SYNGRESS

Industrial Network Security

Securing Critical Infrastructure Networks for Smart Grid, SCADA, and Other Industrial Control Systems

Eric D. Knapp

Industrial Network Security

Securing Critical Infrastructure Networks for Smart Grid, SCADA, and Other Industrial Control Systems

Industrial Network Security

Securing Critical Infrastructure Networks for Smart Grid, SCADA, and Other Industrial Control Systems

Eric Knapp

Technical Editor
James Broad

ELSEVIER

AMSTERDAM • BOSTON • HEIDELBERG • LONDON
NEW YORK • OXFORD • PARIS • SAN DIEGO
SAN FRANCISCO • SINGAPORE • SYDNEY • TOKYO
Syngress is an imprint of Elsevier

SYNGRESS

Acquiring Editor: Angelina Ward
Development Editor: Matt Cater
Project Manager: Jessica Vaughan
Designer: Joanne Blank

Syngress is an imprint of Elsevier
225 Wyman Street, Waltham, MA 02451, USA

Notices
Knowledge and best practice in this field are constantly changing. As new research and experience broaden our understanding, changes in research methods or professional practices may become necessary. Practitioners and researchers must always rely on their own experience and knowledge in evaluating and using any information or methods described herein. In using such information or methods they should be mindful of their own safety and the safety of others, including parties for whom they have a professional responsibility. To the fullest extent of the law, neither the Publisher nor the authors, contributors, or editors, assume any liability for any injury and/or damage to persons or property as a matter of products liability, negligence or otherwise, or from any use or operation of any methods, products, instructions, or ideas contained in the material herein.

Library of Congress Cataloging-in-Publication Data
Knapp, Eric.
Industrial network security : securing critical infrastructure networks for Smart Grid, SCADA, and other industrial control systems / Eric Knapp.
p. cm.
Summary: "This book attempts to define an approach to industrial network security that considers the unique network, protocol and application characteristics of an industrial control system, while also taking into consideration a variety of common compliance controls"–Provided by publisher.
Includes bibliographical references.
ISBN 978-1-59749-645-2 (pbk.)
1. Process control–Security measures. 2. Computer security. I. Title.
TS156.8.K58 2011
670.42'7–dc23
2011018442

British Library Cataloguing-in-Publication Data
A catalogue record for this book is available from the British Library

ISBN: 978-1-59749-645-2

Printed in the United States of America

11 12 13 14 15 10 9 8 7 6 5 4 3 2 1

For information on all Syngress publications visit our website at www.syngress.com.

Contents

This page intentionally left blank

About the Author

Eric D. Knapp is the Director of Critical Infrastructure Markets for NitroSecurity, where he leads the identification, evaluation, and implementation of new security technologies specific to the protection of critical infrastructure, Supervisory Control And Data Acquisition (SCADA), and industrial control networks.

Eric has 20 years of experience in Information Technology, specializing in industrial automation technologies, infrastructure security, and applied Ethernet protocols as well as the design and implementation of Intrusion Prevention Systems and Security Information and Event Management systems in both enterprise and industrial networks. In addition to his work in information security, Eric is an award-winning author. He studied English and Writing at the University of New Hampshire and the University of London and holds a degree in communications.

This page intentionally left blank

About the Technical Editor

James Broad (CISSP, C|EH, C)PTS, Security+, MBA) is the President and owner of Cyber-Recon, LLC, where he and his team of consultants specialize in Information Security, Information Assurance, and Certification and Accreditation and offer other security consultancy services to corporate and government clients.

As a security professional with over 20 years of real-world IT experience, James is an expert in many areas of IT security, specializing in security engineering, penetration testing, and vulnerability analysis and research. He has provided security services in the Nation's most critical sectors including defense, law enforcement, intelligence, finance, and healthcare.

James has a Master's of Business Administration degree with specialization in Information Technology (MBA/IT) from the Ken Blanchard College of Business, Bachelor's degrees in Computer Programming and Security Management from Southwestern University and is currently a Doctoral Learner pursuing a PhD in Information Security from Capella University. He is a member of ISSA and (ISC)$^{2®}$. James currently resides in Stafford, Virginia with his family: Deanne, Micheal, and Temara.

This page intentionally left blank

Foreword

One of the most mysterious areas of information security is industrial system security. No other area of information security contains that many myths, mistakes, misconceptions and outright lies. Information available online, while voluminous, will only lead information security professionals and industrial systems professionals to more confusion and more misconceptions—which may result in not only costly, but also life-threatening, mistakes.

What raises the mystery even higher is that the stakes in the area of industrial security are extremely high. While the loss of trade secret information may kill a business, the loss of electricity generating capability may kill not just one person, but potentially thousands.

And finally the mystery is solved—with this well-researched book on industrial system network security.

The book had a few parts of particular interest to me. I liked that the book covers the "myth of an air gap"—now in the age of wireless, the air gap is not what it used to be and should not be assumed to be "the absolute security." I also liked that safety versus security is covered: industrial engineers might know more about the former while my InfoSec colleagues know more about the latter. Today's interconnected industrial systems absolutely need both! Finally, I also liked the book's focus on risk and impact, and not simply on following the regulatory minimum.

Both information security and industrial engineers, which are currently two distinctly different tribes, would benefit from this book. And, hopefully *Industrial Network Security* will bring the much needed union of both tribes, thus helping us build a more secure business and industrial system.

—Dr. Anton A. Chuvakin
Security Warrior Consulting

This page intentionally left blank

CHAPTER

Introduction

1

INFORMATION IN THIS CHAPTER:

- Book Overview and Key Learning Points
- Book Audience
- Diagrams and Figures
- The Smart Grid
- How This Book Is Organized

BOOK OVERVIEW AND KEY LEARNING POINTS

This book attempts to define an approach to industrial network security that considers the unique network, protocol, and application characteristics of an **industrial control system**, while also taking into consideration a variety of common compliance controls.

Although many of the techniques described herein—and much of the general guidance provided by regulatory standards organizations—are built upon common enterprise security methods and reference readily available information security tools, there is little information available about how to implement these methods. This book attempts to rectify this by providing deployment and configuration guidance where possible, and by identifying why security controls should be implemented, where they should implemented, how they should be implemented, and how they should be used.

BOOK AUDIENCE

To adequately discuss industrial network security, the basics of two very different systems need to be understood: the Ethernet and Transmission Control Protocol/Internet Protocol (TCP/IP) networking communications used ubiquitously in the enterprise, and the **SCADA** and field bus protocols used to manage and/or operate industrial automated systems.

As a result, this book possesses a bifurcated audience. For the plant operator with an advanced electrical engineering degree and a decade of logic programming for Modbus controllers, the basics of industrial network protocols in Chapter 4 have been presented within the context of security in an attempt to not only provide value to such a reader, but also to get that reader thinking about the subtle implications of cyber security. For the information security analyst with a Certified Information Systems Security Professional (CISSP) certification, basic information security practices have been provided within the new context of an industrial control system.

There is an interesting dichotomy between the two that provides a further challenge. Enterprise security typically strives to secure the users and **hosts** on a network while at the same time enables the broad range of open communication services required within modern business. Industrial control systems, on the other hand, strive for the efficiency and reliability of a single, often fine-tuned system. Only by giving the necessary consideration to both sides can the true objective be achieved: a secure industrial network that supports reliable operation while also providing business value to the larger enterprise.

To further complicate matters, there is a third audience: the compliance officer who is mandated with meeting certain regulatory standards in order to survive an audit with minimal penalties and/or fines. Compliance continues to drive information security budgets, and therefore the broader scope of industrial networks must also be narrowed on occasion to the energy industries, where (at least in the United States) electrical energy, nuclear energy, oil, and gas are tightly regulated. Compliance controls are discussed in this book solely within the context of implementing cyber security controls. The recommendations given are intended to improve security and should not be interpreted as advice concerning successful compliance management.

DIAGRAMS AND FIGURES

The network diagrams used throughout this book have been intentionally simplified and have been designed to be as generic as possible while adequately representing industrial networks across a very wide range of industrial systems. As a result, the diagrams will undoubtedly differ from real industrial network designs and may exclude details specific to one particular industry while including details that are specific to another. However, they will provide a high-level understanding of the specific industrial network security controls being discussed.

THE SMART GRID

Although the smart grid is of major concern and interest, for the most part it is treated as any other industrial network within this book, with specific considerations being made only when necessary (such as when considering available **attack vectors**). As a result, there are many security considerations specific to the smart grid that are unfortunately not included. This is partly to maintain focus on the more ubiquitous

ICS and SCADA security requirement, partly due to the relative immaturity of smart grid security and partly due to the specialized and complex nature of these systems. Although this means that specific measures for securing synchrophasers, meters, etc. are not provided, the guidance and overall approach to security that is provided herein is certainly applicable to smart grid networks. For more in-depth reading on smart grid network security, consider *Securing the Smart Grid: Next Generation Power Grid Security* by Tony Flick and Justin Morehouse (ISBN: 978-1-59749-570-7, Syngress).

HOW THIS BOOK IS ORGANIZED

This book is divided into a total of eleven chapters, followed by three appendices guiding the reader where to find additional information and resources about industrial protocols, standards and regulations, and relevant **NIST** security guidelines. An extensive glossary is also provided to accommodate the wealth of both information security and industrial networking terms and acronyms used throughout the book.

The chapters begin with an introduction to industrial networking, and what a cyber attack against an industrial control systems might represent in terms of potential risks and consequences, followed by details of how industrial networks can be assessed, secured, and monitored in order to obtain the strongest possible security, and conclude with a detailed discussion of various compliance controls, and how those specific controls map back to network security practices.

It is not necessary to read this book cover to cover, in order. The book is intended to offer insight and recommendations that relate to both specific security goals as well as the cyclical nature of the security process. That is, if faced with performing a **vulnerability assessment** on an industrial control network, begin with Chapter 6; every effort has been made to refer the reader to other relevant chapters where additional knowledge may be necessary.

Chapter 2: About Industrial Networks

In this chapter, there is a brief introduction to industrial networks as they relate to "**critical infrastructure**," those infrastructures upon which our society, industry, and way of life depend. The dependencies of critical infrastructures upon industrial control systems lead naturally to a discussion of the many standards, regulations, guidance documents, and policies that have been implemented globally to protect these systems. In addition, the chapter introduces the reader to the most basic premises of industrial security.

Of particular note, Chapter 2 also discusses the use of terminology within the book as it relates to the many applications of industrial networks (again, there is also an extensive Glossary included to cover the abundance of new acronyms and terms used in industrial control networks).

Chapter 3: Introduction to Industrial Network Security

Chapter 3 introduces industrial networks in terms of cyber security, by examining the interrelations between "general" networking, industrial networking, and potentially critical infrastructures. Chapter 3 covers the importance of securing industrial networks, discusses the impact of a successful industrial attack, and provides examples of real incidents—including a discussion of the **Advanced Persistent Threat** and the implications of cyber war.

Chapter 4: Industrial Network Protocols

This chapter focuses on industrial network protocols, including **Modbus**, DNP3, OPC, **ICCP**, and others in both their native/original fieldbus form or in modernized TCP/IP or real-time Ethernet implementations. The basics of protocol operation, frame format, and security considerations are provided for each, with security recommendations being made where applicable.

Chapter 5: How Industrial Networks Operate

Industrial networks use specialized protocols because they perform functions that are different than enterprise networks, with different requirements and different security considerations. Chapter 5 discusses control system **assets**, network architectures, control system operations, and how control processes are managed, with special emphasis on smart grid operations.

Chapter 6: Vulnerability and Risk Assessment

Strong security requires a proper assessment of vulnerabilities and risk, which in turn requires that security analysts think like an attacker. Chapter 6 provides a high-level overview of common attack methodologies, and how industrial networks present a unique **attack surface** with common attack vectors to many critical areas. Chapter 6 also discusses vulnerability assessment and patch management strategies.

Chapter 7: Establishing Secure Enclaves

A strong "defense in depth" strategy requires the isolation of functional groups into securable "**enclaves**." Chapter 7 looks at how to separate functional groups and where enclave boundaries should be implemented. Specifics are then provided on how to secure both the perimeter and the interior of enclaves, including common security products, methods, and policies that may be implemented.

Chapter 8: Exception, Anomaly, and Threat Detection

Awareness is the perquisite of action, according to the common definition of **situational awareness**. In this chapter, several contributing factors to obtaining situational awareness are discussed, including how to use anomaly detection, exception reporting, and information correlation for the purposes of threat and risk detection.

Chapter 9: Monitoring Enclaves

Before situational awareness can be achieved, however, a necessary body of information must be obtained. This chapter includes recommendations of what to monitor, why, and how. Information management strategies—including **log** and **event** collection, direct monitoring, and **security information and event management** (**SIEM**)—are discussed, including guidance on data collection, retention, and management.

Chapter 10: Standards and Regulations

There are many regulatory compliance standards applicable to industrial network security, and most consist of a wide range of procedural controls that aren't easily resolved using information technology. There are common cyber security controls (with often subtle but importance variations), however, which reinforce the recommendations put forth in this book. Chapter 10 attempts to map those cyber security–related controls from some common standards—including **NERC CIP**, **CFATS**, **ISO/IEC** 27002:2005, NRC RG 5.71, and NIST 800-82—to the security recommendations made within this book, making it easier for security analysts to understand the motivations of compliance officers, while compliance officers are able to see the security concerns behind individual controls.

Chapter 11: Common Pitfalls and Mistakes

Industrial control systems are highly vulnerable, and often with high consequence. In this chapter, some common pitfalls and mistakes are highlighted—including errors of complacency, common misconfigurations, and deployment errors—as by highlighting the pitfalls and mistakes, it is easier to avoid repeating those mistakes.

CONCLUSION

Writing this book has been an education, an experience, and a challenge. In the months of research and writing, several historic moments have occurred concerning Industrial Control Systems security, including the first ICS-targeted cyber weapon, and one of the most sophisticated cyber attacks to date. The growing number of attacks, new evidence of Advanced Persistent Threats, and a wave of new SCADA- and ICS-specific vulnerabilities are just the tip of the proverbial iceberg.

Hopefully, this book will be both informative and enjoyable, and it will facilitate the increasingly urgent need to strengthen the security of our industrial networks and SCADA systems. Even though the attacks themselves will continue to evolve, the methods provided herein should help to prepare against the inevitable advancement of industrial network threat.

This page intentionally left blank

CHAPTER

About Industrial Networks 2

INFORMATION IN THIS CHAPTER:

- Industrial Networks and Critical Infrastructure
- Relevant Standards and Organizations
- Common Industrial Security Recommendations
- The Use of Terminology Within This Book

Before attempting to secure an industrial network, it is important to understand what an industrial network really is. Because of the diversity of both the industrial networks themselves as well as the markets that they serve, it can be confusing to discuss them in general terms. In addition, the many regulatory agencies and commissions that have been formed to help secure different industrial networks for different markets each introduce their own specific nomenclatures and terminology. Finally, the common misuse of terminology within the media further confuses the issue of what an industrial network truly is.

INDUSTRIAL NETWORKS AND CRITICAL INFRASTRUCTURE

The world of industrial control systems, like many high-tech sectors, possesses its own lexicon to describe the nuances of its industry. Unfortunately, the terms used are also often interchanged and misunderstood. Industrial Control Systems are often referred to in the media as "SCADA," for example, which is both inaccurate and misleading. An industrial network is most typically made up of several distinct areas, which are simplified here as a business network or enterprise, business operations, a supervisory network, and process and control networks (see Figure 2.1). SCADA, or **Supervisory Control and Data Acquisition**, is just one specific piece of an industrial network, separate from the control systems themselves, which should be referred to as Industrial Control Systems (ICS), **Distributed Control Systems** (**DCS**), or **Process Control Systems** (**PCS**). Each area has its own physical and logical security considerations, and each has its own policies and concerns.

The book title "Industrial Network Security: Securing Critical Infrastructure Networks for Smart Grid, SCADA, and Other Industrial Control Systems" was chosen because this text discusses the security concerns of all the networks that make

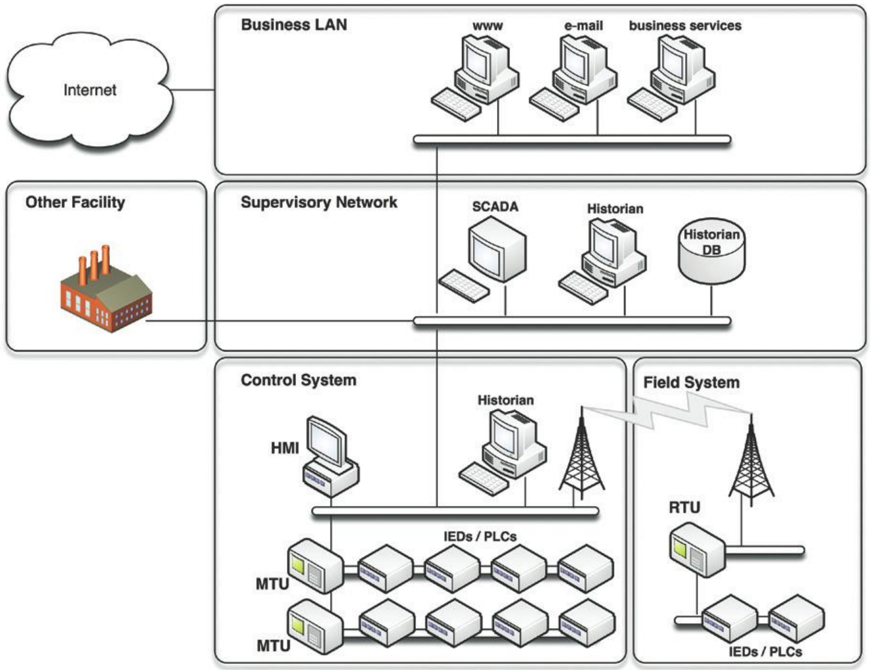


FIGURE 2.1

Sample Industrial Automated Control System Network.

up an industrial network, including the supervisory and distributed control systems, primarily as they apply to critical infrastructure. The business Local Area Network (LAN), the process control network, and whatever supervisory demilitarized zone (DMZ) exists between them are all equally important. To be more specific, it discusses the cyber security of these networks. For the sake of clarity, it is assumed that a strong security policy, security awareness, personnel, and physical security practices are already in place, and these topics will not be addressed except for where they might be used to strengthen specific areas of network security.

Critical Infrastructure

For the purposes of this book, the terms "Industrial Network" and "Critical Infrastructure" are used in somewhat limited contexts. "Industrial Network" is referring to any network operating some sort of automated control system that communicates digitally over a network, and "Critical Infrastructure" is referring to critical *network* infrastructure, including any network used in the direct operation of any system upon which one of the defined "critical infrastructures" depends. Confusing? It is, and this is perhaps one of the leading reasons that our critical infrastructures remain at risk today: many an ICS security seminar has digressed into an argument over semantics, at the sake of any real discussion on network security practices.

Luckily, the two terms are closely related in that the defined critical infrastructure, meaning those systems listed in the **Homeland Security Presidential Directive Seven** (**HSPD-7**), typically utilizes some sort of industrial control systems. In its own words, "HSPD-7 establishes a national policy for Federal departments and agencies to identify and prioritize [the] United States critical infrastructure and key resources and to protect them from terrorist attacks." HSPD-7 includes public safety, bulk electric energy, nuclear energy, chemical manufacturing, agricultural and pharmaceutical manufacturing and distribution, and even aspects of banking and finance: basically, anything whose disruption could impact a nation.[1] However, while some, such as global banking and finance, are considered a part of our critical infrastructure, they do not typically operate industrial control networks, and so are not addressed within this book (although many of the security recommendations will still apply, at least at a high level).

Utilities

Utilities—water, gas, oil, electricity, and communications—are critical infrastructures that rely heavily on industrial networks and automated control systems. Because the disruption of any of these systems could impact our society and our safety, they are listed as critical by HSPD-7; because they use automated and distributed process control systems, they are clear examples of industrial networks. Of the common utilities, electricity is often separated as requiring more extensive security. In the United States and Canada, it is specifically regulated to standards of reliability and cyber security. Oil and gas refining and distribution are systems that should be treated as both a chemical/hazardous material and as a critical component of our infrastructures. It is often regulated as a chemical facility because of these particular qualities.

Nuclear Facilities

Nuclear facilities represent unique safety and security challenges due to their inherent danger in the fueling and operation, as well as the national security implications of the raw materials used. This makes nuclear facilities a prime target for cyber attack, and it makes the consequences of a successful attack more severe. As such, nuclear energy is heavily regulated in the United States by the **Nuclear Regulatory Commission** (**NRC**). The NRC was formed as an independent agency by Congress in 1974 in an attempt to guarantee the safe operation of nuclear facilities and to protect people and the environment. This includes regulating the use of nuclear material including by-product, source, and special nuclear materials, as well as nuclear power.[2]

Bulk Electric

The ability to generate and distribute electricity in bulk is highly regulated. Electrical energy generation and distribution is defined as a critical infrastructure under HSPD-7, and is heavily regulated in North America by **NERC**—specifically via the NERC Critical Infrastructure Protection (CIP) reliability standards—under the authority of the Department of Energy, which is ultimately responsible for the security of the production, manufacture, refining, distribution, and storage of oil, gas, and non-nuclear power.[3]

It's important to note that energy generation and distribution are two distinct industrial network environments, each with its own nuances and special security requirements. Energy generation is primarily concerned with the safe manufacture of a product (electricity), while energy distribution is concerned with the safe and balanced distribution of that product. The two are also highly interconnected, obviously, as generation facilities directly feed the power grid that distributes that energy; bulk energy must be carefully measured and distributed upon production. For this same reason, the trading and transfer of power between power companies is an important facet of an electric utility's operation.

The smart grid—an update to traditional electrical transmission and distribution systems to accommodate digital communications for metering and intelligent delivery of electricity—is a unique facet of industrial networks that is specific to the energy industry that raises many new security questions and concerns.

Although energy generation and distribution are not the only industrial systems that need to be defended, they are often used as examples within this book. This is because the **North American Electric Reliability Corporation** (NERC) has created a reliability standard called "Critical Infrastructure Protection" and enforces it heavily throughout the United States and Canada. Likewise, the NRC requires and enforces the cyber security of nuclear power facilities. Ultimately, all other industries rely upon energy to operate, and so the security of the energy infrastructure (and the development of the smart grid) impacts everything else, so that talking about securing industrial networks without talking about energy is practically impossible.

Is bulk power more important than other industrial systems? That is a topic of heavy debate. Within the context of this book, we assume that all control systems are important, whether or not they generate or distribute energy, or whether they are defined that way by HSPD-7 or any other directive. A speaker at the 2010 Black Hat conference suggested that ICS security is overhyped, because these systems are more likely to impact the production of cookies than they are to impact our national infrastructure.[4] However, even the production of a snack food can impact many lives: through the manipulation of its ingredients or through financial impact to the producer and its workers, for example.

Chemical Facilities

Chemical manufacture and distribution represent specific challenges to securing an industrial manufacturing network. Unlike the "utility" networks (electric, nuclear, water, gas), chemical facilities need to secure their intellectual property as much as they do their control systems and manufacturing operations. This is because the product itself has a tangible value, both financially and as a weapon. For example, the formula for a new pharmaceutical could be worth a large sum of money on the black market. The disruption of the production of that pharmaceutical could be used as a social attack against a country or nation, by impacting the ability to produce a specific vaccine or antibody. Likewise, the theft of hazardous chemicals can be used directly as weapons or to fuel illegal chemical weapons research or manufacture. For this reason, chemical facilities need to also focus on securing the storage and transportation of the end product.

Critical versus Noncritical Industrial Networks

The security practices recommended within this book aim for a very high standard, and in fact go above and beyond what is recommended by many government and regulatory groups. So which practices are really necessary, and which are excessive? It depends upon the nature of the industrial system being protected. What are the consequences of a cyber attack? The production of energy is much more important in modern society than the production of a Frisbee. The proper manufacture and distribution of electricity can directly impact our safety by providing heat in winter or by powering our irrigation pumps during a drought. The proper manufacture and distribution of chemicals can mean the difference between the availability of flu vaccines and pharmaceuticals and a direct health risk to the population. Regardless of an ICS's classification, however, most industrial control systems are by their nature important, and any risk to their reliability holds industrial-scale consequences. However, while not all manufacturing systems hold life-and-death consequences, that doesn't mean that they aren't potential targets for a cyber attack. What are the chances that an extremely sophisticated, targeted attack will actually occur? The likelihood of an incident diminishes as the sophistication of the attack—and its consequences—grow, as shown in Figure 2.2. By implementing security practices to address these uncommon and unlikely attacks, there is a greater possibility of avoiding the devastating consequences that correspond to them.

Although the goal of this book is to secure any industrial network, it focuses on Critical Infrastructure and electric energy in particular, and will reference various standards, recommendations, and directives as appropriate. Regardless of the nature of the control system that needs to be secured, it is important to understand these directives, especially NERC CIP, Chemical Facility Anti-Terrorism Standards (CFATS), Federal Information Security Management Act (FISMA), and the control system security recommendations of National Institute of Standards and Technology (NIST). Each has its own strengths and weaknesses, but all provide a good baseline of best practices for industrial network security (each is explored in more detail in Chapter 10, "Standards and Regulations"). Not surprisingly, the industrial networks that control critical infrastructures demand the strongest controls and regulations around security and reliability, and as such there are numerous organizations helping to achieve just that. The Critical Infrastructure Protection Act of 2001 and HSPD-7 define what they are, while others—such as NERC CIP, CFATS, and various publications of NIST—help explain what to do.

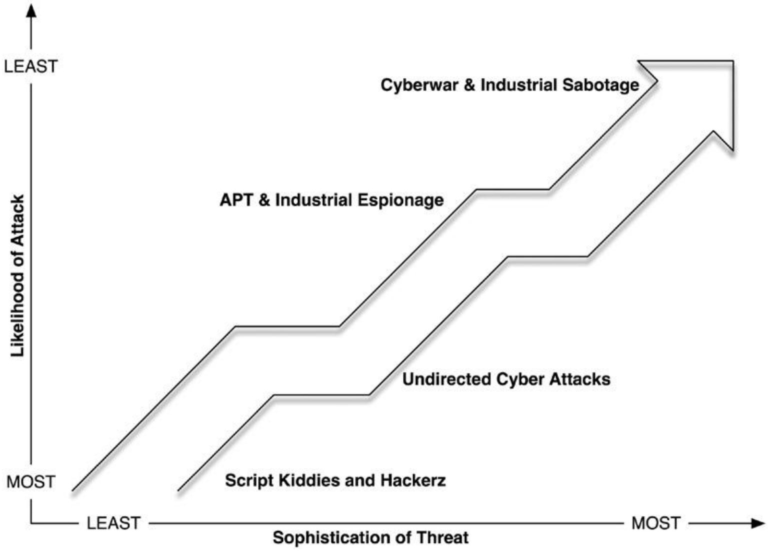


FIGURE 2.2

Likeliness versus Consequence of a Targeted Cyber Attack.

RELEVANT STANDARDS AND ORGANIZATIONS

Many organizations are attempting to define methods of securing our industrial systems. Some are regional, some are national, and some are global. Some are public, some are private. Some—like NERC CIP—carry heavy fines for non-compliance if one falls under their jurisdiction. Others—such as CFATS—offer recommendations for self-assessment and lack the ability to levy penalties for noncompliance.

Each standard is discussed briefly here and in more detail in Chapter 10, “Standards and Regulations.” Although this book does not attempt to provide compliance or audit guidelines, the various standards provide valuable insight into how we should and should not be securing our industrial networks. When considered as a whole, we see common requirement challenges and recommendations that can and should be considered “best practices” for industrial network security.

Homeland Security Presidential DirectiveSeven/HSPD-7

The HSPD-7 attempts to distinguish the critical versus noncritical systems. HSPD-7 does not include specific security recommendations, relying instead upon other federal security recommendations such as those by the NIST on the security of both enterprise and industrial networks, as well as the Homeland Security Risk-Based Performance Standards used in securing chemical facilities.

Which regulations apply to your specific industrial network? Possibly several, and possibly none. Although more information is provided in Chapter 10, “Standards and Regulations," some of the more common regulations are summarized here in order to help you determine which standards you should be striving to meet.

NIST Special Publications (800 Series)

NIST's 800 series documents provide best practices and information of general interest to information security. All 800 series documents concern information security and should be used as references where applicable. Of particular relevance to industrial network security is SP 800-53 ("Recommended Security Controls for Federal Information Systems"), which defines many aspects of information security procedures and technologies, and SP 800-82 ("Guide to Supervisory Control and Data Acquisition [SCADA] and Industrial Control Systems Security"), which discusses industrial control system security specifically. Although of the entire SP 800-53 is applicable to the protection of critical infrastructures, the technical aspects defined under SP 800-53 as Access Control, Security Assessment and Authorization, Configuration Management, Identification and Authentication, Risk Assessment, System and Communications Protection, and System and Information Integrity are directly applicable to industrial networks.[5]

SP 800-82 (currently in draft) details control system architectures, protocols, vulnerabilities, and security controls. Specific security recommendations of SP 800-53 and SP 800-82 are addressed in more detail in Chapter 10, "Standards and Regulations."

NERC CIP

The NERC CIP reliability standard identifies security measures for protecting critical infrastructure with the goal of ensuring the reliability of the bulk power system. Compliance is mandatory for any power generation facility, and fines for noncompliance can be steep. The CIP reliability standards consist of nine sections, each with its own requirements and measures. They are Sabotage Reporting, **Critical Cyber Asset** Identification, Security Management Controls, Personnel & Training, **Electronic Security Perimeter**(s), Physical Security of Critical Cyber Assets, Systems Security Management, Incident Reporting and Response Planning, and Recovery Plans for Critical Cyber Assets.

Nuclear Regulatory Commission

The NRC is responsible for ensuring the safe use of radioactive materials for beneficial civilian (nonmilitary) purposes by licensed nuclear facilities. Part of this responsibility is the establishment of cyber security requirements and recommendations, which are defined primarily within two documents: Title 10 Code of Federal Regulations (CFR), section 73.54 (10 CFR 73.54), and Office of Nuclear Regulatory Research's Regulatory Guide 5.71 (RG 5.71), which explains in detail the specific cyber security requirements of 10 CFR 73.54. RG 5.71 provides recommendations to nuclear agencies or "licensees" in how to secure their facilities against cyber attack. These recommendations indicate that a licensee "shall protect digital computer and communication systems and networks associated with safety, security, emergency preparedness, and any systems that support safety, security and emergency preparedness"[6] and that they shall protect the systems and networks that impact the integrity or confidentiality of data and/or software; deny access to systems, services, and/or data; and prevent any activity that might adversely impact the operation of systems, networks, and associated equipment.[7]

To accomplish this, RG 5.71 makes recommendations in how to identify **critical digital assets**, as well as how to implement a defense in depth strategy to mitigate the adverse effects of a cyber attack against those critical assets, all to "ultimately ensure that the functions of protected assets are not adversely impacted due to cyber attacks."[8]

Important components of RG 5.71 include[9]

- Analyzing Digital Computer Systems and Networks (C.3.1)
- Identification of Critical Digital Assets (C.3.1.3)
- Defense-in-Depth Protective Strategies (C.3.2)
- Security Defensive Architecture (C.3.2.1)
- Establishing Security Controls (C.3.3)
- Technical Controls (C.3.3.10), including
  - Access Control (C.3.3.1.1)
  - Audit and Accountability (C.3.3.1.2)
  - System and Communications Protection (C.3.3.1.3)
  - Identification and Authentication (C.3.3.1.4)
  - System Hardening (C.3.3.1.5)
- Operational Controls (C3.3.2), including
  - Media Protection (C.3.3.2.1)
  - System and Information Integrity (C.3.3.2.3)
  - Incident Response (C.3.3.2.6)
- Continuous Monitoring and Assessment (C.4.1)
- Vulnerability Scans and Assessments (C.4.1.3)
- Change Control (C.4.2)
- Configuration Management (C.3.3.2.9 and C.4.2.1)

In addition, Appendix B of RG 5.71 is exceptionally useful, as it provides in depth detail on recommended security technical controls, of which the following apply directly to network security:[10]

- Access Controls (B.1), including
  - Access Control Policy and Procedures (B.1.1)
  - Account Management (B.1.2)
  - Access Enforcement (B.1.3)
  - Information Flow Enforcement (B.1.4)
  - Separation of Functions (B.1.5)

  - Network Access Control (B.1.15)
  - "Open/Insecure" Protocol Restrictions (B.1.16)
  - Wireless Access Restrictions (B.1.17)
  - Insecure and Rogue Connections (B.1.18)
  - Proprietary Protocol Visibility (B.1.20)
- Audit and Accountability (B.2)
- Critical Digital Asset and Communications Protection (B.3), including
  - Application Partitioning and Security Function Isolation (B.3.2)
  - Transmission Integrity (B.3.6)
  - Use of Cryptography (B.3.10)
  - Session Authenticity (B.3.18)
  - Confidentiality of Information at Rest (B.3.20)
- Identification and Authentication (B.4)
- Removal of Unnecessary Services and Programs (B.5.1)
- Host Intrusion Detection System (B.5.2)

For the most part, the NRC's guidelines are consistent with NIST recommendations. The NRC classifies the criticality of an asset or system based on the risks to operations and safety that could result from its compromise. A severity level (SL) is assigned to a **cyber asset** or mechanism, and the recommendations for cyber security vary based on the assigned SL. There are five SLs, Severity Level 0 to Severity Level 4. One unique recommendation made by the NRC for the protection of nuclear facilities is the use of unidirectional access to the most critical systems, indicated by a severity level of 4—which may be accomplished using a **data diode** or a physical air gap—represents one of the most stringent cyber security practices recommended by any of the regulatory agencies mentioned within this book. The specific recommendation to validate sessions and monitor access to proprietary protocols is also more stringent than the requirements of other regulations—both of which are important considerations when attempting to secure industrial networks, which often use proprietary protocols and/or specialized standard protocols that may or may not include session authentication or validation. Unfortunately, RG 5.71 is purely a recommendation for complying with the broader requirements provided in 10 CFR 73.54 and is not an enforceable standard at this time.

Federal Information Security Management Act

The FISMA may or may not apply to certain critical infrastructures, depending upon their geographic location and/or their jurisdiction within the United States federal government. However, the standards include valid and useful guidelines for the security of critical environments, referring to and relying upon the NIST "800 series" Special Publication documents (especially SP 800-53 and SP 800-82). The management controls of SP 800-53 are divided into 18 security categories:[11]

- Access Control (AC)
- Awareness and Training (AT)

- Audit and Accountability (AU)
- Security Assessment and Authorization (CA)
- Configuration Management (CM)
- Contingency Planning (CP)
- Identification and Authentication (IA)
- Incident Response (IR)
- Maintenance (MA)
- Media Protection (MP)
- Physical and Environmental Protection (PE)
- Planning (PL)
- Personnel Security (PS)
- Risk Assessment (RA)
- System and Services Acquisition (SA)
- System and Communication Protection (SC)
- System and Information Integrity (SI)
- Program Management (PM)

While all of these controls relate to cyber security, the areas that relate most directly to network security practices are Access Control (AC), Audit and Accountability (AU), Configuration Management (CM), Identification and Authentication (IA), Media Protection (MP), Risk Assessment (RA), System and Communication Protection (SC), and System and Information Integrity (SI).[12]

Chemical Facility Anti-Terrorism Standards

CFATS is a set of risk-based performance guidelines published by the Department of Homeland Security. CFATS also consists of 18 Risk-based Performance Standards (**RBPS**s), although these groups differ substantially from those defined by NIST:[13]

- RBPS 1—Restrict Area Perimeter
- RBPS 2—Secure Site Assets
- RBPS 3—Screen and Control Access
- RBPS 4—Deter, Detect, and Delay
- RBPS 5—Shipping, Receipt, and Storage
- RBPS 6—Theft or Diversion
- RBPS 7—Sabotage
- RBPS 8—Cyber
- RBPS 9—Response
- RBPS 10—Monitoring
- RBPS 11—Training
- RBPS 12—Personnel Surety
- RBPS 13—Elevated Threats
- RBPS 14—Specific Threats, Vulnerabilities, or Risks
- RBPS 15—Reporting of Significant Security Incidents
- RBPS 16—Significant Security Incidents and Suspicious Activities

- RBPS 17—Officials and Organization
- RBPS 18—Records

Of these, RBPS 6 (Theft or Diversion), 7 (Sabotage), 8 (Cyber), 14 (Specific Threats, Vulnerabilities, or Risks), and 15 (Reporting of Significant Security Incidents) concern cyber security, with RBPS 8 focusing solely on cyber security. The CFATS RBPSs are not enforceable requirements at this time and are intended as guidance for chemical facilities.[14]

ISA-99

ISA standard 99 (ISA-99) is an industrial control security standard created by the International Society of Automation (ISA) to protect SCADA and process control systems. ISA-99 offers varying security recommendations based on the physical and logical location of the systems being protected as well as their importance to the reliable operation of the system. In order to accomplish this, ISA-99 first attempts to classify functional areas of an industrial system into specific security levels and then provides recommendations for separating these areas into "**zones**." ISA-99 also defines the interconnectedness of zones as well as how to enforce security between zones.

Using the example of an industrial network as illustrated in Figure 2.1, the most public systems such as Internet or Internet-facing systems within the business LAN would continue level 5, while the rest of the business LAN may map to level 4. Supervisory networks (i.e., the SCADA DMZ network) would represent level 3, and so on, with the actual "control system" (the SCADA networks, **HMI** systems, field devices, instrumentation and sensors) at level 0. This concept is very illustrative of the functional isolation of services and the establishment of security "enclaves" (see "Defense in Depth").

ISA-99 organizes security recommendations into seven foundational requirements:[15]

- FR1—Access Control (AC)
- FR2—Use Control (UC)
- FR3—Data Integrity (DI)
- FR4—Data Confidentiality (DC)
- FR5—Restrict Data Flow (RDF)
- FR6—Timely Response to an Event (TRE)
- FR7—Resource Availability (RA)

Each foundational requirement consists of multiple system requirements (SRs). SRs that are especially useful to the protection of industrial networks (excluding policy and procedural recommendations) include[16]

- SR 1.1—IACS user identification and authentication
- SR 1.2—Account management
- SR 3.1—Communication integrity
- SR 3.2—Malicious code protection

- SR 3.3—Security functionality verification
- SR 3.4—Software and information integrity
- SR 4.3—Cryptographic key establishment and management
- SR 5.1—Information flow enforcement
- SR 5.2—Application partitioning
- SR 5.4—Boundary protection
- SR 7.1—Denial of service protection
- SR 7.2—Management of network resources
- SR 7.6—Network and security configuration settings

ISO 27002

ISO 27002 is a set of security recommendations published by the **International Standards Organization** (ISO) and the **International Electrotechnical Commission** (IEC), and may be referred to as ISO/IEC 27002 or ISO/IEC 27002:2005. ISO 27002 defines "Information technology—Security techniques—Code of practice for information security management," and is not specific to industrial network security. ISO standards are widely used internationally and can be easily mapped to the recommendations of NIST, NRC, NERC, and others, as they consist of functional guidelines for risk assessment; security policy and management; governance; asset management; personnel security; physical and environmental security; communications and operations management; access control; asset acquisition, development, and maintenance; incident management; business continuity management; and compliance.[17]

COMMON INDUSTRIAL SECURITY RECOMMENDATIONS

Many of the network security practices that are either required or recommended by the aforementioned organizations are consistent between many or all of the others. Although all recommendations should be considered, these common "best practices" are extremely important and are the basis for many of the methods and techniques discussed within this book. They consist of the following steps: (1) identifying what systems need to be protected, (2) separating the systems logically into functional groups, (3) implementing a defense-in-depth strategy around each system, and (4) controlling access into and between each group.

Identification of Critical Systems

The first step in securing any system is determining what needs to be protected, and this is reflected heavily in NERC CIP, NRC 10 CFR 73.54, and ISA-99. Identifying the assets that need to be secured, as well as identifying their individual importance to the reliable operation of the overall process control system, is necessary for a few primary reasons: it tells us what should be monitored, and how closely; it tells us how to logically segment the network into high-level security enclaves; and it, therefore, indicates where our point security devices (such as firewalls and intrusion detection and prevention systems) should be placed. For North American electric companies, it also satisfies a direct requirement of NERC CIP, and therefore can help to minimize fines associated with noncompliance.

Identifying critical systems isn't always easy, however. The first step is to build a complete inventory of all connected devices. Each of these devices should be evaluated independently. If it performs a critical function, it should be classified as critical. If it does not, consider whether it could impact any other critical devices or operations. Could it impact the network itself, preventing another device from interacting with a critical system and therefore causing a failure? Finally, does it protect a critical system in any way?

The NRC provides a logic map illustrating how to determine critical assets, which is adapted to more generic asset identification in Figure 2.3. This process will help to separate devices into two categories:

- Critical Assets
- Noncritical Assets

However, in many larger operations this process may be over simplified. There may be different levels of "criticality." A general rule to follow once the basic separation of critical versus noncritical has been completed is as follows: are there any critical assets that are not functionally related to other critical assets? If there are, next ask if one function is more or less important than the other. Finally, if there is both a functional separation *and* a difference in the criticality of the system, consider adding a new logical "tier" to your network. Also remember that a device could potentially be critical *and* also directly impact one or more other critical assets. Consider ranking the criticality of devices based on their total impact as well. Each layer of a separation can then be used as a point of demarcation, providing additional layers of defense between each group.

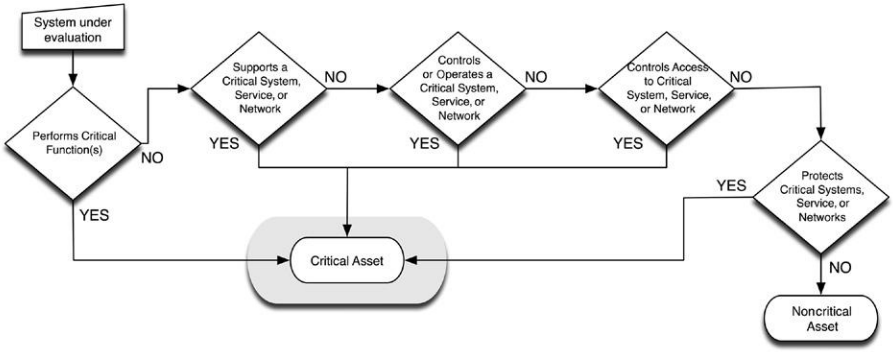

System under evaluation
Performs Critical Function(s)
NO
YES
Supports a Critical System, Service, or Network
NO
YES
Controls or Operates a Critical System, Service, or Network
NO
YES
Controls Access to Critical System, Service, or Network
NO
YES
Protects Critical Systems, Service, or Networks
YES
NO
Critical Asset
Noncritical Asset

FIGURE 2.3

NRC Process Diagram for Identifying Critical Cyber Assets.[18]

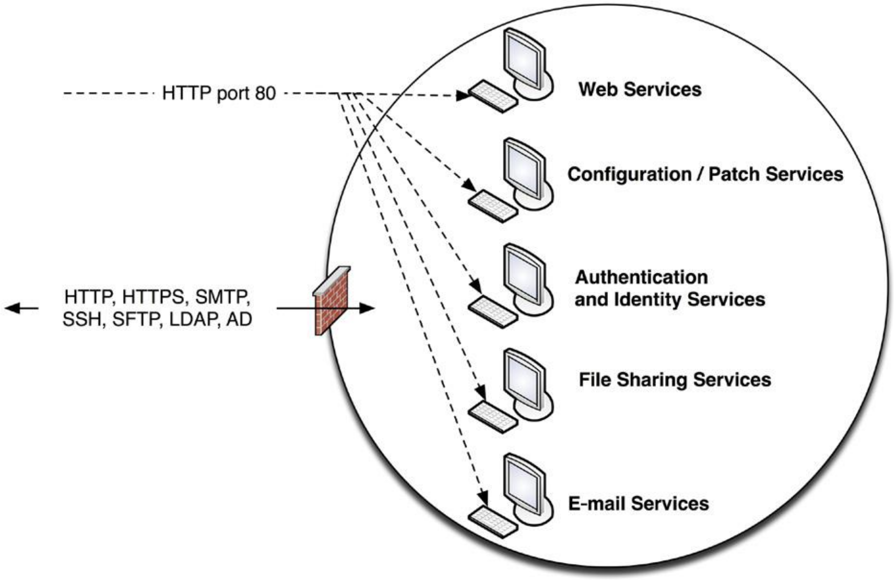

HTTP port 80
HTTP, HTTPS, SMTP, SSH, SFTP, LDAP, AD
Web Services
Configuration / Patch Services
Authentication and Identity Services
File Sharing Services
E-mail Services

FIGURE 2.4

Placing All Services Behind a Common Defense Provides a Broader Attack Surface on All Systems.

Network Segmentation/Isolation of Systems

The separation of assets into functional groups allows specific services to be tightly locked down and controlled, and is one of the easiest methods of reducing the attack surface that is exposed to attackers. Simply by disallowing all unnecessary ports and services, we also eliminate all of the vulnerabilities—known or unknown—that could potentially allow an attacker to exploit those services.

For example, if five critical services are isolated within a single functional group and separated from the rest of the network using a single firewall, it may be necessary to allow several different traffic profiles through that firewall (see Figure 2.4). If an attack is made using an exploit against web services over port 80, that attack may compromise a variety of services including e-mail services, file transfers, and patch/update services.

However, if each specific service is grouped functionally and separated from all other services, as shown in Figure 2.5—that is, all web servers are grouped together in one group, all e-mail services in another group, etc.—the firewall can be configured to disallow anything other than the desired service, preventing an e-mail server from being exposed to a threat that exploits a weakness in HTTP.

In an industrial control system environment, this method of service segmentation can be heavily utilized because there are many distinct functional groups

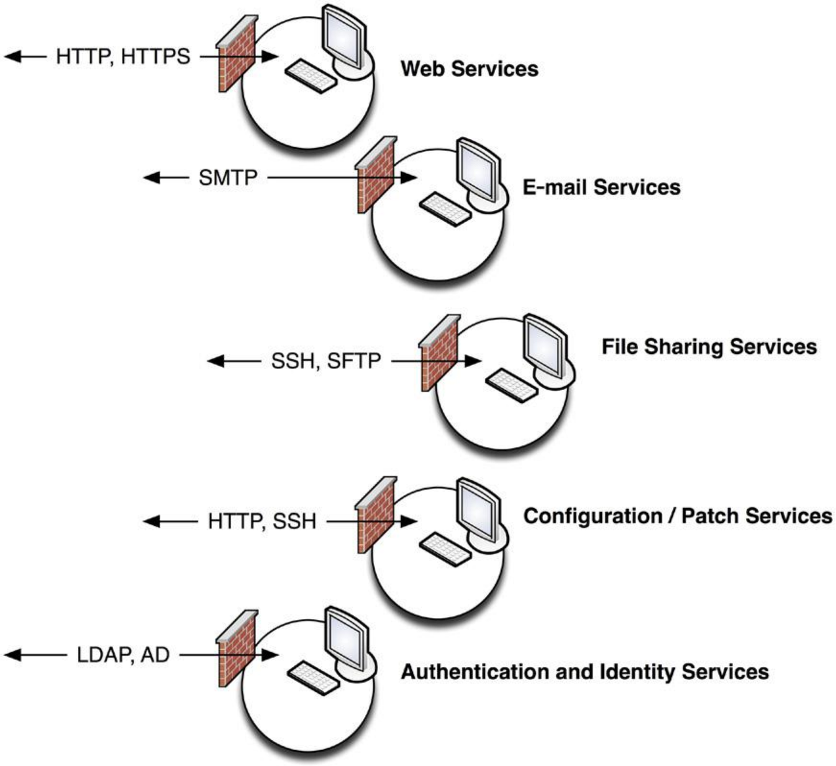


FIGURE 2.5

Separation into Functional Groups Reduces the Attack Surface to a Given System.

within an industrial network that should not be communicating at all outside of established parameters. For example, protocols such as Modbus or DNP3 (discussed in depth in Chapter 4, "Industrial Network Protocols") are specific to SCADA and ICS systems and should never be used within the business LAN and Internet services such as HTTP, SMTP, FTP, and others should never be used within supervisory or control network areas. In Figure 2.6, it can be seen how this layered approach to functional and topological isolation can greatly improve the defensive posture of the network.

Note that within this book, these isolated functional groups or enclaves are often depicted as being separated by a firewall. Although in many cases a separate firewall may be needed for each enclave, the actual method of securing the enclave can vary and could include dedicated firewalls, intrusion detection and prevention devices, application content filters, access control lists, and/or a variety of other controls. In some cases, multiple enclaves can be supported using a single firewall

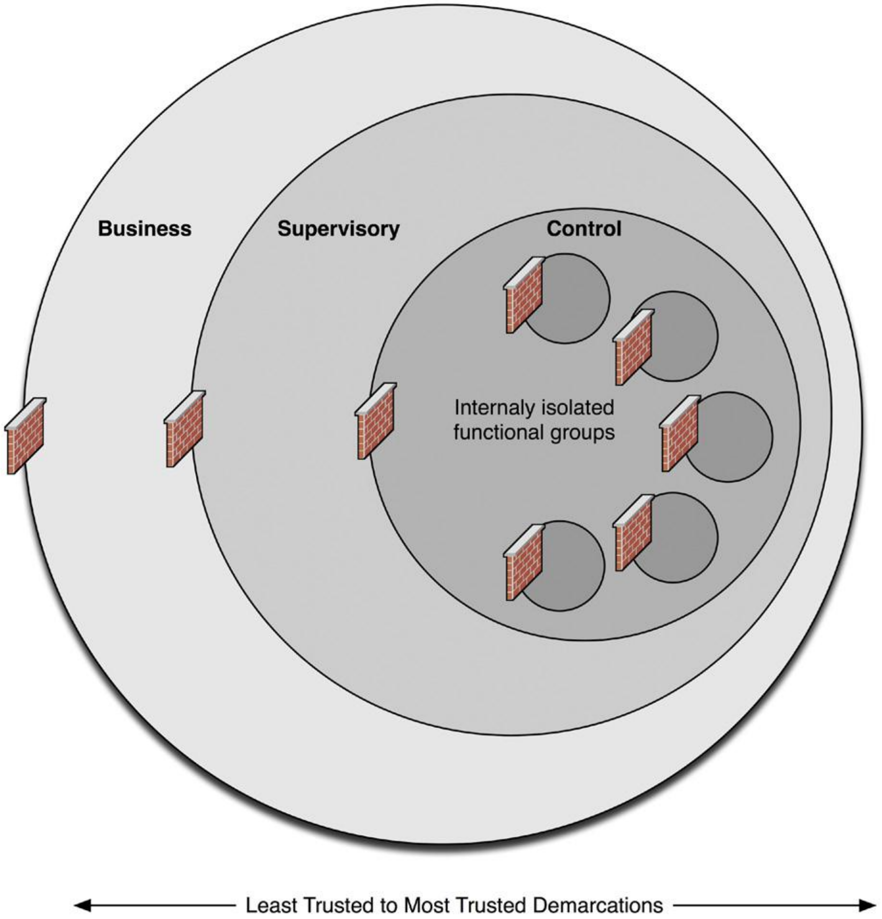


FIGURE 2.6

Topological Defense in Depth Provides Additional Layers of Protection.

through the careful creation and management of policies that implicitly define which servers can connect over a given protocol or port. This is covered in detail in Chapter 7, “Establishing Secure Enclaves.”

CAUTION

Don’t forget to control communications in both directions through a firewall. Not all threats originate from outside. Open, outbound traffic policies can facilitate an insider attack, enable the internal spread of malware, enable outbound command and control capabilities, or allow for data leakage or information theft.

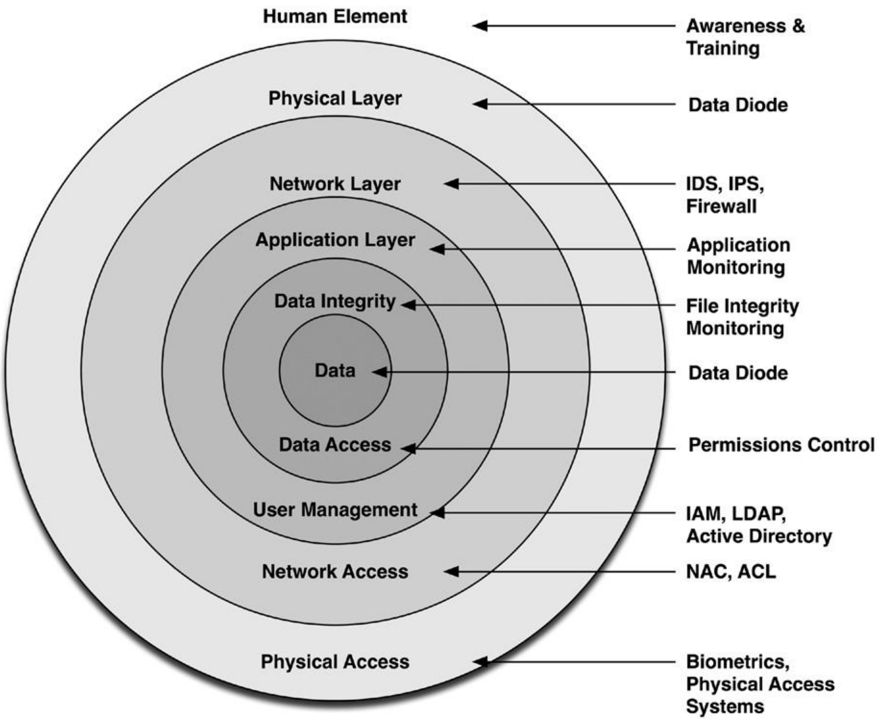


FIGURE 2.7

Defense in Depth with Corresponding Protective Measures.

Defense in Depth

All standards organizations, regulations, and recommendations indicate that a defense-in-depth strategy should be implemented. Although the definitions of "defense in depth" vary somewhat, the philosophy of a layered or tiered defensive strategy is considered a best practice. Figure 2.7 illustrates a common defense-in-depth model, mapping logical defensive levels to common security tools and techniques.

Interestingly, because of the segregated nature of most industrial systems, the term "defense in depth" can and should be applied in more than one context, including

- The layers of the Open Systems Interconnection (OSI) model, from physical (Layer 1) to Application (Layer 7).
- Physical or Topological layers consisting of subnetworks and/or functional groups.
- Policy layers, consisting of users, roles, and privileges.
- Multiple layers of defense devices at any given demarcation point (such as implementing a firewall and an **IDS** or **IPS**).

Table 2.1 Adding Context to User Authentication to Strengthen Access Control

Good	Better	Best
User accounts are classified by authority level	User accounts are classified by functional role	User accounts are classified by functional role and authority
Assets are classified in conjunction with user authority level	Assets are classified in conjunction with function or operational role	Assets are classified in conjunction with function and user authority
Operational controls can be accessed by any device based on user authority	Operational controls can be accessed by only those devices that are within a functional group	Operational controls can only be accessed by devices within a functional group by a user with appropriate authority

Access Control

Access control is one of the most difficult yet important aspects of cyber security. By locking down services to specific users or groups of users, it becomes more difficult for an attacker to identify and exploit systems. The further we can lock down access, the more difficult an attack becomes. Although many proven technologies exist to enforce access control—from network access control (**NAC**), authentication services, and others—the successful implementation of access control is difficult because of the complexity of managing users and their roles and mapping that to the specific devices and services that relate specifically to an employee's operational responsibilities. As shown in Table 2.1, the strength of access controls increases as a user's identity is treated with the additional context of that user's roles and responsibilities within a functional group.

Again, the more layers of complexity applied to the rules of user authentication and access, the more difficult it will be to gain unauthorized access. Some examples of advanced access control include the following:

- Only allow a user to log in to an HMI if the user has successfully badged into the control room (user credentials combined with physical access controls)
- Only allow a user to operate a given control from a specific controller (user credentials limited within a security enclave)
- Only allow a user to authenticate during that user's shift (user credentials combined with personnel management)

TIP

Authentication based on a combination of multiple and unrelated identifiers provides the strongest access control, for example, the use of both a digital and a physical key, such as a password and a biometric scanner.

THE USE OF TERMINOLOGY WITHIN THIS BOOK

Terminology specific to these various organizations and requirements will be used throughout this book. Although they may originate in a compliance mandate such as NERC CIP, they are used in the more open context of security best practices unless otherwise specified. Some terms that will be used extensively are routable and non-routable networks, assets (including cyber assets, critical assets, and critical cyber assets), enclaves, and electronic security perimeters or **ESPs**.

Networks, Routable and Non-routable

Although many think of a "network" as a Transmission Control Protocol/Internet Protocol (TCP/IP) network running on Ethernet, that assumption cannot be made when talking about industrial network security. Because many areas of industrial networks are connected using serial or bus networks, which operate via specific protocols, we need to expand our definition to include these areas of the industrial control systems. To make it easier to discern between the two network types, and to align with NERC CIP terminology, the terms "routable" and "non-routable" are used. A routable network typically means Ethernet and TCP/IP, although other routable protocols such as AppleTalk, DECnet, Novell IPX, and other legacy networking protocols certainly apply. "Routable" networks also include routable variants of SCADA and ICS protocols that have been modified to operate over TCP/IP, such as **Modbus/TCP** or ICCP over TCP/IP.

A "non-routable" network refers to those serial, bus, and point-to-point communication links that utilize **Modbus/RTU**, point-to-point ICCP, fieldbus, and other networks. They are still networks: they interconnect devices and provide a communication path between digital devices, and in many cases are designed for remote command and control.

Routable and non-routable networks generally interconnect at the demarcation between the control systems and the SCADA or supervisory networks, although in some cases (depending upon the specific industrial network protocols used) the two networks overlap. This is illustrated in Figure 2.8 and is discussed in more depth in Chapter 4, "Industrial Network Protocols," and Chapter 5, "How Industrial Networks Operate."

Assets, Critical Assets, Cyber Assets, and Critical Cyber Assets

An asset is a unique device that is used within an industrial control system. Assets are often computers, but also include network switches and routers, firewalls, printers, alarm systems, Human–Machine Interfaces (HMIs), **Programmable Logic Controllers** (**PLCs**), **Remote Terminal Units** (**RTUs**), and the various relays, actuators, sensors, and other devices that make up a typical control loop. As of version 3, NERC CIP defines a "cyber asset" as any device connected via a routable protocol, which limits the role of a cyber asset to those devices communicating on a

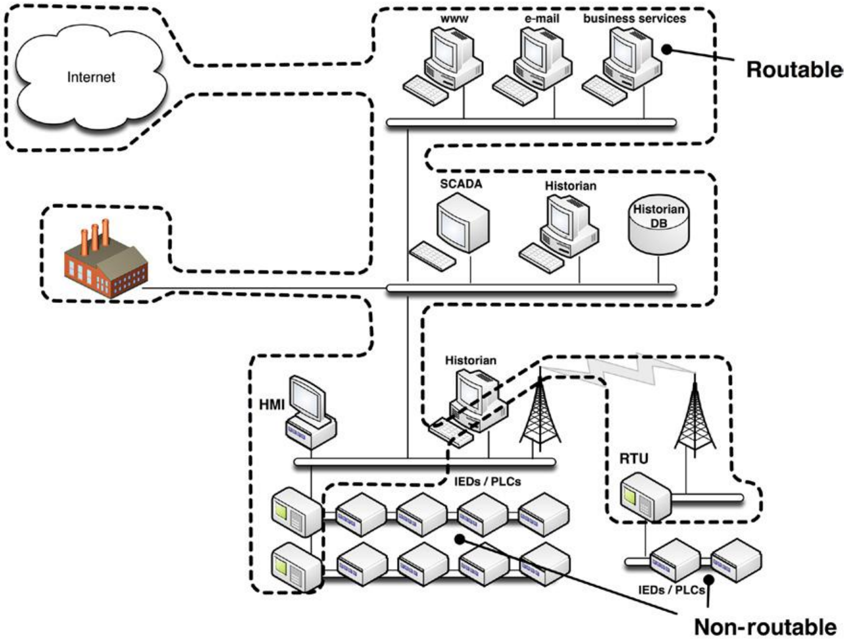


FIGURE 2.8

Routable and Non-routable Areas within an Industrial Control System.

routable LAN.[19] A "critical cyber asset," again as defined by NERC, is a cyber asset whose operation can impact the bulk energy system.[20]

In this book the broader definition of "asset" is used, in order to extend (as much as possible) cyber security to the non-routable devices such as PLCs and RTUs, which have been proven to be both targetable and vulnerable to cyber attack during the 2010 outbreak of **Stuxnet** (see "examples of Industrial Network Incidents" in Chapter 3, "Introduction to Industrial Network Security."

Enclaves

An "enclave" is a convenient term for defining a closed group of assets, similar to the functional "zone and conduit" model supported by ISA-99,[21] that is, the devices, applications, and users that should be interacting with each other legitimately in order to function, as illustrated in Figure 2.9. One example is a control loop: an HMI interfaces with a PLC which interacts with sensors, motors, valves, etc. to perform a specific control function. The "enclave" here includes all devices within the control loop including the PLC and HMI, and ideally the authorized users allowed to use the HMI. Nothing outside of this group should be interacting with anything inside of this group.

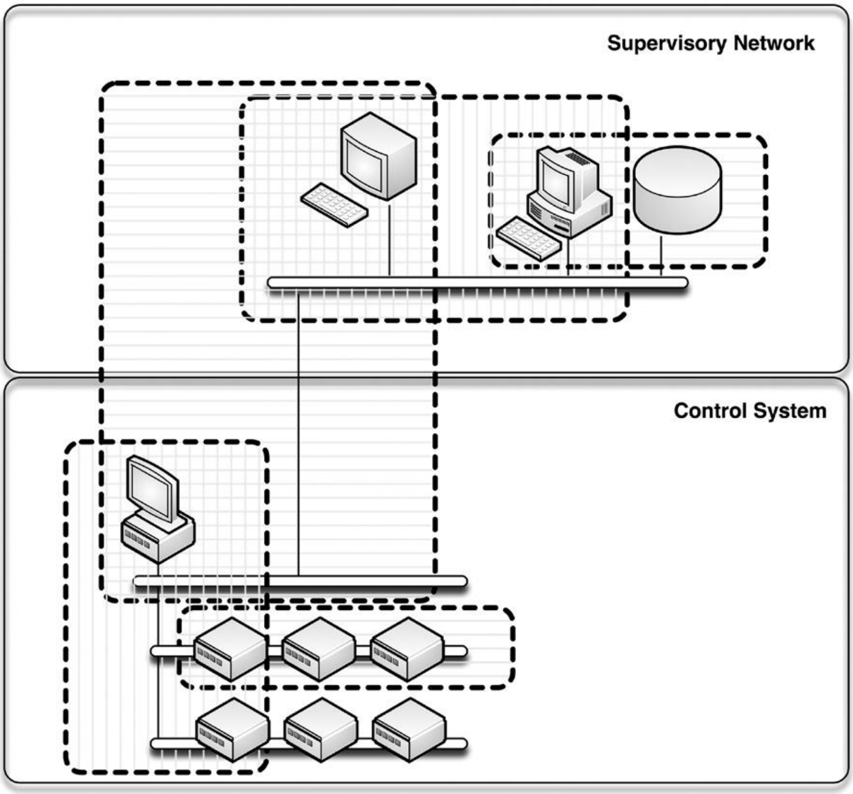


FIGURE 2.9

Basic Security Enclaves, Separating Both Logical and Physical Functional Groups.

NOTE

In the context of this book, an enclave is not (necessarily) a physical grouping of devices: it is a logical delineation of asset communication.

Enclaves are an important aspect of security as they define acceptable versus unacceptable behaviors. However, because a single asset can exist in multiple logical enclaves, the mapping and management of enclaves can become confusing. The concept of enclaves is expanded later in Chapter 7 "Establishing Secure Enclaves"; for now it's enough to understand the term and how it will be used.

Electronic Security Perimeters

The outermost boundary of any closed group of assets (i.e., an "enclave") is called the perimeter. Again, this supports NERC CIP terminology, where "Electronic

Security Perimeter" or "ESP" refers to the boundary between secure and nonsecure enclaves.[22] The perimeter itself is nothing more than the logical "dotted line" around an enclave that separates the closed group of assets within its boundaries from the rest of the network. "Perimeter defenses" are the security defenses established to police the entry into the enclave, and typically consist of a firewall and/or an Intrusion Prevention System (IPS).

A Note on Perimeterless Security

There is much debate about the ESP within the context of NERC and much discussion about a shift toward "perimeterless" security. In a perimeterless approach, there is no strict demarcation where all of our security products are concentrated. The goal is to move away from the "hard outer shell" with "soft gooey center" security practices that NERC's mandate of an ESP unintentionally promotes. Although future changes to NERC CIP may alter the terminology around establishing perimeter defenses, it will remain important to establish and enforce boundaries. This will be discussed further in Chapter 7, "Establishing Secure Enclaves."

SUMMARY

Understanding the basic nature of industrial networks, and examining the many regulations and recommendations put forth by NERC, NIST, NRC, ISA, the ISO/IEC, and other organizations is the foundation of industrial network security. By evaluating an industrial network, identifying and isolating its systems into functional groups or enclaves, and applying a structured methodology of defense in depth and strong access control, the security of the network as a whole will be greatly improved.

ENDNOTES

1. Department of Homeland Security, Homeland security presidential directive 7: critical infrastructure identification, prioritization, and protection. <http://www.dhs.gov/xabout/laws/gc_1214597989952.shtm>, September, 2008 (cited: November 1, 2010).
2. U.S. Nuclear Regulatory Commission, The NRC: who we are and what we do. <http://www.nrc.gov/about-nrc.html> (cited: November 1, 2010).
3. Department of Homeland Security, Homeland security presidential directive/HSPD-7. Roles and responsibilities of sector-specific federal agencies (18)(d). <http://www.dhs.gov/xabout/laws/gc_1214597989952.shtm>, September 2008 (cited: November 1, 2010).
4. J. Arlen, SCADA and ICS for security experts: how to avoid cyberdouchery. in: Proc. 2010 BlackHat Technical Conference, July 2010.
5. National Institute of Standards and Technology, Special Publication 800-53 Revision 3. Recommended Security Controls for Federal Information Systems and Organizations, August 2009, Computer Security Division, Information Technology Laboratory, National Institute of Standards and Technology, Gaithersburg, MD 20899-8930.

6. U.S. Nuclear Regulatory Commission, Regulatory Guide 5.71 (New Regulatory Guide) Cyber Security Programs for Nuclear Facilities, January 2010, U.S. Nuclear Regulatory Commission, Washington, DC.
7. Ibid.
8. Ibid.
9. Ibid.
10. Ibid.
11. National Institute of Standards and Technology, Special Publication 800-53 Revision 3. Recommended Security Controls for Federal Information Systems and Organizations, August 2009, Computer Security Division, Information Technology Laboratory, National Institute of Standards and Technology, Gaithersburg, MD.
12. Ibid.
13. Department of Homeland Security, Risk-based performance standards guidance; chemical facility anti-terrorism standards, Department of Homeland Security, Office of Infrastructure Protection, Infrastructure Security Compliance Division, Washington, DC, May 2009.
14. Ibid.
15. ANSI/ISA-TR99.00.01-2007, Security technologies for industrial automation and control systems, 2007, International Society of Automation (ISA), Research Triangle Park, NC.
16. Ibid.
17. International Standards Organization, ISO/IEC 27002:2005. Information Technology—Security Techniques—Code of Practice for Information Security Management, ISO/IEC, Geneva, Switzerland, 2005.
18. U.S. Nuclear Regulatory Commission, Regulatory Guide 5.71 (New Regulatory Guide) Cyber Security Programs for Nuclear Facilities, U.S. Nuclear Regulatory Commission, Washington, DC, January 2010.
19. North American Electric Corporation, Standard CIP–002–3, Cyber Security, Critical Cyber Asset Identification, North American Electric Corporation (NERC), Princeton, NJ, December 16, 2009.
20. Ibid.
21. ANSI/ISA-TR99.00.01-2007, Security technologies for industrial automation and control systems, 2007, International Society of Automation (ISA), Research Triangle Park, NC.
22. North American Electric Corporation, Standard CIP–002–3, Cyber Security, Critical Cyber Asset Identification, North American Electric Corporation (NERC), Princeton, NJ, December 16, 2009.

CHAPTER

Introduction to Industrial Network Security

3

INFORMATION IN THIS CHAPTER:

- The Importance of Securing Industrial Networks
- The Impact of Industrial Network Incidents
- Examples of Industrial Network Incidents
- APT and Cyber War

Securing an industrial network, although similar in many ways to standard enterprise information security, presents several unique challenges. Because industrial systems are built for reliability and longevity, the systems and networks used are easily outpaced by the tools employed by an attacker. An industrial control system may be expected to operate without pause for months or even years, and the overall life expectancy may be measured in decades. Attackers, on the contrary, have easy access to new exploits and can employ them at any time. Security considerations and practices have also lagged, largely for the same reason: the systems used predate modern network infrastructures, and so they have always been secured physically rather than digitally.

Because of the importance of industrial networks and the potentially devastating consequences of an attack, new security methods need to be adopted. As can be seen in real-life examples of industrial cyber sabotage (see the section "Examples of Industrial Network Incidents"), our industrial networks are being targeted. Furthermore, they are the target of a new threat profile that utilizes more sophisticated and targeted attacks than ever before.

THE IMPORTANCE OF SECURING INDUSTRIAL NETWORKS

The need to improve the security of industrial networks cannot be overstated. Many industrial systems are built using legacy devices, in some cases running legacy protocols that have evolved to operate in routable networks. Before the proliferation of Internet connectivity, web-based applications, and real-time business information systems, energy systems were built for reliability. Physical security was always a concern, but information security was not a concern, because the control systems were air-gapped—that is, physically separated with no common system (electronic or otherwise) crossing that gap, as illustrated in Figure 3.1.

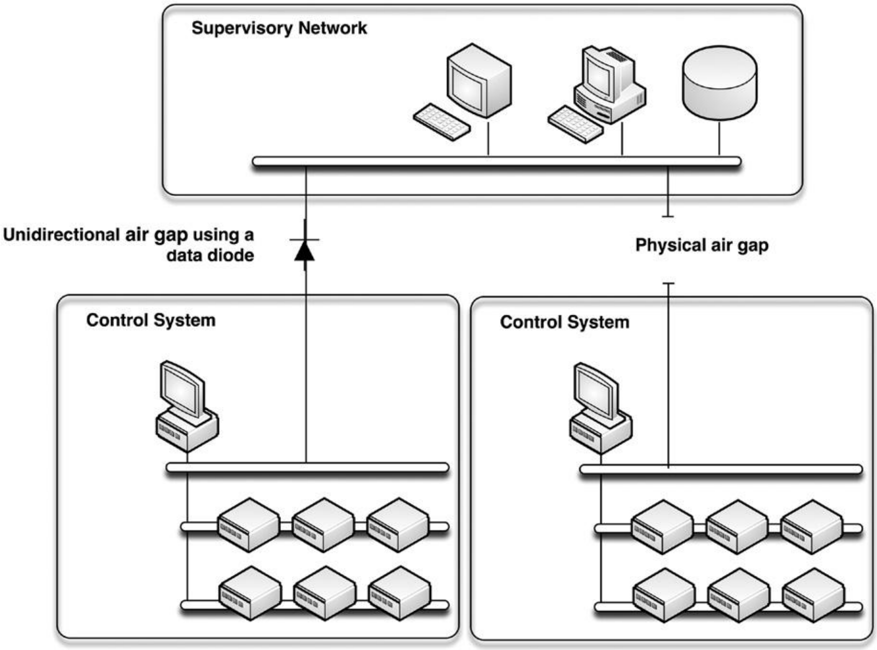


FIGURE 3.1

Air Gap Separation.

Ideally, the air gap would still exist, and it would still apply to digital communication, but in reality it does not. As the business operations of industrial networks evolved, the need for real-time information sharing evolved as well. Because the information required originated from across the air gap, a means to bypass the gap needed to be found. Typically, a firewall would be used, blocking all traffic except what was absolutely necessary in order to improve the efficiency of business operations.

The problem is that—regardless of how justified or well intended the action—the air gap no longer exists, as seen in Figure 3.2. There is now a path into critical systems, and any path that exists can be found and exploited.

Security consultants at Red Tiger Security presented research in 2010 that clearly indicates the current state of security in industrial networks. Penetration tests were performed on approximately 100 North American electric power generation facilities, resulting in more than 38,000 security warning and vulnerabilities.[1] Red Tiger was then contracted by the Department of Homeland Security (DHS) to analyze the data in search of trends that could be used to help identify common attack vectors and, ultimately, to help improve the security of these critical systems against cyber attack.

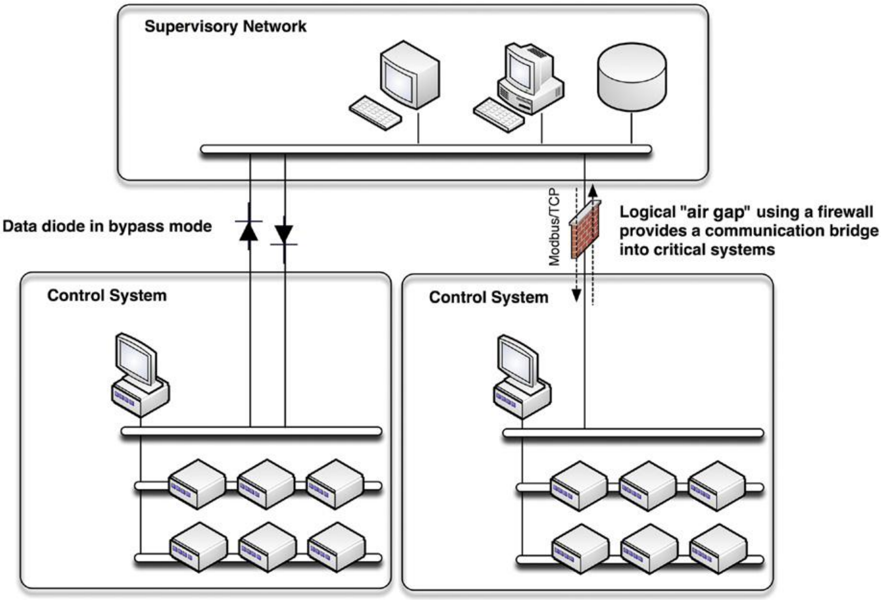

Supervisory Network
Data diode in bypass mode
Modbus/TCP
Logical "air gap" using a firewall provides a communication bridge into critical systems
Control System
Control System

FIGURE 3.2

The Reality of the Air Gap.

The results were presented at the 2010 BlackHat conference and implied a security climate that was lagging behind other industries. The average number of days between the time when the **vulnerability** was disclosed publicly and the time when the vulnerability was discovered in a control system was 331 days: almost an entire year. Worse still, there were cases of vulnerabilities that were over 1100 days old, nearly 3 years past their respective "zero-day."[2]

What does this tell us? It tells us that there are known vulnerabilities that can allow hackers' and cyber criminals' entry into our control networks. A vulnerability that has been disclosed for almost a year has almost certainly been made readily available within open source penetration testing utilities such as **Metasploit** and Backtrack, making exploitation of those vulnerabilities fairly easy and available to a wide audience.

It should not be a surprise that there are well-known vulnerabilities within control systems. Control systems are by design very difficult to patch. By intentionally limiting (or even better, eliminating) access to outside networks and the Internet, simply obtaining patches can be difficult. Because reliability is paramount, actually applying patches once they are obtained can also be difficult and restricted to planned maintenance windows. The result is that there are almost always going to be unpatched vulnerabilities, although reducing the window from an average of 331 days to a weekly or even monthly maintenance window would be a huge improvement.

THE IMPACT OF INDUSTRIAL NETWORK INCIDENTS

Industrial networks are responsible for process and manufacturing operations of almost every scale, and as a result the successful penetration of a control system network can be used to directly impact those processes. Consequences could potentially range from relatively benign disruptions, such as the disruption of the operation (taking a facility offline), the alteration of an operational process (changing the formula of a chemical process or recipe), all the way to deliberate acts of sabotage that are intended to cause harm. For example, manipulating the feedback loop of certain processes could cause pressure within a boiler to build beyond safe operating parameters, as shown in Figure 3.3. Cyber sabotage could result in injury or loss of life, including the loss of critical services (blackouts, unavailability of vaccines, etc.) or even catastrophic explosions.

Safety Controls

To avoid catastrophic failures, most industrial networks employ automated safety systems. However, many of these safety controls employ the same messaging and control protocols used by the industrial control network's operational processes, and in some cases, such as certain fieldbus implementations, the safety systems are supported directly within the same communications protocols as the operational controls, on the same physical media (see Chapter 4, "Industrial Network Protocols," for details and security concerns of industrial control protocols).

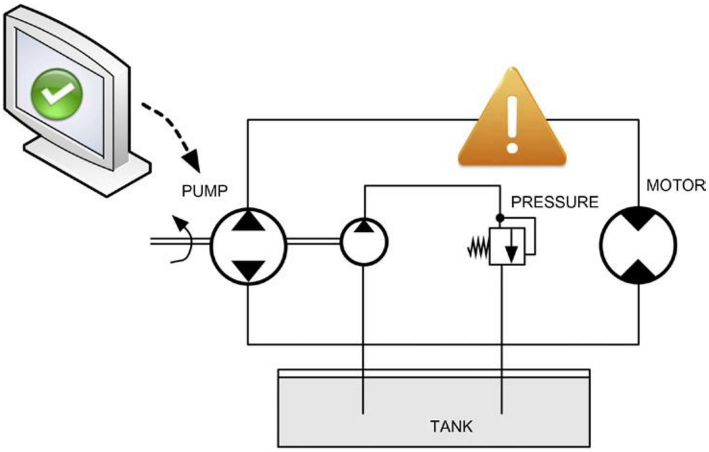


FIGURE 3.3

Disruption of a Control Process Can Cause Catastrophic Failure(s).

Although safety systems are extremely important, they have also been used to downplay the need for heightened security of industrial networks. However, research has shown that real consequences can occur in modeled systems. Simulations performed by the Sandia National Laboratories showed that simple Man-in-the-Middle (MITM) attacks could be used to change values in a control system and that a modest-scale attack on a larger bulk electric system using targeted malware (in this scenario, targeting specific control system front end processors) was able to cause significant loss of generation.[3]

The European research team VIKING (Vital Infrastructure, Networks, Information and Control Systems Management) is currently investigating threats of a different sort. The Automatic Generation Control (AGC) of electric utilities operates in an entirely closed loop—that is, the control process completes entirely within the logic of the SCADA system, without human intervention or control. Rather than breaching a control system through the manipulation of an HMI, VIKING's research attempts to investigate whether the manipulation of input data could alter the normal control loop functions, ultimately causing a disturbance.[4]

TIP

When establishing a cyber security plan, think of security and safety as two entirely separate entities. Do not assume that security leads to safety or that safety leads to security. If an automated safety control is compromised by a cyber attack (or otherwise disrupted), the necessity of having a strong digital defense against the manipulation of operations becomes even more important. Likewise, a successful safety policy should not rely on the security of the networks used. By planning for both safety and security controls that operate independently of one another, both systems will be inherently more reliable.

Consequences of a Successful Cyber Incident

A successful cyber attack on a control system can either

- delay, block, or alter the intended process, that is, alter the amount of energy produced at an electric generation facility.
- delay, block, or alter information related to a process, thereby preventing a bulk energy provider from obtaining production metrics that are used in energy trading or other business operations.

The end result could be penalties for regulatory non-compliance or the financial impact of lost production hours due to misinformation or denial of service. An incident could impact the control system in almost any way, from taking a facility offline, disabling or altering safeguards, and even causing life-threatening incidents within the plant—up to and including the release or theft of hazardous materials or direct threats to national security.[5] The possible damages resulting from a cyber incident vary depending upon the type of incident, as shown in Table 3.1.

Table 3.1 The Potential Impact of Successful Cyber Attacks

Incident Type	Potential Impact
Change in a system, operating system, or application configuration	Introduction of command and control channels into otherwise secure system
	Suppression of alarms and reports to hide malicious activity
	Alteration of expected behavior to produce unwanted and unpredictable results
Change in programmable logic in PLCs, RTUs, or other controllers	Damage to equipment and/or facilities
	Malfunction of the process (shutdown)
	Disabling control over a process
Misinformation reported to operators	Causing inappropriate actions in response to misinformation that could result in a change in programmable logic
	Hiding or obfuscating malicious activity, including the incident itself or injected code (i.e., a rootkit)
Tampering with safety systems or other controls	Preventing expected operations, fail safes, and other safeguards with potentially damaging consequences
Malicious software (malware) infection	May initiate additional incident scenarios
	May impact production, or force assets to be taken offline for forensic analysis, cleaning, and/or replacement
	May open assets up to further attacks, information theft, alteration, or infection
Information theft	Sensitive information such as a recipe or chemical formula are stolen
Information alteration	Sensitive information such as a recipe or chemical formula is altered in order to adversely affect the manufactured product

EXAMPLES OF INDUSTRIAL NETWORK INCIDENTS

Over the past decade, there have been numerous incidents, outages, and other failures that have been identified as the result of a cyber incident. In 2000, a disgruntled man in Australia who was rejected for a government job was accused of using a radio transmitter to alter electronic data within a sewerage pumping station, causing the release of over two hundred thousand gallons of raw sewage into nearby rivers.[6]

In 2007, there was the **Aurora Project:** a controlled experiment by the Idaho National Laboratories (INL), which successfully demonstrated that a controller could be destroyed via a cyber attack. The vulnerability allowed hackers—which in this case were white-hat security researchers at the INL—to successfully open and close breakers on a diesel generator out of synch, causing an explosive failure. In September 2007, CNN reported on the experiment, bringing the security of our power infrastructure into the popular media.[7]

The Aurora vulnerability remains a concern today. Although the North American Electric Reliability Corporation (NERC) first issued an alert on Aurora a few months before CNN's report in June 2007, it has since provided additional alerts, as recent as an October 2010 alert that provides clear mitigation strategies for dealing with the vulnerability.[8]

In 2008, the agent.btz worm began infecting U.S. military machines and was reportedly carried into CENTCOM's classified network on a USB thumb drive later that year. Although the CENTCOM breach, reported by CBS' *60 Minutes* in November 2009, was widely publicized, the specifics are difficult to ascertain and the damages and intentions remain highly speculative.[9]

Not to be confused with the Aurora Project is another recent attack called Operation Aurora that hit Google and others in late 2009 and put the spotlight on the sophisticated new arsenal of cyber war. Operation Aurora used a zero-day exploit in Internet Explorer to deliver a payload designed to exfiltrate protected intellectual property. Operation Aurora changed the threat landscape from denial of service attacks and malware designed to damage or disable networks to targeted attacks designed to operate without disruption, to remain stealthy, and to steal information undetected. Aurora consisted of multiple individual pieces of malware, which combined to establish a hidden presence on a host and then communicate over a sophisticated command and control (C2) channel that employed a custom, encrypted protocol that mimicked common HTTPS traffic on port 443 encrypted via Secure Sockets Layer (SSL).[10] Although CENTCOM and Operation Aurora did not target industrial networks specifically, they exemplifed the evolving nature of threats. In other words, Aurora demonstrated the existence of the "Advanced Persistent Threats" (**APTs**), just as a more recent worm demonstrated the existence of targeted cyber weapons and the machinations of cyber war.

This later worm, of course, is Stuxnet: the new weapon of cyber war, which began to infect industrial control systems in 2010. Any speculation over the possibility of a targeted cyber attack against an industrial network has been overruled by this extremely complex and intelligent collection of malware. Stuxnet is a tactical nuclear missile in the cyber war, and it was not a shot across the bow: it hit its mark and left behind the proof that extremely complex and sophisticated attacks can and do target industrial networks. The worst-case scenario has now been realized: industrial vulnerabilities have been targeted and exploited by an APT.

Although Stuxnet was first encountered in June 2009, widespread discussions about it did not occur until the summer of 2010, after an Industrial Control Systems Cyber Emergency Response Team (ICS-CERT) advisory was issued.[11] Stuxnet uses four zero-days in total to infect and spread, looking for SIMATIC WinCC and PCS 7 programs from Siemens, and then using default SQL account credentials to infect connected Programmable Logic Controllers (PLCs) by injecting a rootkit via the Siemens fieldbus protocol, **Profibus**. Stuxnet then looks for automation devices using a frequency converter that controls the speed of a motor. If it sees a controller operating within a range of 800–1200 Hz, it attempts to sabotage the operation.[12]

Although little was known at first, Siemens effectively responded to the issue, quickly issuing a security advisory, as well as a tool for the detection and removal of Stuxnet. Stuxnet drew the attention of the mass media through the fall of 2010 for being the first threat of its kind—a sophisticated and blended threat that actively targets SCADA systems—and it immediately raised the industry's awareness of advanced threats, and illustrated exactly why industrial networks need to dramatically improve their security measures.

Dissecting Stuxnet

Stuxnet is very complex, as can be seen in Figure 3.4. It was used to deliver a payload targeting a specific control system. It is the first industrial control system rootkit. It can self-update even when cut off from C2 (which is necessary should it find its way into a truly air-gapped system). It is able to inject code into the ladder logic of PLCs, and at that point alter the operations of the PLC as well as hide itself by reporting false information back to the HMI. It adapts to its environment. It uses system-level, hard-coded authentication credentials that were not publicly disclosed. It signed itself with legitimate certificates manufactured using stolen keys. There is no doubt about it at this time: Stuxnet is an advanced new weapon in the cyber war.

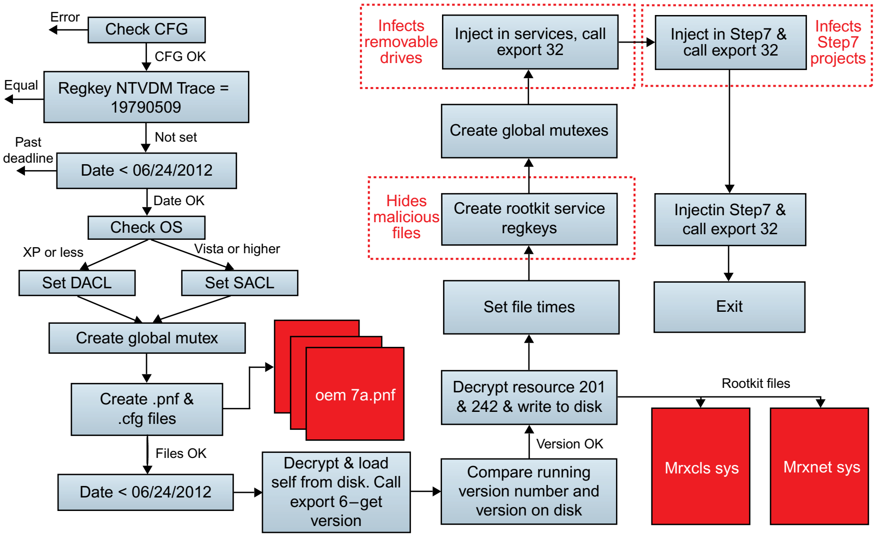


FIGURE 3.4

Stuxnet's Infection Processes.[13]

Courtesy of Symantec.

What It Does

The full extent of what Stuxnet is capable of doing is not known at the time of this writing. What we do know is that Stuxnet does the following:[14]

- Infects Windows systems using a variety of zero-day exploits and stolen certificates, and installing a Windows rootkit on compatible machines.
- Attempts to bypass behavior-blocking and host intrusion protection based technologies that monitor LoadLibrary calls by using special processes to load any required DLLs, including injection into preexisting trusted processes.
- Typically infects by injecting the entire DLL into another process and only exports additional DLLs as needed.
- Checks to make sure that its host is running a compatible version of Windows, whether or not it is already infected, and checks for installed **Anti-Virus** before attempting to inject its initial payload.
- Spreads laterally through infected networks, using removable media, network connections, and/or Step7 project files.
- Looks for target industrial systems (Siemens WinCC SCADA). When found, it uses hard-coded SQL authentication within the system to inject code into the database, infecting the system in order to gain access to target PLCs.
- Injects code blocks into the target PLCs that can interrupt processes, inject traffic on to the Profibus, and modify the PLC output bits, effectively establishing itself as a hidden rootkit that can inject commands to the target PLCs.
- Uses infected PLCs to watch for specific behaviors by monitoring Profibus (The industrial network protocol used by Siemens. See Chapter 4, "Industrial Network Protocols," for more information on Profibus).
- If certain frequency controller settings are found, Stuxnet will throttle the frequency settings from 1410 to 2 Hz, in a cycle.
- It includes the capabilities to remove itself from incompatible systems, lie dormant, reinfect cleaned systems, and communicate peer to peer in order to self-update within infected networks.

What we do not know at this point is what the full extent of damage could be from the malicious code that is inserted within the PLC. Subtle changes in **set points** over time could go unnoticed that could cause failures down the line, use the PLC logic to extrude additional details of the control system (such as command lists), or just about anything. Because Stuxnet has exhibited the capability to hide itself and lie dormant, the end goal is still a mystery.

Lessons Learned

Because Stuxnet is such a sophisticated piece of malware, there is a lot that we can learn from dissecting it and analyzing its behavior. How did we detect Stuxnet? Largely because it was so widespread. Had it been deployed more tactically, it might have gone unnoticed: altering PLC logic and then removing itself from the WinCC hosts that were used to inject those PLCs. How will we detect the next one? The truth is that we may not, and the reason is simple: our "barrier-based"

Table 3.2 Lessons Learned from Stuxnet

Previous Beliefs	Lessons Learned from Stuxnet
Control systems can be effectively isolated from other networks, eliminating risk of a cyber incident	Control systems are still subject to human nature: a strong perimeter defense can be bypassed by a curious operator, a USB drive, and poor security awareness
PLCs and RTUs that do not run modern operating systems lack the necessary attack surface to make them vulnerable	PLCs can and have been targeted and infected by malware
Highly specialized devices benefit from "security through obscurity." Because industrial control systems are not readily available, it is impossible to effectively engineer an attack against them	The motivation, intent, and resources are all available to successfully engineer a highly specialized attack against an industrial control system
Firewalls and Intrusion Detection and Prevention system (IDS/IPS) are sufficient to protect a control system network from attack	The use of multiple zero-day vulnerabilities to deploy a targeted attack indicates that "**blacklist**" point defenses, which compare traffic to definitions that indicate "bad" code are no longer sufficient, and "whitelist" defenses should be considered as a catchall defense against unknown exploits

methodologies do not work against cyber attacks that are this well researched and funded. They are delivered via zero-days, which means we do not detect them until they have been deployed, and they infect areas of the control system that are difficult to monitor.

So what do we do? We learn from Stuxnet and change our perception and attitude toward industrial network security (see Table 3.2). We adopt a new "need to know" mentality of control system communication. If something is not explicitly defined, approved, and allowed to communicate, it is denied. This requires understanding how control system communications work, establishing that "need to know" in the form of well-defined security enclaves, establishing policies and baselines around those enclaves that can be interpreted by automated security software, and **whitelisting** everything.

It can be seen in Table 3.2 that additional security measures need to be considered in order to address new "Stuxnet-class" threats that go beyond the requirements of compliance mandates and current best-practice recommendations. New measures include Layer 7 application session monitoring to discover zero-day threats and to detect covert malware communications. They also include more clearly defined security policies to be used in the adoption of policy-based user, application, and **network whitelisting** to control behavior in and between enclaves (see Chapter 7, "Establishing Secure Enclaves").

TIP

Before Stuxnet, the axiom "to stop a hacker, you need to think like a hacker" was often used, meaning that in order to successfully defend against a cyber attack you need to think in terms of someone trying to penetrate your network. This philosophy still has merit, the only difference being that now the "hacker" can be thought of as having a much greater knowledge of control systems, as well as significantly more resources and motivation. In the post-Stuxnet world, imagine that you are building a digital bunker in the cyber war, rather than simply defending a network, and aim for the best possible defenses against the worst possible attack.

Night Dragon

In February 2011, McAfee announced the discovery of a series of coordinated attacks against oil, energy, and petrochemical companies. The attacks, which originated primarily in China, were believed to have originated in 2009, operating continuously and covertly for the purpose of information extraction,[15] as is indicative of an APT.

Night Dragon is further evidence of how an outside attacker can (and will) infiltrate critical systems. Although the attack did not result in sabotage, as was the case with Stuxnet, it did involve the theft of sensitive information. The intended use of this information is unknown at this time. The information that was stolen could be used for almost anything, and for a variety of motives. It began with SQL injections against corporate web servers, which were then used to access intranet servers. Using standard tools, attackers gained additional usernames and passwords to enable further infiltration to internal desktop PCs and servers. Night Dragon established command and control servers as well as Remote Administration Toolkits (RATs), primarily to extract e-mail archives from executive accounts.[16] Although it is important to note that the industrial control systems of the target companies were not affected, important information could have been obtained regarding the design and operation of those systems, which could be used in a later, more targeted attack. As with any APT, Night Dragon is surrounded with uncertainty and supposition. After all, APT is an act of cyber espionage: one that may or may not develop into a more targeted cyber war.

APT AND CYBER WAR

The terms *APT* and *cyber war* are often used interchangeably, and they can be related, but they differ enough in their intent to justify distinct classifications of modern, sophisticated network threats.

Although both are of concern to industrial networks, it is important to understand their differences and intentions, so that they can be better addressed. It is also important to understand that both are types of threat behaviors that consist of various exploits and are not specific exploits or pieces of malware themselves. That is, "APT" classifies a group of exploits (delivery) to infect a network with malicious code (the payload) that is designed to accomplish a specific goal (information theft). Cyber war similarly classifies a threat that can include distinct delivery mechanisms to deliver payloads of various intents.[17] Although both can utilize similar techniques, exploits, and even common code, the differences between APT and cyber war at a higher level distinguish one from another, as can be seen in Table 3.3.

Just as APT and cyber war differ in intent, they can also differ in their targets, as seen in Table 3.4. Again, the methods used to steal intellectual property for profit

Table 3.3 Distinctions between APT and Cyber War

APT Qualities	Cyber War Qualities
Often uses simple exploits for initial infection	Uses more sophisticated vectors for initial infection
Designed to avoid detection over long periods of time	Designed to avoid detection over long periods of time
Designed to communicate information back to the attacker using covert command and control (C2)	Designed to operate in isolation, not dependent upon remote command and control (C2)
Mechanisms for persistent operation even if detected	Mechanisms for persistent operation or reinfection if detected
Not intended to impact or disrupt network operations	Possible intentions include network disruption

Table 3.4 Information Targets of APT and Cyber War

APT Targets	Cyber War Targets
Intellectual Property	
Application code	Certificates and authority
Application design	Control protocols
Protocols	Functional diagrams
Patents	PCS command codes
Industrial Designs	
Product schematics	Control system designs and schematics
Engineering designs and drawings	Safety controls
Research	PCS weaknesses
Chemicals and Formulas	
Pharmaceutical formulas	Pharmaceutical formulas
Chemical equations	Pharmaceutical safety and allergy information
Chemical compounds	Chemical hazards and controls

and the methods used to steal intellectual property to sabotage an industrial system can be the same. However, by determining the target of attack, insight into the nature of the attack can be inferred. The difference is a subtle one and is made here in an attempt to highlight the level of severity and sophistication that should be considered when securing industrial networks. That is, blended attacks designed to be persistent and undetected represent the APT, while these same blended and stealthy attacks can be weaponized and used for cyber sabotage. APT can be used to obtain information that is later used to construct new zero-day exploits. APT can also be used to obtain information necessary to design a targeted payload—such as the one used by Stuxnet—that can be delivered using those exploits. In other words, the methods, intentions, and impact of cyber war should be treated as even more sophisticated than the APT.

The Advanced Persistent Threat

The Advanced Persistent Threat, or APT, has earned broad media attention in recent years. The Aurora Project and Stuxnet's high publicity increased awareness of new threat behaviors both within and outside of the information security communities: Incident researchers such as Exida (http://www.exida.com), Lofty Perch (http://www.loftyperch.com), and Red Tiger Security (http://www.redtigersecurity.com) specialize in the incident response of APT; organizations such as RISI (Repository of Industrial Security Incidents; http://www.securityincidents.org) have been developed to catalogue incident behavior; and regulatory and CERT organizations have issued warnings for APTs, including an NERC alert issued by the North American Electric Reliability Corporation for both Aurora and Stuxnet, requiring direct action from its member electric utilities, with clear penalties for noncompliance.[18]

With all of this attention, a lot has been determined about how APTs function. One differentiator of an APT is a shift from broad, untargeted attacks to more directed attacks that focus on determining specifics about its target network. APTs spread and learn, and exfiltrate information through covert communications channels. Often, APT relies upon outside C2, although in some cases such as Stuxnet, APT threats are capable of operating in isolation.[19]

Another differentiator of APT from normal malware or hack attempts is an attempt to remain hidden and to proliferate within a network, leading to the persistence of the threat. This typically includes a tiered infection model, where increasingly sophisticated methods of covert communication are established. The most basic will operate in an attempt to obtain information from the target, whereas one or more increasingly sophisticated mechanisms will remain dormant. This tiered model increases the persistence of the threat, where the more difficult to detect infections only awaken after the removal of the initial APT. In this way, "cleaned" machines can remain infected. This is one reason why it is important to thoroughly investigate an APT before attempting disinfection, as the initially detected threat may be easier to deal with, while higher-level programs remain dormant.[20]

The end result of APT's relentless, layered approach is the deliberate exfiltration of data. Proprietary information can achieve anything from increased competitiveness in manufacturing and design (making the data valuable on the black market), to direct financial benefit achieved through the theft of financial resources and records. Highly classified information may also be valuable for the development of further, more sophisticated APTs, or even weaponized threats for use in cyber sabotage and cyber warfare.

Common APT Methods

The methods used by APT are diverse. Within industrial networks, incident data has been analyzed, and specific attacker profiles have been identified. The attacks themselves tend to be fairly straightforward, using Open Source Intelligence (OSINT) to facilitate social engineering, targeted spear phishing (customized e-mails designed to trick readers into clicking on a link, opening an attachment, or otherwise triggering malware), malicious attachments, removable media such as USB drives, and malicious websites as initial infection vectors.[21] APT payloads (the malware itself) range from freely available kits such as Webattacker and torrents, to commercial malware such as Zeus (ZBOT), Ghostnet (Ghostrat), Mumba (Zeus v3), and Mariposa. Malware delivery is typically obfuscated to avoid detection by Anti-Virus and other detection mechanisms.[22]

Once a network is infected, APT strives to operate covertly and may attempt to deactivate or circumvent Anti-Malware software, establish backdoor channels, or open holes in firewalls.[23] Stuxnet, for example, attempts to avoid discovery by bypassing host intrusion detection and also by removing itself from systems that are incompatible with its payload.[24]

Because the techniques used are for the most part common infection vectors and known malware, what is so "advanced" about the APT? One area where APT is often very sophisticated is in the knowledge of its target—known information about the target and the people associated with that target. For example, highly effective spear phishing may utilize knowledge of the target corporation's organization structure (e.g., a mass e-mail that masquerades as a legitimate e-mail from an executive within the company), or of the local habits of employees (e.g., a mass e-mail promising discounted lunch coupons from a local eatery).[25]

Cyber War

Unlike APT, where the initial infections are typically from focused yet simple exploits such as spear phishing (the "advanced" moniker comes from the behavior of the threat after infection), the threats associated with cyber war trend toward more sophisticated delivery mechanisms and payloads.[26] Stuxnet utilized multiple zero-day exploits for infection, for example. The development of one zero-day requires resources: the financial resources to purchase commercial malware or the intellectual resources with which to develop new malware. Stuxnet raised a high degree of speculation about its source and its intent at least partly due to the level of resources required to deliver the worm through so many zero-days. Stuxnet also used "insider intelligence" to focus on its target control system, which again implied that the creators of Stuxnet had significant resources: they either had access to an industrial control system with which to develop and test their malware, or they had enough knowledge about how such a control system was built that they were able to develop it in a simulated environment.

That is, the developers of Stuxnet could have used stolen intellectual property—which is the primary target of the Advanced Persistent Threat—to develop a more weaponized piece of malware. In other words, APT is a logical precursor to cyber war. In the case of Stuxnet, it is pure speculation: at the time of this writing, the creators of Stuxnet are unknown, as is their intent.

Two important inferences can be made by comparing APT and cyber warfare. The first is that cyber warfare is higher in sophistication and in consequence, mostly due to available resources of the attacker and the ultimate goal of destruction versus profit. The second is that in many industrial networks, there is less profit available to a cyber attacker than from others. If the industrial network you are defending is largely responsible for commercial manufacturing, signs of an APT are likely evidence of attempts at intellectual theft. If the industrial network you are defending is critical and could potentially impact lives, signs of an APT could mean something larger, and extra caution should be taken when investigating and mitigating these attacks.

Emerging Trends in APT and Cyber War

Through the analysis of known cyber incidents, several trends can be determined in how APT and cyber attacks are being performed. These include, but are not limited to, a shift in the initial infection vectors and the qualities of the malware used, its behavior, and how it infects and spreads.

Although threats have been trending "up the stack" for some time, where exploits are moving away from network-layer and protocol-layer vulnerabilities and more toward application-specific exploits, even more recent trends show signs that these applications are shifting away from the exploitation of Microsoft software products toward the almost ubiquitously deployed Adobe Portable Document Format (PDF) and its associated software products.

Web-based applications are also used heavily both for infections and for C2. The use of social networks such as Twitter, Facebook, Google groups, and other cloud services is ideal for both because they are widely used, highly accessible, and difficult to monitor. Many companies actually embrace social networking for marketing and sales purposes, often to the extent that these services are allowed open access through corporate firewalls.

The malware itself, of course, is also evolving. There is growing evidence among incident responders and forensics teams of deterministic malware and even the emergence of mutating bots. Stuxnet, again, is a good example: it contains robust logic and will operate differently depending upon its environment. It will spread, attempt to inject PLC code, communicate via C2, lie dormant, or awaken depending upon changes to its environment.

Evolving Vulnerabilities: The Adobe Exploits

Adobe Postscript Document Format (PDF) exploits are an example of the shifting attack paradigm from lower-level protocol and OS exploits to the manipulation of application contents. At a very high level, the exploits utilize PDFs' ability to call and execute code to execute malicious code: either by calling a malicious website or by injecting the code directly within the PDF file. It works like this:

- An e-mail contains a compelling message, a properly targeted spear-phishing message. There is a .pdf attachment.
- This PDF uses a feature, specified in the PDF format, known as a "Launch action." Security researcher Didier Stevens successfully demonstrated that Launch actions can be exploited and can be used to run an executable embedded within the PDF file itself.[27]
- The malicious PDF also contains an embedded file named Discount_at_Pizza_Barn_Today_Only.pdf, which has been compressed inside the PDF file. This attachment is actually an executable file, and if the PDF is opened and the attachment is run, it will execute.
- The PDF uses the JavaScript function exportDataObject to save a copy of the attachment to the user's PC.
- When this PDF is opened in Adobe Reader (JavaScript must be enabled), the exportDataObject function causes a dialog box to be displayed asking the user to "Specify a file to extract to." The default file is the name of the attachment, Discount_at_Pizza_Barn_Today_Only.pdf. The exploit requires that the users' naïveté and/or their confusion will cause them to save the file.
- Once the exportDataOject function has completed, the Launch action is run. The Launch action is used to execute the Windows command interpreter (cmd .exe), which searches for the previously saved executable attachment Discount_at_Pizza_Barn_Today_Only.pdf and attempts to execute it.
- A dialogue box will warn users that the command will run only if the user clicks "Open."

The hack has been used to spread known malware, including ZeusBot.[28] Although it does require user interaction, PDF files are extremely common, and when combined with a quality spear-phishing attempt, this attack can be very effective.

Another researcher chose to infect the benign PDF with another /Launch hack that redirected a user to a website, but noted that it could have just as easily been an exploit pack and/or embedded Trojan binary. The dialogue box used to warn users can also be modified, increasing the likeliness that even a normally cautious user will execute the file.[29]

Antisocial Networks: A New Playground for Malware

Social networking sites are increasingly popular, and they represent a serious risk against industrial networks. How can something as benign as Facebook or Twitter be a threat to an industrial network? By design, social networking sites make it easy to find and communicate with people, and people are subject to social engineering exploitation just as networks are subject to protocol and application exploits.

At the most basic level, they are a source of gathering personal information and end user's trust that can be exploited either directly or indirectly. At a more sophisticated level, social networks can be used actively by malware as a C2 channel. Fake accounts posing as "trusted" coworkers can lead to even more information sharing, or to trick the user into clicking on a link that will take them to a malicious website that will infect the user's laptop with malware. That malware could mine even further information, or it could be walked into a "secure" facility to impact an industrial network directly.

Although no direct evidence links the rise in web-based malware and social networking adoption, the correlation is strong enough that any good security plan should accommodate social networking, especially in industrial networks. According to Cisco, "Companies in the Pharmaceutical and Chemical vertical were the most at risk for web-based malware encounters, experiencing a heightened risk rating of 543 percent in 2Q10, up from 400 percent in 1Q10. Other higher risk verticals in 2Q10 included Energy, Oil, and Gas (446 percent), Education (157 percent), Government (148 percent), and Transportation and Shipping (146 percent)."[30]

Apart from being a direct infection vector, social networking sites can be used by more sophisticated attackers to formulate targeted spear-phishing campaigns, such as the "pizza delivery" exercise. Through no direct fault of the social network operators (most have adequate privacy controls in place), users may post personal information about where they work, what their shift is, who their boss is, and other details that can be used to engineer a social exploitation. Spear phishing is already a proven tactic; combined with the additional trust associated with social networking communities, it is easier and even more effective.

TIP

Security awareness training is an important part of building a strong security plan, but it can also be used to assess current defenses. Conduct this simple experiment to both increase awareness of spear phishing and gauge the effectiveness of existing network security and monitoring capabilities:

1. Create a website using a free hosting service that displays a security awareness banner.
2. For this exercise, create a Gmail account using the name (modified if necessary) of a group manager, HR director, or the CEO of your company (again, disclosing this activity to that individual in advance and obtaining necessary permissions). Assume the role of an attacker, with no inside knowledge of the company: look for executives who are quoted in press releases, or listed on other public documents. Alternately, use the Social Engineering Toolkit (SET), a tool designed to "perform advanced attacks against the human element," to perform a more thorough social engineering penetration test.[31]

3. Again, play the part of the attacker and use either SET or outside means such as Jigsaw.com or other business intelligence websites to build a list of e-mail addresses within the company.
4. Send an e-mail to the group from the fake "executive" account, informing recipients to please read the attached article in preparation for an upcoming meeting.
5. Perform the same experiment on a different group, using an e-mail address originating from a peer (again, obtain necessary permissions). This time, attempt to locate a pizza restaurant local to your corporate offices, using Google map searches or similar means, and send an e-mail with a link to an online coupon for buy-one-get-one-free pizza.

Track your results to see how many people clicked through to the offered URL. Did anyone validate the "from" in the e-mail, reply to it, or question it in any way? Did anyone outside of the target group click through, indicating a forwarded e-mail?

Finally, with the security monitoring tools that are currently in place, is it possible to effectively track the activity? Is it possible to determine who clicked through (without looking at web logs)? Is it possible to detect abnormal patterns or behaviors that could be used to generate signatures, and detect similar phishing in the future?

The best defense against a social attack continues to be security awareness and situational awareness: the first helps prevent a socially engineered attack from succeeding by establishing best-practice behaviors among personnel; the second helps to detect if and when a successful breach has occurred, where it originated, and where it may have spread to—in order to mitigate the damage and correct any gaps in security awareness and training.

CAUTION

Always inform appropriate personnel of any security awareness exercise to avoid unintended consequences and/or legal liability, and NEVER perform experiments of this kind using real malware. Even if performed as an exercise, the collection of actual personal or corporate information could violate your employment policy or even state, local, or federal privacy laws.

Finally, social networks can also be used as a C2 channel between deployed malware and a remote server. One case of Twitter being used to deliver commands to a bot is the @upd4t3 channel, first detected in 2009, that uses standard 140-character tweets to link to base64-encoded URLs that deliver infosteeler bots.[32]

This use of social networking is difficult to detect, as it is not feasible to scour these sites individually for such activity and there is no known way to detect what the C2 commands may look like or where they might be found. In the case of @upd4t3, application session analysis on social networking traffic could detect the base64 encoding once a session was initiated. The easiest way to block this type of activity, of course, is to block access to social networking sites completely from inside industrial networks. However, the wide adoption of these sites within the enterprise (for legitimate sales, marketing, and even business intelligence purposes) makes it highly likely that any threat originating from or directly exploiting social networks can and will compromise the business enterprise.

Cannibalistic Mutant Underground Malware

More serious than the 1984 New World Pictures film about cannibalistic humanoid underground dwellers, the newest breed of malware is a real threat. It is malware with a mind: using conditional logic to direct activity based on its surrounding until it finds itself in the perfect conditions in which it will best accomplish its goal (spread, stay hidden, deploy a weapon, etc.). Again, Stuxnet's goal was to find a particular industrial process control system: it spread widely through all types of networks, and only took secondary infection measures when the target environment (SIMATIC) was found. Then, it again checked for particular PLC models and versions, and if found it injected process code into the PLC; if not, it lay dormant.

Malware mutations are also already in use. At a basic level, Stuxnet will update itself in the wild (even without a C2 connection), through peer-to-peer checks with others of its kind: if a newer version of Stuxnet bumps into an older version, it updates the older version, allowing the infection pool to evolve and upgrade in the wild.[33]

Further mutation behavior involves self-destruction of certain code blocks with self-updates of others, effectively morphing the malware and making it more targeted as well as more difficult to detect. Mutation logic could include checking for the presence of other well-known malware and adjusting its own profile to utilize similar ports and services, knowing that this new profile will go undetected. In other words, malware is getting smarter and it is harder to detect.

Still to Come

Infection mechanisms, attack vectors, and malware payloads continue to evolve. We can expect to see greater sophistication of the individual exploits and bots, as well as more sophisticated blends of these components. Because advanced malware is expensive to develop (or acquire), however, it is reasonable to expect new variations or evolutions of existing threats in the short term, rather than additional "Stuxnet-level" revolutions. Understanding how existing exploits might be fuzzed or enhanced to avoid detection can help plan a strong defense strategy.

What we can assume is that threats will continue to grow in size, sophistication, and complexity.[34] We can also assume that new zero-day vulnerabilities will be used for one or more stages of an attack (infection, propagation, and execution). Also assume that attacks will become more focused, attempting to avoid detection through minimized exposure. Stuxnet spreads easily through many systems and only fully activates within certain environments; if a similar attack were less promiscuous and more tactically inserted into the target environment, it would be much more difficult to detect.

In early 2011, additional vulnerabilities and exploits that specifically target SCADA systems have been developed and released publically, including the broadly publicized exploits developed by two separate researchers in Italy and Russia. The "Luigi Vulnerabilities," identified by Italian researcher Luigi Auriemma included 34 total vulnerabilities against systems from Siemens, Iconics, 7-Technologies, and

DATAC.[35] Additional vulnerabilities and exploit code, including nine zero-days, was released by the Russian firm Gleg as part of the Agora+ exploit pack for the CANVAS toolkit.[36]

Luckily, many tools are already available to defend against these sophisticated attacks, and the results can be very positive when they are used appropriately in a blended, sophisticated defense based upon "Advanced Persistent Diligence."[37]

Defending Against APT

As mentioned in Chapter 2, "About Industrial Networks," the security practices that are recommended herein are aimed high, and this is because the threat environment in industrial networks has already shifted to these types of APTs, if not outright cyber war. These recommendations are built around the concept of "Advanced Persistent Diligence" and a much higher than normal level of situational awareness. This is because APT is evolving specifically to avoid detection by known security measures.[38]

Advanced Persistent Diligence requires a strong Defense-in-Depth approach, both in order to reduce the available attack surface exposed to an attacker, and in order to provide a broader perspective of threat activity for use in incident analysis, investigation, and response. That is, because APT is evolving to avoid detection even through advanced event analysis, it is necessary to examine more data about network activity and behavior from more contexts within the network.[39]

More traditional security recommendations are not enough, because the active network defense systems such as firewalls, UTMs, and IPS are no longer capable of blocking the same threats that carry with them the highest consequences. APT threats can easily slide through these legacy cyber defenses.

Having situational awareness of what is attempting to connect to the system, as well as what is going on within the system is the only way to start to regain control of the system. This includes information about systems and assets, network communication flows and behavior patterns, organizational groups, user roles, and policies. Ideally, this level analysis will be automated and will provide an active feedback loop in order to allow IT and OT security professionals to successfully mitigate a detected APT.

Responding to APT

Ironically, the last thing that you should do upon detecting an APT is to clean the system of infected malware. This is because, as mentioned under section "Advanced Persistent Threats," there may be subsequent levels of infection that exist, dormant, that will be activated as a result. Instead, a thorough investigation should be performed, with the same sophistication as the APT itself.

First, logically isolate the infected host so that it can no longer cause any harm. Allow the APT to communicate over established C2 channels, but isolate the host from the rest of your network, and remove all access between that host and any sensitive or protected information. Collect as much forensic detail as possible in the form of system logs, captured network traffic, and supplement where possible with memory analysis data. By effectively sandboxing the infected system, important information can be gathered that can result in the successful removal of an APT.

In summary, when you suspect that you are dealing with an APT, approach the situation with diligence and perform a thorough investigation:

- Always monitor everything: collect baseline data, configurations, and firmware for comparison.
- Analyze available logs to help identify scope, infected hosts, propagation vectors, etc.
- Sandbox and investigate infected systems.
- Analyze memory to find memory-resident rootkits and other threats living in user memory.
- Reverse engineer-detected malware to determine full scope and to identify additional attack vectors and possible prorogation.
- Retain all information for disclosure to authorities.

NOTE

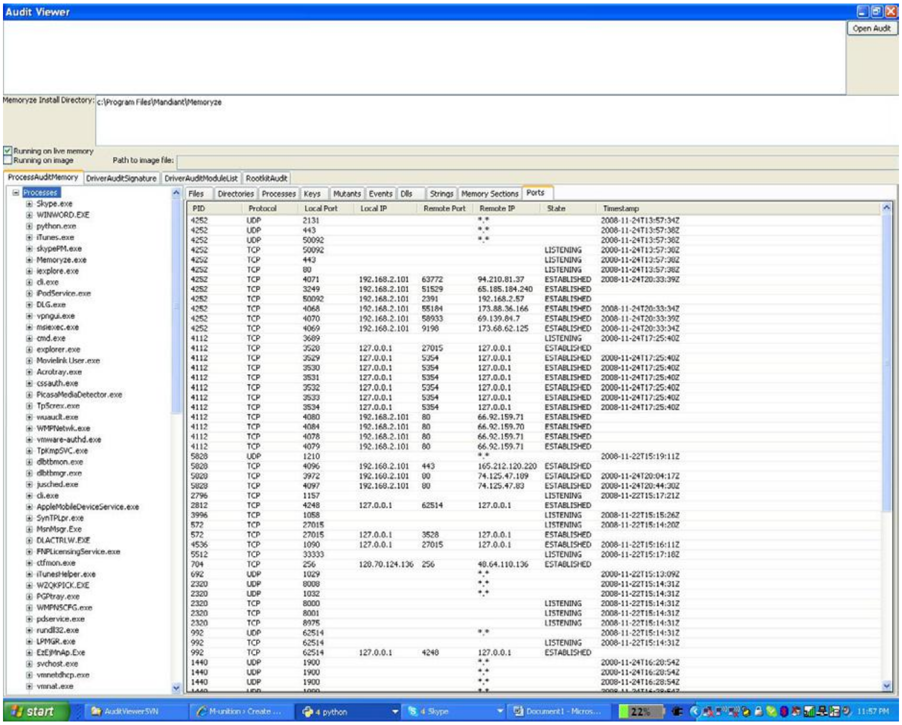
Information collected from an infected and sandboxed host may prove valuable to legal authorities, and depending upon the nature of your industrial network you may be required to report this information to a governing body.

Depending on the severity of the APT, a “bare metal reload” may be necessary, where a device is completely erased and reduced to a bare, inoperable state. The host’s hardware must then be reimaged completely. For this reason, clean versions of operating systems and/or asset firmware should be kept in a safe, clean environment. This can be accomplished using secure virtual backup environments, or via secure storage on trusted removable media that can then be stored in a locked cabinet.

Free tools such as Mandiant’s Memoryze, shown in Figure 3.5, can help you to perform a deep forensic analysis on infected systems. This can help to determine how deeply infected a system might be, by detecting memory-resident rootkits. Memoryze and other forensics tools are available at http://www.mandiant.com.

TIP

If you think you have an APT, you should know that there are security firms that are experienced in investigating and cleaning APT. Many such firms further specialize in industrial control networks. These firms can help you deal with infection as well as provide an expert interface between your organization and any governing authorities that may be involved.



Memoryze Install Directory: c:\Program Files\Mandiant\Memoryze

☑ Running on live memory
□ Running on image
Path to image file:

Processes
- Skype.exe
- WINWORD.EXE
- python.exe
- iTunes.exe
- skypePM.exe
- Memoryze.exe
- iexplore.exe
- cli.exe
- iPodService.exe
- DLG.exe
- vpngui.exe
- msiexec.exe
- cmd.exe
- explorer.exe
- Movielink User.exe
- Acrotray.exe
- cssauth.exe
- PicasaMediaDetector.exe
- TpScrex.exe
- wuauclt.exe
- WMPNetwk.exe
- vmware-authd.exe
- TpKmpSVC.exe
- dlbtbmon.exe
- dlbtbmgr.exe
- jusched.exe
- cli.exe
- AppleMobileDeviceService.exe
- SynTPLpr.exe
- MsnMsgr.Exe
- DLACTRLW.EXE
- FNPLicensingService.exe
- ctfmon.exe
- iTunesHelper.exe
- WZQKPICK.EXE
- PGPtray.exe
- WMPNSCFG.exe
- pdservice.exe
- rundll32.exe
- LPMGR.exe
- EzEjMnAp.Exe
- svchost.exe
- vmnetdhcp.exe
- vmnat.exe

PID	Protocol	Local Port	Local IP	Remote Port	Remote IP	State	Timestamp
4252	UDP	2131			*.*		2008-11-24T13:57:34Z
4252	UDP	443			*.*		2008-11-24T13:57:38Z
4252	UDP	50092			*.*		2008-11-24T13:57:38Z
4252	TCP	50092				LISTENING	2008-11-24T13:57:38Z
4252	TCP	443				LISTENING	2008-11-24T13:57:38Z
4252	TCP	80				LISTENING	2008-11-24T13:57:38Z
4252	TCP	4071	192.168.2.101	63772	94.210.81.37	ESTABLISHED	2008-11-24T20:33:39Z
4252	TCP	3249	192.168.2.101	51529	65.185.184.240	ESTABLISHED	
4252	TCP	50092	192.168.2.101	2391	192.168.2.57	ESTABLISHED	
4252	TCP	4068	192.168.2.101	55184	173.88.36.166	ESTABLISHED	2008-11-24T20:33:34Z
4252	TCP	4070	192.168.2.101	58933	69.139.84.7	ESTABLISHED	2008-11-24T20:33:39Z
4252	TCP	4069	192.168.2.101	9198	173.68.62.125	ESTABLISHED	2008-11-24T20:33:34Z
4112	TCP	3689				LISTENING	2008-11-24T17:25:40Z
4112	TCP	3520	127.0.0.1	27015	127.0.0.1	ESTABLISHED	
4112	TCP	3529	127.0.0.1	5354	127.0.0.1	ESTABLISHED	2008-11-24T17:25:40Z
4112	TCP	3530	127.0.0.1	5354	127.0.0.1	ESTABLISHED	2008-11-24T17:25:40Z
4112	TCP	3531	127.0.0.1	5354	127.0.0.1	ESTABLISHED	2008-11-24T17:25:40Z
4112	TCP	3532	127.0.0.1	5354	127.0.0.1	ESTABLISHED	2008-11-24T17:25:40Z
4112	TCP	3533	127.0.0.1	5354	127.0.0.1	ESTABLISHED	2008-11-24T17:25:40Z
4112	TCP	3534	127.0.0.1	5354	127.0.0.1	ESTABLISHED	2008-11-24T17:25:40Z
4112	TCP	4080	192.168.2.101	80	66.92.159.71	ESTABLISHED	
4112	TCP	4084	192.168.2.101	80	66.92.159.70	ESTABLISHED	
4112	TCP	4078	192.168.2.101	80	66.92.159.71	ESTABLISHED	
4112	TCP	4079	192.168.2.101	80	66.92.159.71	ESTABLISHED	
5828	UDP	1210			*.*		2008-11-22T15:19:11Z
5828	TCP	4096	192.168.2.101	443	165.212.120.220	ESTABLISHED	
5828	TCP	3972	192.168.2.101	80	74.125.47.109	ESTABLISHED	2008-11-24T20:04:17Z
5828	TCP	4097	192.168.2.101	80	74.125.47.83	ESTABLISHED	2008-11-24T20:44:30Z
2796	TCP	1157				LISTENING	2008-11-22T15:17:21Z
2812	TCP	4248	127.0.0.1	62514	127.0.0.1	ESTABLISHED	
3996	TCP	1058				LISTENING	2008-11-22T15:15:26Z
572	TCP	27015				LISTENING	2008-11-22T15:14:20Z
572	TCP	27015	127.0.0.1	3528	127.0.0.1	ESTABLISHED	
4536	TCP	1090	127.0.0.1	27015	127.0.0.1	ESTABLISHED	2008-11-22T15:16:11Z
5512	TCP	33333				LISTENING	2008-11-22T15:17:18Z
704	TCP	256	128.70.124.136	256	48.64.110.136	ESTABLISHED	
692	UDP	1029			*.*		2008-11-22T15:13:09Z
2320	UDP	8008			*.*		2008-11-22T15:14:31Z
2320	UDP	1032			*.*		2008-11-22T15:14:31Z
2320	TCP	8000				LISTENING	2008-11-22T15:14:31Z
2320	TCP	8001				LISTENING	2008-11-22T15:14:31Z
2320	TCP	8975				LISTENING	2008-11-22T15:14:31Z
992	UDP	62514			*.*		2008-11-22T15:14:31Z
992	TCP	62514				LISTENING	2008-11-22T15:14:31Z
992	TCP	62514	127.0.0.1	4248	127.0.0.1	ESTABLISHED	
1440	UDP	1900			*.*		2008-11-24T16:28:54Z
1440	UDP	1900			*.*		2008-11-24T16:28:54Z
1440	UDP	1900			*.*		2008-11-24T16:28:54Z

FIGURE 3.5

Mandiant's Memoryze: A Memory Forensic Package.[40]

SUMMARY

Industrial networks are important and vulnerable, and there are potentially devastating consequences of a cyber incident. Examples of real cyber incidents—from CENTCOM to Aurora to Stuxnet—have grown progressively more severe over time, highlighting the evolving nature of threats against industrial systems. The attacks are evolving into APTs, and the intentions are evolving from information theft to industrial sabotage and the disruption of critical infrastructures.

Securing industrial networks requires a reassessment of our security practices, realigning them to a better understanding of how industrial protocols and networks operate (see Chapter 4, "Industrial Network Protocols," and Chapter 5 "How Industrial Networks Operate"), as well as a better understanding of the vulnerabilities and threats that exist (see Chapter 6, "Vulnerability and Risk Assessment").

ENDNOTES

1. J. Pollet, Red Tiger, Electricity for free? The dirty underbelly of SCADA and smart meters, in: Proc. 2010 BlackHat Technical Conference, Las Vegas, NV, July 2010.
2. Ibid.
3. M.J. McDonald, G.N. Conrad, T.C. Service, R.H. Cassidy, SANDIA Report SAND2008-5954, Cyber Effects Analysis Using VCSE Promoting Control System Reliability, Sandia National Laboratories Albuquerque, New Mexico and Livermore, California, September 2008.
4. A. Giani, S. Sastry, K.H. Johansson, H.Sandberg, The VIKING Project: An Initiative on Resilient Control of Power Networks, Department of Electrical Engineering and Computer Sciences, University of California at Berkeley, and School of Electrical Engineering, Royal Institute of Technology (KTH), Berkeley, CA, 2009.
5. K. Stouffer, J. Falco, K. Scarfone, National Institute of Standards and Technology, Special Publication 800-82 (Final Public Draft), Guide to Industrial Control Systems (ICS) Security, Computer Security Division, Information Technology Laboratory, National Institute of Standards and Technology Gaithersburg, MD and Intelligent Systems Division, Manufacturing Engineering Laboratory, National Institute of Standards and Technology Gaithersburg, MD, September 2008.
6. T. Smith, The Register. Hacker jailed for revenge sewage attacks. <http://www.theregister.co.uk/2001/10/31/hacker_jailed_for_revenge_sewage/>, October 31, 2001 (cited: November 3, 2010).
7. J. Meserve, CNN.com. Sources: Staged cyber attack reveals vulnerability in power grid. <http://articles.cnn.com/2007-09-26/us/power.at.risk_1_generator-cyber-attack-electric-infrastructure>, September 26, 2007 (cited: November 3, 2010).
8. North American Reliability Corporation, Press Release: NERC Issues AURORA Alert to Industry, October 14, 2010.
9. CBS News, Cyber war: sabotaging the system. <http://www.cbsnews.com/stories/2009/11/06/60minutes/main5555565.shtml>, November 8, 2009 (cited: November 3, 2010).
10. McAfee Threat Center, Operation Aurora. <http://www.mcafee.com/us/threat_center/operation_aurora.html> (cited: November 4, 2010).
11. Industrial Control Systems Cyber Emergency Response Team (ICS-CERT), ICSA-10-238-01—STUXNET MALWARE MITIGATION, Department of Homeland Security, US-CERT, Washington, DC, August 26, 2010.
12. E. Chien, Symantec. Stuxnet: a breakthrough. <http://www.symantec.com/connect/blogs/stuxnet-breakthrough>, November 2010 (cited: November 16, 2010).
13. N. Falliere, L.O. Murchu, E. Chien, Symantec, W32.Stuxnet Dossier, Version 1.3, October 2010.
14. N. Falliere, L.O Murchu, E. Chien, Symantec. W32.Stuxnet Dossier, Version 1.1, October 2010.
15. Global Energy Cyberattacks: "Night Dragon," McAfee Foundstone Professional Services and McAfee Labs, Santa Clara, CA, February 10, 2011.
16. Ibid.
17. J. Pollet, Red Tiger, Understanding the advanced persistent threat, in: Proc. 2010 SANS European SCADA and Process Control Security Summit, Stockholm, Sweden, October 2010.
18. North American Reliability Corporation, NERC Releases Alert on Malware Targeting SCADA Systems, September 14, 2010.
19. J. Pollet, Red Tiger, Understanding the advanced persistent threat, in: Proc. 2010 SANS European SCADA and Process Control Security Summit, Stockholm, Sweden, October 2010.

20. K. Harms, Mandiant, Keynote on advanced persistent threat, in: Proc. 2010 SCADA Security Scientific Symposium (S4), Miami, FL, January 2010.
21. J. Pollet, Red Tiger, Understanding the advanced persistent threat, in: Proc. 2010 SANS European SCADA and Process Control Security Summit, Stockholm, Sweden, October 2010.
22. Ibid.
23. Ibid.
24. N. Falliere, L.O. Murchu, E. Chien, Symantec. W32.Stuxnet Dossier, Version 1.1, October 2010.
25. J. Pollet, Red Tiger, Understanding the advanced persistent threat, in: Proc. 2010 SANS European SCADA and Process Control Security Summit, Stockholm, Sweden, October 2010.
26. Ibid.
27. D. Stevens, Escape from PDF. <http://blog.didierstevens.com/2010/03/29/escape-from-pdf>, March 2010 (cited: November 4, 2010).
28. M86 Security Labs, PDF "Launch" Feature Used to Install Zeus. <http://www.m86security.com/labs/traceitem.asp?article=1301>, April 14, 2010 (cited: November 4, 2010).
29. J. Conway, Sudosecure.net. Worm-Able PDF Clarification. <http://www.sudosecure.net/archives/644>, April 4, 2010 (cited: November 4, 2010).
30. Cisco Systems, 2Q10 Global Threat Report, 2010.
31. Social Engineering Framework, Computer based social engineering tools: Social Engineer Toolkit (SET). <http://www.social-engineer.org>.
32. J. Nazario, Arbor networks. Twitter-based Botnet Command Channel. <http://asert.arbornetworks.com/2009/08/twitter-based-botnet-command-channel>, August 13, 2009 (cited: November 4, 2010).
33. J. Pollet, Red Tiger, Understanding the advanced persistent threat, in: Proc. 2010 SANS European SCADA and Process Control Security Summit, Stockholm, Sweden, October 2010.
34. Ibid.
35. D. Peterson, Italian researcher publishes 34 ICS vulnerabilities. Digital Bond. <http://www.digitalbond.com/2011/03/21/italian-researcher-publishes-34-ics-vulnerabilities/>, March 21, 2011 (cited: April 4, 2011).
36. D. Peterson, Friday News and Notes. <http://www.digitalbond.com/2011/03/25/friday-news-and-notes-127>, March 25, 2011 (cited: April 4, 2011).
37. Ibid.
38. Ibid.
39. US Department of Homeland Security, US-CERT, Recommended Practice: Improving Industrial Control Systems Cybersecurity with Defense-In-Depth Strategies, Washington, DC, October 2009.
40. Screenshot, Mandiant's Memoryze memory analysis software. <http://www.mandiant.com> (cited: October 26, 2010).

CHAPTER

Industrial Network Protocols 4

INFORMATION IN THIS CHAPTER:

- Overview of Industrial Network Protocols
- Modbus
- ICCP/TASE.2
- DNP3
- OLE for Process Control
- Other Industrial Network Protocols
- AMI and the Smart Grid

Understanding how industrial networks operate requires a basic understanding of the underlying communications protocols that are used, where they are used, and why. There are many highly specialized protocols used for industrial automation and control, most of which are designed for efficiency and reliability to support the economic and operational requirements of large distributed control systems. Similarly, most industrial protocols are designed for real-time operation to support precision operations.

Unfortunately, this means that most industrial protocols forgo any feature or function that is not absolutely necessary, for the sake of efficiency. Even more unfortunate is that this often includes security features such as authentication and encryption, both of which require additional overhead. To further complicate matters, many of these protocols have been modified to run over Ethernet and Internet Protocol (IP) networks in order to meet the evolving needs of business, potentially exposing these vulnerable protocols to attack.

OVERVIEW OF INDUSTRIAL NETWORK PROTOCOLS

Industrial Network Protocols are often referred to generically as SCADA and/or fieldbus protocols. SCADA protocols are primarily used for the communication of supervisory systems, whereas fieldbus protocols are used for the communication of industrial, automated control systems (ICS or IACS). However, most of the protocols discussed have the ability to perform both functions, and so will be referred to here more generically as Industrial Network Protocols.

Industrial Network Protocols are real-time communications protocols, developed to interconnect the systems, interfaces, and instruments that make up an industrial control system. Most were designed initially to communicate serially over RS-232, RS-485, or other serial connections but have since evolved to operate over Ethernet networks using routable protocols such as TCP/IP.

Four common industrial network protocols will be discussed in some depth, others will be touched upon more briefly, and many will not be covered here: there are literally dozens of industrial protocols, developed by specific manufacturers for specific purposes. Modicon Communication Bus (Modbus), **Inter Control Center Protocol** (ICCP, also known as **TASE.2** or **Telecontrol Application Service Element-2**), Distributed Network Protocol (DNP3), and Object Linking and Embedding for Process Control (OPC) have been selected for more in-depth discussion because they are all widely deployed and they represent several unique qualities that are important to understand within the context of security. These unique qualities include the following:

- Each is used in different (though sometimes overlapping) areas within an industrial network.
- Each provides different methods of verifying data integrity and/or security.
- The specialized requirements of industrial protocols (e.g., real-time, synchronous communication) often make them highly susceptible to disruption.

By understanding the basic principles of how to secure these protocols, it should be possible to assess the risks of other industrial network protocols that are not covered here directly.

MODBUS

Modbus is the oldest and perhaps the most widely deployed industrial control communications protocol. It was designed in 1979 by Modicon (now part of Schneider Electric) that invented the first Programmable Logic Controller (PLC). Modbus has been widely adopted as a de facto standard and has been enhanced over the years into several distinct variants.

Modbus' success stems from its relative ease of use: it communicates raw messages without restrictions of authentication or excessive overhead. It is also an open standard, is freely distributed, and is widely supported by members of the Modbus Organization, which still operates today.

What It Does

Modbus is an application layer messaging protocol, meaning that it operates at layer 7 of the OSI model. It allows for efficient communications between interconnected assets based on a request/reply methodology. It can be used by extremely simple devices such as sensors or motors to communicate with a more complex computer, which can read measurements and perform analysis and control.

To support a communications protocol on a simple device requires that the message generation, transmission, and receipt all require very little processing overhead. This same quality also makes Modbus suitable for use by PLCs and RTUs to communicate supervisory data to a SCADA system.

Because Modbus is a layer 7 protocol, it can operate independently of underlying network protocols, and it has allowed Modbus to be easily adapted to both serial and routable network architectures.

How It Works

Modbus is a request/response protocol using only three distinct Protocol Data Units (PDUs): Modbus Request, Modbus Response, and Modbus Exception Response, as illustrated in Figure 4.1.[1]

Each device communicating via Modbus must be assigned a unique address. A command is addressed to a specific Modbus address, and while other devices may receive the message, only the addressed device will respond.

A session begins with the transmission of an initial **Function Code** and a Data Request within a Request PDU. The receiving device responds in one of two ways. If there are no errors, it will respond with a Function Code and Data Response within a Response PDU. If there are errors, the device will respond with an Exception Function Code and Exception Code within a Modbus Exception Response.

Function Codes and Data Requests can be used to perform a wide range of commands. Some examples of Modbus commands include the following:

- Control an I/O interface.
- Read from an I/O interface.

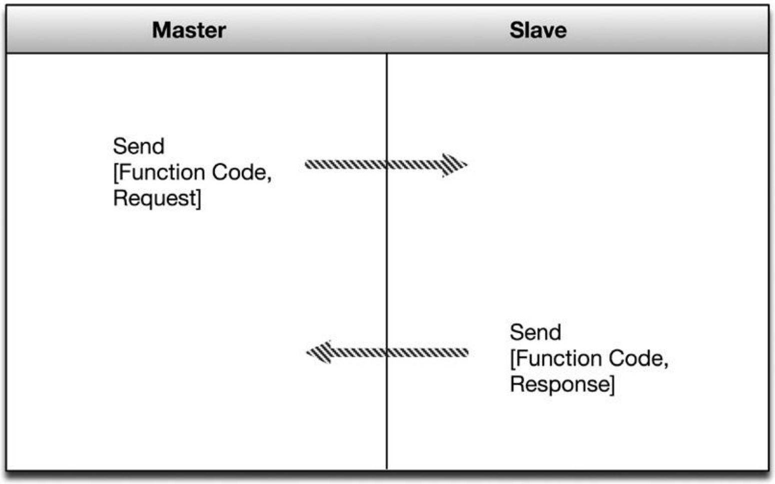


FIGURE 4.1

Modbus Protocol Operation.

- Read the value of a register.
- Write a value to a register (i.e., change the value in a register).

Variants

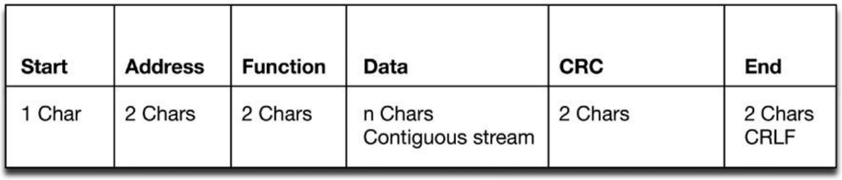
The popularity of Modbus has led to the development of several variations to suit particular needs. These include Modbus RTU and **Modbus ASCII**, which support binary and ASCII transmissions over serial buses, respectively. Modbus TCP is a variant of Modbus developed to operate on modern networks using the IP. **Modbus Plus** is a variant designed to extend the reach of Modbus via interconnected busses.[2]

Modbus RTU and Modbus ASCII

These similar variants of Modbus are used in serial communications, and they are the simplest of the variants. Modbus RTU (Figure 4.2) uses binary data representation, whereas Modbus ASCII (Figure 4.3) uses ASCII characters to represent data. Each uses a simple message format carried within a PDU, consisting of an address, function code, a payload of data, and a checksum, to ensure the message was received correctly.

Modbus TCP

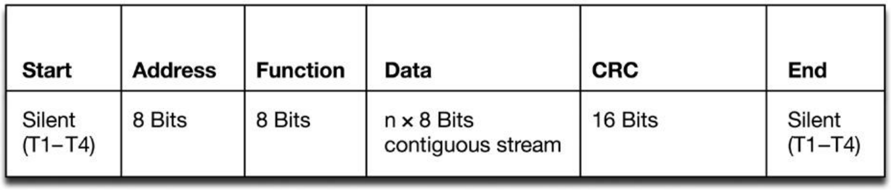
Modbus TCP uses Transmission Control Protocol/Internet Protocol (TCP/IP) to transport Modbus commands and messages over modern routable networks. Early implementations of Modbus TCP abandoned the Modbus checksum, relying on the



Start	Address	Function	Data	CRC	End
1 Char	2 Chars	2 Chars	n Chars Contiguous stream	2 Chars	2 Chars CRLF

FIGURE 4.2

Modbus Frame (Modbus RTU).



Start	Address	Function	Data	CRC	End
Silent (T1–T4)	8 Bits	8 Bits	n × 8 Bits contiguous stream	16 Bits	Silent (T1–T4)

FIGURE 4.3

Modbus Frame (Modbus ASCII).

checksum of the underlying TCP/IP protocol, whereas most current Modbus TCP implementations include the original Modbus checksum within the TCP/IP payload, as shown in Figure 4.4.

Modbus Plus or Modbus+

Modbus Plus is proprietary to Modicon, which sends embedded Modbus messages over an RS-485 communication link. Modbus Plus supports some interesting features, including Modbus bridging to allow multiple buses to interconnect, extending the number of supported nodes indefinitely.

Where It Is Used

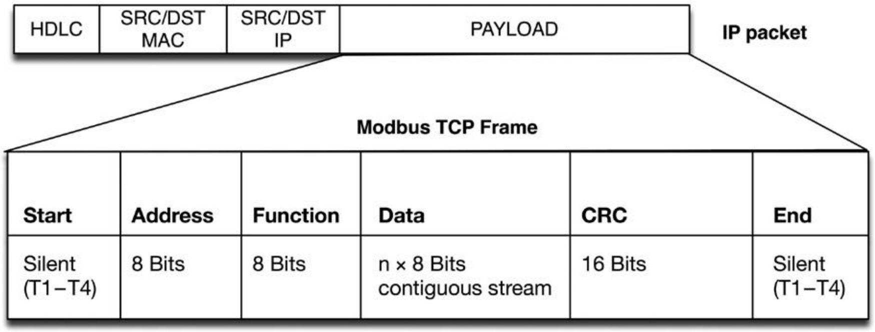
Modbus is typically deployed between PLCs and HMIs, or between a Master PLC and slave devices such as PLCs, HMIs, Drivers, Sensors, I/O devices, etc., as shown in Figure 4.5. Typically up to 247 devices are supported in a single, non-bridged bus.

A common deployment uses Modbus on TCP/IP within a SCADA DMZ or Supervisory LAN, where master HMIs provide a central management capability to a number of Master PLCs, each of which may connect serially over a bus topology to a number of PLCs and/or HMIs, responsible for a distinct control loop.

Security Concerns

Modbus represents several security concerns:

- Lack of authentication. Modbus sessions only require the use of a valid Modbus address and valid function code. One can be easily guessed or spammed, whereas the other is easily obtainable information.
- Lack of encryption. Commands and addresses are transmitted in clear text and can therefore be easily captured and spoofed due to the lack of encryption.



Start	Address	Function	Data	CRC	End
Silent (T1–T4)	8 Bits	8 Bits	n × 8 Bits contiguous stream	16 Bits	Silent (T1–T4)

FIGURE 4.4

Modbus Frame (Modbus TCP).

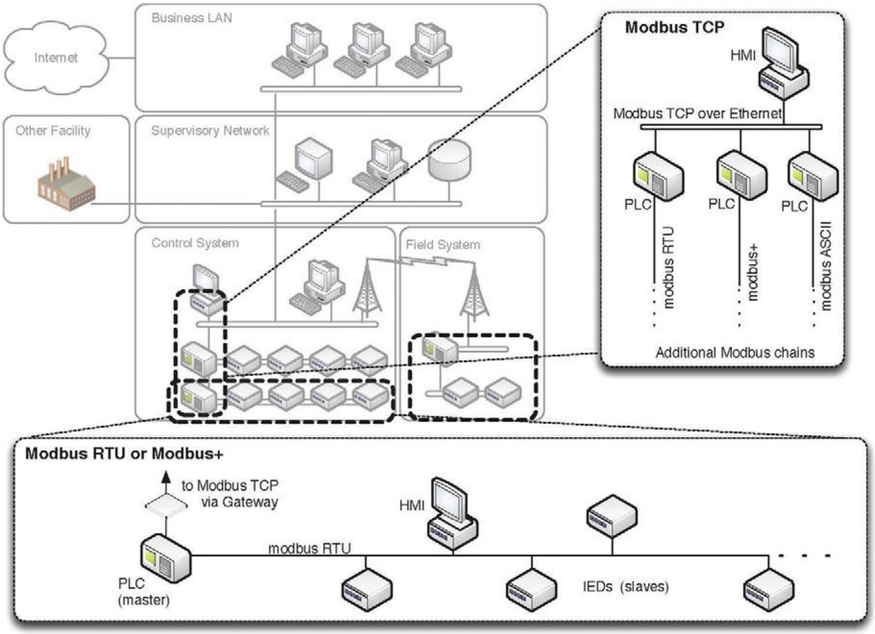


FIGURE 4.5

Typical Modbus Use within the Industrial Network Architecture.

- Lack of message checksum (Modbus TCP only). A spoofed command is even easier over some implementations of Modbus TCP, as the checksum is generated at the transmission layer, not the application layer.
- Lack of broadcast suppression (serial Modbus variants only). All serially connected devices will receive all messages, meaning a broadcast of unknown addresses can be used for effective denial of service (DoS) to a chain of serially connected devices.
- Programmability. By far, the most dangerous quality of Modbus—which is shared with many industrial protocols—is that it is intentionally designed to program controllers, and could be used to inject malicious logic into an RTU or PLC.

Security Recommendations

Modbus, like many industrial control protocols, should only be used to communicate between sets of known devices, using expected function codes, and as such it is easily monitored by establishing clear enclaves and by baselining acceptable behavior. For more information about creating **whitelists**, this topic is discussed in detail in Chapter 8, “Exception, Anomaly, and Threat Detection.”

Some specific examples of Modbus messages that should be of concern include the following:

- Modbus TCP packets that are of wrong size or length.
- Function codes that force slave devices into a “listen only” mode.
- Function codes that restart communications.
- Function codes that clear, erase, or reset diagnostic information such as counters and diagnostic registers.
- Function codes that request information about Modbus servers, PLC configurations, or other need-to-know information.
- Traffic on TCP port 502 that is not Modbus or is using Modbus over malformed protocol(s).
- Any message within an Exception PDU (i.e., any Exception Code).
- Modbus traffic from a server to many slaves (i.e., a potential DoS).
- Modbus requests for lists of defined points and their values (i.e., a configuration scan).
- Commands to list all available function codes (i.e., a function scan).

A SCADA Intrusion Detection System (**SCADA-IDS**) or SCADA Intrusion Prevention System (**SCADA-IPS**) can be easily configured to monitor for these activities using Modbus signatures such as those developed and distributed by Digital Bond. In more critical areas, an application-aware firewall, industrial protocol filter, or **Application Data Monitor** may be required to validate Modbus sessions and ensure that Modbus has not been “hijacked” and used for covert communication, command, and control (i.e., the underlying TCP/IP session on port 502 has not been altered to hide additional communications channels within otherwise normal-looking Modbus traffic). This is discussed in detail in Chapter 7, “Establishing Secure Enclaves.”

CAUTION

SCADA-IPS devices are able to actively block suspect traffic by dropping packets or resetting TCP connections. Any SCADA-IPS devices should be configured to alert on events, rather than block (i.e., operate in IDS mode rather than active IPS mode), as a false positive that blocks a Modbus command could cause an unintentional failure within the control system.

ICCP/TASE.2

The Inter Control Center Protocol (also known as TASE.2 or IEC60870-6, but more commonly referred to as ICCP) is a protocol designed for communication between **control centers** within the energy industry. Unlike Modbus, which was designed for serial command requests, ICCP was designed for bidirectional Wide Area Network (WAN) communication between a utility control center and other control centers, power plants, substations, and even other utilities.

Because many custom and proprietary protocols are used by asset vendors, a common protocol was needed to allow for reliable and standardized data exchange between control centers—especially between control centers that are operated by different owners, produce different products, or perform different operations. Basically, standardization became necessary to support the unique business and operational requirements of industry, especially the electrical utilities that require careful load balancing within a bulk system operated by many disparate facilities. For example, in North America, the division of utilities among several responsible regional entities requires a means of sharing information between utilities as well as the regional entity. Similarly, national and global energy markets require real-time information exchange for load distribution and trading that spans the boundaries of individual utilities.

A working group was formed in 1991 to develop and test a standardized protocol and to submit the specification to the IEC. The initial protocol was called ELCOM-90, or **Telecontrol Application Service Element-1** (**TASE.1**). TASE.1 evolved into TASE.2, which is the most commonly used form of ICCP.[3]

What It Does

ICCP is used to perform a number of communication functions between control centers, including the following:

- Establishing a connection.
- Accessing information (read requests).
- Information transmission (such as e-mail messages or energy market information).
- Notifications of changes, alarms, or other exception conditions.
- Configuration of remote devices.
- Control of remote devices.
- Control of operating programs.

How It Works

The ICCP protocol defines communication between two control centers using a client/server model. One control center (the server) contains application data and defined functions. Another control center (the client) issues requests to read from the server, and the server responds. Communications over ICCP occur using a common format in order to ensure interoperability.

ICCP support is typically either integrated directly into a control system, provided via a gateway product, or provided as software running on Windows or Unix that can then be installed to perform gateway functions.

Although ICCP is primarily a unidirectional client/server protocol, most modern implementations support both functions, allowing a single ICCP device to function as both a client and a server, and thus supporting bidirectional communication (of a sort) over a single connection.

Although ICCP can operate over essentially any network protocol, including TCP/IP, it is commonly implemented using ISO transport on top of TCP port 102, as

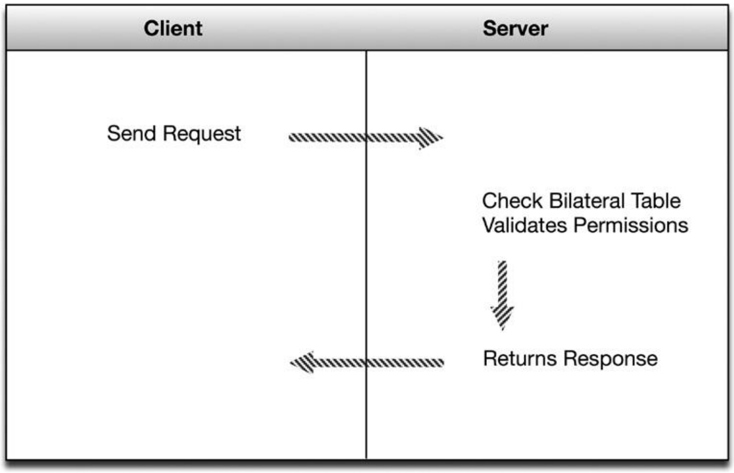


FIGURE 4.6

ICCP Protocol Operation.

defined in RFC 1006. ICCP is effectively a point-to-point protocol due to the use of a "bilateral table" that explicitly defines an agreement between two control centers connected with an ICCP link, as shown in Figure 4.6. The bilateral table is essentially an access control list that identifies which data elements a client can access. The permissions defined within the bilateral tables in the server and the client are the authoritative control over what is accessible to each control center. In addition, the entries in the bilateral tables must match on both the client and the server, ensuring that the permissions are agreed upon by both centers (remembering that ICCP is used to interconnect to other organizations in addition to internal WAN links to substations).[4]

Where It Is Used

ICCP is widely used between control system enclaves and between distinct control centers, as shown in Figure 4.7, for example, between two electric utilities, between two control systems within a single electric utility, between a main control center and a number of substations, etc.

Security Concerns

Like Modbus, ICCP represents several security concerns. ICCP is susceptible to spoofing, session hijacking, and any number of attacks made possible because of:

- Lack of authentication and encryption. ICCP does not mandate authentication or encryption, most often deferring these services to lower protocol layers. Although Secure ICCP does exist, it is not ubiquitously deployed.

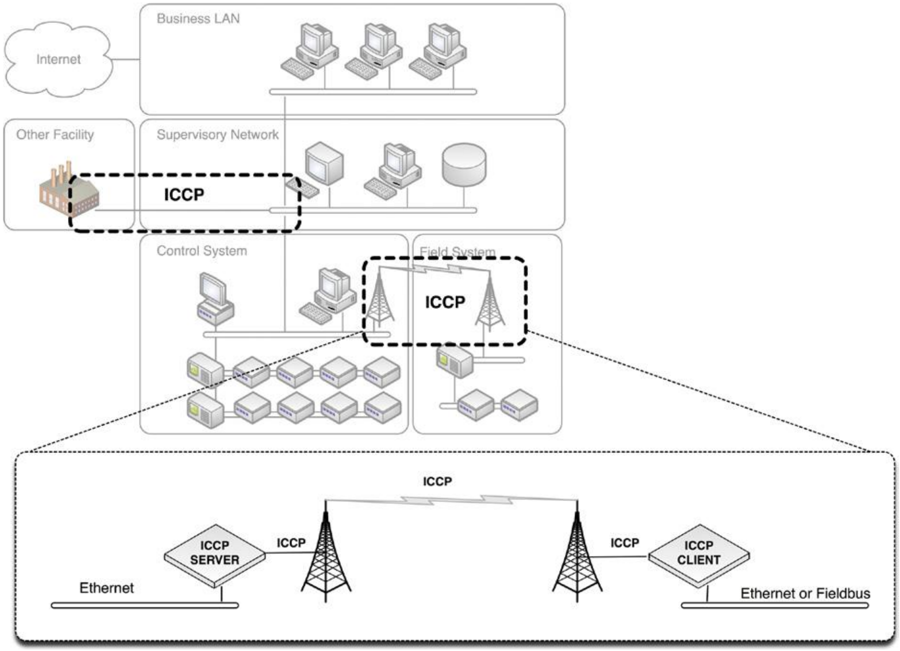


FIGURE 4.7

Typical ICCP Use within the Industrial Network Architecture.

- Explicitly defined trust relationships. The exploitation of bilateral tables could directly compromise security of ICCP servers and clients.
- Accessibility. ICCP is a Wide Area Protocol making it highly accessible and susceptible to many attacks including DoS attacks.

The limited security mechanisms within ICCP are configured on the ICCP **Master station**, meaning that the successful breach of the Master through a Man-in-the-Middle (MITM) or other attack opens the entire communication session up to manipulation.

Security Improvements over Modbus

ICCP offers several improvements over more basic fieldbus protocols such as Modbus, including the following:

- ICCP's use of bilateral tables provides basic control over the communication path by explicitly defining which ICCP clients and servers can communicate.
- A secure version of ICCP exists that incorporates digital certificate authentication and encryption.

Security Recommendations

Secure ICCP variants should be used wherever possible. There are several known vulnerabilities with ICCP that are reported by ICS-CERT. Because there are known exploits in the wild and ICCP is a WAN protocol, proper penetration testing and patching of ICCP servers and clients is recommended.

Extreme care should be taken in the definition of the bilateral table. The bilateral table is the primary enforcement of policy and permissions between control centers and malicious commands issued via ICCP could directly alter or otherwise impact control center operations.

In addition, ICCP clients and servers should be isolated into a unique enclave consisting only of authorized client/server pairs (multiple enclaves can be defined for devices communicating to multiple clients), and the enclave(s) should be thoroughly secured using standard defense-in-depth practices, including a firewall and/or IDS system that enforces strict control over the type, source, and destination of traffic over the ICCP link.

Many malicious behaviors can be detected through monitoring the ICCP link, including the following:

- Intruders gaining unauthorized access to the control center network, via overlooked access points such as dial-up connections to partner or vendor networks.
- Insider threats, including unauthorized information access and transmission, alteration of secure configurations, or other malicious actions can be the result of a physical security breach within a control center, or of a disgruntled employee.
- A DoS attack resulting from repeated information requests ("spamming") that utilize the server's available resources and prevent legitimate operation of the ICCP link.
- Malware infecting the ICCP server or other devices could be used to exfiltrate sensitive information for purposes of sabotage (e.g., theft of command function codes), financial disruption (e.g., alteration of energy metrics used in trading), or various other malicious intents.
- Interception and modification of ICCP messages (i.e., MITM) attacks.

Monitoring of ICCP protocol functions can also detect suspicious or malicious behavior, such as

- Function "read" codes that could be used to exfiltrate protected information.
- Function "write" codes that could be used to manipulate client or server operations.
- Traffic on TCP port 102 that is not ICCP.
- ICCP traffic that is not sourced by and destined to defined ICCP servers or clients.

A SCADA-IDS or SCADA-IPS can be easily configured to monitor for these activities. Digital Bond, a security research team funded by the Department of Energy, has Snort compatible IDS signatures and preprocessors to detect a variety of ICCP-related security events. Again, in more critical areas, an application-aware firewall, industrial protocol filter, or Application Data Monitor may be required to validate ICCP sessions and ensure that ICCP or the underlying RFC-1006 connection have not been "hijacked" and that messages have not been manipulated or falsified.

CAUTION

Any SCADA-IPS devices should be configured with caution and should only alert on events, rather than block them (i.e., operate in IDS mode rather than active IPS mode), as a false positive that blocks an ICCP command could cause an unintentional failure within the control system.

DNP3

DNP3 began as a serial protocol designed for use between master control stations and slave devices or "**outstations**," as well as for use between RTUs and **IEDs** within a control station. Like most control system protocols, DNP3 was extended to work over IP, encapsulated in TCP or UDP packets, in order to make remote RTU communications more easily accessible over modern networks.

One distinction of DNP3 is that it is very reliable, while remaining efficient and well suited for real-time data transfer. It also utilizes several standardized data formats and supports time-stamped (and time-synchronized) data, making real-time transmissions more efficient and thus even more reliable. Another reason that DNP3 is considered highly reliable is due to the frequent use of CRC checks—a single DNP3 frame can include up to 17 CRCs: one in the header and one per data block within the payload (see the section "How it Works"). There are also optional link-layer acknowledgments for further reliability assurance, and—of particular note—variations of DNP3 that support link-layer authentication as well. Because all of this is done within the link-layer frame, it means that additional network-layer checks may also apply if DNP3 is encapsulated for transport over Ethernet.

Unlike Modbus and ICCP, DNP3 is both bidirectional (supporting communications from both Master to Slave and from Slave to Master) and supports exception-based reporting. It is therefore possible for a DNP3 outstation to initiate an unsolicited response, in order to notify the Master of an event outside of the normal polling interval (such as an alarm condition).

What It Does

Like the other ICS protocols that have been discussed, DNP3 is primarily used to send and receive messages between control system devices—only in the case of DNP3, it also does it with a high degree of reliability. Assuming that the various CRCs are all valid, the data payload is then processed. Again, the payload is very flexible and can be used to simply transfer informational readings, or it can be used to send control functions, or even direct binary or analog data for direct interaction with devices such as Remote Terminal Units (RTUs), as well as other analog devices such as IEDs.

As mentioned previously, both the link-layer frame (or LPDU) header and the data payload contain CRCs, and the data payload actually contains a pair of CRC octets for every 16 data octets. This provides an exceptional degree of assurance that any communication errors will be detected. If any errors are detected, DNP3 will retransmit the errored frames. In addition to frame integrity, there are also physical layer integrity issues, and it remains possible that a correctly formed and transmitted frame will not arrive at its destination. To overcome this risk, DNP3 uses an additional link layer confirmation. When link layer confirmation is enabled, the DNP3 transmitter (source) of the frame requests that the receiver (destination) confirms the successful receipt of the frame. If a requested confirmation is not received, the link layer will retransmit the frame. This confirmation is optional because although it increases reliability, it adds overhead that directly impacts the efficiency of the protocol. In real-time environments, this added overhead may not be appropriate.[5]

Once a successful and (if requested) confirmed frame arrives, the frame is processed. Each frame consists of a multipart header and a data payload. The header is significant as it contains a well-defined function code, which can tell the recipient whether it should confirm, read, write, select a specific point, operate a point (initiate a change to a point), directly operate a point (both selecting and changing a point in one command), or directly operate a point without acknowledgment.[6]

These functions are especially powerful when considering that the data payload of the DNP3 frame supports analog data, binary data, files, counters, and other types of data objects. At a high level, DNP3 supports two kinds of data, referred to as class 0 or static data (data that represents a static value or point reading) and event data (data that represents a change, some sort of activity, or an alarm condition). Event data is rated by priority from class 1 (highest) to class 3 (lowest). The differentiation of static and event data, as well as the classification of event data allows DNP3 to operate more efficiently by allowing higher-priority information to be polled more frequently, for example, or to enable or disable unsolicited responses by data type. The data itself can be binary, analog input or output, or a specific control output.[7]

How It Works

DNP3 provides a method to identify the remote device's parameters and then use message buffers corresponding to event data classes 1 through 3 in order to identify incoming messages and compare them to known point data. In this way, the master device is only required to retrieve new information resulting from a point change or change event.

Initial communications are typically a class 0 request from the Master to an Outstation, used to read all point values into the Master database. Subsequent communications will typically either be direct poll requests for a specific data class from the Master; unsolicited responses for a specific data class from an outstation; control or configuration requests from the Master to an RTU, and subsequent periodic class 0 polls. When a change occurs on an outstation, a flag is set to the appropriate

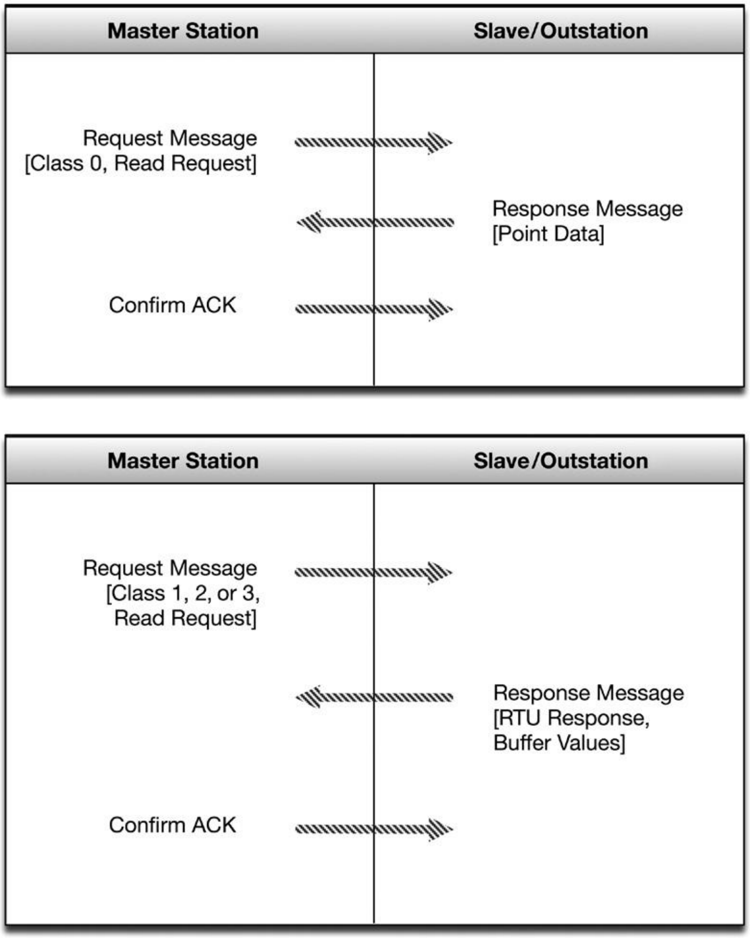


FIGURE 4.8

DNP3 Protocol Operation.

data class. The Master station is then able to poll only those outstations where there is new information to be reported.

This is a huge departure from constant data polling that can result in improved responsiveness and much more efficient data exchange. The departure from a real-time polling mechanism does require time synchronization, however, because the time between a change event and a successful poll/request sequence is variable. Therefore, all responses are time-stamped, so that the events between polls can be reconstructed in the correct order.

Communication is initiated by the Master to the Slave, or in the case of unsolicited responses (alarms) from the slave to the master, as shown in Figure 4.8.

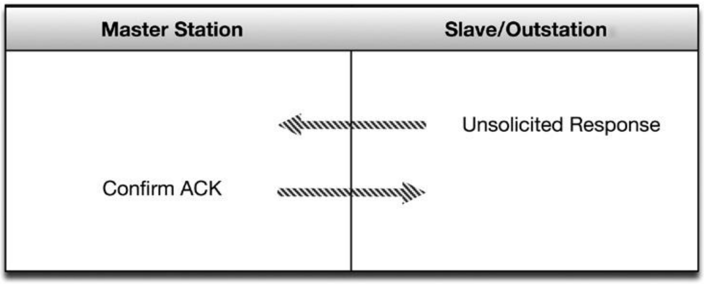


FIGURE 4.9

DNP3 Protocol Operation: Unsolicited Responses Allow Remote Alarm Generation.

Because DNP3 operates bidirectionally and supports unsolicited responses, as shown in Figure 4.9, each frame requires both a source address and a destination address so that the recipient device knows which messages to process, and so that it knows which device to return responses to. Although the addition of a source address (remember that in the other purely Master/Slave protocols, there is no need for a source address as the originating device is always the master) does add some overhead, it does so for the sake of dramatically increased scalability and functionality; as many as 65,520 individual addresses are available within DNP3, and any one of them can initiate communications. An address equals one device (every DNP3 device requires a unique address), although there are reserved DNP3 addresses, including one for broadcast messages (which will be received and processed by all connected DNP3 devices).[8]

Secure DNP3

Secure DNP3 is a DNP3 variant that adds authentication to the response/request process, as shown in Figure 4.10. Authentication is issued as a challenge by the receiving device. A challenge condition occurs upon session initiation (when a master station initiates a DNP3 session with an outstation), after a preset period of time (the default is 20 minutes), or upon a critical request (it is possible to know which requests are critical because the data types and functions of DNP3 are well defined) such as writes, selects, operates, direct operates, starts, stops and restarts, etc.[9]

Authentication occurs using a unique session key, which is hashed together with message data from the sender and from the challenger. The result is an authentication method that at once verifies authority (checksum against the secret key), integrity (checksum against the sending payload), and pairing (checksum against the challenge message). In this way, it is very difficult to perform data manipulation or code injection, or to spoof or otherwise hijack the protocol.[10]

The DNP3 layer 2 frame provides the source, destination, control, and payload, and can operate over a variety of application layers including TCP/IP (typically

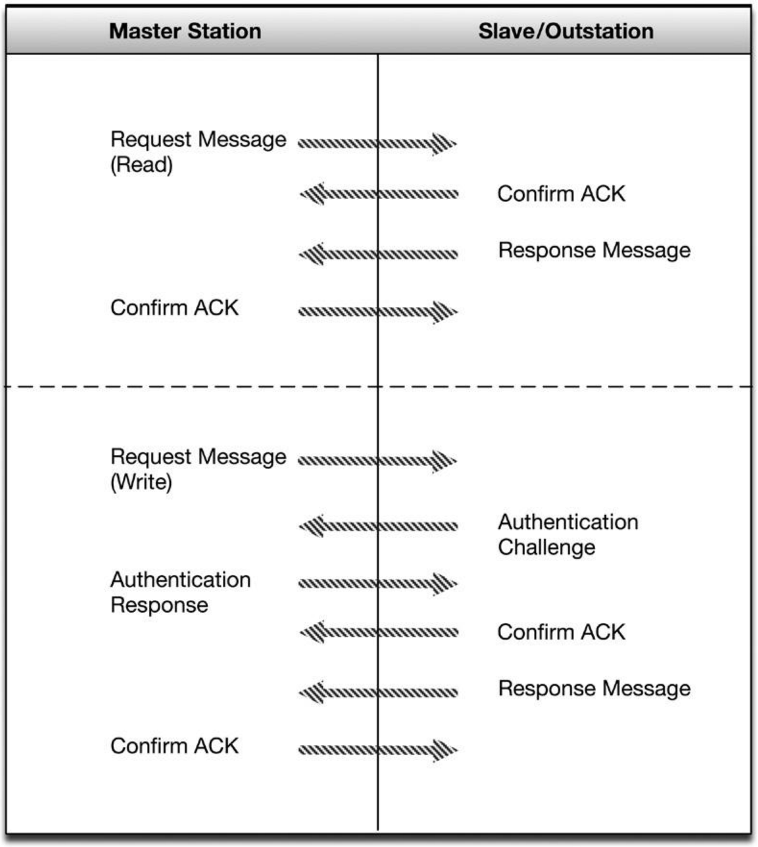


FIGURE 4.10

Message Confirmation and Secure DNP3 Authentication Operation.

using TCP port 20000 or UDP port 20000). The function codes are resident within the CNTRL bytes in the DNP3 frame header, as shown in Figure 4.11.

Where It Is Used

As shown in Figure 4.12, DNP3 is primarily used between a master control station and an RTU in a remote station, over almost any medium including wireless, radio, and dial-up. However, DNP3 is also widely used between RTUs and IEDs. As such it competes directly with the Modbus protocols within several areas of the control system.

Unlike Modbus, however, DNP3 is well suited for hierarchical and aggregated point-to-multipoint topologies in addition to the linear point-to-point and serial point-to-multipoint topologies that are supported by Modbus.[11]

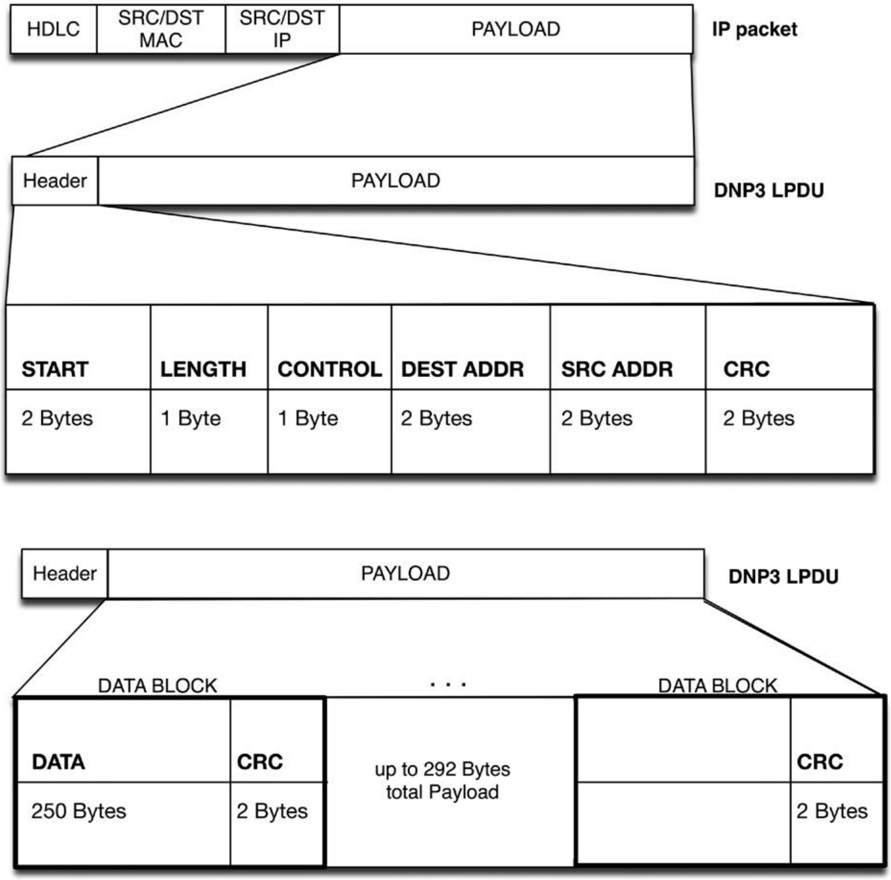


FIGURE 4.11

DNP3 Protocol Framing.

Security Concerns

While much attention is given to the integrity of the data frame, there is no authentication or encryption inherent within DNP3 (although there is within Secure DNP3). Because of the well-defined nature of DNP3 function codes and data types, it then becomes relatively easy to manipulate a DNP3 session.

Also, while DNP3 does include security measures, the added complexity of the protocol increases the chances of vulnerability. As of this writing, there are several known vulnerabilities with DNP3 that are reported by ICS-CERT. Because there are known exploits in the wild and DNP3 is a heavily deployed protocol, proper penetration testing and patching of DNP3 interconnections is recommended.

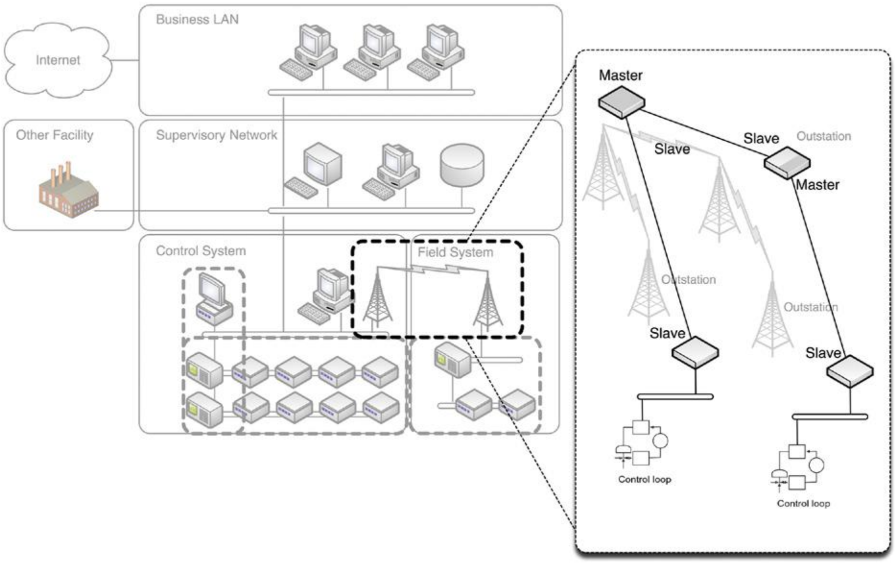


FIGURE 4.12

Typical DNP3 Use within the Industrial Network Architecture.

Some examples of realistic hacks against DNP3 include the use of MITM attacks to capture addresses, which can then be used to manipulate the system. Examples include the following:

- Turning off unsolicited reporting to stifle alarms.[12]
- Spoofing unsolicited responses to the Master to falsify events and trick an operator into taking inappropriate actions.
- Performing a DoS attack through the injection of broadcasts, creating storm behavior within the full extent of the DNP3 system.
- Manipulating the time synchronization data, resulting in synchronization loss and subsequent communication errors.
- Manipulating or eliminating confirmation messages forcing a state of continuous retransmission.
- Issuing unauthorized stops, restarts, or other functions that could disrupt operations.

Security Recommendations

Because a secure implementation of DNP3 exists, the primary recommendation is to implement only Secure DNP3. However, it may not always be possible due to varying vendor support and other factors. In these cases, secure use of the transport layer protocol is advised, such as the use of Transport Layer Security (TLS). In other words, treat your encapsulated DNP3 traffic as highly sensitive information and use every TCP/IP security best practice to protect it.

As always, DNP3 masters and outstations should be isolated into a unique enclave consisting only of authorized devices (multiple enclaves can be defined for devices communicating to multiple clients, or for hierarchical Master/Slave pairs), and the enclave(s) should be thoroughly secured using standard defense-in-depth practices, including a firewall and/or IDS system that enforces strict control over the type, source, and destination of traffic over the DNP3 link.

Many threats can be detected through monitoring the DNP3 session, and looking for specific function codes and behaviors, including the following:

- Use of any non-DNP3 communication on a DNP3 Port (TCP and UDP Port 20000).
- Use of configuration function code 23 (Disable Unsolicited Responses).
- Use of control function codes 4, 5, or 6 (Operate, Direct Operate, and Direct Operate without Acknowledgment).
- Use of application control function 18 (Stop Application).
- Multiple, unsolicited responses over time (Response Storm).
- Any unauthorized attempt to perform an action requiring authentication.
- Any authentication failures.
- Any DNP3 communication sourced from or destined to a device that is not explicitly identified as a DNP3 master or slave device.

As with other industrial protocols, SCADA-IDS or SCADA-IPS can be easily configured to monitor for these activities, while an application-aware firewall or Application Data Monitor may be required to validate DNP3 sessions.

CAUTION

Any SCADA-IPS devices should be configured with caution, and should only alert on events, rather than block them (i.e., operate in IDS mode rather than active IPS mode), as a false positive that blocks a valid DNP3 function code could cause an unintentional failure within the control system.

OLE FOR PROCESS CONTROL

OPC is not an industrial network protocol, but rather an operational framework for the communication of Windows-based process control systems using Microsoft's Object Linking and Embedding (OLE) protocol, which itself heavily utilizes additional communications protocols such as Remote Procedure Call (RPC). That is, OPC is a suite of protocols that collectively enable process control systems to communicate using some of the underlying networking capabilities of Windows.

What It Does

OPC differs from the other fieldbus protocols discussed in this chapter in that its primary function is to interconnect other distributed control systems with Windows hosts, typically connected via an Ethernet TCP/IP network. OPC is an implementation of OLE for Process Control Systems. Originally OPC was Distributed Component Object Model (DCOM) based, and many OPC systems in use today use DCOM, although OPC has more recently been updated to use an object-oriented protocol called OPC-Unified Architecture (OPC-UA).[13]

OPC provides a common communications interface between diverse industrial control systems and products by leveraging Microsoft's DCOM communications API, reducing the need for device-specific drivers. In place of specific communications drivers for each device, simple device drivers could be written to interface with OPC. The use of OPC therefore minimized driver development and allowed for better optimization of core OPC interfaces.[14]

OPC's strengths and weaknesses come from its foundation, which is based upon Microsoft's OLE protocol. OLE is used extensively in Office document generation and is used to embed a common data set in both a Word file and an Excel spreadsheet, for example. This not only allows OPC-connected devices to communicate and interact with minimal operator feedback (as in the case of the Office documents) but also presents significant security challenges.[15]

How It Works

OPC works in a client/server manner, where a client application calls a local process, but instead of executing the process using local code, the process is executed on a remote server. The remote process is linked to the client application and is responsible for providing the necessary parameters and functions to the server, over an RPC.

In other words, the stub process is linked to the client, but when a function is performed, the process is performed remotely, on the server. The server RPC functions then transmit the requested data back to the client computer. Finally, the client process receives the data over the network, provides it to the requesting application, and closes the session, as shown in Figure 4.13.

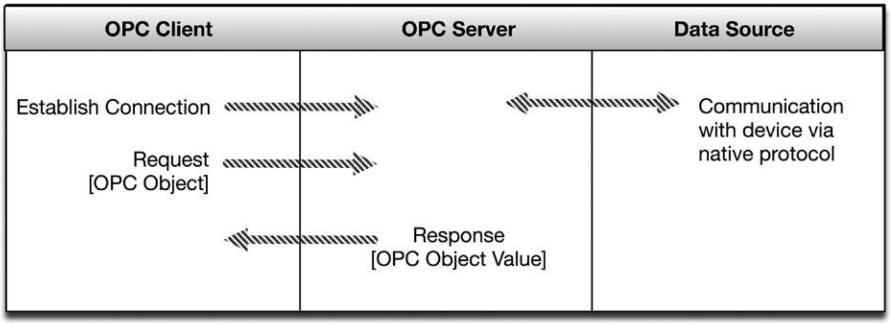


FIGURE 4.13

Typical OPC Protocol Operation.

In Windows systems, the requesting application typically loads RPC libraries at run-time, using a Windows dynamic link library (DLL).[16]

OPC is more complex than previous client/server industrial network protocols because of this interaction with the calling application and the underlying DCOM architecture. It interacts with various aspects of the host operating system, tying it closely to other host processes and exposing the protocol to a very broad attack surface. OPC also inherently supports remote operations (ROPs) that allow OPC to perform common control system functions.[17]

OPC-UA and OPC-XI

The OPC-UA and the OPC Express Interface (XI) are newer variations of OPC that break away from OPC Classic's dependence upon OLE. OPC-UA and OPC-XI preserve the functionality of earlier OPC implementations, while introducing new capabilities including stronger authentication services, encryption, and new transport mechanisms, including SOAP over HTTPS, and binary encoding to improve performance over XML, which has relatively high overhead.[18]

OPC-UA and OPC-XI represent strong improvements over legacy OPC implementations in terms of security. However, legacy OPC systems remain heavily deployed.

Where It Is Used

OPC is primarily a SCADA protocol, and it is used within many areas of industrial networks, including data transfer to data historians, data collection within HMIs, and other supervisory controls, as shown in Figure 4.14. OPC is a Windows interconnection, and so all communications occur either between Windows-based devices, or via OPC gateways that translate the RPC to the native fieldbus format. Unfortunately, OPC is also widely used within an industrial system's business network, including connections to corporate intranets, and even the Internet.[19] Because of the common use of OLE, RPC, and DCOM protocols within OPC, this opens the SCADA environment to a very broad attack surface.

Typically, OPC will be used "upstream" of fieldbus protocols, acting as a gateway between these protocols and Windows-based computing networks.

Security Concerns

OPC's use of DCOM and RPC makes it highly vulnerable to attack, as it is subject to the same vulnerabilities as the more ubiquitously used OLE. In addition, OPC is rooted in the Windows operating system (OS) and is therefore susceptible to attack through exploitation of any vulnerability inherent to the OS.[20]

OPC and related control system vulnerabilities are available only to authorized members of ICS-CERT; however, many OLE and RPC vulnerabilities exist and are well known. Because of the difficulties involved in patching production systems

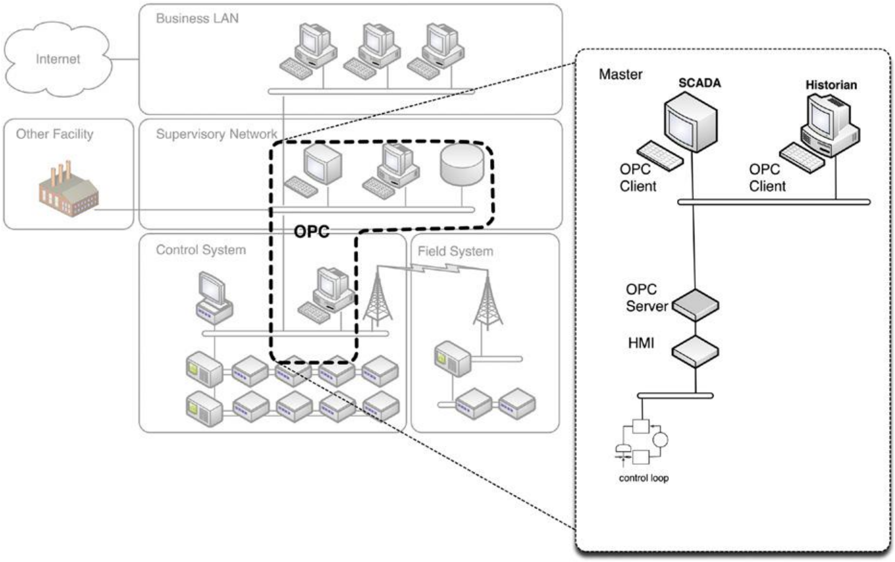


FIGURE 4.14

Typical OPC Use within the Industrial Network Architecture.

within an industrial network (see Chapter 6, "Vulnerability and Risk Assessment"), many of these vulnerabilities are still in place, even if there is an available patch from Microsoft.

Also, because OPC is supported on Windows, many basic host security concerns apply. Many OPC hosts utilize weak authentication, and passwords are often weak when authentication is enforced. Many systems support additional Windows services that are irrelevant to SCADA systems, resulting in unnecessary processes, which often correspond to open ports. These issues open the OPC system up to a broader attack surface. Inadequate or nonexistent logging exacerbates this by providing insufficient forensic detail should a breach occur, as Windows 2000/XP auditing settings do not record DCOM connection requests by default.[21]

In other words, unlike the simple and single-purpose protocols discussed until now, OPC must be treated as a larger system, according to modern OS and network security practices. Given OPC's reliance on Microsoft authentication mechanisms, weak passwords are among the most critical vulnerabilities that can undermine the security of an OPC server. Inadequate logging is also a primary concern, as by default, Windows 2000/XP auditing settings do not record DCOM connection requests, SMB log-ins, or attempts to access system objects.[22]

Other security concerns of OPC include the following:

- Legacy authentication services. Because systems within industrial networks are difficult to upgrade (due to limited maintenance windows, interpretability concerns, and other factors), insecure authentication mechanisms remain in use. For example, Windows 2000 LanMan (LM) and NT LanMan (NTLM) authentication mechanisms are still used by default in many systems. These and other legacy authentication mechanisms may be vulnerable and susceptible to exploitation.[23]

- RPC vulnerabilities. Because OPC uses RPC, it is susceptible to all RPC-related vulnerabilities, including several vulnerabilities that are exposed prior to authentication. Exploitation of underlying RPC vulnerabilities could result in arbitrary code execution, or DoS.[24]
- Unnecessary ports and services. OPC supports other network protocols other than TCP/IP, including NetBIOS Extended User Interface (NetBEUI), Connection Oriented NetBIOS over InterNetwork packet Exchange (IPX), and Hyper Text Transport Protocol (HTTP) Internet services.[25]
- OPC Server Integrity. It is possible to create a rogue OPC server and to use that server for disruption of service, DoS, information theft through bus snooping, or the injection of malicious code.[26]

Security Recommendations

OPC-UA or OPC-XI should be used where possible.

Regardless of the OPC variation used (Classic, UA, or XI), all unnecessary ports and services should be removed or disabled from the OPC server. This includes any and all irrelevant applications, and all unused network protocols. All unused services may introduce vulnerabilities to the system that could result in a compromise of the Windows host, and therefore the OPC network.[27]

OPC servers should be isolated into a unique enclave consisting only of authorized devices, and the enclave(s) should be thoroughly secured using standard defense-in-depth practices, including a firewall and/or IDS/IPS system that enforces strict control over the type, source, and destination of traffic to and from the OPC enclave.

Because OPC is primarily used in a supervisory capacity, IPS can be considered in place of IDS, understanding that an IPS may block SCADA traffic and result in a lack of visibility into control system operations. If information loss will be damaging to the control process or detrimental to business operations, use only IDS.

Many threats can be detected through monitoring OPC networks and/or OPC servers (server activity can be monitored through the collection and analysis of Windows logs), and looking for specific behaviors, including the following:

- The use of non-OPC ports and services initiated from the OPC server.
- The presence of known OPC (including underlying OLE RPC and DCOM) exploits.
- OPC services originating from unknown OPC servers (indicating the presence of a rogue server).
- Failed authentication attempts or other authentication anomalies on the OPC server.
- Successful authentication attempts on the OPC server from unknown or unauthorized users.

Most commercially available IDS and IPS devices support a wide range of detection signatures for OLE and RPC and therefore can also detect many of the underlying vulnerabilities of OPC. Similarly, most open-source and commercial log analysis and threat detection tools are capable of collecting and assessing Windows logs.

TIP

OPC-UA and OPC-XI, as well as certain OPC Classic vulnerabilities, may require the use of a SCADA-IDS or SCADA-IPS rather than an enterprise IDS or IPS. Enterprise devices typically detect exploits via inspection of OLE, RPC, and DCOM and may not be able to detect all threats targeting OPC. In some cases, enterprise IDS and IPS devices may be adapted to detect a wider range of OPC threats, using Snort® compatible preprocessors and detection signatures available from Digital Bond.

OTHER INDUSTRIAL NETWORK PROTOCOLS

There are dozens of industrial protocols—more than can be covered in even cursory detail within this book. Several warrant mention, as they introduce new concepts and/or concerns regarding industrial network security. These include **Ethernet/IP**, Profibus, EtherCAT, Ethernet Powerlink, and **SERCOS III**.

Ethernet/IP

Ethernet/IP uses standard Ethernet frames (ethertype 0x80E1) in conjunction with the Common Industrial Protocol (CIP) suite to communicate with nodes. Communication is typically client/server, although an "implicit" mode is supported to handle real-time requirements. Implicit mode uses connectionless transport—specifically the User Datagram Protocol (UDP) and multicast transmissions—to minimize latency and jitter.

NOTE

The "IP" in Ethernet/IP derives from "Industrial Protocol" and not "Internet Protocol," because of the use of the Common Industrial Protocol (CIP). Similarly, the acronym "CIP" meaning "Common Industrial Protocol" should not be confused with "Critical Infrastructure Protection" of NERC CIP.

The CIP uses object models to define the various qualities of a device. There are three types of objects: Required Objects, which define attributes such as device identifiers, routing identifiers, and other attributes of a device such as the manufacturer, serial number, date of manufacture, etc.; Application Objects, which define input and output profiles for devices; and Vendor-specific Objects, which enable vendors to add proprietary objects to a device. Objects (other than vendor-specific objects) are standardized by device type and function, to facilitate interoperability: if one brand of pump is exchanged for another brand, for example, the application objects will remain compatible, eliminating the need to build custom drivers. The wide adoption and standardization of CIP has resulted in an extensive library of device models, which can facilitate interoperability but can also aid in control network scanning and **enumeration** (see Chapter 6, "Vulnerability and Risk Assessment").

While the Required Objects provide a common and complete set of identifying values, the Application Objects contain a common and complete suite of services for control, configuration, and data collection that includes both implicit (control) and explicit (information) messaging.[28]

Security Concerns

Ethernet/IP is a real-time Ethernet protocol, and as such it is susceptible to any of the vulnerabilities of Ethernet. Ethernet/IP over UDP is transaction-less and so there is no inherent network-layer mechanism for reliability, ordering, or data integrity checks. The CIP also introduces some specific security concerns, due to its well-defined object model.

The following concerns are specific to Ethernet/IP:

- The CIP does not define any explicit or implicit mechanisms for security.
- The use of common Required Objects for device identification can facilitate device identification and enumeration, facilitating an attack.
- The use of common Application Objects for device information exchange and control can enable broader industrial attacks, able to manipulate a broad range of industrial devices.
- Ethernet/IP's use of UDP and Multicast traffic—both of which lack transmission control—for real-time transmissions facilitate the injection of spoofed traffic or (in the case of multicast traffic) the manipulation of the transmission path using injected IGMP controls.

Security Recommendations

Because Ethernet/IP is a real-time Ethernet protocol using UDP and IGMP, it is necessary to provide Ethernet and IP-based security at the perimeter of any Ethernet/IP network. It is also recommended that passive network monitoring be used to ensure the integrity of the Ethernet/IP network, ensuring that the Ethernet/IP protocol is only being used by explicitly identified devices and that no Ethernet/IP traffic is originating from an unauthorized, outside source. This can be accomplished using a SCADA-IDS/IPS or other network monitoring device capable of detecting and interpreting the Ethernet/IP protocol.

Profibus

Profibus (Process fieldbus) is a fieldbus protocol that was originally developed in the late 1980s in Germany by the Central Association for the Electrical Industry. Several specialized variants of Profibus exist, including Profibus-DP (Decentralized Periphery) and Profibus-PA (Process Automation). The standardized variant is Profibus-DP, which itself has three common variants: Profibus DP-V0, DP-V1, and DP-V2, each of which represents a minor evolution of capabilities within the protocol. There are also three profiles for Profibus communication: asynchronous, synchronous, and via Ethernet using ethertype 0x8892. Profibus over Ethernet is also called **Profinet**.[29]

Profibus is a Master/Slave protocol that supports multiple master nodes through the use of token sharing: when a master has control of the token, it can communicate with its slaves (each slave is configured to respond to a single master). In Profibus DP-V2, slaves can initiate communications to the master or to other slaves under certain conditions. Typically, a master Profibus node is a PLC or RTU, and a slave is sensor, motor, or some other control system device.

Security Concerns

Profibus lacks authentication inherent to many of its functions, allowing a spoofed node to impersonate a master node, which in turn provides control over all configured slaves. A compromised master node or a spoofed master node could also be used to capture the token, inject false tokens, or otherwise disrupt the protocol functions, causing a DoS. A rogue master node could alter clock synchronization to slave devices, snoop query responses (across all masters), or even inject code into a slave node.

Profibus over Ethernet (Profinet) is a real-time Ethernet protocol, and as such it is susceptible to any of the vulnerabilities of Ethernet. When used over the IP, it is also susceptible to any vulnerabilities of IP.

NOTE

Stuxnet (see Chapter 3, "Introduction to Industrial Network Security") is an example of Profibus exploitation. Stuxnet compromised PLCs (master Profibus nodes), which were then used to monitor the Profibus and look for specific behaviors associated with frequency controllers. Once the sought-after conditions were detected, Stuxnet then issued commands to the relevant slave nodes to sabotage the process.

Security Recommendations

As with many fieldbus protocols, the inherent lack of authentication and vulnerability of the protocol requires strong isolation of the bus. If Profinet is used, it should be controlled and used only over authenticated and encrypted networks. Monitoring of Ethernet networks for unauthorized or suspicious use of Profinet should be implemented as well, and firewalls and IPS devices should be configured to explicitly deny Profinet outside of well-defined areas.

EtherCAT

EtherCAT is another real-time Ethernet fieldbus protocol, which uses a defined Ethernet ethertype (0x88A4) to transport control systems communications over standard Ethernet networks. To maximize the efficiency of distributed process data communications (which requires only a few bytes per cycle) over Ethernet frames (which vary in size from 46 to 1500 bytes of payload), EtherCAT communicates large amounts of distributed process data with just one Ethernet frame, so that typically only one or two Ethernet frames are required for a complete cycle. Slaves pass the frame(s) to other slaves in sequence, appending its appropriate response, until the last slave returns the completed response frame back.[30]

Security Concerns

EtherCAT is a real-time Ethernet protocol, and as such it is susceptible to any of the vulnerabilities of Ethernet. EtherCAT over UDP is transaction-less and so there is no inherent network-layer mechanism for reliability, ordering or data integrity checks.

As with many real-time Ethernet protocols, EtherCAT is sensitive and highly susceptible to DoS attacks. EtherCAT is easily disrupted via the insertion of rogue Ethernet frames into the network to interfere with time synchronization and is subject to spoofing and MITM attacks due to the lack of bus authentication, requiring the separation of EtherCAT from other Ethernet systems.

Security Recommendations

Because EtherCAT is a real-time Ethernet protocol, it is necessary to provide Ethernet-based security at the perimeter of any EtherCAT network. It is also recommended that passive network monitoring be used to ensure the integrity of the EtherCAT network, ensuring that the EtherCAT protocol is only being used by explicitly identified devices and that no EtherCAT traffic is originating from an unauthorized, outside source. This can be accomplished using a SCADA-IDS/IPS or other network monitoring device capable of detecting and interpreting the EtherCAT protocol. A network monitoring product or probe can also be used to detect Ethernet packets using EtherCAT's specific ethertype.

Ethernet Powerlink

Ethernet Powerlink uses Fast Ethernet as the basis for real-time transmission of industrial control messages. A master node is used to initiate and synchronize cyclic polling of slave devices, by transmitting a master "Start of Cycle" frame that provides a basis for the network synchronization. The master then polls each station; slaves can only respond if they receive a poll request frame, ensuring that all Master/Slave communications occur in sequence. Slave responses are broadcast, eliminating source address resolution. Because collisions are avoided solely via the carefully controlled request/response cycles, Ethernet Powerlink is best used homogeneously: the introduction of other Ethernet-based systems could disrupt synchronization and cause a failure.[31]

Ethernet Powerlink is often used in conjunction with CANopen, an application layer protocol based on CAN (Controller Area Network). CANopen enables the communication between devices of different manufacturers, and the protocol stacks are widely available including open-source distribution for both Windows and Linux platforms. The open nature of CANopen makes Ethernet Powerlink/CANopen a desirable combination for industrial networks requiring inexpensive solutions in Linux environments.[32]

Security Concerns

Ethernet Powerlink is a real-time Ethernet protocol, and as such it is susceptible to any of the vulnerabilities of Ethernet. Ethernet Powerlink is designed for use over all IPs, including TCP, UDP, and HTTP, and it is therefore also susceptible to any corresponding IP vulnerabilities.

As with many real-time Ethernet protocols, Ethernet Powerlink is sensitive and highly susceptible to DoS attacks. Ethernet Powerlink is easily disrupted via the insertion of rogue Ethernet frames into the network, requiring the separation of Ethernet Powerlink from other Ethernet systems. The protocol itself is sensitive and highly susceptible to DoS attacks.

Security Recommendations

Because sensitivity of the cyclic polling mechanism requires separation from other non–Powerlink Ethernet services, Ethernet Powerlink implementations will most likely have a clear demarcation from other networks. This demarcation can be leveraged to further isolate the industrial protocol, through the establishment of strong perimeter defenses at these boundaries.

SERCOS III

SERCOS (Serial Real-time Communications System) is a fieldbus specialized for digital motion control. SERCOS III is a real-time Ethernet communication protocol specifically designed for serial communications between PLCs and IEDs, operating at high speeds within closed loops.[33]

SERCOS III is a Master/Slave protocol that operates cyclically, using a mechanism in which a single Master Synchronization Telegram is used to communicate to slaves, and the slave nodes are given a predetermined time (again synchronized by the master node) during which they can place their data on the bus. All messages for all nodes are packaged into a Master Data Telegram, and each node knows which portion of the MDT it should read based upon a predetermined byte allocation.[34]

An interesting addition to SERCOS III is that, although SERCOS dedicates the use of the bus for synchronized real-time traffic during normal cycles, it allows unallocated time within a cycle to be freed up for other network protocols such as IP. This "IP Channel" allows the use of broader network applications from the same device—for example, a web-based management interface that would be accessible to business networks.[35]

Security Concerns

SERCOS III is a real-time Ethernet protocol, and as such it is susceptible to any of the vulnerabilities of Ethernet. SERCOS III introduces new security concerns through the option to support embedded, open TCP/IP communications. With this option enabled, a compromised RTU or PLC using SERCOS III could be used to launch an in-bound attack into other corporate communications systems, including SCADA and business networks.

Security Recommendations

SERCOS III should be isolated to control loops that require the protocol, and the use of IP channels should be seriously considered and avoided where possible. If IP channels are used, the extent and reach of the IP channel should be enclosed within an explicitly defined enclave consisting of the SERCOS III master node and only those TCP/IP network devices that are absolutely required.

AMI AND THE SMART GRID

The smart grid is a term encompassing many aspects of modern power transmission. Although smart grid technology might seem irrelevant to many industrial network systems outside of the energy industry, it is discussed briefly here because of its broad reach and vulnerable attack surface. The smart grid is a widely distributed communication network that touches both energy production and transmission systems and many end user networks. Therefore, the smart grid represents an easily accessible network that contains many vectors to many possible targets; once compromised, an attacker could use the network to attack the power utility's network, or to attack the networks of connected home and/or business networks.

The term "smart grid" is widely used and generally refers to a new generation of energy distribution built around an Advanced Metering Infrastructure (AMI). AMI promises many new features designed to increase the efficiency and reduce the costs of energy distribution. Common AMI features include Remote Meter Reading; Remote Billing; Demand/Response Energy Delivery; Remote Connect/Disconnect; and Remote Payment and Prepayment.[36]

At a high level, the smart grid requires coordination among the following systems:

- Bulk Generation Systems
- Transmission Systems
- Distribution Systems
- Customer Information and Management Systems
- Usage and Meter Management Systems
- Billing Systems
- Interconnected network systems, including Neighborhood Area Networks (often using wireless mesh technologies); Metropolitan Area Networks (MANs); Home Area Networks (HANs); and Business Area Networks (BANs)

The smart grid is essentially a large, end-to-end communications system interconnecting power suppliers to power consumers (see Figure 4.15). It is made of highly diverse systems, using diverse protocols and network topologies. Smart grids even introduce new protocols. To support home- and business-based service

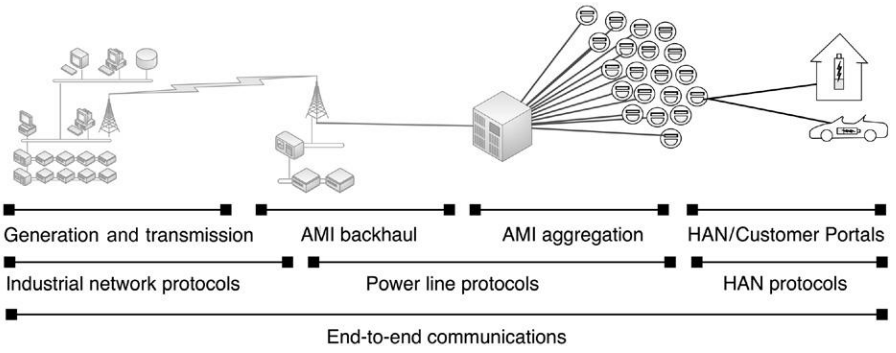

Generation and transmission
AMI backhaul
AMI aggregation
HAN/Customer Portals
Industrial network protocols
Power line protocols
HAN protocols
End-to-end communications

FIGURE 4.15

Smart Grid Operational Areas and Protocols.

portals, Smart Metering introduces HAN and BAN protocols, such as Zigbee and HomePNA, as well as power line protocols such as IEC 61334, Control Network Power Line (PL) Channel Specification, and Broadband over Powerline (BPL). Although the data link and application protocols are too numerous to discuss in detail, it is widely accepted that TCP/IP will be leveraged for network-layer communications.[37]

Although these specific protocols will not be discussed within this book, it is important to recognize that the disparate nature of these systems requires that several distinct operational models and several distinct network architectures combine to form a single end-to-end communications path, as illustrated in Figure 4.15. This means that while many distinct smart grid protocols may be used, the smart grid as a whole should be considered as a single, highly accessible communications network that is highly interconnected.

Security Concerns

The security concerns of smart grid are numerous. AMI represents an extremely large network that touches many private networks and is designed for command and control in order to support remote disconnect, demand/response billing, and other features.[38] Combined with a lack of industry-accepted security standards, the smart grid represents significant risk to connected systems that are not adequately isolated. Specific security concerns include the following:

- Smart meters are highly accessible and therefore require board- and chip-level security in addition to network security.
- Smart grid protocols vary widely in their inherent security and vulnerabilities.
- Neighborhood, home, and business LANs can be used both as an ingress to the AMI, and as a target from the AMI.

- Smart grids are ultimately interconnected with critical power generation and distribution systems.
- Smart grids represent a target to private hackers (for financial gain or service theft) as well as to more sophisticated and serious attackers (for sociopolitical gain or cyber warfare).

Security Recommendations

The best recommendation for smart grid security at this point is for electric utilities to carefully assess smart grid deployments and to perform risk and threat analysis early in the planning stages, and for the end users who are connected to the smart grid to perform a similar assessment of the system as a potential threat vector into the business (or home) network.

Again, clear delineation, separation of services, and the establishment of strong defense in depth at the perimeters will help to minimize any threat associated with the smart grid. For the smart grid operators, this could represent a challenge (especially in terms of security monitoring) due to the broad scale of smart grid deployments, which could contain hundreds of thousands or even millions of intelligent nodes. Therefore, it may be necessary to carve smart grid deployments into multiple, smaller and more manageable security enclaves.

SUMMARY

Industrial networks use a variety of specialized “fieldbus” protocols to accomplish specific tasks, often with careful attention to synchronization and real-time operation. Each protocol has varying degrees of inherent security and reliability, and these qualities should be considered when attempting to secure these protocols. However, because industrial network protocols, in general, lack sufficient authentication or encryption, all are susceptible to cyber attack using relatively simple MITM attacks, which can be used to disrupt normal protocol operations or potentially to alter or otherwise manipulate protocol messages to steal information, commit fraud, or potentially cause a failure of the control process itself.

By understanding each protocol and isolating each into its own carefully defined security enclave, these protocols can be reasonably secured (see Chapter 7, “Establishing Secure Enclaves”). Because each protocol has specific uses within a control system, the creation of enclaves based purely on physical devices is possible and relatively simple. However, as industrial network protocols are used more widely over Ethernet and/or TCP/IP, the creation of clean enclave boundaries becomes more difficult, as boundaries begin to overlap. For this reason, the use of “business” network protocols to transport fieldbus protocols should be avoided unless absolutely necessary, and be especially scrutinized and tested where they are necessary.

ENDNOTES

1. The Modbus Organization, Modbus Application Protocol Specification V1.1b. <http://www.modbus.org/docs/Modbus_Application_Protocol_V1_1b.pdf>, December 2006 (cited: November 24, 2010).
2. Ibid.
3. J.T. Michalski, A. Lanzone, J. Trent, S. Smith, SANDIA Report SAND2007-3345: Secure ICCP Integration Considerations and Recommendations, Sandia National Laboratories, Albuquerque, New Mexico and Livermore, California, June 2007.
4. Ibid.
5. The DNP Users Group, DNP3 Primer, Revision A. <http://www.dnp.org/About/DNP3%20Primer%20Rev%20A.pdf>, March 2005 (cited: November 24, 2010).
6. G.R. Clarke, Deon Reynders Practical Modern SCADA Protocols: DNP3, 60870.5 and Related Systems, Newnes, Oxford, UK and Burlington MA, 2004.
7. The DNP Users Group, DNP3 Primer, Revision A. <http://www.dnp.org/About/DNP3%20Primer%20Rev%20A.pdf>, March 2005 (cited: November 24, 2010).
8. Ibid.
9. Digitalbond SCADAPEDIA, Secure DNP3. <http://www.digitalbond.com/wiki/index.php/Secure_DNP3>, August 2008 (cited: November 24, 2010).
10. Ibid.
11. The DNP Users Group, DNP3 Primer, Revision A. <http://www.dnp.org/About/DNP3%20Primer%20Rev%20A.pdf>, March, 2005 (cited: November 24, 2010).
12. A.B.M. Omar Faruk, Testing & Exploring Vulnerabilities of the Applications Implementing DNP3 Protocol, KTH Electrical Engineering, Stockholm, Sweden, June 2008.
13. The OPC Foundation OPC Task Force, OPC Overview Version 1.0. <http://www.opcfoundation.org/Archive/c218e7b5-4e00-4f95-82ba-7da07eb17883/General/OPC%20Overview%201.00.pdf>, October 27, 1998 (cited: November 24, 2010).
14. Ibid.
15. Digital Bond, British Columbia Institute of Technology, and Byres Research. OPC Security White Paper #2: OPC Exposed (Version 1-3c), Byres Research, Lantzville, BC and Sunrise, FL, November 13, 2007.
16. Microsoft Corporation, RPC Protocol Operation. <http://msdn.microsoft.com/en-us/library/ms818824.aspx> (cited: November 4, 2010).
17. European Organization for Nuclear Research (CERN), A Brief Introduction to OPC™ Data Access. <http://itcofe.web.cern.ch/itcofe/Services/OPC/GeneralInformation/Specifications/RelatedDocuments/DASummary/DataAccessOvw.html>, November 11, 2000 (cited: November 29, 2010).
18. Digital Bond, British Columbia Institute of Technology, and Byres Research, OPC Security White Paper #1 Understanding OPC and How It Is Deployed (Version 1-3b), Byres Research, Lantzville, BC and Sunrise, FL, July 27, 2007.
19. Ibid.
20. Digital Bond, British Columbia Institute of Technology, and Byres Research. OPC Security White Paper #2: OPC Exposed (Version 1-3c), Byres Research, Lantzville, BC and Sunrise, FL, November 13, 2007.
21. Ibid.
22. Ibid.
23. Ibid.

24. Ibid.
25. Ibid.
26. Ibid.
27. Ibid.
28. ODVA, CIP Technology Overview. <http://www.odva.org/Home/ODVATECHNOLOGIES/CIP/CIPTechnologyOverview/tabid/68/Default.aspx>, 2010 (cited: November 24, 2010), Saarbrücken, Germany.
29. V.M. Igure, Security assessment of SCADA protocols: a taxonomy based methodology for the identification of security vulnerabilities in SCADA protocols, VDM Verlag Dr. Müller Aktiengesellschaft & Co. KG, 2008.
30. The EtherCAT Technology Group, Technical introduction and overview: EtherCAT—the Ethernet Fieldbus. <http://www.ethercat.org/en/technology.html#5>, May 10, 2010 (cited: November 24, 2010).
31. P. Doyle, Introduction to Real-Time Ethernet II. The Extension: A Technical Supplement to Control Network, vol. 5, Issue 4, Contemporary Control Systems, Inc., Downers Grove, IL, July 2004.
32. Ethernet POWERLINK Standardization Group, CANopen. <http://www.ethernet-powerlink.org/index.php?id=39>, 2009 (cited: November 24, 2010).
33. SERCOS International, Technology: Introduction to SERCOS interface. <http://www.sercos.com/technology/index.htm>, 2010 (cited: November 24, 2010).
34. SERCOS International, Technology: Cyclic Operation. <http://www.sercos.com/technology/cyclic_operation.htm>, 2010 (cited: November 24, 2010).
35. SERCOS International, Technology: Service & IP Channels. <http://www.sercos.com/technology/service_ip_channels.htm>, 2010 (cited: November 24, 2010).
36. UCA® International Users Group, AMI-SEC Task Force, AMI System Security Requirements, UCA, Raleigh, NC, December 17, 2008.
37. National Institute of Standards and Technology, NIST Special Publication 1108: NIST Framework and Roadmap for Smart Grid Interoperability Standards, Release 1.0, February 2010.
38. UCA® International Users Group, AMI-SEC Task Force, AMI system security requirements, UCA, Raleigh, NC, December 17, 2008.

This page intentionally left blank

CHAPTER

How Industrial Networks Operate

5

INFORMATION IN THIS CHAPTER:

- Control System Assets
- Network Architectures
- Control System Operations
- Control Process Management
- Smart Grid Operations

In addition to understanding how industrial network protocols operate, it is necessary to understand how commonly used devices interact within an industrial network. For operators of industrial control systems, this information may seem overly basic. However, it is important to remember that how control systems *are* connected and how they *should be* connected are not always the same, and so by taking a short step back to the basics we can quickly assess whether there are any basic security flaws in an industrial network design. This requires an understanding of the specific assets, architectures, and operations of a typical industrial network.

CONTROL SYSTEM ASSETS

The first step is to understand the devices used within industrial networks and the roles that they play. These devices, which are discussed in this chapter, include operational devices such as sensors, motors, gauges, and other **intelligent electronic devices**; Remote Terminal Units (RTUs) and/or Programmable Logic Controllers (PLCs); Human Machine Interface (HMI) Control System Assets; Supervisory Management Workstations; Data Historians; and Business Information Consoles or Dashboards.

IEDs

An intelligent electronic device (IED) is any device commonly used within a control system—such as a sensor, actuator, motor, transformers, circuit breakers, and pumps—that is equipped with a small microprocessor that enables it to communicate digitally. These devices communicate almost exclusively using fieldbus protocols, operating as slave nodes, and are controlled via an upstream RTU or PLC. As with all technology, IEDs are growing more and more sophisticated over time, and an IED may perform other tasks, blurring the line between device types. However, to simplify things for the purposes of this book, an IED can be considered to support a specific function (i.e., a motor can spin at different frequencies) within the control system, typically within a specific control loop, whereas RTUs and PLCs are designed for general use (i.e., they can be programmed to control the speed of a motor, to engage a lock, to activate a pump, etc.).

RTUs

A Remote Terminal Unit (RTU) typically resides in a substation or other remote location. RTUs monitor field parameters and transmit that data back to a central monitoring station—typically either a Master Terminal Unit (MTU), or a centrally located PLC, or directly to an HMI system. RTUs, therefore, include remote communications capabilities, consisting of a modem, cellular data connection, radio, or other wide area communication capability. They will typically use industrial network protocols such as DNP3 to communicate between master and remote units, and either DNP3 or Modbus, Profibus or some other common fieldbus protocol to communicate with IEDs (see Chapter 4, “Industrial Network Protocols”).

RTUs and PLCs continue to overlap in capability and functionality, with many RTUs integrating programmable logic and control functions, to the point where an RTU can be thought of as a remote PLC.

PLCs

A programmable logic controller (PLC) is a specialized computer used to automate functions within industrial networks. Unlike desktop computers, PLCs are typically materially hardened (making them suitable for deployment on a production floor) and may be specialized for specific industrial uses with multiple specialized inputs and outputs. PLCs also differ from desktop computers in that they do not typically use a commercially available operating system (OS); instead they rely on blocks of logic code that allow the PLC to function automatically to specific inputs (e.g., from sensors) with as little overhead as possible. PLCs were originally designed to replace relays, and very simple PLCs may be referred to as programmable logic relays (PLRs).

PLCs typically control real-time processes, and so they are designed for simple efficiency. For example, in plastic manufacturing, a catalyst may need to be injected into a vat when the temperature reaches a very specific value; if processing overhead or other latency introduces delay in the execution of the PLC’s logic, it would be very difficult to precisely time the injections, which could result in quality issues. For this reason, the logic used on PLCs is typically very simple and usually based on ladder logic (although almost any programming language could theoretically be supported).

Again, as technology evolves, the line blurs between RTU, PLC, and IED, as can be seen in Emerson Process Management’s ROC800L liquid hydrocarbon

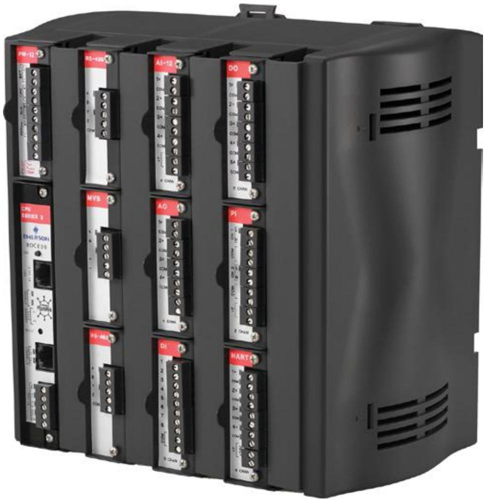


FIGURE 5.1

Emerson Process Management's ROC800L Liquid Hydrocarbon Remote Controller.

Photo courtesy of Emerson Process Management.

remote controller shown in Figure 5.1. This device performs measurement, diagnostics, and remote control in a single device that supports several programmable languages.

Ladder Logic

PLCs often use "ladder logic," a simplistic programming language that is well suited for industrial applications. Ladder logic is based on relay-based logic and can be thought of as a set of connections between inputs (contacts) and outputs (coils). Ladder logic follows a relay function diagram, as shown in Figure 5.2. A path is traced on the left side, across "rungs" consisting of various inputs. If an input relay is "true" the path continues, and if it is "false" it does not. If the path to the right side completes (there is a complete "true" path across the ladder), the ladder is complete and the output coil will be set to "true" or "energized." If no path can be traced, then the output remains "false," and the relay remains "de-energized."[1]

The PLC applies this ladder logic by looking at inputs from digital or analog devices such as sensors that are connected to the outside world and comparing them to set points. PLCs can use a variety of digital and analog communications methods, but typically use a fieldbus protocol such as Modbus or Profibus (see Chapter 4, "Industrial Network Protocols"). If a set point is satisfied, the input is considered "true," and if it is not it is considered "false." Processes defined by ladder logic can be simple or very complex. For example, an "or" condition can allow the rung to complete based on an alternate input condition, as shown in Figure 5.3.

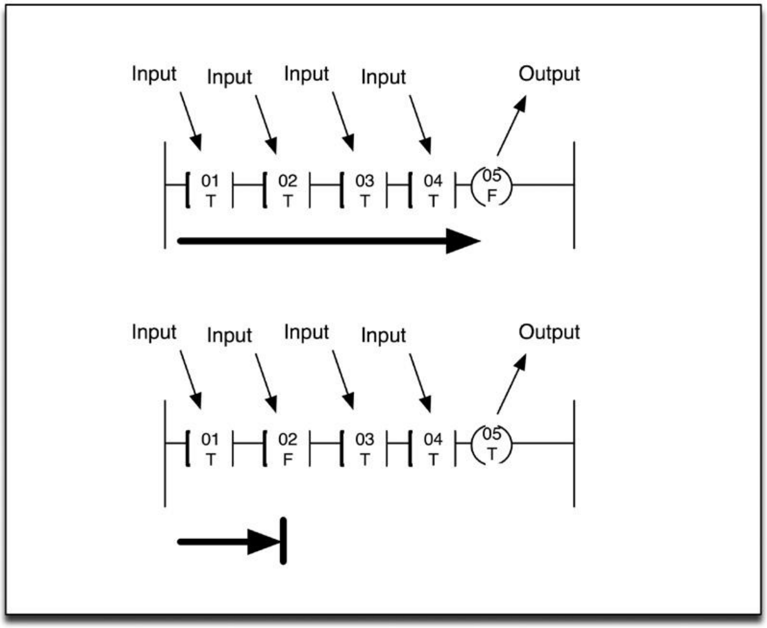


FIGURE 5.2

Example of Simple Ladder Logic, with Both Complete and Incomplete Conditions.

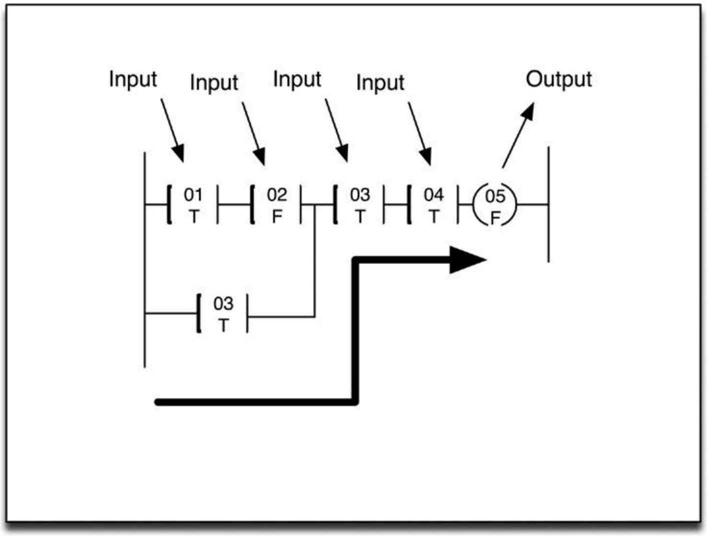


FIGURE 5.3

Example of Simple Ladder Logic, Containing an “or” Condition.

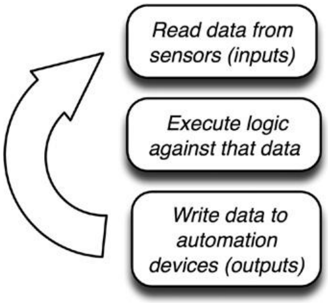

Read data from sensors (inputs)
Execute logic against that data
Write data to automation devices (outputs)

FIGURE 5.4

PLC Operational Flow Diagram.

When an output is finally reached it becomes "true," and the PLC activates the output. This allows the PLC to automate a function (e.g., turning a pump on or off) based on set point parameters (e.g., high and low water levels within a tank).[2]

Ladder logic is created using a software application on a PC and then is programmed onto a PLC by connecting that PC and transferring the ladder logic code onto the PLC. This PC can be a dedicated system or it can be a function of an HMI system. Internal relays may also be used within a PLC—these relays, unlike input relays, do not use inputs from outside but can be used by the ladder logic to lock an input on (true) or off (false) depending upon other conditions of the program. Finally, PLCs can use counters and timers, allowing PLCs to act in defined cycles or pulses, as well as storage.[3]

Sometimes PLCs use "Step Logic," which differs from ladder logic in that each step is tested in isolation and progresses to the next step only upon completion, whereas in ladder logic every step is tested in each scan. Again, almost any programming language could be supported on a modern PLC. However, the end goal is ultimately to automate the relay functions common in industrial systems by checking inputs, applying logic (the program), and adjusting outputs as appropriate,[4] as shown in Figure 5.4.

HMIs

Human machine interfaces (HMIs) are used as an operator control panel to PLCs, RTUs, and in some cases directly to IEDs. HMIs replace manually activated switches, dials, and other controls with graphical representations of the control process and digital controls to influence that process. HMIs allow operators to start and stop cycles, adjust set points, and perform other functions required to adjust and interact with a control process. Because the HMI is software based, they replace physical wires and controls with software parameters, allowing them to be adapted and adjusted very easily.

HMIs are modern software applications running on modern operating systems, and as such they are capable of performing many functions. They act as a bridge between the human operator and the complex logic of one or more PLCs, allowing the operator to function on the process rather than on the underlying logic that performs the function and to control many functions across distributed and potentially complex processes from a centralized location. To accomplish this, the user interface will graphically represent the process being controlled, including sensor values and other measurements, and visible representation of output states (which motors are on, which pumps are activated, etc.).

Humans interact with the HMI through a computer console, and as such must authenticate to the HMI system using password protection. Because HMIs provide supervisory data (visual representation of a control process's current state and values) as well as control (i.e., set point changes), user access may lock out specific functions to specific users. The security of the industrial process therefore relies heavily on access control and host security of the HMI.

The HMI, in turn, interacts with one or more PLCs and/or RTUs, typically using industrial protocols such as OLE for Process Control (OPC) or fieldbus protocols such as Modbus (see Chapter 4, "Industrial Network Protocols").

Supervisory Workstations

A supervisory workstation collects information from assets used within a control system and presents that information for supervisory purposes. Unlike an HMI, a supervisory workstation is primarily read-only; that is, there is no control element to interact directly with the control process, only the presentation of information about that process.

Typically, a supervisory workstation will consist of either an HMI system (with read-only or supervisory access restrictions) or a Data Historian—a device specifically designed to collect a running audit trail of control system operational data. Supervisory workstations can reside within the Supervisory Control and Data Acquisition demilitarized zone (SCADA DMZ) or within the business network—up to and including Internet-facing web portals, Intranets, etc. (see "Control Processes" on page 102).

CAUTION

When placing a supervisory workstation or any other service outside of its intended security enclave, the overall security of that enclave is weakened. For example, by placing a SCADA supervisory console in the business network, the console can be more easily accessed by an attacker and then utilized to communicate back into the SCADA DMZ. This is covered in detail in Chapter 7, "Establishing Secure Enclaves".

Data Historians

A Data Historian is a specialized software system that collects point values and other information from industrial devices and stores them in a purpose-built database.

Most industrial asset vendors—including ABB, Arreva, Emerson, GE, Invensys, Rockwell, Siemens, and others—provide their own proprietary Data Historian systems. In addition, there are third-party industrial Data Historian vendors, such as Canary Labs (www.canarylabs.com), Modiüs (www.modius.com), and OSIsoft (www.osisoft.com), which interoperate with third-party assets and even integrate with third-party Data Historians in order to provide a common, centralized platform for data historization and analysis.

Data points that are historized and stored within a Data Historian are referred to as "tags" and can represent almost anything—the current frequency of a motor or turbine, the rate of airflow through an heating, ventilation, and air-conditioning (HVAC) system, the total volume in a mixing tank, the specific volumes of injected chemical catalysts in a tank, etc. Tags can even represent human-generated values, such as production targets and acceptable loss margins.

Because the information stored within a Data Historian is used by both industrial operations and business management, Data Historians are often replicated across an industrial network. This can represent a security risk, as a Data Historian in a less secure zone (i.e., the business network) could be used as a vector into more secure zones (i.e., the SCADA DMZ). As such, Data Historians should be isolated, secured within their own enclaves, and should be patched regularly to minimize vulnerability.

NOTE

The information collected by a Data Historian is stored centrally within a database. Depending upon the Data Historian used, this could be a commercial Relational Database Management System (RDBMS), specialized columnar or time-series database system, or some other proprietary data storage system. The type of database used is important for several reasons. First, the Data Historian will typically be responsible for collecting information from thousands or even millions of devices. Especially in larger networks, the capabilities of the database in terms of data collection performance can impact the Data Historian's ability to collect operational information in real time. Second, and more importantly within the context of this book, is that commercial RDBMSs may present specific vulnerabilities to cyber attack. The Data Historian and any auxiliary system (database server, network storage, etc.) should be included in any vulnerability assessment, and care should be taken to isolate and secure these systems along with the Data Historian server.

At the time of this writing, OSIsoft holds a dominant position in the Historian market, with 65% market penetration in global industrial automated systems.[5] The OSIsoft PI System integrates with many IT and OT systems including other Data Historians, and as such is a premium target for attack. Again, applying the latest updates and patches can minimize vulnerabilities, while properly isolating and securing PI within its own enclave can minimize accessibility. For more information about the role of Data Historians within control system operations, see "Control Processes: Feedback Loops" and "Control Processes: Business Information Management" later in the chapter.

Business Information Consoles and Dashboards

Business Information Consoles are extensions of supervisor workstations designed to deliver business intelligence information to upper management. They typically consist of read-only representations of the same data obtained from HMI or Data Historian systems. In some cases, a Business Information Console is a physical console: a computer display connected to an HMI or Historian within the SCADA DMZ, but physically located elsewhere (such as an executive office or trading floor). In these cases, the physical display is remotely connected using a remote display or secure remote Keyboard Video Mouse (KVM) switching system. Business information may also be obtained by replicating HMI or Data Historian systems within the business network or by publishing exported information from these systems using an intermediary system, for example, exporting Data Historian information to a spreadsheet and then publishing that spreadsheet to a corporate information portal or intranet. Depending upon the sophistication of the Data Historian, this publishing model may be streamlined and automated. In any case, any published data should be access controlled, and any open communication path from SCADA systems to more openly accessible workstations or portals should be very carefully controlled, isolated, and monitored.

Other Assets

There are many other assets that may be connected to an industrial network other than PLCs, RTUs, HMIs, Historians, and various workstations. Devices such as printers and print servers may be connected to corporate networks, or they may connect directly to a control loop. Physical access control systems such as badge scanners and biometric readers may be used, and these devices may be networked (probably over Transmission Control Protocol/Internet Protocol [TCP/IP]).

Although this book does not attempt to cover every aspect of every device that may be present within an industrial network, it is important to recognize that every device has a potential impact to security and should be assessed if:

1. It is connected to a network of any kind (including wireless networks originating from the device itself, as with some printers).
2. It is capable of transporting data or files, such as removable media (mobile devices).

Even the most harmless seeming devices should be assessed. Check the documentation of devices to make sure that they do not have wireless capabilities, and if they do, secure or disable those features. Many commercially produced devices contain multipurpose microprocessors, which may contain radio or Wi-Fi antennae receivers or transmitters *even if the device is not intended for wireless communication.* This is because it is sometimes more cost-effective to use a commercial, off-the-shelf (COTS) microprocessor with unneeded capabilities; those capabilities may never be enabled by the manufacturer, but if the hardware exists it can be used as an attack vector by hackers.[6]

NETWORK ARCHITECTURES

As with all networks, industrial networks vary considerably. However, because many common functions within industrial systems vary widely—from automation systems, to supervisory and control systems, to business systems—there are natural demarcations within the network where these systems intersect. Table 5.1 indicates some of the major difference between these functional groups. The primary requirement of an industrial automation system is real-time operation and reliability, while the primary requirement of a business network might be high bandwidth and low operation costs. These requirements drive the use of real-time fieldbus protocols within control system processes and control loops, while business networks utilize fast, low-cost Ethernet networks and TCP/IP. SCADA systems sit between these two very different networks. In many ways, SCADA systems share the requirements of the control system itself—they need to be able to operate in real time, for example. However, they must also communicate with business systems over TCP/IP.

For this reason, a DMZ is recommended for supervisory systems. The SCADA DMZ sits between the operational and automation systems that they are supervising and controlling, and the business networks and business information systems. The DMZ is protected from both directions, using a firewall, intrusion detection and/or protection system, a data diode, or other perimeter defensive mechanism to prevent unauthorized traffic from crossing into or out of the DMZ. Logically, this creates three network areas: business, supervisory, and operations, as illustrated in Figure 5.5.

Table 5.1 Differences in Industrial Network Architectures by Function

Function	Industrial Automation	SCADA	Enterprise
Real-time operation	Critical	High	Best effort
Reliability requirements	Critical	High	Best effort
Bandwidth requirements	Low	Low	High
Protocols used	Fieldbus	Fieldbus, TCP/IP	TCP/IP

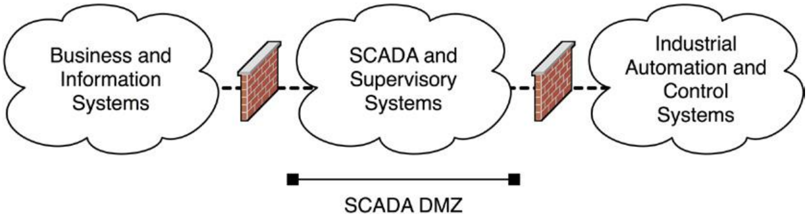

Business and Information Systems
SCADA and Supervisory Systems
Industrial Automation and Control Systems
SCADA DMZ

FIGURE 5.5

Functional Demarcation of Industrial Networks.

The operational and automation systems contain PLCs, RTUs, and IEDs, as well as HMI systems. The SCADA DMZ will also contain HMI systems, as well as Data Historians, MTUs (connecting to remote facilities), and Inter Control Center Protocol (ICCP) clients and servers for communicating with peer systems (see Chapter 4, "Industrial Network Protocols"). Business networks contain common computing and business systems, as well as supervisory workstations and replicated Data Historians.

Topologies Used

Industrial networks are typically very distributed and vary considerably in all aspects, including the link layer and network protocols used, as well as the topology. In the business networks, however, Ethernet and TCP/IP networks are ubiquitous, using a variety of star, tree, and even full-mesh topologies. The ubiquity of Ethernet and TCP/IP make it the "glue" that connects other SCADA and industrial control systems together. SCADA and industrial control system networks may utilize bus, ring, star, and tree topologies depending upon the specific type of control process that is in operation and the specific protocols that are used. For example, an automated control process to sanitize water may use a bus topology with the Modbus protocol, while another control process may use Profibus in a ring topology to control pumping or filtration systems (see Figure 5.6 for examples of topology use within and across an industrial network). The SCADA DMZ must communicate to both sides: on one side a number of industrial network protocols and on the other corporate Ethernet TCP/IP networks. As such, the SCADA DMZ will require protocol gateways to translate between the two environments (see Chapter 4, "Industrial Network Protocols"). These gateways can be standalone network devices, or they may be a built-in function of MTUs, HMIs, PLCs, or other industrial assets.

The specific topology used has little impact on the security of any particular network. More important is the boundary of a network area (which will help to determine how an attacker can migrate between systems) and the protocol(s) used within a network area (which will help to determine how a specific network area may be vulnerable). Although these areas are shown at a very high level in Figure 5.5, each network area that can be differentiated from its neighbors—ICCP interconnects versus OPC SCADA systems versus different control groups using DNP3, Modbus, etc.—can and should be isolated behind a secure periphery (see Chapter 7, "Establishing Secure Enclaves").

Special Topology Considerations

One area that deserves special consideration is the smart grid. As mentioned in Chapter 4, "Industrial Network Protocols," the smart grid is an extensive network providing advanced metering and communications capabilities to power distribution, and as such it is at once specific to the energy industry and yet also a concern

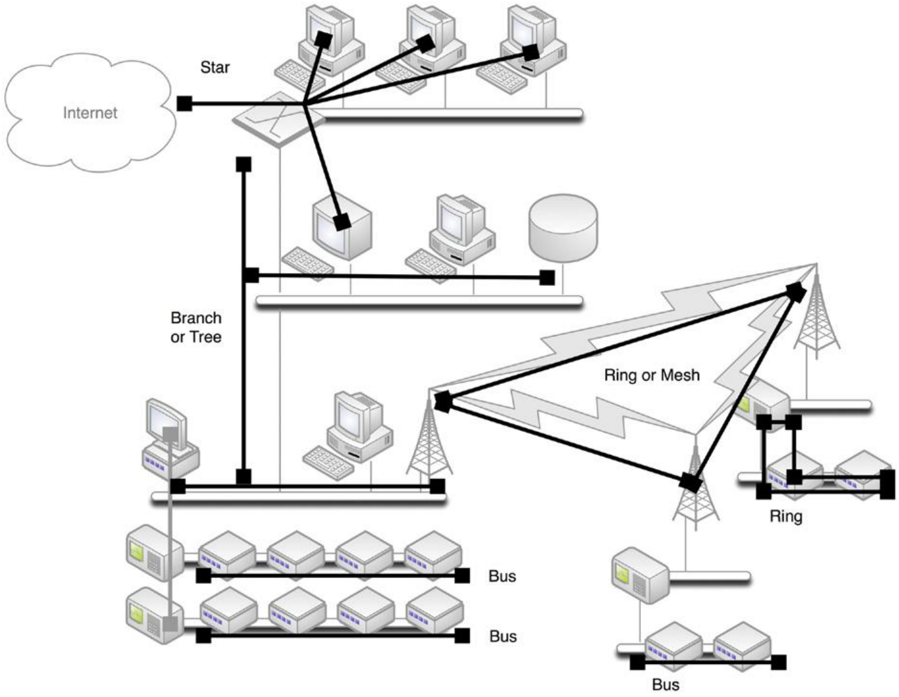


FIGURE 5.6

Common Topologies in Industrial Networks.

for any other industrial network that may connect to the smart grid as a client of the energy industry.

As with all networks, the "smart grid" also varies widely by deployment, and the topologies and protocols used will vary accordingly. However, there is one primary quality that is consistent across any smart grid deployment, and that is its scale and accessibility. As a distribution system designed to deliver power ubiquitously to residences, offices, storefronts, and all aspects of urban infrastructure, even small smart grid deployments create large numbers of nodes and network interconnections, often in hundreds of thousands or even in millions. The scale of a smart grid requires the use of some mechanism to "tier" or hierarchically distribute the nodes.

Represented in terms of an addressable attack surface, smart grids provide broad and easy access to a network that ultimately interconnects to our energy transmission and distribution infrastructure, as well as to many interconnected homes and businesses. In Figure 5.7, the attack surface is illustrated as being exponentially larger as we radiate outward from core energy generation through to the outer reaches of the smart grid.

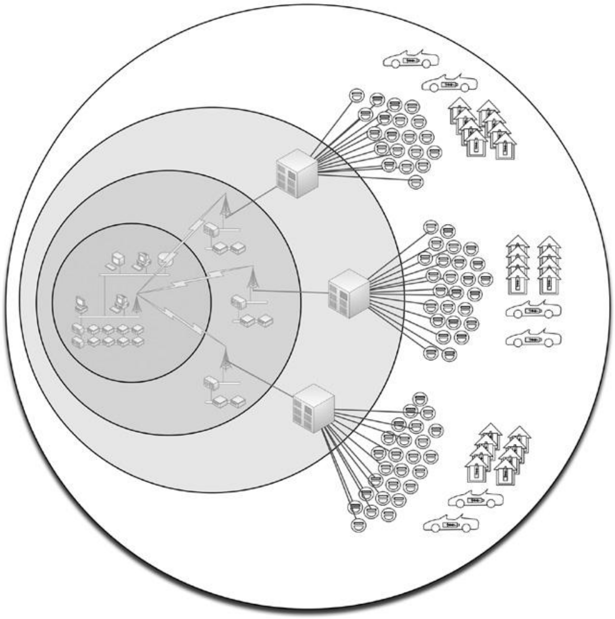


FIGURE 5.7

The Smart Grid Attack Surface Relative to Other Network Areas.

Scalability also plays a role in the development of smart grid devices, putting significant cost pressure on the end-node devices (Smart Meters). Any device deployed at such a large-scale needs to be as efficient to build, deploy, and operate as possible. Because of the costs and complexity of providing security assurance and testing in the various supply, design, and manufacturing stages of Smart Meter development, this business driver is a real concern. As pressures force costs down, there is an increased chance that some physical or network-based vulnerability will find its way into production, and therefore into one of the most easily reachable networks ever built.

CONTROL SYSTEM OPERATIONS

All of the industrial network protocols, devices, and topologies discussed up to this point are used to create and automate some industrial operations: refining oil, manufacturing a consumer product, filtering water, generating electricity, synthesizing and combining chemicals, etc. A typical industrial operation consists of several layers of programmed logic designed to manipulate mechanical controls in order to automate the operation. Each specific function is automated by a control loop, while multiple control loops may be combined or stacked together to automate larger processes.

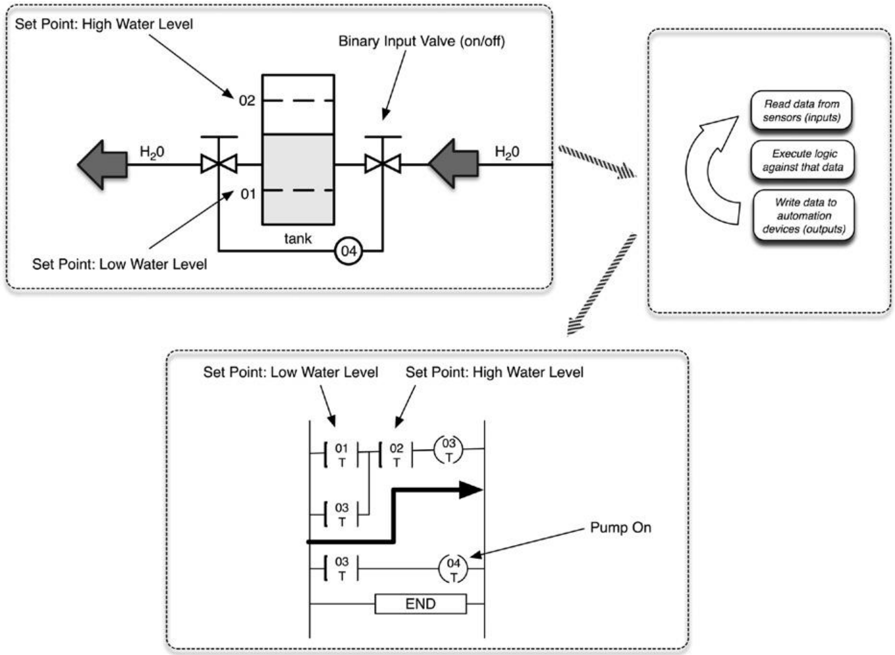

Set Point: High Water Level
Binary Input Valve (on/off)
02
H_20
H_20
01
tank
04
Set Point: Low Water Level
Read data from sensors (inputs)
Execute logic against that data
Write data to automation devices (outputs)
Set Point: Low Water Level
Set Point: High Water Level
01 T
02 T
03 T
03 T
Pump On
03 T
04 T
END

FIGURE 5.8

A Simplified Control Loop in the ON State, Showing the Applied Ladder Logic.

Control Loops

Industrial networks are made up of many specific automated processes, called control loops. The term "loop" derives from the ladder logic that is widely used in these systems: a controller device such as a PLC is programmed with specific logic; the PLC then cycles through its various inputs, applying the logic to adjust outputs or controls, in order to perform a specific function. This cycle or "loop" automates that function.

In a closed loop, the output of the process affects the inputs, fully automating the process. For example, a water heater is programmed to heat water to 90°C. An electric heater coil is then engaged to heat the water, and the water temperature is measured and fed back into the process; when 90°C is reached, the heater will turn off inputs from outside of the specific process. In an open loop, the output of the process does not affect the inputs, such as when the coil of a water heater is manually engaged, independent of the current water temperature. In other words, closed loops provide automated control whereas open loops provide manual control.

Control loops can be very simple, checking a single input, as illustrated in Figures 5.8 and 5.9. For example, a simple loop in an automated lighting process might check a single input (e.g., a light sensor to measure ambient light) and adjust a single output (e.g., the dimmer switch on a lamp). Very complex loops might use

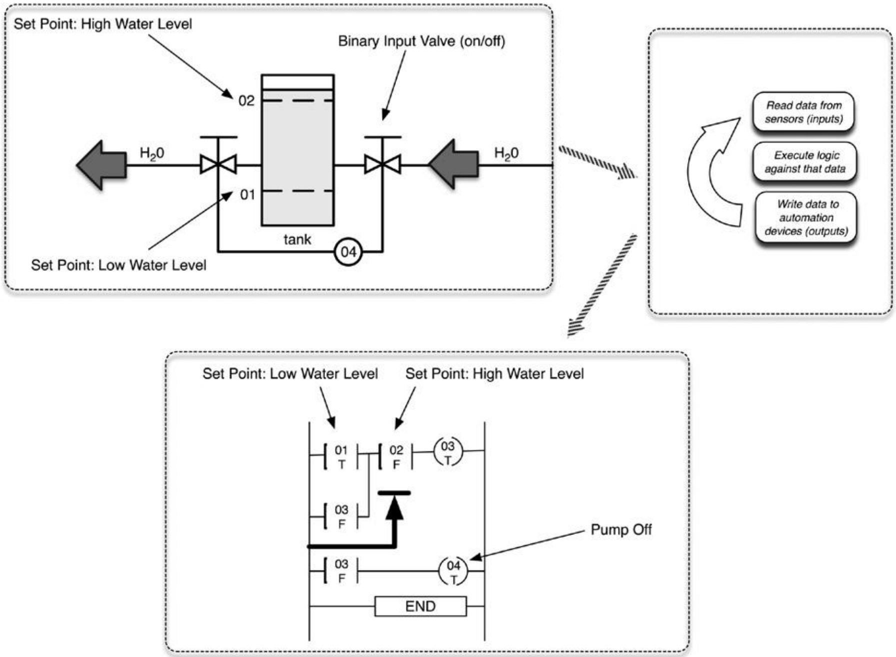

Set Point: High Water Level
Binary Input Valve (on/off)
02
H2O
H2O
01
tank
04
Set Point: Low Water Level
Read data from sensors (inputs)
Execute logic against that data
Write data to automation devices (outputs)
Set Point: Low Water Level
Set Point: High Water Level
01 T
02 F
03 T
03 F
Pump Off
03 F
04 T
END

FIGURE 5.9

A Simplified Control Loop in the OFF State, Showing the Applied Ladder Logic.

multiple inputs (e.g., pressure, volume, flow, and temperature sensors) and adjust multiple outputs (e.g., valves, pumps, heaters) to perform a function that is inherently more complex—in this case, maintaining a constant pressure in a dynamic fluid system.

Multiple control loops may be required to perform even more complex control processes. They may be controlled by a central HMI, or they may themselves be part of a larger control loop, acting as inputs or outputs into another level of logic, controlled by a master or central PLC.

Control Processes

A "control process" is a general term used to define larger automated processes within an industrial operation. Many control processes may be required to manufacture a product or to generate electricity, and each control process may consist of one or many control loops. For example, one process might be to inject an ingredient into a mixer, and that process may consist of a control loop that opens a valve in response to volume measurements within the mixer, temperature, and other

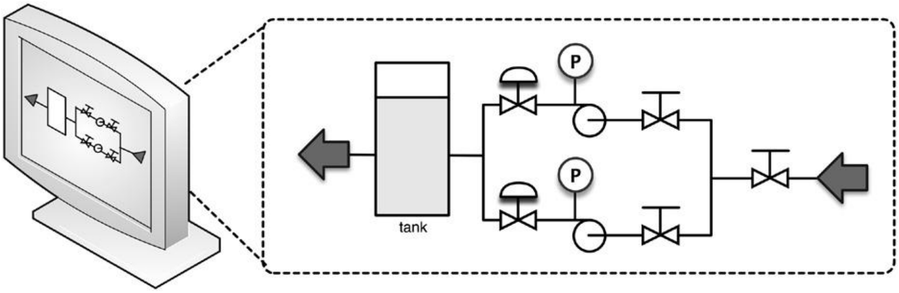


FIGURE 5.10

An HMI Displaying Current Operational Parameters.

conditions. Several such processes can automate the correct timing and combination of several ingredients, which in turn complete a larger process (to make a batter). The mixed batter might then be transported to other entirely separate control processes for baking, packaging, and labeling.

Each process is typically managed using an HMI, which is used to interact with that process. Typically, an HMI will provide relevant readings from one or more control loops, requiring communication to all daughter systems, typically PLCs or RTUs. Some HMIs may include readouts of sensors and other feedback mechanisms, as well as the activity of the PLCs, while others may issue direct control operations and provide controls to adjust the set points of the ongoing control process.

Again, an HMI may control a process consisting of many control loops; therefore, the HMI's network connectivity is typically heterogeneous: connecting over routable protocols (TCP/IP) as well as specialized SCADA and fieldbus protocols and other industrial network protocols to the various PLCs and RTUs that make up the individual loops. Because of this, HMIs are a common attack vector between the routable SCADA and business networks.

Feedback Loops

Every automated process relies on some degree of feedback both within a control loop and between a control loop or process and a human operator. Feedback is generally provided directly from the HMI used to control a specific process, as seen in Figure 5.10. Feedback may also be centralized across multiple processes, through the collection, analysis, and display of information from many systems. For example, a refinery may have several crude oil tanks, each used in a replicated control process. Information from each process can be collected and analyzed together to determine production averages, overages, and variations.

The centralized information management of an industrial control system is typically performed by one or more Data Historian systems. The process of removing data from the real-time environment of an industrial automation process and storing it over time is called "historizing" the data. Once historized, the information can be heavily analyzed, either directly from within the Data Historian or by using an external analysis tool such as a spreadsheet.

Specific control system elements may use their own Data Historian system to historize data locally. For example, an ABB 800xA control system may use the 800xA Information Management Historian, while an Emerson Ovation control system may use the Ovation Process Historian. The need for a common Data Historian that historizes all data across systems derives from the heterogeneous nature of many industrial operations, where different processes may utilize assets manufactured by different vendors, yet all processes need to be evaluated holistically in order to manage and fine-tune overall operations. In addition, there may be value in collecting information from other devices and systems within the industrial network, such as IT systems, HVAC systems, and Physical Security and Access Control systems. The shift from process-specific data historization to operation-wide business intelligence has led to the development of specialized features and functionality within Data Historians.

Business Information Management

Operational monitoring and analysis provides valuable information that can be used by business managers to fine-tune operations, improve efficiencies, minimize costs, and maximize profits. As such, there is a need for replication of the operational process data into the business network.

Supervisory data can be accessed using an HMI or a Data Historian, each of which presents its own security challenges. HMIs provide supervisory and control capabilities, meaning that an HMI user with the correct privileges can adjust parameters of control process (see "Control Process Management" on page 106). By placing an HMI outside of the SCADA DMZ, any firewalls, IDS/IPS, and other security monitoring devices that are in place will need to be configured to allow the communication of the HMI into and out of the SCADA DMZ, effectively reducing the strength of the security perimeter between the SCADA and business networks to user authentication only. That is, if a user account is compromised on the outside HMI system, it can be used to directly manipulate control process(es), without further validation from perimeter security devices.

The use of a Data Historian for business intelligence management presents a similar concern: the security perimeter must be configured to allow the communication between the Data Historian in the Business network and the various systems within the SCADA DMZ that need to be monitored. Unlike an HMI, a replicated Data Historian does not explicitly allow control of the process. Instead, the Data Historian provides a visual dashboard that can be configured to mimic the informational qualities and graphical representation of an HMI so that information about a process can be viewed in a familiar format.

TIP

Because the replication of Data Historian systems into the business network is for information purposes only, these systems can be effectively connected to the SCADA DMZ using a **unidirectional gateway** or data diode (see Chapter 7, "Establishing Secure Enclaves"). This preserves the security perimeter between business and supervisory networks, by allowing only outbound data communications. However, data outbound (from the SCADA DMZ to the business network) should still be secured using one or more security devices such as a firewall, IDS/IPS, or **application monitor**.

Data is collected by a Data Historian through a variety of methods including direct communication via industrial network protocols such as Modbus, Profibus, DNP3, and OPC (see Chapter 4, "Industrial Network Protocols"); via direct insertions in the Data Historian's database using Object Linking and Embedding Database (OLEDB), Open Database Connectivity (ODBC), Java Database Connectivity (JDBC), etc.; or using standard data exchange protocols such as the Simple Network Management Protocol (SNMP) and Syslog. Most Data Historians support multiple methods of data collection to support a variety of industrial applications. Once the information has been collected, it is stored within a database schema along with relevant metadata that helps to apply additional context to the data, such as batch numbers, shifts, or virtually anything (depending upon the Data Historian's available features and functionality).

Data Historian systems also provide access to historized data, typically through the same supported interfaces, with the possible addition of more ubiquitous protocols such as HTTP. Historized data can therefore be retrieved via direct SQL queries, via HTTP requests, via direct fieldbus protocol reads, or via other means. The data itself could be presented in almost any format, including binary files, XML, etc.

Historized data may be accessed directly via the Data Historian's operator console or could be integrated at almost any level into supplementary Business Intelligence Management systems. In some cases, the Data Historian may also be integrated with Security Information and Event Management systems (SIEMs), Network Management Systems (NMSs), and other network and/or security monitoring systems.

TIP

Unnecessary ports and services are a security concern on Data Historians, just as they are on any other industrial cyber asset. Contact your Data Historian vendor to determine how to disable unused data interfaces, in order to minimize the available attack surface of the Data Historian.

CONTROL PROCESS MANAGEMENT

A control process is initially established through the programming of PLCs to build a control loop. In a fully automated loop, the process is controlled entirely through the comparison of established set points against various inputs. In a water heater, a set point might be used to establish the high-temperature range of 90°C, and an input would take temperature measurements from a sensor within the water heater tank. The PLC's logic would then compare the input to the set point to determine whether the condition has been met (it is "true") or not (it is "false"), in this example disengaging or engaging the heater element, respectively.

When an operator manages a control process, he or she uses real-time information about the state of the process from an HMI to determine whether manual intervention is required (open loop) or adjustments to established set points are required (closed loop). The HMI facilitates both, by providing software controls to adjust the various set points of a control loop while also (typically) providing controls to directly affect the loop.

In the case of set point adjustments, the HMI software is used to write new set points in the programmable logic of the loop controller (the PLC or RTU). This might translate to function code 6 ("Write Single Register") in a Modbus system, although the specific protocol function is typically hidden from the operator; the HMI translates the function into human-readable controls presented within a graphical user interface (GUI), as represented in Figure 5.11.

In contrast, the HMI could also be used to override a specific process and force an output, for example, using function code 5 ("Write Single Coil") to write a single output to either the on ("true") or the off ("false") state.[7] Again, the specific function code used to write the output state is hidden from the operator.

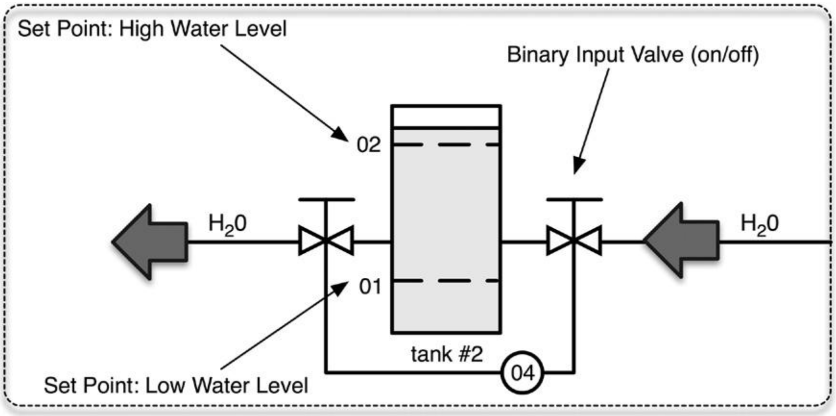

Set Point: High Water Level
Binary Input Valve (on/off)
02
H₂0
H₂0
01
tank #2
04
Set Point: Low Water Level

FIGURE 5.11

An HMI's GUI Representation of a Control Loop.

NOTE

The specific function codes used vary among industrial network protocols, and many protocols support vendor-proprietary codes. Although these protocols are discussed in Chapter 4, "Industrial Network Protocols," this book does not document protocol function codes. Refer to your user documentation for supported function codes.

This represents a significant security concern. If an attacker is able to successfully compromise the HMI, fully automated systems can be permanently altered through the manipulation of set points. For example, by changing the high-temperature set point to 100°C, the water in a water heater tank could boil, potentially increasing the pressure enough to rupture the tank. Direct changes to a process loop's output controls can also be forced by an attacker. In this example, the water heater's coil could be engaged manually by the attacker. In the case of Stuxnet, malware inserted into a PLC listened to a Profibus for an indication of a specific frequency converter operating at a specific frequency range. If those conditions were found, multiple commands are sent to the controller, alternating the operating frequency and essentially sabotaging the process.[8]

SMART GRID OPERATIONS

Smart grid operations consist of several overlapping functions, intercommunicating and interacting with each other. These include Customer Information systems, Billing Systems, Demand Response systems, Meter Data Management Systems, and Distribution management systems. These systems communicate with an Advanced Metering Infrastructure (AMI) Headend, which in turn feeds local distribution and metering, as shown in Figure 5.12. The AMI Headend will typically

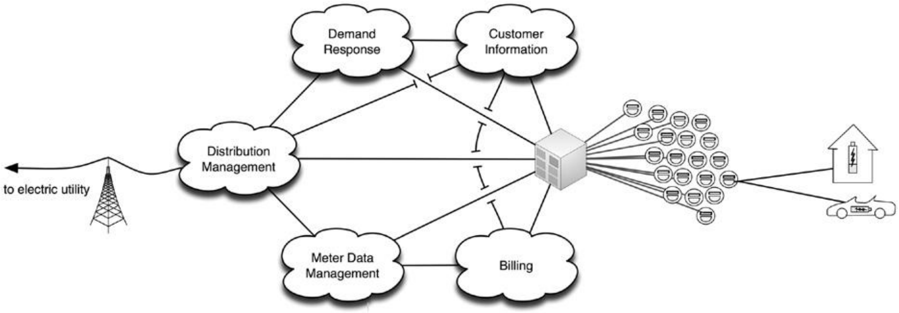


FIGURE 5.12

Components of a Typical Smart Grid Deployment.

connect to large numbers of Smart Meters, serving a neighborhood or urban district, which in turn connect to home or business networks.

The Customer Information system supports the business relationship between the utility and the customer, and may connect to both the customer premise (via customer service portals) as well as the utility back-end systems (e.g., corporate CRM). Meter Data Management systems store data, including usage statistics, energy generation fed back into the grid, Smart Meter device logs, and other meter information, from the Smart Meter. Demand Response systems connect to Distribution Management systems and Customer Information systems as well as the AMI Headend to manage system load based on consumer demand and other factors.[9]

Smart grid deployments consist of multiple AMI Headends, which may interconnect via a mesh network (where all Headends connect to all other Headends) or hierarchical network (where multiple Headends aggregate back to a common Headend), and may support hundreds of thousands or even millions of meters. All of this represents a very large and distributed network of intelligent end nodes (Smart Meters) that ultimately connects back to energy transmission and distribution.[10] The benefits of this allow for intelligent command and control of energy usage, distribution, and billing.[11] The disadvantage of such a system is that the same end-to-end command and control pathways could be exploited to attack one, any, or all of the connected systems. Some specific threats concerning smart grids include

- Bill Manipulation/Energy Theft—An attack initiated by an energy consumer with the goal of manipulating billing information to obtain free energy.[12]
- Unauthorized Access from Customer End Point—Use of an intelligent AMI end node (a Smart Meter or other connected device) to gain unauthorized access to the AMI communications network.[13]
- Interference with Utility Telecommunications—Use of unauthorized access to exploit AMI system interconnections in order to penetrate the bulk electric generation, transmission, and distribution system.[14]
- Mass Load Manipulation—The use of mass command and control to manipulate bulk power use, with the goal of adversely affecting the bulk electric grid.[15]
- Denial of Service—Using intelligent nodes to communicate to other nodes in a storm condition, with the goal of saturating communications channels and preventing the AMI from functioning as designed.

The AMI Headend is a prime target due to its central position in the smart grid: all end nodes, business systems, operational systems, and distributed control systems connect to (or through) the Headend. Compromise of the AMI Headend would provide a vector of attack to many systems. Similarly, if any other connected system were compromised the next hop would likely be to the Headend. Therefore, all inbound and outbound communications at the Headend should be carefully monitored and controlled (see Chapter 9, "Monitoring Enclaves").

SUMMARY

Industrial networks operate differently from enterprise networks and use specialized devices including IEDs, RTUs and/or PLCs, HMIs, Control System Assets, Supervisory Management Workstations, Data Historians, and Business Information Consoles or Dashboards. These devices utilize specialized protocols to provide the automation of control loops, which in turn make up larger industrial control processes. These automated control processes are managed and supervised by operators and managers within both SCADA and business network areas, which requires the sharing of information between two disparate systems with different security requirements.

This is exemplified in the smart grid, which shares information between multiple disparate systems, again across different networks each of which has its own security requirements. Unlike traditional industrial network systems, however, the smart grid represents a massive network with potentially millions of intelligent nodes, all of which communicate back to the energy provider, and possibly to other homes, businesses, or industrial facilities consuming power from the grid.

By understanding the assets, architectures, topologies, processes, and operations of industrial systems and smart grids, it is possible to examine them and perform a security assessment in order to identify prevalent attack vectors, or paths of entry that an attacker could use to exploit the industrial network.

ENDNOTES

1. PLCTutor.com, Ladder logic. <http://www.plctutor.com/relay-ladder-logic.html>, October 19, 2000 (cited: November 29, 2010).
2. P. Melore, PLC operations. <http://www.plcs.net/chapters/howworks4.htm>, (cited: November 29, 2010).
3. P. Melore, The guts inside. <http://www.plcs.net/chapters/parts3.htm>, (cited: November 29, 2010).
4. PLCTutor.com, PLC operations. <http://www.plctutor.com/plc-operations.html>, October, 19, 2000 (cited: November 29, 2010).
5. OSIsoft, OSIsoft company overview. <http://www.osisoft.com/company/company_overview.aspx>, 2010 (cited: November 29, 2010).
6. J. Larson, Idaho National Laboratories, Control systems at risk: sophisticated penetration testers show how to get through the defenses, in: Proc. 2009 SANS European SCADA and Process Control Security Summit, October, 2009.
7. The Modbus Organization, Modbus application protocol specification V1.1b, Modbus Organization, Inc. Hopkinton, MA, December 2006.
8. E. Chien, Symantec. Stuxnet: a breakthrough. <http://www.symantec.com/connect/blogs/stuxnet-breakthrough>, November, 2010 (cited: November 16, 2010).
9. G. Locke, US Department of Commerce and Patrick D. Gallagher, National Institute of Standards and Technology, Smart Grid Cyber Security Strategy and Recommendations, Draft NISTIR 7628, NIST Computer Security Resource Center, Gaithersburg, MD, February 2010.

10. UCA® International Users Group, AMI-SEC Task Force, AMI system security requirements, UCA, Raleigh, NC, Dec 17, 2008.
11. Ibid.
12. Raymond C. Parks, SANDIA Report SAND2007-7327, Advanced Metering Infrastructure Security Considerations, Sandia National Laboratories, Albuquerque, New Mexico and Livermore, California, November 2007.
13. Ibid.
14. Ibid.
15. Ibid.

CHAPTER

Vulnerability and Risk Assessment

6

INFORMATION IN THIS CHAPTER:

- Basic Hacking Techniques
- Accessing Industrial Networks
- Determining Vulnerabilities
- Vulnerability Management

In order to protect an industrial network from attack, it is important to understand how an attacker might approach an industrial network, gain access, and ultimately gain control. The basic hacking methodology and techniques of "identify, enumerate, and penetrate" are often discussed within the context of a typical Ethernet and Transmission Control Protocol/Internet Protocol (TCP/IP) enterprise network. However, in industrial networks, the methodology holds true but the techniques are subtly different. The entry points and attack vectors into an industrial system, the vulnerabilities of industrial systems, devices and protocols, and the exploits built against them must be understood before these systems can be effectively secured.

A discussion of vulnerabilities needs even greater consideration, as industrial networks are sensitive to traditional scanning techniques, and by their nature difficult to patch and reconfigure in order to minimize vulnerability. Therefore, it is important to understand where attacks may originate from, the paths or vectors that may be used, the targets, and their specific vulnerabilities.

Once an understanding of how a successful attack might occur is attained, the process of securing and isolating functional groups can be started. The information obtained from a strong vulnerability assessment and patch management strategy will facilitate the process of both defining and securing these functional groups (discussed in detail in Chapter 7, "Establishing Secure Enclaves").

BASIC HACKING TECHNIQUES

In order to defend a network against an attacker, it is important to be able to think like an attacker—and that means understanding the basics of hacking. The tools and techniques for hacking vary widely, although there are well-known and common methodologies that are often employed. By analyzing possible methods of gaining unauthorized entry into an industrial network, the perimeter can be strengthened accordingly. Note that these methods are all methods of attacking, and do not define the attack itself. That is, these steps define the process of how an attacker might gain entry into your network to deliver some sort of malicious payload such as a virus or malware; they do not define the payload itself. Protecting against the delivery of the payload comes first; protecting the users, services and hosts within your network from the payload (any malicious code, virus, bot, Trojan, etc.) comes after. The latter is discussed in Chapter 7, "Establishing Secure Enclaves".

The Attack Process

While there are numerous ways to penetrate a network, most involve some (if not all) of the following defined steps. They are performing some sort of reconnaissance activity to learn more about the target network; scanning the network to determine what the network looks like and what services are available for exploitation; enumeration, which is the process of identifying operating systems and users, including the determination of authentication credentials of users on the network; and then an attack—typically either an abrupt disruption such as a Denial of Service (DOS), or an attempt to penetrate and infect the network.[1] Examining these steps as a process, it can be seen that successfully penetrating a network is more difficult than simply disrupting it from the outside (see Figure 6.1). For example, if the goal of an attack is to disrupt an outward-facing service such as HTTP, an attack can be as simple as a targeted DOS against an organization's primary Internet access point. It is easy because the target is fully exposed, by design. Conversely, penetrating a network—either to disrupt an internal system that is not exposed, or to steal or alter information or other resources—requires that one or more layers of defense must be compromised. For the purposes of developing a best-practice defensive strategy, all industrial systems should be fully enclosed and protected within secure boundaries (see Chapter 7, "Establishing Secure Enclaves"); however, in reality many critical industrial networks are fully exposed (see the section "Targeting an Industrial Network").

Reconnaissance

The initial reconnaissance, or "foot printing" of a target, enables an attacker to understand the organization's security posture. By properly researching a target, an attacker can conclude information about the company and its employees, the company's Internet presence, internal and external networks and domains, and potential points of entry into those networks.[2]

Many readily available Internet services and search engines can be used for foot printing. Many companies openly publish information about partners, member organizations, and even employee blogs—any of which might equate to a way in. Partners typically interact with a company via a partner portal that may provide access to a greater range of information and services. Blog-friendly companies might implement special web services to aggregate employee RSS feeds.

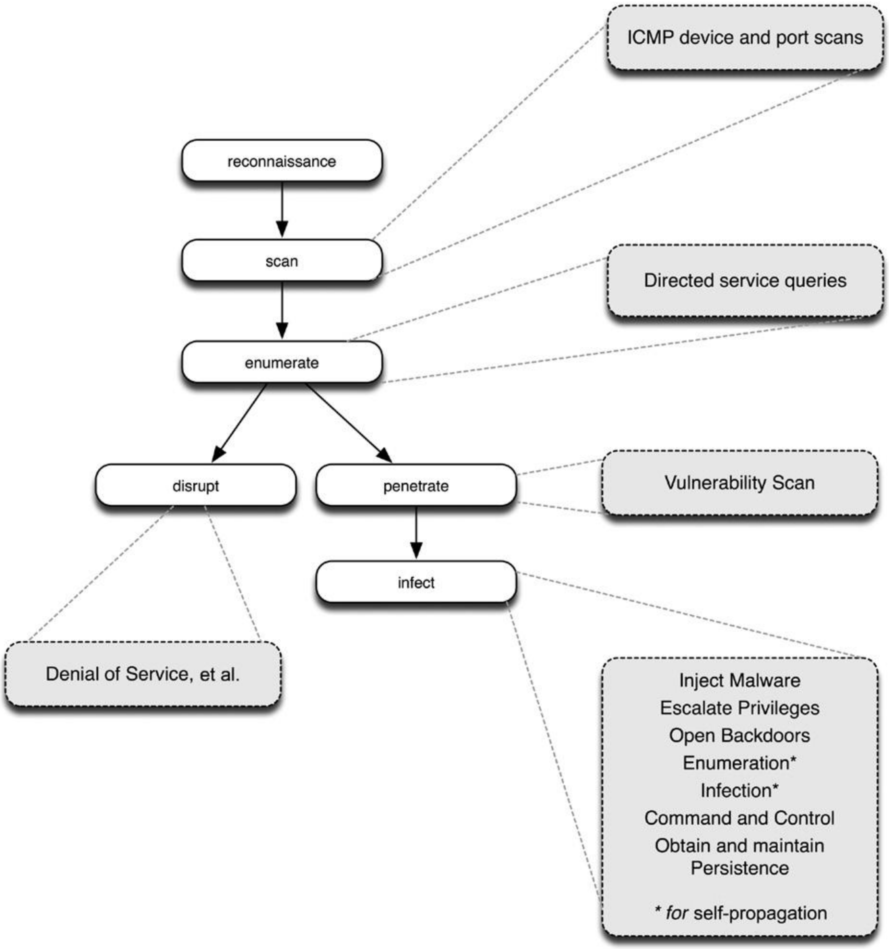


FIGURE 6.1

Basic Hacking Techniques in Traditional Enterprise Networks.

Any information that can be obtained is important because it could identify an entry point into the network, or it may be leveraged directly for social-engineering efforts, with results ranging from additional information gathering to targeted spear-phishing.

The tools available for network reconnaissance include: open-source aggregation services such as Maltego (www.paterva.com/web5/); social networking sites such as Facebook and LinkedIn; or more advanced tools such as the Social Engineer Toolkit (SET), a specialized tool set designed to "perform advanced attacks against the human element."[3]

For reconnaissance of network domains, IP space, extranets, and other essentials of network foot printing, domain queries, and lookups provide useful information about the available network(s) as well as specific devices within the network. DNS information can also be used to locate additional related domains (using point of contact searches), or simply provide a relevant user identity (including address and phone number) that might be leveraged as part of a social-engineering attack. Once a device within a network has been identified, it can be scrutinized to obtain more detail—such as using a command line tool like traceroute to learn about the routers, firewalls, and other devices that might sit along the path to the target.[4]

Scanning

Scanning a network typically begins with broad attempts to identify network devices and hosts using a ping sweep, and then leveraging additional capabilities of the Internet Control Message Protocol (ICMP) to determine additional information, such as the network mask (which allows you to derive subnet information), as well as open TCP and User Datagram Protocol (UDP) ports (which allows you to identify operating services, as most services map to well-known ports).[5]

Again, there are many tools that are available to facilitate network scanning, including tools like Fyodor's popular **Nmap** scanner, a free network scanning tool that combines ping sweeps, port scans, operating system detection, and service detection (by looking up well-known ports) and service version detection (by connecting to identified servers and obtaining reported version information). Nmap (www.nmap.org) is widely used; it is available on all major operating systems, and many minor ones including Amiga,[6] and is thoroughly documented in 16 languages.[7] Metasploit (www.metasploit.com) is another popular penetration testing tool that includes network scanning modules.

Enumeration

Enumeration refers to the process of identifying valid users and/or account credentials, as well as shared network resources that those user accounts might be able to access. The process involves establishing connections (or attempting to) and performing directed queries using tools like net view (for NT domains), or applications such as **finger** (for Unix user information) and rpcinfo (for identifying remote procedure calls that may be running).[8] The concept is that if an open entry point does not exist, a valid user account can be used to breach the network via a closed entry point. Once a username is known, passwords can be guessed (using knowledge gained during reconnaissance), brute-forced using password generators, or obtained from captured network traffic during the authentication process.

Once again, common tools such as Metasploit include ready-to-use enumeration modules. As of version 3.5.0, Metasploit included modules for the enumeration of MySQL and MS-SQL services, Oracle database users, DNS services, SAP BusinessObjects, Apache web servers, Wordpress blogs, Server Message Block (SMB) users and shares, Simple Mail Transfer Protocol (SMTP) users, Session Initiation Protocol (SIP) users, and even SMTP and Telnet authentications.[9]

Disruption, Infection and Persistence

The intentions of the attacker dictate what further actions might be taken. Does the attacker want to kill a service, or hide within the network to steal information over time?

If the goal is simply to disrupt a process or service, only understanding that service—for example, knowing an outward-facing IP address of a server—can be enough. There is no need to actually penetrate a web server to disrupt a company's ability to serve web pages, for example, because a simple DOS attack can be sufficient to break that service. In this very simple example the path to disruption shown in Figure 6.1 can be taken prior to enumeration. It may be necessary to first penetrate the network to a degree prior to breaking a service, however. For example, to attack a system or service that operates only internally (such as a supervisory system), it may first be necessary to penetrate one or more layers of defense (see the section "Disruption and Penetration of Industrial Networks").

Once a system has been accessed, it can be infected, referring to the successful installation and execution of some type of malware code on a device. The nature of the malware could be simple or complex, ranging from botnets to more advanced rootkits and/or memory-resident malware. Once a system is infected, the attacker can do almost anything, for example, opening backdoors, escalating privileges, spreading the infection to other devices, and establishing command and control functionality, etc.

Persistence means that the attacker's goal is to penetrate the network and lie hidden, listening, and waiting. Malware introduced as part of a persistent threat will attempt to remain hidden. As stated in Chapter 3, "Introduction to Industrial Network Security," this is one of the foundations of the Advanced Persistent Threat (APT). Persistence requires the following additional steps that must be taken:[10]

- Establish outbound connections or backdoors for command and control
- Continue to farm user credentials to access additional systems
- Escalate privileges and obtain data for exfiltration
- Maintain persistence by deleting logs and other evidence of the infection, rewriting legitimate services to hide command and control and other functionality, and evading detection through mutation

The last step sets requires that all steps in the attack process remain hidden. This could mean operating entirely in memory, or it could mean rewriting an existing service so that the outbound command and control can operate secretly within a legitimate service—something that is expected and will not raise a red flag if seen by a network security analyst (Chapter 9, "Monitoring Enclaves," discusses how to use security monitoring tools to help detect these covert communications). Persistent threats might also include several layers of infection, with dormant "backup" malware waiting to take over if the running exfiltration services are detected and removed.[11] In this way, the threat remains active in a new unknown form even after the original threat has been discovered and cleaned—perhaps even more so due to unwarranted complacency that can be felt after "eliminating" the original threat.

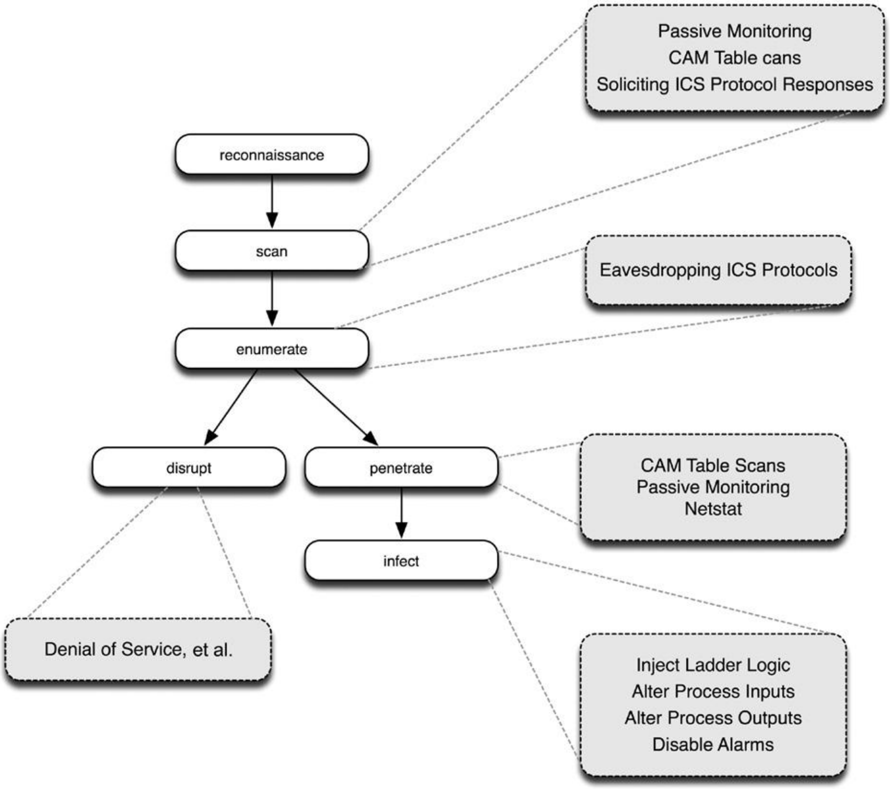


FIGURE 6.2

Basic Hacking Techniques Modified for Industrial Networks.

Targeting an Industrial Network

While the basic hacking methods discussed above apply to industrial networks, there are additional considerations—at all stages of an attack—when targeting a control system, as illustrated in Figure 6.2. Industrial control systems, because they utilize specialized systems and protocols, present new opportunities to an attacker. However, enterprise network hacking methods remain available as well, presenting a greater overall attack surface, which can be an advantage to an attacker.

Industrial networks can be difficult to attack if properly isolated, however. The establishment of secure zone or enclaves and a clear delineation between business, supervisory, and operation systems provides additional layers that an attacker must penetrate before reaching the most critical—and the most vulnerable system. Once the attacker has penetrated surrounding enclaves, they must discover the continued path into the control system.

Finally, in addition to normal user accounts and authentications, there are device Master/Slave relationships that can be discovered and manipulated to gain "authenticated" access to control system assets. In other words, there are Reconnais-sance, Scanning, and Enumeration techniques specific to Supervisory Control and Data Acquisition (SCADA) and distributed control systems (DCS) environments.

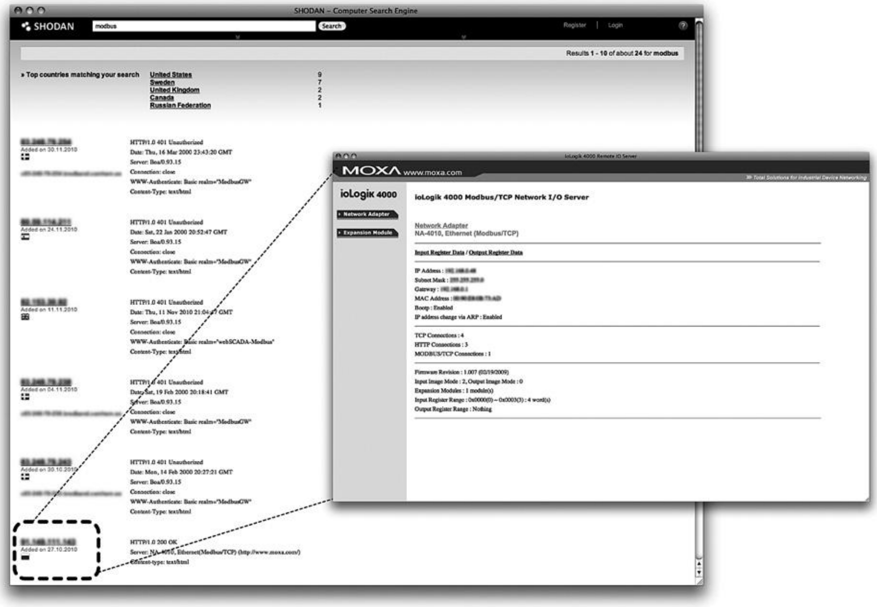
Industrial Reconnaissance

Industrial networks, protocols, assets, and systems are specialized. They are not commonly available, however, so an attacker intent on infiltrating an industrial system may focus reconnaissance efforts on information about the specific systems in use. As with enterprise hacking, reconnaissance can focus on public information about a company in order to learn the types of control system assets being used, the shift change schedule, and what other companies partner, service, or trade with the target company. Because many asset vendors use different and sometimes proprietary industrial protocols, knowing the specific assets used within the control system can indicate to an attacker what to look for in terms of systems, devices, and protocols.

Unfortunately, information can be obtained as easily as for any other network. Websites like the Sentient Hyper-Optimized Data Access Network (SHODAN) allow Internet-connected devices to be searched by port and protocol, country, and other filters. Any server, network switch or router, or other networked device using HTTP, FTP, SSH or Telnet is indexed by SHODAN (shodanhq.com). As a result, the site can easily identify devices utilizing SCADA protocols over any of these services (as seen in Figure 6.3).[12] This is an important step, as control systems are not easily procured, and therefore not easily reverse-engineered to find vulnerabilities. However, by understanding the control system devices in use the attacker is able to look for existing well-known vulnerabilities, or acquire device-specific research about the device through **backchannels** in order to determine vulnerabilities or backdoors. For example, in the case of Stuxnet, a hard-coded authentication process was used to gain access to the target Programmable Logic Controllers (PLCs). There has been much speculation in general about how a malware author might know this "insider information." It could have been someone with insider knowledge, access to a production DCS, or—depending upon the sophistication of the attacker—this level of industrial-grade information could have been obtained via the deployment of APTs that are intent on discovering control system schematics, source code, and other information. Black market information sources might already posses the information from existing APTs.

Scanning Industrial Networks

As mentioned in the section "Scanning," a network scan can identify hosts as well as the ports and services those hosts are using. In industrial networks, network scanning works in much the same way. The results of the scan can quickly identify SCADA and DCS communications, allowing the attacker to focus on these items. For example, a device found using port 502 is known to be using Modbus and is therefore very likely an HMI system or some supervisory workstation that is communicating with the HMI (see Table 6.1).



Results 1 - 10 of about 24 for modbus

» Top countries matching your search

United States	9
Sweden	7
United Kingdom	2
Canada	2
Russian Federation	1

```
[redacted]
Added on 30.11.2010
[redacted]
HTTP/1.0 401 Unauthorized
Date: Thu, 16 Mar 2000 23:43:20 GMT
Server: Boa/0.93.15
Connection: close
WWW-Authenticate: Basic realm="ModbusGW"
Content-Type: text/html

[redacted]
Added on 24.11.2010
HTTP/1.0 401 Unauthorized
Date: Sat, 22 Jan 2000 20:52:47 GMT
Server: Boa/0.93.15
Connection: close
WWW-Authenticate: Basic realm="ModbusGW"
Content-Type: text/html

[redacted]
Added on 11.11.2010
HTTP/1.0 401 Unauthorized
Date: Thu, 11 Nov 2010 21:04:47 GMT
Server: Boa/0.93.15
Connection: close
WWW-Authenticate: Basic realm="webSCADA-Modbus"
Content-Type: text/html

[redacted]
Added on 04.11.2010
[redacted]
HTTP/1.0 401 Unauthorized
Date: Sat, 19 Feb 2000 20:18:41 GMT
Server: Boa/0.93.15
Connection: close
WWW-Authenticate: Basic realm="ModbusGW"
Content-Type: text/html

[redacted]
Added on 30.10.2010
[redacted]
HTTP/1.0 401 Unauthorized
Date: Mon, 14 Feb 2000 20:27:21 GMT
Server: Boa/0.93.15
Connection: close
WWW-Authenticate: Basic realm="ModbusGW"
Content-Type: text/html

[redacted]
Added on 27.10.2010
HTTP/1.0 200 OK
Server: NA-4010, Ethernet(Modbus/TCP) (http://www.moxa.com/)
Content-type: text/html
```

ioLogik 4000 Remote IO Server
MOXA www.moxa.com
» Total Solutions for Industrial Device Networking
ioLogik 4000
Network Adapter
Expansion Module

ioLogik 4000 Modbus/TCP Network I/O Server

Network Adapter
NA-4010, Ethernet (Modbus/TCP)

Input Register Data / Output Register Data

IP Address : [redacted]
Subnet Mask : [redacted]
Gateway : [redacted]
MAC Address : [redacted]
Bootp : Enabled
IP address change via ARP : Enabled

TCP Connections : 4
HTTP Connections : 3
MODBUS/TCP Connections : 1

Firmware Revision : 1.007 (02/19/2009)
Input Image Mode : 2, Output Image Mode : 0
Expansion Modules : 1 module(s)
Input Register Range : 0x0000(0) ~ 0x0003(3) : 4 word(s)
Output Register Range : Nothing

FIGURE 6.3

SHODAN Screenshot with Drill-in to Target Modbus Device.

Table 6.1 SCADA and DCS Well-known Ports and Services

Port	Service
102	ICCP
502	Modbus TCP
530	RPC
593	HTTP RPC
2222	Ethernet/IP
4840	OPC UA
4843	OPC UA over TLS/SSL
19,999	DNP-Sec
20,000	DNP3
34,962–34,964	Profnet
34,980	EtherCAT
44,818	Ethernet/IP

However, there is a caveat when scanning industrial networks: because many industrial network protocols are extremely sensitive to latency and/or latency variation (jitter), a "hard scan" could actually cause the industrial network to fail. The lesson here is that, if the intention is disruption of services, all it takes is a simple network scan to achieve your goal. It is easy enough to scan through a firewall,[13] meaning that if real-time protocols are only protected by a firewall, they are highly prone to DOS attacks using very basic hacking techniques. If the goal of the attacker is more complex, network scans need to be performed more sensitively. This means using a "soft scan" versus large sweeps—for example, inspecting router tables or even sniffing traffic passively (see the section "Determining Vulnerabilities"). Successful scan results can quickly map known SCADA and DCS systems by filtering on the ports and services listed in Table 6.1.

CAUTION

Table 6.1 is only a partial list of some of the more common industrial ports and services. Many industrial devices utilize proprietary or unregistered port numbers. Always consult asset documentation to determine if special ports are used, and for what service, so that a comprehensive list of SCADA and DCS ports can be built.

Once a target system is identified, the scanning can continue—this time using the inherent functions of the industrial network protocols rather than commercial scanning tools. The following examples will obtain device information from industrial networks:

- Sniffing Ethernet/IP traffic to obtain Critical Infrastructure Protection (CIP) device identifiers and attributes
- Sweeping DNP3 requests that solicit a response (e.g., `REQUEST_LINK_STATUS`) to discover DNP3 slave addresses[14]
- Capture an EtherCAT frame or a SERCOS III Master Data Telegram to obtain all slave devices and time synchronization information

Each industrial protocol utilizes its own function codes, and some proprietary function codes may be used on specific devices (necessitating some reconnaissance). For example, on SERCOS (Serial Real-time Communications System) networks, all slave devices can be easily identified via a short packet capture, as all communications to all nodes are packaged into a common message. Obtaining a SERCOS Master Data Telegram may also allow an attacker to identify designated time slots for communications to a specific device, as well as what cycles are available for open TCP/IP use.

Again, Stuxnet has exemplified the disruptive potential of this type of scanning. Once Stuxnet establishes itself in the logic of a target PLC, it listens to Profibus communications using these same techniques in order to detect specific frequency settings of specific frequency controllers. Stuxnet then manipulates the PLC outputs in order to sabotage the process.[15]

NOTE

Scanning an industrial network can effectively act as a DOS attack. Because many industrial protocols are real time, and the processes tightly synchronized, the introduction of additional packets into a real-time network can be disruptive. This means that an attacker who does not want to immediately disrupt an industrial network may scan quietly: performing low-and-slow scans, or using the "scan and spread" methodology of Stuxnet, where the malware crawls invasively but quietly through the network examining its surroundings at it goes in order to find target systems, rather than performing fast and loud sweeps.

Enumerating Industrial Networks

Because many industrial systems are Windows based, Windows user accounts can be enumerated in standard ways and be fully applicable to industrial operations. This is especially true of OPC Classic systems that rely on Windows OLE and DCOM, where obtaining authentication to the host allows essentially full control over the OPC environment. However, despite the lack of authentication in the underlying network protocols, enumeration can extend to specific identities and roles within a control system. Useful authentication information might include the following:

- HMI users
- ICCP server credentials (the bilateral table)
- Master node addresses (for any Master/Slave industrial protocol)
- Historian database authentication

Accessing an HMI would allow direct control of that HMI's managed processes, and/or theft of information about that process. Obtaining ICCP server credentials would allow an ICCP server to be spoofed, enabling either steal or manipulate information being transmitted between control centers. If a Master node address is obtained, the attacker could spoof that Master node, obtaining control over a control loop without requiring access to the HMI (the attacker could inject function codes directly on the bus at this point).

In many cases, user roles and privileges are stored centrally, in an **LDAP** or an **Active Directory** server, providing attackers with a clear target for enumeration attempts. This is why it is important to functionally isolate both physical devices and logical services into established enclaves. This is also why NIST 800-82 (Guide to Industrial Control Systems [ICS] Security) recommends using a combination of account verification methods, including "something known (e.g., PIN number or password), something possessed (e.g., key, dongle, smart card), something you are such as a biological characteristic (e.g., fingerprint, retinal signature), a location (e.g., Global Positioning System [GPS] location access), the time when a request is made, or a combination of these attributes."[16] By abstracting authentication across multiple physical and digital attributes, enumeration becomes very difficult and can be effectively limited. That is, it may be possible to obtain a username or even a password, but full authentication remains elusive.

Disruption and Penetration of Industrial Networks

As mentioned in the section "Scanning Industrial Networks," simply scanning an industrial network can be enough to disrupt it: many of the industrial protocols are sensitive enough that the introduction of a significant amount of unexpected traffic will result in protocol failure, and an effective DOS condition. This is a significant concern: it is possible to perform a network scan through a firewall,[17] and even easier to packet-flood through an open port on a firewall. That is, by identifying what traffic is allowed through the firewall, the attacker can then use allowed traffic to scan through the firewall, using a soft scan for true reconnaissance or a hard scan for disruption of service. If the firewall is well configured, a scan may not be possible, but all firewalls will allow some traffic through. By spoofing legitimate communications, abnormal amounts of traffic can be injected into a control network, causing a DOS.

TIP

The more strictly defined, a firewall's rules are, the more difficult it will be to identify and spoof "allowed" traffic. When configuring a firewall, always begin with "deny all," and then configure "allow" rules according to the following guidelines:

1. Only "allow" traffic that is absolutely necessary for the operation of the devices specific to the enclave that is being secured. If too many "allow" conditions are needed, consider breaking the enclave into additional functional groups.
2. Always explicitly define the source and destination IP address and port. That is, use "allow from [a specific IP address and port] to [a specific IP address and port]" rather than "allow all from [a specific IP address]."
3. Especially for critical control systems, supplement the firewall with an IDS/IPS, application monitor or similar device to detect hidden channels or exploits inside of allowed protocols. An IDS/IPS with rate-based anomaly detection, for example, could detect and prevent a potentially disruptive packet-flood condition.

If the goal is not disruption, but penetration (and possibly persistence), we can again look to Stuxnet as an example of the types of infiltration techniques that might be deployed. Stuxnet employs a variety of scanning and mutation mechanisms for industrial network penetration. By looking for specific conditions in the network environment before performing additional tasks, Stuxnet is able to distribute itself widely despite maintaining a very focused target. Stuxnet reacts to its environment as follows:

- In the "enterprise phase" it looks for a target HMI before mutating to penetrate the HMI.
- In the "industrial phase" it infects the HMI, looks for target PLCs, and then again mutates, injecting malware into the PLC.
- In the "operational phase" it uses the PLC to look for certain IEDs operating with specific parameters before injecting commands to sabotage the process.

This simplified description of how Stuxnet operates highlights the following important considerations:

- The initial attack vectors leverage common enterprise hacking techniques.
- A compromised SCADA or DCS asset can be used to detect and penetrate additional industrial systems.
- Even "nonroutable" systems (such as a fieldbus consisting of PLCs and IEDs) are susceptible to infection, and can be used to penetrate even further into the industrial process.

Threat Agents

Industrial networks are different in many ways from enterprise networks, and as such they attract a different type of attacker. Who would want to deliberately breach an industrial network? An attack on an industrial network is not difficult, although it does require specialized knowledge and therefore the attacker will require more resources. There also is not an obvious benefit to attacking most industrial networks, as there might be from a financial services network or a retailer. The bad news is that there are attackers, and they fall into several distinct classes. The Government Accountability Office (GAO) has identified several classes of attackers, or "threat agents" in DHS parlance. They include the following[18]:

- General hackers looking for individual prestige (referred to as "attackers" by the GAO, although the term "attacker" is used more generally in this book to refer to any threat)
- Botnet operators and spammers, identified as having the same skill sets as general hackers, but with the intent of further distributing spambots and other botnets
- Criminal groups looking to obtain money, either as ransom against the threat of a disruptive attack, or through direct monetary theft
- Insiders, including disgruntled employees, technology or business partners, or recently terminated employees or partners
- Phishers, treating industrial networks as another population of users susceptible to identity theft
- Spyware and malware authors
- Foreign intelligence services, as part of information gathering and espionage efforts
- Terrorists, seeking to destroy or disrupt critical infrastructures
- Industrial spies, who—much like foreign intelligence services—perform espionage, but for the purpose of acquiring intellectual property from competitive companies and/or nations

At first, the list of identified threat agents does not stand apart from what might be expected from an enterprise network attacker. However, the last three (foreign intelligence agencies, terrorists, and industrial spies) quickly put the risk of industrial network attack in perspective. Mapping the GAO's classifications to the likelihood and sophistication of an attack (as depicted in Chapter 2, "About Industrial

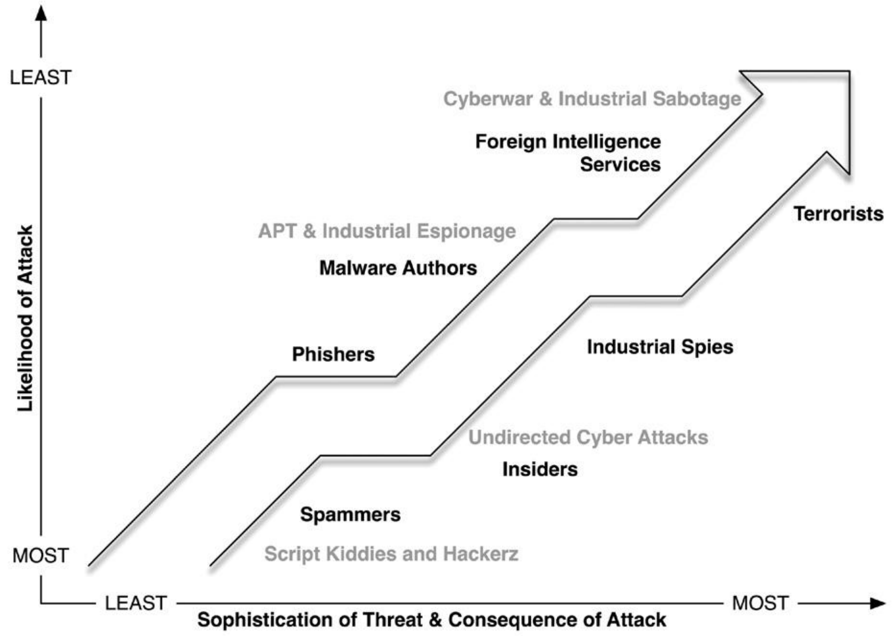

LEAST
Likelihood of Attack
MOST
LEAST
Sophistication of Threat & Consequence of Attack
MOST
Cyberwar & Industrial Sabotage
Foreign Intelligence Services
Terrorists
APT & Industrial Espionage
Malware Authors
Phishers
Industrial Spies
Undirected Cyber Attacks
Insiders
Spammers
Script Kiddies and Hackerz

FIGURE 6.4

Threat Agents versus Likelihood and Sophistication of Attack.

Networks," Figure 2.2), we can now also see the consequences of such an attack, as illustrated in Figure 6.4.

ACCESSING INDUSTRIAL NETWORKS

In an ideal situation, the most critical systems of an industrial network are well protected behind strong layered defenses, making a basic attack difficult if not impossible. In reality, there are many entry points or attack vectors into industrial systems. The most obvious is via the business network, but in many cases there are entry points directly into "secure" SCADA demilitarized zones, and even into the control systems' networks themselves. As shown in Figure 6.5, direct entry is possible into almost any network zone, and from there an attacker might easily penetrate into other areas.

The whole concept of "perimeter defense" only works if an attacker actually needs to break through that defense. If a perimeter can be bypassed, it adds little value to the overall security of the network. An example of this is implementing a data diode to make inbound data communications impossible, and then allowing uncontrolled use of removable media within the secure enclave. Securing an

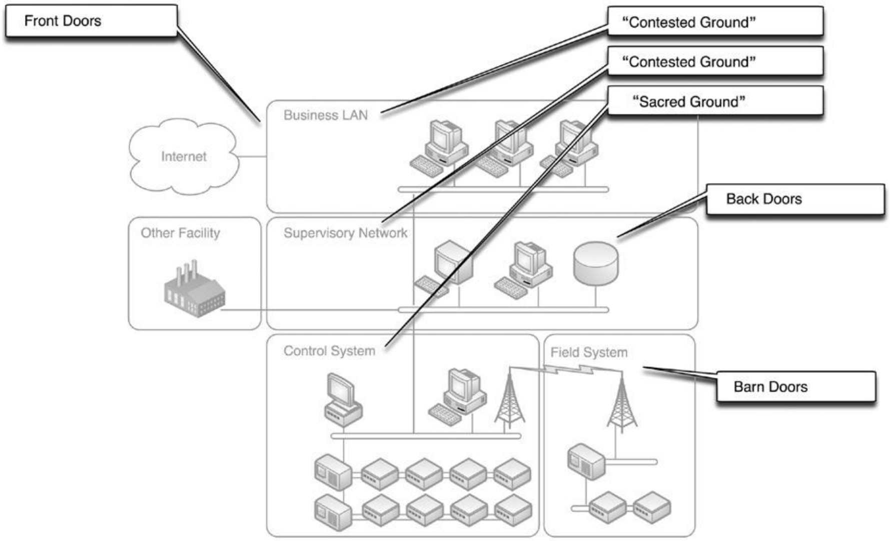


FIGURE 6.5

Entry Points into Industrial Networks.

industrial network therefore begins with understanding how an attack might gain entry and then putting the necessary defenses in place. To complicate matters, the clean delineation of industrial networks into three well-defined enclaves (Business, SCADA, and Control) is overly simplified. In real industrial networks, there are many—potentially dozens—of enclaves that need to be isolated and protected; if any one system is vulnerable, and there is lack of separation between that system and others, then an attack vector exists.

Further complication is introduced by smart grids. As discussed in Chapter 5, "How Industrial Networks Operate," the smart grid presents an unprecedentedly large attack surface. The sheer scale of a smart grid deployment makes these networks easily accessible, both physically and digitally. In addition, a smart grid communicates with several systems that are (hopefully) logically separated into distinct enclaves. A breach of the smart grid, therefore, can potentially open many entry points into different areas on the industrial network.

The Business Network

Unlike SCADA networks and control systems, business networks rely on connectivity. Out of necessity, they allow more open communications, both inbound and outbound, in order to support the various normal functions of business: sales, marketing, accounting, administrative, and other business functions all rely upon networked systems, many of which utilize web applications or even cloud computing resources. Therefore, unlike industrial network enclaves, the business network

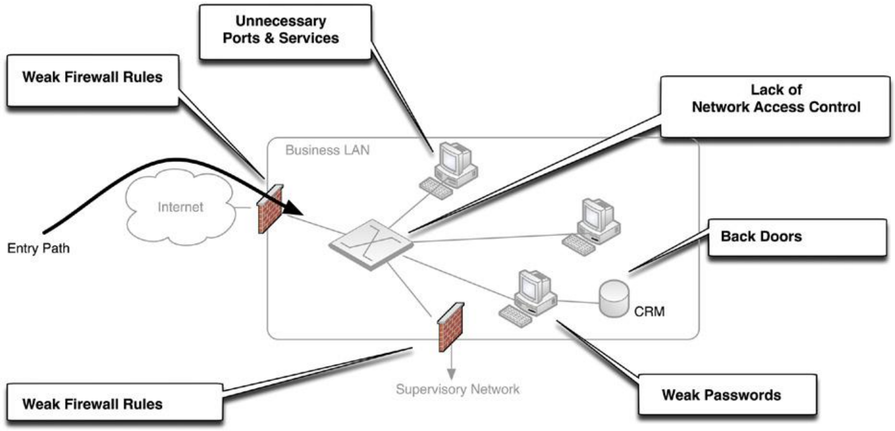


FIGURE 6.6

Entry Points into the Business Network.

must allow connectivity to the Internet. Also, unlike industrial control systems, the network-, user-, and application-behavior patterns in an enterprise vary widely.

Unfortunately, this makes the business network highly exposed to attack. If vulnerabilities exist, it is a simple matter to discover and exploit them remotely. Even more unfortunate is that the business systems also rely upon information from SCADA and DCS systems, and as such these services are sometimes made accessible from within the business network. When the business network is inevitably compromised, it becomes a primary attack vector into these supervisory and control systems. The business network, therefore, should be considered "contested ground," and when assessing the security of industrial networks it should be treated as if it were already compromised.

The primary entry point to the business network, as shown in Figure 6.6, is from the Internet. According to the SANS Institute, the leading methods of entry continues to be unpatched client software and vulnerable Internet-facing web servers, reinforcing the trend toward application-based vulnerabilities (vs. previous trends of operating system and protocol stack vulnerabilities).[19] With the previously described attack methodologies in mind (see the section "Basic Hacking Techniques"), steps should be taken to mitigate these vulnerabilities by limiting the attacker's ability to identify and enumerate important systems and services. This involves the following steps:

- Properly controlling and monitoring inbound and outbound traffic
- Disabling all unnecessary ports and services
- Enforcing strong authentication and access control policies
- Minimizing backdoor access through application vulnerability assessment and patching
- Controlling the use of removable media, remote access and other rogue I/O where control is possible (and establishing security awareness and policies where it is not)

Weaknesses in any of these areas will increase the attack surface of the network, as can also be seen in Figure 6.6. However, the highly dynamic and interconnected nature of modern business practices requires a more open approach to information sharing (see the section "Poorly Configured Firewalls").

It should be noted that access to the business network from the SCADA DMZ is possible. Although no substantial evidence of past attacks has been observed along this path,[20] all security demarcations should enforce communications in both directions.

The SCADA DMZ

Where the business network is "contested ground," the SCADA network is the "middle ground," the demilitarized zone between the business and process control systems. Here, the network is bridged between standard business applications and services and specialized process control applications and services, as well as between common Ethernet TCP/IP networks and either Ethernet or serial fieldbus networks. If we assume that the business network has been compromised (for the sake of establishing a strong security profile), the same vulnerabilities and entry points exist. As can be seen in Figure 6.7, different systems are in use, but they present many of the same inbound vectors for an attacker.

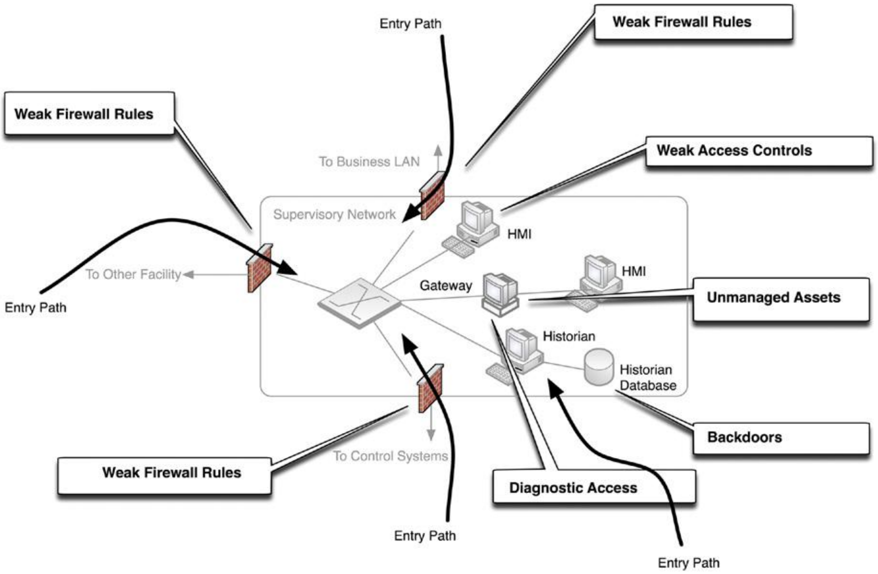


FIGURE 6.7

Entry Points into the SCADA DMZ.

Again, weak firewall rules and access control provide a primary entry point from the business network into the SCADA DMZ. Legitimate reasons for allowing communications through the firewall exist, and these can introduce entry points into industrial network enclaves, via the business network. However, there are also inbound entry paths that lead directly into the supervisory enclave(s), bypassing the business network. These entry points include the following:

- Inter-control center communications over ICCP
- Remote access connections to field stations
- Connections to the Control System
- Diagnostic access to SCADA devices via dial-up or remote access

Each entry path requires security demarcation, using (at a minimum) a properly configured firewall. See Chapter 7, "Establishing Secure Enclaves," for recommendations on how to provide strong perimeter defenses.

The Control System

If the business network is "contested ground" and the SCADA DMZ is "middle ground" than the Control System is "sacred ground." Within the context of industrial network security, the control system represents the ultimate target: the devices and systems that actually control the industrial process which needs to be protected. Theoretically, the Control System has very limited access, but in practice there are multiple points of entry. These include not only the obvious path from the SCADA DMZ, but also direct entry paths from wireless and diagnostics systems, as shown in Figure 6.8.

Because the control system is likely the target of more sophisticated attacks from more dangerous threat agents, minimizing any direct entry path into these networks is critical. Again, direct vectors of attack can be minimized through the isolation of critical functional groups, secured by strong defense-in-depth practices (see Chapter 7, "Establishing Secure Enclaves").

Common Vulnerabilities

In addition to identifying paths into an industrial network, understanding the vulnerabilities associated with industrial network systems will give the attacker—or the defender—an advantage. While many vulnerabilities are derived from software bugs in applications or network protocol stacks, other vulnerabilities are derived from weak security practices and policies, poor network design, and other easily addressable factors. Some of these vulnerabilities, including poor firewall configurations, weak authentication, unmanaged and/or insecure network access, and remote access vulnerabilities, can be addressed easily.

Poorly Configured Firewalls

Poorly configured firewalls represent the largest vulnerability to any network, because firewalls are still relied upon as the primary (and in some cases the only)

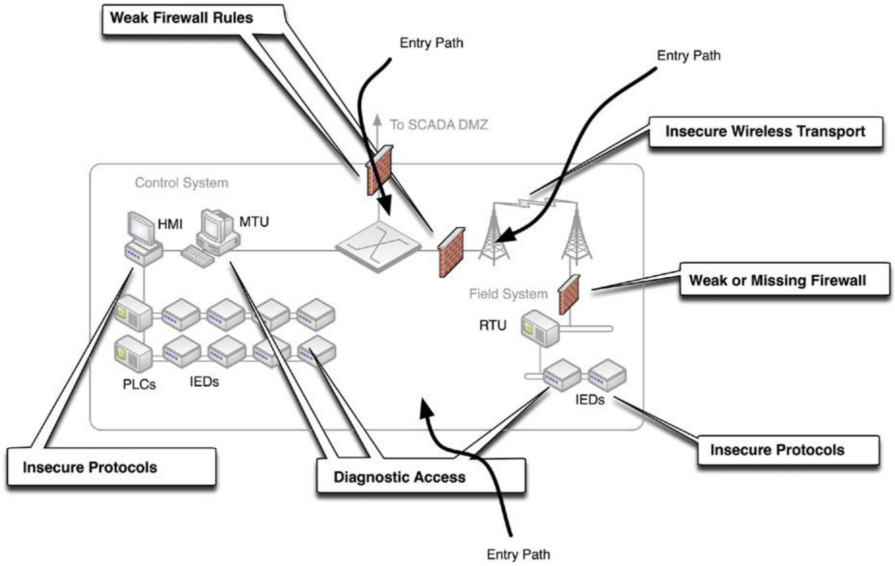


FIGURE 6.8

Entry Points into the Control System.

method of cyber defense. Firewall misconfigurations derive from a number of factors, including a combination of legitimate business requirements and increasingly complex firewall policies that are required to accommodate them. In addition, poor network housekeeping can result in open policies allowing network traffic types that are no longer in use (but could still be leveraged by an attacker to gain entry into the network). They are also derived from a lack of understanding; for example, many firewalls include only inbound traffic policies and allow any outbound traffic free reign, ignoring the very real possibility that an attacker could be residing inside the network perimeter attempting to communicate outwards—such as a command and control agent of APT, looking to exfiltrate information about control system functions, configurations, and operations.[21]

Table 6.2 highlights in general terms some common firewall misconfigurations, indicates how they introduce risk, and gives recommendations on how to remediate the issue(s). Note that more detailed recommendations for security configurations are provided in Chapter 7, "Establishing Secure Enclaves."

Unnecessary Ports and Services

NERC CIP and other regulations dictate the disclosure of all open ports and services and all cyber assets, and recommend that any unused or unnecessary ports and services be closed or disabled. Looking at the intricacies of firewall configurations, the reason is clear. Every open port and service represents a network communication path that could be used maliciously, and as such the number of open ports

Table 6.2 Common Firewall Policies with Recommendations

Firewall Rule	Business Justification	Issues	Recommendations
“Permit All” policies for traffic from the business network to the Internet	Employees in the business network require access to outside world for a variety of business functions	Unless source and destination addresses and ports are explicitly defined, devices such as printers or rogue PCs are exposed to the Internet over port 80. There is also no control in place to prevent a rogue web server inside the business network from initiating or accepting HTTP connections	Specify source and destination IP address details on all firewall rules. Consider web content firewall to restrict access to websites that have inappropriate content and/or malware
Allow SCADA protocols to pass from the SCADA DMZ to the business network	Executive access to an HMI console or other SCADA system is desired for business planning, strategy, trading or other legitimate business purpose	Unless source and destination addresses and ports are explicitly defined, the SCADA protocols will be exposed to all nodes in the corporate LAN. If the business firewall does not inherently or explicitly deny SCADA traffic to the Internet, the HMI is directly accessible from the Internet	Inherently disallow SCADA protocols across all firewalls, and only explicitly allow them where needed, using `allow` rules that specify both source and destination address and port. SCADA and DCS protocols can be detected using network monitoring as well for added situational awareness
Allow business services from the business network to pass into the Control System	These types of policies are usually caused by oversight. In some cases they are the result of unintended bi-directionality caused by other rules designed to allow traffic from the SCADA Network into the business network	Any business service (such as web, e-mail, file sharing, etc.) available to SCADA or Control Systems provides an open entry path for an attacker	Always define firewall policies to enforce traffic in both directions, and inherently deny all traffic into critical network areas. Common business services can be detected using network monitoring as well for added situational awareness
Allow contiguous service use across multiple enclaves	To facilitate business operations across functional units, communications, fire sharing and other services may be allowed within Business, SCADA and Control System networks	If allowed contiguously, an exploited service will provide a clear path through any and all additional enclaves, potentially bypassing their perimeter defenses entirely	Configure enclave boundaries using “disjointed” policies—that is, do not allow the same service to operate within any two adjacent enclaves[a]

[a] *Internet Engineering Lab (IEL) Group for Advanced Information Technology (GAIT), NISCC Good Practice Guide on Firewall Deployment for SCADA and Process Control Networks, version 1.4, National Infrastructure Security Coordination Center (NISCC), February 15, 2005.*

and services correlates directly to the complexity of firewall, IDS/IPS, and other security device configurations. As complexity increases, so does the likelihood of a misconfiguration that will result in network vulnerability.

Application Backdoors

Many business applications and control system applications utilize a database, and databases remain highly vulnerable to attack unless properly configured and secured. Using SQL injection techniques, an attacker can gain control of the database, and from there he or she can migrate control to the database server or the application server, such as a company's customer management and billing systems or a SCADA system's Historian system. Most control systems not only log activity to a Historian, but the data is also replicated across the perimeter between the SCADA DMZ and business network.[22] These attacks are especially dangerous because the databases (and the applications which they backend) are explicitly reachable by authoritative systems and users. Many OPC systems store local connection databases which might enable an attacker to quickly migrate into several systems inside a SCADA DMZ or even the Control System itself.

Ideally, databases should only be accessible by the application which it backends, using strong user authentication. Hard-coded internal authentication, such as was exploited by Stuxnet, should be avoided. To avoid these vulnerabilities, limit connectivity between both users and applications, as well as applications and databases. If data from a SCADA database need to be replicated to a business network, use hard defenses such as unidirectional data diodes, which allow traffic to move only in one direction (out of the SCADA DMZ) and prevent a compromised database on the business network from letting an attacker migrate into the SCADA DMZ.

Asset Controls

Even a data diode can be compromised physically. A user can walk into a secure facility and plug in a USB drive or iPod, well inside the secure perimeter established by firewalls, IDS/IPS devices, or diodes. It is therefore important to control the assets themselves. All assets should require authentication, and all unnecessary services should be disabled. This includes device mounting services, file sharing services, and other commonly overlooked computer functions (Stuxnet initially spread via removable USB drives, although network-based vulnerabilities and even print-spooler vulnerabilities were utilized.

TIP

Though not covered in this book, asset procurement and supply chain assurance should also be considered to reduce the risk of new equipment, pre-infected with malware, from being procured and deployed.

Wi-Fi Access

Wireless network represents significant risk due to multiple vulnerabilities associated with network architecture, access control configurations, and even component-level vulnerabilities. This is because wireless networks are accessible easily from the air: they possess antennae capable of receiving wireless transmissions. The tools required to detect a Wi-Fi network are readily available, and there is an extensive list of tools available to hack into wireless networks. These tools include discovery, mapping, traffic analysis ("sniffing"), client evaluators, wireless frame generators, and encryption cracking tools designed to break wireless authentication.[23]

Even disabled wireless systems can present a challenge. A researcher from the Idaho National Labs presented methods for hacking into devices at the component level, utilizing Wi-Fi antennae that were built into the microprocessor silicon but never enabled by the manufacturer.[24] Also, beyond the control of most security administrators are exploitations of the normal functions of Wi-Fi access points. Jamming a Wi-Fi signal using off-the-shelf components could cause an access point to enter a state of continuous reconnection attempts. The result, if sensitive Ethernet fieldbus protocols are in use, could be a processwide DOS.[25]

The best mitigating factor for wireless access is to avoid it where possible, and to thoroughly isolate and secure access points wherever they are located. That is, assume that Wi-Fi access will be successfully detected and that authentication will be cracked, and treat the access point as contested ground, separating it from other networks.

Remote Access, VPNs and Mobile Apps

Remote access, if not implemented properly, can represent significant risk. Especially when considering the potential threat agents in an industrial network attack scenario, the remote end of the connection simply cannot be trusted. A laptop with a VPN client can be stolen. Extranets can be easily breached. Mobile SCADA and control applications for smart phones and other mobile devices expound the problem even further.

To avoid inherent vulnerabilities with remote access, always treat the access point (whether a VPN client, application server, etc.) as if it were directly exposed to the Internet, and do not terminate remote access directly into critical networks. Also, when performing vulnerability analysis and penetration tests, make sure to include all remote interfaces into the network.

Diagnostic access/Dial-up Access/Field Access

Some remote access mechanisms do terminate into critical systems—sometimes directly into a critical cyber asset. Because industrial networks are built around reliability, and control systems are sometimes difficult to access physically (especially remote stations or plants), many vendors of industrial products include mechanisms to access field devices remotely. If a system has a remote dial-up modem interface for diagnostics or for backup communication, an attacker can potentially bypass every single defensive measure in place and call into the asset directly. A simple war-dialing attack (where an attacker rapidly dials every combination of possible telephone numbers using specialized software) will quickly locate modems, putting any asset with exposed dial-up interfaces at extreme risk.

Remember that many industrial assets do not require authentication, and for those that do, it is still common to find default passwords in use in many field devices.[26] Securing long-distance communication facilities can be difficult. Lines are typically terminated at a PBX or other telecommunications demarcation device, which is most often under the authority of the corporate IT department. Special care should be taken for remote access over these lines: ideally, all field access of this sort would operate over private lines that terminate in a controlled corporate environment, limiting access to devices located within a central, controlled environment.

The Smart Grid

Once again, the smart grid represents new inbound attack vectors into industrial systems. The nature of Smart Metering technology and its close marriage to power transmission and distribution essentially turn the entire T&D infrastructure into a potential entry point into the utility's network. These entry points are numerous and are therefore not depicted in Figures 6.6, 6.7, or 6.8. However, they include access via the T&D communications infrastructure, via customer service and billing apps within the business network, via generation and usage applications within the SCADA network, and so on. Almost every physical system, network, and application is tied in some way to the smart grid, requiring strong security on these systems to prevent an attacker using them as an inbound vector (see the section "Smart Grid Operations" in Chapter 5, "How Industrial Networks Operate").

DETERMINING VULNERABILITIES

Understanding what are the entry points through which an attacker might attempt to penetrate an industrial network is one thing; understanding how the attack might succeed is another. As discussed in the section "Basic Hacking Techniques," an attacker will attempt to gather information and scan networks for entry points. Next comes enumeration, where an attacker will attempt to obtain user and authentication information; and then penetration. During both of these steps an attacker will look for vulnerabilities that can be exploited in order to obtain access and control. Many vulnerabilities are well known, and therefore vulnerability management and patch management are closely related (see the section "Vulnerability Management"). For example, dozens of common industrial network vulnerabilities are identified in NIST SP 800-82. While the list is extensive—ranging from procedural vulnerabilities cause by inadequate security policies (e.g., lack of training, awareness, documented security procedures, etc.), to platform configuration vulnerabilities (i.e., unpatched systems, default configuration use, missing weak or default password use, etc.), software vulnerabilities (inherent vulnerabilities such as buffer overflows, the use of vulnerable protocols or services such as DCOM, insufficient logging, etc.), the lack of sufficient malware protection, improperly configured networks (weak or missing network security controls, lack of encryption, lack of access control, lack of redundancy, etc.), inadequate network authentication, inadequate or missing perimeter protection, and lack of communication integrity checking[27]—most are directly addressable by implementing the security best practices described within this book. For others, diligent vulnerability scanning and management is required (see Table 6.3).

Mapping suggested actions to the ICS vulnerabilities identified by NIST shows the necessity of performing thorough vulnerability assessments and isolating the detected ports and services into clearly defined and secure enclaves.

NOTE

Table 6.3 represents only a subset of the identified ICS vulnerabilities identified by NIST. For a full list of identified ICS vulnerabilities, please refer to the latest version of NIST Special Publication 800-82, available at http://csrc.nist.gov/publications/PubsSPs.html.

Why Vulnerability Assessment Is Important

Apart from the known architecture, procedural or configuration vulnerabilities that can be easily addressed in advance through security best practices, to manage vulnerabilities it is first necessary to assess the network to determine what specific vulnerabilities exist, and where. This requires either extensive manual assessments of each network asset or the use of an automated vulnerability assessment (VA) tool. Automated tools greatly facilitate the process of vulnerability assessment through a combination of various network and asset scans and the correlation of the results to known vulnerabilities, typically from a central data repository of known threats that is maintained by the VA tool.

The scanning of networks and assets in an attempt to find unpatched vulnerabilities is also one of the initial steps of an attack, as discussed in the section "Basic Hacking Techniques." VA scanning tools are often used by hackers, and in fact the very popular exploit framework Metasploit (www.metasploit.com) was acquired by vulnerability assessment vendor Rapid7 in October, 2009 specifically to "[bring] richer exploitability data to customers and partners . . . enabling them to better identify, prioritize and remediate critical security issues."[28]

When a vulnerability is found, it must be remediated. Remediation can be achieved by applying a patch to the vulnerable system (if one is available) to eliminate the vulnerability, adjusting the system's network or operating configuration to functionally eliminate the vulnerability, or by eliminating the vulnerable system altogether.[29] If the system is critical and cannot be remediated, the option of last resort is to isolate the vulnerable system in order to effectively quarantine the vulnerability. This process of vulnerability management and remediation is covered in more detail in the section "Vulnerability Management."

Table 6.3 Common Vulnerabilities Defined by SP 800-82[a] with Security Recommendations

Policy and Procedure Vulnerabilities	**Suggested Actions**
Inadequate security architecture and design	Identify functional groups and separate into security enclaves with appropriate security measures (see Chapter 7, "Establishing Secure Enclaves")
Few or no security audits on the ICS	Implement centralized log and event collection and reporting (see Chapter 9, "Monitoring Enclaves")
Lack of ICS specific configuration change management	Monitor control processes for changes using Historians and/or importing Historian data into other security tools such as SIEM (see Chapter 9, "Monitoring Enclaves")
Platform Configuration Vulnerabilities	**Suggested Actions**
OS and vendor software patches may not be developed until significantly after security vulnerabilities are found OS and application security patches are not maintained OS and application security patches are implemented without exhaustive testing	Perform regular **VA** scans and follow Vulnerability Management best practices (see the sections " Vulnerability Assessment in Industrial Networks" and "Vulnerability Management")
Default Configuration Vulnerabilities	**Suggested Actions**
Lack of adequate password policy No password used Password disclosure Password guessing	Monitor network for weak passwords using content inspection such as content aware IDS, content firewalls, ADM or transaction monitors (see Chapter 9, "Monitoring Enclaves"). Also, look for password weakness and/or password stagnation during the vulnerability assessment process (see the section "Vulnerability Scanning for Configuration Assurance")
Platform Software Vulnerabilities	**Suggested Actions**
Buffer overflow Installed security capabilities not enabled by default Mishandling of undefined, poorly defined, or "illegal" conditions	Perform regular VA scans and follow Vulnerability Management best practices (see the sections " Vulnerability Assessment in Industrial Networks" and "Vulnerability Management")
OLE for Process Control (OPC) relies on Remote Procedure Call (RPC) and Distributed Component Object Model (DCOM) Use of insecure industry-wide ICS protocols	Monitor the network for SCADA and DCS protocols outside of their defined enclaves (see Chapter 9, "Monitoring Enclaves")
Unneeded services running	Vulnerability scans can identify open ports and services on a host, while network flow monitoring can identify services in use in the network (see section " Vulnerability Assessment in Industrial Networks" and Chapter 9, "Monitoring Enclaves")

(Continued)

Table 6.3 (Continued)

Platform Software Vulnerabilities	Suggested Actions
Intrusion detection/prevention software not installed	Implement host- and/or network-based intrusion detection and prevention systems. At a minimum, Host IDS (**HIDS**) should be used on all critical assets, while Network IDS (**NIDS**) should be used at all enclave perimeters (see Chapter 7, "Establishing Secure Enclaves")
Logs not maintained	All logs should be centrally collected and managed using a **Log Management** and/or SIEM system. For those devices that are incapable of producing logs, compensating measures should be implemented, such as passive network monitoring to produce logs in proxy, or the use of Historian or other information stores in lieu of activity logs (see the section "Vulnerability Assessment in Industrial Networks" and Chapter 9, "Monitoring Enclaves")
Incidents are not detected	Implement a central event analysis and correlation system to detect and document potential incidents (see Chapter 9, "Monitoring Enclaves")
Platform Malware Protection Vulnerabilities	**Suggested Actions**
Malware protection software not installed	Implement Host and/or Network Anti-Malware (see Chapter 7, "Establishing Secure Enclaves")
Malware protection software or definitions not current	Include Anti-Malware definitions in the Vulnerability Management process (see "Vulnerability Management"). Consider a Host and/or Network Anti-Malware solution based on whitelisting rather than signature-based detection (see Chapter 7, "Establishing Secure Enclaves")
Malware protection software implemented without exhaustive testing	Include Anti-Malware definitions in the Vulnerability Management process (see section "Vulnerability Management").
Network Configuration Vulnerabilities	**Suggested Actions**
Weak network security architecture	Identify functional groups and separate into security enclaves with appropriate security measures (see Chapter 7, "Establishing Secure Enclaves")
Data flow controls not employed	Implement firewalls, IDS/IPS, routing and/or ACL controls to enforce data flow control. Analyze network flows using Network Management System (NMS), Network Behavior Anomaly Detection (NBAD), SIEM, or other tools to monitor data flow violations (see Chapter 7, "Establishing Secure Enclaves")

(Continued)

Table 6.3 (Continued)

Network Perimeter Vulnerabilities	Suggested Actions
No security perimeter defined Firewalls non-existent or improperly configured	Identify functional groups and separate into security enclaves with appropriate security measures at the perimeter of each enclave (see Chapter 7, "Establishing Secure Enclaves")
Control networks used for non-control traffic	Monitor within control network enclaves for non-DCS traffic. Also implement exception rules at other demarcations to prevent non-control traffic that originates from a control network. For example, adding `Deny %DCS_IP_ADDRS to Any` rules at the Internet firewall (see Chapter 7, "Establishing Secure Enclaves")
Control network services not within the control network	Monitor for SCADA and DCS outside of their respective enclaves, and deny this traffic at enclave perimeters (see Chapter 7, "Establishing Secure Enclaves," and Chapter 9 "Monitoring Enclaves")
Inadequate firewall and router logs	Enable logging on all networked devices, and implement centralized log collection and analysis (see Chapter 9 "Monitoring Enclaves")
No security monitoring on the ICS network	Implement SCADA- and DCS-capable NIDS or other network probe(s) in control system networks to act as a passive ICS security monitoring device Ideally use these devices as part of a larger security information monitoring solution such as a SIEM
Communication Vulnerabilities	**Suggested Actions**
Authentication of users, data or devices is substandard or nonexistent	Implement centralized authentication management. Ideally, monitor user activity using a monitoring tool with **IAM** context
Lack of integrity checking for communications	Implement whitelisting technology to assure only validated communications are initiatedImplement application-layer content inspection devices or similar technology to verify the integrity of all communications to assure that "validated" applications and protocols have not been modified on the wire
Wireless Connection Vulnerabilities	**Suggested Actions**
Inadequate authentication between clients and access points	Secure all entry points from wireless, dial-up and other remote access methods via isolated enclaves (see Chapter 7, "Establishing Secure Enclaves")

[a] *K. Stouffer, J. Falco, K. Scarfone, National Institute of Standards and Technology, Special Publication 800-82 (Final Public Draft), Guide to Industrial Control Systems (ICS) Security, September, 2008.*

Before managing vulnerabilities, however, they must first be identified. However, as discussed earlier under "Basic Hacking Techniques," scanning industrial networks can be detrimental to operations, as many industrial network protocols are sensitive to latency, jitter, and/or unexpected network utilization (see Chapter 4, "Industrial Network protocols"). Because of this it is necessary to understand how to perform a non-disruptive scan, when to scan, and where.

Vulnerability Assessment in Industrial Networks

Because many industrial network protocols are sensitive to latency and/or latency variation (jitter), any large amount of introduced network traffic could cause a failure. Aggressive scans that actively probe many addresses and ports in rapid succession could overwhelm an industrial network. Likewise, broad scans that attempt to penetrate through multiple network hops could unintentionally reach and disrupt a process control system. Because of this, vulnerability scans should be performed differently depending upon what the network it is scanning: in a non-production test environment, both hard and soft scan methods should be used; while in production environments, only soft scan methods should be used. By subjecting an industrial system to aggressive scanning techniques in a test environment, it can be determined if the scan will actually disrupt the network, and if so how. Any failure should be treated as a vulnerability, and addressed according to established vulnerability management procedures (see section "Vulnerability Management"). Obviously, a scan that causes a failure will also fail to determine all open vulnerabilities, so soft scans should also be performed—first in the test environment, and then in production environments during scheduled maintenance windows.

Soft scans replace hard techniques with somewhat more intrusive techniques that require much less network overhead, effectively reaching the same goal (to find open ports and services, patch levels, etc.) but with less risk to sensitive networks. If attempting to identify the devices on a network, instead of a ping sweep using a tool such as Nmap, which can potentially introduce significant traffic to the network, the Content-Addressable Memory (CAM) tables of Ethernet switches or the router tables of Ethernet routers can be examined to both identify valid network addresses and also determine where they are located in the network. If a device within the network is compromised, it can also be used to sniff live network traffic, extracting source and destination address information of all nodes that are actively communicating on the network.[30]

To identify open ports and services, soft techniques again require gaining access to a system or device and looking at local network statistics. In place of "outside in" sweeps that query device after device to identify available services, the "inside out" approach utilizes tools such as netstat to list all network connections from a single host. Netstat is able to show the protocol, the source address, destination address, and the state of the connection. By looking at connection information, a list of active systems and the services that they are using can be compiled; the more host connection tables that can be examined, the more complete the list will be.[31] Many commercial VA tools support soft scanning in this manner, using valid user credentials to legitimately authenticate to systems and gather network and service information from the host operating system.

Mapping a list of discovered network devices and their open ports and services to known vulnerabilities can be done passively as well. In place of actively scanning for vulnerabilities using a VA tool, a passive scan can be performed using the same tool. Passive VA scanning sniffs network traffic (or in some tools, accepts pcap files from previously captured traffic) to perform "soft" detection of devices, ports and services, and then reconstructs the communication so that a banner analysis can be performed.[32] Banner analysis—the process of examining information banners in application services—can identify version information and other specifications about a particular service that is in use. Mapping this information to a database of known vulnerabilities, such as the Common Vulnerabilities and Exposures (CVE) list published by the MITRE Corporation, identifies known vulnerabilities that are present in the network. This mapping can be automated, as part of a passive VA scan, or it can be performed manually.[33]

When scanning an active control system, safe scan methods should always be used. If manual control of the system is possible, personnel capable of performing manual control must be present during the security testing.[34] Ideally, a test system will be available for "hard testing" in a non-production environment. If possible, build out a test system, or plan to perform hard tests against production systems during scheduled maintenance windows or other periods of downtime. This will allow you to identify and document the detrimental effects of a hard scan, so that compensating measures can be introduced. For example, it is possible to block ping sweeps and may be possible to implement traffic throttling controls or to provide additional network separation that can protect against more sophisticated and aggressive scans.

Vulnerability Scanning for Configuration Assurance

Assessing vulnerabilities and documenting them is a compliance requirement of NERC CIP-007, specifically requirement R2 (which requires the identification of open ports and services) and R8 (which requires a vulnerability assessment). However, the vulnerability assessment process can be used for additional compliance purposes as well. One example is the Bandolier project by Digital Bond. Bandolier is a Department of Energy–funded project designed to audit configuration files in control system environments. Written for the Nessus vulnerability scanner, Bandolier helps to validate the security configuration of a workstation, identify ports and services, detect default passwords and accounts and verify password requirements, and audit the status of malware prevention software.[35] Bandolier is an example of a soft vulnerability scan configuration: it uses valid credentials to authenticate to each system and then uses local tools such as netstat to obtain port and services information, capture application banners, etc.[36] Many of these functions could be performed manually as well, although the use of an automated tool can simplify the process and help to eliminate oversight. In some cases additional monitoring tools can be implemented to help with configuration assurance. For example, a passive application monitor that is capable of decoding and analyzing application sessions could compare authentication credentials against known defaults, determining weak passwords, etc.

Configuration assurance can also be addressed through configuration management, which allows a proven configuration to be validated and then managed, notifying the administrator when a configuration change occurs. Configuration control is an important part of Vulnerability Management and can help maintain a compliant configuration once one has been established (see the section "Vulnerability Management").

NOTE

There are many commercial vulnerability assessment tools. Although Bandolier was written for the Nessus scanner, optimized vulnerability scanning options for industrial networks may be supported by other scanners as well. When evaluating vulnerability assessment products or services, ask the vendor or consultant about how the specific tool(s) support SCADA and DCS protocols, if there are specialized scan profiles, audit files, or other customizations that support safe scanning within industrial control systems.

Where to Perform VA Scans

Especially in industrial networks that consist of (sometimes) clearly defined security enclaves and (most of the times) overlapping or poorly defined security enclaves, knowing where to perform a scan is as important as knowing how to perform a scan. A rule of thumb is whenever attempting to functionally isolate a group of devices or services, first quantify that group into a defined enclave (see Chapter 7, "Establishing Secure Enclaves") and then perform a penetration test against it. This requires first scanning the network immediately outside of the enclave, as well as scanning from within the enclave to ensure that there are no outbound vulnerabilities.

The reason for this is simple: in a network that contains nested security enclaves, some enclaves may not be immediately vulnerable from every other enclave. For example, it may not be possible to exploit an RTU directly from the Internet. However, if an attacker breaches the business network and then the SCADA DMZ, there may be new vulnerabilities present. Penetration testing should first be performed from the outermost entry points (such as an Internet firewall or a VPN gateway). Next, the tests should be repeated from "one level in," assuming that the first line of defense has been compromised.

This accomplishes two things: first, it helps to detect all addressable vulnerabilities within nested enclaves; second, it helps to identify ports and services that are in use so that "disjointing" of communications policies can be implemented where possible. For example, in a control system enclave that utilizes Modbus TCP, port 502 will be open to support this protocol. To protect that traffic, port 502 should be filtered in the next outermost enclave (in this example, it is perhaps the SCADA DMZ), if possible. In our example, the SCADA DMZ probably requires Modbus

TCP connections as well, and so port 502 should be filtered at the outer boundary of the DMZ instead. In this way, the protocol is "disjointed," meaning there is no single path from the Internet to the control system over port 502.

Cyber Security Evaluation Tool

The Cyber Security Evaluation Tool (CSET) is a software tool available from the National Cyber Security Division of the DHS. CSET walks through a control system vulnerability assessment process, and then produces guidelines for vulnerability remediation. The CSET recommendations are made after comparing the assessment against relevant NIST, ISO, NERC, and other standards. Note that the CSET assessment is a user survey designed to evaluate policies against recommendations, and is not a "vulnerability assessment" as described in the section "Vulnerability Assessment in Industrial Networks."[37]

VULNERABILITY MANAGEMENT

Once a vulnerability has been identified, it needs to be eliminated. Depending upon the nature of the vulnerability, it may be addressable via a software patch or a configuration adjustment, or it may need to be removed.[38] Patching and configuration-based remediation require careful management: patches, while technically a "fix" for a known vulnerability, represent new code and should be carefully tested prior to installation. Configuration changes may also fix a vulnerability, but may impact other systems or devices and therefore require controlled testing as well. Configuration adjustments could include direct adjustments to the vulnerable system itself, such as disabling vulnerable or unused services or modifying user privileges, or configuration adjustments to outside systems, such as modifying firewall or IPS policies or limiting access via router Access Control Lists (ACLs) to block the vulnerable service.[39] Obtaining patches can also prove challenging: downloading up-to-date software from vendor websites requires connectivity to the Internet, yet with proper security enclaves in place, the systems requiring an upgrade should not have this connectivity. Instead, a patch management enclave should be established, providing an additional barrier between online patch management and the systems requiring upgrades. Patches obtained in this way still must be transferred to the system needing an upgrade, and even if patches are "walked in" using removable media, the process represents a potential attack vector (albeit a small one). Figure 6.9 illustrates the vulnerability assessment and remediation process, accounting for the isolation of patch management and configuration management.

If the vulnerability is not addressable via a patch or configuration change, the vulnerable service should be removed. This may be easy or difficult depending upon the criticality of the vulnerable system. If an HMI is found to posses Microsoft Internet Information Service (IIS) vulnerabilities, the decision is easy: web services should not be necessary or allowed within SCADA and control networks, and so the

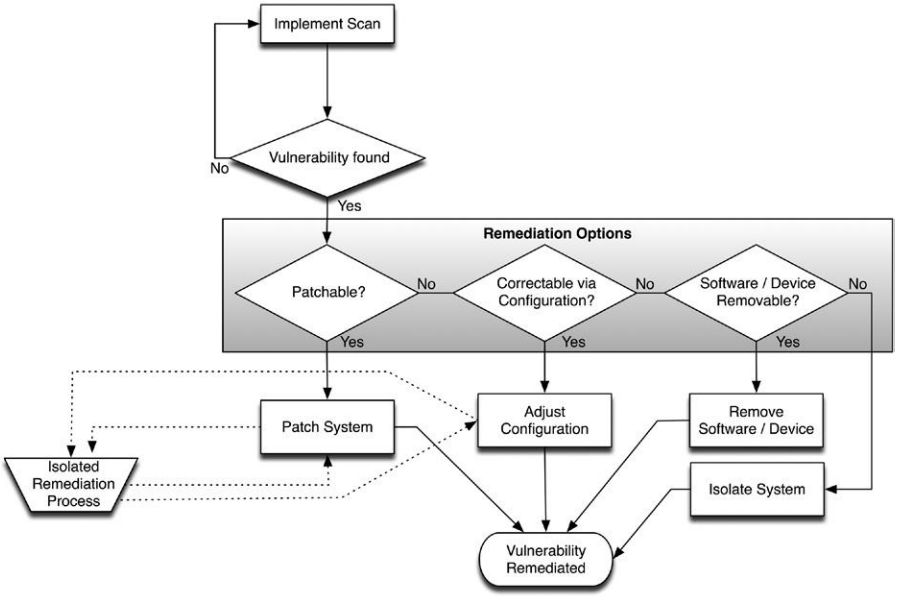


FIGURE 6.9

Vulnerability Assessment Methodology.

service can be safely removed. However, if the same HMI has vulnerabilities in core OPC services and there is no patch available, you cannot simply disable the service as it would effectively disable the HMI. In these cases, the offending system should be quarantined: effectively locking down its security enclave and preventing any access exceptions at the perimeter. For example, if there is an HMI system with OPC vulnerabilities, all systems that connect legitimately via OPC should be grouped into an enclave and isolated from remaining systems with strong firewall and/or IPS policies. More advanced threat and malware detection should also be considered in these circumstances, such as protocol or application inspection at the perimeter, to ensure that all OPC protocol traffic is legitimate and benign. This is covered in more detail in Chapter 7, “Establishing Secure Enclaves.”

Patch Management

The most secure and effective method of obtaining and applying a software patch is through the use of a dedicated patch management system. Because this system will be responsible for connecting to the Internet and downloading unverified software, it should be treated with caution and carefully isolated from other systems. The adequate sandboxing of patch management systems is paramount, as patch management system introduce significant risk. According to NIST SP 800-40, creating a Patch and Vulnerability Management Program, the following risks may be introduced when obtaining patches through a commercial patch management system:[40]

- The software vendor's patch might have been corrupted or infected with malware, either prior to distribution or during the distribution process.
- The patch management system could become infected, compromising all subsequently obtained patches.
- The patch management tool could be used by an attacker as centralized attack vector to industrial systems, leveraging the patch distribution capabilities.
- The patch management system could be used by an attacker to identify participating systems, as well as which patches have/have not been applied to participating systems.
- Once breached, the patch management software could be used to elevate privileges of participating systems, gain administrative access to participating systems.

Locate the patch management system within an enclave that already has open Internet access, such as the business network. If the patch management system needs to be located in SCADA or DCS networks (e.g., if the business network is geographically separate), create a unique enclave for patch management with true air gap boundaries. The patch management system is responsible for downloading and testing patches, configuration files, upgrades, and other third-party material; testing it for malware; and then archiving the validated files to read-only media (preventing any subsequent infection or manipulation). The entire patch management process is illustrated in Figure 6.10.

Applying patches, once validated, also requires caution. Even clean files may impact the operation of the target system in some unintended manner. This is especially true on industrial devices that utilize legacy versions of operating systems, as new software updates may only be tested and supported by the vendor on newer OS versions. To ensure that the new patch or configuration will not impact the target system, full testing of the application should be performed on a functionally identical test system. Ideally, an isolated test network should be maintained that contains an offline version of all systems in use. If patching an operating system (such as a Windows update or service pack), all applications on that host that are in use should be fully tested, as even minor changes in the OS could unintentionally affect third-party software.[41]

Automated updates are supported on many newer systems, and third-party commercial solutions are also available to automate the distribution and application of patches. Automation offers many benefits, including greater assurance that obtained patches are successfully applied. However, the systems used to obtain patches and to distribute patches should be kept separate, or there will be an automated mechanism in place that allows inbound Internet attacks to spread directly to industrial systems. If required, implement two instances of the patch management system: one to retrieve patches in isolation and one to distribute the validated patches after they have been hand carried across a true air gap.

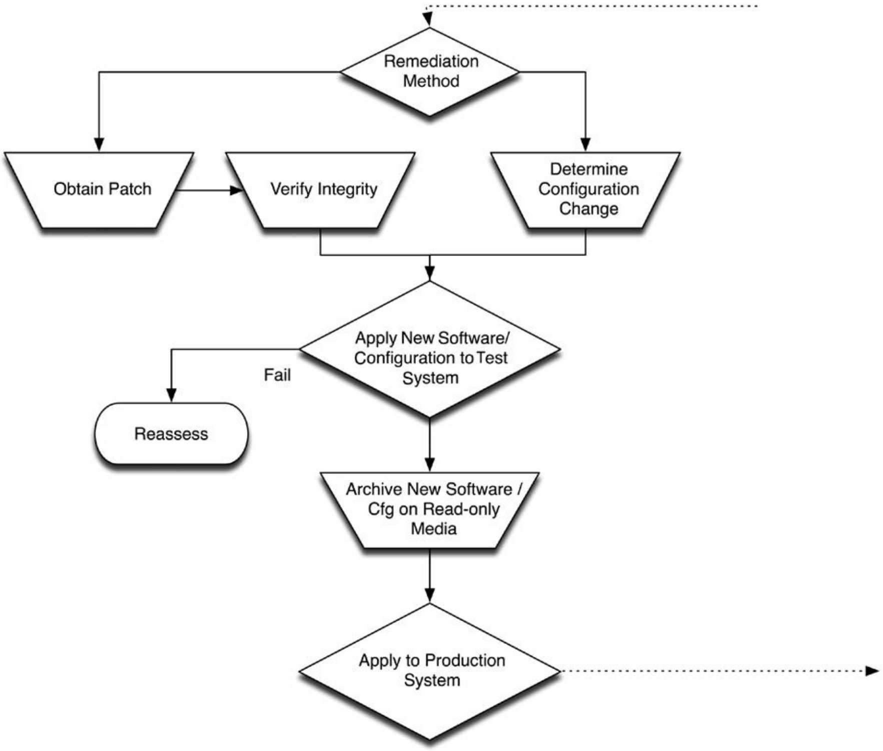

Remediation Method
Obtain Patch
Verify Integrity
Determine Configuration Change
Apply New Software/ Configuration to Test System
Fail
Reassess
Archive New Software / Cfg on Read-only Media
Apply to Production System

FIGURE 6.10

Patch Management Methodology.

NOTE

Note that the required sandboxing of these systems can be facilitated using virtual machines (VMs), allowing the patch management system to be easily restored to a known clean state after every use. This will prevent the patch management itself from being infected with malware that could then compromise all subsequently obtained patches.

Configuration Management

Configuration management refers to the process of documenting the active configurations of all systems, validating known “good” configurations, monitoring all systems to ensure the use of known “good” configurations, and monitoring all systems for any subsequent changes to the validated configuration. The logic is simple: any change could introduce risk, so once a system has been appropriately patched and configured, that system should not be changed. If a new vulnerability is found, the configuration management process allows a new configuration to be validated, at which point systems are reassessed against the new valid configuration.

As with validated patches and upgrades, a digital record of validated configurations should be stored on read-only media, ensuring that there is a clean, unadulterated copy of the configuration for use in turning up new systems, rolling back test systems, etc.

NOTE

Because many attacks attempt to adjust the configurations of the penetrated hosts (e.g., escalating privileges, disabling logging, etc.), configuration management tools can be useful for security monitoring and threat detection as well. This is discussed in more detail in Chapter 9, "Monitoring Enclaves."

Device Removal and Quarantine

When a vulnerability cannot be remediated via patch or configuration management, the vulnerable system should be removed. If the service is critical, and there is no viable and secure replacement, the only alternative is completely isolate that vulnerability. Quarantining a service based upon enforced policies requires that all access to the vulnerable service is cut off from any non-essential communications, with all essential communications being encrypted for further protection. The service should also be disjointed across enclaves, with explicitly defined "deny" policies at all encompassing firewalls, IDS/IPS, and other devices. This ensures that no direct access to the vulnerable service exists. Quarantined enclaves are discussed in Chapter 7, "Establishing Secure Enclaves."

SUMMARY

Understanding how an attack might be performed, the importance of identifying and remediating vulnerabilities becomes clear. The importance of establishing secure enclaves that isolate functional groups—especially control system functions, which utilize specialized protocols that can facilitate the most basic steps of an attack: scanning to identify targets, enumerating the client/server relationships within the control protocol (as there is no real authentication to enumerate), and the delivery of a malware payload and/or the direct control over a process.

To adequately assess vulnerabilities in a control network, however, appropriate vulnerability assessment techniques must be used in order to prevent potential disruption of sensitive industrial protocols that may be in use. In addition, the various entry points—the paths through which an attacker might attempt to gain access to the industrial control network—must be understood, so that vulnerability detection can used as part of a real penetration test: identifying all the vectors and exploits that are available to an attacker enable vulnerabilities to be remediated, significantly increasing the overall security of the industrial network. No network should be assumed safe, and assessment of vulnerabilities from various internal networks should be performed.

Once an understanding of how various systems might be vulnerable is achieved, the process of securing and isolating them can begin. The information obtained from a strong vulnerability assessment and patch management strategy will also facilitate the process of both defining and securing these functional groups into secure enclaves, discussed in detail in Chapter 7, "Establishing Secure Enclaves."

ENDNOTES

1. S. McClure, J. Scambray, G. Kurtz, Hacking Exposed: Network Security Secrets and Solutions, McGraw Hill, 1999.
2. Ibid.
3. Computer Based Social Engineering Tools: Social Engineer Toolkit (SET). <http://www.social-engineer.org/framework/Computer_Based_Social_Engineering_Tools:_Social_Engineer_Toolkit_(SET)> 2010 (cited: December 23, 2010).
4. S. McClure, J. Scambray, G. Kurtz, Hacking Exposed: Network Security Secrets and Solutions, McGraw Hill, 1999.
5. Ibid.
6. nmap.org. Introduction. <http://nmap.org>, 2010 (cited: December 23, 2010).
7. nmap.org. Nmap Reference Guide. <http://nmap.org/docs.html>, 2010 (cited: December 23, 2010).
8. S. McClure, J. Scambray, G. Kurtz, Hacking Exposed: Network Security Secrets and Solutions, McGraw Hill, 1999.
9. Rapid7 LLC. The Metasploit framework. <http://www.metasploit.com/framework>, 2010 (cited: December 23, 2010).
10. B. Sterling, The advanced persistent threat attack. <http://www.wired.com/beyond_the_beyond/2010/01/the-advanced-persistent-threat-attack/> January 30, 2010 (cited: December 23, 2010).
11. Ibid.
12. See www.zdnet.com/blog/security/shodan-search-exposes-insecure-scada-systems/7611.
13. S. McClure, J. Scambray, G. Kurtz, Hacking Exposed: Network Security Secrets and Solutions, McGraw Hill, 1999.
14. M. Franz, DNP3 Recon, Digital bond. <http://www.digitalbond.com/index.php/2006/10/18/dnp3-recon/>, October 18, 2006 (cited December 23, 2010).
15. N. Falliere, L.O. Murchu, E. Chien, Symantec, W32.Stuxnet Dossier, version 1.3, October 2010.
16. K. Stouffer, J. Falco, K. Scarfone, National Institute of Standards and Technology, Special Publication 800-82 (Final Public Draft), Guide to Industrial Control Systems (ICS) Security, Section 6.3.1 Identification and Authentication, September, 2008.
17. S. McClure, J. Scambray, G. Kurtz, Hacking Exposed: Network Security Secrets and Solutions, McGraw Hill, 1999.
18. K. Stouffer, J. Falco, K. Scarfone, National Institute of Standards and Technology. Special Publication 800-82 (Final Public Draft), Guide to Industrial Control Systems (ICS) Security, September 2008.
19. The SANS Institute, Top cyber security risks; executive summary. <http://www.sans.org/top-cyber-security-risks/summary.php>, 2010 (cited: December 23, 2010).

20. J. Pollet, Electricity for free? The dirty underbelly of SCADA and Smart Meters. Red Tiger Security. in: Proc. 2010 BlackHat Technical Conference, July, 2010.
21. United States Computer Emergency Readiness Team (US-CERT), Overview of cyber vulnerabilities. <http://www.us-cert.gov/control_systems/csvuls.html#under>, 2010 (cited: December 23, 2010).
22. United States Computer Emergency Readiness Team (US-CERT), Overview of cyber vulnerabilities. <http://www.us-cert.gov/control_systems/csvuls.html#under>, 2010 (cited: December 23, 2010).
23. Wi-Foo.com, Recon and attack tools. <http://www.wi-foo.com/ViewPageefe4.html?siteNodeId=55&languageId=1&contentId=-1>, 2006 (cited: December 23, 2010).
24. J. Larson, Idaho National Laboratories, Control systems at risk: sophisticated penetration testers show how to get through the defenses. in: Proc. 2009 SANS European SCADA and Process Control Security Summit, October, 2009.
25. S4 Briefings book (Wi-Fi jamming presentation).
26. United States Computer Emergency Readiness Team (US-CERT), Overview of cyber vulnerabilities. <http://www.us-cert.gov/control_systems/csvuls.html>, 2010 (cited: December 23, 2010).
27. K. Stouffer, J. Falco, K. Scarfone, National Institute of Standards and Technology, Special Publication 800-82 (Final Public Draft), Guide to Industrial Control Systems (ICS) Security, September, 2008.
28. Rapid7, LLC, Press Release: Rapid7 acquires metasploit. <http://www.rapid7.com/news-events/press-releases/2009/2009-rapid7-acquire-metasploit.jsp>, October 21, 2009 (cited: December 23, 2010).
29. P. Mell, T. Bergeron, D. Henning, Special Publication 800-40 Version 2, Creating a Patch and Vulnerability Management Program, National Institute of Standards and Technology (NIST), November, 2005.
30. D.P. Duggan, Sandia Report SAND2005-2846P, Penetration Testing of Industrial Control Systems, Sandia National Laboratories, March, 2005.
31. Ibid.
32. R. Deraison, R. Gula, T. Hayton, Passive vulnerability scanning introduction (revision 14), tenable network security, <http://www.nessus.org/whitepapers/passive_scanning_tenable.pdf>, October 13, 2009 (cited: December 23, 2010).
33. D.P. Duggan, Sandia Report SAND2005-2846P, Penetration Testing of Industrial Control Systems, Sandia National Laboratories, March, 2005.
34. Ibid.
35. Digital Bond SCADApedia, Bandolier and NERC CIP. <http://www.digitalbond.com/wiki/index.php/Bandolier_and_NERC_CIP>, 2010 (cited: December 23, 2010).
36. Digital Bond, Bandolier. <http://www.digitalbond.com/index.php/research/bandolier/>, 2010 (cited: December 23, 2010).
37. Control Systems Security Program (CSSP), Cyber Security Evaluation Tool: overview, United States Computer Emergency Readiness Team (US-CERT). <http://www.us-cert.gov/control_systems/satool.html>, 2010 (cited: December 23, 2010).
38. P. Mell, T. Bergeron, D. Henning, Special Publication 800-40, Version 2, Creating a Patch and Vulnerability Management Program, National Institute of Standards and Technology (NIST), November, 2005.
39. Ibid.
40. Ibid.
41. K. Stouffer, J. Falco, K. Scarfone, National Institute of Standards and Technology, Special Publication 800-82 (Final Public Draft), Guide to Industrial Control Systems (ICS) Security, Section 6.3.1 Identification and Authentication, September, 2008.

CHAPTER

Establishing Secure Enclaves

7

INFORMATION IN THIS CHAPTER:

- Identifying Functional Groups
- Establishing Enclaves
- Securing Enclave Perimeters
- Securing Enclave Interiors

The concepts of Defense in Depth, as discussed up to this point, have focused on the separation of devices, ports, services, and even users into functional groups. The logic is simple: by isolating functional groups, the attack surface of any one group is minimized. The group itself can be secured using a variety of products and techniques, turning the group into a secure enclave. The enclave will be much more difficult to penetrate because the isolation of its services will deter attempts to scan and enumerate the enclosed network devices.

Unfortunately, enclaves are typically defined only in very broad terms, separating the industrial network into as few as two or three enclaves: the control system, the business LAN, and in some cases a supervisory demilitarized zone between them. In some cases, such as in nuclear facilities, a five-tier enclave system is used, based on the Nuclear Regulatory Commission guidelines defined in RG 5.71.[1] Enclaves can—and should—be defined much more precisely. However, before this can happen, the functional groups themselves need to be defined. While simple in concept, this can be a difficult and time-consuming process. It begins by logically grouping networks, assets, the operations that they perform, and even the users who are responsible for those operations. These overlapping groups are then examined to identify the common denominators between systems. The result is an enclave: exclusive collections of only those systems that are necessary to perform a specific function.

Once defined, the enclave then needs to be secured. Ideally, every enclave would be secured to the highest degree possible. Realistically, costs and other factors make this goal unattainable. Therefore, it is also necessary to identify those enclaves that represent the highest risk to safety and reliability, so that the strongest perimeter defenses can be implemented where they are needed the most (understanding the criticality of an enclave may be required for regulatory compliance purposes as well). Perimeter defenses may consist of firewalls, Network IDS and

IPS devices (NIDS and **NIPS**), router Access Control Lists (ACLs), application monitors, and/or similar security products—all of which can and should be configured to isolate the defined members of an enclave.

While perimeter defense is important, the enclave interior must also be secured to protect the enclave against inside attacks and/or an attack that somehow circumvents the established perimeter defenses (such as walking malware into a control system using a physical device, or injecting malware from outside of the control system using an unknown access point or vulnerability). Interior defenses consist primarily of host security systems, such as Anti-Virus, Anti-Malware, Host IDS (HIDS), and **application whitelisting** systems. As with perimeter defenses, internal defenses should be configured in concert with the authorized parameters of established and documented enclaves.

While this chapter will cover the identification of an enclave as well as the methods of perimeter and asset defense, it is also important to define the expected behavior of an enclave and to monitor all activities within each enclave—both for the obvious alerts that might be generated by perimeter and host security products and for behavioral anomalies within the enclave. Baselining enclave activity is covered in Chapter 8, "Exception, Anomaly, and Threat Detection," while monitoring enclave activity is covered in Chapter 9, "Monitoring Enclaves."

IDENTIFYING FUNCTIONAL GROUPS

The first step of building a secure enclave is to identify any and all functional groups, so that you can determine what each enclave consists of, and where its perimeters are. A "functional group" refers to anything directly involved in or responsible for a given function. When identifying functional groups, assess all assets (physical devices), systems (software and applications), users, protocols, and other items. Attempt to separate two items, such as a protocol from an asset. If the two can be separated without impacting either item's primary function, they belong to two functional groups. For example, if some HMI systems use the DNP3 protocol, create a list of all devices currently communicating over DNP3. Assess each to see if DNP3 is necessary to its function or not (it may support multiple protocols, and may be actively using a different protocol to perform its functions). If not remove it from the functional group, and if possible disable the unused protocol on the HMI as well. The result will be a list of all assets legitimately using that protocol (see "Protocols").

Similarly, consider which assets are connected to each other on the network, both physically and logically. Each represents a functional group based on network connectivity (see "Network Connectivity"). Again, assess each item in question individually, and if it does not need to belong, remove it.

A functional group can be based on almost anything. Common functional groups to consider when building enclaves in industrial networks include Control Loops, Supervisory Controls, Control Processes, Control Data Storage, Trading

Communications, Remote Access, and even less tangible groups such as User groups and Industrial Protocol groups.

Network Connectivity

Functional groups based on network connectivity are easy to understand because networks by nature connect devices together: how the different devices are connected on the network clearly qualify those items that belong to an interconnected group and those that are excluded by a hard perimeter. Networks should be considered both physically (what devices are connected to other devices via network cables or wireless connections) and logically (what devices share the same routable network space or subnet).

Physical network boundaries are easy to determine using a network map. Ideally (although not realistically) all control system networks will have a hard physical boundary in the form of an air gap. Realistically, there will be interconnection points consisting of a single link, preferably through a firewall and/or other defensive devices.

CAUTION

Wireless networks are easy to overlook as physical network connections. However, any two devices with wireless antennae, regardless of whether they have logical connection to the wireless network in question, should be considered "physically" connected. The separation provided by authenticated wireless access is a logical separation. To truly separate two wireless-capable devices at the physical level, the antennae of one device would need to be disabled, or a barrier capable of disrupting the wireless connection needs to be placed between the two devices.

Logical network boundaries are defined by the use of routers to separate a physical network into multiple address spaces. The router provides a logical demarcation between each network. This forces all communications from one logical network to another to go through the router, where ACLs and other protective measures can be implemented.

Note that VLANs are a type of logical boundary, but one that is enforced at layer 2 rather than layer 3. VLANs use a standardized tag in the Ethernet packet header to determine how they are handled by the router: traffic destined for the same VLAN is switched, while traffic destined for a different VLAN is routed. VLANs, however, are not recommended for security, as it is possible to modify the packet header to hop VLANs, bypassing the router.[2]

Control Loops

A control loop consists of the devices responsible for a particular automated process (see Chapter 5, "How Industrial Networks Operate"). Applying this list of devices to a functional group is relatively simple. In most instances, a control loop will consist

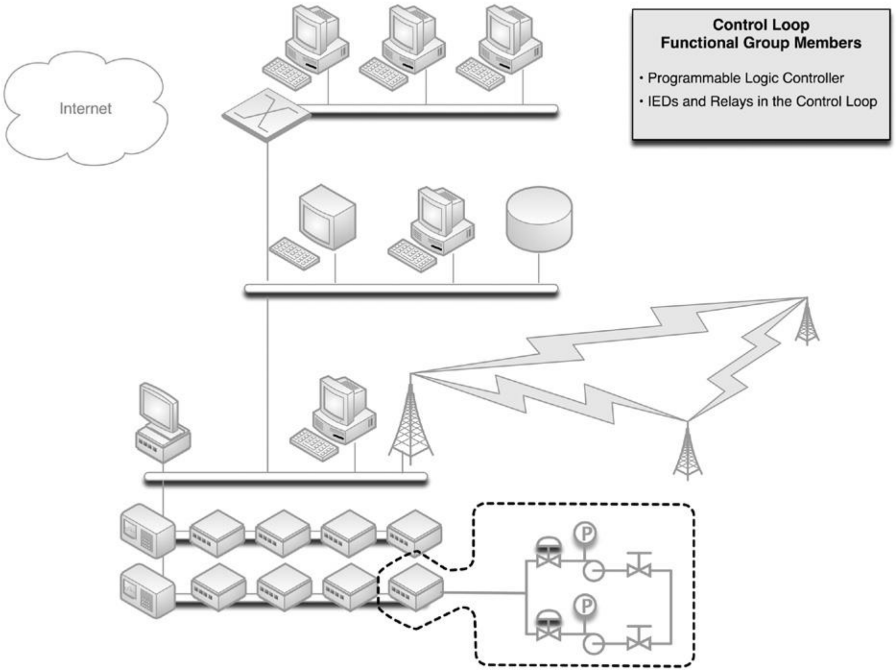

Internet
Control Loop
Functional Group Members
• Programmable Logic Controller
• IEDs and Relays in the Control Loop
P
P

FIGURE 7.1

A Functional Group Based on a Control Loop.

of a PLC and any related inputs and outputs, as illustrated in Figure 7.1. If an IED is a direct input or output of the control logic, those devices share a functional group with the controller; if not, they do not.

Where defining a functional group based on network connectivity is a broad example that might result in a handful of functional groups, building a functional group based on a control loop is a very precise example. The functional groups created will be numerous, and each will contain a relatively small number of devices (a specific PLC or RTU and a collection of relays and IEDs).

Supervisory Controls

Each control loop is also connected to some sort of supervisory control—typically an HMI—that is responsible for the configuration, monitoring, and management of the automated process. Because the HMI is responsible for the PLC, these two devices belong to a common functional group. However, because the HMI is not directly responsible for those IEDs connected to the PLC, these items are not necessarily in a common functional group as the HMI (they belong to a common functional group based on some other common criteria, such as protocol use). Figure 7.2 shows a common supervisory functional group.

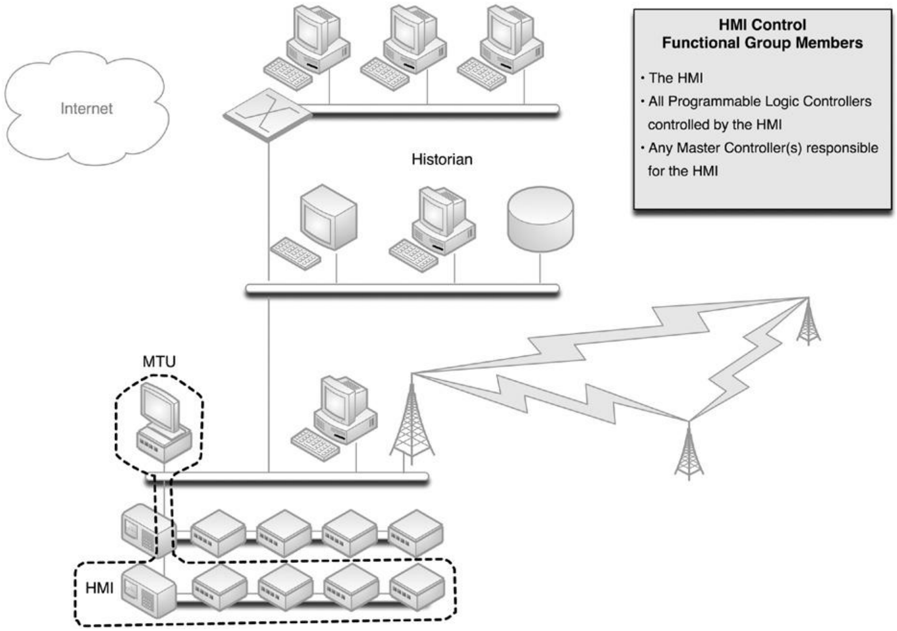

Internet
Historian
HMI Control
Functional Group Members
• The HMI
• All Programmable Logic Controllers controlled by the HMI
• Any Master Controller(s) responsible for the HMI
MTU
HMI

FIGURE 7.2

A Functional Group Based on an HMI.

All PLCs controlled by the HMI are included, as are any "master" HMI or control management systems that might have responsibility or control over the initial HMI (see Chapter 5, "How Industrial Networks Operate"). Other HMIs are not included, as they are not the responsibility of the initial HMI. Rather, each HMI would represent its own functional group. If a common master controller is in use to manage multiple HMIs, each HMI's distinct functional group will contain the same master, creating an overlap between multiple functional groups.

NOTE

There are many other devices, such as I/O drives, printers, and safety systems that may also be connected to an HMI and therefore might also be included in the HMI's functional group. However, these items are not shown in Figure 7.2 in order to simplify the illustration.

Control Processes

If a Master Controller or Master Terminal Unit (MTU) is used to manage multiple HMIs, each responsible for a specific part of a larger control process (see Chapter 5, "How Industrial Networks Operate"), that device represents the root of

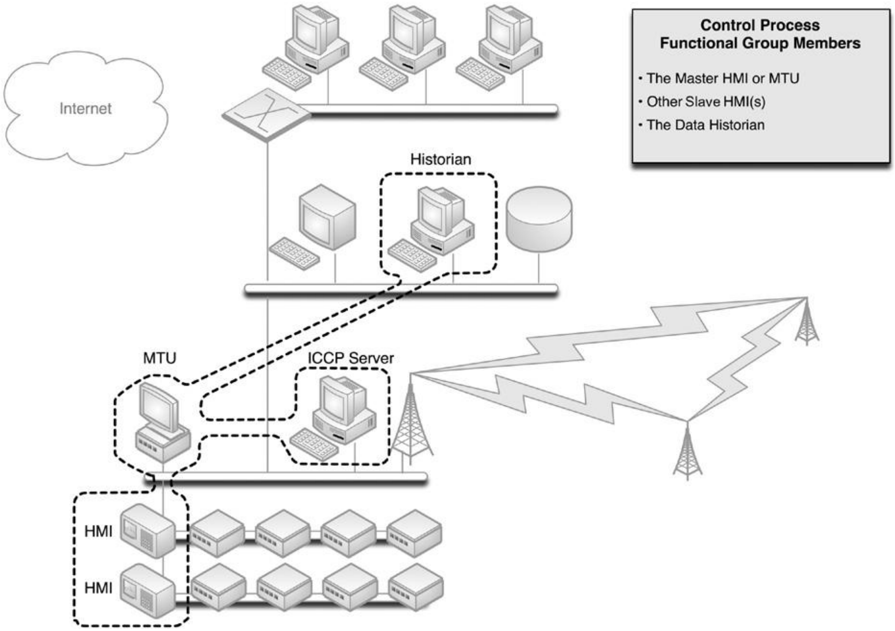


FIGURE 7.3

A Functional Group Based on a Control Process.

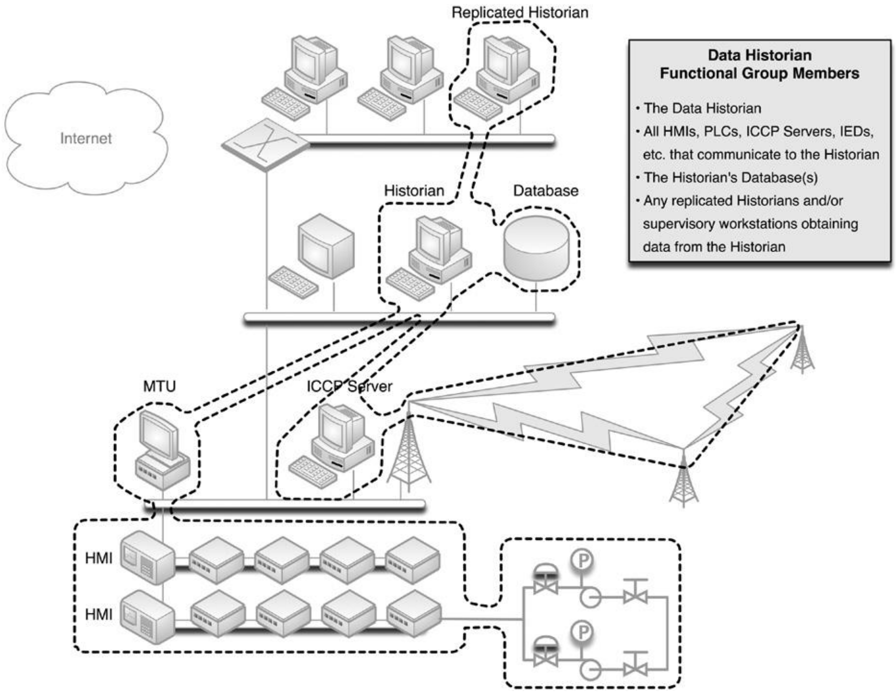
yet another functional group—this time containing all relevant HMIs, as shown in Figure 7.3.

This example also introduces the concept of process communication and historization. If an MTU interfaces with an ICCP server, for example, in order to communicate bulk electrical load to another electrical entity, the ICCP server should also be included in the MTU's functional group. Similarly, if the process information from the MTU is fed into a Data Historian, that system should also be included.

Control Data Storage

Many industrial automation and control system devices generate data, reflecting current configurations, the status of a process, alarms, and other information. This information is typically collected and "historized" by a Data Historian (see Chapter 5, "How Industrial Networks Operate"). The Data Historian system may connect to many—potentially all—devices throughout the control system network, supervisory network, and in some cases the business network, as illustrated in Figure 7.4.

Not shown here are other devices such as Network Attached Storage (NAS) devices, Storage Area Networks (SAN), and other devices that may be present to support the data storage requirements of a Historian, especially in larger industrial operations.



Data Historian Functional Group Members

- The Data Historian
- All HMIs, PLCs, ICCP Servers, IEDs, etc. that communicate to the Historian
- The Historian's Database(s)
- Any replicated Historians and/or supervisory workstations obtaining data from the Historian

FIGURE 7.4

A Functional Group Based on Historization.

Trading Communications

The need to communicate between control centers is sufficient enough to justify a specialized industrial protocol, developed specifically for that task: the Inter Control Center Communication Protocol, or ICCP (see Chapter 4, “Industrial Network Protocols”). ICCP connections require explicitly defined connections between clients and servers, and therefore, any operation utilizing ICCP to communicate with a field facility and/or a peer company will have one or more ICCP servers and one or more ICCP clients (these can be a single physical server or multiple distributed servers). This is the first example of a functional group that extends over Wide Area Networks, as illustrated in Figure 7.5.

One thing to remember when assessing this functional group is that the remote client devices are all explicitly defined, even if owned by another company and hosted at its facility. These remote clients should be included within the functional group, as they have a direct relationship to any local ICCP servers that may be in use.

Because ICCP connections are typically used for trading, access to operational information is necessary. This could be a manual or automated informative process,

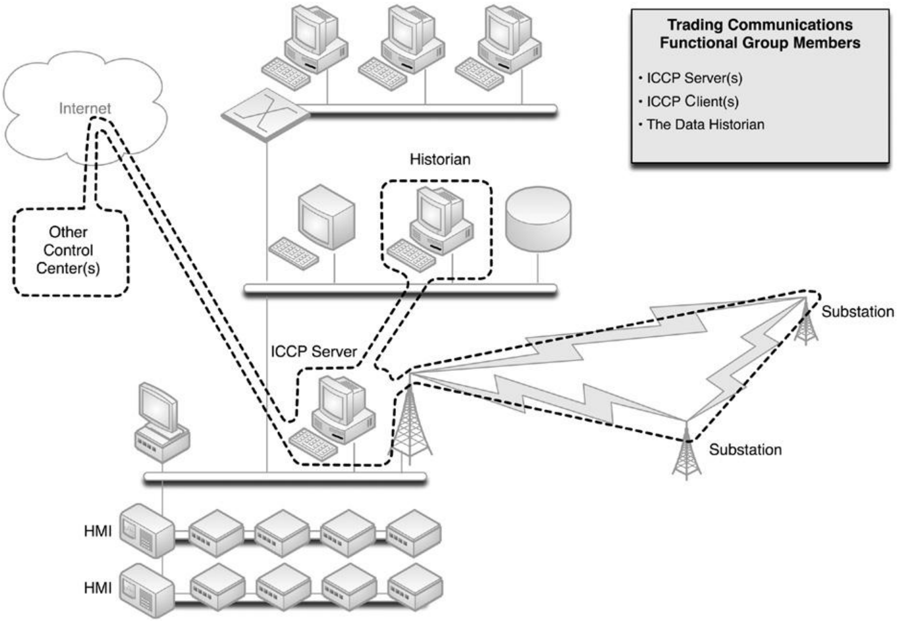


FIGURE 7.5

A Functional Group Based on the Inter Control Center Protocol for Trading Communication.

which most likely involves the historized data stores of the Data Historian (or a subsystem thereof); as such, the Data Historian is included in this example of a "Trading Communications" enclave.

Remote Access

ICCP is but one, specialized method of remotely accessing a system. Many control systems and industrial devices—including HMIs, PLCs, RTUs, and even IEDs—allow remote access for technical support and diagnostics. This access could be via dial-up connection, or via a routable network connection. Remote access to control system devices, if it is provided, should be controlled via specialized virtual private networks (VPNs) or remote access servers (RAS), and should only allow explicitly defined, point-to-point connections from known entities, over secure and encrypted channels. These explicitly defined users, the devices that they access, and any VPN or RAS systems that are used constitute a remote access functional group, as illustrated in Figure 7.6.

By functionally isolating remote connections, additional security can be imposed. This is extremely important in order to avoid an open and inviting vector to an attacker.

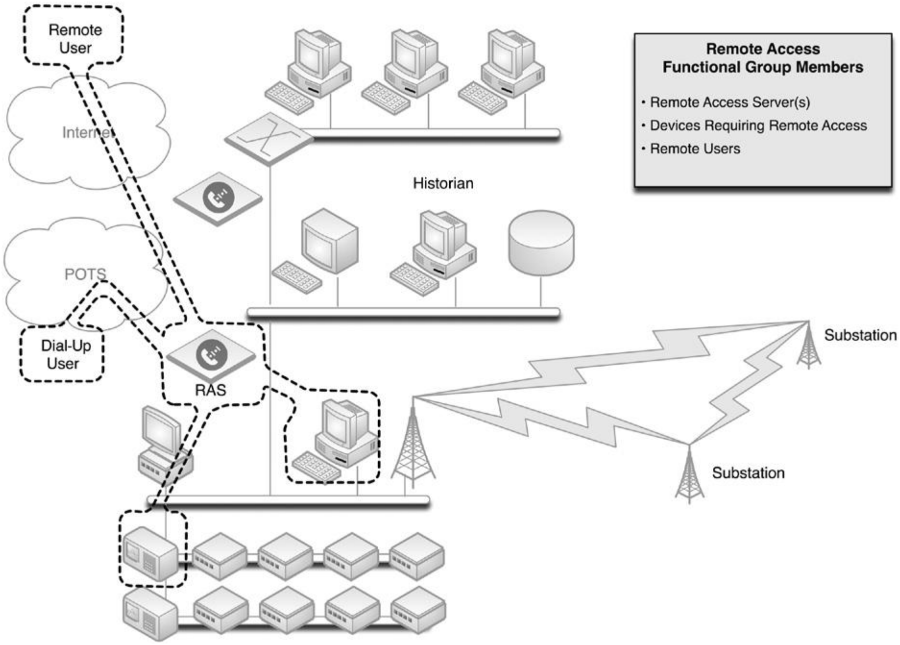

Remote User
Internet
POTS
Dial-Up User
RAS
Historian
Substation
Substation
Remote Access Functional Group Members
• Remote Access Server(s)
• Devices Requiring Remote Access
• Remote Users

FIGURE 7.6

A Functional Group Based on Remote Access.

Users and Roles

Every system is ultimately accessed by either a user or another system. Until now, functional groups have been built around the latter: explicitly defining which devices should legitimately be communicating with other devices. For human interaction, such as an operator accessing an HMI to adjust a process, it is just as important to define which users should legitimately be communicating with which devices. This requires a degree of Identity and Authentication Management (IAM), which defines users and their roles. The most well-known example of an IAM is Microsoft's Active Directory services, although many other commercial IAM systems exist. Figure 7.7 illustrates the concept of a functional group containing a user and those devices that the user is allowed to interface.

Mapping roles and responsibilities to devices can be tedious but is very important, as the resulting functional group can be used to monitor for unauthorized access to a system by an otherwise legitimate user. That is, an employee with control system access to a certain HMI, upon termination of his or her employment, might decide to tamper with other systems. By placing a user in a functional group with only those devices he or she should be using, this type of activity could be easily detected and possibly prevented (remember, defining functional groups is only

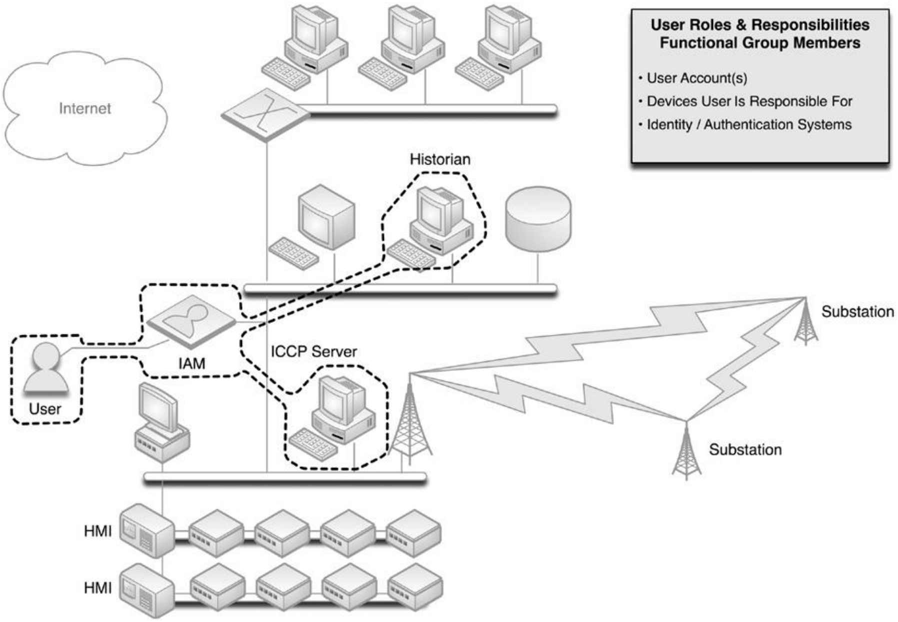


FIGURE 7.7

A Functional Group Based on Users and Roles.

the first step to building a secure enclave. The groups must be further refined into actual enclaves, and then secured internally and at the perimeter, as discussed under "Securing Enclave Perimeters" and "Securing Enclave Interiors").

Protocols

The protocols that a device uses in industrial networks can be explicitly defined, and so it should be, in order to create functional groups based on protocols. Only devices that are known to use DNP3 should ever use DNP3, and if any other device uses DNP3, it is a notable exception that should be detected quickly and prevented outright if possible. The areas where a specific industrial protocol is commonly used has already been discussed in Chapter 4, "Industrial Network Protocols." Now, the specific devices using specific industrial protocols should be identified and recorded, in order to build one more important functional group, as shown in Figure 7.8.

Criticality

Enclave-based security is about isolating common influencing factors into functional groups so that they can be kept separate and secure from other noninfluencing factors.

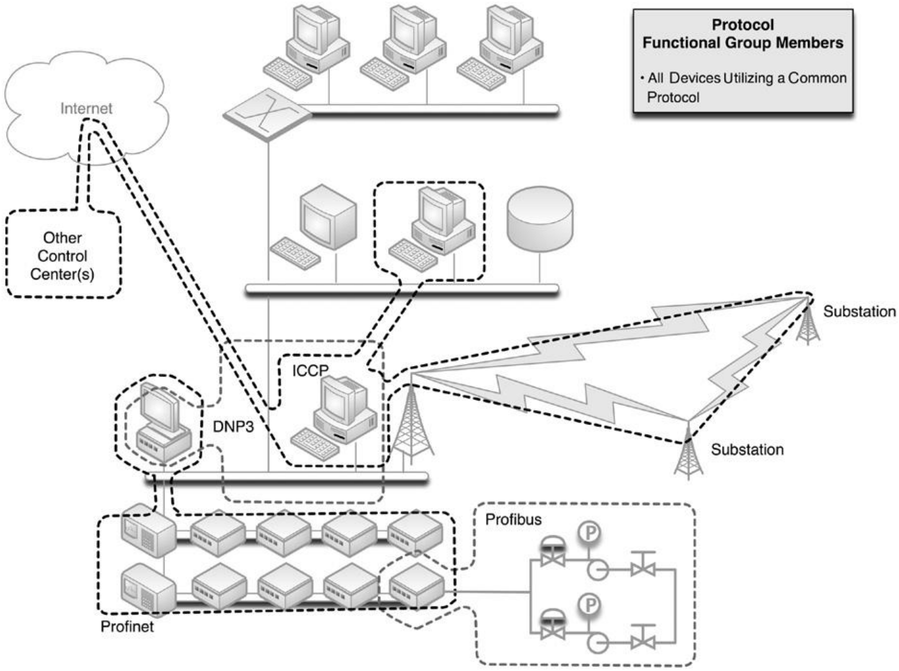


FIGURE 7.8

A Functional Group Based on Protocols.

The NRC dictates within CFR 73.54 that the criticality of assets be determined so that they can be separated into five logical security zones.[3] The NRC security zones are an example of enclave-based security, using a functional grouping based on criticality. NRC regulations also provide an example of how stronger security measures should be used as the criticality of the enclave increases, as the NRC regulatory Guide 5.71 clearly differentiates the level of security provided between zones.

Critical assets, as defined by NERC, are those that can impact the operation of the bulk electric system.[4] They might include control centers, transmission substations, generation systems, disaster recovery systems, black start generators, load shedding systems and facilities, special protection systems, etc.[5] They can be identified using a simple methodology (see Chapter 2, "About Industrial Networks"). Determining the criticality of an enclave is a similarly straightforward process, and uses a similar methodology, as illustrated in Figure 7.9.

Critical assets are extrapolated to the critical function group(s) to which they belong, which may or may not contain other critical and/or noncritical assets. Any enclave that includes that function group (and therefore the critical asset) is a critical enclave.

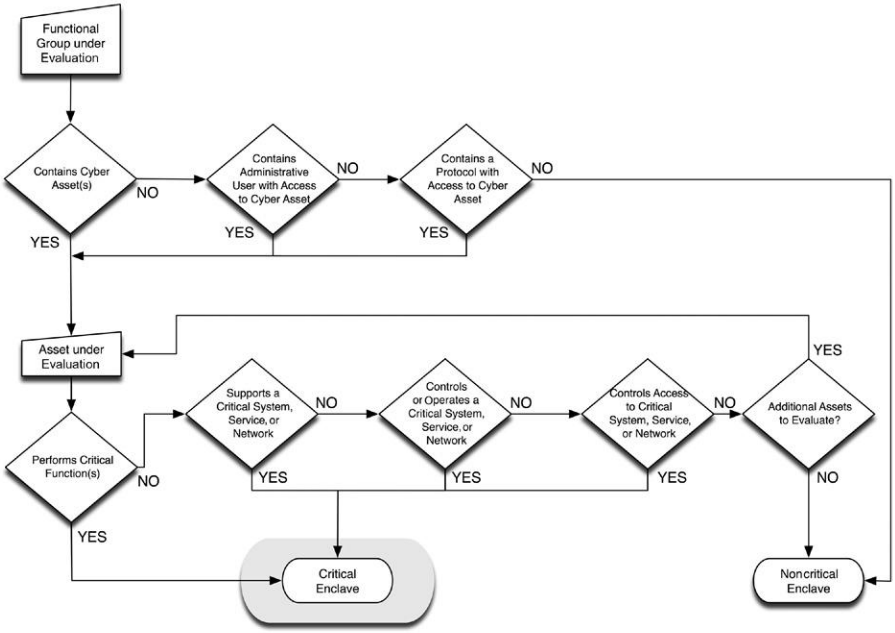

Functional Group under Evaluation
Contains Cyber Asset(s)
NO
Contains Administrative User with Access to Cyber Asset
NO
Contains a Protocol with Access to Cyber Asset
NO
YES
YES
YES
Asset under Evaluation
YES
Supports a Critical System, Service, or Network
NO
Controls or Operates a Critical System, Service, or Network
NO
Controls Access to Critical System, Service, or Network
NO
Additional Assets to Evaluate?
Performs Critical Function(s)
NO
YES
YES
YES
NO
YES
Critical Enclave
Noncritical Enclave

FIGURE 7.9

Determining the Criticality of an Enclave.

TIP

While grading the importance of an asset for compliance can be construed as a means to measure accountability (and fines), it also allows us to improve threat detection and measure the severity of an event should one occur. By taking the time and making the effort to identify critical assets and enclaves, you can also greatly improve your threat detection capability, by configuring security monitoring tools to weight the perceived severity of suspicious activities, ranking them in order of consequence and priority. This is discussed in more detail in Chapter 9, "Monitoring Enclaves."

However, simply defining functional groups around criticality to identify enclaves will result in very few enclaves (a total of five, using the NRC guidelines). In contrast, the more enclaves that are defined the stronger the security of the industrial network as a whole, and so a broader methodology—which identifies many more distinct enclaves—is preferred. Therefore, criticality should be assessed within the context of the previously defined functional groups. In this way the most critical systems will be protected by an additional layer of separation—within the inherent security of the enclave itself and then the additional protections between critical and noncritical items within that enclave. This will help to secure critical devices from the insider threat, such as a disgruntled employee who already has legitimate

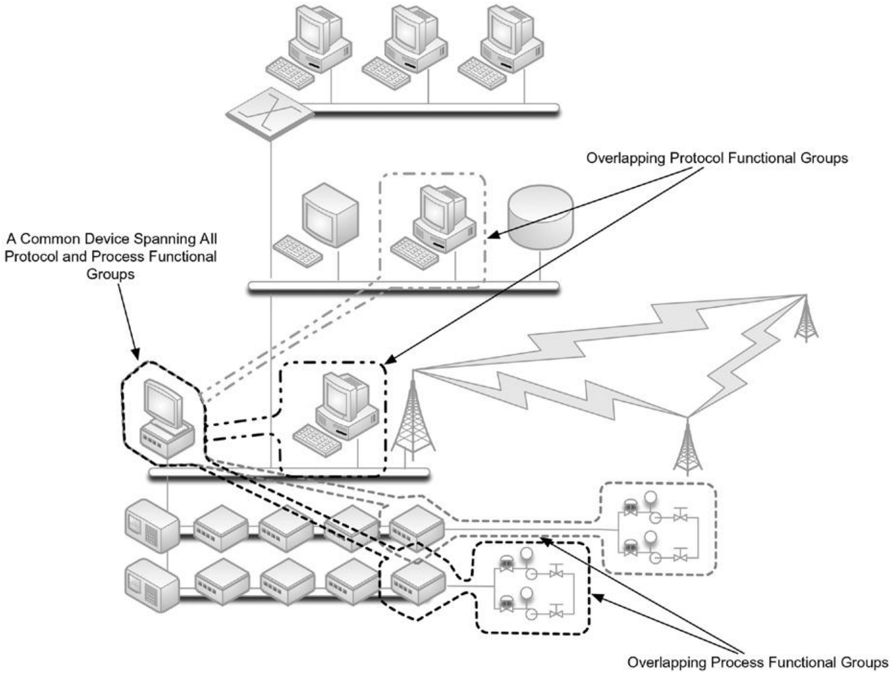

Overlapping Protocol Functional Groups
A Common Device Spanning All Protocol and Process Functional Groups
Overlapping Process Functional Groups

FIGURE 7.10

Overlapping Functional Groups.

physical and logical access to the parent enclave. It also prevents lateral attack from one critical system to the next: if all critical systems are grouped together solely because they are all "critical," a successful breach of one critical system puts the entire critical infrastructure at risk.

Using Functional Groups to Identify Enclaves

Defining groups based on services, protocols, criticality, and other factors is an excellent way to eliminate unknown, unauthorized devices from a group. Simply, if two devices do not share a common quality, there is no way for them to communicate. Unfortunately, many devices support multiple protocols, applications, services, and other qualities, resulting in multiple overlapping functional groups. Figure 7.10 shows two overlapping functional groups, based on a common controller, as well as two functional groups based on protocol, which then partially overlap with the first group.

This illustrates the difficulties of defining clear-cut groups when so many variables are in play. Superimposing Figures 7.1 through 7.8 atop each other creates many overlapping functional groups, which are difficult to make sense of (as shown in Figure 7.11). Ideally, every functional group would contain a clear demarcation

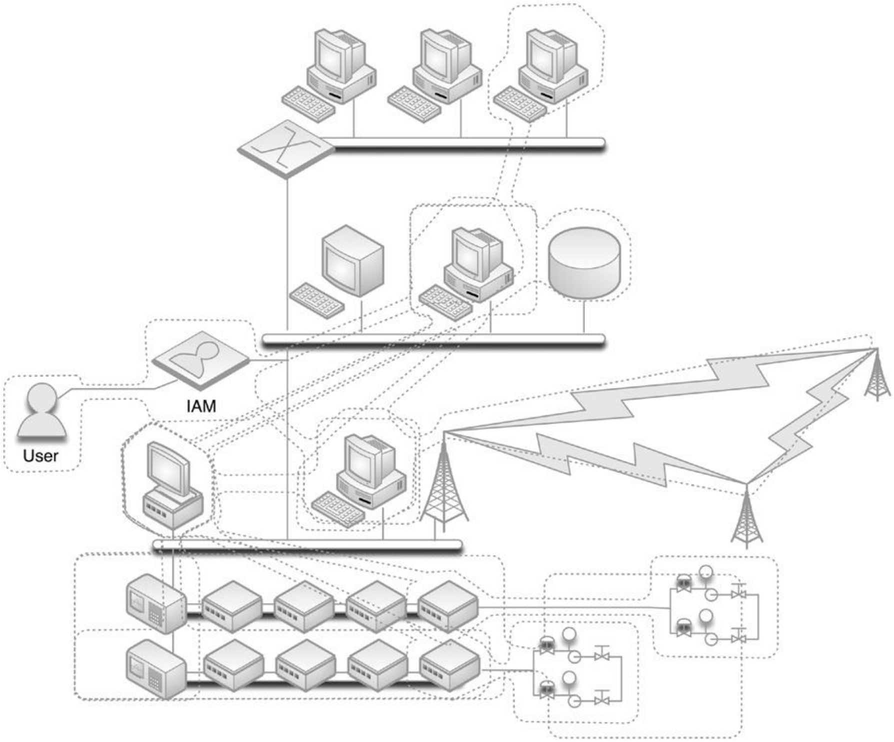


FIGURE 7.11

Many Overlapping Functional Groups.

from every other group, and each demarcation would be secured using a unique protective device (see "Securing Enclave Perimeters"). However, in many cases it is necessary to simplify the functional groups using a common quality shared between groups, effectively combining overlapping functional groups into a single, larger enclave.

Ultimately, the process of distilling the many functional groups into manageable ones will result in several defined security enclaves, with a clear understanding of the boundaries of that enclave, and the users, devices, and protocols that are contained within.

TIP

Carefully document each functional group as well as the devices, services, protocols, and users within it. These lists will come in handy when establishing the enclaves (see "Establishing Enclaves") as well as when implementing perimeter defenses (see "Securing Enclave Perimeters") and monitoring enclave behavior (see Chapter 9, "Monitoring Enclaves").

ESTABLISHING ENCLAVES

Once the process of pairing down the dozens of functional groups has been completed and the groups have been consolidated where necessary into larger overlapping groups, the enclaves can be established. Logically, the enclaves have already been defined at this point, with each consolidation of functional groups equating to a single security enclave.

The process of establishing enclaves can be summarized as follows:

1. Identifying the boundaries of each enclave so that perimeter defenses can be deployed in the correct location.
2. Making any necessary changes to the network so that the network architecture aligns with the defined enclaves.
3. Documenting the enclave for purposes of policy development and enforcement.
4. Documenting the enclave for purposes of security device configuration.

NOTE

Establishing an enclave is simply a means of mapping those functional groups that need to be isolated to the network architecture, policies, and configurations that are necessary to enforce that isolation. That is, the enclave itself is just a logical entity, which must then still be secured (see "Securing Enclave Perimeters" and "Securing Enclave Interiors").

Identifying Enclave Perimeters

Once an enclave is identified, it must be mapped to the network so that clear electronic perimeters can be defined. While this process is required under NERC regulation CIP 005[6], it is a necessary process that should be performed for any industrial network regardless of regulatory concerns, as an enclave can only be secured if there are defined and control entry points. In many cases the demarcation of the enclave will be very clear; for example, there may be a single network connection between a control center's supervisory LAN and the control system network. In some instances, multiple connections might exist; for example, a power generation facility might connect to both supervisory and control networks, as well as directly to substations or remote field stations. All network connections into or out of an enclave comprise that enclave's electronic perimeter.

TIP

Wireless, dial-up, and other remote connectivity are easy to overlook when identifying perimeters. If a wireless access point is located inside the enclave, a wireless user could connect directly to that enclave via a Wi-Fi connection. The access point, therefore, is part of the enclave's perimeter, even if it is physically connected well inside the enclave. When securing the perimeter, the access point must also be secured (see "Securing Enclave Perimeters"). Consideration of all remote connection points in defining enclave perimeters will result in a more secure enclave, and will also comply with NERC CIP regulatory requirement CIP 005 R1.1 and R1.2, which dictate that access points to the ESP "shall include any externally connected communication end point (e.g., dial-up modems) terminating at any device within the Electronic Security Perimeter."[7]

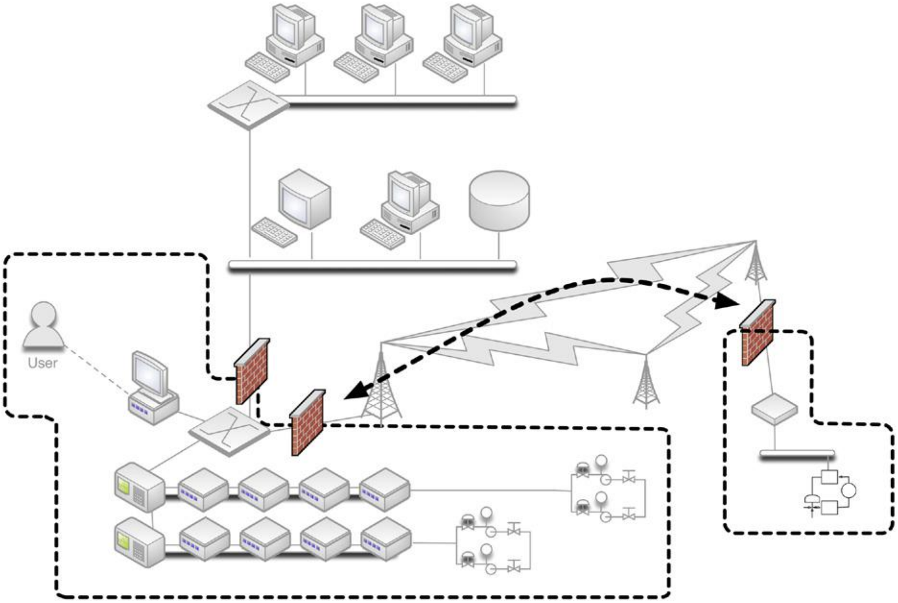


FIGURE 7.12

A Geographically Split Enclave.

In some instances, such as the one illustrated in Figure 7.12, a single enclave may consist of multiple, geographically or otherwise separated groups. In these cases, the enclave is still considered to be a single enclave. If there are any network connections between the two (or more) locations, they should be held to the same controls as the rest of the enclaves. That is, there should be no communications across those links that do not originate and terminate within the enclave, and if outside communication is required (i.e., a communication that either originates or terminates outside of the enclave), it must occur through defined and secure access points. One common method of interconnecting distributed enclaves is the use of a dedicated VPN or other encrypted gateway, while for extremely critical enclaves, a dedicated network connection or fiber cable may be used so that physical separation is maintained.

The goal is that each enclave be isolated as strictly as possible, with as few connections as possible between that enclave and any other directly adjacent (or surrounding) enclave. Figure 7.13 shows how, by providing a single access point in and out of an enclave, that point can be secured using a perimeter security device such as a firewall or IPS. In the event of a single enclave that is split (geographically or by another enclave), inter-enclave communication can still be allowed: in this case through the use of perimeter firewalls, which effectively enforce a point-to-point route between the split enclaves (this path should also be encrypted).

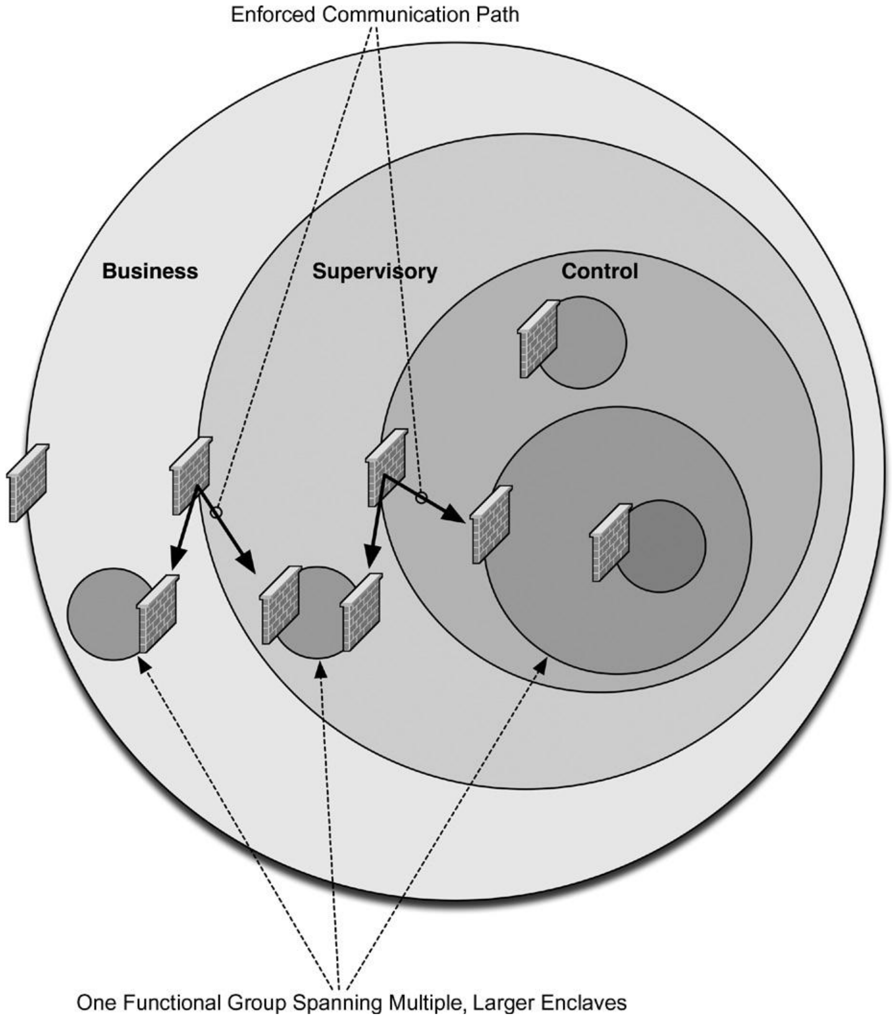

Enforced Communication Path
Business
Supervisory
Control
One Functional Group Spanning Multiple, Larger Enclaves

FIGURE 7.13

Enclave Perimeters.

In scenarios where an enclave needs to be extended across another enclave boundary, consider the functional goals of that extension. For example, in many cases a business user may require access to information originating from within a secure SCADA enclave. However, there is no requirement for the business user to communicate back into the SCADA environment. In situations like these, one-way communications can and should be enforced, either by provisioning intermediate perimeters (e.g., the firewalls shown in Figure 7.13) to disallow inbound traffic or through the use of a data diode or unidirectional gateway.

Network Alterations

No device that does not belong to a defined enclave should be directly connected to that enclave or to any device within that enclave.

In many cases, however, there will be devices identified that have access to or are connected to an enclave even though they do not belong to any of the functional groups within that enclave. For example, a printer or workstation that does not belong to the enclave might be connected to a local switch or router interface, or (as in the example under "Identifying Enclave Perimeters") a wireless access point. The aberration may be the result of improper network design or improper network addressing—whatever the result, these exceptions need to be resolved before an enclave can be secured.

In other cases, it may not be possible to clearly identify the perimeter of an enclave. For example, if supervisory, control, and enterprise systems are all interconnected via a flat network (a network that is switched purely at layer 2, without network routing or other separation of devices) or a wireless network, it will not be possible to isolate any group from any other. In these cases, a complete network redesign may be necessary to separate the enclaves to the point where only devices that belong in an enclave are directly connected to it via the network.

Enclaves and Security Policy Development

Once enclaves are defined and the necessary adjustments to the network architecture are made, a distinct milestone is reached. With defined enclaves in place, the organization is armed with the information needed to satisfy several compliance requirements of NERC CIP, ISA-99, CFATS, and others.

Documenting all enclaves within the context of the organization's security policy provides many benefits, by clearly identifying what systems may be accessed by what other systems, and how. This will facilitate policy documentation for compliance, security training and review materials, and similar security policy functions required by NERC CIP 003,[8] NERC CIP 005,[9] ISA-99 FR5,[10] CFATS Risk Based Performance Standards 8.2,[11] and NRC 10 CFR 73.54 and NRC RG 5.71 section C.3.2.[12]

Documentation of enclaves also defines how ongoing security assessments and vulnerability testing should be measured. This is again useful for compliance, including NERC CIP 008,[13] ISA-99,[14] CFATS Risk Based Performance Standards 8.5,[15] and NRC CFR 73.54 and NRC RG 5.71 section C.13.[16]

Enclaves and Security Device Configurations

Documentation can be a function of security as well as compliance. Firewalls, IDS and IPS systems, Security Information and Event Management (SIEM) systems, and many other security systems support the use of variables, which are used to map hard security configurations to organizational security policies.

For each enclave, the following lists should be maintained at a minimum:

- Devices belonging to the enclave, by IP address
- Users with authority over the enclave, by username or other identifier
- Protocols, Ports, and Services in use within the enclave

If additional metrics are identifiable, additional lists should be created. Depending on the number of enclaves that have been defined, this may require several lists—three (device, users, and ports/services) for every established enclave. Additional lists could also be maintained, for example, users by shift, in addition to users defined solely by enclave. However, unless there is a centralized authentication system in use, maintaining these lists may be cumbersome.

When finished, these variables will appear as follows:

```
$ControlSystem_Enclave01_Devices
192.168.1.0/24
10.2.2.0/29

$ControlSystem_Enclave01_Users
jcarson
jrhewing
kdfrog
mlisa

$ControlSystem_Enclave01_PortsServices
TCP 502 #Modbus TCP
TCP 20000 #DNP3
```

The creation of these variables will assist in the creation of firewall and IDS rules for the enforcement of the enclave's perimeter, as discussed under "Securing Enclave Perimeters," and will also allow for security monitoring tools to detect policy exceptions and generate alarms, as discussed in Chapter 9, "Monitoring Enclaves."

NOTE

In this book, variables are defined using `var VariableName [value1, value2, value3, etc.]` and referenced using `$VariableName`, in line with standard Snort syntax. However, depending on the device used, the specific syntax for defining and referencing variables may differ. For example, while a variable is defined as follows using Snort

```
var ControlSystem_Enclave01_Devices 192.168.1.0/24
```

the same example for an iptables firewall is defined within the iptables configuration file as follows:

```
ControlSystem_Enclave01_Devices 192.168.1.0/24
```

To define a usable variable that maps to an enclave, `var ControlSystem_Enclave01_Devices [192.168.1.0/24, 10.2.2.0/29]` is used, and then that variable is referenced within a specific rule using `$ControlSystem_Enclave01_Devices`. This is a logical extension of the classic `$HOME_NET` variable used in many IDS policies, only applied to a specific enclave. This allows for exception-based detection of unauthorized behavior within the enclave, as seen in the following rule to detect any traffic with a destination IP of a device within the defined control system enclave:

```
alert tcp any any -> $ControlSystem_Enclave01_Devices
```

With enclaves defined, and relevant variables defined for each, the enclaves can now be secured using perimeter and host security devices.

SECURING ENCLAVE PERIMETERS

Establishing an Electronic Security Perimeter (ESP) around a defined enclave provides direct protection against unauthorized access to the enclosed systems and also prevents the enclosed systems from accessing external systems from the inside out. To establish an ESP and effectively secure inbound and outbound traffic, two things must occur:

1. All inbound and outbound traffic must be forced through one or more known network connections that can be monitored and controlled.
2. One or more security devices must be placed in-line at each of these connections.

For each enclave, appropriate security devices should be selected and implemented using the recommendations below.

Selecting Perimeter Security Devices

At a minimum, a firewall is typically required. Additional security—provided by IDS, IPS, and a variety of specialized and hybrid devices such as Unified Threat Management (UTM) devices, Network Whitelisting devices, Application Monitors, Industrial Protocol Filters, etc.—may be desired as well. Typically, the criticality of the enclave (see “Criticality”) dictates the degree of security that is required. Table 7.1 maps the criticality of an enclave to required security measures of NERC CIP and NRC CFR 73.54, as well as recommended enhancements to improve security beyond regulatory requirements.

Table 7.1 recommends that both a firewall and an IPS be used at each security perimeter. This is because firewalls and IPS devices serve different functions: firewalls enforcing what types of traffic are allowed to pass through the perimeter; and Intrusion Prevention Systems closely examining the traffic that is allowed through in order to detect “legitimate” traffic with malicious intent—that is, exploit code, malware, etc—that is transferred over allowed paths. Using both devices together provides two mutual benefits: first, it allows the IPS to perform **deep packet inspection** (**DPI**) on all traffic allowed in through the firewall; second, the firewall limits the allowed traffic based on the defined parameters of the security enclave, freeing the IPS to focus its resources on just that traffic and therefore enabling it to enforce a more comprehensive and robust set of IPS rules.

Table 7.1 Perimeter Security Requirements by Criticality

Criticality	Required Security	Recommended Enhancements
4 (highest)	NRC CFR 73.54: Unidirectional Perimeter, NERC CIP 005: Firewall or IDS or IPS	Application layer monitoring, Firewall, IDS and IPS
3	NRC CFR 73.54: Unidirectional Perimeter, NERC CIP 005: Firewall or IDS or IPS	Application layer monitoring, Firewall, IDS and IPS
2	NERC CIP 005: Firewall or IDS or IPS	Firewall and IDS and IPS
1	NERC CIP 005: Firewall or IDS or IPS	Firewall and IPS
0 (lowest)	NERC CIP 005: Firewall or IDS or IPS	Firewall and IPS

For even greater protection, DPI can be used to analyze specific industrial protocol functions. This may require the use of specialized SCADA IDS or SCADA firewall devices that are designed to identify these protocol functions, or even the use of an ICS protocol filter or application monitoring tool that provides DPI across all packets within a session—providing detection and analysis capability to protocol and application contents that span multiple packets. This provides an even deeper look into the contents of network traffic. Figure 7.14 illustrates the increased security capability of firewalls, IDS/IPS devices, and application session monitoring systems.

In the most critical areas, application layer session monitoring provides a valuable and necessary level of assurance, as they are able to detect both low-level protocol anomalies (such as a base64-encoded application stream inside of HTTP, used by many APTs and botnets) and application policy violations (such as an unauthorized attempt to write a new configuration to a PLC). However, unless monitoring very simple application protocols, where the desired contents are distinctly packaged within a single packet or frame, the application session must be reassembled prior to monitoring as illustrated in Figure 7.15.

The most stringent perimeter security device may be the data diode, also referred to as a unidirectional gateway. A data diode is, very simply, a one-way network connection—often a physically restricted connection that uses only one fiber-optic strand from a transmit/receive pair. By only using TX optics, it is physically impossible for any digital communications to occur in a highly sensitive network area containing control system devices, while supervisory data may be allowed to communicate out of that highly secure enclave into the SCADA DMZ or beyond. In certain instances, such as for the storage of highly sensitive documents, the diode may be reversed, such that information can be sent into a secure enclave that is then physically prevented from communicating that information back outside of the enclave.

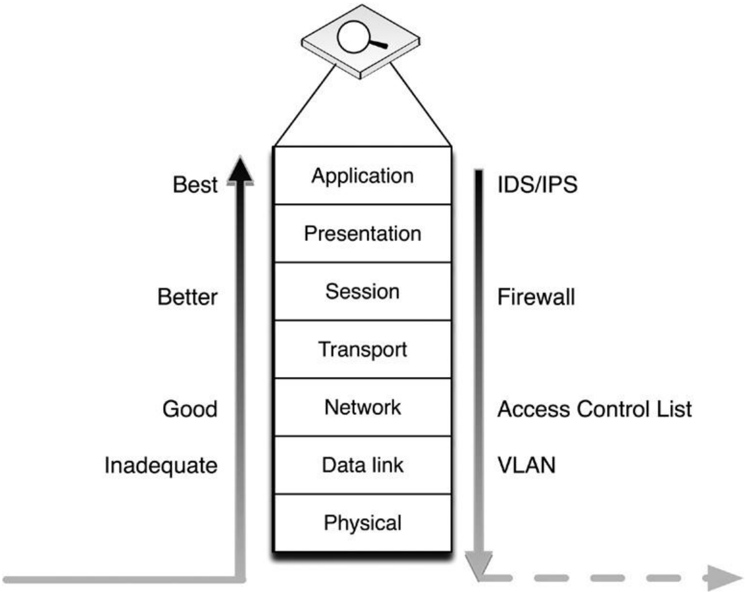


FIGURE 7.14

Relative Capabilities of Common Security Devices.

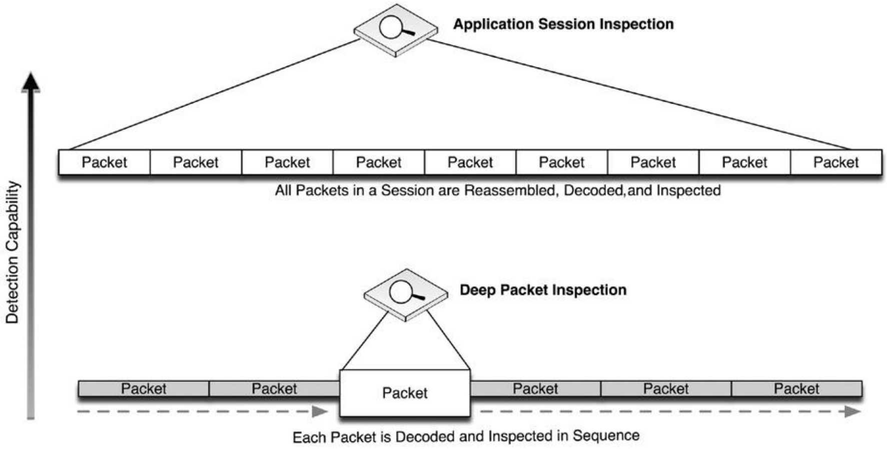

Application Session Inspection
Detection Capability
Packet
Packet
Packet
Packet
Packet
Packet
Packet
Packet
Packet
All Packets in a Session are Reassembled, Decoded, and Inspected
Deep Packet Inspection
Packet
Packet
Packet
Packet
Packet
Packet
Each Packet is Decoded and Inspected in Sequence

FIGURE 7.15

Application Session Inspection vs. Deep Packet Inspection.

Implementing Perimeter Security Devices

Once appropriate security product(s) have been identified, they must be installed and configured appropriately. Luckily, the process of identifying, establishing, and documenting enclaves will simplify this process. The following guidelines will help to configure firewalls, IDS/IPS devices, and application monitors using the variables defined earlier under "Establishing Enclaves."

Firewall Configuration Guidelines

Firewalls control communication using a defined configuration policy, typically consisting of `Accept` (allow) and `Drop` (deny) statements. Most firewalls will enforce a configuration in sequence, such that starting with a broadly defined policy, such as `Deny All`, which will drop all inbound traffic by default. These broad rules can then be overruled by subsequent, more focused rules. Therefore, the following firewall policy would only allow a single IP address to communicate outside of the firewall on port 80 (HTTP).

```
Deny All
Allow 10.0.0.2 to Any Port 80
```

NOTE

Firewall rule examples are written generically so that they can be more easily understood. Depending on the firewall used, specific rule syntax may have to be used, while some firewalls are configured exclusively via a graphical user interface.

Determining what rules should be configured is typically easier in an industrial network because the nature of an industrial network is such that there is no need to accommodate the full diversity of applications and services typically found in an enterprise network. This is especially true when configuring a specific firewall against a specific enclave: the enclave will by its nature be limited in scope, resulting in concise firewall policies. The method of properly configuring an enclave firewall is as follows:

1. Begin with bidirectional `Deny All` rules.
2. Configure specific exceptions, using the defined variables `$ControlSystem_Enclave01_Devices` and `$ControlSystem_Enclave01_PortsServices`.
3. Verify that all `Allow` rules are explicitly defined (i.e., no `All` rules).

One simple way to configure a firewall is to follow the guidelines of the National Infrastructure Security Coordination Center (NISCC) "Good Practice Guide on Firewall Deployment for SCADA and Process Control Networks," using the defined enclave variables as detailed in Table 7.2.[17]

Table 7.2 NISCC Firewall Configuration Guidelines with Enclave Variables[a]

NISCC Recommendations	Example Rule Using Enclave Variables	Notes
Start with universal exclusion as a default policy	`Deny All / Permit None`	Firewalls should explicitly deny all traffic inbound and outbound as the default policy.

Ports and services between the control system environment and an external network should be enabled and permissions granted on a specific case by case basis	`Allow 10.2.2.120 port 162 to 192.168.1.15 port 162`
	`#Allow SNMP traps from router ip 10.2.2.120 to network management station ip 192.168.1.15, authorized by John Doe on April 1 2005`
		Comments used within the firewall configuration file can be used to document special cases, permissions, and other details.

All "permit" rules should be both IP address and TCP/UDP port specific, and stateful if appropriate, and shall restrict traffic to specific IP address or range of addresses	N/A	This guideline can be enforced by using `$ControlSystem_Enclave01_Devices` and `$ControlSystem_Enclave01_PortsServices` to define rules.
All traffic on the SCADA and DCS network(s) are typically based only on routable IP protocols, either TCP/IP or UDP/IP; thus, any non-IP protocol should be dropped	N/A	By using `$ControlSystem_Enclave01_PortsServices` within all defined rules, only protocols explicitly allowed within that enclave will be accepted by the firewall, and all others will be dropped by the overarching `Deny All` rule.

Prevent traffic from transiting directly from the Process Control / SCADA network to the enterprise network; all traffic should terminate in the DMZ	`Deny [Not $Neighboring Enclave1, Not $Neighboring Enclave2] to $ControlSystem_Enclave01_Devices`
	`Deny $ControlSystem_Enclave01_Devices to [Not $Neighboring Enclave1, Not $Neighboring Enclave2]`
		By configuring a rule on each enclave that explicitly denies all traffic to and from any enclave that is NOT a neighboring enclave will prevent any transitive traffic. All traffic will need to be terminated and reestablished using a device local to that enclave.

Any protocol allowed between the DCS and the SCADA DMZ is explicitly NOT allowed between SCADA DMZ and enterprise networks (and vice versa)	At the demarcation between the enterprise network and SCADA DMZ:
	`Deny $ControlSystem_Enclave01_PortsServices to $EnterpriseNetwork_Enclave01_Devices`
	At the demarcation between the DCS and SCADA DMZ:
	`Deny $EnterpriseNetwork_Enclave01_PortsServices to $ControlSystem_Enclave01_ Devices`
		These rules enforce the concept of "disjointing" protocols, and further prevents transitive communication from occurring across an enclave.

Allow outbound packets from the PCN or DMZ only if those packets have a correct source IP address assigned to the PCN or DMZ devices	N/A	Explicitly defined `Deny All` rules combined with explicitly defined known-good IP addresses using `$ControlSystem_Enclave01_Devices` ensures that all outbound packets are from a correct source IP. Firewalls may also be able to detect spoofed IP addresses. In addition, network activity monitoring using a Network Behavior Anomaly Detection (NBAD), Security Information and Event Management (SIEM), or Log Management solution may be able to detect instances of a known-good IP address originating from an unexpected device based on MAC Address or some other identifying factor (see Chapter 9, "Monitoring Enclaves")
Control network devices should not be allowed to access the Internet	At the Internet firewall: `Deny [$ControlSystem_Enclave01_ Devices, $ControlSystem_Enclave02_ Devices, $ControlSystem_Enclave03_ Devices, $ControlSystem_Enclave04_ Devices]`	Because all devices in all enclaves have been identified and mapped into variables, these devices can be explicitly denied at the Internet firewall.
Control system networks shall not be directly connected to the Internet, even if protected via a firewall	N/A	Using the enclave approach, no control system should be directly connected to the Internet (see "Establishing Enclaves").
All firewall management traffic be: **1.** Either via a separate, secured management network (e.g., out of band) or over an encrypted network with two-factor authentication **2.** Restricted by IP address to specific management stations	N/A	This recommendation supports the establishment of a Firewall Management enclave using the methods described earlier under "Establishing Enclaves." By placing all firewall management interfaces and management stations in an enclave, which is isolated from the rest of the network, the traffic can be kept separate and secured.

[a] *National Infrastructure Security Coordination Center, NISCC Good Practice Guide on Firewall Deployment for SCADA and Process Control Networks. British Columbia Institute of Technology (BCIT). February 15, 2005.*

Intrusion Detection and Prevention (IDS/IPS) Configuration Guidelines

IDS and IPS devices inspect network packets for signs of malicious code or exploits. Intrusion Detection refers to passive inspection. An IDS examines packets and compares them against a set of detection signatures, and issues an alert when there is a match. Intrusion Prevention refers to active inspection, where traffic is matched against IDS rules, but where specific actions can be taken in addition to alerting. IDS actions can include `Alert` (generate a custom message and log the packet), `Log` (log the packet), and `Pass` (ignore the packet), while IPS actions can also include `Drop` (drop the packet and log it), `Reject` (drop the packet and initiate a TCP reset to kill the session), and `sDrop` (drop the packet, but do not log it). In addition, both IDS and IPS rules can use the `Activate` and `Dynamic` actions, the former of which activates another rule, and the latter of which remains idle until activated by an `Activate` rule.[18]

Both IDS and IPS devices can be deployed either out-of-line using a network span or tap port or in-line using two network interfaces, although an IPS can only actively block traffic if it is deployed in-line.

An enabled collection of IDS/IPS detection signatures is referred to as an IDS/IPS policy, and this policy will dictate what types of threats may be detected by the device, as well as the degree and scope of events that will be generated. While active blocking of malicious traffic is important, the IDS/IPS events that are generated can also be analyzed to provide other important indicators—including network behavior, larger threat incidents, etc. (see Chapter 9, "Monitoring Enclaves"). Signatures generally follow a format similar to a firewall rule, where there is an identified source and destination address and/or port, as well as an action. In addition, IDS/IPS signatures may match against specific contents of a packet, looking for patterns within the packet that indicate a known exploit (i.e., a "signature"). Common IDS/IPS signature syntax follows the de facto standards defined by Snort, an open-source IDS project owned by SourceFire. An example signature is written as follows:

```
[Action] [Protocol] [Source Address] [Source Port] [Direction
Indicator] [Destination Address] [Destination Port] [Rule Options]
```

which when written in correct syntax looks like

```
drop tcp 10.2.2.1 80 -> 192.168.1.1 80 (flags: <optional snort
flags>; msg: "<message text>"; content: <this is what the rule is
looking for>; reference: <reference to external threat source>;)
```

To highlight the difference between a firewall rule and an IDS/IPS signature, consider the following example:

```
drop tcp 10.2.2.1 80 -> any any
```

Without any rule options, the previous rule is essentially the same as the firewall rule `Deny 10.2.2.1port 80`, which would block all traffic originating from 10.2.2.1 on port 80, effectively preventing that user from accessing the web (via HTTP port

80). However, the ability to match packet contents within the rule options enables an IDS/IPS device to control traffic at a much more granular level, such as

```
drop tcp 10.2.2.1 80 -> any any (msg: “drop http POST”; content:
“POST”;)
```

This rule functions differently, only dropping traffic from the source address in question if the HTTP traffic contains a `POST` request (used by many web forms or applications attempting to upload a file to a web server over HTTP).

NOTE

IDS/IPS rule examples are written using Snort syntax, as it is the de facto signature creation language. However, many IDS or IPS devices support proprietary rule syntax, GUI rule editors, or other rule creation methods. Depending on the product used, the example rules in this book may or may not function as intended. All rules should always be tested prior to deployment.

As with a firewall configurations, determining the exact IDS/IPS policy to be enforced is the first step in correctly configuring the device. Also as with firewalls, the enclave variables defined earlier under “Establishing Enclaves” are valuable tools that can be used to write succinct and highly relevant signatures. However, unlike a firewall which starts with a simple `Deny All` rule, an IDS/IPS should be deployed “large”—with many active signatures—and then pruned back to the specific requirements of the enclave. A method of properly configuring an IDS/IPS is as follows:

1. Begin with a more robust signature set, with many active rules.
2. If a protocol or service is not allowed in the enclave, replace any specific detection signatures associated with that protocol or service with a broader rule that will block all traffic from that protocol or service (i.e., drop unauthorized ports and services).
3. If a protocol or service is allowed in the enclave, keep all detection signatures associated with that protocol or service active.

   3a. For all active signatures, assess the appropriate action, using Table 7.3.
4. Keep all IDS signatures current and up to date.

Remember that an IDS or IPS can be used in a purely passive mode, to analyze traffic that is allowed, including traffic within an enclave (that is, between two devices within the same enclave, that do not cross an enclave perimeter). Passive monitoring will generate alerts and logs that can be useful in many security operations, including forensic investigations, threat detection, and compliance reporting (see Chapter 9, “Monitoring Enclaves,” and Chapter 10, “Standards and Regulations”).

IDS/IPS rules should be tailored to the appropriate enclave using the variables defined in “Establishing Enclaves.” A typical Snort variable is established using the `var` command, as follows:

Table 7.3 Determining Appropriate IDS/IPS Actions

Allowed Port or Service?	Source	Destination	Criticality of Service	Severity of Event	Recommended Action	Note
No	Any	Any	Any	Any	`Reject`	Any communication not explicitly allowed within the enclave should be `Rejected` to disrupt unauthorized sessions and deter an attack.
Yes	Inside Enclave	Inside Enclave	High	Any	`Alert`	Active blocking or rejection of traffic that originates and terminates within an enclave could impact operations. For example, a false positive could result in legitimate control system traffic being blocked or rejected.

Yes	Inside Enclave	Inside Enclave	Low	Any	`Alert`
					or
					`Pass`
						For noncritical services, logging is recommended but not necessary (`Alert` actions will provide valuable event and packet information that could assist in later incident investigations).

Yes	Outside Enclave	Inside Enclave	High	Low (events from obfuscated detection signatures or informational events)	`Alert`	Many detection signatures are broad to detect a wider range of potential threat activity. These signatures should `Alert` only to prevent unintentional interruption of control system operations.

Yes	Outside Enclave	Inside Enclave	High	High (explicit malware or exploit detected by a precisely tuned signature)	Block, Alert	If inbound traffic to a critical system or asset contains known malicious payload, the traffic should be blocked to prevent outside cyber incidents or sabotage.
Yes	Inside Enclave	Outside Enclave (explicitly allowed destination address)	Any	Any	Alert	This traffic is most likely legitimate. However, alerting and logging the event will provide valuable event and packet information that could assist in later incident investigations.
Yes	Inside Enclave	Outside Enclave (unknown destination address)	Any	Any	Block or Reset	This traffic is most likely illegitimate. Generated alerts should be addressed quickly: if the event is a false positive, necessary traffic could be unintentionally blocked; if the event is a threat, it could indicate that the enclave has been breached.

```
var VARIABLE_NAME <alphanumeric value>.
```

The `var` command can be used ubiquitously, or specialized `ipvar` and `portvar` can be used exclusively for IP addresses and ports, respectively.[19] In the enclave method described earlier under "Establishing Enclaves," variables would be defined as

```
ipvar ControlSystem_Enclave01_Devices [192.168.1.0/24, 10.2.2.0/29]
var ControlSystem_Enclave01_Users [jcarson, jrhewing, kdfrog,
mlisa]
portvar ControlSystem_Enclave01_PortsServices [502, 20000]
```

These variables can then be used extensively throughout the active detection signatures. For example, a signature designed to detect a known SCADA buffer overflow attack that is available within the Metasploit framework might appear as follows. (The following rule has been deliberately obfuscated; the complete rule can be obtained from Digital Bond at www.digitalbond.com.)

```
alert tcp !$ControlSystem_Enclave01_Devices -> $ControlSystem_
Enclave01_Devices 20222 (msg: "SCADA ODBC Overflow Attempt";
content: <long string in the second application packet in a TCP
session>; reference:cve,2008-2639; reference:url, http://www
.digitalbond.com/index.php/research/ids-signatures/m1111601/;
sid:1111601; rev:2; priority:1;)
```

NOTE

Many Snort rules reference the `$HOME_NET` or `$MY_NET` variable. The use of multiple `$ControlSystem_Enclave01_Devices` variables (one for each defined enclave) accomplishes the same purpose, effectively defining a unique `$HOME_NET` for each enclave. The nomenclature of `$ControlSystem_Enclave01_Devices` is deliberately verbose in order to easily identify the variable's contents, so that the examples within this book are easier to understand.

Additional examples include signatures designed to specifically block known infection vectors used by Stuxnet.[20] The first example looks for one of the early delivery mechanisms for the Stuxnet malware: specifically, a shortcut image file delivered via a WebDav connection. The second example detects Semens WinCC connection attempts, used in early Stuxnet infection phases.

```
tcp !$ControlSystem_Enclave01_Devices $HTTP_PORTS ->
$ControlSystem_Enclave01_Devices any (msg: "Possible Stuxnet
Delivery: Microsoft WebDav PIF File Move Detected"; flow:from_
server; content: "MOVE"; offset:0; within:5; content: ".pif";
distance:0; classtype:attempted-user; reference:cve, 2010-
2568; reference:osvdb,66387; reference:bugtraq,41732;
reference:secunia,40647; reference:research,20100720-01;
sid:710072205; rev:1;)
```

```
tcp any any -> any 1433 (msg: "Possible Stuxnet Infection: Siemens
Possible Rootkit.TmpHider connection attempt"; flow:to_server;
content: "Server=|2e 5c|WinCC|3b|uid=WinCCConnect|3b|pwd=2W
SXcder”; classtype:suspicious-login; reference:cve,2010-2772;
reference:osvdb,66441; reference:bugtraq,41753; sid:710072201;
rev:2;)
```

Recommended IDS/IPS Rules

Basic recommendations for IDS/IPS configuration include active `block` rules to

1. Prevent any undefined traffic from crossing enclave boundaries (where the disruption of the communication will not impact the reliability of a legitimate service).
2. Prevent any defined traffic containing malware or exploitation code from crossing enclave boundaries.
3. Detect and log suspicious or abnormal activity within an enclave (see “Securing Enclave Interiors” and Chapter 9, “Monitoring Enclaves”).
4. Log normal or legitimate activity within an enclave, which may be useful for compliance reporting (see Chapter 10, “Standards and Regulations”).

> **CAUTION**
>
> A false positive (a rule that triggers in response to unintended traffic, typically due to imprecisions in the detection signature) can block legitimate traffic and in a control system legitimate traffic could represent a necessary operational control. Only use `block` IPS rules where absolutely necessary, and only after extensive testing.

The greater the extent of functional isolation and separation into defined enclaves, the more concise and effective the IDS/IPS policy will be. Some basic IDS and IPS rules suitable for use in enclave perimeters include the following:

- `Block` any industrial network protocol packets that are the wrong size or length.
- `Block` any network traffic that is detected inbound to or outbound from any enclave where that is not expected or allowed.
- `Block` any industrial network protocol packets that are detected in any enclave where that protocol is not expected or allowed.
- `Alert` any authentication attempts, in order to log both successful and failed logins.
- `Alert` any industrial network port scans.
- `Alert` any industrial network protocol function codes of interest, such as:
  - “Write” functions, including codes that write files or that clear, erase, or reset diagnostic counters.
  - “System” functions, including codes that stop or restart a device.
  - “System” functions that disable alerting or alarming.
  - “Read” functions that request sensitive information.
  - “Alarm” or “Exception” codes and messages.

While SCADA IDS/IPS devices may be able to detect and trigger upon industrial network protocol function codes and commands, specialized application monitoring devices may be more capable of analyzing the contents of application layer protocols.

> **CAUTION**
>
> IDS and IPS signatures are only able to block known threats, meaning that the IDS/IPS policy must be kept current in order to detect more recently identified attacks (virus, exploits, etc). Therefore, IDS/IPS products must be included within the overall Patch Management Strategy in order for the devices to remain effective (see Chapter 6, "Vulnerability and Risk Assessment").

Anomaly based Intrusion Detection

So far, only signature-based detection has been discussed. However, many IDS and IPS systems also support detection based on anomaly detection. Anomaly detection uses statistical models to detect when something unusual is happening, on the premise that unexpected behavior could be the result of an attack.

The exact capabilities will vary from product to product, as there is no standard anomaly detection mechanism. Theoretically, anything monitored by the IDS could be used for anomaly detection. Because network flows are highly quantifiable, anomaly detection is often used to identify abnormal behavior in what devices are communicating, and how. Referred to as Network Anomaly Detection, these systems are able to detect a sudden increase in outbound traffic, an increase in sessions, an increase in total bytes transmitted, an increase in the number of unique destination IP addresses, or other quantifiable metrics.

Anomaly detection is useful because it does not require an explicitly defined signature in order to detect a threat. This allows anomaly detection systems to identify zero day attacks or other threats for which no detection signature exists. At the same time, however, anomaly detection trends toward a higher number of false positives, as a benign change in behavior can lead to an alert. It is for this reason that anomaly-based threat detection is typically used passively, generating alerts rather than actively blocking suspect traffic.

In industrial networks—especially in well-isolated control system enclaves—network behavior tends to be highly predictable, making anomaly detection more reliable.

Anomaly detection systems may be referred to as "rule-less" detection systems. This is because they do not pattern match against a defined signature, although they do use rules. However, unlike a normal IDS rule, anomaly rules are often based on thresholds and/or statistical deviations, such as in the following example:

```
TotalByteCount from $Control_System_Enclave01_Devices increases by
>20%
```

An example of a threshold rule would use a hard upper- or lower-limit, most likely derived automatically by the anomaly detection system:

```
TotalDestinationIPs > 34
```

As a general guideline, the greater the variation of the network traffic being monitored, the greater the chances of anomaly detection rules to generate a false positive.

Anomaly detection can be used across devices as well, using an information consolidation tool such as a Security Information and Event Management (SIEM) system. This system-level anomaly detection is discussed in more detail in Chapter 8, "Exception, Anomaly, and Threat Detection."

Application and Protocol Monitoring in Industrial Networks

Because many industrial operations are controlled using specialized industrial network protocols that issue commands, read and write data, etc. using defined function codes, specialized devices can leverage that understanding along with Firewall, IDS, and IPS technology to enforce communications based on the specific operations being performed across the network.

In addition to the inspection of industrial protocol contents (e.g., DNP3 function codes), the applications themselves—the software that controls how those protocols are used—can also be inspected. This degree of Application Monitoring, also referred to as Session Inspection, allows the contents of an application (e.g., HMI, Web Browser) to be inspected even though it might exist across a large number of individual packets. That is, inspection can occur up to and include the contents of a file being transferred to a PLC, a virus definition downloaded from a web browser of update server, etc. Application Monitors provide a very broad and very deep look into how network traffic is being used, and are therefore especially useful in environments where both control systems and enterprise protocols and applications are in use.

Many specialized security devices are available for SCADA and control system environments that use either application or protocol monitoring to this degree. At the time of this writing, these devices include the Tofino Industrial Security Appliance and the Secure Crossing Zen Firewall, as well as other broader-use enterprise Application Data Monitors. The two former devices were designed specifically to identify the operations being performed within industrial protocols, to prevent unauthorized operations. The latter refers to a more general-purpose enterprise security appliance, which is able to support the most common industrial network protocols. Each of these specialized devices has specific strengths and weaknesses, which are summarized in Table 7.4.

Because these devices are highly specialized, configurations can vary widely. In general terms, a firewall capable of SCADA protocol inspection may utilize a rule as follows to block any protocol function from writing a configuration or register, or executing a system command (such as a device restart):

```
Deny [$ControlSystem_ProtocolFunctionCodes_Write,
$ControlSystem_ProtocolFunctionCodes_System]
```

Table 7.4 A Comparison of Industrial Security Devices

Security Product	Functionality	Strengths	Weaknesses	Rule Example
SCADA Firewall	Traffic policy enforcement	Enables separation of networks, ports and services	Does not block hidden threats or exploits within "allowed" traffic	Allow only TCP port 502 (Modbus TCP)
SCADA IDS/IPS	Detects malware and exploits within traffic	Prevents exploitation of vulnerabilities via authorized ports and services	"Blacklist" methodology can only detect and block known threats	Block Modbus packets containing known malware code
SCADA UTM or hybrid security appliance	Combines firewall, IDS/IPS, VPN, and other security functions	Combination of security functions facilitates "defense in depth" via a single product	Security functions maintain their component weaknesses (i.e., the whole is equal to but not greater than the sum of its parts)	Allow only TCP port 502 with "read only" function codes Allow outbound TCP 502 only via encrypted VPN to other SCADA enclaves
SCADA Content Firewall or Application Firewall	Traffic policy enforcement	Enables content-based traffic separation, based on industrial network protocols	Assesses content of a single packet only (lacks session reassembly or document decode)	Allow only "Read only" Modbus TCP functions
Deep Session Inspection (application content monitoring)	Session Reassembly File/Content Decode File/Content Capture	Functions of a SCADA content firewall, plus visibility into full application session and document contents to detect APT threats and insider data theft; provides strong security in hybrid enterprise/industrial areas such as SCADA DMZ	Typically limited to TCP/IP inspection, making session inspection less suitable for deployment in pure control system environments	Alert on Modbus TCP traffic on ports other than TCP 502 Alert on any traffic with base64-encoded content
Network Whitelist	Allows only defined "good" traffic	Prevents all malicious traffic by allowing only known, good traffic to pass	Requires proper baselining of correct network behavior	Can make legitimate changes in network operations more difficult

An IDS capable of SCADA protocol inspection may utilize a rule as follows, which looks for a specific function code within a DNP3 packet:

```
tcp any any -> $ControlSystem_Enclave01_Devices 502 (msg: “DNP
function code 15, unsolicited alarms disabled”; content:“|15|”;
offset:12; rev:1;)
```

In contrast, an application monitor performing full session decode may use syntax similar to the following rule to detect windows .LNK files within application traffic, which could indicate a possible Stuxnet delivery attempt.

```
FILTER_ID=189
NORM_ID=830472192
ALERT_ACTION=log-with-metadata
ALERT_LEVEL=13
ALERT_SEVERITY=10
DESCRIPTION=A Microsoft Windows .LNK file was detected
EXPRESSION=(objtype==application/vnd.ms-lnk)
```

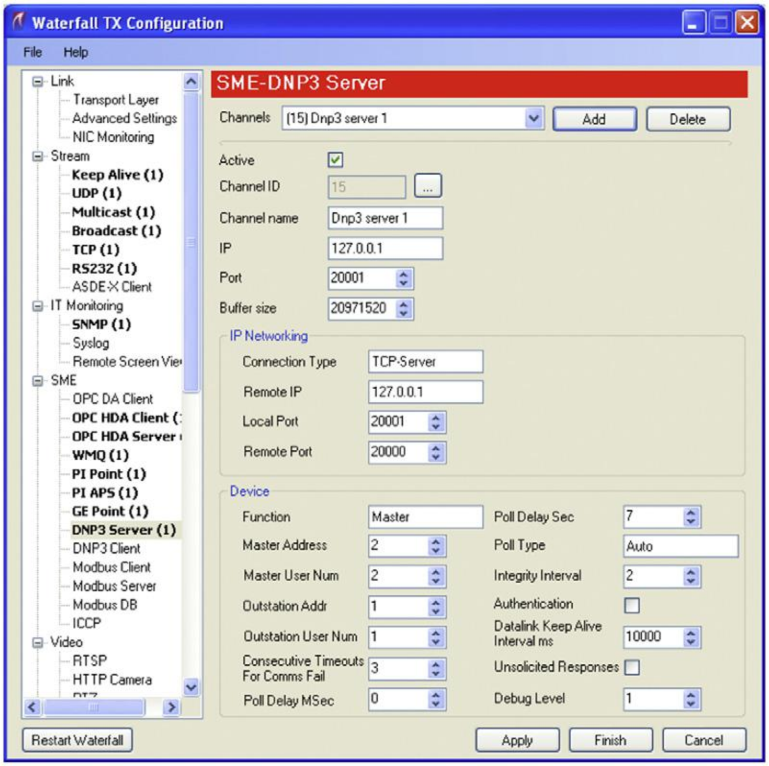
Data Diodes and Unidirectional Gateways

Data diodes and unidirectional gateways work by physically preventing return communications over a fiber-optic connection, typically through the physical removal of the RX optics. This provides absolute physical layer security at the sake of bidirectional communications. Because the connection in one direction does not exist, data diodes are true air gaps, albeit in only one direction.

Because many network applications and protocols require bidirectional communication (such as TCP/IP, which requires a variety of handshakes and acknowledgments to establish and complete a session), considerations should be taken when using data diodes in order to ensure that the remaining one-way data path is capable of transferring the required traffic. To accommodate this concern, many data diode vendors implement a software-based solution, where the physical diode exists between two servers. These servers support a variety of bidirectional applications, so that the bidirectional requirements can be met fully at the transmitting end, and so that the receiving end can then spoof the behavior of the original transmitter—essentially tricking the application to operate over a one-way link. This allows an additional level of control over the applications and services that can be transmitted over the diode or gateway. An example of enabling DNP3 services over a unidirectional gateway is shown in Figure 7.16. While data diodes are physical layer devices that do not require any specific configuration, the communication servers may need to be correctly configured before these applications work correctly over the diode.

SECURING ENCLAVE INTERIORS

Unlike enclave perimeters, which by their definition have a clear point of demarcation that can be monitored and controlled, enclave interiors consist of specific



SME-DNP3 Server

Channels: [15] Dnp3 server 1

Active ☑

Channel ID: 15

Channel name: Dnp3 server 1

IP: 127.0.0.1

Port: 20001

Buffer size: 20971520

IP Networking

Connection Type: TCP-Server

Remote IP: 127.0.0.1

Local Port: 20001

Remote Port: 20000

Device

Function	Master	Poll Delay Sec	7
Master Address	2	Poll Type	Auto
Master User Num	2	Integrity Interval	2
Outstation Addr	1	Authentication	□
Outstation User Num	1	Datalink Keep Alive Interval ms	10000
Consecutive Timeouts For Comms Fail	3	Unsolicited Responses	□
Poll Delay MSec	0	Debug Level	1

FIGURE 7.16

Enabling DNP3 over a Unidirectional Gateway.

devices as well as a variety of network communications between those devices. Securing an enclave's interior is primarily accomplished through host-based security, which controls end-user authentication to a device, how that device communicates on the network, what files are accessed by that device, and what applications may be executed by it. Although monitoring the communications between hosts within an enclave is also useful for detecting threats, this is discussed in Chapter 9, "Monitoring Enclaves," and will not be discussed in this chapter.

This chapter discusses three distinct areas of host security, including:

- Access Control, including user authentication and service availability
- Host-Based Network Security, including host firewalls and host intrusion detection systems (HIDS)
- Anti-Malware systems such as Anti-Virus (**AV**) and application whitelists (**AWL**)

Table 7.5 Varying Levels of Host Security Options

Device	Suitable Security Measures
HMI or similar device running a modern operating system. Application is not time sensitive	• Host firewall • HIDS • Anti-Virus or Application Whitelist • Disable all unused ports and services
HMI or similar device running a modern operating system. Application is time sensitive	• Host firewall • Disable all unused ports and services
PLC, RTU, or similar device running an embedded commercial OS	• Host firewall or HIDS if available • External security controls
PLC, RTU, IED, or similar device running an embedded operating environment	• External security controls

Selecting Interior Security Systems

As a matter of best practices, all host access controls and host network security solutions should be implemented on all networked devices. However, not all network devices are capable of running additional security software, and in some cases the software may incur latency or unacceptable processor overhead. Table 7.5 shows which devices are typically capable of running the common methods of host security.

Where possible, one option of each type—access control, network security, and Anti-Malware—should be used. Especially where host security options are not possible, an external security control should be implemented.

CAUTION

Major control system asset vendors often recommend and/or support the use of particular host security options and may even perform regression testing to validate authorized tools.[21] This is an important consideration, especially when utilizing time-sensitive applications that could be affected by delay. In addition, many control system assets may use proprietary extensions or modifications of commercial operating systems that may conflict with some host security solutions.[22] Therefore, asset vendors should always be consulted prior to the installation of a commercial host security product.

Host Firewalls

A host firewall works just like a network firewall, and acts as an initial filter between the host and any attached network(s). The host firewall will allow or deny inbound traffic based on the firewall's specific configuration. Typically, host firewalls are session-aware firewalls that allow control over distinct inbound and outbound application sessions.

As with network firewalls, host firewalls should be configured according to the guidelines presented under "Firewall Configuration Guidelines": starting with `Deny`

`All` policies, and `Allow` rules should only be added for the specific ports and services used on that particular asset.

Host IDS

Host IDS (HIDS) systems work like Network IDS, only they reside on a specific asset and monitor systems internal to that asset. Typically, HIDS devices may monitor system settings and configuration files, applications, and/or sensitive files.[23] These devices are differentiated from Anti-Virus and other host security options in that they can perform network packet inspection, and can therefore be used to directly mimic the behavior of a Network IDS by monitoring the host systems network interface(s) to detect or prevent inbound threats. HIDS can therefore be configured using the guidelines presented under "Intrusion Detection and Prevention (IDS/IPS) Configuration Guidelines." Because an HIDS may also be able to inspect local files, the term is sometimes used for other host-based security devices such as Anti-Virus systems, or propriety host security implementations that provide overlapping security functions.

As with Network IDS, an HIDS device will generate alerts detailing any violations of the established policy. If the system is able to actively block the violation, it may be referred to as a Host IPS (**HIPS**).

> **CAUTION**
>
> Like Network based IDS/IPS systems, HIDS products require regular signature updates in order to detect more recently identified threats. HIDS should therefore be included in the overall Patch Management Strategy (see Chapter 6, "Vulnerability and Risk Assessment").

Anti-Virus

Anti-Virus systems are designed to inspect files for malware. They work similarly to an IDS (and IDS systems can be used to detect malware), using signature-based detection to validate system files. When a signature matches known indications of a virus, Trojan, or other malware, the suspect file is typically quarantined so that it can be cleaned or deleted.

> **CAUTION**
>
> Like other signature-based detection systems, Anti-Virus systems require regular signature updates. Anti-Virus systems should therefore be included in the overall Patch Management Strategy (see Chapter 6, "Vulnerability and Risk Assessment").

Application Whitelisting

Application whitelisting (AWL) offers a different approach to host security than traditional HIDS, Anti-Virus, and other "blacklist" technologies. A "blacklist" solution compares the monitored object to a list of what is known to be bad. This presents two issues: the first is that the blacklist must be continuously updated as new threats are discovered; the second is that there is no way to detect or block certain attacks, such as zero-days, and/or known attacks for which there is no available signatures. In contrast, a "whitelist" solution creates a list of what is known to be good and applies very simple logic: if it is not on the list, block it.

AWL solutions apply this logic to the applications on a host. In this way, even if a virus or Trojan does penetrate the control system's perimeter defenses and finds its way onto a target system, the host itself will stop that malware from executing—rendering it inoperable.

AWL is well suited for use in control systems, where an asset should have explicitly defined ports and services. In addition, there is no need to continuously download, test, evaluate, and install signature updates. Rather, the AWL only needs to be updated and tested when the applications used on the host system are updated.

However, because AWL operates at the lowest levels of an operating environment, it introduces new code into the execution paths of all applications and services on that host. This adds latency to all functions of the host, which may cause unacceptable delay for time-sensitive operations, and requires full regression testing.

External Controls

When it is simply not possible to use host-based security tools, external tools may be required. For example, certain IDS, Firewalls, and other network security devices that are specialized for control system operations may be used to monitor and protect these assets. Many of these devices support serial as well as Ethernet interfaces, and can be deployed directly in front of a specific device or group of devices, including deployment within a specific process or loop.

Other external controls, such as Security Information and Event Management systems, may monitor a control system more holistically, using information available from other assets (such as an MTU or HMI), from other information stores (such as a Data Historian), or from the network itself. This information can be used to detect risk and threat activity across a variety of systems. This will be discussed more in Chapter 9, "Monitoring Enclaves."

External controls, especially passive monitoring and logging, can also be used to supplement those assets that are already secured via a host firewall, host-based IDS, Anti-Virus, AWL, etc.

SUMMARY

Through the identification and isolation of functional groups, quantifiable security enclaves can be defined. These enclaves can and should be secured at both the enclave perimeter and within the enclave interior, using a variety of tools including both network- and host-based firewalls, network- and host-based intrusion detection and prevention systems (IDS/IPS), Application Monitoring, Anti-Virus, and/or Application whitelisting (AWL).

In addition to the direct security benefits of these various controls, each also provides useful alerting capabilities. The information collected from these and other devices can be used to identify and establish baseline behavior, and thereafter to detect exceptions and anomalies (see Chapter 8, "Exception, Anomaly, and Threat Detection"). Logs and events from these enclave security measures are also useful for overall activity and behavior monitoring (see Chapter 9, "Monitoring Enclaves").

ENDNOTES

1. U.S. Nuclear Regulatory Commission, Regulatory Guide 5.71 (New Regulatory Guide), Cyber Security Programs for Nuclear Facilities, January, 2010.
2. D. Taylor, Intrusion detection FAQ: are there vulnerabilities in VLAN implementations? VLAN Security Test Report, The SANS Institute. <http://www.sans.org/security-resources/idfaq/vlan.php>, July 12, 2000 (cited: January 19, 2011).
3. U.S. Nuclear Regulatory Commission, 73.54 Protection of digital computer and communication systems and networks. <http://www.nrc.gov/reading-rm/doc-collections/cfr/part073/part073-0054.html>, March 27, 2009 (cited: January 19, 2011).
4. North American Reliability Corporation, Standard CIP-002-3. Cyber Security—Critical Cyber Asset Identification. <http://www.nerc.com/files/CIP-002-3.pdf>, December 16, 2009 (cited: January 19, 2011).
5. Ibid.
6. North American Reliability Corporation, Standard CIP-005-3. Cyber Security—Electronic Security Perimeter(s). <http://www.nerc.com/files/CIP-005-3.pdf>, December 16, 2009 (cited: January 19, 2011).
7. Ibid.
8. North American Reliability Corporation, Standard CIP-003-3. Cyber Security—Security Management Controls. <http://www.nerc.com/files/CIP-003-3.pdf>, December 16, 2009 (cited: January 19, 2011).
9. North American Reliability Corporation, Standard CIP-005-3. Cyber Security—Electronic Security Perimeter(s). <http://www.nerc.com/files/CIP-005-3.pdf>, December 16, 2009 (cited: January 19, 2011).
10. International Society of Automation, Standard ANSI/ISA-99.02.01-2009, Industrial Automation and Control System Security.
11. Department of Homeland Security, Risk-Based Performance Standards Guidance, Chemical Facility Anti-Terrorism Standards, May, 2009.
12. U.S. Nuclear Regulatory Commission, Regulatory Guide 5.71 (New Regulatory Guide), Cyber Security Programs for Nuclear Facilities, January, 2010.
13. North American Reliability Corporation, Standard CIP-008-3. Cyber Security—Incident Reporting and Response Planning. <http://www.nerc.com/files/CIP-008-3.pdf>, December 16, 2009 (cited: January 19, 2011).
14. International Society of Automation, Standard ANSI/ISA-99.02.01-2009, Industrial Automation and Control System Security.
15. Department of Homeland Security, Risk-Based Performance Standards Guidance, Chemical Facility Anti-Terrorism Standards, May, 2009.

16. U.S. Nuclear Regulatory Commission, Regulatory Guide 5.71 (New Regulatory Guide), Cyber Security Programs for Nuclear Facilities, January, 2010.
17. National Infrastructure Security Coordination Center, NISCC Good Practice Guide on Firewall Deployment for SCADA and Process Control Networks, British Columbia Institute of Technology (BCIT), February 15, 2005.
18. Snort.org, SNORT Users Manual 2.9.0. <http://www.snort.org/assets/156/snort_manual.pdf>, December 2, 2010 (cited: January 19, 2011).
19. Ibid.
20. NitroSecurity, Inc., Network Threat and Analysis Center, Nitrosecurity.com, January, 2011.
21. K. Stouffer, J. Falco, K. Scarfone, National Institute of Standards and Technology, Special Publication 800-82 (Final Public Draft), Guide to Industrial Control Systems (ICS) Security, Section 6.3.1 Identification and Authentication, September, 2008.
22. Ibid.
23. Ibid.

CHAPTER

Exception, Anomaly, and Threat Detection

8

INFORMATION IN THIS CHAPTER:

- Exception Reporting
- Behavioral Anomaly Detection
- Behavioral Whitelisting
- Threat Detection

By defining enclaves, clear policies about what is allowed and what is not have already been obtained. In addition, the operation of each enclave should be well defined and relatively predictable. This supports two important types of behavioral analysis: exception reporting and anomaly detection.

Exception Reporting refers to an automated system that notifies the security administrator whenever a defined policy has been violated. In the context of enclave-based security, this means a notification that the defined enclave has been violated: some user, system, or service is interacting with the enclave in a way that is contrary to security policies established at the perimeter and/or within the enclave interior (see Chapter 7, "Establishing Secure Enclaves"). If we expect one behavior but see another, we can view this behavior as a potential threat and take action accordingly.

Anomaly Detection picks up where policy-based detection ends, by providing a "rule less" method of identifying possible threat behavior. Simply, anomaly detection takes action when something out of the ordinary happens. In an industrial system—especially if a strong defense-in-depth posture is maintained and enclaves are appropriately separated—the normal behavior can be determined, and variations in that behavior should be minimal. The operational behavior of an industrial network should be relatively predictable, making anomaly detection effective once all "normal" actions have been defined.

The effectiveness of anomaly detection pivots on that basic understanding of behavior, however. Understanding how baseline behavior can be measured is the first step to implementing a usable anomaly detection strategy.

Taken together, clearly defined policies and anomaly detection can provide an additional function: behavioral whitelisting. Behavioral whitelisting combines an understanding of what is known good/bad behavior (policies) with an understanding of expected behaviors, to define what is "known good behavior." Just as whitelists of other known good elements (IP addresses, applications, users, etc.) can be used to enforce perimeter and interior enclave defenses, these higher level behavioral whitelists can help to deter broader threats, even across enclaves.

Although each method is effective on its own, attacks rarely occur in clear, direct paths (see Chapter 6 “Vulnerability and Risk Assessment”). Therefore, to detect more sophisticated threats, all anomalies and exceptions need to be assessed together, along with the specific logs and events generated by network switches, routers, security appliances, and other devices. Event correlation looks across all systems to determine larger threat patterns that can more clearly identify a security incident. However, event correlation is only as good as the data that is available, requiring that all of the above detection techniques are used to generate a comprehensive base of relevant security information. It also requires proper monitoring of networks and devices, as discussed in the next chapter, “Monitoring Enclaves.”

CAUTION

Automated tools for the detection of exceptions, anomalies, and advanced threats are effective measures to help notify security analysts of incidents that may need to be addressed. However, no tool should be trusted completely; the experience and insight of a human analyst is a necessary component in the security monitoring and analysis process. While tools are often sold with the promise of being “an analyst in a box,” even the most well-tuned systems will still produce false positives and false negatives, therefore requiring the additional layer of human intellect to complete the assessment.

EXCEPTION REPORTING

In Chapter 7 (“Establishing Secure Enclaves”), specific policies have been developed and enforced by firewalls, Intrusion Detection System/Intrusion Prevention System (IDS/IPS) devices, application monitors, and other security devices. Apart from the clear examples of when a specific firewall or IPS rule triggers an alert, these policies can be used to assess a variety of behaviors. Exception reporting looks at all behaviors, and unlike a hard policy defined at an enclave perimeter, which makes black-and-white decisions about what is good and bad, exception reporting can detect suspicious activities by compiling a wealth of seemingly benign security events.

This level of assessment could encompass any measurable function of an enclave (or enclaves), including network traffic patterns, user access, operational controls, etc. At a very basic level, exception reporting might be used to inform an operator when something that should not have been allowed (based on enclave perimeter policies) has occurred. The first example in Table 8.1 is an example of this: as it should not be possible for an inbound network communication to originate from an unrecognized IP address—that should have been prevented by the default `Deny All` firewall policy.

Table 8.1 Examples of Suspicious Exceptions

Exception	Policy being Enforced	Detected by	Recommended Action
A network flow originates from a different enclave than the destination IP address	Network separation of functional groups/enclaves	Firewall, Network Monitor, Network IDS/IPS, etc. using `$Enclave_IP` variables	Alert only, to create a report on all interenclave communications
Network traffic originating from foreign IP addresses is seen within a secured enclave	Isolation of critical enclaves form the Internet	Log Manager/Analyzer, SIEM, etc. correlating !`$Enclave_IP` variables and geolocation data	Critical Alert to indicate possible penetration of a secure enclave
An authorized user accessing the network from a new or different IP address	User access control policies	Log Manager/Analyzer, SIEM, etc. correlating `$Enclave_IP` variables to user authentication activity	Alert only, to create a report on abnormal administrator activity
An unauthorized user performing administrator functions	User access control policies	Log Manager/Analyzer, SIEM, etc. correlating !`$Admin_users` variables to application activity	Critical Alert to indicate potential unauthorized privilege escalation
An industrial protocol is used in nonindustrial enclaves	Network separation of functional groups by protocol	Network Monitor, Network IDS/IPS, Application Monitor, Industrial Protocol Monitor, etc. using !`$Enclave_Protocol` variables	Alert only, to create a report of abnormal protocol use
`Write` function codes are used outside of normal business hours	Administrative control policies	Application monitoring detects `$Modbus_Administrator_Functions` Identity or authentication systems indicate normal administrative shifts SIEM or other log analysis tool correlates administrative functions against expected shift hours	Alert only, to create an audit trail of unexpected admin behavior
An industrial protocol using `Write` function codes is originating from a device authenticated to a nonadministrative user	User access control policies	Application monitoring detects `$Modbus_Administrator_Functions` Authentication logs indicate a nonadministrative user SIEM or other log analysis tool correlates authentication logs with control policies and industrial protocol functions	Critical Alert to indicate possible insider threat or sabotage

Other, less obvious uses for exception reporting are exemplified in the last example in Table 8.1, where two completely different detection methods (an application monitoring system and a log analysis system) indicate a policy exception that otherwise might seem benign; the function codes in question are only a concern if being executed by an authorized user.

Exception reporting can be automated using many log analysis or security information management systems, which are designed to look at information (typically log files) from many sources, and correlate this information together (for more information on how to generate this information, see Chapter 9, "Monitoring Enclaves"). Without an understanding of the policies that are in place, however, exceptions cannot be determined.

BEHAVIORAL ANOMALY DETECTION

Sometimes, an exception might be seen in a network's expected behavior, rather than in adherence to a policy. These anomalies can be detected by comparing monitored behavior against known "normal" values. This can be done in a variety of ways: manually, based on real-time monitoring; manually, via log review; automatically, using a Network Behavior Anomaly Detection (NBAD) product, Log Analysis, or Security Information and Event Management (SIEM) tool; or automatically, by exporting data to a dedicated spreadsheet or other statistical application. Whether performed manually or automatically, an anomaly cannot be detected without an established baseline of activity to compare against. Once a baseline has been established for a given metric (such as the volume of network traffic, the number of active users, etc.), that metric must be monitored using one or more of the methods described in Chapter 9, "Monitoring Enclaves."

Measuring Baselines

Baselines are time-lagged calculations based on running averages. They provide a basis (base) for comparison against an expected value (line). Baselines are useful for comparing past behaviors to current behaviors, but can also be used to measure network or application capacity, or almost any other operational metric that can be tracked over time. A baseline should not be confused with a trend analysis—a baseline is a value: nothing more, nothing less. Using that metric in an analysis of past observed behavior and future predicted behavior is a trend analysis, a forward-looking application of known baselines to predict the continuation of observed trends.

A baseline can be simple or complex—anything from a gut understanding of how a system works to a sophisticated statistical calculation of hard, quantifiable data. The simplest method of establishing a baseline is to take all data collected over a period of time and use whatever metric is available to determine the average over time. This is a commonly used method that is useful in determining whether something is occurring above or below a fixed level. In Figure 8.1, for example,

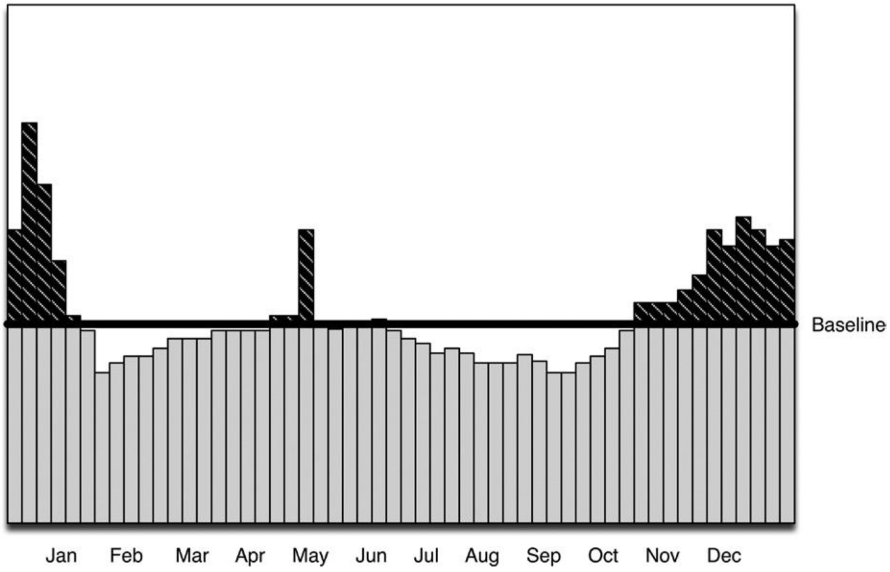


FIGURE 8.1

A Flat Average of All Events over One Year.

it can be clearly seen that production output is either above or below the average production level for the previous 12 months. The specific peaks and valleys could represent anything from a stalled process to normal variations in process schedules.

This may or may not be useful for operations management; in a security context, this type of baseline provides little value. Knowing that `59,421,102 events over 30 days=1,980,703 events per day average` cannot tell us if the current day's event volume of 2,000,000 is meaningful or not, without some additional context. Does the yearly average include weekends and other periods of downtime? If it does, the actual per day expected values of a workday could be considerably higher. For purposes of behavioral analysis, a more applicable method would be a similar calculation that excluded known periods of downtime and created a flat baseline that was more relevant to periods of operation. Better still are time-correlated baselines, where an observed period of activity is baselined against data samples taken over a series of similar time periods. That is, if looking at data for 1 week, the baseline might indicate the expected patterns of behavior over a period of several weeks. Figure 8.2 illustrates how this affects the flatline average with a curved baseline that visualizes a drop in activity during weekends and shows an expected peak on Thursdays.

Time-correlated baselines are very useful because they provide a statistical analysis of observed activity within relevant contexts of time—essentially providing historical context to baseline averages.[1] Without such a baseline, a spike in activity

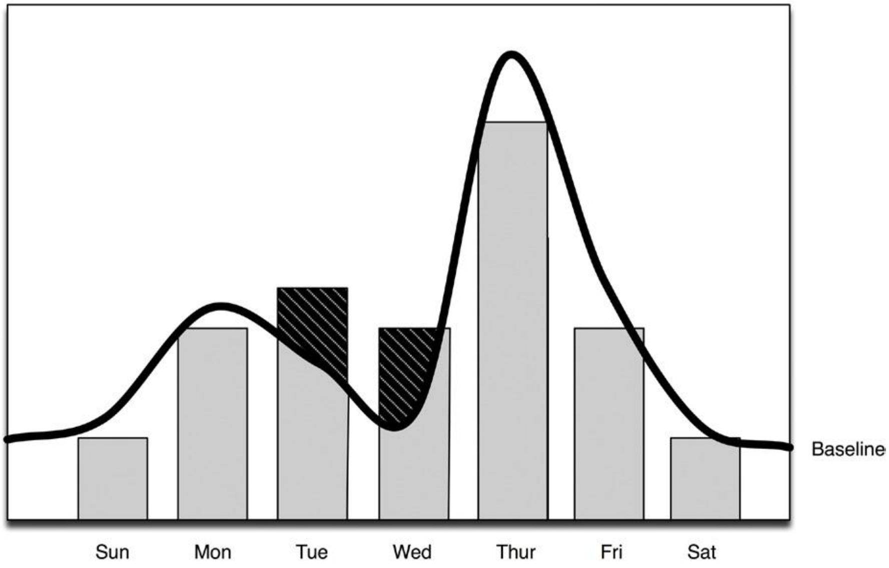


FIGURE 8.2

A Time-Correlated Baseline Shows Dip on Weekends, Peak on Thursdays.

on Thursday might be seen as an anomaly and spur an extensive security analysis, rather than being clearly indicated as normal behavior. Consider that there may be scheduled operations at the beginning of every month, at specific times of the day, or seasonally, all causing expected changes in event volumes.

Baselines, in whatever form, can be obtained in several ways, all beginning with the collection of relevant data over time, followed by statistical analysis of that data. Although statistical analysis of any metric can be performed manually, this function is often supported by the same product/system used to collect the metric, such as a Data Historian or an SIEM system (see Table 8.2 for examples).

Anomaly Detection

An anomaly is simply something that happens outside of normal parameters. Many firewalls and IDS/IPS devices may support anomaly detection directly, providing an additional detection capability at the enclave perimeter. Holistically, all behaviors can be assessed for more systematic anomalies indicative of larger threats. Luckily, having defined expected (baseline) behaviors anomalies can be easily identified. In addition, many automated systems—including NBAD, Log Management, and SIEM systems—are available to facilitate anomaly detection across a number of different sources.

Behavioral anomaly detection is useful because there is no dependency upon a detection signature, and therefore unknown threats or attacks can be identified. In

Table 8.2 Measurement and Analysis of Baseline Metrics

Behavior	Measured Metric(s)	Measured by	Analyzed by
Network Traffic	• Total unique Source IPs • Total unique Destination IPs • Total unique TCP/UPD ports • Traffic Volume (total flows) • Traffic Volume (total bytes) • Flow duration	• Network switch/router flow logs (i.e., netFlow, jFlow, sFlow, or similar) • Network probe (i.e., IDS/IPS, network monitor, etc.)	• Network Behavior Anomaly Detection (NBAD) system • **Log Management system** • SIEM system
User Activity	• Total unique active users • Total logons • Total logoffs • Logons by user • Logoffs by user • Activity (e.g., configuration changes) by user	• Application Logs • Database logs and/or transaction analysis • Application logs and/or session analysis • Centralized authentication (LDAP, Active Directory, IAM)	• Log Management system • SIEM system NOTE: user activity may need additional layers of correlation to consolidate multiple usernames/accounts associated with a single user
Process/Control Behavior	• Total unique function codes • Total number per individual function code • Total set point or other configuration changes	• Industrial Protocol Monitor • Application Monitor • Data Historian tags	• Data Historian • SIEM System
Event/Incident Activity	• Total events • Total events by criticality/severity • Total events by security device	• Security device (i.e., firewall, IPS) logs	• Application Monitor • Industrial Protocol Filter

addition, although often thought of exclusively in terms of network anomalies, any metric that is collected over time can be statistically analyzed and used for anomaly detection.

For example, an unexpected increase in network latency—measurable by easily obtained network metrics such as Transmission Control Protocol (TCP) errors, the size of the TCP receive window, the round-trip duration of a `ping` (TTL)—can indicate risk to the industrial network.[2] However, as can be seen in Table 8.3, anomalies can indicate normal, benign variations in behavior as well as potential threats.

Table 8.3 Examples of Suspicious Anomalies

Normal Behavior	Anomaly	Detected By	Indication
All Modbus communications to a group of PLCs originates from the same three HMI workstations	A fourth system communicates to the PLCs	A >20% increase in the number of unique source IP addresses, from analysis of: • Network flows • Security event logs from firewalls, IPS devices, etc. • Application logs • Etc.	• A new, unauthorized device has been plugged into the network (e.g., an administrator's laptop) • A rogue HMI is running using a spoofed IP address • A new system was installed and brought online
Every device has a single MAC address and a single IP address	An IP address is seen originating from two or more distinct MAC addresses	>1 MAC Adresses per IP, from analysis of: • Network flows • Security event logs from firewalls, IPS devices, etc. • Application logs • Etc.	• An attacker is spoofing an address • A device has failed and been replaced with new hardware
A process within a Control System enclave is running a consistent control loop for extended periods	Traffic increases above expected volumes	A >20% increase in the total network traffic, in bytes, from analysis of network flows	• An unauthorized service is running • A scan or *pen test* is being run • A shift change is underway • A new batch or process has started
	Traffic decreases below expected levels	A >20% decrease in the total network traffic, in bytes, from analysis of network flows	• A service has stopped running • A networked device has failed or is offline • A batch or process has completed

Changes to Programmable Logic	Industrial network monitor such as a SCADA IDS Ladder Logic/ Code Review	Any variation in the individual function codes and/or frequency of any function code, from analysis of • Industrial Protocol Monitors • Application Monitors • SCADA IDS/IPS logs	• A process has been altered • A new process has been implemented • An old process has been removed • A process has been sabotaged
Authorized Users log on to common systems at the beginning of a shift	• Unauthorized user logs on to a system normally accessed by administrators only • Authorized users log on to a system outside of normal shift hours • Authorized users log on to unknown of unexpected systems	Any variation seen from analysis of authentication logs from • Active Directory Operating System logs • Application Logs	• Personnel changes have been made • An administrator is on leave or absent and duties have been delegated to another user • A rogue user has authenticated to the system • An administrator account has been compromised and is in use by an attacker

In other words, the rate of false positives tends to be higher using anomaly detection techniques.

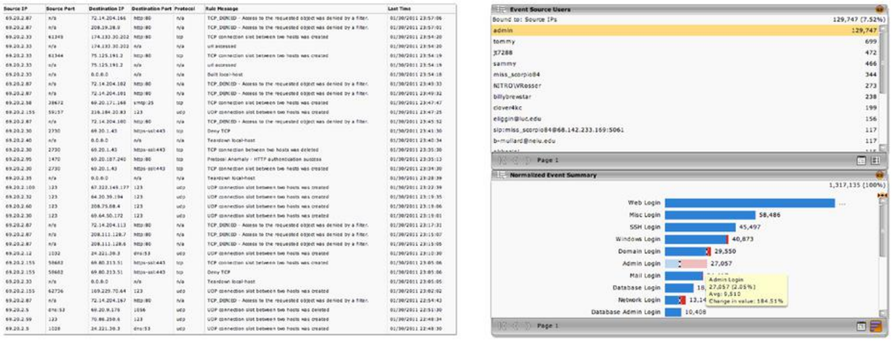
Analyzing IT vs. OT Metrics

Up to this point, the discussion of anomaly detection has focused largely on security events derived from information technology (IT) tools. Even when looking at specialized security products for industrial network monitoring, these devices operate on the same paradigm as IT security devices to detect and block suspicious and/or "out of policy" events, and then generate an alert.

Anomaly Detection Tools

Anomaly detection can be done using anything from "gut feelings," to manual statistical analysis using a spreadsheet or mathematical application, to specialized statistics software systems, to network and security data analysis systems such as certain Log Management and SIEM systems. Time-series databases, such as those used by Data Historians, can also be used for anomaly detection; while these systems do not typically represent anomalies within the specific context of network security, a Historian configured to show comparative overlays of security events over time could easily identify dangerous anomalies that might indicate a cyber attack.

NBAD, Log Management, and SIEM tools are predominantly used for security-related anomaly detection. NBAD systems are focused exclusively on network activity and may or may not support the specific industrial network protocols used within a Supervisory Control and Data Acquisition (SCADA) or Distributed Control Systems (DCS) environment. As such, the use of a Log Management or an SIEM system may be better suited for anomaly detection in industrial networks. For example, Figure 8.3 shows a visual representation of anomalous authentication behavior for the `admin` user (on the right) versus the same data shown without context (on the left); the security tool has done the necessary statistical analysis to



Source IP	Source Port	Destination IP	Destination Port	Protocol	Rule Message	Last Time
69.20.2.87	n/a	72.14.204.166	http:80	n/a	TCP_DENIED - Access to the requested object was denied by a filter.	01/30/2011 23:57:06
69.20.2.87	n/a	208.19.28.9	http:80	n/a	TCP_DENIED - Access to the requested object was denied by a filter.	01/30/2011 23:57:01
69.20.2.33	61349	174.133.30.202	http:80	tcp	TCP connection slot between two hosts was created	01/30/2011 23:54:20
69.20.2.33	n/a	174.133.30.202	n/a	n/a	url accessed	01/30/2011 23:54:20
69.20.2.33	61344	75.125.191.2	http:80	tcp	TCP connection slot between two hosts was created	01/30/2011 23:54:19
69.20.2.33	n/a	75.125.191.2	n/a	n/a	url accessed	01/30/2011 23:54:19
69.20.2.33	n/a	0.0.0.0	n/a	n/a	Built local-host	01/30/2011 23:54:18
69.20.2.87	n/a	72.14.204.102	http:80	n/a	TCP_DENIED - Access to the requested object was denied by a filter.	01/30/2011 23:49:33
69.20.2.87	n/a	72.14.204.101	http:80	n/a	TCP_DENIED - Access to the requested object was denied by a filter.	01/30/2011 23:49:32
69.20.2.58	28672	69.20.171.168	smtp:25	tcp	TCP connection slot between two hosts was created	01/30/2011 23:47:47
69.20.2.155	59157	216.184.20.83	123	udp	UDP connection slot between two hosts was created	01/30/2011 23:47:25
69.20.2.87	n/a	72.14.204.100	http:80	n/a	TCP_DENIED - Access to the requested object was denied by a filter.	01/30/2011 23:45:52
69.20.2.30	2730	69.20.1.43	https-ssl:443	tcp	Deny TCP	01/30/2011 23:41:30
69.20.2.40	n/a	0.0.0.0	n/a	n/a	Teardown local-host	01/30/2011 23:40:34
69.20.2.30	2730	69.20.1.43	https-ssl:443	tcp	TCP connection between two hosts was deleted	01/30/2011 23:35:30
69.20.2.95	1470	69.20.187.240	http:80	tcp	Protocol Anomaly - HTTP authentication success	01/30/2011 23:35:13
69.20.2.30	2730	69.20.1.43	https-ssl:443	tcp	TCP connection slot between two hosts was created	01/30/2011 23:34:30
69.20.2.35	n/a	0.0.0.0	n/a	n/a	Teardown local-host	01/30/2011 23:28:39
69.20.2.100	123	67.222.149.177	123	udp	UDP connection slot between two hosts was created	01/30/2011 23:22:39
69.20.2.32	123	64.30.39.194	123	udp	UDP connection slot between two hosts was created	01/30/2011 23:19:35
69.20.2.60	123	208.75.88.4	123	udp	UDP connection slot between two hosts was created	01/30/2011 23:19:06
69.20.2.30	123	69.64.50.172	123	udp	UDP connection slot between two hosts was created	01/30/2011 23:19:01
69.20.2.87	n/a	72.14.204.113	http:80	n/a	TCP_DENIED - Access to the requested object was denied by a filter.	01/30/2011 23:17:31
69.20.2.87	n/a	208.111.128.7	http:80	n/a	TCP_DENIED - Access to the requested object was denied by a filter.	01/30/2011 23:15:07
69.20.2.87	n/a	208.111.128.6	http:80	n/a	TCP_DENIED - Access to the requested object was denied by a filter.	01/30/2011 23:15:05
69.20.2.12	1032	24.221.30.3	dns:53	udp	UDP connection slot between two hosts was created	01/30/2011 23:10:30
69.20.2.155	58682	69.80.213.51	https-ssl:443	tcp	TCP connection slot between two hosts was created	01/30/2011 23:05:06
69.20.2.155	58682	69.80.213.51	https-ssl:443	tcp	Deny TCP	01/30/2011 23:05:06
69.20.2.33	n/a	0.0.0.0	n/a	n/a	Teardown local-host	01/30/2011 23:05:05
69.20.2.155	62726	169.229.70.64	123	udp	UDP connection slot between two hosts was created	01/30/2011 23:02:02
69.20.2.87	n/a	72.14.204.167	http:80	n/a	TCP_DENIED - Access to the requested object was denied by a filter.	01/30/2011 22:54:43
69.20.2.5	dns:53	69.20.9.176	1056	udp	UDP connection slot between two hosts was deleted	01/30/2011 22:51:30
69.20.2.59	123	70.86.250.6	123	udp	UDP connection slot between two hosts was created	01/30/2011 22:48:34
69.20.2.5	1028	24.221.30.3	dns:53	udp	UDP connection slot between two hosts was created	01/30/2011 22:48:30

Bound to: Source IPs	129,747 (7.52%)
admin	129,747
tommy	699
jt7288	472
sammy	466
miss_scorpio84	344
NITRO\WRosser	273
billybrewstar	238
clover4kc	199
eliggin@luc.edu	156
sip:miss_scorpio84@68.142.233.169:5061	117
b-mullard@neiu.edu	117

Admin Login
27,057 (2.05%)
Avg: 9,510
Change in value: 184.51%

FIGURE 8.3

Representation of Anomalous Administrator Logins Using an SIEM System.

Image courtesy of NitroSecurity.

show a 184% increase in administrator logins and has also brought that anomaly to the attention of the security analyst.

As shown in Table 8.3, this requires that the Log Management or SIEM system is used to collect relevant data over time from those systems used in perimeter and interior enclave security, as well as any relevant network traffic data obtained from network switches and routers.

TIP

When selecting an analysis tool for industrial network anomaly detection, consider the greatest relevant time frame for analysis and ensure that the system is capable of automating anomaly detection over sufficient periods of time. Many systems, such as Log Management and SIEM systems, are not designed exclusively for anomaly detection and may have limitations as to how much information can be assessed and/or for how long.

To ensure the tool is right for the job, look at the operational lifespan of specific processes and use time-correlated baselines to determine normal activities for those processes. If a process takes 3 hours, analysis of $n \times 3$ hours of process data is needed for anomaly detection, where n represents the number of sampled operations. The greater the n, the more accurate the baseline, and therefore the more accurate the anomaly detection.

BEHAVIORAL WHITELISTING

Whitelisting is well understood in the context of access control and application whitelisting (AWL) for host malware prevention. However, the concept of whitelisting has many roles within control system environments, where access, communication, processes, policies, and operations are well defined. Using the controlled nature of these systems and the enclave-based policies defined in Chapter 7, “Establishing Secure Enclaves,” whitelists can be defined for a variety of network and security metrics, including users, assets, applications, and others.

Whitelists can be actively enforced via a `Deny !Whitelist` policy on a firewall or IPS, or can be used throughout a network by combining network-wide monitoring and exception reporting with dynamic security controls. For example, if an exception is seen to a policy within an enclave, a script can be run to tighten the specific perimeter defenses of that enclave.

User Whitelists

Understanding user activity—especially of administrative users—is useful for detecting cyber attacks, both by insiders (e.g., a disgruntled employee) as well as by outside attackers. Locking critical functions to administrative personnel, and then following best practices of user authentication and access control, means that an attack against a critical system should have to originate from an administrative user account. In reality, enumeration is a standard process in a cyber attack because administrative accounts can be used for malicious intent (see Chapter 6, "Vulnerability and Risk Assessment"). They can be hijacked or used to escalate other rogue accounts in order to enable nonauthorized users' administrator rights.

Fortunately, authorized users have been identified and documented (see Chapter 7, "Establishing Secure Enclaves"), and this allows us to whitelist user activities. As with any whitelist, the list of known users needs to be established and then compared to monitored activity. In this case, authorized users can be identified using a directory service or an Identity and Authentication Management (IAM) system, such as Lightweight Directory Access Protocol (LDAP), Microsoft Active Directory, or other commercial IAM systems from IBM, Oracle, Sun, and others.

As with exception reporting, the whitelist is first defined and then monitored activity is compared against it. If there is an exception, it becomes a clear indicator that something outside of established policies is occurring. In the case of a user whitelist, all known good user accounts are used as a detection filter against all login activity. If the user is on the list, nothing happens. If the user is not on the list, it is assumed bad and an alert is sent to security personnel. This accomplishes an immediate flag of all rogue accounts, default accounts, or other violations of the authentication policies. Consider Stuxnet, which uses a default (albeit hidden) set of authentication credentials to access a Programmable Logic Controller (PLC). Assuming that the operator was unaware of this account and had not explicitly defined it as a "known good" or whitelisted account, it could have triggered an exception, alerting cyber security officers of its existence.

NOTE

In the case of hidden accounts and hard-coded backdoor authentications, as used in Stuxnet, normal connections would also be flagged as an exception, because those accounts would most likely not appear on the whitelist. This could generate a potential excess of false-positive alerts. However, it would also draw attention to the existence of default authentications within the system so that these accounts could be more closely monitored. For example, the WinCC authentication could be monitored in conjunction with baseline analysis. If the default account was then used by malware, it would still be possible to detect the threat via anomaly detection.

Asset Whitelists

Once an inventory of cyber assets is completed—either automatically via an appropriate soft network scan (see Chapter 6, "Vulnerability and Risk Assessment") or manual inventory—the resulting list of known, authorized devices can be used to whitelist known good network devices.

Again, unlike perimeter-based security policies which may only allow known good devices into an enclave, a network asset whitelist can be applied to devices within an enclave. If a spoofed address or rogue device appears within an enclave, it can still be detected via exception reporting against the list of known good devices so that action can be taken.

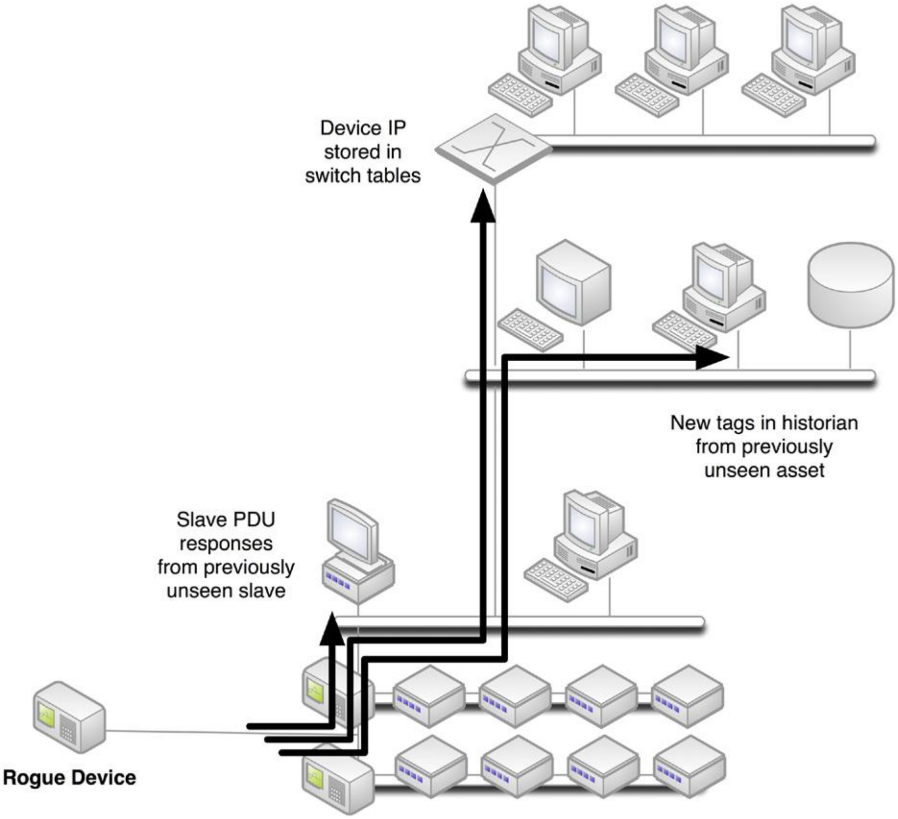


FIGURE 8.4

Information Flow Relevant to a Rogue Device IP.

A classic use case for asset whitelisting is the use of mobile media, which can be carried past perimeter defenses and attached directly to a protected network, well within a secure enclave. This could be benign—an employee bringing an iPhone inside a control system that has WiFi enabled—or it could be a deliberate vehicle for sabotage. Either way, the IP address of the device will be detected by switches, routers, network monitors, and security devices, and will eventually be seen in logs or events that are centralized and managed, as illustrated in Figure 8.4. At this point, simple comparison against the defined whitelist will identify the presence of a non-authorized device. This example represents significant risk, as the mobile device also connects directly to a 3G or 4G cellular network, which bypasses all defensive measures of the ESP, and opens the enclave up for attack or exploitation.

The whitelists themselves would need to be generated and applied to the central management system—most likely a Log Management or an SIEM system that is capable of looking at device metrics across the entire network. Depending upon the specific monitoring product used, the whitelist might be built through the use of a defined system variable (much like the generation of enclave-specific variables in firewalls and IDS/IPS devices, as discussed in Chapter 7, "Establishing Secure Enclaves"), configurable data dictionaries, manually scripted detection signatures, etc.

Application Behavior Whitelists

Applications themselves can be whitelisted per host using an AWL product. However, application behavior can also be whitelisted within the network. As with asset whitelisting, application behavior whitelists need to be defined so that good behavior can be differentiated from bad behavior. Like asset whitelists, application behavior whitelists can be utilized by a central monitoring and management system by defining a variable of some sort within a Log Management or an SIEM system. However, because of the nature of industrial network protocols, many application behaviors can be determined directly by monitoring those protocols and decoding them in order to determine the underlying function codes and commands being executed (see Chapter 4, "Industrial Network Protocols"). This allows for in-line whitelisting of industrial application behavior in addition to network-wide whitelisting offered by a Log Management or SIEM system. If in-line whitelisting is used, via an industrial security appliance or application monitor, network whitelisting may still be beneficial for assessing application behavior outside of industrial control systems (i.e., for enterprise applications and SCADA applications that do not utilize industrial protocols).

Some examples of application behavior whitelisting in industrial networks are as follows:

- Only read-only function codes are allowed.
- Master PDUs or Datagrams are only allowed from predefined assets.
- Only specifically defined function codes are allowed.

Some examples of application behavior whitelisting in enterprise networks are as follows:

- Only encoded HTTP web traffic is allowed and only on Port 443.
- Only `POST` commands are allowed for web form submissions.
- Human–Machine Interface (HMI) applications are only allowed on predefined hosts.

Some examples of application behavior whitelisting across both environments together are as follows:

- Write commands are only allowed in native fieldbus protocols and not over TCP/IP.
- HMI applications in supervisor networks are only allowed to use `read` functions over TCP/IP-based protocols.

In other words, unlike AWL systems, which only allow certain authorized applications to execute, application behavior whitelisting only allows applications that do execute to function in specifically defined ways on the network.

For example, an AWL system is installed on a Windows-based HMI. The AWL allows for the HMI application to execute, as well as a minimal set of necessary operating system services, and the networking services required to open Modbus network sockets so that the HMI can communicate to a series of RTUs and PLCs. However, the AWL does not control how the HMI application is used, and what commands and controls it can enforce on those RTUs and PLCs. The HMI, although protected by AWL, can be used by a disgruntled employee to shut down key systems, randomly change set points, or otherwise disrupt operations. Network-based application behavior whitelisting looks at how the HMI application is being used and compares that to a defined whitelist of authorized commands—in this case, a list of known good Modbus function codes. Functions that are not explicitly defined may then be actively blocked or they may be allowed but the system may generate an alert to notify administrators of the violated policy.

Industrial protocol or application monitoring tools should possess a base understanding of industrial protocols and their functions, allowing behavioral whitelists to be generated directly within the device. For network-wide behavioral whitelisting, variables or data dictionaries need to be defined. Common variables useful in application behavioral whitelisting include these same application function codes—the specific commands used by industrial protocols, ideally organized into clear categories (read, write, system commands, synchronization, etc.).

Examples of Beneficial Whitelists

Many whitelists can be derived using the functional groups defined in Chapter 7, “Establishing Secure Enclaves.” Table 8.4 identifies some common whitelists, and how those whitelists can be implemented and enforced.

Smart-Lists

The term “Smart-Lists” was first introduced at the SANS Institute’s 2010 European SCADA and Process Control Summit in London, United Kingdom. “**Smart-Listing**” combines the concept of behavioral whitelisting with a degree of deductive intelligence. Where blacklists block what is known to be bad, and whitelists only allow what is known to be good, Smart-Lists use the latter to help dynamically define the former.

For example, if a critical asset is using AWL to prevent malicious code execution, the AWL software will generate an alert when a nonauthorized application attempts to execute. What can now be determined is that the application is not a known good application for that particular asset. However, it could be a valid application that is in use elsewhere, and has attempted to access this asset unintentionally. A quick correlation against other whitelists can then determine if the application under scrutiny is an acceptable application on other known assets. If it is, the “Smart-Listing” process might result in an informational alert and nothing more. However, if the application under scrutiny is not defined anywhere within the

Table 8.4 Examples of Behavioral Whitelists

Whitelist	Built Using	Enforced Using	Indications of a Violation
Authorized devices by IP	• Network monitor or probe (such as a Network IDS) • Network scan	• Firewall • Network Monitor • Network IDS/IPS	A rogue device is in use
Authorized applications by port	• Vulnerability assessment results • Port scan	• Firewall • Network IDS/IPS • Application Flow Monitor	A rogue application is in use
Authorized applications by content		• Application Monitor	An application is being used outside of policy
Authorized Function Codes/Commands	• Industrial network monitor such as a SCADA IDS • Ladder Logic/ Code Review	• Application Monitor • Industrial Protocol Monitor	A process is being manipulated outside of policy
Authorized Users	• Directory Services • IAM	• Access Control • Application Log Analysis • Application Monitoring	A rogue account is in use

system as a known good application, the Smart-Listing process can deduce that it is malicious in nature, and define it within the system as a known bad application and proactively defend against it, by initiating a script or other active remediation mechanism to block that application wherever it might be detected.

"Smart-Listing" therefore combines what we know from established whitelists with deductive logic in order to dynamically adapt our blacklist security mechanisms (such as firewalls and IPS devices) to block newly occurring threats. This process is illustrated in Figure 8.5. First, an alert is generated that identifies a violation of an established policy; second, the nature of that alert is checked against other system-wide behavior; and finally a decision is made—if it is "bad" a script or other automation service may be used to dynamically update firewall, IDS/IPS, and other defenses so that they can actively block this activity. If not, the activity might generate an alert, or be ignored.

Smart-Listing is a relatively new concept which could greatly benefit enclave defenses by allowing them to automatically adapt to evasive attacks as well as insider attacks. Smart-Listing is especially compelling when used with overarching security management tools (see Chapter 9, "Monitoring Enclaves") as it requires complex event association and correlation. Although it has yet to be determined how widely this technique will be adopted by security analysis and information

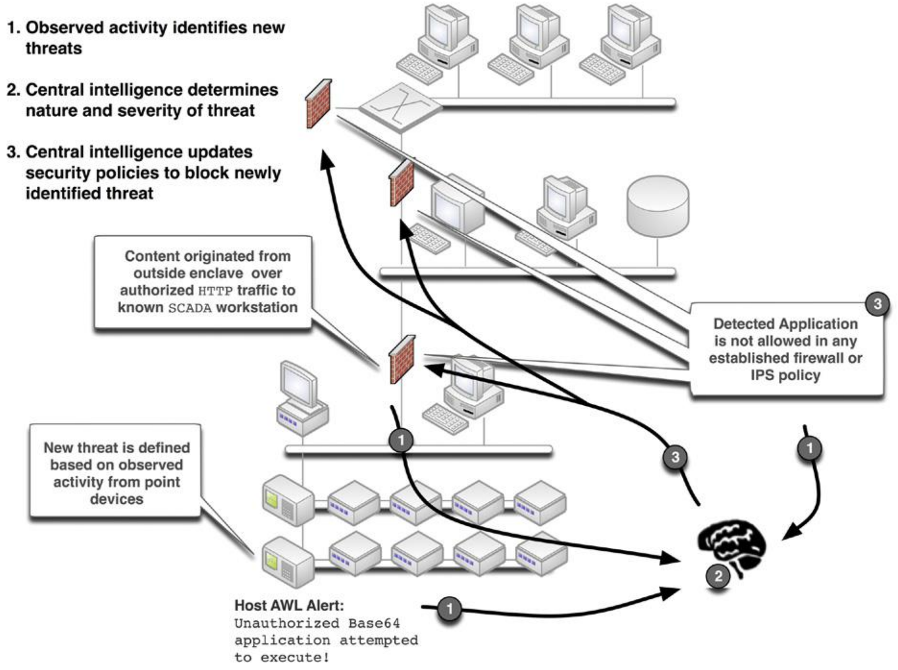

1. Observed activity identifies new threats
2. Central intelligence determines nature and severity of threat
3. Central intelligence updates security policies to block newly identified threat
Content originated from outside enclave over authorized HTTP traffic to known SCADA workstation
Detected Application is not allowed in any established firewall or IPS policy
New threat is defined based on observed activity from point devices
Host AWL Alert:
Unauthorized Base64 application attempted to execute!
1
1
1
2
3
3

FIGURE 8.5

Smart-Listing.

management vendors, at present the techniques can be performed manually, using any number of Log Management or SIEM tools.

THREAT DETECTION

Used independently, the specific detection techniques discussed up to this point—security device and application logs, network connections, specific alerts generated by exception reporting or anomaly detection, and violations of whitelists—provide valuable data points indicating events where a specific policy was violated. Even simple attacks, however, consist of multiple steps. For the detection of an incident (vs. a discrete event), it is, therefore, necessary to look at multiple events together and search for larger patterns. For example, many attacks will begin with a scanning technique, followed by an enumeration technique, followed by an attempt to successfully authenticate against an enumerated account. This pattern might equate to firewall alerts indicating a ping sweep, followed by access to an `/etc/passwd`, followed by a brute force login. The detection of this larger threat pattern is known as event correlation. As cyber attacks continue to increase in sophistication, event correlation methods have continued to expand, considering event data from a wider net of point security devices, additional event contexts such as user privileges or asset vulnerabilities, and searching for more complex patterns.

With Stuxnet, however, another factor was introduced that further complicated the event correlation process. Prior to Stuxnet, a threat had never before involved events from both IT and Operational Technology (OT) systems. With the evolution of threat patterns across both systems, the correlation of events across both IT and OT systems is also necessary. However, event correlation systems were not designed to accommodate OT systems, presenting challenges in the detection of the most serious threats to industrial networks.

Event Correlation

Event correlation simplifies the threat detection process by making sense of the massive amounts of discrete event data, analyzing it as a whole to find the important patterns and incidents that require immediate attention. Although early event correlation focused on the reduction of event volumes in order to simplify event management—often through filtering, compressing, or generalizing events[3]—newer techniques involve state logic to analyze event streams as they occur, performing pattern recognition to find indications of network issues, failures, attacks, intrusions, etc.[4] Event correlation is useful in several ways, including facilitating human security assessments by making the large volumes of event data from a wide variety of sources more suitable for human consumption and comprehension, by automatically detecting clear indications of known threat patterns to easily detect incidents of cyber attack and sabotage and by facilitating the human detection of unknown threat patterns through event normalization. The process of event correlation is depicted in Figure 8.6.

First, events are compared against a defined set of known threat patterns or "correlation rules." If there is a match, an entry is made in a (typically) memory-resident state tree; if another sequence in the pattern is seen, the rule progresses until a complete match is determined. For example, if a log matches the first condition of a rule, a new entry is made in the state tree, indicating that the first condition of a rule has been met. As more logs are assessed, there may be a match for a subsequent condition of an existing branch, at which point that branch is extended. A log may meet more than one condition of more than one rule, creating large and complex state trees. For example, even a simple "brute force attack" rule can create several unique branches. Consider the rule

```
If [5 consecutive failed logins] from [the same source IP] to [the
same destination IP] within [5 minutes]
```

This example would create one branch for the first failed login event "A" from any IP address to any other IP address. The next matching login event "B" would extend that initial branch while also generating a new branch (with a new timer):

```
A + B
B
```

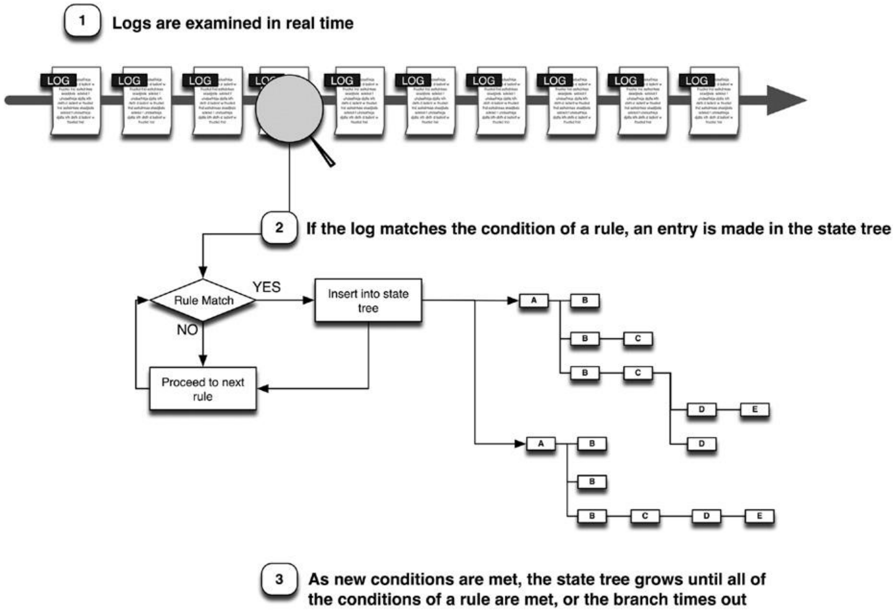

1 Logs are examined in real time
LOG
2 If the log matches the condition of a rule, an entry is made in the state tree
Rule Match
YES
NO
Insert into state tree
Proceed to next rule
A
B
C
D
E
3 As new conditions are met, the state tree grows until all of the conditions of a rule are met, or the branch times out

FIGURE 8.6

The Event Correlation Process.

The third matching login event "C" would extend the first two branches while also creating a third:

```
A + B + C
B + C
C
```

This will continue ad inifitum until all of the conditions are met, or until a branch's timer expires. If a branch completes (i.e., all conditions are met), the rule triggers.

Note that events are collected from many types of information sources, such as firewalls, switches, authentication servers, etc. Therefore, before they can be effectively correlated they must be normalized into a common event taxonomy. Normalization categorizes activities into a common framework so that similar events can be correlated together even if the originating log or event formats differ.[5] Without normalization, many additional correlation rules would be required in order to check a condition (in this example a failed login) against all possible variations of that event that may be present (Windows logins, Linux logins, CMS application logins, etc.).

For purposes of threat detection, the entire event correlation process is typically performed in memory at the time individual logs and events are collected. However, correlation can also be performed manually by querying larger stores of already collected events to find similar patterns.[6]

Table 8.5 Example Event Correlation Rules

Threat Pattern	Description	Rule
Brute Force Attack	Passwords are guessed randomly in quick succession in order to crack the password of a known user account	A number `N` of `Failed Logon` events, followed by one or more `Successful Logon` events, from the same `Source IP`
Outbound Spambot behavior	A spambot (malware designed to send spam from the infected computer) is sending bulk unsolicited e-mails to outside addresses	A large number `N` of `Outbound SMTP` events, from one internal `IP Address`, each destined to a unique `e-mail address`
HTTP Command and Control	A hidden communication channel inside of HTTP is used as a command and control channel for malware	`HTTP` traffic is originating from servers that are not `HTTP` servers
Covert botnet, command, and control	A distributed network of malware establishing covert communications channels over applications that are otherwise allowed by firewall or IPS policy	Traffic originating from `N` number of `$ControlSystem_Enclave01_Devices` to `!$ControlSystem_Enclave01_Devices` with contents containing `Base64` coding.

Examples of event correlation rules are provided in Table 8.5. Event correlation may be very basic (e.g., a brute force attack) or highly complex—up to and including tiered correlation that consists of correlation rules within correlation rules (e.g., a brute force attack followed by a malware event).

Data Enrichment

Data enrichment refers to the process of appending or otherwise enhancing collected data with relevant context obtained from additional sources. For example, if a username is found within an application log, that username can be referenced against a central IAM system to obtain the user's actual name, departmental roles, privileges, etc. This additional information "enriches" the original log with this context. Similarly, an IP address can be used to enrich a log file, referencing IP reputation servers to see if there is known threat activity associated with that IP address, or by referencing geolocation services to determine the physical location of the IP address by country, state, or postal code (see "Additional Context" in Chapter 9, "Monitoring Enclaves," for more examples of contextual information).

Data enrichment can occur in two primary ways: the first is by performing a lookup at the time of collection and appending the contextual information into the log; the second is to perform a lookup at the time the event is scrutinized by the SIEM or Log Management system. Although both provide the relevant context, each has advantages and disadvantages. Appending the data at the time of collection provides the most accurate representation of context and prevents

Table 8.6 Common Logon Events Depicted by Varying Log Formats[a]

Log Source	Log Contents	Description
Juniper Firewall	`<18> Dec 17 15:45:57 10.14.93.7 ns5xp: NetScreen device_id 5 ns5xp system-warning-00515: Admin User jdoe has logged on via Telnet from 10.14.98.55:39073 (2002-12-17 15:50:53)`	Successful Logon
Cisco Router	`<57> Dec 25 00:04:32:%SEC_LOGIN-5-LOGIN_SUCCESS:Login Success [user:jdoe] [Source:10.4.2.11] [localport:23] at 20:55:40 UTC Fri Feb 28 2006`	Successful Logon
Redhat Linux	`<122> Mar 4 09:23:15 localhost sshd[27577]: Accepted password for jdoe from ::ffff:192.168.138.35 port 2895 ssh2`	Successful Logon
Windows	`<13> Fri Mar 17 14:29:38 2006 680 Security SYSTEM User Failure Audit ENTERPRISE Account Logon Logon attempt by: MICROSOFT_AUTHENTICATION_PACKAGE_V1_0 Logon account: JDOE Source Workstation: ENTERPRISE Error Code: 0xC000006A 4574`	Successful Logon

[a] *A. Chuvakin, Content aware* SIEM. *http://www.sans.org/security-resources/idfaq/vlan.php, February, 2000 (cited: January 19, 2011).*

misrepresentations that may occur as the network environment changes. For example, if IP addresses are provided via the Dynamic Host Configuration Protocol (DHCP), the IP associated with a specific log could be different at the time of collection than at the time of analysis. However, although more accurate, this type of enrichment also burdens the analysis platform by increasing the amount of stored information. Also, it is important to ensure that the original log file is maintained for compliance purposes, requiring the system to replicate the original raw log records prior to enrichment. The alternative, providing the context at the time of analysis, removes these additional requirements at the cost of accuracy. Although there is no hard rule indicating how a particular product enriches the data that it collects, traditional Log Management platforms tend toward analytical enrichment, whereas SIEM platforms tend toward enrichment at the time of collection, possibly because most SIEM platforms already replicate log data for parsing and analysis, minimizing the additional burden associated with this type of enrichment.

Normalization

Event normalization is a classification system, which categorizes events according to a defined taxonomy, such as the Common Event Expression Framework provided by the MITRE Corporation.[7] Normalization is a necessary step in the correlation process, due to the lack of a common log format.[8] Consider a logon activity. Table 8.6 provides a comparison of authentication logs from a variety of sources.

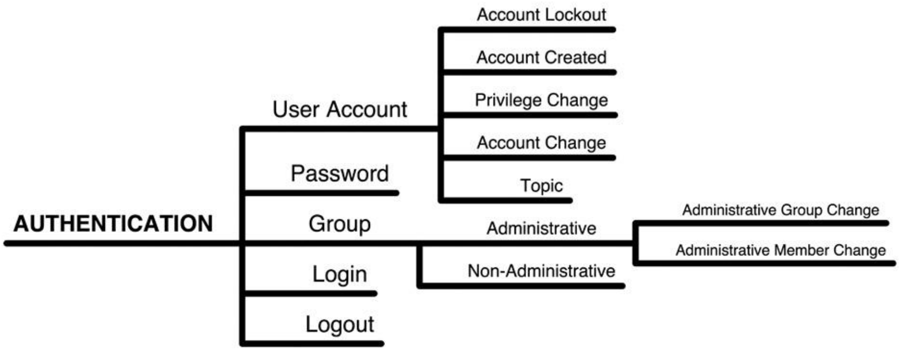


FIGURE 8.7

A Partial Representation of a Tiered Normalization Taxonomy.

Although each example in Table 8.6 is a logon, the way the message is depicted varies sufficiently that without a compensating measure such as event normalization, a correlation rule looking for "logons" would need to explicitly define each known logon format. In contrast, event normalization provides the necessary categorization so that a rule can reference a "logon" and then successfully match against any variety of logons. Because this level of generalization may be too broad for the detection of specific threat patterns, most normalization taxonomies utilize a tiered categorization structure, as illustrated in Figure 8.7.

Cross-source Correlation

Cross-source correlation refers to the ability to extend correlation across multiple sources so that common events from disparate systems (such as a firewall and an IPS) may be normalized and correlated together. As correlation systems continue to mature, the availability of single-source correlation is dwindling. Cross-source correlation remains an important consideration of threat detection capability. The more types of information that can be correlated, the more effective the threat detection will be, and the fewer false positives, as shown in Table 8.7.

As more systems are monitored (see Chapter 9, "Monitoring Enclaves"), the potential for expanding cross-source correlation increases accordingly—ideally with all monitored information being normalized and correlated together.

Tiered Correlation

Tiered correlation is simply the use of one correlation rule within another correlation rule. For example, a brute force attempt on its own may be indicative of a cyber incident, or it may not. If it is a cyber attack, there is no further determination of what the attack is, nor its intent. By stacking correlation rules within other rules, additional rules can be enabled to target more specific attack scenarios, as shown in Table 8.8.

Table 8.7 Single-Source vs. Cross-source Correlation

Single-source Correlation Example	Cross-source Correlation Example
Multiple `Failed Logon` followed by one or more `Successful Logon`	Multiple `Failed Logon` events by an `Admin user` of `Critical Assets`, followed by one or more `Successful Logon`
Any `Successful Logon` to a `Critical Asset`	Any `Successful Logon` to a `Critical Asset`, by either a `Terminated Employee` or by an `Admin User` at a time outside of `Normal shift hours.`
`HTTP` traffic is originating from servers that are not `HTTP` servers	`HTTP` traffic is originating from servers that are not `HTTP` servers' `IP addresses` with a geographic location outside of the United States

Table 8.8 Tiered Correlation Examples

Description	Rule
Brute Force Attack	A number `N` of `Failed Logon` events, followed by one or more `Successful Logon` events, from the same `Source IP`
Brute Force Malware Injection	A number `N` of `Failed Logon` events, followed by one or more `Successful Logon` events, from the same `Source IP`, followed by a `Malware Event`
Brute Force followed by Internal Propagation	A number `N` of `Failed Logon` events, followed by one or more `Successful Logon` events, from the same `Source IP`, followed by a `Network Scan` originating from the same `Source IP`
Internal Brute Force Enumeration using Known Password	A number `N` of `Failed Logon` events from the same `Source IP`, each with a unique `username` but a different `password`

The third example in Table 8.8 illustrates the use of normalization within correlation by using a `Malware Event` as a general condition of the rule. The fourth example illustrates the value of content inspection for the purposes of threat detection by exposing application authentication parameters to the correlation engine.

Correlating between IT and OT Systems

Up until now, correlation has been discussed solely within the context of IT networks running standard enterprise systems and protocols. Operational systems must also be analyzed, however, requiring that metrics within the OT network be correlated to events in the IT network. The challenge here is the disparity of the two systems, and the information collection models used within each—IT systems

Table 8.9 Correlation of IT and OT Systems[a]

Incident	IT Event	OT Event	Condition
Network instability	Increased Latency, measured by TCP errors, reduction of TCP receive windows, increased round-trip TTL, etc.	Reduction in Efficiency, measured by historical batch comparisons	Manifestation of network condition in operational processes Deliberate cyber sabotage
Operational change	No detected event	Change to operational set points, or other process change(s)	Benign process adjustment Undetected cyber sabotage
Network breach	Detected threat or incident using event correlation, to determine successful penetration of IT system(s)	Change to operational set points, or other process change(s)	Benign process adjustment Undetected cyber sabotage
Targeted Incident	Detected threat or incident directly targeting industrial SCADA or DCS systems connected to IT networks	Abnormal change to operational set points, unexpected PLC code writes, etc.	Potential "Stuxnet-class" cyber incident or sabotage

[a] *B. Singer, Correlating Risk Events and Process Trends. Proceedings of the SCADA Security Scientific Symposium (S4). Kenexis Security Corporation and Digital Bond Press., 2010.*

are monitored heavily for performance and security using a wide range of available tools, whereas OT systems are monitored primarily for process efficiency and performance, using a more limited range of tools consisting of Data Historians, spreadsheets, and statistical modeling applications (see Chapter 9, "Monitoring Enclaves").

However, even benign network behaviors of the IT network can impact operations, and threats do exist across both IT and OT systems. By correlating IT conditions against OT conditions, a good deal can be determined about potential cyber incidents.[9] For example, Table 8.9 shows several instances where IT systems can impact OT systems.

To fully leverage the automated correlation capability built into most IT SIEM products, OT data must first be collected into the SIEM, and then the normalization of one metric to another must be made using a common threat taxonomy.

SUMMARY

With enclave security measures in place, a larger picture of security-related activity begins to form. By measuring these activities and analyzing them, exceptions from the established security policies can be detected. In addition, anomalous activities can be identified so that they may be further investigated.

This requires well-defined policies and also requires that those policies are configured within an appropriate information analysis tool. Just as with perimeter defenses to the enclave, carefully built variables defining allowed assets, users, applications, and behaviors can be used to aid in detection of security risks and threats. If these lists can be determined dynamically, in response to observed activity within the network, the "whitelisting" of known good policies becomes "Smart-Listing"—which can help strengthen perimeter defenses through dynamic firewall configuration or IPS rule creation.

As various threat detection techniques are used together, the event information can be further analyzed by event correlation systems to find larger patterns that are more indicative of serious threats or incidents. Widely used in IT network security, event correlation is beginning to "cross the divide" into OT networks, at the heels of Stuxnet and other sophisticated threats that attempt to compromise industrial network systems via attached IT networks and services.

Everything—measured metrics, baseline analysis, and whitelists—all rely on a rich base of relevant security information. Where does this security information come from? Chapter 9, "Monitoring Enclaves," discusses what to monitor, and how, in order to obtain the necessary base of data required to achieve "situational awareness" and effectively secure an industrial network.

ENDNOTES

1. F. Salo, Anomaly Detection Systems: Context Sensitive Analytics. NitroSecurity, Inc. Portsmouth, NH, December 2009.
2. B. Singer, Correlating Risk Events and Process Trends. Proceedings of the SCADA Security Scientific Symposium (S4). Kenexis Security Corporation and Digital Bond Press, Sunrise, FL, 2010.
3. R. Kay, QuickStudy: event correlation. Computerworld.com <http://www.computerworld.com/s/article/83396/Event_Correlation?taxonomyId=016>, July 28, 2003 (cited: February 13, 2011).
4. Softpanorama, Event correlation technologies. <http://www.softpanorama.org/Admin/Event_correlation/>, January 10, 2002 (cited: February 13, 2011).
5. The MITRE Corporation, About CEE (common event expression). <http://cee.mitre.org/about.html>, May 27, 2010 (cited: February 13, 2011).
6. M. Leland, Zero-day correlation: building a taxonomy. NitroSecurity, Inc. <http://www.youtube.com/watch?v=Xtd0aXeLn1Y>, May 6, 2009 (cited: February 13, 2011).
7. The MITRE Corporation, About CEE (common event expression). <http://cee.mitre.org/about.html>, May 27, 2010 (cited: February 13, 2011).

8. A. Chuvakin, Content aware SIEM. <http://www.sans.org/security-resources/idfaq/vlan.php>, February 2000 (cited: January 19, 2011).
9. B. Singer, Correlating risk events and process trends. Proceedings of the SCADA Security Scientific Symposium (S4). Kenexis Security Corporation and Digital Bond Press, 2010, Sunrise, FL.

CHAPTER

Monitoring Enclaves

9

INFORMATION IN THIS CHAPTER:

- Determining What to Monitor
- Successfully Monitoring Enclaves
- Information Management
- Log Storage and Retention

The first step of information analysis requires a certain degree of information collection, so that there is a healthy body of data to assess. Collecting information relevant to cyber security requires knowing what to monitor and how to monitor it.

Unfortunately, there is a lot of information that could be relevant to cyber security, and because there are many unknown threats and exploitations, even information that may not seem relevant today may be relevant tomorrow as new threats are discovered. Even more unfortunate is that the amount of seemingly relevant data is already overwhelming—sometimes consisting of millions or even billions of events in a single day, with even higher rates of events occurring during a period of actual cyber attack.[1] It is therefore necessary to assess which events, assets, applications, users, and behaviors should be monitored—as well as any additional relevant systems that can be used to add context to the information collected therefrom, such as threat databases, user information, vulnerability assessment results, etc.

An additional challenge arises from the segregated nature of a properly secured industrial network: deploying a single monitoring and information management system across multiple otherwise-separated enclaves violates those enclaves and introduces potential risk. The methods used to monitor established enclaves must be considerate of the separation of those enclaves, and the data generated from this monitoring need to be managed accordingly as well. While there are benefits to fully centralized information management, the information being generated may be sensitive and may require "need to know" exposure to security analysts. Therefore, centralized monitoring and management needs to be overlaid with role-based information access, and some enclaves may require full separation—forgoing the efficiencies of central management so that the analysis, information management and reporting of sensitive information can be kept local in order to maintain absolute separation of duties between, for example, a highly critical safety system and a less secure supervisory system.

In order to deal with massive volumes of log and event data that can result from monitoring established network enclaves, and the challenges of highly distributed and segregated enclaves, best practices in information management—including short- and long-term information storage—must be followed. This is necessary both in order to facilitate the threat detection process, and also as a mandate for relevant compliance requirements, such as North American Electric Reliability Corporation Critical Infrastructure Protection (NERC CIP), NRC Title 10 CFR 73.54, Chemical Facility Anti-Terrorism Standards (CFATS), and others (see Chapter 10, “Standards and Regulations”).

DETERMINING WHAT TO MONITOR

The trite answer to “what to monitor” is “everything.” However, everything that we monitor results in information that must be managed. Every data point results in a log record, or perhaps a security or safety alert. Assets, users, applications and the networks that interconnect them all require monitoring. Because there are so many assets, users, applications, and networks that need to be monitored, the total amount of information generated every second in even a moderately sized enterprise can be staggering.[2] While products exist to automate security event and information management, the total amount of information that is available for analysis can quickly overwhelm the information analysis and storage capacity of these tools. Therefore, security monitoring requires some planning and preparation in order to ensure that all necessary information is obtained, without overloading and potentially crippling the tools that the information is intended to feed.

One approach is to segregate monitoring by enclave. Just as the separation of functional groups into enclaves helps minimize risk, it also helps to minimize the total information load that is generated by that enclave; that is, there are limited assets and activities within an enclave, and therefore there are less total logs and events.

To further complicate matters, operational technology (OT) activities and metrics must also be considered when securing industrial networks—representing new data types from yet another potentially overwhelming source of: new assets such as Remote Terminal Units (RTUs), Programmable Logic Controllers (PLCs), Intelligent Electronic Devices (IEDs), and other industrial assets; applications such as Human–Machine Interfaces (HMIs), and Historians; and networks such as fieldbus and smart grid networks.

TIP

When considering network monitoring and information management, it is helpful to benchmark the information load currently being produced in both IT and OT networks. IT networks require identifying which devices need to be monitored. This means understanding what servers, workstations, firewalls, routers, proxies, etc. (almost every IT device is capable of producing logs of some sort) are important—the process of determining critical assets described in Chapter 2, “About Industrial Networks,” and Chapter 7, “Establishing

Secure Enclaves," is helpful here. Once it has been determined which devices need to be monitored, the event load generated by these devices needs to be calculated. One method is to measure event load of a period of time that contains both normal and peak activity, and divide the total number of events by the time period (in seconds) to determine the average event per second (EPS) load of the network. Alternately, a worst-case calculation can be based entirely on peak event rates, which will result in a higher EPS target.[3]

In OT networks, most assets do not produce events or logs at all, and therefore they cannot be measured. They do produce information, however. This can be easily derived by looking at historized data from the control plants, and/or through the use of specialized industrial protocol monitors. Determine which assets you wish to monitor, and use the Data Historian system to determine the amount of information collected from these assets over time. This information will need to be normalized and centralized—either automatically via an SIEM or similar product, or manually via human time and effort—so it may be prudent to limit the amount of historized data that need to be exposed for security assessment. Some Historian tags—especially system tags concerning authentication, critical alarm tags concerning point or operational changes, stopped or failed processes, etc.—are obvious choices, while others may have little relevance to security.

Once the initial benchmark is obtained, add 10% for growth, and 10% for headroom. When sizing the IT network, it is also prudent to plan for "peak averages" where peak traffic rates occur for extended periods of time (i.e., the peak becomes the average), as this condition can occur during an extended attack, or as a result of a successful breach and subsequent infection with malware.[4] OT systems, on the other hand, may report different conditions but are less likely to report higher numbers of conditions unless the control process being historized has been significantly altered.

So what really needs to be monitored? The following guidelines help to identify what systems should be monitored.

Security Events

Security events are those events generated by security products: network- or host-based firewalls, Anti-Virus systems, intrusion detection and prevention systems, application monitors, application whitelisting systems, etc. Ideally, any event generated by a security device should be relevant, and therefore, these devices should be used for promiscuous monitoring. Realistically, false positives can dilute the relevance of security events.

NOTE

The term "false positive" is often misused. Because security logs and events originate from so many sources and are often generated quickly and in large quantities, false positives are often associated with what are seemingly irrelevant security data. When an alert is generated because a benign activity matches a detection signature of an Intrusion Prevention System (IPS), the result is a false positive. Similarly, if an Anti-Virus system falsely indicates that a file is infected, the result is a false positive. False positives make security analysis more difficult by generating extra data points that need to be assessed, potentially clouding real incidents from detection.

False positives can be minimized or eliminated through tuning of the faulty detection signatures: a process that should be performed regularly to ensure that detection devices are operating as efficiently as possible. However, while false positives often result in large amounts of unnecessary or irrelevant data, not all irrelevant data are false positives. Because of this common misconception, many security analysts and even security vendors are tempted to overly tune devices to eliminate any alert that occurs in large numbers. The issue with overly aggressive tuning is that while it will make incidents easier to manage in day-to-day operations, it can introduce false negatives—that is, when a real threat fails to create an alert, or when a correlation rule fails to trigger because a necessary condition was suppressed by over-tuning (see Chapter 8, "Exception, Anomaly, and Threat Detection"). Remembering that event correlation signatures are signature-matching rules that detect known threat patterns, the elimination of smaller seemingly irrelevant events can prevent detection of the larger pattern. Similarly, as new patterns are discovered by security researchers, event data that seem irrelevant today may become relevant in the future.

To ensure accurate threat detection and correlation, all legitimately produced events should be retained short term for live analysis (i.e., kept online) and long term for forensic and compliance purposes (i.e., kept offline) regardless of how irrelevant they may seem at the time of collection. Only true false positives—the events generated due to a false signature match—should be eliminated via tuning or filtering.

When considering the relevance of security events in industrial networks, consider the source of the event and its relevance to the specific enclave being monitored. For example, all enclaves should have at least one perimeter security device such as a firewall or IPS, but there may be host security events (Anti-Virus application whitelisting) and possibly internal Intrusion Detection System (IDS) or IPS, firewalls or other security devices (see Chapter 7, "Establishing Secure Enclaves"). One example is industrial security appliances that use industrial protocol and application monitoring to enforce how industrial protocols are used.

These logs might provide much more specific data to an enclave than do general security events, as seen in the example below from a Tofino industrial security appliance:

```
IP_DST = 192.168.1.1 LEN = 55 TOS = 0 TTL = 128 PROTO = TCP PORT_
SRC = 4516 PORT_DST = 502 SEQ = 3893700258 ACK_SEQ = 1852284905 URG = 0
ACK = 1 PSH = 1 RST = 0 SYN = 0 FIN = 0 Description: Function Code List:
The function code (16) is not in permitted function code list
```

In contrast, a generic Snort IDS might produce a syslog identifying a perimeter policy violation, such as the attempted Windows update shown below, but cannot provide the context of application function codes within the industrial network (see Chapter 4, "Industrial Network Protocols").

```
Jan 01 00:00:00 [69.20.59.59] snort: [1:2002948:6] ET POLICY
External Windows Update in Progress [**] [Classification: Potential
Corporate Privacy Violation] [Priority: 1] {TCP} 10.1.10.33:1665 ->
192.168.25.35:80
```

Assets

Assets—the physical devices within the network—also provide security data, typically in the form of logs. Assets can produce logs that track activity on a variety of levels: the operating system itself produces many logs, including system logs, application logs, and file system logs.

System logs are useful for tracking the status of devices and the services that are (or are not running), as well as when patches are (or are not) applied. Logs are useful for determining the general health of an asset as well as the validation that approved ports and services are running. These are also valuable in tracking which users (or applications) have authenticated to the asset, satisfying several compliance requirements. The following represent individual records from a Redhat Linux system log showing a successful user login, and a Windows failed authentication:

```
<345> Mar 17 11:23:15 localhost sshd[27577]: Accepted password for
knapp from ::ffff:10.1.1.1 port 2895 ssh2
<345> Fri Mar 17 11:23:15 2011 680 Security SYSTEM User Failure
Audit ENTERPRISE Account Logon attempt by:
MICROSOFT_AUTHENTICATION_PACKAGE_V1_0 Logon account: KNAPP Source
Workstation: ENTERPRISE Error Code: 0xC000006A 4574
```

Although syslog is ubiquitously used across a variety of systems, other event logging systems are used as well—the most notable of which is the Windows Management Instrumentation (WMI) framework. WMI produces auditable events in a structured data format that can be used against scripts (for automation) as well as by other Windows operating system functions.[5] Because syslog is so widely supported, WMI events are often logged using a Windows syslog agent to stream WMI events over syslog.

The following WMI event example indicates the creation of a new process on a Windows server:

```
Computer Name: WIN-0Z6H21NLQ05
Event Code: 4688
Type: Audit Success (4)
User Name:
Category: Process Creation
Log File Name: Security
String[%1]: S-1-5-19
String[%2]: LOCAL SERVICE
String[%3]: NT AUTHORITY
String[%4]: 0x3e5
String[%5]: 0xc008
String[%6]: C:\Windows\System32\RacAgent.exe
String[%7]: %%1936
String[%8]: 0xc5e4

Message: A new process has been created. Subject: Security ID:
S-1-5-19 Account Name: LOCAL SERVICE Account Domain: NT AUTHORITY
Logon ID: 0x3e5 Process Information: New Process ID: 0xc008 New
Process Name: C:\Windows\System32\RacAgent.exe Token Elevation Type:
TokenElevationTypeDefault (1) Creator Process ID: 0xc5e4 Token
Elevation Type indicates the type of token that was assigned to the
new process in accordance with User Account Control policy. Type 1
is a full token with no privileges removed or groups disabled. A
```

```
full token is only used if User Account Control is disabled or if
the user is the built-in Administrator account or a service account.
Type 2 is an elevated token with no privileges removed or groups
disabled. An elevated token is used when User Account Control is
enabled and the user chooses to start the program using Run as
administrator. An elevated token is also used when an application
is configured to always require administrative privilege or to
always require maximum privilege, and the user is a member of the
Administrators group. Type 3 is a limited token with administrative
privileges removed and administrative groups disabled. The limited
token is used when User Account Control is enabled, the application
does not require administrative privilege, and the user does not
choose to start the program using Run as administrator.
```

The same event, when collected via syslog using a WMI agent such as Snare, might look like this:

```
<12345> Fri Mar 17 11:23:15 2011||WIN-0Z6H21NLQ05||4688||Audit
Success (4)||||Process Creation||Security||S-1-5-19||LOCAL
SERVICE||NT AUTHORITY||0x3e5||0xc008||C:\Windows\System32\RacAgent.
exe||%%1936||0xc5e4
```

Application logs (covered in more detail under the section "Applications") provide a record of application-specific details such as logon activities to an HMI, configuration changes, and other details that indicate how an application is being used.

File system logs typically track when files are created, changed, or deleted, when access privileges or group ownerships are changed, and similar details. File system logging is included in Windows using the Windows File Protection (WFP) within WMI, which is an "infrastructure for management data and operations on Windows-based operating systems."[6] File monitoring in Unix and Linux systems is performed using **auditd**, and there are also commercial file integrity monitoring (FIM) products available such as Tripwire (www.tripwire.com) and nCircle (www.ncircle.com). These logs are extremely valuable for assuring the integrity of important files stored on an asset—such as configuration files (ensuring that the asset's configurations remain within policy), and the asset's log files themselves (ensuring that logged activities are valid and have not been tampered with to cover up indications of illicit behavior).

Configurations

Configuration monitoring refers to the process of monitoring baseline configurations for any indications of change,[7] and is only a small part of Configuration Management (CM). Basic configuration monitoring can be done at a rudimentary level through a combination of host configuration file monitoring (to establish the baseline), system and application log monitoring (to look for change actions) and FIM (to ensure that configurations are not altered). While this does not provide true CM, it does provide an indication as to when established configurations are altered, providing a valuable security resource.

Full CM systems provide additional key functions, typically mapping at least partially to the security controls outlined in NIST SP 800-53 under the section

"Configuration Management," which provides a total of nine configuration management controls:[8]

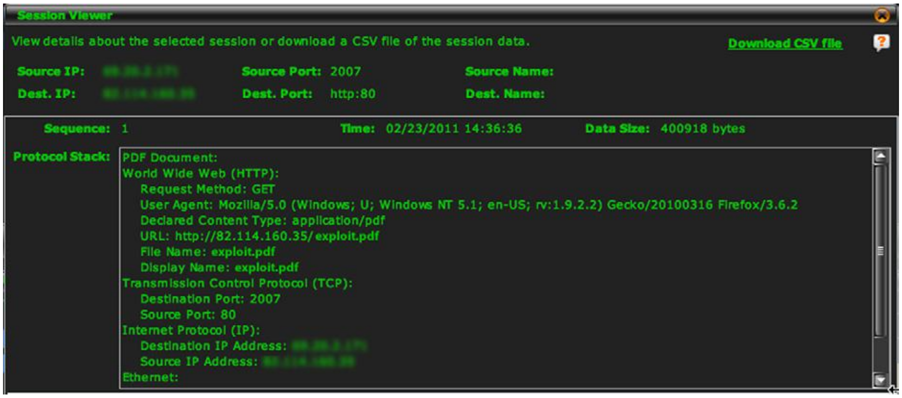
- Configuration management policy and procedures—establishes a formal, documented configuration management policy
- Baseline configurations—identifying and documenting all aspects of an asset's configurations to create a secure template against which all subsequent configurations are measured
- Change control—monitoring for changes and comparing changes against the established baseline
- Security impact analysis—the assessment of changes to determine and test how they might impact the security of the asset
- Access restrictions for change—limiting configuration changes to a strict subset of administrative users
- Configuration settings—identification, monitoring and control of security configuration settings and changes thereto
- Least functionality—the limitation of any baseline configuration to provide the least possible functionality to eliminate unnecessary ports and services
- Information service (IS) component (asset) inventory—establishing an asset inventory to identify all assets that are subject to CM controls, as well as to detect rogue or unknown devices that may not meet baseline configuration guidelines
- Establishment of a configuration management plan—assigning roles and responsibilities around an established CM policy to ensure that CM requirements are upheld

Configuration management tools may also offer automated configuration controls to allow batch configurations of assets across large networks, which is useful for ensuring that proper baseline configurations are used in addition to improving desktop management efficiencies. For the purposes of security monitoring, it is the monitoring and assessment of the configuration files themselves that is a concern. This is because an attacker will often attempt to either escalate user privileges in order to obtain higher levels of access, or alter the configurations of security devices in order to penetrate deeper into secured enclaves—both of which are detectable with appropriate CM controls in place.

The logs produced by the CM are therefore a useful component of overall threat detection by using change events in combination with other activities such as an event correlation system. For example, a port scan, followed by an injection attempt on a database, followed by a configuration change on the database server is indicative of a directed penetration attempt. Change logs are also highly beneficial for compliance and regulatory purposes, with configuration and change management being a common requirement of most industrial security regulations (see Chapter 10, "Standards and Regulations").

Applications

Applications run on top of the operating system and perform specific functions. While monitoring application logs can provide a record of the activities relevant



View details about the selected session or download a CSV file of the session data.

Source IP: [redacted] **Source Port:** 2007 **Source Name:**
Dest. IP: [redacted] **Dest. Port:** http:80 **Dest. Name:**

Sequence: 1 **Time:** 02/23/2011 14:36:36 **Data Size:** 400918 bytes

Protocol Stack:

```
PDF Document:
World Wide Web (HTTP):
   Request Method: GET
   User Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.9.2.2) Gecko/20100316 Firefox/3.6.2
   Declared Content Type: application/pdf
   URL: http://82.114.160.35/exploit.pdf
   File Name: exploit.pdf
   Display Name: exploit.pdf
Transmission Control Protocol (TCP):
   Destination Port: 2007
   Source Port: 80
Internet Protocol (IP):
   Destination IP Address: [redacted]
   Source IP Address: [redacted]
Ethernet:
```

FIGURE 9.1

Application Session Details from an Application Monitor.

to those functions, direct monitoring of applications using a dedicated application monitoring product or application content firewall will provide a granular account of all application activities. Application logs can include when an application is executed or terminated, who logs into the application, and specific actions performed by users once logged in. The information contained in application logs is a summary, as it is in all log records. A sample application log record generated by an Apache web server is provided below:

```
Jan 01 00:00:00 [69.20.32.12] 93.80.237.221 - - [24/
Feb/2011:01:56:33 -0000] "GET/spambot/spambotmostseendownload.
php HTTP/1.0" 500 71224 "http://yandex.ru/yandsearch?text =video.
krymtel.net" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1;
MRA 4.6 (build 01425))"
```

For more detailed accounting of application activity, an application monitoring system can be used. For example, while it is possible that malware might be downloaded over HTTP, and be indicated in a log file such as the example shown above, monitoring an application's contents across a session could indicate malware that is embedded in a file being downloaded from an otherwise normal-seeming website, as shown in Figure 9.1.

Networks

Network flows are records of network communications, from a source to one or more destinations. Flows are typically tracked by network infrastructure devices such as switches and routers. Flow collection is typically proprietary to the network device manufacturer (e.g., Cisco supports netFlow, Juniper supports jFlow, etc.), although many vendors also support the sFlow standard.

Monitoring flows provides an overview of network usage over time (for trending analysis, capacity planning, etc.) as well as at any given time (for impact analysis, security assessment, etc.), and can be useful for a variety of functions, including the following:[9]

- Network diagnosis and fault management
- Network traffic management or congestion management
- Application management, including performance management, and application usage assessments
- Application and/or network usage accounting for billing purposes
- Network security management, including the detection of unauthorized devices, traffic, etc.

Network flow analysis is extremely useful for security analysis because it provides the information needed to trace the communications surrounding a security incident back to its source. For example, if an application whitelisting agent detects malware on an asset, it is extremely important to know where that malware came from, as it has already breached the perimeter defenses of the network and is now attempting to infect machines. By correlating the malware attempt to network flows, it may be possible to trace the source of the malware and may also provide a path of propagation (i.e., where else did the virus spread to).

For industrial network security, network flow analysis also provides an indication of network performance, which is important because of the negative impact that network performance can have on process quality and efficiency, as shown in Table 9.1. For example, an increase in latency can cause certain industrial protocols to fail, halting industrial processes.[10]

User Identities and Authentication

Monitoring users and their activities is an ideal method for obtaining a clear picture of what is happening on the network, and who is responsible. User monitoring is also an important component of compliance management, as most compliance regulations require specific controls around user privileges, access credentials, roles, and behaviors.

Unfortunately, the term “user” is vague: there are user account names, domain names, host names, and of course the human user’s identity. While the latter is what is most often required for compliance management (see Chapter 10, “Standards and Regulations”), the former are what is typically provided within digital systems. Authentication to a system requires a username and password, from a machine that has a host name, which might be one of several hosts in a named domain. The application itself might then authenticate to another backend system (such as a database), which has its own name and to which the application authenticates using yet another set of credentials. To further complicate things, the same human operator might need to authenticate to several systems, from several different machines, and may use a unique username on each.

Table 9.1 Network Flow Details

Flow Detail	What It Indicates	Security Ramifications
SNMP interface indices (ifIndex in IF-MIB)	The size of the flow in terms of traffic volume (bytes, packets, etc.), as well as errors, latency, discards, physical addresses (MAC addresses), etc.	SNMP details can provide indications of abnormal protocol operation that might indicate a threat More germane to industrial networks, the presence of interface errors, latency, etc. can be directly harmful to the correct operation of many industrial protocols (see Chapter 4, “Industrial Network Protocols”)
Flow start time	When a network communication was initiated and when it ended	Essential for the correlation of communications against security events
Flow end time	Collectively, the start and stop timestamps also indicate the duration of a network communications	
Number of bytes/ packets	Indicates the “size” of the network flow, indicative of how much data is being transmitted	Useful for the detection of abnormal network access, large file transfers, as might occur during information theft (e.g., retrieving a large database query result, downloading sensitive files, etc.)
Source and destination IP addresses	Indicates where a network communication began and where it was terminated	Essential for the correlation of related logs and security events (which often track IP address details)
Source and destination port	Note that in non-IP industrial networks, the flow may terminate at the IP address of an MI or PLC even though communications may continue over specialized industrial network protocols	IP addresses may also be used to determine the physical switch or router interface of the asset, or even the geographic location of the asset (through the use of a geo-location service)

It is therefore necessary to normalize users to a common identity, just as it is necessary to normalize events to a common taxonomy. This can be done by monitoring activities from a variety of sources (network, host, and application logs), extracting whatever user identities might be present, and correlating them against whatever clues might be preset within those logs. For example, if a user authenticates to a Windows machine, launches an application and authenticates to it, and then the application authenticates to a backend system, it is possible to track that activity back to the original username by looking at the source of the authentications and the time at which they occurred; because they occurred from the same physical console in clear succession, it can be assumed that all three authentications were by the same user.

As the systems become more complex and distributed, and as the number of users increases, each with specific roles and privileges, this can become cumbersome, and an automated identity management mechanism may be required.

This process is made simpler through the use of common directories, such as Microsoft Active Directory and/or the **Lightweight Directory Access Protocol** (LDAP), which act as identity directories and repositories. However, there may still be several unique sets of credentials per human operator. The difficulty lies in the lack of common log formats, and the corresponding lack of universal identities between diverse systems. User monitoring therefore requires the extraction of user information from a variety of network and application logs, followed by the normalization of that identity information. John Doe might log into a Windows domain using the username j.doe, have an e-mail address of jdoe@company.com, log into a corporate intranet or CMS as johnnyd, etc. To truly monitor user behavior, the recognition of j.doe, jdoe, and johnnyd as a single identity is necessary.

Several commercial **Identity Access Management** (IAM) systems (also sometimes referred to as Identity and Authentication Management systems) are available to facilitate this process. Some commercially available IAM systems include: Novell, Oracle Identity Management (www.oracle.com/technetwork/middleware/id-mgmt/overview), Sun Identity Management (www.sun.com/software/index.jsp?cat=Identity%20Management&tab=3), and Tivoli Identity Manager (www.01.ibm.com/software/tivoli/products/identity-mgr). Other third-party identity solutions, such as Securonix Identity Matcher (www.Securonix.com) offers features of both a centralized directory and IAM by mining identity information from other IAMs and normalizing everything back to a common identity.[11] More sophisticated SIEM and Log Management systems might also provide identity correlation features to provide user normalization. Whatever method is used, by managing and controlling authentications to multiple systems via a centralized IAM, an authoritative source of identity is provided, as shown in Figure 9.2.

Once the necessary identity context has been obtained, it can be utilized in the information and event management process to cross-reference logs and events back to users. For example, in Figure 9.3, an SIEM dashboard shows both network and event details associated with their source users.

Additional Context

While user identity is one example of contextual information, there is a wealth of additional information available that can provide context. This information—such as vulnerability references, IP reputation lists, and threat directories, etc.—supplements the monitored logs and events with additional valuable context. Examples of contextual information are provided in Table 9.2.

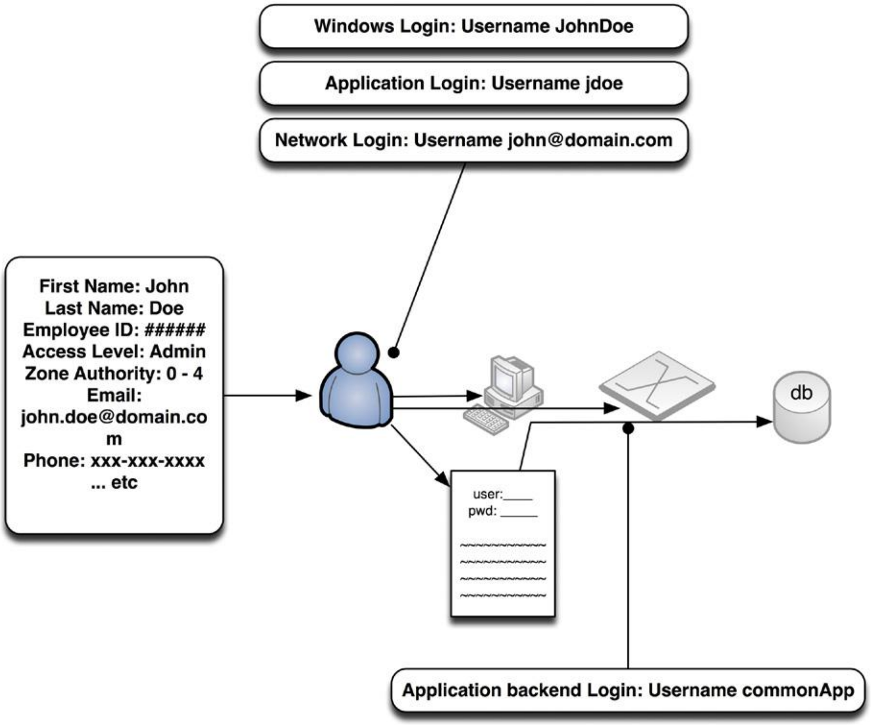
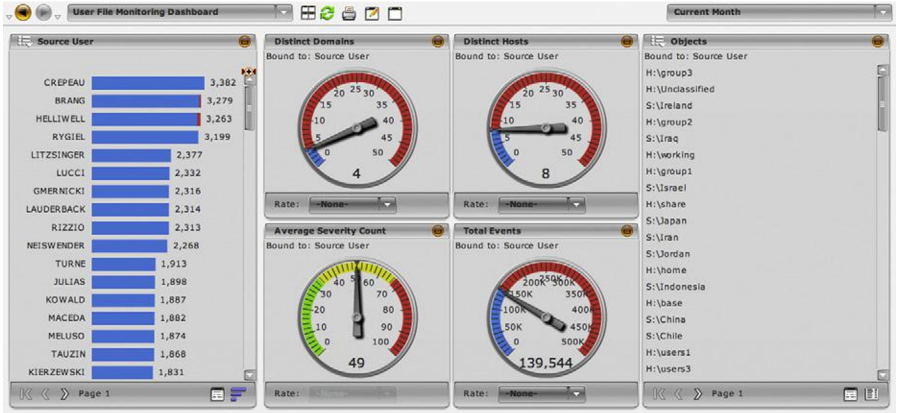


FIGURE 9.2

Normalization of User Identity.


User File Monitoring Dashboard

Current Month

Source User

CREPEAU	3,382
BRANG	3,279
HELLIWELL	3,263
RYGIEL	3,199
LITZSINGER	2,377
LUCCI	2,332
GMERNICKI	2,316
LAUDERBACK	2,314
RIZZIO	2,313
NEISWENDER	2,268
TURNE	1,913
JULIAS	1,898
KOWALD	1,887
MACEDA	1,882
MELUSO	1,874
TAUZIN	1,868
KIERZEWSKI	1,831

Page 1

Distinct Domains
Bound to: Source User
4
Rate: -None-

Distinct Hosts
Bound to: Source User
8
Rate: -None-

Average Severity Count
Bound to: Source User
49
Rate: -None-

Total Events
Bound to: Source User
139,544
Rate: -None-

Objects
Bound to: Source User
H:\group3
H:\Unclassified
S:\Ireland
H:\group2
S:\Iraq
H:\working
H:\group1
S:\Israel
H:\share
S:\Japan
S:\Iran
S:\Jordan
H:\home
S:\Indonesia
H:\base
S:\China
S:\Chile
H:\users1
H:\users3

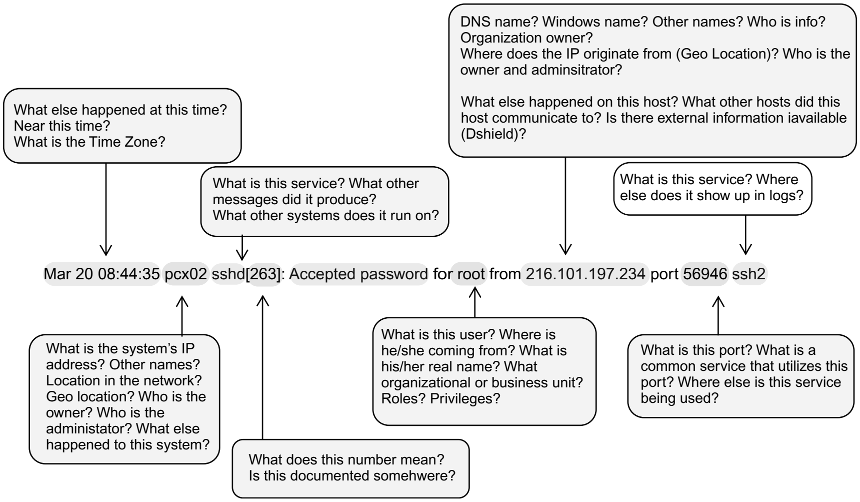
Page 1

FIGURE 9.3

User Activity Related to File Access as Displayed by an SIEM.

Table 9.2 Contextual Information Sources and Their Relevance

Information Source	Provided Context	Security Implications
Directory services (e.g., active directory)	User identity information, asset identity information, and access privileges	Provides a repository of known users, assets, and roles that can be leveraged for security threat analysis and detection, as well as for compliance
Identity and Authentication Management systems	Detailed user identity information, usernames and account aliases, access privileges, and an audit trail of authentication activity	Enables the correlation of users to access and activities based upon privilege and policy. When used to enrich security events, provides a clear audit trail of activity versus authority that is necessary for compliance auditing
Vulnerability scanner	Asset details including the operating system, applications in use (ports and services), patch levels, identified vulnerabilities, and related known exploits	Enables security events to be weighted based upon the vulnerability of their target (i.e., a Windows virus is less concerning if it is targeting a Linux workstation) Also provides valuable asset details for use in exception reporting, event correlation, and other functions
Penetration tester	Exploitation success/failure, method of exploitation, evasion techniques, etc.	Like with a vulnerability scanner, pen test tools provide the context of an attack vector. Unlike VA scan results, which show what could be exploited, a pen test indicates what has been exploited—which is especially useful for determining evasion techniques, detecting mutating code, etc.
Threat database/CERT	Details, origins and recommendations for the remediation of exploits, malware, evasion techniques, etc.	Threat intelligence can be used in a purely advisory capacity (e.g., providing educational data associated with a detected threat), or in an analytical capacity (e.g., in association with vulnerability scan data to weight the severity calculation of a detected threat) Threat intelligence may also be used as "watchlists," providing a cross-reference against which threats can be compared in order to highlight or otherwise call out threats of a specific category, severity, etc.



DNS name? Windows name? Other names? Who is info? Organization owner?
Where does the IP originate from (Geo Location)? Who is the owner and adminsitrator?

What else happened on this host? What other hosts did this host communicate to? Is there external information iavailable (Dshield)?

What else happened at this time? Near this time? What is the Time Zone?

What is this service? What other messages did it produce? What other systems does it run on?

What is this service? Where else does it show up in logs?

```
Mar 20 08:44:35 pcx02 sshd[263]: Accepted password for root from 216.101.197.234 port 56946 ssh2
```

What is the system's IP address? Other names? Location in the network? Geo location? Who is the owner? Who is the administator? What else happened to this system?

What is this user? Where is he/she coming from? What is his/her real name? What organizational or business unit? Roles? Privileges?

What is this port? What is a common service that utilizes this port? Where else is this service being used?

What does this number mean? Is this documented somehwere?

FIGURE 9.4

A Log File, Illustrating the Lack of Context Image.

Courtesy of Dr. Anton A. Chuvakin, Security Warrior Consulting.

Contextual information is always beneficial, as the more context is available for any specific event or group of events, the easier it will be to assess relevant to specific security and business policies. This is especially true because the logs and events being monitored often lack the details that are most relevant, such as usernames (see Figure 9.4).[12]

However, contextual information adds to the total volume of information already being assessed; as such, it is most useful when used to enrich other security information in an automated manner (see section "Information Management").

Behavior

Behavior is not something that is directly monitored. Rather, it is the analysis of any monitored metric (obtained from a log, network flow, or other source) over time. The result is an indication of expected versus unexpected activity, which is extremely useful for a wide range of security functions, including anomaly based threat detection, as well as capacity or threshold-based alarming. Behavior is also a useful condition in security event correlation (see Chapter 8, "Exception, Anomaly, and Threat Detection").

Behavior analysis is often provided by security log and event monitoring tools, such as Log Management systems, SIEMs, and Network Behavior Anomaly Detection (NBAD) systems. If the system used for the collection and monitoring of security information does not provide behavioral analysis, an external tool such as a spreadsheet or statistics program may be required.

SUCCESSFULLY MONITORING ENCLAVES

Understanding what to monitor is only the first step: actually monitoring all of the users, networks, applications, assets, and other activities still needs to happen. The discussion of what to monitor focused heavily on logs, as log files are designed to describe activities that have occurred, are fairly ubiquitous, and are well understood. However, log files are not always available, and they may not provide sufficient detail in some instances. Therefore, monitoring is typically performed using a combination of methods, including the following:

- Log collection and analysis
- Direct monitoring or network inspection
- Inferred monitoring via tangential systems

Except in pure log-collection environments, where logs are produced by the assets and network devices that are already in place, specialized tools are required to monitor the various network systems. In addition, the results of monitoring—by whatever means—need to be dealt with; while manual logs and event reviews are possible (and allowed by most compliance regulations), automated tools are available and are recommended.

However, the central analysis of monitored systems is contrary to a security model built upon functional isolation. That is, industrial networks should be separated into functional enclaves, and centralized monitoring requires that log and event data either remain within a functional group—limiting the value for overall situation awareness—or be shared between enclaves—potentially putting the security of the enclave at risk. In the first scenario, logs and events are not allowed across the enclave perimeter; they may be collected, retained, and analyzed only by local systems within that enclave. In the second scenario, special considerations must be made for the transportation of log and event data across enclave perimeters to prevent the introduction of a new inbound attack vector. A common method is to implement special security controls—either a data diode, unidirectional gateway, or a firewall configured to explicitly deny all inbound communications—to ensure that the security data is only allowed to flow toward the centralized management system. Especially in industrial networks where critical systems in remote areas need to operate reliably, a hybrid approach may be used: providing local security event and log collection and management so that the enclave can operate in total isolation, while also pushing security data to a central location to allow for more complete situational awareness across multiple enclaves.

Log Collection

Log collection is just that: the collection of logs from whatever sources produce them. This is often simply a matter of directing the log output to a log aggregation point, such as a network storage facility and/or a dedicated Log Management system. Directing a log is often as simple as directing the syslog to the IP address of the aggregator. In some cases, such as Windows WMI, events are stored locally within a database rather than as log files. These events must be retrieved, either directly (by authenticating to Windows and querying the event database) or indirectly (via a software agent such as Snare, which retrieves the events locally and then transmits them via standard syslog).

Direct Monitoring

Direct monitoring refers to the use of a probe or other device to examine network traffic or hosts directly. Direct monitoring is especially useful when the system being monitored does not produce logs natively (as is the case with many industrial network assets, such as RTUs, PLCs and IEDs). It is also useful as a verification of activity reported by logs, as log files can be altered deliberately in order to hide evidence of malicious activities. Common monitoring devices include Firewalls, Intrusion Detection Systems (IDSs), **Database Activity Monitors** (**DAMs**), Application Monitors, and Network Probes. These are often available commercially as software or appliances, or via open source distributions such as Snort (an IDS available at www.snort.org), Wireshark (a network sniffer and traffic analyzer available at www.wireshark.org), and the wireless sniffer Kismet (www.kismetwireless.net).

Often, network monitoring devices produce logs of their own, which are then collected for analysis along with other logs. Because the logs are produced without any direct interaction with the system being monitored, network monitoring devices are sometimes referred to as "passive logging" devices. Database Activity Monitors, for example, monitor database activity on the network—often on a span port or network tap. The DAM decodes network packets and then extracts relevant SQL transactions in order to produce logs. There is no need to enable logging on the database itself, and as a result there is no performance impact to the database servers.

In industrial networks, it is similarly possible to monitor industrial protocol use on the network, providing "passive logging" to those industrial control assets that do not support logging. Passive monitoring is especially important in these networks, as many industrial protocols operate in real time and are highly susceptible to network latency. This is one reason why it is difficult to deploy logging agents on the devices themselves (which would also complicate asset testing policies), making passive network logging an ideal solution.

In some instances, the device may use a proprietary log format or event streaming protocol that must be handled specially. For example, Cisco's Security Device Event Exchange protocol (SDEE), used by most Cisco IPS products, requires a username and password in order to authenticate with the security device so that events can be retrieved on demand, and/or "pushed" via a subscription model. While the end result is the same, it is important to understand that syslog is not absolutely ubiquitous.

Inferred Monitoring

Inferred monitoring refers to situations where one system is monitored in order to infer information about another system. For example, many applications connect to a database; monitoring the database in lieu of the application itself will provide valuable information about how the application is being used, even if the application itself is not producing logs or being directly monitoring by an Application Monitor.

NOTE

Network-based monitoring inevitably leads to the question, "Is it possible to monitor encrypted network traffic?" Many industrial network regulations and guidelines recommend the encryption of control data . . . so how can this data be monitored via a network probe? There are a few options, each with benefits and weaknesses. The first is to monitor the sensitive network connection between the point of encryption and the traffic source. That is, encrypt network traffic externally using a network-based encryption appliance, and place the network probe immediately between the asset and the encryption. While effective, this does technically weaken the security of the network. The second option is to utilize a dedicated network-based decryption device, such as the Netronome SSL Inspector (www.sslinspector.com). These devices perform deliberate, hardware-based Man-in-the-Middle attacks in order to break encryption and analyze the network contents for security purposes. A third option is not to monitor the encrypted traffic at all, but rather to monitor for instances of data that should be encrypted (such as industrial protocol function codes) but are not—producing exception alerts indicating that sensitive traffic is not being encrypted.

To determine which tools are needed, start with your enclave perimeter and interior security controls (see Chapter 7, "Establishing Secure Enclaves") and determine which can or cannot produce adequate monitoring. If they can, start by aggregating logs from the absolute perimeter (the demarcation between the least critical enclave and the Internet—typically the business enterprise LAN) to a central log aggregation tool (see the section "Information Collection and Management Tools"). Next, begin aggregating logs from those devices protecting the most critical enclaves, and work outward until all available monitoring has been enabled, or until the capacity of your log aggregation has become saturated. At this point, if there are remaining critical assets that are not being effectively monitored, it may be necessary to increase the capacity of the log aggregation system.

TIP

Adding capacity does not always mean buying larger, more expensive aggregation devices. Distribution is also an option: keep all log aggregation local within each enclave (or within groups of similar enclaves), and then aggregate subsets of each enclave to a central aggregation facility for centralized log analysis and reporting. While this type of event reduction will reduce the effectiveness of threat detection and will produce less comprehensive reports from the centralized system, all the necessary monitoring and log collection will remain intact within the enclaves themselves, where they can be accessed as needed.

If all logs are being collected and there are still critical assets that are not adequately monitored, it may be necessary to add additional network monitoring tools to compensate. This process is illustrated in Figure 9.5.

Additional monitoring tools could include any asset or network monitoring device, including host-based security agents, or external systems such as an

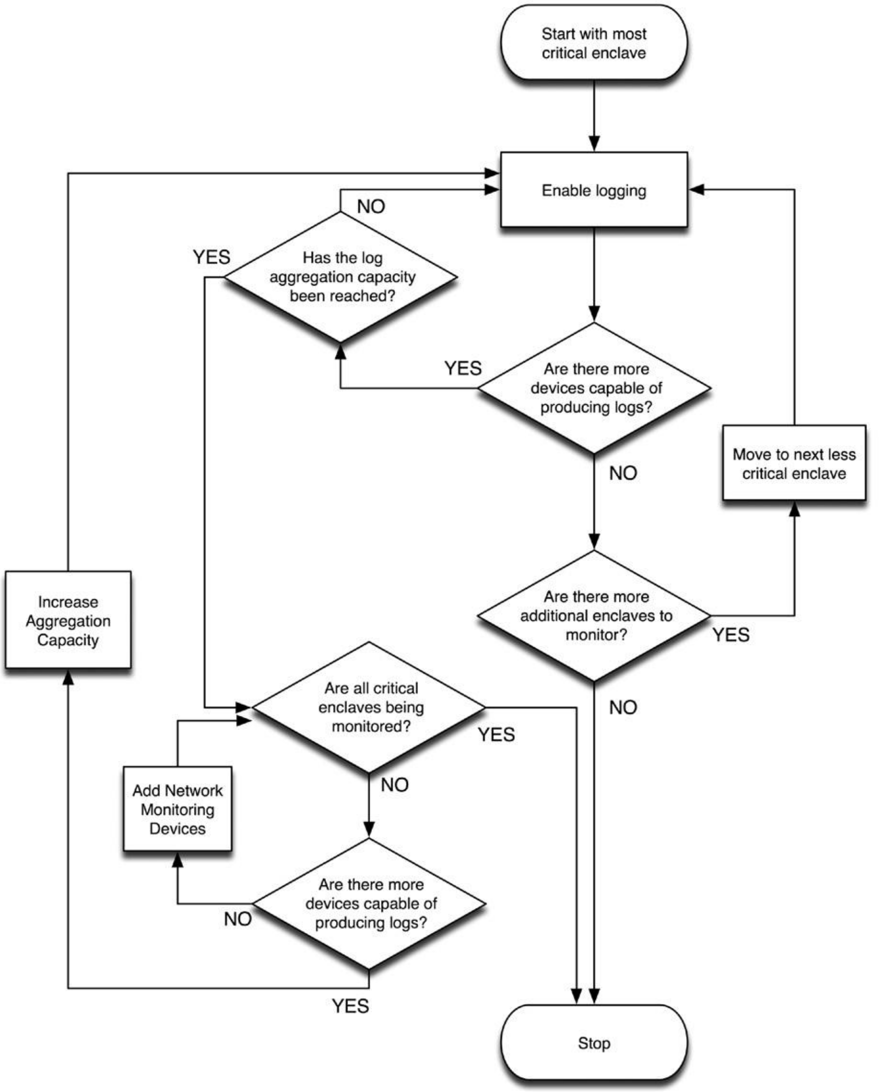

Start with most critical enclave
Enable logging
NO
YES
Has the log aggregation capacity been reached?
YES
Are there more devices capable of producing logs?
NO
Move to next less critical enclave
Increase Aggregation Capacity
Are there more additional enclaves to monitor?
YES
NO
Are all critical enclaves being monitored?
YES
NO
Add Network Monitoring Devices
Are there more devices capable of producing logs?
NO
YES
Stop

FIGURE 9.5

Process for Enabling Enclave Monitoring.

Intrusion Detection System, an Application Monitor, or an Industrial Protocol Filter. Network-based monitoring tools are often easier to deploy, because they are by nature nonobtrusive—and, if configured to monitor a spanned or mirrored interface, do not incur latency.

CAUTION

Remember—when aggregating logs it is still necessary to respect the boundaries of all established enclaves. If logs need to be aggregated across enclaves (which is helpful for the detection of threats as they move between enclaves) make sure that the enclave perimeter is configured to only allow the movement of logs in one direction; otherwise, the perimeter will be compromised. In most instances, simply creating a policy that explicitly states the source (the device producing logs) and the destination (the log aggregation facility) for the specified service (e.g., syslog, port 514) is sufficient in order to enforce a restricted one-way transmission of the log files. For critical enclaves, physical separation using a data diode or unidirectional gateway may be required to assure that all log transmissions occur in one direction, and that there is no ability for malicious traffic to enter the secure enclave from the logging facility.

Information Collection and Management Tools (Log Management Systems, SIEMs)

The "log collection facility" is typically a Log Management system or a Security Information and Event Management (SIEM) system. These tools range from very simple to very complex and include free, open-source, and commercial options. Some options include Syslog Aggregation and Log Search, commercial Log Management systems, the Open Source Security Information Management (**OSSIM**) system, and commercial Security Information and Event Management systems.

Syslog Aggregation and Log Search

Syslog allows log files to be communicated over a network. By directing all syslog outputs from supported assets to a common network file system, a very simple and free log aggregation system can be established. While inexpensive (essentially free), this option provides little added value in terms of utilizing the collected logs for analysis, requiring the use of additional tools such as open source Log Search or IT Search tools, or through the use of a commercial Log Management System or SIEM. In addition, if logs are being collected for compliance purposes as well as for security monitoring, additional measures will need to be taken to comply with log retention requirements. These requirements include nonrepudiation and chain of custody, ensuring that files have not been altered, or accessed by unauthorized users. Again, this can be obtained without the help of commercial systems, although it does require additional effort by IT managers.

Log Management Systems

Log Management systems provide a commercial solution for log collection, analysis and reporting. Log Management systems provide a configuration interface to manage log collection, as well as options for the storage of logs—often allowing the administrator to configure log retention parameters by individual log source. At the time of collection, Log Management systems also provide the necessary nonrepudiation features to ensure the integrity of the log files, such as "signing" logs with a calculated hash that can be later compared to the files as a checksum. Once collected, the logs can then also be analyzed and searched, with the ability to

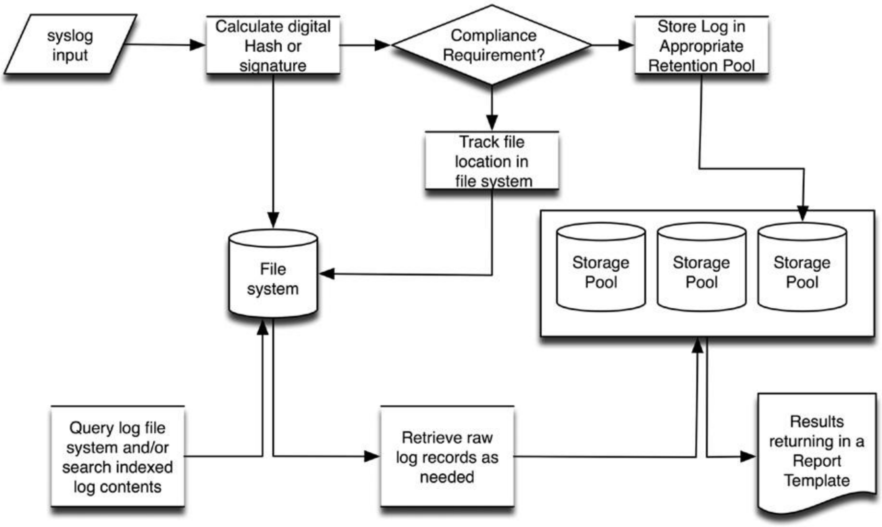

syslog input
Calculate digital Hash or signature
Compliance Requirement?
Store Log in Appropriate Retention Pool
Track file location in file system
File system
Storage Pool
Storage Pool
Storage Pool
Query log file system and/or search indexed log contents
Retrieve raw log records as needed
Results returning in a Report Template

FIGURE 9.6

Typical Log Management Operations.

produce pre-filtered reports in order to present log data relevant to a specific purpose or function—such as compliance reports, which produce log details specific to one or more regulatory compliance controls, as shown in Figure 9.6.

Security Information and Event Management Systems

Security Information and Event Management Systems, or SIEMs, extend the capabilities of Log Management systems with the addition of specific analytical and contextual functions. According to security analysts from Gartner, the differentiating quality of an SIEM is that it combines the log management and compliance reporting qualities of a Log Management or legacy Security Information Management (SIM) system with the real-time monitoring and incident management capabilities of a Security Event Manager (SEM).[13] Further, an SIEM must support "data capture from heterogeneous data sources, including network devices, security devices, security programs and servers,"[14] making the qualifying SIEM an ideal platform for providing situational awareness across enclaves perimeters and interiors.

Many SIEM products are available, including the open source OSSIM project (www.sourceforge.net/projects/os-sim/), as well as several commercial SIEMs, competing across a variety of markets, and offering a variety of value-added features and specializations.

Because an SIEM is designed to support real-time monitoring and analytical functions, it will parse the contents of a log file at the time of collection, storing the parsed information in some sort of structured data store, typically a database or a specialized flat-file storage system. By parsing out common values, they are more readily available for analytics, helping to support the real-time goals of the

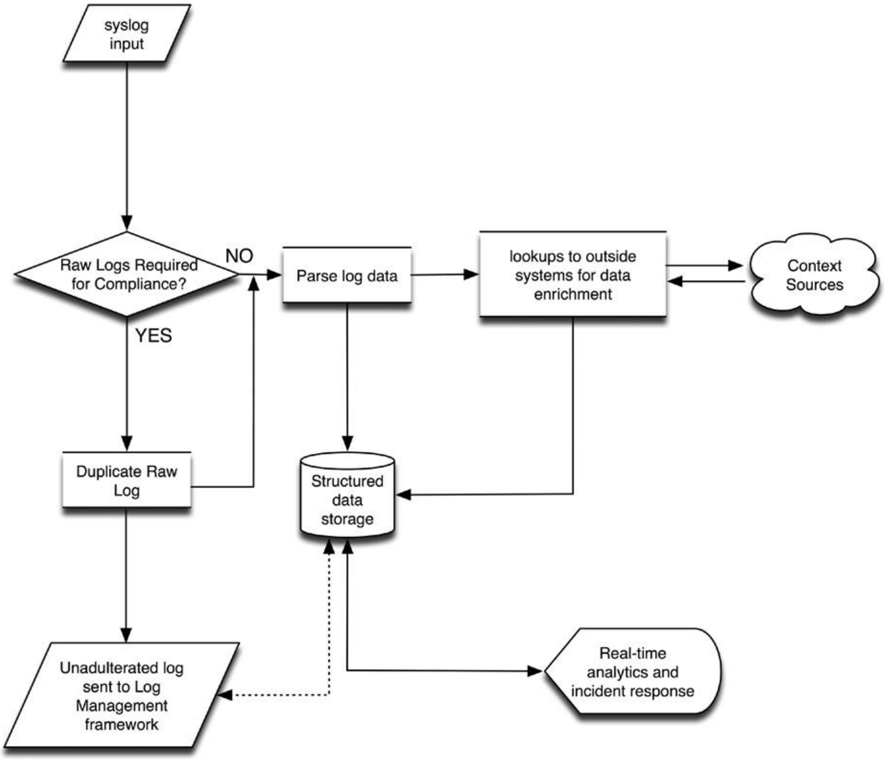

syslog input
Raw Logs Required for Compliance?
NO
YES
Parse log data
lookups to outside systems for data enrichment
Context Sources
Duplicate Raw Log
Structured data storage
Unadulterated log sent to Log Management framework
Real-time analytics and incident response

FIGURE 9.7

Typical SIEM Operations.

SIEM, as shown in Figure 9.7. The parsed data are used for analytics, while raw log data are managed separately by a more traditional Log Management framework that will hash the logs and retain them for compliance. Because the raw log file may be needed for forensic analysis, a logical connection between the log file and the parsed event data is typically maintained within the data store.

SIEM platforms are often used in Security Operations Centers (SOCs), providing intelligence to security operators that can be used to detect and respond to security concerns. Typically, the SIEM will provide visual dashboards to simplify the large amounts of disparate data into a more human-readable form.

NOTE

Log Management and SIEM platforms are converging as information security needs become more closely tied to regulatory compliance mandates. Many traditional Log Management vendors now offer SIEM features, while traditional SIEM vendors are offering Log Management features.

Data Historians

Data Historians are not security monitoring products, but they do monitor activity (see Chapter 5, "How Industrial Networks Operate") and can be a useful supplement to security monitoring solutions in several ways, including the following:

- Providing visibility into control system assets that may not be visible to typical network monitoring tools
- Providing process efficiency and reliability data that can be useful for security analysis

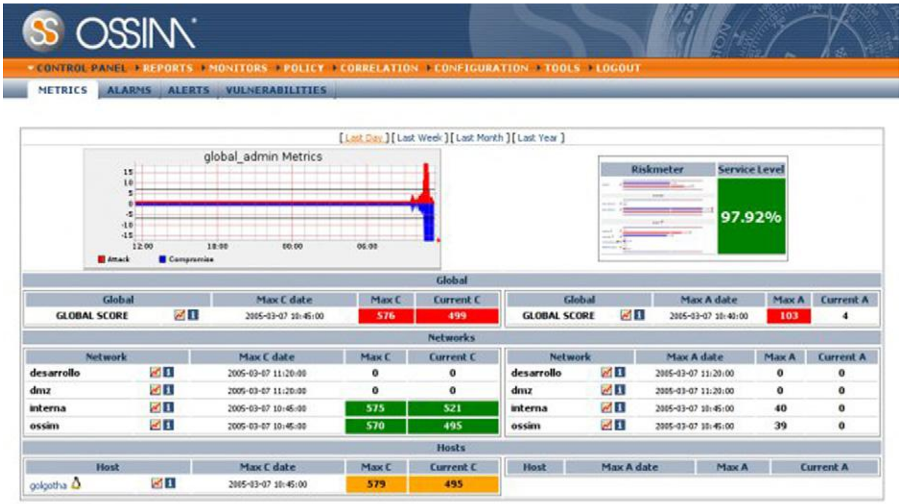
Because most security monitoring tools are designed for enterprise network use, they are typically restricted to TCP/IP networks and therefore have no visibility into large portions of most industrial plants, which may utilize serial connectivity or other non-routable protocols. However, with many industrial protocols evolving to operate over Ethernet and/or over TCP/IP, these processes can be impacted by enterprise network activities. By using the operational data provided by a Historian, the security analysis capabilities of SIEM are made available to operational data, allowing threats that originate in IT environments but target OT systems (i.e., Stuxnet), to be more easily detected and tracked by security analysts. In addition, by exposing IT network metrics to operational processes, those activities that could impact the performance and reliability of industrial automations systems can be detected as well, for example, increased network flow activity, heightened latency, or other metrics that could impact the proper operation of industrial network protocols (see Chapter 4, "Industrial Network Protocols").

Monitoring Across Secure Boundaries

As mentioned in the section "Successfully Monitoring Enclaves," it is sometimes necessary to monitor systems across secure enclave boundaries. This requires enclave perimeter security policies that will allow the security logs and events generated by the monitoring device(s) to be transferred to a central management console. Data diodes are ideal for this application as they force the information flow in one direction—away from the secured enclaves and toward the central management system. If a firewall is used, any "hole" provided for logs and events represents a potential vector of attack; the configuration must therefore explicitly limit the communication from the originating source(s) to the destination management system, by IP and Port, with no allowed return communication path. Ideally, this communication would be encrypted as well, as the information transmitted could potentially be sensitive in nature.

INFORMATION MANAGEMENT

Having successfully collected the necessary information, the next step in security monitoring is to utilize the relevant security information that has been collected. Proper analysis of this information can provide the situational awareness necessary to detect incidents that could impact the safety and reliability of the industrial network.



Global

Global		Max C date	Max C	Current C
GLOBAL SCORE		2005-03-07 10:45:00	576	499

Global		Max A date	Max A	Current A
GLOBAL SCORE		2005-03-07 10:40:00	103	4

Networks

Network		Max C date	Max C	Current C
desarrollo		2005-03-07 11:20:00	0	0
dmz		2005-03-07 11:20:00	0	0
interna		2005-03-07 10:45:00	575	521
ossim		2005-03-07 10:45:00	570	495

Network		Max A date	Max A	Current A
desarrollo		2005-03-07 11:20:00	0	0
dmz		2005-03-07 11:20:00	0	0
interna		2005-03-07 10:45:00	40	0
ossim		2005-03-07 10:45:00	39	0

Hosts

Host		Max C date	Max C	Current C
golgotha		2005-03-07 10:45:00	579	495

Host	Max A date	Max A	Current A

FIGURE 9.8

The Open Source Security Information Management Project.

Ideally, the SIEM or Log Manager will perform many underlying detection functions automatically—including normalization, data enrichment, and correlation (see Chapter 8, "Exception, Anomaly, and Threat Detection")—providing the security analyst with the following types of information at their disposal:

- The raw log and event details obtained by monitoring relevant systems and services, normalized to a common taxonomy
- The larger "incidents" or more sophisticated threats derived from those raw events
- The associated necessary context to what has been both observed (raw events) and derived (**correlated events**)

Typically, an SIEM will represent a high-level view of the available information in a dashboard or console, as illustrated in Figure 9.8, which shows the dashboard of the Open Source Security Information Management (OSSIM) platform. With this information in hand, automated and manual interaction with the information can occur. The information can be queried directly, to achieve direct answers to explicit questions; it can be formulated into a report to satisfy specific business, policy or compliance goals; it can be used to proactively or reactively notify a security or operations officer of an incident; and it can be used to further investigate incidents that have already occurred.

Queries

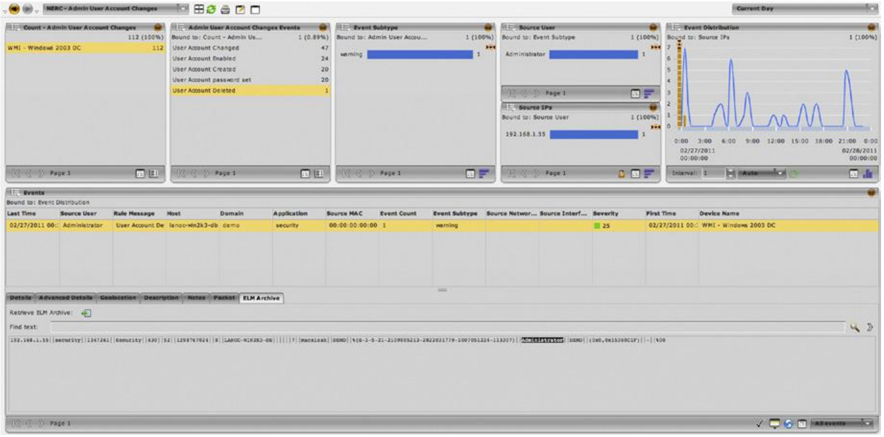
The term "query" refers to a request for information from the centralized data store. This can sometimes be an actual database query, using Structured Query Language (SQL), or may be a plain-text request to make the information more accessible by users without database administration skills (although these requests may use SQL queries internally, hidden from the user). Common examples of initial queries include the following:

- Top ten talkers (by total network bandwidth used)
- Top talkers (by unique connections or flows)
- Top events (by frequency)
- Top events (by severity)
- Top events over time
- Top applications in use
- Open ports

These requests can be made against any or all data that is available in the data store (see the section "Data Availability"). Queries can be focused by providing additional conditions or filters, providing results more relevant to a specific situation. For example:

- Top 10 talkers during nonbusiness hours
- Top talkers using specific industrial network protocols
- All events of a common type (e.g., user account changes)
- All events targeting a specific asset or assets (e.g., critical assets within a specific enclave)
- All ports and services used by a specific asset or assets
- Top applications in use within more than one enclave

Query results can be returned in a number of ways: in delimited text files, via a graphical user interface or dashboard, via pre-formatted executive reports, via an alert that is delivered by text or e-mail, etc. Figure 9.9 shows user activity filtered



NERC - Admin User Account Changes

Current Day

Count - Admin User Account Changes

	112 (100%)
WMI - Windows 2003 DC	112

Admin User Account Changes Events

Bound to: Count - Admin Us...	1 (0.89%)
User Account Changed	47
User Account Enabled	24
User Account Created	20
User Account password set	20
User Account Deleted	1

Event Subtype

Bound to: Admin User Accou...	1 (100%)
warning	1

Source User

Bound to: Event Subtype	1 (100%)
Administrator	1

Source IPs

Bound to: Source User	1 (100%)
192.168.1.55	1

Event Distribution

Bound to: Source IPs 1 (100%)

Interval: 1 Auto

Events

Bound to: Event Distribution

Last Time	Source User	Rule Message	Host	Domain	Application	Source MAC	Event Count	Event Subtype	Source Networ...	Source Interf...	Severity	First Time	Device Name
02/27/2011 00::	Administrator	User Account De	lanoc-win2k3-db	demo	security	00:00:00:00:00	1	warning			🟩 25	02/27/2011 00::	WMI - Windows 2003 DC

Retrieve ELM Archive:

Find text:

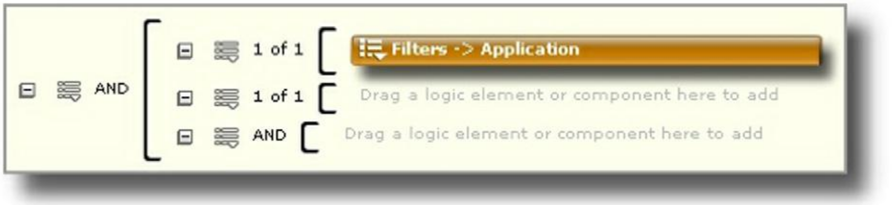
```
192.168.1.55||security||1347261||Security||630||52||1298767024||8||LANOC-WIN2K3-DB||||||7||macaisak||DEMO||%{S-1-5-21-2109995213-2822031779-1007051224-113207}||Administrator||DEMO||(0x0,0x15260C1F)||-||%08
```

FIGURE 9.9

An SIEM Dashboard Showing Administrative Account Changes.

by a specific event type—in this example, administrative account change activities that correspond with NERC compliance requirements.

A defining function of an SIEM is to correlate events to find larger incidents (see Chapter 8, “Exception, Anomaly, and Threat Detection”). This includes the ability to both define correlation rules, as well as present the results via a dashboard. Figure 9.10 shows a graphical event correlation editor that allows the logical conditions (such as “if A and B then C”), while Figure 9.11 shows the result of an incident query: the selected incident (an HTTP Command and Control Spambot) being derived from four discrete events.

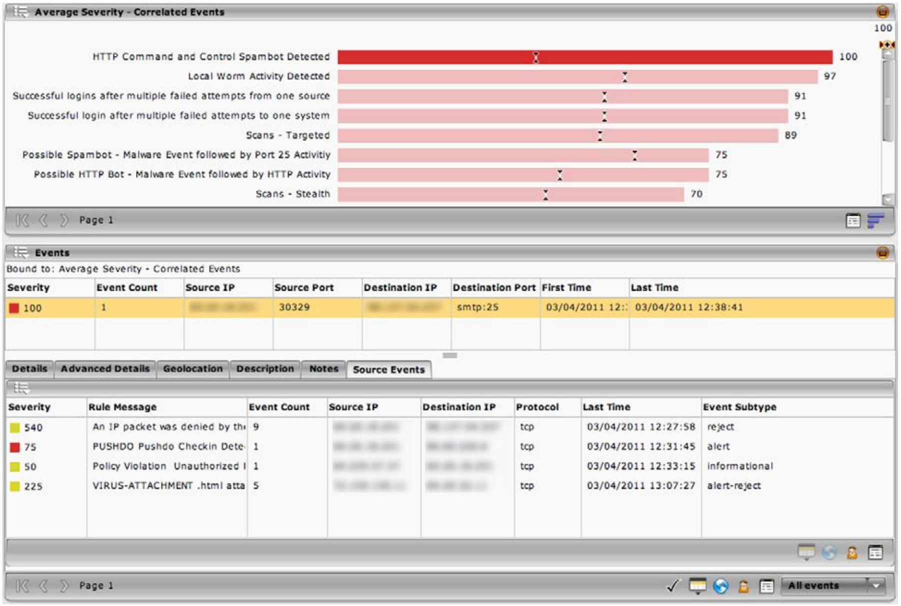


Drag a logic element or component here to add

Drag a logic element or component here to add

FIGURE 9.10

An Example of a Graphical Interface for Creating Event Correlation Rules.


Average Severity - Correlated Events

100
HTTP Command and Control Spambot Detected 100
Local Worm Activity Detected 97
Successful logins after multiple failed attempts from one source 91
Successful login after multiple failed attempts to one system 91
Scans - Targeted 89
Possible Spambot - Malware Event followed by Port 25 Activitiy 75
Possible HTTP Bot - Malware Event followed by HTTP Activity 75
Scans - Stealth 70
Page 1

Events

Bound to: Average Severity - Correlated Events

Severity	Event Count	Source IP	Source Port	Destination IP	Destination Port	First Time	Last Time
🟥 100	1	[redacted]	30329	[redacted]	smtp:25	03/04/2011 12::	03/04/2011 12:38:41

Details | Advanced Details | Geolocation | Description | Notes | Source Events

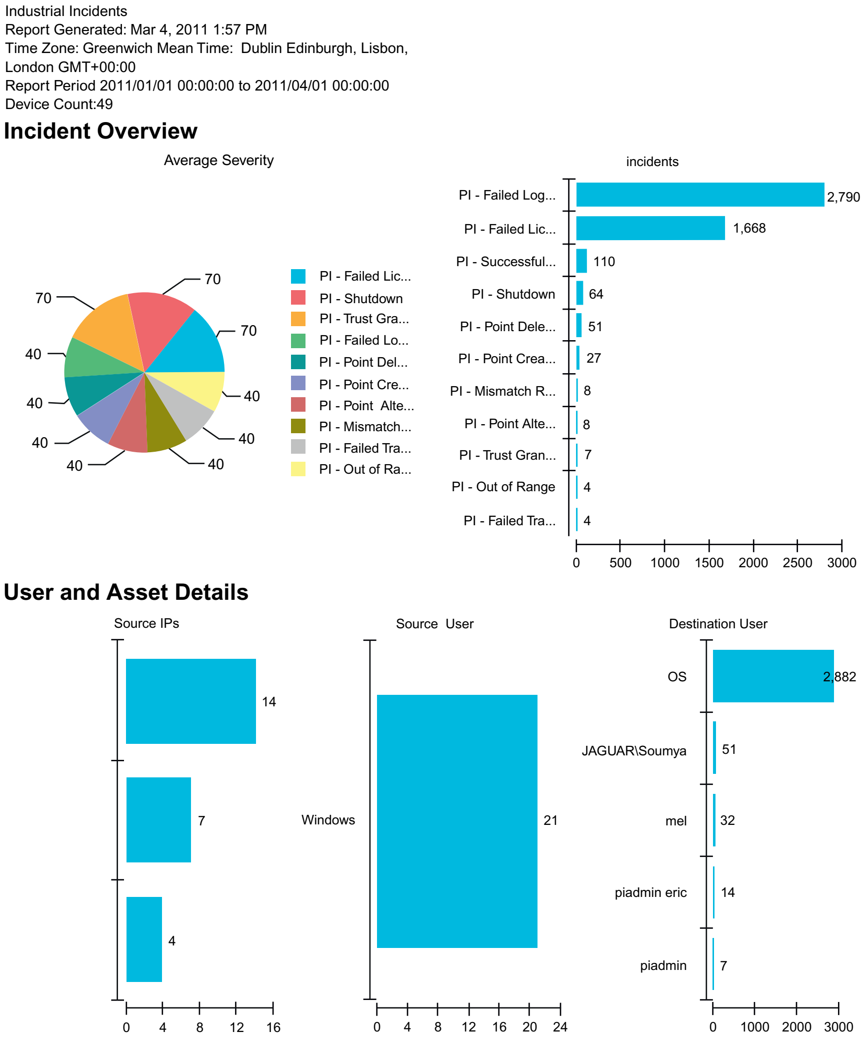
Severity	Rule Message	Event Count	Source IP	Destination IP	Protocol	Last Time	Event Subtype
🟨 540	An IP packet was denied by th	9	[redacted]	[redacted]	tcp	03/04/2011 12:27:58	reject
🟥 75	PUSHDO Pushdo Checkin Dete	1	[redacted]	[redacted]	tcp	03/04/2011 12:31:45	alert
🟨 50	Policy Violation Unauthorized I	1	[redacted]	[redacted]	tcp	03/04/2011 12:33:15	informational
🟨 225	VIRUS-ATTACHMENT .html atta	5	[redacted]	[redacted]	tcp	03/04/2011 13:07:27	alert-reject

FIGURE 9.11

An SIEM Dashboard a Correlated Event and Its Source Events.

Reports

Reports select, organize, and format all relevant data from the enriched logs and events into a single document. Reports provide a useful means to present almost any data set: from a summary of high-level incidents for executives, to precise and comprehensive documentation that provides minute details for internal auditing or for compliance. An example of a report generated by an SIEM is shown in Figure 9.12, which provides a quick summary of PI authentication failures and point change activity.



Industrial Incidents
Report Generated: Mar 4, 2011 1:57 PM
Time Zone: Greenwich Mean Time: Dublin Edinburgh, Lisbon, London GMT+00:00
Report Period 2011/01/01 00:00:00 to 2011/04/01 00:00:00
Device Count:49

Incident Overview

User and Asset Details

FIGURE 9.12

An SIEM Report Showing Industrial Activities.

Alerts

Alerts are active responses to observed conditions within the SIEM. An alert can be a visual notification in a console or dashboard, a direct communications (e-mail, page, text message, etc.) to a security administrator, or even the execution of a custom script. Common alert mechanisms used by commercial SIEMs include the following:

- Visual indicators (e.g., red, orange, yellow, green)
- Direct notification to a user or group of users
- Generation and delivery of a specific report(s) to a user or group of users
- Internal logging of alert activity for audit control
- Execution of a custom script or other external control
- Generation of a ticket in a compatible help desk or incident management system

Several compliance regulations, including NERC CIP, CFATS, and NRC RG 5.71, require that incidents are appropriately communicated to proper authorities inside and/or outside of the organization. By creating a useable variable or data dictionary with appropriate contacts within the SIEM, the alerting mechanism of an SIEM can facilitate this process by automatically generating appropriate reports and delivering them to key personnel.

Incident Investigation and Response

SIEM and Log Management systems are also useful for incident response, because the structure and normalization of the data allows an incident response team to drill into a specific event to find additional details (often down to the source log file contents and/or captured network packets), and to pivot on specific data fields to find other related activities. For example, if there is an incident that requires investigation and response, it can be examined, and relevant details such as the username, IP address, etc. can be quickly determined. The SIEM can then be queried to determine what other events are associated with the user, IP, etc.

In some cases the SIEM may support active response capabilities, including the following:

- Allowing direct control over switch or router interfaces via SNMP, to disable network interfaces
- Executing scripts to interact with devices within the network infrastructure, to re-route traffic, isolate users, etc.
- Execute scripts to interact with perimeter security devices (e.g., firewalls) to block subsequent traffic that has been discovered to be malicious
- Execute scripts to interact with directory or IAM systems to alter or disable a user account in response to observed malicious behavior

These responses may be supported manually or automatically, or both.

CAUTION

While automated response capabilities can improve efficiencies, they should be limited to noncritical enclaves and/or to enclave perimeters, and all automated responses should be carefully considered and tested prior to implementation. A false positive could trigger such a response and cause the failure of an industrial operation, with potentially serious consequences.

LOG STORAGE AND RETENTION

The end result of security monitoring, log collection, and enrichment is a large quantity of data in the form of log files, which must be stored for audit and compliance purposes (in the cases where direct monitoring is used in lieu of log collection, the monitoring device will still produce logs, which must still be retained). This represents a few challenges, including how to ensure the integrity of the stored files (a common requirement for compliance), how and where to store these files, and how they can be kept readily available for analysis.

Nonrepudiation

Nonrepudiation refers to the process of ensuring that a log file has not been tampered with, so that the original raw log file can be presented as evidence, without question of authenticity, within a court of law. This can be achieved in several ways, including digitally signing log files upon collection as a checksum, utilizing protected storage media, or the use of third-party FIM systems.

A digital signature is typically provided in the form of a hash algorithm that is calculated against the log file at the time of collection. The result of this calculation provides a checksum against which the files can be verified to ensure they have not been tampered with: if the file is altered in any way, the hash will calculate a different value and the log file will fail the integrity check; if the checksum matches, the log is known to be in its original form.

The use of appropriate storage facilities can ensure nonrepudiation as well. For example, by using Write Once Read Many (WORM) drives, raw log records can be accessed but not altered, as the write capability of the drive prevents additional saves. Many managed storage area network (SAN) systems also provide varying levels of authentication, encryption, and other safeguards.

An FIM may already be in use as part of the overall security monitoring infrastructure, as described in the section "Assets." The FIM observes the log storage facility for any sign of changes or alterations, providing an added level of integrity validation.

Data Retention/Storage

The above security monitoring tools all require the collection and storage of security information. The amount of information that is typically required could easily surpass 170 GB over an 8-hour period for a medium-sized enterprise collecting information at approximately 20,000 EPS.[15]

Data retention refers to the amount of information that is stored long term, and can be measured in volume (the size of the total collected logs in bytes) and time (the number of months or years that logs are stored for). The length of time a log is retained is important, as this metric is often defined by compliance regulations—for example, NERC CIP requires that logs are retained for anywhere from 90 days to up to 3 years, depending upon the nature of the log.[16] By determining which logs are needed for compliance and for how long they must be kept, the amount of physical storage space that is required can be calculated. Factors that should be considered include the following:

- Identifying the quantity of inbound logs
- Determining the average log file size
- Determining the period of retention required for logs
- Determining the supported file compression ratios of the Log Management or SIEM platform being used

Table 9.3 illustrates how sustained log collection rates map to total log storage requirements over a retention period of 7 years, resulting in a few terabytes of storage up to hundreds of terabytes or even petabytes of storage.

Depending upon the nature of the organization, there may be a requirement to retain an audit trail for more than one standard or regulation, often with different retention requirements. As with NERC CIP, there may also be a change in the retention requirements depending upon the nature of the log, and whether an incident has occurred. All of this adds up to even greater, long-term storage requirements.

TIP

Because event rates can vary (especially during a security incident), make sure that the amount of available storage has sufficient headroom to accommodate spikes in event activity.

Data Availability

Data availability differs from retention, referring to the amount of data that is accessible for analysis. Also called "live" or "online" data, the total data availability determines how much information can be analyzed concurrently—again, in either volume (bytes and/or total number of events) or time. Data retention affects the ability of an SIEM to detect "low and slow" attacks (attacks that purposefully occur over a long time in order to evade detection), as well as to perform trend analysis and anomaly detection (which by definition requires a series of data over time; see Chapter 8, "Exception, Anomaly, and Threat Detection").

Table 9.3 Log Storage Requirements Over Time

Logs per Second	Logs per Day (in Billions)	Logs per Year (in Billions)	Average Bytes per Event	Retention Period in Years	Raw Log Size (TB)	Compressed Bytes (TB) 5:1	Compressed Bytes (TB) 10:1
100,000	8.64	3,154	508	7	10,199	2,040	1020
50,000	4.32	1,577	508	7	5,100	1,020	510
25,000	2.16	788	508	7	2,550	510	255
10,000	0.86	315	508	7	1,020	204	102
5,000	0.43	158	508	7	510	102	51
1,000	0.09	32	508	7	102	21	11
500	0.04	16	508	7	51	11	6

TIP

In order to meet compliance standards, it may be necessary to produce a list of all network flows within an enclave that originated from outside of that enclave, for the past 3 years. For this query to be successful, 3 years of network flow data needs to be available to the SIEM at once. If the SIEM's data availability is insufficient (for example, it can only keep 1 year of data active), there is a work-around: by archiving older data sets, the information can be stored in volumes consistent with the SIEM's data availability. By querying the active data set, a partial result is obtained. By then restoring the next-previous backup or archive, two additional queries can be run, producing multiple partial result sets of 1 year each. These results can then be combined to obtain the required 3-year report. Note, however, that this requires extra effort on the part of the analyst. In addition, on some legacy SIEMs the archive/retrieval process may interfere with or interrupt the collection of new logs until the process is complete.

Unlike data retention, which is bound by the available volume of data storage (disk drive space), data availability is dependent upon the structured data that is used by the SIEM for analysis. Depending upon the nature of the data store, the total data availability of the system may be limited to a number of days, months, or years. Typically, databases are limited by one or more of the following:

- The total number of columns (indices or fields)
- The total number of rows (discreet records or events)
- The rate at which new information is inserted (i.e., collection rate)
- The rate at which query results are required (i.e., retrieval rates)

Depending upon the business and security drivers behind information security monitoring, it may be necessary to segment or distribute monitoring and analysis into zones to meet performance requirements. Some factors to consider when calculating the necessary data availability include the following:

- The total length of time over which data analysis may be required by compliance standards
- The estimated quantity of logs that may be collected in that time based on event estimates
- The incident response requirements of the organization: certain military or other critical installations may require rapid-response initiatives that necessitate fast data retrieval
- The desired granularity of the information that is kept available for analysis (i.e., are there many vs. few indices)

SUMMARY

With enclave security measures in place, a larger picture of security-related activity begins to form. By measuring these activities and analyzing them, exceptions from the established security policies can be detected. In addition, anomalous activities can be identified so that they may be further investigated.

This requires well-defined policies and also requires that those policies are configured within an appropriate information analysis tool. Just as with perimeter defenses to the enclave, carefully built variables defining allowed assets, users, applications, and behaviors can be used to aid in detection of security risks and threats. If these lists can be determined dynamically, in response to observed activity within the network, the "whitelisting" of known-good policies, becomes "smart-listing"—which can help strengthen perimeter defenses through dynamic firewall configuration or IPS rule creation.

As various threat detection techniques are used together, the event information can be further analyzed by event correlation systems to find larger patterns that are more indicative of serious threats or incidents. Widely used in IT network security, event correlation is beginning to "cross the divide" into OT networks, at the heels of Stuxnet and other sophisticated threats that attempt to compromise industrial network systems via attached IT networks and services.

Everything—measured metrics, baseline analysis, and whitelists—all rely on a rich base of relevant security information. Where does this security information come from? The networks, assets, hosts, applications, protocols, users, and everything else that is logged or monitored contributes to the necessary base of data required to achieve "situational awareness" and effectively secure an industrial network.

ENDNOTES

1. J.M. Butler. Benchmarking Security Information Event Management (SIEM). The SANS Institute Analytics Program, February, 2009.
2. Ibid.
3. Ibid.
4. Ibid.
5. Microsoft. Windows Management Instrumentation. <http://msdn.microsoft.com/en-us/library/aa394582(v=VS.85).aspx>, January 6, 2011 (cited: March 3, 2011).
6. Ibid.
7. National Institute of Standards and Technology, Special Publication 800-53 Revision 3. Recommended Security Controls for Federal Information Systems and Organizations, August, 2009.
8. Ibid.
9. sFlow.org. Traffic Monitoring using sFlow. <http://www.sflow.org/sFlowOverview.pdf>, 2003 (cited: March 3, 2011).
10. B. Singer, Kenexis Security Corporation, in: D. Peterson (Ed.), Proceedings of the SCADA Security Scientific Symposium, 2: Correlating Risk Events and Process Trends to Improve Reliability, Digital Bond Press, 2010.
11. Securonix, Inc., Securonix Indentity Matcher: Overview. <http://www.securonix.com/identity.htm>, 2003 (cited: March 3, 2011).
12. A. Chuvakin, Content Aware SIEM. <http://www.sans.org/security-resources/idfaq/vlan.php> February, 2000 (cited: January 19, 2011).

13. M. Nicolett, K.M. Kavanagh, Magic quadrant for security information and event management, Gartner Document ID Number: G00176034, May 13, 2010.
14. Ibid.
15. J.M. Butler, Benchmarking Security Information Event Management (SIEM). The SANS Institute Analytics Program, February, 2009.
16. North American Electric Reliability Corporation. NERC CIP Reliability Standards, version 4. <http://www.nerc.com/page.php?cid=2|20> February 3, 2011 (cited: March 3, 2011).

This page intentionally left blank

CHAPTER

Standards and Regulations

10

INFORMATION IN THIS CHAPTER:

- Common Standards and Regulations
- Mapping Industrial Network Security to Compliance
- Mapping Compliance Controls to Network Security Functions
- Common Criteria and FIPS Standards

There are hundreds of cyber security standards and regulations imposed by governments and industry, which provide everything from "best practices" recommendations to hard requirements that are enforced through penalties and fines. Common standards include the North American Electric Reliability Corporation's (NERC's) Critical Infrastructure Protection (CIP) Reliability Standards, the U.S. Department of Homeland Security's Chemical Facility Anti-Terrorism Standards (CFATS), the Regulated Security of Nuclear Facilities by the U.S. Nuclear Regulatory Commission (NRC), and general ICS security recommendations published by NIST in Special Publication 800-82. International standards include ISA-99 and ISO/IEC 27002:2005.

There are many specific compliance controls within these standards, as well as scores of additional compliance standards, which are not covered in this book. While efforts to maintain compliance with one or more of these regulations can be challenging and complex enough to fill an entire book dedicated to that topic, these controls often map directly to security best practices. These practices often agree with each other to a certain degree, although there are subtle differences among the various standards and regulations that can prove valuable when securing an industrial network. By mapping common security functions to common compliance controls, the best of each can be implemented as required (by a regulating authority) or as desired (for the sole purpose of strengthening the security of the industrial system).

Finally, there are standards and regulations that do not apply to industrial networks at all, but rather to the products that might be utilized by an industrial network operator to help secure (see Chapter 7, "Establishing Secure Enclaves") and monitor (see Chapter 9, "Monitoring Enclaves") the network. Among these are the international Common Criteria standards, and various FIPS standards including the FIPS 140-2 Security Requirements for Cryptographic Modules.

COMMON STANDARDS AND REGULATIONS

As mentioned in Chapter 2, "About Industrial Networks," industrial networks are of interest to several national and international regulatory and standards organizations. In the United States and Canada, NERC is well known because of the NERC CIP reliability standards, which heavily regulate security within the North American bulk electric system. NERC operates independently under the umbrella of the Federal Energy Regulatory Commission (FERC), which regulates natural gas, oil, and electric transmission, as well as hydropower projects. The Department of Energy (DoE) and Department of Homeland Security (DHS) also produce several security recommendations and requirements, including the Chemical Facility Anti-Terrorism Standards (CFATS), the Federal Information Security Management Act (FISMA), and Homeland Security Presidential Directive Seven, which all refer back to several special publications of the National Institute of Standards and Technology (NIST), particularly SP 800-53 "Recommended Security Controls for Federal Information Systems and Organizations" and SP 800-82 "Guide to Industrial Control Systems (ICS) Security." The International Standard Association's standard for the Security for Industrial Automation and Control Systems (ISA-99), and the International Standards Organization (ISO) Standard ISO/IEC 27002:2005 provide security recommendations that are applicable to industrial control networks.

NERC CIP

It is hard to discuss Critical Infrastructure security without referring to the North American Electric Reliability Corporations' Critical Infrastructure Protection reliability standards (NERC CIP). Although NERC CIP standards are only enforceable upon North American bulk electric systems, the standards represented are technically sound and in alignment with other standards, and are presented in the spirit of improving the security and reliability of the electric industry.[1] Further, the critical infrastructures of the electric utilities—specifically the distributed control systems responsible for the generation of electricity and the stations, substations, and control facilities—utilize common industrial network assets and protocols, making the standards relevant to a wider base of industrial network operators.

NERC consists of nine separate configuration management controls:

- CIP-001-4—Sabotage Reporting. Requires that all disturbances or unusual occurrences, suspected or determined to be caused by sabotage, shall be reported to the appropriate systems, governmental agencies, and regulatory bodies.[2]
- CIP-002-4—Critical Cyber Asset Identification. Requires the identification and documentation of the Critical Cyber Assets associated with the Critical Assets that support the reliable operation of the Bulk Electric System. These Critical Assets are to be identified through the application of a risk-based assessment.[3]
- CIP-003-4—Security Management Controls. Requires that Responsible Entities have minimum security management controls in place to protect Critical Cyber Assets.[4]

- CIP-004-4—Personnel and Training. Requires that personnel having authorized cyber or authorized unescorted physical access to Critical Cyber Assets, including contractors and service vendors, have an appropriate level of personnel risk assessment, training, and security awareness.[5]
- CIP-005-4—Electronic Security Perimeter(s). Requires the identification and protection of the Electronic Security Perimeter(s) inside which all Critical Cyber Assets reside, as well as all access points on the perimeter.[6]
- CIP-006-4—Physical Security of Critical Cyber Assets. Ensures the implementation of a physical security program for the protection of Critical Cyber Assets.[7]
- CIP-007-4—Systems Security Management. Requires Responsible Entities to define methods, processes, and procedures for securing those systems determined to be Critical Cyber Assets, as well as the other (noncritical) Cyber Assets within the Electronic Security Perimeter(s).[8]
- CIP-008-4—Incident Reporting and Response Planning. Ensures the identification, classification, response, and reporting of Cyber Security Incidents related to Critical Cyber Assets.[9]
- CIP-009-4—Recovery Plans for Critical Cyber Assets. Ensures that recovery plan(s) are put in place for Critical Cyber Assets and that these plans follow established business continuity and disaster recovery techniques and practices.[10]

NOTE

The NERC CIP standards have been mapped to common security controls under the section "Mapping Industrial Network Security to Compliance."

CFATS

The Risk-Based Performance Standards (RBPS) for the Chemical Facilities Anti-Terrorism Standards (CFATS) outlines various controls for securing the cyber systems of chemical facilities. Specifically, RBPS Metric 8 ("Cyber") outlines controls for (1) security policies, (2) access control, (3) personnel security, (4) awareness and training, (5) monitoring and incident response, (6) disaster recovery and business continuity, (7) system development and acquisition, (8) configuration management, and (9) audits.

Controls of particular interest are Cyber Metric 8.2.1, which requires that system boundaries have been identified and secured using perimeter controls, which supports the enclave security model. Metric 8.2 includes perimeter defense, access control (including password management), the limiting of external connections, and "least-privilege" access rules.[11]

Metric 8.3 (Personnel Security) also requires that specific user access controls are established, primarily around the separation of duties, and the enforcement thereof by using unique user accounts, access control lists, and other measures.[12]

Metric 8.5 covers the specific security measures for the monitoring of asset security (primarily patch management and Anti-Malware), network activity, log collection and alerts, and incident response, whereas Metric 8.8 covers the ongoing assessment of the architecture, assets, and configurations to ensure that security controls remain in compliance.[13]

Of particular note are RBPS 6.10 (Cyber Security for Potentially Dangerous Chemicals), RBPS 7 (Sabotage), RBPS 14 (Specific Threats, Vulnerabilities, and Risks), and RBPS 15 (Reporting)—all of which include cyber security controls outside of the RBPS 8 recommendations for cyber security. RBPS 6.10 implicates ordering and shipping systems as specific targets for attack that should be protected according to RBPS 8.[14] RBPS 7 indicates that cyber systems are targets for sabotage and that the controls implemented "deter, detect, delay, and respond" to sabotage.[15] RBPS 14 requires that measures are in place to address specific threats, vulnerabilities, and risks, inferring a strong vulnerability assessment plan,[16] whereas RBPS 15 defines the requirements for the proper notification of incidents when they do occur.[17]

NOTE

The CFATS standards as defined by the Risk-Based Performance Standards (RBPS) have been mapped to common security controls under the section "Mapping Industrial Network Security to Compliance."

ISO/IEC 27002:2005

The ISO/IEC 27002:2005 Standard is an international standard published by the International Standards Organization (ISO), the International Electrotechnical Commission (IEC), and the American National Standards Institute (ANSI). Although ISO/IEC 27002:2005 provides less guidance for the specific protection of industrial networks, it is useful in that it maps directly to many additional national security standards in Australia and New Zealand, Brazil, Chile, Czech Republic, Denmark, Estonia, Japan, Lithuania, the Netherlands, Poland, Peru, South Africa, Spain, Sweden, Turkey, United Kingdom, Uruguay, Russia, and China.[18]

NOTE

ISO/IEC 27002:2005 follows the C-I-A information security model, prioritizing Confidentiality, Integrity, and Availability in that order. However, depending upon the criticality of the industrial network and relevant safety concerns, these priorities may differ. For example, in nuclear facilities, the reliability and safety of the plant are paramount, whereas the confidentiality of information is less critical.

As with NERC CIP and CFATS, ISO/IEC 27002:2005 focuses on risk assessment and security policies in addition to purely technical security controls. The technical controls that are discussed include asset management and configuration management controls, separation and security controls for network communications, specific host security controls regarding access control, and Anti-Malware protection. Of particular interest are a group of controls around security incident management—the first of the standards discussed in this book to specifically mention the anticipation of a security breach using anomaly detection. Specifically, ISO/IEC mentions "malfunctions or other anomalous system behavior may be an indicator of a security attack or actual security breach."[19]

NOTE

Excerpts from the ISO/IEC 27002:2005 Standard have been mapped to common security controls under the section "Mapping Industrial Network Security to Compliance."

NRC Regulation 5.71

NRC Regulation 5.71 (RG 5.71) provides security recommendations for complying with Title 10 of the Code of Federal Regulations (CFR) 73.54. It consists of an in-depth discussion of the general requirements of cyber security, to specific requirements of planning, establishing, and implementing a cyber security program. Specific to RG 5.71 is the use of a five-zone network separation model, with one-way communications being required between zones 0 and 1 (the most critical enclaves of the five zones). One-way communications gateways, such as data diodes, allow outbound communications while preventing any return communications, promising an ideal security measure for the transmission of information from a secure zone to an outside supervisory system.

Although many of the recommendations in RG 5.71 are general in nature, RG 5.71 also includes three appendices, which provide a well-defined security plan template as well as specific technical security and operational controls for each recommendation.[20]

NIST SP 800-82

The National Institute of Standards and Technology (NIST) has published a working draft of a "Guide to Industrial Control Systems (ICS) Security," which includes recommendations for Security, Management, Operational, and Technical controls in order to improve control system security. This NIST publication is currently still in draft form and represents recommendations, not hard regulations. However, the controls presented are comprehensive and map well to additional NIST recommendations, such as those provided in SP 800-53 ("Recommended Security Controls for Federal Information Systems and Organizations") and SP 800-92 ("Guide to Computer Security Log Management").[21]

MAPPING INDUSTRIAL NETWORK SECURITY TO COMPLIANCE

There are literally hundreds of security regulations and recommendations that are published globally; many are applicable to industrial networks; some are enforced, some not; some are regional; some are applicable to all industrial networks, while some (such as NERC CIP) apply to specific industries. Although most standards and regulations focus on a variety of general security measures (including physical security, security policy development and planning, training, etc.), each has specific controls and measures for cyber security.

TIP

Many enforced compliance regulations (e.g., NERC CIP) require that "**compensating controls**" be used where a requirement cannot be feasibly met. Using additional compliance standards as a guide, alternate "compensating controls" may be identified. Therefore, even if the compliance standard is not applicable to a particular organization, the recommendations made within may prove useful.

These cyber security measures often overlap, although there are differences—both subtle and strong—among them. Efforts to normalize all the available controls to a common "compliance taxonomy" are being led by organizations such as the Unified Compliance Framework (UCF), which has currently mapped close to 500 Authority Documents to a common framework consisting of thousands of individual controls.[22] The advantages of a common mapping are significant and include the following:

- Facilitating compliance efforts for organizations that are responsible for multiple sets of compliance controls. For example, a nuclear energy facility that must track industrial regulations such as NRC Title 10 CFR 73.54, NRC RG 5.71, and **NEI** 08/09 requirements, as well as business regulations such as Sarbanes Oxley (SOX). Understanding which specific controls are common among all regulations prevents the duplication of efforts and can significantly reduce the costs of collecting, maintaining, storing, and documenting the information necessary for compliance.
- Facilitating the implementation of specific security controls by providing a comprehensive list of controls that must be implemented across all relevant standards and regulations.

This chapter begins to map the security and compliance requirements for this purpose; however, owing to the extensive nature of most regulations, as well as the changing nature of specific compliance control documents, only a select sample of common controls has been included in this text.

CAUTION

Figures 10.1, 10.2, and 10.3 and the corresponding Tables 10.1, 10.2, and 10.3 show how various compliance controls apply to different areas of control system security. Although every attempt has been made to reference common and relevant controls, which provide insight into best security practices, the controls represented in this chapter are far from all-inclusive.

This text should not be used as a sole resource for any regulatory compliance effort. Always reference source compliance standards documents and/or contact the standards organization directly to ensure that all required compliance controls are fully understood in order to avoid possible penalties or fines.

Perimeter Security Controls

Figure 10.1 and Table 10.1 map specific security controls to those requirements of the NERC CIP, CFATS, ISO 27002, NRC RG 5.71, and NIST SP 800-82 (draft) standards that are most relevant to perimeter security (see the section "Securing Enclave Perimeters" in Chapter 7, "Establishing Secure Enclaves").

CAUTION

These mappings are only intended to provide a high-level awareness of how security and compliance interrelate. Although based upon the most recent publications of each relevant standard at the time of writing, they do not represent a comprehensive list of the requirements and recommendations for all controls. Specifically cited requirements are excerpts from the original standards documentation only and do not represent the full scope of the referenced standard. Recommendations are provided for the purposes of improving security; adherence to these recommendations does not guarantee compliance with any referenced standard. Always reference the most current publication of the original standards document(s) when planning regulatory compliance efforts.

Figures 10.1, 10.2, and 10.3 illustrate where specific compliance controls can be implemented within the network, whereas the corresponding Tables 10.1, 10.2, and 10.3 then outline the corresponding controls and provide recommendations on how to implement appropriate security measures.

Figure 10.1 and Table 10.1 focus specifically on perimeter security controls and how they can be implemented to support regulatory requirements.

Host Security Controls

Figure 10.2 and Table 10.2 map specific security controls to those requirements of the NERC CIP, CFATS, ISO 27002, NRC RG 5.71, and NIST SP 800-82 (draft) standards that are most relevant to host security (see the section "Securing Enclave Interiors" in Chapter 7, "Establishing Secure Enclaves").

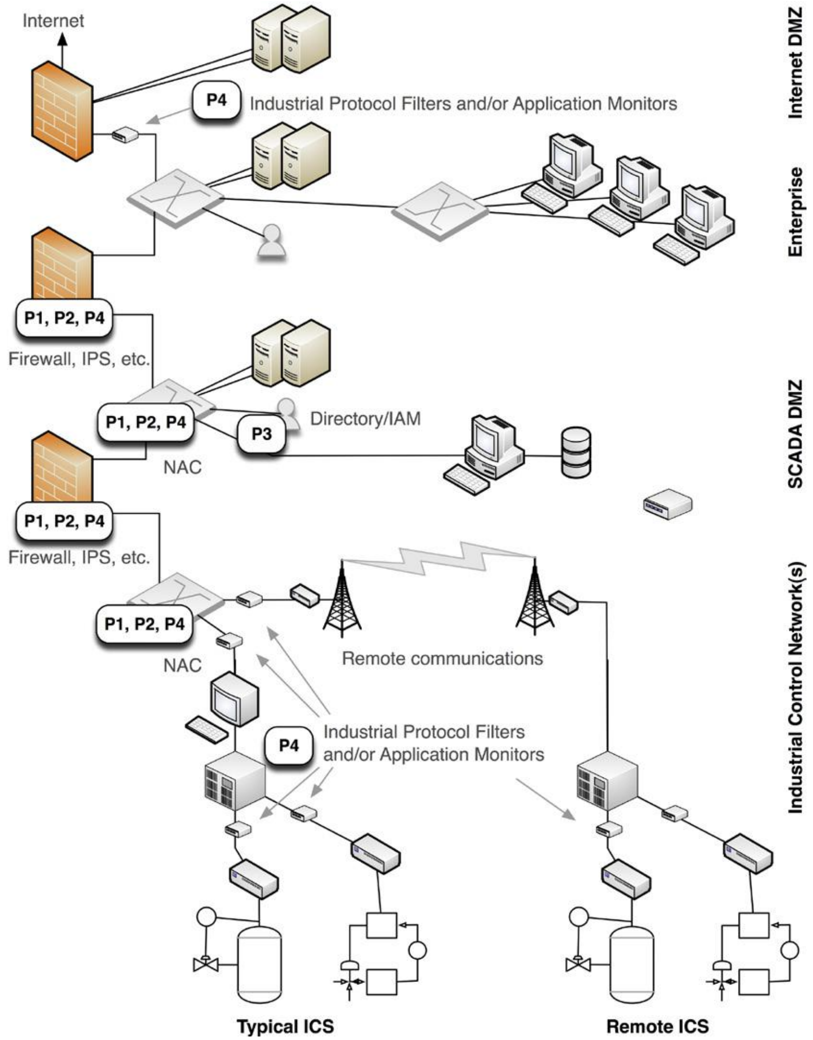


FIGURE 10.1

Compliance Requirements Mapped to Perimeter Security Controls.

CAUTION

These mappings are only intended to provide a high-level awareness of how security and compliance interrelate. Although based upon the most recent publications of each relevant standard at the time of writing, they do not represent a comprehensive list of the requirements and recommendations for all controls. Specifically cited requirements are excerpts from the original standards documentation only and do not represent the full scope of the referenced standard. Recommendations are provided for the purposes of improving security; adherence to these recommendations does not guarantee compliance with any referenced standard. Always reference the most current publication of the original standards document(s) when planning regulatory compliance efforts.

Figure 10.2 and Table 10.2 focus specifically on host security controls, and how they can be implemented to support regulatory requirements.

Table 10.1 Compliance Requirements Mapped to Perimeter Security Controls

Compliance Control	Recommendations
P1—Electronic Security Perimeter	
NERC CIP-005-4 R1, Electronic Security Perimeter CIP-005-4 R1 requires that the Electronic Security Perimeter (ESP), which should be established around "every Critical Cyber Asset," is identified and documented, including all access points to the ESP.[a]	Construct security perimeters at the edge of all enclaves, using multiple layered defenses (e.g., a firewall and an IPS, and/or Industrial Protocol Filters and Industrial Application Monitors). Implementing network flow monitoring at the perimeter will facilitate the detection and reporting of assets (by IP) on both sides of the perimeter and will also allow monitoring of the perimeter for communication violations, using an event or log management system.
CFATS RBPS Metric 8.2.1, Systems Boundaries The Risk-Based Performance Standard Metric 8.2.1 requires that an electronic perimeter be identified and also that appropriate security controls are implemented "to limit access across those boundaries."[b]	Consider naming devices that make up an ESP using a common nomenclature that identifies the ESP to which those devices belong, as well as the enclave(s) that it protects, in order to facilitate reporting, filtering, and other information management functions.
CFATS RBPS Metric 8.5.1, Cyber Security Controls The Risk-Based Performance Standard Metric 8.5.1 requires security controls to "prevent malicious code from exploiting critical assets," although there is no indication of whether network- or host-based protection is required.[c]	Implement an IPS to detect malware within inbound network traffic. Where an IPS is not feasible, utilize an IDS (or an IPS configured to operate in a "passive" or "IDS" mode) via a span port or network tap. The detection capability of the IDS is the same, without the risk of incurring latency or other connectivity issues.
ISO/IEC 27002:2005, Control 10.6.1, Network Controls Control 10.6.1 requires that networks be "adequately managed and controlled," to protect the systems and applications using the network. The control specifically calls out the protection of information in transit and offers guidance including the separation of duties, and the establishment of controls "to safeguard the confidentiality and integrity of data passing over public networks or over wireless networks, and to protect the connected systems and applications." While guidance for specific security controls is not provided, ISO/IEC 27002:2005, Control 10.6.1 does require that those controls utilize "appropriate logging and monitoring" of "security relevant actions."[d]	Network devices (switches, routers, etc.) should be configured to provide layer 3 separation of enclaves, with explicit access controls in place where possible. Network management and security monitoring systems should be functionally isolated as well, via layer 1 (physically isolated network connections), layer 2 (VLAN), or layer 3 (subnet) separation. Communication enforcement between enclaves (explicit source-to-destination rules) should be implemented at a minimum, and encryption should be used where interconnections occur across less secure networks. An IDS or IPS may also be used to detect and/or block intrusion attempts, whereas an Industrial Protocol Filter or Application Monitor can detect the attempted misuse of industrial protocols. All measures should be configured to produce verbose logs, for collection and management (see Chapter 9, "Monitoring Enclaves").

(*Continued*)

Table 10.1 (Continued)

Compliance Control	Recommendations
ISO/IEC 27002:2005, Control 11.4.5, Segregation of Networks Control 11.4.5 supports the separation of functional groups into enclaves, defined as “groups of information services, users, and information,” going on to include the concept of separating out enclaves based upon the criticality of systems and assets. Guidance for how to perform the separation of enclaves includes separating the network into multiple network domains; installing a firewall between networks, utilizing virtual private networks to control access. Additional guidance includes separating networks using “network device functionality” such as layer 3 routing, layer 2 switching, and access control lists—as well as strong authentication and encryption.[e]	Establish each domain as a separate enclave and implement an electronic perimeter consisting of a firewall and/or IPS, at a minimum. While network separation using layer 2 (VLAN) or layer 3 (Network) controls is recommended, these methods are significantly less secure and should be supplemented with a strong electronic perimeter, if used. To separate Industrial Networks using layer 2 and 3 protocols (see Chapter 4, “Industrial Network Protocols”), implement an Industrial Protocol Filter, or use an Application Monitor capable of operating as an Industrial Protocol Filter.
ISO/IEC 27002:2005, Control 11.4.6, Network Connection Control Control 11.4.6 refers specifically to shared network environments; that is, those networks that cross organizational or functional boundaries, such as a business intelligence workstation for the visualization of operational or production metrics to a business user located in a nonsecure network. Guidance includes strong network access controls, including date and time considerations for access control (i.e., the enforcement of “shifts”). Guidance also includes the “capability of users [to] be restricted through network gateways that filter traffic by means of predefined tables or rules.”[f] Although the method of restricting traffic is not specified, examples of restrictions are given, which include messaging applications, file transfers, interactive access, and application access—the latter two of which imply a control against the use of executable code or scripts in a network environment.	The reachability of networks should be controlled by electronic perimeter devices such as a firewall or IPS, whereas accessibility should be controlled via network access control, enforced within the network infrastructure. Applications and services should also be isolated within unique enclaves, and therefore be separated at the network layer, with explicitly defined network access control in addition to perimeter defenses.

ISO/IEC 27002:2005, Control 11.4.7, Network Routing Control Control 11.4.7 continues the trend of network-based separation recommended in Controls 11.4.5 and 11.4.6, this time focusing on network routing—specifically, to "ensure that computer connections and information flows do not breach the access control policy of the business applications."[g] The guidance refers to source and destination address checking, implying standard TCP/IP routing protocols for layer 3 separation of network traffic. However, the use of security gateways is also recommended to validate routing by checking source and destination addresses.	All routers should separate enclaves at layer 3, with explicitly defined Access Control Lists (ACLs) enforced where supported to control the specific source and destination addresses allowed to communicate between networks. Layer 2 network separation via Virtual Local Area Networks (VLANs) is less secure and, although VLANs may be used where needed, they should not be used as a secure means of network separation.
ISO/IEC 27002:2005, Control 11.6.2, Sensitive System Isolation Control 11.6.2 further supports the enclave model of functional separation and isolation, specifically requiring that critical systems should be implemented on a dedicated computing environment, which is isolated from other systems.[h]	All sensitive systems should be isolated within secured enclaves (see Chapter 7, "Establishing Secure Enclaves").
NRC RG 5.71, Control B.1.4, Information Flow Enforcement RG 5.71 Control B.1.4 is written in the context of documentation, but concerns control over "the flow of information, in near-real time" within and between Critical Digital Assets (**CDAs**) in accordance with defense-in-depth security practices.[i] This includes documenting those information flows that are allowed (basically, establishing a perimeter security policy as described in Chapter 7, "Establishing Secure Enclaves") and monitoring both access and information flows.	Information flow between systems can be controlled at the network level via an IPS or firewall, which can also "deter, detect, prevent, and respond" to unauthorized communication flows. One-way communications can be enforced via a carefully configured firewall with explicitly defined "deny all" rules in one direction, or via dedicated unidirectional network gateways or data diodes.
Information flows are required to be controlled using "domain-type enforcement" and monitored for indications of malicious communications attempts.[j] NRC RG 5.71 Control B.1.4 presents a strict perimeter security requirement for critical zones, requiring the implementation of one-way data flows from the highest-security enclaves to less secure enclaves.	Completing the feedback loop between monitoring, analysis, and defense provides dynamic control; this can be automated using more advanced SIEM or Log Management tools in response to detected threats, by appropriately configuring perimeter defenses (firewalls, NAC, IPS) in response to the threat.

(*Continued*)

Table 10.1 (Continued)

Compliance Control	Recommendations
NRC RG 5.71, Control B.3.4, Denial-of-Service Protection RG 5.71 Control B.3.4 is written in the more general context of protecting CDAs against denial-of-service attacks. The control specifically requires both network-based protection ("· · · restrict[ing] the ability of users to launch denial-of-service attacks against other CDAs or networks") and host-based security considerations ("· · · configuring CDAs to manage excess capacity, bandwidth, or other redundancy to limit the effects of information-flooding and saturation types of denial-of-service attacks.").[k]	To protect against denial of service (DoS) or information-flooding, limit the direct visibility to (i.e., make it more difficult to find) and the accessibility to (i.e., make it more difficult to connect) all outward facing services. Implement an IPS and/or firewall at the perimeter to block DoS behavior, as well as to block inbound scan attempts that could lead to DoS behavior. Make sure that any perimeter devices are capable of filtering unwanted traffic in excess of the maximum line rate of the network connection being protected, in order to prevent dropped traffic.
NIST SP 800-82, Network Architecture Control 5.3.2, Firewall between Corporate Network and Control Network NIST SP 800-82 Control 5.3.2 outlines how to deploy a firewall between corporate and control networks. The recommendation indicates that "ICS networks and corporate networks can be segregated to enhance cyber security using different architectures. By introducing a simple two-port firewall between the corporate and control networks · · · a significant security improvement can be achieved. Properly configured, a firewall significantly reduces the chance of a successful external attack on the control network."[l]	Configure firewall policies according to the recommendations in Chapter 7, "Establishing Secure Enclaves." Consider a layered defensive strategy consisting of one or more additional security measures in addition to a firewall—such as an IPS or Application Monitor. Consider implementing a DMZ (see NIST SP 800-82, Network Architecture Control 5.3.4) or paired firewalls (see NIST SP 800-82, Network Architecture Control 5.5) to terminate and reestablish connections between enclaves when using a common protocol (to "disjoint" the protocol).
NIST SP 800-82, Network Architecture Control 5.3.4, Firewall with DMZ between Corporate Network and Control Network NIST SP 800-82 Control 5.3.4 expands upon the recommendations of Control 5.3.2, recommending that a DMZ be used to provide access to certain systems—such as a data historian or business intelligence workstation—to both corporate and control networks, while maintaining security between the two.	When utilizing firewall DMZs to allow a device to communicate with multiple enclaves, always ensure that any and all devices within the DMZ have appropriate measures to prevent forwarding or relaying communications, in order to prevent the DMZ-connected devices from being used as a stepping-stone into other enclave(s). Consider a layered defensive strategy consisting of one or more additional security measures in addition to a firewall—such as an IPS or Application Monitor.

Specifically, NIST recommends that "· · · each DMZ holds one or more critical components, such as the data historian, the wireless access point, or remote and third party access systems. In effect, the use of a DMZ-capable firewall allows the creation of an intermediate network."[m] The establishment of a firewall DMZ is fairly straightforward and supported by most firewalls. Apart from the firewall interfaces that connect to the corporate network and the control network, additional interfaces are used to connect to those systems that are accessed by both.	Consider strengthening the DMZ further through the use of paired firewalls, as described in "NIST SP 800-82, Network Architecture Control 5.5." Use different policies on each perimeter to prevent "pass through" communications; i.e., "disjoint" the network connectivity.
NIST SP 800-82, Network Architecture Control 5.3.5, Paired Firewalls between Corporate Network and Control Network NIST SP 800-82 Control 5.3.5 recommends the establishment of a SCADA DMZ network. That is, an isolated network containing supervisory controls, historians, and other resources that require access by the corporate and control networks. Unlike the configuration described in Control 5.3.4, two firewalls are used in a pair: one between the corporate network and the DMZ network, and one between the DMZ network and the control network. The result is "· · · a DMZ-like network zone sometimes referred to as a Manufacturing Execution System (MES) layer," referred to as a SCADA DMZ in this book, where "the first firewall blocks arbitrary packets from proceeding to the control network or the shared historians. The second firewall can prevent unwanted traffic from a compromised server from entering the control network, and prevent control network traffic from impacting the shared servers."[n]	This is the preferred method of separation (shown in Figure 10.1), as it enforces explicitly defined and disjointed communication policies bi-directionally. Consider a layered defensive strategy consisting of one or more additional security measures in addition to a firewall—such as a SCADA IPS, Industrial Protocol Filter, or Application Monitor.
NIST SP 800-82, Network Architecture Control 5.5, General firewall policies for ICS Control 5.5 of NIST SP 800-82 covers basic recommendations for firewall configurations, including configuring firewalls with bidirectional	Always explicitly define the source and destination IP and port in all firewall policies, so that potentially vulnerable traffic will only be allowed between trusted assets.

(Continued)

Table 10.1 (Continued)

Compliance Control	Recommendations
`Deny All` rules and then explicitly enabling only “traffic [that is] absolutely required for business needs is every organization’s basic premise.” NIST’s recommendations in Control 5.5 include guidance on the “absolutely required” means, noting that “Many important protocols used in the industrial world, such as HTTP, FTP, OPC/DCOM, EtherNet/IP, and MODBUS/TCP, have significant security vulnerabilities.”[o] Using SQL as an example, SQL is often used for historian data access but is also a major inbound attack vector. Simply allowing SQL traffic, therefore, is not recommended. If SQL is allowed, it should be exclusively limited to specific IPs and ports. Alternatively, additional methods of historian access should be investigated.	For critical enclaves, further protection can be provided via deeper analysis of the allowed traffic. Implement a SCADA IPS, Industrial Protocol Filter, or Application Monitoring device to provide deep packet inspection of allowed traffic, to detect exploits against allowed protocols (see Chapter 7, “Establishing Secure Enclaves”).
NIST SP 800-82, Security Controls 6.2.6.2, Intrusion Detection and Prevention NIST SP 800-82 is one of the few documented guidelines to specifically recommend using an IDS. Specifically NIST SP 800-82 Control 6.2.6.2 recommends using an IDS “to monitor events on a network, such as traffic patterns, or a system, such as log entries or file accesses, so that they can identify an intruder breaking into or attempting to break into a system.”[p] NIST also points out that an IDS can “ensure that unusual activity such as new open ports, unusual traffic patterns, or changes to critical operating system files is brought to the attention of the appropriate security personnel.”[q] Note that Control 6.2.6.2 specifically calls for IDS, and not IPS, and references the IDS for monitoring use cases rather than active protection.	NIST SP 800-82 Security Control 6.2.6.2 highlights the benefits of intrusion detection vs. intrusion prevention, using the IDS to detect unusual activity (i.e., anomaly detection) as well as traffic on “new open ports” (i.e., using the IDS to detect policy violations). By following the guidelines provided in Chapter 7, “Establishing Secure Enclaves,” any deviations from authorized activities should be easily identified, via the use of “whitelist” variables defining known good ports, protocols, users, assets, etc. With the proper policies in place, Intrusion Prevention Systems (IPS) can be used in place of IDS to provide active protection. However, any detection policy configured to block traffic should be carefully considered to ensure that there is no possibility of disrupting a critical process as the result of a false positive.

P2—Network and Perimeter Monitoring	
NERC CIP-005-4 R2, Electronic Access Control CIP-005-4 R2 is an example of a compliance control designed to ensure that operational or organizational processes are established, yet that can be facilitated by the proper implementation of security controls. In this case, CIP-005-4 R2 requires that the “organizational processes and technical and procedural mechanisms for control of electronic access at all electronic access points to the Electronic Security Perimeter(s)” be implemented and documented.[r]	Implement access policies at the perimeter according to Chapter 7, “Establishing Secure Enclaves.” The use of a centralized configuration management system to track and monitor these access policies as defined within the electronic security perimeter device(s) can facilitate the documentation requirements. In addition, monitoring of the ESP itself can provide further evidence that the ESP is in place and (assuming that ESP logs indicate successful and failed access attempts to the ESP) that the correct controls are in place.
CFATS RBPS Metric 8.5.2, Network Monitoring Metric 8.5.2 calls for network monitoring traffic to identify unauthorized access and to detect malicious code. This requirement suggests the use of an IDS, which can do both tasks. In addition, RBPS Metric 8.5.2 specifies that the network monitoring result in immediate alerts, and that logs of all alert activity be produced. Again, this suggests the use of an IDS or application-aware firewall, although these devices are not specified. Of particular interest in RBPS Metric 8.5.2 is an exception allowing “network monitoring [to] occur on-site or off-site. Where logging of cyber security events on their networks is not technically feasible (e.g., logging degrades system performance beyond acceptable operational limits).”[s] This could be interpreted as allowing the use of an external network monitoring device (such as an IDS or network probe) to be connected via a network tap or mirrored interface, to monitor control networks where an in-line device could interfere with network operations; or it could be interpreted as allowing the remote collection and analysis of logs.	Monitoring a network in near-real time for unauthorized access or the introduction of malware requires either an active network monitoring solution such as an IDS or IPS, or the centralized analysis of alerts and logs by a Log Management or SIEM solution. In the latter case, the Log Management or SIEM system must be able to collect, correlate, and analyze logs and alerts generated by perimeter security monitoring tools such as firewalls, IDS, IPS, etc. and to provide the necessary threat detection capability required to detect the malicious access.

(*Continued*)

Table 10.1 (Continued)

Compliance Control	Recommendations
P3—Network Access and Authentication	
NERC CIP-005-4 R3, Monitoring Electronic Access CIP-005-4 R3 requires that a process for monitoring and logging access be established and documented. This control specifically requires the need to monitor access to the ESP as well as nonroutable network access via dial-up (in Control R3.1), where feasible, and generate alerts when an unauthorized access attempt is detected (in Control R3.2).[t]	Utilize a security event management system (SIEM) to centrally collect and display security events from the electronic security perimeter device(s). (See Chapter 9, "Monitoring Enclaves.") Practice a combined process of automated monitoring (i.e., the use of a SIEM for automated analysis and threat detection) with manual review of the ESP using a real-time "Security Operations Center" dashboard—typically provided by the same monitoring tools. The SIEM will not only facilitate the analysis of logs, but will also assist with the required documentation of the monitoring process(es).
ISO/IEC 27002:2005, Control 11.4.1, Policy of use of Network Services Control 11.4.1 requires the use of "least privilege" access control, where a minimal set of privileges are provided to a username, based upon the role of the human operator, and what services he or she has been authorized to use. Guidance provided by ISO/IEC 27002:2005 Control 11.4.1 includes the identification of which networks are accessible, and the mapping of user access to networks and services based upon a mapping of user authority to specific enclaves.[u]	Implement Network Access Control (NAC), and a central authentication system, directory system, or Identity Access Management (IAM) system to manage users, roles, and privileges. Map user privileges to specific `allow` policies in perimeter firewalls and IPSs, and to network access control lists within the network infrastructure.
ISO/IEC 27002:2005, Control 11.4.2, User Authentication for External Connections Control 11.4.2 simply requires that "appropriate authentication methods" are used by remote users. Guidance suggests that appropriate methods include "a cryptographic based technique, hardware tokens, or a challenge/response protocol" such as what	Remote access should only be provided using secure remote access mechanisms such as a VPN. Once terminated locally, remote users should be further restricted via network access control, and preferably be isolated to a unique enclave that is separated from local networks via an electronic perimeter.

is commonly associated with VPN authentication or dial-back procedures when using remote dial-up connectivity via modems. Node authentication (authenticating to the connected host rather than the network access service) is specified as an alternative for the connection of user groups to a shared network or facility.[V]	In other words, remote access should never allow direct authentication to a cyber asset. For example, a SCADA application installed on a laptop or smartphone should not be able to communicate directly to control system devices, unless the connection is made via a separately established and authenticated VPN, which terminates into a monitored and secured enclave. In this example, where control system access is being remotely granted, the additional layers of authentication (terminating the VPN into an isolated enclave that then requires additional authentication) prevents direct access to other critical networks from remote users. If remote ports are enabled when in use and otherwise disabled as a policy, the state of these access devices should be monitored using the Simple Network Management Protocol (SNMP) or a similar mechanism, so that network and security analysts can account for instances where remote access is enabled.
ISO/IEC 27002:2005, Control 11.4.4, Remote Diagnostic and Configuration Port Protection Control 11.4.4 is actually a physical security control as well as a logical security control, mandating that the configuration ports of cyber assets be controlled to prevent unauthorized access. This compliance control highlights the ability to physically circumvent cyber security controls (by physically accessing a local communication port) and to logically circumvent physical security controls (by accessing configurations via the network where physical access is prevented).[W]	Logical access to configuration ports can be controlled through the use of strict access controls, requiring one or more authenticated connections, as discussed in response to "ISO/IEC 27002:2005, Control 11.4.2, User Authentication for External Connections." If the remote access port supports local access control, the interface can be protected further by only allowing inbound connections from these secure sources—in other words, the physical configuration port could be accessed locally, but it would not function; only connections established via an authenticated source would be allowed.
NRC RG 5.71, Control B.1.4, Information Flow Enforcement As stated above under P1—Electronic Security Perimeter, NRC RG 5.71 Control B.1.4 is written in the context of documentation, but concerns control over "the flow of information, in near-real time" within and between Critical Digital Assets (CDAs) in accordance with defense-in-depth security practices.[X] This includes an access	NRC RG 5.71 Control B.1.4 introduces the concept of authenticated flow control between assets. Where authentication is not supported at the protocol layer (e.g., in the case of certain industrial network protocols, as discussed in Chapter 4, "Industrial Network Protocols"), the authentication will need to be enforced using external controls, such as Network Access Control, VPN, Domain authentication, etc.

(*Continued*)

Table 10.1 (Continued)

Compliance Control	Recommendations
control mechanism in addition to establishing an electronic security perimeter, in order to capture and log all inbound communications for purposes of information flow enforcement. NRC RG 5.71 Control B.1.4 also requires that information flows are deeply inspected, and that even encrypted data be scrutinized via content checking controls.[y]	The requirement to inspect the contents of a flow encourages the use of an Application Monitor or Industrial Protocol filter capable of determining (and analyzing) the payload of the information flow. In the case of encrypted traffic, the traffic must either be inspected prior to encryption or after decryption. In the case of network-based encryption, this can be accomplished by monitoring just outside of the encrypted link (just before the traffic has been encrypted, or just after it has been decrypted). If it is host-based data encryption, or if the encrypted connection is not fully assessable, it may be necessary to implement a network-based SSL inspection product. These devices effectively perform a hardware-accelerated Man-in-the-Middle attack on the encrypted traffic to allow full content inspection with minimal impact to network performance.
NRC RG 5.71, Control B.1.15, Network Access Control NRC's regulatory Guideline 5.71 Control B.1.15 specifically recommends the use of Network Access Control techniques, including "MAC address locking, physical or electrical isolation, static tables, encryption, or monitoring."[z]	Network access control is supported on most modern Ethernet switches and/or routers. If it is not, dedicated network access control (NAC) devices may be used. For non-IP industrial control networks, access control can be provided by whitelisting known industrial behaviors based upon device IDs and industrial protocol function codes (see Chapter 4, "Industrial Network Protocols").
P4—Network and Perimeter Ports and Services	
NRC RG 5.71, Control B.1.16, "Open/Insecure" Protocol Restrictions Like NERC CIP-007-4 R2, NRC RG 5.71 Control B.1.16 requires that protocol use be limited and controlled on the network. However, Control B.1.16 acknowledges that many industrial protocols lack security controls, and therefore requires that additional precautions be taken when using these protocols, including mechanisms to prevent protocols from initiating commands across an enclave perimeter.[aa]	To secure against the malicious use of open or insecure protocols, all enclave perimeters should carefully control the protocols that are/are not allowed to communicate, so that insecure protocols are fully restricted to those areas where they are necessary. In addition, to prevent unauthorized exploitation of or misuse of open or insecure protocols where they are allowed, an Industrial Control System Firewall, SCADA IPS, or Industrial Protocol Filter should be used that is capable of full protocol, session, and application monitoring, so that any misuse of these protocols will be detected. For example, Industrial Protocol filter may be able to allow or deny industrial protocol traffic based upon the specific function codes contained within the protocol frame (see Chapter 4, "Industrial Network Protocols"), effectively preventing that protocol from initiating commands across an enclave boundary.

[a] *North American Reliability Corporation, Standard CIP-005-4—Cyber Security—Electronic Security* Perimeter(s). <*http://www.nerc.com/files/CIP-005-4*.pdf>, *February 3, 2011 (cited: March 3, 2011).*

[b] *Department of Homeland Security, Risk-Based Performance Standards Guidance; Chemical Facility Anti-Terrorism Standards (CFATS), May 2009.*

[c] *Ibid.*

[d] *International Standards Organization/International Electrotechnical Commission (ISO/IEC), INTERNATIONAL ISO/IEC STANDARD 27002:2005 (E), Information technology—Security Techniques—Code of Practice for Information Security Management, first edition 2005-06-15.*

[e] *Ibid.*

[f] *Ibid.*

[g] *Ibid.*

[h] *Ibid.*

[i] *U.S. Nuclear Regulatory Commission, Regulatory Guide 5.71 (New Regulatory Guide) Cyber Security Programs for Nuclear Facilities, January 2010.*

[j] *Ibid.*

[k] *Ibid.*

[l] *K. Stouffer, J. Falco, K. Scarfone, National Institute of Standards and Technology, Special Publication 800-82 (Final Public Draft), Guide to Industrial Control Systems (ICS) Security, Section 5.3.2, Firewall Between Corporate Network and Control Network, September 2008.*

[m] *Ibid.*

[n] *Ibid.*

[o] *Ibid.*

[p] *D. Peterson, Intrusion detection and cyber security monitoring of SCADA and DCS networks,* ISA <http:*//whitepapers.techrepublic.com.com/whitepaper.aspx?&docid=126355&promo=*100511>, 2004 *(cited: March 3, 2011).*

[q] *K. Stouffer, J. Falco, K. Scarfone, National Institute of Standards and Technology, Special Publication 800-82 (Final Public Draft), Guide to Industrial Control Systems (ICS) Security, Section 6.2.6.2 Intrusion Detection and Prevention, September 2008.*

[r] *North American Reliability Corporation, Standard CIP-005-4—Cyber Security—Electronic Security* Perimeter(s). <*http://www.nerc.com/files/CIP-005-4*.pdf>, *February 3, 2011 (cited: March 3, 2011).*

[s] *Department of Homeland Security, Risk-Based Performance Standards Guidance, Chemical Facility Anti-Terrorism Standards (CFATS), May 2009.*

[t] *North American Reliability Corporation, Standard CIP-005-4—Cyber Security—Electronic Security* Perimeter(s), <*http://www.nerc.com/files/CIP-005-4*.pdf>, *February 3, 2011 (cited: March 3, 2011).*

[u] *International Standards Organization/International Electrotechnical Commission (ISO/IEC), INTERNATIONAL ISO/IEC STANDARD 27002:2005 (E), Information Technology—Security Techniques—Code of Practice for Information Security Management, first edition 2005-06-15.*

[v] *Ibid.*

[w] *Ibid.*

[x] *U.S. Nuclear Regulatory Commission, Regulatory Guide 5.71 (New Regulatory Guide) Cyber Security Programs for Nuclear Facilities, January 2010.*

[y] *Ibid.*

[z] *Ibid.*

[aa] *Ibid.*

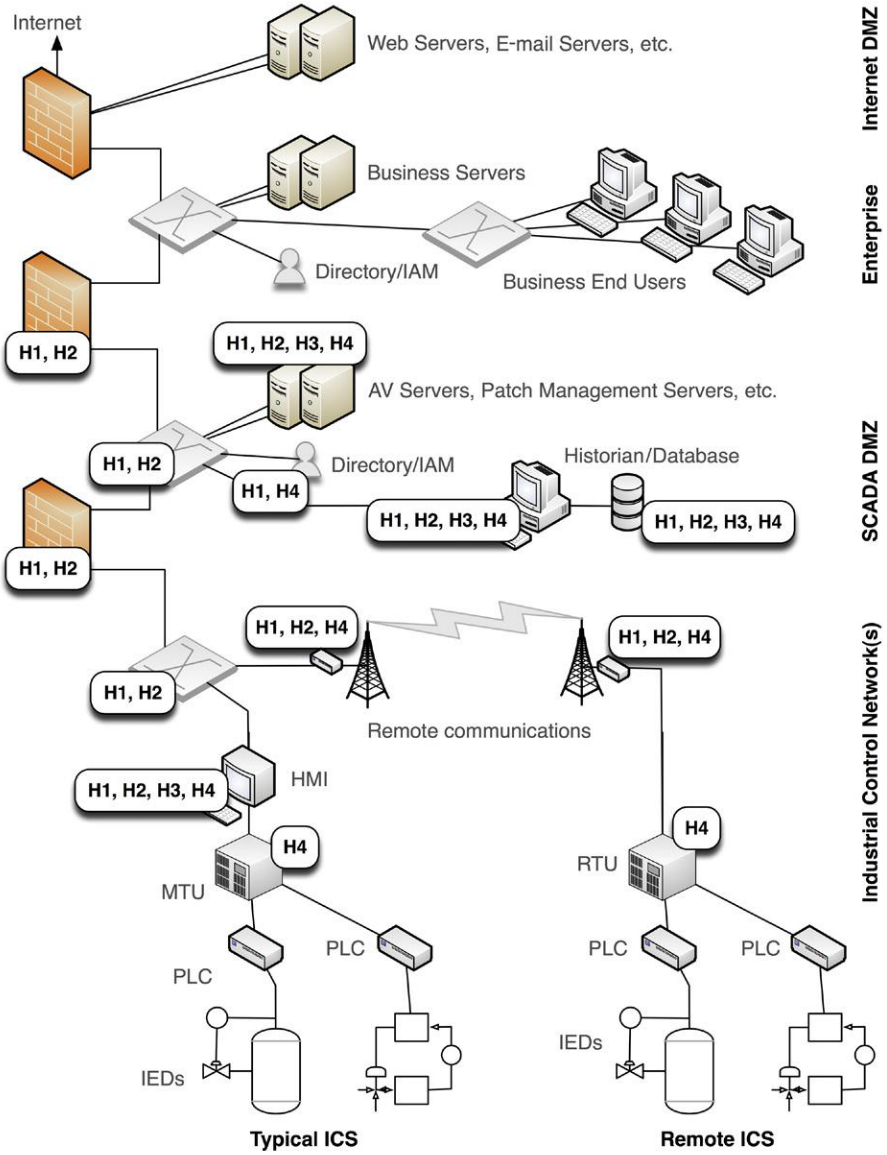


FIGURE 10.2

Compliance Requirements Mapped to Host Security Controls.

Table 10.2 Compliance Requirements Mapped to Host Security Controls

Compliance Control	Recommendations
H1—Asset Configurations	
NERC CIP-003-4 R6, Change Control and Configuration Management Like many NERC CIP requirements, CIP-003-4 R6 is written in the context of establishing and documenting a process. However, CIP-003-4 R6 does specifically require organizations to "implement supporting configuration management activities to identify, control and document all entity or vendor-related changes to hardware and software components of Critical Cyber Assets."[a] This could be accomplished manually, although configuration management tools would automate change identification, control, and documentation.	Configuration management and change control are important considerations for host security, as an unauthorized change from a "known good" configuration can negate host security controls. Security configurations should be compared against an authorized configuration file and monitored for changes. Although many configuration files can be monitored using host OS auditing or external FIM products, a commercial Configuration/Change Management system (or a commercial security monitoring tool with integrated CM features) may be justified in networks with a large number of assets.
NERC CIP-007-4 R3, Security Patch Management CIP-007-4 R3 specifically ties security patch management on individual cyber assets to the change controls required under CIP-003. Patch management requirements include tracking patches, evaluating them, and testing and implementing patches on all cyber assets within the ESP.[b]	Patch management should be performed in a separate, secure, and controlled environment; so that patches can be obtained free of risk (i.e., no open communications are established from live production networks to the Internet) and so that adequate testing and verification of patches can be performed prior to implementation (see Chapter 6, "Vulnerability and Risk Assessment").
CFATS RBPS Metric 8.8.2, Cyber Asset Identification CFATS Metric 8.8.2 requires that all "hardware, software, information, and services" have been identified and that all unnecessary items have been disabled, and requires that any remaining vulnerabilities be accommodated by compensating security controls.[c] This implies that in addition to ports and services on a particular asset, an entire asset or system may need to be removed if it is determined to be unnecessary.	Metric 8.8.2 requires a conglomeration of asset and vulnerability identification, all of which are met through a combination of network discovery and vulnerability assessment, as discussed in Chapter 6, "Vulnerability and Risk Assessment."
ISO/IEC 27002:2005, Control 10.1.2, Change Management The change management controls defined under ISO/IEC 27002:2005, Control 10.1.2, requires change management for all information processing facilities, and the provided guidance includes	Monitoring configuration files using host file system auditing (e.g., Linux auditd), and/or a commercial configuration management or change management system will identify change activities, as well as produce necessary reports and audit trails.

(*Continued*)

Table 10.2 (Continued)

Compliance Control	Recommendations
the identification of changes, the assessment of any potential impacts of those changes, and a system to generate detailed reports of all changes—to support an audit log of all change activity as well as to notify “relevant persons” when changes occur.[d]	By comparing new configurations against authorized configurations, configuration assurance is provided. This comparison may be performed manually using host tools (e.g., diff), and/or a commercial configuration management or change management system. File comparisons may also be a supported feature of certain SIEM or Log Management systems. Unauthorized changes may be an indication of malicious activity, as many attacks will attempt to alter network or security settings, add or change user credentials, or make other configuration changes as part of the attack process.
ISO/IEC 27002:2005, Control 12.6.1, Control of Technical Vulnerabilities ISO/IEC 27002:2005’s control of Technical Vulnerabilities is in-line with other vulnerability controls, requiring that vulnerabilities be identified, evaluated, and addressed and that “appropriate, timely action should be taken in response to the identification of potential technical vulnerabilities.”[e] Specific guidance offered includes maintaining a complete inventory of all assets, including details about installed software and version numbers/patches of the software.	The Control of Technical Vulnerabilities is similar to other vulnerability assessment and patching controls, with the additional guidance of providing “timely” information about vulnerabilities, as well as “timely” action when vulnerabilities are identified. This implies that vulnerability assessment should occur frequently, so that developing vulnerabilities can be quickly identified and remediated. By performing ongoing Vulnerability Assessments against a segregated test environment, frequent scans are possible without introducing risk to production systems.
NRC RG 5.71, Control B.5.3, Changes to File System and Operating System Permissions The NRC outlines clear asset configuration management requirements, including the file system and operating system permissions controls in NRC RG 5.71, Control B.5.3, which requires least-privilege access to “data, commands, files and account[s]” for both users and system services, as well as the documentation of any changes to access permissions or other security settings.[f]	This control requires that strong user/access policies are in place to prevent unnecessary privileged access, which will limit the impact of a compromised user account (or the actions of a disgruntled employee) to the least possible scale. Change management requirements are included to ensure that privilege escalations do not occur—which can be enforced using configuration file auditing and/or a commercial change management system. Validation is also possible by monitoring account changes as they are represented in system and application logs, and/or in application contents (by directly monitoring applications for account commands).

NIST SP 800-82, Network Architecture Control 6.2.4, Configuration Management NIST's recommendations for configuration control include restricting access configuration settings, and setting security controls to the "most restrictive mode," with specific guidance for "maintaining, monitoring, and documenting configuration control changes."[g] NIST SP 800-82 Control 6.2.4 also refers to the extensive configuration management guidance provided within the Configuration Management (CM) section of NIST SP 800-53, "Recommended Security Controls for Federal Information Systems and Organizations."	Monitor configuration files for indications of manipulation or change, via local file system auditing and/or a commercial CM system. This recommendation is almost identical to the corresponding NRC regulation 5.71 Control B.5.3, although it specifically references NIST Special Publication 800-53 for CM guidance, which identifies nine areas of configuration control: policies and procedures; baseline configurations; change control; security impact analysis; access restrictions for change; configuration settings; least functionality; IS component (asset) inventory; and the establishment of a configuration management plan.[h]
H2—Ports and Services	
NERC CIP-007-4 R2, Ports and Services CIP-007-4 R2 mandates that only required ports and services be enabled on a cyber asset (and that there is a documented process to ensure it).[i]	Unnecessary ports and services should be detected and disabled as part of earlier configuration and vulnerability assessments—and regular assessments can be used to further validate that only necessary services are in operation. However, malicious code will commonly open new ports or enable new services, requiring a continuous assessment of ports and services in order to truly ensure that only authorized services are in use. This can be accomplished by monitoring network activity in addition to host and perimeter configurations. Network flow analysis will clearly indicate which ports are actively in use, and can be used to generate an alarm when an unknown or unauthorized port is used (see Chapter 9, "Monitoring Enclaves").
NRC RG 5.71, Control B.5.1, Removal of Unnecessary Services and Programs Like NERC CIP-007-4 R2, the NRC's guidelines call for the removal of all unnecessary ports and services. However, the NRC goes further in defining "applications, utilities, system services, scripts, configuration files, databases, and other software and the appropriate configurations, including revisions or patch levels, for each of the computer systems associated with the CDAs."[j]	This control supports the concept of application whitelisting by requiring that known good applications are identified and documented, and that all other applications are removed. This can be achieved using AWL, which will also then prevent new software or services from being installed or executed in the future.

(*Continued*)

Table 10.2 (Continued)

Compliance Control	Recommendations
Apart from the added detail about the types of unnecessary items that must be removed or disabled, NRC RG 5.71 Control B.5.1 also specifies that patches be included in the assessment of "necessary" elements to prevent potential disruption of service or weakening of the security controls caused by the implementation of an unnecessary software patch.	"Whitelisting" can also be achieved by documenting known good applications and using this list as a variable that can be referenced by threat detection and network monitoring tools. For example, Vulnerability Assessment scanners (which will probe the assets directly) will detect most applications that are in use. Application traffic on the network can also be detected by network monitoring tools. Network connections can be mapped to applications based upon the TCP/UDP port; if there is traffic on these ports, the corresponding application is in use. For critical environments, application monitoring will provide a deeper look into application traffic. This will help detect applications running over nonstandard ports, applications that are masquerading as other applications, and even malware operating covertly inside of other applications. By comparing real-time application activity with the defined list of authorized applications, security administrators can be alerted of unauthorized application use.
H3—Anti-Malware	
NERC CIP-007-4 R4, Malicious Software Prevention NERC CIP-007-4 R4 technically requires the use of "antivirus software *and* other malicious software ("malware") prevention tools,"[k] indicating that at least two Anti-Malware controls should be implemented where technically feasible. However, while the wording of NERC CIP-007-4 R4 makes it difficult to determine which specific security controls should be implemented, its intentions are clear: to "detect, prevent, deter, and mitigate the introduction, exposure, and propagation of malware on all Cyber Assets within the Electronic Security Perimeter(s)."[l]	NERC CIP-007-4 R4 specifically requires the use of antivirus software. However, antivirus software requires regular patching and verification. Consider using Application Whitelisting instead, although a **technical feasibility exception** may be required as a result. Many AWL products are able to operate in fully isolated environments once properly configured, and require minimal patching (they only need to be updated when new software is applied to the host).

CFATS RBPS Metric 8.5.1, Cyber Security Controls The DHS' Risk-Based Performance Standards do not specify or recommend specific Anti-Malware controls, only that controls must be implemented to prevent malicious code from exploiting critical systems. However, RBPS Metric 8.5.1 goes somewhat further to require that appropriate security patches and updates are tested and applied "as soon as possible."[m] This is an important consideration, especially using AV systems that can only protect against known malware definitions, making AV patching an important necessity.	This control is similar to NERC CIP-007-4 R4, except that it does not specifically call out antivirus software, using the more general label of "malicious code" prevention. This allows the implementation of alternate controls such as Application Whitelisting. AWL will also facilitate the testing and patching of updates, as AWL profiles typically only require updating to accommodate other upgrades (i.e., unless you upgrade an authorized application, the AWL does not need to be regularly patched to protect against malware).
ISO/IEC 27002:2005, Control 10.4.1, Controls Against Malicious Code ISO/IEC 27002:2005's contribution to Anti-Malware methods can be found in Control 10.4.1, which requires that "Detection, prevention, and recovery controls to protect against malicious code" are implemented, along with "appropriate user awareness procedures."[n] The specific guidance of Control 10.4.1 again calls out awareness as a control against malware (which is true, considering that a significant amount of infections still occur through phishing attacks). Guidance also recommends that system access and change management controls be used to protect against malware.[o]	Utilize an antivirus system for strict adherence with this control, which requires "detection and repair." Application Whitelisting (AWL) is also an adequate control for preventing malware, although malicious code repair is not a function of most AWL systems. Note the inclusion of change management controls specific to Anti-Malware efforts, indicating that Anti-Malware systems should be closely monitored and maintained as part of formal assessment and patching. This is necessary to ensure that the Anti-Virus software is up to date, and that it is using the most current malware detection signatures.
NRC RG 5.71, Control B.5.2, Host Intrusion Detection System The NRC's host security recommendations extend to the use of Host Intrusion Detection Systems. Specifically, a HIDS should be configured to detect "dynamic file name patterns, system and user accounts, execution of unauthorized code, host utilization, and process permissions, to enable the system to detect cyber attacks." In other words, the HIDS should be configured to detect known malware and exploit patterns, and to alert security personnel when an event occurs.[p]	The requirement that the HIDS be configured to prevent the execution of "unauthorized code" implies that whitelisting behavior is preferred over blacklisting behavior. Although AWL is not specifically mentioned, traditional antivirus systems are blacklisting systems, where only code that is explicitly defined as bad is blocked. Whitelisting reverses this by defining what is good. This can be achieved through careful hardening of the host using a strong permissions-based operating system such as Linux, and/or through the use of an application whitelisting agent, which watches system files and low-level code execution (and in many cases memory-resident code as well).

(*Continued*)

Table 10.2 (Continued)

Compliance Control	Recommendations
NRC RG 5.71 Control B.5.2 also recommends specific logging methods to ensure event log integrity and requires that HIDS rule updates are performed to keep the HIDS in-line with known threat patterns.[q]	Interestingly, the requirement also states that the HIDS should not adversely impact operations,[r] which may be an acknowledgement to the potential application latency imposed by some AWL solutions. Proper testing and evaluation of whitelisting solutions should resolve any concerns. The requirement that the HIDS create logs supports the need for situational awareness (provided by holistic log review) and accountability (via an audit trail). Of additional interest is the inclusion of specific guidance for upgrades and patching, which suggests the use of traditional AV vs. AWL. With adequate host Anti-Malware in place (see Chapter 7, "Establishing Secure Enclaves"), RG 5.71 Control B.5.2 should be satisfied regardless of the type of HIDS that is implemented.
NIST SP 800-82, Network Architecture Control 6.2.6.1, Malicious Code Detection NIST SP 800-82 also identifies the need for host-based security and recommends the use of an antivirus, defined as a product that "evaluate[s] files on a computer's storage devices against an inventory of known malware signature files. If one of the files on a computer matches the profile of a known virus, the virus is removed through a disinfection process (e.g., quarantine, deletion), so it cannot infect other local files or communicate across a network to infect other files," going on to point out that "antivirus software can be deployed on workstations, servers, firewalls, and handheld devices"—an important consideration as mobile devices become more ubiquitous.[s]	Control 6.2.6.1 identifies antivirus products specifically, precluding AWL or other HIDS solutions. However, consider utilizing these additional measures in addition to antivirus in order to better secure critical hosts. This control specifically requires the protection of handheld devices—which is valid advice and an acknowledgement that mobile devices (such as smart phones) can be an inbound vehicle for malware.

H4—Authentication	
CFATS RBPS Metric 8.2.5, Password Management Password management controls, as defined by RBPS Metric 8.2.5, require that authentication methods are documented and enforced for all administrative and end user accounts, and that strong passwords are used (e.g., default passwords are not allowed).[t] Interestingly, RBPS Metric 8.2.5 allows for compensating controls to be implemented where changing a default password is "not technically feasible (e.g., a control system with a hard-coded password."[u]	Implement a centralized authentication system (Active Directory or a Commercial IAM) to track user authentication requirements. Monitor application contents (either via deep packet inspection via IPS or deep session inspection via an Application Monitor) for instances of weak passwords or known default passwords. Implement exception-based reporting using known good accounts as a whitelist of account behavior, to detect when legitimate but unknown or unauthorized authentications occur (e.g., a valid authentication against a hard-coded password, as used by Stuxnet).
CFATS RBPS Metric 8.3.2, Unique Accounts RBPS Metric 8.3.2 extends the provisions of Metric 8.2.5, which prohibit the use of default accounts, by requiring that unique accounts be used for all users and administrators, and prohibiting the sharing of accounts. The exception is "in instances where users function as a group (e.g., control system operators) and user identification and authentication is role based, then appropriate compensating security controls (e.g., physical controls) have been implemented."[v]	Usernames can be extracted from any monitoring system capable of examining authentications (i.e., packet—or session deep packet inspection, database or application monitoring, application log monitoring, IAM monitoring, etc.). The correlation of usernames and network flows can be used to identify where accounts are being shared across multiple physical consoles. Where multiple users are sharing a single physical console, alternate measures will need to be implemented, as there is no adequate cyber separation of user activity.
CFATS RBPS Metric 8.3.4, Access Control Lists RBPS Metric 8.3.4 further strengthens access control by requiring that an access control list be maintained, and by ensuring that administrative accounts are adjusted or deleted as appropriate when an administer leaves the organization or otherwise no longer requires access.[w]	Although access control lists are often thought of in terms of network switches and routers, host ACLs are an effective way to limit access to a particular asset. Validation of active accounts can be done by monitoring the authentication activity on the network and correlating that information against centralized account management systems such as Active Directory, or a commercial IAM.
ISO/IEC 27002:2005, Control 11.2.1, User Registrations ISO/IEC 27002:2005 requires the use of a formal "registration and de-registration" procedure to ensure the accuracy of all access privileges to information systems. Guidance includes the use of unique user IDs, and evaluation of user and system access privileges with the business or system owner, removing access privileges from users who no longer need them, checking for and removing duplicate accounts, and similar procedural controls.[x]	These functions are supported by most IAM systems, including the centralization of user accounts and the normalization of multiple accounts to a common user identity; the roles and privileges of that user; and the ability to centrally apply, ensure, or revoke privileges.

(*Continued*)

Table 10.2 (Continued)

Compliance Control	Recommendations
NRC RG 5.71, Control B.1.3, Access Enforcement NRC RG 5.71's Access Enforcement Controls (B.1.3) require that authorized access is only provided in accordance with established procedures. Specific security controls that are required under B.1.3 include the use of "dual authorization" for critical privileged functions and the creation of any privileged account.[y]	The NRC's guidance for access control extends beyond most authentication controls by adding controls that specifically require dual authorization for critical access, and by requiring that authentication methods should not interfere or adversely impact performance of the operational system being authenticated to. Although many authentication and access control recommendations can be met using a commercial IAM platform, the dual-authorization requirement should ideally use a separate set of credentials that are managed separately. In this way, a successful enumeration attack against the IAM itself would not compromise access to these critical systems.
NRC RG 5.71, Control B.4.2, User Identification and Authentication NRC RG 5.71 Control B.4.2 specifically requires the implementation of "identification and authentication technology to uniquely identify and authenticate individuals and processes acting on behalf of users interacting with CDA and ensuring that CDAs, security boundary devices, physical controls of the operating environment, and individuals interacting with CDAs, are uniquely identified and authenticated and that all processes acting on behalf of users are equally authenticated and identified; ensuring that the authentication technology employs strong multifactor authentication using protected processing levels."[z]	This control requires that users are uniquely identified and authenticated—a function of IAM systems where multiple accounts can be reconciled to a common human user identity. However, the requirement to track user activity (including applications that are authenticating on behalf of a user) can present challenges. For example, if a poorly written application uses account pooling (where many users authenticate to the application, but then the application authenticates to a backend system using a single account), the original user identity can be lost. This challenge can be resolved by correlating logs from all stages of authentication. However, to correctly log backend authentications, a database monitoring tool or custom software agent may be required. If the application in question is customizable, strengthening the backend authentications to include session details and user credentials is advisable.

NRC RG 5.71, Control B.4.3, Password Requirements NRC RG 5.71 Control B.4.3 extends the strength of Identification and Authentication controls of B.4.2 by requiring the use of strong passwords and secure password management. B.4.3 defines a strong password as having a "length and complexity commensurate with the required security," and requires that passwords be changed regularly. In addition, master passwords must be stored securely, and any authorization to change master passwords must be strictly controlled."[aa]	Strong password requirements may be implemented and maintained using a common authentication or IAM system, which will provide the necessary password strength controls as well as password storage and recovery controls. In addition to password management, active monitoring for password violations—including weak passwords or default passwords—is recommended as an additional checksum to ensure that only strong passwords are in use, even if there are misconfigurations or errors in password provisioning at the central IAM.
NIST SP 800-82, Network Architecture Control 6.3.2, Access Control NIST SP 800-83 refers access control requirements back to NIST SP 800-53's Access Control (AC) recommendations, which "specifies controls for managing information system accounts, including establishment, activating, modifying, reviewing, disabling, and removing accounts [and] cover access and flow enforcement issues such as separation of duties, least privilege, unsuccessful login attempts, system use notification, previous logon notification, concurrent session control, session lock, and session termination."[ab]	NIST SP 800-53 Access Control recommendations are thoroughly defined, consisting of 22 individual controls, including requirements to monitor established sessions to enforce time-outs, break inactive sessions, etc. to enforce postauthentication access control.[ac]

[a] *North American Reliability Corporation, Standard CIP-003-4—Cyber Security—Security Management* Controls. <http*://www.nerc.com/files/CIP-003-4.*pdf>, *February 3, 2011 (cited: March 3, 2011).*

[b] *North American Reliability Corporation, Standard CIP-007-4—Cyber Security—Systems Security* Management. <http*://www.nerc.com/files/CIP-007-4.*pdf>, *February 3, 2011 (cited: March 3, 2011).*

[c] *Department of Homeland Security, Risk-Based Performance Standards Guidance, Chemical Facility Anti-Terrorism Standards (CFATS), May 2009.*

[d] *International Standards Organization/International Electrotechnical Commission (ISO/IEC), INTERNATIONAL ISO/IEC STANDARD 27002:2005 (E), Information Technology—Security Techniques—Code of Practice for Information Security Management, first edition 2005-06-15.*

[e] *Ibid.*

(*Continued*)

[f] *U.S. Nuclear Regulatory Commission, Regulatory Guide 5.71 (New Regulatory Guide) Cyber Security Programs for Nuclear Facilities, January 2010.*

[g] *K. Stouffer, J. Falco, K. Scarfone, National Institute of Standards and Technology, Special Publication 800-82 (Final Public Draft), Guide to Industrial Control Systems (ICS) Security, Section 6.2.4 Configuration Management, September 2008.*

[h] *National Institute of Standards and Technology, Special Publication 800-53 Revision 3, Recommended Security Controls for Federal Information Systems and Organizations, August 2009.*

[i] *North American Reliability Corporation, Standard CIP-007-4—Cyber Security—Systems Security* Management. <http://www.nerc.com/files/CIP-007-4.pdf>, *February 3, 2011 (cited: March 3, 2011).*

[j] *U.S. Nuclear Regulatory Commission, Regulatory Guide 5.71 (New Regulatory Guide) Cyber Security Programs for Nuclear Facilities, January 2010.*

[k] *North American Reliability Corporation, Standard CIP-007-4—Cyber Security—Systems Security* Management. <http://www.nerc.com/files/CIP-007-4.pdf>, *February 3, 2011 (cited: March 3, 2011).*

[l] *Ibid.*

[m] *Department of Homeland Security, Risk-Based Performance Standards Guidance; Chemical Facility Anti-Terrorism Standards (CFATS), May 2009.*

[n] *International Standards Organization/International Electrotechnical Commission (ISO/IEC), INTERNATIONAL ISO/IEC STANDARD 27002:2005 (E), Information Technology—Security Techniques—Code of Practice for Information Security Management, first edition 2005-06-15.*

[o] *Ibid.*

[p] *U.S. Nuclear Regulatory Commission, Regulatory Guide 5.71 (New Regulatory Guide) Cyber Security Programs for Nuclear Facilities, January 2010.*

[q] *Ibid.*

[r] *Ibid.*

[s] *K. Stouffer, J. Falco, K. Scarfone, National Institute of Standards and Technology, Special Publication 800-82 (Final Public Draft), Guide to Industrial Control Systems (ICS) Security, Section 6.2.6.2 Malicious Code Detection, September 2008.*

[t] *Department of Homeland Security, Risk-Based Performance Standards Guidance; Chemical Facility Anti-Terrorism Standards (CFATS), May 2009.*

[u] *Ibid.*

[v] *Ibid.*

[w] *Ibid.*

[x] *International Standards Organization/International Electrotechnical Commission (ISO/IEC), INTERNATIONAL ISO/IEC STANDARD 27002:2005 (E), Information Technology—Security Techniques—Code of Practice for Information Security Management, first edition 2005-06-15.*

[y] *U.S. Nuclear Regulatory Commission, Regulatory Guide 5.71 (New Regulatory Guide) Cyber Security Programs for Nuclear Facilities, January 2010.*

[z] *Ibid.*

[aa] *Ibid.*

[ab] *K. Stouffer, J. Falco, K. Scarfone, National Institute of Standards and Technology, Special Publication 800-82 (Final Public Draft), Guide to Industrial Control Systems (ICS) Security, Section 6.3.2 Access Control, September 2008.*

[ac] *National Institute of Standards and Technology, Special Publication 800-53 Revision 3, Recommended Security Controls for Federal Information Systems and Organizations, August 2009.*

Security Monitoring Controls

Figure 10.3 and Table 10.3 map specific security controls to those requirements of the NERC CIP, CFATS, ISO 27002, NRC RG 5.71, and NIST SP 800-82 (draft) standards that are most relevant to security monitoring, log management, and situational awareness (see Chapter 9, "Monitoring Enclaves").

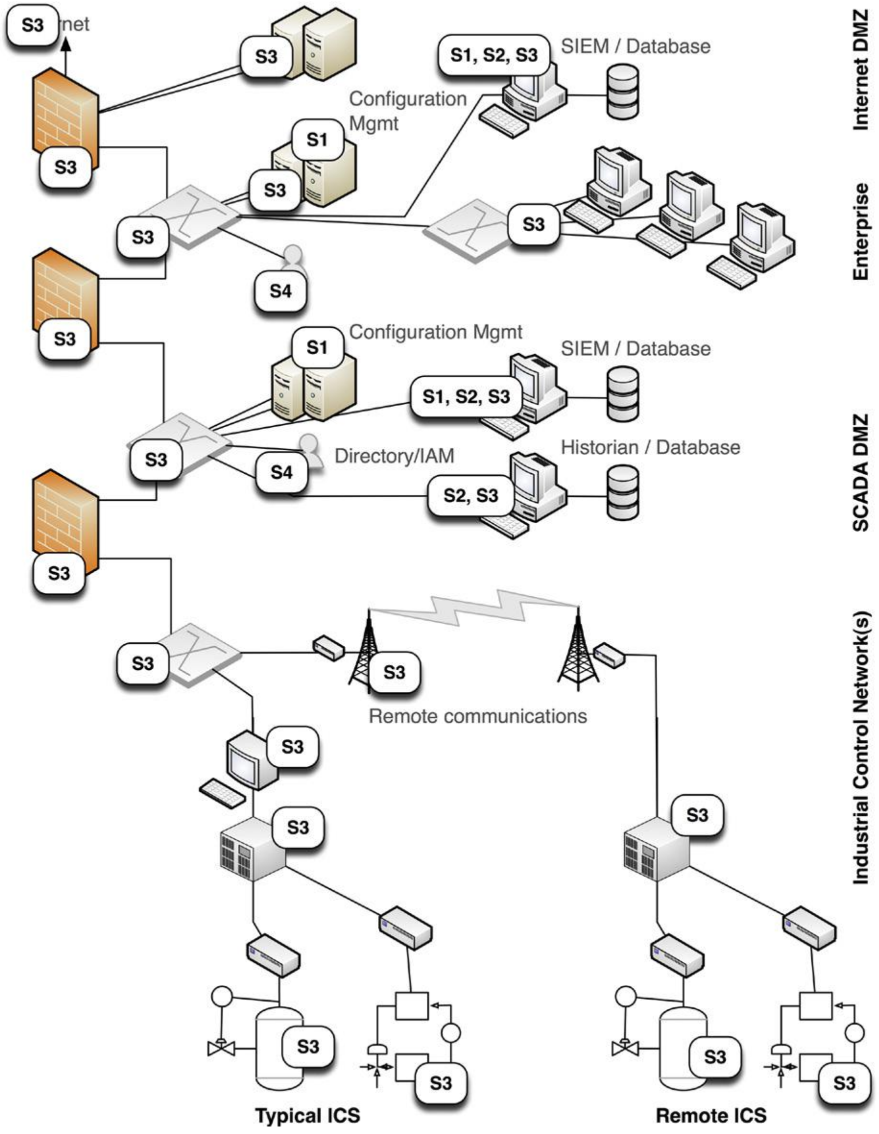


FIGURE 10.3

Compliance Requirements Mapped to Security Monitoring Controls.

Table 10.3 Compliance Requirements Mapped to Security Monitoring Controls

Compliance Control	Recommendations
S1—Asset Configurations	
NERC CIP-003-4 R6, Change Control and Configuration Management NERC CIP-003-4 R6 concerns configuration and change controls of cyber assets and requires the implementation of "supporting configuration management activities to identify, control, and document all entity or vendor-related changes to hardware and software . . . "[a]	To satisfy the requirement "to identify, control, and document all entity or vendor-related changes to hardware and software" requires that—whatever configuration management system is in place—adequate logs are produced to document any changes that may be made. Ideally, those logs should be centrally collected and managed for compliance auditing purposes.
NERC CIP-005-4 R4, Cyber Vulnerability Assessment NERC CIP-005-4 R4 requires the assessment, review, and documentation of vulnerabilities, including a review of ports and services in use, validity of access at perimeters, and identification of default accounts. Although the vulnerability assessment procedure required under CIP-005-4 R4 can be satisfied by an annual vulnerability assessment scan (either automated via a VA product or performed manually), many of these requirements can also be met—or at least facilitated—using continual network and security monitoring tools. "The Responsible Entity shall implement and document an electronic or manual process(es) for monitoring and logging access at access points to the Electronic Security Perimeter(s) twenty-four hours a day, seven days a week."[b]	Although CIP-005-4 R4 does not specifically mention the configurations of assets, a vulnerability assessment should be the basis of all asset configurations, as it will identify open ports and services, unnecessary services that are in use, and possible vulnerabilities that should be fixed via a patch or by the removal of the software. Outside of the vulnerability assessment process, security monitoring tools can be used to detect default passwords, weak passwords, account changes, violations of the electronic security perimeter, the presence of unauthorized ports and services, etc. Security monitoring should ideally be used in conjunction with a vulnerability assessment system, to correlate vulnerabilities identified by the VA tool with the activities being observed by the security monitoring tool(s).
NERC CIP-007-4 R3, Security Patch Management NERC CIP-007-4 R3 requires the use of security patch management, or "compensating controls" where patches are not or cannot be installed.[c]	Patch management is a challenge in industrial networks due to the requirements to obtain and test patches in a controlled environment prior to deployment, and because of the minimal maintenance windows that are available for production upgrades in most industrial systems.

	Therefore, the establishment of compensating measures—under the assumption that unpatched assets will continue to be present—is a sound practice to avoid noncompliance. These compensation measures could include the isolation of systems and least-privilege access controls to those systems. Establishing (and documenting) highly segmented and secured enclaves may be sufficient as a compensating measure (always verify compliance audit assumptions with a local, qualified compliance consultant). Regardless, security monitoring should be used to obtain a clear picture of all incidents or events surrounding the asset(s) in question, and to ensure the proper function of any compensating controls. The logs and events analyzed should also be retained for supporting documentation of the compensating controls.
NERC CIP-007-4 R8, Cyber Vulnerability Assessment NERC CIP-007-4 R8 is an example of how security monitoring can be used to validate specific security controls. CIP-007-4 R8 requires documentation of a vulnerability assessment, the vulnerabilities that may have been identified, and a plan to remediate the vulnerability, including the execution status of that action plan.”[d] By using a broader security monitoring solution to manage VA results (along with other relevant activities and events), the required VA documentation can be coupled with an audit trail of any changes made to remediate specific vulnerabilities.	Vulnerability assessments can be a useful aid to configuration assessment and management. Once configurations are established, tested, and locked down, there should no longer be any open vulnerabilities. In addition, central logging and monitoring software such as a Log Management system or SIEM will have assimilated the VA scan data, such that any new vulnerabilities, changes in software versions or patch levels, etc., will provide a clear indication that an asset has changed. This can be used to validate the results of configuration file monitoring or configuration management systems to ensure that only good configurations are in use.
NRC 5.71, Control A.4.1.3, Vulnerability Scans and Assessments NRC requires a periodic vulnerability scanning and assessment to validate all security controls. Unlike NERC CIP, NRC RG 5.71 controls specify the “periodic vulnerability scanning and assessments of the security controls, defensive architecture and of all CDAs to identify	NRC 5.71 specifically requires that vulnerability assessment scans be performed quarterly, and “when new vulnerabilities . . . are identified,”[f] which necessitates (potentially) very frequent scanning. In addition, NRC 5.71 requires the “promotion”[g] of interoperability among automation tools, validating the recommendation to use integrated event monitoring and VA solutions.

(*Continued*)

Table 10.3 (Continued)

Compliance Control	Recommendations
NRC requires a periodic vulnerability scanning and assessment to validate all security controls. Unlike NERC CIP, NRC RG 5.71 controls specify the "periodic vulnerability scanning and assessments of the security controls, defensive architecture and of all CDAs to identify security deficiencies. The CST performs assessments of security controls and scans for vulnerabilities in CDAs and the environment [no less frequently than once a quarter] or as specified in the security controls in Appendices B and C to RG 5.71, whichever is more frequent, and when new vulnerabilities that could potentially affect the effectiveness the security program and security of the CDAs are identified. In addition, the CST employs up-to-date vulnerability scanning tools and techniques that promote interoperability among tools and automate parts of the vulnerability management process."[e]	Most Log Management and SIEM solutions integrate with VA scanners to the extent that observed events within the infrastructure (from log and event sources) can be correlated against known vulnerabilities (from the most recent VA scan). To support this requirement, consider using the SIEM or Log Management solution to produce regularly scheduled exception reports—either daily or weekly—to indicate both open vulnerabilities on assets, as well as anomaly-based events that may indicate a new exploit against which vulnerability assessments have not been run. This latter report can then be used by a security analyst to assess whether there is sufficient evidence of "new vulnerabilities" to justify an interim vulnerability scan.
NRC 5.71, Control A.4.2.1, Configuration Management NRC RG 5.71 Control A.4.2.1 ties the change management process and vulnerability assessment process together and "ensures that changes made are conducted using these configuration management procedures to avoid the introduction of additional vulnerabilities, weaknesses, or risks into the system."[h]	NRC 5.71 Control 4.2.1 specifically requires that any change to an asset's configuration requires additional vulnerability assessment of that asset. This direct linking of CM and VM requirements further supports the promotion of an integrated solution that can manage configuration management as well as the integrated vulnerability assessment data and the event/alarm data promoted in Control 4.1.3. Although not hard requirements (the tasks could be performed manually using isolated systems), these recommendations for integrated security management indicate that a more advanced security management platform be used. At the time of this writing, there are commercial SIEM and Log Management solutions available that are evolving in this direction.

S2—Documentation	
NERC CIP-005-4 R1, Electronic Security Perimeter NERC CIP requires extensive documentation, including the requirements of CIP-005-4 R1, which mandate the documentation of the ESP, including all assets within the ESP and all access points to the ESP, including access control and security monitoring assets that may be used.[i]	Controls such as CIP-005-4 R1, which require "documentation of assets," may be streamlined using the same Log Management or SIEM systems used to manage the logs that the asset(s) produce. Assets may also be detectable or discoverable using SNMP/In addition by producing logs, the asset(s) comprising the ESP are identified to the central reporting system. Finally, monitoring network flows can identify which asset(s) are deployed on either side of the ESP, as can extrapolation of source and destination IP addresses that may be present in firewall or IPS logs, generated by the ESP.
NERC CIP-005-4 R3, Monitoring Electronic Access CIP-005-4 R3 requires a 24-hour, 7-day monitoring and documentation of electronic access at all access points to the ESP.[j]	Monitoring electronic access at the host level can be achieved by monitoring authentication to the host itself, and can be further scrutinized by monitoring application and/or database access. That is, the human operator logs into the host machine (host authentication logs), launches an application and logs into it (application authentication logs), and as the user performs tasks (application activity logs), the application itself will typically connect to a backend database (database authentication logs). Note that each monitored authentication may use a different set of credentials, which can make it difficult to track a session from the end user to the backend system(s) without the aid of an identity management system. For unauthenticated access, most commercial VA scanners will be able to identify vulnerabilities through which access might be obtained, and are able to produce comprehensive reports indicating the results of such a scan.
NERC CIP-005-4 R4, Cyber Vulnerability Assessment NERC CIP-005-4 R4 mandates that the results of a vulnerability be documented, that a plan be documented to remediate any vulnerabilities found, and that the status of that plan also be documented.[k]	Most commercial VA scanners are able to produce comprehensive reports indicating the results of the scan. The Bandolier project by Digital Bond (www.digitalbond.com) provides a plug-in to the Nessus VA scanner to map the results of the vulnerability assessment to specific NERC CIP controls.[l] Although no tool can automate the generation of an organizational procedure, the actions taken to execute a remediation plan can be monitored and logged to produce a viable audit trail of the remediation activities. This could include the application of patches, disabling of ports or services, etc.

(*Continued*)

Table 10.3 (Continued)

Compliance Control	Recommendations
NERC CIP-005-4 R5, Documentation Review and Maintenance NERC CIP-005-4 R5 requires that all of Electronic Security Perimeter requirements, including access points to the ESP, access controls, and monitored access to the ESP up to and including individual user access to the ESP, be documented and that all relevant logs are maintained and reviewed.[m]	In addition to the event logs of ESP devices, which can easily be collected, reviewed, and retained, CIP-005-4 R5 requires that all individual user account access activity (i.e., authentications) be monitored and logged. To meet minimal compliance requirements, simple log reports of account authentications from VPNs, network access controls, and other relevant logs are likely to be sufficient. Although CIP-005-4 R5 does not necessitate that actual end user identities are tracked (i.e., there is no need to determine how specific user accounts map back to human operators), user account normalization is recommended, as it will facilitate incident detection as well as incident reporting if/when an incident does occur. Like most NERC CIP documentation requirements, CIP-005-4 R5 requires that all logs be retained for a period of 90 days. However, if a log is determined to be associated with a cyber incident, the required retention period increases to 3 years.[n] According to the 2010 Verizon Data Breach Report, the majority of incidents are not detected for months after they occur.[o] Because any "benign" log could be associated with an incident long after the minimum 90-day period, it is recommended that all logs are retained for the full 3 years required by NERC CIP-008-4 R2.
NERC CIP-007-4 R2, Ports and Services NERC CIP-007-4 R2 requires that only those ports and services that are allowed are enabled, and that processes are put in place to ensure that unauthorized ports and services are not in use.[p]	The documentation of those ports and services that are enabled requires either a manual assessment of assets or the use of a vulnerability assessment scanner. Most VA tools have documentation and reporting features to facilitate this type of documentation requirement.

	However, malicious code will commonly open new ports or enable new services, requiring a continuous assessment of ports and services in order to truly ensure that only authorized services are in use. This can be accomplished by monitoring network activity in addition to host and perimeter configurations. Network flow analysis will clearly indicate which ports are actively in use and can be used to generate an alarm when an unknown or unauthorized port is used (see Chapter 9, "Monitoring Enclaves"). Unauthorized ports and services that are in use should be immediately remediated.
NERC CIP-008-4 R2, Cyber Security Incident Documentation NERC differentiates between standard activity logs (what might be thought of as "events" by information security analysts) and identified Cyber Security Incidents. Although most documentation only needs to the retained for 90 days, documentation that is relevant to a Cyber Security Incident must be retained for 3 years.[q]	CIP-008 R2 presents a "worst case" requirement for log retention for 3 years, mandated for any log associated with a cyber security incident. As mentioned above in regards to NERC CIP-005-4 R5, even "benign" logs that are not initially related to a cyber security incident could be associated with an incident after the minimum 90-day retention period is over. If the log is discarded after 90 days, insufficient evidence may be available to adequately document an incident when/if one occurs. Therefore, it is recommended that all logs are retained for the full 3 years required by NERC CIP-008-4 R2.
CFATS RBPS Metric 8.5.4, Incident Reporting CFATS RBPS Metric 8.5.4 illustrates a common reporting requirement across many compliance standards, which is that all incidents must be reported to a higher authority. In the case of Metric 8.5.4, incidents must be reported to senior management and to the DHS's US-CERT at www.us-cert.gov."[r]	Incident reporting controls such as RBPS Metric 8.5.4 can often be met using the alarm or notification functions of information management systems. First, define notification lists of important administrators and/or outside agencies that must be notified of an incident. Define a report containing the necessary summary information of the suspected incident, and configure the security tool (typically a SIEM) to automatically distribute the summary report to the responsible entities. The report template should include the contact information for the next level of escalation, as well as incident handling and escalation procedures, thereby automating incident reporting procedures at several layers of authority.

(*Continued*)

Table 10.3 (Continued)

Compliance Control	Recommendations
CFATS RBPS Metric 8.9.1, Audits As with Metric 8.5.4, Metric 8.9.1 involves the communication of security information to senior management. Where Metric 8.5.4 concerns specific incidents, Metric 8.9.1 concerns the delivery of the results of regularly conducted audits against the facility's cyber security policies.[s]	As with Control 8.5.4, the security information management tool used to collect and report on relevant security data can and should be configured to distribute reports to appropriate managers and/or other authorities, as needed.
ISO/IEC 27002:2005, Control 10.10.1, Audit Logging ISO/IEC 27002:2005 controls for Audit Logging require that logs recording "user activities, exceptions, and information security events" should be collected and retained for an undefined period. Further guidance suggests that these audit logs should include a variety of details including user IDs, dates, times, terminal identities, authentication results, configuration changes, application use, and many other relevant security events.[t]	The required information can be obtained by collecting security events and by monitoring users, networks, hosts, applications, and/or database transactions and configuration files as detailed in Chapter 9, "Monitoring Enclaves." Note that Control 10.10.1 requires that audit logs should include many details that are not typically provided by all logs. This can be addressed through event enrichment, by correlating logs together based upon common indicators and either adding additional detail to the log at the time of collection, or providing the additional context at the time of analysis (i.e., when the report is generated). See Chapter 8, "Exception, Anomaly, and Threat Detection," for more information about data enrichment.
ISO/IEC 27002:2005, Control 13.1.1, Reporting Information Security Events ISO/IEC Control 13.1.1 requires that security events be reported to higher levels of authority within the organization. The control also recommends that an established point of contact should be used, and that he or she should be identifiable and accessible.[u]	As with CFATS RBPS Metric 8.5.4, ISO/IEC 27002:2005's reporting controls can be facilitated through simple configurations to the security information reporting system (typically a Log Management or SIEM solution). By defining notification lists of responsible managers and using these lists for automated incident notifications, the incident will be "reported as quickly as possible," in that it will occur automatically at the time of detection. Limiting these incident reports to summary information of the suspected incident and including escalation procedures with additional contact information included allows responsible staff to assess, approve, and forward incident reports upward through the management chain as is necessary, with minimal delay.

ISO/IEC 27002:2005, Control 13.2.3, Collection of Evidence ISO/IEC 27002:2005, Control 13.2.3 concerns the ability to produce evidence of a cyber security incident where required for legal action.[v]	Although many compliance controls require security logs and information to be stored in a secure manner, with considerations made for nonrepudiation, ISO/IEC 27002:2005, Control 13.2.3 specifically requires that this information be retrievable for presentation to some judicial authority. The operational consideration here is that among potentially billions of stored logs, finding all the relevant logs (and only the relevant logs) related to a particular incident can be daunting. As is often the case, there are a variety of tools available to facilitate this task: from ubiquitous tools such as `grep`, to log search or "IT search" products, to both open-source and commercial Log Management and SIEM products—all of which provide a means of searching through large volumes of log files, to provide a single relevant filtered log cache.
NRC RG 5.71, Control C.5, Records Retention and Handling NRC RG 5.71 requires the establishment of procedures to ensure that all necessary records are developed, reviewed, and retained. Control C.5 specially requires that a facility retain records including, but not limited to, "all digital records, log files, audit files, and nondigital records that capture, record, and analyze network and CDA events. These records are retained to document access history and discover the source of cyber attacks or other security-related incidents affecting CDAs or SSEP functions or both. [Licensee/Applicant] will retain superseded portions of these records for at least 3 years after the record is superseded, unless otherwise specified by the NRC."[w]	Monitoring activities and events in the industrial infrastructure provides the necessary digital records, log files, and audit files to partially satisfy NRC RG 5.71, Control C.5, as well as most other record retention requirements from other compliance standards. Although logs and events cannot measure or document the formation or implementation of a security plan per se, certain logs can also be used to support the requirements to show how certain activities were managed, reviewed, reacted to, and completed. For example, many SIEM systems include workflow integration with help desk ticketing systems, or they may provide limited ticketing systems as a core function of the SIEM. By logging these SIEM-derived activities along with network-derived logs and events, a broader array of governing procedures and policies can be audited.
S3—Monitoring	
NERC CIP-005-4 R3, Monitoring Electronic Access NERC CIP-005-4 R3 specifically requires that electronic access be monitored at all access points to the ESP.[x]	Security perimeters are often thought of in terms of what is not allowed, and as a result, alerts generated by most perimeter security devices (firewalls, IPS, etc.) typically involve traffic that was denied access at the perimeter.

(*Continued*)

Table 10.3 (Continued)

Compliance Control	Recommendations
	However, the requirement to monitor and log access at access points to the Electronic Security Perimeter requires that alerts are generated, collected, and retained that pertain to traffic that is ***allowed*** access at the perimeter. Depending upon the nature of the enclave being secured, this could result in a significant number of alerts; for excample, if a business intelligence server is communication constantly with one or more HMI systems, there will a cotinuous number of new sessions established through the ESP Network flow analysis can also be used to indicate access to/through an ESP. By logging flow records that originate and terminate on different sides of the perimeter. To accommodate this additional event (and/or flow) load, consider using event aggregation (see Chapter 9, “Monitoring Enclaves”) to reduce the total number of collected events. Alternatively, further separate assets into more specialized functional groups, in order to reduce the number of inbound and outbound traffic to any given enclave (see Chapter 7, “Establishing Secure Enclaves”). Note that in the latter case, the total number of access alerts will remain the same and will still need to be supported and managed by a properly sized Log Management or SIEM system. However, the additional level of distribution may facilitate event collection.
NERC CIP-007-4 R5 Account Management NERC CIP-005-4 R5 requires the establishment of procedures to generate logs “of sufficient detail to create historical audit trails of individual user account access activity for a minimum of ninety days.”[y]	Although auditing account activity is typically thought of in terms of authentication logs produced by individual servers or applications, or in other cases by a centralized directory or authentication system, account activities can also be directly monitored. Certain monitoring tools such as **Database Monitor**s, Protocol Monitors or Filters, and Application Monitors can detect and log authentications as they occur, via the capture and decoding of network traffic. This provides a valuable verification that authentication logs are valid and can also help to ensure that authentications do not use default or weak credentials.

NERC CIP-007-4 R2, Ports and Services NERC CIP-007-4 R2 requires the implementation of a process to ensure that only authorized ports and services are in use.[z]	It is possible to monitor for ports and services that are actively in use within the network. This level of monitoring will detect ports and services that may have been enabled after the initial asset inventory and assessment has been completed, and so is recommended even when manual assessments have been made. Network flow monitoring is most useful for this, as it will also indicate the source and destination IP address of the communication, in order to identify the assets that are using unauthorized ports and services. Unauthorized ports and services that are in use should be immediately remediated.
NERC CIP-007-4 R6, Security Status Monitoring NERC CIP-007-4 R6 provides a clear mandate for security monitoring, requiring that all cyber assets within the ESP “implement automated tools or organizational process controls to monitor system events that are related to cyber security.”[aa]	“Automated tools” to monitor security events seem to clearly indicate the use of a Log Management and/or SIEM system (see Chapter 9, “Monitoring Enclaves”). Although NERC does not specifically call out the types of activities and events that require monitoring, the more vague requirement to monitor what is “related to cyber security”[ab] should be assumed to require user, host, network, application, change, and other events.
CFATS RBPS Metric 8.5.2, Network Monitoring RBPS Metric 8.5.2 recommends the use of network monitoring to detect “unauthorized access or the introduction of malicious code,” and requires that immediate alerts be generated, that the resulting security logs be reviewed, and that alerts be responded to in a timely manner. Network monitoring may occur on-site or off-site. Where logging of cyber security events on their networks is not technically feasible (e.g., logging degrades system performance beyond acceptable operational limits), appropriate compensating security controls (e.g., monitoring at the network boundary) are implemented.”[ac]	RBPS Metric 8.5.2 requires network monitoring and deep packet inspection to detect unauthorized access or malicious code, that is, an Intrusion Detection System. However, there are some interesting exceptions: the first allowing on-site or off-site review of security events; and the second allowing for perimeter-only monitoring where network monitoring within an enclave is not feasible. Because industrial systems and protocols are often sensitive to latency, and any in-line network monitoring or deep packet inspection will incur at least some degree of latency, it is foreseeable that such an exception will be justifiable in some cases. However, rather than removing network monitoring to the enclave perimeters, consider implementing an IDS, Industrial Protocol Filter, or Application Monitor within the enclave using passive deployment methods. For example, implement a network tap to “mirror” traffic that needs to be monitored to the monitoring device(s). In this way, the “live” traffic will remain untouched, with no impact caused by the network monitoring facilities. In addition, highly specialized industrial monitoring devices may be available that provide low-latency/low-impact in-line monitoring.

(*Continued*)

Table 10.3 (Continued)

Compliance Control	Recommendations
ISO/IEC 27002:2005, Control 10.10.2, Monitoring System Use Control 10.10.2 of the ISO/IEC 27002:2005 Standard mandates procedures for monitoring the use of information systems, as well as for the regular review of the monitored activity (i.e., logs). This control provides further guidance for the areas of information systems that should be monitored, which include authorized user activities, failed authentications, system alerts or failures, and changes to security settings or configurations.[ad]	This control requires both positive alerts (alerts on successful admin access and all successful admin operations), negative alerts (system alerts and failures), and change events (configuration changes and attempted changes). As with NERC CIP-005-4 R3, ISO/IEC 27002:2005, Control 10.10.2 will produce increased amounts of events, as both benign and malicious activities are logged.
NRC RG 5.71, Control C.3.5, Security Alerts and Advisories NRC RG 5.71 introduces the concept of external monitoring, for the purpose of information gathering in order to strengthen existing security controls. Information sources that should be monitored include “security alerts, bulletins, advisories, and directives from credible external organizations as designated by the NRC.” Monitored information should then be evaluated to determine the scope and need of implementing new or modified security controls.[ae]	NRC RG 5.71 Control 3.5 incorporates external threat intelligence into security monitoring. This is a useful monitoring approach where updated threat lists can be used to dynamically populate “blacklist” security defenses. That is, the security devices such as firewalls, Intrusion Prevention Systems, protocol filters, etc. can block known sources of malware or other threats by matching against a dynamically populated variable. For example, `Deny $ThreatList to Any` rule would block any source in the `$ThreatList` variable. See Chapter 7, “Establishing Secure Enclaves,” for more information on blacklist policies.
NRC RG 5.71, Control C.4.1, Continuous Monitoring and Assessment For the monitoring of internal systems, the NRC provides guidance for ongoing security monitoring and assessment, which include the requirement that “Automated support tools are also used, as appropriate, to accomplish near-real time cyber security management,” including “Ongoing assessments to verify that the security controls implemented for each CDA remain in place	Not surprisingly, the guidelines presented by NRC RG 5.71 focus on the *continuous* monitoring and the *continuous* assessment of cyber activities. Here, rather than focusing on long-term information repositories for proof of compliance, the requirement is on the security of the network.

throughout the life cycle; Verification that rogue assets have not been connected to the infrastructure; Periodic assessments of the need for and effectiveness of the security controls . . . [and the] Periodic security program review to evaluate and improve the effectiveness of the program."[af]	NRC RG 5.71 Control C.4.1 also specifically requires network monitoring to detect the presence of rogue assets. This can be accomplished using most Network Management Systems but is also a good example of how security monitoring can be combined with anomaly detection techniques and correlation to detect policy violations. For example, an alert could be generated if the total number of Source IP Addresses seen within 24 hours of network flow records increases. In a business enterprise, this type of detection would likely result in a high percentage of false positives. In an industrial network, where operations are well defined and controlled, any variation could indicate the presence of rogue assets.
S4—Authentication	
NERC CIP-007-4 R5, Account Management NERC CIP-007-4 R5 requires the establishment of procedures to generate logs "of sufficient detail to create historical audit trails of individual user account access activity for a minimum of ninety days."[ag]	The requirement for user account and authentication activity logs is common among many compliance controls. Of note in NERC CIP-005-4 R5 is the term "sufficient detail" in regard to recreating an audit trail of user account activity. This is important because of the inconsistencies in log formats. Where one log may clearly indicate a user account name and the results of authentication attempts, others may not. In addition, certain applications may pool user accounts, accessing backend resources through a common identifier and password, as was the case with Stuxnet.[ah] There is room for interpretation in "individual user accounts," which could also be significant, as the ability to create an audit trail based on an individual user requires some method of normalizing user identities across multiple systems, while tracking user accounts would only require a distinct audit trail for each account; potentially tracking several accounts in parallel for a single end user (see the section "User Identities and Authentication" in Chapter 9, "Monitoring Enclaves").

(*Continued*)

[a] North American Reliability Corporation, Standard CIP-003-4—Cyber Security—Security Management Controls, <http://www.nerc.com/files/CIP-003-4.pdf>, February 3, 2011 (cited: March 3, 2011).
[b] North American Reliability Corporation, Standard CIP-005-4—Cyber Security—Electronic Security Perimeter(s), <http://www.nerc.com/files/CIP-005-4.pdf>, February 3, 2011 (cited: March 3, 2011).
[c] North American Reliability Corporation, Standard CIP-007-4—Cyber Security—Systems Security Management. <http://www.nerc.com/files/CIP-007-4.pdf>, February 3, 2011 (cited: March 3, 2011).
[d] Ibid.
[e] U.S. Nuclear Regulatory Commission, Regulatory Guide 5.71 (New Regulatory Guide) Cyber Security Programs for Nuclear Facilities, January 2010.
[f] Ibid.
[g] Ibid.
[h] Ibid.
[i] North American Reliability Corporation, Standard CIP-005-4—Cyber Security—Electronic Security Perimeter(s). <http://www.nerc.com/files/CIP-005-4.pdf>, February 3, 2011 (cited: March 3, 2011).
[j] Ibid.
[k] Ibid.
[l] DigitalBond, Inc. Bandolier and NERC CIP. <http://www.digitalbond.com/tools/bandolier/bandolier-and-nerc-cip/> (cited: March 3, 2011).
[m] North American Reliability Corporation, Standard CIP-005-4—Cyber Security—Electronic Security Perimeter(s). <http://www.nerc.com/files/CIP-005-4.pdf>, February 3, 2011 (cited: March 3, 2011).
[n] Ibid.
[o] W. Baker, M. Goudie, A. Hutton, C.D. Hylender, J. Niemantsverdriet, C. Novak, et al. 2010 Data Breach Investigations Report, Verizon. <http://www.verizonbusiness.com/resources/reports/rp_2010-data-breach-report_en_xg.pdf>, 2010 (cited: March 3, 2011).
[p] North American Reliability Corporation, Standard CIP-007-4—Cyber Security—Systems Security Management. <http://www.nerc.com/files/CIP-007-4.pdf>, February 3, 2011 (cited: March 3, 2011).
[q] Ibid.
[r] Department of Homeland Security, Risk-Based Performance Standards Guidance; Chemical Facility Anti-Terrorism Standards (CFATS), May 2009.
[s] Ibid.
[t] International Standards Organization/International Electrotechnical Commission (ISO/IEC). INTERNATIONAL ISO/IEC STANDARD 27002:2005 (E), Information Technology—Security Techniques—Code of Practice for Information Security Management, first edition 2005-06-15.
[u] Ibid.
[v] Ibid.
[w] U.S. Nuclear Regulatory Commission, Regulatory Guide 5.71 (New Regulatory Guide) Cyber Security Programs for Nuclear Facilities, January 2010.
[x] North American Reliability Corporation, Standard CIP-005-4—Cyber Security—Electronic Security Perimeter(s). <http://www.nerc.com/files/CIP-005-4.pdf>, February 3, 2011 (cited: March 3, 2011).
[y] North American Reliability Corporation, Standard CIP-007-4—Cyber Security—Systems Security Management. <http://www.nerc.com/files/CIP-007-4.pdf>, February 3, 2011 (cited: March 3, 2011).
[z] Ibid.
[aa] Ibid.
[ab] Ibid.
[ac] Department of Homeland Security, Risk-Based Performance Standards Guidance; Chemical Facility Anti-Terrorism Standards (CFATS), May 2009.
[ad] International Standards Organization/International Electrotechnical Commission (ISO/IEC), INTERNATIONAL ISO/IEC STANDARD 27002:2005 (E), Information technology—Security techniques—Code of practice for information security management, first edition 2005-06-15.
[ae] U.S. Nuclear Regulatory Commission, Regulatory Guide 5.71 (New Regulatory Guide) Cyber Security Programs for Nuclear Facilities, January 2010.
[af] Ibid.
[ag] North American Reliability Corporation, Standard CIP-007-4—Cyber Security—Systems Security Management. <http://www.nerc.com/files/CIP-007-4.pdf>, February 3, 2011 (cited: March 3, 2011).
[ah] N. Falliere, L.O. Murchu, E. Chien, Symantec. W32.Stuxnet Dossier, Version 1.1, October 2010.

CAUTION

These mappings are only intended to provide a high-level awareness of how security and compliance interrelate. Although based upon the most recent publications of each relevant standard at the time of writing, they do not represent a comprehensive list of the requirements and recommendations for all controls. Specifically cited requirements are excerpts from the original standards documentation only and do not represent the full scope of the referenced standard. Recommendations are provided for the purposes of improving security; adherence to these recommendations does not guarantee compliance with any referenced standard. Always reference the most current publication of the original standards document(s) when planning regulatory compliance efforts.

Figure 10.3 and Table 10.3 focus specifically on security monitoring and auditing controls, and how they can be implemented to support regulatory requirements.

MAPPING COMPLIANCE CONTROLS TO NETWORK SECURITY FUNCTIONS

As an additional reference for mapping compliance and security controls, Table 10.4 provides a reverse mapping, indicating which specific security functions are required by which compliance controls.

COMMON CRITERIA AND FIPS STANDARDS

Unlike other standards, Common Criteria and FIPS aim to certify security products, rather than security policies and processes. The Common Criteria for Information Technology Security Evaluation ("Common Criteria" or "CC") is an international framework that is currently recognized by Australia/New Zealand, Canada, France, Germany, Japan, the Netherlands, Spain, the United Kingdom, and the United States.[23] FIPS standards are defined by NIST in Federal Information Processing Standards Publications or FIPS PUBs. Although there are several FIPS standards, it is the FIPS 140-2 Standard that validates information encryption that is most relevant to information security products.

Common Criteria

Common Criteria's framework defines both functional and assurance requirements, which security vendors can test against in order to validate the security of the product in question.[24] Certification by an authorized Common Criteria testing facility provides a high level of assurance that specific security controls have been appropriately specified and implemented into the product.

Table 10.4 Mapping Compliance Controls to Network Security Functions

Compliance Regulation	Category	Mapping	Chapter
NERC CIP-003-4 R6	Asset Configurations	H1	Chapter 6, "Vulnerability and Risk Assessment"
NERC CIP-003-4 R6	Asset Configurations	S1	Chapter 9, "Monitoring Enclaves: Determining What to Monitor"
NERC CIP-005-4 R1	ESP	P1	Chapter 7, "Establishing Secure Enclaves"
NERC CIP-005-4 R1.6	Documentation	S2	Chapter 9, "Monitoring Enclaves: Log Storage and Retention"
NERC CIP-005-4 R2	Access Control	P2	Chapter 7, "Establishing Secure Enclaves: Securing Enclave Perimeters"
NERC CIP-005-4 R3	Monitoring	P3	Chapter 8, "Exception, Anomaly, and Threat Detection"
NERC CIP-005-4 R3	Monitoring	S3	Chapter 9, "Monitoring Enclaves: Determining What to Monitor"
NERC CIP-005-4 R3	Documentation	S2	Chapter 9, "Monitoring Enclaves: Log Storage and Retention"
NERC CIP-005-4 R4.5	Asset Configurations	S1	Chapter 9, "Monitoring Enclaves: Determining What to Monitor"
NERC CIP-005-4 R4.5	Asset Configurations	S2	Chapter 9, "Monitoring Enclaves: Determining What to Monitor"
NERC CIP-005-4 R5.3	Documentation	S2	Chapter 9, "Monitoring Enclaves: Log Storage and Retention"
NERC CIP-007-4 R5.1.2	Authentication	S4	Chapter 9, "Monitoring Enclaves: Determining What to Monitor"
NERC CIP-007-4 R5.1.2	Monitoring	S3	Chapter 9, "Monitoring Enclaves: Determining What to Monitor"
NERC CIP-007-4 R5.1.2	Documentation	S2	Chapter 9, "Monitoring Enclaves: Log Storage and Retention"
NERC CIP-007-4 R2	Ports and Services	H2	Chapter 7, "Establishing Secure Enclaves: Securing Enclave Interiors" Chapter 8, "Exception, Anomaly, and Threat Detection"
NERC CIP-007-4 R2	Monitoring	S3	Chapter 9, "Monitoring Enclaves: Determining What to Monitor"
NERC CIP-007-4 R2	Documentation	S2	Chapter 9, "Monitoring Enclaves: Log Storage and Retention"
NERC CIP-007-4 R3.2	Asset Configurations	S1	Chapter 9, "Monitoring Enclaves: Determining What to Monitor"
NERC CIP-007-4 R3.2	Asset Configurations	S2	Chapter 9, "Monitoring Enclaves: Determining What to Monitor"
NERC CIP-007-4 R4	Asset Configuration	H1	Chapter 6, "Vulnerability and Risk Assessment"
NERC CIP-007-4 R5	Anti-Malware	H3	Chapter 7, "Establishing Secure Enclaves"

NERC CIP-007-4 R6	Monitoring	S3	Chapter 9, “Monitoring Enclaves: Determining What to Monitor”
NERC CIP-007-4 R8.4	Asset Configurations	S1	Chapter 9, “Monitoring Enclaves: Determining What to Monitor”
NERC CIP-007-4 R8.4	Documentation	S2	Chapter 9, “Monitoring Enclaves: Determining What to Monitor”
NERC CIP-008-4 R2	Documentation	S2	Chapter 9, “Monitoring Enclaves: Log Storage and Retention”
CFATS RBPS Metric 8.2.5	Authentication	H4	Chapter 7, “Establishing Secure Enclaves: Securing Enclave Interiors” Chapter 9, “Monitoring Enclaves: Determining What to Monitor”
CFATS RBPS Metric 8.3.2	Authentication	H4	Chapter 7, “Establishing Secure Enclaves: Securing Enclave Interiors” Chapter 9, “Monitoring Enclaves: Determining What to Monitor”
CFATS RBPS Metric 8.3.4	Authentication	H4	Chapter 7, “Establishing Secure Enclaves: Securing Enclave Interiors” Chapter 9, “Monitoring Enclaves: Determining What to Monitor”
CFATS RBPS Metric 8.5.1	Anti-Malware	H3	Chapter 7, “Establishing Secure Enclaves: Securing Enclave Interiors”
CFATS RBPS Metric 8.5.1	ESP	P1	Chapter 7, “Establishing Secure Enclaves”
CFATS RBPS Metric 8.5.2	Monitoring	P2	Chapter 7, “Establishing Secure Enclaves: Securing Enclave Perimeters” Chapter 8, “Exception, Anomaly, and Threat Detection”
CFATS RBPS Metric 8.5.2	Monitoring	S3	Chapter 9, “Monitoring Enclaves: Determining What to Monitor”
CFATS RBPS Metric 8.5.4	Documentation	S2	Chapter 9, “Monitoring Enclaves: Log Storage and Retention”
CFATS RBPS Metric 8.8.2	Asset Configuration	H1	Chapter 6, “Vulnerability and Risk Assessment”
CFATS RBPS Metric 8.9.1	Documentation	S2	Chapter 9, “Monitoring Enclaves: Log Storage and Retention”
ISO/IEC 27002:2005 Control 10.1.2	Asset Configuration	H1	Chapter 6, “Vulnerability and Risk Assessment”
ISO/IEC 27002:2005 Control 10.4.1	Anti-Malware	H3	Chapter 7, “Establishing Secure Enclaves: Securing Enclave Interiors”
ISO/IEC 27002:2005 Control 10.6.1	ESP	P1	Chapter 7, “Establishing Secure Enclaves”

(*Continued*)

Table 10.4 (Continued)

Compliance Regulation	Category	Mapping	Chapter
ISO/IEC 27002:2005 Control 10.10.1	Documentation	S2	Chapter 9, “Monitoring Enclaves: Log Storage and Retention”
ISO/IEC 27002:2005 Control 10.10.2	Monitoring	S3	Chapter 9, “Monitoring Enclaves: Determining What to Monitor”
ISO/IEC 27002:2005 Control 11.2.1	Authentication	H4	Chapter 7, “Establishing Secure Enclaves: Securing Enclave Interiors” Chapter 9, “Monitoring Enclaves: Determining What to Monitor”
ISO/IEC 27002:2005 Control 11.4.1	Authentication	P3	Chapter 7, “Establishing Secure Enclaves: Securing Enclave Perimeters” Chapter 8, “Exception, Anomaly, and Threat Detection”
ISO/IEC 27002:2005 Control 11.4.2	Authentication	P3	Chapter 7, “Establishing Secure Enclaves: Securing Enclave Perimeters” Chapter 8, “Exception, Anomaly, and Threat Detection”
ISO/IEC 27002:2005 Control 11.4.4	Authentication	P3	Chapter 7, “Establishing Secure Enclaves: Securing Enclave Perimeters” Chapter 8, “Exception, Anomaly, and Threat Detection”
ISO/IEC 27002:2005 Control 11.4.5	ESP	P1	Chapter 7, “Establishing Secure Enclaves”
ISO/IEC 27002:2005 Control 11.4.6	ESP	P1	Chapter 7, “Establishing Secure Enclaves”
ISO/IEC 27002:2005 Control 11.4.7	ESP	P1	Chapter 7, “Establishing Secure Enclaves”
ISO/IEC 27002:2005 Control 11.6.2	ESP	P1	Chapter 7, “Establishing Secure Enclaves”
ISO/IEC 27002:2005 Control 12.6.1	Asset Configuration	H1	Chapter 6, “Vulnerability and Risk Assessment”

ISO/IEC 27002:2005 Control 13.1.1	Documentation	S2	Chapter 9, “Monitoring Enclaves: Log Storage and Retention”
ISO/IEC 27002:2005 Control 13.2.3	Documentation	S2	Chapter 9, “Monitoring Enclaves: Log Storage and Retention”
NRC RG 5.71 Control A4.1	Monitoring	S3	Chapter 9, “Monitoring Enclaves: Determining What to Monitor”
NRC RG 5.71 Control A4.1.3	Asset Configurations	S1	Chapter 9, “Monitoring Enclaves: Determining What to Monitor”
NRC RG 5.71 Control A4.2.1	Asset Configurations	S1	Chapter 9, “Monitoring Enclaves: Determining What to Monitor”
NRC RG 5.71 Control A4.2.2	Asset Configurations	S1	Chapter 9, “Monitoring Enclaves: Determining What to Monitor”
NRC RG 5.71 Control A5	Documentation	S2	Chapter 9, “Monitoring Enclaves: Log Storage and Retention”
NRC RG 5.71 Control B1.3	Authentication	H4	Chapter 7, “Establishing Secure Enclaves: Securing Enclave Interiors” Chapter 9, “Monitoring Enclaves: Determining What to Monitor”
NRC RG 5.71 Control B1.4	ESP	P1	Chapter 7, “Establishing Secure Enclaves”
NRC RG 5.71 Control B1.4	Authentication	P3	Chapter 7, “Establishing Secure Enclaves: Securing Enclave Perimeters” Chapter 8, “Exception, Anomaly, and Threat Detection” Chapter 9, “Monitoring Enclaves: Determining What to Monitor”
NRC RG 5.71 Control B1.15	Authentication	P3	Chapter 7, “Establishing Secure Enclaves: Securing Enclave Perimeters” Chapter 8, “Exception, Anomaly, and Threat Detection” Chapter 9, “Monitoring Enclaves: Determining What to Monitor”
NRC RG 5.71 Control B1.16	Ports and Services	P4	Chapter 7, “Establishing Secure Enclaves: Securing Enclave Perimeters”
NRC RG 5.71 Control B3.4	ESP	P1	Chapter 7, “Establishing Secure Enclaves”
NRC RG 5.71 Control B4.2	Authentication	H4	Chapter 7, “Establishing Secure Enclaves: Securing Enclave Interiors” Chapter 9, “Monitoring Enclaves: Determining What to Monitor”

(*Continued*)

Table 10.4 (Continued)

Compliance Regulation	Category	Mapping	Chapter
NRC RG 5.71 Control B4.3	Authentication	H4	Chapter 7, “Establishing Secure Enclaves: Securing Enclave Interiors” Chapter 9, “Monitoring Enclaves: Determining What to Monitor”
NRC RG 5.71 Control B5.1	Ports and Services	H2	Chapter 7, “Establishing Secure Enclaves: Securing Enclave Interiors” Chapter 8, “Exception, Anomaly, and Threat Detection”
NRC RG 5.71 Control B5.2	Anti-Malware	H3	Chapter 7, “Establishing Secure Enclaves: Securing Enclave Interiors”
NRC RG 5.71 Control B5.3	Asset Configuration	H1	Chapter 7, “Establishing Secure Enclaves: Securing Enclave Interiors” Chapter 9, “Monitoring Enclaves: Determining What to Monitor”
NRC RG 5.71 Control C3.5	Monitoring	S3	Chapter 9, “Monitoring Enclaves: Determining What to Monitor”
NIST SP 800-82 Control 5.3.2	ESP	P1	Chapter 7, “Establishing Secure Enclaves”
NIST SP 800-82 Control 5.3.4	ESP	P1	Chapter 7, “Establishing Secure Enclaves”
NIST SP 800-82 Control 5.3.5	ESP	P1	Chapter 7, “Establishing Secure Enclaves”
NIST SP 800-82 Control 5.5	ESP	P1	Chapter 7, “Establishing Secure Enclaves”
NIST SP 800-82 Control 6.2.6.2	ESP	P1	Chapter 7, “Establishing Secure Enclaves”
NIST SP 800-82 Control 6.2.4	Asset Configuration	H1	Chapter 6, “Vulnerability and Risk Assessment”
NIST SP 800-82 Control 6.2.6.1	Anti-Malware	H3	Chapter 7, “Establishing Secure Enclaves: Securing Enclave Interiors”
NIST SP 800-82 Control 6.3.2	Authentication	H4	Chapter 7, “Establishing Secure Enclaves: Securing Enclave Interiors” Chapter 9, “Monitoring Enclaves: Determining What to Monitor”
NIST SP 800-82 Control 6.3.3	Documentation	S2	Chapter 9, “Monitoring Enclaves: Log Storage and Retention”

The evaluations required prior to certification are extensive and include the following:

- Protection Profiles (PP)
- Security Target (ST)
- Security Functional Requirements (SFRs)
- Security Assurance Requirements (SARs)
- Evaluation Assurance Level (EAL)

The Security Target defines what is evaluated during the certification process, providing both the necessary guidance during evaluation as well as high-level indication of what has been evaluated after an evaluation is complete.[25]

The Security Targets are translated to the more specific Security Functional Requirements (SFRs), which provide the detailed requirements against which the various STs are evaluated. The SFRs provide a normalized set of terms and requirements designed so that different STs for different products can be evaluated using common tests and controls, to provide an accurate comparison.

When common requirements are established for a particular product type or category, typically by a standards organization, they can be used to develop a common Protection Profile (PP), which is similar to an ST in that it provides a high-level indication of the assessment, but different in that the specific targets are predefined within the PP.[26] For example, there is a Common Criteria Protection Profile for Intrusion Detection and Prevention Systems that defines the specific security targets that an IDS or IPS must meet to earn certification.

Perhaps the most commonly identified CC metric is the Evaluation Assurance Level, or EAL. EALs measure Development (ADV), Guidance Documents (AGD), Lifecycle Support (ALC), Security Target Evaluation (ASE), Tests (ATE), and Vulnerability Assessment (AVA).[27] There are seven total assurance levels, EAL 1 through EAL 7, each of which indicates a more extensive degree of evaluation against a more exhaustive set of requirements for each of these components. For example, to compare just one of the evaluation requirements (AVA, Vulnerability Assessment), CC EAL 1 provides a basic level of assurance, using a limited security target, and a vulnerability assessment consisting only of a search for potential vulnerabilities in the public domain.[28] In contrast, EAL 3 requires a "vulnerability analysis . . . demonstrating resistance to penetration attackers with a basic attack potential,"[29] and EAL 4 requires a "vulnerability analysis . . . demonstrating resistance to penetration attackers with an Enhanced-Basic attack potential" (i.e., more sophisticated attack profiles for a more thorough vulnerability assurance level).[30] At the most extensive end of the certification assurance spectrum is EAL 7, which requires "complete independent confirmation of the developer test results, and an independent vulnerability analysis demonstrating resistance to penetration attackers with a high attack potential."[31]

It is important to understand that the EAL level does not measure the level of security of the product that is under evaluation but rather measures the degree to which the product's security is tested. Therefore, a higher EAL does not necessarily indicate a more secure system. It is the specific security target(s) being evaluated that indicate the functional requirements of the system. When comparing like systems that are tested against identical targets, the higher EAL indicates that those targets were more thoroughly tested and evaluated, and therefore, the higher EAL provides additional confidence in the proper and secure function of the system.

FIPS 140-2

The Federal Information Processing Standards Publication (FIPS PUB) 140-2 establishes the requirements for the "cryptographic modules" that are used within a cyber asset or system. There are four qualitative levels of FIPS validation, Levels 1 through 4, which like Common Criteria's EALs intend to validate increasingly thorough assurance. With FIPS 140-2, this assurance is in the form of cryptographic integrity: basically, how resistant encrypted boundaries are to penetration.[32] FIPS 140-2 covers the implementation and use of Symmetric and Asymmetric Keys, the Secure Hash Standard, Random Number Generators, and Message Authentication.[33] The specific validation levels represent increasingly more stringent controls to prevent physical access to information with the encrypted boundary. For example, FIPS 140-2 Level 2 requires that data cannot be accessed, physically, even through the removal of disk drives or direct access to system memory. Level 3 provides stronger physical controls to prevent access to and tampering, even through ventilation holes, whereas Level 4 even accommodates environmental failures to protect the encrypted data against recovery during or following a failure.[34]

SUMMARY

Understanding how regulatory standards and regulations can impact the security of a network or system will help at all stages of industrial network security planning and implementation. For example, specific compliance controls might dictate the use of certain products or services to improve security, and/or how to configure specific security products.

The security products themselves are subject to regulation as well, of course. The Common Criteria standards provide a means for evaluating the function and assurance of a product in a manner designed to facilitate the comparison of similar products, whereas FIPS standards such as FIPS 140-2 can provide further validation of specific security functions (in this case, encryption) used by a product.

ENDNOTES

1. M. Asante, NERC, Harder questions on CIP compliance update: ask the expert, 2010 SCADA and Process Control Summit, The SANS Institute, March 29, 2010.
2. North American Reliability Corporation, Standard CIP-001-4—Sabotage Reporting. <http://www.nerc.com/files/CIP-001-4.pdf>, February 3, 2011 (cited: March 3, 2011).

3. North American Reliability Corporation, Standard CIP-002-4—Cyber Security—Critical Cyber Asset Identification. <http://www.nerc.com/files/CIP-002-4.pdf>, February 3, 2011 (cited: March 3, 2011).
4. North American Reliability Corporation, Standard CIP-003-4—Cyber Security—Security Management Controls. <http://www.nerc.com/files/CIP-003-4.pdf>, February 3, 2011 (cited: March 3, 2011).
5. North American Reliability Corporation, Standard CIP-004-4—Cyber Security—Personnel and Training. <http://www.nerc.com/files/CIP-004-4.pdf>, February 3, 2011 (cited: March 3, 2011).
6. North American Reliability Corporation, Standard CIP-005-4—Cyber Security—Electronic Security Perimeter(s). <http://www.nerc.com/files/CIP-005-4.pdf>, February 3, 2011 (cited: March 3, 2011).
7. North American Reliability Corporation, Standard CIP-006-4—Cyber Security—Physical Security of Critical Cyber Assets. <http://www.nerc.com/files/CIP-006-4.pdf>, February 3, 2011 (cited: March 3, 2011).
8. North American Reliability Corporation, Standard CIP-007-4—Cyber Security—Systems Security Management. <http://www.nerc.com/files/CIP-007-4.pdf>, February 3, 2011 (cited: March 3, 2011).
9. North American Reliability Corporation, Standard CIP-008-4—Cyber Security—Incident Reporting and Response Planning. <http://www.nerc.com/files/CIP-008-4.pdf>, February 3, 2011 (cited: March 3, 2011).
10. North American Reliability Corporation, Standard CIP-001-9—Cyber Security—Recovery Plans for Critical Cyber Assets. <http://www.nerc.com/files/CIP-009-4.pdf>, February 3, 2011 (cited: March 3, 2011).
11. Department of Homeland Security, Risk-Based Performance Standards Guidance; Chemical Facility Anti-Terrorism Standards, May 2009.
12. Ibid.
13. Ibid.
14. Ibid.
15. Ibid.
16. Ibid.
17. Ibid.
18. International Standards Organization/International Electrotechnical Commission (ISO/IEC), About ISO. http://www.iso.org/iso/about.htm (cited: March 21, 2011).
19. International Standards Organization/International Electrotechnical Commission (ISO/IEC), International ISO/IEC Standard 27002:2005 (E), Information Technology—Security Techniques—Code of Practice for Information Security Management, first edition 2005-06-15.
20. U.S. Nuclear Regulatory Commission, Regulatory Guide 5.71 (New Regulatory Guide) Cyber Security Programs for Nuclear Facilities, January 2010.
21. K. Stouffer, J. Falco, K. Scarfone, National Institute of Standards and Technology, Special Publication 800-82 (Final Public Draft), Guide to Industrial Control Systems (ICS) Security, September 2008.
22. The Unified Compliance Framework, What is the UCF? <http://www.unifiedcompliance.com/what_is_ucf> (cited: March 21, 2011).
23. The Common Criteria Working Group, Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model, Version 3.1, Revision 3 Final, July 2009.

24. Ibid.
25. Ibid.
26. Ibid.
27. The Common Criteria Working Group, Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Components, Version 3.1, Revision 3 Final, July 2009.
28. Ibid.
29. Ibid.
30. Ibid.
31. Ibid.
32. National Institute of Standards and Technology, Information Technology Laboratory, Federal Information Processing Standards Publication 140-2, Security Requirements for Cryptographic Modules, May 25, 2001.
33. Ibid.
34. Ibid.

CHAPTER

Common Pitfalls and Mistakes

11

INFORMATION IN THIS CHAPTER:

- Complacency
- Misconfigurations
- Compliance vs. Security
- Scope and Scale

Even with best of intentions, a qualified staff, budget, and time, it can be difficult to implement strong security measures into any network, and even more so into an industrial network. While assessing real, deployed networks, many common pitfalls and mistakes are made. One of the most common and dangerous mistakes that can be made is complacency, either as a result of overconfidence or because of stubborn refusal to recognize the threats that exist against industrial networks. Other pitfalls include simple misconfigurations of both assets and security devices, resulting in a false sense of security while the industrial network systems are vulnerable; confusing security best practices with compliance requirements; and finally—even when everything else is done correctly—substandard security products that, despite precise efforts to configure and tune them correctly, simply fail to function under the increased load of an actual cyber incident.

COMPLACENCY

Complacency is a danger to any security profile. Just as a boxer needs to always maintain a proper guard, network security professionals need to assume a similarly defensive posture. However, it is easy to let our guard down when there is no real belief or conviction that a threat is real or when there is overconfidence in the defenses that have already been established. The following examples are the result of complacency in some form or another.

Vulnerability Assessments vs. Zero-Days

Most recommendations, including those made within this book, include some form of vulnerability assessment and/or penetration testing. This process will potentially uncover areas of risk that can then be addressed by patching systems and strengthening the policies of firewalls or IPSs. At the end of this process, a common misperception is that the network is now 100% secure, as there are no more open vulnerabilities against it.

However, any vulnerability assessment or penetration test will only identify how susceptible a network is against known attacks, exploits, and penetration techniques. In reality, there are unknown threats that cannot be accounted for. Therefore, no security plan is fully complete without some method of accounting for unknown attacks. This includes

1. Using multiple layers of defense. Some defensive products may have different detection signatures, more accurate profiles, or different threat research that might allow one product to detect something that another product missed.
2. Using alternate threat detection mechanisms. In addition to "blacklist" based protection, such as what is provided by a firewall or an IPS, utilize anomaly detection products to detect abnormal behavior that could indicate a possible threat, and/or utilize "whitelisting" to block anything that is not specifically identified as a known good service or application.
3. Finally, using the full capability of security monitoring and analysis tools to provide Situational Awareness across the network as a whole, potentially identifying unknown threats that might go undetected by perimeter security devices.

Real Security vs. Policy and Awareness

Security is a process that not only demands a well thought out information security practice but also depends heavily on the human element. Even the strongest network perimeter can be circumvented by an end user—either intentionally as an act of sabotage or in innocence and ignorance. Even a true air gap can be breached if a worker enters the physical security zone and plugs in an unauthorized device, such as a USB drive, an iPod, or some other intelligent mobile device.

While a strong network security practice will anticipate and account for many of these scenarios (e.g., by removing or disabling USB connectors on critical assets), it will never be possible to anticipate every event. Only a properly trained and motivated staff can ultimately ensure that the established technical controls will operate successfully.

Conversely, a well-established security training program coupled with the best and most honest intentions of the entire employee base cannot protect the network against a real threat unless the proper technical security controls are also in place. Knowing not to visit public websites from inside of a "secure" enclave is not enough; if the connection is openly allowed, there is a clear and direct vector into the enclave that can be exploited by an attacker.

The Air Gap Myth

In a time where open networking protocols and wireless networks are prominently used, there is still the misperception that a true air gap exists, protecting critical industrial systems simply because they are not connected to the IT network.

In reality, even a real air gap (if one truly does exist) is of little use in defending against cyber attacks, because cyber attacks have evolved past physical wires. Many assets that were not designed or intended to support wireless network communications include embedded Wi-Fi capabilities at the microprocessor level,[1] which can be exploited by attackers ranging from the skilled cyber terrorist to a disgruntled worker with an understanding of wireless technologies.[2]

In addition, there is the high possibility that a threat could be walked into a critical network, stepping across the air gap with the aid of a human carrier. Only strong security awareness and strong technical security controls can truly "gap" a networked system.

MISCONFIGURATIONS

If complacency is an error of intention, misconfigurations are errors of implementation. In a 10-year study completed in 2010, configuration weaknesses accounted for 16% of exploits in industrial control systems.[3] With misconfigurations responsible for such a high level of risk, it is no wonder that security recommendations from NERC CIP, CFATS, NRC RG 5.71, ISO/IEC 27002:2005, and the NSIT SP 800 documents focus so heavily on configuration control and management. Simple errors can negate all of the benefits of a specific security device (such as using the default password on a firewall), exposing an entire enclave, while misconfigured hosts can provide easy penetration and propagation through a network once it is breached.

The use of default accounts and passwords is a common misconfiguration. Others include the lack of outbound monitoring or policy enforcement in perimeter controls and the introduction of intentional security holes for a legitimate business purpose, which is given the affectionate moniker of "the executive override." Perhaps the most common configuration error, however, is the "set it and forget it" approach. Because effective security is an ongoing process, any configuration that is not continuously assessed, monitored, and managed will eventually shift its alignment with the desired security policies, opening unintentional holes through which an attack can occur.

TIP

While the process of performing vulnerability assessments and penetration tests should uncover most configuration issues, there are also configuration assessment tools to help with configuration assurance. These tools—which may be part of a configuration management system, a SIEM, or as a standalone product—look for common errors in configuration files. For example, a firewall configuration policy should not include "allow all" policies, or policies that do not explicitly define the source and destination IP address(es) and port(s). Especially when combined with regular vulnerability assessments, these tools can simplify the process of assuring the strength of a security configuration, so that the validated configuration files can then be monitored and controlled.

Default Accounts and Passwords

The use of default accounts and passwords is common and dangerous. The initial stages of most attacks involve the enumeration of legitimate system and user identities, a process that is necessary to determine vulnerabilities so that an exploit can be attempted (see Chapter 6, "Vulnerability and Risk Assessment"). If the username and password of a system are already known, the attacker—whether an outside entity or an internal user—can simply and easily authenticate, often with administrative privileges since most default accounts exist for the purpose of initial setup and configuration of other user accounts. Regardless of how secure the system is otherwise, the system is now highly vulnerable and at risk: security configurations can be altered to allow broader access, software can be installed, new accounts can be created, etc. In essence, the successful administrative login to any system is the end game of most hacking attempts.

The use of default passwords, or to a lesser degree weak passwords, therefore is a primary concern. A quick search on the web will provide most default passwords, as well as sites that specifically track and document default credentials, making them easy to obtain.[4] However, these default password lists can be used for benevolent intent as well. The solution is simple: disable all default accounts where possible, and require unique user accounts with strong credentials.

Unfortunately, unless the device in question enforces strong password controls, it is difficult to ensure that all unique user accounts will use strong passwords. Luckily, both default and weak passwords are easy to detect. By using these sources the same way a hacker would, it is possible to define a blacklist of known default passwords, which can then be used by various security products to detect when a default password is in use. Weak passwords can also be easily detected, using regular expressions. For example, the following regular expression checks for a password that is a minimum of eight characters, with at least one uppercase letter, one lowercase letter, and one number.[5]

```
^(?=.{8,})(?=.*\d)(?=.*[a-z])(?=.*[A-Z])(?!.*\s).*$
```

Applied as a detection signature, the following might be used to detect either weak passwords or default passwords:

```
((password != /^(?=.{8,})(?=.*\d)(?=.*[a-z])(?=.*[A-Z])
(?!.*\s).*$/) || (password == $defaultPasswords))
```

Whatever measures are taken to eliminate default passwords and enforce strong password use, establishing unique and strongly authenticated accounts is one of the most basic and necessary steps in securing any network.

Lack of Outbound Security and Monitoring

It is easy to think of an "attack" as an inbound event: someone is attempting to break into the industrial network from "the outside." However, as shown in Chapter 7, "Establishing Secure Enclaves," there are many access control points to consider, and the "outside" of one enclave may be the "inside" of another. In addition, there are inside attackers including but not limited to disgruntled employees or “trusted” third parties. It is critical to enforce access control and traffic flow in both directions: both into and out of every enclave in order to ensure that an inbound attack is not originating from inside the network.

In addition, many breaches result in the infection and propagation of malware, which will typically attempt to connect back out of the network to a public IP. Depending on the sophistication of the attack, the outbound connection may be well hidden or obvious, but if firewall and IPS policies are only enforcing traffic in one direction, it does not matter. Monitoring is equally as important: even if the perimeter security policies are strong enough to stop the malicious outbound traffic, the fact that the traffic originated from the inside indicates that there is a malicious entity (user or malware) inside your network. Monitoring will alert you to this, and can also help indicate where the attacks are originating from.

The Executive Override

The “Executive Override” is an intentional policy allowing traffic through a perimeter firewall for a nonessential use (at least from the perspective of industrial operations, there may be a very legitimate business case for the exception). It is almost unavoidable as business operations continue to evolve, but it is addressable.

One example of the “Executive Override” is the need for real-time process data within the business enterprise (in the least secure zone of the network!) so that financial and trading analysis can be made using the absolute latest information on production yields, quality, manufacturing efficiency, etc. This will often be done by extending Historian data through one or more firewalls, “poking holes” in the security perimeter of (potentially) several enclaves. The result, when implemented poorly, is a direct vector of attack from the executive console to the Historian system, which resides inside of a critical enclave.

To secure these inevitable requirements, establish a purely supervisory enclave that consists of purely read-only information data (i.e., no “control”), and replicate the necessary Historians or HMIs into this enclave over a unidirectional gateway or data diode for physical layer separation. Now, these replicated systems can be allowed to communicate to less secure zones without risk of any malicious backwash into the critical zones.

The Ronco Perimeter

The “set it and forget it” process extolled by clever Ronco kitchen products (www.ronco.com) may be suitable for a rotisserie cooker, but it is not suitable for cyber security. Cyber security is a process, not a product, and therefore needs to be continuously assessed, adjusted, and improved. Even after a vulnerability assessment is complete and security policies are locked down, there are still steps to be taken. Specifically, these steps include

- Monitoring the newly established configuration to make sure it does not change, by an unknown administrator, a disgruntled insider, or an attacker modifying defenses in order to penetrate deeper into the network.

- Identifying and adapting to new threat types, including new zero-day attacks, new social engineering schemes, and attacks introduced by new technology.
- Adding new security controls, and/or adjusting the configurations of existing controls to minimize risk in an ongoing manner.

The smartphone is an excellent example of how the introduction of new technology needs to be accommodated by security policies. Are these devices capable of transporting files (and potentially malware)? Can they be mounted as a removable drive via USB, wireless, or Bluetooth? Is it possible to route between a cellular carrier network and local Wi-Fi networks supported by these devices? Are there existing controls to prevent misuse of intelligent mobile platforms, or are new controls needed?

COMPLIANCE VS. SECURITY

Compliance controls represent any number of guidelines and/or mandates that have been developed in order to ensure that organizations have correctly planned and implemented the necessary security measures to protect whatever sensitive materials, systems, or services may need protecting. While the controls discussed in this book (see Chapter 10, "Standards and Regulations") relate specifically to cyber security, there are compliance regulations spanning almost every foreseeable aspect of information security, including fraud prevention, privacy, safety, financial responsibility, financial integrity, and more.

While compliance controls are presumably developed with good intentions, they can sometimes impact the security process that they are trying to assure. This is because the necessary steps that need to be taken to prove that you have implemented and enforced certain security controls are different from the controls themselves. As a result, it is a common pitfall to focus efforts on obtaining the necessary documentation to earn "compliance check off boxes" rather than on the security goals of the standard. This can result in misplaced efforts, especially in preparation for an audit, at which point security analysis might be temporarily repurposed to the role of the compliance analyst.

NOTE

While it is often possible to become compliant with a particular standard or regulation without implementing strong security controls, the documentation and audit trails required by most compliance standards are often easier to obtain when those security controls are in place. Compliance officers and security analysts should work together early in the planning stages to ensure that the intended security controls are implemented in a manner that satisfies both areas of responsibility. Compliance does not equal security, and security does not equal compliance; however, the two can be obtained together.

Audit Fodder

The various requirements that are excerpted in Chapter 10 from NERC CIP, CFATS, NRC, ISO, and NIST could be summarized quickly as the need to implement measures to protect against attacks, to log and review that activity on a continuous basis, and to prove it by retaining those logs for a set period of time. The issue is that last piece: to prove compliance by retaining logs.

Unfortunately, simply collecting event logs for compliance is not going to help with security, unless those logs are collected from a security net that is properly cast, correctly configured, and regularly reviewed. While log retention can prove that certain measures are in place, and while documented plans and policies can prove that an organization's intentions are well founded, neither can entirely prove or disprove that the ongoing security practices are good, bad, efficient, successful, or complete.

The issue comes when systems are implemented with the primary goal to provide the evidence of compliance and the secondary goal of security. The result in these cases is "audit fodder," mounded high to satisfy the requirements of the auditor. Instead, security measures should be designed and implemented for security first. The resulting logs and documentation should satisfy the reporting and auditing requirements of the standard, and the network will be more secure for the effort. If the compliance requirements are still not met after the best efforts to secure the network are complete, continue the process of assessment, remediation, monitoring, response, retention, and then back to assessment again—repeating the cycle until all requirements are met.

The "One Week Compliance Window"

The pre-audit reallocation of resources has been observed in many organizations. With an impending audit, documentation must be put in order. Logs that have been retained for (potentially) many years need to be cross-referenced, correlated, collated, and formatted into suitable reports. Networks need to be mapped, vulnerabilities detected and resolved, patches applied, and antivirus signatures updated. In short, the network needs to be put in perfect order, and cleanly represented on paper for review by a compliance auditor. In many cases, however, the technical resource that is utilized to perform this extensive work is the only resource that is available: the security analyst(s). The skilled security professionals are tasked with cleaning house, taking their attention away from real-time, day-to-day security operations.

The result is a flurry of activity that actually weakens the network while the organization "becomes compliant." Afterward the staff members are reassigned to their original duties, and if all goes well the organization will remain in compliance until the next audit occurs. In reality, new systems are implemented, new patches are applied, there is a merger or a purchase or a reduction—something that misaligns the current security policies and practices with the auditable compliance goals. Until the next audit cycle occurs, these errors may go unnoticed or disregarded.

The result is obvious though not always possible or realistic: ensure that dedicated compliance resources are in place to separate the responsibility of the audit from the security practices. Ironically, this separation can be facilitated through the closer integration of security and compliance efforts, for example, by mapping compliance controls to specific security events, so that as incident(s) occur the responsible security analysts are immediately aware of the impact that the incident(s) may have on the organization's compliance goals. Chapter 10, "Standards and Regulations," begins the process of mapping compliance and security controls together. This effort may also be facilitated through the use of the Unified Compliance Framework (www.unifiedcompliance.com).

SCOPE AND SCALE

Another common mistake made when attempting to secure a control system is to think of the industrial network as an isolated system. While once air-gapped from the rest of the organization, industrial and automated control systems are now dependent on and heavily influenced by many other systems: the business or enterprise network, new communications infrastructures that are integrated with power systems (i.e., the smart grid), new technologies, tools, etc. The result is that control systems must be assessed (at least for security purposes) as a dynamic system. Without sufficient planning for outside influences and unforeseen growth, the best-laid plans can fail after implementation.

CAUTION

When implementing new security products, proper sizing and configuration of those products is critical. However, most vendors rate products differently. Similar products may be marketed using entirely different metrics, making it difficult to choose the correct tool for the job.

Especially in an industrial network (where there is likely a compliance requirement to thoroughly test new assets in any case), insist on a trial of significant length to ensure that the product is sufficient for the scope and scale of the network it will be deployed in. Because it is also difficult to effectively measure the various necessary qualities of a network, this trial should be performed in a full test network environment that replicates the production network as closely as possible. Such a test environment should be maintained in its own isolated and secured enclave, and to the greatest degree possible it should contain the same network assets and systems that are in production environments. The use of virtual machines (VMs) can simplify the process of establishing test networks by enabling the easy reimaging of certain systems. However, while certain systems may be able to be virtualized for simplicity, due to the nature of many industrial assets, at least a partly built physical test environment will likely be required.

Project-Limited Thinking

Two common axioms in information security are "Security is a Process, not a Product" and "Every door is a back door." Taking this advice under consideration, security cannot be treated as a onetime project, with limited scope and definable goals. Rather, security policies should be continuously assessed and reassessed as part of the global information technology strategy. Because of the relatively static nature of industrial operations, there may be an inclination to implement perimeter defenses, lock down configurations, and then consider the job to be finished.

However, even if the industrial network(s) remain unchanged, the networks that surround them and interact with them are likely to evolve, often at a rapid pace. New tools and technologies will be implemented that could impact the industrial network in ways that were never considered. The sudden introduction of wireless networking in Smart Phones, for example, can suddenly introduce traffic into an industrial network as the phone attempts to discover wireless access points and negotiate connections; in this example, an executive touring a plant floor with a Smart Phone in his or her pocket has introduced unexpected change into the industrial network.

Especially in industrial networks where the enterprise or business network and the operational networks may be managed by separate groups, the point of demarcation between the two networks needs constant scrutiny as well. For example, the firewall between the SCADA DMZ and the Business Network might allow certain traffic on a certain port as part of a legitimate policy. Later, a new system or application could be introduced on the Business Network that uses the same port for a different function. Likewise, the enabled system on the Business Network (the firewall rule should explicitly define the source and destination IPs that are allowed to communicate) could be misconfigured, new software could be installed, or some other change made that ultimately violates the originally established policy—after all, business networks are more dynamic than industrial networks by nature. If both networks are not continuously assessed, these changes may go unnoticed, invalidating the security perimeter.

It is therefore necessary to think about industrial network security as broadly as possible, with full consideration of all outside influencing factors—even if those factors are outside of the responsibilities of the industrial network operator.

Insufficiently Sized Security Controls

The last pitfall to be discussed involves the improper implementation of automated security systems. These systems include specific security devices—such as a firewall, IPS, industrial protocol filter, application monitor, or whitelisting agent—as well as systems designed to provide the required situational awareness, log retention, and reporting—such as a SIEM or Log Management system. In any case, the tool may not be big enough to complete the required task. For firewalls and IPSs, it might be an issue of throughput, latency, or the completeness of the rule set. For situational awareness tools, it might be a limitation on the types of logs, the amount of logs, or the rate at which logs can be assessed. The result is the same: the system will eventually fail.

This pitfall occurs partly because of the difficulties in measuring the required performance, especially in the case of situational awareness tools. The types of devices, the properties of the network(s), how the network is used, and other factors all influence the rate at which event logs are produced.[6] As the rate of new events increases, the need to store more log files increases, as does the hardware requirements of the system itself, so that collected logs can be effectively managed, analyzed, reported against, etc. Additional difficulties arise during times when an incident occurs. If there is an attempted breach, an unauthorized change in configurations, or some other policy violation, all properly configured security devices will begin to generate an increased number of logs. In the event of a malware infection or an Advanced Persistent Threat, the incident can be prolonged, extending the spike in event volume into an ongoing plateau that can quickly overburden systems that have not been designed with sufficient headroom for growth.[7]

SUMMARY

With the proper intentions, a well-informed network security administrator can plan, implement, and execute best-in class security measures for any industrial network. By following the basic guidelines presented in this book, as well as those provided by various compliance standards, regulatory guides, and other publications referenced in the Appendices of this book, industrial networks will be more secure, protecting the valuable—and often critical—automated processes that they operate.

ENDNOTES

1. J. Larson, Idaho National Laboratories, Control systems at risk: sophisticated penetration testers show how to get through the defenses, in: Proc. 2009 SANS European SCADA and Process Control Security Summit, October, 2009.
2. J. Brodsky, A. McConnell, M. Cajina, D. Peterson, Security and reliability of wireless LAN protocol stacks used in control systems, in: Proc. SCADA Security Scientific Symposium (S4), Kenexis Security Corporation, 2010, Digital Bond Press.
3. J. Pollet, R. Tiger, Electricity for free? The dirty underbelly of SCADA and Smart Meters, in: Proc. 2010 BlackHat Technical Conference, July, 2010.
4. C. Sullo, CIRT.net. Default passwords. <http://cirt.net/passwords>, October 4, 2007 (cited: March 15, 2011).
5. I. Spaanjaars, Regular expression for a strong password. <http://imar.spaanjaars.com/297/regular-expression-for-a-strong-password>, May 14, 2004 (cited: March 15, 2011).
6. J.M. Butler, Benchmarking security information event management (SIEM), The SANS Institute Analytics Program, February, 2009.
7. Ibid.

Glossary

Active Directory Microsoft's Active Directory (AD) is a centralized directory framework for the administration of network devices and users, including user identity management, and authentication services. AD utilizes the Lightweight Directory Access Protocol (LDAP) along with domain and authentication services.

Advanced Persistent Threat The Advanced Persistent Threat (APT) refers to a class of cyber threat designed to infiltrate a network, remain persistent through evasion and propagation techniques. APTs are typically used to establish and maintain an external command and control channel through which the attacker can continuously exfiltrate data.

Anti-Virus Anti-Virus (AV) systems inspect network and/or file content for indications of infection by malware. Signature-based AV works by comparing file contents against a library of defined code signatures; if there is a match the file is typically quarantined to prevent infection, at which point the option to clean the file maybe available.

Application Monitor/Application Data Monitor An application content monitoring system which functions much like an Intrusion Detection System, only performing deep inspection of a session rather than of a packet, so that application contents can be examined at all layers of the OSI model, from low level protocols through application documents, attachments, etc. Application Monitoring is useful for examining industrial network protocols for malicious content (malware).

Application Whitelisting Application Whitelisting (AW) is a form of whitelisting intended to control which executable files (applications) are allowed to operate. AW systems typically work by first establishing the "whitelist" of allowed applications, after which point any attempt to execute code will be compared against that list. If the application is not allowed, it will be prevented from executing. AW often operates at low levels within the kernel of the host operating system.

APT See **Advanced Persistent Threat**.

Asset An asset is any device used within an industrial network.

Attack Surface The attack surface of a system or asset refers to the collectively exposed portions of that system or asset. A large attack surface means that there are many exposed areas that an attack could target, while a small attack surface means that the target is relatively unexposed.

Attack Vector An attack vector is the direction(s) through which an attack occurs, often referring to specific vulnerabilities that are used by an attacker at any given stage of an attack.

auditd auditd is the auditing component of the Linux Auditing System, responsible for writing audit events to disk. The Linux Auditing System is a useful tool for monitoring file access and file integrity in Linux systems.

AV See Anti-Virus.

AWL See Application Whitelisting.

Backchannel A backchannel typically refers to a communications channel that is hidden or operates "in the background" to avoid to detection, but is also used in reference to hidden or covert communications occurring back towards the originating sender, that is, malware hidden in the return traffic of a bidirectional communication.

Blacklisting (see "**Whitelisting**") Blacklisting refers to the technique of defining known malicious behavior, content, code, etc. Blacklists are typically used for threat detection, comparing network traffic, files, users, or some other quantifiable metric against a relevant blacklist. For example, an Intrusion Prevention System (IPS) will compare the contents of network packets against blacklists of known malware, indicators of exploits, and other threats so that offending traffic (i.e., packets that match a signature within the blacklist) can be blocked.

CDA See **Critical Digital Asset**.

CFATS The Chemical Facility Anti-Terrorism Standard, established by the United States Department of Homeland Security to protect the manufacture, storage and distribution of potentially hazardous chemicals.

Compensating Controls The term "compensating controls" is typically used within regulatory standards or guidelines to indicate when an alternative method than those specifically address by the standard or guideline is used.

Control Center A control center typically refers to an operations center where a control system is managed. Control centers typically consist of SCADA and HMI systems that provide interaction with industrial/automated processes.

Correlated Event A correlated event is a larger pattern match consisting of two or more regular logs or events, as detected by an event correlation system. For example, a combination of a network scan event (as reported by a firewall) followed by an injection attempt against an open port (as reported by an IPS) can be correlated together into a larger incident; in this example, an attempted reconnaissance and exploit. Correlated events maybe very simple or very complex, and can be used to detect a wide variety of more sophisticated attack indicators.

Critical Cyber Asset A critical cyber asset is a cyber asset that is itself responsible for performing a critical function, or directly impacts an asset that performs a critical function. The term "critical cyber asset" is used heavily within NERC reliability standards for Critical Infrastructure Protection.

Critical Digital Asset A "critical digital asset" is a digitally connected asset that is itself responsible for performing a critical function, or directly impacts an asset that performs a critical function. The term "critical digital asset" is used heavily within NRC regulations and guidance documents. Also see: **Critical Cyber Asset**.

Critical Infrastructure Any infrastructure whose disruption could have severe impact on a nation or society. In the United States, Critical Infrastructures are defined by the Homeland Security Presidential Directive Seven as: Agriculture and Food; Banking and Finance; Chemical; Commercial Facilities; Critical Manufacturing; Dams; Defense Industrial Base; Drinking Water and Water Treatment Systems; Emergency Services; Energy; Government Facilities; Information Technology; National Monuments and Icons; Nuclear Reactors, Materials, and Waste; Postal and Shipping; Public Health and Healthcare; Telecommunications; and Transportation Systems.

Cyber Asset A digitally connected asset; that is, an asset that is connected to a routable network, that is, a Host. The term Cyber Asset is used within the NERC reliability standards, which defines a Cyber Asset as: any Asset connected to a routable network within a control system; any Asset connected to a routable network outside of the control system; and/or any Asset reachable via dial-up.[1]

DAM See **Database Activity Monitor**.

Data Diode A data diode is a “one way” data communication device, often consisting of a physical-layer unidirectional limitation. Using only one half of a fiber optic “transmit/receive” pair would enforce unidirectional communication at the physical layer, while proper configuration of a network firewall could logically enforce unidirectional communication at the network layer.

Database Activity Monitor A Database Activity Monitor (DAM) monitors database transactions, including SQL, DML and other database commands and queries. A DAM may be network- or host-based. Network-based DAMs monitor database transactions by decoding and interpreting network traffic, while host-based DAMs provide system-level auditing directly from the database server. DAMs can be used for indications of malicious intent (e.g.,SQL injection attacks), fraud (e.g., the manipulation of stored data) and/or as a means of logging data access for systems that do not or cannot produce auditable logs.

Database Monitor See **Database Activity Monitor.**

DCS See **Distributed Control System**.

Deep Packet Inspection The process of inspecting a network packet all the way to the application layer (layer 7) of the OSI model. That is, past datalink, network or session headers to inspect all the way into the payload of the packet. Deep Packet Inspection is used by most intrusion detection and prevention systems (IDS/IPS), newer firewalls, and other security devices.

Distributed Control System An industrial control system deployed and controlled in a distributed manner, such that various distributed control systems or processes are controlled individually. See also: **Industrial Control System**.

DPI See **Deep Packet Inspection**.

Electronic Security Perimeter An Electronic Security Perimeter (ESP) refers to the demarcation point between a secured enclave, such as a control system, and a less trusted network, such as a business network. The ESP typically includes the devices, such as firewalls, IDS, IPS, Industrial Protocol Filters, Application Monitors, and similar devices, that secure the demarcation points.

Enclave A logical grouping of assets, systems and/or services that defines and contains one (or more) functional groups. Enclaves represent network “zones” that can be used to isolate certain functions in order to more effectively secure them.

Enumeration Enumeration is the process of identifying valid identities of devices and users in a network; typically as an initial step in a network attack process. Enumeration allows an attacker to identify valid systems and/or accounts that can then be targeted for exploitation or compromise.

ESP See **Electronic Security Perimeter**.

Ethernet/IP Ethernet/IP is a real-time Ethernet protocol supporting the Common Industrial Protocol (CIP), for use in industrial control systems.

Event An event is a generic term referring to any datapoint of interest, typically alerts that are generated by security devices, logs produced by systems and applications, alerts produced by network monitors, etc.

Finger The finger command is a network tool that provides detailed information about a user.

Function Code Function Codes refer to various numeric identifiers used within industrial network protocols for command and control purposes. For example, a function code may represents a request from a Master device to a Slave device(s), such as a request to read a register value, to write a register value, to restart the device, etc.

HIDS Host IDS. A Host Intrusion Detection System, which detects intrusion attempts via a software agent running on a specific host. A HIDS detects intrusions by inspecting packets and matching the contents against defined patterns or "signatures" that indicate malicious content, and produce an alert.

HIPS Host IPS. A Host Intrusion Prevention System, which detects and prevents intrusion attempts via a software agent running on a specific host. Like a HIDS, a HIPS detects intrusions by inspecting packets and matching the contents against defined patterns or "signatures" that indicate malicious content. Unlike a HIDS, a HIPS is able to perform active prevention by dropping the offending packet(s), resetting TCP/IP connections, or other actions in addition to passive alerting and logging actions.

HMI A Human Machine Interface (HMI) is the user interface to the processes of an industrial control system. An HMI effectively translates the communications to and from PLCs, RTUs, and other industrial assets to a human-readable interface, which is used by control systems operators to manage and monitor processes.

Homeland Security Presidential Directive Seven The United States Homeland Security Presidential Directive Seven (HSPD-7) defines the 18 critical infrastructures within the US, as well as the governing authorities responsible for their security.

Host A host is a computer connected to a network: that is, a Cyber Asset. The term differs from an Asset in that hosts typically refer to computers connected to a routable network using the TCP/IP stack—that is, most computers running a modern operating system and/or specialized network servers and equipment—where an Asset refers to a broader range of digitally connected devices, and a Cyber Asset refers to any Asset that is connected to a routable network.[2]

HSPD-7 See **Homeland Security Presidential Directive Seven**.

IACS Industrial Automation Control System. See **Industrial Control System**.

IAM See **Identity Access Management**.

ICCP See **Inter Control Center Protocol**.

ICS See **Industrial Control System**.

Identity Access Management Identity Access Management refers to both: the process of managing user identities and user accounts, as well as related user access and authentication activities within a network; and a category of products designed to centralize and automate those functions.

IDS Intrusion Detection System. Intrusion Detection Systems perform deep packet inspection and pattern matching to compare network packets against known "signatures" of malware or other malicious activity, in order to detect a possible network intrusion. IDS operates passively by monitoring networks either in-line or on a tap or span port, and providing security alerts or events to a network operator.

IEC See **International Electrotechnical Commission**.

IED See **Intelligent Electronic Device**.

Industrial Control System An Industrial Control System (ICS) refers to the systems, devices, networks, and controls used to operate and/or automate an industrial process. See also: **Distributed Control System**.

Intelligent Electronic Device An Intelligent Electronic Device (IED) is an electronic component—such as a regulator, circuit control, etc.—that has a microprocessor and is able to communicate, typically digitally using fieldbus, real-time Ethernet or other industrial protocols.

Inter Control Center Protocol The Inter Control Center Protocol (ICCP) is a real-time industrial network protocol designed for wide area intercommunication between two or more control centers. ICCP is an internationally recognized standard published by the International Electrotechnical Commission (IEC) as IEC 60870-6. ICCP is also referred to as the Telecontrol Application Service Element-2 or TASE.2.

International Electrotechnical Commission The International Electrotechnical Commission (IEC) is an international standards organization that develops standards for the purposes of consensus and conformity among international technology developers, vendors, and users.

International Standards Organization The International Standards Organization (ISO) is a network of standards organizations from over 160 countries, which develops and publishes standards covering a wide range of topics.

IPS Intrusion Prevention System. Intrusion Protection Systems perform the same detection functions of an IDS, with the added capability to block traffic. Traffic can typically be blocked by dropping the offending packet(s), or by forcing a reset of the offending TCP/IP session. IPS works in-line, and therefore may introduce latency.

ISO See **International Standards Organization**.

LDAP See **Lightweight Directory Access Protocol**.

Lightweight Directory Access Protocol The Lightweight Directory Access Protocol (LDAP) is a standard published under IETF RFC 4510, which defines a standard process for accessing and utilized network-based directories. LDAP is used by a variety of directories and Identity Access Management (IAM) systems.

Log A log is a file used to record activities or events, generated by a variety of devices including computer operating systems, applications, network switches and routers, and virtually any computing device. There is no standard for the common format or structure of a log.

Log Management Log Management is the process of collecting and storing logs for purposes of log analysis and data forensics, and/or for purposes of regulatory compliance and accountability. Log Management typically involves collection of logs, some degree of normalization or categorization, and both short-term storage (for analysis) and long-term storage (for compliance).

Log Management System A system or appliance designed to simplify and/or automate the process of Log Management. See also: **Log Management**.

Master Station A Master station is the controlling asset or host involved in an industrial protocol communication session. The Master station is typically responsible for timing, synchronization, and command and control aspects of an industrial network protocol.

Metasploit Metasploit is a commercial exploit package, used for penetration testing.

Modbus Modbus is the Modicon Bus protocol, used for intercommunication between industrial control assets. Modbus is a flexible Master/Slave command and control protocol available in several variants including Modbus ASCII, Modbus RTU, Modbus TCP/IP, and Modbus Plus.

Modbus ASCII A Modbus variant that uses ASCII characters rather than binary data representation.

Modbus Plus A Modbus extension that operates at higher speeds, which remains proprietary to Shneider Electric.

Modbus RTU A Modbus variant that uses binary data representation.

Modbus TCP A Modbus variant that operates over TCP/IP.

NAC See **Network Access Control**.

NEI The Nuclear Energy Institute is an organization dedicated to and governed by the United States nuclear utility companies.

NERC See **North American Electric Reliability Corporation**.

NERC CIP The North American Electric Reliability Corporation reliability standard for Critical Infrastructure Protection.

Network Access Control Network Access Control (NAC) provides measures of controlling access to the network, using technologies such as 802.1X (port network access control) to require authentication for a network port to be enabled, or other access control methods.

Network Whitelisting see "**Whitelisting**".

NIDS Network IDS. A Network Intrusion Detection System detects intrusion attempts via a network interface card, which connects to the network either in-line or via a span or tap port.

NIPS Network IPS. A Network Intrusion Prevention Detection System detects and prevents intrusion attempts via a network-attached device using two or more network interface cards to support inbound and outbound network traffic, with optional bypass interfaces to preserve network reliability in the event of a NIPS failure.

NIST The National Institute of Standards and Technology is a non-regulatory federal agency within the United States Department of Commerce, whose mission is to promote innovation through the advancement of science, technology, and standards. NIST provides numerous research documents and recommendations (the "Special Publication 800 series") around information technology security.

nmap Nmap or "Network Mapper" is a popular network scanner distributed under GNU General Public License GPL-2 by nmap.org.

North American Electric Reliability Corporation The North American Electric Reliability Corporation is an organization that develops and enforces reliability standards for and monitors the activities of the bulk electric power grid in North America.

NRC See **Nuclear Regulatory Commission**.

Nuclear Regulatory Commission The United States Nuclear Regulatory Commission (NRC) is a five-member Presidentially appointed commission responsible for the safe use of radioactive materials including but not limited to nuclear energy, nuclear fuels, radioactive waste management, and the medical use of radioactive materials.

OSSIM OSSIM is an Open Source Security Information Management project, whose source code is distributed under GNU General Public License GPL-2 by AlienVault.

Outstation An outstation is the DNP3 slave or remote device. The term outstation is also used more generically as a remote SCADA system, typically interconnected with central SCADA systems by a Wide Area Network.

PCS Process Control System. See **Industrial Control System**.

Pen test A Penetration Test. A method for determining the risk to a network by attempting to penetrate its defenses. Pentesting combines vulnerability assessment techniques with evasion techniques and other attack methods to simulate a "real attack."

PLC See **Programmable Logic Controller**.

Process Control System See **Industrial Control System**.

Profibus Profibus is an industrial fieldbus protocol defined by IEC standard 61158/IEC 61784-1.

Profinet Profinet is an implementation of Profibus designed to operate in real time over Ethernet.

Programmable Logic Controller A Programmable Logic Controller (PLC) is an industrial device that uses input and output relays in combination with programmable logic in order to build an automated control loop. PLCs commonly use Ladder Logic to read inputs, compare values against defined set points, and (potentially) write to outputs.

Project Aurora A research project that demonstrated how a cyber attack could result in the explosion of a generator.

RBPS Risk Based Performance Standards are recommendations for meeting the security controls required by the Chemical Facility Anti-Terrorism Standard (CFATS), written by DHS.

Red Network A "red network" typically refers to a trusted network, in contrast to a "black network" which is less secured. When discussing unidirectional communications in critical networks, traffic is typically only allowed outward from the red network to the black network, to allow supervisory data originating from critical assets to be collected and utilized by less secure SCADA systems. In other use cases, such as data integrity and fraud prevention, traffic may only be allowed from the black network into the red network, to prevent access to classified data once it has been stored.

Remote Terminal Unit A Remote Terminal Unit (RTU) is a device combining remote communication capabilities with programmable logic for the control of processes in remote locations.

RTU See **Remote Terminal Unit**.

SCADA See **Supervisory Control and Data Acquisition**.

SCADA-IDS SCADA aware Intrusion Detection System. An IDS system designed for use in SCADA and ICS networks. SCADA-IPS devices support pattern matching against the specific protocols and services used in control systems, such as Modbus, ICCP, DNP3, and others. SCADA-IDS are passive, and are therefore suitable for deployment within a control system, as they do not introduce any risk to control system reliability.

SCADA-IPS SCADA aware Intrusion Prevention System is an IPS system designed for use in SCADA and ICS networks. SCADA-IPS devices support pattern matching against the specific protocols and services used in control systems, such as Modbus, ICCP, DNP3, and others. SCADA-IPS are active and can block or blacklist traffic, making them most suitable for use at control system perimeters. SCADA-IPS are not typically deployed within a control system for fear of a false-positive disrupting normal control system operations.

Security Information and Event Management Security Information and Event Management (SIEM) combines Security Information Management (SIM or Log Management) with Security Event Management (SEM) to provide a common centralized system for managing network threats and all associated information and context.

SERCOS III SERCOS III is the latest version of the Serial Real-time Communications System, a real-time Ethernet implementation of the popular SERCOS fieldbus protocols.

Set Points Set points are defined values signifying a target metric against which programmable logic can operate. For example, a set point may define a high temperature range, or the optimum pressure of a container, etc. By comparing set points against sensory input, automated controls can be established. For example, if the temperature in a furnace reaches the set point for the maximum ceiling temperature, reduce the flow of fuel to the burner.

SIEM See **Security Information and Event Management**.

Situational Awareness Situational Awareness is a term used by the National Institute of Standards and Technology (NIST) and others to indicate a desired state of awareness within a network in order to identify and respond to network-based attacks. The term is a derivative of the military command and control process of perceiving a threat, comprehending it, making a decision and taking an action in order to maintain the security of the environment. Situational Awareness in network security can be obtained through network and security monitoring (perception), alert notifications (comprehension), security threat analysis (decision making), and remediation (taking action).

Smart-Listing A term referring to the use of both blacklisting and whitelisting technologies in conjunction with a centralized intelligence system such as a SIEM in order to dynamically adapt common blacklists in response to observed security event activities. See also: **Whitelisting** and **Blacklisting**.

Stuxnet An advanced cyber attack against an industrial control system, consisting of multiple zero-day exploits used for the delivery of malware that then targeted and infected specific industrial controls for the purposes of sabotaging an automated process. Stuxnet is widely regarded as the first cyber attack to specifically target an industrial control system.

Supervisory Control And Data Acquisition Supervisory Control and Data Acquisition (SCADA) refers to the systems and networks that communicate with industrial control systems to provide data to operators for supervisory purposes, as well as control capabilities for process management.

TASE.1 See **Telecontrol Application Service Element-1**.

TASE.2 See **Telecontrol Application Service Element-2**.

Technical Feasibility/Technical Feasibility Exception (TFE) The term "Technical Feasibility" is used in the NERC CIP reliability standard and other compliance controls to indicate where a required control can be reasonably implemented. Where the implementation of a required control is not technically feasible, a Technical Feasibility Exception can be documented. In most cases, a TFE must detail how a compensating control is used in place of the control deemed to not be feasible.

Telecontrol Application Service Element-1 The initial communication standard used by the ICCP protocol. Superseded by **Telecontrol Application Service Element-2**.

Telecontrol Application Service Element-2 The Telecontrol Application Service Element-2 standard or TASE.2 refers to the ICCP protocol. See also: **Inter Control Center Protocol**.

Unidirectional Gateway A network gateway device that only allows communication in one direction, such as a Data Diode. See also: **Data Diode**.

User Whitelisting The process of establishing a "whitelist" of known valid user identities and/or accounts, for the purpose of detecting and/or preventing rogue user activities. See also: **Application Whitelisting**.

VA See **Vulnerability Assessment**.

Vulnerability A vulnerability refers to a weakness in a system that can be utilized by an attacker to damage the system, obtain unauthorized access, execute arbitrary code, or otherwise exploit the system.

Vulnerability Assessment The process of scanning networks to find hosts or assets, and probing those hosts to determine vulnerabilities. Vulnerability Assessment can be automated using a Vulnerability Assessment Scanner, which will typically examine a host to determine the version of the operating system and all running applications, which can then be compared against a repository of known software vulnerabilities to determine where patches should be applied.

Whitelists Whitelists refers to defined lists of "known good" items: users, network addresses, applications, etc., typically for the purpose of exception-based security where any item not explicitly defined as "known good" results in a remediation action (e.g., alert, block, etc.). Whitelists contrast blacklists, which define "known bad" items.

Whitelisting Whitelisting refers to the act of comparing an item against a list of approved items for the purpose of assessing whether it is allowed or should be blocked. Typically referred to in the context of Application Whitelisting, which prevents unauthorized applications from executing on a host by comparing all applications against a whitelist of authorized applications.

Zone A zone refers to a logical boundary or enclave containing assets of like function and/ or criticality, for the purposes of facilitating the security of common systems and services. See also: **Enclave**.

ENDNOTES

1. North American Reliability Corporation. Standard CIP-002-4 – Cyber Security – Critical Cyber Asset Identification. <http://www.nerc.com/files/CIP-002-4.pdf>, February 3, 2011 (cited: March 3, 2011).
2. Ibid.

This page intentionally left blank

APPENDIX A

Protocol Resources

INFORMATION IN THIS CHAPTER:

- Modbus Organization
- DNP3 Users Group
- OPC Foundation
- Common Industrial Protocol/ODVA

While industrial network protocols were covered at a high level in Chapter 4, “Industrial Network Protocols”, fully understanding how these protocols work will facilitate the assessment and security of industrial networks. The following organizations provide in-depth documentation and support for the four leading industrial network protocols: Modbus, Distributed Network Protocol (DNP3), OPC, and Common Industrial Protocol (CIP).

MODBUS ORGANIZATION

- The Modbus Organization is a group consisting of independent users and automation device manufacturers who manage the development and use of the Modbus protocols. Their website contains information about the Modbus protocols, as well as technical resources for development, integration and testing of Modbus. It also includes directories of Modbus suppliers and industrial devices utilizing Modbus (http://www.modbus.org/).

DNP3 USERS GROUP

- The DNP Users Group is a nonprofit organization that maintains and promotes the Distributed Network Protocol (DNP3). Their website provides documentation on the uses and benefits of DNP3, as well as technical documents and conformance testing. It also includes member directories and listings of all conformance tested products (http://www.dnp.org).

OPC FOUNDATION

- The OPC Foundation is an organization that maintains the open specifications of the OPC protocol, in an effort to standardize and ensure interoperability of process data communications. Their site includes the latest resources for OPC Classic, OPC UA, and OPC XI (.NET). It provides whitepapers, sample code, technical specifications and software development kits. The website also includes member directories and product lists, as well as technical support, webinars, and other resources (http://www.opcfoundation.org/).

COMMON INDUSTRIAL PROTOCOL/ODVA

- ODVA is an international association made of automation companies, which manages the development of DeviceNet, EtherNet/IP, CompoNet, and ControlNet protocols utilizing the Common Industrial Protocol (CIP). The ODVA website provides technical specifications, conformance testing policies, training and other resources. It also includes member and product directories (http://www.odva.org).

APPENDIX B

Standards Organizations

INFORMATION IN THIS CHAPTER:

- North American Reliability Corporation (NERC)
- The United States Nuclear Regulatory Commission (NRC)
- United States Department of Homeland Security (DHS)
- International Standards Association (ISA)
- The International Standards Organization (ISO) and International Electrotechnical Commission (IEC)

While a limited selection of regulatory standards and compliance controls have been discussed in Chapter 10, “Standards and Regulations,” there are many additional controls that are either mandated or recommended by North American Reliability Corporation (NERC), the United States Nuclear Regulatory Commission (NRC), United States Department of Homeland Security (DHS), International Standards Association (ISA), and the International Standards Organization/International Electrotechnical Commission (ISO/IEC). The following organizations provide useful resources, including access to the most recent versions of compliance standards documents.

NORTH AMERICAN RELIABILITY CORPORATION (NERC)

The North American Reliability Corporation is tasked by the Federal Energy Regulatory Commission (FERC) to ensure the reliability of the bulk power system in North America. NERC enforces several reliability standards, including the reliability standard for Critical Infrastructure Protection (NERC CIP). In addition to these standards, NERC publishes information, assessments and trends concerning bulk power reliability, including research of reliability events as they occur.

The NERC CIP standards are comprised of nine standards documents, all of which are available from NERC’s website at: http://www.nerc.com/page.php?cid=2|20.

THE UNITED STATES NUCLEAR REGULATORY COMMISSION (NRC)

The United States Nuclear Regulatory Commission is responsible for the safe use of radioactive materials, including nuclear power generation and medical applications of radiation. The NRC publishes standards and guidelines for Information Security, as well as general information and resources about nuclear materials and products, nuclear waste materials, and other concerns.

NRC Title 10 CFR 73.54

NRC Title 10 of the Code of Federal Regulations, Part 73.54 regulates the “Protection of digital computer and communication systems and networks” used in member Nuclear Facilities. More information on CFR 73.54 is available from NRC’s website at: http://www.nrc.gov/reading-rm/doc-collections/cfr/part073/part073-0054.html.

NRC RG 5.71

The United States Nuclear Regulatory Commission’s Regulatory Guide 5.71 offers guidance on how to protect digital computer and communication systems and networks. RG 5.71 is not a regulatory standard but rather guidance on how to comply with the standard, which is Title 10 of the Code of Federal Regulations, Part 73.54. Information on RG 5.71 is available from NRC’s website at: http://nrc-stp.ornl.gov/slo/regguide571.pdf.

UNITED STATES DEPARTMENT OF HOMELAND SECURITY (DHS)

The Department of Homeland Security’s (NHS) mission is to protect the United States from a variety of threats including (but not limited to) counter-terrorism and cyber security. One area where cyber security concerns and anti-terrorism overlap is in the protection of chemical facilities, which are regulated under the Chemical Facilities Anti-Terrorism Standards (CFATSs). CFATS includes a wide range of security controls, which can be measured against a set of Risk-Based Performance Standards (RBPSs).

Chemical Facilities Anti-Terrorism Standard

The Chemical Facility Anti-Terrorism Standards (CFATSs) are published by the United States Department of Homeland Security, and they encompass many areas of chemical manufacturing, distribution and use including cyber security concerns. More information on CFATS can be found on the DHS’s website at: http://www.dhs.gov/files/laws/gc_1166796969417.shtm.

CFATS Risk-Based Performance Standards

The United States Department of Homeland Security also publishes recommendations in the form of Risk-Based Performance Standards (RBPSs) for CFATS.

These standards provide guidance for the compliance to the Chemical Facility Anti-Terrorism Standards. More information on the CFATS RBPS can be found on the DHS's website at: http://www.dhs.gov/xlibrary/assets/chemsec_cfats_riskbased_performance_standards.pdf.

INTERNATIONAL STANDARDS ASSOCIATION (ISA)

The International Standards Association (ISA) and the American National Standards Institute (ANSI) have published three documents concerning industrial network security under the umbrella of ISA-99. These documents are: ANSI/ISA-99.02.01-2009, "Security for Industrial Automation and Control Systems: Establishing an Industrial Automation and Control Systems Security Program"; ANSI/ISA-99.00.01-2007, "Security for Industrial Automation and Control Systems: Concepts, Terminology and Models"; and ANSI/ISA-TR99.00.01-2007, "Security Technologies for Manufacturing and Control Systems."

These documents, as well as additional information and resources relevant to ISA-99 are available at the ISA website, at: http://www.isa.org/MSTemplate.cfm?MicrositeID=988&CommitteeID=6821.

THE INTERNATIONAL STANDARDS ORGANIZATION (ISO) AND INTERNATIONAL ELECTROTECHNICAL COMMISSION (IEC)

The International Standards Organization (ISO) and the International Electrotechnical Commission (IEC) produced the ISO/IEC 27002:2005 standard for "Information technology—Security techniques—Code of practice for information security management." While ISO/IEC 27002:2005 does not apply exclusively to SCADA or industrial process control networks, it provides a useful basis for implementing security in industrial networks, and is also heavily referenced by a variety of international standards and guidelines.

More information on the ISO/IEC 27002:2005 can be found on the ISO website at: http://www.iso.org/iso/catalogue_detail?csnumber=50297.

This page intentionally left blank

INFORMATION IN THIS CHAPTER:

- National Institute of Standards and Technology, Special Publications 800 Series

The National Institute of Standards and Technology (NIST), Special Publications (SP) 800 series present security best practices and guidelines resulting from the Information Technology Lab's research. NIST provides over 100 specialized documents, providing specific information security guidance for a wide range of industries and use cases.

NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY, SPECIAL PUBLICATIONS 800 SERIES

Several of NIST SP 800 documents, listed below, address concepts of information and system security that are highly relevant to industrial network security. The full index of SP 800 documents, including those mentioned here, can be found online at http://csrc.nist.gov/publications/PubsSPs.html.

- SP 800-12, An Introduction to Computer Security: The NIST Handbook, October 1995.
- SP 800-30, Risk Management Guide for Information Technology Systems, July 2002.
- SP 800-40, Version 2, Creating a Patch and Vulnerability Management Program, November 2005.
- SP 800-41 (Draft), Guidelines on Firewalls and Firewall Policy, July 2008.
- SP 800-53, Guide for Assessing the Security Controls in Federal Information Systems: Building Effective Security Assessment Plans, July 2008.
- SP 800-60, Revision 1, Guide for Mapping Types of Information and Information Systems to Security Categories, August 2008.
- SP 800-82 (Final Public Draft), Guide to Industrial Control Systems (ICS) Security, September 2008.
- SP 800-92, Guide to Computer Security Log Management, September 2006.
- SP 800-94, Guide to Intrusion Detection and Prevention Systems (IDPS), February 2007.
- SP 800-113, Guide to SSL VPNs, July 2008.
- SP 800-118 (Draft), Guide to Enterprise Password Management, April 2009.
- SP 800-128 (Draft), Guide for Security Configuration Management of Information Systems, August 2009.

This page intentionally left blank

Index