Common Cyber Security Vulnerabilities Observed in DHS Industrial Control Systems Assessments

July 2009

Homeland
Security

Control Systems Security Program

National Cyber Security Division

EXECUTIVE SUMMARY

The U.S. Department of Homeland Security (DHS) National Cyber Security Division's Control Systems Security Program (CSSP) performs cyber security assessments of Industrial Control Systems (ICS) to help industry improve the security of the ICS used in critical infrastructures throughout the United States. A key part of this mission is the assessment of ICS to identify vulnerabilities that could put critical infrastructures at risk from a cyber attack.

This report presents results from 15 ICS assessments performed under the CSSP from 2004 through 2008. Although information found in individual stakeholder reports is protected from disclosure, the security of the critical infrastructure as a whole can be improved by sharing information on common security problems with those in industry responsible for developing and maintaining ICS. For this reason, vulnerability information was collected, analyzed, and organized in a way that the most prevalent issues could be identified and mitigated by those responsible for individual systems without disclosing the identity of the associated ICS product.

Common vulnerabilities were derived from correlated vulnerabilities identified by the 15 ICS assessments and grouped into general categories. Poor network protocol implementations, information disclosure, and authentication problems contain the most report findings. The common assessment findings are described under their respective categories along with specific examples and recommendations. General recommendations are based on empirical knowledge gained through performing security assessments on ICS products and operational installations.

The assessment findings were also categorized by where the security problems occur within the ICS software, hardware, and network components. Proprietary ICS applications had the highest vulnerability count because most CSSP assessments to date have been on new vendor product releases.

This information will benefit vendors, asset owners, and other stakeholders responsible for securing the systems that control the nation's critical infrastructure. System vendors learn of common weaknesses in ICS applications, services, and protocols, and how to better secure their products. Asset owners can evaluate possible weaknesses in their installed system configurations and learn how they can fix or mitigate them with secure firewall configurations, intrusion detection systems, and network architectures. Understanding the types of vulnerabilities commonly found and mitigating them can serve to help protect the systems currently in development as well as those already installed in critical infrastructure applications.

This report represents a steadily growing understanding of ICS security issues and methods for mitigating current vulnerabilities as well as new technologies and approaches being developed in response to ICS security challenges. The assessment effort is expanding to new technologies as CSSP seeks a continuing understanding of the control systems being planned and deployed.

CONTENTS

EXECUTIVE SUMMARY iii

ACRONYMS ix

1. INTRODUCTION 1

2. ASSESSMENT METHODOLOGIES 2
   - 2.1 Approach Limitations 3
   - 2.2 Impact on ICS Security 5

3. COMMON ICS ASSESSMENT FINDINGS 6
   - 3.1 CSSP Findings Mapped to the ISA SP99 Reference Model Levels 10
     - 3.1.1 Level 4: Enterprise Systems 11
     - 3.1.2 Level 3: Operations Management 12
     - 3.1.3 Level 2: Supervisory Control 13
     - 3.1.4 Level 1: Local or Basic Control 14

4. UNDERSTANDING COMMON ICS VULNERABILITIES 16
   - 4.1 Common ICS Software Vulnerabilities 16
     - 4.1.1 Poor Code Quality 16
     - 4.1.2 Vulnerable Web Services 17
     - 4.1.3 Poor Network Protocol Implementations 18
     - 4.1.4 Information Disclosure 22
     - 4.1.5 Poor Patch Management 25
     - 4.1.6 ICS Performs Weak Authentication 26
     - 4.1.7 Least User Privileges Violation 27
     - 4.1.8 Summary of Common ICS Software Vulnerabilities 27
   - 4.2 Common ICS Configuration (Implementation) Vulnerabilities 29
     - 4.2.1 Poor Patch Management 29
     - 4.2.2 Weak User Authentication 30
     - 4.2.3 Information Disclosure 32
     - 4.2.4 Summary of Common ICS Configuration Vulnerabilities 33
   - 4.3 Common ICS Network Vulnerabilities 33
     - 4.3.1 Common ICS Network Design Vulnerabilities 34
     - 4.3.2 Common ICS Network Component Configuration (Implementation) Vulnerabilities 34
     - 4.3.3 Summary of Common ICS Network Vulnerabilities 35

5. ICS SECURITY RECOMMENDATIONS 37
   - 5.1 Recommendations for Vendors 37
     - 5.1.1 Create a Security Culture 38
     - 5.1.2 Enhance ICS Test Suites 39
     - 5.1.3 Create and Test Patches 40
     - 5.1.4 Redesign Network Protocols for Security 41
     - 5.1.5 Increase Robustness of Network Parsing Code 41
     - 5.1.6 Create Custom Protocol Parsers for Common IDSs 41
     - 5.1.7 Document Necessary Services and Communication Channels 42

5.1.8 Implement and Test Strong Authentication and Encryption Mechanisms 42
5.1.9 Improve Security through External Software Security Assessments 43
5.2 Recommendations for ICS Owners and Operators 43
5.2.1 Restrict ICS User Privileges to only those Required 44
5.2.2 Change All Default Passwords and Require Strong Passwords 45
5.2.3 Test and Apply Patches 45
5.2.4 Protect Critical Functions with Network Security Zones and Layers 46
5.2.5 Customize IDS Rules for the ICS and Closely Monitor Logs 47
5.2.6 Force Security through External Software Security Assessments 47

6. SUMMARY 49

7. REFERENCES 50

Appendix A—CSSP Assessment Processes 51

Appendix B—Common Vulnerability Identification 55

Appendix C—Terms and Definitions 59

FIGURES

Figure 1. Percentage of CSSP assessment findings in each vulnerability category. 7
Figure 2. Percentage of CSSP assessment findings per ICS component type. 9
Figure 3. Percentage of CSSP assessment finding types. 9
Figure 4. Categorization of CSSP assessment findings using ISA99 reference model levels. 11
Figure 5. Types of CSSP assessment findings in ISA SP99 Level 4. 12
Figure 6. Percentage of component types with CSSP assessment findings in ISA SP99 Level 4. 12
Figure 7. Types of CSSP assessment findings in ISA SP99 Level 3. 13
Figure 8. Percentage of component types with CSSP assessment findings in ISA SP99 Level 3. 13
Figure 9. Types of CSSP assessment findings in ISA SP99 Level 2. 14
Figure 10. Percentage of component types with CSSP assessment findings in ISA SP99 Level 2. 14
Figure 11. Types of CSSP assessment findings in ISA SP99 Level 1. 15
Figure 12. Percentage of component types with CSSP assessment findings in ISA SP99 Level 1. 15
Figure 13. Generic Man-in-the-Middle attack. 22
Figure 14. CSSP assessment findings related to ICS products. 28
Figure 15. CSSP assessment ICS implementation findings. 33
Figure 16. CSSP assessment network security findings. 36

TABLES

Table 1. Summary of common CSSP ICS assessment findings. 8
Table 2. Reference model for ISA99 standards. 10
Table 3. Summary of common ICS software assessment findings. 28
Table 4. Summary of common ICS configuration findings. 33
Table 5. Summary of common ICS assessment network findings. 36

ACRONYMS

ARP	address resolution protocol
CAPEC	Common Attack Pattern Enumeration and Classification
CRADA	Cooperative Research and Development Agreement
CSSP	Control Systems Security Program
CVE	common vulnerability enumeration
DCOM	Distributed Component Object Model
DHS	U.S. Department of Homeland Security
DLL	dynamic-link library
DMZ	demilitarized zone
DNP	distributed network protocol
DOE-OE	U.S. Department of Energy - Office of Electricity Delivery and Energy Reliability
DoS	denial-of-service
FTP	File Transfer Protocol
HMI	human-machine interface
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol over Secure Socket Layer
IACS	Industrial Automation and Control Systems
ICCP	Inter-Control Center Communications Protocol
ICMP	Internet Control Message Protocol
ICS	industrial control system(s)
IDS	intrusion detection system(s)
IEC	International Electrotechnical Commission
IP	Internet Protocol
IPSec	Internet Protocol security
IT	Information Technology
LAN	local area network
LM	LAN Manager (password hash)
MAC	media access control
MitM	Man-in-the-Middle
NDA	nondisclosure agreement
NTLM	NT LAN Manager
OLE	Object Linking and Embedding
OPC	OLE for Process Control

OS	operating system
PLC	programmable logic controller
RPC	Remote Procedure Call
rsh	remote shell
RTU	remote terminal unit
SCADA	Supervisory Control and Data Acquisition
SDL	security development lifecycle
SIS	Safety Instrumented System
SQL	Structured Query Language
SSH	Secure Shell
SSL	Secure Sockets Layer
TASE	Telecontrol Application Service Element
VPN	virtual private network
WAN	wide area network

Common Cyber Security Vulnerabilities Observed in DHS ICS Assessments

1. INTRODUCTION

The U.S. Department of Homeland Security (DHS) National Cyber Security Division established the Control System Security Program (CSSP) to help industry and government improve the security of the industrial control systems (ICS) used in critical infrastructures throughout the United States. A key part of the CSSP mission is the assessment of ICS to identify vulnerabilities that could put critical infrastructures at risk for a cyber attack. Once these vulnerabilities are identified, mitigation strategies are developed to enhance ICS security.

CSSP has established a collaborative effort among vendors, owners/operators, industry partners, and other national laboratories to provide an assessment environment where ICS can be evaluated for security vulnerabilities. This controlled environment allows realistic assessments of systems and components without the adverse consequences resulting from potential system failures.

The CSSP performs assessments to evaluate vendors' ICS software and assess security issues due to the interdependencies and network design of operational ICS installations. Operational ICS assessments use nonintrusive methods, such as reviewing the production system network diagrams and firewall rules, and performing a hands-on assessment of a duplicate nonproduction installation of the system.

Assessment efforts focus on identifying and understanding the vulnerabilities in ICS that require access to the hardware and software that comprise these systems. This report documents common findings generated from the cyber security assessments.

The term "ICS," as used throughout this report, includes Supervisory Control and Data Acquisition Systems, Process Control Systems, Distributed Control Systems, and other control systems specific to any of the critical infrastructure industry sectors. Although differences in these systems exist, their similarities enable a common framework for discussing and defining security controls. Standard cyber security concepts apply to all computer hardware and software and can be discussed in general as well as specific terms to issues common in ICS. General causes of common ICS vulnerabilities and associated recommendations are discussed in this report. High-level analysis of the problem areas provides insight into the current state of ICS security as indicated by assessments of the latest products available and selected ICS installations.

First, the CSSP assessment methodologies are discussed. Next, the common ICS assessment findings are analyzed according to the different security priorities and vulnerabilities commonly found in each of the ISA99 network levels. Then the common ICS vulnerabilities are presented according to categories that describe a general problem observed in multiple ICS security assessments. These general categories are grouped by (1) vulnerabilities inherent in the ICS product; (2) vulnerabilities caused during the installation, configuration, and maintenance of the ICS; and (3) the lack of adequate protection due to poor network design or configuration. Sanitized assessment findings are listed with the common vulnerability descriptions to aid in understanding the issues. General recommendations based on empirical knowledge gained through performing ICS security assessments are then grouped by software development recommendations for ICS vendors, ICS network configuration, and maintenance recommendations for ICS owners. More information on the CSSP assessment process, common vulnerability recommendation identification, and terms and definitions is available in the appendixes.

2. ASSESSMENT METHODOLOGIES

The primary goal of the CSSP cyber assessments is to improve the security of the critical infrastructure by delivering to each industry partner a report of all security problems found during the assessment along with associated recommendations for improving the security of their product or infrastructure (as appropriate). The CSSP has performed assessments on a large variety of systems. For each assessment, the assessment plan and methodology are tailored to provide the most value to the customer owning the system. System configurations also vary considerably depending on ICS functionality, negotiated objectives, and whether the assessment was conducted in a laboratory or onsite. In all cases, the architecture and boundaries for the system under test are carefully determined. Assessment targets are developed individually for each assessment based on the system configuration and assessment focus in order to address the concerns of the partners. Although a common approach is used for all assessments, the details of each assessment vary; the fact that a vulnerability was not listed on a particular system report does not imply that it did not exist on that system. Common vulnerabilities listed in this report are, therefore, limited to those tested for and found in multiple systems.

Laboratory assessments are designed to evaluate vendor-specific products and services, such as custom protocols, field equipment, applications, and services. The model is to assess systems in multiple phases: (1) a baseline system assessment that identifies vulnerabilities in the vendor's default configuration and (2) an evaluation of the system following implementation of mitigation strategies based on baseline assessment results. In some cases, more than two assessments have been performed on different versions of an ICS. Assessment projects typically leverage a full-disclosure approach with the vendor and asset-owner partners. The CSSP focus is on the ICS and its perimeter. By collecting background architecture, policy, and configuration data from a project partner, the team can perform a more thorough assessment of the system. Penetration testing is a security validation process performed by many commercial entities. CSSP does not simulate a blind attack or penetration of the system, but instead works with the project partner to gain the best understanding possible and provide insight to help mitigate vulnerabilities found in their ICS.

A laboratory assessment generally starts with a basic information technology (IT) assessment of the system, including port scanning, vulnerability scanning, network mapping, password cracking, and network sniffing and fuzzing. In addition to the IT assessment, specific targets or functional pieces of the system are evaluated. These targets are referred to as assessment targets. Testing is often conducted on the ICS local area network (LAN), with the assumption that the attacker has penetrated perimeter protection and is on the ICS network. Typical assessment targets may be "Changing Alarms and Commands" or "Unauthorized Database Access." If the test environment contains connections external to the ICS network, such as to the corporate network, field equipment, or demilitarized zone (DMZ), these connections can be assessed. Typical assessment targets for this portion of the ICS may be "Compromise the Front End Processor" or "Assess Vulnerabilities in DMZ Servers."

Assessment targets are given a priority based on the level of functionality they provide to the system and their operational impacts to the system. Each target is allocated an appropriate amount of testing time according to its priority level. The timeframe may be modified during the assessment based on testing results. Depending on the complexity of the system, testing time for laboratory assessments is generally allotted up to 900 cyber security researcher-hours. The impact to the system is described, and a mitigation strategy is proposed for each finding identified.

Onsite system assessments generally assess how securely external connections, firewall configurations, intrusion detection systems (IDS), network architecture, and any other components are deployed and installed. These assessments generally leverage findings from laboratory assessments with the associated ICS vendor. This technique is commonly coined "ground truthing" because laboratory assessment findings are validated on installed systems. This interaction includes discussion on the viability of possible mitigations and defenses. Onsite assessments generally include 2 weeks of assessment at the asset owner’s site, and can take up to 300 cyber security researcher-hours.

Assessment plans are tailored to each system and to each vendor. Objectives outlined in the assessment plan cover steps that might be potential goals of a real attacker attempting to exploit the ICS and cause damage to equipment, interrupt service, or harm people or the environment.

The methodology used for ICS cyber security assessments includes the following activities:

- ICS target selection

  Together with vendor or plant personnel, the CSSP assessment team identifies a list of assessment target areas. These targets are specific objectives for which the system owners identified or requested as priorities for security review, along with those identified to be of strategic interest to potential attackers. These targets are included in subsequent assessments of the same system.

- Identification of vulnerabilities in selected targets

  Using a combination of commercial, proprietary, and open-source tools, the cyber assessment team discovers information about the targets that may allow the assessment team to compromise the predefined targets. During the course of this discovery, the assessment team may also identify additional targets.

- Attempts to exploit the identified vulnerabilities

  With the information gleaned from antecedent activities, the cyber assessment team attempts to exploit the vulnerabilities they have identified.

- Characterization of identified vulnerabilities

  By documenting their course of action and the results of each activity, the assessment team characterizes the vulnerabilities they have found in terms of the exposure of the vulnerability, the impact of exploiting the vulnerability, and the simplicity of exploiting the vulnerability.

- Recommendations for remediation of identified vulnerabilities

Having characterized the identified vulnerabilities, the assessment team provides their best recommendation to remediate the vulnerabilities. These recommendations are based primarily on the experience of the assessment team and input from the ICS vendor. They may not be fiscally feasible or reflect the operational constraints of the process ICS, but should be considered for risk management purposes. Appendix A contains more assessment information.

2.1 Approach Limitations

This report represents an attempt to assess the most critical vulnerabilities that could put ICS at risk from a cyber attack but does not provide a complete accounting of all possible vulnerabilities associated with ICS.

Assessment reports detail the system under test, the targeted components, the tests performed, the results, and mitigation recommendations. This focus on identifying security problems and solutions was chosen because securing products used in critical infrastructure is part of the process. The focus has not been to simply measure security levels and produce reliable, repeatable, and comparable statistics required for common vulnerability reporting.

Knowledge gained from CSSP assessments performed to assess and help improve the security of ICS does not have the exposure, scrutiny, and security culture change that common IT applications have received. Understanding the unique priorities, vulnerabilities, impacts, limitations, and operation inherent in ICS has been a research effort with evolving assessment procedures. As a result, assessments are customized and tailored to the specific ICS security constraints that the vendor and CSSP assessment teams determine to be critical to the protection of key resources. These security constraints are a combination of targets based on perceived vulnerabilities, impacts or exposure of system components, functionality, and evaluation of security measures built into, or placed around, the ICS.

Standard assessment methodology has a common general theme but not all systems have been tested for the same vulnerabilities. The CSSP assessment methodology is based on this idea of identifying security weaknesses through an attacker's perspective and communicating the security issues to the industry partner from this perspective. CSSP ICS assessments are guided by assessment goals created and customized with industry partners in order to evaluate the ICS functionality both partners feel are highest priority for increasing the security of the particular product or installed ICS. The scope of assessments varied from a proof of concept test of a single attack against a single ICS component, to level of effort identification of possible attack vectors that would allow compromise of the ICS. A broad assessment scope has been very successful in helping increase awareness of the "out-of-the-box" attack methods for which the ICS sector needs to defend. However, this approach has not produced documented data on a consistent set of vulnerabilities for each system assessed.

A lack of consistent vulnerability data occurs because, many assessment goals are result-based, which tests whether an action such as causing a breaker to close could be accomplished by any means, instead of a methodical assessment approach that tests for a given set of potential security flaws. Also, another factor that contributed to the disparity of vulnerability tests was that not all systems configured for assessment have the same set of components and functionality. ICS inherently have many configuration options, such as:

- Operating system
- Functionality shared on the same computer
- Amount of redundancy
- Connected field devices
- Proprietary protocols used to communicate with field devices
- Security features such as security zones, intrusion detection, and up-to-date methods.

The size and architecture of assessed systems varied widely. For example, assessed systems ranged from:

- An ICS with the most basic functionality on a single local area network (LAN) with a few computers
- An ICS with partial optional functionality, simulated data, and some network security defense devices
- All available ICS functionality connected to control hardware with the recommended network architecture and perimeter defenses
- An operational system with duplicate system for interactive testing.

This disparity leads to differences in assessment focus based on what was available on the system for assessment and priority lists for which part of the system will be evaluated in the allotted timeframe.

The observations and recommendations in this report are based on common security practices and the experience of the assessment team. There are many security issues and solutions common to ICS which have been generalized in this report. The possible affects on the unique operation, maintenance, and architecture of each ICS installation should be evaluated before implementing security mitigations. Observations and recommendations are based on the findings documented in CSSP assessment reports that have been tailored toward the vendor and CSSP assessment teams' knowledge and methods.

2.2 Impact on ICS Security

Reports generated from the assessment of ICS products and installations have been used by vendors and asset owners to understand and mitigate cyber vulnerabilities found during each assessment. Although not all findings have been addressed, most systems have been modified to improve security based on assessment reports. After-action validation of mitigations to identified security flaws are performed by the CSSP assessment team to help ensure the security assessments are successful in increasing critical infrastructure security. Some of the vendors have been forthright in sharing the results with their customers, and some have felt that any disclosure of vulnerabilities could lead to exposure of their customers to potential cyber attacks. Whether specific results were shared with ICS customers or not, CSSP has shared general ICS security knowledge gained through the assessment process . Cyber security and ICS researchers have presented results at varying levels of detail, and provided security training to attendees of the various vendor user group conferences. ICS security awareness has also been increased through training, presentations, and documents generated from multiple assessments' results.

Onsite assessments have been conducted on installed ICS in order to help secure the particular site, verify laboratory findings, and gather common recommendations. This brings laboratory vendor assessments full circle to help the ICS software, operational implementations, and the associated critical infrastructure become more secure.

3. COMMON ICS ASSESSMENT FINDINGS

Common CSSP assessment findings in this report were not derived from a comprehensive or even consistent set of assessments or fully operational ICS. The common assessment findings that follow are similar security problems found on two or more unique ICS configurations. Common assessment findings and common ICS vulnerabilities refer to common vulnerabilities found in CSSP assessments. They are derived from detailed assessment findings. In order to maintain ICS product anonymity and bring findings up to a high enough level, different detailed findings were grouped into one common vulnerability. For example, the common vulnerability, ICS protocol uses weak authentication, was derived from the following assessment findings:

- MitM altering of ICS communication possible between ICS and controller equipment
- Firmware update uses weak integrity checks
- MitM altering of ICS inter-process communication possible between ICS components
- MitM altering of ICS communication possible between controller and field equipment.

Each of these detailed findings may actually apply to one or multiple systems. There may also be other findings that were not listed because they were too indicative of the affected system. Remember, ICS Protocol authentication may not have been evaluated on all systems. Therefore, an inference cannot be made that vulnerabilities not included in this report are not common to ICS or that three detailed findings under a common vulnerability means that only three systems assessed were susceptible to that vulnerability.

Table 1 lists the common vulnerabilities found through CSSP security assessments. A common vulnerability describes findings from a minimum of two different ICS product lines or installed systems. Vulnerability descriptions are generalized to remove specific vendor-identifying information and details that would hinder the ability to group common vulnerabilities.

As with all common vulnerability categorization methods, overlaps exist between the different categories that comprise each taxonomy. For example, poor code quality is a category that could contain all application types as subcategories (i.e., the database, Web, human-machine interface (HMI), and services applications that make up an ICS product). Each of these subcategories would then contain a “lack of input validation” common vulnerability. Another option is to list lack of input validation as a general vulnerability with or without specifying where it has been found to be a problem. Many different categories can conceivably be used to view the vulnerability data, and will most likely result in different vulnerabilities being categorized together.

At a high level, common vulnerabilities in this document are categorized differently based on how the problem is being viewed. Table 1 groups common ICS vulnerabilities according to nine general security problems that sum up the main weaknesses that ICS products and installations are prone to have due to legacy code, lack of security training and requirements, and ICS operational requirements. These categories are not mutually exclusive, but highlight the main causes of vulnerabilities that put ICS at risk to cyber attack. Figure 1 shows the percentage of CSSP assessment findings for each of the categories in Table 1.

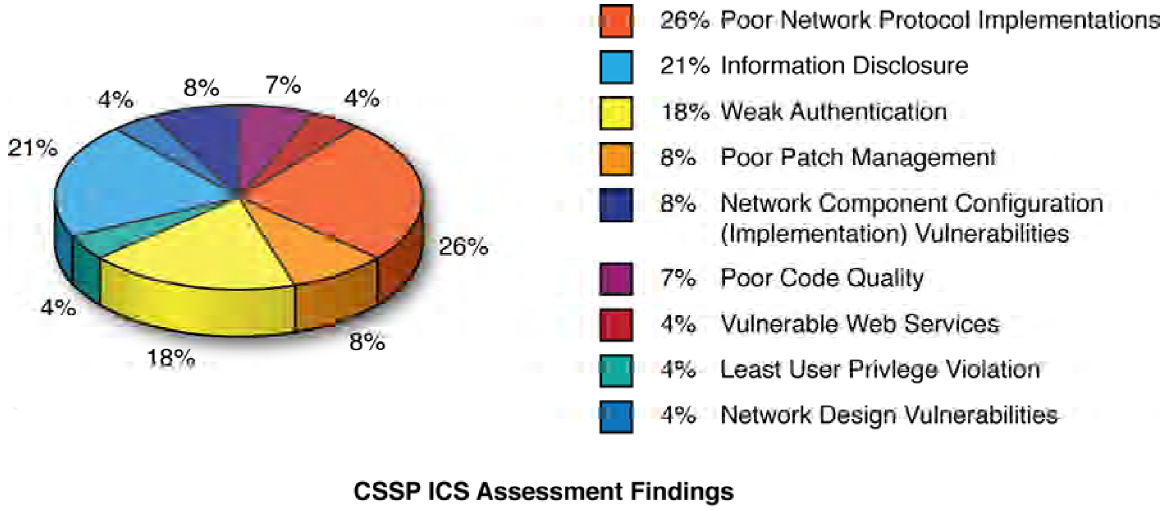

Figure 1. Percentage of CSSP assessment findings in each vulnerability category.

Table 1. Summary of common CSSP ICS assessment findings.

Category	Common Vulnerability
Poor Code Quality	Use of potentially dangerous functions in proprietary ICS application
Vulnerable Web Services	Poor authentication
	Directory traversal enabled
	Unauthenticated access to Web server
Poor Network Protocol Implementations	Lack of input validation: Buffer overflow in ICS service
	Lack of input validation: Lack of bounds checking in ICS Service
	ICS protocol uses weak authentication
	ICS protocol uses weak integrity checks
	ICS product relies on standard IT protocol that uses weak encryption
Poor Patch Management	Unpatched or old versions of third-party applications incorporated into ICS software
	Unpatched operating system
Weak Authentication	ICS uses standard IT protocol that uses weak encryption
	Use of standard IT protocol with clear-text authentication
	Client-side enforcement of server-side security
	Improper security configuration
	No password required
	Weak passwords
	Weak password requirements
Least User Privileges Violation	Unauthorized directory traversal allowed
	Services running with unnecessary privileges
Information Disclosure	Unencrypted proprietary ICS protocol communication
	Unencrypted nonproprietary ICS protocol communication
	Unencrypted services common in IT systems
	Open network shares on ICS hosts
	Weak protection of user credentials
	Information leak through unsecure service configuration
Network Design Vulnerabilities	Lack of network segmentation
	Firewall bypassed
Network Component Configuration Vulnerabilities	Access to specific ports on host not restricted to required IP addresses
	Port security not implemented on network equipment

ICS are made up of process equipment, process control hardware, network devices, and computers. Vulnerabilities in network devices and protocols, or the operating systems, ICS software, and other software running on the ICS computers could allow an attacker to gather information about, disrupt, or manipulate ICS operations. The percentage of CSSP assessment vulnerabilities that were found in each ICS component are shown in Figure 2 below.

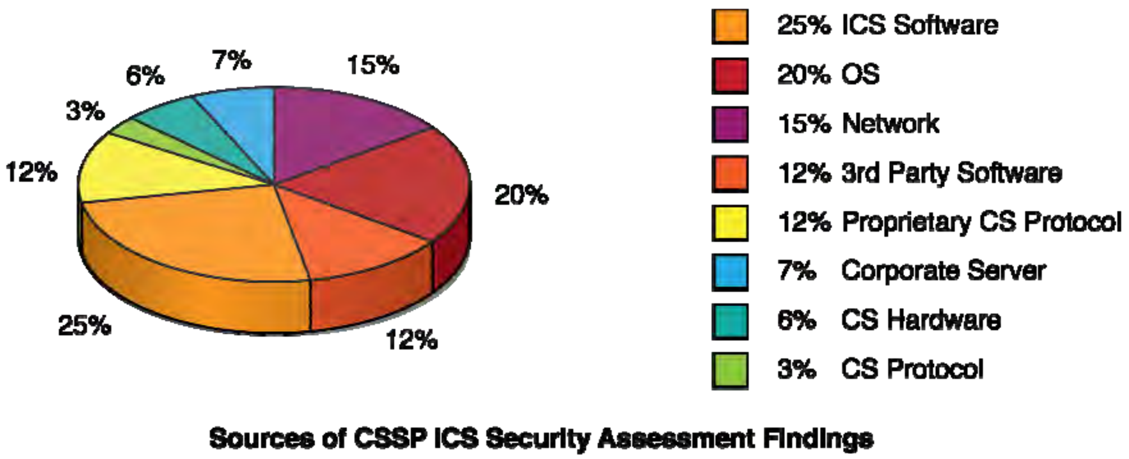

Figure 2. Percentage of CSSP assessment findings per ICS component type.

Each CSSP finding is a security risk to the ICS, but may not actually qualify as a vulnerability. To promote better understanding of assessment findings, Figure 3 displays the categories of finding types. Configuration problems are security risks due to the way a system or application was installed and configured. This includes password policies, firewall, IDS and switch rules and placement, Web server access rules, etc. Configuration problems are not vulnerabilities in the component itself but in specific component usage.

In cyber security, known vulnerabilities are vulnerabilities that have been publically announced. Someone has written or possibly published exploits for that vulnerability. All CSSP known vulnerability findings had patches available. This report does not include findings from automated vulnerability scans unless the component was verified to be vulnerable. No large focus is generated for known vulnerabilities in CSSP assessments because the goal is to help identify security problems specific to the ICS software, which has not been subjected to the scrutiny as the more widespread products. Typically, assessment findings note that vulnerable versions of operating systems, services, or applications are available on the ICS and should be patched, upgraded, or removed. Known vulnerabilities are therefore a bigger problem than Figure 3 portrays because patch management is not straightforward on ICS because of the risk that a patch or upgrade may adversely affect operations. The vendor and owner must be able to test the affect of each patch thoroughly on a representative system, and vendors may have to change ICS code to work with new application versions.

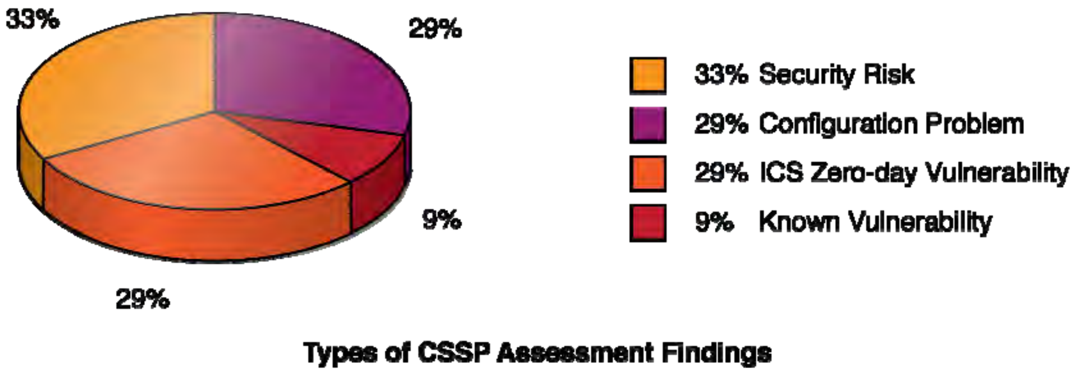

Figure 3. Percentage of CSSP assessment finding types.

ICS zero days are vulnerabilities found in ICS specific applications and protocols during assessments. Zero days are known as such even though they have been disclosed to the vendor who may have disclosed them to customers at some level. This report does not contain information on the number of ICS vulnerabilities that have been fixed or whether patches were released for existing systems.

The security risk category includes all findings that cannot be called a vulnerability per se, but do put the ICS at a higher risk of attack. For example, having open ports on a host is necessary for communication. Otherwise, any open port could be disconnected from the network and it would only be at risk to physical attack. Services or applications running on a system open up different network ports to be able to communicate to the outside world. Each open port provides a possible access path for an attacker that can be used to send exploits and receive data. An attacker can only gain access to and receive information from the ICS through an open port. The more ports and services that are accessible, the greater the risk of successful exploits due to existing vulnerabilities in the services. Even if no known vulnerabilities exist in the current system, new vulnerabilities are found every day in the applications and services that run on computers. Some of these vulnerabilities are published shortly after their discovery, and some are kept a close secret, allowing a few hackers to exploit computers at will, with no patches available to stop them. Decreasing the number of installed applications and services decreases the likelihood of an attacker finding a vulnerability on the computer. Example assessment findings categorized as security risks are:

- Use of a high-risk service with a history of vulnerability announcements
- System has a large number of services running
- Unencrypted communication
- Lack of IDS.

3.1 CSSP Findings Mapped to the ISA SP99 Reference Model Levels

The ISA reference model creates a framework for referencing general Industrial Automation and Control Systems (IACS) network levels.[1] Although all CSSP assessment system networks were not designed consistently, this framework allows the findings to be consistently categorized by logical network layers. Each level represents a class of functionality.[a] Table 2 lists the ISA SP99 reference model levels and associated IACS and Supervisory Control and Data Acquisition (SCADA) functions.

Table 2. Reference model for ISA99 standards.

ISA Level	Functions	ISA99 Standard IACS Functions	SCADA Reference Model Functions
Level 4	Enterprise Systems	Business Planning & Logistics	Engineering Systems
Level 3	Operations Management	Operations Management	System Management Supervisory Control
Level 2	Supervisory Control	Supervisory Control	Site Monitoring & Local Display
Level 1	Local or Basic Control	Basic Control Safety and Protection	Local Control Protection
Level 0	Process	Equipment Under Control	Equipment Under Control

[a] Note: The ISA IACS models are currently in the process of being updated.

The majority of functionality evaluated in CSSP assessments was at the supervisory control level. None of the assessments used for this report listed findings at the process level. Figure 4 illustrates the percentage of CSSP assessment findings identified in each of the ISA reference model levels. An Internet level was added for findings referring to information found on the Internet that could be used in an attack.

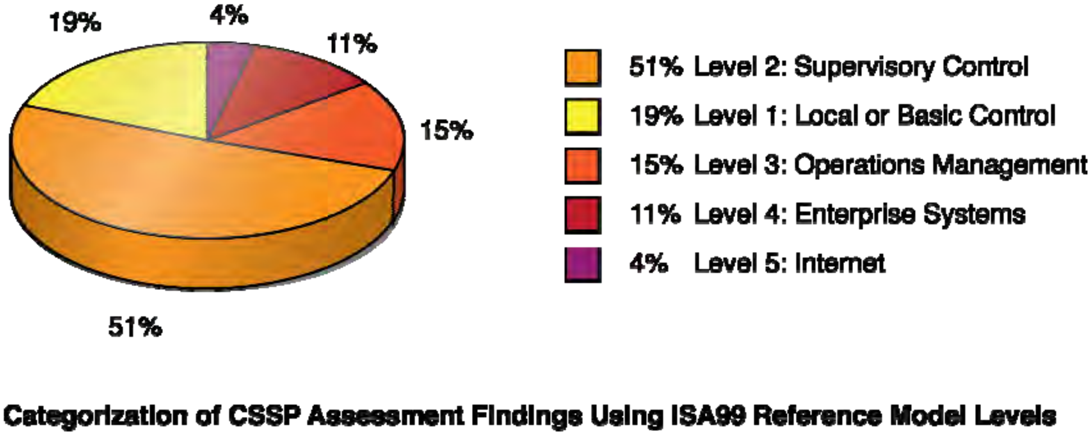

Figure 4. Categorization of CSSP assessment findings using ISA99 reference model levels and an Internet level.

3.1.1 Level 4: Enterprise Systems

Level 4 can be thought of as the systems on the enterprise network used for business planning and logistics. This level includes the business-related activities needed to manage a manufacturing organization. Functions include enterprise or regional financial systems and other enterprise infrastructure components such as production scheduling, operational management, and maintenance management for an individual plant or site in an enterprise. Engineering systems are also considered to be in this level; however, CSSP engineering station findings were categorized in Level 2 because they were specific to the "normal" configuration where the engineering station was placed in the HMI LAN.

Figure 5 groups the unsecure Web server configurations, weak passwords, IDS and firewall problems, and ICS information found on corporate LANs into the configuration problem category. Web HMI Application zero-day vulnerabilities were also included in this level. Although vulnerabilities in Level 4 may aid an attacker by providing information about the ICS, the business planning and logistics functions are separate from the ICS functions in the lower layers. Compromise should only provide an attacker with process information or a launching point into the ICS (unless a Web HMI on the corporate network is allowed control of the process). The ISA model does not include ICS functions in Level 4 because supervisory or process control functions need to be isolated and protected as much as possible due to high possible consequences.

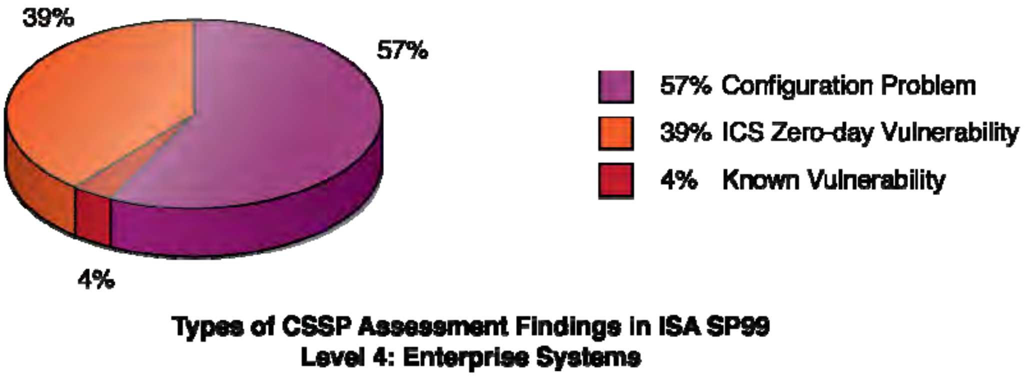

Figure 5. Types of CSSP assessment findings in ISA SP99 Level 4.

Figure 6 shows that most CSSP assessment findings in Level 4 are in proprietary ICS applications used for business planning and logistics, including Web HMI applications. The rest are security risks due to the configuration of common applications and services, operating systems, and network components.

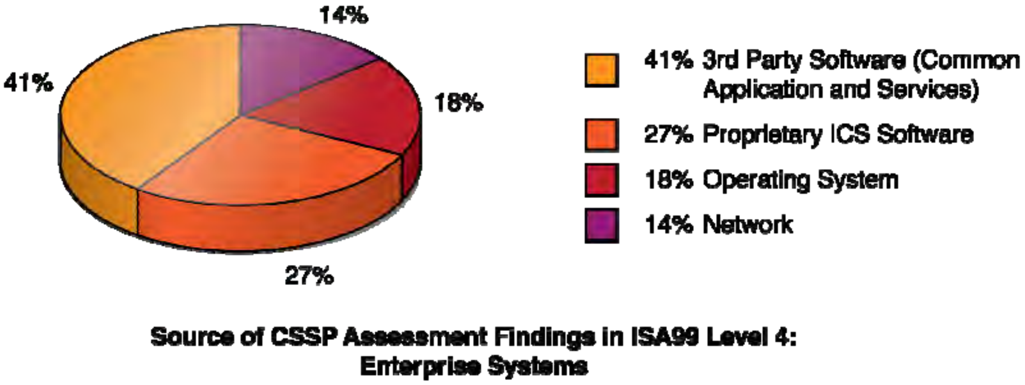

Figure 6. Percentage of component types with CSSP assessment findings in ISA SP99 Level 4.

3.1.2 Level 3: Operations Management

Level 3 includes functions that manage ICS production (e.g., planning, scheduling, and quality assurance). The CSSP assessment components categorized in this level include all ICS management functions that can be separated from the supervisory control LAN such as data historians, ICS Web servers, Inter-Control Center Communications Protocol (ICCP) servers, and Object Linking and Embedding (OLE) for Process Control (OPC) servers.

Figure 7 illustrates that the majority of findings are ICS zero-day vulnerabilities with significant numbers of configuration problems and known vulnerabilities. Figure 8 shows many Level 3 network perimeter defense problems have been found along with vulnerabilities in network protocols used to communicate with Level 3 applications. If the Level 3 perimeter defenses allow communications using vulnerable protocols or connections to vulnerable applications, an attacker may be able to gain a foothold onto the ICS network.

The elimination of known vulnerabilities is extremely important because they are generally easily discovered and exploited. ICS users are well advised to configure network perimeter defenses as securely as possible to prevent access to the ICS. All network protocols allowed between Level 4 enterprise systems and the outer layer of the ICS network should be minimized and secured along with the applications that handle them. Finally, the importance of compartmentalization and security of all other traffic and applications at this layer to prevent a Level 3 compromise from going deeper into the ICS network cannot be underemphasized.

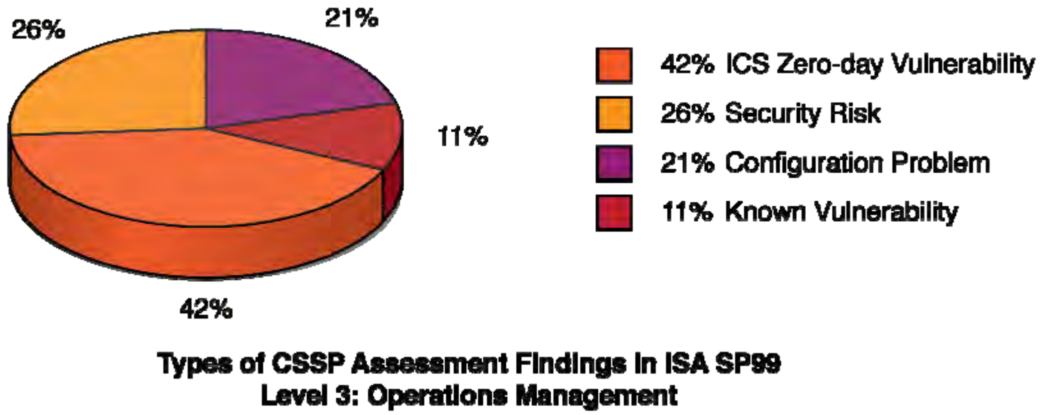

Figure 7. Types of CSSP assessment findings in ISA SP99 Level 3.

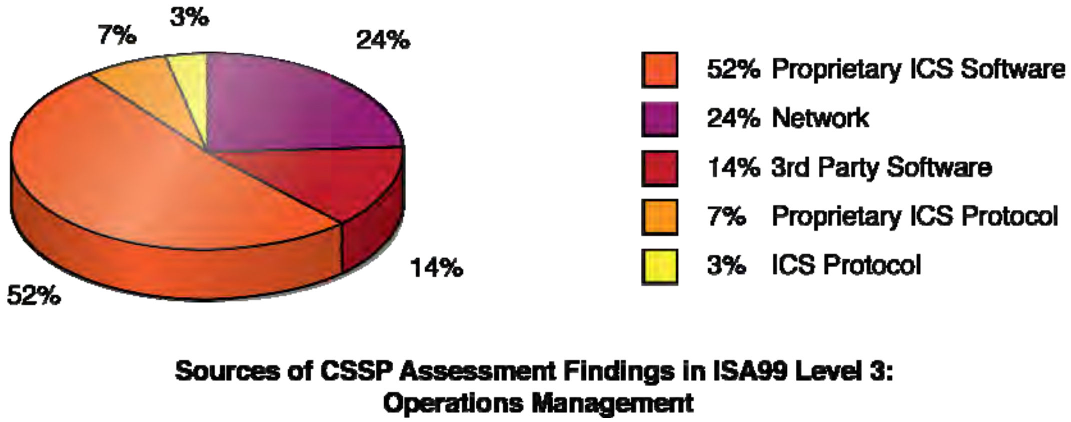

Figure 8. Percentage of component types with CSSP assessment findings in ISA SP99 Level 3.

3.1.3 Level 2: Supervisory Control

Level 2 includes the functions involved in monitoring and controlling the physical process. These functions include operator HMI applications that can send control commands and receive monitor data, alarms, and alerts.

Known vulnerabilities and configuration problems can provide pathways from a compromised Level 3 component to the vulnerable Level 2 component. Operating systems and common third-party software account for the known vulnerabilities in Figure 9. Configuration problems such as weak passwords and poor network design and access rules can also allow unauthorized access to Level 2 components.

Proprietary ICS protocols allowed between Levels 2 and 3 have been found susceptible to Man-in-the-Middle (MitM) attacks, which can be used for ICS information gathering and manipulation. The applications written to handle ICS traffic have been found vulnerable to invalid input exploits that can cause denial-of-service (DoS) of ICS communications or unauthorized access to Level 2 components. DoS of the protocols used to transfer ICS data on the supervisory control network can cause high impacts to systems that require monitor and control availability.

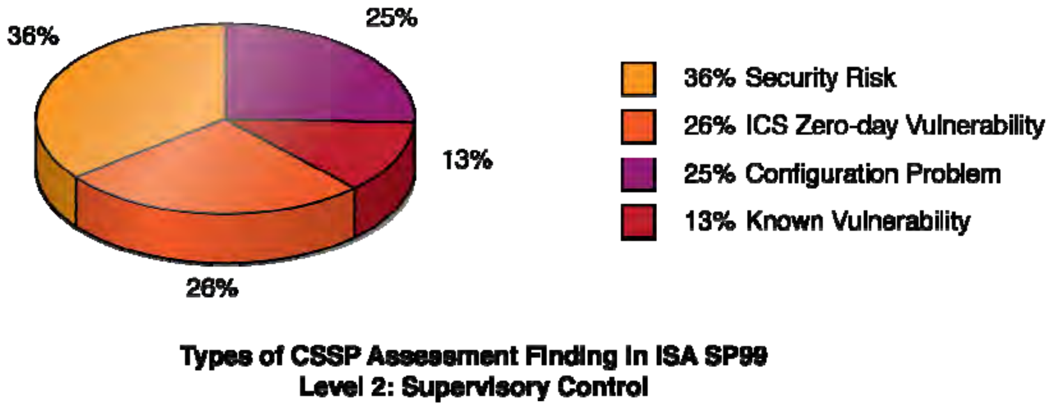

Figure 9. Types of CSSP assessment findings in ISA SP99 Level 2.

Figure 10 shows that CSSP assessments found the most Level 2 security problems in operating systems, followed by proprietary ICS software used for supervisory control, common third-party applications, network devices, and then proprietary ICS communication protocols. Vulnerabilities and configuration problems with services shipped with the operating system are included in the Operating System count.

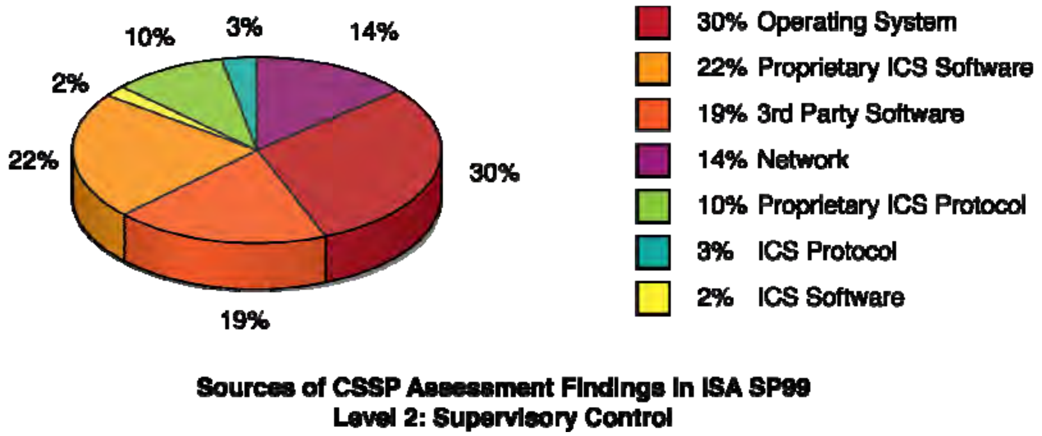

Figure 10. Percentage of component types with CSSP assessment findings in ISA SP99 Level 2.

3.1.4 Level 1: Local or Basic Control

Level 1 is the control network that connects the supervisory control level to lower-level control modules, including the functions used for sensing and manipulating the physical process.

Figure 11 shows that half of the Level 1 assessment findings are ICS zero-day vulnerabilities. This vulnerability type is mapped to the ICS software, firmware, network protocols, hardware devices (e.g., programmable logic controllers [PLCs], controllers, safety integrated systems, and remote terminal units [RTUs]) and security devices that make up 73% of the Level 1 vulnerable components in Figure 12. If an attacker is able to gain access to this level and reverse engineer the protocols and process, a malicious prevention or manipulation of sensing and control functions at the lowest level can occur.

Known vulnerabilities and configuration problems in the operating systems and applications running on the control network devices and ICS devices provide opportunities for unauthenticated access to the control network. CSSP assessments have found Web servers on PLCs and switches that do not require authentication. Many communication protocols used to transfer process data, program ICS hardware, etc., are plain text and use weak authentication and data integrity methods.

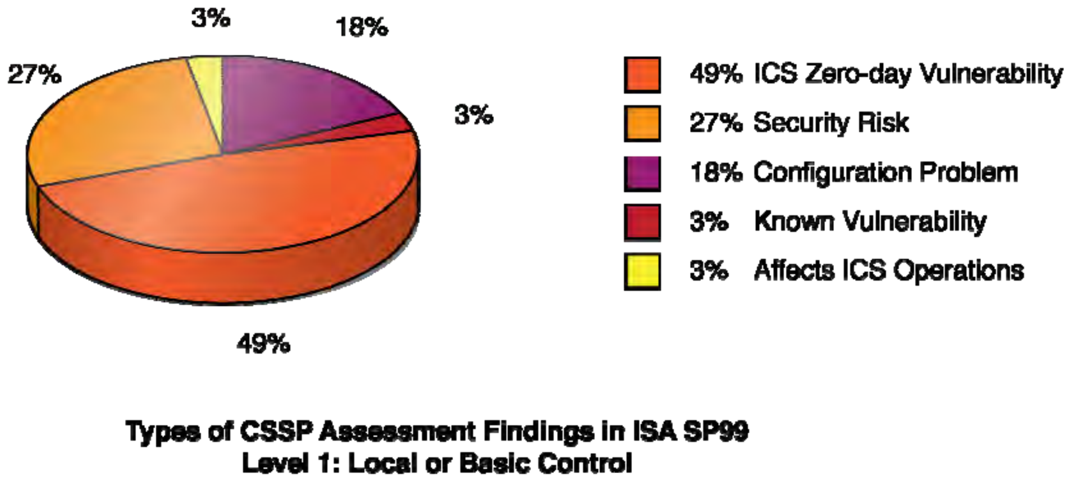

Figure 11. Types of CSSP assessment findings in ISA SP99 Level 1.

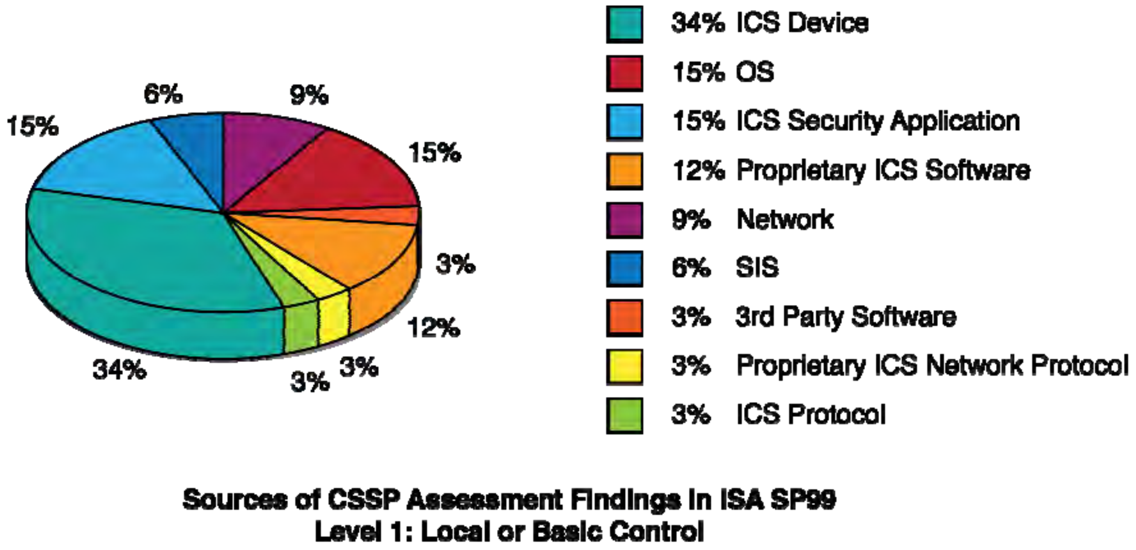

Figure 12. Percentage of component types with CSSP assessment findings in ISA SP99 Level 1.

4. UNDERSTANDING COMMON ICS VULNERABILITIES

CSSP ICS security assessments have identified the vulnerabilities described in this section in a majority of the systems that included the associated functionality. In addition to a subset of these common vulnerabilities, additional vulnerabilities unique to the individual ICS software and implementations were identified. All these vulnerabilities can be mitigated by following secure software design and development principles, and secure platform, software, and network configuration guidelines.

The difference in securing ICS and a typical computer system is in the nonstandard ICS components that do not use off-the-shelf hardware or software and have non-typical security priorities. Custom ICS hardware and software have not been scrutinized like common computer products. Therefore, although they contain a high degree of vulnerabilities, very few of them have been publically announced. While adding security measures to ICS components, keep in mind the importance of functional requirements. Unlike typical IT systems, ICS security objectives are typically prioritized as:

1. Availability
2. Integrity
3. Confidentiality.

Violating operational requirements while implementing security features in ICS could cause more damage than a cyber attack.

4.1 Common ICS Software Vulnerabilities

The ICS vendor software assessment findings are described in the following sections. Sanitized assessment details are listed with each common vulnerability description to aid in understanding the real issues. Multiple assessments may have findings that match the same vulnerability details, and one assessment may have multiple specific detailed vulnerabilities relating to one common vulnerability. Some common vulnerabilities have only one detailed example that describes all findings from the associated assessments. The number of systems that were found at risk to a given vulnerability is not listed in order to avoid any implication that all systems were tested for that vulnerability and to help lend anonymity to the ICS associated with common vulnerabilities and the related specific details listed.

4.1.1 Poor Code Quality

Poor code quality refers to code issues that are not necessarily vulnerabilities, but indicate that it was not securely developed. These products are more likely to contain vulnerabilities than those that were developed using secure development concepts and other good programming practices. “If a program is complex, difficult to maintain, not portable, or shows evidence of neglect, then there is a higher likelihood that weaknesses are buried in the code.”[2]

ICS code review and reverse engineering exercises indicate that ICS software has not been designed or implemented using secure software development concepts in general. The relatively greater ages of core ICS applications increase the likelihood of development as stand-alone systems with only reliability and efficiency as requirements. However, new ICS applications tend to suffer from the same lack of secure coding principles.

4.1.1.1 Use of Potentially Dangerous Functions in Proprietary ICS Applications

Otherwise known as unsafe function calls, the application calls a potentially dangerous function that could introduce vulnerability if used incorrectly, but the function also can be used safely. The problem with using unsafe functions is that the developer is responsible for validating input. The number of publicly announced buffer overflow and other malformed input vulnerabilities is evidence that implementing this validation is a high risk.

Unsafe C/C++ function calls are the most notorious potentially dangerous functions. All have safe counterparts, so there is no reason to use unsafe functions or not replace them in existing code. The strcpy() function in C is an example of a potentially dangerous function because of introducing a buffer overflow vulnerability. If the input to strcpy can in any way be influenced, a chance exists that an attacker can find a way to circumvent the developer's logic. In many cases, the logic is only based on what would normally happen, and a buffer overflow attack is successful because the developer decided that no one would ever create a username longer than 1,024 characters. The attacker simply needs to try a few usernames to figure out that more than 1,024 characters causes problems. The developer can test to make sure nothing larger than the memory buffer he created is sent to strcpy(), but strncpy() eliminates this risk by requiring that the buffer size is specified. The following are specific assessment findings associated with unsafe C/C++ function calls:

- Several instances of unsafe function calls found in proprietary communications processing code
- Unsafe C/C++ function calls in proprietary ICS code
- Unsafe C/C++ functions in OPC dynamic-link libraries (DLLs)
- Use of potentially dangerous functions in proprietary ICS application.

Recommendation: ICS applications tend to suffer from poor code quality. Vendors and asset owners who write custom applications should train developers in secure coding practices. All custom software should undergo thorough code review via both manual and automated processes to identify security issues while the code is still in the development stage. ICS-specific protocols should be redesigned to include strong authentication and integrity checks. IT products deployed on the ICS network should also have passed a security review. Asset owners should explicitly address the security of these products during the procurement process.

4.1.2 Vulnerable Web Services

Many ICS have recently incorporated Web applications and services to allow remote supervisory control, monitoring, or corporate ICS data analysis. ICS assessments have found unauthorized directory traversal and authentication problems with ICS Web implementations. Many of the poor code quality and input validation findings in this section refer to proprietary Web applications.

4.1.2.1 Directory Traversal Enabled

Web application directory traversal vulnerabilities occur when file paths are not validated. Directory traversals are commonly associated with Web applications, but all types of applications can have this class of vulnerability. Directory Traversals occur when the developer uses a path provided by the user, but fails to validate the path in order to ensure that the user can only access the necessary files. For example, the classic Hypertext Transfer Protocol (HTTP) "GET" directory traversal attack is performed by submitting "../" to tell the operating system (OS) to look up one directory. If the HTTP server was vulnerable to a directory traversal attack, this GET request would cause the HTTP to get the /etc/passwd file. Directory traversal attacks can be used to gather information by downloading files, or to gain access to the ICS by uploading the exploit code to be executed. The damage that a directory traversal vulnerability can cause is related to the permission of the application that was vulnerable. If the vulnerable application has limited read/write permissions, the attacker may not be able to do anything of importance. However, when running as system or root, then the damages can be extensive. Being able to download arbitrary files is more common then being able to upload files.

The following are specific assessment findings associated with this vulnerability:

- Web servers on multiple assessment systems had directory traversal enabled
- Web server directory browsing was enabled.

Recommendation: The file permissions on the Web server need to be set to grant the least privileges necessary. The system design needs to be evaluated to reduce necessary file access as much as possible. Features on the Web server, such as unrestricted browsing, need to be disabled and additional security of HTTP can be gained by utilizing the Secure Sockets Layer (SSL) where possible. The Web server should filter input to screen incoming filenames and exclude the ".." string. Disabling unused ports and keeping the Web server patched to current standards are good practices. Write permissions are most dangerous, but read permissions may disclose valuable information or information that can be used for an attack.

4.1.2.2 Unauthenticated Access to Web Server

Web services developed for the ICS tend to be vulnerable to attacks that can exploit the ICS Web server to gain unauthorized access. System architectures often use network demilitarized zones (DMZs) to protect critical systems and to limit exposure of network components. Vulnerabilities in ICS DMZ Web servers may provide the first step in the attack path by allowing access within the ICS exterior boundary. Vulnerabilities in lower level component's Web servers can provide more steps in the attack path.

ICS assessments have also found poor authentication, poor session tracking, Structured Query Language (SQL) injection, and cross-site scripting vulnerabilities that can allow unauthorized access to Web servers and applications. These types of issues were categorized in Section 4.1.3, Poor Network Protocol Implementations.

The following are specific assessment findings associated with this vulnerability:

- Web server on controller required no authentication
- Unauthenticated access to Web HMI Web server
- Web HMI Web server username/password authentication bypass.

Recommendation: ICS applications should use well known and tested third-party Web servers to serve their Web applications. Web applications should be thoroughly tested for malformed input and other vulnerabilities that could lead to a compromise of the ICS Web server.

4.1.3 Poor Network Protocol Implementations

Network protocols specify how information is packaged and sent across a computer network. For every network protocol, an application (known as server) must wait for and process the data off the network. All ICS products use at least one protocol created specifically for ICS component communication. In order to communicate using standard (nonproprietary) ICS protocols, each ICS vendor must implement their own server application to process the network traffic.

Part of the protocol specification is if and how authentication, integrity checks, and confidentiality will be implemented. Services that employ weak authentication methods can be exploited to gain unauthorized privilege. Poorly protected credentials can be found in documentation or code, sniffed "off the wire," cracked, or guessed.

4.1.3.1 Lack of input validation: Buffer Overflow in ICS Service

Input validation is used to ensure that the content provided to an application does not grant an attacker access to unintended functionality or privilege escalation. Buffer overflows are the result of programmer oversight. This usually happens because the programmer only considered what should happen and what could happen by mistake, but not all the "out of the box" possibilities such as entering a 2,000-character-long last name.

Buffer overflows result when a program tries to write more data into a buffer than the space allocated in memory. The "extra" data then overwrites adjacent memory, and ultimately results in abnormal operation of the program. A careful and successful memory overwrite can cause the program to begin execution of actual code submitted by the attacker. Most exploit code allows the attacker to create an interactive session and send commands with the privileges of the program with the buffer overflow. Network protocol implementations, which do not validate input values can be vulnerable to buffer overflow attacks.

Services written by ICS vendors frequently suffer from coding practices that allow attackers to supply unexpected data and thus modify program execution. Some ICS protocol implementations are vulnerable to packets that are malformed or contain illegal or otherwise unexpected field values. Even though some ICS protocols are commonly used, the services that receive and interpret the protocol traffic are usually customized to the vendor product. Vulnerabilities in these services were a main target of many assessments because buffer overflows in the ICS services are possible entry points onto the ICS components.

The following are five specific assessment findings associated with this vulnerability:

1. Multiple assessments found buffer overflows in ICS protocol implementations
    a. Stack-based buffer overflows allowed remote code execution on ICS hosts
    b. Heap-based buffer overflows allowed remote code execution on ICS hosts
2. Multiple assessments found buffer overflows in ICS application services
    a. A buffer overflow was found in a proprietary historian database (historian communication protocol server application)
    b. Username and Password Buffer Overflows in Web HMI Web Server
    c. Exploitable stack overflow in OPC server.

Recommendation: All code should be written to validate input data. All programmers should be trained in secure coding practices, and all code should be reviewed and tested for input functions that could be susceptible to buffer overflow attacks. All input should be validated, not just those proven to cause buffer overflows. Input should be validated for length, and buffer size should not be determined based on an input value. Length validation is especially important in the C and C++ programming languages, which contain string and memory function calls that can be used insecurely.

Even if values are never input directly by a user, data will not always be correctly formatted, and hardware or operating system protections are not always sufficient. Most buffer overflows identified in CSSP assessments were in the server applications that process ICS protocol traffic. In most cases, values input from network traffic were intercepted and altered in transit. Therefore, network data bounds and integrity checking should be implemented.

Perform a code review of all ICS applications responsible for handling network traffic. Network traffic cannot be trusted, so better security and sanity checks need to be implemented so fuzzing attempts will not cause crashes or a DoS.

4.1.3.2 Lack of Input Validation: Lack of Bounds Checking in ICS Service

The lack of input validation for values that are expected to be in a certain range, such as array index values, can cause unexpected behavior. For instance, unvalidated input, negative, or too large numbers can be input for array access and cause essential services to crash.

ICS applications frequently suffer from coding practices that allow attackers to supply unexpected data and thus modify program execution. Even though ICS applications pass valid data values during normal operation, a common vulnerability discovery approach is to alter or input unexpected values.

The following are specific assessment findings associated with this vulnerability:

- DoS caused by out of range index values:
    - Crashed ICS communications service by altering input value to negative number

- Crashed proprietary fault tolerant network equipment protocol.

Recommendation: All code should be written to validate input data. Every programmer should be trained in secure coding practices. All code should be reviewed and tested for input functions that could be susceptible to buffer overflow attacks. All input should be validated, not just those proven to cause buffer overflows. Input values should be validated.

Even if values are never input directly by a user, data will not always be correctly formatted, and hardware or operating system protections can be insufficient. Further ICS traffic may be intercepted and altered in transit. Therefore, Network data value and integrity checking should be implemented.

4.1.3.3 ICS Protocol Uses Weak Authentication

Commands from the HMI cause actions in the ICS. Alarms are sent to the HMI that notify operators of triggered events. The integrity and timely delivery of alarms and commands is critical in an ICS.

Weak authentication in ICS protocols allows replay or spoof attacks to send unauthorized messages, and a possibility of sending messages that update the HMI or RTU must be considered. The attacker may be able to cause invalid data to be displayed on a console or create invalid commands or alarm messages.

Clear-text authentication credentials can be sniffed and used by an attacker to authenticate to the system.

The following specific assessment findings associated with this vulnerability were identified on multiple assessments:

- Common ICS protocol uses weak authentication between controller and field equipment
- Proprietary ICS protocol uses weak authentication between controller and HMI
- Firmware update uses weak authentication
- Proprietary ICS protocol uses weak authentication between ICS components
- Blind trust relationships based on the IP address as specified in the /etc/hosts file
- Lack of secure authentication for session initiation and message authentication means the attacker can initiate sessions or alter established sessions with little difficulty
- HMI login transmits passwords in clear text, which allows remote attackers to sniff the operator password
- Remote telnet-style applications with weak authentication run in plain text on the ICS network
- HMI LAN communication protocol authentication by IP address.

Recommendation: The system design needs to implement strong authentication into ICS communication protocols and encrypt communications if appropriate and possible. Secure authentication and data integrity checks should be used to ensure that process commands and updates have not been altered in transit. These security procedures offer protection against spoofing attacks, in which false information is sent to the operator’s console in order to give them an altered view from reality. Authentication also protects against unauthorized commands being sent to the ICS process devices.

Physical access to the controller while the controller is disconnected from a production Ethernet network should be required for firmware updates. Ensuring that updates occur in this environment will help prevent possible exploitation and will also prevent the information disclosure of the device’s firmware. Authentication and data integrity checks should also be used to protect against unauthorized physical access and manipulation of firmware files.

4.1.3.4 ICS Protocol Uses Weak Integrity Checks

The lack of, or weak, data integrity checks prevent a protocol from detecting bad data. An attacker is able to manipulate alarm or command messages sent over the wire if the ICS protocol has poor integrity checks. This manipulation creates an identical effect where the attacker may be able to cause invalid data to be displayed on a console or create invalid command or alarm messages.

If an attacker has access to ICS communication paths and reverse engineered the ICS network communications protocol, manipulation is possible of the data flowing between the systems components. This includes commands and messages sent to update operator screens and control field equipment. Altering the operator’s view of the system received from the ICS can be used to either trick the operator into performing actions or hide what an attacker is doing with the system.

If integrity check values or “checksums” are omitted from a protocol, no way of determining if data have been corrupted in transmission can be found. Likewise, if integrity check values are easily reverse engineered and duplicated, data manipulation in transmission is invisible upon security inspection.

The following specific assessment findings associated with this vulnerability were identified on multiple assessments:

- MitM altering of ICS communication possible between ICS and controller equipment
- Firmware update uses weak integrity checks
- MitM altering of ICS inter-process communication possible between ICS components
- MitM altering of ICS communication possible between controller and field equipment.

MitM is possible when the communication protocol does not insure the identity of each communication partner or the integrity of the message. If an attacker can pose as a trusted communication partner (if necessary) and formulate the correct integrity check values for a new or altered message, the communication channel is at risk.

Manipulating the communications on a control network requires an in-depth understanding of the protocol to be manipulated. The cyber assessment team is generally able to gather enough information about a network protocol to perform a network layer attack against the system. Most effective network attacks use the address resolution protocol (ARP) MitM attack to achieve their objectives.

The ARP MitM attack is a popular method used by an attacker to gain access to the network flow of a target system. In this style of attack the network ARP cache of machines on the LAN are targeted, confusing whom they think they are communicating with. The ARP protocol is used to determine which hardware addresses coincide with the IP addresses on the network. The MitM attack is initiated by sending gratuitous ARP commands to confuse each host. These ARP commands tell the two hosts that the attacker computer is really the computer they want to send data. When a successful MitM attack is performed, the hosts on each side of the attack are unaware that their network data is taking a different route through the attacker’s computer. The attacker computer then needs to forward all packets to the intended host so the connection stays in sync and does not time out. Figure 13 illustrates a typical MitM attack.

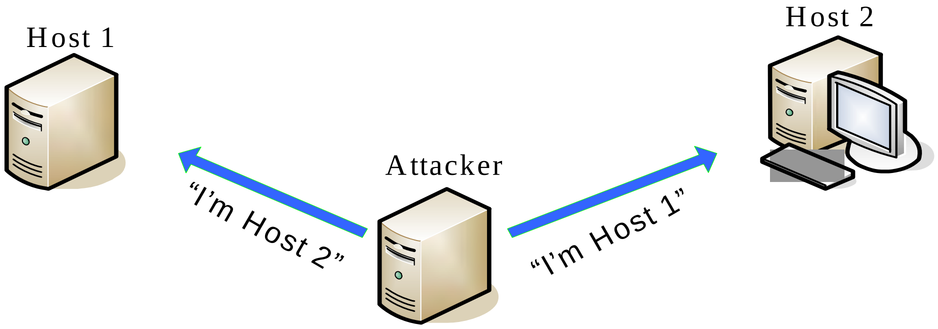

Figure 13. Generic Man-in-the-Middle attack.

The MitM attack is effective against any switched network because it effectively puts the attacker computer between the two hosts. This means the hosts send their data to the attacker's (compromised) computer thinking it is the host they intended to send the data. The attacker generally needs to be able to compromise a host on (or between) the victim computers' LANs.

With a full ARP MitM attack in place, manipulation of ICS devices and/or modification of data flowing back to the operator's console to give false information of the state of the system (spoofing) can occur. This tampering could allow an attacker to manipulate the system or the operator's response.

Recommendation: Data integrity checks need to be designed and implemented in ICS communication protocols. Use hardcoded ARP tables for static IP addresses or dynamic ARP inspection of dynamic IP addresses, if feasible. Monitoring the network traffic for changing media access control (MAC) addresses using an Intrusion Detection System (IDS), such as ARPWatch, can help detect MitM attacks. Using port security on all network equipment is another good practice, which helps protect against unauthorized physical connections into the network.

The vulnerabilities that were exploited by the assessment team are inherent in the protocols. The only recommended mitigations for field device protocols are to change to a secure alternative protocol or to tunnel the traffic over an encrypted channel that would require "bump-in-the-wire" devices to handle the encryption, at least on the field end.

Reworking the protocol with sequence numbers that are more difficult to predict and incorporating authentication is another option, but this would be expensive and difficult to retrofit to the existing installed base.

Access to an operational ICS network and ICS devices should be restricted. Authentication and/or encryption of the firmware upgrade process should be required. Requiring physical access to upgrade a controller prevents unauthorized remote firmware downloads.

4.1.4 Information Disclosure

An information leak is the intentional or unintentional disclosure of information to an actor that is not explicitly authorized to have access to that information. The information is either (1) regarded as sensitive within the product's own functionality, such as a private message or (2) provides information about the product or its environment that could be useful in an attack, but is normally not available to the attacker such as the configuration of the process being controlled.

Credentials sent across the network in clear text leave the system at risk to the unauthorized use of a legitimate user's credentials. If attackers are able to capture usernames and passwords, they will be able to log onto the system with that user's privileges. Any unencrypted information concerning the ICS source code, topology, or devices is a potential benefit for an attacker and should be limited.

One of the greatest security issues the assessment teams have identified is the widespread use of unencrypted plain-text network communications protocols. Many applications and services use protocols that include human-readable characters and strings. Network “sniffing” tools, many of which are freely downloadable, can be used to view this type of network traffic. As a result, the content of the ICS communication packets can be intercepted, read, and manipulated. Vulnerable data in this scenario include usernames, passwords, and ICS commands. Examples of these applications and services are proprietary ICS protocols and remote access services, such as telnet, File Transfer Protocol (FTP), and remote shell (rsh), which do not even encrypt the password or obfuscate it with a one-way hash function.

4.1.4.1 Unencrypted Proprietary ICS Protocol Communication

Clear-text communications without authentication and integrity checks offer an attacker the opportunity to intercept and alter the communications. The captured communications may be used to reverse engineer the proprietary protocols and modify or insert commands in ways that suit the attacker’s purpose.

All systems evaluated used clear-text protocols on the ICS network. Most of these protocols were proprietary to the system vendor. However, being unpublished does not equate to being secure because reverse engineering was accomplished on each system tested. To reverse engineer a protocol, network packets are captured and analyzed for patterns to understand the inner workings of the protocol. Once a protocol has been reverse engineered, an attacker can use it to perform unauthorized operations on the system (using a MitM or replay attack) and to keep such operations undetected by spoofing the operator’s console. A more advanced skill level is required to reverse engineer a protocol.

The following are specific assessment findings associated with this vulnerability:

- The backbone proprietary ICS communication protocol is clear text and susceptible to being reverse engineered
- Proprietary ICS distributed communications protocol used for Web HMI communication
- Unencrypted historian protocol used to connect to an untrusted network
- Communication between the controller and HMI was unencrypted and susceptible to reverse engineering
- Proprietary clear-text protocol between controller and HMI that allowed for easy packet monitoring.

Recommendation: When possible, standard secure versions of protocols should be used. When proprietary protocols are used, ideally, they should be encrypted and every message’s integrity validated. In situations where encryption of messages or provision for encrypted channels is infeasible, access to the proprietary protocols and associated communications should be kept to a minimum level and, preferably, kept within the confines of a well-protected ICS security zone.

4.1.4.2 Unencrypted Nonproprietary ICS Protocol Communication

Clear-text communications without authentication and integrity checks offer an attacker the opportunity to intercept and alter the communications. The captured communications may be used to better understand the specific structural function of the various field devices in the system, and to modify or insert commands in ways that suit the attacker’s purpose.

The following is a specific common assessment finding associated with this vulnerability:

- Communication between the controller and process equipment was unencrypted and susceptible to MitM and reverse engineering to alter values to and from field devices.
- OPC communications cross the firewall. OPC uses standard Distributed Component Object Model (DCOM)/Remote Procedure Call (RPC) protocols that are decipherable. These communications are able to be decoded and reveal what methods and objects are being passed between the client and server.

Recommendation: Future protocols should be designed with greater security including encrypted messaging. If possible, immediate application of encrypted channels would be beneficial. If supported by the field devices, configure the field equipment to only allow connections from the IP addresses of the systems that are expected to connect to those devices. While not preventing information leakage, this mitigation could make a successful attack more difficult.

4.1.4.3 Unencrypted Services Common in IT Systems

Unsecure services developed for IT systems have been adopted for use in ICS for common IT functionality. Although more secure alternatives exist for most of these services, active unused or obsolete services still exist in many ICS. Unfortunately, lingering obsolete services in ICS led to vulnerabilities readily accessible to an attacker who has gained a toehold on the ICS or has access to an unencrypted communication channel to or from the system.

If an attacker is able to capture a username and password, he is able to legitimately log onto the system with that user's privileges. In addition, in order to strategically attack an ICS, the attacker must perform discovery of the particular ICS environment. Attackers accomplish this by monitoring the ICS communication traffic to see which computers are performing specific functions and the protocols and commands used. These issues illustrate the security issue of data confidentiality. For this reason, plain-text protocols should be eliminated where possible and at a minimum, plain-text remote access services should be replaced with encrypted services such as secure shell (SSH). Encrypting other communications, such as proprietary ICS protocols, is a complex task and should be carefully addressed by the system vendor. Another issue to consider prior to encrypting everything is that it prevents the ability to implement network-monitoring tools on communication channels.

The following are specific assessment findings associated with this vulnerability:

- Use of clear-text Information Technology (IT) protocols on ICS LAN (e.g., telnet, FTP, "r" services) identified in multiple assessments
- Network file system, which has relatively limited security features, is used as the network file system
- Telnet access available on controller
- Post authentication sniffing or hijacking opportunities available on the dial-up connection.

Recommendation: Encryption is a direct answer to information leaks due to clear-text communication. Unfortunately, encryption is not always feasible on ICS networks. Timing concerns may make encryption impractical, and in addition, encryption reduces the ability to monitor network traffic and to troubleshoot the system.

Unsecure versions of common IT services should be replaced where possible by their secure versions. ICS use common IT protocols for common IT functionality, such as network device management, remote logins, or file transfers. Because they are not used for real-time functionality, they can be replaced with their secure counterparts in most cases. SSH can replace all file transfer and remote login protocols such as FTP, telnet, and rlogin with encrypted versions. Any communication can be "tunneled" through SSH. HTTP can be sent over the Secure Socket Layer (HTTPS). Users of these products should be aware more secure network file sharing solutions available. ICS vendors and customers should follow IT security practices and use the current secure versions of common protocols. When replacement is not feasible, access to the services should be minimized, and unencrypted communication should be limited to within the ICS whenever possible. Communications between security zones should be secured as much as possible.

4.1.4.4 Open Network Shares on ICS Hosts

The storage of ICS artifacts, such as source code and system configuration on a shared file system, provides significant potential for information mining by an attacker.

The following are assessment finding examples associated with this vulnerability:

- Publically available network shares on ICS hosts
- Two shares discovered on work station and server computers
- Common shares on multiple systems
- Files available for read access
- Information leak through shared directories
- Large number of publically available network shares on ICS hosts
- The source code for the ICS is shared on ICS hosts. Source code could be downloaded and used to find vulnerabilities.

Recommendation: Share files to only the computers and accounts that require them. Restrict the read and write permissions of these shared files and directories to the minimum required for each user. Restrict ability to create network shares to the users that need this functionality (generally administrators). Use network segmentation and firewall rules that block access to file sharing ports (e.g., TCP Port 139 and 445 on Windows systems).

4.1.5 Poor Patch Management

Vulnerabilities in ICS can occur because of flaws, misconfigurations, or poor maintenance of their platforms, including hardware, operating systems, and ICS applications. These vulnerabilities can be mitigated through various security controls, such as OS and application patching, physical access control, and security software (e.g., antivirus software).

A computer system is vulnerable to attack from the time a vulnerability is discovered and publicly disclosed, to when a patch is generated, disseminated, and finally applied. The number of publicly announced vulnerabilities has been steadily increasing over the past decade to the point where patch management is a necessary part of maintaining a computer system. Although patching may be difficult in high-availability environments, unpatched systems are often trivial to exploit due to the ease of recognizing product version and the readiness of exploit code.

4.1.5.1 Unpatched or Old Versions of Third-party Applications Incorporated into ICS Software

In multiple assessments, unpatched or old versions of applications were built into the ICS. Some had newer versions available just for security fixes. These applications possess vulnerabilities that may provide an attack path into the system. The software is well known, and available exploit code makes them an easy target.

The following are assessment finding examples associated with this vulnerability:

- Vulnerable database version
- Vulnerable Web server version
- OPC relies on RPC and DCOM—without updated patches, OPC is vulnerable to the known RPC/DCOM vulnerabilities.

Recommendation: The vendor bears responsibility to incorporate the latest versions of third-party (and OS) software into the current version of the ICS product. The vendor should also support customers in patch testing and providing patches for their own software.

4.1.6 ICS Performs Weak Authentication

Even if a protocol provides for strong authentication, implement correctly with strong passwords that remain private. Users are responsible for creating and protecting authentication credentials. Application developers are responsible for supporting strong passwords and protecting authentication credentials in the software. System integrators and administrators are responsible for configuring the systems to require and protect strong passwords as well.

4.1.6.1 ICS Uses Standard IT Protocol with Weak Encryption

Some standard IT encryption protocols used in assessment systems were exploited due to encryption weaknesses. A published attack was used in multiple assessments to crack a terminal service encryption and view the user credentials during authentication.

The following are common specific assessment findings associated with this vulnerability:

- Remote display application encryption can be cracked
- LAN Manager (LM) password hashes found in ICS network traffic.

Recommendation: Perform the necessary background research before choosing and properly implementing an encryption solution. Keep informed on published vulnerabilities and weaknesses of the deployed protocols and keep patches up-to-date.

The use of LM password hashes is a bad practice due to the easy decoding provided by tools such as John the Ripper and the Rainbow Tables. Users must assume that any passwords used on the network that were stored as LM hashes are compromised. Prevent storage of the LM hash if it is not needed for backward compatibility. Windows 2000 and later systems also create stronger NT LAN manager (NTLM) hashes, but create LM hashes for interoperability with older Windows systems.

4.1.6.2 Use of Standard IT Protocol with Clear-text Authentication

Clear-text authentication credentials can be sniffed and used by an attacker to authenticate the system.

The following are sanitized findings associated with this vulnerability from multiple assessments:

- Standard IT clear-text authentication protocol services are running on multiple ICS hosts
- Telnet access available on controller
- Clear-text IT protocols are used by the ICS (e.g., telnet, FTP, “r” services). This finding was common to multiple assessments.

Recommendation: Reduce the number of necessary services as much as possible. If necessary services are vulnerable to attack, these services should be replaced with more secure counterparts. For example, the clear-text protocols FTP, telnet, rshell, rexec, and rlogin can be replaced with SSH and secure FTP (a straightforward procedure for system access). This effort is not trivial if these services are integrated into the system functionality and may require rewriting code, architecting secure authentication, or even reengineering system communications.

4.1.6.3 Client-Side Enforcement of Server-Side Security

Applications that authenticate users locally trust the client that is connecting to a server to perform the authentication. Because the information needed to authenticate is stored on the client side, a moderately skilled hacker may easily extract that information or modify the client to not require authentication.

The following are specific assessment findings associated with this vulnerability:

- Client side validation of HMI application username
- Client side user and password validation for remote controller configuration.

Recommendation: Implement robust authentication by the server or component that is granting access.

4.1.7 Least User Privileges Violation

4.1.7.1 Unauthorized Directory Traversal Allowed

Findings were reported that directory traversal was allowed beyond intended file access. Either remotely connecting to a Web server, database, open network share or proprietary ICS application can accomplish this task.

The following are specific assessment findings associated with this vulnerability:

- Proprietary ICS protocol allowed ICS system hosts to read or overwrite files on other hosts, without any logging
- Documentation and configuration information was being shared freely (read only)
- Common shares on multiple systems
- Oracle listener allows arbitrary file write and read/write database access.

Recommendation: Ensure share permissions for nonessential folders are removed. Whenever possible, shared folders should only allow read access. Ensure that even read-only shares are not providing critical information to public queries.

4.1.7.2 Services Running with Unnecessary Privileges

Services are restricted to the user rights granted through the user account associated with them. Exploitation of any service could allow an attacker a foothold on the ICS network with the exploited service's permissions. Privilege escalation can be accomplished by exploiting a vulnerable service running with more privileges than the attacker has currently obtained. If successfully exploited, services running as a privileged user would allow full access to the exploited host.

This vulnerability was very common. The following are some specific assessment findings associated with this vulnerability:

- Manager account overused
- Remote exploitation of ICS application services allowed root-level access on ICS hosts
- Database service running as administrator.

Recommendation: By default, some ICS installations start services as the root user and root group. Many services do not need to be started with this privilege level, and doing so exposes system resources to preventable risks. By restricting necessary privileges during ICS design and implementation, the window of exposure and criticality of impact is significantly reduced in the event that a flaw is found in that service. Essentially, running with minimum privileges is a recommended practice because it reduces the potential harm that a service can cause due to a bug, accident, or malicious exploit. The most secure service available should be used for a given functionality and then kept patched and up-to-date to help prevent exploitation.

4.1.8 Summary of Common ICS Software Vulnerabilities

ICS software mostly suffers from the lack of secure software design and coding practices. ICS network protocols and associated server applications are prone to MitM data viewing and alteration, as well as compromise through invalid input. This lack of security culture contributes to poor code quality, network protocol implementations that rely on weak authentication and allow information disclosure, and vulnerable custom ICS Web services.

ICS software generally uses third-party applications such as common Web servers, remote access services, and encryption services. Many out-of-date and vulnerable third-party software applications and services have been identified on new ICS version; all indications show that the ICS vendor is not supporting third-party patch management for their software.

Table 3 lists the ICS software categories and vulnerabilities identified in multiple CSSP assessments. Figure 14 shows the category findings' frequency of occurrence.

Table 3. Summary of common ICS software assessment findings.

Category	Common Vulnerability
Poor Code Quality	Use of potentially dangerous functions in proprietary ICS application
Vulnerable Web Services	Directory traversal enabled
	Unauthenticated access to Web server
Poor Network Protocol Implementations	Lack of input validation: Buffer overflow in ICS service
	Lack of input validation: Lack of bounds checking in ICS service
	ICS protocol uses weak authentication
	ICS protocol uses weak integrity checks
	ICS product relies on standard IT protocol that uses weak encryption
Poor Patch Management	Unpatched or old versions of third-party software incorporated into ICS software
Weak Authentication	ICS use standard IT protocol with weak encryption
	Use of standard IT protocol with clear-text authentication
	Client-side enforcement of server-side security
Least User Privileges Violation	Unauthorized directory traversal allowed
	Services running with unnecessary privileges
Information Disclosure	Unencrypted proprietary ICS protocol communication
	Unencrypted nonproprietary ICS protocol communication
	Unencrypted services common in IT systems
	Open network shares on ICS hosts

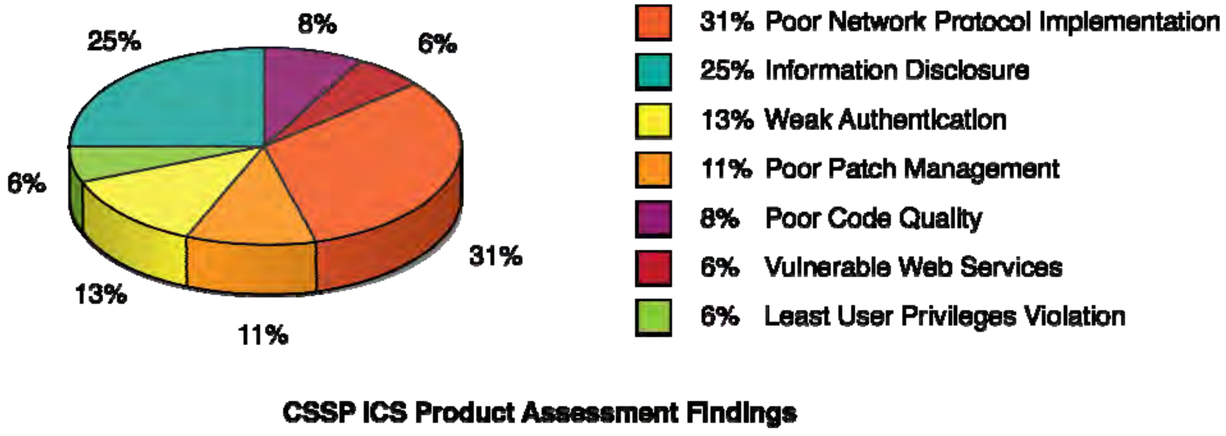

Figure 14. CSSP assessment findings related to ICS products.

4.2 Common ICS Configuration (Implementation) Vulnerabilities

Vulnerabilities in the previous section are inherent in the ICS products. Other vulnerabilities can be introduced by the way the ICS is installed and maintained. Each ICS installation is a unique combination of components and functionality offered by an ICS product vendor. ICS are generally such major purchases in time and money required that very few systems from each ICS product line are delivered before features are added and a new version is released. A large investment of financial and personnel resources needed for ICS upgrade contributes to a lack of, or insufficient, standard procedures for securely configuring each ICS product.

All vendors have different standard processes for building, testing, and installing an ICS. Some vendors have integrators who work with customers to create and install the system. Other vendors have just a product model. Often, integration consultants with specific ICS product training are available for installation and configuration. All systems are unique; generally with new features introduced in each one, the level of security in each ICS installation is dependent on those responsible for installing and configuring the operating systems, ICS applications, and third-party applications.

Common security problems that can arise from ICS configuration are unpatched OS, application, and service vulnerabilities, failure to configure and implement applications and services securely (i.e., selecting security options and protecting credentials), changing all default passwords, setting password policies to require strong passwords, limiting user accounts, applications and services to only the required permissions, installing or enabling security features correctly, and restricting unnecessary connections.

Assurance of a secure configuration can be increased through automated security configuration packages and detailed instructions provided by the ICS vendor. Automated disabling of unnecessary services and applications and lists of required applications and services with associated permissions required should be included in instructions. Required ports and components allowed to connect should also be defined. Owners should require this information during the procurement process to insure the ability to securely configure their systems.

Although some vulnerability is inherent in ICS products, many ICS component vulnerabilities are dependent on how an ICS product was implemented. Even though security configuration can be limited by the design of the ICS, ICS owners can control their risk of cyber attack by securely configuring their systems.

The ICS assessment findings that are due to installation and configuration errors are described below. These issues also apply to the maintenance of the operational ICS.

4.2.1 Poor Patch Management

Change management is paramount to maintaining the integrity of both IT and ICS. Unpatched software represents one of the greatest vulnerabilities to a system. Software updates on IT systems, including security patches, are typically applied in a timely fashion based on appropriate security policy and procedures. In addition, these procedures are often automated using server-based tools. Software updates on ICS cannot always be implemented on a timely basis because these updates need to be thoroughly tested by the vendor of the industrial control application and the end user of the application before being implemented and ICS outages often must be planned and scheduled days/weeks in advance. The ICS may also require revalidation as part of the update process. Another issue is that many ICS utilize older versions of operating systems that are no longer supported by the vendor. Consequently, available patches may not be applicable. Change management is also applicable to hardware and firmware. The change management process, when applied to ICS, requires careful assessment by ICS experts (e.g., control engineers) working in conjunction with security and IT personnel.

4.2.1.1 Unpatched Operating System

Vulnerabilities that have had patches available for a long time are still being seen on ICS. Unpatched operating systems open ICS to attack through known operating system service vulnerabilities. For example, in 2003 the Slammer worm disabled an Ohio Davis-Besse nuclear power plant safety monitoring system for nearly 5 hours. The Davis-Besse plant was in a maintenance cycle at this time and not generating power. According to reports, plant computer engineers had not installed the patch for the Microsoft SQL vulnerability that Slammer exploited. In fact, they did not know there was a patch, which Microsoft released 6 months before Slammer struck.[3]

The following are sanitized findings associated with this vulnerability from multiple assessments:

- Operating system vendor patches were not applied
- System computers vulnerable to operating system service vulnerabilities
- Vulnerable version of Sendmail
- Sun rpc.cmsd has an integer overflow problem in xdr_array
- Vulnerable version of RPC
- Inconsistent application of current patches on HMIs.

Recommendation: A timely patch management process is critical to reduce vulnerabilities. OS patches repair vulnerabilities in the OS that could allow an attacker to exploit the computer. The importance to system security of keeping OS patches up-to-date cannot be over emphasized. However, patching ICS machines can present unique challenges. Among the factors to consider are system functionality, security benefit, and timeliness. This process requires elements of IT, IT security, process control engineering, and senior management and incorporates elements of an Incident Response Plan, a Disaster Recovery Plan, testbed testing, and a Configuration Management Plan. Where patching is not an option, work-arounds and defense-in-depth techniques and tactics can be used.[4]

Statically linked libraries need to be independently kept up-to-date if they are different from the libraries associated with the operating system.

4.2.2 Weak User Authentication

4.2.2.1 Improper Security Configuration

A common problem found during assessments was that even though secure authentication applications were used, installations and configurations were not correct.

The following are specific assessment findings associated with this vulnerability:

- Security options not enabled
- OPC client and server use the Microsoft LM authentication process, which contains known vulnerabilities.

Recommendation: Instructions for secure installation and proper configuration for each application need to be followed and tested. Do not allow login information to be hard coded into scripts and user programs or stored so that reauthentication on that computer is never required again.

LM hashes should be disabled on all Windows hosts and domain controllers. If LM authentication is required, update the configuration settings so that only the new NTLM network authentication is used. Because LM hashing does not support passwords longer than 14 characters, users can prevent a LM hash from being generated for their password by using a password at least 15 characters in length.

4.2.2.2 No Password Required

Some assessments discovered applications that had been configured without passwords, which means that anyone able to access these applications are guaranteed to be able to authenticate and interact with them.

The following are specific assessment findings associated with this vulnerability:

- Database service was configured without a password on multiple assessments
- Null connection allows remote hosts to query each system for information without requiring authentication
- Password length can have zero characters. Any user on the system can have a blank password.

Recommendation: Strong passwords need to be required and deployed on networking, client, and server equipment. Passwords should be implemented on ICS components to prevent unauthorized access.

4.2.2.3 Weak Passwords

Poorly chosen passwords can easily be guessed by humans or computer algorithms to gain unauthorized access. The longer and more complex a password is, the time to guess or crack the password increases. Cracking a password can be trivial or virtually impossible depending on the combination of different character types used with larger password length.

The following are specific assessment findings associated with this vulnerability:

- Some ICS hosts had very weak 3-character administrative passwords
- The weak passwords were recovered and provided root-level access to all system resources
- Default SNMP community string was used by 89 hosts
- Several weak passwords were found
- Default password had not been changed.

Recommendation: A policy mandating the use of strong passwords for all cyber assets inside the electronic perimeter with a reasonable lifespan limit needs to be mandated and enforced. Usage of common administrative passwords needs to be discouraged.

4.2.2.4 Weak Password Requirements

Password policies are needed to define when passwords must be used, how strong they must be, and how they must be maintained. Without a password policy, systems might not have appropriate password controls, making unauthorized access to systems more likely. Passwords that are short, simple (e.g., all lower-case letters), or otherwise do not meet typical strength requirements are vulnerable to being cracked. Password strength also depends on whether the specific ICS application was designed to support more stringent passwords.

The following are specific assessment findings associated with this vulnerability:

- Many of the accounts, including the administrator account, had no password expiration date
- Account lockout policy not defined
- Password complexity disabled
- Password history set to remember zero previous passwords.

Recommendation: Password policies should be developed as part of an overall ICS security program taking into account the capabilities of the ICS and its personnel to handle more complex passwords. System administrators should enforce the usage of strong passwords. A password strength policy should contain the following attributes: (1) minimum and maximum length (2) require mixed character sets (alpha, numeric, special, mixed case); (3) do not contain user name; (4) expiration; and (5) no password reuse. Authentication mechanisms should always require sufficiently complex passwords and require that they be periodically changed. [42]

4.2.3 Information Disclosure

4.2.3.1 Weak Protection of User Credentials

User credentials should be vigorously protected and made inaccessible to an attacker. Whenever credentials are passed in clear text, they are susceptible to being captured and then cracked if necessary by the attacker. If stored password hashes are not properly protected, they may be accessed by an attacker and cracked. In every case, the lack of protection of user credentials may lead to the attacker gaining increased privileges on the ICS and thus being able to more effectively advance the attack.

The following are specific assessment findings associated with this vulnerability:

- Services such as FTP, telnet, and rlogin transmit user credentials in clear text
- OPC client responds with both newer NTLM and older LM password hashes, making discovery of passwords easier
- Password hash files are not properly secured
- Found LM password hashes.

Recommendation: Properly secure password files by making hashed passwords more difficult to acquire (e.g., restrict access by using a shadow password file or equivalent on UNIX systems). Replace or modify services so that all user credentials are passed through an encrypted channel.

LM password hashes are crackable by freely available tools within seconds. All Windows hosts support LM passwords and all versions before Windows Vista and Windows Server 2008 compute and store passwords using the LM hash algorithm by default. LM hashes should be disabled on all Windows hosts and domain controllers. OPC Client security policies should be configured so that only the NTLM response is given. Because LM hashing does not support passwords longer than 14 characters, users can prevent a LM hash from being generated for their password by using a password at least 15 characters in length.

4.2.3.2 Information Leak through Insecure Service Configuration

Information that can be used in determining system vulnerabilities can be gathered from services that have been configured to reply with debug or other information. For example, null sessions could be used to enumerate user accounts on the system and allow the use of available network shares. An attacker can use this sort of information to configure a sniffer more accurately or to check if the attacker already has login information for one of the given accounts. This information also could be potentially useful for a social engineering attempt.

The following are specific assessment findings associated with this vulnerability:

- Controller answered to an ICMP_MASKREQ query
- Controller answered to an Internet Control Message Protocol (ICMP) timestamp, which allows an attacker to know the date that is set on the machine
- Nessus was able to gather NetBIOS names from the HMI and was able to determine the MAC address for the network adapter
- Information leak through debug information: Web interface had debugging enabled
- Information leak through directory viewing: Web interface allowed directory viewing
- DNS queries indicate hostnames reveal function of host (e.g., antivirus, keyserver)

- Null sessions enabled
- ICMP responses from the router were different based on whether the target address referenced a valid machine.

Recommendation: Any information that is not necessary to the functionality should be removed in order to lower both the overhead and the possibility of security sensitive data being sent.

4.2.4 Summary of Common ICS Configuration Vulnerabilities

Table 4 lists the common vulnerabilities related to ICS configuration issues that were identified with the assessment activities for the CSSP. Figure 15 shows that the most ICS configuration problems found were due to weak passwords and password policies.

Table 4. Summary of common ICS configuration findings.

Category	**Common Vulnerability**
Poor Patch Management	Unpatched operating system
Weak User Authentication	Improper security configuration
	No password required
	Weak passwords
	Weak password requirements
Information Disclosure	Weak protection of user credentials
	Information leak through insecure service configuration

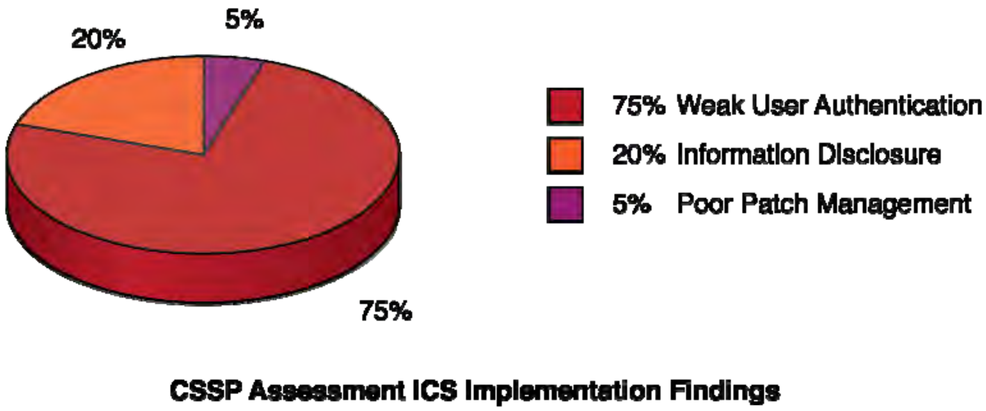

Figure 15. CSSP assessment ICS implementation findings.

4.3 Common ICS Network Vulnerabilities

The network architecture needs to be securely designed and implemented to allow remote control and monitoring of a process and provide process data for business functions while preventing any other traffic from entering or leaving the control network. Security zones with access control rules that limit the traffic allowed in and out of the zone, will reduce the risk of intentional or unintentional attacks from sources outside the zones, to attacks from allowed IP addresses that exploit the protocols allowed through the given security zone's perimeter. The security features built into the protocols used to transfer data in and out of the control network must be relied on to prevent attacks that pass access control requirements. Security features, such as authentication and integrity checks, can be wrapped around unsecure protocols that must be used for communication with the ICS, making understanding the limitations of protection they do and *do not* provide essential for proper implementation.

An effective cyber security program for an ICS should apply a strategy known as “defense-in-depth,” layering security mechanisms such that the impact of a failure in any one mechanism is minimized.

4.3.1 Common ICS Network Design Vulnerabilities

The network infrastructure environment within the ICS has often been developed and modified based on business and operational requirements, with little consideration for the potential security impacts of the changes. Over time, security gaps may have been inadvertently introduced within particular portions of the infrastructure. Without remediation, these gaps may represent backdoors into the ICS.

4.3.1.1 Lack of Network Segmentation

Minimal or no security zones allows vulnerabilities and exploitations to gain immediate full control of the systems, which could cause high-level consequences.

The following are specific assessment findings associated with this vulnerability:

- Lack of internal segmentation of the ICS production network: ICCP servers not on DMZ
- Lack of internal segmentation of the ICS production network: Host with dedicated serial link for data transfer using high-risk application not on DMZ
- Control-related systems are accessible on the corporate LAN
- Connections to and from remote facilities to the ICS do not pass through a firewall.

Recommendation: At a minimum, the ICS network should be separated from the corporate network by a firewall, and a DMZ should be implemented to provide the corporate network access to the required information from the ICS network. The systems located in the DMZ are not production systems and also should be treated as hostile. Exceptions between the DMZ and the ICS networks should be kept to an absolute minimum, and exceptions from the corporate to the ICS should be eliminated. Additional security zones can be created within these segments.

4.3.1.2 Firewall Bypassed

Backdoor network access is also not recommended and could cause direct access to ICS for attackers to exploit and take full control of the system.

The following are specific assessment findings associated with this vulnerability:

- Physical cables connected directly to the ICS LAN, bypassing firewall
- SSH server bridges corporate and ICS LANs, bypassing firewall
- Third network card on ICCP server connects directly to ICS LAN.

Recommendation: A firewall should limit access to the different LAN segments to only necessary communication. The ICS network should be separated from the corporate network by a firewall, and a DMZ is implemented to provide the corporate network access to the required information from the ICS network. The systems located in the DMZ are not production systems and should be treated as hostile. Exceptions between the DMZ and the ICS networks should be kept to an absolute minimum, and exceptions from the corporate to the ICS should be eliminated.

4.3.2 Common ICS Network Component Configuration (Implementation) Vulnerabilities

Firewall and router filtering deficiencies allow access to ICS components through external and internal networks. The lack of incoming access restrictions creates access paths into critical networks.

The lack of outgoing access restrictions allows access from internal components that may have been compromised. For an attacker to remotely control exploit code running on the user’s computer, a return connection must be established from the victim network. If outbound filtering is implemented correctly, the attacker will not receive this return connection and cannot control the exploited machine.

4.3.2.1 Access to Specific Ports on Host Not Restricted to Required IP Addresses

Detailed findings under this common vulnerability involve firewall rules restricting access to specific ports, but not IP addresses. A common finding was that network device access control lists did not restrict management access to the required IP addresses.

Another common detailed finding was that firewall rules allowed access to unused IP addresses traceable to legacy configuration of the firewall allowed access to unused IP addresses. This finding illuminates an attack path by using this IP address in order to be allowed through the firewall.

The remaining specific assessment details associated with this vulnerability involved access to specific ports being given to either an entire address space or were not restricted by an IP address at all. Assessment findings that fall under this vulnerability are firewall rules that are based on address groups that include a wider range than should be allowed.

The following are specific assessment findings associated with this vulnerability:

- More configuration needs to be performed on the personal firewalls
- Router inside and outside interfaces had 24-bit netmask rather than 16-bit
- Access lists defined but not applied. No inbound filtering
- Access lists incorrect for required ports
- Access to network printer services on corporate LAN was not restricted by password protection or access control list
- E-mail client on DMZ had access to corporate LAN and Internet
- Inadequate outgoing access restrictions.

Recommendations: Firewall rules that apply to functional groups should use defined finite groups that are restricted to required IP addresses. Firewall rules that are no longer needed should be removed as part of a change management procedure or periodic system review or audit. Access control lists should be used to limit management access of network equipment to only those who need it.

4.3.2.2 Port Security Not Implemented on Network Equipment

Unauthorized network access through physical access to network equipment includes the lack of physical access control to the equipment, including the lack of security configuration functions that limit functionality even if physical access is obtained. The common finding was a lack of port security on network equipment. A malicious user who has physical access to an unsecured port on a network switch could plug into the network behind the firewall to defeat its incoming filtering protection.

Recommendation: Port security should be implemented to limit connectivity to hardware interfaces. Given the static nature of ICS environments, port security can be used to ensure MAC addresses do not change and new devices are not introduced to the network. Actions, such as limiting known MAC addresses to specific interfaces and disabling unused interfaces, should be implemented to assist in network security. Given the static nature of the environment, port security can be used to ensure MAC addresses do not change and new devices are not introduced to the network.

4.3.3 Summary of Common ICS Network Vulnerabilities

Table 5 lists the common vulnerabilities related to ICS network vulnerabilities that were identified with the assessment activities for the CSSP. Figure 16 shows that most assessment findings were due to insufficient access controls.

Table 5. Summary of common ICS assessment network findings.

Category	Common Vulnerability
Network Design Vulnerabilities	Lack of network segmentation
	Firewall bypassed
Network Component Configuration (Implementation) Vulnerabilities	Access to specific ports on host not restricted to required IP addresses
	Port security not implemented on network equipment

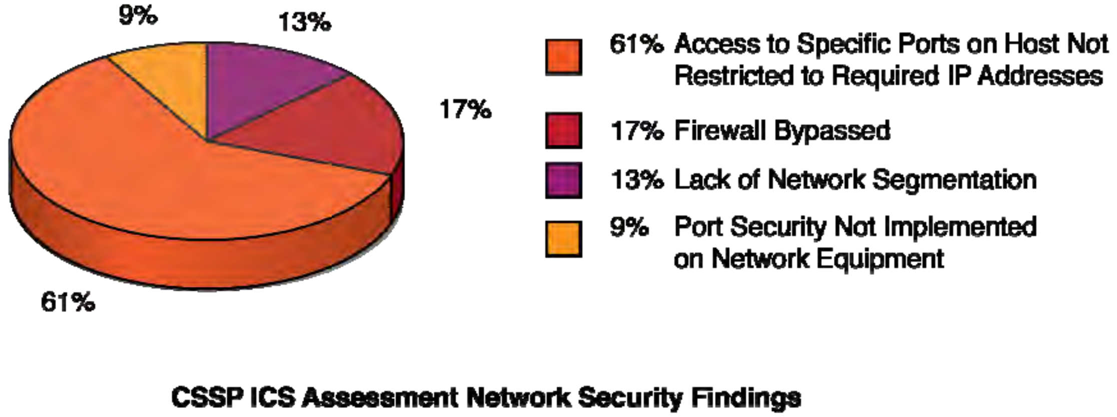

Figure 16. CSSP assessment network security findings.

5. ICS SECURITY RECOMMENDATIONS

ICS vendors and owners can learn and apply many common computer security concepts and practices to secure and protect their systems. Security should be designed and implemented by qualified security and ICS experts who are able to verify that the solutions are effective and can make sure that the solutions do not impair the system's reliability and timing requirements.

ICS vendors and asset owners are encouraged use this report as a guide to help focus further efforts to improve the overall security of their systems. They should investigate whether the identified vulnerabilities affect their systems and if so, follow the recommendations in this report along with more detailed and tailored recommendations from other resources. The classes of vulnerabilities identified in this report can help identify problem areas for self-assessment activities that can be conducted to identify and mitigate vulnerabilities in ICS networks, components, services, and code.

By mitigating the vulnerabilities identified in this report, an ICS can be made more secure, but additional vulnerabilities most likely exist in all systems. The path to a more secure system is a continuous journey and as new attack scenarios are identified or developed, new defenses must be implemented. In addition to the specific mitigations and recommendations made for the vulnerabilities called out in the previous sections of this report, several general recommendations are given below.

ICS have different performance and reliability requirements and use operating systems and applications that may be considered unconventional to typical IT support personnel. Furthermore, the goals of safety and efficiency can sometimes conflict with security in the design and operation of ICS (e.g., requiring password authentication and authorization should not hamper or interfere with emergency actions for ICS.) All security solutions must not compromise critical functionality. All security functions integrated into the ICS must be tested (i.e., offline on a comparable ICS) to prove that they do not compromise normal ICS functionality.

In order to reduce the risk of a successful attack against an ICS, the likelihood of a high-impact incident can be reduced by implementing as many perimeter protection and vulnerability reduction strategies as possible (aka defense-in-depth). A mitigation strategy should not be chosen from a list of possible mitigations for a given identified or possible vulnerability. As many mitigation techniques as reasonably possible should be employed to stand in a line of defense and prevent access to vulnerable components and network traffic. The probability that an attack is able to defeat or circumvent security defenses is increasingly reduced as the number of security measures are implemented and gaps are filled in the line of protection formed by the other security features on the ICS. However, the risk of the layers of defense to the operation of the ICS must be considered and mitigated as well.

The operational and risk differences between ICS and IT systems create the need for increased sophistication in applying cyber security and operational strategies. A cross-functional team of control engineers, ICS operators, and IT security professionals needs to work closely together to understand the possible implications of the installation, operation, and maintenance of security solutions in conjunction with ICS operation. IT professionals working with ICS need to understand the reliability impacts of information security technologies before deployment. Some of the OSs and applications running on ICS may not operate correctly with commercial-off-the-shelf IT cyber security solutions because of specialized ICS environment architectures.

5.1 Recommendations for Vendors

Vendors need to incorporate security into every phase of the product development life cycle and rely on manual and automated means to ensure proper bounds checking. Once products are deployed, vendors need to establish a process to manage and mitigate product security defects. The vendor team should consist of representatives of key business functions, such as product development, public relations, and legal. A single point of contact leads resolution on reported security issues and must assist asset owners in addressing reported security issues in a timely manner. Common industry practice is to host a "/security" Web page off the corporate main domain where information on security issues and the designated contact or team can easily be found. The vendor is responsible for responding to reported security concerns that include issue validation, patch development, patch testing and validation, and response coordination.

ICS security assessment reports show a common need to increase secure coding practices. The three most common problems are the lack of input validation, authentication, and integrity checks. The top nine ICS vendor recommendations are listed and then discussed below.

- Educate/train developers in secure coding and create a culture that emphasizes security
- Expeditiously test and provide security patches to affected customers
- Create the necessary communication paths that are needed to quickly notify customers of security problems, and create the methods needed to provide patches in an effective way
- Implement and strenuously test strong authentication and encryption mechanisms
- Dramatically increase the robustness of network parsing code
- Document how the systems use the network so that effective firewall and IDS rules can be created
- Pay for a third-party security source code audit and fix the problems identified during the audit
- Redesign network protocols to avoid common problems and enhance security
- Enhance test suites to perform more testing for failure with emphases on testing for potential vulnerabilities
- Create custom protocol parsers for common IDS so that they can be more effective.

5.1.1 Create a Security Culture

Educate/train developers in secure coding and create a culture that emphasizes security.

The security development lifecycle (SDL), created by Microsoft in 2002 as a response to heightened awareness of cyber security threats, is a high-visibility example of a security culture change. This process was developed to catch security flaws during the product development lifecycle, not just after the product is released. For example, Microsoft has created a culture that promotes safe code development by forcing all new code to pass a set of tests before incorporation into the main product. All developers were put through secure development training to support this new culture. Performance evaluation of software products, as well as the product managers and their teams, also changed to include a focus on security. Although new Microsoft vulnerabilities are still abundant 6 years later, this culture change has made a significant difference in the security level of Microsoft products.

ICS products have gained considerable attention in recent years as the cyber security threats due to connection to the Internet have been realized. Microsoft and other hardware, operating system, and software application vendors have experienced the cost and difficulties that arise from public announcement of security flaws to force quicker patch response time. Those companies willing to embrace a security culture change will benefit from fewer security patches for deployed systems and greater customer confidence and loyalty. Public announcements of ICS vulnerabilities are starting to appear and ICS protocol dissectors are becoming available.

ICS vendors must adapt to changing customer needs for security in the products used to control physical systems where compromise can have catastrophic consequences. Even Microsoft has experienced difficulty bolting security onto a mature product and impossible to find and prevent all bugs. Security must also compete with functionality for product time and budget. Vendors must accept that security improvements will require an investment. The sooner security is integrated into the product, the better chance it has of competing in a market where ICS products are required to survive cyber attack without compromising critical functionality.

ICS vendors should work toward a culture where software security best practices are adopted throughout the product development organizations and software development life cycles are adjusted to use the best practices. Security practices should be consolidated, integrated, and centralized into a security process that supports the defined strategy for creating the most secure product possible. Most important is a change in attitudes to a realization that security is important because it is associated with consequences for everyone. ICS vendors can create a security cultural change within their companies by incorporating ICS product security into personnel performance.

Numerous resources are available for information and training on building a security culture and software security best practices. ICS vendors can use the following software security best practices to create more secure products:

- Develop or acquire the necessary personnel security skills
- Define security requirements to protect critical functions
- Identify ICS component designs that violate security
- Develop secure design or redesign of identified components
- Require secure source coding handling to protect against malicious vulnerabilities
- Perform thorough security testing
- Provide security documentation.

Many ICS vulnerabilities are due to the lack of input validation. Programmers should be trained in secure coding practices to minimize vulnerabilities such as buffer overflows that are due to programmer oversight. All code should be reviewed and tested for input functions that could be susceptible to buffer overflow attacks. The C and C++ unsafe string and memory function calls should be replaced with their safe counterparts. Input validation should be used to ensure that the content provided to an application does not grant an attacker access to unintended functionality or privilege escalation. All input should be validated, not just those proven to cause buffer overflows. Input should be validated for length and buffer size should not be determined based on an input value. Even if values are never input directly by a user, data are not necessarily correctly formatted, and hardware or operating system protections are not always sufficient. Buffer overflows in applications that process network traffic can be exploited by intercepting and altering input values in transit. Therefore, network data bounds and integrity checking should be implemented as well.

As a layer of defense, compiler protection options should be used when compiling C/C++ code to increase the difficulty for an attacker to execute exploit code. This decreases the impact of a vulnerability from an exploit that allows the attacker to run commands on the computer or use it as a launching point along an attack path into the core of the ICS to a DoS-type attack.

5.1.2 Enhance ICS Test Suites

ICS product test suites should be enhanced to perform testing to failure with an emphasis on potential vulnerabilities. ICS software code logic has been found to only test for failures and other problems that may occur during normal operations.

The design and code logic of ICS products should prevent all invalid or unwanted cases, even if they should never occur. ICS experts can be blinded by their goal of creating a system that works reliably and protects against normal failures and mistakes. The connection of ICS to other networks has created the threat of cyber attack. ICS test suites should include “out of the box” scenarios that test all kinds of input values and abnormal conditions. This requires tests built by individuals who can create comprehensive and “out of the box” scenarios and are not involved in the design and implementation of the ICS product.

The CSSP assessment methodology is based on this idea of identifying security weaknesses through an attacker’s perspective and communicating the security issues to the industry partner from this perspective. This testing approach has been a very successful in increase awareness of the “out-of-the-box” attack methods the ICS sector needs to defend against.

Resources such as the Common Attack Pattern Enumeration and Classification project can help in developing test packages:

> Building software with an adequate level of security assurance for its mission becomes more and more challenging every day as the size, complexity, and tempo of software creation increases and the number and the skill level of attackers continues to grow. These factors each exacerbate the issue that, to build secure software, builders must ensure that they have protected every relevant potential vulnerability; yet, to attack software, attackers often have to find and exploit only a single exposed vulnerability. To identify and mitigate relevant vulnerabilities in software, the development community needs more than just good software engineering and analytical practices, a solid grasp of software security features, and a powerful set of tools. All of these things are necessary but not sufficient. To be effective, the community needs to think outside of the box and to have a firm grasp of the attacker’s perspective and the approaches used to exploit software.
>
> Attack patterns are a powerful mechanism to capture and communicate the attacker’s perspective. They are descriptions of common methods for exploiting software. They derive from the concept of design patterns applied in a destructive rather than constructive context and are generated from in-depth analysis of specific real-world exploit examples.
>
> To assist in enhancing security throughout the software development lifecycle, and to support the needs of developers, testers and educators, the Common Attack Pattern Enumeration and Classification (CAPEC) is sponsored by the Department of Homeland Security as part of the Software Assurance strategic initiative of the National Cyber Security Division. The objective of this effort is to provide a publicly available catalog of attack patterns along with a comprehensive schema and classification taxonomy.”[5]

5.1.3 Create and Test Patches

Expeditiously test and provide security patches to affected customers. Create the necessary communication paths that are needed to quickly notify customers of security problems and create the methods needed to provide patches in an effective way. Currently, most ICS venders have poor methods of notifying customers about potential security problems and patches. Experience has shown that patches generated as the result of previous security assessments have been slow in being deployed with many end users unaware about the existence of the patches. ICS vendors should create and maintain security mailing lists and also test the procedures needed to notify the end users about security problems. Increasing accessibility for end users to obtain the necessary information will greatly increase the use and effectiveness of patching. Many ICS vendors do publish security information, but frequently locate this information in an obscure location on their website that can easily be overlooked. This information should have a more prominent location and should be easy for the users to find. If this advice is followed, ICS vendors would provide strong benefit by making it easier for end users to obtain and install patches.

Vendors should test and approve OS patches, along with all other third-party software. Products and services such as NTP should be kept at current version and patch levels prior to deployment at asset owner sites and be included in the patch testing process. ICS products that have third-party services and applications incorporated into their functionality should be designed so that these applications can be updated or replaced as easily as possible.

ICS vendor software vulnerabilities should be patched and made available to affected customers as well.

5.1.4 Redesign Network Protocols for Security

ICS network protocols and the service applications that implement them need to be redesigned for security. Most ICS network protocols were designed with the original ICS code base to be fast and only avoid failure issues and are not designed to provide robust authentication and integrity checks. Many ICS protocol designs contain common security pitfalls. A number of characteristics of a secure protocol are relevant to this discussion.

- Secure protocols should be simple. The more complex a protocol is, the higher the likelihood of bugs and vulnerabilities within the implementation.
- Protocols should also minimize duplicate data. If data appear multiple times within the protocol, then portions of the implementation will invariably use one version of the data while other portions use another version. This allows an attacker to put the implementation into an unknown state by sending conflicting versions of the data.
- Protocols with many optional fields and features are less secure because no two implementations will agree on what is optional and tend to make incorrect assumptions.
- Secure protocols are also targeted; they contain enough functionality to get the job done and nothing more. If protocols contain seldom used or never used components then those components tend to be more buggy and contain more vulnerabilities than the components that are actually being used because they will be tested to a lesser degree.
- Secure protocols also have secure authentication methods and options for encryption or data integrity. Security by obscurity cannot be relied on because insider knowledge or reverse engineering can be used to recreate valid network packets. Some ICS protocol analyzers have already been developed, and one should expect to see more given the increasing interest in ICS security.

When possible, network protocols should be redesigned to improve security by avoiding common security pitfalls, avoiding designs that lead to implementation issues, and by including secure authentication and encryption methods.

5.1.5 Increase Robustness of Network Parsing Code

Dramatically increase the robustness of network parsing code. Part of every network protocol is an associated program to build packets or process the traffic off the network. These applications are written by the ICS vendor for their propriety protocols as well as for common ICS protocols, such as OPC, ICCP, and Distributed Network Protocol Version 3 (DNP3). If these applications contain invalid input vulnerabilities such as buffer overflows, exploitation by anyone who is able to gain access to the ICS host and port is possible. Such action could cause a communication DoS, with an attacker gaining access to the computer with the privileges of the account service was running as, or other problems for the ICS.

Data integrity checks need to be designed and implemented into ICS communication protocols. The lack of, or weak, data integrity checks prevent a protocol from detecting bad data. An attacker can take advantage of the poor integrity checks to send malformed packets in order to cause DoS attacks or to trigger a buffer overflow and compromise the system. An attacker does not always have to send malformed packets for manipulation of otherwise valid alarm or command messages sent over the wire if the ICS protocol has poor integrity checks.

5.1.6 Create Custom Protocol Parsers for Common IDSs

ICS vendors should create parsers for their custom protocols that can be used by common IDSs. In this manner, intrusion detection monitoring is made more effective by providing the ability to watch for illegal or abnormal values in ICS traffic. The bulk of the current IDS technology is focused on detecting exploits, not vulnerabilities. These systems are not very effective in the ICS environment due to the lack of known exploits to detect. If dissectors for the ICS protocols exist, rules could be written for the IDSs that verify network messages are within reasonable bounds and attempt to detect an exploitation of vulnerability.

5.1.7 Document Necessary Services and Communication Channels

Document how the ICS system components use the network so that effective firewall and IDS rules can be created. For each ICS component, document the necessary services along with the associated port ranges and which components are allowed to initiate a connection to that component.

Provide complete documentation and/or automated setup of security features to allow for quicker, easier, and more consistent implementation of ICS components and security features. Security features that are obtuse or difficult to configure and implement are typically not used or are used incorrectly in the field installations of ICS. Security features that are inconsistently implemented or provide inconsistent results are considered a risk to reliability and availability of the ICS in an operational environment.

5.1.8 Implement and Test Strong Authentication and Encryption Mechanisms

Implement and strenuously test strong authentication and encryption mechanisms. Applications that process network traffic or accept network connections must use strong authentication to prevent unauthorized access and messages. Weak authentication in network protocols allows replay or spoof attacks to send unauthorized messages. Poor authentication also allows unauthorized users or computers to connect to a device or application. The lack of authentication in most ICS-specific network protocols allows for manipulation of time synchronization and process alarms, commands, and data updates. Poor authentication in protocol server applications allows unauthorized access to ICS components, including ICS hardware. Proven authentication services should be used when available.

Experienced personnel in authentication and encryption systems involved in creating these systems should be a part of any cyber security staff. Authentication and encryption systems are complex and one small mistake or oversight can render the authentication or encryption ineffective. Test rigorously that the authentication and encryption system are working correctly before deploying the solutions.

Use a well-vetted encryption algorithm that is currently considered to be strong by experts in the field, and select well-tested implementations. Design software so that one cryptographic algorithm can be replaced with another, improving upgrade capability to stronger algorithms. Periodically ensure that current methods used have not been broken. Many old algorithms and implementations have become obsolete or discovered to be flawed.

Securely manage and protect cryptographic keys. Keys should be strong and should not be hard-coded, default, published, or discoverable in any other way.

A remote end-point joins the trusted domain when it is allowed to remotely connect to the ICS network. If virtual private network (VPN) endpoints (hosts) are compromised, an attacker can utilize the VPN connection when it is established. Importantly, these hosts must be secured to the maximum extent possible. End-point management software can be used to help determine the security posture of the remote device and how it is allowed to connect to the protected network, but should not be the only defense measure. VPN access should only be granted to the minimum set of hosts and users when necessary and those VPN connections should be restricted to only allow access to the necessary components.

Internet Protocol security (IPSec) and VPN tunneling cannot be used as a replacement for fixing vulnerabilities. A VPN connection extends the attack surface of the system to the VPN client's computer. An attacker cannot be prevented from compromising a VPN endpoint computer and using the VPN tunnel as an encrypted pathway to exploit the vulnerabilities.

IPsec can be used for confidentiality, integrity, authenticity, and/or replay protection. If an attacker intends to disable IPSec or perform a DoS if he may attempt to gain access to any point between two IPSec partners. The implementation of IPSec included with Microsoft Windows XP, Windows Server 2003 and newer uses the identity proofing afforded by Active Directory. This authentication can be intercepted, causing IPSec to fail. This failure can cause a DoS if the IPSec policy is set to require IPSec for communications. If the IPSec policy is set to request, then an attacker can force IPSec to disable itself it they interfere with the communications long enough to fall back onto unencrypted channels. The decision for configuring this implementation of IPSec with a "request" policy versus a "require" policy should be made based on whether the communication between the IPSec partners must be confidential (or insure integrity, authenticity, or replay protection) or the availability of communication based on criticality.

A number of concerns about the mal-effects of cryptography on ICS have also been raised. The four most common concerns are latency, bandwidth, availability, and IDS interaction. Difficulty of implementation and viewing traffic for trouble-shooting are other issues that can prevent encryption from being used in operational ICS. Still, IPSec and VPN tunneling should not be used as layers of defense.

5.1.9 Improve Security through External Software Security Assessments

ICS software vendors should pay for a third-party security source code audit and fix the problems identified during the audit. Independent source code auditing can help ensure quality and security in software products. An outside professional opinion of software design and implementation based on the actual source code and build process of the ICS product will greatly enhance quality and security, or confirm the security of the product.

ICS software can have large, complicated and legacy codebases. ICS operations require high availability, and update scenarios are complicated. Unlike the standard off-the-shelf computer software model, the cost of security fixes and support and maintenance has traditionally been transferred to the ICS customer. With the new focus and requirements for ICS security, including ICS product vulnerabilities starting to be publicly announced, vendors may find the cost of code audits and associated code changes to be very cost effective versus fixing single vulnerabilities as they are publically announced.

5.2 Recommendations for ICS Owners and Operators

An effective cyber security program for ICS should apply a strategy known as defense-in-depth, layering security mechanisms such that the impact of a failure in any one mechanism is minimized. Implementing security controls, such as intrusion detection software, antivirus software, and file integrity checking software, where technically feasible, will prevent, deter, detect, and mitigate the introduction, exposure, and propagation of malicious software to, within, and from the ICS.

The most successful method for securing an ICS is to gather industry- recommended practices and engage in a proactive, collaborative effort between management, the controls engineer and operator, the IT organization, and a trusted automation advisor. This team should draw upon the wealth of information available from ongoing federal government, industry groups, vendor, and standards organizational activities. ICS owners should perform risk-based assessments on their systems and tailor the recommended guidelines and solutions to meet their specific security, business, and operational requirements.

Planning efforts need to be implemented for prioritization of the tasks necessary to enhance ICS security. Important considerations in this process are cost, probability, and consequence. Decisions concerning methods of mitigating cyber vulnerabilities include balancing the risk of system compromise by an intruder with the risk of potentially degrading system operability. Above all, the ICS must be reliable and perform its required mission. Therefore, the suggested approach is to build security into a system before it is put into production or add security into an existing system in small increments. When adding security to a production system, test on a backup system first to allow quick recovery to the previous configuration in the event any security measure affects system operation. Always weigh the risks and add the appropriate amount of security measures for the specific situation.

Asset owners must use procurement specifications to ensure that security development life-cycle requirements are met by the vendor. Asset owners also may hire independent security assessment teams to review demonstration vendor products for security issues prior to purchase. Vulnerability and patch management programs and policies must be established and enforced.

Good defense in-depth perimeter protections should be used to help prevent access to vulnerable components and communication on ICS networks. Part of a good defense in-depth strategy is identifying and mitigating known vulnerabilities and weaknesses in the system that may help an attacker manipulate or cause damage to the system. Continuous monitoring of IDS logs can allow system administrators to catch and block attempts to circumvent these defenses before serious damage is done.

Firewalls, IDS, and antivirus solutions should be deployed and properly configured at all appropriate locations. Asset owners must identify and deploy security workarounds, defense-in-depth strategies, and use monitoring (access logs and intrusion detection systems) to mitigate risk introduced by the presence of unpatched vulnerabilities until patches can be properly tested and deployed.

Owners/Operators are recommended to increase the security of their systems by completing the following recommendations:

- Redesign network layouts to take full advantage of firewalls, VPNs, etc.
- Implement a network topology for the ICS that has multiple layers, with the most critical communications occurring in the most secure and reliable layer
- Restrict physical access to the ICS network and devices
- Expeditiously deploy security patches after testing all patches under field conditions on a test system if possible, before installation on the ICS
- Work with vendor to test and apply patches for all operating systems and software on the ICS networks
- Customize IDSs for the ICS hosts and networks
- Restrict ICS user privileges to only those that are required to perform each person's job (i.e., establishing role-based access control and configuring each role based on the principle of least privilege)
- Develop a password management plan to enforce strong passwords with minimum length, mixed character sets, expiration, no password reuse, etc., and change all default passwords.

5.2.1 Restrict ICS User Privileges to only those Required

A common problem with applications and services is that they are run with system or root-level privileges. If this case is applicable, and an attacker is able to redirect execution, exploit code will run with those same privileges giving him full access to that device. A number of software products run with these super user permissions by default even though their functions do not require them. Therefore, permission levels of applications and services should be lowered to that necessary for their required functions.

Another common problem is allowing users to operate a computer system (consoles, servers, etc.) with more permissions than necessary. User accounts used for interactive logon should be carefully evaluated for the lowest set of permissions necessary.

File access should then be restricted to those who require access. If network access to a file is necessary, restrict access as much as possible and require strong authentication.

5.2.2 Change All Default Passwords and Require Strong Passwords

In some ICS operations, user IDs, and passwords are shared among the different operators of the system. This sharing must exist, in many cases, because of the criticality of the system operation. Unacceptable consequences might occur because of a locked user ID or a forgotten password. Typical continual manning of operating consoles provides additional physical security that reduces the need for distinct operator user IDs and passwords. If user-level authentication is not an option for operators, ensure all users have separate accounts for all other account types in the ICS to help increase security and accountability. These prudent actions can prevent an attacker from using a user ID and password obtained from the business LAN to gain access to the ICS DMZ and/or the ICS LAN and also prevent authorized users from performing actions that cannot easily be attributed to them.

ICS and networking equipment should not be left with the default manufacturer passwords. Default passwords can give an attacker easy access to the equipment that controls the process. Unless required by the ICS software, default passwords should always be changed to robust, unpublished passwords. In the case that the software uses hardcoded passwords, work with the vendor to fix this vulnerability. Implement a password policy that enforces strong passwords to greatly impede password cracking and guessing.

Passwords have been found in control rooms on small pieces of paper on the bottom of the keyboard, in a drawer, etc. If a password is too complicated and difficult to remember, or changes too often, users will undermine their security in order to remember them. Complex passwords do protect against some of the advanced password cracking attacks, but they create a physical and social engineering vulnerability that could be exploited by an attacker. Therefore, passwords should not be auto-generated, but instead created from passphrases or other memorable means.

5.2.3 Test and Apply Patches

ICS owners must rely on their ICS vendor in some part for validation of patch compatibility before applying them to their operational system. One way to reduce this problem is to reduce the number of applications that need patched.

Services or applications running on a system open up different network ports to be able to communicate to the outside world. Each open port provides a possible access path for an attacker that can be used to send exploits and receive data. An attacker can only gain access to and receive information from the ICS through an open port. The more ports and services that are accessible, the greater the risk of successful exploits due to existing vulnerabilities in the services.

New vulnerabilities are found every day in the applications and services that run on computers. Some of these vulnerabilities are published shortly after their discovery, and some are kept a close secret, allowing a few hackers to exploit computers at will, with no patches available to stop them. Decreasing the number of installed applications and services decreases the likelihood of an attacker finding a vulnerability on the computer. Therefore, all unneeded applications and services should be removed. Also, adequate resources must be allocated to ensure that all services and applications are completely patched and up-to-date using the process described in the preceding patches section.

The patching process should be worked closely with vendor support to ensure ICS application integrity is maintained. Before stopping any services or programs, the vendor should confirm that the service is not needed for system functionality. For conformation, any patch process test should be performed on a backup or development system first, to isolate the primary system from any potential damage. For example, a standard security measure is to shut off the auxiliary services such as echo, chargen, daytime, discard, and finger. However, if the echo port is being used as the system pulse to confirm that the system is up and running, shutting off these services would disable the entire system.

5.2.4 Protect Critical Functions with Network Security Zones and Layers

In many cases, the individuals in charge of the ICS network do not have adequate security training. This situation is generally due to a lack of funding or appreciation for the importance of this training. Training provides an understanding of the security implications of a given network architecture and how to design a more secure network. Educating or hiring network administrators with skills to design and manage the ICS network and its perimeter defenses with the most current security techniques is essential. Network attacks must be prevented, detected, or stopped before they have the opportunity to affect critical ICS functions. ICS security is largely dependent on the effectiveness of the network design to prevent unauthorized access. Network administrators need to understand security concepts such as layering, security, and functionality zones, and specific access rules to restrict all communication to only that which is necessary for system functionality. If the network administrator has designed the network correctly, an attacker is limited to finding vulnerabilities in the authorized users/systems, protocols, or associated applications/servers allowed into each network segment, without being detected.

To provide defense-in-depth, firewalls can be used to separate different layers of the ICS network (i.e., the HMI level LAN from the ICS DMZ from the Enterprise network). These layers can be further segregated into security zones to protect systems from attack through compromised systems on that layer. Multiple DMZs, or security zones, should be created for separate functionalities and access privileges, such as peer connections, the data historian, the OPC server or ICCP server in SCADA systems, the security servers, replicated servers, and development servers.

Any connection into the ICS LAN is considered part of the perimeter. Often these perimeters are not well documented and some connections are neglected. All entry points into the ICS LAN should be known and strictly managed by a security policy. Route all connections to the ICS LAN through the firewall, with no connections circumventing it. Network administrators need to keep an accurate network diagram of their ICS LAN and its connections to other protected subnets, DMZs, the corporate network, and the outside.

Well-configured firewalls are critical to ICS security. Communications should be restricted to that necessary for system functionality. ICS traffic should be monitored, and rules should be developed that allow only necessary access. Any exceptions created in the firewall rule set should be as specific as possible, including host, protocol, and port information. All rules should be concise and well documented. The IDS sensors can then be used to audit the firewall rule set.

A common oversight is not restricting outbound traffic. Firewall rules should consider both directions through the firewall. An exploit that cannot connect back to the attacker is limited to blind attacks. An attacker needs to obtain information from and send files and commands to the ICS network. To remotely control exploit code running on an ICS computer, a return connection must be established from the ICS network. Because of the nature of most vulnerabilities, exploit code must be small and contain just enough code to get an attacker onto the computer; insufficient space is present to add expensive logic for the attacker to get advanced functionality. Therefore, additional instructions are needed from the attacker to continue with the discovery portion of the attack. If outbound filtering is implemented correctly, the attacker will not receive this return connection and cannot discover and control the exploited machine.

The top priority of most ICS installations is availability. The risk to availability of any security feature must be weighed against the expected added security benefit (lowered risk). ICS network administrators may not want to risk the chance of impacting ICS functionality by redesigning the network or updating rules as components are added or removed. In this case, network traffic can be monitored for a long enough period to be confident all possible scenarios have occurred. Rules can then be created starting with the standard restrictions; working toward a rule set that excludes all unnecessary traffic. Once the necessary traffic has been determined, a safer configuration can then be created that blocks all traffic with exceptions for the specific host, protocol, and port combinations that require access in each direction through the firewall.

Greater assurance that network security changes will not affect operations can be obtained by implementing changes as IDS rules. IDS logs can be monitored for alerts identifying traffic that would have been prevented by the new segmentation or access rules. All proposed network changes can be tested as IDS rules for as long as necessary to provide assurance that they will not affect critical functions. Because IDSs do not prevent access, closely monitor IDS logs during this period and immediately investigate unexpected communication.

5.2.5 Customize IDS Rules for the ICS and Closely Monitor Logs

The configuration and deployment of IDS for an ICS is not as straightforward as it is for typical computer networks. IDS signatures are available to detect a wide range of attacks, but the signatures required to monitor for malicious traffic in control networks are not adequate. When looking at the unique communications protocols used in ICS, such as Modbus or DNP3, specific payload and port numbers have traditionally not been a part of the signatures seen in a contemporary IDS. In short, modern IDSs deployed on ICS networks may be blind to the types of attacks that an ICS would experience.

When deploying IDS in an ICS network, the ability to add unique signatures must be used. Removal of some default signatures and response capability is commonplace, as it may have no relevance to ICS network. However, analysis must be made to ensure some of the inherent capability of the IDS is leveraged, with some of the capability refined and augmented. Many security vendors, including those specializing in ICS security, have created signatures for the IDS that are deployed in control architectures. Rules sets and signatures unique to that domain be used are imperative when deploying IDS on ICS networks. Developing security signatures and rules in a cooperative relationship with the ICS vendor are shown through study as very advantageous.

One of the common problems observed in industry is that tools deployed for network monitoring are implemented but improperly updated, monitored, or validated. Assigned individuals should be trained and given the responsibility of monitoring system data logs and keeping the various tool configurations current.

IDS logs can also be used to identify normal communication between each of the ICS components. All unexpected traffic can be investigated and either added to the required communication list or blocked by firewalls.

A one-to-one mapping of firewall rules and IDS signatures should exist so when a firewall rule is not successfully applied; the IDS sensor will alert and allow administrators to take corrective action on the firewall.

The external IDS sensor is used for notification of malicious attempts on the firewall and for monitoring egress rules from the ICS out to the DMZ or corporate networks. The internal IDS sensor and the DMZ IDS sensor are used to closely monitor the exceptions in the firewall for malicious activity.

Intrusion detection is not a single product or technology. A comprehensive set of tools providing network monitoring can give an administrator a complete picture of how the network is being utilized. Implementing a variety of these tools will help create a defense-in-depth architecture that will be more effective in identifying attacker activities.

5.2.6 Force Security through External Software Security Assessments

ICS customers can require a security audit of an ICS product and fixes in order to meet specified security levels as part of the procurement process. This allows the ICS customers to identify security risks of the products and determine whether they are acceptable and/or able to be mitigated. ICS owners can also have external security audits on their existing systems to identify risks that need to be mitigated. Security audits also help fulfill regulatory requirements, but the audit should be used to help secure the ICS as much as possible, not just to fill a requirement.

As ICS industry security requirements have begun to be created, some facilities have learned that they can get away with documenting exceptions to the rules. The requirements developed in an effort to help ICS owners increase their security levels have failed in some cases. ICS owners should look at the development of standards as an opportunity to obtain assistance in securing their assets. Requirements such as yearly security audits can be viewed by those responsible for ICS systems as help in convincing management to spend money on security.

6. SUMMARY

CSSP has conducted 15 ICS assessment and identified two hundred and forty five vulnerabilities that could put the critical systems at risk from a cyber attack. The CSSP assessments are performed to assist in the identification and mitigation of vulnerabilities to support the reduction of risk to critical infrastructure. Assessments were designed to test vendor-specific products and services such as custom protocols, field equipment, applications, and services. Onsite system assessments generally assessed how securely external connections, firewall configurations, IDS, network architecture, and other components are deployed and installed.

The identified common vulnerabilities from the CSSP assessments are being shared to increase security awareness and mitigation. ICS vendors and owners can learn and apply many common computer security concepts and practices to secure and protect their systems. Security should be designed and implemented by qualified security and ICS experts who are able to verify that the solutions are effective and can make sure that the solutions do not impair the system's reliability and timing requirements. Given the nature of the vulnerabilities found in ICS, asset owners cannot always directly fix them. Thus, as asset owners wait for vendor patches and fixes, the design and implementation of defense-in-depth security strategies that aid in protecting the ICS from attack is part of an effective proactive security program.

Attack strategies are constantly evolving to compensate for increasing defense mechanisms. Vendors should offer or support security products and features that can be used as layers of defense to help protect ICS installations. Owners should add the additional network perimeter layers of defense, and actively update and monitor the system. Increasing the hurdles required to attack a system decreases the chance that attackers are able to subvert all hurdles and increases the chance that the attackers will give up before accomplishing their goals. Designing security into the system and using secure coding and security best practices can also minimize damage from attacks by insiders, social engineers, or anyone else with access behind the ICS network perimeter.

ICS product vendors are responsible to deliver systems that are able to survive attack without compromising critical functionality. ICS owners have the responsibility to ensure that the physical systems they operate do not put lives, economy, or environment at risk by failing to perform due diligence in procuring, configuring, securing, and protecting the ICS for critical infrastructure.

7. REFERENCES

1. ANSI/ISA–99.00.01–2007, "Security for Industrial Automation and Control Systems Part 1: Terminology, Concepts, and Models," October 2007, pages 69–73.
2. CWE (Common Weaknesses Enumeration), MITRE, http://cwe.mitre.org/, January 11, 2009.
3. Kevin Poulsen, "Slammer worm crashed Ohio nuke plant network," http://www.securityfocus.com/news/6767, August 2003.
4. Control Systems Security Center, "Control Systems Cyber Security: Defense in Depth Strategies," http://csrp.inl.gov/Recommended_Practices.html#nogo, May 2006.
5. Jaikumar Vijayan, "Gates pushed change in security culture at Microsoft," http://www.computerworld.com/action/article.do?command=viewArticleBasic&articleId=9102998, June 2008.

Appendix A

CSSP Assessment Processes

Appendix A

CSSP Assessment Information

The current overall impact of the CSSP on the stakeholder community encompasses strong user interest in the information from the assessment reports. CSSP assessment team members in conjunction with the vendors are able to inform ICS owners of vendor-approved findings and recommendations at vendor user group conferences and through other vendor-approved channels. ICS owners have requested assessments of their sites, and both utilities and vendors have requested follow-up assessments on the vendors' ICS software. After-action validation of mitigations to identified security flaws are also performed to help ensure the security assessments are successful in increasing critical infrastructure security. General knowledge gained through performing multiple ICS security assessments and correlating assessment report findings is shared through security awareness training, presentations, and reports such as this.

A-1. Laboratory Assessments

The CSSP performs assessments under Cooperative Research and Development Agreements (CRADAs) or Nondisclosure Agreements (NDAs). The CSSP assessment team develops the test plans, performs the cyber security assessments, and reports the results. These agreements also protect the vendor from public disclosure of the assessment findings.

Assessments are usually performed on the latest release of the vendors' systems that is represented within industry which is based on a typical or turnkey installation. This allows CSSP assessment team to influence the system that is currently under development. Identifying a baseline or default architecture is generally difficult because every installation includes some degree of custom configuration. Vendors may support multiple operating systems, features, and ICS protocols. Customers can choose from an assortment of functionality, which can be separated on different servers or combined on one, and many levels of redundancy can be provided. Therefore, CSSP works closely with each vendor to ensure that to the extent possible, the components most commonly found in customer facilities are included in the test architecture.

A-2. Onsite Assessments

Onsite assessments differ from laboratory assessments largely in the amount of time spent on testing and the functional areas of focus for the system assessment. Onsite assessments are completed in 2 or 3 weeks, as compared to 2 or 3 months for more in-depth laboratory assessments. Because an intrusive examination of the operational ICS itself is not appropriate, and system owners and operators cannot modify the ICS software themselves, the focus for onsite assessments is usually network security and perimeter protection. These two areas of focus from the main areas system owners and operators have direct control over for mitigating findings on their assessment and the associated vendor assessment.

The focus of the assessment is changed in order to analyze a fielded system and mechanisms for implementation and protection in production. This changes the assessment to resemble a "network security layers of defense" analysis. This includes a review and tour of the production system to help identify through documentation, observation, and conversation any possible security problems with the production system and network configuration without putting the operational system at risk. Analysis of the actual site's ICS software is done on a backup or test system so that the custom installation can be evaluated based on vulnerabilities found in the associated vendor laboratory assessment. Any mitigating recommendations can then be given to the customer while the vendor fixes the underlying problem.

A-3. REFERENCES

Permann, May, Kenneth Rohde, “Cyber Assessment Methods for SCADA Security,” http://www.isa.org/InTechTemplate.cfm?Section=Article_Index1&template=/ContentManagement/ContentDisplay.cfm&ContentID=49890, November 2005.

Appendix B

Common Vulnerability Identification

Appendix B

Common Vulnerability Identification

B-1. Identification of Recommended Mitigations

Reported assessment findings were extracted, combined, and categorized as described in *Common Cyber Security Vulnerabilities Observed in DHS ICS Assessments* and were given corresponding recommendations and mitigations. These recommendations and mitigations are based on those indicated in the assessment reports. The mitigations are general in nature, with the intent of being applicable to findings identified in multiple assessments. As such, they are high-level generic recommendations and require further refinement before implementation on any specific system. Most of the assessments to date only evaluated the ICS software, rather than hardware. Therefore, a majority of the recommendations require vendor development, not just a configuration change that can be done by the end users. Based on typical maintenance agreements, changes may have to be approved by the maintenance provider prior to implementation. All changes will have to be tested to determine the impact to production and operations. Some may even require extensive rewrites and are not feasible for incorporation into current software releases. In these cases, other defensive measures need to be defined and implemented. Each system needs to be considered on an individual basis following the applicable standards, policies, and procedures respective to all contracts and legal obligations.

B-2. Frequency of Occurrence

Each assessment had different goals and vulnerability identification coverage. Therefore, specific common vulnerabilities are reported, but the number of assessments that tested for that particular vulnerability is not necessarily known.

Not all systems were tested identically, not all vulnerability types were looked for in all systems, and no single standardized testing methodology was used. In addition, not all vulnerabilities found are included in this report. To prevent identifying a specific system or vendor, this report includes only findings determined to be common to multiple systems. Future assessments will test for all common vulnerabilities identified in this report. However, new common vulnerabilities also will be identified in the future that have not been tested for in all assessments.

Appendix C

Terms and Definitions

Appendix C

Terms and Definitions

Access Authorization

Access authorization restricts access to or from a computer, server, Web site, or network to a group of users through the application of authentication systems. These systems can protect either the whole computer, such as through an interactive logon screen, or individual services, such as an FTP server. Many methods are available for identifying and authenticating users, such as passwords, identification cards, smart cards, and biometric systems.

Access Control List

An access control list is a list of permissions attached to a firewall, server, or other device on a network. The list specifies who or what is allowed to access the device and what operations are allowed to be performed on the device.

Antivirus Software

Antivirus software consists of a computer program that attempts to identify, neutralize, or eliminate malicious software (i.e., viruses, Trojan horses, malware, spyware).

ARP

Address resolution protocol (ARP) is the standard method for finding a host's hardware address when only its network layer address is known.

Buffer Overflow

There are two types: stack buffer overflow and heap buffer overflow. Both types of overflow occur when an amount of data larger than the target data buffer area is written to that buffer. The extra data overwrites adjacent memory locations in either the stack (temporary memory) or the heap (dynamic memory) with corrupt data values causing erroneous program results or malicious code to be executed.

Change Management

The change management process is the process of requesting, determining attainability, planning, implementing, and evaluation of changes to a system. It has two main goals: supporting the processing of changes and enabling traceability of changes.

Control System

A device or set of devices to manage, command, direct, or regulate the behavior of other devices or systems.

DMZ

A demilitarized zone (DMZ), more appropriately known as demarcation zone or perimeter network, is a physical or logical subnetwork that interfaces an organization's external services to a larger, untrusted network, usually the Internet. The DMZ adds an additional layer of security to an organization's Local Area Network (LAN).

Encryption

Encryption is the process of transforming information (referred to as plaintext or clear text) using an algorithm (called a cipher) to make it unreadable to anyone except those possessing special knowledge, usually referred to as a key.

Exploit

An exploit (from the same word in the French language, meaning “achievement” or “accomplishment”) is a piece of software, a chunk of data, or sequence of commands that take advantage of a bug, glitch, or vulnerability in order to cause unintended or unanticipated behavior to occur on computer software, hardware, or something electronic (usually computerized). This frequently includes such things as gaining control of a computer system or allowing privilege escalation or a denial-of-service attack.

Finding

An item identified during an assessment. It can be a vulnerability, an observation, a weakness, a flaw, a code error, or a concern.

Firewall

Firewalls can either be hardware devices or software programs. They provide some protection from online intrusion. They are systems that help protect computers and computer networks from attack and subsequent intrusion by restricting the network traffic that can pass through them, based on a set of system administrator defined rules.

Fuzzing or Fuzz Testing

A software testing technique that uses random data, also known as “fuzz,” as input to the software. This technique attempts to exercise code by using values that may be outside the normal range of values for which the software was designed. By doing this the testing, it will uncover areas of the code that were inadequate in handling input values outside the normally desired ranges.

ICCP

The Inter-Control Center Communications Protocol (ICCP or IEC 60870-6/TASE.2) is being specified by utility organizations throughout the world to provide data exchange over wide-area networks (WANs) between utility control centers, utilities, power pools, regional control centers, and Non-Utility Generators. ICCP is also an international standard: International Electrotechnical Commission (IEC) Telecontrol Application Service Element 2 (TASE.2).

Ground Truthing

The technique of verifying that results obtained from lab testing or simulations are repeatable in real-world situations. An example: lab results show a particular configuration creates a vulnerability. Ground truthing of this is accomplished by checking the production system and verifying that indeed a vulnerability exists.

Information Leaks

Inside information that is carelessly disseminated such as passwords written on sticky notes or shared among users. This can also include information items such as user IDs, passwords, and other system information that is not encrypted when transmitted or when stored.

Least Privileges

The technique of assigning privileges for doing certain functions to only those that require them. For example, restricting the ability to create new user accounts to only the system administrator or a user that should only be able to query a database, but has privileges to delete the folder containing the database file.

Man-in-the-Middle Attack

The man-in-the-middle (MitM) attack or bucket-brigade attack is a form of active eavesdropping in which the attacker makes independent connections with computers that communicate with one another and relays messages between them, making them believe that they are talking directly to each other over a private connection when in fact the entire conversation is controlled by the attacker.

OPC

Object Linking and Embedding (OLE) is a technology that allows embedding and linking to documents and other objects developed by Microsoft. OLE for Process Control (OPC) is the standards specification for the communication of UUreal-time plant data between control devices from different manufacturers.

Protocol

A protocol is the set of standard rules for data representation, signaling, authentication, and error detection required to send information over a communications channel.

Reliability

Reliability is the ability of a system to perform and maintain its functions in routine circumstances as well as hostile or unexpected circumstances.

Safety System

A Safety System or Safety Instrumented System (SIS) is a control system consisting of sensors, one or more controllers, and final elements. The purpose of an SIS is to monitor an industrial process for potentially dangerous conditions and to alarm or execute preprogrammed action to either prevent a hazardous event from occurring or mitigate the consequences of such an event should it occur.

Social Engineering Awareness

Keeping employees aware of the dangers of social engineering and having a policy in place to prevent social engineering can reduce successful breaches of the network and servers.

Taxonomy

The science, laws, or principles of classification.