U.S. Department of Energy
Office of Electricity Delivery
and Energy Reliability

INL/EXT-08-13979

Common Cyber Security Vulnerabilities Observed in Control System Assessments by the INL NSTB Program

November 2008

NSTB

National SCADA Test Bed

Enhancing control systems security in the energy sector

INL/EXT-08-13979

Common Cyber Security Vulnerabilities Observed in Control System Assessments by the INL NSTB Program

November 2008

Idaho National Laboratory
Idaho Falls, Idaho 83415
http://www.inl.gov

Prepared for the
U.S. Department of Energy
Office of Electricity Delivery and Energy Reliability
Under DOE Idaho Operations Office
Contract DE-AC07-05ID14517

Common Cyber Security Vulnerabilities Observed in Control System Assessments by the INL NSTB Program

INL/EXT-08-13979

November 2008

Approved by:

David G. Kuipers
Project Manager
National SCADA Test Bed Program

11/17/2008
Date

EXECUTIVE SUMMARY

Idaho National Laboratory (INL) performs cyber security assessments of control systems under private sector and government programs. This report applies to assessments conducted on behalf of the National Supervisory Control and Data Acquisition (SCADA) Test Bed (NSTB) program. The mission is to help industry and government improve the security of the control systems used in critical energy infrastructures throughout the United States. A key part of this mission is the assessment of control systems to identify vulnerabilities that could put critical infrastructures at risk from a cyber attack.

This document presents results from 16 control system assessments performed under the NSTB program from 2003 through 2007. Information found in individual stakeholder reports is protected from disclosure. Researchers recognized that many systems had similar types of findings, and that this information could be of great benefit to those in industry responsible for control system security. For this reason, vulnerability information was collected, analyzed, and organized into general categories based on a control system security metrics approach.

Common vulnerabilities were derived from similar assessment findings and grouped into general security dimensions and sub-categories based on analysis using an INL-developed taxonomy. Mitigation strategies for these findings are provided in each section.

Information found in this report can 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 control system applications, services, and protocols, and how to better secure their products while asset owners can evaluate possible weaknesses in their installed system configurations and 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 how to mitigate them can serve to help protect the systems currently in development as well as those already installed in critical infrastructure applications.

The current overall impact of the INL NSTB program on the stakeholder community encompasses strong user interest in the information from the assessment reports. Utilities have requested assessments of their sites, and both utilities and vendors have requested follow-up assessments on the vendors' control system software. Knowledge gained through control system security assessments has been shared with over 1,800 individuals through security awareness training, and specific vendor-approved findings have been presented at vendor user group conferences.

This document represents a steadily growing understanding of control system security issues and methods for mitigating current vulnerabilities as well as new technologies and approaches being developed in response to security challenges. The effort is expanding to new technologies, such as substation automation and Smart Grid, as the program seeks a continuing understanding of the systems being planned for and deployed in the energy sector critical infrastructure.

CONTENTS

FIGURES

TABLES

ACRONYMS

ACL	Access Control List
AG	Attack Group
ARP	Address Resolution Protocol
BEA	Battelle Energy Alliance
CIA	Confidentiality, Integrity, Availability
CMU	Carnegie Mellon University
CRADA	Cooperative Research and Development Agreement
CWE	Common Weaknesses Enumeration
DMZ	Demilitarized Zone
DNP	Distributed Network Protocol
DOE	Department of Energy
DOE-OE	Department of Energy-Office of Electricity Delivery and Energy Reliability
DoS	Denial-of-Service
EMS	Energy Management System
FEP	Front-end Processor
FTP	File Transfer Protocol
HMI	Human Machine Interface
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol over Secure Socket Layer
IAM	Information Security Assessment Methodology
ICCP	Inter-Control Center Communications Protocol
IDS	Intrusion Detection Systems
IEC	International Electrotechnical Commission
IED	Intelligent Electronic Device
INFOSEC	Information Security
INL	Idaho National Laboratory
IP	Internet Protocol
IT	Information Technology
LAN	Local Area Network
MAC	Media Access Control
MITM	Man-in-the-Middle
MS	Microsoft
NDA	Nondisclosure Agreement

NFS	Network File System
NIST	National Institutes of Technology
NSA	National Security Agency
NSTB	National Supervisory Control and Data Acquisition Test Bed
PLC	Programmable Logic Controllers
RTU	Remote Terminal Unit
SCADA	Supervisory Control and Data Acquisition
SG	Security Group
SIS	Safety Instrumental System
SNL	Sandia National Laboratory
SQL	Structured Query Language
SSH	Secure Shell
SSL	Secure Sockets Layer
TASE	Telecontrol Application Service Element
USB	Universal Serial Bus
VPN	Virtual Private Network
VTP	Virtual Trunking Protocol
WAN	Wide Area Network
XML	eXtensible Markup Language

Common Cyber Security Vulnerabilities Observed in Control System Assessments by the INL NSTB Program

1. INTRODUCTION

The U.S. Department of Energy (DOE) established the National Supervisory Control and Data Acquisition (SCADA) Test Bed (NSTB) program to assist industry and government improve the security of control systems used in the nation's critical energy infrastructures. The NSTB Program is funded and directed by the DOE Office of Electricity Delivery and Energy Reliability (DOE-OE). A key part of the program is the assessment of control systems to identify and provide mitigation approaches for vulnerabilities that could put the systems at risk to a cyber attack.

Assessments are performed in the Idaho National Laboratory (INL) SCADA Test Bed and in operational installations at utilities, generation plants, and energy management facilities. Control system vendor software is tested in the SCADA Test Bed at INL. Operational control system assessments use nonintrusive methods, such as reviewing the production system network diagrams and firewall rules, and performing a hands-on assessment of a duplicate non-production installation of the system.

SCADA systems are of the greatest interest in the assessments because they are the most common type of control system in the energy sector, controlling equipment ranging from valves in oil and gas pipelines to switches and breakers in the national electric grid. If compromised, an attacker could utilize these systems to cause catastrophic damage or outages directly or by exploiting paths to critical end devices or connected SCADA systems.

2. ASSESSMENT METHODOLOGIES

The primary goal of the INL cyber assessment tasks is to improve the security of the energy 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). With this in mind, INL has performed assessments on a large variety of systems. Some systems have been assessed after being set up in the INL test bed while others have been assessed onsite in production facilities. Regardless of where the work took place, an assessment plan and methodology are tailored to provide the most value to the customer owning the system. To accomplish this, assessment methodologies differed significantly between laboratory and onsite assessments, and even across systems.

System configurations varied considerably depending on control system functionality, negotiated objectives, and whether the assessment was conducted in the 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 because a comprehensive system assessment is not possible given project funding and the complex nature of control systems. 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 a control system. After a system has been tested in the laboratory, an assessment is usually performed on an asset-owner installed version of the system. This provides an opportunity to work with owners and operators and help validate the impact and possible mitigation strategies for vulnerabilities identified in the laboratory assessment.

Assessment projects typically leverage a full-disclosure approach with the vendor and asset-owner partners. The INL focus is on the control system 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. INL 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 control systems.

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. Typical tools used in testing can be found in the Permann and Rohde report.[1] 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 control system local area network (LAN), with the assumption that the attacker has penetrated perimeter protection and is on the control system network. Typical assessment targets may be "Changing Alarms and Commands" or "Unauthorized Database Access." If the test environment contains connections external to the control system 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 control system 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 time frame 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 control system vendor. This has been 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 control system and cause damage to equipment, interrupt service, or harm people or the environment.

2.1 Laboratory Assessments

Laboratory assessments are performed under Cooperative Research and Development Agreements (CRADAs) or Nondisclosure Agreements (NDAs) between the system vendors and Battelle Energy Alliance (BEA). BEA, which currently manages the INL contract, is hereafter referred to as INL. Under these agreements, the vendors provide hardware, software, and technical support. INL 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 using a base, typical or turnkey installation. This allows INL to influence the system that is currently under development. Identifying a baseline or default architecture is generally difficult since every installation includes some degree of custom configuration. Vendors may support multiple operating systems, features, and control system 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, INL 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.

2.2 Onsite Assessments

Onsite assessments are performed under a CRADA or NDA agreement between INL and the industry partner. Onsite assessments differ from laboratory assessments largely in the amount of time spent on testing and the functional areas of a system the testing focuses on. Onsite assessments are completed in 2 or 3 weeks, as compared to 2 or 3 months for more in-depth laboratory assessments. Since an intrusive examination of the operational control system itself is not appropriate and system owners and operators cannot modify the control system software themselves, the focus for onsite assessments is usually network security and perimeter protection. These are the main areas they have direct control of 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 how it is implemented and protected 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 control system 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.

2.3 Impact on Stakeholder Community

Since the inception of the program, the reports generated from the assessment of control system 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. 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 results were shared with control system customers or not, security awareness has been increased by sharing control system security knowledge gained through the assessment process. Cyber security and control system researchers have presented results and provided security training to attendees of the various vendor user group conferences. Most user groups have established breakout sessions and working groups dedicated wholly to control system cyber security issues. Some of these sessions have evolved into user consortiums and other arrangements with vendors to fund additional control system assessments.

The NSTB onsite assessments have been conducted at utilities in order to help secure the particular site, verify laboratory findings, and gather common recommendations that are then shared with other utilities. This brings laboratory vendor assessments full circle to help the control system software, its implementations in the field, and the associated critical infrastructure become more secure.

3. COMMON ASSESSMENT FINDINGS

The INL Control Systems and Cyber Security Test Center is a research facility designed to evaluate control systems representative of those used in the United States critical energy infrastructure. It was established through a collaboration of the DOE-OE NSTB and the Department of Homeland Security Control System Security Program to assess and help improve the security of control systems, which have not had the exposure, scrutiny, and security culture change that common IT applications have received. Understanding the unique priorities, vulnerabilities, impacts, limitations, and operation inherent in control systems has been a research effort with evolving assessment procedures. As a result, assessments are customized and tailored to the specific control system and the aspects of the system that its vendor and INL determine require the most scrutiny. Assessments are a combination of targets based on perceived vulnerability, impact or exposure of system components and functionality, and evaluation of security measures built into, or placed around, the control system.

Even though there is a common general theme and assessment methodology, all systems have not been tested for the same vulnerabilities. For example, 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. Another factor that contributed to the disparity of vulnerability tests was that all systems configured for assessment do not have the same set of components and functionality. Control systems inherently have many configuration options, such as:

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

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

- A control system with the most basic functionality on a single LAN with a few computers
- A control system with partial optional functionality, simulated data, and some network security defense devices
- All available control system functionality connected to a remote terminal unit (RTU) with the recommended network architecture and perimeter defenses
- An operational system connected to the U.S. power grid 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.

Common NSTB assessment findings in this report were not derived from a comprehensive or even consistent set of assessments or fully operational control systems. The common assessment findings that follow are similar security problems found on two or more unique control system configurations. It cannot be inferred that vulnerabilities not included in this report are not common to control systems or that three similar findings under a common vulnerability means that only three systems assessed were susceptible to that vulnerability.

Table 1 lists the common vulnerabilities categorized based on the taxonomy described in Appendix A. The security dimensions represent the main system attributes that affect the risk of loss from cyber attacks. These security dimensions are further subdivided into categories of problems that lead to vulnerabilities. A common vulnerability describes findings from a minimum of two assessments.

The assessment findings described in this report are organized according to the security dimension and category in which they belong. Vulnerability descriptions are generalized to remove specific vendor-identifying information and details that would hinder the ability to group common vulnerabilities. Sanitized assessment details are included inside 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 with a given vulnerability are not listed to avoid any implication that all systems were tested for that vulnerability and to help lend anonymity to the control systems associated with common vulnerabilities and related specific details.

Table 1. Summary of common NSTB control system assessment findings to date.

Security Dimension	Category	Common Vulnerability
Security Group (SG) Knowledge	Change Management Deficiencies	Network access rules not removed when no longer needed - 3.1.1.1
	Documentation Deficiencies	Inaccurate critical asset documentation - 3.1.2.1
Attack Group (AG) Knowledge	Information Leaks	Unencrypted services common in IT systems – 3.2.1.1
		Weak protection of user credentials – 3.2.1.2
		Unencrypted proprietary control system protocol communication – 3.2.1.3
		Unencrypted non-proprietary control system protocol communication – 3.2.1.4
		Unencrypted Inter-Control Center Communications Protocol (ICCP) communication – 3.2.1.5
Access	Firewall Filtering Deficiencies	Access to specific ports on host not restricted to required Internet Protocol (IP) addresses – 3.3.1.1
		Access not restricted to required one-way communication initiation – 3.3.1.2
	Remote Access Deficiencies	Unauthorized access through Virtual Private Network (VPN) – 3.3.2.1
		Unauthorized access through remote display applications – 3.3.2.2
	Physical Access	Physical access to network equipment – 3.3.3.1
Vulnerability	Lack of Input Validation	Buffer overflow in control system service – 3.4.1.1
		Integer overflow in control system service – 3.4.1.2
		Lack of bounds checking in control system service – 3.4.1.3
		Structured Query Language (SQL) injection vulnerability – 3.4.1.4

Security Dimension	Category	Common Vulnerability
		Control system protocol uses weak integrity checks – 3.4.2.4
	Communication Protocols with Weak or No Authentication	Standard IT protocol encryption can be defeated – 3.4.2.1
		Standard IT protocol uses clear text authentication
		Control system protocol uses weak authentication – 3.4.2.2
	Weak User Authentication	Improper security configuration – 3.4.3.1
		No password required – 3.4.3.2
		Weak passwords – 3.4.3.3
	Least Privileges not Enforced	Unauthorized directory traversal allowed – 3.4.4.1
		Services running with unnecessary privileges – 3.4.4.2
		Control system protocols with unnecessary functionality – 3.4.4.3
	Unpatched Systems	Unpatched operating system – 3.4.5.1
		Unpatched third-party application – 3.4.5.2

3.1 Security Group Knowledge Security Dimension

The security group knowledge ideal is defined as aspects of the system or associated management processes that impact the ability of the security group to know the system and manage changes to the system. Assessment findings show that inadequate documentation and communications prevented security groups from having accurate knowledge of network and asset configurations.

3.1.1 Vulnerability Category: Change Management Deficiencies

The security group knowledge of the system may not be accurate if changes to the system are not properly managed. If there are no tools or processes to support the tracking of changes, the security group knowledge of the control system will be outdated. Unmanaged changes lead to more access allowed than necessary because legacy firewall exceptions, undocumented connections, and unused user accounts exist.

3.1.1.1 Common Vulnerability: Network access rules not removed when no longer needed

Rules that allowed unnecessary network access were found during assessments (see Section 4.3). This indicated that network equipment was not being updated when the network configuration or communication paths changed. This showed a lack of change management procedures for removing firewall rules when hosts were removed from the network. Legacy firewall configurations indicate inaccurate knowledge of the current network and may indicate that network administrators are not aware of some potential attack paths. Unneeded firewall access rules may expose vulnerabilities on hosts for potential exploitation. The unneeded firewall rules also may increase the possibility of introducing errors while making modifications to firewall configurations.

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

- Hosts do not appear to exist; however, there are rules pertaining to these hosts

- All hosts on the control system LAN have unrestricted access to the IP address of a since-removed external server
- Routers have what appear to be legacy configuration entries.

Recommendation: Change management procedures need to be created and enforced to ensure unneeded access rules are removed as they become unnecessary.

3.1.1.2 Mitigations for change management deficiencies category

Change management is necessary for accurate system state knowledge and accountability. It is also necessary for compliance to industry regulations. Create and follow change management processes to update router, switch, and firewall configurations, user accounts, etc. This change management procedure should also include compliance and security group documentation updates.

3.1.2 Vulnerability Category: Documentation Deficiencies

There are different types of technical documentation and all play an important role for the smooth operation of systems and organizations. It is important to maintain current information on hardware and software component configurations used in the control system's architecture. Other materials are important for maintaining systems, such as user guides for installation, reference, and repair. Business policy and procedures, emergency response procedures, security policies, as well as training manuals improve the communication and management process.

3.1.2.1 Common Vulnerability: Inaccurate documentation

In many onsite assessments, documentation was found to either be inaccurate or outdated. Inaccurate network and asset documentation may lead to inadequate access control implementations and other security protections by the security group.

Four specific assessment findings associated with this vulnerability are:

- Critical asset documentation was incorrect or outdated
- There was a disconnect between the corporate IT networking and control system networking groups, which fostered misconfiguration of critical networking resources
- Network did not match drawings or documentation
- IT and control system departments used different names for the same servers.

Recommendation: Timely and accurate communications between the IT and control system departments needs to be promoted. Integrated plant-specific security plans, policies, and procedures need to be established with the support of senior management, IT and control system groups to be applied and standardized between facilities or subsystems within a facility.

Specifically, configuration management procedures that apply to keeping documentation current when assets are acquired or removed must be used and enforced. Periodic reviews and audits should be scheduled to ensure configuration documentation is accurate.

3.1.2.2 Mitigations for documentation deficiencies category

Once the asset documentation has been completed, documents must be kept current through the same change management procedures that keep the network, host, and account configurations current. Whenever assets are added or removed, or the network is reconfigured, the change management

procedures should be followed to update any affected network and critical asset regulatory compliance and security group documentation.

3.1.3 Vulnerability Category: Procedure Deficiencies

No common vulnerabilities were found in the procedure deficiencies category. This does not mean that there were not common procedure deficiencies in the systems assessed. It should only be inferred that due to the nature of the custom assessment plans and systems, common security group procedure problems were not tested for and discovered in at least two systems.

3.1.4 Security Group Knowledge Summary

Two categories with common vulnerabilities were identified in the security group knowledge dimension. Change management deficiencies led to legacy network access rules not being removed from firewalls and routers, which allowed access paths to hosts and ports that were no longer needed. Documentation deficiencies resulted in inaccurate critical asset documentation (see Table 2).

Table 2. Frequency of security group knowledge common vulnerabilities by category.

Vulnerability Category	Number of Unique Detailed Finding Descriptions
Documentation Deficiency	4
Change Management Deficiency	3
Procedure Deficiency	0

3.1.4.1 Summary of Recommendations for Security Group Knowledge Dimension

System tools and associated management processes should provide the security group with current and accurate knowledge of the system and manage changes to the system. This includes configuration and change management tools and procedures that support the tracking of changes, collection and analysis of system logs, and forensics. Common findings indicate the need for a greater focus on documentation and information sharing of accurate network and asset configurations.

Once the asset documentation has been completed, documents must be kept current using the same configuration control procedures used for network, host, and account configurations. Whenever assets are added or removed, or the network is reconfigured, the change management procedures must be followed to update any affected network and critical asset regulatory compliance and security group documentation.

Change management processes must be enforced to ensure system knowledge and documentation is current and accurate.

3.2 Attack Group Knowledge Security Dimension

The Attack Group Knowledge dimension of security is the set of system attributes, processes, or actions that provide potential attackers with the means to gain information about the system.

An attacker must find information about the target control system and discover details about the process under its control before he can create an attack against it. If the attacker's goal is merely to shut down the process, very little discovery is needed. However, if the attacker intends a surgical attack or process manipulation, specific details are needed. Some primary sources of information about the process

on the control system LAN are the traffic, points database, and the operator's screens. The points database assigns numbers to each device, such as a switch or breaker. An attacker planning a surgical strike on the process needs the point information because, at the protocol level, each device is often referred to by number only. The operator's screens are generally the easiest way to understand the process.

Assessment findings show that use of clear text communications and access to system source code and configuration data are potential weaknesses that could aid an attacker in compromising the integrity of the control system. The open source availability of information regarding the software, devices, topology, configuration, and purpose of a control system was not significantly addressed in any of the assessments to date.

3.2.1 Vulnerability Category: Information Leaks

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

3.2.1.1 Common Vulnerability: Unencrypted Services Common in IT Systems

Unsecure services developed for IT systems have been adopted for use in control systems for common IT functionality. Although more secure alternatives exist for most of these services, active unused or obsolete services still exist in many control systems. Unfortunately, this has led to vulnerabilities readily accessible to an attacker who has gained a toe hold on the control system or has access to an unencrypted communication channel to or from the system. For example, management of network devices using clear text communication provides an attacker the opportunity to learn about the system's configuration and to more effectively plan his continued attack. Further, the storage of control system artifacts, such as source code and system configuration on an unprotected file system, provides significant potential for information mining by an attacker.

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

- Use of clear text IT protocols on control system LAN (e.g., telnet, ftp, "r" services). This finding was common to multiple assessments.
- Network File System (NFS), which has relatively limited security features, is used as the network file system.
- The clear text IT protocol, telnet, is used to access and manage network equipment.
- The clear text IT protocols, File Transfer Protocol (FTP) and Hypertext Transfer Protocol (HTTP), are used for configuration and file transfers to RTUs.
- Access to clear text vendor proprietary services for management of network equipment is not limited solely to the necessary devices.
- Management of network equipment from the corporate network into the control network is accomplished using a clear text service.

Recommendation: Encryption is the obvious answer to information leaks due to clear text communication. Unfortunately, encryption is not always feasible on control system 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. Control systems use common IT protocols for common IT functionality, such as network device

management, remote logins, or file transfers. Since they are not used for real-time functionality, they can be replaced with their secure counterparts in most cases. Secure shell (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. Hypertext Transfer Protocol (HTTP) can be sent over the Secure Socket Layer (HTTPS). There are more secure network file sharing solutions available also. Control system 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 control system whenever possible. Communications between security zones should be secured as much as possible.

3.2.1.2 Common Vulnerability: 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 control system and thus being able to more effectively advance his attack.

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

- Services such as FTP, telnet, and rlogin transmit user credentials in clear text
- Password hash files are not properly secured.

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.

3.2.1.3 Common Vulnerability: Unencrypted proprietary control system 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 to modify or insert commands in ways which suit the attacker's purpose.

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

- A proprietary protocol and service required the firewall to allow incoming connections to the control system on a large number of ports using unencrypted communication
- A common finding was that the backbone control system proprietary communication protocol was clear text and susceptible to being reverse engineered.

Recommendation: When possible, standard secure versions of services should be used. When proprietary protocols are used, they should, ideally, be encrypted and every message's integrity validated. If it is not feasible to encrypt messages or provide encrypted channels, 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 control system security zone.

3.2.1.4 Common Vulnerability: Unencrypted Non-proprietary Control System 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 which suit the attacker's purpose.

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

- Communication between the front end processor (FEP) and RTUs was unencrypted and susceptible to reverse engineering to effectively alter values to and from field devices.

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, it would be useful to 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.

3.2.1.5 Common Vulnerability: Unencrypted ICCP communication

The Inter-Control Center Communications Protocol (ICCP) is an open protocol used internationally in the electric power industry to exchange data among utilities, regional transmission operators, independent power producers, and others. The clear text data exchange may provide many opportunities for an attacker to intercept, understand, and alter the data being exchanged.

A specific assessment finding associated with this vulnerability is clear text ICCP traffic.

Recommendation: The preferred mitigation is to adopt the secure version of ICCP so that all communication is appropriately encrypted. This solution is dependent on whether data-sharing peers support secure ICCP. Whether ICCP is encrypted or not, the connection is still with an outside entity; treat ICCP as an untrustworthy connection, place the ICCP server in a separate DMZ, and monitor it closely.

3.2.1.6 Mitigations for Information Leaks Category

Availability of system information including topology, configuration, user credentials, protocol specification, and specific device function are of significant value to an attacker. As a consequence, a variety of mitigations would reduce the potential for information leakage.

Wherever encryption of messages is feasible, it should be done. This may include the replacement of standard insecure services with their secure counterparts or may, in some cases, involve migration away from unencrypted proprietary protocols to more standard and secure services.

Control system communications should be isolated to the extent possible with a network partitioning and defense-in-depth strategies that use switches, firewalls, access control lists, and DMZs, thus minimizing an attacker's access to different portions of the control system's messaging.

Also, all unnecessary system documentation should be removed from the system. Required system documentation should only reside on the necessary system devices, which should be minimized, and access to that information should be restricted to the greatest extent possible.

3.2.2 Vulnerability Category: Open Source Information Available

Availability of open source information concerning the software, devices, topology, configuration, and purpose of a control system may be the first criteria in an attacker's identification and selection of a target. To a degree, the accessibility of the information is not strictly in the control of individual owners or vendors. However, it is possible to establish a process to baseline the information that has been made

publicly available on the product, and to then track and respond to the increase or decrease of the available information over time.

3.2.2.1 Common Vulnerability: Default Passwords

In the assessments used to generate the set of common vulnerabilities, there was no systematic investigation of open source information related to the control system, and little analysis of the potential uses of such information to an attacker. The investigations that were done focused on default passwords and their use in the operational system. If default passwords are retained, it would be a significant windfall for an attacker. However, this potential vulnerability was not found on multiple assessments, and therefore does not yet qualify as a common vulnerability.

Recommendation: Not applicable.

3.2.2.2 Mitigations for open source information available category

The first step in mitigation of the potential vulnerabilities associated with publicly available, open-source information is to become aware of the information that is published and available about the control system.

The second step is to develop mitigations for the publicly available information. In the case of information about a code error or security flaw being published, a good patch management program might be used to ensure that timely solutions to counter the threat are implemented. Information, such as default passwords, can be assumed to be available information and should be changed. A full discussion of possible mitigations is pending a more thorough investigation of the types and qualities of open source information that is available to potential attackers and commonality of the resulting vulnerabilities.

3.2.3 Attack Group Knowledge Summary

Only one category with common vulnerabilities was identified in the attack group knowledge dimension. Information leaks were caused by unencrypted control system protocols and common IT services, and weak protection of user credentials.

3.3 Access Security Dimension

Access is the dimension of security that provides a potential attacker with the ability to send or receive data to or from a component of the control system from the attacker's location. The first step in gaining control of a control system is to gain unauthorized access to the control system LAN. Fortunately, most control system networks are no longer directly accessible from the Internet. It is now common industry practice to separate the business LAN from the control system LAN with a firewall. The firewall helps prevent unauthorized access and isolates the control system LAN from worms and other maladies that may infect the corporate network. Often a separate LAN, the DMZ, is created to share data between the corporate and control system LANs and to keep non-control system applications off the control system LAN.

To access the control system LAN, the attacker must first bypass the perimeter defense provided by the firewall or find another avenue onto the control system LAN. The attacker can use a number of proven techniques, such as piggybacking, on a connection or exploiting a service allowed through the firewall, discovering an auto-answer modem or connection circumventing the firewall, or gaining access

through a trusted peer site. The attacker does need to maintain access to the control system network in order to accomplish the rest of the attack.

Vulnerabilities that were reported fell into the categories of physical access attack vectors, network partitioning deficiencies, firewall filtering deficiencies, and services allowed into the control system. Most access violation findings fell into the firewall filtering deficiencies category. Inadequate incoming access restrictions findings outweighed all other access control vulnerabilities reported.

3.3.1 Vulnerability Category: Firewall Filtering Deficiencies

Firewall and router filtering deficiencies include access to control system 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.

3.3.1.1 Common Vulnerability: 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. This was because legacy configuration of the firewall allowed access to unused IP addresses. This provides 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 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 four specific assessment findings associated with this vulnerability:

- Network device access control lists did not restrict management access to the required IP addresses
- Unrestricted telnet access allowed to DMZ network equipment
- Legacy configuration of firewall allows access to unused IP addresses
- Access to resources specified in the firewall rules apply to entire corporate network.

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.

3.3.1.2 Common Vulnerability: Access not Restricted to Required One-way Communication Initiation

Detailed findings under this common vulnerability involve firewall rules that do not restrict access based on direction.

Two specific assessment findings associated with this vulnerability are:

- Bi-directional rules allow connectivity to and from the control system DMZ for various control system protocols
- Network equipment allowed to initiate connections to management servers.

Recommendation: Ensure that the communications paths require bi-directional communication. If not required, modify each rule to only allow one-way communication initiation. For example, if devices such as networking hardware have no need to initiate sessions out of the DMZ, remove the configuration rules that allow this.

3.3.1.3 Mitigations for Firewall Filtering Deficiencies Category

A well-configured firewall is critical to control system LAN security. Communications should be restricted to only what is necessary for system functionality. System 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.

3.3.2 Vulnerability Category: Remote Access Deficiencies

Unauthorized access may be gained through vulnerable or misconfigured services allowed into the control system. Remote access paths can possibly provide access to control system components through external and internal networks.

3.3.2.1 Common Vulnerability: Unauthorized Access Through VPN

Virtual private network (VPN) access is commonly used for remote access into control systems for vendor maintenance access and possibly system administrator access.

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

- Selective VPN access routes into the control system network were misconfigured
- VPN concentrator was configured incorrectly.

Recommendations: Make sure VPN access is configured correctly and securely. When using a VPN, the VPN concentrator should enforce the security policy instead of relying on the client to enforce security upon itself.

The remote end-point joins the trusted domain when it is allowed to remotely connect to the control system network. If VPN endpoints (hosts) are compromised, an attacker can utilize the VPN connection when it is established. Therefore, it is important that these hosts be secured as much as possible. End-point management software can be used to determine the security posture of the remote device and how it is allowed to connect to the protected network. VPN access should only be granted to the minimum set of hosts and users necessary.

3.3.2.2 Common Vulnerability: Unauthorized Access Through Remote Display Applications

Remote display applications (such as Remote desktop, Citrix, VNC, PC Call Anywhere, Hummingbird, etc.) are commonly used for remote access into the control system from other networks. This provides possible attack paths if these applications can be exploited.

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

- Remote desktop passwords were common between security zones (corporate and control system networks)
- It was possible to escalate privileges from a non-control system application remotely published by the display application to a control system application.

Recommendations: Make sure remote display access is configured correctly and securely, and is kept patched. Ensure access is limited to authorized personnel. Credentials for different security zones should be different to prevent unauthorized access due to knowledge of authentication information in another zone. Remote display applications should not be relied on to "sandbox" a user into the display application's environment to prevent the user from accessing the underlying system.

3.3.2.3 Mitigations for Remote Access Deficiencies

In order to gain remote access, an attacker must find a path through the perimeter defenses. This can be done by piggybacking on a connection or exploiting a service allowed through the firewall. Therefore, all connections allowed in and out of the control system network(s) need to be configured securely and monitored. Secure remote access services should be used and kept patched and current from vulnerabilities, which could allow an attacker to gather connection credentials or utilize the connection for their own purposes.

Remote access extends the attack surface to include the remote endpoints. If an attacker is able to compromise a remote endpoint they can use the privileges of that endpoint to facilitate further attacks into the critical network. Therefore the security of the endpoints is critical and needs to be maintained.

3.3.3 Vulnerability Category: Physical Access

Physical access vulnerability allows unauthorized physical access to critical assets. This includes physical access to control system field devices that can be physically controlled or damaged, and physical connection points into the data transfer paths of the control system such as Universal Serial Bus (USB), switch, and other ports. Examples are devices located outside the protected area and any other path that allows unrestricted physical access to critical assets.

A well-accepted rule of computer security is that once an attacker has acquired physical access to a machine, it is generally trivial for that attacker to fully compromise the system. As technology improves, this is becoming less of an issue, but for now, if an attacker has physical access to a machine, the attacker can generally breach its security.

3.3.3.1 Common Vulnerability: Physical Access to Network Equipment

Unauthorized network access through physical access to network equipment includes the lack of physical access control to the equipment. It also includes the lack of security configurations 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.

The specific assessment finding was no port security on the network switch.

Recommendation: Port security should be implemented to limit connectivity to hardware interfaces. Given the static nature of control system environments, port security can be used to ensure Media Access Control (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.

3.3.3.2 Mitigations for Physical Access Category

Follow CIP-006-1, "Physical Security of Critical Cyber Assets." Protect all critical cyber assets by isolating them in secured areas inside the physical security perimeter.

Physical access to an asset generally implies compromise. There are a number of simple steps a user can take to mitigate the risk of attacks via physical access. The first and most important of these is to not allow physical access to a system by untrustworthy people. Additional security measures should also be taken, which restrict access and damage potential once physical access is obtained (i.e., network port security and strong authentication).

3.3.4 Access Security Dimension Summary

Three categories with common vulnerabilities were identified in the access security dimension: firewall filtering deficiencies, remote access, and physical access. Firewall filtering deficiencies allowed access to ports on more IP addresses than necessary and did not restrict the direction the connections could be initiated. Unauthorized remote access could be gained by taking advantage of VPN and remote display communication paths. The lack of physical access protections allowed access to the control system network (see Figure 1).

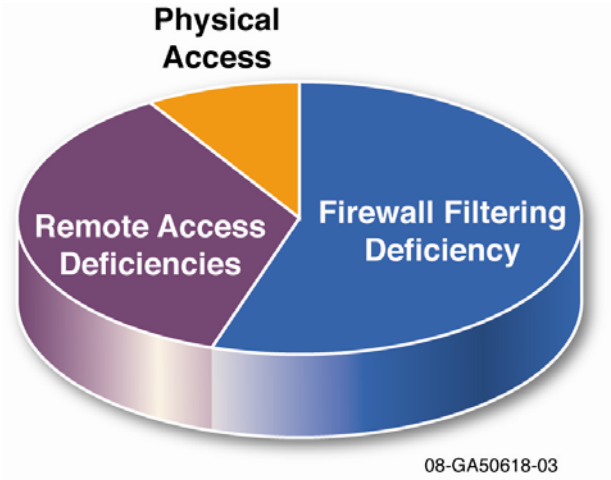

Figure 1. Frequency of access dimension common vulnerabilities by category.

3.3.4.1 Summary of Recommendations for Access Security Dimension

Disable or remove unused services. Documentation, traffic analysis, shutting off individual services, and other such methods can be used to identify the services necessary on each control system component. This should be done by the control system vendor. After used services are identified, they should be analyzed for necessity, especially if they are associated with known vulnerabilities. All necessary communication should be done using secure services that are kept patched and up-to-date to help prevent attackers from utilizing the allowed connection for their own purposes.

Use a firewall to prevent access to ports and services that are unneeded or cannot be disabled or removed. Firewall rules also need to restrict access to only the required IP addresses and the required communication directions. Change management procedures and periodic network reviews.

Do not allow physical access to the critical assets. Additional security measures should also be taken to restrict access and damage potential once physical access is obtained.

3.4 Vulnerability Security Dimension

The vulnerability security dimension is that dimension that deals with weaknesses in the control system that allows an attacker to advance towards or accomplish malicious objectives. Vulnerabilities may be software flaws or configuration mistakes.

After an intruder has discovered enough information regarding the control system, he may be able to attack or manipulate it. In general, the easiest way for an intruder to control control system components is to send commands directly to the front-end equipment. Most front-end equipment, such as RTU, programmable logic controllers (PLCs), protocol converters, or FEPs lack even basic authentication. To control such equipment, in most cases, an attacker need only establish a connection and issue a properly formatted command. The operator's screen could possibly be exported back to the attacker as well, giving him operator-level awareness and control of the process.

An attacker could also perform man-in-the-middle (MITM) attacks on the control system protocols. Once the attacker knows the protocol, he can modify the packets in transit. By inserting packets into the network, he can issue arbitrary commands. By modifying replies, he can give the operator a false picture of the control system state. Thus, the attacker could both spoof and control the system.

Because control systems are not as prevalent as IT systems and have not been targeted by vulnerability discovery groups, and because of the nature of testing at the NSTB, many of the vulnerabilities documented in the assessment reports were discovered by the assessment team and are therefore undisclosed, or zero day vulnerabilities.

3.4.1 Vulnerability Category: Lack of Input Validation

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. It is the result of programmer oversight.

3.4.1.1 Common vulnerability: Buffer Overflow in Control System Service

Buffer overflows result when a program tries to write more data into a buffer than the space it was 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 arbitrary code provided by the attacker resulting in the system being remotely controlled.

Network protocol implementations that do not check input strings for length put the system at risk for buffer overflow exploits. The C and C++ programming languages are especially vulnerable, containing string and memory function calls that can be used insecurely Other languages, such as Fortran, can also be vulnerable.

Services written by control system vendors frequently suffer from coding practices that allow attackers to supply unexpected data and thus modify program execution. Even though some control system protocols are commonly used, the services that receive and interpret are usually customized to the vendor product. Vulnerabilities in these services were a main target of many laboratory assessments because buffer overflows in the control system services are possible entry points onto the control system LAN (if the traffic is allowed).

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

- Multiple assessments found buffer overflows in control system protocol implementations
- Multiple assessments found buffer overflows in control system applications

- Multiple assessments found buffer overflows in proprietary Distributed Network Protocol (DNP) implementations
- Buffer overflows were found in FEP programs
- A buffer overflow was found in a proprietary real-time database
- Buffer overflows were found in ICCP implementations.

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. This 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, it cannot be assumed that data will always be correctly formatted or that hardware or operating system protections are sufficient. Most buffer overflows identified in NSTB assessments were in the server applications that process control system protocol traffic. In most cases, values input from network traffic were intercepted and altered in transit. Network data bounds and integrity checking should therefore be implemented.

3.4.1.2 Common vulnerability: Integer overflow in control system service

Integer overflows result when an arithmetic operation attempts to create a numeric value that is larger than can be represented within the available storage space. For instance, adding 1 to the largest value that can be represented constitutes an integer overflow. The most common result in these cases is for the value to "wrap" around to the lowest value possible. This can lead to unintended behavior if the value is assumed to be positive and wraps around to a negative value. Many other unexpected situations may occur from operations that create invalid values that are too small or large, including potential buffer overflows if the unexpectedly small number is used as the number of bytes to allocate for a buffer.

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

The following is a specific finding associated with this vulnerability that was identified in multiple assessments:

- Integer overflow found in control system protocol.

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 values should be validated.

Even if values are never input directly by a user, it cannot be assumed that data will always be correctly formatted or that hardware or operating system protections are sufficient. It is possible for control system traffic to be intercepted and altered in transit. Network data value and integrity checking should therefore be implemented.

3.4.1.3 Common Vulnerability: Lack of Bounds Checking in Control System 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, if input is not validated, negative or too large numbers can be input for array access and cause essential services to crash.

Control system applications frequently suffer from coding practices that allow attackers to supply unexpected data and thus modify program execution. Even though control system 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:

- Crashed control system communications service by altering input value to negative number
- Lack of validation of number of points sent resulted in denial-of-service (DoS)
- DoS caused by out of range index values.

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 values should be validated.

Even if values are never input directly by a user, it cannot be assumed that data will always be correctly formatted or that hardware or operating system protections are sufficient. It is possible for control system traffic to be intercepted and altered in transit. Network data value and integrity checking should therefore be implemented.

3.4.1.4 Common Vulnerability: SQL Injection Vulnerability

Structured Query Language (SQL) injection vulnerabilities were identified in multiple assessments because control systems use real-time databases, configuration databases, and multiple historian databases in different security zones. Compromise of any of these databases may provide valuable information and the opportunity to insert or corrupt data, as well as control of the database server.

SQL injections were found by looking at the source code of applications and by trial and error. SQL injections typically occur when the developer is creating an SQL query that includes data provided from a user and does not filter escape sequences.

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

- Multiple SQL injection vulnerabilities found for control system commercial database
- Shell commands invoked through SQL queries on DMZ historian server
- Non-parameterized SQL statements are contained in some of the back-end Java coding; this may be susceptible to second order SQL injections.

Recommendation: Fix the vulnerable method of building SQL queries from non-validated user input by either adding filters to check user input and properly escaping any character that might lead to an SQL injection or switching to prepared SQL query statements. With prepared statements, the SQL query is compiled with placeholders and those placeholders are later filled with the data provided by the user. Since the SQL statement has already been compiled, it is not possible to perform an SQL injection.

Care still has to be taken to make sure there are no SQL injections within procedures and functions defined on the database. The prepared statements only prevent SQL injections from the user interface

(usually a Web application), and will have no effect in preventing SQL injections that might exist in functions and procedures defined within the database. Use parameterized SQL statements throughout the code.

Try to minimize the damage an attacker can do with an SQL injection by only granting access to the minimum number of resources that the Web application needs to operate. For example, if the Web application is only displaying data, revoke all write privileges from that Web application. It is also a good idea to have separate accounts for running versus developing the Web applications. This removes the ability to create tables and functions from the runtime user.

3.4.1.5 Mitigations for Lack of Input Validation Category

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. Control system-specific protocols should be redesigned to include strong authentication and integrity checks. IT products deployed on the control system network should also have passed a security review. Asset owners should explicitly address the security of these products during the procurement process.

3.4.2 Vulnerability Category: Vulnerable Communication Protocols

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.

3.4.2.1 Common Vulnerability: Standard IT Protocol Uses 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 is a common specific assessment finding associated with this vulnerability:

- Remote desktop encryption can be cracked.

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.

3.4.2.2 Common vulnerability: Standard IT Protocol Uses Clear Text Authentication

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

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

- Standard IT clear text authentication protocols are used by the control system
- Standard IT clear text authentication protocol services are running on multiple control system hosts.

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. This is straightforward 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.

3.4.2.3 Common Vulnerability: Control Ssystem Protocol Uses Weak Authentication

Commands from the Human-Machine Interface (HMI) cause actions in the control system. Alarms are sent to the HMI that notify operators of triggered events. The integrity and timely delivery of alarms and commands is critical in a control system.

Weak authentication in control system protocols allows replay or spoof attacks to send unauthorized messages. This means it is possible to send messages that update the HMI or RTU. The attacker may be able to cause invalid data to be displayed on a console or create invalid commands or alarm messages.

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

- Common control system protocol uses weak authentication between control system and field equipment (RTU)
- Proprietary control system protocol uses weak authentication between control system components.

Recommendation: The system design needs to implement strong authentication into control system communication protocols and encrypt communications if possible.

3.4.2.4 Common Vulnerability: Control System 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 control system protocol has poor integrity checks. This has the same effect as above, 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 control system communication paths and reverse engineered the control system network communications protocol, it is possible to manipulate the data flowing between the system 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 control system can be used to either trick the operator into performing actions the attacker wants, or to hide what an attacker is doing with the system.

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

- MITM altering of control system communication possible between control system and field equipment (RTU)
- MITM altering of control system communication possible between control system components.

Recommendation: Data integrity checks need to be designed and implemented in control system communication protocols. Use hardcoded Address Resolution Protocol (ARP) tables for static IP addresses or dynamic ARP inspection of dynamic IP addresses, if feasible. Monitoring the network traffic for changing MAC addresses using an 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.

3.4.2.5 Mitigations for vulnerable communications protocols

Control system vendors should implement secure-by-default settings for their control system products. These should include protocols that use strong encryption schemes for message authentication. control system-specific protocols should be redesigned to include strong authentication and integrity checks. During control system configuration, vendors, owner/operators, and integrators should pay special attention to end-point security, including proper storage of encryption keys, signing of digital certificates, and enablement of security settings.

3.4.3 Vulnerability Category: Weak User Authentication

Even if a protocol provides for strong authentication, it must be implemented correctly with strong passwords that are kept private. Users are responsible for creating and protecting authentication credentials.

3.4.3.1 Common Vulnerability: 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 five specific assessment findings associated with this vulnerability:

- Unsecure SSH implementation
- Access authentication for password management, accounting, and accessibility not implemented on firewall
- Login information remembered
- Network device authentication encryption can be broken. (Stronger encryption could have been chosen.)
- X11 implementation provides weak authentication.

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

3.4.3.2 Common Vulnerability: No Password Required

Some assessments discovered applications that had been configured without passwords.

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

- No authentication required between corporate clients and Web server on DMZ
- The Oracle listener was configured without a password on multiple assessments
- Network equipment required no Virtual Trunking Protocol (VTP) domain password.

Recommendation: Strong passwords need to be required and deployed on networking, client, and server equipment.

3.4.3.3 Common Vulnerability: Weak Passwords

The longer and more complex a password is, the longer it will take for it to be guessed or cracked. Cracking a password can be trivial or virtually impossible depending on the combination of different character types used and password length.

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

- Weak passwords were cracked
- Common administrative account passwords were used between LANs.

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.

3.4.3.4 Mitigations for Weak User Authentication Category

Mitigations for this vulnerability category include requiring login for all system management functions, enabling software features to require strong passwords, and implementing multi-factor authentication.

3.4.4 Vulnerability Category: Least Privileges Not Enforced

The least privileges policy grants privileges to a user only for necessary functions. Not every user of a computer requires administrative privileges with access to all files and folders. For further information see National Institute of Technology's (NIST's), "Recommended Security Controls for Federal Information Systems," SP800-53, which organizes security controls into families for ease of use.

Assessment findings show how access to files and applications was possible, even though it was not within the given process or user's need or permission.

3.4.4.1 Common Vulnerability: Unauthorized Directory Traversal Allowed

Findings were reported that directory traversal was allowed beyond intended file access. This was accomplished through either remotely connecting to the Web server or using a file transfer protocol.

The following are specific assessment findings associated with this vulnerability:

- Web server allowed directory browsing
- Web server allowed file traversal using ".." string for higher directories
- File transfer protocol permissions allowed directory traversal.

Recommendation: The file permissions on the Web server need to be set granting 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. Filter input to screen incoming filenames to exclude the ".." string. Disabling unused ports and keeping the Web server patched to current standards are also good practices. Write permissions are most dangerous, but read permissions may disclose valuable information or information that can be used for attack.

3.4.4.2 Common vulnerability: 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 control system 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:

- Many SQL connections were running with full admin rights
- Many common remote access services were running as root or administrator
- Web application executed scripts with root privileges, which could allow local users in that group to execute arbitrary commands as root by modifying the scripts
- Control system console application required administrative access to the system
- A backup service was running as administrator
- Remote exploitation of control system application services allowed root-level access on control system hosts.

Recommendation: By default, some control system installations start services as the root user and root group. Many services do not need to be started with this much privilege, and doing so exposes system resources to preventable risks. By restricting necessary privileges during control system 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 if it were to misbehave 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.

3.4.4.3 Common vulnerability: Control system protocols with unnecessary functionality

Some control system protocols could be exploited to perform unintended actions. In this case, protocols were designed for functionality without restricting unauthorized actions.

The following are specific assessment findings associated with this vulnerability:

- HMI was able to insert arbitrary alarms into the alarm database
- Historian was able to insert data into the real time database
- All control system hosts were able to modify the real-time database.

Recommendation: control system protocols and/or applications need to be redesigned to only include intended functionality and rights.

3.4.4.4 Mitigations for Least Privileges not Enforced Category

The principle of least privilege recommends that accounts have the least amount of privilege required to perform business processes. This encompasses user rights, resource permissions such as CPU limits, memory, network, and file system permissions.

A common problem is applications and services running with system or root-level privileges. If this is the case, and an attacker is able to exploit the application, the exploit code will run with system privileges. A number of software products run with these super user permissions by default, and yet will still function properly if running as a less-privileged user.

Another common problem is allowing users to operate computer systems (consoles, servers, etc.) with more permissions than necessary. For example, if the operator only needs access to the HMI, he should only have permission for that application. Under no circumstances should the operator be granted

administrative privileges. User accounts used for interactive logon should be carefully evaluated for the proper set of permissions.

A related issue is file permissions. File shares should be restricted to only those users who require access and limited to the access level they require. For example, if control system information is shared to everyone on the network, even if the control system network is segmented, an intruder gaining access to the control system network will have access to all control system-specific information. Restrict necessary communications and lower permission levels of users and applications to allowable functions. Share files only to required computers and user accounts**.** Give each user and process the minimal privileges necessary for system operation.

3.4.5 Vulnerability Category: Unpatched Systems

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.

3.4.5.1 Common Vulnerability: Unpatched Operating System

Unpatched operating systems open control systems 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 (MS)-SQL vulnerability that Slammer exploited. In fact, they did not know there was a patch, which Microsoft released 6 months before Slammer struck. See http://www.securityfocus.com/news/6767.

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

- Operating system vendor patches were not applied
- System computers vulnerable to operating system service vulnerability.

Recommendation: A timely patch management process is critical to reduce vulnerabilities. This process requires elements of IT, IT security, process control engineering, and senior management. It needs to incorporate 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.

3.4.5.2 Common Vulnerability: Unpatched Third-party Applications

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

The following are six assessment findings examples associated with this vulnerability:

- Unpatched and vulnerable Web server
- Unpatched and vulnerable database

- Buffer overflow in unpatched Oracle database
- Unpatched ICCP stacks
- Unpatched SSL libraries
- Unpatched XML libraries.

Recommendation: A timely patch management process is critical to reduce vulnerabilities. This process requires elements of IT, IT security, process control engineering and senior management. In-depth test bed testing of issued patches is important to identify adverse affects they may have on the control system before deploying patches on operational systems. Where patching is not an option, work arounds and defense in-depth techniques and tactics can be used. Statically linked libraries need to be independently kept up-to-date if they are different from the libraries associated with the operating system.

3.4.5.3 Mitigations for Unpatched Systems Category

Unpatched systems, including operating systems, control system software, and other applications, are often found in control system environments. Locating such systems is often achieved by a simple vulnerability scan. Systems may go unpatched due to high availability operational requirements and the challenges associated with extensive patch testing and multiple deployment locations.

Asset owners should monitor sources of vulnerability information for applicability to their environment. The asset owner should couple the vulnerability information feed with the configuration management process to identify the resources that are potentially affected by the vulnerabilities. The asset owner should have a risk management paradigm established to guide the decision of whether and when a patch should be deployed. The asset owner should have established a patch management methodology that includes thorough patch testing prior to deployment. The asset owner should have a deployment methodology that results in minimal down time and patches systems via predetermined priority ranking.

IT software is often included in vendor control systems. Vendors should use due diligence to ensure that the products they incorporate were written under a development process that emphasized security from the beginning of the development lifecycle. Vendors should ensure that IT products are at current version and patch levels prior to deployment at asset owner sites and support timely testing of patches for customers.

Asset owners should institute a rigorous change and patch management paradigm to identify patch releases and apply them in as timely a manner as possible. The decision of patch timeliness should be based on a sound understanding of risk and the threat environment, taking into consideration the simplicity, exposure, and impact of an attacker exploiting the unpatched vulnerability.

3.4.6 Vulnerability Security Dimension Summary

Five categories with common vulnerabilities were identified in the vulnerability dimension. The lack of input validation led to buffer overflows, integer overflows, out of bounds conditions, and SQL injection vulnerabilities in control system services. IT and control system communication protocols provided weak or no encryption or integrity checks. Weak user authentication was caused by improper security configurations, weak passwords, and the lack of required passwords. The lack of least privilege restrictions led to unauthorized directory traversal and services running as privileged users. Vulnerabilities were also found due to unpatched operating systems and third-party applications (see Figure 2).

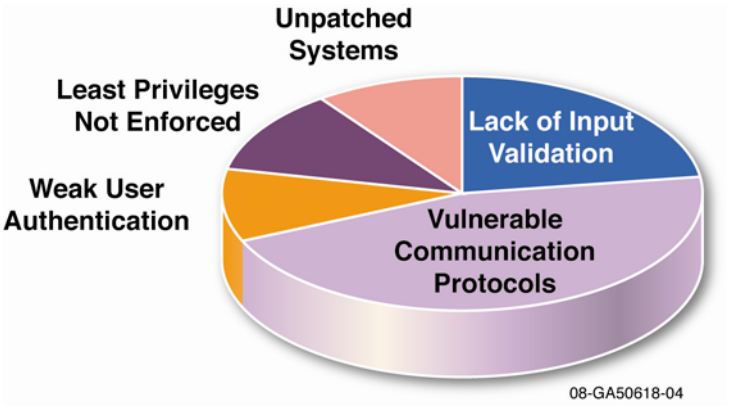

Figure 2. Frequency of vulnerability security dimension common vulnerabilities by category.

3.4.6.1 Summary of Recommendations for Vulnerability Security Dimension

Asset owners must use procurement specifications to ensure that security development lifecycle requirements are met by the vendor. Asset owners may also 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. Firewalls, intrusion detection systems, 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 utilize monitoring (i.e., access logs and intrusion detection systems) to mitigate risk introduced by the presence of unpatched vulnerabilities until patches can be properly tested and deployed.

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.

3.5 Damage Potential Security Dimension

Damage potential is the amount of loss that a malicious attacker has the power to cause once they have compromised the control system. The amount of damage that can be caused by a compromised control system is determined by the type of process that it controls and by the nature of engineered safety systems (e.g., physical safety mechanisms, redundant systems, and defense in depth measures may be in place preventing significant damage despite successful attack on the electronic control system).

For onsite assessments this estimate is feasible and may be based on a worse-case loss estimate or an existing safety risk analysis. For laboratory assessments there is no field-scale physical process that is being controlled, so the damage potential must be evaluated differently, perhaps by noting whether there is an independent safety system associated with the rest of the system.

For assessments in general the safety system should be evaluated from a security perspective. To date, thorough investigations of the electronic safety mechanisms, independence of the safety instrumented

systems, and ease of safety system compromise have not occurred on the NSTB program. Consequently, there are no common vulnerabilities to report at this time.

3.6 Detection Security Dimension

Intrusion detection is the act of detecting actions that attempt to compromise the confidentiality, integrity, or availability of a resource. Intrusion detection does not include prevention of intrusions. Intrusion detection can be performed manually or automatically. Manual intrusion detection is done by examining log files or other evidence for signs of intrusions, including off normal network traffic. Automated approaches use intrusion detection systems to monitor system logs, network traffic flow, and packets. When a "probable intrusion" is identified, an alert is sent. Determining what the probable intrusion actually is and taking some form of action to stop it or prevent it from happening is outside the scope of intrusion detection systems. Intrusion prevention systems are not generally recommended along paths of critical functionality.

Intrusion detection was not a focus for laboratory assessments, but in many cases it was noted whether the system gave any indication of abnormal conditions, such as alerts on the operator screen. There was a general lack of adequate control system indicators of abnormal conditions. Onsite assessments also found intrusion detection systems to be lacking in their installation, monitoring, and/or updating. Better indicators of abnormal system traffic and behavior should be built into operator screens. IDS systems should be deployed, tailored to the control system architecture and traffic, and continuously monitored. Control system networks are generally static in nature and IDS rules can therefore be developed to look for abnormal behavior such as a protocol that should not be used between two computers.

3.7 Recovery Security Dimension

Recovery is associated with the ability to restore the control system from a compromised state to an uncompromised state. It is dependent on the reliability and security of both the backup and the restore processes and mechanisms. To date, NSTB assessments have not thoroughly assessed the security and potential vulnerabilities in these processes and mechanisms, so there are no common vulnerabilities to note at this time.

4. SUMMARY

The U.S. DOE established the NSTB Program to help industry and government improve the security of the control systems used in the nation's critical energy infrastructures. A key part of the program is the assessment of control systems to identify vulnerabilities that could put the systems at risk from a cyber attack. Assessments were performed in the INL SCADA Test Bed and in operational control system installations.

Laboratory 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.

In order to help prevent disclosure of vulnerabilities that may not have been patched, assessment agreements with control system vendors and owners include some form of non-disclosure agreements that give the vendors and owners the right to control dissemination of assessment findings. 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 unnecessary exposure of their customers to potential cyber attacks. Whether results were shared with control system customers, security awareness has been increased by sharing general control system security knowledge, which was gained through the assessment process. Cyber security and control system researchers have presented results and provided security training to attendees of the various vendor user group conferences. Many user groups have now established breakout sessions dedicated wholly to control system cyber security issues. Some of these breakout sessions have evolved into user consortiums that work together to fund additional testing on the control systems they use.

This common control system vulnerability report provides another avenue for sharing the control system security knowledge gained through the NSTB program. After searching for standard vulnerability categorization methods to present the common assessment findings, the INL control system cyber security technical metrics taxonomy was selected as the best fit. Seven control system security dimensions were divided into subcategories of findings identified in order to support security analysis at multiple levels of granularity. Detailed vulnerability descriptions were then generalized to remove system identifying information and to aid in creation of common vulnerability categories. Some sanitized individual vulnerability details were included in the vulnerability discussions in this report to promote better understanding of the findings. Vulnerabilities were considered common when at least two assessments had findings that could be generally described as the same vulnerability.

The common vulnerabilities in Table 2 indicate that control systems often integrate third-party applications and services with known vulnerabilities as well as integrating clear text and weak authentication communication protocols into their functionality. Other vulnerabilities are built into these systems because of unsecure custom software design, implementation, and coding practices.

Onsite assessment findings indicated that control system administrators need to be more diligent in verifying and documenting their current system configuration. They must also more effectively determine the required communication paths based on IP address, port number, and direction, secure the communications protocols to the extent possible, and then block all other communications. Intrusion detection and antivirus applications can then be used to help detect attacks.

Control system 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 control system 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 control systems, asset owners cannot always directly fix them. Thus, as they wait for vendor patches and fixes, the asset owners should design and implement defense in depth security strategies that aid in protecting the control system from attack.

5. REFERENCES

1. Permann, May, Kenneth Rohde, “Cyber Assessment Methods for SCADA Security,” November 2005, http://www.isa.org/InTechTemplate.cfm?Section=Article_Index1&template=/ContentManagement/ContentDisplay.cfm&ContentID=49890.
2. McCumber, John, “Assessing and Managing Security Risk in IT Systems: A Structured Methodology,” Auerbach Publications, 2004.
3. Berg, Michael, Jason Stamp, “A Reference Model for Control and Automation Systems in Electric Power,” Sandia National Laboratories report SAND2005-1000C, 2005.
4. CWE (Common Weaknesses Enumeration), MITRE, August 23, 2008, http://cwe.mitre.org/.
5. IATRP, “INFOSEC Assurance Training and Rating Program,” August 23, 2008, http://www.iatrp.com/iam.php.
6. Seacord, Robert C., Allen Householder, “A Structured Approach to Classifying Security Vulnerabilities,” Carnegie Mellon University Technical Note, CMU/SEI-2005-TN-003, 2005.
7. McQueen, Miles, Wayne Boyer, Sean McBride, Marie Farrar, Zachary Tudor; “Measurable Control System Security through Ideal Driven Technical Metrics,” S4: SCADA Security Scientific Symposium, January 23, 2008.

Appendix A

Taxonomy Methodologies

Appendix A

Taxonomy Methodologies

This section describes the taxonomy used to categorize the findings in this aggregated report. There is no generally accepted taxonomy for control system security assessments. Several published taxonomies were considered, but they were not chosen for this report because they did not provide a good fit to the INL assessment paradigm. For example, the Confidentiality, Integrity, Availability (CIA) McCumber Cube model[2] is a well-known framework for information security evaluation, but it is a high-level classification of what is protected rather than a taxonomy of assessment findings and is not oriented to control systems. Other taxonomies considered were the Sandia National Laboratory (SNL) Reference Model,[3] Mitre Common Weaknesses Enumeration (CWE),[4] National Institutes of Technology (NIST) and National Security Agency (NSA) Information Security (INFOSEC) Assessment Methodology (IAM),[5] and Carnegie Mellon University (CMU) structured vulnerability classification.[6]

A-1. Security Dimension Taxonomy

The INL taxonomy was used to categorize each finding first at a primary level, then at a secondary level of granularity. The top level of the taxonomy is one of seven abstract dimensions of security. These seven security dimensions are the basis of control system cyber security technical metrics proposed by McQueen et al.[7] These seven security dimensions represent the main system attributes that affect the risk of loss from cyber attacks. Every item of interest (i.e., finding, vulnerability, weakness, concern, or observation) identified in the collection of reviewed assessment reports mapped to at least one of the seven dimensions of security. However, vulnerabilities were not found in all security dimensions due to the focus of assessments. And assessments did not analyze damage potential and recovery ability.

These security dimensions were further subdivided by the categories of findings identified in order to support analysis at multiple levels of granularity. The categories were influenced by the assessment findings, and are expected to be modified and extended as more assessments are completed.

Vulnerability descriptions were generalized to remove specific vendor-identifying information and details that would hinder the ability to group common vulnerabilities. Sanitized individual details are referenced in the vulnerability discussion in this report to promote understanding of the findings. A finding is defined as a minimum of two assessments having the same vulnerability.

A-2. Security Dimension Definitions

A-2.1 Security Group Knowledge

The security group knowledge ideal is defined as aspects of the system or associated management processes that impact the ability of the security group to know the system and manage changes to the system. It includes aspects of the system and processes associated with configuration management, tools (or lack of tools) that support the tracking of changes, and the collection and analysis of system logs and forensics. Ideally, the security group has perfect knowledge of the control system.

A-2.2 Attack Group Knowledge

The Attack Group Knowledge dimension of security is the set of system attributes, processes, or actions that provide potential attackers with means to gain information about the system. It includes

software defects or configuration settings that return information when the system is probed by an unauthenticated user, any information about the system that can be obtained from public sources, and design or implementation weaknesses that allow a user with no authenticated privilege to gain information by listening on communication paths. Ideally, potential attackers have zero knowledge about the control system.

A-2.3 Access

Access is the dimension of security that provides a potential attacker with the ability to send or receive data to/from a component of the control system from the attacker's location. This includes physical access to control system components, access to control system components through external and internal networks, and access from internal components that may have been compromised. Since there is always the possibility of unknown "zero day" vulnerabilities existing for a component, access does not address whether or not the communication channel is known to provide useful information, nor whether attacking it will provide the attacker with any particular desired result. Ideally, there are no data pathways between the control system and the location of potential attackers.

A-2.4 Vulnerabilities

A vulnerability is any defect or weaknesses in the control system that can be exploited to gain unauthorized privilege or compromise its operation. This includes, but is not limited to, known vulnerabilities, the ability to identify previously unknown vulnerabilities, and estimates of the number of zero day vulnerabilities in the system. This dimension of security excludes defects that allow information to be obtained once access is gained without also explicitly gaining privilege. If a single defect allows an attacker to gain information and also gain privilege, that defect is defined as a vulnerability. Ideally there are no defects in design or implementation that potential attackers can exploit to gain unauthorized privilege. Note that for an attacker to gain unauthorized privilege and compromise control system operations, all three of the following conditions must be present: attacker knowledge, access, and vulnerability.

A-2.5 Damage Potential

The damage potential is defined as the amount of loss that a malicious attacker has the power to cause once they have gained privilege on a control system. It does not include any weaknesses associated with the process of gaining malicious control. Although actual damage may be reduced by a quick response to an attack, this dimension does not include any effects associated with attack detection or control system recovery. Ideally, there are mechanisms independent of the control system, which prevent damage to the plant even if the control system is fully compromised.

A-2.6 Detection

Detection is the dimension of security that is associated with the ability to detect attacks and provide timely notification. This includes anti-virus software, intrusion detection systems, intrusion prevention systems, system logging, installation or implementation of detection mechanisms, and the effectiveness of those mechanisms. Ideally, any attack is detected instantly and the Security Group is notified immediately of the detection.

A-2.7 Recovery

Recovery is the dimension of security associated with the ability to restore the control system from a compromised state to an uncompromised state. It includes the reliability of both the backup and the

restore facilities and the time required to recover from an attack. Ideally, a compromised control system can be instantly restored to an uncompromised state.

A-2.8 Categories

The security dimensions were subdivided into categories of findings (see Table A-1). The categories were influenced by the assessment findings; thus, they are expected to be modified and extended as more assessments are completed.

Table A-1. Control system cyber security taxonomy used to classify NSTB assessment findings.

Security Dimension	Category
Security Group (SG) Knowledge	Change management deficiency (Legacy configurations not removed)
	Documentation deficiency (operation and maintenance requirements, training requirements, audit requirements, backup requirements, Roles & Responsibilities, etc.)
	Procedure deficiency (lack of standard/documented procedures)
Attack Group (AG) Knowledge	Information leaks, enumeration (e.g., clear text communications)
	Open source information available to attackers
Access	Firewall filtering deficiency
	Remote access deficiencies
	Physical access to data transfer paths (e.g., unrestricted USB ports, devices outside protected area)
Vulnerability	Lack of input validation
	Communication protocols with weak or no authentication
	Weak user authentication
	Least privileges not enforced
	Unpatched systems
Damage Potential	Lack of safety system
	Safety system not independent of control system
Detection	Host detection mechanism deficiency
	Network detection mechanism deficiency
	Lack of anti-virus software
Recovery	Excessive recovery time
	Lack of ability to recover

A-3. Identification of Recommended Mitigations

Reported assessment findings were extracted, combined, and categorized as described above, and were given corresponding recommendations and/or mitigations. These recommendations and/or mitigations are based upon 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 control system software. 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.

A-4. 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 are not necessarily known.

Note that not all systems were tested identically, not all vulnerability types were looked for in all systems, and no single standardized testing methodology was used. Not every assessment examined each defined security dimensions. 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, but it is expected that new common vulnerabilities will be identified in the future that have not been tested for in all assessments.

Appendix B

Terms and Definitions

Appendix B

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. There are many methods for identifying and authenticating users, such as passwords, identification cards, smart cards, and biometric systems.

ACL or Access Control List
An Access Control List (ACL) 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 sub-network that interfaces an organization's external services to a larger, untrusted network, usually the Internet. The purpose of a DMZ is to add an additional layer of security to an organization's Local Area Network (LAN).

DNP3
Distributed Network Protocol is a set of communications protocols used between components in process automation systems. Its main use is in utilities, such as electric and water companies. Usage in other industries is not common, although technically possible. Specifically, it was developed to facilitate communications between various types of data acquisition and control equipment. It plays a crucial role in SCADA systems, where it is used by SCADA Master Stations (Control Centers), Remote Terminal Units (RTUs), and Intelligent Electronic Devices (IEDs). It is used only for communications between a master station and RTUs or IEDs. ICCP, the Inter-Control Center Protocol, is used for inter-master station communications.

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 the software was designed for. By doing this the testing, it will uncover areas of the code that were inadequate in handling input values outside of 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 attack (MITM) 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.

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/or 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.