INL/CON-07-13581
PREPRINT

Measurable Control System Security Through Ideal Driven Technical Metrics

SCADA Security Scientific Symposium

Miles McQueen
Wayne Boyer
Sean McBride
Marie Farrar
Zachary Tudor

January 2008

The INL is a
U.S. Department of Energy
National Laboratory
operated by
Battelle Energy Alliance

Measurable Control System Security through Ideal Driven Technical Metrics

Miles McQueen, Wayne Boyer, Sean McBride
Idaho National Laboratory
{miles.mcqueen, wayne.boyer, sean.mcbride}@inl.gov

Marie Farrar
Securicon, LLC
marie.farrar@Securicon.com

Zachary Tudor
George Mason University
ztudor@gmu.edu

Abstract: The Department of Homeland Security National Cyber Security Division supported development of a small set of security ideals as a framework to establish measurable control systems security. Based on these ideals, a draft set of proposed technical metrics was developed to allow control systems owner-operators to track improvements or degradations in their individual control systems security posture. The technical metrics development effort included review and evaluation of over thirty metrics-related documents. On the bases of complexity, ambiguity, or misleading and distorting effects the metrics identified during the reviews were determined to be weaker than necessary to aid defense against the myriad threats posed by cyber-terrorism to human safety, as well as to economic prosperity. Using the results of our metrics review and the set of security ideals as a starting point for metrics development, we identified thirteen potential technical metrics - with at least one metric supporting each ideal.

Two case study applications of the ideals and thirteen metrics to control systems were then performed to establish potential difficulties in applying both the ideals and the metrics. The case studies resulted in no changes to the ideals, and only a few deletions and refinements to the thirteen potential metrics. This led to a final proposed set of ten core technical metrics. To further validate the security ideals, the modifications made to the original thirteen potential metrics, and the final proposed set of ten core metrics, seven separate control systems security assessments performed over the past three years were reviewed for findings and recommended mitigations. These findings and mitigations were then mapped to the security ideals and metrics to assess gaps in their coverage. The mappings indicated that there are no gaps in the security ideals and that the ten core technical metrics provide significant coverage of standard security issues with 87% coverage.

Based on the two case studies and evaluation of the seven assessments, the security ideals demonstrated their value in guiding security thinking. Further, the final set of core technical metrics has been demonstrated to be both usable in the control system environment and provide significant coverage of standard security issues.

Keywords: Cyber Security Metrics, Control System Security

1 Introduction

Electronic control systems that operate much of the Nation's critical infrastructure are increasingly connected to public networks. Therefore, control systems and the associated critical infrastructure are at risk from cyber attacks. Meaningful metrics are needed to make informed decisions that affect system security. The Department of Homeland Security National Cyber Security Division supported development of a small set of security ideals and associated metrics to provide control system owners/operators guidance in managing their system security.

A metric is a standard of measurement [Jac07]. The scope of this paper is limited to quantitative technical metrics. A cyber security technical metric is the security relevant output from an explicit mathematical model that makes use of objective measurements of a technical object. Other types of metrics (such as operational and organizational metrics, and metrics that are qualitative such as "low impact" or "highly unlikely") can provide insights about security but are beyond the scope of this work.

An important use of technical metrics is in the estimation of risk where risk is defined as the probability of an event times the consequence of the event. The risk we would like to measure is the expected value of the loss from cyber attacks per unit time. Risk is usually measured in dollars or lives. The estimation of risk could provide the ability to weigh the benefits versus costs of security counter measures. However, a credible estimation of cyber security risk in real world control systems is not currently feasible because the problem involves an unpredictable intelligent adversary and very complex systems. Previous work [MBF05] proposed "mean time-to-compromise" as a security metric and proposed a simple method for calculating it as a function of the number of known vulnerabilities. A method was also proposed for estimating risk reduction for a simple control system using the mean time-to-compromise metric [MBF06]. Unfortunately, those methods require simplifying assumptions that are not valid in general.

In our opinion a good set of metrics should support the concept of risk estimation within the practical constraints of what is currently objectively measurable and under the control of the defender. A good set of metrics should have the following attributes: The number of metrics should be small (less than 20)[1] to be manageable; the metrics should be easy to understand, measurable and objective; the metrics should be directly related to security risk; and the set of metrics should represent the most important measurable security attributes of the system. Previous work [BM07] introduced the concept of security metrics based on seven ideals of security and proposed a set of metrics intended to meet the above criteria. This paper is an extension of that work.

2 This Paper's Contribution

This paper presents a set of technical metrics for control systems that are ideal driven. The ideals and associated metrics are identified and described. The ideals and associated metrics were evaluated and refined by case studies and by evaluation of several security assessments. The proposed metrics can be used to assess security improvements, guide security thinking, and make risk assessments.

---

[1] NIST 800-55 [SBS03] recommends that to keep the set of metrics manageable, the number of metrics should be about 10 and no more than 20.

3 Survey of Previously Proposed Metrics

Thirty guides and standards documents (including, for example, references [CCH06], [CSC06], [RKJ06], [SBS03]) were reviewed in search of technical metrics that have previously been defined and recommended [INL06]. A sampling of security metrics used by some industries were also included in the investigation. Most of the metrics found in the standards and guides do not meet our definition of a technical metric. We found no case where a standards document recommended the use of a specific metric or set of metrics. The specific metrics described in standards documents are generally provided as examples rather than as recommended metrics.

Each of the few identified technical metrics was analyzed by considering the circumstances in which the metric provides a meaningful security representation and when it is misleading. We evaluated the strengths and weaknesses and concluded that existing metrics have serious weaknesses. For example, many of the metrics were simply a percent of the system components that implemented a certain type of security control mechanism. However, the fractional implementation of a given security mechanism does not necessarily correlate to risk. A specific metric defined in industry is "Average number of vulnerabilities per system component". This metric has the following strengths: It is easy to understand and it is easy to obtain estimates using automatic scanning tools. The problem of using an average is that all vulnerabilities and all components of the network are given equal weight. Consider the case where there is one easily exploitable vulnerability that allows penetration of a critical system component while there are zero known vulnerabilities on the other system components. Now consider a case where there are no known vulnerabilities on critical components, no vulnerabilities that allow penetration from an external site, but there are many minor vulnerabilities on non-critical system components. The former case is a high-risk situation, but the metric indicates low risk while the latter case is a low-risk situation, but the metric indicates high risk. The assumption that all vulnerabilities and all components are of equal value is false for most systems. The metric can be improved by counting the number of vulnerabilities for each group of components with similar security implications and for vulnerabilities with similar effects (e.g. external penetration versus privilege escalation).

The results of our investigation of existing technical metrics showed the need for the definition of a small set of technical metrics that operators of control systems can use to gain better insight into their security risk.

4 Security Ideals

Seven ideals are the basis for our proposed metrics. Each ideal is associated with an abstract dimension of cyber security and represents a system condition at a given point in time such that perfection has been achieved for its associated dimension of security. The seven dimensions of security and the respective ideals are listed in Table 1. We chose the ideals in Table 1 based on our study and experience in the cyber security field and we assert that each of these ideals is strongly related to security risk.

It is generally accepted that the objective of computer security is the protection of confidentiality, availability and integrity of computer systems. Security principles support that objective. We assert that our seven security ideals are consistent with generally accepted security principles. To support that assertion we successfully mapped security principles from

Bishop [Bis03], Neumann [Neu95], Schneier [Sch00], NIST [SG96] and Summers [Sum97] to our seven ideals.

4.1 Security Group (SG) knowledge

The first abstract dimension is Security Group (SG) knowledge. The security group represents the group of people (or person) who are responsible for the security of the control system. In the ideal situation, the security group has perfect knowledge of the system including all the components, how they fit together, and the vulnerabilities. That knowledge is needed to protect the system from attackers. Perfect knowledge of the system implies a configuration

Table 1. Seven abstract dimensions of security and associated ideals

Security Dimension	**Ideal**
1. Security Group (SG) knowledge	1. Security Group (SG) knows current control system perfectly
2. Attack Group (AG) knowledge	2. Attack Group (AG) knows nothing about the control system
3. Access	3. The control system is inaccessible to AGs
4. Vulnerabilities	4. The control system has no vulnerabilities
5. Damage potential	5. The control system cannot cause damage
6. Detection	6. SG detects any attack instantly
7. Recovery	7. SG can restore control system integrity instantly

management process that includes the SG in the planning of all changes and provides a mechanism for notifying the SG of any unauthorized changes. We assert that security risk is strongly correlated with SG knowledge.

4.2 Attack Group (SG) knowledge

The second abstract dimension is Attack Group (AG) knowledge. The attack group represents any potential attacker. Ideally, anyone who is not authorized to use the control system should be prevented from gaining knowledge of its design or configuration and be unable to obtain any information that would allow them to plan an attack. We assert that security risk is very low when potential attackers are unable to obtain any information about the control system. Users may become members of the AG when their actions on the system go beyond what they are authorized to perform, whether inadvertently or intentionally (the “insider threat”).

4.3 Access

The third abstract dimension is Access. The ideal situation from a security perspective is to provide no access to the control system from any location where there are potential attackers. That ideal includes the absence of any electronic connections between the Internet and the control system. Even though authentication mechanisms are designed to prevent unauthorized use of data transfer paths, we assert that the existence of all paths, authenticated or not, negatively impacts security.

4.4 Vulnerabilities

The fourth abstract dimension of security is **Vulnerabilities**. A vulnerability is any weakness or defect in the system that provides a potential attacker with a means to gain privilege intended for authorized users only. An exploit of a vulnerability leads to a compromise. An ideal system has no weaknesses and no defects.

4.5 Damage Potential

The fifth abstract dimension is Damage Potential. An ideal control system cannot cause damage. Since risk is the expected value of loss, the damage potential is directly proportional to risk. 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.

4.6 Detection

The sixth abstract dimension is Detection. An ideal control system includes detection mechanisms that alert the Security Group whenever there is an unauthorized event on the control system.

4.7 Recovery

The seventh abstract dimension is Recovery. An ideal control system can be restored to an uncompromised state immediately. Recovery time is related to Damage Potential because the cost of a successful attack correlates with the length of time that the control system is in a compromised state. Damage will tend to be less severe if the time to recover is minimized. However, the relationship between Recovery Time and Damage Potential is highly non-linear and highly system dependent.

4.8 Ideals Drive Metrics

Although perfection is probably not feasible for any of these seven dimensions of cyber security, the measurement of how closely each ideal has been achieved is a useful way to think about security metrics. The achievement of zero security risk is not realistic. From these seven ideals, we defined a small set of metrics that provide a measure of how successfully the system approaches each ideal.

5 Identification of Technical Metrics

A potential set of thirteen technical metrics related to the security ideals is listed in Table 2. Each metric is associated with one security ideal and there is at least one metric associated with each of the seven cyber security ideals. Each defined metric is intended to answer the question "what can be objectively measured on the system that is a reasonable representation of how closely the ideal has been realized?" The following sections briefly discuss each metric in our potential set.

Table 2. Potential Set of Thirteen Technical Security Metrics with corresponding ideal.

Security Ideal	Metric
1. Security Group (SG) knows current control system perfectly	Rogue Change Days
	Component Test Count
2. Attack Group (AG) knows nothing about the control system	Data Transmission Exposure
3. The control system is inaccessible to AGs	Reachability Count
	Attack Path Depth
	Root Privilege Count
4. The control system has no vulnerabilities	Known Vulnerability Days
	Password Crack Time
	Attack Surface
5. The control system cannot cause damage	Worst Case Loss
6. SG detects any attack instantly	Detection Mechanism Deficiency Count
	Detection Performance
7. SG can restore control system integrity instantly	Restoration Time

5.1 Rogue Change Days

Rogue Change Days is the number of rogue changes multiplied by number of days the changes were unknown to the Security Group (SG). A rogue change is any change to the system configuration without prior notification to the SG.

A key assertion is that the security risk from changes to the system without notification to the security group is, on average, worse than for changes which are planned and implemented in a well managed system. One weakness of this metric is that it does not include any measure of the actual security impact of the rogue changes.

For this metric the set of objects under change control must first be established and a version identifier must be saved for each object to establish a baseline. Periodically the current version identifier is scanned and compared to the previously saved identifier. Examples of objects under configuration management are: Programmable logic controllers (PLCs), Human machine interfaces (HMIs), critical computer files, network devices attached to the local network, etc.

Each type of configured object must have an associated mechanism for identification that produces an identifier that an audit program can obtain from the object. For example, computer files may have a hash function applied to the file content to calculate an identifier that can be used to determine if the file has changed.

Mathematical model:

S_T == An ordered set of version identifiers for all configured objects, measured at time T.

S_{T+k} == An ordered set of version identifiers for all configured objects, measured at time T + k.

TSC_{T+k} == Number of mismatches between sets S_T and S_{T+k}

CC_{T+k} == Changes introduced into the system only after notification of the security group,

RC_{T+k} == TSC_{T+k} - CC_{T+k} is the number of Rogue Changes between the current measurement of the system and the previous measurement of the system.

Rogue Change Days == RC_{T+k} * k

5.2 Component Test Count

Component Test Count is the number of control system components that have not undergone independent security testing. This metric is included in our potential set because we recognize the importance of security testing. A key assertion is that independent security testing of the system components will reduce risk. An independent test is one that is performed by personnel that are not under the direct employ of the component vendor.

5.3 Data Transmission Exposure

Data Transmission Exposure is the unencrypted data transmission volume. A key assertion is that any data that can be monitored by a potential attacker increases the security risk. Some data are clearly more sensitive than others but to make the metric easier to obtain we propose that this metric be a count of the number of unencrypted communication channel pairs in use by devices within the control system boundary. For a TCP/IP network, it is the number of unencrypted machine TCP-port pairs in use (as observable by network monitoring). Some network paths are more critical than others so the security manager may choose to categorize network connections (e.g. publicly accessible, internal) and track this metric for each network category.

5.4 Reachability Count

Reachability Count is the number of access points (relative to a specific point of origin such as the Internet). A key assertion is that a reduction in the number of access points tends to reduce the cyber security risk.

This metric requires complete network configuration information including connectivity, firewall/router rules and open ports. It also requires information about physical access to computer ports. An electronic scan from the point of origin is one method for obtaining information about the network communication paths. Physical access to portable storage media drives can be done by inspection.

Mathematical model:

N_s == Number of (services) that respond to data transmitted from the point of origin. For TCP/IP networks, it is the number of open TCP/UDP ports that can be reached from the point of origin.

N_o == Number of active physical network ports with outgoing network connectivity from a control system machine to the point of origin. "Outgoing network connectivity" means the network configuration allows the physical port to originate two-way connection-oriented sessions to some machine located at the point of origin. (Note: strict one-way outgoing data transmission is OK) Examples of physical network ports that meet this definition of "outgoing network connectivity" are: 1) An Ethernet card connected to the control system network and with unrestricted outgoing TCP/IP connectivity to the Internet, 2) a dialup modem on a machine that is also connected to the control system, 3) a wireless network card on a laptop computer that is also connected to the control system.

N_p == Number of physical access points to unrestricted portable storage media drives, including unrestricted USB ports.

N_T == Total reachability count

$N_T = N_s + N_o + N_p$

The security manager may choose to combine the network and physical reachability counts or track them separately. Because of the possibility of penetration of the perimeter the security manager may choose to calculate this metric at multiple points of origin within the network perimeter such as at the DMZ or behind each firewall. The measurement of reachable ports/services includes all the cases of crafted packets that exploit known vulnerabilities in firewalls and routers, such as the spoofing of IP addresses and packet fragmentation to disguise the targeted TCP port number.

The point of origin for physical access may be "outside the fence" or some other partially controlled area or combination of areas within the fence as defined by the security manager. Examples of restricted portable storage media drives that should not be included in the count of physical access points are:

- USB ports that are disconnected, physically disabled, or locked.
- USB ports that have host-based or device-based port encryption.
- USB ports restricted by end-point control software.

5.5 Attack Path Depth

Attack Path Depth is the minimum number of independent single machine compromises required for a successful network attack. This metric emphasizes the need to avoid a protection configuration that can be defeated by a single exploit or compromise. There may be common vulnerabilities on various paths of entry, therefore the attack steps may not be truly independent and this metric may be optimistic. To calculate this metric, determine the reachability (as defined for the reachability metric) from each network access point to every machine on the network. Machine X can be compromised from machine Y if machine Y is reachable from machine X.

Mathematical model:

> Attack Path Depth == Minimum number of compromises required to reach any machine in the set S from the public network by traversing reachable network paths. S is the set of machines such that if any machine in the set S is compromised then the attack is considered to be successful.

5.6 Root Privilege Count

Root Privilege Count is the number of unique user IDs with administration (root) access privilege. A key assertion is that risk is strongly related to the principle of least privilege. This metric is a simple measure of how well this principle is being followed.

5.7 Known Vulnerability Days

Known Vulnerability Days is the sum of known and unpatched vulnerabilities, each multiplied by their exposure time interval. A key assertion is that the longer a vulnerability is known the greater the risk it will be exploited.

Mathematical model:

N = Number of known vulnerabilities that currently apply to the system.

T_i = Discovery date of vulnerability i

t = current date

T == Total vulnerability days

$$T = \sum_{i=1}^{N} (t - Ti)$$

For publicly disclosed vulnerabilities, the discovery date is the disclosure date from the public vulnerability database. For vulnerabilities that are discovered locally, such as configuration errors, it is the local discovery date. Vulnerabilities that apply to the system may be identified by vulnerability test tools and by comparing system components to the components associated with publicly disclosed vulnerabilities. The system should be scanned for vulnerabilities often (suggest weekly or when there is a known configuration change). Public vulnerability databases should be checked regularly and often (suggest daily). This metric is affected by vulnerability discovery rate and by patch rate. Vulnerabilities may result from design errors, implementation errors and from mis-configurations such as inappropriate trusted relationships between machines. Some vulnerabilities are more significant than others. Tools such as Attack Graphs [XBM06] can be used to determine priority categories for all known vulnerabilities. The Common Vulnerability Scoring System (CVSS) [Sch05] is another suggested mechanism for prioritizing known vulnerabilities. This metric should be applied separately for each vulnerability category.

5.8 Password Crack Time

Password Crack Time is the shortest time (in days) needed to crack a single password for any account on the system. A key assertion is that the system security tends to improve when the password crack time increases. This metric is a valid measure of the minimum amount of time an attacker would need to compromise the system by password cracking. The password age should be subtracted from the password cracking time. One weakness of this metric is that it does not measure the strength of other authentication mechanisms but passwords are the most common form of authentication.

Data collected for this metric is the encrypted password files (hashes) from all machines on the system. For example, all password files from UNIX servers, Configuration data for Web Servers, Database Servers, Windows workstations, Control System HMI, etc. A password cracking tool is then applied to each password file instance. The metric is simply the minimum time needed to crack a single password.

Password cracking tools are available commercially and for free download [JTR06]. Data should be collected whenever passwords change. This metric is an important measure because passwords are by far the most common form of authentication. The value of the metric should be greater than the password expiration time. This metric is independent of password policies because it measures the least amount of time an attacker would need to crack a password if the encrypted password data is available to the attacker. If a very weak password is used, (including a default vendor supplied password) an attacker can guess the password without obtaining the encrypted password files and this metric would detect that high risk situation because good password cracking tools crack very weak passwords virtually instantly. Passwords used for authentication at the perimeter are particularly important and therefore perhaps should be measured for strength separately from other passwords used on the system. The security manager should ensure that vendor supplied passwords and passwords commonly used by maintenance personnel are included in the password cracker's dictionary. A list of commonly used and publically disclosed default passwords will be found in [INL07].

5.9 Attack Surface

Attack Surface is a measure of potential vulnerability. Key assertions are 1) vulnerabilities exist that are currently unknown to the defender and 2) the attack surface complexity, including external interfaces is strongly correlated to the potential for the discovery of new vulnerabilities. Attack surface has been proposed as a security metric for software systems by Manadhata and Wing [MW05]. This metric is considered to be potentially very valuable as an indirect measure of vulnerability.

5.10 Worst Case Loss

Worst Case Loss is the maximum dollar value of the damage/loss that could be inflicted by malicious personnel via a compromised control system. A key assertion is that system risk is strongly related to worst case loss. Although there can be successful attacks where the actual loss is much less than worst case, we assert that a reduction in the worst case loss reduces the potential for loss and therefore reduces risk. The worst case loss can probably be estimated from an existing safety analysis associated with the plant. The metric is the answer to the question "If the control system is under the control of a malicious person, what damage can be done?". If safety systems are not completely independent of the control system (for example a safety system connected to the control system network) safety systems may also be compromised by an attacker, therefore it should not neccessarily be assumed that such a safety system will prevent damage when estimating the Worst Case Loss.

5.11 Detection Mechanism Deficiency Count

Detection Mechanism Deficiency Count is the number of externally accessible devices without any malware detection or attack detection mechanisms. A key assertion is that detection mechanisms reduce risk especially when applied to devices that can be used as entry points for attacks.

5.12 Detection Performance

Detection Performance is a measure of the effectiveness of the detection mechanisms (intrusion detection system, anti-virus software, etc.) implemented on the system. The metric can be defined as detection probability discounted by false alarm rate. The metric should be applied separately to each of the detection mechanisms used on the system.

A suggested mathematical model:
N = Number of attack test cases
D = Number of attack test cases detected
Pd = D/N = Probability of detection.
F = Number of false alarms during tests.
Pfa = F/(D + F) = Probability of false alarm.

Detection Performance = Pd * (1 – Pfa)

This metric is difficult to obtain currently but is theoretically measurable. There is some public data available but better tests and tools are needed. Some intrusion detection products have been evaluated by Lincoln Laboratories [Mel03].

5.13 Restoration Time

Restoration Time is the worst case elapsed time to restore the system to a known uncorrupted version. The metric can be determined running a test to measure the actual time elapsed from the "start" of worst case compromise to a fully restored and 100% operational system. If it is impractical to perform that kind of a test on an operational system then this data should be

collected for actual security events if they have occurred. If a recovery test is not feasible, then a worst case recovery analysis may be used to estimate recovery time.
A suggested mathematical model:

T_0 = Start time (Time compromise is detected, or test start time)
T_r = Time at which recovery is complete and the system is 100% operational.
Restoration time = Maximum value of all instances of (T_r -T_0)

6 Evaluation of Thirteen Potential Metrics

We used case studies and analyzed security assessments to evaluate the thirteen potential metrics and to guide enhancements.

6.1 Case Studies

The potential set of proposed technical metrics was applied in two case studies of operating control systems. The purpose of these studies was to identify the practical limitations associated with data collection and to provide specific examples of how the metrics could be obtained.

6.1 1 Case Study 1

The first case study was of a Distributed Control System (DCS) for a chemical processing plant. Figure 1 is a simplified network diagram of the case study system.

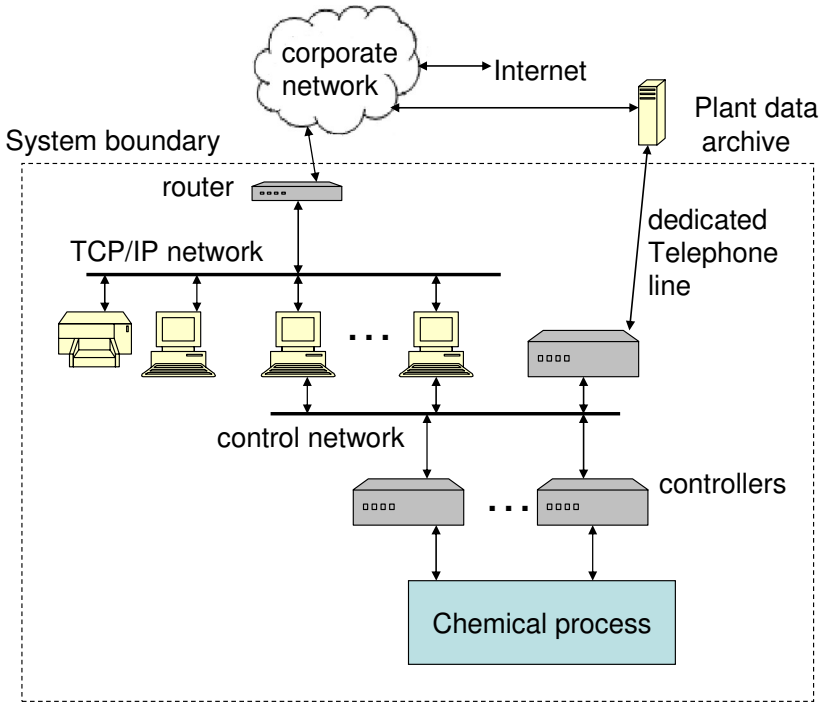

Figure. 1. Case Study 1 Control System Network Diagram

Notice that the system is connected to the Internet through the corporate network. The router that provides connectivity between the corporate network and the local TCP/IP network restricts access to the control system with an access-control-list so only the incoming TCP/IP connections with origination addresses that match the control list are allowed through the router. The system boundary is defined to be the processing plant and the control system networks that are within the control room. A dedicated telephone line connects the control system to the Plant data archive system which has direct connectivity to the corporate network. The corporate network affects the security of the control system but for this study the corporate network was not considered to be part of the system.

The DCS for this case study consists of a TCP/IP network that provides connectivity for 11 workstations and 2 printers, and a proprietary control network that provides connectivity to approximately 30 distributed controller nodes that control and monitor the plant. The workstations on the TCP/IP network consist of standard IT hardware, standard IT operating system software and application software supplied by the DCS vendor. The controller nodes consist of specialized control hardware and software supplied by the same DCS vendor. The metric values obtained for study 1 are shown in Table 3. Table 3 also shows the ideal value for each metric and the suggested target value. The suggested target value is the value we estimate could be obtained by changing the system configuration to improve security yet retain required functionality. This case study verified that the values of the metrics listed in Table 3 could be obtained using tools that are readily available.

Table 3. Case Study 1 Metrics Values

Metric Name	**Metric Value**	**Ideal target value**	**Suggested target value**
Rogue Change Days	0	0	0
Password Crack Time	> 30 days	∞	>30 days
Data Transmission Exposure	23	0	1
Reachability Count (N_T)*	164	0	1
Physical (N_p)	2	0	0
Services (N_S)	149	0	1
Outgoing (N_o)	13	0	0
Root Privilege Count	3	0	1
Attack Path Depth	2	∞	4
Worst Case Loss	$100M	$0	unknown
Detection Mechanism Deficiency Count	12	0	0
Known Vulnerability Days (high priority)	16,416 vuln. days	0	0
Known Vulnerability Days (low priority)	15,877 vuln. days	0	0
Restoration Time	120 minutes	0	120 minutes

*$N_T = N_p + N_S + N_o$

Metrics that were changed or refined following Case Study 1:

1. **Data Transmission Exposure** was originally defined to be simply the number of

unencrypted bytes transmitted per day between the control system and the point of measurement. This was found to be impractical and of questionable value. We concluded that a better definition is a count of the number of unencrypted machine communication channel pairs in use. For our case study (a TCP/IP network), it is the number of unencrypted machine TCP-port pairs in use (as observable by network monitoring).

2. **Reachability Count** includes a count of "outgoing network connectivity (N_o)". N_o was originally defined to be the number of machine-port pairs that have network connectivity from inside the network to the point of origin, where connectivity means the network configuration allows the machine to originate two-way connection-oriented sessions to some facility located at the point of origin. We found that by this definition, machines with TCP/IP connectivity and with no outgoing firewall restrictions would have very large values for this metric (there are 65535 possible TCP ports) that was not representative of the risk. Therefore N_o was redefined to be the number of active physical network ports with outgoing network connectivity from a control system machine to the point of origin.
3. **Root Privilege Count** was included in the initial proposed metrics for ideal 3 (The control system is inaccessible to AGs) because it is a simple measure of least privilege. It was originally defined to be the number of accounts that have administrative privilege. The case study showed that there are many machines that all have identical accounts. We decided to count all those duplicate accounts as a single account.
4. **Attack Surface** was deleted. The metric has the potential to be a valuable measure of vulnerability but needs more research and tools before it is practical.
5. **Detection Performance** was deleted. This metric is important and it is theoretically measurable. But currently the tools do not exist that would make this metric practical.

Other lessons learned from case study 1:

1. The **Component Test Count** metric was found to be difficult to measure. An unresolved question is: Do tests become obsolete with the passage of time or when there is a new version of the component? If so, then how do you determine when the tests are obsolete? It became clear during the case study that we did not know how to identify the components that should constitute a complete control system (How are software components to be counted? What level of decomposition of hardware is appropriate? etc.). We did not find any components that had clearly undergone independent testing. Therefore, the value for that metric could not be obtained. We concluded the metric should be better defined or discarded.
2. The scanning of machines that are part of a live control system may not be allowed because of the potential impact on operations. Scanning to identify open ports and vulnerabilities can cause some machines to crash. Fortunately, it may be viable to scan machines that are temporarily disconnected from the control system network or were setup in a laboratory environment and representative of the live machines. Then the results may be extrapolated to include counts for the whole system.
3. The collecting of security related information is very sensitive. To develop a complete picture requires the cooperation of the control system engineers and the IT operations personnel.
4. An important observation of this study was that all security improvements that were recommended by security experts and by following recommended practices [DHS07] correspond to improvements in the values of one or more of the proposed metrics. That implies that the metrics are, indeed, strongly correlated to security.

6.1.2 Case Study 2

Case Study 2 was for a power distribution Supervisory Control and Data Acquisition (SCADA) system. Figure 2 is a simplified network diagram of the system that shows the main components and connectivity. The SCADA controls seven power distribution substations. There is one PLC and one Voice over Internet Protocol (VoIP) phone physically located at each of the seven substations. There are 25 electric power meters that are connected directly to the network. One Engineering Workstation (EWS), two Front End Processors (FEP), three HMI hosts,, one printer and the SCADA firewall are physically located in the main substation control room. All devices are logically connected to a single TCP/IP network through routers and switches. The only connection from the SCADA network to the external network is through the SCADA firewall.

Data were collected by testing on a duplicate laboratory system when feasible rather than the live SCADA system to avoid potential disruption of operations. For the cases where the laboratory system did not provide sufficient information, data was also collected on the live operating SCADA system.

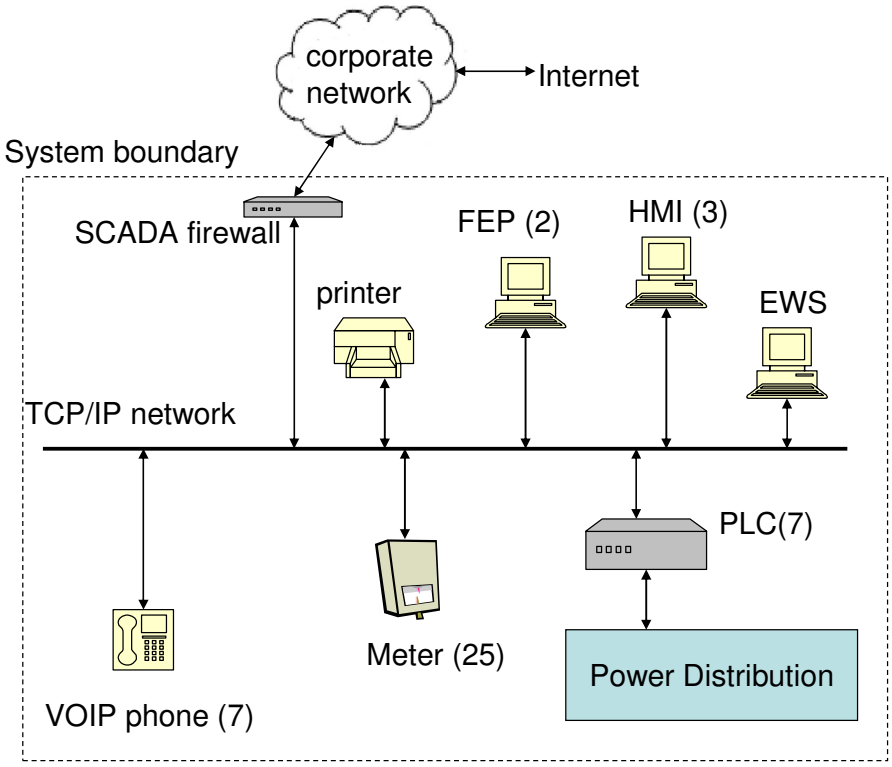

Figure 2. Case Study 2, SCADA Network Diagram.

For this case study, all the passwords were cracked in one day using John the Ripper [JTR06]. The password hashes were LAN Manager hashes which are known to be easy to crack. The metric values obtained for Case Study 2 are shown in Table 4.

Table 4. Case Study 2 Metrics Values

Metric Name	**Metric Value**	**Ideal target value**	**Suggested target value**	**Comment**
Rogue Change Days	unknown	0	0	Mechanisms are needed to detect rogue changes
Password Crack Time	0 days	∞	30+ days	Suggest changing the passwords more often. Password age is 4 years.
Data Transmission Exposure (Internet)	0	0	0	
Data Transmission Exposure (Intranet)	28	0	0	Avoid using unencrypted protocols through the firewall.
1. Reachability Count (external) (N_T)*	0	0	0	External reachability count total
1. Physical (N_p)	0	0	0	No physical access outside the control room
1. Services (N_S)	0	0	0	External connectivity only through a VPN
1. Outgoing (N_o)	0	0	0	firewall prevents outgoing connections
2. Reachability Count (local) (N_T)*	91	0	1	Local Reachability count total. From within control room and/or from VPN
2. Physical (N_p)	4	0	0	HMI and EWS have unrestricted physical access USB ports inside the control room.
2. Services (N_S)	87	0	1	Suggest further firewall rules to restrict VPN access.
2. Outgoing (N_o)	0	0	0	firewall prevents outgoing connections
Worst Case Loss	unknown	$0	unknown	Worst case power outage from a cyber attack is estimated to be 6 hours duration. The dollar value of such an outage has not been estimated.
Detection Mechanism Deficiency Count	46	0	0	There are no detection mechanisms on the system. The addition of AV software and firewall restrictions can reduce metric value to zero.
Vulnerability Exposure (high priority)†	36,318 vuln. days	0	0	Recommend the operating system be patched to current level. The PLC vulnerability has no available fix but PLC reachability improvement could mitigate the vulnerability.
Vulnerability Exposure (low priority) †	18,624 vuln. days	0	0	All known vulnerabilities should be mitigated if feasible.
Attack Path Depth	1	∞	2	Suggest further firewall rules to restrict VPN access to increase attack path depth.
Restoration Time	72 hours	0	5 hours	Worst case restoration time is high because it requires a complete software rebuild.

*$N_T = N_p + N_S + N_o$

†High priority vulnerabilities allow penetration and/or CVSS score is greater than 7.

Table 4 also shows the ideal value for each metric and the suggested target value. The suggested target value is the value we estimate could be obtained by changing the system configuration as suggested by the comments in the table.

Metrics that were changed or refined as a result of Case Study 2:

- **Root Privilege Count** was discarded. This metric was included in the initial proposed metrics for ideal 3 (The control system is inaccessible to AGs). It was originally assumed to be easy to measure. The question of whether to count similar accounts on multiple machines was raised again as it was during Case Study 1. Since the appropriate definition for this metric and its correlation to security became clouded, it was discarded.
- **Component Test Count** was replaced. The metric was again found to be difficult to measure. Therefore, the metric was replaced by a new metric: **Security Evaluation Deficiency Count** described below in section 6.1.3.

Some other lessons learned from Case Study 2 are:

1. The System had a mechanism for detecting one type of rogue change, (CISCO switches have the "port security" feature enabled which means that if the MAC address of the connected device doesn't match the registered value for that port, then the port is disabled). But the **Rogue Change Days** metric value could not be determined because of the lack of any other data. We believe most systems do not currently employ tools that compare the current configuration to the baseline configuration. However, there are tools available (for example, Tripwire[TW06]) that can provide this functionality. Therefore, we believe the **Rogue Change Days** metric should remain on the list of core/recommended metrics.
2. As was observed in Case Study 1, all security improvements that were recommended by security experts and by following recommended practices [DHS07] correspond to improvements in the values of one or more of the proposed metrics. This provides further evidence that the metrics are, indeed, strongly correlated to security.

6.1.3 Case Study Results- New Core Metrics

The two case studies resulted in the refinement of our proposed metrics. The resulting core metrics are summarized in Table 5 with corresponding ideals. The definitions of some metrics were changed to make them more meaningful and more practical. Three metrics were deleted (Root privilege count, Attack Surface, Detection Performance) and the **Component Test Count** metric was replaced by **Security Evaluation Deficiency Count.**

Security Evaluation Deficiency Count is the number of control system network devices that have not undergone a security evaluation. A key assertion is that security evaluation of the network components identifies vulnerabilities and leads to improved security of the control system.

Table 5. Set of Ten Core Technical Security Metrics with Corresponding Ideal.

Security Ideal	Metric
1. Security Group (SG) knows current control system perfectly	Rogue Change Days
	Security Evaluation Deficiency Count
2. Attack Group (AG) knows nothing about the control system	Data Transmission Exposure
3. The control system is inaccessible to AGs	Reachability Count
	Attack Path Depth
4. The control system has no vulnerabilities	Known Vulnerability Days
	Password Crack Time
5. The control system cannot cause damage	Worst Case Loss
6. SG detects any attack instantly	Detection Mechanism Deficiency Count
7. SG can restore control system integrity instantly	Restoration Time

6.2 Validation of Ideals and Metrics Using Security Assessments

To further validate the security ideals, the modifications made to the original thirteen potential metrics and the final proposed set of ten core metrics, seven separate control systems security assessments were reviewed for findings and recommended mitigations. These findings and mitigations were then mapped to the security ideals, to the thirteen potential metrics, and to the final proposed set of ten core metrics to identify gaps and weaknesses in their coverage.

The seven control system security assessments were performed by several cyber security assessors, external from the authors, assessing control systems considered representative of those frequently deployed in industry. These seven assessments produced 217 findings and 199 recommended mitigations. When possible, assessment findings were paired with a corresponding recommended mitigation which resulted in 225 ***evaluation cases*** (several findings and recommendations lacked counterparts).

Each evaluation case was then mapped to one ideal and one metric when it was possible. Each attempted mapping to a metric was characterized as a "strong map," "weak map," or "unmapped."

A "strong map" indicated that the evaluation case would be included in the metric calculation according to the previously established metric description.

A "weak map" indicated that the evaluation case would probably be included in the metric calculation according to the metric description but the associated mitigation might move the metric in the wrong direction, or that while the mitigation would move the metric in the correct direction the underlying security issue would not change.

An "unmapped" characterization indicated that none of the proposed metrics addressed the evaluation case.

6.2.1 Analysis of Evaluation Case Mapping to Ideals

The first significant finding was that all evaluation cases mapped well to the proposed security ideals. This indicates that the set of security ideals accounted for every finding and recommendation made by the cyber security assessors. See Table 6.

Table 6. Evaluation Case Mapping to Ideals

Ideal	Number of Evaluation Cases Mapped	Percent of All Evaluation Cases
1	7	3%
2	41	18%
3	80	36%
4	82	36%
5	7	3%
6	7	3%
7	1	0%
Total	225	100%

It is notable that 90% of the cases mapped to ideals 2, 3, and 4. This is not surprising as cyber security assessments frequently relate to an attack group learning about, gaining access to, and exploiting vulnerabilities in a control system – which are the concepts addressed in these ideals.

6.2.2 Analysis of Evaluation Case Mappings to Metrics

To assess the validity of the technical metrics each evaluation case which had been mapped to an ideal was further mapped to one of the metrics for that ideal. This was done once for the set of thirteen potential metrics and again for the set of ten core metrics. The mapping results were then used to assess whether the potential set of thirteen metrics encompassed most of the evaluation cases, whether the metrics for each ideal provided reasonable coverage for that ideal, and whether the set of ten core metrics provided a similar level of coverage.

6.2.2.1 Mapped versus Unmapped Counts

After mapping evaluation cases to the potential set of thirteen metrics, analysis determined that 90% of the evaluation case mappings were categorized as “strong map” or “weak map”. Thus only the remaining 10% of the evaluation cases were “unmapped”. See Table 7.

Table 7. Mapped and Unmapped Evaluation Cases to Thirteen Potential Metrics

	Number	Percent
Strong or weakly mapped cases	203	90%
Unmapped cases	22	10%
Total	225	100%

The evaluation cases were then mapped to the set of ten core technical metrics. The results are found in Table 8 and indicate that despite removing three of the thirteen metrics fully 87% of the evaluation cases are still categorized as a strong or weak mapping and there has been only a slight increase (3%) in the number of evaluation cases that couldn’t effectively be mapped to some metric.

Table 8. Mapped and Unmapped Evaluation Cases to Ten Core Metrics

	Number	Percent
Strong or weakly mapped cases	196	87%
Unmapped cases	29	13%
Total	225	100%

6.2.2.2 Mapped versus Unmapped Counts for Each Ideal's Metrics

The evaluation case mappings to the thirteen potential metrics were examined for insight into how well the metric(s) for each ideal covered the evaluation cases mapped to that ideal. Based on the results found in Table 9, the metrics for five of the seven ideals resulted in over 90% coverage. Interestingly, note that of the total number of unmapped cases, a majority reside in ideal 2 (Attack group knows nothing about the control system). Unfortunately, practical metrics for ideal 2 are difficult to identify; this observation highlights the need for future improvement. One other noteworthy result from Table 9 is that 71% of evaluation cases mapped to ideal 5 were unable to be mapped to a corresponding metric. Investigation of this fact showed that assessors did not seek worst case loss information, and that there is no metric to cover losses that may not be included in calculation of the worst case loss metric. More detailed analysis of unmapped evaluation cases can be found in [INL07].

Table 9. Mapped and Unmapped Evaluation Cases for Each Ideal Using the Thirteen Potential Metrics

	Number			As Percent of Total	
Ideal	Total evaluation cases	Strongly or weakly mapped	Unmapped	Strongly or weakly mapped	Unmapped
1	7	7	0	100%	0%
2	41	28	13	68%	32%
3	80	79	1	99%	1%
4	82	79	3	96%	4%
5	7	2	5	29%	71%
6	7	7	0	100%	0%
7	1	1	0	100%	0%
Total	225	203	22	90%	10%

The evaluation case mappings to the ten core metrics were then examined in a similar fashion as above. The results may be found in Table 10. The most unfortunate aspect of the mapping of evaluation cases to the core metrics is that ideal 6 (Security group detects any attack instantly) metrics now only have 29% coverage of the ideal. Consequently, consideration was given to reintroducing the Detection Performance metric for ideal 6 but was decided against since there is currently no accepted and credible method for assessing the performance of many of the possible detection mechanisms.

Table 10. Mapped and Unmapped Evaluation Cases for Each Ideal Using the Ten Core Metrics

	Number			As Percent of Total	
Ideal	Total evaluation cases	Strongly or weakly mapped	Unmapped	Strongly or weakly mapped	Unmapped
1	9	9	0	100%	0%
2	41	28	13	68%	32%
3	80	77	3	96%	4%
4	80	77	3	96%	4%
5	7	2	5	29%	71%
6	7	2	5	29%	71%
7	1	1	0	100%	0%
Total	225	196	29	87%	13%

6.2.2.3 Counts of Mapped Evaluation Cases for Each Metric

A further decomposition of the evaluation case mappings, shown in Table 11, examined the distribution of mappings across each of the thirteen metrics. The results indicate that two metrics, reachability count and known vulnerability days, capture a majority (67%) of mapped findings. It is not clear whether this indicates the overall importance of those two metrics or that it indicates a bias in the assessment processes. It is also useful to note that every metric had at least one mapping to it.

Table 11. Mapped and Unmapped Evaluation Cases for Each of the Thirteen Potential Metrics

Ideal	Metric	Strongly or weakly mapped	Percent
1	rogue change days	3	1%
1	component test count	4	2%
2	data transmission exposure	28	14%
3	reachability count	74	36%
3	root privilege count	2	1%
3	attack path depth	3	1%
4	known vulnerability days	62	31%
4	attack surface	2	1%
4	password crack time	15	7%
5	worst case loss	2	1%
6	detection mechanism deficiency count	2	1%
6	detection performance	5	2%
7	restoration time	1	0%
	Total strongly or weakly mapped cases	203	100%

As expected, the distribution of mappings across metrics changed very little when the thirteen potential metrics were reduced to the ten core metrics. While previously capturing 67% of the

mappings the two predominant metrics, reachability count and known vulnerability days, now capture 70%. See Table 12.

Table 12. Mapped and Unmapped Evaluation Cases for Each of the Ten Core Metrics

Ideal	Metric	Strongly or weakly mapped	Percent
1	rogue change days	3	2%
1	security evaluation deficiency count	6	3%
2	data transmission exposure	28	14%
3	reachability count	74	38%
3	attack path depth	3	2%
4	known vulnerability days	62	32%
4	password crack time	15	8%
5	worst case loss	2	1%
6	detection mechanism deficiency count	2	1%
7	restoration time	1	1%
	Total matched cases	196	100%

6.2.2.4 Strong versus Weakly Mapped Counts for Each Metric

An examination of the mappings of evaluation cases to each of the thirteen potential metrics, Table 13, showed that mappings were characterized as strong for 86% of the evaluation cases. Rogue change days mappings were characterized as weakly mapped for a third of evaluation cases. This was due to the fact that logging a change does not necessarily imply alerting the security group of that change. A competent security group is expected to recognize this limitation in applying this metric. More detailed analysis of the mappings can be found in [INL07].

Table 13. Mapped and Unmapped Evaluation Cases for Each of the Thirteen Potential Metrics

		Number			As percent of Total	
Ideal	Metric	Strongly or weakly mapped	Strongly mapped	Weakly mapped	Strongly mapped	Weakly mapped
1	rogue change days	3	2	1	67%	33%
1	component test count	4	0	4	0%	100%
2	data transmission exposure	28	24	4	86%	14%
3	reachability count	74	69	5	93%	7%
3	root privilege count	2	0	2	0%	100%
3	attack path depth	3	3	0	100%	0%
4	known vulnerability days	62	54	8	87%	13%
4	attack surface	2	0	2	0%	100%
4	password crack time	15	15	0	100%	0%
5	worst case loss	2	0	2	0%	100%
6	detection mechanism deficiency count	2	2	0	100%	0%
6	detection performance	5	5	0	100%	0%
7	restoration time	1	0	1	0%	100%
	Total cases strongly or weakly mapped	203	174	29	86%	14%

After removing the root privilege count, attack surface, and detection performance metrics from the potential set of thirteen metrics the overall strength of metric mappings decreased slightly - falling from 86% to 84%, as shown in Table 14. The only other notable change for the mapping to the ten core metrics is the increase of two strongly mapped evaluation cases that resulted from the replacement of the **Component Test Count** metric by the **Security Evaluation Deficiency Count** metric.

Table 14. Mapped and Unmapped Evaluation Cases for Each of the Ten Core Metrics

		Number			As percent of Total	
Ideal	Metric	Strongly or weakly mapped	Strongly mapped	Weakly mapped	Strongly mapped	Weakly mapped
1	rogue change days	3	2	1	67%	33%
1	security evaluation deficiency count	6	2	4	33%	67%
2	data transmission exposure	28	24	4	86%	14%
3	reachability count	74	69	5	93%	7%
3	attack path depth	3	3	0	100%	0%
4	known vulnerability days	62	47	15	76%	24%
4	password crack time	15	15	0	100%	0%
5	worst case loss	2	0	2	0%	100%
6	detection mechanism deficiency count	2	2	0	100%	0%
7	restoration time	1	0	1	0%	100%
	Total cases strongly or weakly mapped	196	164	32	84%	16%

6.2.2.3 Summary of Evaluation Case Mappings to Metrics Results

The evaluation cases were mapped to the potential set of thirteen metrics and then to the set of ten core metrics. The mapping results show that the set of ten core metrics does not provide as much coverage as the potential thirteen metrics yet still captures 87% of all security assessment findings and recommended mitigations. As a result, the ten core metrics have been shown to capture a significant portion of security issues found in the seven control system assessments.

7 Conclusions

A set of seven security ideals [Table 1] were developed to guide the creation of a small set of technical metrics to aid in the measurement of control system security. Through the application of the ideals and proposed metrics in two control system security case studies it was demonstrated that the ideals provided a useful framework for thinking about security and that the final proposed set of technical metrics [Table 5] provided an excellent but somewhat incomplete security snapshot.

Seven control system security assessments were also reviewed to aid in the identification of gaps in either the security ideals or the proposed technical metrics. From these reviews it was discovered that all of the security findings and recommended mitigations could be clearly mapped to an ideal. Consequently, no additions or deletions to the security ideals were needed. Further, it was discovered that approximately 87% of the assessment security findings and mitigations could be mapped to one of the technical metrics.

While we are aware that there is no known scientifically defensible method for measuring security, based on this work we assert that it is possible to measure attributes of a control system that are correlated with standard security issues. Given the results of the case studies and review of the assessments we conclude that the proposed set of seven security ideals provide a strong framework for thinking about control system security and that the final proposed set of ten core technical security metrics provide control system owners/operators with valuable insight for the management of their control system security.

8 Future Work

The measuring of security is an extremely difficult problem partly because the technology is complex and because security is aimed at protecting against an unpredictable intelligent adversary. The value of the proposed security ideals and technical metrics now need to be applied over time in an industrial setting and correlated with actual attacks on the associated control systems.

Further, the development of these ideals and metrics has helped to show the need for more control system security research into security models and measurement tools. The research should include the development of more technical metrics that would provide either greater coverage of security issues or improved correlation to security. Measures we are considering for investigation include the extension of attack surface concepts to control system and facility level security, improved measures and models of detection performance, and the value of various security testing processes.

9 References

[Bis03] Bishop, M., *Computer Security Art and Science*, Addison Wesley, pp. 343-349, 2003.

[BM07] Boyer, W.F, McQueen, M.A., "Ideal Based Cyber Security Technical Metrics for Control Systems", CRITIS'07 2nd International Workshop on Critical Information Infrastructures Security, October 3-5, 2007.

[CCH06] Chew, E., Clay, A., Hash, J., Bartol, N., Brown, A., Guide for Developing Performance Metrics for Information Security, NIST Special Publication 800-80, May 2006.

[CSC06] Chemical Sector Cyber Security Program (CSCSP), Guidance for Addressing Cyber Security in the Chemical Industry, Technical Report, CSCSP, May 2006.

[DHS07] U.S. Department of Homeland Security, Control System Security Program Recommended Practices, http://csrp.inl.gov/Recommended_Practices_Working_Group.html

[INL06] INL Report to the Department of Homeland Security, INL/EXT-06-12016, Cyber Security Metrics, December 2006.

[INL07] INL Report to the Department of Homeland Security, INL/EXT-07-13562, Control System Technical Security Metrics Report, December 2007.

[JTR06] John the Ripper, (http://www.openwall.com)

[Jac07] Jacquith, A., Security Metrics, Addison Wesley, 2007.

[MBF05] McQueen, M. A., W. F. Boyer, M. A. Flynn, G. A. Beitel, "Time-to-compromise Model for Cyber Risk Reduction Estimation", First Workshop on Quality of Protection, Sept. 2005.

[MBF06] McQueen, M. A., W. F. Boyer, M. A. Flynn, G. A. Beitel, "Quantitative Cyber Risk Reduction Estimation Methodology for a Small SCADA Control System", Proceedings of the 39th Hawaii International Conference on System Sciences, pp. 226, Jan. 2006.

[Mel03] Mell P, V Hu, R Lippmann, J Haines, and M Zissman, *An Overview of Issues in Testing Intrusion Detection Systems*, http://csrc.nist.gov/publications/nistir/nistir-7007.pdf, Interagency Report (IR) 7007, National Institute of Standards and Technology, Gaithersburg, Maryland, June 2003, Web site visited September 19, 2006.

[MW05] Manadhata, P., Wing, J. M., An Attack Surface Metric, Technical Report CMU-CS-05-155, July 2005

[Neu95] Neumann, P. G., *Computer Related Risks*, Addison Wesley, pp. 244, 1995.

[RKJ06] Ross, R., S. Katzke, A. Johnson, M. Swanson & G. Rogers, System Questionnaire with NIST SP 800-53: Recommended Security Controls for Federal Information Systems, Technical Report, NIST, References and Associated Security Control Mappings, Gaithersburg, Maryland, March 2006,

[SBS03] Swanson, M., N. Bartol, J. Sabato, J. Hash & L. Graffo, NIST Special Publication 800-55: Security Metrics Guide for Information Technology Systems, Technical Report, National Institute of Standards and Technology (NIST), Gaithersburg, Maryland, July 2003

[Sch00] Schneier, B., *Secrets & Lies*, Wiley, pp. 367-380, 2000.

[Sch05] Schiffman, M., A Complete Guide to the Common Vulnerability Scoring System (CVSS), Technical Report, Forum for Incident Response and Security Teams (FIRST), June 7, 2005.

[SG96] Swanson, M., Guttman, B., *Generally Accepted Principles and Practices for Securing Information Technology Systems, NIST 800-14*, September 1996.

[Sum97] Summers, R. C., *Secure Computing Threats and Safeguards*, McGraw Hill pp. 251-252, 1997.

[TW06] Tripwire, http://www.tripwire.com/products/index.cfm

[XBM06] Ou, X., Boyer, W., McQueen, M., A Scalable approach to Attack Graph Generation, 13th ACM Conference on Computer and Communications Security, CCS'06, October 30 through November 3, 2006.