Automatic Security Analysis Using Security Metrics

Kun Sun, Sushil Jajodia
Center for Secure Information Systems
George Mason University
Email: {ksun3, jajodia}@gmu.edu

Jason Li, Yi Cheng, Wei Tang
Intelligent Automation Inc.
Rockville, MD
Email: {jli,ycheng,wtang}@i-a-i.com

Anoop Singhal
Computer Security Division
NIST
Email: anoop.singhal@nist.gov

***Abstract*—Security metrics are valuable for measuring and comparing the amount of security provided by different systems and configurations. However, meaningful security metrics for networked systems are significantly difficult to define, evaluate, interpret, and visualize. We design a system that provides security metrics collection, security metrics management, and security metrics visualization for scalable and automatic security analysis. We first identify a set of new security metrics. Then, we show how to collect simple security metrics from the computers in a sample network. Next, we use Analytic Hierarchy Process (AHP) mechanism to compose two sophisticated security metrics, *Criticality* and *Security Score*, which are critical to measure the security risk. We also develop visualization tools to help administrators better understand and evaluate the system security using security metrics.**

I. Introduction

Our society has become increasingly dependent on the reliability and proper functioning of a vast number of interconnected information systems. To improve the security of interconnected information systems, it is critical to measure the amount of security provided by different systems and configurations. The study of security metrics has recently drawn significant attention. There are a number of standardized enumerations (e.g., [1], [2], [4], [12], [13]) to help measurement in the cyber security.

However, metrics for security have been much less satisfactory. Most of the existing metrics are of questionable utility. Meaningful security metrics for networked systems are significantly more difficult to define, evaluate, interpret, and use intelligently. The following requirements should be met before security metrics can have greater utility.

- *Defining meaningful metrics*. Metrics are generated from analysis, so they may be either objective or subjective human interpretations of raw data. Moreover, the assumptions for security metrics may be poorly founded or changing with time.
- *Collecting effective measurements.* Metrics are derived from measurements. We need to know what measurements should be collected to derive the metrics. For example, administrators need to know the network reachability information to evaluate the security of a network.
- *Composing sophisticated metrics.* Security metrics can cover from finer-grained (e.g., single-system) metrics through hierarchical composition of metrics. Composability of sophisticated metrics is a difficult problem, considering the unpredictable composition compatibility and interoperability among networked systems.
- *Showing analysis results.* In many cases, presentation of security metrics is as important as the data content. Visual representations can dramatically enhance administrators' abilities to understand security issues.

This work was supported by the NIST SBIR award number SB1341-09-SE-0626. The work of Sushil Jajodia was supported in part by the Army Research Office MURI award number W911NF-09-1-0525.

In this paper, we present our work on developing a scalable and automatic security analysis system that provides security metrics collection, security metrics management, and security metrics visualization components. We implement a prototype of the system, which can help administrators better understand the security of the computer and networks. In summary, our major contributions include:

1) *Identify a set of new security metrics.* After studying the well defined security metrics in the literature (e.g.,[5], [6]), we identify a number of new security metrics such as Patch Risk, Criticality, Security Score, and Time Series, and discuss the importance of each metrics.
2) *Collect security metrics from system and network measurements.* Our system can automatically collect simple security metrics, which can be used to compose more sophisticated security metrics. For example, we use Nessus vulnerability scanner [3] to scan the vulnerabilities on each computer, and use Common Vulnerability Scoring System (CVSS) [4] to obtain the security vector for each vulnerability. Moreover, we can automatically obtain network reachability information from network router configuration files.
3) *Compose sophisticated security metrics.* We use Analytic Hierarchy Process (AHP) [8] mechanism to compose two sophisticated security metrics, Criticality and Security Score. Criticality is to evaluate the importance of one computer in the network. Security Score provides an explicit number to evaluate the security of a computer or a network.
4) *Visualize security metrics.* We develop a GUI interface that allows users query the security of a computer, a subnet, or a network. Besides the traditional visualization graphs like Scatter, Pie, Ring, Bars, Histogram, and Quartile, our tool provides what-if analysis and time series plots to visualize the security changes of the computers and networks.

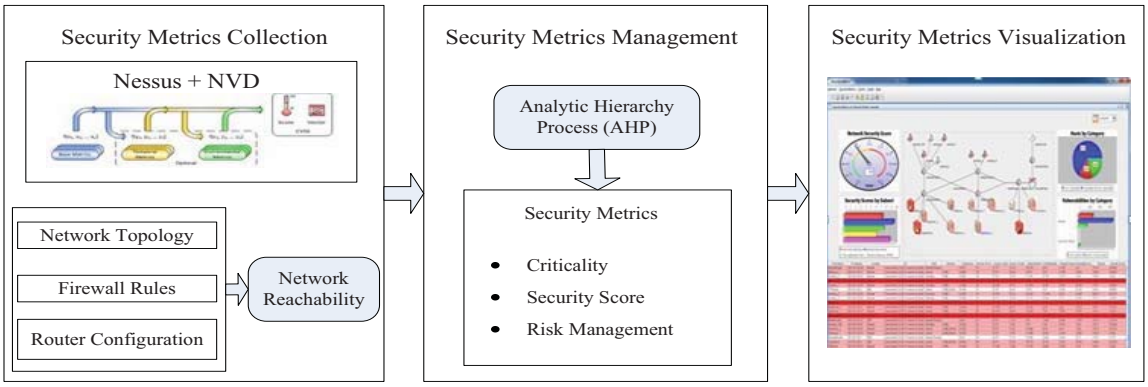

Fig. 1. System Architecture.

The rest of this paper is organized as follows. Section 2 presents the system architecture. We introduce several new security metrics in Section 3. We present our work on automatically collecting simple security metrics in Section 4. We present the security metrics composition using AHP in Section 5. In Section6, we describe the security metrics visualization. We discuss the related work in Section 7. Section 8 concludes this paper.

II. System Architecture

Our system architecture is shown in Figure 1. It consists of three major components: *security metrics collection*, *security metrics management*, and *security metrics visualization*. Security metrics collection focuses on collecting operational security metrics, such as vulnerability scanning from Nessus scanner, security score from NVD/CVSS, and network reachability information. We use Nessus vulnerability scanner [3] to perform real-time vulnerability scanning and analysis on a network. Then, we use Common Vulnerability Scoring System (CVSS) [4] to obtain more detailed information about each vulnerability. Network reachability information is critical to compose network-level security metrics. It can be collected from network topology, firewall rules, and router configuration.

Security metrics management targets at deriving more sophisticated security metrics based on the simple security metrics. We use Analytic Hierarchy Process (AHP) [8][9] to derive two sophisticated security metrics, Criticality and Security Score. AHP is a structured technique for dealing with complex decisions. It is useful for security analysis because it adopts the subjective opinions from Subject Matter Experts according to specific security requirements of each organization into mathematical calculation.

The security visualization component helps administrators better understand the security status of the computers and the network. The visualization tool can show security analysis results in various graphs and dashboards according to users requirements and display settings. It allows users enter various queries to check the security of individual computer or a network.

In this work, we implement a system prototype to help administrators evaluate network security and prioritize incident response actions. In the following, we will introduce the details for each component in the system.

III. Security Metrics Identification

We summarize a number of security metrics well defined in the literature (e.g.[5], [6]), and identify several new security metrics. Due to the space limitation, we will only present the new security metrics.

A. Patch Risk

Patch Risk is the risk to apply patches for fixing vulnerabilities in applications. Deploying a patch can have an adverse effect on critical applications. When an operating system is patched, some software may or may not function properly from that point forward. Moreover, some patches themselves require patching. For example, a security update released by Microsoft in January 2001 to patch a security exploit in Exchange 2000 server actually required a patching of its own [7]. Therefore, patching requires certain analysis, and there is a patch risk associated with patch management. The patch risk of a patch can be derived according to the trustworthiness of its provider and how long the patch has been released and verified.

B. Time Series

Time series metric shows the changes of security of a computer or a network in a period of time. This metric can tell users whether the security of a computer is improved, or falls below a pre-determined threshold. CVSS [4] provides Temporal Metrics to reflect that the threat posed by a vulnerability may change over time. However, our time series metric not only includes the Temporal Metrics in CVSS, but also includes the changes of the security score on a computer after adding new Zero-day vulnerabilities or fixing existing vulnerabilities.

C. Criticality

Criticality is a combined metric to evaluate the importance of one computer in the network. It can be derived from four other security metrics: (1) location of the computer in the network (e.g., intranet, Internet, or DMZ), (2) services and applications running on a computer, (3) role of the computer (e.g., server, router, firewall, or desktop), and (4) asset value on the computer. For example, when we consider the location

TABLE I
VSDT FOR CVE-2009-0022

Entry Identifier	CVE-2009-0022
Score	6.3
Severity	Medium
Vector	(AV:N/AC:M/Au:S/C:C/I:N/A:N)
Vuln_Types	Conf
Range	Network
Vuln_Soft	Samba 3.2.0, 3.2.1, 3.2.2, 3.2.3, 3.2.4, 3.2.5, 3.2.6
Description	When registry shares are enabled, allows remote authenticated users to access the root filesystem via a crafted connection request that specifies a blank share name.

metrics, one compromised http server in DMZ has less security impact than another compromised HTTP server located in the intranet.

D. Security Score

Security score provides an explicit number to evaluate the security of a computer or a network. It can help administrator compare the security between two computers or two networks. The security score for an individual vulnerability can be directly obtained using the CVSS score of the vulnerability in the CVSS database [4]. However, when the same computer is used to accomplish different missions, users may give different weights on vulnerability's impacts on authentication, integrity, and availability. Such a one-shot score may not be useful or meaningful for evaluating vulnerabilities in mission-awareness situations.

Furthmore, we could derive a security score for a computer by considering CVSS scores of all the vulnerabilities on the computer. Similarly, we could calculate a security score for a network with the knowledge of the network reachability information and security scores for all the computers in the network. However, because of the interdependence among the vulnerabilities on different computers in one network, it is a challenge to derive a meaningful security score for a network. We will study it in the future work.

IV. Security Metrics Collection

Our system can automatically collect meaningful measurement data from three resources: (1) NVD/CVSS database, (2) Nessus scanner, and (3) router configuration files. We obtain vulnerabilities information on each computer by scanning the whole network using Nessus scanner and then finding the details of each vulnerability in NVD/CVSS database. We can derive the network reachability information from router configuration files.

A. NVD/CVSS

National Vulnerability Database (NVD) [2] is a comprehensive vulnerability database that integrates all publicly available U.S. Government vulnerability resources. It provides large quantities of Common Vulnerability Scoring System (CVSS) scores, which is an industry standard for assessing the severity of computer system security vulnerabilities [4]. CVSS is composed of three metric groups: Base, Temporal, and Environmental, each consisting of a set of metrics.

In this work, we download the most updated version of NVD/CVSS database from NIST website [4], which provides the XML database on CVSS score/vector and an XML schema. Then, we develop an XML parser to extract security metrics (e.g., Score, Vector) that are interested to administrators. We define a data structure named Vulnerability Scoring and Description Table (VSDT) to store the parsing results, which can reduce the database size and speed up the operations on deriving more complicated security metrics. A VSDT entry example for vulnerability record CVE-2009-0022 [10] is shown in Table I.

B. Vulnerability Scanning

The Nessus vulnerability scanner [3] is an active scanner that features high-speed vulnerability discovery, sensitive data discovery, and vulnerability analysis. We use Nessus scanner [3] to collect metrics about real vulnerabilities on computers. For each identified vulnerability on either Linux or Windows operating system, we find a patch or an update to fix each of them. Linux does not provide patches to vulnerabilities like Microsoft does. Instead, a new version of the package is created and the system updates itself by uninstall the old package and then installing the new package. Through this process, we collect the first-hand vulnerability information to be used in our toolkit demonstration and we are able to show security changes in our toolkit when we fix some vulnerabilities using the corresponding patches.

C. Network Reachability

Our system can derive network reachability information from popular OPNET network design software or router files from CISCO routers, and then automatically generates a network reachability file in XML. A segment of network reachability XML file is shown in Figure 2. It includes node entries and link entries. Node entries include hosts, routers or switches, and link entries are connections between the nodes. Each node has one or more interface entries, which contain one or more ports. The interface for an router may contain multiple routing rules.

V. Security Metrics Management

Simple security metrics (e.g. number of vulnerabilities on one computer, number of computers in a network) can be directly obtained from the measurement in a system, However, to get an in-depth understanding of the network security, more

```
<?xml version="1.0" encoding="UTF-8"?>
<netconfig>

<node icon="router" name="ExternalRouter" type="Router" x="0.9" y="0.4"
location="DMZ" OS="Windows XP">
  <intf IP="128.105.120.1" id="WebServers" mask="255.255.255.128">
    <accessGroup direction="in" id="100">
      <accessList action="permit" destIP="128.105.120.1" destMask="0.0.0.15"
id="100" port="0" protocol="tcp" srcIP="0.0.0.0" srcMask="255.255.255.255"/>
    </accessGroup>
  </intf>
</node>
... ...
<node icon="ftpserver" name="FTPServer" type="Host" x="0.75" y="0.6"
location="DMZ" OS="Windows XP">
  <intf IP="128.105.120.2">
    <port portnum="21" protocol="ftp"/>
    <port portnum="22" protocol="ssh"/>
  </intf>
</node>
... ...
<link destNode="InternalSwitch2" name="internal switch 1 to internal switch 2"
srcNode="InternalSwitch1"/>
<link destNode="MiddleSwitch22" name="middle switch 21 to middle switch 22"
srcNode="MiddleSwitch21"/>

</netconfig>
```

Fig. 2. Network Configuration XML File.

powerful analysis techniques are required to compose sophisticated security metrics such as security score and criticality. In the following, we first introduce a decision making technique, Analytic Hierarchy Process (AHP), and then illustrate how to apply AHP to compose two sophisticated security metrics: criticality and security score.

A. Analytic Hierarchy Process (AHP)

The Analytic Hierarchy Process (AHP) [8][9] is a structured technique for dealing with complex decisions. It is useful for security analysis because it combines the objectivity of mathematics and the subjectivity of psychology to evaluate information and make decisions [9].

Users of the AHP first decompose their decision problem into a hierarchy of more easily comprehended sub-problems, each of which can be analyzed independently. Once the hierarchy is built, some subject matter experts systematically evaluate its various elements by comparing them. In making the comparisons, they can use concrete data about the elements, or they can use their judgments about the elements' relative meaning and importance. It is the essence of the AHP that expert judgments, and not just the underlying information, can be used in performing the evaluations. Then, AHP converts these evaluations to numerical values that can be processed and compared over the entire range of the problem. A numerical weight is derived for each element of the hierarchy, allowing diverse elements to be compared to one another in a rational and consistent way.

B. Composing Criticality

Criticality evaluates the importance of one computer in the network. To derive the criticality value of a computer in a sample network, we first build an AHP architecture that contains two levels of criteria, as shown in Figure 3. Criticality can be evaluated by four first-level criteria: (1) *Service*, such as HTTP, FTP, SMTP, SSH, etc.; (2) *Location*, such as intranet, extranet, DMZ, etc.; (3)*Role*, such as desktop, server, router, Firewall, etc.; (4) *Asset*, such as important file or database on the host. Because our sample network consists of 21 hosts, the second-level criteria are the 21 hosts.

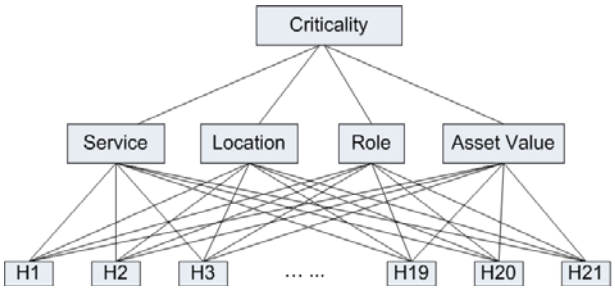

Fig. 3. AHP Architecture for Criticality.

TABLE II
PAIRED COMPARISON ON CRITICALITY

Criteria	Service	Location	Role	Asset
Serivce	1.0	0.5	0.5	0.3333
Location	2.0	1.0	1.0	0.6667
Role	2.0	1.0	1.0	0.6667
Asset	3.0	1.5	1.5	1.0

Subject Matter Experts use a paired comparison approach to rate the relative importance or preference for each criterion level by level, from top to bottom. Table II shows the ranking of the importance of each first-level criterion relative to the others. In this example, experts decide that location criterion and role criterion are both 2 times as important as service criterion; asset criterion is 3 times as important as the service criterion and 1.5 times as important as both location and role criteria.

Next, Subject Matter Experts use a paired comparison matrix to decide the proper weights for each criterion. Given four first-level criteria for criticality, we have a 4×4 matrix M. We get a normalized matrix $M^{'}$ by summing each column of matrix M and dividing each element of the matrix with the sum of its column. Finally, the weights for first-level criteria can be obtained by averaging across the rows, as shown in matrix M^{*}. The sum of weights of all criteria will always equal to 1.

$$M = \begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{2} & \frac{1}{3} \\ 2 & 1 & 1 & \frac{2}{3} \\ 2 & 1 & 1 & \frac{2}{3} \\ 3 & \frac{3}{2} & \frac{3}{2} & 1 \end{bmatrix} => M^{'} = \begin{bmatrix} \frac{1}{8} & \frac{1}{8} & \frac{1}{8} & \frac{1}{8} \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ \frac{3}{8} & \frac{3}{8} & \frac{3}{8} & \frac{3}{8} \end{bmatrix} =>$$

$$M^{*} = \frac{1}{4} \times \begin{bmatrix} 1+\frac{1}{2}+\frac{1}{2}+\frac{1}{3} \\ 2+1+1+\frac{2}{3} \\ 2+1+1+\frac{2}{3} \\ 3+\frac{3}{2}+\frac{3}{2}+1 \end{bmatrix} = \begin{bmatrix} 0.125 \\ 0.25 \\ 0.25 \\ 0.375 \end{bmatrix}$$

In this example, the weight for service is 0.125, the weights for both location and role are 0.25, and the weight for asset

is 0.375. Similarly, a paired comparison matrix of the second-level criteria can be decided for each first-level criteria. We will have a 21×21 matrix to show the ranking of *Asset* of each host relative to the others. Then we could calculate the weight of asset $W_i(asset)$ for host i. Similarly, we have three other paired comparison matrices for service, location and role separately, and calculate the weights $W_i(service)$, $W_i(location)$, and $W_i(role)$ for host i. The final criticalness weight C_i for $Host_i$ can be calculated as

$$C_i = 0.125 \times W_i(service) + 0.25 \times W_i(location) + 0.25 \times W_i(role) + 0.375 \times W_i(asset)$$

C. Composing Security Score

From CVSS database [4], we can obtain the based score and security vectors for individual vulnerability. However, when the same computer is used to accomplish different missions, users may give different weights on vulnerability's impacts on authentication, integrity, and availability. For example, if a company provides an online warehouse for information sharing, it may pay more attention to data availability than data confidentiality. Thus, it may give different weights for availability, integrity, and confidentiality, which are given the same weight in CVSS scoring system. Again, we can use Subject Matter Expert and Analytic Hierarchy Process (AHP) to decide the weights for availability, integrity, and authentication according to system requirements.

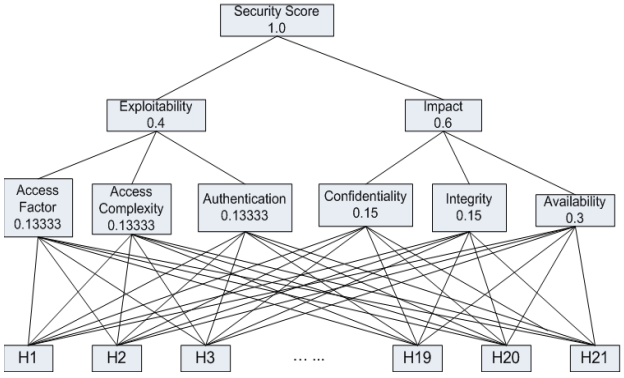

Fig. 4. AHP Hierarchy for Security Score.

The AHP architecture for security score of vulnerability is shown in Figure 4. The root is the weight for Security Score, which equals to 1. The first-level criteria are Exploitability and Impact. Table III shows the paired comparison for the first-level criteria on security score. In this example, the experts decide that impact criterion is 1.5 times as important as exploitability criterion. The second-level criteria for Exploitability are access vector, access complexity, and authentication. When we suppose they have equal weights, Table IV shows the corresponding paired comparison. The second-level criteria for Impact are confidentiality impact, integrity impact and availability impact. In this example, experts decide that availability criterion is 2 times as important as confidentiality criterion and integrity criterion. Table V shows the paired comparison matrix for the second-level criteria on Impact. The weights for the first and second levels criteria are shown in Figure 4. Due to space limitation, we omit the details of weight calculation for each host in the network.

TABLE III
PAIRED COMPARISON ON SECURITY SCORE

Criteria	Exploitability	Impact
Exploitability	1.0	0.6667
Impact	1.4999	1.0

TABLE IV
PAIRED COMPARISON ON EXPLOITABILITY

Criteria	Access Vector	Access Complexity	Authentication
Access Vector	1.0	1.0	1.0
Access Complexity	1.0	1.0	1.0
Authentication	1.0	1.0	1.0

TABLE V
PAIRED COMPARISON ON IMPACT

Criteria	Confidentiality	Integrity	Availability
Confidentiality	1.0	1.0	0.5
Integrity	1.0	1.0	0.5
Availability	2.0	2.0	1.0

VI. SECURITY METRICS VISUALIZATION

We develop a visualization tool in Java to help administrators better understand the security metrics. Besides the traditional visualization graphs like Scatter, Pie, Ring, Bars, Histogram, and Quartile, we include the time series plot and support what-if analysis in our visualization tool. We use a Java chart library called JFreeChart [11] to draw all the charts.

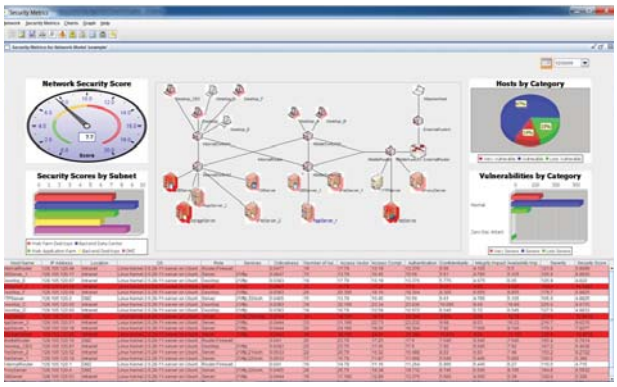

Fig. 5. Main Dashboard.

Dashboard is a powerful tool to show the results of an integrated system that monitored internal and external threats, vulnerabilities, and asset criticality, etc. The main dashboard of our security metrics system is shown in Figure 5. It has five menus. Network menu contains submenus to import network topology from popular OPNET network design software and CISCO route files, open/close network model and browse

interactive network topology graph. Security Metrics menu includes network snapshot display, host snapshot display, and time series/bar charts display for security metrics information. A time series plot can show that the network security becomes either better or worse over time, as shown in Figure 6. AHP menu may be used to display and edit AHP graph for host criticalness and security score. In Graph menu, we can restore zoomed view, save graph to jpeg file or print the graph. Help menu provides usage guidance of our toolkit.

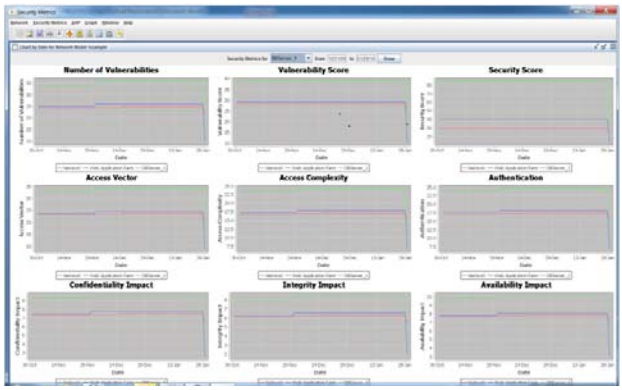

Fig. 6. Time Series Analysis for Host.

VII. Related Work

There are a number of standardized enumerations to help measurement in the information assurance, cyber security, and software assurance space. MITRE's Making Security Measurable effort [1] facilitates both effective security process coordination and the use of automation to assess and improve enterprise security. Seddigh et al. [18] introduce an information assurance metrics taxonomy for IT Network assessment. Their taxonomy has three categories, security, Quality of Service (QoS) and availability, based on their novel definition of information assurance. Stoddard et al. [15] propose an initial security metrics taxonomy for process control systems based on ISO/IEC 17799 [16] and ANSI/ISATR99.00.01- 2004 [17] standards.

Jaquith [5] gives detailed directions on how to quantify, classify, and measure information security operations in enterprise environments. He summarizes seventy-five different metrics that organizations use to assess their security posture. We identify a number of new security metrics that are not included in his work.

NVD/CVSS [4] is an industry standard for assessing the severity of computer system security vulnerabilities. Our tool can directly obtain the simple security metrics from CVSS database, and use them to derive more sophisticated metrics. Christian [19] pointed out CVSS is not used to its full potential when temporal metrics and environment metrics are left out. She propose to improve CVSS-based vulnerability prioritization by gathering context information from either available data or artificially created data.

In [20], Anoop et al. identify two layers in enterprise network security metrics: the component metrics and the cumulative metrics. The component metrics are about individual components properties which in many cases can be obtained from standard data sources like the National Vulnerability Database (NVD). The cumulative security metrics account for both the baseline metrics of individual components and the interactions among components.

VIII. Conclusion

We present our work on developing an automatic security analysis system that provides security metrics collection, security metrics management, and security metrics visualization. We implement a prototype toolkit that can help administrators better understand the security of the computer and networks. We show that AHP is a powerful mechanism to derive sophisticated security metrics. In our future work, we are interested in composing more enterprise-level security metrics in our system.

References

[1] Robert A. Martin, Making Security Measurable and Manageable. MILCOM 2008, November 19, 2008
[2] National Vulnerability Database (NVD), https://nvd.nist.gov
[3] Nessus vulnerability scanner, http://www.nessus.org/nessus/
[4] Common Vulnerability Scoring System (CVSS), http://nvd.nist.gov/download.cfm
[5] Andrew Jaquith, Security Metrics: Replacing Fear, Uncertainty, and Doubt, Addison Wesley, 2007.
[6] The CIS Security Metrics, Center for Internet security, https://www.cisecurity.org/tools2/metrics/CIS_Security_Metrics_v1.0.0.pdf.
[7] Vikram Phatak, Santosh Pawar, Vulnerability Protection: A Buffer for Patching, A Lucid Security Technical White Paper February 2004
[8] Kardi Teknomo, Analytic Hierarchy Process (AHP) Tutorial, http://people.revoledu.com/kardi/tutorial/AHP/index.html
[9] Analytic Hierarchy Process (AHP), http://www.mindtools.com/pages/article/newTED_88.htm
[10] Vulnerability Summary for CVE-2009-0022, http://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2009-0022.
[11] JFreeChart, http://www.jfree.org/index.html
[12] National Institute of Standards and Technology. Technology assessment: Methods for measuring the level of computer security. NIST Special Publication 500-133, 1985.
[13] M. Swanson, N. Bartol, J. Sabato, J. Hash, and L. Graffo. Security metrics guide for information technology systems. NIST Special Publication 800-55, 2003.
[14] Vaughn, R., Henning, R. and Siraj, A.: Information Assurance Measures and Metrics: State of Practice and Proposed Taxonomy. Proc. of 36th Hawaii Int. Conf. on System Sciences HICSS 03. (2003)
[15] Stoddard, M. et al.: Process Control System Security Metrics State of Practice. I3P Institute for Information Infrastructure Protection Research Report No. 1, Aug. (2005)
[16] ISO/IEC 17799:2005: Information Technology Security Techniques Code of Practice for Information Security Management. International Organization of Standardization (2005)
[17] ANSI/ISA-TR99.00.01-2004: Security Technologies for Manufacturing and Control Systems Standards. ANSI, Washington, D.C. (2004)
[18] Seddigh, N., Pieda, P., Matrawy, A., Nandy, B., Lambadaris, I., Hatfield, A.: Current Trends and Advances in Information Assurance Metrics. Proc. of the 2nd Annual Conference on Privacy, Security and Trust (PST 2004), Fredericton, NB, Oct. (2004)
[19] Christian Fruhwirth, Improving CVSS-based vulnerability prioritization with business context information(info), Fifth Workshop on Security Metrics (MetriCon 5.0), Washington, DC. 2010
[20] Techniques for enterprise network security metrics. Anoop Singhal and Xinming Ou. Proceedings of the 5th Annual Workshop on Cyber Security and Information Intelligence Research: Cyber Security and Information Intelligence Challenges and Strategies (CSIIRW) , Extended Abstract, April, 2009.