An Architecture Based on Agent-manager Model for Automated Data Collection of Security Metrics

Liniquer Kavrokov Vieira, Rodrigo Sanches Miani, Bruno Bogaz Zarpelão, Gean Breda, Leonardo Mendes
School of Electrical and Computer Engineering
University of Campinas (UNICAMP)
Campinas – SP - Brazil
{liniquer, rsmiani, bzarpe, gean, lmendes} @decom.fee.unicamp.br

***Abstract*— The requirement of organizations on computer network makes information security a key element to the evolution and continuity of services in our society. Security metrics are developed in order to offer a quantitative and objective basis for security assurance. This study proposes an architecture based on the agent-manager management model to allow the automated data collection from several components in a computer network, aiming to expand the security metrics application. A tool for measurement and automated data collection of metrics based on the architecture proposed were developed and applied in a real computer environment. Tests were performed showing that the architecture proposed is able to integrate information control and support the security monitoring process.**

Keywords-computer network; security metrics; security management; automated data collection.

I. INTRODUCTION

Current changes in computer networks have led to a paradigm shift. The network transmission of voice, data and video has been converging into a single multi-service network platform based on IP (Internet Protocol) [1]. This multi-service network can provide access to applications and content such VoIP (Voice Over Internet Protocol), video over demand and IPTV (Internet Protocol Television). Therefore, triple play services can be provided on a single connection, making customers dependent on the reliability and availability of these networks.

Despite the benefits that an integrated computer network can provide, several incidents and issues related to information security may exist. Some examples are the absence of security protocols in access points, unauthorized access to an information system, computer security vulnerabilities and exploits or misconfiguration of firewalls in computer networks [2].

These incidents may allow possible attacks and also compromise the whole infrastructure of an organization. In large-scale computing networks, identifying failures become a complex task for network administrators. A complex scenario composed by equipments and software provided by different vendors, requires the reduction of human intervention to prevent errors for the process of large amount of data.

Considering that preventive measures may also fail, it is important to adopt tools that might be used to avoid security incidents and prevent intrusions for a rapid response and supporting post-event activities [3].

Important organizations in security field such as SANS (SysAdmin, Audit, Network, Security) and NIST (National Institute of Standards and Technology) recommend the use of metrics in enterprises. Security metrics provide a quantitative approach to measure and analyze security controls and effectiveness at the security management [4]. Due to the lack of consensus on the term definition, most authors relate the concept of security metrics to quantifying and analyzing specific security data [5].

Nevertheless, using metrics to measure security levels usually requires a high cost-time for data collection. The, development of tools to automate the metrics collection process might be an efficient way to measure security in complex and heterogeneous computer network. Moreover, it contributes to decrease the data time collection and also to support the active security monitoring process [3].

This paper proposes an architecture based on a manager-agent management model for data collection from different components in IP-based computer networks, aiming to expand security metrics application. The proposed architecture is also able to integrate information control and support the security monitoring process.

The main motivations of this work are:

- The risk associated to security incidents and the importance of improving the security management;
- The lack of integrated tools to support the collection and analysis of security data in heterogeneous networks;
- High cost to quantify information security data, considering the time that network administrators could spend to manually collect and analyze information security data;

This paper is organized as follows. Section 2 presents some related work in the field. Section 3 presents security metrics and automated data collection. We introduce in Section 4 the criteria to select security metrics. Session 5 describes the architecture proposed in this paper. Next, Section 6 presents a case study conducted in a real computer network and discusses some experimental results. Finally, Section 7 offers the conclusion and poses future research questions.

 ISBN: 978-1-61208-209-7

II. RELATED WORK

Security metrics appear as an option to measure and assess information security issues and also to provide a feedback from organizations network [4].

Security metrics development and its application are discussed in [6] and [7]. Jaquith [4] states that any quantification problem that results in a numeric value can be seen as a metric. Its application can provide many benefits to information security management such as the evaluation of security risks, alerting to impending troubles, understanding the security infrastructure weaknesses and encouraging the use of technology.

Miani [5] proposes a methodology to facilitate the process of data analysis obtained with metrics application. The objective is to measure structural problems and vulnerabilities of network services. In Miani's work, metrics are collected in a real communication computer network. One of the difficulties found was the absence of a specific tool to help the data collection. Two following requirements were cited in this paper: recording security data in a specific database and presenting statistical data.

Ertürk [3] proposes a framework for security monitoring based on security metrics, which aim to examine the information security levels continuously over time. This requires collecting, processing cumulative data and reporting results according to organization requests. The author uses several monitoring tools to collect metrics, such as PRTG Network Monitor, Nessus Vulnerability Scanner, and others, which makes control and data integration more difficult.

Network management systems that enable the monitoring of network communication systems are found in [1] and [8]. Lee [1] presents a viable platform for network management based on architecture model TMN (Telecommunications Management Network) which uses JAVA technology and enables the control of functions such as, sending and receiving requests from network components, alarms and data performance. In this case, it is possible to collect automated data, but this is not the purpose of author.

This paper seeks to solve these difficulties proposing an integrated solution for automated data collection of metrics. It aims at supporting the monitoring of several components and improving the integration of data collected from different sources in IP-based computer networks.

III. SECURITY METRICS AND AUTOMATED DATA COLLECTION

Security metrics are developed aiming to help the assessment of security processes and controls developed by organizations. It also might be used to measure cost, effectiveness and improvements controls of organizations [4].

Security data in heterogeneous networks can be collected using basically two methods: manually or through automated processes. When performed manually, it can require a higher time and effort for collection. Whereas in automated processes, it is necessary to use complex routines of tools like protocol analyzers and in-depth analysis of audit logs [5].

Procedures for data collection on security metrics can be automated using a management platform capable of extracting data from different locations in a computer network. According to Lee [1], a management platform can enable the integration of several resources, such as the creation of graphical user interface, distributed process, networking facilities and object orientated paradigms. It can be implemented, for example, by using object oriented concepts and management API's supported by JAVA language programming [1].

However, when using several management platforms to collect metrics, it makes the control and data integration become more difficult. This paper proposes a solution for automated data collection from different components in a computer network using an integrated platform in order to helps the network administrator to have comprehensive and data integrated control from security information.

Therefore, the development of an integrated platform for security data collection can provide several benefits, such as [4]: increased accuracy in data collection, repeatability, high frequency of measurement, reliability, transparency and auditability.

IV. METRICS SELECTION

CIS (Center for Internet Security) [6] is a nonprofit organization whose goal is to improve information security of private and public sectors in general. One of its works has been to develop a guide which helps organizations selecting some feasible security metrics. This guide primarily states that in order to implement a metrics program, one must select a set of metrics that satisfies the interests of organizations and supports security managers in decisions making [6].

Inappropriate metrics application or poorly planned metrics can provide a false sense of security and loss of credibility. However, there is no consensus about attributes which makes good metric. Jaquith [4] proposed the following set of desirable features for security metrics: i) Defined consistently, without subjective criteria; ii) Easy to collect, preferably in an automated way; iii) Expressed as a percentage or cardinal numbers, not to qualitatively labeled as "high", "medium" and "low"; iv) Expressed by using at least one measure unit, such as "defects", "hours" or "dollars"; v) Significant enough so that decisions can be taken based on metrics results;

Besides having a set of metrics with the features mentioned above, it is necessary to have a plan and a preliminary evaluation of which data should be gathered. Some organizations collect more data than necessary, whether assuming that it is always better to have extra data, or because it is easier to collect a large amount of data, and then determine which will be used for developing a security metric implementation plan.

In order to facilitate the process of metrics development, security frameworks such as NIST [2], COBIT [10] and ISO/IEC [11] have been proposed taxonomies to classify metrics. Savola [12] considerates the following levels when classifying metrics: security metrics for cost-benefit analysis; trust metrics for risk analysis in terms of business; security

 ISBN: 978-1-61208-209-7

metrics for managing information and security metrics for products, systems and services (SDT - Security, Dependability and Trust).

Another point to be considered is whether the metric can be automated or not. Jaquith [4] classifies as technical metrics that allow the identification and diagnosis of security problems across the infrastructure (physical and logical) of an organization and usually do not require human intervention to collect data. A comprehensive taxonomy is also suggested in [2] based on management, technical and organizational classification. In this work we focused on this kind of metrics.

One of the contributions of this work is the creation of criteria to select metrics that can be automated. Automated metrics can be seen as metrics whose data can be collected on servers, desktops and network equipments using protocols grouped by layers of network.

The criteria are classified according to the protocols from OSI model, which might be used to read or collect data from metrics, for example, the protocols from: i) application layer, such as, SNMP, NetFlow, IPFIX and HTTP; ii) network layer, for example, ICMP; iii) transport layer, such as TCP and UDP. These criteria are based on TCP/IP related polling techniques, because TPC/IP is the point of convergence of different services in next generation multi-service networks. It is also important to mention that new criteria for metrics selection can be added as needed, take into consideration the network layer protocol used to collect the data.

The main goal of the criteria is to assist the metrics selection process prioritizing the metrics that could be gathered in an automated way Thus, many benefits can be provided, for example, helping decision making, shortening the time of metrics selection and ensuring greater confidence during the selection process and automation.

Metrics that do not meet all criteria presented previously or require manual intervention are also important and can be implemented by the organization. However, they are out of scope of this work.

V. DATA COLLECTION ARCHITECTURE

After selecting a specific set of metrics based on the automation criteria, it is necessary to develop an architecture that helps the implementation of a security metrics program. This paper proposes an architecture based on an agent-manager management model that is well-known in the network management field through a computer configured as a manager and other components playing the role of agents.

Agent-Manager management model is basically composed by three components [9]: a network management station, that monitors managed resources and adjust them according to the network administrators interests; a set of managed objects that correspond to an agent (switches, servers, routers) and to their associated data; and a management communication protocol used by the station manager and by agents to exchange data.

Figure 1 illustrates an agent-manager structure in a computer network. In this example, a computer is used as a management station that sends data requests to other components via SNMP (Simple Network Management Protocol). In the SNMP architecture, there is a database of managed objects called MIB (Management Information Base) that contains data about the managed device. This data is sent to the manager. In doing so, the management station might be able to execute metrics calculation routines using such collected data, providing useful information to network administrators.

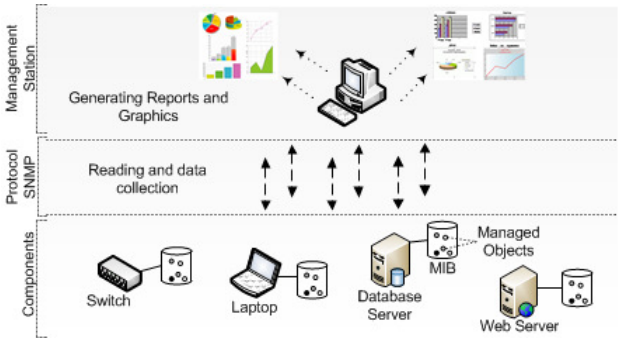

Figure 1. Example of a real management structure

It is possible to note in Figure 1 that only the SNMP protocol is used to exchange data between the management station and the components. Furthermore, several monitoring tools can be used to collect metrics and generate reports, which make control and data integration more difficult.

Whatever, it is important to collect data from several sources using different protocols and only one tool. Then, the architecture proposed in our work is also composed by three elements: i) monitored computer network, where the data to be collected is located; ii) communication protocols for data collection; iii) management station which collect data, transform data into metrics and generate reports and graphics.

Figure 2 shows an overview of the proposed architecture. In this example, different protocols can be used to exchange data between the management station and components. Moreover, a tool for measurement and automated data collection is used to integrate information control and support the security monitoring process.

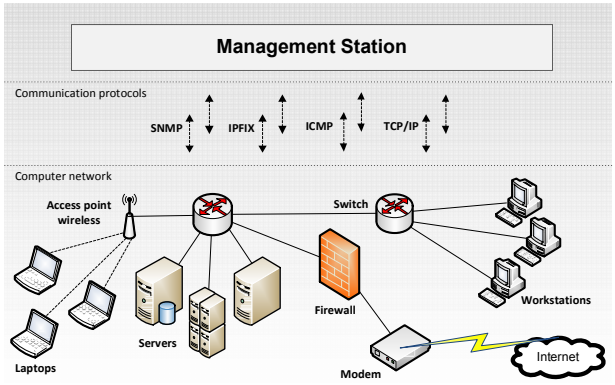

Figure 2. Overview of proposed architecture.

The following subsections present a detailed description of each component, their interactions and how they can help

with the improvement of security metrics application and collaborating with data integration.

A. *Computer Network*

The architecture defined in this paper aims to support the monitoring of several components establishing a computer network. Some examples of such components are: workstations, switches, wireless routers, servers, and laptops.

This set of heterogeneous components has a vast amount of data that might be important to organization's security management. Most of this components support management protocols and have a management database which is accessible and manageable.

Therefore, this data gathering can be automated through an integrated solution contributing to the network management processes and supporting the evaluation of network security level. Next subsection introduces some protocols that may be used to collect security data.

B. *Communication Protocols*

The TCP/IP (Transport Control Protocol/Internet Protocol) is a collection of protocols that can be used to perform the communication in a network. Socket [13] is an interface that provides a communication mechanism to computer applications using TCP or UDP protocol. Therefore, applications can be implemented through socket to read network data such as which servers has an Internet connection established, the number of ports opened or closed on a server, and others.

Another important protocol is the SNMP [13]. This is a standard protocol for TCP/IP network management defined by IETF (Internet Engineering Task Force). SNMP requires few resources to be implemented using UDP transport protocol to effectively exchange data between agents and managers. There is a database of managed objects called MIB (Management Information Base), which contains data about the managed device, such as the uptime of a server. The agent provides the interface between the manager and physical device being managed, providing a monitoring of different network components such as switches, servers and wireless routers.

A different way to collect data from network devices is through NetFlow protocol developed by Cisco Systems [13], which enables the network administrator to export and collect data flow information from routers that support this technology.

The IETF has been working to standardize NetFlow considering Cisco NetFlow version 9 [13] as a starting point. This effort is called IPFIX (Internet Protocol Flow Information Export) [13]. IPFIX optimizes the export of data through a template, supporting extensibility and adapting to different scenarios. In addition, a file format is specified for storing data that has been received. It facilitates the interoperability and reusability among a wide variety of flow storage, processing and analysis tools.

C. *Management Station*

The management station needs a tool to perform data collection from different devices in computer network. Its architecture is based on a MVC model (model-view-controller). This model divides the features of a system in layers, facilitating the maintenance and creation of new interfaces. The three layers and their functions are [14]:

- Model: Represents the logical layer. The whole business rule and data persistence are located in this layer. In this work it also aims to make requests to read and collect data from the computer network.
- View: User interaction layer or application interface. In this layer the data collected are pre-processed and viewed by network administrators through reports and graphs.
- Controller: This layer is responsible for determining the application flow, processing user actions and transmitting data to model layer.

Figure 3 illustrates the layers of this management station and its interaction between the network administrator and the computer network.

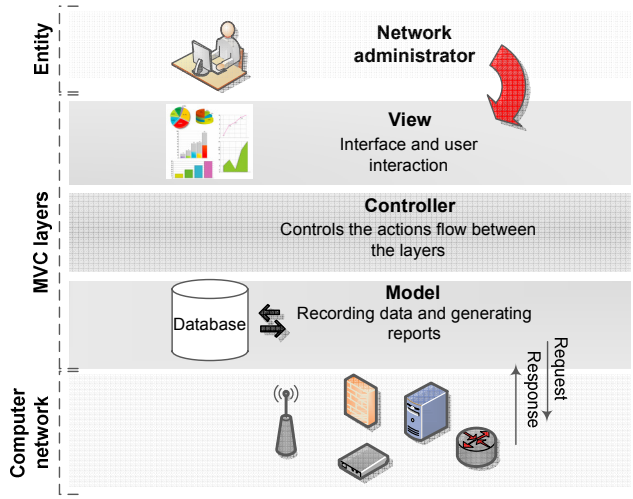

Figure 3. MVC Layers and its interactions.

In this example, the network administrator performs an interaction with view layer that allows managing computer network. Then, the controller layer processes the actions to the model layer, which performs requisitions to agents via communications protocols. The protocol returns the collected data to model layer and stores it in a database. With data collected and stored, network administrator can execute queries and analyze information using, for example, graphs and reports.

VI. A PRACTICAL APPLICATION

In order to demonstrate and validate the proposed solution, tests were performed in a real computer network environment, the Communication Networks Laboratory (LaRCom) at the University of Campinas (Unicamp).

Currently, LaRCom has a computer network that offers a 1Gb Ethernet network, whose external internet connection is provided by Unicamp. It is also composed by several

 ISBN: 978-1-61208-209-7

equipments such as application servers, database servers, printers, switches, VoIP terminals and laptops, all working on IP protocol. This infrastructure supports technology innovation projects that use this network to develop new solutions and tools such as the development of e-government to local governance and technologies for education.

Due to the intense use of services provided by LaRCom, some issues might occur, such as: slowness in certain servers, connection problems related to the Internet link, instability on firewall, deficiency in electrical power supply and others. In order to minimize these problems and help the security manager's duties, some security metrics were applied in this network. Then, it was developed an Integrated Platform for Security Metrics Analysis (IPSMA) tool based on the architecture proposed in this paper. IPSMA was implemented using Java technology and Oracle database management system to store data.

A. *Security Metrics Application*

First of all, some components of LaRCom were selected and monitored. The selection criteria here were choosing the essential and frequently used components. Figure 4 presents an overview of LaRCom and which components were selected to be monitored.

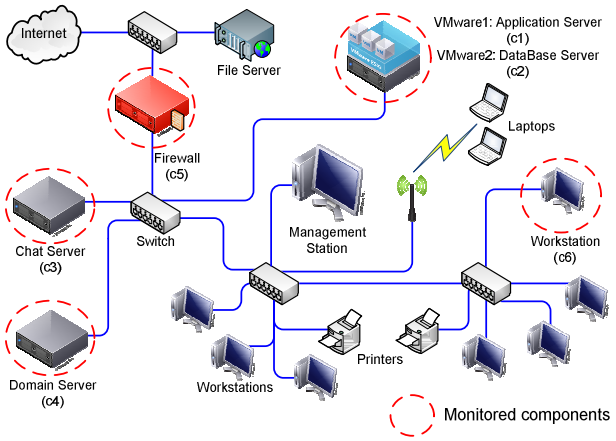

Figure 4. LaRCom network and monitored components.

Based on [2], [4] and [6], we selected metrics which satisfies the interests of organizations and effective the security management. The criteria presented in Section 4 also were used assisting metrics selection process and choosing which metrics would be automated and applied in the components of LaRCom. The following metrics were implemented: i) Uptime; ii) Number of open ports;

Uptime [4] has as a main goal to calculate the time that a computer, application or server is running. We used SNMP protocol to collect such data. Another metric used in this work, is the number of open ports [3]. Socket is used to perform communication with servers and identify the open ports. SNMP4J and Socket API were used for data collection of metrics, respectively. The metrics automation process was implemented using object oriented concepts supported by JAVA language programming.

Furthermore, it was developed a thread to collect each metric in order to initiate and control simultaneously the frequency of data collection. Data collection was performed continuously and automatically without intervention, during a period of one month without interruptions.

An illustration from IPSMA can be seen in Figure 5. This tool performs the automated data collection from metrics, generating reports and graphics. Moreover, it can help the network administrator to manage security metrics application and controls from organization.

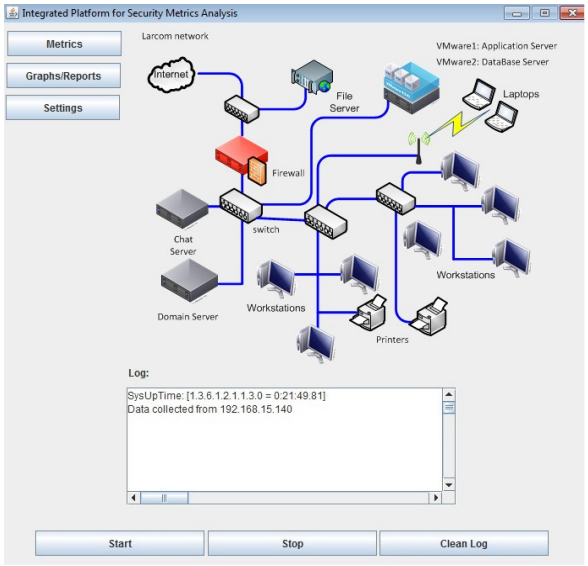

Figure 5. Integrated Platform for Security Metrics Analysis

Finally, Figure 6 shows the real network following the proposed architecture presented in Figure 2.

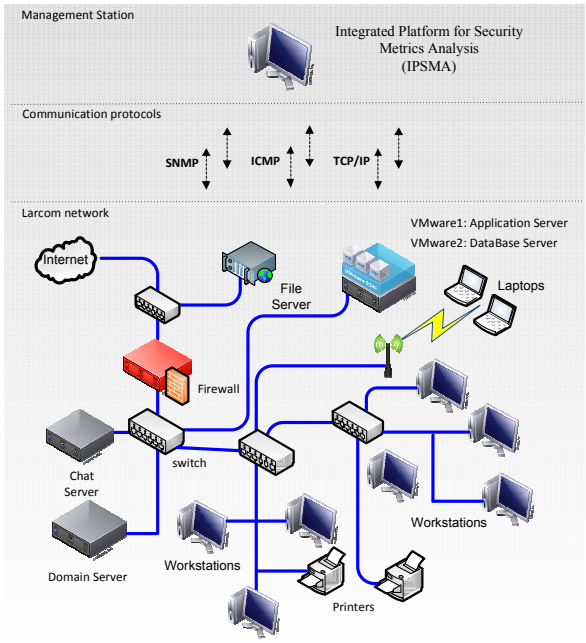

Figure 6. Real network following the proposed architecture

 ISBN: 978-1-61208-209-7

In this example, IPSMA was installed on the management station which uses different protocols to read and collect data from components. It contributes to integrate information control and support the security monitoring process.

B. *Discussion and Results*

Uptime [4] is a classic metric used to measure operations such as "uptime" and "downtime". Downtime is the total amount of time that resources were out of service planned or unplanned. Uptime is the total time for a given period minus any downtime.

Important data can be collected by applying this metric, such as: i) the total amount of time that resources were out of service due to regular maintenance; ii) the total elapsed time related to unexpected service outages; ii) the total time for a given period that a computer was up;

Pham [15] also presents a system to assess the overall security assurance. In this paper is used a metric to evaluate the time that two stations and a server are running. The results indicate that the station does not response to network requests at some points.

In our architecture also was applied this metric using only one platform to implement and collect data from metrics instead of several tools.

Figure 7 presents the total amount of time that components of LaRCom were up during a period of one month.

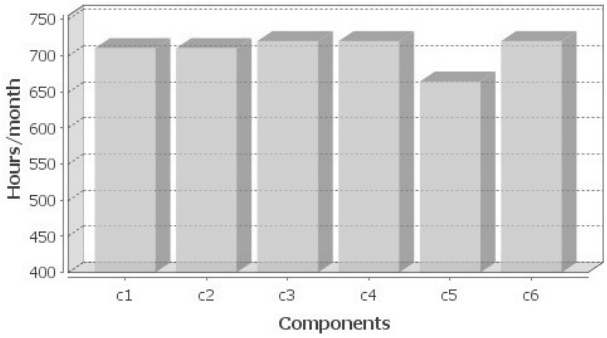

Figure 7. Total amount of "uptime" from components

As we can observe, the firewall (c5) stayed up an amount of hours smaller than other components. This data visualization can be detailed, for example, by the investigation of the number of shutdowns of each component per week, as seen in Table 1.

TABLE I. TOTAL NUMBER OF SHUTDOWNS OF EACH COMPONENT

	C1	C2	C3	C4	C5	C6
W1	-	-	-	-	1	-
W2	2	2	-	-	2	-
W3	-	-	-	-	4	-
W4	1	1	-	-	4	-

When analyzing Table 1 and Figure 8, it is possible to note that the application server (c1) and database server (c2) had three "shutdowns". According to the network administrator, these shutdowns were required and planned. The firewall (c5) had eleven unplanned "shutdowns". So, this fact alerted the network administrator of a potential risk or instability in equipment C5. Soon, it was found that the computer had some hardware troubles and it was sent to maintenance.

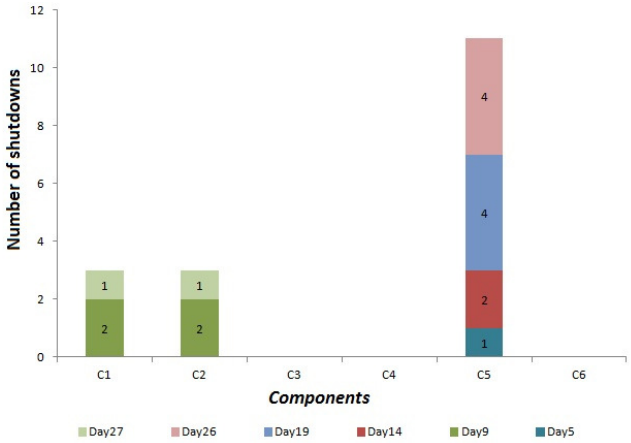

Figure 8. Days when the components have "shutdown"

Another metric was applied aiming to verify which server ports were opened, and also the frequency at which a certain door remains open. Thus, important data can be obtained to support security management such as: number of computers that allow remote access via terminal service, vulnerabilities which can be exploited through the open ports without the consent of the network administrator and others [16].

Table 2 presents the top five servers open ports of LaRCom and the number of components with open ports.

TABLE II. TOP FIVE OPENED PORTS OF LARCOM

Port	Service	Number
80	HTTP	3
3389	Terminal services	3
1521	Database connection	3
53	DNS (Domain name Server)	2
22	SSH	1

In addition, SANS [17] also reported the top ten ports that can be exploited by attacks or source of critical. When comparing Table 2 with [17], we found that there are ports such as 80, 3389 and 53 which were found in both lists. These ports are considered dangerous by [17] and the security policy in relation to these ports should be reconsidered. Ports 80 and 22 which are operating system standard also are considered easy targets of automated attacks and should be changed.

 ISBN: 978-1-61208-209-7

Ertürk [3] proposes a framework for security measurement, collection and reporting data. Further, the implementation was applied in a public organization.

The author also uses a metric to verify the number of open ports on servers. In this case, Nessus Vulnerability Scanner tool is used to collect data from several components. Nevertheless, other metrics are presented by the author and several monitoring tools, such as PRTG Network Monitor, are used to collect data. Due to this fact our architecture showed better to integrate information control and support the active security monitoring process using only one tool.

VII. CONCLUSION AND FUTURE WORK

In this paper, we proposed an architecture for automated collection of security metrics. Based on this architecture, also developed an Integrated Platform for Security Metrics Analysis performing a case study in a real computer network. A practical application was important to test and validate the behavior of architecture.

This application has brought some benefits to information security management, such as: an integrated solution able to support the monitoring of several components in a computer network; reduction in data collection time and greater reliability through automation; development of a historical database, which could be used to discover unexpected relationships and to reveal other predictive interactions that might exist.

The case study revealed some issues on the firewall and also breaches in the laboratory security police, related to several open ports in the investigated servers. This information is used to security analysis and can be improving by adding new security metrics.

Future work includes: i) development of new metrics using other network protocols, such as IPFIX; ii) improvement of the architecture in order to allow distributed management, solving a possible scalability issue caused by the large number of managed components; iii) deployment of the proposed architecture in other network scenarios.

ACKNOWLEDGMENT

The authors would like to thank the State of São Paulo Research Foundation (FAPESP) and the Coordination for the Improvement of Higher Education Staff (CAPES) that supports this work.

REFERENCES

[1] J. Lee, “Enabling network management using Java technologies,” IEEE Communications Magazine, Vol. 38, January 2000, pp. 116–123.

[2] M. Swanson, N. Bartol, J. Sabato, J. Hash, and L. Graffo, “Security Metrics Guide for Information Technology Systems,” National Institute of Standards and Tachnology, Special Publication 800–55, 2003. http://csrc.nist.gov/publications/nistpubs/800-55-Rev1/SP800-55-rev1.pdf retrieved: April, 2012.

[3] V. Ertürk, “A Framework Based on Continuous Security Monitoring”, A thesis submitted to the Graduate School of Informatics of The Middle East Technical University, 2008. http://etd.lib.metu.edu.tr/upload/12610139/index.pdf retrieved: April, 2012.

[4] A. Jaquith, Security Metrics – Replacing Fear, Uncertainty and Doubt, Addison-Wesley, 2007.

[5] R. Miani, B. Zarpelao, and L. Mendes, “Application of Security Metrics in a Metropolitan Network: A Case Study,” The 7th International Telecommunications Symposium (ITS 2010), 2010.

[6] The Center for Internet Security, “Quick Start Guide,” Vol. 1.0.0, November 2010. http://www.cisecurity.org retrieved: April, 2012.

[7] P. Black, K. Scarfone, and M. Souppaya, “Cyber Security Metrics and Measures,” National Institute of Standards and Tachnology, 2009. http://xlinux.nist.gov/~black/Papers/cyberSecurityMetrics2007proof.pdf retrieved: April, 2012.

[8] J.Park, N. Joo, and T. Kim, “Java-Based Network Management Environment,” IEEE Internacional Conference on Communications, Vol. 2, June 1998, pp 1124–1128.

[9] H. Hegering, S. Abeck, and B. Neumair, Integrated Management of Networked Systems – Concepts, Architectures and Their Operational Application, Morgan Faufmann Publishers, 1998.

[10] http://www.isaca.org/COBIT/Pages/default.aspx retrieved: April, 2012.

[11] http://www.iso.org retrieved: April, 2012.

[12] R. Savola, “Towards a Security Metrics Taxonomy for the Information and Communication Ttechnology Industry,” Proc. Internacional Conference on Software Engineering Advances (ICSEA 07), 2007, pp. 60.

[13] http://tools.ietf.org retrieved: April, 2012.

[14] An Oracle White Paper, Oracle Application Development Framework Overview, 2011. http://www.oracle.com/technetwork/developer-tools/adf/adf-11-overview-1-129504.pdf retrieved: April, 2012.

[15] N. Pham, L. Baud, P. Bellot, and M. Riguidel, “A Near Real-time System for Security Assurance Assessment,” Proc. The Third Internacional Conference on Internet Monitoring and Protection (ICMP 08), 2008, pp. 1252–1260.

[16] http://www.vulnerabilityassessment.co.uk retrieved: April, 2012.

[17] http://isc.sans.edu/top10.html retrieved: April, 2012.

 ISBN: 978-1-61208-209-7