6-2 Automated Vulnerability Monitoring Techniques for IT Assets on the Network

Takeshi TAKAHASHI

Each organization needs to manage vulnerabilities of its IT assets, but it may be difficult for some organizations to allocate enough resources for that. This paper introduces a system that monitors an organization's internal IT assets and their related vulnerability information. The system streamlines the operations within each organization. The core of this system is the automatic generation of identifiers of IT assets. This paper also introduces a prototype system, with which we discuss the issues to be solved as our future work.

1 Introduction

To maintain security in an organization, one must understand the vulnerabilities of IT assets on the managed network in real time as much as possible. However, implementing such work manually requires large amounts of labor, and the fact is that many organizations find it difficult to allocate sufficient human resources. Far from understanding in real time whether the assets are vulnerable, many organizations do not even reach the level of necessary vulnerability information in their organization. It is also necessary to take care to regularly update vulnerability information, but even more organizations do not do that.

To manage vulnerabilities, one must first know what IT assets are in the organization. The importance of identifying IT assets is also clearly explained in ISMS, etc., and many organizations such as large companies handle this. However, small organizations are a step behind.

Also, understanding vulnerabilities is not a onetime activity. As time passes, IT assets undergo changes, and these changes should be tracked in real time. What is needed is the development of technology that can achieve IT asset management and the accompanying vulnerabilities management through automation, without increasing human resources.

1.1 Related research

The proposed method will take into account the results of various related research, and help reduce the labor involved in managing IT assets and vulnerabilities in an organization, which is the challenge targeted in this paper. This section provides an overview of related research. For details, refer to Chapter 2 of document [1], and each reference document.

a) IT assets management tool: There are already various tools for collecting IT assets information. Registry information is a widely used source from where information is collected, and this can be collected by the tool that comes along with the OS. reg.exe that comes with Windows 8 is one such tool, wherein registry information in the CUI can be collected. Also, the Get-ChildrenItem commandlet is available in Windows PowerShell, so registry information in the CUI can be acquired. Besides the tools that come with the OS, there are also various other software management tools; most of them are integrated tools that proactively also collect information from sources other than the registry.

b) Open information repository/schema: Several organizations have started providing the information that they have accumulated on asset vulnerabilities online and in repositories. Among them, notable repositories are NVD[2] and JVN[3]. The authors had proposed and built a prototype of a huge database by combining these repositories. This database will serve as a knowledge base that can be searched across.

c) Information schema: In many of the repositories mentioned above, the information is described using standardized XML schema. In the NVD mentioned above, as one of the keys for searching vulnerabilities information, the ID of the IT asset that is prone to vulnerability is described in CPE[4] format. Standards

such as CPE and SWID[5] are established for the IT asset ID, and there is also an ID list (dictionary) based on these standards. In addition to the schema for gathering information, schema for exchanging information are also standardized. For example, there is the IODEF[6] schema and its extended technology IODEF-SCI[7], which are used for sharing incident information. In IODEF-SCI, various XML information can be exchanged.

1.2 Our approach

There are various IT asset information collection tools, but except for the reg.exe and PowerShell tools that come with the OS, most are proprietary tools. Also, the information collected may not be complete, so proprietary tools are also used to compensate for technology/information which is lacking. Moreover, proprietary vulnerabilities information is used for auto-retrieving the vulnerabilities information of one's own IT assets. By using proprietary technology and information, one can avoid using less reliable information, but unless the tool provider prepares/checks/provides the information, one cannot deal with new IT asset/vulnerabilities information. In today's scenario where the distribution of open information has started to progress, we aim to ensure scalability by maximizing use of open data, and also aspire to build free software with all its system specifications disclosed, thereby encouraging its use for security automation.

1.3 Contributions of this paper

In this research, we use open data, and labor saving Intranet management is achieved by automating asset management and vulnerabilities management. First, for asset management, network information of the in-house network and information about IT devices such as PCs and smartphones are periodically collected. Then, by searching the knowledge base in which various security-related information is accumulated, the collected information is given ID and structured. Next, for vulnerabilities management, confirm whether the IT devices within the in-house network are vulnerable. Specifically, with the ID of the collected IT asset information as the key, confirm whether there is vulnerabilities information saved in the knowledge base. When corresponding vulnerabilities information that has not be handled is confirmed, immediately send a warning message to the administrator, to enable the administrator to respond quickly. We are also investigating an architecture wherein the network will implement the initial response (triage) autonomously, but that is not within the scope of this paper; only the results of the initial investigation are shared in this paper. We are considering combination with existing technologies for information that must be reliable; the proposed system complements existing technologies and does not replace them.

The results of the initial investigation indicate that by using open standards and information, warnings can be issued for vulnerabilities in real time. This shows that labor-saving in security operations can be encouraged in an organization. Also, activities for establishing various information schemas with the purpose of automating security operations are being done actively; by building specific labor saving tools in which these schemas are actu-

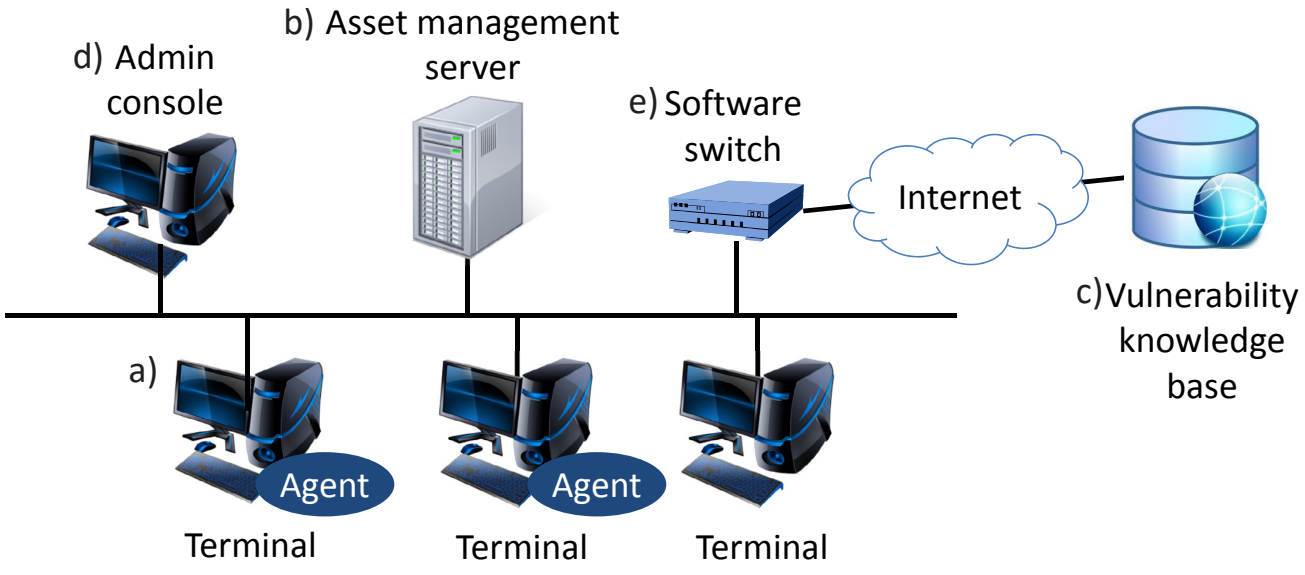

Fig. 1 Configuration example of proposed system

ally used, the importance of these activities is clarified. For details, refer to document[8], as this paper is a summary of that document.

2 System architecture

The proposed system constantly checks IT asset information, and converts that information to an ID. Based on that ID, it searches vulnerabilities information in the knowledge base, and if it finds vulnerabilities information, a warning is given. This section first defines the roles needed for the proposed system, and it next provides a summary of the process wherein these roles are linked and operate.

2.1 Role model

Figure 1 is an example of the configuration of the proposed system. In the proposed system, five types of roles are mandatory: terminal, asset management server, knowledge base, administrator terminal, and software switch.

a) Terminal: Generally, the terminals in an organization are the ones that the employees use for their work. Normally, a software module called an "Agent" is installed, but there are some organizations that have not installed this.

b) Asset management server: This is a server that collects/stores various types of organizational information such as information collected by the Agent, and information collected by the asset management server itself. This role determines the ID of each IT asset information by communicating with the knowledge base, and saves it together with the corresponding IT asset information. Then with this ID as key, it searches whether vulnerabilities information exists in the knowledge base, and if needed, creates and sends a warning message to the administrator.

c) Knowledge base: This is a database where various types of information on security are stored[1]. In this paper, among the various information, only the vulnerabilities information described in CVE/CVRF format, and the CPE dictionary in which the correspondence of CPE-ID and IT asset information is described, are used.

d) Administrator terminal: This is the terminal that the system administrator uses; it is also the terminal where a warning is sent when an abnormality related to vulnerabilities is detected. Currently, this terminal is placed in the same network segment as the terminal, but it can also actually be in another network and connect via the mobile phone network.

e) Software switch: This is a network device which is the demarcation line of the management network segment; it can take various forms such as switch or router. In this paper, this role does not have a particular role to perform, but when considering automation of initial responses in the future, this role will control the traffic.

3 Summary of processes of the proposed system

Figure 2 shows a summary of the processes the proposed system. The process established in this system is such that after the three stage process is implemented, a warning message is sent if needed. First, the proposed system collects information related to the IT asset on the connected network. Next, the identifier of the IT asset to be managed is generated from the collected information, and using that identifier, the vulnerabilities information in the knowledge base is searched. If vulnerabilities information is allocated, it means that IT assets being managed by the proposed system are vulnerable, so a warning message is sent.

The identifier and vulnerabilities information of IT asset information allocated in the above mentioned process are described in detail in Subsection **3.1**.

3.1 Allocation of identifier of IT asset information

Regarding the identifier for uniquely identifying IT assets, although there are standards such as CPE and SWID, this time CPE (which is also used in NVD) is used. In this system, the CPE identifier that corresponds to the IT asset information is allocated by text searching the CPE diction-

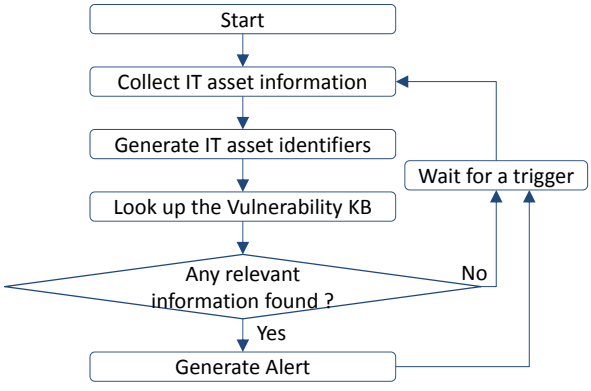

Fig. 2 Summary of process of proposed system

ary. In other words, when the Agent collects terminal information such as OS name and name of the installed applications, and sends it to the asset management server, the asset management server that receives the information will search the CPE-ID in the knowledge base based on the application name written in that terminal information. There are very few cases where there will be complete match in text search, so the ratio of matching of the search result is quantified, and when that ratio crosses the threshold value (the value specified beforehand by the settings file), it is judged as a match and CPEID is allocated. When CPEID is allocated, this information is saved/stored along with the terminal information.

In the IT asset information collected from sources such as the registry, there is also information written in Japanese kanji or hiragana/katakana, not in a Western alphabet. However, CPE can handle multiple languages, and there are already several expressions in Japanese in the CPE dictionary. For this reason, even for the information written in Japanese that is included in IT asset information collected by this prototype, the corresponding CPE-ID can also be acquired.

3.2 Allocation of vulnerabilities information

When the security vulnerability is verified using CPE-ID that is auto-generated by the procedure mentioned in Subsection **3.1**, the XML that matches with the NVD element <vulm:product> is searched from the CPE described in the software information of the terminal information. If there is a matching NVD, it is judged as vulnerable, and a warning is output in IODEF-SCI format.

4 Information schema

The proposed system operates on the premise that the information is correctly structured. Therefore, for all the information, the schema is either specified or defined, and its correct usage must be adhered to.

4.1 IT assets information

In the proposed system, the collected IT assets information is structured and saved. Standardized IT asset information schema such as Asset Identification[9] or ARF[10] can be used as schema at that time, but these standards are over engineered for the structure of the current prototype, so an independent schema was used. In the future, if the information that should be collected increases, or there is a case where the need has arisen for sharing of information with other organizations, the usage of these standardized schemas shall be re-examined.

端末一覧画面

TermID	IP	Update	Insert		
080027667EE0	192.168.56.102	2016-02-18T01:06:36	2016-02-18T01:06:36	OK	CPE detail
0800278E56AC	192.168.56.101	2016-02-18T01:01:18	2016-02-18T00:43:54	OK	CPE detail

Fig. 3 Terminal list screen

```
<SoftwareDetailInfo version="1">
  <SoftwareName>秀丸エディタ64 (8.51)</SoftwareName>
-<CPE name="cpe:/a:hidemaru:editor:8.51">
    <source name="official-dictionary" matchMethod="match-name-version" matchRate="1
  </CPE>
  <CVE>CVE-2015-0903</CVE>
  <SoftwareVersion>8.51</SoftwareVersion>
  <Publisher>有限会社サイトー企画</Publisher>
</SoftwareDetailInfo>
-<SoftwareDetailInfo version="1">
  <SoftwareName>Oracle VM VirtualBox Guest Additions 5.0.12</SoftwareName>
  <SoftwareVersion>5.0.12.0</SoftwareVersion>
  <Publisher>Oracle Corporation</Publisher>
</SoftwareDetailInfo>
-<SoftwareDetailInfo version="1">
  <SoftwareName>7-Zip 9.38 (x64 edition)</SoftwareName>
-<CPE name="cpe:/a:7-zip:7-zip:9.38">
    <source name="local-dictionary" matchMethod="exact-match" matchRate="100.0"/>
  </CPE>
  <SoftwareVersion>9.38.00.0</SoftwareVersion>
  <Publisher>Igor Pavlov</Publisher>
  <Size>0x12a7</Size>
  <InstallDate>20150317</InstallDate>
</SoftwareDetailInfo>
-<SoftwareDetailInfo version="1">
  <SoftwareName>Microsoft .NET Framework 4.5.2</SoftwareName>
-<CPE name="cpe:/a:microsoft:.net_framework:4.5">
    <source name="official-dictionary" matchMethod="match-name-version" matchRate="6
  </CPE>
  <CVE>CVE-2012-0163</CVE>
  <CVE>CVE-2012-4776</CVE>
```

Fig. 4 IT asset information that is being managed

4.2 Warning message

This system sends a warning message when vulnerabilities information about IT assets in the organization are found. We are using IODEF-SCI for the schema of this message. In this message, both CVE-ID of the vulnerability and CPE that identifies the IT asset are embedded.

IODEF is an information structure standard created for exchanging incident information between organizations, so mandatory fields such as fields for writing the sender's information are prepared, and it also has excellent extendibility. Also, CVE and CPR etc. can be embedded in the IODEF document using IODEFSCI. Therefore, as using IODEF-SCI serves the purpose, it is used as the schema of this message.

5 Prototype structure

This section describes a prototype of the proposed method. Information structure such as the registry differs depending on the OS and its version, so this prototype was implemented for Windows 7. However, it was implemented to also partially support Windows XP, Windows 8 and Linux OS.

This prototype collects the information of each terminal, and that information gets stored in the server. The list of terminals saved can be viewed as seen in Fig. 3, and by clicking on the IDs of the terminals specified in this Figure, the IT asset information of the corresponding terminal shown in Fig. 4 can be acquired. Further, each terminal sends information to the server when the Agent module is installed, so for the terminals shown in Fig. 3, the IT asset information is already saved in the server.

Figure 4 shows that the CPE tag and CVE tag are in various information, and that the information is provided by the asset management server and knowledge base. Information other than these is the information collected from terminals via the Agent etc. Each time new software is saved in the terminal, all the information is updated. Similarly, even when a new vulnerability information is registered on the knowledge base side, this CVE tag is updated.

When a vulnerability is found, a warning message is issued. It is sent by email in the current implementation. Actually, these warning messages are created assuming that the administrator can read them on a smartphone when out of the office. Figure 5 shows the warning message screen.

6 Summary

This paper shares the status of our investigation aimed at automated management of IT assets and related vulnerabilities in an organization, using open vulnerability information and relevant tools. In the proposed system, a method for searching the knowledge base is adopted. The knowledge base search has two levels: the ID allocation for IT asset information, and allocation of corresponding vulnerability information. We verified the feasibility of this method. However, many issues remain (refer to Section 4 of document [8]), particularly the allocation precision of IT asset information and wild card expression support for vulnerabilities information are pressing problems to be investigated. We also intend to investigate the initial response after vulnerability information is acquired, and automation technology using SDN.

Acknowledgments

We wish to extend our heartfelt gratitude to Dr. Koji Nakao, Senior Researcher, and Dr. Kazumasa Taira, Research Center Director, for their support in conducting this research.

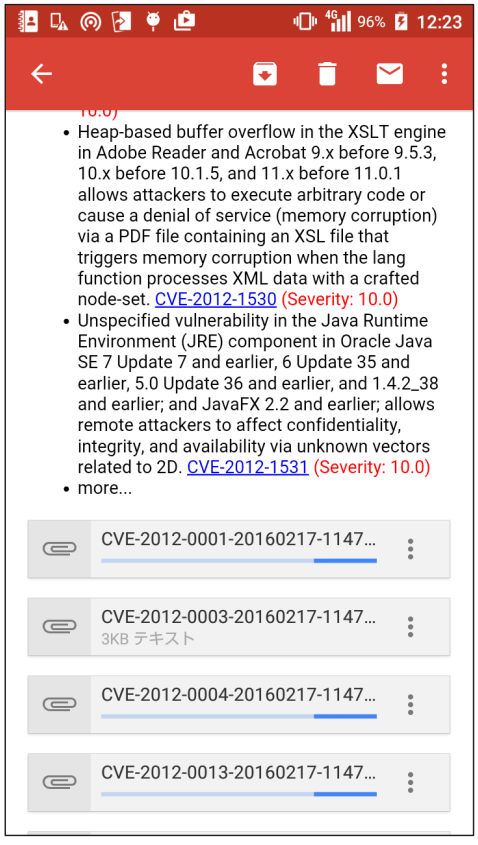

Fig. 5 Warning message to administrator

References

1 T. Takahashi, Y. Kadobayashi, "Reference Ontology for Cybersecurity Operational Information," The Computer Journal, 2015.

2 National Institute of Standards and Technology, "National Vulnerability Database Version 2.2," 2014. [Online]. Available: http://nvd.nist.gov/.

3 JPCERT/CC and IPA, "Japan Vulnerability Notes," 2014. [Online]. Available: http://jvn.jp.

4 International Telecommunications Union, "Common platform enumeration," ITU-T Recommendation X.1528, 2012.

5 International Organization for Standardization/International Electrotechnical Commission, "Software asset management -- Part 2: Software identification tag," ISO/IEC 19770-2:2009, 2009.

6 The Internet Engineering Task Force, "The Incident Object Description Exchange Format," RFC 5070, Dec. 2007.

7 The Internet Engineering Task Force, "An Incident Object Description Exchange Format (IODEF) Extension for Structured Cybersecurity Information," RFC 7203, April 2014.

8 T. Takahashi, D. Miyamoto, K. Nakao, "Toward Automated Vulnerability Monitoring using Open Information and Standardized Tool," IEEE International Conference on Pervasive Computing and Communications, 2016.

9 National Institute of Standards and Technology, "Specification for Asset Identification 1.1," NIST Interagency Report 7693, 2011.

10 National Institute of Standards and Technology, "Specification for the Asset Reporting Format 1.1," NIST Interagency Report 7694, 2011.

Takeshi TAKAHASHI, Ph.D.

Senior Researcher, Cybersecurity Laboratory,
Cybersecurity Research Institute
Cybersecurity, Network Security