Review of k-Zero Day Safety Network Security Metrics to Measure the Risk on Different Vulnerabilities

Ms. Suchita D. Pawar
Department of computer Engineering
JSPM, Hadpsar
PUNE, INDIA
E-mail: suchitadpawar18@gmail.com

Prof. H. A. Hingoliwala
Department of computer Engineering
JSPM, Hadpsar
PUNE, INDIA
E-mail: ali_hyderi@yahoo.com

***Abstract*—** Today's computer networks face intelligent attackers who combine multiple vulnerabilities to penetrate networks with destructive impact. The overall network security cannot be determined by simply counting the number of vulnerabilities. Due to the less predictable nature of software flaws we can't measure the security risk of unknown vulnerabilities. This affects to security metrics, because a safer configuration would be of little value if it were equally vulnerable to zero-day attacks. In this paper, instead of just measuring how much such vulnerability would be required for compromising network assets we can also attempting to rank unknown vulnerabilities. By using collaborative filtering technique to different (types of) zero-day vulnerabilities and novel security metrics for uncertain and dynamic data we propose a Flexible and Robust k-Zero Day Safety security model to rank the zero-day attacks.

Keywords- *vulnerability, zero-day attacks, collaborative filtering.*

*****

I. INTRODUCTION

Today Internet connects and enables a growing list of critical activities from which people expect services and revenues. In other words, they trust these systems to be able to provide data and elaborations with a degree of confidentiality, integrity, and availability compatible with their needs. Unfortunately, this trust is often not based on a rational assessment of the risk to which the system could be exposed [2]. Users typically know only the interface of the system and, for example, they have too little information for evaluating the confidentiality of their credit card number: it could be even transmitted on an SSL armored link, but this does not help if on the other side it will be stored on a publicly available database! [2].

The scale and severity of security threats to computer networks have continued to grow at an ever increasing pace. One of the main difficulties in securing computer networks is the lack of means for directly measuring the relative effectiveness of different security solutions in a given network, because "you cannot improve what you cannot measure."[8]

A variety of authors have noted that identifying vulnerabilities in isolation is only a small part of securing a network, and that a significant issue is identifying which vulnerabilities an attacker can take advantage of through a chain of exploits [1]. For example, an attacker might exploit a defect in a particular version of ftp to overwrite the .rhosts file on a victim machine. In the next step, the attacker could remotely log in to the victim. In a subsequent step, the attacker could use the victim machine as a base to launch another exploit on a new victim, and so on [1].

II. BACKGROUND

Every organization is at risk for zero-day exploits regardless of size. These exploits will often circulate for months until the vulnerability is made public, leaving organizations unprotected.

There were more zero-day vulnerabilities discovered in 2013 than in any previous year according to Symantec's Internet Security Report of 2014. "The 23 zero-day vulnerabilities discovered represent a 61 percent increase over 2012 and are more than the two previous years combined"

Analysis of zero day vulnerabilities by following methods

A. Statistical-based techniques

Statistical-based techniques for the detection of exploits depend on publically known past exploits attack profiles. This defense technique adjusts the historical exploit's profile parameters to detect new attacks. The quality of the detection is directly related to threshold limits set by the vendor or security professional using this technique. This technique find out what normal activity is and anything outside of normal is blocked or flagged.

The system that is utilizing this technique is online. Existing techniques in this approach perform static analysis and/or dynamic analysis on the packet payloads to detect the invariant characteristics reflecting semantics of malicious codes (e.g., behavioral characteristics of the decryption routine of a polymorphic worm)

B. Signature-based technique

Signature-based detection is often used by virus software vendors who will compile a library of different malware signatures ie virus definitions. They will match these signatures with local files, network files, email or web downloads depending on settings chosen by the user. These libraries are constantly being updated for new signatures of new exploited vulnerabilities. For detection of exploited vulnerabilities this technique requires a signature to be in the signature library because of that purpose virus software vendors are frequently updating their virus definitions. Cost is main constraints because of continuously updating virus definitions in libraries.

Signature-based techniques are classified by content-based, semantic-based and vulnerability-based signatures and are somewhat effective against polymorphic worms.

C. Behavior-based technique

The Behavior-based techniques look for the essential behavior of worms which do not require the examination of payload byte patterns

Main aim of such techniques is to predict the future behavior of a web server, server or affected machine in order to deny any behaviors that are not expected. Those behaviors are analysis by the current and past interactions with the web server, server or affected machine. This technique relies on the ability to predict the flow of network traffic.

D. Hybrid detection-based technique

Hybrid-based techniques combine heuristics with various combinations of the three previous techniques. Using a hybrid model technique will overcome a weakness in any single technique.

The benefits of their hybrid technique are four fold:

- This technique identifying zero-day attacks from data collected automatically on high interaction honeypots.
- This technique designed by combining the advantages of existing techniques and minimizing their disadvantages.
- This technique does not need prior knowledge of zero-day attacks and uses Honeynet as an anomaly detector.
- This technique can detect zero-day attacks in its early phase and can contain the attack before major penalty occur.

III. LITERATURE SURVEY

In [1] P. Mell, K. Scarfone, and S. Romanosky (2006), main goal of Common vulnerability Scoring System (CVSS) is "Good enough" for non-expert administrator, Relative Simplicity and efficient representation of vector. They provide security analysts and vendors standard ways for assigning numerical scores to known vulnerabilities that are already available in public vulnerability databases, such as the National Vulnerability Database (NVD). But Temporal/Environmental aspects not well-tested, Requires good documentation. In[2] Common Weakness Scoring System (CWSS) the process of discovering new vulnerabilities, automated and human analysis will find weaknesses.

In [5] R. Lippmann, K. Ingols, C. Scott, K. Piwowarski, K. Kratkiewicz, M. Artz, and R.Cunningham (2006), Defense in depth is a common strategy that uses layers of firewalls to protect Supervisory Control and Data Acquisition (SCADA) subnets and other critical resources on enterprise networks. NetSPA (NETwork Security and Planning Architecture) verifies and, if necessary, provides suggestions to restore defense in depth for large enterprise networks. NetSPA successfully imported vulnerability scanner and firewall configuration information and was able to produce attack graphs and make recommendations in only a few minutes.

In [9] Mohammed, M.M.Z.E.; Chan, H.A; Ventura, N.; Pathan, A-S.K. (2013), Their technique first tries to detect zero-day polymorphic worms and then tries to prevent them. "STF observes all network traffic at an edge network and the Internet. The traffic is passed simultaneously to both Honeynet and IDS/IPS (Intrusion Detection System/Intrusion Prevention System) sensors through a port mirroring switch". Suspicious Traffic Filter (STF) is the first defense layer from zero-day attack. Zero-day Attack Evaluation (ZAE) takes input (malicious traffic) from STF to evaluate and analyze captured zero-day attack. Signature Generator (SG) generates new signature for zero day attack and updates the signature database in STF.

In [3] M. Frigault, L. Wang, A. Singhal, and S. Jajodia (2008), In this paper Explores the causal relationships between vulnerabilities and measuring network security in a dynamic environment. In this module used tool for measuring network security by integrating attack graphs generated by the TVA system with CVSS scores provided by NVD. Tool accuracy is important to get optimal result.

In [5] J. Homer, X. Ou, and D. Schmidt (2009), apply probabilistic reasoning to produce a sound risk measurement. Running time of algorithm depends on size of data sets and interconnection in attack graph. If in attack graph their exist

lots of interconnectivity in exploits then it not able to generate optimal result.

IV. PROPOSED SYSTEM

In this model we can able to count known as well as unknown dynamic vulnerabilities and design optimal firewall rule policy to block them. We can also calculate the risk of vulnerability to affect the security of system. Considering the risk value rank the vulnerability to reduce the cost of system security. Collaborative filtering technique is used for ranking the vulnerability

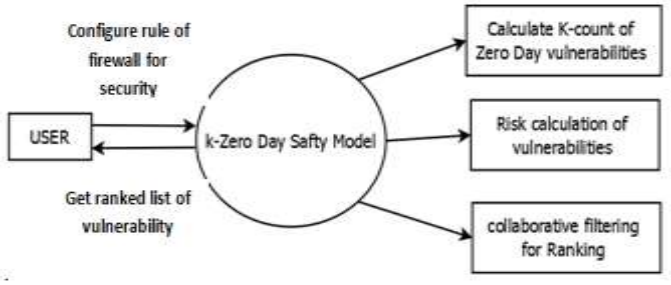

Fig1. Module Structure

Procedure: K0d_Bwd
Input: Firewall Rule list ie a set of assets allowed rule A and blocked rule B
Output: A non negative real no. k
Method:
For each rule in rule list
Let L be the status of rule representing action and mode;
While at least one of the following is possible
Capture all data packets in network;
Matches all data packet information with rule list;
Count allowed data packets Fi= $\sum_{a \in A} A$;
Let k=min(kod(Fi∩E0,ø):Fi is a count of allowed packets in rule list .
Return: k

Algorithm 1: To compute value of k

A. *Computing k count*

In this model firewall rule list dataset is designed by using network rules and used jpcap and WinPcap software's to capture the data packets travelled in network. After capturing data packets matches their sources and destination ip addresses in the firewall rule list. ip addresses of data packets are allowed can able to attack our system that packets transferring protocol count as vulnerability. Then optimized firewall rule list for security.

B. *Calculating Risk of vulnerability*

Using captured record in this module we can draw attack graph of vulnerabilities. Byasian network attack graph technique is used to design network attack graph. By using that network graph we can apply probabilistic reasoning to produce a risk measurement of vulnerability.

C. *Ranking the vulnerability*

In this module we can use Collaborative filtering technique is used for ranking the vulnerability.

V. CONCUSION AND FUTURE SCOPE OF INHANCEMENT

In this project we design the security model for zero day attack. We are able to catch the total count of known and dynamic vulnerabilities in network which affect our system security. In previous system we are not able to calculate the risk of vulnerability as well as not able to rank the vulnerabilities for network hardening, this system provide this function. In this model we are using collaborative filtering for ranking vulnerabilities. In this model we are design practical model for firewall system. We configure optimal list of firewall rule list to make our system more secure and find the known as well as unknown and dynamic vulnerabilities in network.

The scope of our metric is limited by the three basic assumptions about zero-day vulnerabilities (the existence of network connectivity, vulnerable services on destination host, and initial privilege on source host). The model will be more suitable for application to the evaluation of penetration attacks launched by human attackers or network propagation of worms or bots in mission critical networks. An important future work is to broaden the scope by accommodating other types of attacks (e.g., a time bomb which requires no network connection).

REFERENCES

[1] P. Mell, K. Scarfone, and S. Romanosky, "Common Vulnerability Scoring System," IEEE Security and Privacy, vol. 4, no. 6, pp. 85-89, Nov./Dec. 2006.(24)
[2] MITRE Corp., "Common Weakness Scoring System (CWSS)," http://cwe.mitre.org/cwss/, 2010.(37)
[3] M. Frigault, L. Wang, A. Singhal, and S. Jajodia, "Measuring Network Security Using Dynamic Bayesian Network," Proc. Fourth ACM Workshop Quality of Protection (QoP '08), 2008.(9)
[4] Kaur, R.; Singh, M., "Efficient hybrid technique for detecting zero-day polymorphic worms," Advance Computing Conference (IACC), 2014 IEEE International ,pp.95,100, 21-22 Feb. 2014.
[5] J. Homer, X. Ou, and D. Schmidt, "A Sound And Practical Approach to Quantifying Security Risk in Enterprise Networks," technical report, Kansas State Univ., 2009.(12)
[6] R. Lippmann, K. Ingols, C. Scott, K. Piwowarski, K. Kratkiewicz, M. Artz, and R. Cunningham, "Validating and Restoring Defense in Depth Using Attack Graphs," Proc. IEEE Conf. Military Comm. (MILCOM' 06), pp. 981-990, 2006.(20)
[7] N. Poolsappasit, R. Dewri, and I. Ray, "Dynamic Security Risk Management Using Bayesian Attack Graphs," IEEE Trans. Dependable Secure Computing, vol. 9, no. 1, pp. 61-74, Jan. 2012.(31)

[8] L. Wang, S. Jajodia, A. Singhal, and S. Noel, “k-Zero Day Safety: Measuring the Security Risk of Networks against Unknown Attacks,” Proc. 15th European Conf. Research Computer Security (ESORICS ’10), pp. 573-587, 2010.(41)

[9] Mohammed, M.M.Z.E.; Chan, H.A; Ventura, N.; Pathan, A-S.K., "An Automated Signature Generation Method for Zero-Day Polymorphic Worms Based on Multilayer Perceptron Model," Advanced Computer Science Applications and Technologies (ACSAT), 2013 International Conference on , vol., no., pp.450,455, 23-24 Dec. 2013

[10] Alosefer, Y.; Rana, O.F., "Predicting client-side attacks via behavior analysis using honeypot data," Next Generation Web Services Practices (NWeSP), 2011 7th International Conference on Next Generation Web Services Practices, pp.31,36, 19-21 Oct. 2011.

[11] D. Balzarotti, M. Monga, and S. Sicari, “Assessing the Risk of Using Vulnerable Components,” Proc. ACM Second Workshop Quality of Protection (QoP ’05), pp. 65-78, 2005.