3TH INTERNATIONAL CONFERENCE ON COMBINATORICS, CRYPTOGRAPHY AND COMPUTATION
Iran University of Science &Technology

Proceedings of the 3nd International Conference on Combinatorics, Cryptography and Computation (I4C2018)

Innovative Cyber-Security Metrics for Intrusion Prevention

Marjan Keramati, Fatemeh Sadat Halataei
Semnan University , Department of Computer Science
Iran, Semnan, Semnan
Keramati_marjan@semnan.ac.ir; halataei@semnan.ac.ir

ABSTRACT

Network services are becoming more developed and sophisticated to maintain secure. It is extremely critical to preserve system's availability and user's quality of experience. Also it is noticeable that, attackers are becoming more intelligent too. Consequently, it seems necessary to do intrusion prevention in computer networks. Because of the inevitable parameter cost, network hardening should be done in a cost effective manner. So, applying the best security measure is not possible but with being provided with some security metrics for finding the most perilous attacks and also a criteria for security improvement assessment when applying different countermeasures. In this paper we proposed an intrusion prevention model which makes finding the most disastrous vulnerabilities and security improvement estimation possible. Also, one cost model is proposed which helps the security administrators to quantify the cost of possible network hardening methods.

KEYWORDS: Intrusion Prevention, Minimum Cost Network Hardening, Vulnerability, Security Metric, patch, exploit.

1 INTRODUCTION

Nowadays, malicious incidents are known to be the most hazardous threats to both human's life and different political, economic, social and…..organizations. Attackers usually endanger the confidentiality and integrity of data or the availability of some critical services. As a result, it seems crucial to harden computer systems against cyber-attacks.

Securing computer services should be done by considering the trade-offs between attacks damage, security costs and security benefits. Killing a process or adapting the firewall configuration are two instances of security measures. Employing security measures brings cost to the network both in terms of financial and non-financial ones. So, the security administrator has to be cautious about doing cost benefit trade-off while performing network immunization. For instance, when one attack with high damage level is progressing, a strong countermeasure has to be applied immediately. Consider an attacker is detected in the system, trying to compromise main databases in one organization. In such a case, "Blocking All Traffic by Firewall" or "Disabling the Database" seems reasonable (Shameli-Sandi et al., 2017)

Three fundamental questions come into mind when tackling abovementioned issues: “how to find the most disastrous potential incidents?”, “how to estimate the cost of preventing vulnerabilities exploitation?” and “how to measure the security benefits of applying possible countermeasures?”.

The main goal of this paper is to answer the mentioned questions. To come up with the challenges we need some quantifiable security metrics. The main difficulties with some existing solutions are that, usually security metrics are not quantifiable that makes them impractical in real world. The problem is seen in both standard common vulnerability scoring systems such as CVSS (Web-1) and CWE (Web-2) and various academic attempts. Secondly and more importantly, the existing methods usually do risk assessment regardless of the temporal features of vulnerabilities which makes security assessment inaccurate.

The proposed security assessment method outweighs the similar ones in below terms:

- Introducing quantifiable security metrics
- Dynamic risk assessment of vulnerabilities
- Defining cost model

Also, introduced method considers isolated security metrics for below purposes:

- System readiness for disaster prevention
- Vulnerabilities elimination priority
- Risk assessment of vulnerabilities

This paper is organized as follows: first there is a brief review on some similar works on security assessment. The proposed method is discussed in section 3. The results of applying the suggested method on one popular software is shown in section 4. After declaring the limitations in 5, we conclude in 6.

2 RELATED WORKS

Recently, several works have been done in the fields of defining security metrics, incident’s risk assessment, and intrusion response and prevention systems. Here we have only a short review on some of them.

There are points that should be considered in defining security metrics. Good metrics can be measured consistently, are inexpensive to collect, expressed numerically, have units of measure, and have specific context (Jaquith, 2007). One of the standardization efforts in security assessment is the Common Vulnerability Scoring System (CVSS) (Web-1). Two main problems with CVSS are: qualitative nature of their scores and neglecting temporal features of vulnerabilities in risk estimation.

Efforts like (Islam, Wang 2008) (Noel et al., 2003)are instances of researches in minimum cost network hardening. Proposed methods are based on attack graph and they try to find the best result by traversing it. But, finding the optimal solution scales exponentially with the size of the attack graph (Albanese et al., 2012). So, in (Islam, Wang 2008) authors tried to find the near optimal solution to reduce this complexity. Ref (Albanese et al., 2012) is one of the most efficient efforts in this area as it proposed a methodology that can find a near optimal solution in linear time. (Web-2) is the collection of the authors achievements in minimum cost network hardening containing (Jaquith, 2007).As the main shortcoming with (Islam ,Wang 2008), (Noel et al., 2003), (Albanese et al., 2012)is their inability for measuring the amount of security improvement for each optimal solution. So, doing cost-benefit trade-off is impossible.

Authors in (Aziz et al., 2012), present some ideas on defining and implementing a new Cyber-security risk metric for measuring the readiness of organizations, in terms of the availability of their resources in dealing with new attack incidents launched against their infrastructures whilst recovering from ongoing incidents. Their new metric, the Mean Blind Spot, is defined as the average interval between the recovery time of an existing incident and the occurrence time of a new incident.

In (Shameli-Sandi et al., 2017), they proposed a new model to combine the Attack Graph and Service Dependency Graph approaches to calculate the impact of an attack more accurately compared to other existing solutions. To show the effectiveness of their model, a sophisticated multistep attack was designed to compromise a web server, as well as to acquire root privilege.

In (Wang et al., 2012), a novel, precious security metric is defined formally for ranking unknown vulnerabilities in computer networks. In (Nzoukou et al., 2013), an approach is proposed for measuring a network's mean time-to-compromise by considering both known and zero day attacks. Specifically, it first devises models of the mean time for discovering and exploiting individual vulnerabilities. Then it employs Bayesian networks to derive the overall mean time-to-compromise by aggregating the results of individual vulnerabilities. Approach in (Ghani et al., 2013), targets the quantitative understanding of vulnerability severity, taking into account the potential economic damage a successful vulnerability exploit can cause. their approach utilizes the Multiple Criteria Decision Analysis (MCDA) methods to perform a prioritization of the existing vulnerabilities within the target system. In (Keramati 2017), we proposed a framework for vulnerability risk assessment. The introduced method is an improvement over CVSS. Considering the temporal features of vulnerabilities in probability estimation and developing a novel method for Impact assessment of CVSS version 2 are two most noticeable novelty of the mentioned vulnerability ranking system.

In (Keramati et al., 2011), we proposed an attack graph based security metric for risk evaluation of possible attacks in computer networks. The approach can be used for doing cost-benefit trade-off in network hardening. The main problem with the most above mentioned approaches is that, their defined security metrics are unmeasurable. In this paper we tried to fill this gap as much as possible.

3 PROPOSED METHOD

The introduced model is designed in a way to answer below basic questions:

1. How to find the most disastrous potential incidents?
2. How to measure the security benefits of applying possible countermeasures?
3. How to estimate the cost of preventing vulnerabilities exploitation?

The mentioned questions have been answered in this paper by defining some quantifiable security metrics. These security metrics are declared in the following sub-sections

3.1 Finding the most disastrous potential incidents

A vulnerability is defined as software defect or weakness in the security system which might be exploited by a malicious user causing loss or harm (Alexander, 2008). Eliminating all vulnerabilities of the network can harden the network completely. But in practice this idea cannot be applicable. This is because, eliminating vulnerabilities itself can bring the system into undesirable conditions that are described in continuance. For removing the vulnerabilities, we can either patch the vulnerability or eliminate the conditions that are required for exploiting it. The most straightforward way is patching the vulnerabilities but, unconfirmed patches may bring the system into instability and introduce more bugs. Thirdly, patching on OS kernel level often needs the system to be rebooted, and some organizations are intolerant of affecting availability (Wang 2012). This is an example of an undesirable condition occurs as a result of removing one vulnerability in the network. In the field of network security, these unwanted effects are referred to as cost. So, I should say that because of cost, in doing intrusion prevention it is impossible to harden the network completely and we have to do minimum cost network hardening.

By considering the above fact, here we defined some security metrics that quantify following terms:

- System's Intrusion Prevention readiness
- Average frequency of a specific vulnerability in a computer system

System's Intrusion Prevention readiness. We defined the Intrusion prevention readiness of a system, as the capability of the system in remediating potential attacks before their occurrence in the system. We quantified this concept by the security metric in (1)

$$Intrusion\ Prevention\ Readiness = \frac{MTTR}{MTTE} \quad (1)$$

$$MTTR(Vul_i) = Date\ of\ \ Patch\ Release - Date\ of\ Exposure \quad (2)$$

$$MTTE(Vul_i) = Date\ of\ \ Exploit\ Tool\ Release - Date\ of\ Exposure \quad (3)$$

In (1), $MTTR(Vul_i)$ (Mean Time to Recovery) is calculated by counting the number of days between the day of vulnerability exposure to the public and the day of patch release. Also, $MTTE(Vul_i)$ is measured by counting the number of days between the day of vulnerability exposure to the public and the day of exploit tool introduction.

Quantifying the above security metrics is not possible without being providing with comprehensive databases of vulnerabilities containing thorough information about vulnerabilities exposure dates, patch release dates and exploit release dates and applying data mining methods. So, we focused on probabilistic methods for making estimation on the three mentioned dates (exposure, patch release, exploit introduction).

Various probabilistic attempts have been done in making predictions about different characteristics of vulnerabilities. They are especially important when performing risk assessment of Zero-Day vulnerabilities. Papers such as (Shabana Janani, 2014), (Chatzipoulidis, 2015) are two valuable examples. But the problem is that, the suggested probabilistic distribution functions are application dependent and the best fit for distribution parameters was found for a few discrete applications. So, as these models are not generalized we didn't utilize their ideas. But, (Frei et al., 2006) is a precious work that provided a statistic model for estimating the probability of patch release date and exploit tool introduction. They proved that, Weibull Cumulative Distribution Function with assigned parameters in (4) is a reasonable match for the probability of Patch availability after disclosure.

$$F(x) = 1 - \exp\left(-\frac{x}{\lambda}\right)^k . \qquad (4)$$
$$k = 4.040, \quad \lambda = 0.209$$

Also, they demonstrate that Exploit availability before and after disclosure is found to be best matched with a Pareto distribution as parametrized in (5).

$$F(x) = 1 - \left(\frac{k}{x}\right)^\alpha . \qquad (5)$$
$$k = 0.00161\,, \quad \alpha = 0.260$$

In (4), (5), x is the age of the vulnerability that is calculated by counting the days between the date of the first disclosure and the date the CVSS Scoring is conducted (for example Today)

Here we made use of the above findings to quantify $MTTR(Vul_i)$ and $MTTE(Vul_i)$. We analyzed that, as it is shown in (6), (7), $MTTR(Vul_i)$ has indirect relationship with the probability of patch existence. Also $MTTE(Vul_i)$ has indirect relationship with the probability of exploit tool availability.

$$MTTR(Vul_i) \propto \frac{1}{patch\ \ Introduction\ Probability} \qquad (6)$$

$$MTTR(Vul_i) \propto \frac{1}{Exploit\ Tool\ Availability\ Probability} \qquad (7)$$

3.2 Average frequency of a specific vulnerability in a computer system.

Vulnerabilities are not the same in terms of lifespan, the speed of remediation and the degree of exploitability. But security specialists have always been struggling to introduce patch for vulnerabilities as soon as possible. But, there are various vulnerabilities that is no patch for them. So, they remain in the software in subsequent versions too. Also, some vulnerabilities are not limited to only one software product and it exists in more than one software in a system simultaneously.

We considered this fact and defined security metric in (8) that reflects the prevalence level of each vulnerability across different version and products. In (8), NOP is the number of products Vul_i exists in. Also, NOV_j specifies the number of versions of product j that contains Vul_i.

$$prevalence\ level(Vul_i) = \frac{\sum_{j=1}^{NOP} NOV_j}{NOP} \qquad (8)$$

By combining security metrics in (1), (8) in (9) we can compare vulnerabilities urgency's for elimination.

$$Vulnerability\ Urgency\ Degree = \frac{prevalence\ level(Vul_i)}{Intrusion\ Prevention\ Readiness(Vul_i)} \quad (9)$$

3.3 Measuring the security benefits of applying possible countermeasures.

As it is emphasized before, network hardening should be done in a cost effective manner. So, estimating the security improvement of applying different security measures for intrusion prevention seems to be necessary. Here we suggested security metric in (10) for this purpose.

$$Security\ Improvement = Risk(After\ Hardening\ ) - Risk(Before\ Hardening\ ). \quad (10)$$

According to (Joh , Malaiya, 2011), the formal definition of risk is shown in (11):

$$Risk = Likelihood\ of\ an\ adverse\ event \times Impact\ of\ the\ adverse\ event\ . \quad (11)$$

$$Risk(Network) = \sum_{i=1}^{n} Risk(Vul_i) \qquad (12)$$

In (11), adverse event means vulnerability exploitation. Here we defined and aggregate security metrics for quantifying both attack occurring probability and its impact on Confidentiality, Integrity and Availability. The risk level of each network is measurable by (12)

In CVSS, for each indexed vulnerability with CVE identifier (Web-3), there is a parameter called Base Score, which reflects the risk of the vulnerability based on its intrinsic features. Base Score can be compared with the risk parameter in (11) as it is calculated by estimating the likelihood of attack occurring and the Impact of attack occurring on the three mentioned security parameters.

But, there are some basic drawbacks in CVSS which makes risk estimation inaccurate. In (Keramati, 2017), we went through the risk estimation process in CVSS and extracted its problems.

Also in (Keramati 2017), we introduced a new vulnerability scoring system for risk assessment of single vulnerabilities with the perspective of filling the gaps of CVSS Scoring system. Some main challenges with CVSS Scoring System are:

- It ignores the temporal features of vulnerabilities in probability estimation
- It is incapable of ranking multi-step attacks in the network
- Scores diversity is so limited in CVSS
- The symmetric nature of Impact parameter makes risk estimation inaccurate.

CVSS Base Score is function of Intrinsic Exploitability of the vulnerability and its impact of exploitation on security parameters. Here we considered the challenges and proposed a model for risk estimation. Security metric in (14) is defined for probability assessment.

$$Exploitability\ Degree(Vul_i) = \frac{Number\ of\ \ exploits\ (Vul_i)+1}{Exposure\ Time\ (Vul_i)} \quad (13)$$

$$prob(Vul_i) = \frac{CVSS\ Exploitability\ (Vul_i)}{10} \times\ Exploitability\ Degree(Vul_i) \quad (14)$$

Impact Estimation of Vul_i is done by the method we introduced in (Keramati 2017). The proposed method improves both score's diversity and accuracy.

The most considerable point about the defined security metric is that, they can be measured quantifiably. All the required information is available in vulnerability and exposure systems like CVSS (Web-1), CWE (Web-2), CWSS (Web-3) and etc.

3.4 Estimating the cost of preventing vulnerabilities exploitation

As it is mentioned in the previous section, we can prevent one vulnerability exploitation by:

- Employment of patch
- Removing the necessary conditions for its exploitation

Also, it is obvious that, applying security measures undoubtedly brings cost to the system. For example, patching on OS kernel level often needs the system to be rebooted, and some organizations are intolerant of affecting availability (Wang 2012).

Vulnerabilities in computer systems are exploited only if necessary conditions are provided. By considering such a fact, we suggested a model for cost estimation in (15).

$$CostofRemoving(Vul_i) = Cost\ of\ Patch(Vul_i) + Cost\ of\ Removing\ Conditions(Vul_i) \quad (15)$$

$$Cost\ of\ Removing\ Conditions(Vul_i) = \sum_{j=1}^{n} \frac{1}{ConditionSetSize_j} \quad (16)$$

In (16), n is the number of discrete sets that make exploiting the vulnerability possible. Each condition set consists of some requirements such as the existence of some service on the system and some user access levels. It should be noticed that, for each condition set to be effective, all its member should exist simultaneously. Because of this "and" relationship between the members the cost is in inverse relation with the condition set size. $ConditionSetSize_j$ Is the number of members in set number j.

Note that quantifying the security metric in (15) requires security administrator's collaboration. This is because $Cost\ of\ Patch(Vul_i)$ varies from one network to another based on the network's security policy. But, $Cost\ of\ Removing\ Conditions(Vul_i)$ can be extracted easily from some security databases like cvedetails (Web-5).

.

4 EXPERIMENTAL RESULTS

The proposed model can be used in computer network system. Here we only applied the method on one commonly used software, Vmware. There are 18 vulnerabilities in Vmware 2013. Defined security metrics have been calculated for these 18 vulnerabilities. Results are shown in Table 1.

$Risk$ Reflects the amount of security improvement occurs as a result of removing the associated vulnerability.

5 LIMITATIONS

It is worth mentioning that the real attacks in computer networks are multi-step attacks in which an attacker exploits more than one vulnerability in a specified manner to reach his goal. Our method is only capable of dealing with single vulnerabilities.

Secondly and more importantly, there is a lack of an efficient algorithm in the process of response selection in a cost effective manner.

The cost model neglects the chain of vulnerabilities or the sequence of vulnerabilities that must be exploited in a specified manner for intruding the network.

Thirdly and finally our model does not improve any specific Intrusion Prevention System. It only tries to recommend a model to find more disastrous attacks, and provide the security administrators with a tool to quantify the security improvement level of applying different security measures.

6 CONCLUSION

In this paper we introduced a model for intrusion prevention. A network immunization system is efficient when performs network hardening in a cost effective manner. Decrement of quality service as a result of limiting user access to the services and unavailability of critical systems in the period of patch employment are two examples of cost.

In the process of performing minimum cost network hardening, it is necessary to firstly find the most perilous attacks and then analyse the effectiveness of possible countermeasures. Estimating the security benefit of applying security metrics is critical when considering cost-benefit trade-off in the network hardening process.

In this paper we defined some quantifiable security metrics which makes finding the most dangerous vulnerabilities and estimating security benefit level possible. We also suggested a cost model that helps security administrators to quantify the cost of removing vulnerabilities.

In the future we are going to generalize our system to perform network hardening against multistep attacks. In other words, we are going to utilize security models to make this goal possible.

REFERENCES

Albanese M., Jajodia S., and Noel S., (2012). “Time-Efficient and Cost-Effective Network Hardening Using Attack Graphs,” in Proceedings of the 42nd Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN 2012), Boston, Massachusetts, USA, June 25-28.

Alexander C. (2008),” Market Risk Analysis: Quantitative Methods in Finance”, Wiley.

Aziz B., Malik A., Jung J. (2017) Check Your Blind Spot: A New Cyber-Security Metric for Measuring Incident Response Readiness. In: Großmann J., Felderer M., Seehusen F. (eds) Risk Assessment and Risk-Driven Quality Assurance. RISK 2016. Lecture Notes in Computer Science, vol 10224. Springer, Cham.

Chatzipoulidis A., Michalopoulos D., Mavridis I., (2015) “Information infrastructure risk prediction through platform vulnerability analysis”, The Journal of Systems and Software 106 (2015) 28–41.

Frei, S. & May, S. & Fiedler, U. & Plattner, B. (2006). Large-scale vulnerability analysis. LSAD ’06: Proceedings of the 2006 SIGCOMM workshop on Large-scale attack defense, pp. 131–138.

Ghani H., Luna J., & Suri N. (2013). “Quantitative assessment of software vulnerabilities based on economic-driven security metrics”. International Conference on Risks and Security of Internet and Systems, 1-8.

Islam, T., and Lingyu Wang. (2008). "A Huristic Approach to Minimum Cost Network Hardening Using Attack Graphs." New Technologies, Mobility and Security. IEEE, 1-5.

Jaquith, (2007). “Security Metrics: Replacing Fear, Uncertainty, and Doubt”, Addison Wesley Publication.

Joh H., & Malaiya Y. K. (2011). Defining and Assessing Quantitative Security Risk Measures Using Vulnerability Lifecycle and CVSS Metrics. Proc. Int. Conference on Security and Management, 10-16, 2011.

Keramati M., (2017). “Dynamic Risk Assessment System for the Vulnerability Scoring.” International Journal of Information & Communication Technology Research, 9(4), 57-68.

Keramati M., Asgharian H. and Akbari A. (2013), "Cost-aware network immunization framework for intrusion prevention," 2011 IEEE International Conference on Computer Applications and Industrial Electronics (ICCAIE), Penang, pp. 639-644.

Noel, Steven, Sushil Jajodia, Brian O'Berry, and Michael Jacobs. (2003), "Efficient Minimum-Cost Network Hardening Via Exploit Dependency Graphs." 19th Annual Computer Security Applications Conference. IEEE Computer Society, 2003. 86-92.

Nzoukou W., Wang L., Jajodia S., & Singhal A. (2013). A unified framework for measuring a network's mean time-to-compromise. Proc. 32nd Int'l. Symp. on Reliable Distributed Systems (SRDS), 215-224.

Shabana Janani C. (2014),” Distribution Fitting for Zero Day Vulnerability Life Spans: A Quantitative assessment with reference to Weibull Distribution”, Great Lakes Herald , 8 (2014) 59-69.

Shameli-Sendi A., Dagenais M., and Wang L.(2017). “Realtime Intrusion Risk Assessment Model based on Attack and Service Dependency Graphs”. In Computer Communications, 253–272.

Wang C., Bao Y., Liang X, Zhang T. (2012).” Vulnerability Evaluating based on attack graph”, International Conference, ISCTCS 2012,pp 555-563.

Wang L., Jajodia S., Singhal A. & Noel S. (2014) .k-Zero Day Safety: A Network Security Metric for Measuring the Risk of Unknown Vulnerabilities. IEEE Trans. Dependable Sec. Comput, 11(1), 30-44.

Web sites:

Web-1: http://www.first.org/cvss/, consulted 7, September, 2018

Web-2: http://cwe.mitre.org/, consulted 7, September, 2018

Web-3: http://cve.mitre.org/, consulted 7, September, 2018

Web-4: http://cwe.mitre.org/cwss/cwss_v1.0.1.html, consulted 7, September, 2018

Web-5: https://www.cvedetails.com/, consulted 7, September, 2018

TABLE I. THE RESULTS OF APPLYING THE PROPOSED METHOD ON VMWARE 2013

		Intrusion Prevention	*prevalence leve*	*Vulnerability Urgency Degr*	*Risk*
1	CVE-2013-6366	0.9731	1	0.9731	0.7747
2	CVE-2013-5973	0.9729	8	7.7832	0.1646
3	CVE-2013-5972	0.9731	3	2.9193	0.2999
4	CVE-2013-5971	0.9732	9	8.7588	0.6613
5	CVE-2013-5970	0.9732	2.5	2.4330	0.0735
6	CVE-2013-3658	0.9732	5.5	5.3526	0.2563
7	CVE-2013-3657	0.9733	5.5	5.3532	0.4842
8	CVE-2013-3520	0.9737	7	6.8159	0.9684
9	CVE-2013-3519	0.9738	3.2	3.1162	0.4229
10	CVE-2013-3107	0.9738	1	0.9738	0.0980
11	CVE-2013-3080	0.9738	1	0.9738	0. 6152
12	CVE-2013-3079	0.9738	1	0.9738	0. 6152
13	CVE-2013-1662	0.9734	11.5	11.1941	0.5229
14	CVE-2013-1661	0.9734	7	6.8138	0.0490
15	CVE-2013-1659	0.9740	8	7.7920	0.3768
16	CVE-2013-1406	0.9741	7.8	7.5980	0.2999
17	CVE-2013-1405	0.9741	3.33	3.2438	0.7690
18	CVE-2012-6326	0.9740	4.5	4.3830	0.0854