IJICTR
International Journal of Information & Communication Technology Research

Research Note

Volume 9- Number 4 – Autumn 2017 (57-68)

Dynamic Risk Assessment System for Vulnerability Scoring

Marjan Keramati *
Computer Science Department
Semnan University
Semnan, Iran
Keramati_marjan@semnan.ac.ir

Received: February 18, 2017 - Accepted: September 19, 2017

Abstract—**One of the key factors that endangers network security is software vulnerabilities. So, increasing growth of vulnerability emergence is a critical challenge in security management. Also, organizations constantly encounter the limited budget problem. Therefore, to do network hardening in a cost-benefit manner, quantitative vulnerability assessment for finding the most critical vulnerabilities is a vital issue. The most prominent vulnerability scoring systems is CVSS (Common Vulnerability Scoring System) that ranks vulnerabilities based on their intrinsic characteristics. But in CVSS, Temporal features or the effect of existing patches and exploit tools in risk estimation of vulnerabilities are ignored. So, CVSS scores are not accurate. Another deficiency with CVSS that limits its application in real networks is that, in CVSS, only a small set of scores is used for discriminating between numerous numbers of vulnerabilities. To improve the difficulties with existing scoring systems, here some security metrics are defined that rank vulnerabilities by considering their temporal features beside their intrinsic ones. Also, by the aim of improving scores diversity in CVSS, a new method is proposed for Impact estimation of vulnerability exploitation on security parameters of the network. Performing risk assessment by considering the type of the attacker which endangers the network security most is another novelty of this paper.**

Keywords- CVSS; Risk; Vulnerability; Impact; Network Hardening; Security Metric; exploit; patch

I. INTRODUCTION

These days with widespread utilization of computer networks in financial systems, economics, airports and because of the existence of the vast number of threatening attacks in computer systems, improving network security level has become one of the most urgent requirements of the human life.

This necessity requires thorough understanding of the source of attacks in networks. Attacks occur because of vulnerabilities in software systems. In computer systems, vulnerability means a bug, a flaw, a weakness, or an exposure of an application, system, device, or service which could lead to a failure of Confidentiality, integrity or availability [1].

As an inherent part of each service is its vulnerabilities or potential attacks, in order to increase the security level of each organization, estimating the risk of vulnerabilities in a quantitative manner should be considered as one of the most important challenges in improving network security.

Currently, security administrators suffer from the lack of comprehensive standards for measuring the risk of vulnerabilities. Some valuable but limited systems are now available that describe and sometimes rank various aspects of each vulnerability.

One widely used example is Common Vulnerability Scoring System or CVSS [1]. CVSS is an open framework for ranking vulnerabilities from three perspectives, intrinsic qualities of vulnerability or Base

* Corresponding Author

Group, Temporal group which is characteristics of each vulnerability that change over time and Environmental group that scores features of each vulnerability that is unique to the user's environment. On the other hand, CVSS that is now maintained by FIRST can be considered as the most prominent and usable vulnerability scoring system.

CVSS, provides users with an aggregate score based of the inherent features of the vulnerability that reflects the severity of it in both quantitative and qualitative manner.

But, despite of its broad viewpoint to different features of vulnerabilities, CVSS suffers from some serious weak points that are explained in detail in the subsequent sections.

One of the CVSS challenges that is addressed in this paper is that, CVSS Calculator can only calculate Base Score Group of Vulnerabilities. This violation leads to inappropriate risk evaluation. This is because, the threat posed by a vulnerability may change over time and it is crucial for the security manager to have an exact estimation of the vulnerability's risk to remediate those vulnerabilities which impose the greatest danger to the organization.

Another most known approach for scoring vulnerabilities is Common Weakness Enumeration or CWE [2] that is a community developed dictionary of software weakness types. In comparison with CVSS, CWE in conjunction with Common Weakness Scoring System (CWSS) provides more comprehensive and detailed information about Temporal and Environmental characteristics of each vulnerability [3]. So, it seems that these scoring systems may be more effective in risk estimation. But, the problem with them is that, CWE has only a qualitative description of vulnerabilities. So, this negative point prevents the CWE to measure the risk of vulnerabilities quantitatively. In contrast, CWSS, in spite of having a quantitative description of vulnerabilities from different viewpoints (intrinsic, temporal and Environmental features) lacks the calculator. So it is not practically applicable in vulnerability scoring process.

Generally, one of the major problems with existing systems is that, in security ranking, they cannot reflect the variable feature of a vulnerability over time. So, the accuracy of these systems is low. This claim is true because, introducing patch for a vulnerability or developing methods of exploiting it over time changes the risk of vulnerability.

Accurate scoring is a serious issue. This is because, limited budget has been always one of the permanent challenges with each organization. So, hardening the network in a cost effective manner is a vital need that can be achieved by exact classification of vulnerabilities to find the most dangerous ones.

By considering this challenge, in this paper a novel method for scoring vulnerabilities is introduced that scores each known vulnerability by an overall score composed of both intrinsic and temporal features of the vulnerability such as probability of exploitability tool availability and patch presence. So, by the use of the proposed method, the score changes over time to more accurately evaluate the risk of each vulnerability. Improving scores diversity is another result of developing our scoring system. This scoring system is composed of some existing and newly defined security metrics that can be measured quantitatively.

Improvements to our system over CVSS is that, the proposed method:

- Is more accurate because of considering temporal features of each vulnerability in scoring
- Performs dynamic risk evaluation of vulnerability that changes over time.
- More accurately assesses the impact of each vulnerability by considering the relative importance of the three security parameters (Confidentiality, Integrity and Availability).
- Has more diversity than CVSS in Vulnerability Scoring
- Reduces the False Positive and False Negative rates in vulnerability Scoring
- Perform risk evaluation by considering the type of the attacker that threaten the system most.

In the following, after a brief review on some related works, CVSS is introduced and its challenges are discussed. In section IV, the proposed method is introduced and finally, after investigating the effectiveness of our method, the results of applying our framework to one widely used service are shown in section VI.

II. RELATED WORKS

Currently, there are numbers of standard and non-standard security ranking systems that differ in what they measure and the method of scoring (qualitative or quantitative).

Standard Works

Vulnerability Scoring Systems are divided into qualitative and quantitative forms. In addition to some mentioned above examples in the Introduction section, few other systems are as below.

CERT/CC is one example that assigns a score between 0 to 180 to the security level by considering items such as whether the infrastructure is at risk and what kinds of preconditions are required to exploit the vulnerability [4].

Microsoft also has a vulnerability scoring mechanism in the Microsoft Security Response Center Security Bulletin Severity Rating System [5]. Microsoft's proprietary scoring system describes the difficulty of exploitation and the overall impact of the vulnerability. This system can assign each vulnerability's severity in four different levels: Critical, Important, Moderate or Low.

XForce is the IBM's scoring system that scores vulnerabilities by a qualitative manner in three levels: High, medium and low [6].

The other qualitative system is the Symantec Security Response Threat Severity Assessment from Symantec that its scores are in five levels [7]. As mentioned above, US-CERT and CVSS are two examples of quantitative systems. Mozilla Foundation has also its own qualitative vulnerability rating system with four levels of severity [8].

One of the most important problems with these systems is that, they ignore the effect of environmental and time dependent features in measuring the impact of vulnerabilities.

The most widely used scoring systems is CVSS that scores vulnerabilities both quantitatively and qualitatively. This system will be explained in more details in the next sections.

Nonstandard works (Academic works)

Recently, many valuable efforts have been done in the field of vulnerability scoring and defining security metrics. Here we have only a brief review on some more related and recent ones.

In [9], a new potential value loss metric was proposed for rating vulnerabilities. The main purpose in [9] is to improve score diversity and vulnerability distribution evenness in comparison to CVSS.

Reference [10] has introduced a new method for assessing the severity of each host in the network as the total weights of its vulnerabilities using CVSS base, temporal and environmental Sub-Scores. They did that by combining related sub-scores and modeling problem parameters in a mathematical framework.

In [11], a framework is introduced for software risk evaluation with respect to the vulnerability lifecycle.

The method in [12] is a new scoring system with the aim of advancing a new approach to measure the severity cost for each host by combining CVSS sub-scores. The work in [12] is the vulnerability scoring system that has better accuracy than CVSS. Because, it considers temporal features of vulnerabilities in prioritizing them by calculating the probability of implementing exploits and patch development for vulnerabilities over time. It is claimed that, their approach is applicable in doing network hardening in a cost effective manner.

In [13], a set of polynomial approaches is proposed for measuring the level of k-zero day safety in networks by analyzing their vulnerabilities. In [14], a novel, exact security metric is defined formally for ranking unknown vulnerabilities in computer networks.

Paper in [15] has an innovation in improving the accuracy of CVSS framework by changing and correcting its formulas and considering environmental factors in scoring vulnerabilities. Authors in [17] have developed a security rating system with considerable better diversity than CVSS. But the problem with some systems like [15] and also [17] is that, they rank vulnerabilities despite of temporal and environmental features of them.

Authors in [18] proposed a new quantitative vulnerability scoring system that used the idea of normal distributions to improve the vulnerability scoring system. Reference [18] also considers the user security requirement which is dependent on the organization's context.

In this paper by the aim of improving the mentioned weaknesses of CVSS (Lack of accuracy and diversity), one framework has been implemented for scoring each indexed vulnerability. The paper is the extension of the vulnerability scoring system we proposed in [24].

III. CVSS AND ITS CHALLENGES

In this section, after a short review of CVSS, some of its challenges are discussed.

A. A brief review of CVSS

CVSS is the most widely used vulnerability scoring system. The most substantial feature with CVSS is the existence of the Calculator.

Currently, Version 2 of CVSS calculator is available and can be used for risk estimation of all indexed vulnerabilities. Also, Version 3 is being developed from the perspective of improving the Version 2's difficulties. Now the version 3 of CVSS calculator is only useable for risk estimation of the limited number of vulnerabilities (vulnerabilities, which are indexed from 2016 and in some cases for the ones which are indexed from 2011) but it is expected to be developed for risk estimation of all indexed vulnerabilities.

So, as our vulnerability scoring system has been developed by utilizing some intrinsic features of vulnerabilities from CVSS, we did our best to make our system as much as compatible with both versions.

Before introducing our CVSS based security metrics, we have a brief review of CVSS system and both of its calculators.

CVSS estimates the risk of each known vulnerability by considering three following characteristics of it [1]:

- **Base Score Group**: is reflective of the intrinsic and fundamental characteristics of a vulnerability that are constant over time and among user environments.
- **Temporal Group**: represents the characteristics of a vulnerability that change over time, but not among user environments.
- **Environmental Group**: demonstrates the characteristics of a vulnerability that are relevant and unique to a particular user's environment.

CVSS Calculator takes as input the identifier for each indexed vulnerability and determines the score of the vulnerability by considering the mentioned different factors. This identifier is called CVE. CVE is a dictionary of publicly known information security vulnerabilities and exposures [19].

In Version 3 by the aim of amending Version 2 scoring system uses different sub metrics in the three mentioned groups.

One important weak point with CVSS is that, Temporal and Environmental Groups of CVSS are not scored in CVSS. So, vulnerability scoring cannot be done accurately by using CVSS. Also, only intrinsic features of vulnerabilities can be extracted from CVSS.

So, in this paper, we only focused on Base Score group. In the following the sub metrics of Base Score Group of both versions are provided.
Base Score Group of CVSS consists of below metrics: [1]

- **Exploitability**
  In Version 2, this metric consists of following parameters:
  - ✓ **Access Vector**: This metric reflects how the vulnerability is exploited. The more the remote an attacker can be to attack a host, the greater the vulnerability score.
  - ✓ **Access Complexity**: This metric measures the amount of complexity required to exploit the vulnerability once an attacker has gained access to the target system.
  - ✓ **Authentication**: This metric measures the number of times an attacker must authenticate to a target in order to exploit a vulnerability.

  Version 3 sub parameters of this metric are as below too:
  - ✓ **Access Vector**: This metric reflects how the vulnerability is exploited. The more the remote an attacker can be to attack a host, the greater the vulnerability score.
  - ✓ **Access Complexity**: This metric describes the conditions beyond the attacker's control that must exist in order to exploit the vulnerability. In version 3 in opposition to version 2 the assessment of this metric excludes any requirements for user interaction in order to exploit the vulnerability
  - ✓ **Privileges Required:** This metric describes the level of privileges an attacker must possess before successfully exploiting the vulnerability.
  - ✓ **User Interaction:** This metric captures the requirement for a user, other than the attacker, to participate in the successful compromise the vulnerable component

- **Impacts**: CVSS provides metrics that reflect the impact of exploiting the vulnerability on three security parameters (Confidentiality, Integrity, and Availability) and introduce an aggregate measure for scoring the total impact based on these three security measures.

B. *Some CVSS Challenges*

In this paper, the goal is to introduce a vulnerability scoring system by trying to improve the existing CVSS challenges.

Some of the CVSS difficulties are as below:

- Upon the last version of CVSS Calculator, the Temporal Scores group of CVSS is not defined in CVSS yet. So, CVSS cannot reflect the effect of introducing new patches and novel facilities for exploiting the vulnerability in risk estimation.
- Another difficulty with CVSS is that, only a small range of discrete values is used for scoring the huge number of vulnerabilities. So, diversity goes low and CVSS cannot discriminate between vulnerabilities efficiently. Consequently, scores become unusable for vulnerability prioritization and determining the most dangerous ones for elimination.
- According to [10], one major difficulty with the Impact Score of CVSS is that, the relative importance of three security parameters is ignored in Impact estimation. But the reality is that, Integrity Impact is more severe than the Availability Impact. This is because, violation of integrity very often affects the Availability Impact negatively. Also integrity violation is more difficult to be noticed. Similarly, Confidentiality Impact is more severe than the Integrity Impact, because the violation of Confidentiality is the hardest one to detect.

Mentioned CVSS weak points, affects the accuracy and diversity of risk scores adversely. So, in this paper, by the aim of developing a vulnerability scoring system with improved scores diversity accuracy, some mentioned CVSS weak points were tried to be amended. In the next section, the proposed method is described.

IV. PROPOSED VULNERABILITY SCORING METHOD

According to [11], the formal definition of risk is shown in (1). One interpretation of an adverse event in this definition can be vulnerability exploiting. Based on (1), by the aim of vulnerability ranking, we proposed some security metrics to measure the probability of exploiting each vulnerability in terms of its intrinsic and temporal features. Also, we proposed an approach for estimating the impact of exploiting each vulnerability on the three security parameters (Confidentiality, Integrity and Availability). Our risk estimation method has more diversity than CVSS. So, discrimination of vulnerabilities can be done more efficiently and in a more accurate manner.

$$Risk = Likelihood\ of\ an\ adverse\ event \times Impact\ of\ the\ adverse\ event\ . \quad (1)$$

In the two subsequent sections the proposed method for estimating the probability and the impact of

exploiting it on security parameters of the network are described.

Our approach has been developed in such a way to be compatible with both version 2 and 3 of CVSS calculator. This has been done by defining some CVSS based security metrics for these two versions separately.

A. *Estimating the Probability of Exploiting each Vulnerability*

As mentioned in section III, one important weak point of CVSS is that, reported risk scores are calculated based on the intrinsic features only and temporal features are ignored. This feature leads to not accurate scoring. Also, this feature can be considered as one reason of limited available scores in CVSS.

In this paper, by the aim of improving the accuracy and diversity of CVSS Scores, one new method has been proposed for dynamic probability estimation of vulnerability exploitation. This goal has become possible by assessing the probability of exploiting tool availability.

Generally, in this paper, the probability of vulnerability exploiting is measured by considering both intrinsic and temporal features of it. This estimation is done by defining some related security metrics based on number of CVSS Sub-Scores and some probability distributions.
In calculating the Intrinsic Probability of exploiting each vulnerability, below Sub-Scores of CVSS Base Score Group are utilized.

- **Access Vector:** This metric reflects how the vulnerability is exploited. The more remote an attacker can be to attack a host, the greater the score [1].
- **Access Complexity:** This metric measures the complexity required to exploit the vulnerability once an attacker has gained access to the target system. For example, consider a buffer overflow in an Internet service: once the target system is located, the attacker can launch an exploit at will. Other vulnerabilities, however, may require additional steps in order to be exploited. For example, a vulnerability in an email client is only exploited after the user downloads and opens a tainted attachment [1].

In version 3, following sub score is also used. The reason is that in version 3, in contrast to version 2, Access Complexity excludes any requirements for user interaction in order to exploit the vulnerability (such conditions are captured in the User Interaction metric)

- **User Interaction**: This metric captures the requirement for a user, other than the attacker, to participate in the successful compromise the vulnerable component. This metric determines whether the vulnerability can be exploited solely at the will of the attacker, or whether a separate user (or user-initiated process) must participate in some manner. This metric value is greatest when no user interaction is required

Selection of two above Base Group sub-scores (Access Vector, Access Complexity) were validated by extracting the Base Group sub scores (Access vector, Access Complexity, Authentication) for all indexed vulnerabilities (1988-2016) from CVSS calculator Version2. The results are shown in Fig. 1, Fig. 2, and Fig. 3. It is evident that, possible levels in Access Complexity and Access vector has considerably higher dispersion than Authentication. So, utilizing these two sub scores for the aim of improving risk scores, diversity will be rational in spite of picking Authentication too.

In this relation, in order to make our approach compatible with CVSS Version 3 calculator, User Interaction sub-score is also considered beside Access Complexity. So, by using our approach doing risk assessment based on CVSS Version 3 will be possible too.

Currently, CVSS Version 3 can be used for ranking nearly all vulnerabilities which are indexed after 2016. So, we extracted Access Vector, Access Complexity and User Interaction sub-scores for all the indexed vulnerabilities from CVSS Version 3 calculator. The results are shown in Fig. 4, Fig. 5, and Fig. 6 respectively. Acceptable dispersion of all possible values for the mentioned sub-scores makes utilizing these sub-scores for vulnerability discrimination reasonable.

Possible values for these mentioned Sub-Scores are listed in Table I and Table II for version 2 and version 3 of CVSS calculator respectively. It can be concluded that, the probability of exploiting each vulnerability is in direct relationship with its Access Vector and Access Complexity Sub-Scores.
In this paper, the dynamic probability of exploiting each vulnerability that is calculated based on its temporal features is inspired from Exploitability (called, Exploit Code Maturity in Version 3) Sub-Score of CVSS Temporal groups. A brief description of this score is here [1]:

- **Exploitability:**

This metric measures the current state of exploit techniques or code availability. Public availability of easy-to-use exploit code increases the number of potential attackers by including those who are unskilled, Thereby increases the danger of the vulnerability.
Based on [20], the probability of exploits availability can be evaluated by the Pareto distributions that is shown in (2).

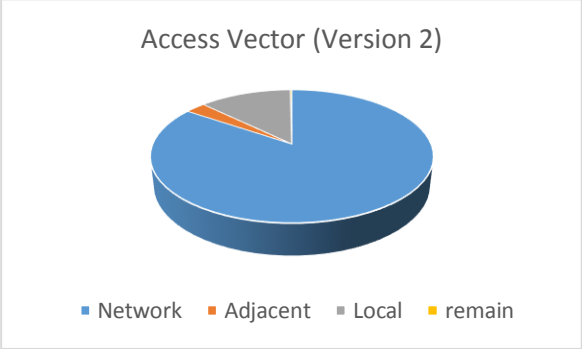

Figure 1. Access Vector levels scattering for all Indexed Vulnerabilities (1988-2016)

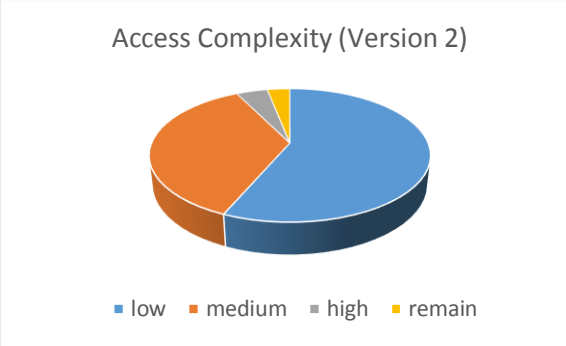

Figure 2. Access Complexity levels scattering for all Indexed Vulnerabilities (1988-2016)

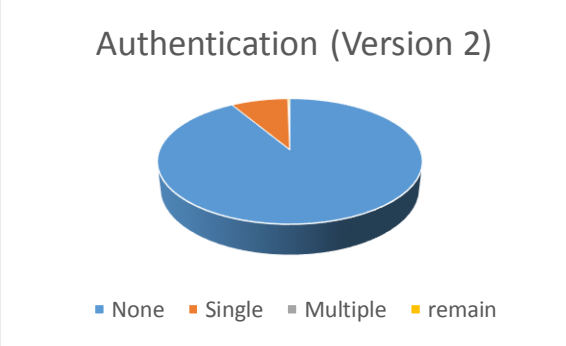

Figure 3. Authentication levels scattering for all Indexed Vulnerabilities (1988-2016)

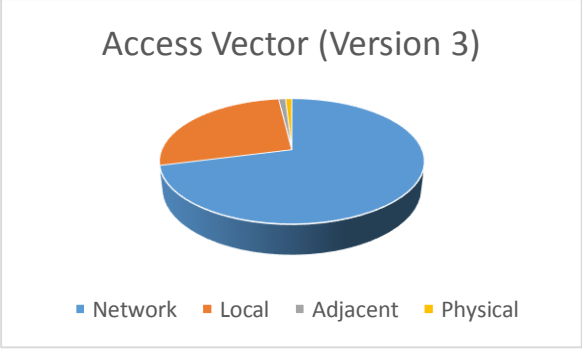

Figure 4. Access Vector levels of CVSS Version3 scattering for all Indexed Vulnerabilities (2016)

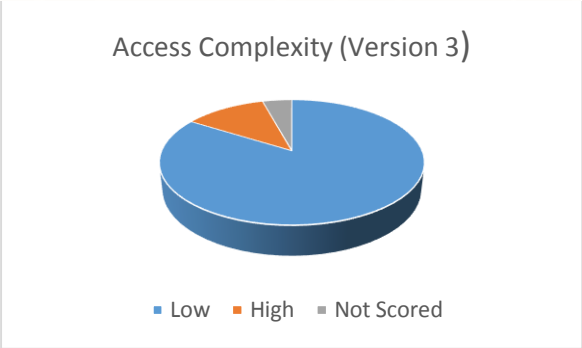

Figure 5. Access Complexity levels of CVSS Version3 scattering for all Indexed Vulnerabilities (2016)

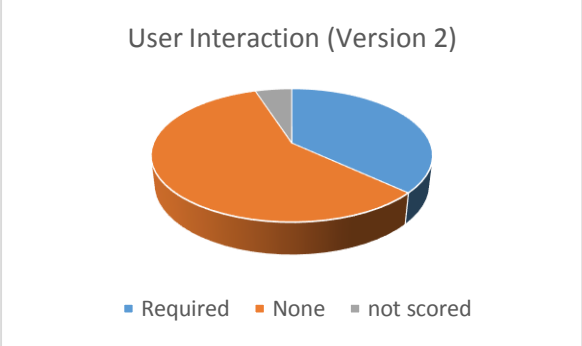

Figure 6. User Interaction levels of CVSS Version3 scattering for all Indexed Vulnerabilities (2016)

TABLE I. POSSIBLE VALUES FOR SOME CVSS SUB-SCORES (VERSION 2)

Access Vector	Metric value	Access Complexity	Metric value	I_C, I_I, I_A	Metric value
Local (L)	0.395	High (H)	0.35	None	0
Adjacent Network (A)	0.646	Medium (M)	0.61	Partial	0.275
Network (N)	1	Low (L)	0.71	Complete	0.66

Parameters for the best match with real data are shown too. In (2), 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) [20].

$$F(x) = 1 - \left(\frac{k}{x}\right)^{\alpha}. \qquad (2)$$

$$k = 0.00161\,, \quad \alpha = 0.260$$

TABLE II. POSSIBLE VALUES FOR SOME CVSS SUB-SCORES (VERSION 3)

Access Vector	Metric value	Access Complexity	Metric value	User Interaction	Metric value	I_C, I_I, I_A	Metric value
Physical (P)	0.2	Low (L)	0.77	None	0.85	None	0
Local (L)	0.55	High (H)	0.44	Required	0.62	Low	0.22
Adjacent Network (A)	0.62	_____	_____	_____	_____	High	0.56
Network (N)	0.85	_____	_____	_____	_____	_____	_____

Now the important step is to aggregate the three mentioned parameters (Access Vector, Access Complexity, and Exploitability) in one index to estimate the dynamic probability of exploiting each vulnerability by considering their individual effects.
The three parameters are in below relationships with vulnerability exploitation probability:

- **Access Complexity**

The lower the Access Complexity, the higher the numeric score and the higher the probability of vulnerability exploiting.

- **Access Vector**

The more remote an attacker can be to attack to a host, the greater the vulnerability score and the probability goes higher.

- **Exploitability**

The more powerful exploit tools exists for the vulnerability, it can be exploited with higher probability.

- **User Interaction**
  The probability of vulnerability exploiting goes higher when no user interaction is required and this metric value is greatest in this case.

Probability estimation in this paper is dependent on CVSS Calculator. This process is described for Version 2 and 3 of CVSS Calculator separately in the following.
Before description of the aggregation policy of above metrics, it is needed to say that our vulnerability scoring method specifies the risk of network vulnerabilities by the aim of hardening the network against two types of attackers:

1. **External attackers**

These attackers can steal the secret information of the local network. So, they jeopardize the privacy of it. In each local network, if there is a vulnerability that its Access Vector is assigned to Network, this network is threatened by External attackers the most.

2. **Non-skilled attackers**

If the network is threatened by these types of attackers, the probability of an attack occurring and consequently, service failure goes higher. Among the above parameters for measuring the intrinsic and dynamic vulnerability exploitation probability, Access Complexity (and User Interaction in CVSS Version 3) and Exploitability of each vulnerability can determine the simplicity degree of exploiting it.
In each network, if there are vulnerabilities with low Access Complexity (and User Interaction in CVSS Version 3), these vulnerabilities can be exploited by non-skilled attackers. Also, if the exploitability tools are existent for high numbers of network vulnerabilities, we can say, the percentage of attacks goes higher. Because, in this case exploiting the vulnerabilities by the attackers need not effort nor knowledge.
In order to create the relationship between above mentioned CVSS parameters and kind of attackers, we define two below parameters with the constraint in (3). β, γ indicates that, in risk estimation, which kind of attackers have higher priority for resisting against them and are corresponding to External Attackers and Non-Skilled Attackers respectively. So, we consider β as the coefficient for Access Vector and γ to both exploitability and Access Complexity (and User Interaction in CVSS Version 3).

$$\beta + \gamma = 1 . \qquad (3)$$

Now we should develop a standard method for efficient aggregation of the included parameters in probability estimation.
Our probability estimation problem can be addressed by using the effective prioritization methods called, MCDA (Multiple Criteria Decision Analysis). Such methods are concerned with the task of ranking a finite number of decision alternatives, each of which is explicitly described in terms of different characteristics called decision criteria which have to be taken account simultaneously. MCDA problems can be stated as below [23]:
There are a number, say m, of alternatives to be evaluated in terms of a number, say n, of decision criteria. Each criterion is associated with a weight of importance, denoted as w_i. The higher the weight is, the more important the criteria are assumed to be.

These weights are normalized. So, they add up to one or we have $\sum_{i=1}^{n} w_i = 1$.
Our problem of probability estimation can be modeled by the MCDA structure.
Vulnerabilities are alternatives, Access Complexity (and User Interaction in CVSS Version 3), Access Vector and exploitability are the criterion and β, γ are the weights (w_i).
Based on the requirements of our problem, among the various methods of MCDA, we chose the Weighted Sum Model or WSM for probability estimation. Relation (4) shows WSM Model. a_{ij} reflects the relative importance of alternative A_i in the set of all alternatives when they are evaluated in terms of criterion C_j.[21].

$$\sum_{j=1}^{n} a_{ij}\, w_i \,. \qquad (4)$$

So our probability estimation problem can be a model based on MCDA as it is shown in (5) and (6) for CVSS Calculator Version 2 and 3 respectively.
$AC(V_i)$,$AV(V_i)$, $Exp(V_i)$, UI corresponds to the Access Complexity, Access Vector, User Interaction and Exploitability of vulnerability V_i respectively.

$$Prob\,(V_i) = \beta \times AV(V_i) + \gamma \times 0.5 \times \big(AC(V_i) + Exp(V_i)\big)\,. \quad (5)$$

$$Prob\,(V_i) = \beta \times AV(V_i) + \gamma \times (\tfrac{1}{3}) \times \big(AC(V_i) + Exp(V_i) + UI(V_i)\big)\,. \quad (6)$$

B. Assessing the Impact of exploiting vulnerabilities

The other improvement of our approach is the novel method for assessing the Impact of exploiting each vulnerability on the three security parameters of the network (Confidentiality, Integrity and Availability). In CVSS, these three security parameters are scored for each known vulnerability as it is shown in Table I and II. The range of overall Impact score is between 0 and 10. I_C, I_I, I_A are the impact of exploiting each vulnerability on Confidentiality, Integrity, and Availability respectively. The impact parameter of CVSS is calculated based on the three above parameters with the relation (7) and (8) for version 2 and 3 respectively.

$$Impact_{AP_i} = 10.41 \times \big(1 - (1 - I_C) \times (1 - I_I) \times (1 - I_A)\big)\,. \quad (7)$$

$$Impact = \begin{cases} 6.42 \times ISC_{Base}, & \text{Scope Unchanged} \\ 7.52 \times [ISC_{Base} - 0.029] - 3.25 \times [ISC_{Base} - 0.02]^{15} & Scope\ Changed \end{cases}$$

$$ISC_{Base} = 1 - [(1 - Impact_{Conf}) \times (1 - Impact_{Integ}) \times (1 - Impact_{Avail})] \quad (8)$$

According [17], one major difficulty with the Impact Score of CVSS is that, the relative importance of three security parameters is ignored in Impact estimation. But the reality is that, Integrity Impact is more severe than the Availability Impact. Because violation of integrity very often affects the Availability Impact and also it is more difficult to be noticed. Similarly, Confidentiality Impact is more severe than the Integrity Impact, because, violation of Confidentiality is the hardest one to detect.
In this paper, this point is considered in impact assessment policy and we introduced a novel approach that has more diversity in Impact parameter than CVSS.
Note that, according to Table I and II, there are 27 possible combination of the three above Impact parameters. But, due to the symmetric nature of (7) the number of separated Impact scores produced by (7) is only 11. Also, because of the symmetric nature of (8), the number of produced unique Impact scores by Version 3 will be less than 27. Such realities leads to the same Impact Score for more than one vulnerability with different natures.
In this paper by the aim of improving this challenge (same Impact Scores for more than one Impact parameters pattern) and by considering the mentioned relative importance between three security parameters, we assigned each possible combination of three security parameters (Confidentiality, Integrity and Availability) a different level. So, we will have 27 possible values for Impact estimation in our Vulnerability Scoring System that shows its improvement in terms of diversity and accuracy over CVSS.
For example, combinations pnp, npp and ppn (and the same cases) have the same Impact Score in CVSS Version2 but, our method scores them in three different levels.
This paper's Impact estimation policy is shown in Table III. (The same idea can be applied for Version 3 of CVSS.). In order to make the Impact parameter within the same range as CVSS, the impact of each Combination is calculated by dividing its associated rank by 2.7.
An important point to be noticed is that, we didn't assigned zero for the Combination "NNN". (This combination means, exploiting the associated vulnerability has no impact on none of the three security parameters). The reason is that, exploiting of such vulnerability, despite of its no impact on three security parameters can help the attacker to gain more privileges on the network and these privileges can help him/her to exploit other vulnerabilities in the network.
Based on (1), by defining security metrics for estimating the probability of vulnerability exploitation and a new policy for determining the Impact of them, all the requirements for risk measurement are satisfied. Our risk assessment system in spite of determining the risk of each vulnerability numerically, has a qualitative description of the risk level of vulnerabilities that can be useful in network hardening. (Class high needs the most cost and class low needs the minimum cost).
Our qualitative description of risk scores is the same as what CVSS do for this purpose. CVSS Classifies Vulnerabilities according to the Base Score as below:

- 0 <Base Score <4: qualification degree=low
- 4 <=Base Score <7 : qualification degree=medium

- 7 <=Base Score =<10 : qualification degree=high

TABLE III. IMPACT ASSESSMENT POLICY FOR RANKING POSSIBLE COMBINATION OF SECURITY PARAMETERS

Combination	Rank	Combination	Rank	Combination	Ralue
NNN	1	NPP	5	CPN	22
PPP	17	PPN	16	PCN	19
CCC	27	PNP	11	PNC	12
NNP	2	NCC	9	PCC	21
NPN	4	CCN	25	CPC	24
PNN	10	CNC	15	CCP	26
NNC	3	NCP	8	PPC	18
NCN	7	CNP	14	CPP	23
CNN	13	NPC	6	PCP	20

V. VALIDATION OF THE PROPOSED APPROACH

As it is mentioned in section III, CVSS ranks vulnerabilities based on two parameters below:

- Exploitability: in Version 2, three sub scores are determinative in scoring Exploitability sub score. Each of these sub scores can have three different amounts. So, 27 different Exploitability scores are available for differentiating between vulnerabilities.
  In Version 3, four sub scores are used for vulnerability ranking. By considering their possible values, at most 48 different scores will be produced by version 3 [1].
- Impact: as it is shown in section IV, only 11 parameters are available for Impact estimation of available vulnerabilities.

Consequently, in CVSS, at most, 297 scores in Version 2 (528 scores in Version 3) can be produced for risk estimation of the huge number of vulnerabilities.
In the proposed approach, risk estimation is done based on two parameters below:

- Vulnerability probability estimation: Continuous nature of (2) improves the risk scores diversity considerably in comparison to CVSS.
- Impact of Vulnerability exploiting on security parameters of the network: in section IV, it is shown that, scores diversity of Impact parameter is $(\frac{27}{11})$ times greater than CVSS.

As a result in the proposed approach, scores diversity has considerably improved in comparison to CVSS. Considering Temporal features of vulnerabilities and Impact Estimation based on the relative importance of security parameters has been done in order to improve the accuracy in vulnerability scoring too.
In the next section, the results of applying the proposed method on two well-known service are proposed.

VI. EXPERIMENTAL RESULTS

The most important usage of our risk evaluation method is to find the most perilous vulnerabilities for doing minimum cost network hardening. In this relation, We utilized our method for risk evaluation of McAfee 2015 and McAfee 2016 vulnerabilities [22]. Required parameters for risk evaluation and the results are shown in Table IV and Table V. (Risk is calculated for the case in which$\beta =, \gamma = \frac{1}{2}$)
Each known vulnerability is indexed by one identifier called CVE in CVSS [19].
The effectiveness of our approach can be shown by comparing the results of risk assessment in our Vulnerability Scoring System and what CVSS reports of the considered vulnerabilities (Base Score).
By analyzing the results, we can have below interpretations for investigating the accuracy and diversity improvement in the proposed Vulnerability Scoring System in comparison to CVSS.

Accuracy Improvement Checking

- ✓ Changes in the severity level (low, medium and high) of vulnerabilities in our method in comparison with CVSS is because of considering temporal features beside the intrinsic ones in risk assessment. Applying such parameters in risk assessment is so important because they reflect the availability degree of exploitation tools.
  Vulnerability number 3 in McAfee 2015 (TABLE IV) and vulnerability number 1 in McAfee 2016 (TABLE V) are two such examples. Such changes are substantial to examine. This is because of the importance of doing efficient minimum cost network hardening.
  As a result of the differences between the elimination cost of various classes, vulnerability class changes from high to medium or low or from medium to low can have considerable effect on cost saving.

- ✓ Changes from low to medium or high indicates the existence of false negative reports in CVSS and changes from high to medium or medium to low reflects the false positive reports in CVSS.

TABLE IV. RISK EVALUATION OF MCAFEE 2015 VULNERABILITIES

NUM	CVE	Impact	*AV*	*AC*	Exp	Base	Level	Risk ($\beta = \frac{1}{2}, \gamma = \frac{1}{2}$)	Level
1	CVE-2015-2757	0.7407	1	0.35	0.9171	4	Low	0.6050	Low
2	CVE-2015-1616	6.2963	1	0.35	0.9243	6.5	Medium	5.1540	Medium
3	CVE-2015-4559	1.4815	1	0.61	0.8399	4.3	Medium	1.2777	Low
4	CVE-2015-3028	5.9259	1	0.35	0.9133	5.5	Medium	4.8345	Medium
5	CVE-2015-3030	3.7037	1	0.35	0.9133	4	Low	3.0216	Low
6	CVE-2015-2759	6.2963	1	0.61	0.9163	6.8	Medium	5.5507	Medium
7	CVE-2015-1305	10	0.395	0.61	0.9206	6.9	Medium	5.8015	Medium
8	CVE-2015-7612	6.2963	1	0.61	0.9149	6.8	Medium	5.5485	Medium
9	CVE-2015-7310	6.2963	1	0.71	0.9224	6.5	Medium	5.7177	Medium
10	CVE-2015-7238	3.7037	0.395	0.71	0.9249	2.1	Low	2.2453	Low
11	CVE-2015-7237	3.7037	1	0.71	0.9249	5	Medium	3.3656	Low
12	CVE-2015-3987	10	1	0.71	0.9496	7.2	High	9.1490	High
13	CVE-2015-3030	3.7037	1	0.71	0.9523	4	Low	3.3910	Low
14	CVE-2015-3029	3.7037	1	0.71	0.9523	4	Low	3.3910	Low
15	CVE-2015-2859	5.9259	1	0.61	0.9458	5.8	Medium	5.2679	Medium
16	CVE-2015-2760	1.4815	1	0.61	0.9529	3.5	Low	1.3196	Low
17	CVE-2015-2758	6.2963	1	0.61	0.9529	6.5	Medium	5.6083	Medium
18	CVE-2015-2053	1.4815	1	0.61	0.9548	4.3	Medium	1.3203	Low
19	CVE-2015-1619	1.4815	1	0.61	0.9551	3.5	Low	1.3204	Low
20	CVE-2015-1618	3.7037	1	0.71	0.9551	4	Low	3.3936	Low
21	CVE-2015-1617	1.4815	1	0.61	0.9551	3.5	Low	1.3204	Low
22	CVE-2015-0922	3.7037	1	0.71	0.9567	5	Medium	3.3951	Low
23	CVE-2015-0921	3.7037	1	0.71	0.9567	4	Low	3.3951	Low

TABLE V. RISK EVALUATION OF MCAFEE 2015 VULNERABILITIES

	CVE	Base Score	**Level**	Impact	Exp	AC	AV	UI	**Risk** ($\beta = \frac{1}{2}, \gamma = \frac{1}{2}$)	**Level**
1	CVE-2016-4535	7.5	High	1.1111	0.9449	0.77	0.85	0.85	0.9472	Low
2	CVE-2016-4534	3	Low	1.8519	0.9449	0.44	0.55	0.85	1.1991	Low
3	CVE-2016-3984	5.1	Medium	2.2222	0.9479	0.77	0.55	0.85	1.5622	Low
4	CVE-2016-3983	7.5	High	2.5926	0.9479	0.77	0.85	0.85	2.2114	Low
5	CVE-2016-3969	6.1	Medium	5.9259	0.9481	0.77	0.85	0.62	4.8277	Medium
6	CVE-2016-2199	8.8	High	10	0.9530	0.77	0.85	0.62	8.1550	High
7	CVE-2016-1715	6.6	Medium	6.6667	0.9541	0.44	0.55	0.62	4.0712	Medium
8	CVE-2015-8773	7.5	High	1.1111	0.9531	0.77	0.85	0.85	0.9487	Low
9	CVE-2015-8772	9.1	High	5.5556	0.9531	0.77	0.85	0.85	4.7436	Medium
10	CVE-2015-8765	8.3	High	6.2963	0.9543	0.77	0.85	0.85	5.3774	Medium

Diversity Improvement Checking

- ✓ McAfee 2015: CVSS differentiates between these 23 vulnerabilities by 11 unique squares. This is in the case that, our Vulnerability Scoring System ranks these 23 vulnerabilities by 20 different scores.
- ✓ McAfee 2016: Vulnerabilities of this service are differentiated by our Scoring System with unique scores. (Ten risk different scores for ten vulnerabilities.) And CVSS discriminates them with only 8 scores.

VII. CONCLUSION AND FUTURE WORKS

One of the basic challenges with available existing vulnerability scoring systems is the lack of accuracy that occurs as the result of ignoring temporal and environmental features of vulnerabilities. This deficiency is really serious because, introducing exploits and patches over time can change the exploitability of the vulnerability considerably.

In this paper, we developed a new scoring system that assesses the risk of Known vulnerabilities by considering their Temporal features. Our system is an improvement over CVSS Scoring System that is the most used one in this area.

In this paper, a novel method was introduced for assessing the Impact of vulnerability exploitation on three security parameters (Confidentiality, Integrity, Availability) by considering the relative importance between them.

Performing risk assessment by considering the type of the attacker that endangers the system the most is another unique novelty of our vulnerability scoring system.

Improvement of accuracy and diversity in vulnerability ranking in comparison to CVSS is the most significant feature of our Vulnerability Scoring System.

In the future we are going to improve the proposed framework by considering the environmental factors in scoring the vulnerabilities too. Another important future improvement should be developing a method for risk assessment of multi-step attacks in computer networks

. REFERENCES

[1] http://www.first.org/cvss/ (accessed December, 11, 2016)

[2] http://cwe.mitre.org/(accessed May, 7, 2016)

[3] http://cwe.mitre.org/cwss/cwss_v1.0.1.html(accessed May, 7, 2016)

[4] http://www.kb.cert.org/vuls/html/fieldhelp (accessed May, 7, 2016)

[5] https://technet.microsoft.com/en-us/security/gg309177.aspx(accessed May, 7, 2016)

[6] http://www-935.ibm.com/services/us/iss/xforce/faqs.html (accessed May, 7, 2016)

[7] http://www.symantec.com/security_response/severityassessment.jsp. (Accessed May, 7, 2016)

[8] http://www.mozilla.org/security/announce/. (accessed May, 7, 2016)

[9] Wang, Y., & Yang, Y. PVL: A Novel Metric for Single Vulnerability Rating and Its Application in IMS. Journal of Computational Information Systems, 8 (2), 579-590, 2012.

[10] Thaier Hamid, Carsten Maple and Paul Sant. Article: Methodologies to Develop Quantitative Risk Evaluation Metrics. International Journal of Computer Applications 48 (14): 17-24, June 2012.

[11] H. Joh and Y. K. Malaiya, "Defining and Assessing Quantitative Security Risk Measures Using Vulnerability Lifecycle and CVSS Metrics," Proc. Int. Conference on Security and Management (SAM11), 2011, pp. 10-16.

[12] Frühwirth, C. & Männistö, T. Improving CVSS-based vulnerability prioritization and response with context information. Proceedings of International Workshop on Security Measurement and Metrics (MetriSec), 2009, PP. 535-544.

[13] Massimiliano Albanese, Sushil Jajodia, Anoop Singhal, Lingyu Wang, "An efficient approach to assessing the risk of zero-day vulnerabilities," Proc. 10th International Conference on Security and Cryptography (SECRYPT 2013), Reykjavik, Iceland, July 29-31, 2013

[14] Lingyu Wang, Sushil Jajodia, Anoop Singhal, Pengsu Cheng, Steven Noel: k-Zero Day Safety: A Network Security Metric for Measuring the Risk of Unknown Vulnerabilities. IEEE Trans. Dependable Sec. Comput. 11(1): 30-44 (2014)

[15] GALLON, L. Vulnerability discrimination using cvss framework. In New Technologies, Mobility and Security (NTMS), 4th IFIP International Conference, 2010, pp. 1 –6.

[16] Liu, Q. & Zhang, Y. VRSS: A new system for rating and scoring vulnerabilities. Computer Communications. 34 (3), 2011, PP. 264-273.

[17] Spanos, G. & Sioziou, A. & Angelis L. WIVSS: a new methodology for scoring information systems vulnerabilities. Panhellenic Conference on Informatics.2013, PP. 83-90

[18] Ghani, H. & Luna, j. & Suri, N. Quantitative assessment of software vulnerabilities based on economics-driven security metrics. International Conference on Risks and Security of Internet and Systems (CRiSIS)., 2013, pp. 1-8.

[19] http://cve.mitre.org/. (accessed May, 7, 2016)

[20] Frey, S. & May, S. & Fiedler, U. & Plattner, B. Large-scale vulnerability analysis. LSAD '06:Proceedings of the 2006 SIGCOMM workshop on Large-scale attack defense. pp. 131–138, 2006.

[21] Triantaphyllou, E. & Baig, K. (2005). The Impact of Aggregating Benefit and Cost

[22] http://www.cvedetails.com/vendor/345/Mcafee.html (accessed December, 17, 2016).

[23] Criteria in Four MCDA Methods. In IEEE Transactions on Engineering Management. 52 (2), pp. 213-226.

[24] M. Keramati, "New Vulnerability Scoring System for dynamic security evaluation," *2016 8th International Symposium on Telecommunications (IST)*, Tehran, 2016, pp. 746-751.

Marjan Keramati received both her undergraduate and graduate degrees in Computer System Architecture from Iran University of Science and Technology. Currently, she is Faculty Member in Semnan University, Department of Computer Science. Also, she is Editorial Board Member in the International Journal of Cases on Information Technology (USA). Besides, she is the member of Technical Commission of Standard Codification and registered one National Standard in the field of network security in 2017. Publishing papers in International Journals and Conferences, Journal paper reviewing in various prestigious International Journals and being both Scientific and Executive Committee member in International Conferences are the other examples of her academic activities. Her research Interests include: Risk Evaluation, Security Metrics, Security Modeling ,Vulnerability Analysis, Cloud Computing Security, Intrusion Prevention Systems, Intrusion Response Systems.