A Holistic Approach to Evaluate Cyber Threat

Márcio Monteiro[1], Thalysson Sarmento[1], Alexandre Barreto[1] and Paulo Costa[2]
[1]Instituto de Controle do Espaço Aéreo, São José dos Campos, Brazil
[2]C4I Center, George Mason University, Fairfax, USA
E-mails: {contemmcm, thalyssontfs, barretoabb}@icea.gov.br, pcosta@c4i.gmu.edu

***Abstract*— Several vulnerability databases and standards are currently available for assessing the degree of security of IT infrastructures in general. These standards focus on different aspects of the systems, while generally failing to provide support for holistic analyses - a key aspect in ensuring a secure IT infrastructure. This work aims to address this gap by presenting a new methodology for evaluating the overall security risks of a networked system that adopts an ontology-based approach we presented in previous work. We leverage current security standards and databases, while also considering the human factors to build a broader and interconnected view. Our methodology is meant to achieve a more realistic picture of the network security, hence improving situation awareness for its administrators. To illustrate our approach, this paper brings a case study applying the new methodology to a few target networks. The proof of concept is meant to underscore the methodology's effectiveness in assessing the security of the whole network.**

I. INTRODUCTION

Cyber security assessment has a importance role in a modern society. has become more interconnected through computer systems and networks. It is well-established that cyber threats can cause on corporations severe economic losses and damages to their reputation [1]. As a result, investments on cyber security has been growing significantly, even during market crises [2].

A basic standard for cyber security assessment is the Common Vulnerabilities and Exposures (CVE), which is the *de facto* standard to report and communicate software vulnerabilities between organizations and entities. Currently, the CVE has been standardized by the Telecommunication Standardization Sector of the International Telecommunication Union (ITU-T) [3] and is being heavily used by automatic security assessment tools (*e.g.*, Nessus and OpenVAS) to identify software vulnerabilities on target hosts.

On top of CVE, another standard was established to score the vulnerabilities with respect to their severity, impact and exploitation capacity. This standard is called Common Vulnerability Scoring System (CVSS). One of the most important CVSS databases is hosted and managed by the National Vulnerability Database (NVD), which provides the scores for most known vulnerabilities.

Although those standards are very efficient in cataloging and prioritizing software vulnerabilities, system administrators are usually interested in knowing how vulnerable is their entire network, no only individual hosts.

For instance, if a web server is highly protected against external threats, but vulnerable hosts in the same local area network have open access to the server, this condition should impact the overall score of the system. In addition, users can also be considered vulnerabilities of the system, as they could be deceived (or "exploited") somehow to execute malicious software. Then, security unaware or careless users should also impact the overall score of the system.

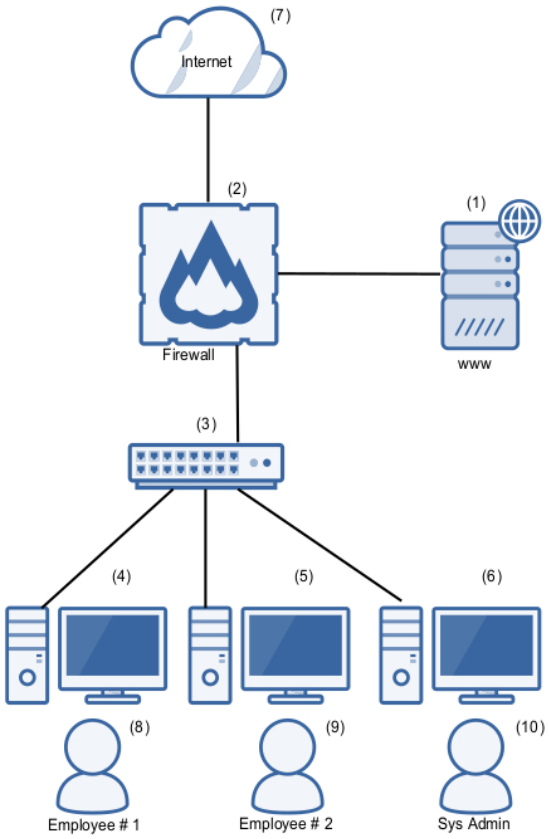

Fig. 1: How secure is this network?

In this work we propose to analyze those aspects (CVE, CVSS and human factors) in a unified manner for a target network, where vulnerabilities scores are propagated through the network's trusted relationships (intentional or not). This way, we provide an overall security metric that can be used to classify entire networks.

This work is organized as follows: Sec. II briefly details the main attributes of CVE and CVSS; Sec. III presents the proposed metric; and Sec. IV concludes with final remarks.

II. Overview

A. Common Vulnerabilities and Exposures

The Common Vulnerabilities and Exposures (CVE) is a standard for cataloging vulnerabilities of computer systems. It consists of a list of information of security vulnerabilities and exposures, mainly reported by the community, aiming to provide common names for publicly known problems. It allows to share data about vulnerability capabilities (tools, repositories, and services).

The main attributes of a CVE are:

- CVE identifier number (i.e., CVE-1999-0067);
- Vulnerability type: buffer overflow, cross site request forgery (CSRF), cross site scripting (XSS), directory traversal, incorrect access control, insecure permissions, integer overflow, missing SSL certificate validation, SQL injection, XML external entity (XXE), and others or unknown;
- Vendor of the product(s);
- List of vulnerable products and versions;
- Attack type: context-dependent, local, physical, remote, other;
- Impact: code execution, denial of service, escalation of privileges, information disclosure, other.

Currently, the MITRE Corporation is responsible for managing CVE identifiers generation and publication through its web site [4]. In addition, MITRE also delegates this attribution to its several CVE numbering authorities (CNAs).

B. Common Vulnerability Scoring System

The Common Vulnerability Scoring System (CVSS) is an open framework for describing specific characteristics of software vulnerabilities. It consists of three metric groups: base, temporal, and environmental.

The *base* group represents the intrinsic qualities of a vulnerability, the *temporal* group reflects the characteristics of a vulnerability that changes over time, and the *environmental* group represents characteristics of a vulnerability that are unique to the user's environment.

In this work, we focus on the *base* metric, which produces a score ranging from 0.0 to 10.0. It is composed by the impact subscore (ranging from 0 to 6) and the exploitability subscore (ranging from 0 to 4). However, the overall CVSS score of a single vulnerability is also impacted by the *temporal* and *environmental* metrics. Readers are encouraged to refer to [5] for more information on CVSS specifications and formulas.

The main attributes of CVSS base score are:

- Attack vector (AV): network (N), adjacent network (A), local (L), and physical (P);
- Attack complexity (AC): low (L), high (H);
- Privileges required (PR): none (N), low (L), high (H);
- User interaction (UI): none (N), required (R);
- Scope (S): unchanged (U), changed (C);
- Confidentiality impact (C): none (N), Low (L), high (H);
- Integrity impact (I): none (N), Low (L), high (H);
- Availability impact (A): none (N), Low (L), high (H);

Usually, the CVSS is represented as a vector string, a compressed textual representation of the values used to derive the score. String (1) below is an example of a CVSS vector string.

$$\text{CVSS:3.0/AV:N/AC:L/PR:L/UI:R/S:C/C:L/I:L/A:H} \quad (1)$$

The equations adopted to calculate the CVSS base score are provided in Sec. III.

C. Human Factors

Human factors play an important role in the security of an organization, since users are used as both targets and vectors of attacks. Several social engineering methods can be employed to obtain key information and select the most vulnerable employees.

In this work we propose to model the users' "vulnerabilities" as a CVSS-like metric. In other words, the users would also be rated by the impact and exploitability subscores. As an example, users with high privileges in the network would have a high impact factor, because if they get "compromised" that would grant intruders deeper access to the network.

On the other hand, users unaware of security issues or careless about it can be considered highly "exploitable", that is, they can be easily deceived to execute malicious software on their computers. There are numerous methods to do so, such as telephone calls from fake IT staff, phishing campaigns, malicious websites, etc.

To prevent such situations, the staff should perform security awareness training. Besides, the corporation should have a solid information security policy and all means should be employed to enforce it.

III. The Proposed Metric

System administrators usually focus heavily in protecting their networks against external cyber attacks. For this reason, the insider threats might receive insufficient attention and, consequently, the security can be impacted. Considering that every host connected to the Internet is a potential attack vector through phishing campaigns (someone trying to convince the user to execute the malicious code) and applications vulnerabilities (browsers, e-mail and document readers), and that the protection against known hosts is reduced, then a single host can severely compromise the security of the entire network.

The proposed metric in this work is obtained by a five-step approach, each one being required for computing the overall security of a given network. The technique involves building a graph representing the overall network as well as the relationship between each step. The relative importance of each step is assessed using multi-criteria decision analysis concepts.

There are different approaches for building such graph and defining the metric. However, the specific aspects of the cyber security domain involving different perspectives (e.g. technical, human factor, standards, etc.) naturally led us to reuse/adopt the ontology-based approach previously presented in [6]. The general idea is to use semantic techniques

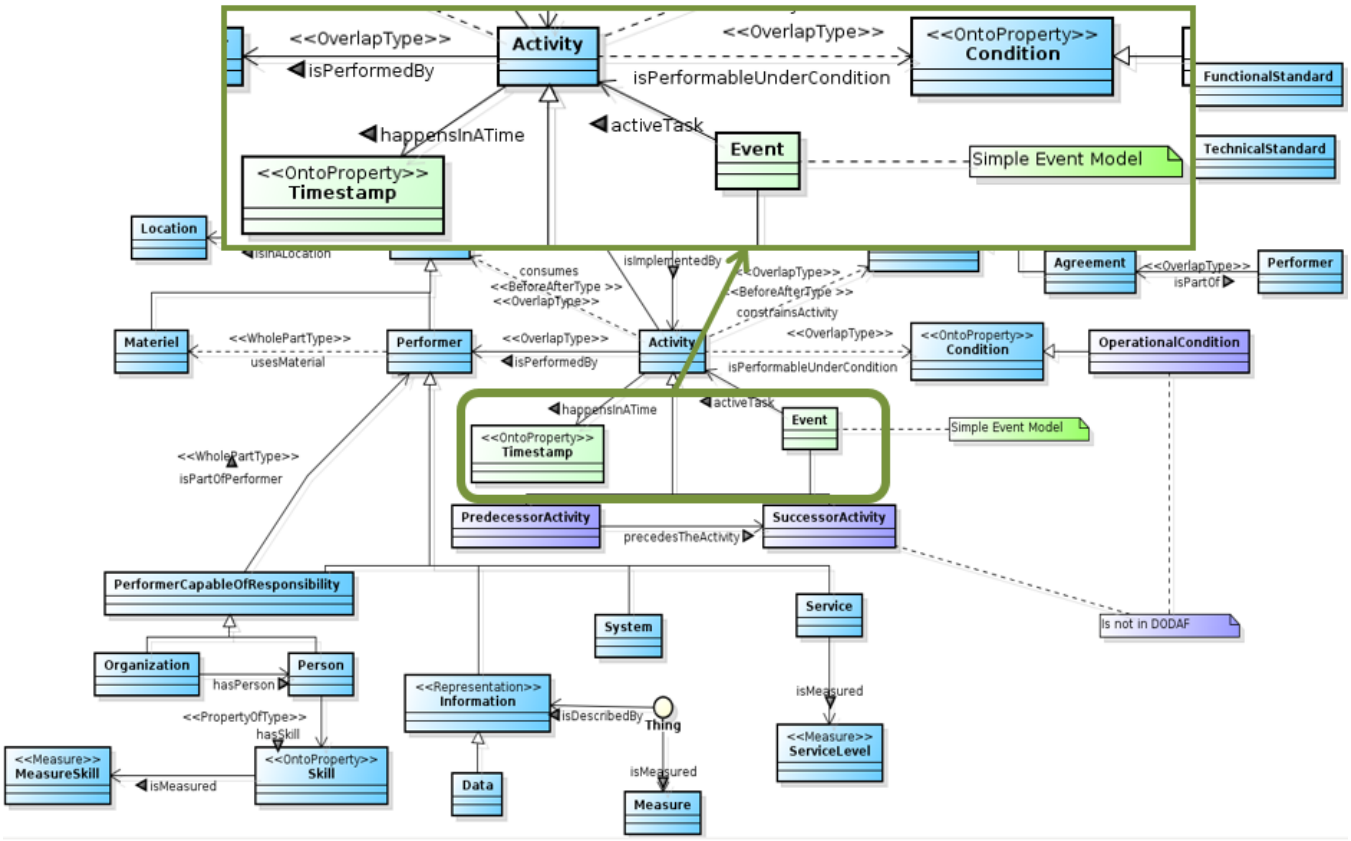

Fig. 2: Mission Ontology

in supporting the definition of the target mission, its support task, as well as the services and network configuration required to accomplish a mission. As in the aforementioned previous version, in this work we use the DoDAF Conceptual Data Model to represent the concepts involved in the mission. The difference, however, is that in this work we extend this approach by incorporating time and event descriptions [7].

The ontology is presented in Figure 2, which conveys the queries that can be performed in cyber-situation awareness: **WHAT** (Activity), **WHY** (Goal and Desire Effect), **HOW** (Resource and Guidance), **WHO** (Performer), **WHERE** (Location), and **WHEN** (Timestamp and Event).

1) Complete inventory: The first step consists in obtaining a complete and detailed asset inventory record of the target network, including hubs, switches, routers, software list, etc. This is fundamental for every security approach and should not be a problem for security aware corporations.

2) Communications: The second step consists in mapping the communication between the assets (including the users). If the network contains N assets, this can be mapped into a $N \times N$ matrix.

Taking Fig. 1 as example, we can derive its access matrix as presented in Table I, where the rows represent the asset with communication initiative, the columns represent the communication destination, and a cell filled with a 'Y' informs that such communication is allowed (or that there is nothing forbidding such communication).

TABLE I: Trusted relationships between assets (matrix).

	1	2	3	4	5	6	7	8	9	10
1	-						Y			
2	Y	-	Y	Y	Y	Y	Y			
3		Y	-	Y	Y	Y				
4				-	Y	Y	Y			
5				Y	-	Y	Y			
6	Y	Y	Y	Y	Y	-	Y			
7	Y						-			
8				Y				-		
9					Y				-	
10	Y					Y				-

To generate the aforementioned table, a SPARQL query is performed on the Mission Ontology. This greatly simplifies the otherwise complex task of discovering and mapping connections, in spite of these being hidden or not.

An alternative representation of Table I can be achieved through directed graphs, as depicted in Fig. 3. The main advantage of this approach is that it makes relatively easier to identify nodes with a higher impact higher to the overall security of the network. Also, it becomes possible to derive attack chains throughout the network.

3) Vulnerabilities assessment: The third step is to obtain the CVE IDs and CVSS base vector string for all N hosts of

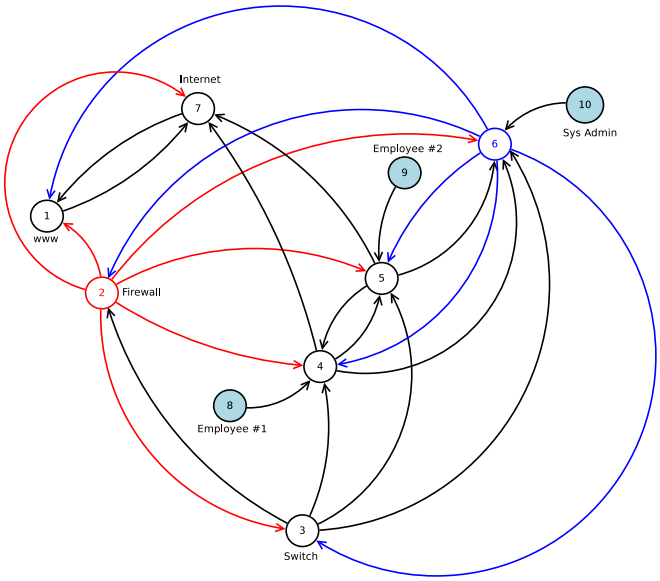

Fig. 3: Trusted relationships between assets (graph).

the network. There are many automated tools that can help in obtaining this information, such as the Nessus Vulnerability Scanner [8] and the Open Vulnerability Assessment System (OpenVAS) [9].

4) Calculating Scores: Once the vulnerabilities are obtained, for every CVSS string we need to compute the impact sub score α and the exploitability sub score β.

The impact sub score α can be computed according to (2):

$$\alpha = \begin{cases} 6.42 \times ISC_{\text{Base}}, & \text{if S} = \text{U}, \\ 7.52 \times [ISC_{\text{Base}} - 0.029] - & \\ 3.25 \times [ISC_{\text{Base}} - 0.02]^{15}, & \text{if S} = \text{C} \end{cases} \tag{2}$$

where

$$ISC_{\text{Base}} = 1 - [(1 - \Delta_{\text{C}}) \times (1 - \Delta_{\text{I}}) \times (1 - \Delta_{\text{A}})]. \tag{3}$$

The confidentiality impact (C), integrity impact (I) and availability impact (A) parameters are given by:

$$\Delta_{\text{C/I/A}} = \begin{cases} 0.56, & \text{if C/I/A} = \text{Low (L)}, \\ 0.22, & \text{if C/I/A} = \text{High (H)}, \\ 0, & \text{if C/I/A} = \text{None (N)}. \end{cases} \tag{4}$$

The exploitability sub score can be computed as:

$$\beta = 8.22 \times \Delta_{\text{AV}} \times \Delta_{\text{AC}} \times \Delta_{\text{PR}} \times \Delta_{\text{UI}}. \tag{5}$$

The attack vector (AV) parameter is given by (6):

$$\Delta_{\text{AV}} = \begin{cases} 0.85, & \text{if AV} = \text{Network (N)}, \\ 0.62, & \text{if AV} = \text{Adjacent Network (A)}, \\ 0.55, & \text{if AV} = \text{Local (L)}, \\ 0.20, & \text{if AV} = \text{Physical (P)}. \end{cases} \tag{6}$$

On the sequence, the attack complexity (AC) parameter is given by (7)

$$\Delta_{\text{AC}} = \begin{cases} 0.77, & \text{if AC} = \text{Low (L)}, \\ 0.44, & \text{if AC} = \text{High (H)}. \end{cases} \tag{7}$$

For unmodified scope (S:U), the following equation applies for the privileges required (PR) parameter:

$$\Delta_{\text{PR}} = \begin{cases} 0.85, & \text{if PR} = \text{None (N)}, \\ 0.62, & \text{if PR} = \text{Low (L)}, \\ 0.27, & \text{if PR} = \text{High (H)}. \end{cases} \tag{8}$$

However, for modified scope (S:C), the following equation applies for PR:

$$\Delta_{\text{PR}} = \begin{cases} 0.85, & \text{if PR} = \text{None (N)}, \\ 0.68, & \text{if PR} = \text{Low (L)}, \\ 0.50, & \text{if PR} = \text{High (H)}. \end{cases} \tag{9}$$

Finally, the user interaction (UI) parameter can be given by (10):

$$\Delta_{\text{UI}} = \begin{cases} 0.85, & \text{if UI} = \text{Not Required (N)}, \\ 0.62, & \text{if UI} = \text{Required (R)}. \end{cases} \tag{10}$$

5) Computing the proposed metric: After computing the impact sub score (α) and exploitability sub score (β), for every vulnerability found in previous steps we need to assemble a P matrix, where the first column ($p_{i,1}$, $\forall i$) corresponds to the impact sub score (α), and the second column ($p_{j,2}$, $\forall j$) corresponds to the exploitability sub score (β). Then, we need to append three additional points to this matrix such that its final version is according to (11):

$$P = \begin{bmatrix} p_{1,1} & p_{1,2} \\ \vdots & \vdots \\ p_{N,1} & p_{N,2} \\ 0 & 0 \\ \max(p_{1,1}, \ldots, p_{N,1}) & 0 \\ 0 & \max(p_{1,2}, \ldots, p_{N,2}) \end{bmatrix} \tag{11}$$

where the function $\max(\cdot)$ returns the maximum value of its arguments and N denotes the number of vulnerabilities found on previous steps.

Finally, we must compute the convex hull of the matrix P and its 2D area (considering the outmost vulnerabilities as vertices of the polygon), and divide resulting area by the highest possible CVSS subscores ($6 \times 4 = 24$). Conducting the calculations this way ensures that the proposed metric is presented as percentage. The results are then used to rate the network security according the intervals presented on Table II.

Fig. 4 depicts an example of a fictitious network composed of three nodes. The overall vulnerability metrics has been appointed as 70.4476 %, which corresponds to the rating *Highly Vulnerable*, according to Table II. Every marker on

TABLE II: Ratings

Min (%)	Max (%)	Rating
00.00	00.00	None
00.01	39.99	Low
40.00	69.99	Medium
70.00	89.99	High
90.00	100.0	Critical

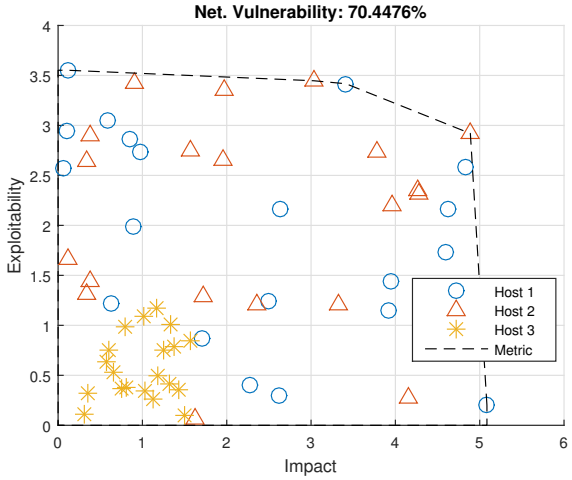

Fig. 4: Vulnerability assessment using the proposed metric for a highly insecure network.

this figure corresponds to a CVSS metrics (impact and exploitability sub scores).

Likewise, Fig. 5 presents a second network with less severe individuals vulnerabilities throughout the nodes of the network. Notice that the overall vulnerability was 16.7402 %, which corresponds to the rating *Low*, according to Table II.

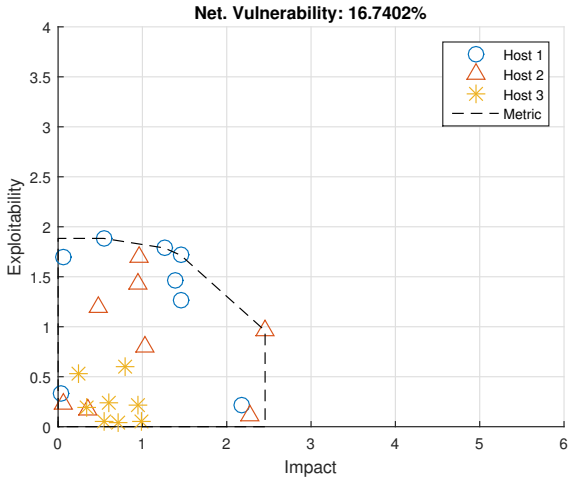

Fig. 5: Vulnerabilities of hosts of the network.

IV. Final Remarks

This work presented an ontology-based approach for analyzing the vulnerability of a network in a holistic way, using multiple-criteria analysis and modeling the human factor as CVSS v3 base scores. An example on a fictitious network was performed in order to demonstrate the practicality of the proposed metric. Further, the reuse of concepts previously defined in an existing ontology we had developed suggests that the approach can be generalized to encompass the diverse aspects that permeate the way different corporations are structured.

Acknowledgment

Márcio Monteiro, Thalysson Sarmento and Alexandre Barreto would like to thank the financial support of the Brazilian agencies MCTI and FINEP (Ref. 04/2013/12).

References

[1] CNN Money, "Cybercrime costs the average U.S. firm $15 million a year," 2015, [accessed 05-Sept-2016]. [Online]. Available: http://money.cnn.com/2015/10/08/technology/cybercrime-cost-business/

[2] Reuters, "Cyber security investing grows, resilient to market turmoil," 2015, [accessed 05-Sept-2016]. [Online]. Available: http://fortune.com/2015/09/23/cyber-security-investing/

[3] Study Group 17, *ITU-T Recommendation X.1520: Common vulnerabilities and exposures*, Std., April 2011.

[4] MITRE, "Common vulnerabilities and exposures – the standard for information security vulnerability names," [accessed 05-Sept-2016]. [Online]. Available: https://cve.mitre.org/

[5] FIRST, "Common vulnerability scoring system v3.0: Specification document – version 1.7," [accessed 05-Sept-2016]. [Online]. Available: https://www.first.org/cvss/specification-document

[6] A. Bareto, "Cyber-argus framework – measuring cyber-impact on the mission," Ph.D. dissertation, Instituto Tecnológico de Aeronáutica, Brazil, 7 2013.

[7] W. R. e. a. VAN HAGE, "Design and use of the simple event model (sem)," *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 9, no. 2, Sep 2011.

[8] Tenable Network Security.

[9] "Open vulnerability assessment system (OpenVAS)," [accessed 05-Sept-2016]. [Online]. Available: http://www.openvas.org