Network Vulnerability Analysis

Abstract

A network vulnerability analysis (NVA) is the process of identifying and evaluating those security loopholes that might exist in an enterprise before the network is hacked. This paper discussed network vulnerability analysis and its processes. It also demonstrate some approaches to conducting vulnerability analysis for any network environment using manual tools and showed other automatic tools used for conducting vulnerability analysis. The paper also addressed the reasons for network vulnerability analysis and finally discussed vulnerabilities of the network. Findings from this study indicate that network vulnerability analysis is important. However, conducting vulnerability analysis does not necessarily prevent security on its own; instead, it reflects a snapshot of the environment at a particular point in time and pointed out the rationale for regular conducting of vulnerability analysis.

Keywords: Network; Vulnerability; Analysis; Assessment; Network Mapping.

1. Introduction / Literature review

The quest to investigate and eradicate vulnerability in networks is becoming the paramount concern for most security expert and network administrator. Thus, finding and closing the security holes in the network is the only way to protect organization's network from any possible attack notwithstanding the organizations well-managed firewall, an updated antivirus, and intrusion detection system [1]. Of most essence is bearing in mind that profit making is not always the intension of an attacker but, in most cases can be born as a result of an unhappy client, unsatisfied employee, or even a disgruntled contractor all for the sake of self-satisfaction and revenge.

----

* Corresponding author.

A vulnerability assessment is the process of identifying and analyzing those security vulnerabilities that might exist in an enterprise [2]. It should be noted that vulnerability assessments alone do not prevent security incidents, nor do their results provide any indication of a current or past security incident. Conducting an assessment does not necessarily improve security on its own; instead, it reflects a snapshot of the environment at a particular point in time, and its goal is simply to identify and analyze weaknesses present in a technical environment. Vulnerability assessments are not exploitative by nature (compared to, for example, ethical hacking or penetration tests). In conducting a vulnerability assessment, practitioners (or the tools they employ) will not typically exploit vulnerabilities they find. Instead, a vulnerability assessment serves an altogether different purpose: it allows an enterprise to focus on reconnaissance and discover weaknesses in its environment. According to [2], validating the vulnerability through a penetration test—that is, actually staging and executing an exploit to prove that the loophole can allow an attacker to gain access to the network—may not be necessary because the weakness is already known or suspected through vulnerability assessment. Network- based vulnerability assessment aims at compiling an inventory of systems and services attached to the network and, for each system and service, identify the weaknesses and vulnerabilities visible and exploitable on the network by using methodologies such as network mapping, port mapping, vulnerability programs, and stopping unneeded services [5]. The baseline and narrowing the scope of the vulnerability assessment should also be considered. Traditional vulnerability scanners merely identify a list of individual vulnerabilities but are unable to depict the correlations between these vulnerabilities. Given the complexity of today's network infrastructure, it is not enough to consider the presence or absence of vulnerabilities in isolation [6]. What is needed is a systematic approach that models threats against the system. The model should allow system administrators to understand the actual outline of the network vulnerability and help them manage the risk in a proper manner.

The speed of exploit deployment is revealing weakness in corporate policies as a result of relatively infrequent manual penetration testing. Even the perimeter defences (anti-virus, firewall and IPS/IDS) which are vital could be bypassed by determined effort to reach and exploit known vulnerabilities that reside just inside the fence. Hence, the introduction of an automated network scanning mechanism and consolidated reporting to identify and track mitigation of known vulnerabilities is establishing a higher overall security level often using already existing budget and manpower [7]. Past work proposed applying post process —raw‖ attack graph in evaluating Network security so that the result can be abstracted and becomes easier for a human user to grasp [8]. This method showed that while visualization is a major problem caused by the multitude of attack paths in an attack graph, a more severe problem is the distorted risk picture it renders to both human users and quantitative vulnerability assessment models. Consequently, it was proposed that abstraction be done before attack graphs are computed, instead of after. This way, the distortion in quantitative vulnerability assessment metrics can be prevented, at the same time improving visualization as well. A graph-based approach to network vulnerability analysis allows analysis of attacks from both outside and inside the network. It can analyze risks to a specific network asset, or examine the universe of possible consequences following a successful attack [9]. The graph- based tool can identify the set of attack paths that have a high probability of success (or a low "effort" cost) for the attacker. The attack graph. Nodes identify a stage of attack, for example the class of machines the attacker has accessed and the user privilege level he or she has compromised. The arcs in the attack graph represent attacks or stages of attacks. By assigning probabilities of success on the arcs or costs representing level-of-effort for the attacker, various graph algorithms such as shortest-path algorithms can identify the attack paths with the highest probability of success. In previous generations of systems, a risk adverse vulnerability posture dictated custom hardware and software solutions. The next generation of information systems and infrastructures are built upon the concept of acceptable risk where the security features and system architecture are deemed to provide sufficient protection over the life of the data processed [10]. Following this, the Network Vulnerability Tool (NVT) concept develops and applies a single topological system model. This model supports the information needs of multiple vulnerability analysis tools using an integrated knowledge solicitation and translation framework. As part of this effort, vulnerability tools from COTS, GOTS, and research laboratory sources were surveyed, and a representative sample tool collection was selected for inclusion in the NVT prototype. The prototype integrates and interactively applies multiple existing vulnerability assessment technologies, resulting in a cohesive, combined vulnerability/risk assessment. Network security does not stop at patch management and running antivirus software. It requires checking configurations, knowing issues in third-party applications, as well as potentially troublesome hardware that in their default configuration can be harmful to the network's security. These processes are what constitute a vulnerability assessment [3].

2. Types of Vulnerability Assessment

There are four types of vulnerability assessment. The network-based scan, the host-based scan, wireless-based scan and the application-based scan vulnerability assessments [2]. However, this paper focused on the two major parts. Network-based analysis identifies vulnerable systems on the entire network. This test should be conducted first to provide the immediate results of highly severe vulnerabilities that needed a quick fix. For instance, a firewall not configured correctly or vulnerable web server, which is considered very severe vulnerabilities, can be detected easily by running a network vulnerability test [1]. Host-based analysis is an important exercise because it identifies vulnerabilities on the organization's internal systems by providing an extra layer of security testing such as analyzing access limits of the hosts from accessing confidential data of the organization. This analysis works on client-server model where client files should be installed on every machine that you want to check.

3. Why Vulnerability Analysis Should be Conducted

Despite fully patching the network security and deploying antivirus solutions, hackers might still be able to exploit a number of misconfiguration such as: Unnecessary open shares, Unused user accounts, Unnecessary open ports, Rogue devices connected to your systems, Dangerous script configurations, Servers allowing use of dangerous protocols, Incorrect permissions on important system files, Running of unnecessary, potentially dangerous services [3]. Apart from these misconfiguration, when running a vulnerability assessment on your network you might find several security issues with a wide range of software and hardware including: Default passwords on certain devices, Unnecessary services running on some devices, Running web services that contain known vulnerabilities, Dangerous applications such as peer-to-peer applications and Third-party applications that are a vulnerability to known exploits. Other reasons are to find: Undisabled User Account of employees who left the system and who may log into the system and cause some havoc, Unpassworded Open Shares through which hackers use to spread antivirus, there is a need to continually assess the vulnerability of a system’s network. Hence, the vulnerability identification is to determine the points through which the security of the network can be compromised. The output of this process helps decision maker to identify appropriate controls for reducing the risk in the risk mitigation process.

4. The Vulnerability Assessment Processes

A good vulnerability assessment is based on a methodical approach irrespective of any tool and strategy been used [1]. The generalized vulnerability assessment follows the processes below: Determine the target systems: defining the IP (i.e., internet protocol) addresses of the target devices that need to be analyzed which will be used to extract responses in return of probes so that the system will mark the IP addresses as valid host. Locate the live systems: performing many fingerprinting techniques to detect the live host and their information. List services: scanning ports starts which finds out vulnerable ports on the network and the services associated with specific port numbers thus, discovering the TCP (Transmission Control Protocol) and UDP (User Datagram Protocol) services which are open on the network or system. Different scanning types such as TCP ACK and TCP window scanning should be used to avoid firewalls and IDS. The addresses or port numbers of open ports are logged and saved for later usage. Recognize services: recognizing the services on every open port of a network identified in list services by frequently sending similar requests while the assessment tool examines the responses against a set of signatures. Once a match is made between signatures of known application, the data is saved for later use and tool starts running tests on other services. Identify applications: pinpointing vendor and type of every service which was discovered to ensure that the vulnerability test of one application is not effecting or crashing other applications. An application causing another application to crash leads to inaccurate results. The common cause of a false result is imperfect application identification before the test was conducted. Identify vulnerabilities in network: given that all the open ports on the network are mapped, all familiar services are also mapped to a particular application. The vulnerability analysis test starts where active configuration probes are conducted and in the end a set of custom attacks on network which will define either a stated vulnerability exists in a system or not. Report vulnerabilities: lastly, discovered vulnerabilities on the network are reported which depicts which vulnerabilities are highly severe and which are not and also provide solutions on how the reported vulnerabilities will be fixed. One thing which most of the assessment tools have in common is the ability to show tendency report of how a tested network progressed over time. The network administrator can also choose the report summary to present it to his organization's management.

5. Analysis of Network Vulnerabilities

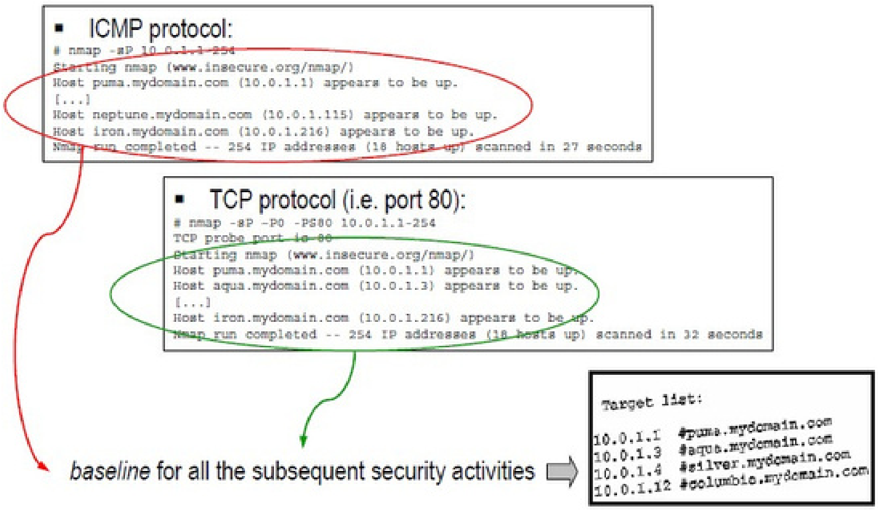
acquisition and Vulnerability assessment [5].

5.1 Target acquisition

Target acquisition is when a complete knowledge of the network environment is analyzed in order to identify all the alive hosts and network-attached devices residing on the portion of the network under analysis, along with their available services. The techniques used for target acquisition are network mapping and port mapping

5.1.1 Network mapping

This is also referred to as *IP scanning* or *host discovery*. There are different ways of performing network mapping under different constraint and conditions such as using (a) system-provided information (i.e. ping and traceroute, ICMP queries, routing tables, DNS interrogation with nslookup, DNS zone transfers, etc). (b) specific tools (i.e nmap, fping, pinger, etc). Examples of network mapping is shown in fig 1.



- ICMP protocol:

```
# nmap -sP 10.0.1.1-254
Starting nmap (www.insecure.org/nmap/)
Host puma.mydomain.com (10.0.1.1) appears to be up.
[...]
Host neptune.mydomain.com (10.0.1.115) appears to be up.
Host iron.mydomain.com (10.0.1.216) appears to be up.
Nmap run completed -- 254 IP addresses (18 hosts up) scanned in 27 seconds
```

- TCP protocol (i.e. port 80):

```
# nmap -sP -P0 -PS80 10.0.1.1-254
TCP probe port is 80
Starting nmap (www.insecure.org/nmap/)
Host puma.mydomain.com (10.0.1.1) appears to be up.
Host aqua.mydomain.com (10.0.1.3) appears to be up.
[...]
Host iron.mydomain.com (10.0.1.216) appears to be up.
Nmap run completed -- 254 IP addresses (18 hosts up) scanned in 32 seconds
```

baseline for all the subsequent security activities

```
Target list:
10.0.1.1   #puma.mydomain.com
10.0.1.3   #aqua.mydomain.com
10.0.1.4   #silver.mydomain.com
10.0.1.12  #columbia.mydomain.com
```

Figure 1: Example of port mapping

Source: IBM Software Group.

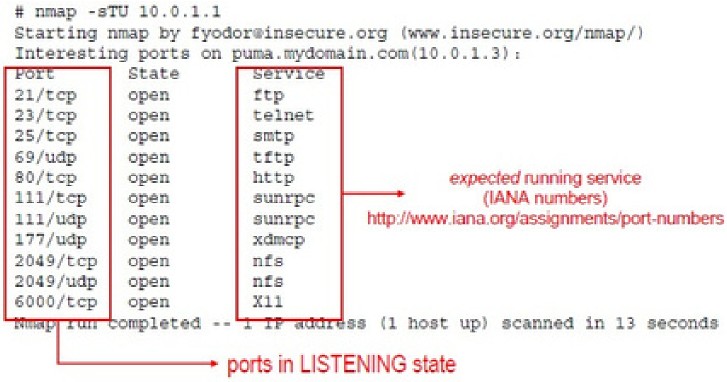
5.1.2 Port Mapping

Also known as port scanning is the process of connecting to TCP and UDP ports of the target system to identify (a) what TCP and UDP ports are in a LISTENING state. (b) what TCP and UDP services are running on the target system. (c) what RPC registered RPC programs are running on the target system. And (d) other services unintentionally expose. Port mapping is carried out with programs known as port mappers or port scanners such as nmap, strobe and netcat. Examples of port mapping is shown in fig 2.

Other Common Port Mapping Techniques used for Analysis

5.1.3 TCP connect() scanning

This method uses a complete 3-way handshake connection (SYN, SYN/ACK,ACK) carried out through the connect() system call to check the listening (i.e open) ports. The destination host can be configured to reset (RST) the connection if the port is closed.



```
# nmap -sTU 10.0.1.1
Starting nmap by fyodor@insecure.org (www.insecure.org/nmap/)
Interesting ports on puma.mydomain.com(10.0.1.3):
Port        State       Service
21/tcp      open        ftp
23/tcp      open        telnet
25/tcp      open        smtp
69/udp      open        tftp
80/tcp      open        http
111/tcp     open        sunrpc
111/udp     open        sunrpc
177/udp     open        xdmcp
2049/tcp    open        nfs
2049/udp    open        nfs
6000/tcp    open        X11
Nmap run completed -- 1 IP address (1 host up) scanned in 13 seconds
```

Figure 2: Example of network mapping (using nmap)

Source: IBM Software Group

5.1.4 SYN scanning

This is also known as ―half-open‖ scanning. This is where the source host sends a SYN packet. The destination host replies with a SYN/ACK if the port is listening. The destination host can be reset as needed

5.1.5 Fragmentation scanning

Since layer 4 protocol packets are hidden inside layer 3 packet fragments, they are less likely to be detected by the old packet filters such as ethereal. Hence, newer versions of packet filters like wireshakes can be used for filtering packets.

5.1.5 UDP scanning

In this method, the source host sends UDP datagram. If the port is closed the destination host sends back an ―ICMP port unreachable‖ message. Thus, the listening of UDP open ports can be determined.

5.2 Vulnerability Analysis

Vulnerability analysis is when programs are used to automatically identify not just host and open ports but any associated vulnerabilities instead of relying on human interpretation of the result as in the case of target acquisition [5].

Automated network-based vulnerability scanning programs assist in:

1. Extracting information from the target hosts (O.S. version, open ports, active services and protocols, version of each running service, exported resources and shares, valid accounts), this phase is also known as enumeration.
2. Checking all the details against publicly available sources of known vulnerability information and vendor security alert to see if known potential vulnerabilities may affect the host.
3. Performing test and heuristics to confirm the existence of a real vulnerability (whenever possible and according to the level of aggression chosen)
4. Rating the risk of the vulnerability.
5. Mapping each finding to their related security advisory or alert
6. Providing fixing direction
7. Creating reports

Network-based vulnerability scanning programs

□ARC SARA – Security Auditor’s Research Assistant
□ eEye Digital Security Retina
□ CyberCop Scanner
□ ISS Internet Security Scanner (Pier, 2018)
□ Nessus
□ OpenVas
□ Enterprise Configuration Manager
□ Symantec Enterprise Security Manager (Irfan, 2018)

6. Vulnerabilities of the Network

Network vulnerabilities are considered as those things that pose a potential avenue for attack or security breach against the network. The Network vulnerabilities that were discussed in one case study were as follows: [11]

6.1 Insecure / exposed ports

The TCP/UDP were all configured as open ports thereby accepting all kinds of packet that come through the network. Most of the system’s USB ports were opened without restrictions. Users can plug in flash drive from any source on any system on the network. The network is therefore open to both external attacks DOS and internal attacks like malicious worms

6.2 Indiscreminate enabling of service

Unused and irrelevant services were left enables on the network device. This can serve as a hole to the network. The proper thing to do is to disable all unused services

6.3 Inproper system configuration

The following configuration flaws were noticed with the network. (a) Unsecured user account (b) Easier guessed password (used across all the network devices (c) Misconfigured services where default settings were used in running configurations and misconfigured network equipments. (e.g. multi-layer 10g Ethernet switch, 48-port 3750g Cisco switch, Cisco SRW 2024P, 24-ports gigabit switch, cisco Linksys P1000 wireless-N router, etc)

6.4 Poor anti-virus implementation

Most of the anti-virus like MaCaffee, Kaspaschy and Avast anti-virus used on network device are out of date. Some devices are used without anti-virus update. This makes the system open to malware attack.

6.5 Poor firewall deployment

The installed firewall on the network was without any external or additional intrusion detection system (IDS) / sensor. This makes the network to rely only on the firewall internal intrusion prevention system (IPS). Virtually any attack that bypasses the firewall attacks all the devices on the network

6.6 Poor intrusion detection system (IDS) setups

The IDS used was the network based IDS through poorly configuration because most of the settings were still left in the default setting and was placed behind the firewall that provide access to internal network

6.7 Week password implementation

Most password used on network devices were too short and consist of only alphabets. This can be easily brute forced or guessed. This means the network is subjected to various password attacks.

6.8 Easy access to information

Access to network information does not follow any security procedure as privilege levels were not properly implemented. Most users have equal clearance / right to network information and stored data on the network.

6.9 Downloading of files and applications from untrusted sites

No restriction on downloads and on sites to visit. This can serve as a hole for attack especially virus, worms, and other malicious software

6.10 Unsecured application / programs as a result of poor programming / practices

Most of the applications running on the network devices were designed without security consciousness. They are mostly used on the network without patches.

6.11 Application back doors

Too many back doors to different applications are running on the network devices. This may serve as a hole to any form of attack especially attacks launched by any skilled programmer

6.12 Lack of appropriate security policies

Available network security policy was not appropriately documented alongside with the network design.

6.13 Poor attention to security indications

Most users don't give proper attention to the warning messages or security indicators on the network

6.14 Disgruntled employees

Most of the staff in the ICT department do not receive their salaries as and at when due. This makes them open to doing anything for money, even if it is changing students GPA

6.15 Lack of efficient physical security

Apart from the entrance gate to the school, there is no other physical security mechanism in place for securing the network activities, devices, staff and students data

6.16 Insufficient security training and awareness

Poor security training of staff working directly with the school network was observed

6.17 Carelessness on the part of users

Most users fail to log out after login in to a system

6.18 Missing patches in Operating System

This may allow a rude insider to run an unauthenticated command prompt or other backdoor path into the web environment

6.19 Lack of written security policy

An unwritten security policy cannot be consistently be applied or enforced. This was noticed with the network under review.

6.20 Poor logical access controls

Inadequate monitoring and auditing within the network and its infrastructure was observed. This allows attacks and unauthorized use to continue, wasting the school resources. This could result in legal action or termination against IT technicians, IT management, or even other leadership that allow this unsafe condition to persist.

6.21 Unprocedure software / hardware installations

It was noticed that some units within the network section of the organization make an unauthorized changes to the network topology or to the installation of unapproved applications. This can create security holes

6.22 Poor / no disaster recovery plan

The organization's ICT unit has no proper discovery plan for now. This may allow chaos, panic and confusion to occur when someone attacks the organization's network.

6.23 Little politics was observed within the ICT unit

This may lead to inconsistent security policy because they are not working as a team.

7. Limitations

1. Analysis of network vulnerabilities in this study does not represent the entire analysis and hence may not be generalized
2. Vulnerabilities of networks covered in this study points out to a particular case study. Other vulnerabilities may be available in other instances.

8. Recommendations

1. Broad study on analysis of network vulnerabilities that would accommodate more approaches for evaluating networks should be worked on.
2. Other networks situations should be studied to uncover vulnerabilities of networks.

9. Conclusion

Patch management and antivirus protection are only the first step in securing a network. A good vulnerability assessment is the next logical move. Networks are a dynamic entity, they evolve and change constantly. A vulnerability assessment should be set to run constantly and inform the administrator every time change is detected to make the utmost of network security protection.

References

[2] Isaca. ―Security Vulnerability Assessment.‖ Internet: https://cybersecurity.isaca.org/info/cyber-aware/images/ISACA_WP_Vulnerability_Assessment_1117.pdf, 2017 [Jun. 19, 2018].

[3] E. Carabott. ―Why You Need to Run a Vulnerability Assessment.‖ Internet:

https://techtalk.gfi.com/vulnerability-assessment/, 2011 [Jun. 19, 2018].

[4] R. Bond. ―The Benefits of a Vulnerability Assessment.‖ Internet: https://www.hitachi-systems-security.com/blog/the-benefits-of-a-vulnerability-assessment/, 2017 [Jun. 19, 2018].

[5] L. Pier. ―Network-based vulnerability assessment.‖ Internet: https://pdfs.semanticscholar.org/presentation/c504/91c807ee198ae5825b026de1a4bacb97c473.pdf, 2018 [Jun. 19, 2018].

[6] N. Poolsappasit. ―Towards An Efficient Vulnerability Analysis Methodology For Better Security Risk Management.‖ Department of Computer Science, Colorado State University, Internet: https://dspace.library.colostate.edu/bitstream/handle/10217/40477/Poolsappasit_colostate_0053A_10071.pdf;sequence=1 2010 [Jun. 20, 2018].

[7] R. Noam. ―Automating Vulnerability Assessment, Beyond Security.‖ Internet: http://beyondsecurity.com/ AVDS_Whitepaper.pdf, 2018. [Jun. 20, 2018].

[8] S. Zhang, O. Xinming, & H. John. ―Effective Network Vulnerability Assessment through Model Abstraction.‖ Internet: http://people.cs.ksu.edu/~zhangs84/papers/dimva11.pdf, 2011 [Jun. 20, 2018].

[9] P. Cynthia, & P. Laura. ―A Graph-Based System for Network-Vulnerability Analysis.‖ Internet: http://web2.utc.edu/~djy471/CPSC4660/graph-vulnerability.pdf, 1999 [Jun. 20, 2018].

[10] R. H. Ronda. & L. F. Kelvin. ―The Network Vulnerability Tool (NVT) – A System Vulnerability Visualization Architecture.‖ Internet: https://csrc.nist.gov/csrc/media/publications/conference-paper/1999/10/21/proceedings-of-the-22nd-nissc-1999/documents/papers/p7.pdf, 1999 [Jun. 20, 2018].

[11] J. C. Iyiama. CSC 911. Class Lecture, Topic: ―Vulnerabilities of Network.‖ Presco Campus, Department of Computer Science, Ebonyi State University, Nigeria, Sep. 14, 2018.