

Web Vulnerability Scanning

Made By : Priyanshu Saklani (RA1811030010060)
Nitin Chauhan (RA1811030010061)

Abstract:

The world is exceedingly reliant on the Internet. Nowadays, web security is biggest challenge in the corporate world. It is considered as the principle framework for the worldwide data society.

Web applications are prone to security attacks. Web security is securing a web application layer from attacks by unauthorized users. A lot of the issues that occur over a web application is mainly due to the improper input provided by the client. This paper discusses the different aspects of web security and it's weakness.

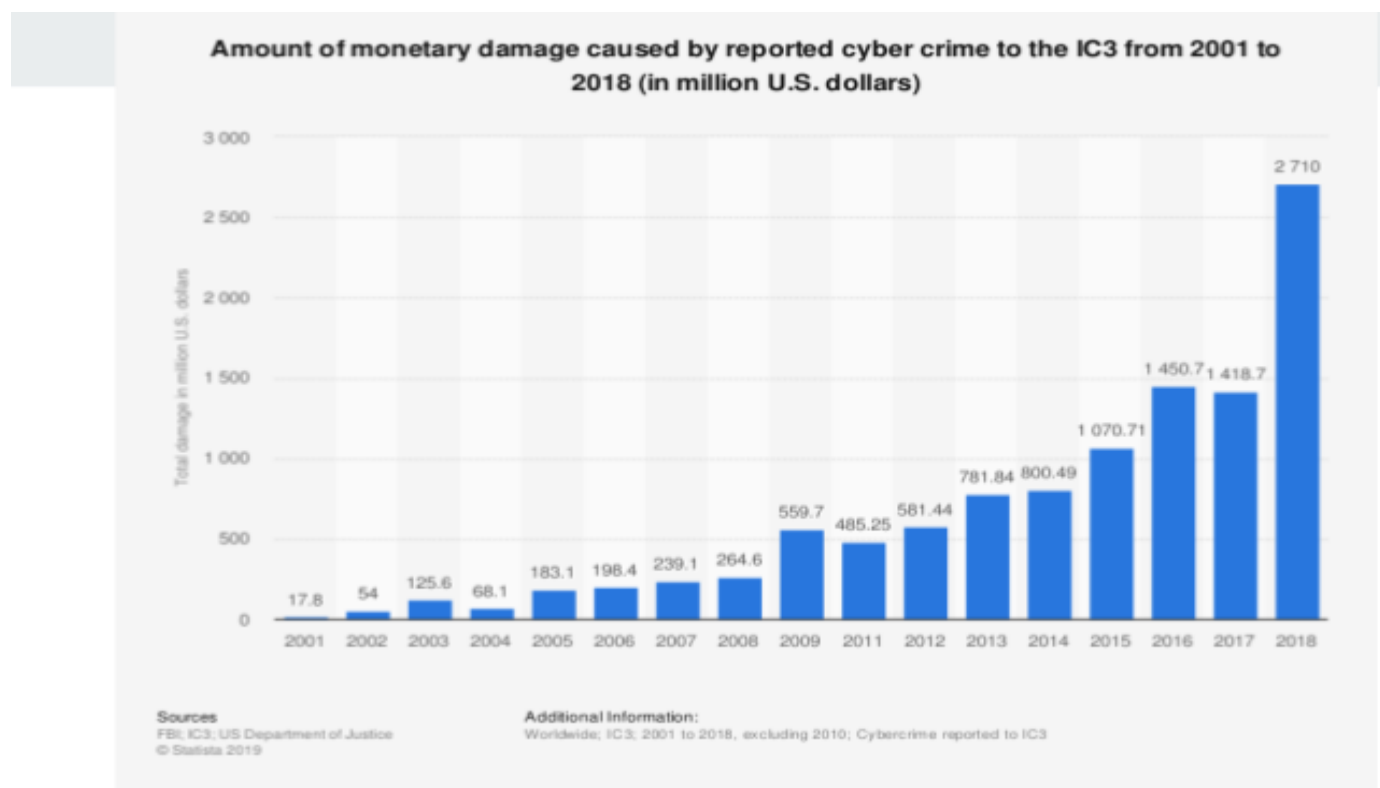
The main elements of web security techniques such as the passwords, encryption, authentication and integrity are also discussed in this paper. The anatomy of a web application attack and the attack techniques are also covered in details.

This report explores a number of methods for combatting this class of threats and assesses why they have not proven more successful. This paper proposes a better way for minimizing these type of web vulnerabilities. It also provides the best security mechanisms for the said attacks.

INTRODUCTION:

Over the last years the number of IT security incidents has been constantly increasing among companies and personal computers.

Data breaches are becoming normal day by day as globe progresses towards information technology . News like Facebook data breach impacts 500 million users and others like WannaCry Ransomware attack is surfacing all over .Data security on the internet is synonymous with a website and a computer network that connects to one another. In the context of computer networks, any existing data on a computer that is connected to another computer, is unsafe, so need to do some way to secure the data so that cannot be accessed by another computer.



Domain Study :

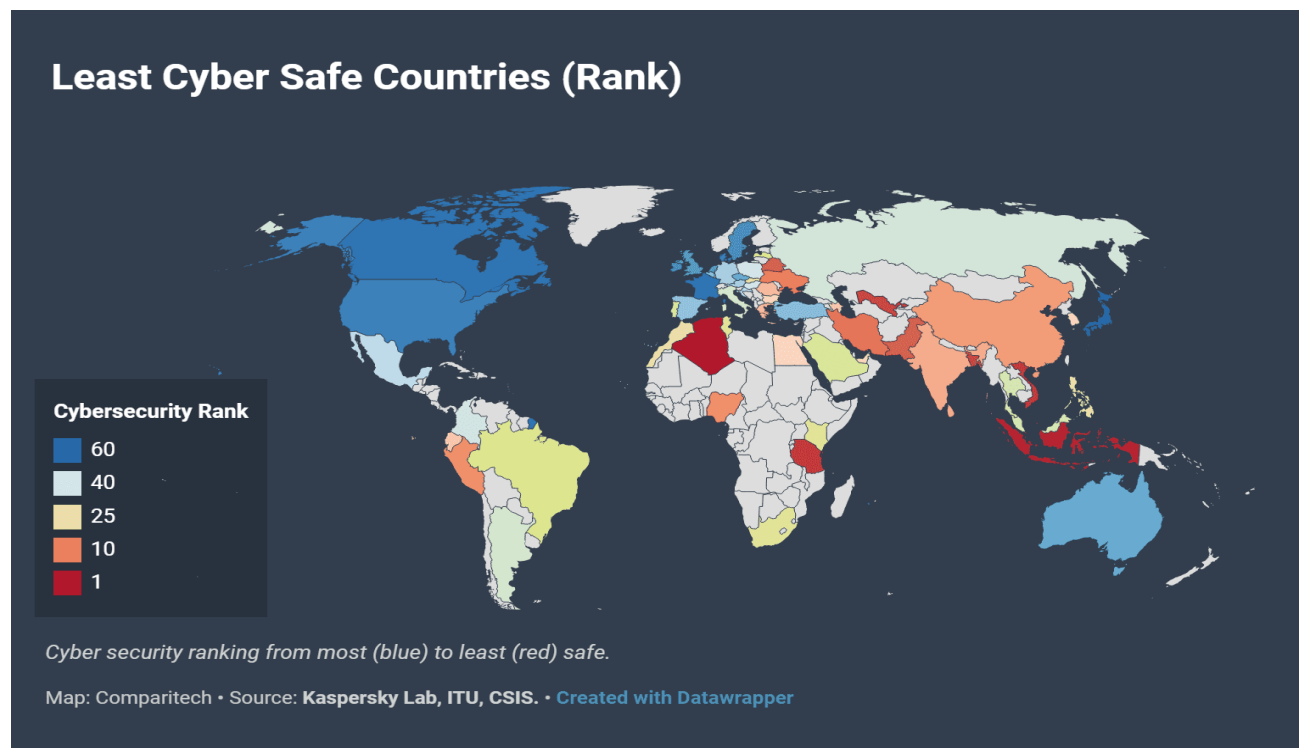
As we progress to live in the age of information, basically the age of IT, we must also take in consideration the malicious things it brings with it . About 80 % percent of countries are vulnerable to cyber attacks which not only brings down the reputation of the country as a whole but also lead to economic problems , less trust by other countries and increase in risk of future cyber attacks .

List of those countries with lowest malware infection rates in computers

Sweden-19.88%, Finland- 20.65% ,Norway-21.63% ,Japan-22.24%
,Belgium-22.78%, United Kingdom-23.38%,
Switzerland-23.94%,Germany-24.12%,Denmark-24.34%.

Now the list of those with highest malware infection rates in computers

China-49%,Taiwan-47.34%,Turkey-40.99%,Russia-38.95%,Guatemala-37.56%,
Mexico-36.89%, Peru-36.23%, Ecuador-36.22%, Brazil-34.68%.



LITERATURE SURVEY

This report performs a detailed analysis of existing studies on web service security published in journals/conferences

[21] DOS attack	An automated plug-in based on Black box testing is proposed that can evaluate DOS attacks on web services.	Other
[22] DOS attack	Propose an approach, for identifying DOS attacks on web services, which are intrusion tolerant.	Attack detection
[23] DOS attack	A gateway system is proposed based on Schema hardening as well as WSDL Compiler to protect the services from DOS attack by filtering malicious SOAP Messages.	Attack prevention
[24] SQL/XML injection	An architecture as well as filtering policy is proposed and tested to prevent web service attacks such as Injection, Coercive parsing.	Attack prevention
[25] DOS attack	A Vector Quantization based Intrusion detection system for DOS Attacks on web services to achieve better true detection rates.	Attack detection
[26] DOS attack	A mechanism is proposed with adaptive rule updates to detect and mitigate DOS Attacks.	Attack detection
[27] XML Signature injection	An approach of Schema hardening is discussed for fending XML signature attacks.	Attack prevention
[28] SQL injection, XPath injection	An approach to prevent SQL/XPath Injection attacks on web services by combining statement learning as well as service protection.	Attack prevention
[29] SQL injection, XPath injection	A Learning based approach, to detect XPath & SQL Injection attacks based on the concept of anomaly detection.	Attack prevention
[30] DOS attack	A proposed approach for testing the security of service platforms against DOS Attacks, by multi-phase testing.	Other
[31] XML injection	A hybrid approach that applies ontology on the knowledge database for knowledge based detection of XML Injection attacks on web services.	Attack detection
[32] XML injection, DOS attack	Identify the different attacks on web services, suggest techniques to counter the attacks as well as develop a self-adaptive hardening scheme for message validation.	Other
[33] DOS attack, XML injection	A series of tests are conducted on SOAP requests to identify possible attacks on web services.	Attack detection
[34] XML injection	A XML Injection attack detection method based on XML based SOAP message tree verification.	Attack detection
[35] DOS attack	A scheme is identified for defending web services against DOS as well as XML based DOS attack.	Attack detection/Attack prevention

[1]	XPath injection	An architecture that uses a run-time monitoring mechanism to identify malicious queries thus preventing XPath Injection.	Attack prevention
[2]	SQL injection	SQL vulnerabilities in web services is detected based on mutation operator related automated testing approach.	Vulnerability detection
[3]	DOS attack	An adaptable algorithm for testing web services by parsing incoming XML messages for DOS attack.	Attack detection
[4]	SQL,XPath injection	A comparison of existing vulnerability scanners against 300 public web services to identify security flaws.	Vulnerability detection
[5]	SQL,XPath injection	An automated approach for XPath/SQL injection vulnerability detection in web services.	Vulnerability detection
[6]	SQL injection	A systematic approach for web services to detect SQL injection vulnerabilities using penetration testing tool.	Vulnerability detection
[7]	DOS attack	Model an architecture to apply on web services and create a filter defense system to protect against XML based DOS.	Attack detection/Attack prevention
[8]	XML injection attack	An approach is proposed with pluggable API as well as security services in the middleware to detect and overcome XML Injection attacks.	Attack detection
[9]	Spoofing, DOS	An automated pluggable API Model for network level threat detection was proposed.	Attack detection
[10]	DOS attack	A proposed real time agent based classification mechanism for detecting and preventing DOS attacks on web services.	Attack detection/Attack prevention
[11]	SQL,XML, XPath injection	A novel approach to complement existing vulnerability detection by forming sound and precise slices thus identifying false and true positives.	Other
[12]	Spoofing	Proposed a misuse pattern called Spoofing web services to prevent the same attack on web services.	Attack prevention
[13]	XML injection	A hybrid learning, universal approximator model is proposed to detect XML SOAP based attacks on web services.	Attack detection/Attack prevention
[14]	SQL,XML injection DoS attack	An intrusion detection systems based on fuzzy rules is suggested to prevent Injection as well as Denial-of-service attacks.	Attack prevention
[15]	DOS attack	A content introspection framework is suggested to prevent XML based Denial-of- service attack on web services.	Attack prevention

OBJECTIVE OF THE WORK

Objective of this report is to make every individual aware about the cyber threats which going around the world right now and also make them self sufficient to scan vulnerabilities in their home networks fr possible threats and other malicious activities .

This will not only keep them safe in their home environment but will also be a good practice for their official work.

Tools generates graphical reports that identify application vulnerabilities and exposure risks, and ranks the priority of threats. Tools also can perform an advanced analysis of your site structure, content and configuration to identify inherent exposure to future or emerging threats

PROPOSED MODEL

We will talk about two Security Tools namely

1.NMAP

2.Metasploit

NMAP

Nmap (*Network Mapper*) is a free and open-source network scanner created by Gordon Lyon (also known by his pseudonym *Fyodor Vaskovich*). Nmap is used to discover hosts and services on a computer network by sending packets and analyzing the responses.

Nmap provides a number of features for probing computer networks, including host discovery and service and operating system detection. These features are extensible by scripts that provide more advanced service detection, vulnerability detection, and other features. Nmap can adapt to network conditions including latency and congestion during a scan.

Nmap started as a Linux utility and was ported to other systems including Windows, macOS, and BSD. It is most popular on Linux, followed by Windows.

Features

Nmap features include:

- Host discovery – Identifying hosts on a network. For example, listing the hosts that respond to TCP and/or ICMP requests or have a particular port open.
- Port scanning – Enumerating the open ports on target hosts.
- Version detection – Interrogating network services on remote devices to determine application name and version number.
- TCP/IP stack fingerprinting – Determining the operating system and hardware characteristics of network devices based on observations of network activity of said devices.

- Scriptable interaction with the target – using Nmap Scripting Engine¹ (NSE) and Lua programming language.

Nmap can provide further information on targets, including reverse DNS names, device types, and MAC addresses.¹

Typical uses of Nmap:

- Auditing the security of a device or firewall by identifying the network connections which can be made to, or through it
- Identifying open ports on a target host in preparation for auditing.
- Network inventory, network mapping, maintenance and asset management.
- Auditing the security of a network by identifying new servers.
- Generating traffic to hosts on a network, response analysis and response time measurement.
- Finding and exploiting vulnerabilities in a network.
- DNS queries and subdomain search

Output

Nmap provides four possible output formats. All but the interactive output is saved to a file. Nmap output can be manipulated by text processing software, enabling the user to create customized reports.

Interactive

presented and updated real time when a user runs Nmap from the command line. Various options can be entered during the scan to facilitate monitoring.

XML

a format that can be further processed by XML tools. It can be converted into a HTML report using XSLT.

Grepable

output that is tailored to line-oriented processing tools such as grep, sed, or awk.

Normal

the output as seen while running Nmap from the command line, but saved to a file.

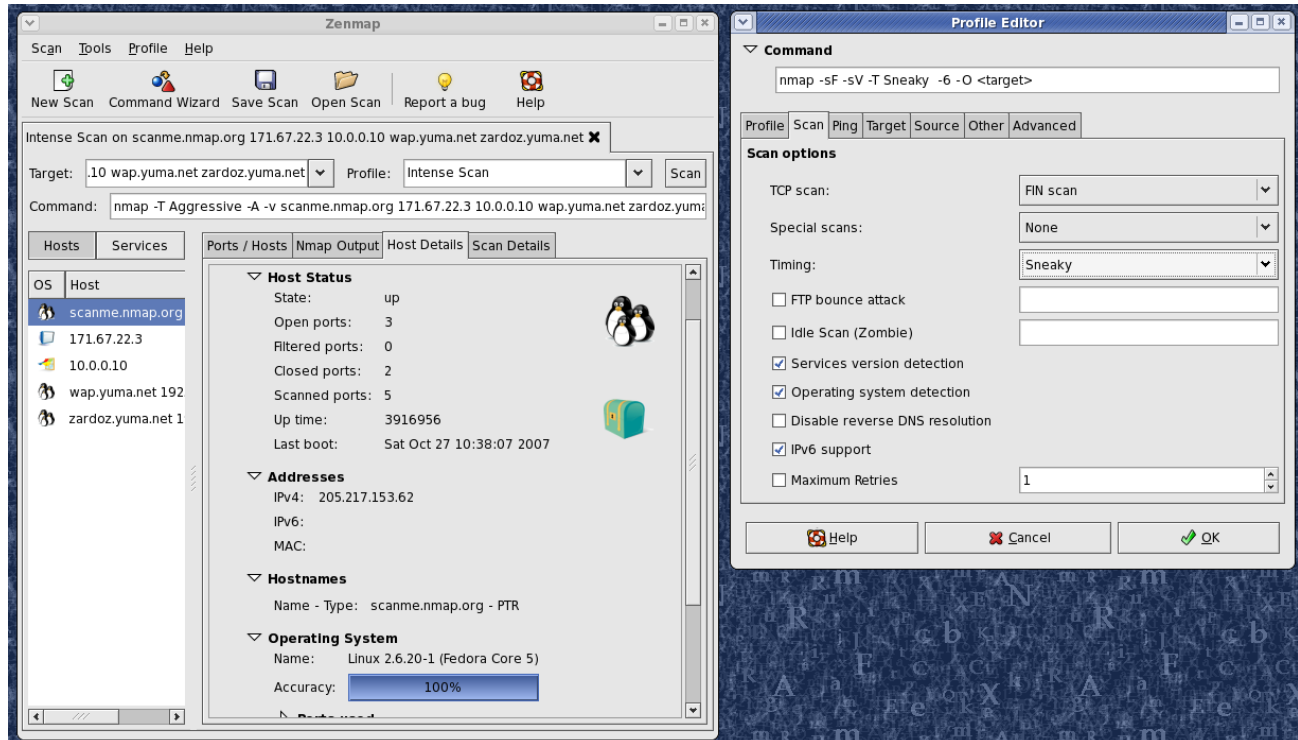
Download and Installation

Nmap.org is official web page for Nmap

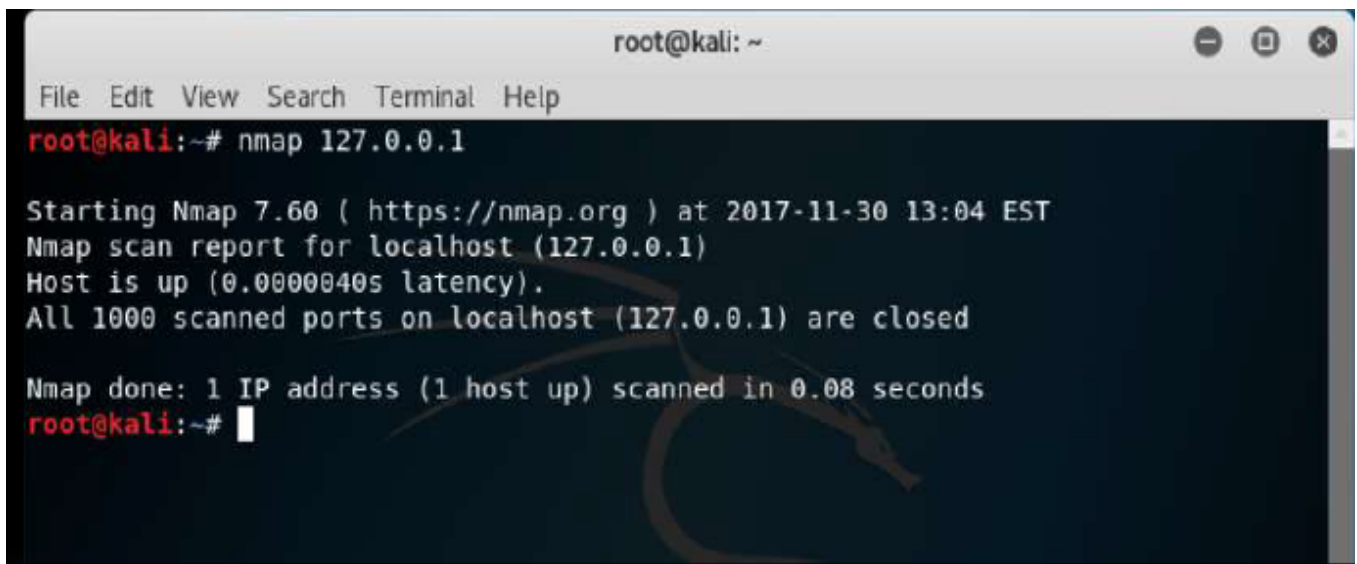
<https://nmap.org/download.html> is the link to the download section of the web site

It is available for all major OS like Windows , Linux , Solaris ,etc.

Now we'll use NMAP to scan network to check for different vulnerabilities . We get IP address of the machine which is going to be scanned ,and use different NMAP



S



Using Nmap: IP Address

Using Nmap: Scanning Network

```
root@kali: ~  
File Edit View Search Terminal Help  
root@kali:~# nmap 10.0.2.0/24  
  
Starting Nmap 7.60 ( https://nmap.org ) at 2017-11-30 13:14 EST  
Nmap scan report for 10.0.2.1  
Host is up (0.000056s latency).  
Not shown: 999 closed ports  
PORT      STATE SERVICE  
53/tcp    open  domain  
MAC Address: 52:54:00:12:35:00 (QEMU virtual NIC)  
  
Nmap scan report for 10.0.2.2  
Host is up (0.00029s latency).  
Not shown: 997 closed ports  
PORT      STATE SERVICE  
631/tcp   open  ipp  
49152/tcp open  unknown  
49153/tcp open  unknown  
MAC Address: 52:54:00:12:35:00 (QEMU virtual NIC)  
  
Nmap scan report for 10.0.2.3  
Host is up (0.00053s latency).  
All 1000 scanned ports on 10.0.2.3 are filtered  
MAC Address: 08:00:27:A4:C2:07 (Oracle VirtualBox virtual NIC)
```

Nmap: Scripts

```
root@kali: ~  
File Edit View Search Terminal Help  
root@kali:~# nmap -sC scanme.nmap.org  
  
Starting Nmap 7.60 ( https://nmap.org ) at 2017-11-30 13:58 EST
```

Nmap tells you about the open ports and number of devices connected to the network .

Close the ports which are not patched as they are vulnerable to exploits and tally the number of machines connected to the network to check if any external sources are present in the topology.

Metasploit Project

The **Metasploit Project** is a computer security project that provides information about security vulnerabilities and aids in penetration testing and IDS signature development. It is owned by Boston, Massachusetts-based security company **Rapid7**.

It's best-known subproject is the open-source **Metasploit Framework**, a tool for developing and executing exploit code against a remote target machine. Other important sub-projects include the Opcode Database, shellcode archive and related research.

The Metasploit Project includes anti-forensic and evasion tools, some of which are built into the Metasploit Framework. Metasploit is pre-installed in the Kali Linux operating system.

The basic steps for exploiting a system using the Framework include.

1. Optionally checking whether the intended target system is vulnerable to an exploit.
2. Choosing and configuring an *exploit* (code that enters a target system by taking advantage of one of its bugs; about 900 different exploits for Windows, Unix/Linux and macOS systems are included).
3. Choosing and configuring a *payload* (code that will be executed on the target system upon successful entry; for instance, a remote shell or a VNC server).
Metasploit often recommends a payload that should work.

4. Choosing the encoding technique so that hexadecimal opcodes known as "bad characters" are removed from the payload, these characters will cause the exploit to fail.
5. Executing the exploit.

This modular approach – allowing the combination of any exploit with any payload – is the major advantage of the Framework. It facilitates the tasks of attackers, exploit writers and payload writers.

Metasploit runs on Unix (including Linux and macOS) and on Windows. The Metasploit Framework can be extended to use add-ons in multiple languages.

To choose an exploit and payload, some information about the target system is needed, such as operating system version and installed network services. This information can be gleaned with port scanning and TCP/IP stack fingerprinting tools such as Nmap.

Vulnerability scanners such as Nessus, and OpenVAS can detect target system vulnerabilities. Metasploit can import vulnerability scanner data and compare the identified vulnerabilities to existing exploit modules for accurate exploitation.

Following is a list of important terms used in the field of hacking.

Adware – Adware is software designed to force pre-chosen ads to display on your system.

Attack – An attack is an action that is done on a system to get its access and extract sensitive data.

Back door – A back door, or trap door, is a hidden entry to a computing device or software that bypasses security measures, such as logins and password protections.

Bot – A bot is a program that automates an action so that it can be done repeatedly at a much higher rate for a more sustained period than a human operator could do it. For example, sending HTTP, FTP or Telnet at a higher rate or calling script to create objects at a higher rate.

Botnet – A botnet, also known as zombie army, is a group of computers controlled without their owners' knowledge. Botnets are used to send spam or make denial of service attacks.

Brute force attack – A brute force attack is an automated and the simplest kind of method to gain access to a system or website. It tries different combination of usernames and passwords, over and over again, until it gets in.

Buffer Overflow – Buffer Overflow is a flaw that occurs when more data is written to a block of memory, or buffer, than the buffer is allocated to hold.

Clone phishing – Clone phishing is the modification of an existing, legitimate email with a false link to trick the recipient into providing personal information.

Cracker – A cracker is one who modifies the software to access the features which are considered undesirable by the person cracking the software, especially copy protection features.

Denial of service attack (DoS) – A denial of service (DoS) attack is a malicious attempt to make a server or a network resource unavailable to users, usually by temporarily interrupting or suspending the services of a host connected to the Internet.

DDoS – Distributed denial of service attack.

Exploit Kit – An exploit kit is software system designed to run on web servers, with the purpose of identifying software vulnerabilities in client machines communicating with it and exploiting discovered vulnerabilities to upload and execute malicious code on the client.

Exploit – Exploit is a piece of software, a chunk of data, or a sequence of commands that takes advantage of a bug or vulnerability to compromise the security of a computer or network system.

Firewall – A firewall is a filter designed to keep unwanted intruders outside a computer system or network while allowing safe communication between systems and users on the inside of the firewall.

Keystroke logging – Keystroke logging is the process of tracking the keys which are pressed on a computer (and which touchscreen points are used). It is simply the map of a computer/human interface. It is used by gray and black hat hackers to record login IDs and passwords. Keyloggers are usually secreted onto a device using a Trojan delivered by a phishing email.

Logic bomb – A virus secreted into a system that triggers a malicious action when certain conditions are met. The most common version is the time bomb.

Malware – Malware is an umbrella term used to refer to a variety of forms of hostile or intrusive software, including computer viruses, worms, Trojan horses, ransomware, spyware, adware, scareware, and other malicious programs.

Master Program – A master program is the program a black hat hacker uses to remotely transmit commands to infected zombie drones, normally to carry out Denial of Service attacks or spam attacks.

Phishing – Phishing is an e-mail fraud method in which the perpetrator sends out legitimate-looking emails, in an attempt to gather personal and financial information from recipients.

Phreaker – Phreakers are considered the original computer hackers and they are those who break into the telephone network illegally, typically to make free longdistance phone calls or to tap phone lines.

Rootkit – Rootkit is a stealthy type of software, typically malicious, designed to hide the existence of certain processes or programs from normal methods of detection and enable continued privileged access to a computer.

Shrink Wrap code – A Shrink Wrap code attack is an act of exploiting holes in unpatched or poorly configured software.

Social engineering – Social engineering implies deceiving someone with the purpose of acquiring sensitive and personal information, like credit card details or user names and passwords.

Spam – A Spam is simply an unsolicited email, also known as junk email, sent to a large number of recipients without their consent.

Spoofing – Spoofing is a technique used to gain unauthorized access to computers, whereby the intruder sends messages to a computer with an IP address indicating that the message is coming from a trusted host.

Spyware – Spyware is software that aims to gather information about a person or organization without their knowledge and that may send such information to another entity without the consumer's consent, or that asserts control over a computer without the consumer's knowledge.

SQL Injection – SQL injection is an SQL code injection technique, used to attack data-driven applications, in which malicious SQL statements are inserted into an entry field for execution (e.g. to dump the database contents to the attacker).

Threat – A threat is a possible danger that can exploit an existing bug or vulnerability to compromise the security of a computer or network system.

Trojan – A Trojan, or Trojan Horse, is a malicious program disguised to look like a valid program, making it difficult to distinguish from programs that are supposed to be there designed with an intention to destroy files, alter information, steal passwords or other information.

Virus – A virus is a malicious program or a piece of code which is capable of copying itself and typically has a detrimental effect, such as corrupting the system or destroying data.

Vulnerability – A vulnerability is a weakness which allows a hacker to compromise the security of a computer or network system.

Worms – A worm is a self-replicating virus that does not alter files but resides in active memory and duplicates itself.

Cross-site Scripting – Cross-site scripting (XSS) is a type of computer security vulnerability typically found in web applications. XSS enables attackers to inject client-side script into web pages viewed by other users.

Zombie Drone – A Zombie Drone is defined as a hi-jacked computer that is being used anonymously as a soldier or 'drone' for malicious activity, for example, distributing unwanted spam e-mails.

Vulnerability scanning with Metasploit

Now we are going to see how to perform vulnerability assessments of network and web applications by using Metasploit built-in plug-ins. First we will start with OpenVAS; before jumping into msfconsole, you have to install OpenVAS in your system.

To run OpenVAS, type in load openvas in msfconsole and it will load and open the VAS plug-in from its database.

```
File Edit View Bookmarks Settings Help

      =[ metasploit v4.8.1-1 [core:4.8 api:1.0]
+ -- --=[ 1231 exploits - 751 auxiliary - 205 post
+ -- --=[ 324 payloads - 31 encoders - 8 nops

msf > load openvas
[*] Welcome to OpenVAS integration by kost and averagesecurityguy.
[*]
[*] OpenVAS integration requires a database connection. Once the
[*] database is ready, connect to the OpenVAS server using openvas_connect.
[*] For additional commands use openvas_help.
[*]
[*] Successfully loaded plugin: OpenVAS

root : .ruby.bin
```

Now type in openvas_help and it will show all usage commands for OpenVAS.

```
File Edit View Bookmarks Settings Help

msf > openvas_help
[*] openvas_help          Display this help
[*] openvas_debug          Enable/Disable debugging
[*] openvas_version        Display the version of the OpenVAS server
[*]
[*] CONNECTION
[*] =====
[*] openvas_connect         Connects to OpenVAS
[*] openvas_disconnect      Disconnects from OpenVAS
[*]
[*] TARGETS
[*] =====
[*] openvas_target_create    Create target
[*] openvas_target_delete    Deletes target specified by ID
[*] openvas_target_list      Lists targets
[*]
[*] TASKS
[*] =====
[*] openvas_task_create      Create task
[*] openvas_task_delete      Delete a task and all associated reports
[*] openvas_task_list        Lists tasks
[*] openvas_task_start       Starts task specified by ID
[*] openvas_task_stop        Stops task specified by ID
[*] openvas_task_pause       Pauses task specified by ID
[*] openvas_task_resume      Resumes task specified by ID
[*] openvas_task_resume_or_start Resumes or starts task specified by ID
[*]
[*] CONFIGS
[*] =====
[*] openvas_config_list      Lists scan configurations
[*]
[*] FORMATS
[*] =====
[*] openvas format list      Lists available report formats

root : .ruby.bin
```

We have to connect our OpenVAS to its server by giving the command `openvas_connect` and it will show the full usage command, which is `openvas_connect username password host port <ssl-confirm>` for connecting to the server. In my case, the command is `openvas_connect rohit toor localhost 9390 ok`

```
File Edit View Bookmarks Settings Help
msf > openvas_connect
[*] Usage:
[*] openvas connect username password host port <ssl-confirm>
msf > openvas_connect rohit toor localhost 9390 ok
[*] Connecting to OpenVAS instance at localhost:9390 with username rohit...
[+] OpenVAS connection successful

root : .ruby.bin
```

As can we can see in the above figure, our OpenVAS connection is successful. Now we will create a target for scanning. The command for creating a target is `openvas_target_create <scan name> <target IP> <any comments>`. In the below figure, we can see my scan name is windows7, the target is 192.168.0.101 and the comment is new_scan, so the command is `openvas_target_create "windows7" 192.168.0.101 "new_scan"`

```
File Edit View Bookmarks Settings Help
msf > openvas target create "windows7" 192.168.0.101 "new scan"
[*] OK, resource created: 178aa13a-31cc-4509-984c-7b1b05a8b766
[+] OpenVAS list of targets

ID  Name      Hosts      Max Hosts  In Use  Comment
--  -
0   Localhost localhost  1          1
1   windows7  192.168.0.101  1          0      new_scan

msf >

root : .ruby.bin
```

After creating the target, we want to see the OpenVAS's scan configuration list, so type in `openvas_config_list`.

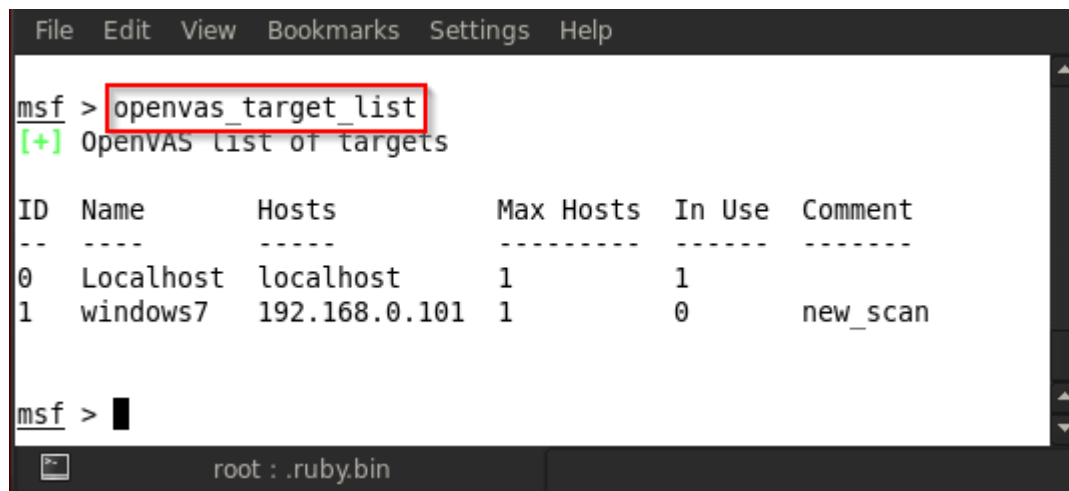


```
msf > openvas_config_list
[+] OpenVAS list of configs

ID  Name
--  ---
0   Full and fast
1   Full and fast ultimate
2   Full and very deep
3   Full and very deep ultimate
4   empty

msf > 
```

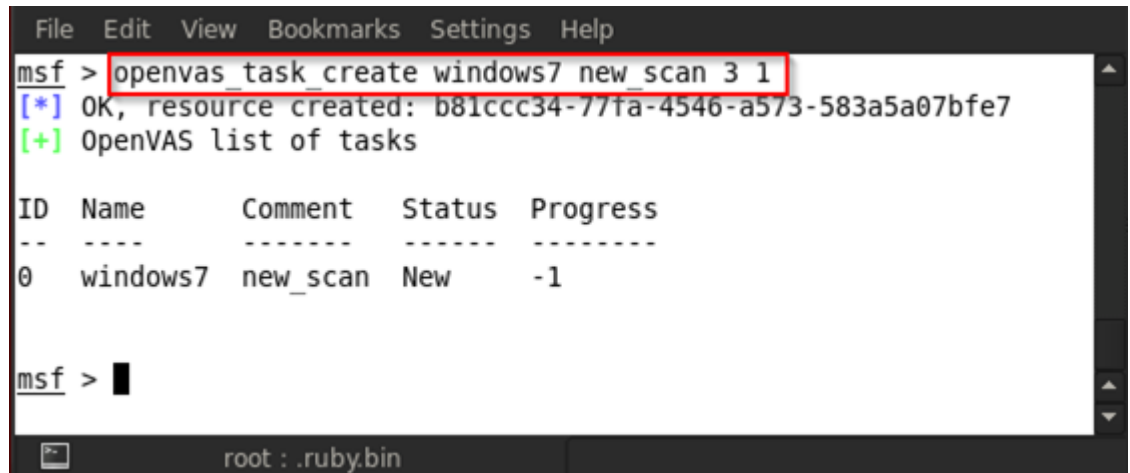
OpenVAS has four types of scan configuration; we will select this as per requirement. Next type in `openvas_target_list` and it will show your created targets



```
msf > openvas_target_list
[+] OpenVAS list of targets

ID  Name      Hosts      Max Hosts  In Use  Comment
--  ---      -
0   Localhost localhost  1         1       new_scan
1   windows7  192.168.0.101  1         0
```

Now we have a target and we have also seen the scan configuration, so we will create a task for scanning our target machine.



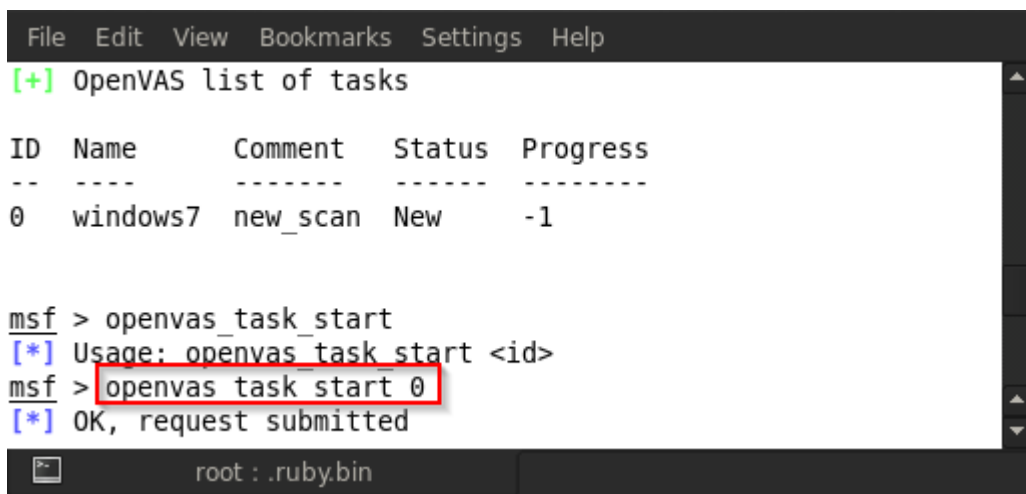
```
File Edit View Bookmarks Settings Help
msf > openvas_task_create windows7 new_scan 3 1
[*] OK, resource created: b81ccc34-77fa-4546-a573-583a5a07bfe7
[+] OpenVAS list of tasks

ID   Name      Comment   Status   Progress
--   -
0    windows7  new_scan  New      -1

msf >
```

root : .ruby.bin

Go create a task, the command is `openvas_task_create <scanname> <comment> <scanconfig ID> <targetID>`. For example, in the above figure, we type in `openvas_task_create windows7 new_scan 3 1`. We can see that our task is created and the task ID is 0 for our target machine. Now start the task by typing in `openvas_task_start <taskID>`. Here we are using `openvas_task_start 0`.



```
File Edit View Bookmarks Settings Help
[+] OpenVAS list of tasks

ID   Name      Comment   Status   Progress
--   -
0    windows7  new_scan  New      -1

msf > openvas_task_start
[*] Usage: openvas_task start <id>
msf > openvas task start 0
[*] OK, request submitted

root : .ruby.bin
```

```
File Edit View Bookmarks Settings Help

msf > openvas_task_list
[+] OpenVAS list of tasks

ID  Name      Comment    Status    Progress
--  -
0   windows7  new_scan   Running   80

msf >
```

root : .ruby.bin

The progress is now 80%, which means it's almost complete. When the scan is complete, the progress will show -1. and the status will show "Done."

```
File Edit View Bookmarks Settings Help

msf > openvas_task_list
[+] OpenVAS list of tasks

ID  Name      Comment    Status    Progress
--  -
0   windows7  new_scan   Done      -1

msf >
```

root : .ruby.bin

Our scan is completed now, so we can download the report; type in `openvas_report_list` and it will show all reports from its database.

```
File Edit View Bookmarks Settings Help

msf > openvas_report_list
[+] OpenVAS list of reports

ID  Task Name      Start Time                      Stop Time
--  -
0   Example task   Tue Aug 25 21:48:25 2009        Tue Aug 25 21:52:16 2009
1   windows7       Sun Dec 8 22:21:03 2013         Sun Dec 8 22:44:35 2013

msf >
```

root : .ruby.bin

There are several formats for downloading the report.
Type in `openvas_format_list` and it will list all available


```
msf > openvas_format_list
[+] OpenVAS list of report formats

ID  Name  Extension  Summary
--  -
0   CPE   csv        Common Product Enumeration CSV table.
1   HTML  html       Single page HTML report.
2   ITG   csv        German "IT-Grundschutz-Kataloge" report.
3   LaTeX tex       LaTeX source file.
4   NBE   nbe        Legacy OpenVAS report.
5   PDF   pdf        Portable Document Format report.
6   TXT   txt        Plain text report.
7   XML   xml        Raw XML report.

msf >
```

After choosing the format, we can download the report by using this command: `openvas_report_download <report id> <format id> <path for saving report> <report name>`. Here we are using `openvas_report_download 1 5 /root/Desktop report`

```
[*] Usage: openvas report download <report id> <format id> <path> <report_name>
msf > openvas_report_download 1 5 /root/Desktop report
[*] Saving report to /root/Desktop/report
msf >
```



The OpenVAS has a bug in the report format: Whenever I tried to download PDF or XML formats, it gives blank report, so again I download the report in HTML format and this format is working

Conclusion :

This study provides a comprehensive survey of existing methods in the research area of web applications vulnerabilities. We highlighted several open issues that still needs to be addressed. In this paper, we reviewed classification and detection of web vulnerabilities with different approaches like static analysis, dynamic analysis, hybrid analysis, combined three analyses for scanners and machine learning. We also reviewed various types of web vulnerabilities with different classification. The input validation vulnerabilities and improper session management and methods to perceive web vulnerabilities.

