



Sri Lanka Institute of Information Technology

Web Security Audit

IE2062 – Web Security

Domain

Verizon media – yahoo.com

Student Registration Number	Student Name
[REDACTED]	Ekanayake E.M.I.R

Table of Contents

Abstract.....	3
Introduction	4
What is a Web Audit?	4
How to start a web audit	5
OWASP	6
Domain selection.....	8
Information Gathering	13
Information of Domain name.....	13
Sublist3r	15
Crt.sh.....	17
Nmap scan.....	19
Vulnerability scan	22
Nikto Scan.....	23
Netsparker	27
Vulnerability analysis	32
Conclusion	40
References :.....	41

Abstract

A web security audit is a popular topic these days in the world. These days, web applications, online platforms and also work through the internet are rapidly increasing. Most of the people who are in the IT industry engage with these activities. Because nowadays, people are considering their security and privacy. Therefore, most of the web applications are consider their security.

According to the web security module assigned to students who are studying this module, to perform a web security audit as a module assignment. Then create documentation for it. In this report describe how I perform the web security audit based on [yahoo.com](http://www.yahoo.com).

At the beginning of the report, give a brief introduction to the web security audit. Then discussed how to perform a web security audit and what resources have to get information for the audit. After that part, describe the domain that I selected to perform the web security audit. Then consider the information gathering phase in the audit. in this part discussed what tools are have for information gathering and facts about the domain that I selected. The next part of the report about a web audit considers about vulnerability scan which is a special phase in the audit. In that part discussed what tools can use to do a vulnerability scan and the result of a vulnerability scan of my audit. The latter part of the report contains a vulnerability analysis of I selected domain. Finally, the conclusion of the web security audit is including.

Introduction

Nowadays, the uses of the web application are rapidly increasing. Most of the people in this era doing their day to day life works and also their jobs are doing through these web application.

While rapidly increasing these web applications at the same time web application security also take a major part in the world. Because users' privacy can be accessed through this web application, which can lead to anonymous illegal activity. So web application security is a very important topic as well as a issue that needs to be addressed around the world. If we not consider about this web security, hackers can access these web applications for some malicious activities and also they can do any illegal things in the world using web applications. We have to do various things to avoid these types of activities. Web auditing is one of the most important and most needed solutions for this matter.

What is a Web Audit?

A website security audit is an audit that looks at the infrastructure, core, extensions, and themes for vulnerabilities and shortcomings of the selected web application. In general, through this web security audit review the codes that include in the web application. So through this web audit can identify vulnerabilities in the selected web site. While a security audit purpose is to evaluate and pinpoint the vulnerable area. If it is found that there has any vulnerability in the website through this audit, the pentester can emulate hackers and attack situations and then exploit this vulnerability which is found through the audit. So it helps to find solutions and fix these vulnerabilities. This is the main advantage of doing a web security audit. Web security has several types,

- Automated security audit
- Manual security audit
- Professional security audit

Also, vulnerability scanners are the most used tools to test web security. [1]

How to start a web audit ...

First, we want to select the domain to launch this web security audit. Here we can't access all domains in the world to doing this web security because they do not give any permission to scan their website. Only some of the domains are permitted to do this security audit. So first we want to select one of the domains from these permitted domains.

We have some platform to do these bug bounty programs. Through these platforms, we can select one of the domain and start our web security auditing. Listed below some of the platforms which we can use to select the domain.

- Hackerone
- Bug crowd
- Google

Also, some web security related organizations like,

- OWSAP
- PCI checks
- SANS

are given to the checklist to do a proper web security audit. When we are following one of these guidelines and we can do a proper web application security audit. According to the OWSAP given 10 main steps to do an audit. If we follow these guidelines, we can do a good web security audit.

OWASP

OWASP is the Open Web Application Security Project which found on 2001. It's a well-known community among information technology for supply articles, tools, technologies, and also methodologies of web application security. It provides a checklist for continue to web security audit properly. [2]

OWASP Checklist

1. Information Gathering
2. Configuration and deploy management testing
3. Identity management testing
4. Authentication testing
5. Authorization testing
6. Session management testing
7. Data validating testing
8. Error handling
9. Cryptography
10. Business logic testing
11. Client-side testing

OWASP Top 10

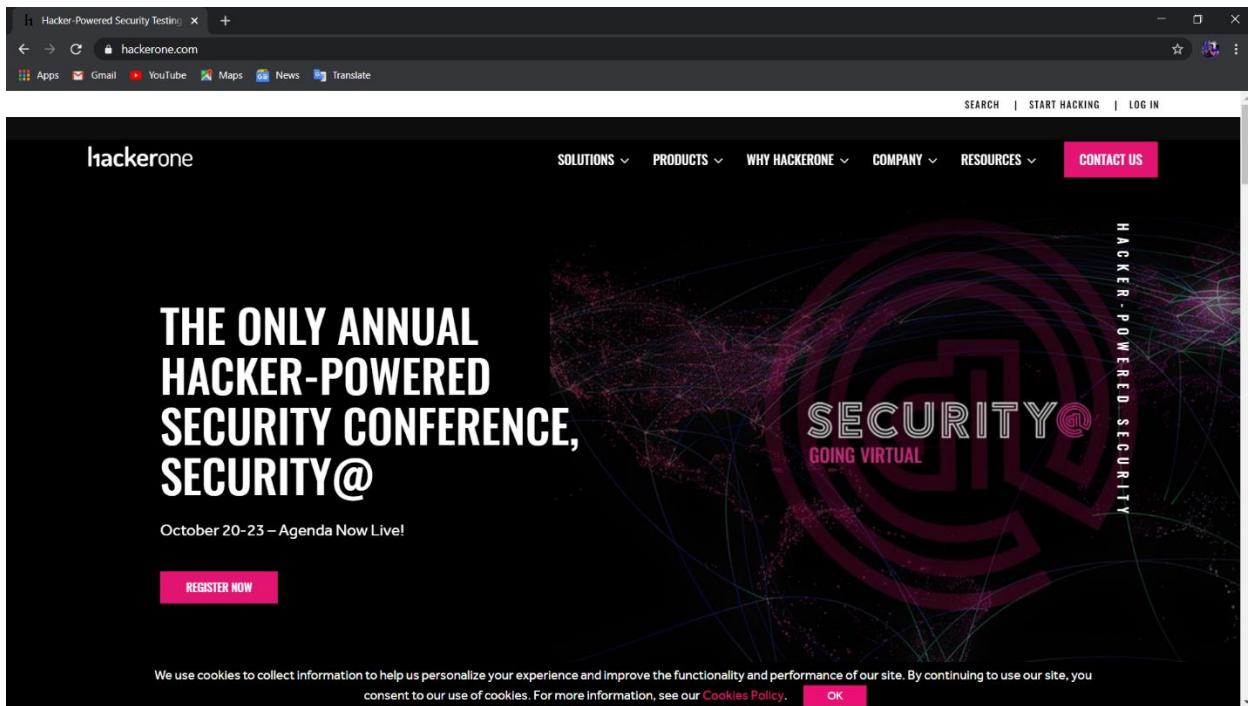
- Injection
- Broken Authentication
- Sensitive Data Exposure
- XML External Entities (XXE)
- Broken Access control
- Security misconfigurations

- Cross-Site Scripting (XSS)
- Insecure Deserialization
- Using Components with known vulnerabilities
- Insufficient logging and monitoring [3]

I decided to follow some of the OWASP testing guidelines and methodology for my web security auditing.

Domain selection

We have so many platforms to select for the domain for this web security audit. After going through some of these platforms and I was selected to HackerOne for selected to a domain for doing my web security audit. Hackerone is good vulnerability coordination and bug bounty platform. It connects with cybersecurity researchers. In addition, HackerOne also paid for these bug bounty programs.



Then I searched permitted domains for doing my bug bounty program. This platform has so many permitted domains for doing security audits. Then I selected some of the domains from this permitted domains list and I did some consideration of the facts about I selected domains. In the Hackerone platform are given some description, some rules for want to consider when doing the bug bounty program and some important facts of the domain. Then I consider the facts and rules of the domains and I selected to <https://www.verizonmedia.com/> domain to doing my web security audit.

Verizon Media - Bug Bounty Program

hackerone.com/verizonmedia?type=team

SOLUTIONS | PRODUCTS | WHY HACKERONE | COMPANY | RESOURCES | CONTACT US

Submit report

Bug Bounty Program
Launched on Feb 2014

Managed by HackerOne
Bounty splitting enabled

Reports resolved: 6600 | Assets in scope: 58 | Average bounty: \$400-\$500

Rewards

Critical	High	Medium	Low
\$10,000	\$3,000	\$500	\$100

Last updated on October 15, 2020. [View changes](#)

Response Efficiency

6 hrs	Average time to first response
3 days	Average time to triage
2 months	Average time to bounty

Verizon Media - Bug Bounty Program

hackerone.com/verizonmedia?type=team

SOLUTIONS | PRODUCTS | WHY HACKERONE | COMPANY | RESOURCES | CONTACT US

Welcome to Verizon Media

Paranoids

We are Paranoid

Our information security team is known as the Paranoids, and we're committed to protecting our brands and our users. As part of this commitment, we invite security researchers to help protect Verizon Media and its users by proactively identifying security vulnerabilities via our bug bounty program. Our program is inclusive of all Verizon Media brands and offers competitive rewards for a wide array of vulnerabilities. We encourage security researchers looking to participate in our bug bounty program to review our policy to ensure compliance with our rules and also to help you safely verify any vulnerabilities you may uncover.

2 months
Average time to bounty

97% of reports
Meet response standards
Based on last 90 days

Program Statistics
Updated Daily

>\$11,090,000	Total bounties paid
\$400 - \$500	Average bounty range
\$5,000 - \$40,000	Top bounty range
\$520,000	Bounties paid in the last 90 days
662	Reports received in the last 90 days

Then I again going through the rules and I understood what can we do in the audit and also what can not be doing in the audit. Also in the platform is given more details about what are the

responsibilities, Rewards, rules of engagement when we are doing adult for this domain. In addition, supply what is the valued vulnerabilities in this domain.

The screenshot shows a web browser window with the title "Verizon Media - Bug Bounty Pro". The URL is "hackerone.com/verizonmedia?type=team". The page displays a table of vulnerabilities accepted by Verizon. The columns are: Severity (low), Severity (high), CWE-ID, Common Weakness Enumeration, and Bug Examples. The rows list various vulnerabilities across different severity levels and CWE categories.

Severity (low)	Severity (high)	CWE-ID	Common Weakness Enumeration	Bug Examples
Critical	Critical	CWE-78	OS Command Injection	Remote Code Execution; Code Injection; LDAP Injection
Critical	Critical	CWE-120	Classic Buffer Overflow	Buffer Overflow
High	Critical	CWE-89	SQL Injection	SQL Injection
Medium	Critical	CWE-918	Server-Side Request Forgery	SSRF (unrestricted); Content-Restricted SSRF; Error-based SSRF (true/false); Blind SSRF
High	Critical	CWE-732	Incorrect Permission Assignment for Critical Resource	IDOR; Horizontal Privilege Escalation; Vertical Privilege Escalation
Critical	Critical	CWE-91	XML Injection	XML Injection
Critical	Critical	CWE-611	Improper Restriction of XML External Entity Reference	XXE
High	Critical	CWE-134	Uncontrolled Format String	Insecure Deserialization
High	Critical	CWE-250	Execution with Unnecessary Privileges	Privilege Escalation to System Account
Medium	High	CWE-444	Inconsistent Interpretation of HTTP Requests	HTTP Request Smuggling
Low	Critical	CWE-829	Inclusion of Functionality from Untrusted Control Sphere	Server Side Includes Injection; Local File Inclusion; Directory Traversal
Medium	High	CWE-306	Missing Authentication for Critical Function	Exposed Administrative Interface
Medium	Critical	CWE-862	Missing Authorization	Horizontal Privilege Escalation; Vertical Privilege Escalation; IDOR

	Low	Critical	CWE-200 Information Exposure	User Enumeration with PII; Credentials on GitHub; Confidential Information Exposure
	Informative	High	CWE-863 Incorrect Authorization	Authorization Bypass; Account Takeover; Social Media Takeover (Brand, <12mo); Social Media Takeover (Personal); Social Media Takeover (Brand, >12mo)
	Medium	High	CWE-798 Use of Hard-coded Credentials	Hard Coded Credentials
	Medium	High	CWE-434 Unrestricted Upload of File with Dangerous Type	Unfiltered File Upload
	Low	High	CWE-203 Information Exposure Through Discrepancy	PHP Admin Information page; MySQL Information page (w/ credentials); Apache Status page
	Medium	Medium	CWE-494 Download of Code Without Integrity Check	S3 Bucket Upload
	Low	Medium	CWE-311 Missing Encryption of Sensitive Data	Cleartext Submission of Passwords
	Low	Medium	CWE-807 Reliance on Untrusted Input in a Sensitive Context	
	Low	Medium	CWE-79 Cross-Site Scripting	Stored XSS; POST-Based XSS; GET-Based XSS; DOM-Based XSS; Flash-based XSS; CSS Injection
	Medium	Medium	CWE-352 Cross-Site Request Forgery	State-Changing CSRF; Non-State-Changing CSRF
	Low	Medium	CWE-16 Misconfiguration	Subdomain Takeover; Dangling DNS Record
	Medium	Medium	CWE-93 CRLF Injection	CRLF Injection
	Low	Low	CWE-601 Open Redirect	Open Redirect
	Informative	Low	CWE-327 Use of a Broken or Risky Cryptographic Algorithm	Weak CAPTCHA
	Informative	Low	CWE-307 Improper Restriction of Excessive Authentication Attempts	Lack of Rate Limiting on Login; CAPTCHA Bypass

And also HackerOne is supplied to scope to what subdomains can we audit. This is very important facts because through this scope we can know our limitations. Sometimes when we trying to access the subdomains which are out of scope, we have to chance banned from the domain. Hence this scope is helpful to us.

The screenshot shows the Verizon Media - Bug Bounty Program page on hackerone.com. The main content area is titled "Scopes" and lists "In Scope" domains:

Domain	Critical	Eligible
data.mail.yahoo.com	Critical	Eligible
le.yahooapis.com	Critical	Eligible
onepush.query.yahoo.com	Critical	Eligible
proddata.xobni.yahoo.com	Critical	Eligible
apis.mail.yahoo.com	Critical	Eligible

Below this, there is a section titled "Moloch" with the sub-section "Review the Code". It contains the following steps:

- Source Code [↗](#)
- Submit a PR to fix/update the code - [fork ↗](#) the codebase then submit a [PR ↗](#)
- Visit our web page at <https://molo.ch> ↗ for pre-built rpm/deb and instructions for running yourself.

At the bottom of the table, it says "Source code" again with "Critical" and "Eligible" status indicators.

After finding and referring to this information and I started to information gathering part according to the OWASP checklist.

Information Gathering

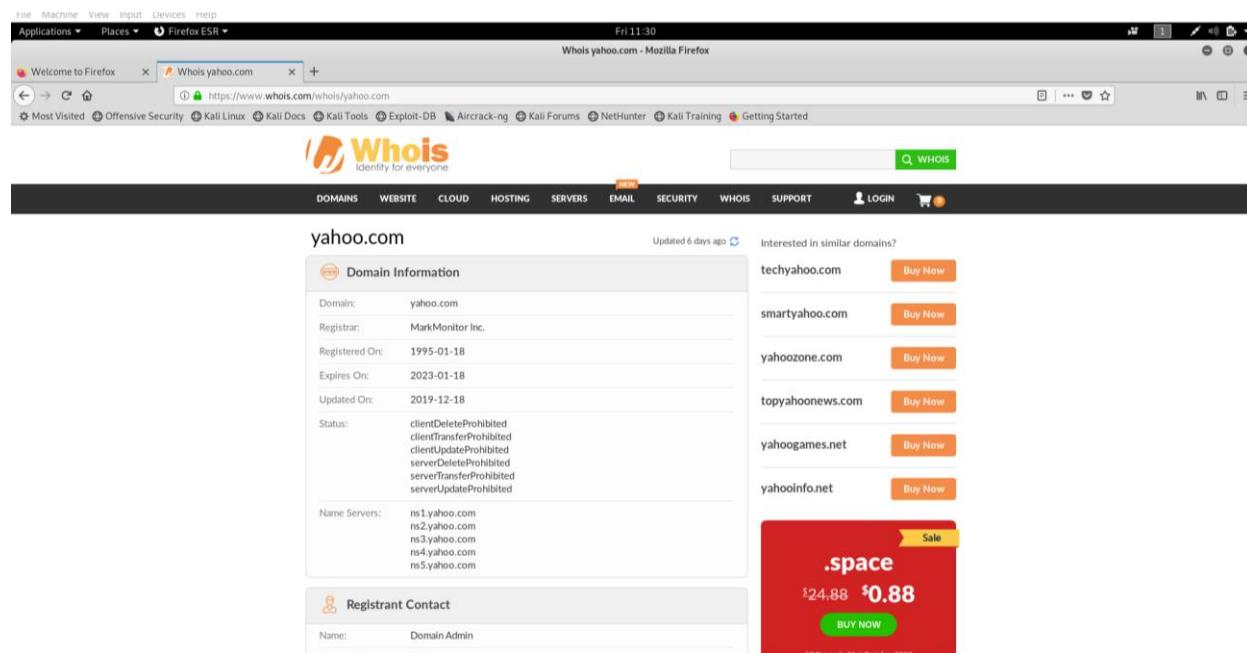
Most of the information about I selected domain is given in the platform. By that, we can get a clear idea about the domain. But when we are doing a web security audit, we want more details. Hence we have to do the information gathering part.

I used some tools for done information gathering.

- ✓ Sublist3r
- ✓ Crt.sh
- ✓ Nmap

Information of Domain name

I used the “Whois” feature to get more information about I selected domain. It includes server names, status, registered date, update date, and more details.



The screenshot shows a Firefox browser window with the title "Whois yahoo.com - Mozilla Firefox". The address bar displays "Whois yahoo.com". The main content area shows the Whois information for the domain "yahoo.com". The "Domain Information" section includes fields for Domain, Registrar, Registered On, Expires On, Updated On, Status, and Name Servers. The "Status" field lists several prohibited actions. The "Registrant Contact" section shows the Name as "Domain Admin". To the right of the main content, there is a sidebar with a list of similar domains like "techyahoo.com", "smartyahoo.com", etc., each with a "Buy Now" button. Below this is a promotional banner for ".space" domains, showing a price of \$0.88.

Domain Information	
Domain:	yahoo.com
Registrar:	MarkMonitor Inc.
Registered On:	1995-01-18
Expires On:	2023-01-18
Updated On:	2019-12-18
Status:	clientDeleteProhibited clientTransferProhibited clientUpdateProhibited serverDeleteProhibited serverTransferProhibited serverUpdateProhibited
Name Servers:	ns1.yahoo.com ns2.yahoo.com ns3.yahoo.com ns4.yahoo.com ns5.yahoo.com

Registrant Contact	
Name:	Domain Admin

kali [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Applications ▾ Places ▾ Firefox ESR ▾

Fri 11:30 Whois yahoo.com - Mozilla Firefox

Welcome to Firefox Whois yahoo.com +

https://www.whois.com/whois/yahoo.com

Most Visited Offensive Security Kali Linux Kali Docs Kali Tools Exploit-DB Aircrack-ng Kali Forums NetHunter Kali Training Getting Started

Email: domain-admin@oath.com

Raw Whois Data

Domain Name: yahoo.com
 Registry Domain ID: 3643624 DOMAIN.COM-VHSN
 Registrar WHOIS Server: whois.markmonitor.com
 Registrar URL: http://www.markmonitor.com
 Updated Date: 2020-07-30T11:35:29-0700
 Creation Date: 1996-09-01T00:00:00-0700
 Registrar Registration Expiration Date: 2023-01-18T21:00:00-0800
 Registrar: MarkMonitor, Inc.
 Registrar IAB ID: 292
 Registrant House Number: Email: abusecomplaints@markmonitor.com
 Registrant Abuse Contact Phone: +1.2083095770
 Domain Status: clientUpdateProhibited (https://www.icann.org/epp#clientUpdatePrc)
 Domain Status: clientTransferProhibited (https://www.icann.org/epp#clientTransferPrc)
 Domain Status: clientDeleteProhibited (https://www.icann.org/epp#clientDeletePrc)
 Domain Status: serverUpdateProhibited (https://www.icann.org/epp#serverUpdatePrc)
 Domain Status: serverTransferProhibited (https://www.icann.org/epp#serverTransferPrc)
 Domain Status: serverDeleteProhibited (https://www.icann.org/epp#serverDeletePrc)
 Registry Registrant ID:
 Registrant Name: Domain Admin
 Registrant Organization: Oath Inc.
 Registrant Street: 22000 AOL Way
 Registrant City: Dulles
 Registrant State/Province: VA
 Registrant Postal Code: 20166
 Registrant Country: US
 Registrant Phone: +1.4083493300
 Registrant Phone Ext:
 Registrant Fax:
 Registrant Fax Ext:
 Registrant Email: domain-admin@oath.com
 Registry Admin ID:
 Admin Name: Domain Admin
 Admin Organization: Oath Inc.
 Admin Street: 22000 AOL Way
 Admin City: Dulles
 Admin State/Province: VA
 Admin Postal Code: 20166
 Admin Country: US
 Admin Phone: +1.4083493300
 Admin Phone Ext:
 Admin Fax:
 Admin Fax Ext:

Admin Email: domain-admin@oath.com
 Registry Tech ID:
 Tech Name: Domain Admin
 Tech Organization: Oath Inc.
 Tech Street: 22000 AOL Way
 Tech City: Dulles
 Tech State/Province: VA
 Tech Postal Code: 20166
 Tech Country: US
 Tech Phone: +1.4083493300
 Tech Phone Ext:
 Tech Fax:
 Tech Fax Ext:
 Tech Email: domain-admin@oath.com
 Name Server: ns4.yahoo.com
 Name Server: ns1.yahoo.com
 Name Server: ns2.yahoo.com
 Name Server: ns3.yahoo.com
 Name Server: ns3.yahoo.com
 DNSSEC: unsigned
 URL of the ICANN WHOIS Data Problem Reporting System: http://wdprs.internic.net/
 >>> Last update of WHOIS database: 2020-10-09T04:58:35-0700 <<<

For more information on WHOIS status codes, please visit:
<https://www.icann.org/resources/pages/epp-status-codes>

If you wish to contact this domain's Registrant, Administrative, or Technical contact, and such email address is not visible above, you may do so via our web form, pursuant to ICANN's Temporary Specification. To verify that you are not a robot, please enter your email address to receive a link to a page that facilitates email communication with the relevant contact(s).

Web-based WHOIS:
<https://domains.markmonitor.com/whois>

If you have a legitimate interest in viewing the non-public WHOIS details, send your request and the reasons for your request to whoserequest@markmonitor.com and specify the domain name in the subject line. We will review that request and ask for supporting documentation and explanation.

The data in MarkMonitor's WHOIS database is provided for information purposes, and to assist persons in obtaining information about or related to a domain's registration record. While MarkMonitor believes the data to be accurate, the data is provided "as is" with no guarantee or warranties regarding its accuracy.

By submitting a WHOIS query, you agree that you will use this data only for lawful purposes and that, under no circumstances will you use this data to:
 (1) allow, enable, or otherwise support the transmission by email, telephone, or facsimile of mass, unsolicited, commercial advertising, or spam; or
 (2) enable the collection, distribution, or use of personally identifying information about individuals from this data without their consent, except as required by law, or as needed to contact you about your interaction with the data or to contact MarkMonitor (or its systems) or the domain name contacts (or its systems).

MarkMonitor reserves the right to modify these terms at any time.

By submitting this query, you agree to abide by this policy.

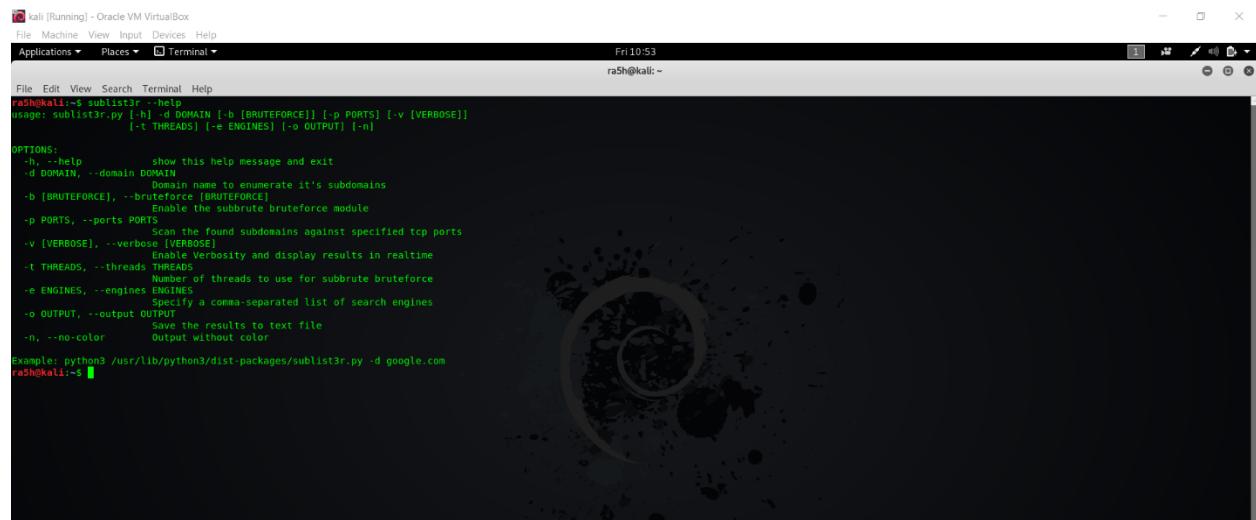
MarkMonitor Domain Management(TM)
 Protecting companies and consumers in a digital world.

Visit MarkMonitor at <http://www.markmonitor.com>
 Contact us at +1.8007459229
 In Europe, at +44.02032062220
 ...

Sublist3r

Sublist3r is a tool for gather subdomains in the domains. When using this tool I can found subdomains in the domain which I selected. Sublist3r also enumerates subdomains using Netcraft, Virustotal, ThreatCrowd, DNSdumpster, and ReverseDNS. [4]

I am also used this sublist3r for my information gathering part. Then I able to find the subdomains of yahoo.com that I selected domain.



The screenshot shows a terminal window on a Kali Linux desktop environment. The title bar reads "kali [Running] - Oracle VM VirtualBox". The terminal window has a dark background with white text. It displays the usage information for the sublist3r tool:

```
ra5h@kali:~$ sublist3r --help
usage: sublist3r.py [-h] [-D DOMAIN] [-b [BRUTEFORCE]] [-p PORTS] [-v [VERBOSE]]
                   [-t THREADS] [-e ENGINES] [-o OUTPUT] [-n]
OPTIONS:
-h, --help      show this help message and exit
-D DOMAIN, --domain DOMAIN
               domain name to enumerate it's subdomains
-b [BRUTEFORCE], --bruteforce [BRUTEFORCE]
               Enable the subbrute bruteforce module
-p PORTS, --ports PORTS
               scan the found subdomains against specified tcp ports
-v [VERBOSE], --verbose [VERBOSE]
               increase verbosity and display results in realtime
-t THREADS, --threads THREADS
               Number of threads to use for subbrute bruteforce
-e ENGINES, --engines ENGINES
               specify a comma-separated list of search engines
-o OUTPUT, --output OUTPUT
               write the results to text file
-n, --no-color   output without color
Example: python3 /usr/lib/python3/dist-packages/sublist3r.py -d google.com
ra5h@kali:~$
```

```

kali [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Applications ▾ Places ▾ Terminal ▾
File Edit View Search Terminal Help
rash0x@kali:~$ 
rash0x@kali:~$ 
rash0x@kali:~$ 
rash0x@kali:~$ sublist3r -d yahoo.com
# Sublist3r v3.0.0
# Coded By Ahmed Aboul-Ela - @aboulla
[+] Enumerating subdomains now for yahoo.com
[+] Searching now in Baidu..
[+] Searching now in Yahoo..
[+] Searching now in Google..
[+] Searching now in Bing..
[+] Searching now in Ask..
[+] Searching now in DuckDuckGo..
[+] Searching now in DNSdumpster..
[+] Searching now in VirusTotal..
[+] Searching now in ThreatCrowd..
[+] Searching now in SSL Certificates..
[+] Searching now in PassiveDNS..
[+] Total Unique Subdomains Found: 114637
ur.yahoo.com<br>ai.yah0.com<br>br.yahoo.com<br>ca.yahoo.com<br>de.yahoo.com<br>es.yahoo.com<br>fr.yahoo.com<br>hk.yahoo.com<br>id.yahoo.com<br>ie.yahoo.com<br>in.yahoo.com<br>it.yahoo.com<br>kr.yahoo.com<br>mx.yahoo.com<br>ph.yahoo.com<br>qc.yahoo.com<br>sg.yahoo.com<br>tw.yahoo.com<br>uk.yahoo.com<br>us.yahoo.com<br>vn.yahoo.com<br>www.yahoo.com<br>yahoo.com
uk.118800.yahoo.com
360.yahoo.com
hu.360.yahoo.com
bl0g.360.yahoo.com
su.blog.360.yahoo.com
ca.blog.360.yahoo.com
de.blog.360.yahoo.com
fr.blog.360.yahoo.com
it.blog.360.yahoo.com
blogs.360.yahoo.com
ra.360.yahoo.com
de.360.yahoo.com
download.360.yahoo.com
fr.360.yahoo.com
img.360.yahoo.com
message.360.yahoo.com
uk.360.yahoo.com
bssauth.3721.yahoo.com
img.3721.yahoo.com
hk.7-eleven.yahoo.com

hk.7-eleven.yahoo.com
Erdavisi01@yahoo.com
ha2.vl-123.bas1-edg.a4e.yahoo.com
ha2.vl-123.bas1-edg.a4e.yahoo.com
ha2.vl-123.bas1-edg.a4e.yahoo.com
ha2.vl-124.bas1-edg.a4e.yahoo.com
te-01.bas1-edg.a4e.yahoo.com
te-84.bas1-edg.a4e.yahoo.com
te-91.bas1-edg.a4e.yahoo.com
vl-107.bas1-edg.a4e.yahoo.com
vl-122.bas1-edg.a4e.yahoo.com
vl-123.bas1-edg.a4e.yahoo.com
vl-124.bas1-edg.a4e.yahoo.com
vl-124.bas1-edg.a4e.yahoo.com
te-94.bas2-1-edg.a4e.yahoo.com
vl-100.bas2-1-edg.a4e.yahoo.com
vl-122.bas2-1-edg.a4e.yahoo.com
vl-123.bas2-1-edg.a4e.yahoo.com
vl-124.bas2-1-edg.a4e.yahoo.com
lbx1-1-edg.a4e.yahoo.com
lbx2-1-edg.a4e.yahoo.com
config.yodvip.a4e.yahoo.com
111.f51.ymdb.a4e.yahoo.com
110.f51.ymdb.a4e.yahoo.com

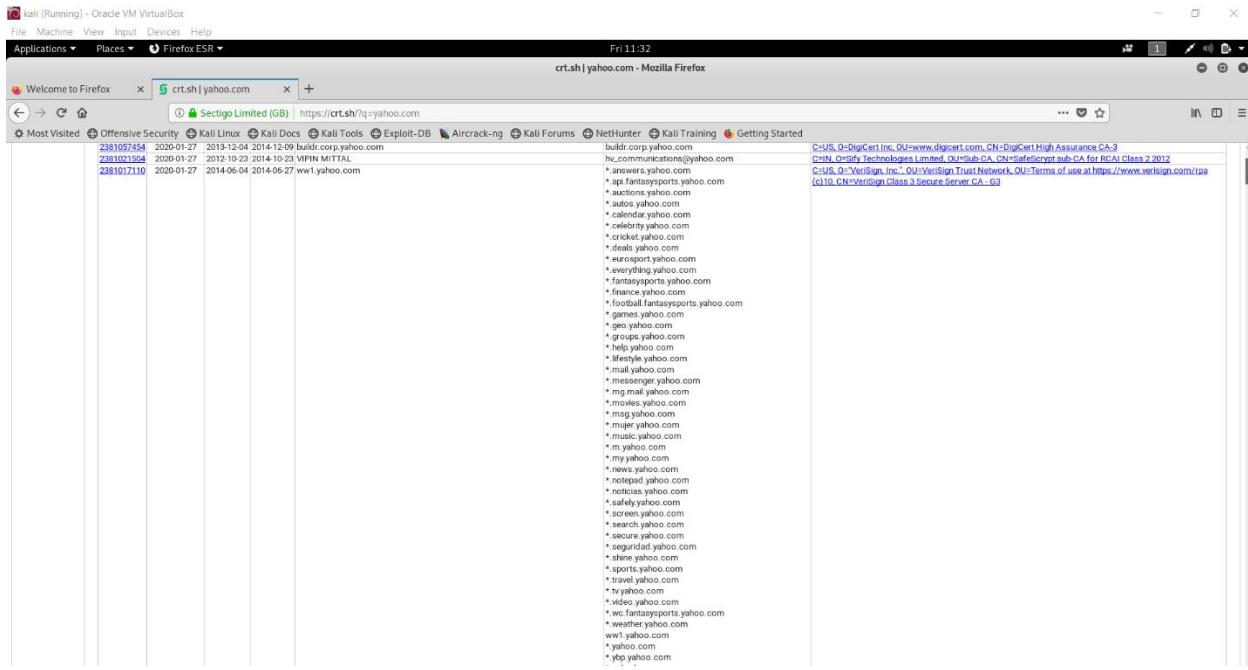
```

I used this sublist3r tool to find 114637 subdomains on yahoo.com, the domain I selected.

Crt.sh

Furthermore, I used “**Crt.sh**” to find subdomains for doing my web security auditing. **Crt.sh** is also like a sublist3r tool that also can find subdomains of the selected domain.

A screenshot of a Firefox browser window titled "crt.sh | yahoo.com - Mozilla Firefox". The address bar shows "crt.sh | yahoo.com". The main content area is titled "crt.sh Identity Search" and contains a search bar with "Match: ILIKE Search: yahoo.com". Below the search bar, there's a "Criteria" dropdown set to "Type: Identity Match: ILIKE". The main table lists numerous certificates, each with columns for "Certificates", "ext_id", "Issued At", "Not Before", "Not After", "Common Name", "Matching Identities", and "Issuer Name". The "Certificates" column lists IDs such as 238295144, 238295482, 2382973482, 238297375, 2382722910, 2382722499, 238227680, 2381997081, 238199033, 238180081, 238169485, 238169663, 238156107, 2381531941, 2381536499, 2381518150, 2381506044, 2381439781, 2381426611, 2381395266, 2381393804, 2381393196, and 2381392872. The "Common Name" column shows various email addresses and domain names like "brandenligu@yahoo.com", "timothydccherry@yahoo.com", "smtp.mail.yahoo.com", "nairvkj@yahoo.com", "nairvkj@yahoo.com", "lacheny@yahoo.com", "lohet@yahoo.com", "chandranasasthi@yahoo.com", "d_trivedi@yahoo.com", "kishoreengineering@yahoo.com", "R_MMSP817@YAHOO.COM", "greddy_associates@yahoo.com", "rajeshtchoragle@yahoo.com", "dsurja@yahoo.com", "tvb_satur@yahoo.com", "CERVECERIAONFELIPE@YAHOO.COM", "CAYOGESHVVAS@YAHOO.COM", "GRYM2000@YAHOO.COM", "ca021979@yahoo.com", "salm.bharai@yahoo.com", "greddy_associates@yahoo.com", "greddy_associates@yahoo.com", and "TCB29@YAHOO.COM". The "Issuer Name" column provides detailed information about the certificate issuers, including company names, addresses, and unique identifiers.



After these several stages, I selected some subdomains because this domain has so many subdomains and we can not do to scan all subdomains in that domain. Then I do vulnerability scans for I selected subdomains in the domain.

Then I shifted to another stage of the information-gathering part. Before doing a vulnerability scan I did a Nmap scan.

Several subdomains I selected,

- ✓ Mail.yahoo.com
- ✓ Onepush.query.yahoo.com
- ✓ Proddata.xobni.yahoo.com
- ✓ Data.mail.yahoo.com
- ✓ Le.yahoo.apis.com
- ✓ Apis.yahoo.mail.com

Nmap scan

Nmap is short for network mapper. This Nmap is open source tools for network discovery and also vulnerability scanner. And also it is known as a network monitoring system. When using this Nmap scanning, we can identify,

- Open ports
- What devices are running on the system
- Available host

Nmap collects information by sending raw packet systems to the port. Listening to the feedback on those sent raw packets determines whether the port is open or closed. Nmap used transport layer protocol include UDP, TCP and also SCTP. The packets which Nmap sends return with IP addresses of ports and some of the other data in the ports and also system. through this Nmap scan we can able to get an idea of the network in the system. and also Nmap can identify the operating systems running on the network systems. It is known as OS fingerprinting. [5]

Nmap commands :

```
raSh@kali:~$ nmap --help
nmap [v0.9.1] http://nmap.org
Usage: nmap [Scan Type(s)] [Options] {target specification}
TARGET SPECIFICATION:
  Can pass hostnames, IP addresses, networks, etc.
  Ex: scanme.nmap.org, microsoft.com/24, 192.168.0.1; 10.0.0-255.1-254
  -l <inputfilenames>: Input from list of hosts/networks
  -iL <inputfile>: Input from list of targets
  --exclude <host1[,host2...]>: Exclude hosts/networks
  --excludefile <exclude_file>: Exclude list from file
HOST DISCOVERY:
  -SL List Scan - simply List targets to scan
  -SN Ping Scan - disable port scan
  -PN Ping Only - skip host discovery
  -PE/PN/PNU/PNP/PortList: TCP SYN/ACK, UDP or SCTP discovery given ports
  -PO[protocol]list: IP Protocol Ping
  -n/-R: Never do DNS resolution/Always resolve [default: sometimes]
  -dns-servers <serv1,serv2...>: Specify custom DNS servers
  -sS: Scan like OS's DNS resolver
  -traceroute: Trace hop path to each host
SCAN TECHNIQUES:
  -SS/t/sA/SU/M: TCP SYN/Connect()|ACK/Window/Maimon scans
  -SU UDP Scan
  -SN/SP/SA/TCP null, FIN, and Xmas scans
  -SF/SV/AS: TCP scan flags
  -SF/AS: Scan flags: Custom TCP scan flags
  -sI <zombie host/probeuri>; Idle scan
  -SY/z: SCTP INIT/COOKIE-ECHO scans
  -SO: IP protocol scan
  -b <FTP relay host>; FTP bounce scan
PORT SPECIFICATION AND ORDER:
  -p <port>: Target port only (e.g. 80)
  --exclude-ports <port ranges>: Exclude the specified ports
  -E: -p22,-p165535: -p U:55,111,137,T:21-25,80,139,8080,S:9
  --exclude-ports <port ranges>: Exclude the specified ports from scanning
  -F: Fast mode - Scan fewer ports than the default scan
  -R: Scan ports consecutively - don't randomize
  --top-ports <numbers>: Scan <numbers> most common ports
  -P<hosts>: Target hosts can be comma separated
  --script-args <script>: Scan ports more common than <ratio>
SERVICE/VERSION DETECTION:
  -SV: Probe open ports to determine service/version info
  --version-intensity <level>: Set from 0 (light) to 9 (try all probes)
  --version-light: Limit to most likely probes (intensity 2)
  --version-all: Try every single probe (intensity 9)
  -V: --script-trace: Show detailed version scan activity (for debugging)
SCRIPT SCAN:
  -SC: equivalent to --script=default
  --scripts=<Lua scripts>; <Lua scripts> is a comma separated list of
    directories, script-files or script-categories
  --script-args=<args>; <n>=<v>...>; provide arguments to scripts

--script-args=<filename>; provide NSE script args in a file
--script-trace: Show all data sent and received
--script-updatedb: Update the script database
--script-help=<Lua scripts>; Show help about scripts.
  <Lua scripts> is a comma-separated list of script-files or
    script-categories.

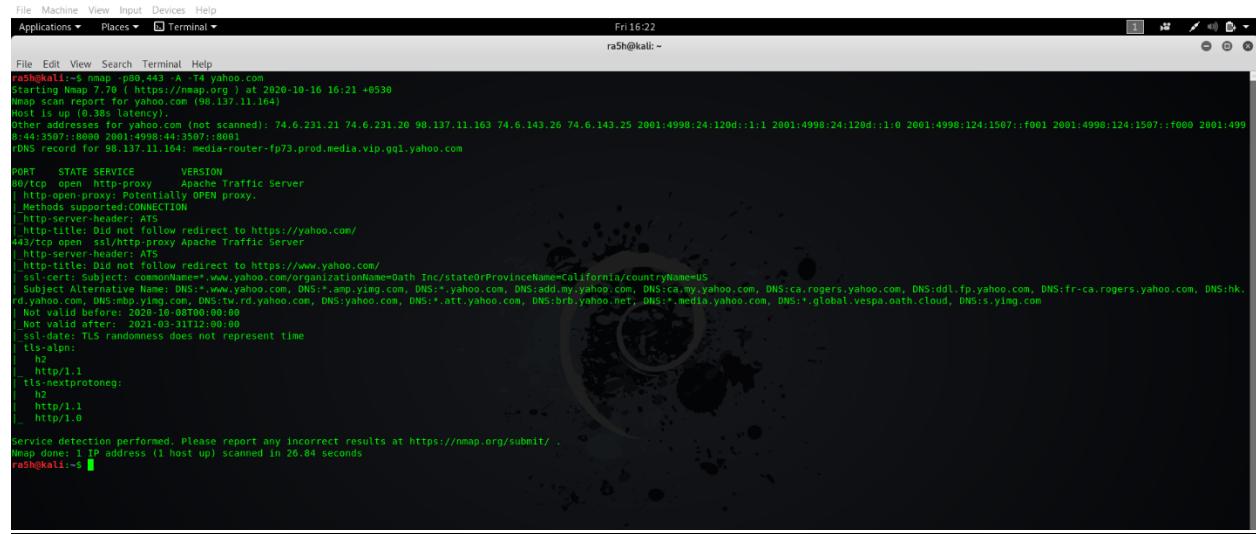
OS DETECTION:
  -O: Enable OS detection
  --osscan-limit: Limit OS detection to promising targets
  --osscan-guess: Guess OS more aggressively
TIMING AND PERFORMANCE:
  -T<0-5>: Set timing template (higher is faster)
  --min-hostgroup/max-hostgroup <size>: Parallel host scan group sizes
  --min-parallelism/max-parallelism <n>: Probe parallelism
  --min-rtt-timeout/min-rtt-timeout/initial-rtt-timeout <time>: Specifies
    minimum round trip time
  --max-retries <tries>: Coax number of port scan probe retransmissions.
  --host-timeout <time>: Give up on target after this long
  --scan-delay/-max-scan-delay <time>: Adjust delay between probes
  --min-rate <numbers>: Send packets no slower than <numbers> per second
  --max-rate <numbers>: Send packets no faster than <numbers> per second
FIREWALL/IDS EVASION AND SPOOFING:
  -f: -mtu <val>; fragment packets (optionally w/given MTU)
  -D <decoy1,decoy2[,ME]...>; Cloak a scan with decoys
  -S <IP Address>; Spoof source address
  -e <Interface>; Use specified interface
  -g <source-port> <numbers>: Set a given port number
  -p <proxy-list[,url3]...>; Relay connections through HTTP/SOCKS4 proxies
  --data <hex string>; Append a custom payload to sent packets
  --data-string <string>; Append a custom ASCII string to sent packets
  --data-length <nme>; Append random data to sent packets
  --ip-options <options>; Send packets with specified IP options
  -T <value>; Set TCP timeout
  --spoof-source <address/prefix>/<vendor name>; Spoof your MAC address
  --badsum: Send packets with a bogus TCP/UDP/SCTP checksum

OUTPUT:
  -O<XML/JSON/YAML> <file>; Output scan in normal, XML, JSON or YAML
  -oX <file>; And Grepable format, respectively, to the given filename.
  -oG <file>; And Grepable XML format, respectively, to the given filename.
  -vv Increase verbosity level (use -vv or more for greater effect)
  -d: Increase debugging level (use -dd or more for greater effect)
  --reason: Display the reason a port is in a particular state
  --open: Only show open (or possibly open) ports
  --packet-trace: Show all the packets sent and received
  --list-trace: List the trace files (for debugging)
  --append-output: Append to rather than clear or specified output files
  --resume <filename>; Resume an aborted scan

--packet-trace: Show all packets sent and received
--ifflist: Print host interfaces and routes (for debugging)
--append-output: Append to rather than clobber specified output files
--resume <filename>; Resume an aborted scan
--stylesheet <path/URL>; XSL stylesheet to transform XML output to HTML
--webxml: Reference stylesheet from Nmap.org for more portable XML
--no-stylesheet: Prevent associating of XSL stylesheets w/XML output
MISC:
  -6: Enable IPv6 scanning
  -A: Enable OS detection, version detection, script scanning, and traceroute
  --datadir <dirname>; Specify custom Nmap data file location
  --send-eth/-send-ip: Send using raw ethernet frames or IP packets
  -S <source IP>; Set source IP unless the user is fully privileged
  --unprivileged: Assume the user lacks raw socket privileges
  -V: Print version number
  -h: Print this help summary page.

EXAMPLES:
nmap -v -A scanme.nmap.org
nmap -v -sT -p 192.168.0.9/16 10.0.0.0/8
nmap -v -sR 10000 -Pn -p 80
SEE THE MAN PAGE (https://nmap.org/book/man.html) FOR MORE OPTIONS AND EXAMPLES
```

Nmap scan of yahoo.com



```
File Machine View Input Devices Help
Applications Places Terminal Fri 16:22
raSh@kali:~

File Edit View Search Terminal Help
raSh@kali:~$ nmap -A -T4 98.137.11.164
Starting Nmap 7.70 ( https://nmap.org ) at 2020-10-16 16:23 +0530
Nmap scan report for yahoo.com (98.137.11.164)
Host is up (0.38s latency).
Other addresses for yahoo.com (not scanned): 74.6.231.21 74.6.231.20 98.137.11.163 74.6.143.26 74.6.143.25 2001:4998:24:120d::1:1 2001:4998:24:120d::1:0 2001:4998:124:1507::f001 2001:4998:124:1507::f000 2001:4998:124:1507::f001 2001:4998:124:1507::f000
DNS record for 98.137.11.164: media-router-fp73.prod.media.vip.gq1.yahoo.com

PORT      STATE SERVICE      VERSION
80/tcp    open  http-proxy   Apache Traffic Server
| http-open-proxy: Potentially OPEN proxy
|_ Methods supported:CONNECT
|_ http-title: Did not follow redirect to https://yahoo.com/
443/tcp   open  ssl/http proxy Apache Traffic Server
| http-server-header: ATS
| http-title: Did not follow redirect to https://www.yahoo.com/
| ssl-cert: Subject: commonName=www.yahoo.com/organizationName=Oath Inc/stateOrProvinceName=California/countryName=US
| SubjectAltName:DNS="*.y1.yahoo.com",DNS="*.mp.yimg.com",DNS="add.my.yahoo.com",DNS="ca.yahoo.com",DNS="ca.rogers.yahoo.com",DNS="dnl.fp.yahoo.com",DNS="fr-ca.rogers.yahoo.com",DNS="hk.yd.yahoo.com",DNS="mp.yimg.com",DNS="rd.yahoo.com",DNS="yahoo.com",DNS="yahadoo.com",DNS="*.alt.yahoo.com",DNS="ibb.yahoo.net",DNS="*.media.yahoo.com",DNS="global.vespa.oath.cloud",DNS="s.yimg.com"
| Not valid before: 2020-10-06T00:00:00
| Not valid after: 2021-03-31T12:00:00
| ssl-date: TLS randomness does not represent time
| tls-alpn:
|_ http/1.1
|_ http/1.1
|_ http/2
|_ http/1.1
|_ http/1.0

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 26.84 seconds
raSh@kali:~$
```

Command of the above screenshot shows, ports 80 and 443 mean HTTP and HTTPS scripts and –A mean an active OS detection, version detection and also script scanning. –T means timing template. And the last one is the target domain that I need to scan.

According to a Nmap scan of the yahoo.com, this domain used server as **ATS(Apache Traffic Server)**.

After finishing this stage, I do a vulnerability scan as another part of my web security auditing.

Vulnerability scan

A vulnerability scan is an automated technology that attempts to identify vulnerabilities in the system. through this scan, we can identify shortcomings in the system. vulnerability scan has various types.

- External vulnerability scan
- Internal vulnerability scan
- Authenticated vulnerability scan
- Unauthenticated vulnerability scan [6]

Also, we have to use many tools for doing this vulnerability scan. Some are,

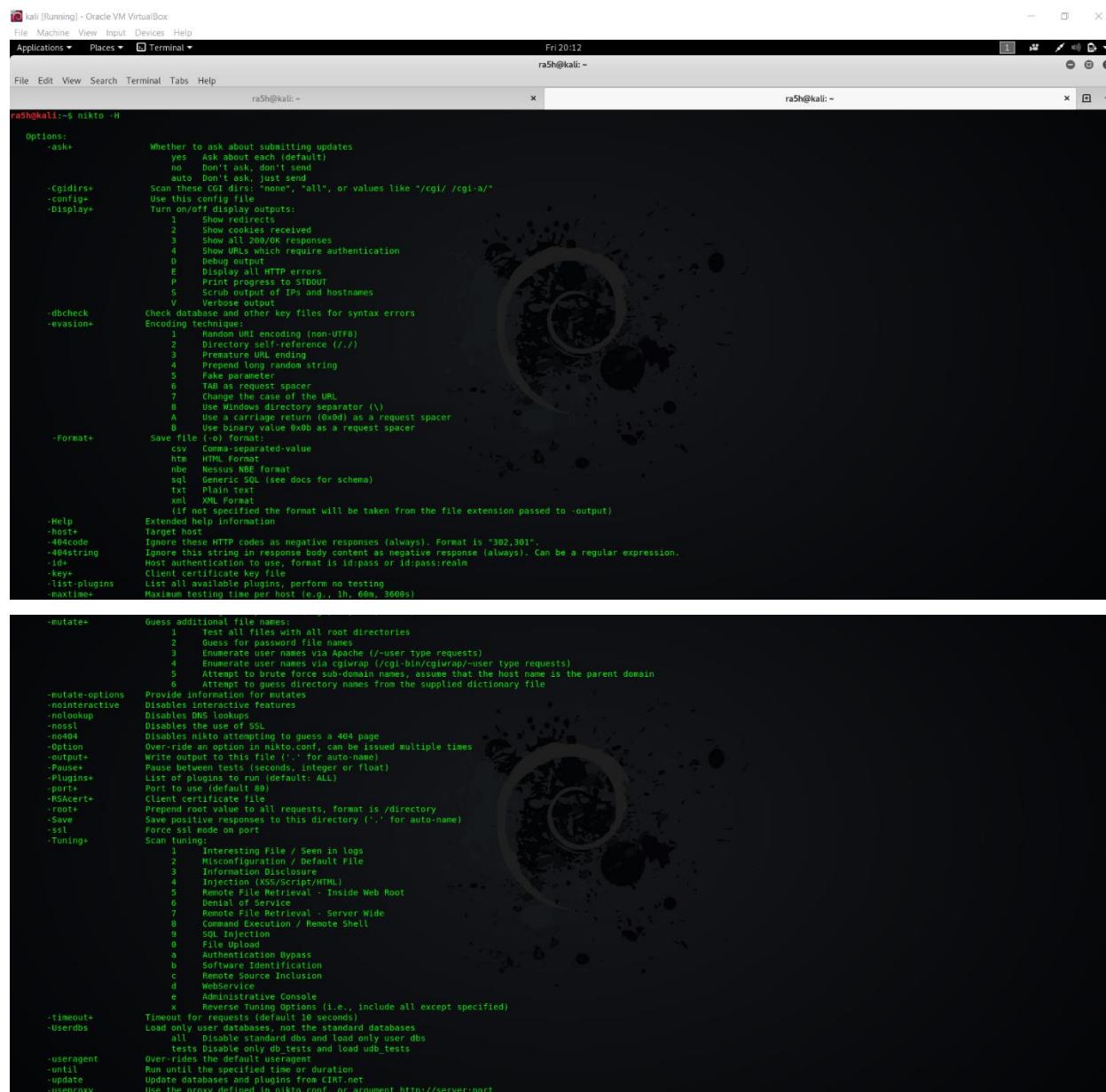
- Nikto scan
- Nessus
- Netsparker
- Burp

I selected **Nikto scan** and **Netsparker** tools from these tools to continue to my web security audit.

Nikto Scan

Nikto scan is a well known vulnerability scanner that scans web servers for vulnerable and dangerous files, outdated server software, miss configuration and other problems. It captures cookies received and then print it. This scanner helps to detect security-related issues in web script and web server configuration. Nikto is open-source software.

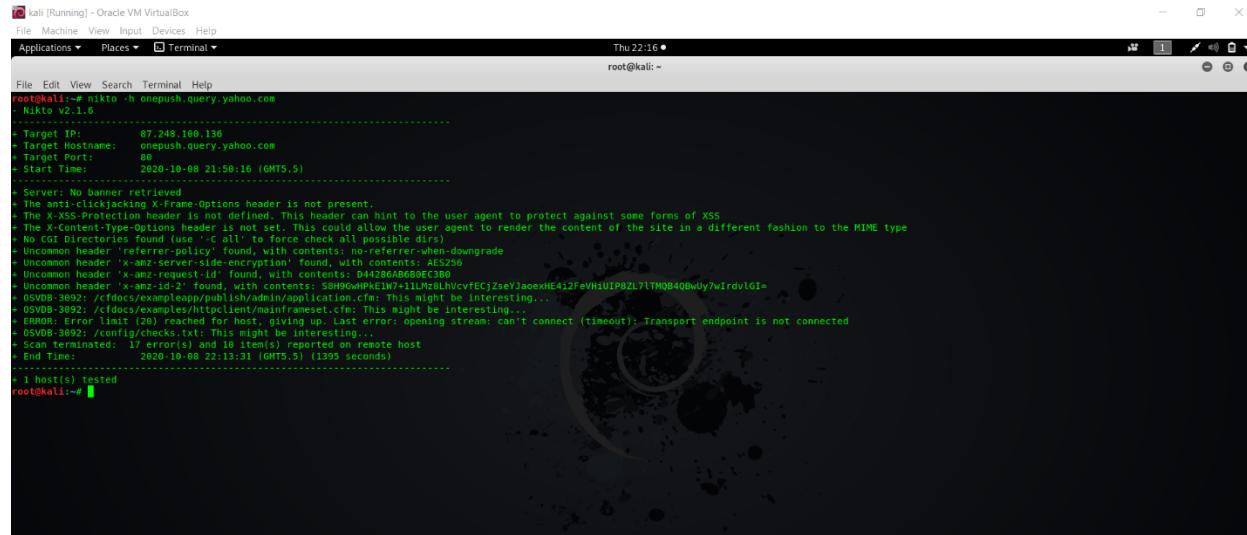
These are some keywords of Nikto :



```
ra5h@kali:~$ nikto -h
Options:
  -ask+      Whether to ask about submitting updates
            yes Ask about each (default)
            no  Don't ask, don't send
            auto Don't ask, just send
  -Cgidirs+  Scan These CGI dirs: "none", "all", or values like "/cgi/ /cgi-a/"
  -config+   Use this config file
  -Display+  Turn on/off display outputs:
            0  Standard red/blue
            1  Show responses received
            2  Show all 200/OK responses
            3  Show URLs which require authentication
            4  Debug output
            D  Display all HTTP errors
            E  Prepend long random string to STDOUT
            F  Print progress to STDOUT
            S  Scrub output of IPs and hostnames
            V  Verbose output
  -dbcheck   Check database and other key files for syntax errors
  -evasion+  Encoding technique:
            1  Random URI encoding (non-UTF8)
            2  Direct URL self-reference (/./)
            3  Prostotype URL encoding
            4  Prepend long random string
            5  Fake parameter
            6  TAB as request spacer
            7  Change the case of the URL
            8  Use multiple directory separator (\)
            A  Use a carriage return (0xd) as a request spacer
            B  Use binary value 0x0b as a request spacer
  -Format+   Save file (-o) format:
            csv  Comma-separated-value
            htm  HTML Format
            nbe  Nessus-like format
            sql  Generic SQL (see docs for schema)
            txt  Plain text
            xml  XML Format
            (if not specified the format will be taken from the file extension passed to -output)
  -Help      Extended help information
  -hosts+   Target hosts
  -404code  Ignore these HTTP codes as negative responses (always). Format is "302,301".
  -404string Ignore this string in response body content as negative response (always). Can be a regular expression.
  -id+      Host authentication to use, format is idpass or idpass:realm
  -key+     Client certificate key file
  -list-plugins List all available plugins, perform no testing
  -maxtime+ Maximum testing time per host (e.g., 1h, 60m, 5600s)

  -mutate+   Guess additional file names:
            1  Test for files with all root directories
            2  Guess for password file names
            3  Enumerate user names via Apache (/user type requests)
            4  Enumerate user names via cgiwrap (/cgi-bin/cgiwrap/user type requests)
            5  Attempt to brute force sub-domain names, assume that the host name is the parent domain
            6  Attempt to guess directory names from the supplied dictionary file
  -mutate-options Provide information for mutates
  -nointeractive Disables interactive features
  -nolockup  Disables DNS lookups
  -nossl    Disables the use of SSL
  -no404   Disables nikto attempting to guess a 404 page
  -Option+   Over-ride an option in nikto.conf, can be issued multiple times
  -outputs+ Write output to this file ('-' for auto-name)
  -Plugins+  Plugin options (multiple, integer or float)
  -Plugins+  List of plugins to run (default: ALL)
  -port+    Port to use (default 80)
  -RSACert+ Client certificate file
  -root+    Prepend root value to all requests, format is /directory
  -Save+    Save positive responses to this directory ('.' for auto-name)
  -ssl+    Force ssl mode on port
  -Tuning+   Scan tuning:
            1  Interesting File / Seen in logs
            2  Misconfiguration / Default File
            3  Information Disclosure
            4  Injection (XSS/Script/HTML)
            5  Remote File Inclusion - Inside Web Root
            6  Denial of Service
            7  Remote File Retrieval - Server Wide
            8  Command Execution / Remote Shell
            9  SQL Injection
            0  File Upload
            a  Authentication Bypass
            b  Software Identification
            c  Remote Source Inclusion
            d  WebService
            e  Administrative Console
  -timeout+  Timeout for requests (default 10 seconds)
  -Userdbs  Load only user databases, not the standard databases
            all  Disable standard dbs and load only user dbs
            tests Disable only db_tests and load udb_tests
  -useragent Over-rides the default useragent
  -until+   Run until the specified time or duration
  -update+  Update database and plugins from CERT.net
  -useproxy  Use the proxy defined in nikto.conf, or argument http://server:port
```

Nikto scan for Onepush.query.yahoo.com

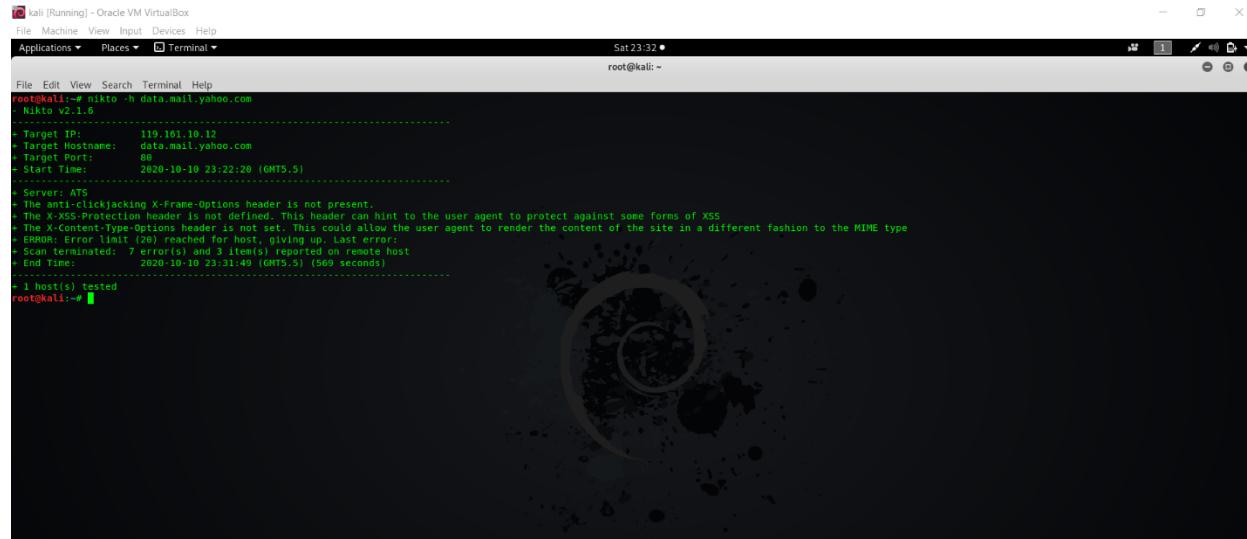


```
kali [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Applications ▾ Places ▾ Terminal ▾
Thu 22:16 •
root@kali: ~

File Edit View Search Terminal Help
root@kali:~# nikto -h onepush.query.yahoo.com
Nikto v2.1.5
=====
+ Target IP:      87.248.100.136
+ Target Hostname: onepush.query.yahoo.com
+ Target Port:    80
+ Start Time:    2020-10-08 21:58:16 (GMT5.5)
+ Threads:        4
+ Timeout:       10
+ Threads:        4
+ Server: No banner retrieved
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XSS
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type
+ No /ai Directories found (use --ai to force check all possible dirs)
+ Uncommon header 'x-amz-cdn-connection-downgrade' found, with contents: AES256
+ Uncommon header 'x-amz-server-side-encryption' found, with contents: D44236A6B0ECE3B0
+ Uncommon header 'x-amz-request-id' found, with contents: D44236A6B0ECE3B0
+ Uncommon header 'x-amz-id-2' found, with contents: 50H9QWHPKEW1W+1ILH2BLhvvcfCjZseYJaaexH412FevV11IP82L7fTM0B40BdywIrdvLG1
+ OSVDB-3692: /cfdocs/exampleapp/publish/admin/application.cfm: This might be interesting...
+ OSVDB-3692: /cfdocs/exampleapp/examples/httpclient/mainfrmset.cfc: This might be interesting...
+ Error: Error writing file /tmp/nikto2020-10-08-21-58-16-1395-1444888888: Can't open stream: Can't connect (timeout): Transport endpoint is not connected
+ OSVDB-3692: /config/checks.txt: This might be interesting...
+ Scan terminated: 17 error(s) and 10 item(s) reported on remote host
+ End Time:       2020-10-08 22:13:31 (GMT5.5) (1395 seconds)

+ 1 host(s) tested
root@kali:~#
```

Nikto scan for data.mail.yahoo.com



```
kali [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Applications ▾ Places ▾ Terminal ▾
Sat 23:32 •
root@kali: ~

File Edit View Search Terminal Help
root@kali:~# nikto -h data.mail.yahoo.com
Nikto v2.1.5
=====
+ Target IP:      119.161.10.12
+ Target Hostname: data.mail.yahoo.com
+ Target Port:    80
+ Start Time:    2020-10-10 23:22:20 (GMT5.5)
+ Threads:        4
+ Timeout:       10
+ Threads:        4
+ Server: ATS
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XSS
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type
+ ERROR: Error limit (20) reached for host giving up. Last error:
+ Scan terminated: 7 error(s) and 1 item(s) reported on remote host
+ End Time:       2020-10-10 23:31:49 (GMT5.5) (569 seconds)

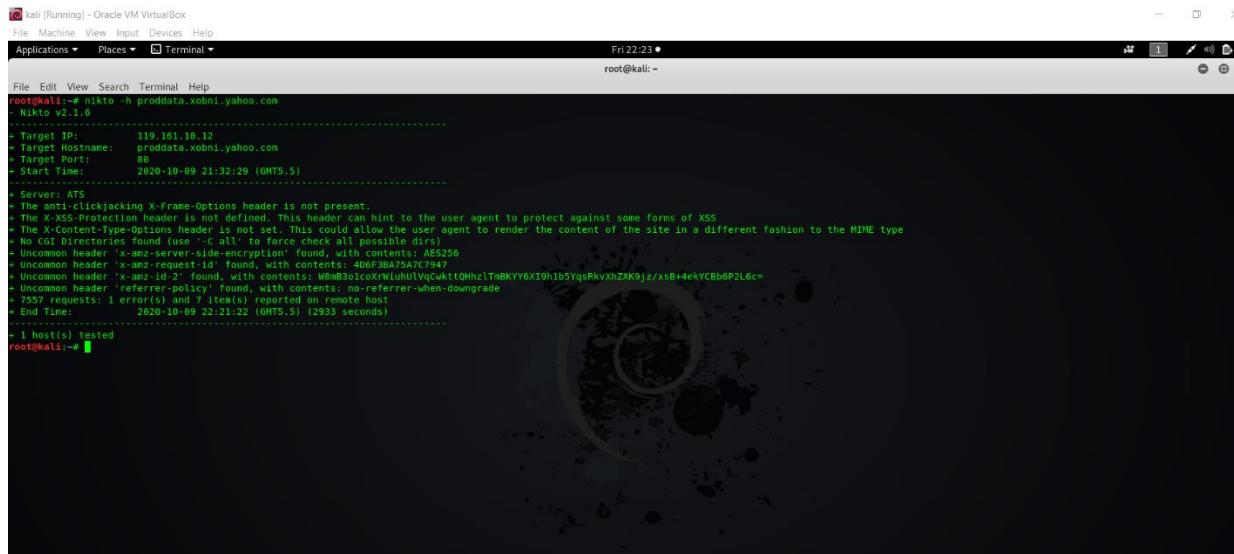
+ 1 host(s) tested
root@kali:~#
```

Nikto scan for mail.yahoo.com

Nikto scan for apis.mail.yahoo.com

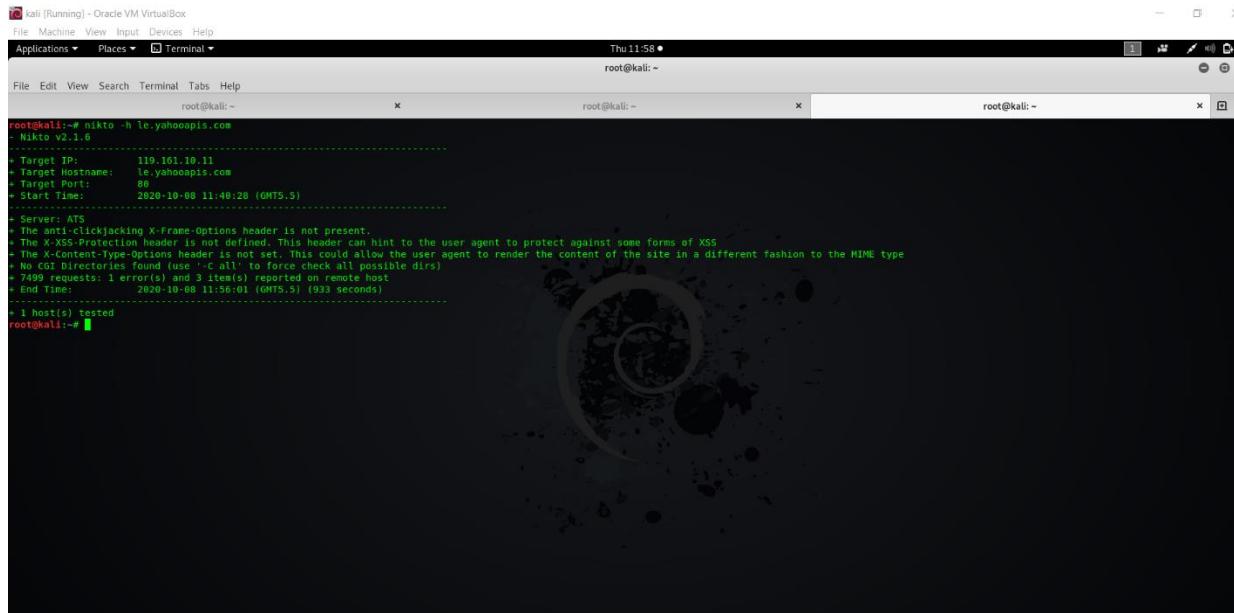
```
kali [Running] - Oracle VM VirtualBox  
File Machine View Input Devices Help  
Applications Places Terminal Fri 23:09 •  
root@kali: ~  
File Edit View Search Terminal Tabs Help  
root@kali: ~  
root@kali:~# nikto -h apis.mail.yahoo.com  
Nikto v2.1.6  
+ Target IP: 100.10.236.37  
+ Target Hostname: apis.mail.yahoo.com  
+ Target Port: 80  
+ Start Time: 2020-10-09 22:25:34 (GMT5.5)  
+ Server: ATS  
+ Url: /  
+ Test: Clickjacking X-Frame-Options header is not present.  
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XSS  
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type  
+ No CGI Directories found (use '-C all' to force check all possible dirs)  
+ Uncommon header 'x-amz-id-2' found, with contents: Miss38binI0440XD/HG7V2vKMRZVwRHLY4svENVkVx829fTS1WSUe4mLBgAcB9c3/y460dYrU=
```

Nikto scan for proddata.xobni.yahoo.com



```
kali [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Applications ▾ Places ▾ Terminal ▾
Fri 22:23 ●
root@kali: ~
File Edit View Search Terminal Help
root@kali:~# nikto -h proddata.xobni.yahoo.com
Nikto v2.1.5
=====
+ Target IP:      119.161.10.12
+ Target Hostname: proddata.xobni.yahoo.com
+ Target Port:    80
+ Start Time:   2020-10-09 21:32:29 (GMT5.5)
+ Threads:      4
+ Timeout:      10
+ Threads:      4
+ Server: ATS
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XSS
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type
+ No SSL Directories found (use --force-check-all to force check all possible dirs)
+ Uncommon header 'x-amz-server-side-encryption' found, with contents: AES256
+ Uncommon header 'x-amz-request-id' found, with contents: 406F3BA75A7C7947
+ Uncommon header 'x-amz-id-2' found, with contents: WmB3o1cXWluhUVqCwkttOHhzTnBKYY6X9h1b5YqsRkvXhZK9jz/xsB+ekYCB6bP2L6cm
+ Uncommon header 'referrer-policy' found, with contents: no-referrer-when-downgrade
+ 7557 requests; 1 error(s) and 7 item(s) reported on remote host
+ End Time:    2020-10-09 22:21:22 (GMT5.5) (2933 seconds)
+ 1 host(s) tested
root@kali:~#
```

Nikto scan for le.yahooapis.com



```
kali [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Applications ▾ Places ▾ Terminal ▾
Thu 11:58 ●
root@kali: ~
File Edit View Search Terminal Tabs Help
root@kali:~# nikto -h le.yahooapis.com
Nikto v2.1.5
=====
+ Target IP:      119.161.10.11
+ Target Hostname: le.yahooapis.com
+ Target Port:    80
+ Start Time:   2020-10-09 11:48:28 (GMT5.5)
+ Threads:      4
+ Timeout:      10
+ Threads:      4
+ Server: ATS
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XSS
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type
+ No SSL Directories found (use --force-check-all to force check all possible dirs)
+ 7449 requests; 1 error(s) and 3 item(s) reported on remote host
+ End Time:    2020-10-09 11:56:01 (GMT5.5) (933 seconds)
+ 1 host(s) tested
root@kali:~#
```

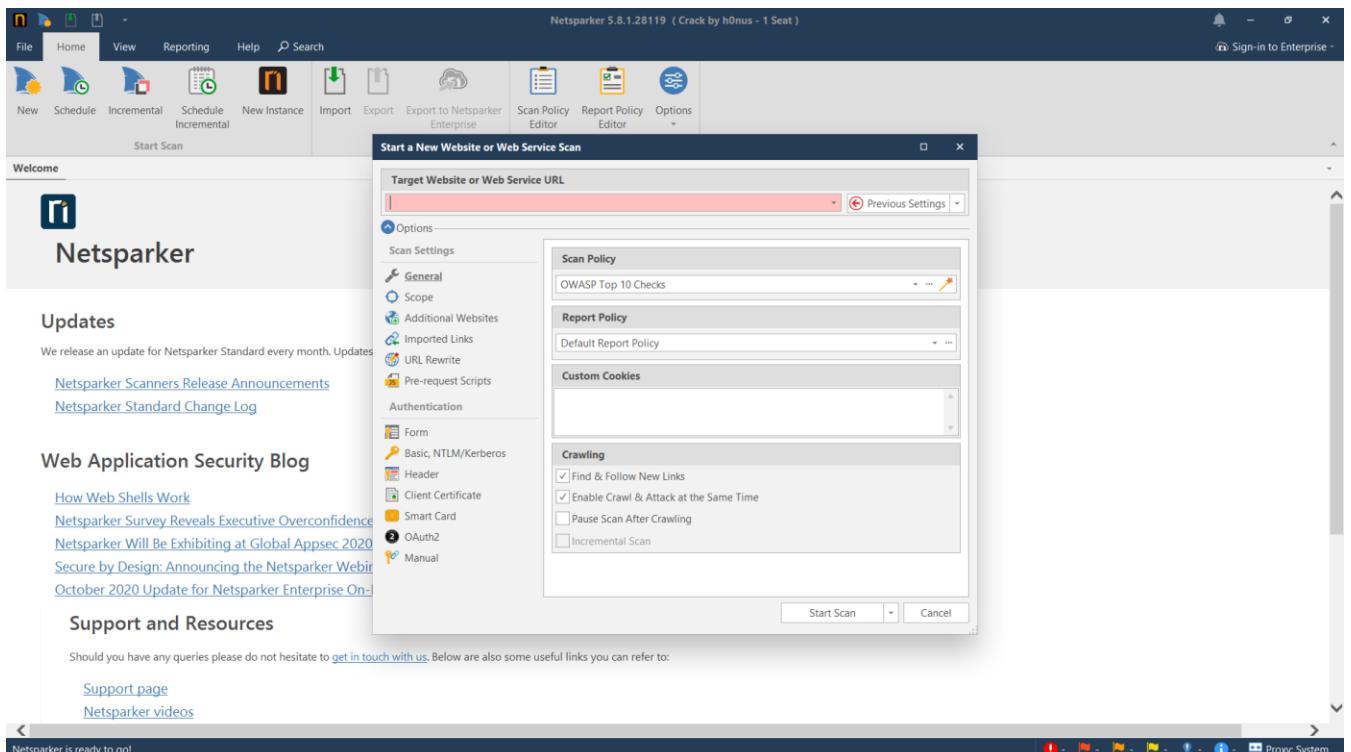
Netsparker

Netsparker is also an automated web application security scanner. It can scan web applications, web services and also websites. Netsparker can detect security issues of the system. and also it automatically discovers and protect your current web assets. Netsparker is not a just scanner it is a vulnerability management solution. Because the Netsparker provides so many facts of detected vulnerability. So we can easily identify what is the type of this vulnerability and also we can gain an understanding of the solution we can take to fix this vulnerability. One of the important of this application, it provides proof of the vulnerability. Netsparker can generate a proof when it identifies the following vulnerabilities, [7]

- SQL Injection
- Boolean SQL Injection
- Blind SQL Injection
- Remote File Inclusion (RFI)
- Command Injection
- Blind Command Injection
- XML External Entity (XXE) Injection
- Remote Code Evaluation
- Local File Inclusion (LFI)
- Server-side Template Injection
- Remote Code Execution
- Injection via Local File Inclusion

Also, lots of benefits have to this type of proof-based scanning. Furthermore, we can select the scan policy and also report policy in the Netsparker scanner.

The interface of the Netsparker



Netsparker scan of the onepushquery.yahoo.com

The screenshot shows the Netsparker interface for a completed scan of `apis.mail.yahoo.com`. The main window displays the `HTTP Request / Response` tab, showing a raw request and response. The request is a GET to `http://apis.mail.yahoo.com` with various headers. The response is an HTTP 404 Not Found page with the following content:

```
<!DOCTYPE html>
<html lang="en-us">
<head>
```

The left sidebar lists findings under the `Issues - Previous Settings` tab, including issues like missing X-Frame-Options Header, Content Security Policy (CSP) Not Implemented, and Weak Ciphers Enabled. The right sidebar shows the `Netsparker Assistant (0)` and a knowledge base with 8 items.

Netsparker scan of the data.mail.com

The screenshot shows the Netsparker interface for a completed scan of `data.mail.yahoo.com`. The main window displays the `HTTP Request / Response` tab, showing a raw request and response. The request is a GET to `http://data.mail.yahoo.com` with various headers. The response is an HTTP 404 Not Found page with the following content:

```
<!DOCTYPE html>
<html lang="en-us">
<head>
```

The left sidebar lists findings under the `Issues - Previous Settings` tab, including issues like missing X-Frame-Options Header, Content Security Policy (CSP) Not Implemented, and Weak Ciphers Enabled. The right sidebar shows the `Netsparker Assistant (0)` and a knowledge base with 8 items.

Netsparker scan of the mail.yahoo.com

The screenshot shows the Netsparker application interface. At the top, there's a toolbar with File, Home, View, Reporting, Help, and various icons for controlled scan, send to request builder, copy URL, etc. Below the toolbar is a browser view showing the URL mail.yahoo.com. The main area is divided into two panes: "HTTP Request / Response" and "Browser View". In the "Request" pane, a raw HTTP request is shown:

```

GET / HTTP/1.1
Host: mail.yahoo.com
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Accept-Encoding: gzip, deflate
Accept-Language: en-us,en;q=0.5
Cache-Control: no-cache
User-Agent: Mozilla/5.0 (Windows NT 10.0; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.77 Sa
X-Scanner: Netsparker

```

The "Response" pane shows the raw response:

```

HTTP/1.1 302 Redirect
Server: ATS
Connection: keep-alive
Content-Length: 305
Content-Language: en
Content-Type: text/html
Location: https://mail.yahoo.com/
Date: Thu, 15 Oct 2020 05:20:24 GMT
Cache-Control: no-store
<HTML>
<HEAD>

```

On the left, there's a "Sitemap - Previous Settings" tree and an "Issues - Previous Settings" list. A "Knowledge Base" pane on the right lists items like "Maximum Signature Exceeded" and "SSL [1]". The bottom status bar indicates "Session loaded successfully. Scan status: finished."

Netsparker scan of the apis.mail.yahoo

This screenshot shows the Netsparker interface scanning the apis.mail.yahoo domain. The layout is similar to the previous screenshot, with a toolbar, browser view, and two main panes for requests and responses. The "Request" pane shows:

```

GET / HTTP/1.1
Host: apis.mail.yahoo.com
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Accept-Encoding: gzip, deflate
Accept-Language: en-us,en;q=0.5
Cache-Control: no-cache
User-Agent: Mozilla/5.0 (Windows NT 10.0; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.77 Sa
X-Scanner: Netsparker

```

The "Response" pane shows:

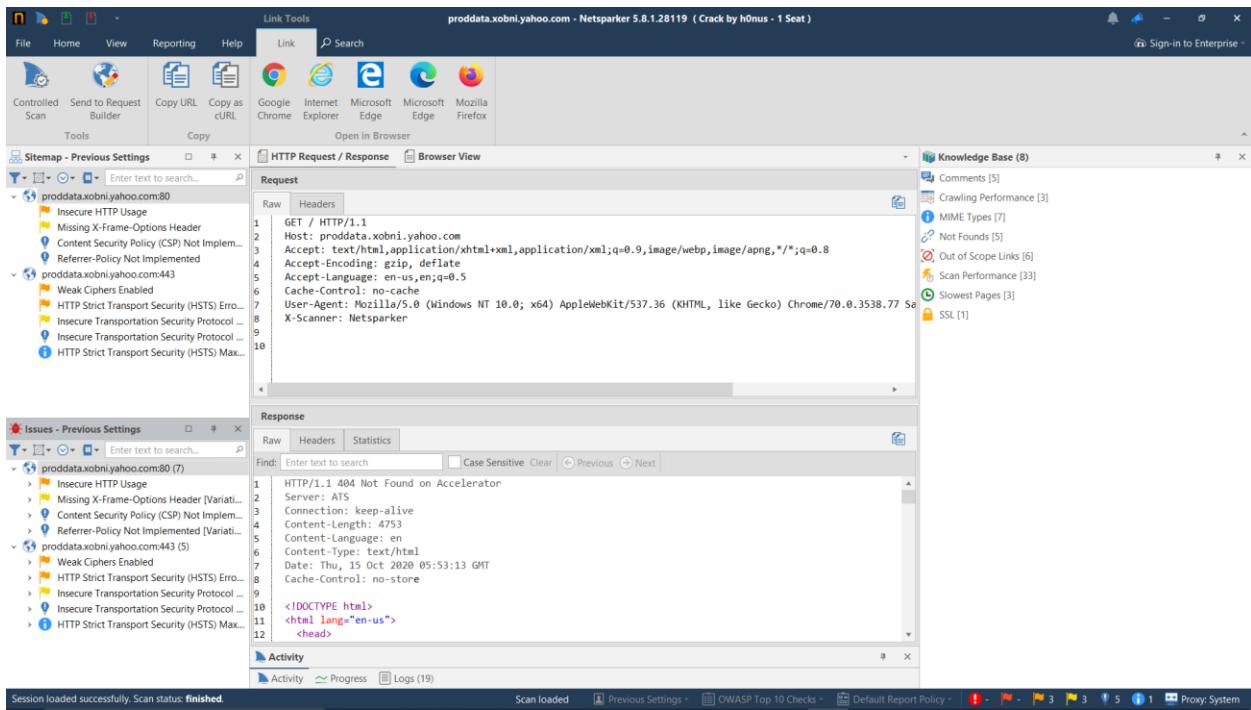
```

HTTP/1.1 404 Not Found on Accelerator
Server: ATS
Connection: keep-alive
Content-Length: 4750
Content-Language: en
Content-Type: text/html
Date: Thu, 15 Oct 2020 05:56:55 GMT
Cache-Control: no-store
<!DOCTYPE html>
<html lang="en-us">
<head>

```

The left sidebar shows a "Sitemap - Previous Settings" tree and an "Issues - Previous Settings" list. The "Knowledge Base" pane on the right lists items such as "Comments [5]", "Crawling Performance [3]", and "SSL [1]". The status bar at the bottom says "Session loaded successfully. Scan status: finished."

Netsparker scan of the proddata.xobni.yahoo.com

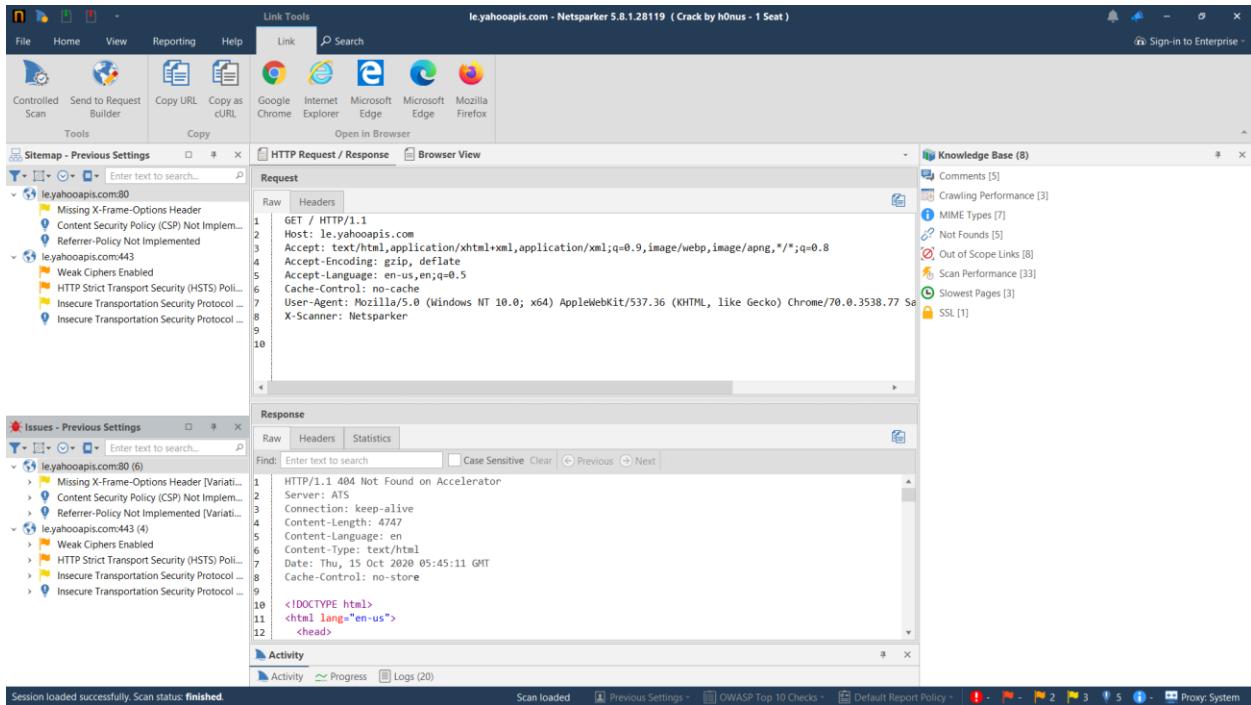


The screenshot shows the Netsparker interface with the following details:

- Link Tools:** Includes Scan, Send to Request Builder, Copy URL, Copy as cURL, and Tools.
- Browser View:** Shows icons for Google Chrome, Internet Explorer, Microsoft Edge, Mozilla Firefox, and Open in Browser.
- Sitemap - Previous Settings:** Lists the host proddata.xobni.yahoo.com:80 with various vulnerabilities:
 - Insecure HTTP Usage
 - Missing X-Frame-Options Header
 - Content Security Policy (CSP) Not Implemented
 - Weak Ciphers Enabled
 - HTTP Strict Transport Security (HSTS) Errors
 - Insecure Transportation Security Protocol
 - Insecure Transportation Security Protocol
 - HTTP Strict Transport Security (HSTS) Max-Age
- HTTP Request / Response:** Displays the raw request and response headers for a GET / HTTP/1.1 request to proddata.xobni.yahoo.com:80. The response shows a 404 Not Found error page with the following content:

```
<!DOCTYPE html>
<html lang="en-us">
<head>
```
- Knowledge Base (8):** Lists various security topics with counts:
 - MIME Types [7]
 - Not Found [5]
 - Out of Scope Links [6]
 - Scan Performance [33]
 - Slowest Pages [3]
 - SSL [1]
- Issues - Previous Settings:** Lists issues for proddata.xobni.yahoo.com:80 and proddata.xobni.yahoo.com:443, including:
 - Insecure HTTP Usage
 - Missing X-Frame-Options Header
 - Content Security Policy (CSP) Not Implemented
 - Weak Ciphers Enabled
 - HTTP Strict Transport Security (HSTS) Errors
 - Insecure Transportation Security Protocol
 - Insecure Transportation Security Protocol
 - HTTP Strict Transport Security (HSTS) Max-Age
- Activity:** Shows activity logs (19).

Netsparker scan of the le.yahooapis.com



The screenshot shows the Netsparker interface with the following details:

- Link Tools:** Includes Scan, Send to Request Builder, Copy URL, Copy as cURL, and Tools.
- Browser View:** Shows icons for Google Chrome, Internet Explorer, Microsoft Edge, Mozilla Firefox, and Open in Browser.
- Sitemap - Previous Settings:** Lists the host le.yahooapis.com with various vulnerabilities:
 - Missing X-Frame-Options Header
 - Content Security Policy (CSP) Not Implemented
 - Weak Ciphers Enabled
 - HTTP Strict Transport Security (HSTS) Policies
 - Insecure Transportation Security Protocol
 - Insecure Transportation Security Protocol
- HTTP Request / Response:** Displays the raw request and response headers for a GET / HTTP/1.1 request to le.yahooapis.com. The response shows a 404 Not Found error page with the following content:

```
<!DOCTYPE html>
<html lang="en-us">
<head>
```
- Knowledge Base (8):** Lists various security topics with counts:
 - MIME Types [7]
 - Not Found [5]
 - Out of Scope Links [8]
 - Scan Performance [33]
 - Slowest Pages [3]
 - SSL [1]
- Issues - Previous Settings:** Lists issues for le.yahooapis.com:80 and le.yahooapis.com:443, including:
 - Missing X-Frame-Options Header
 - Content Security Policy (CSP) Not Implemented
 - Weak Ciphers Enabled
 - HTTP Strict Transport Security (HSTS) Policies
 - Insecure Transportation Security Protocol
 - Insecure Transportation Security Protocol
- Activity:** Shows activity logs (20).

After these scans of Nikto and also Netsparker I find some vulnerabilities of these subdomains.

Vulnerability analysis

Vulnerabilities identified in the Onepush.query.yahoo.com

✓ Insecure HTTP Usage

This insecure HTTP header usage is a medium vulnerability which identifies by Netsparker. The identity of this vulnerability according to OWASP is 2017-A3. What happens here is that when implemented target website however HTTP request is not redirected to HTTPS. This vulnerability is vulnerable to Man in the Middle (MITM) attack because the user will not able to get the benefits of HTTP strict transport security (HSTS) and almost renders the implementation is useless.

Solution

Configure webserver to redirect HTTP requests to HTTPS requests.

✓ Weak Ciphers enabled

This is a confirmed medium vulnerability identified by Netsparker. Classification of this vulnerability is OWASP 2017 –A3 or CWE-327. This vulnerability is identified during secure communication.

The impact of this weak ciphers enabled vulnerability is attackers can decrypt secure socket layer (SSL) traffic between web application and users.

Solutions

Configure webserver to disallow when using weak ciphers.

In the apache server, modify the SSLCipher suite.

✓ Cookie not marked as HTTP only

This vulnerability can expose to cross-site scripting attacks. Through this vulnerable attacker can easily hijack the victim's session.

- ✓ **Same-site cookie not implemented**

This vulnerable can expose to the CSRF attack.

- ✓ **Missing X-frame option header**

- ✓ **Insecure transportation security protocol supported**

Vulnerabilities identified in the data.mail.yahoo.com

✓ Missing X-frame option header

The x-frame option header missing means this application can be the risk of a clickjacking attack. A clickjacking attack is that prompts a user to click on a web page feature in malicious. This can cause to download malicious software into the users' devices.

Solution

- Sending the proper X-Frame-Options in HTTP response headers.

- **X-Frame-Options: DENY**

It completely denies being loaded in frame/iframe.

- **X-Frame-Options: SAMEORIGIN**

It allows only if the site which wants to load has the same origin.

- **X-Frame-Options: ALLOW-FROM <URL>**

It grants a specific URL to load itself in an iframe.

✓ HSTS Policy is not enabled

✓ Weak Ciphers enabled

This is a confirmed medium vulnerability identified by Netsparker. Classification of this vulnerability is OWASP 2017 –A3 or CWE-327. This vulnerability is identified during secure communication.

The impact of this weak ciphers enabled vulnerability is attackers can decrypt secure socket layer (SSL) traffic between web application and users.

Solutions

Configure webserver to disallow when using weak ciphers.

In the apache server, modify the SSLCipher suite.

- ✓ **Insecure transportation security protocol supported.**

Vulnerabilities identified in the mail.yahoo.com

✓ Vulnerable to cross-site scripting

In here attackers can execute some dynamic scripts. This can cause several attacks.

Mostly risk on hijacking attack. Other risks of this vulnerable are phishing attack, intercepting data and perform MiTM attack and changing the user interface.

Solutions

- Enabled content security policy(CSP).
- Outputs and inputs are should be encoded.
- Include well-structured whitelist libraries

✓ Autocomplete is enable

In here, some sensitive data like username, password, credit card no and CVV are automatically filled. If the users select the save option and this sensitive data is going to be cached by the browser. And then attackers can steal these users' sensitive data.

Solution

The “Autocomplete” attribute is disabled.

✓ Detected robot.txt file

In the Netsparker scan detected this text file that includes some sensitive data. So attackers might be can discover this file and then it issues to users' data.

✓ Cross-site referrer leakage through referrer policy.

Through this vulnerability can leak some domain information through the referrer header. When we check the available option by using external reference and use one suit for your need.

Vulnerabilities identified in the apis.mail.yahoo.com

✓ Weak Ciphers enabled.

This is a confirmed medium vulnerability identified by Netsparker. Classification of this vulnerability is OWASP 2017 –A3 or CWE-327. This vulnerability is identified during secure communication.

The impact of this weak ciphers enabled vulnerability is attackers can decrypt secure socket layer (SSL) traffic between web application and users.

Solutions

Configure webserver to disallow when using weak ciphers.

In the apache server, modify the SSLCipher suite.

✓ HSTS Policy is not enabled.

HSTS policy is used to communicate with the server to the user. HSTS is the security policy of the web servers that implement for users to interact with a website using a secure (HTTPS) connection.

Solution

Configure webserver to redirect HTTP requests to HTTPS.

✓ Missing X-Frame-Options Header.

✓ Referrer-Policy Not Implemented.

Referrer-Policy is a security header that implements to prevent cross-domain referer leakage. The lack of a Referrer-Policy header might affect the privacy of the users and also the website.

Solutions

Implement Referrer-Policy response header.

Declare the meta tags.

Vulnerabilities identified in the proddata.xobni.yahoo.com

✓ Insecure HTTP Usage.

This insecure HTTP header usage is a medium vulnerability which identifies by Netsparker. The identity of this vulnerability according to OWASP is 2017-A3. What happens here is that when implemented target website however HTTP request is not redirected to HTTPS.

This vulnerability is vulnerable to Man in the Middle (MITM) attack because the user will not able to get the benefits of HTTP strict transport security (HSTS) and almost renders the implementation is useless.

Solution

Configure webserver to redirect HTTP requests to HTTPS requests.

✓ Missing X-frame option header.

The x-frame option header missing means this application can be the risk of a clickjacking attack. A clickjacking attack is that prompts a user to click on a web page feature in malicious. This can cause to download malicious software into the users' devices.

Solution

Sending the proper X-Frame-Options in HTTP response headers.

✓ Referrer-Policy Not Implemented.

Referrer-Policy is a security header that implements to prevent cross-domain referer leakage. The lack of a Referrer-Policy header might affect the privacy of the users and also the website.

Vulnerabilities identified in the le.yahooapis.com

✓ Content Security Policy (CSP) Not Implemented.

CSP is used to mitigate Cross-site Scripting attacks. This vulnerability has not a direct impact on the website but if the website is vulnerable to a Cross-site Scripting attack, then CSP can prevent the successful exploitation of that vulnerability.

Solution

Enable CSP on the website by sending the Content-Security-Policy in HTTP response headers.

✓ Weak Ciphers enabled.

This is a confirmed medium vulnerability identified by Netsparker. Classification of this vulnerability is OWASP 2017 –A3 or CWE-327. This vulnerability is identified during secure communication.

The impact of this weak ciphers enabled vulnerability is attackers can decrypt secure socket layer (SSL) traffic between web application and users.

Solutions

Configure webserver to disallow when using weak ciphers.

In the apache server, modify the SSLCipher suite.

✓ (HSTS) Policy Not Enabled

✓ Referrer-Policy Not Implemented.

Referrer-Policy is a security header that implements to prevent cross-domain referer leakage. The lack of a Referrer-Policy header might affect the privacy of the users and also the website.

Solutions

Implement Referrer-Policy response header.

Declare the meta tags.

Conclusion

Overall, through this web security audit performed on several subdomains that I selected in yahoo.com. While this testing I identified several critical vulnerabilities as well as a lot of medium vulnerabilities on these subdomains. Furthermore through this documentation provide an idea of web security auditing and also what benefits are coming from this audit to web application. Also in this documentation discussed how to perform web auditing and also what tools can use to do the audit. As well as a discussed result about what identified vulnerabilities are in I selected domain.

Today, most people are using web applications for their work. Also, they access their sensitive data through this web application. Therefore web applications want to follow a good security system because if not users' data have at risk. That's why security analysis is recommended for doing a web security audit and implement or improve their security. It can protect their users' data and also they can prevent cyber attacks.

References :

- [1] J. Varghese, Website Security Audit, Astra, October 9, 2020.
- [2] OWASP Mobile Top 10 Risks, Promon.
- [3] OWASP Top 10 Security Risks & Vulnerabilities, Sucuri Guides.
- [4] sublist3r Package Description, Kali Tools.
- [5] M. Ferranti, What is Nmap? Why you need this network mapper, Network World , AUG 17, 2018.
- [6] What is a Vulnerability Scan?, PACKETLABS.
- [7] What is Netsparker?, Netsparker.