BUG BOUNTY ASSIGNMENT

Domain	**: https://www.jimdo.com/**
Student Name	**: P.A. Daham Thameera**
Student Registration Number	**: IT20077624**
Date of submission	**: 18/10/2021**
Batch	**: Group 13.1**

Sri Lanka Institute of Information Technology
Web Security - IE2062 [2021/JUL]

Table of Contents

Acknowledgment

I would like to express my sincere gratitude to Dr. Lakmal Rupasinghe for his relentless effort in guiding us and advising us through difficult and unfamiliar phases of the project and helping us gain practical knowledge and skills in the subject.

It is with heartfelt appreciation that I thank Ms. Chetana Lyanapathirana, Ms. Lanisha Rugghahakotuwa, and Ms. Chathuri Udagedra for helping me throughout the semester to gain new knowledge and understand concepts of Web security and how to implement them in the practical world.

Purpose

The purpose of this assignment is the assess vulnerabilities of the web application. So, https://www.hackerone.com/ (Fig.1) platform is used to find the websites and web applications for the Bug Bounty hunting. And there are a lot of Bug Bounty hunting platforms to improve our vulnerability assessing skills. As an example, https://www.bugcrowd.com/ is one of the Bug Bounty hunting platforms. So, the purpose of using this Hackerone platform is because this website legally protects us to do Bug Bounty hunting for real-world web applications.

Using these websites benefits to get powerful knowledge about the penetration testing tool and how to use those tools. And these web audit reports are giving an excellent understanding of how to handle cybersecurity profession skills.

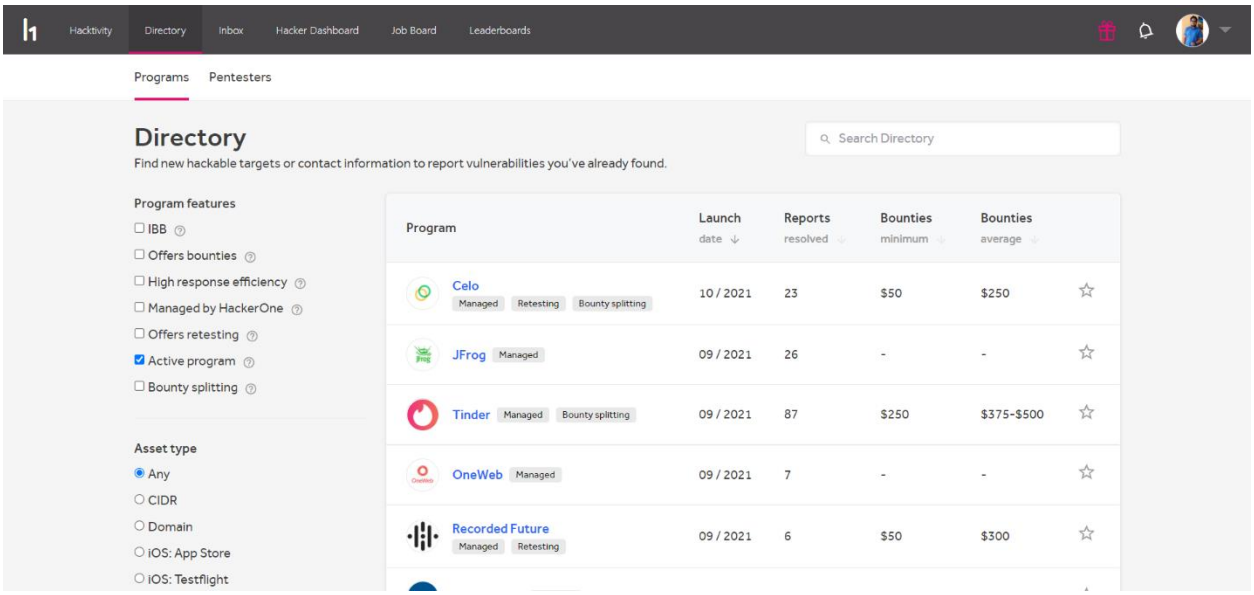


Fig. 1. https://hackerone.com/

Introduction

Web security is critical to web-based companies and businesses because cybercrime is increasing day by day. Every moment attackers are finding new paths for exploiting the web applications. And attackers develop their skills not only for fun they focus on money also. That is why ransomware attacks are most popular these days. Because of that protection is a must for web applications to defend against this type of cybercrime.

So, a lot of web-based companies and businesses are assigned to Bug Bounty programs to detect the vulnerabilities and fix those vulnerable domains before getting into attack. Hackerone(https://www.hackerone.com/) is one of the platforms that help web-based companies to fix vulnerabilities through Bug Bounty programs. And Hackerone platform and web-based companies are paying for penetration testing their web domains. So, I selected a web-based company called Jimdo(https://www.jimdo.com) for my Bug Bounty hunting program (Fig. 2). Jimdo platform is used by customers to create their websites (Fig. 3). This includes the ability to add custom JavaScript code to their website. And Jimdo offers a large number of subdomains to this Bug Bounty hunting program and also the little number of reports submitted because they did not pay for those reports and the service.

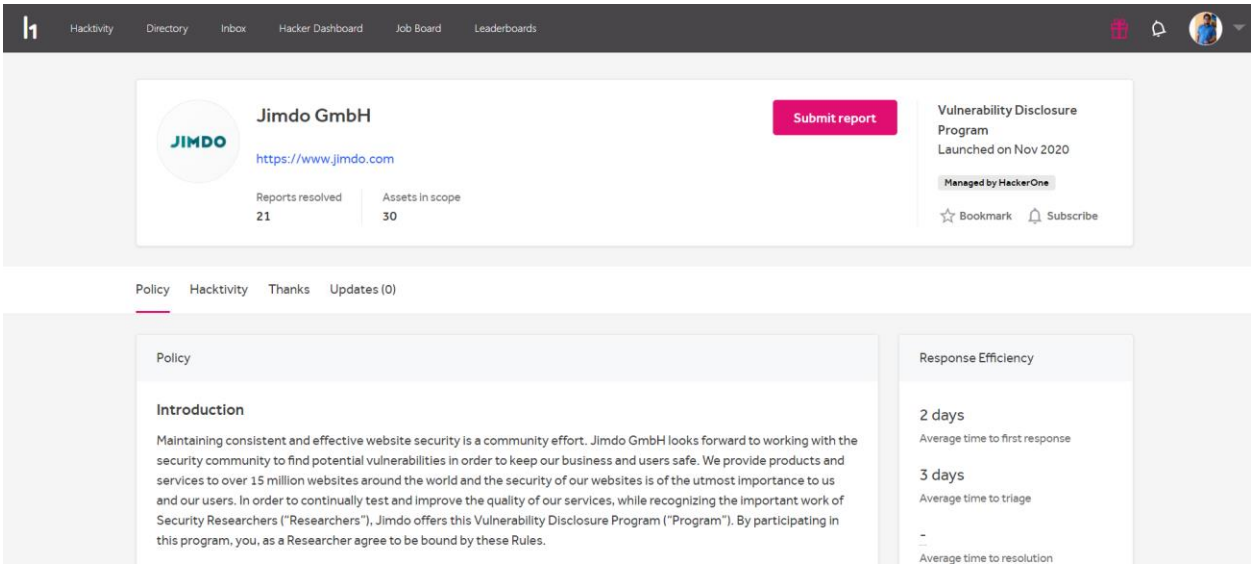


Fig. 2. Jimdo's Hackerone page

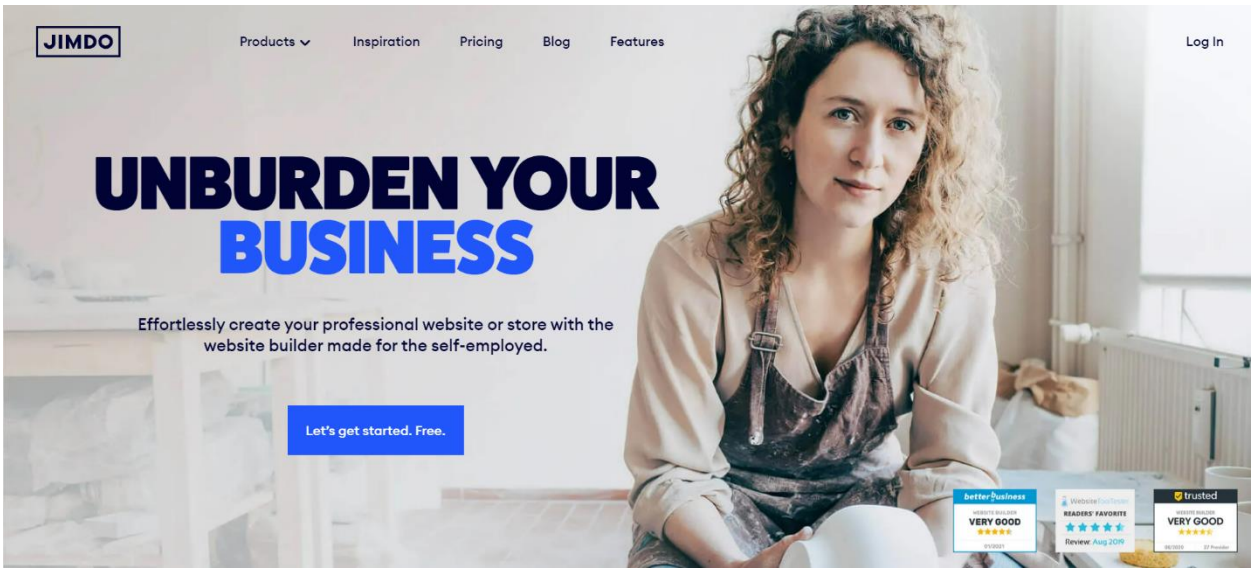


Fig. 3. https://www.jimdo.com

This Bug Bounty Assignment is used to be done according to the following web application security testing methodology (Fig. 4).

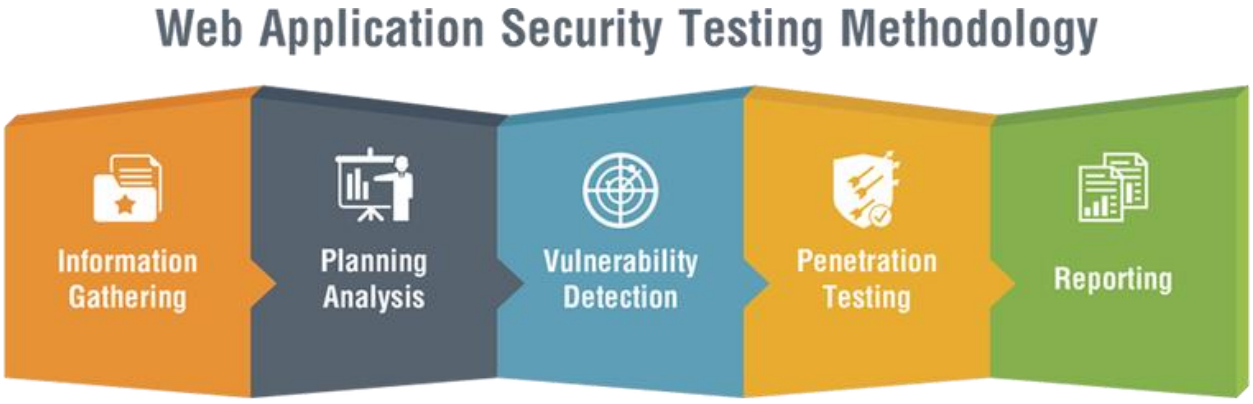


Fig. 4. web application security testing methodology

Before moving into the information gathering stage, we need to consider the top 10 web application's Security Risks and vulnerabilities in 2021. Because we can get an excellent idea for success in our information gathering stage. According to the Sucuri Guides, The Open Web Application Security Project (OWASP) is an online community that creates web application security papers, techniques, documentation, tools, and technologies. The OWASP Top 10 is a list of the top ten most frequent application Security Risks and vulnerabilities [1].

- 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

So, these are the top 10 vulnerabilities found by the OWASP in 2021. We need to focus on these types of vulnerabilities according to the scope and rules provided by Jimdo web-based company and the Hackerone platform.

The scope of the investigation was restricted to 30 domains of Jimdos's primary web application.

- www.jimdo.com
- cms.jimdo.com
- cms.e.jimdo.com
- jimdofree.com
- help.jimdo.com
- jimdo.design
- logo.e.jimdo.com
- account.e.jimdo.com

These are the domains selected to perform this Bug Bounty hunting program.

Information Gathering

Information gathering is the first step to building a strong foundation for this Bug Bounty hunting program. Because this step is about collecting the critical details of the targeted web application. If this step is not done well entire project can be a useless effort. So, more information means that we can capture more vulnerabilities from targeted domains. As an example, we have to find the targeted domain's IP addresses, details about open ports in the targeted domain, and what type of protection they use to protect their web application. According to the All About Testing (AAT), "The more useful information you have about a target, the more you can find vulnerabilities in the target and find more serious problems in the target by exploiting them." [2]. So, perfect information gathering is key to unlocking vulnerabilities from the target and it will help improve our vulnerability scanning process.

Information Gathering can be divided into two parts. They are,

1. Passive information gathering
    - Passive information gathering is collecting information from the targeted domain without invoking any kind of communication with the target systems.
2. Active information gathering
    - Active information gathering is collecting information from the targeted domain involves monitoring the target systems by building communication with the target. This method is detectable to the targeted system.

Considering the Passive and Active information gathering, there are many tools to gather information from the target domain using both methods. They are,

1. Passive information gathering tools
    - sublist3r
    - nslookup
    - whois
    - whatweb
    - dig
    - Netcraft (https://sitereport.netcraft.com/)
    - Whois Lookup (https://whois.domaintools.com/)

2. Active information gathering tools
    - Nmap
    - Dmitry

These are the information-gathering tools used to analyze the targeted web domain. And I give priority to Passive information gathering tools. Because Active information gathering is very noisy. But we need active information gathering to analyze information about what is open ports are in our targeted system.

1. Passive information gathering tools

➢ Sublist3r

Sublist3r is a subdomain enumeration tool. That means this is a tool to identify the unique subdomains associated with the target domain. Because of this tool, we can gather more information about subdomains. This tool is not built-in and comes with Kali Linux operating system and first, we need to install this tool in the Kali Linux operating system.

- Download the Sublist3r [3].

```
┌──(daham㉿kali)-[~/Documents/Tools]
└─$ git clone https://github.com/aboul3la/Sublist3r.git

Cloning into 'Sublist3r'...
remote: Enumerating objects: 383, done.
remote: Total 383 (delta 0), reused 0 (delta 0), pack-reused 383
Receiving objects: 100% (383/383), 1.12 MiB | 1.42 MiB/s, done.
Resolving deltas: 100% (212/212), done.
```

Fig. 5. Download the Sublist3r using github link

- Check the downloaded location and go into the Sublist3r directory.

```
┌──(daham㉿kali)-[~/Documents/Tools]
└─$ ls
Sublist3r

┌──(daham㉿kali)-[~/Documents/Tools]
└─$ cd Sublist3r

┌──(daham㉿kali)-[~/Documents/Tools/Sublist3r]
└─$ ls
LICENSE      README.md         setup.py  sublist3r.py
MANIFEST.in  requirements.txt  subbrute
```

Fig. 6. Go into Sublist3r directory

- Install Python3-pip in Kali Linux.

Fig. 7. Install Python3-pip

- Install Dependencies in the Sublist3r directory.

Fig. 8. Install Dependencies

- Install argparse module in the Sublist3r directory.

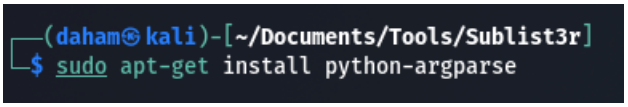


Fig. 9. Install argparse module

- Check Sublist3r is ready to use and test the tool.

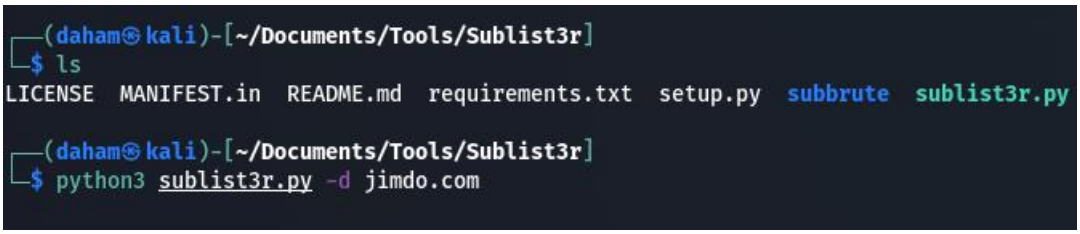


Fig. 10. Checking the tool

After installing Sublist3r next step is to do scan the main domain to capture subdomains in the targeted system (Jimdo.com).

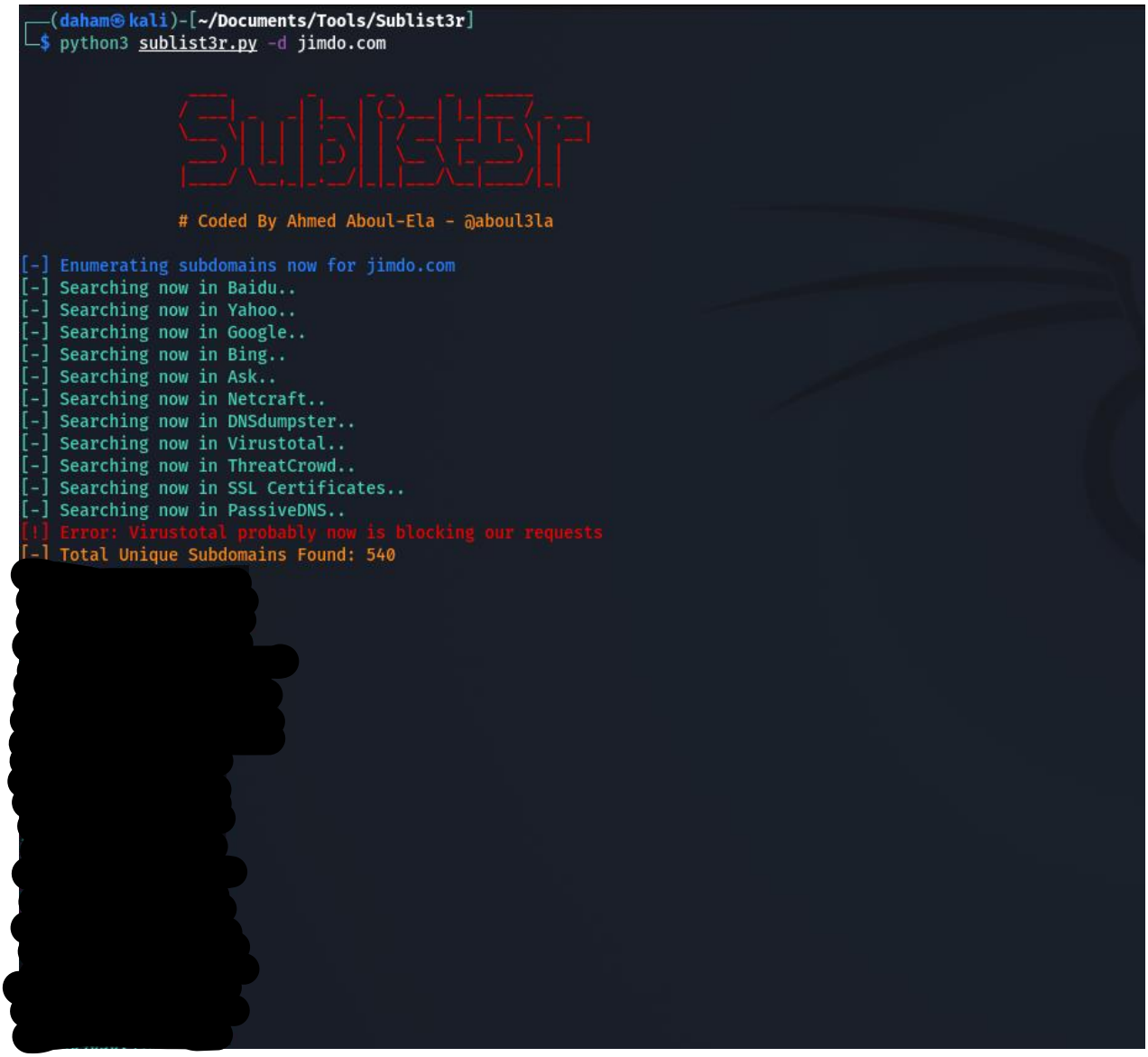


Fig. 11. Sublist3r scan result

After the scan, the Sublist3r tool found 540 unique subdomains related to the main domain (Jimdo.com).

- **Nslookup**

Nslookup is perfect DNS enumeration. That means this is a tool for gathering information about the Domain Name System (DNS) of the targeted system. Nslookup tool help to find out the information related to DNS record names, IP addresses of a target, DNS domain names, and the MX records for the domain or the NS servers of the domain. This tool is already built in the Kali Linux environment. So, I gather the information that

all selected domains to get a better understanding of DNS information related to the web application (Jimdo.com).

- Gather information about the IP address of the hostname.

Fig. 12. IP address of the hostname (Jimdo.com)

- Gather information about the mail exchange (MX) records.

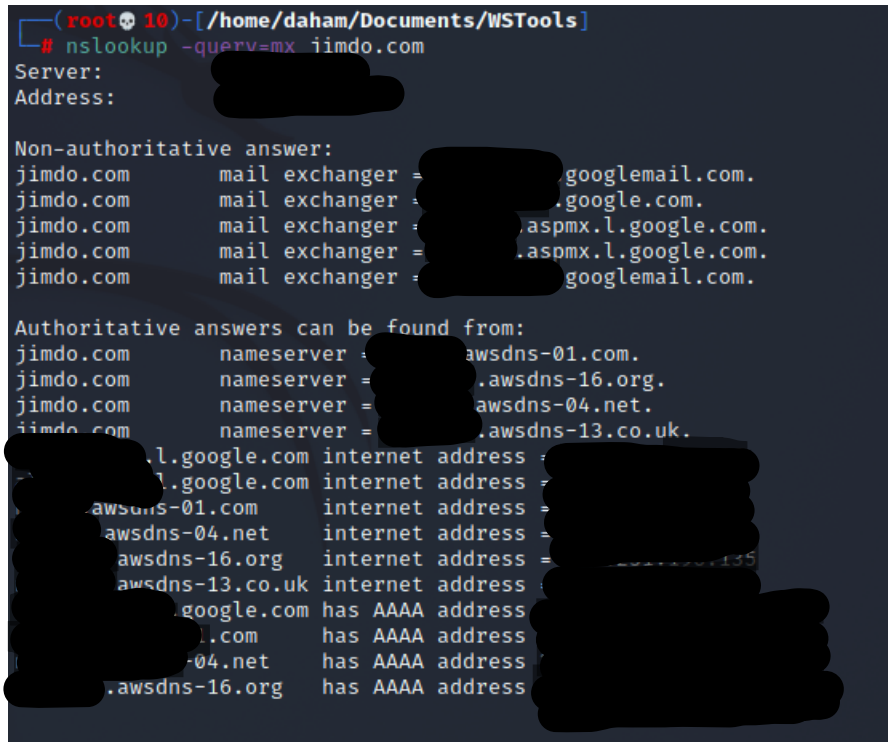


Fig. 13. MX records (-query=mx) of the Jimdo.com

- Gather information about the nameserver (NS) records.

Fig. 14. NS records (-query=ns) of the Jimdo.com

- Gather information about the "start of authority" (SOA) records. That means we can get details about the domain or region, like the administrator's email address, how long the server should wait between refreshes, and the very last time the domain was modified.

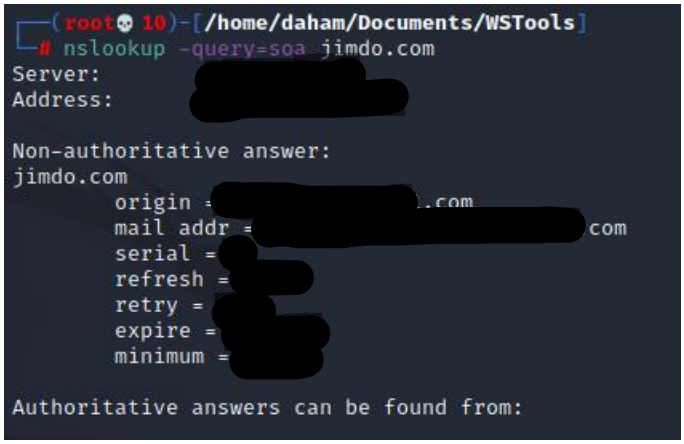


Fig. 15. OSA records (-query=OSA) of the Jimdo.com

- “Any” keyword can use gather all the above information using only one command. So, I use that command to gather information on the in-scope domains.

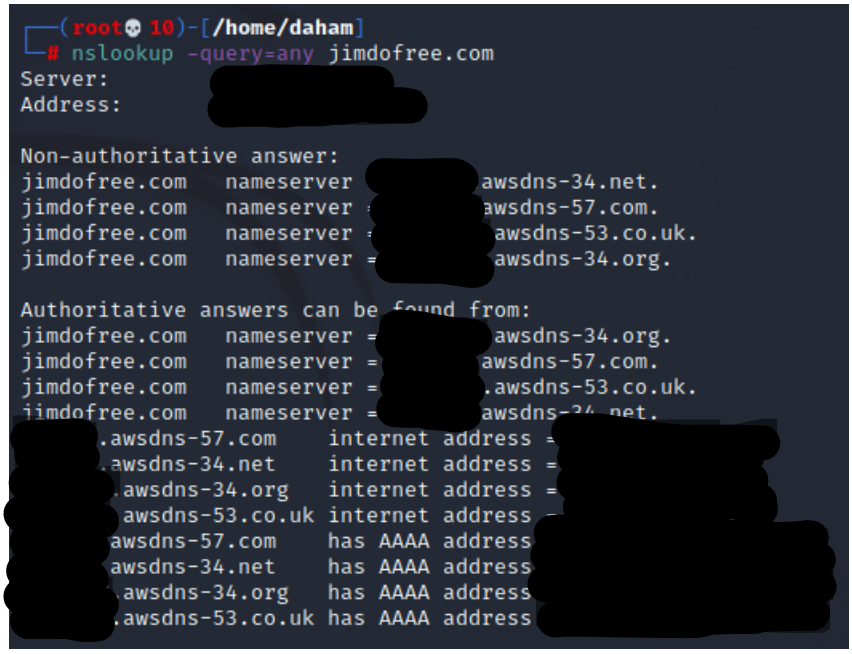


Fig. 16. Gather all information using "-query=any" examples

➢ **Whois**

Whois command gathers information related to targeted domain unknown and distant hosts, server information, network details, and many more details. This command also has a lot of filtering options and uses that "whois --help" command to grant filtering techniques (Fig. 17.).

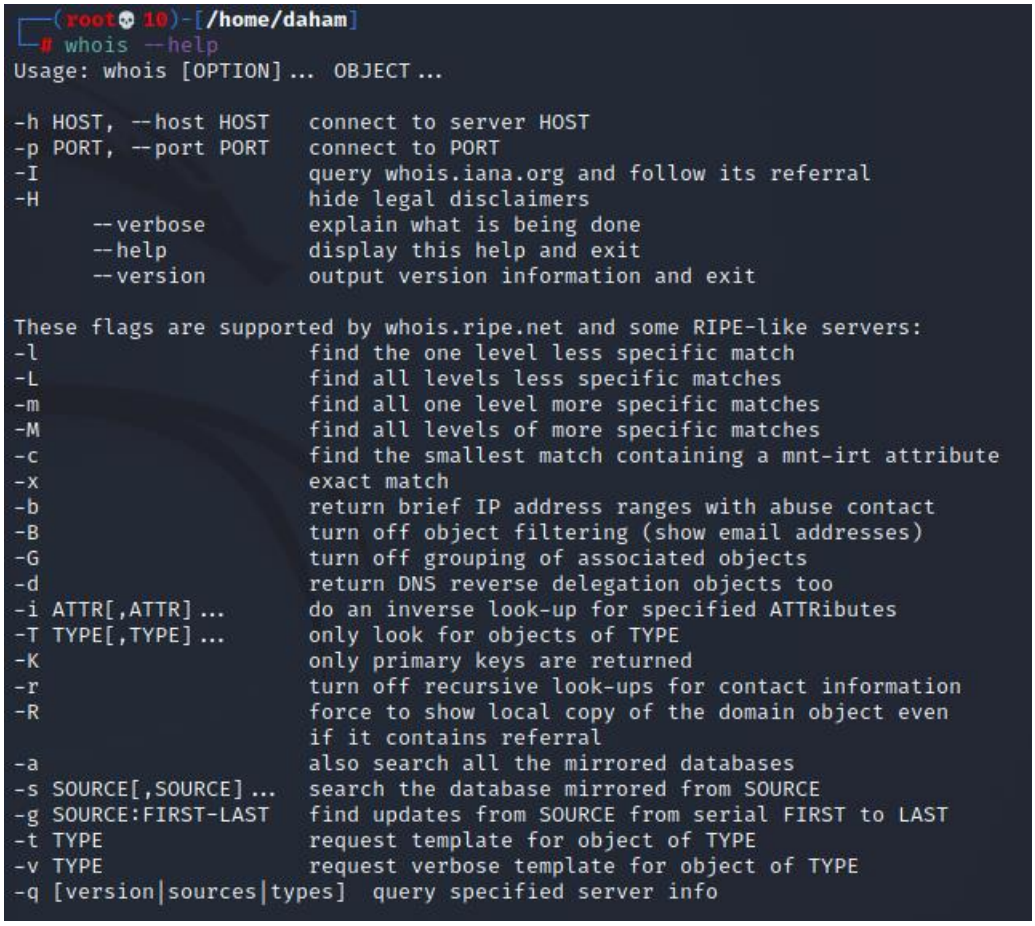


Fig. 17. Whois --help

I did not want to filter the output because I need a full detailed report for my information gathering process. So, these are the sample output of this command (Fig. 18.).

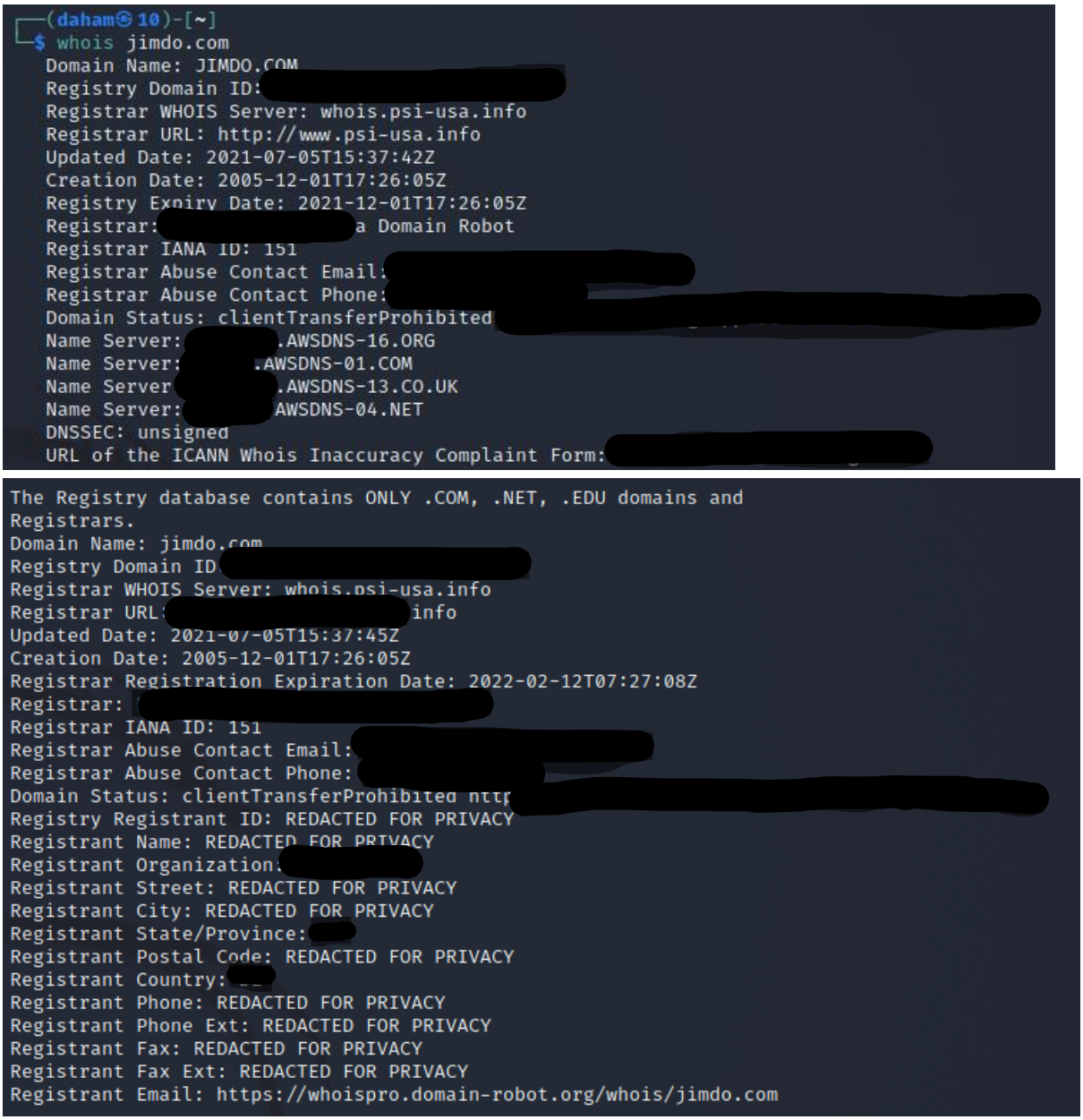


Fig. 18. Whois Jimdo.com

- **Whatweb**

According to Kali Linux, “WhatWeb identifies websites. It recognizes web technologies including content management systems (CMS), blogging platforms, statistic/analytics packages, JavaScript libraries, web servers, and embedded devices.” [4]. This tool is very powerful because we can capture a lot of details using this Whatweb tool. Specially, we can gather information about what type of protection mechanism is used that the targeted domain to

protect their web application. But the output information is not sorted well. So, we can use filtering options to gather information in a sorted way.

Fig. 19. whatweb -h

But I did not filter the output because I need informative result about what kind of protection method that the targeted domain use to protect their web application and the filtering is a time-consuming process.

- Gather information related to www.jimdo.com

Fig. 20. whatweb www.jimdo.com

- Gather information related to csm.jimdo.com

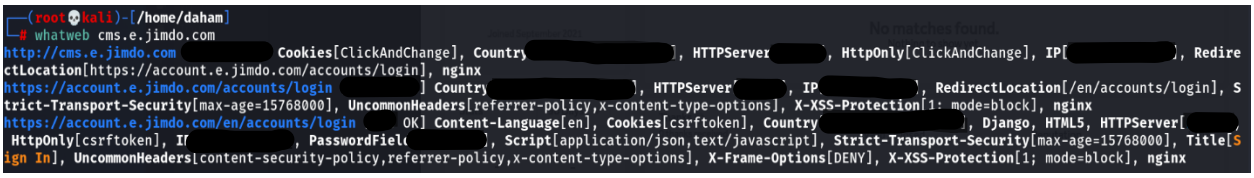


Fig. 21. whatweb csm.jimdo.com

- Gather information related to csm.s.jimdo.com

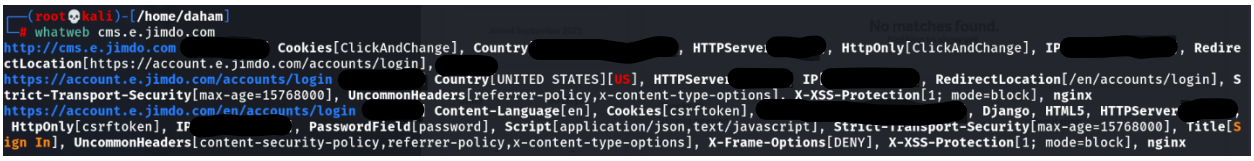


Fig. 22. whatweb cms.e.jimdo.com

- Gather information related to jimdofree.com

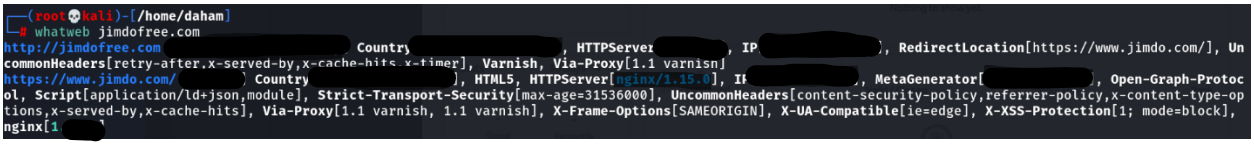


Fig. 23. whatweb jimdofree.com

- Gather information related to dash.e.jimdo.com

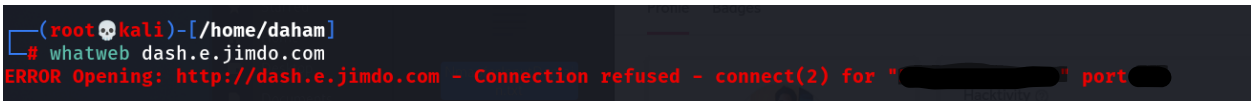


Fig. 24. whatweb dash.e.jimdo.com

- Gather information related to help.jimdo.com

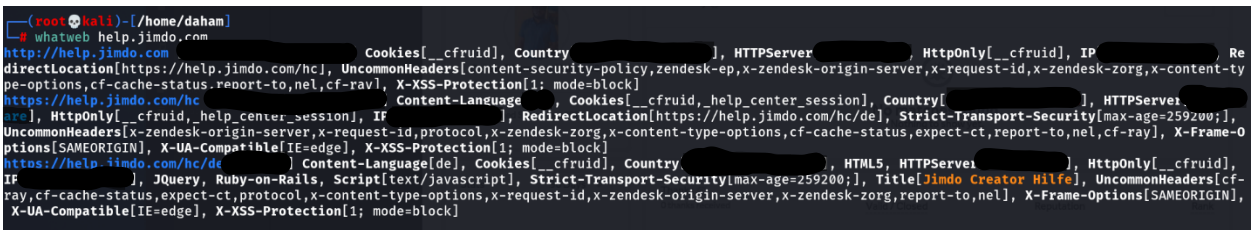


Fig. 25. Whatweb help.jimdo.com

- Other gathered targeted subdomains.

Fig. 26. Other targeted domains

- Gather information about www.jimdo.com in a sorted way with filtering methods.
  - ✓ Scan www.jimdo.com with verbose plugin descriptions (./whatweb -v www.jimdo.com) [4].
  - ✓ An aggressive scan of www.jimdo.com detects the exact version of WordPress (./whatweb -a 3 www.jimdo.com) [4].

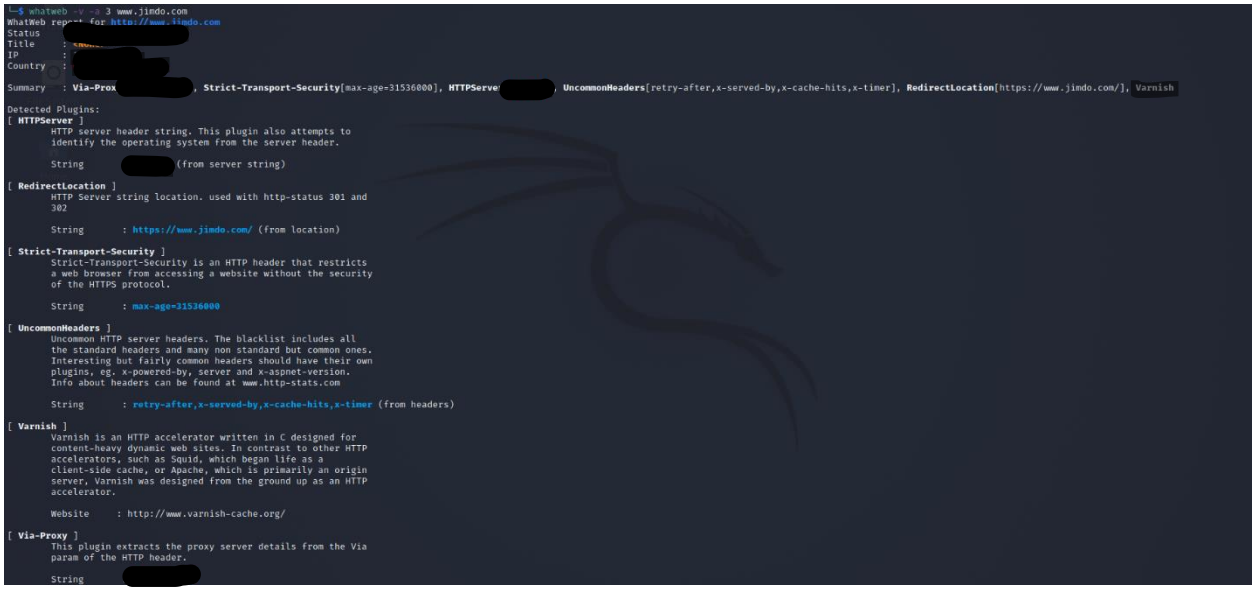


Fig. 27. whatweb -v -a 3 www.jimdo.com

Consider the information I gathered, these are my ideas related to targeted domains.

- ✓ X-Frame-Options HTTP Header is in the DENY use only two domains (Fig.21 & Fig.22). And those are the login pages of this web application. DENY is about the page must not be embedded into another page within an iframe [5]. So, using a frame to hijack the usernames and passwords using clickjacking attacks is protected.
- ✓ X-Frame-Options HTTP Header is in the SAMEORIGIN used by other web domains including the main domain. SAMEORIGIN is about the website can only be embedded in a site that's paired in terms of scheme, hostname, and port [5].
- ✓ X-XSS-Protection HTTP Header is in the 1; mode=block use by all domains. Using X-Frame-Options HTTP Header to detect the cross-site scripting attack. And using 1; mode=block to enable the filter and completely blocks the page [6].
- ✓ According to fig. 27, this domain uses Nginx to version [redacted] as the HTTP server software. This version is older, and it might be vulnerable to exploiting the target domain.
- ✓ Consider about HTTP Strict Transport Security (HSTS) according to the web application status code, it did not use much secure HSTS method to protect their web site. So, 301(Moved Permanently) and 302(Found) methods do vulnerable to attackers. Mainly that the SSL Stripping attacks can happen to this website.

So, the Whatweb tool give me a perfect understanding of targeted domains and the above details are the resent why I am mostly focused on this tool.

➢ Dig

Domain Information Groper (dig) is used for gathering information relevant to Domain Name System (DNS). This command is also the same as the nslookup command. But dig command present the information sorted way than the nslookup command.

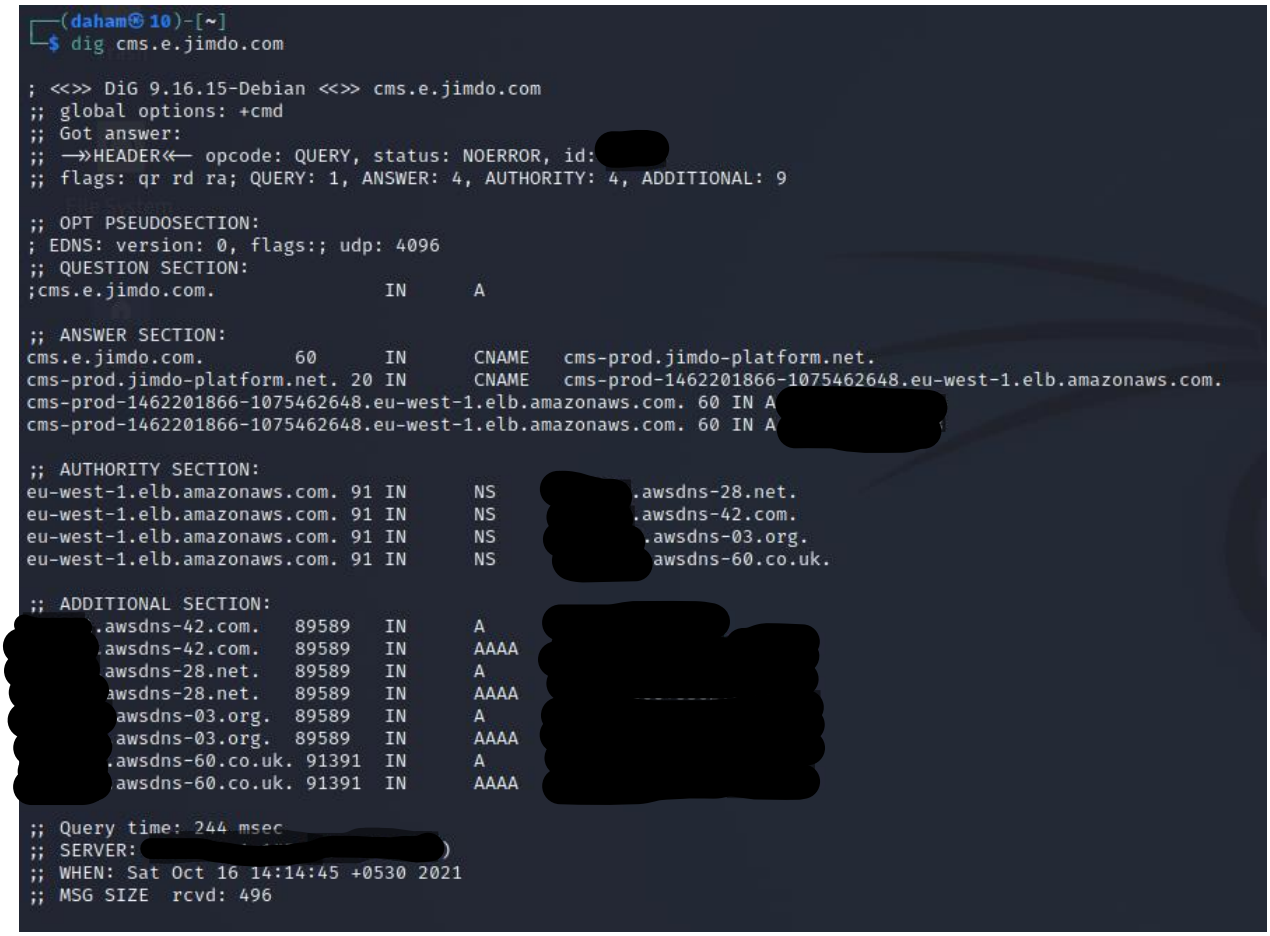


Fig. 28. dig cms.e.jimdo.com

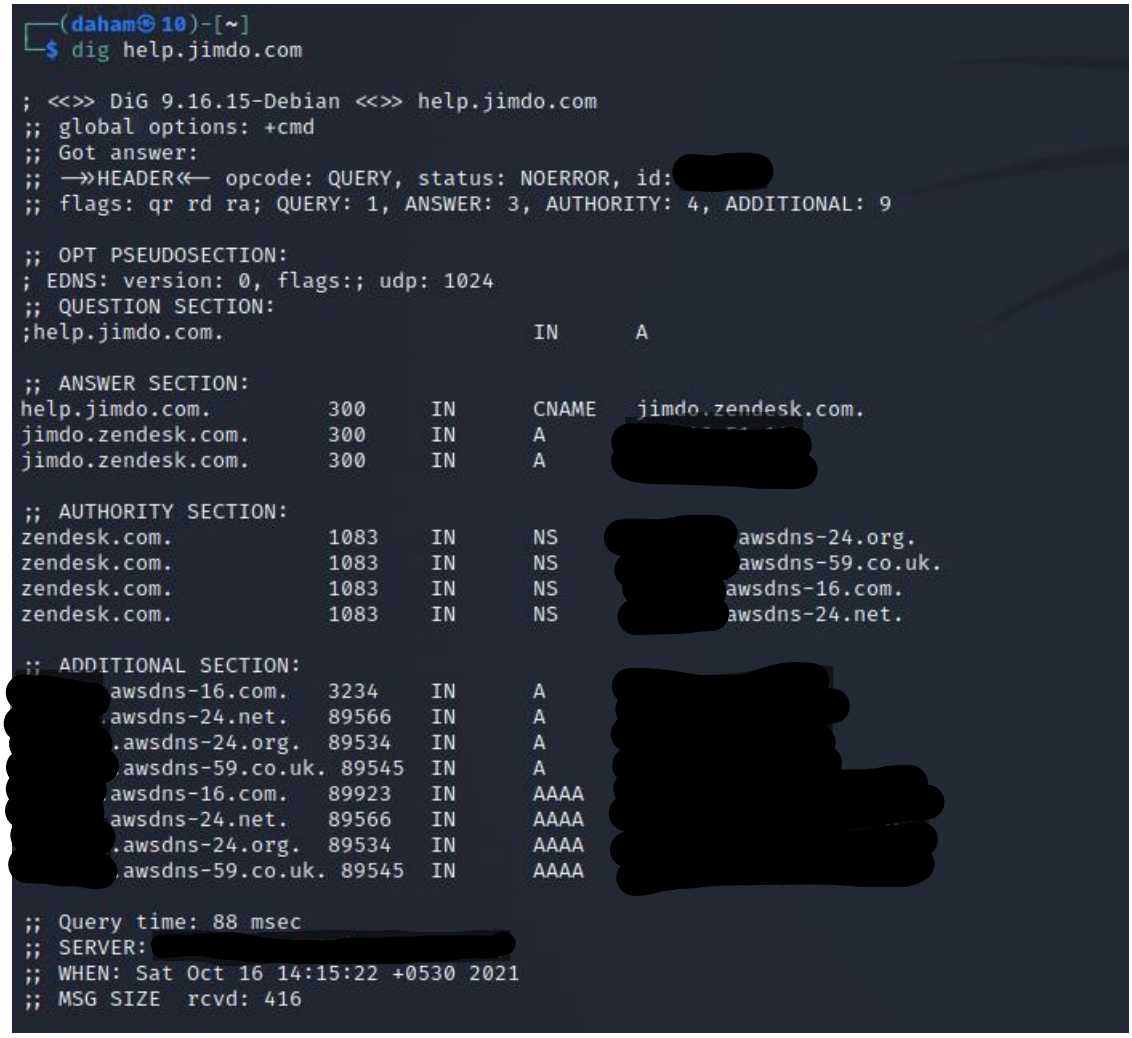


Fig. 29. dig help.jimdo.com

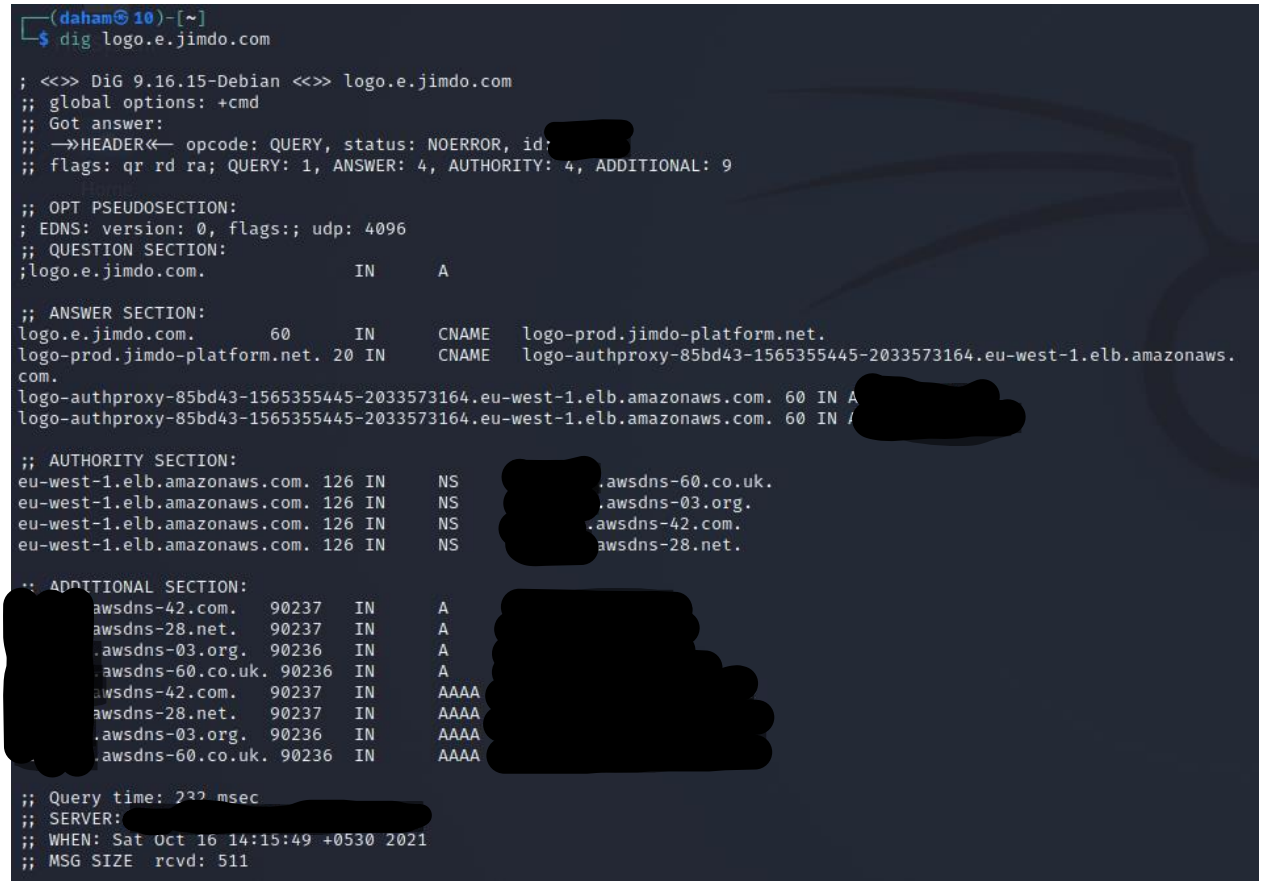


Fig. 30. dig logo.e.jimdo.com

➢ Netcraft

Netcraft (https://sitereport.netcraft.com/) is an online web tool used to gather information related to technologies utilized in web application development. This tool is helping to identify out-of-date software modules used to develop the web application. These outdated software modules can be vulnerable to exploitation.

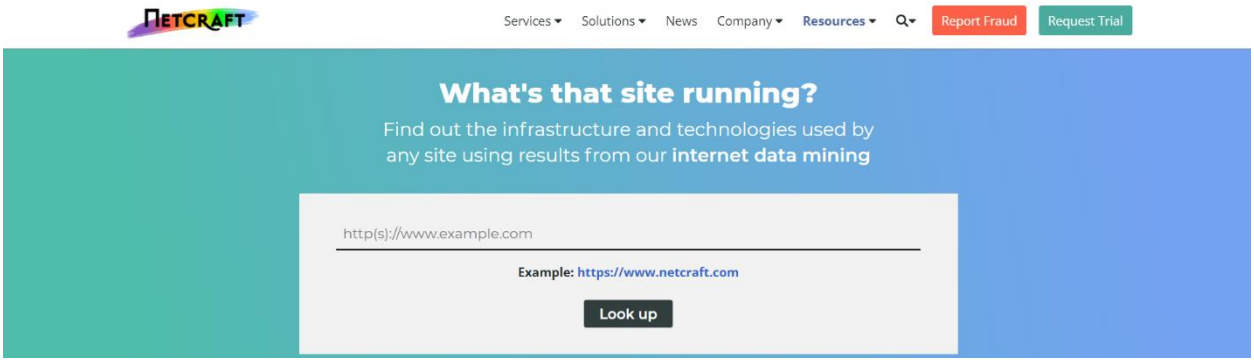


Fig. 31. https://sitereport.netcraft.com/

This is the main interface of the Netcraft tool. We have to enter the domain name to get the details from this tool.

- Gather details about the Network and Background of the targeted domain.

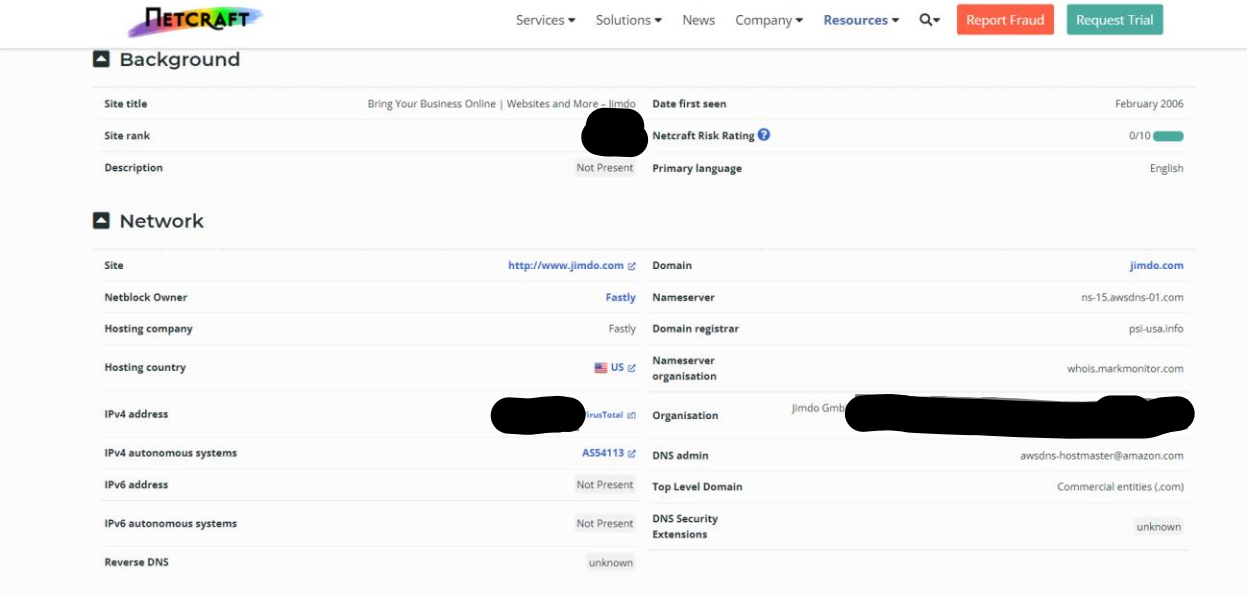


Fig. 32. Details about Network and Background

- Gather information regarded to IP Delegation of the targeted domain.

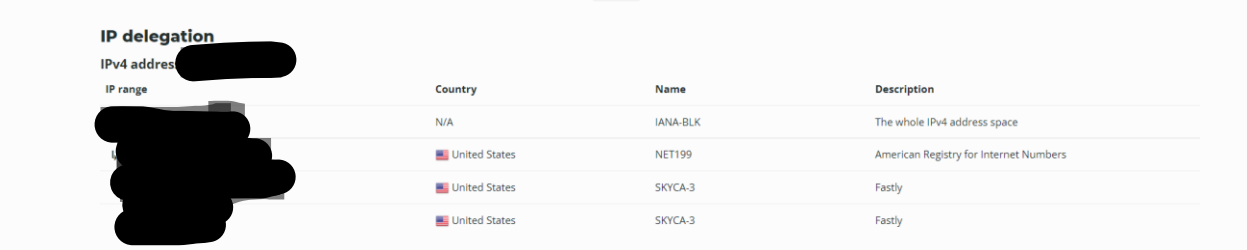
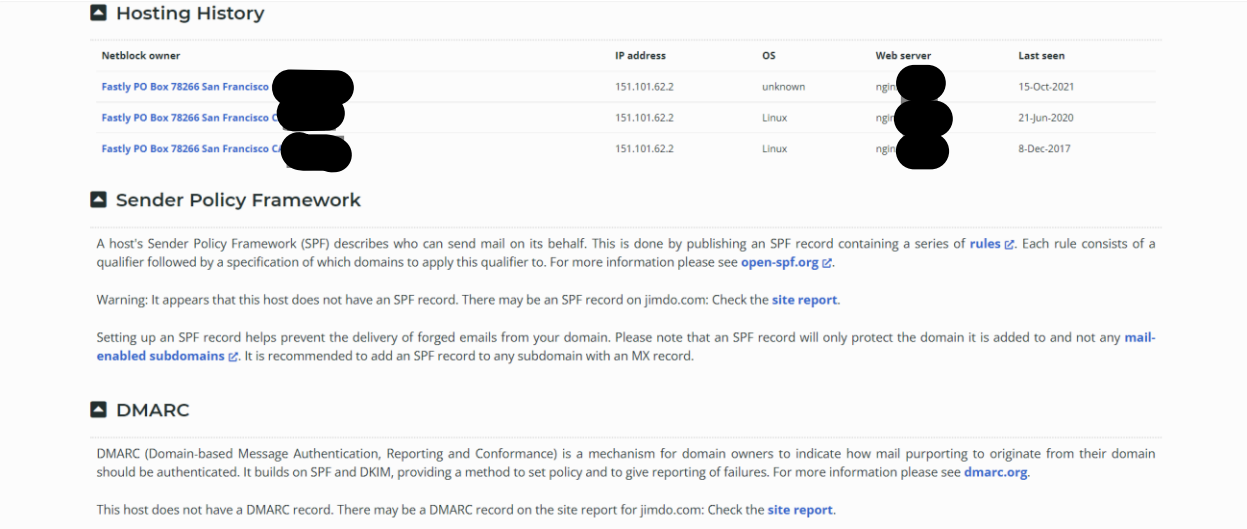


Fig. 33. information regarded to IP Delegation

- Gather the information about Hosting History, Sender Policy Framework, and DMARC of the targeted domain. And we can find the same older HTTP server founded using the whatweb command (Nginx version 1.15.0).



Hosting History

Netblock owner	IP address	OS	Web server	Last seen
Fastly PO Box 78266 San Francisco [redacted]	151.101.62.2	unknown	ngin[redacted]	15-Oct-2021
Fastly PO Box 78266 San Francisco C[redacted]	151.101.62.2	Linux	ngin[redacted]	21-Jun-2020
Fastly PO Box 78266 San Francisco C[redacted]	151.101.62.2	Linux	ngin[redacted]	8-Dec-2017

Sender Policy Framework

A host's Sender Policy Framework (SPF) describes who can send mail on its behalf. This is done by publishing an SPF record containing a series of **rules**. Each rule consists of a qualifier followed by a specification of which domains to apply this qualifier to. For more information please see **open-spf.org**.

Warning: It appears that this host does not have an SPF record. There may be an SPF record on jimdo.com: Check the **site report**.

Setting up an SPF record helps prevent the delivery of forged emails from your domain. Please note that an SPF record will only protect the domain it is added to and not any **mail-enabled subdomains**. It is recommended to add an SPF record to any subdomain with an MX record.

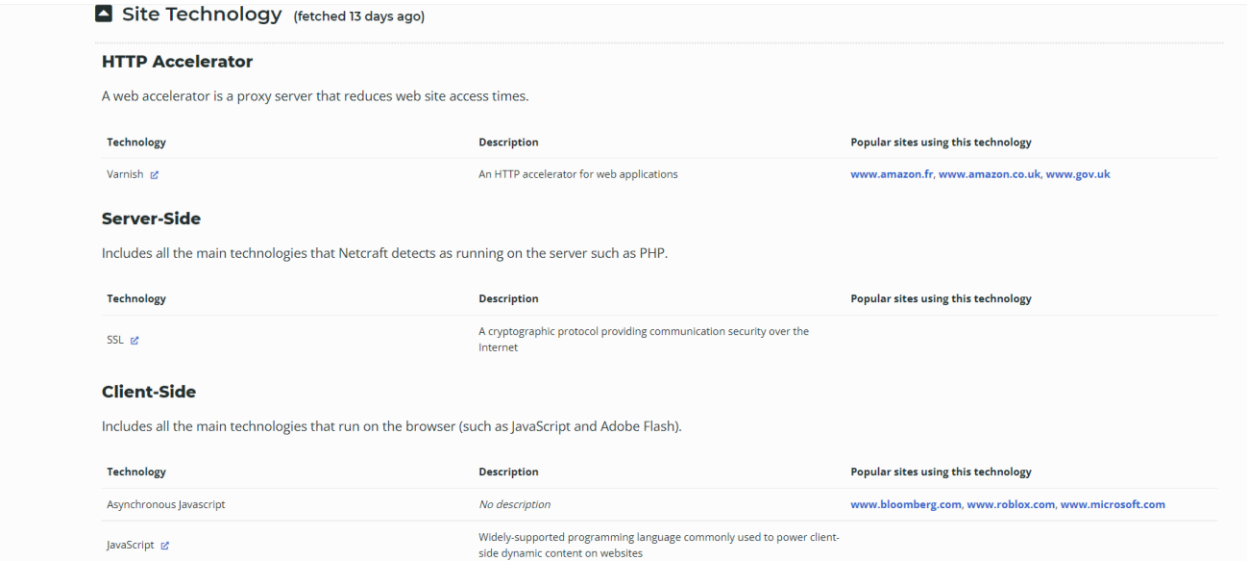
DMARC

DMARC (Domain-based Message Authentication, Reporting and Conformance) is a mechanism for domain owners to indicate how mail purporting to originate from their domain should be authenticated. It builds on SPF and DKIM, providing a method to set policy and to give reporting of failures. For more information please see **dmarc.org**.

This host does not have a DMARC record. There may be a DMARC record on the site report for jimdo.com: Check the **site report**.

Fig. 34. Gather the information about Hosting History, Sender Policy Framework, and DMARC

- Gather the information about Site Technology.



Site Technology (fetched 13 days ago)

HTTP Accelerator

A web accelerator is a proxy server that reduces web site access times.

Technology	Description	Popular sites using this technology
Varnish	An HTTP accelerator for web applications	www.amazon.fr, www.amazon.co.uk, www.gov.uk

Server-Side

Includes all the main technologies that Netcraft detects as running on the server such as PHP.

Technology	Description	Popular sites using this technology
SSL	A cryptographic protocol providing communication security over the Internet	

Client-Side

Includes all the main technologies that run on the browser (such as JavaScript and Adobe Flash).

Technology	Description	Popular sites using this technology
Asynchronous Javascript	*No description*	www.bloomberg.com, www.roblox.com, www.microsoft.com
JavaScript	Widely-supported programming language commonly used to power client-side dynamic content on websites	

Content Delivery Network

A content delivery network or content distribution network (CDN) is a large distributed system of servers deployed in multiple data centers in the Internet. The goal of a CDN is to serve content to end-users with high availability and high performance.

Technology	Description	Popular sites using this technology
Fastly CDN	Content Delivery Network	

RSS Feed

RSS Rich Site Summary is a family of web feed formats used to publish frequently updated works such as blog entries, news headlines, audio, and video in a standardized format.

Technology	Description	Popular sites using this technology
RSS	Standardized web feed format used to publish frequently updated works	www.cnblogs.com, www.heise.de, www.tagesschau.de

Character Encoding

A character encoding system consists of a code that pairs each character from a given repertoire with something else such as a bit pattern, sequence of natural numbers, octets, or electrical pulses in order to facilitate the transmission of data (generally numbers or text) through telecommunication networks or for data storage.

Technology	Description	Popular sites using this technology
UTF8	UCS Transformation Format 8 bit	

NETCRAFT Services Solutions News Company Resources Report Fraud Request Trial

HTTP Compression

HTTP compression is a capability that can be built into web servers and web clients to make better use of available bandwidth, and provide greater transmission speeds between both.

Technology	Description	Popular sites using this technology
Gzip Content Encoding	Gzip HTTP Compression protocol	www.sozcu.com.tr, www.newsit.gr, www.varzesh3.com

Web Browser Targeting

Web browser targeting enables software applications to make use of specific functions of the browser as well as optimizing the application for specific browser versions.

Technology	Description	Popular sites using this technology
Document Compatibility Mode	A meta-tag used in Internet Explorer 8 to enable compatibility mode	www.msn.com, www.amazon.com, www.instagram.com
X-Content-Type-Options	Browser MIME type sniffing is disabled	www.linkedin.com, accounts.google.com, mail.yahoo.com
Strict Transport Security	Web security policy mechanism whereby a web server declares that complying user agents are to interact with it using only secure HTTP connections	twitter.com, outlook.office.com, web.whatsapp.com
X-Frame-Options Same Origin	Do not allow this site to be rendered within an iframe	www.google.com, www.baidu.com, www.twitch.tv
Referrer Policy	Restrict referrer information included in subsequent requests	www.canva.com, www.qwant.com, www.bbc.co.uk
Content Security Policy Report	Detect, mitigate and report attacks in the browser	www.startpage.com, teams.microsoft.com, yandex.ru
X-XSS-Protection Block	Block pages on which cross-site scripting is detected	www.paypal.com, docs.microsoft.com, mail.google.com
Content Security Policy	Detect and mitigate attacks in the browser	

Doctype

A Document Type Declaration, or DOCTYPE, is an instruction that associates a particular SGML or XML document (for example, a webpage) with a Document Type Definition (DTD).

Technology	Description	Popular sites using this technology
HTML5	Latest revision of the HTML standard, the main markup language on the web	

HTML 5

HTML5 is a markup language for structuring and presenting content for the World Wide Web and a core technology of the Internet. It is the fifth revision of the HTML standard.

Technology	Description	Popular sites using this technology
Video Tag	Native browser video playback	www.apple.com, www.pexels.com, www.notion.so
Viewport meta tag	HTML5 tag usually used for mobile optimization	coinmarketcap.com, login.live.com

Fig. 35. Information about Site Technology.

➢ **Whois Lookup**

Whois Lookup (https://whois.domaintools.com/) is an online web tool used to gather information about the hosted company, owner of a target, Server Type, and location of servers.

Fig. 36. *https*://whois.domaintools.com/

- Gather the IP information using the targeted domain IP address.

IP Information for [redacted]

– Quick Stats

IP Location	[redacted] San Francisco Fastly
ASN	[redacted] AS54113 FASTLY, US (registered Oct 04, 2011)
Whois Server	whois.arin.net
IP Address	[redacted]

```
NetRange:       [redacted]
CIDR:           [redacted]
NetName:        [redacted]
NetHandle:      NET-199-232-0-0-1
Parent:         NET199 (NET-199-0-0-0-0)
NetType:        Direct Assignment
OriginAS:
Organization:   Fastly (SKYCA-3)
RegDate:        2016-04-14
Updated:        2016-04-14
Ref:            https://rdap.arin.net/registry/ip/199.232.0.0

OrgName:        Fastly
OrgId:          SKYCA-3
Address:        PO Box 78266
City:           San Francisco
StateProv:      CA
PostalCode:     94107
Country:        US
RegDate:        2011-09-16
Updated:        2021-09-20
Ref:            https://rdap.arin.net/registry/entity/SKYCA-3
```

```
OrgAbuseHandle: ABUSE4771-ARIN
OrgAbuseName:   Abuse Account
OrgAbusePhone:
OrgAbuseEmail:
OrgAbuseRef:    https:                /registry/entity/ABUSE4771-ARIN

OrgNOCHandle: FNO19-ARIN
OrgNOCName:   Fastly Network Operations
OrgNOCPhone:
OrgNOCEmail:
OrgNOCRef:    https                   registry/entity/FNO19-ARIN

OrgTechHandle: FRA19-ARIN
OrgTechName:   Fastly RIR Administrator
OrgTechPhone:
OrgTechEmail:
OrgTechRef:    https:                /registry/entity/FRA19-ARIN
```

Fig. 37. Gather the IP information

2. Active information gathering tools

➢ Nmap

Nmap is a tool used to recognize the state of ports, the host is up and running or not, and much other useful information can gather using this tool. Nmap tool also can be used to scan vulnerabilities inside the targeted domain. But now I use this tool only to gather information about the open ports or those ports are filtered, closed, or unfiltered. So, using the Nmap tool to execute SYN scan to gather the details about the open port of the targeted domains.

- Gather open port information about the www.jimdo.com web domain.

```
└─$ sudo nmap -sS www.jimdo.com -iL NmapforInputDomain.txt -oN Jimdo.txt
Starting Nmap 7.91 ( https://nmap.org ) at 2021-10-14 14:10 +0530
Nmap scan report for www.jimdo.com
Host is up (0.044s latency).
Not shown: 997 filtered ports
PORT    STATE SERVICE
    cp  open  smtp
    tcp  open  http
    tcp open  https
```

Fig. 38. Open ports of the www.jimdo.com

- Gather open port information about the cms.jimdo.com web domain.

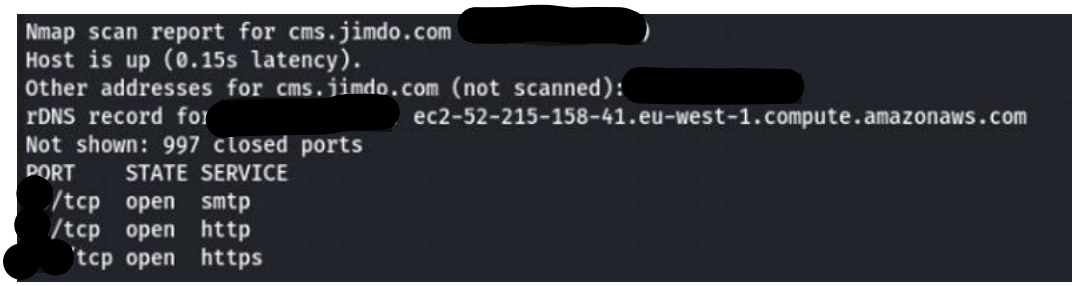


Fig. 39. Open ports of the cms.jimdo.com

- Gather open port information about the cms.e.jimdo.com web domain.

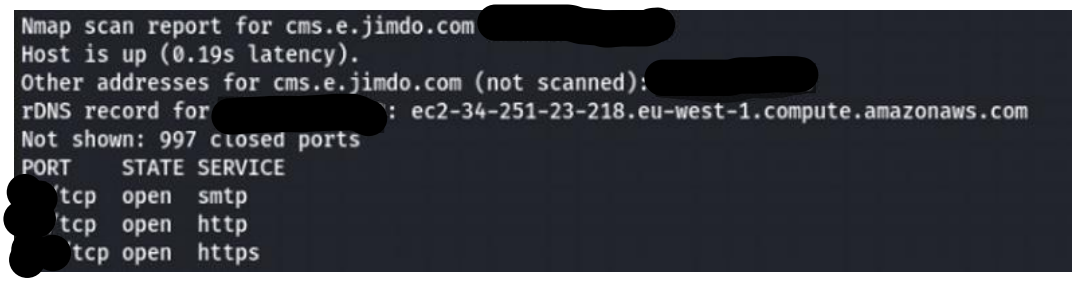


Fig. 40. Open ports of the cms.e.jimdo.com

- Gather open port information about the jimdofree.com web domain.

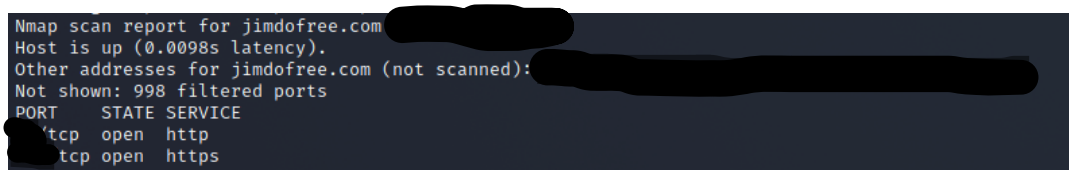


Fig. 41. Open ports of the jimdofree.com

- Gather open port information about the help.jimdo.com web domain. There are some additional open pots.
  - ✓[redacted]tcp open http-proxy
  - ✓[redacted]/tcp open https-alt

Fig. 42. Open ports of the hlpe.jimdo.com

- Gather open port information about the jimdo.design web domain.

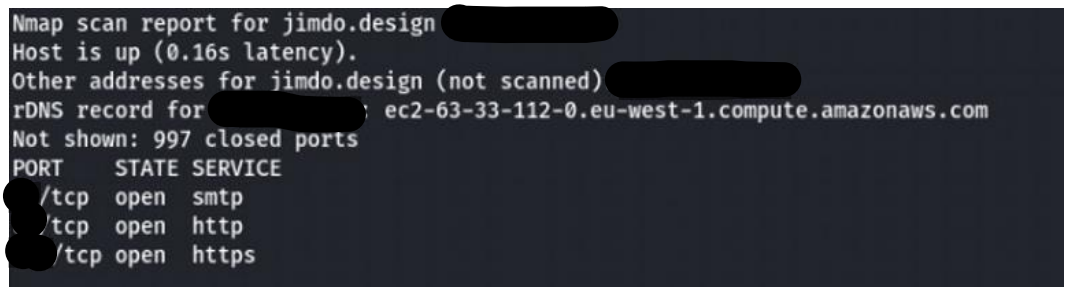


Fig. 43. Open ports of the jimdo.design

- Gather open port information about the logo.e.jimdo.com web domain.

Fig. 44. Open ports of the logo.e.jimdo.com

➢ Dmitry

Dmitry is a collection of information-gathering tools. Because of that, this tool is a combination or package of tools. Using this tool, we can gather details related to Whois lookup web tool information, Netcraft information, and open port details. Because this tool gathers information about open ports, Dmitry is an Active information gathering tool.

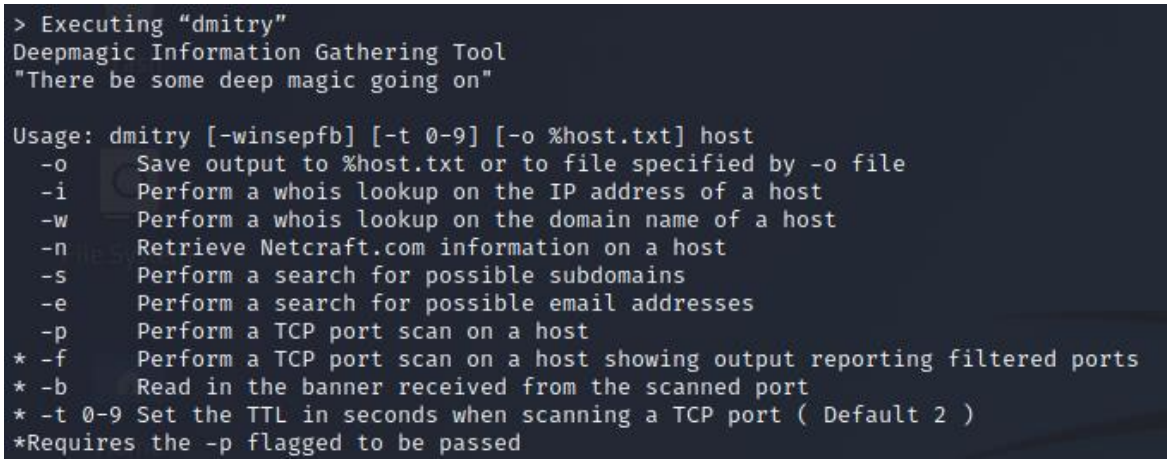
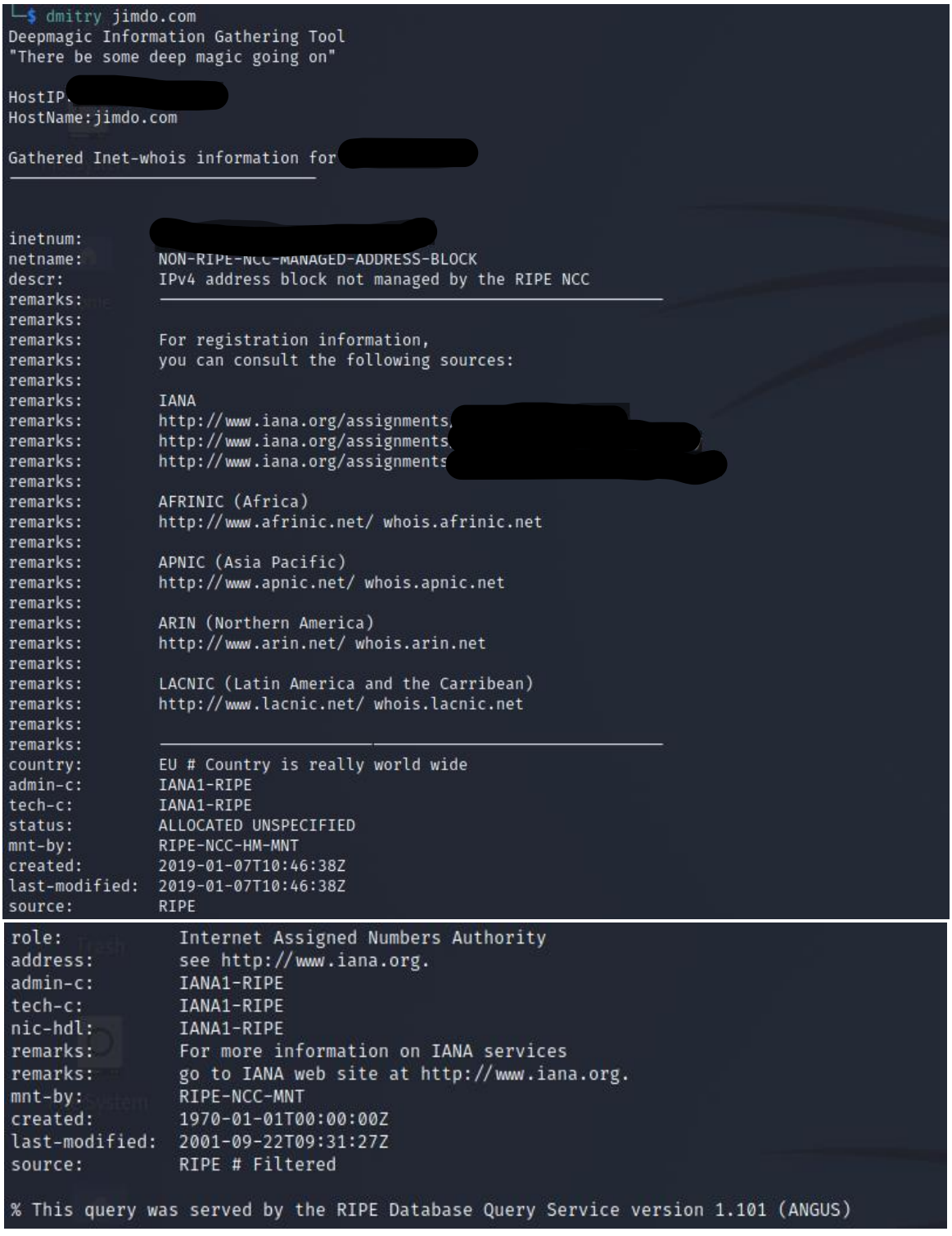


Fig. 45. Dmitry filtering commands

- Gathering Information related to Inet-whois according to Jimdo domain IP address.



```
└─$ dmitry jimdo.com
Deepmagic Information Gathering Tool
"There be some deep magic going on"

HostIP:[redacted]
HostName:jimdo.com

Gathered Inet-whois information for[redacted]
---------------------------------


inetnum:        [redacted]
netname:        NON-RIPE-NCC-MANAGED-ADDRESS-BLOCK
descr:          IPv4 address block not managed by the RIPE NCC
remarks:        ----------------------------------------------------------
remarks:
remarks:        For registration information,
remarks:        you can consult the following sources:
remarks:
remarks:        IANA
remarks:        http://www.iana.org/assignments[redacted]
remarks:        http://www.iana.org/assignments[redacted]
remarks:        http://www.iana.org/assignments[redacted]
remarks:
remarks:        AFRINIC (Africa)
remarks:        http://www.afrinic.net/ whois.afrinic.net
remarks:
remarks:        APNIC (Asia Pacific)
remarks:        http://www.apnic.net/ whois.apnic.net
remarks:
remarks:        ARIN (Northern America)
remarks:        http://www.arin.net/ whois.arin.net
remarks:
remarks:        LACNIC (Latin America and the Carribean)
remarks:        http://www.lacnic.net/ whois.lacnic.net
remarks:
remarks:        ----------------------------------------------------------
country:        EU # Country is really world wide
admin-c:        IANA1-RIPE
tech-c:         IANA1-RIPE
status:         ALLOCATED UNSPECIFIED
mnt-by:         RIPE-NCC-HM-MNT
created:        2019-01-07T10:46:38Z
last-modified:  2019-01-07T10:46:38Z
source:         RIPE

role:           Internet Assigned Numbers Authority
address:        see http://www.iana.org.
admin-c:        IANA1-RIPE
tech-c:         IANA1-RIPE
nic-hdl:        IANA1-RIPE
remarks:        For more information on IANA services
remarks:        go to IANA web site at http://www.iana.org.
mnt-by:         RIPE-NCC-MNT
created:        1970-01-01T00:00:00Z
last-modified:  2001-09-22T09:31:27Z
source:         RIPE # Filtered

% This query was served by the RIPE Database Query Service version 1.101 (ANGUS)
```

Fig. 46. Inet-whois information

- Gathering Information related to Netcraft according to Jimdo domain.

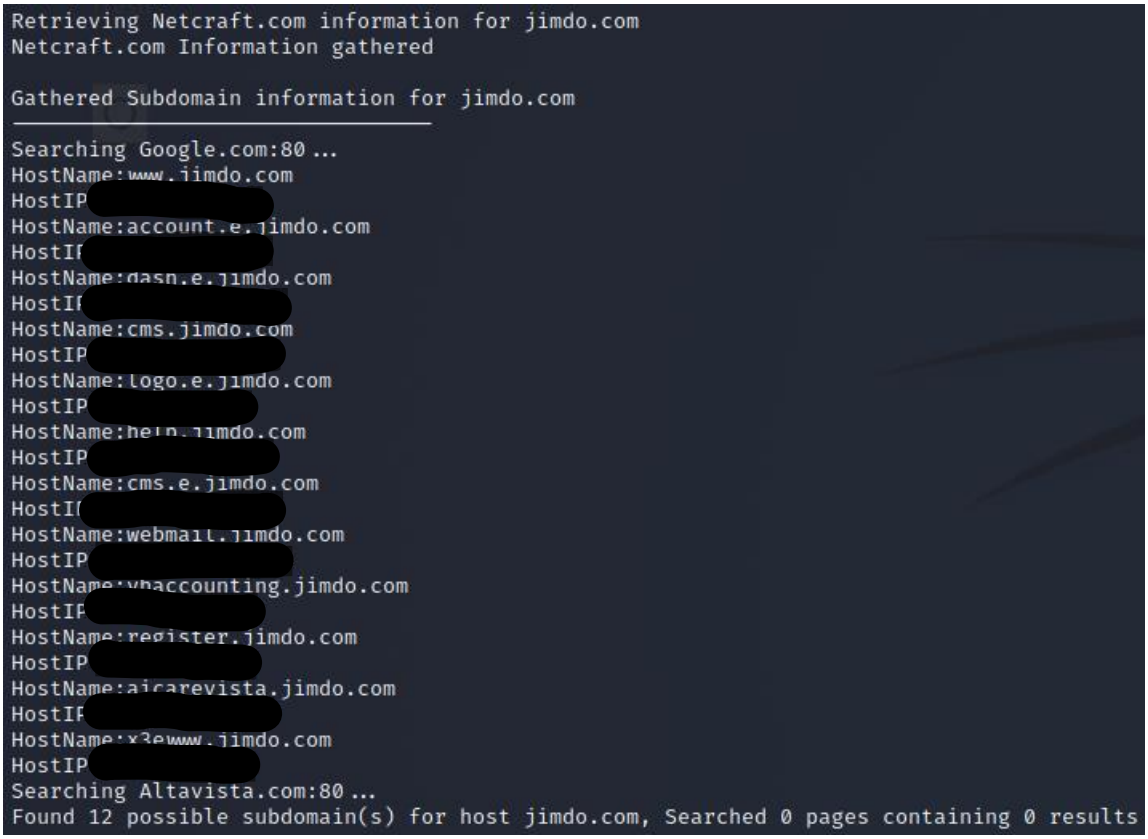


Fig. 47. Netcraft information

- Gathering Information related to E-mail and state of TCP port according to Jimdo domain.

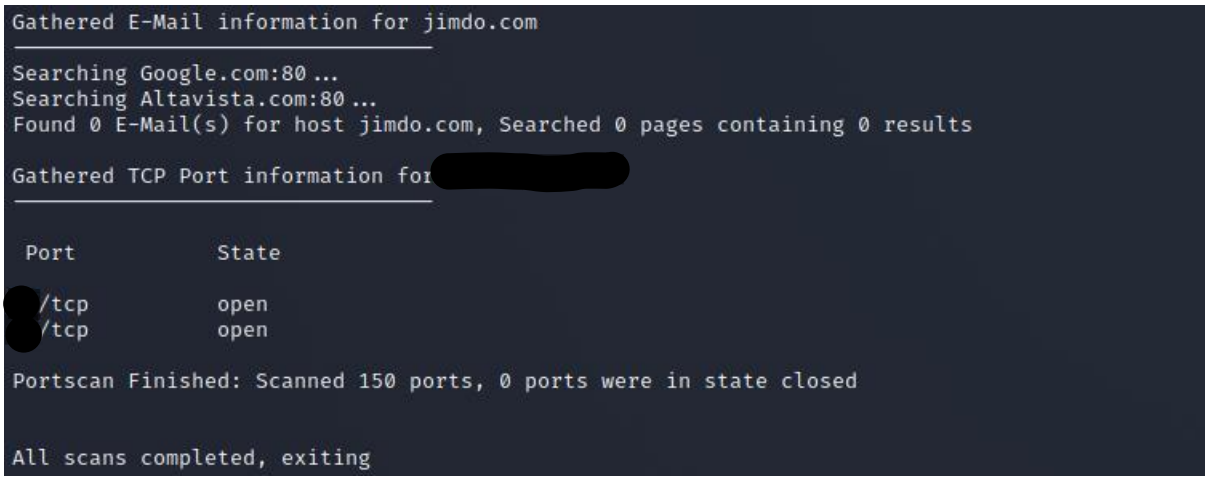


Fig. 48. E-mail and TCP port information

These are the Passive and Active tools I use to gather information about the www.jimdo.com domain.

Planning and Analysis

After the information gathering stage, we need to analyze those details to plan what we focused on next stagers. The planning stage is very essential because vulnerability detection is a time-consuming process and with the plan, we can do vulnerability detection in a targeted way. So, we can save our time and vulnerability detection also can be done in a very efficient manner.

So, after the information gathering process that the collected data can be sorted down according to the technical details such as, Web server details, Application server details, and Database server details. And also, that the state of the ports and the HTTP protection methods are the details focused on to execute the vulnerability scan.

- Technical Details
  - Web server
    - ✓ HTTPS server is Nginx
      - www.jimdo.com (version : [redacted])
      - cms.jimdo.com (version not defined)
      - cms.e.jimdo.com (version not defined)
      - jimdofree.com (version : [redacted])
      - jimdo.design (version : [redacted])
    - ✓ HTTP server is Nginx
      - cms.e.jimdo.com (version not defined)
      - jimdo.design (version : [redacted])
    - ✓ HTTP server is Varnish
      - www.jimdo.com (version not defined)
      - jimdofree.com (version not defined)
    - ✓ HTTP/HTTPS server is Cloudflare
      - help.jimdo.com (version not defined)
  - Application server
    - ✓ Python
      - cms.jimdo.com
      - account.e.jimdo.com
    - ✓ PHP
      - jimdofree.com

- Database server
  - ✓ PostgresSQL
    - cms.jimdo.com
  - ✓ MySQL
    - cms.e.jimdo.com
    - jimdofree.com
    - account.e.jimdo.com

- Open ports details are in the Nmap scan report done in the information gathering stage.
- HTTP security details are in the Wahtweb scan report done in the information gathering stage.

After that select vulnerability scanning tools according to the gathered information and plan the vulnerability scanning according to the information analysis details.

Vulnerability Detection

Vulnerability Detection is a very important stage in Bug Bounty assessment. Because before moving to the penetration testing stage we need to identify vulnerabilities in the particular system. According to Balbix, “Vulnerability scanning is the process of identifying security weaknesses and flaws in the system.” [7].

There are two vulnerability detection methods. They are the automated scanning method and the manual scanning method. I use both of these methods to detect vulnerabilities in the targeted system. Most of the tools can scan vulnerabilities in the system for both of these two methods. Manual scanning is something like a filtered way of scanning and automated scanning is go through all subdomains in the system and scans all vulnerabilities in the system. The automated scanning method is very easy, but it is a time-consuming method. Because that manual scanning is an efficient way of the vulnerability detection method.

So, detecting those vulnerabilities can be done using the Vulnerability Detection tools. There is a lot of open source and paid tools. They are,

- Legion
- Nikto
- Nmap
- Arachni
- Uniscan
- Netsparker
- Nessus
- Owasp Zap

So, I choose that the most suitable vulnerability detection tool according to the gathered information and the usability of those tools. Because some of those tools are not freeware. So, Legion, Nikto, Arachni, Uniscan, Netsparker, Owasp Zap are the tool chosen for use in this Bug Bounty assessment.

➢ Legion

Legion is an open-source network vulnerability detection tool to discover online devices in a network, obtain useful information about targeted systems, and expose targeted system exploits. This tool is a combination of vulnerability detecting tools. Such as Nmap, Whatweb, sslyzer, vulners, SMBenum, and Shodan tools are used in the Legion tool. So, do not need to use Nmap and other tools to detect vulnerabilities in the targeted system.

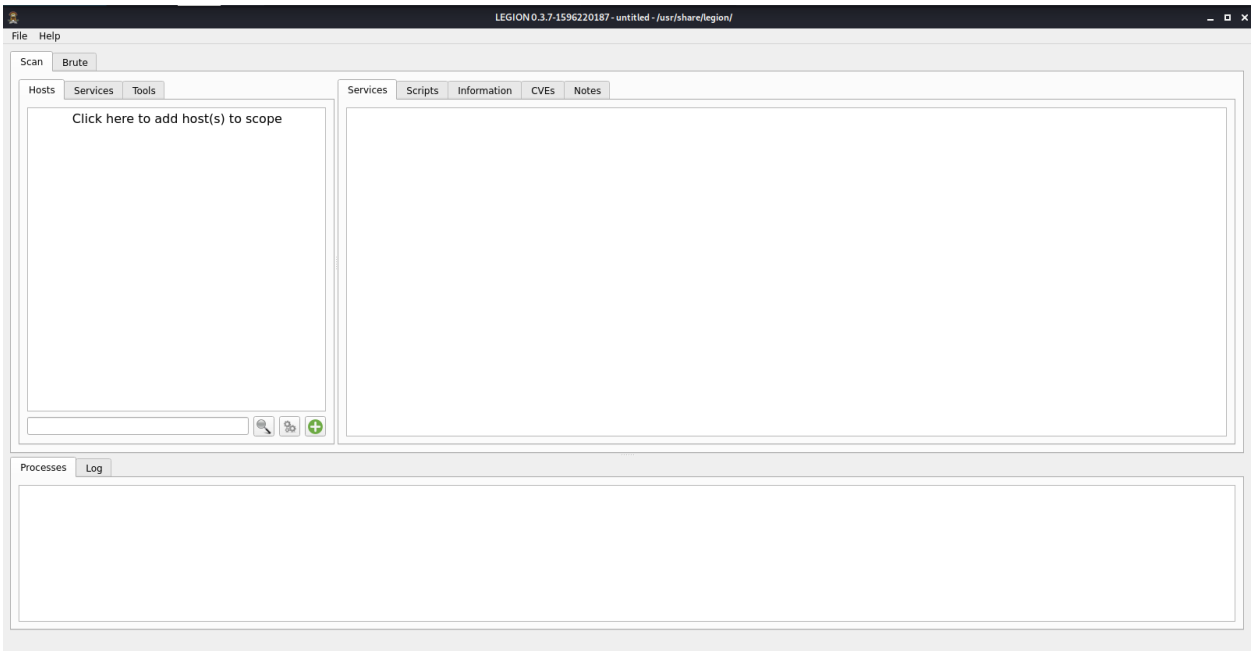


Fig. 49. Dashboard of the Legion Tool

This is the dashboard of the Legion tool. Using the green plus button we can do any type of customization to scan vulnerabilities and provide relevant subdomain links to this tool.

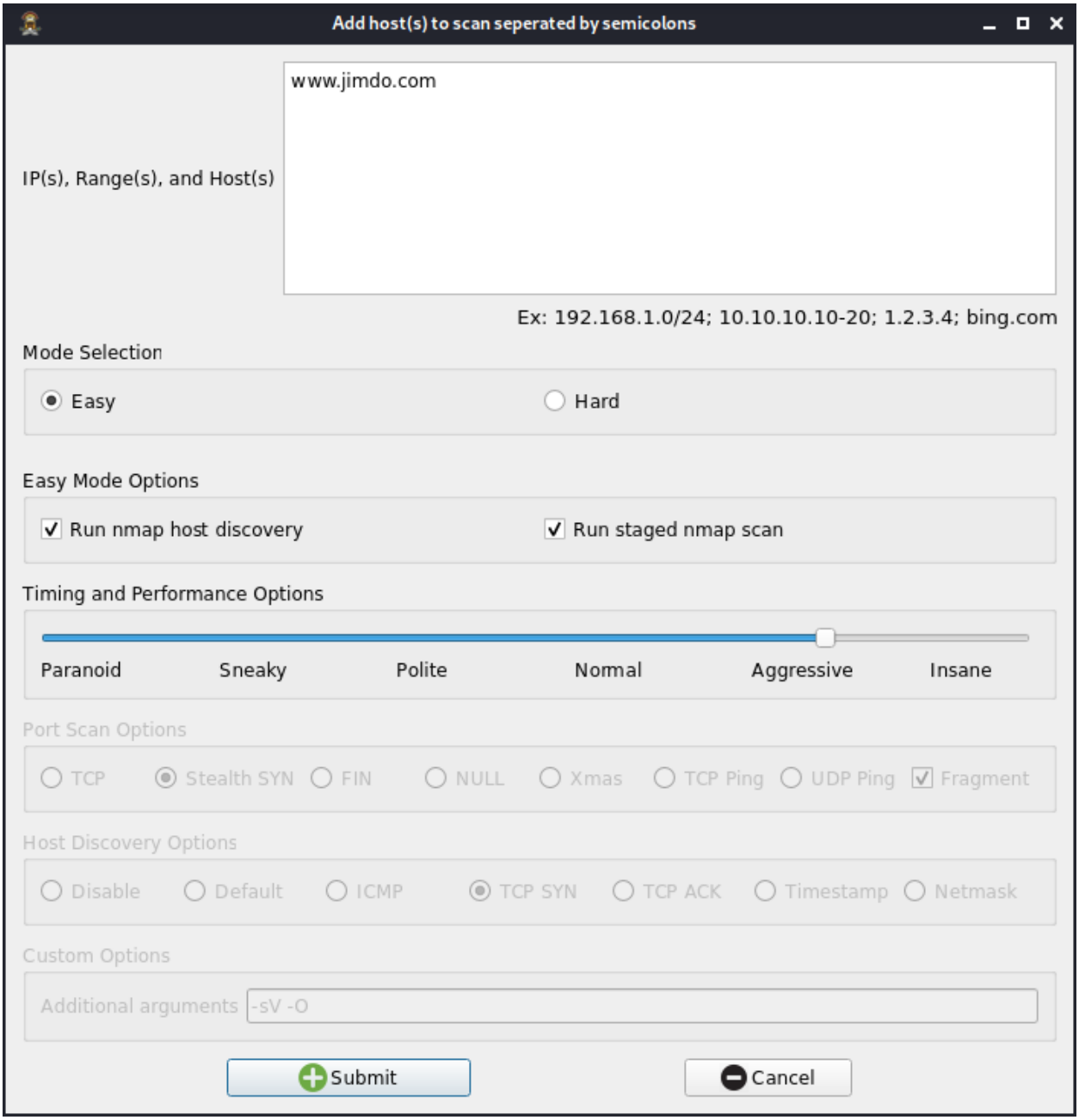


Fig. 50. Customization to scan vulnerabilities

So, I choose automated scan because I need a full scan report of targeted domains and this tool did not take much time to scan. Targeted domain IP address or hostname can use to identify the targeted system and even automated scan this tool provides some Nmap customization methods. After that the customization process is done, we need to submit to get the scanning result from this tool.

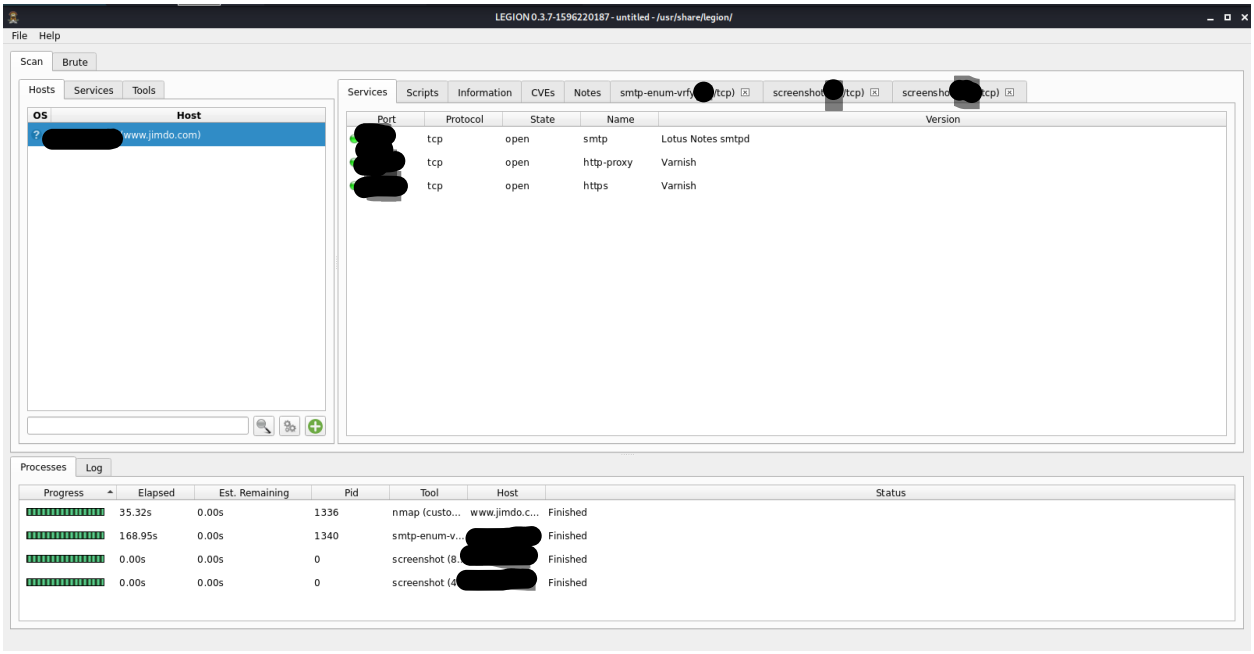


Fig. 51. Port scanning result

Port scanning also gives the same result given through the Nmap scanning done in the information gathering stage. Because that is the same tool used in this scan. [redacted] port(HTTP), [redacted] port(HTTPS), [redacted] port(SMTP) are the open port in the targeted domain. Port 80 can use to exploit vulnerabilities. Because that port is not a protected HTTP port.

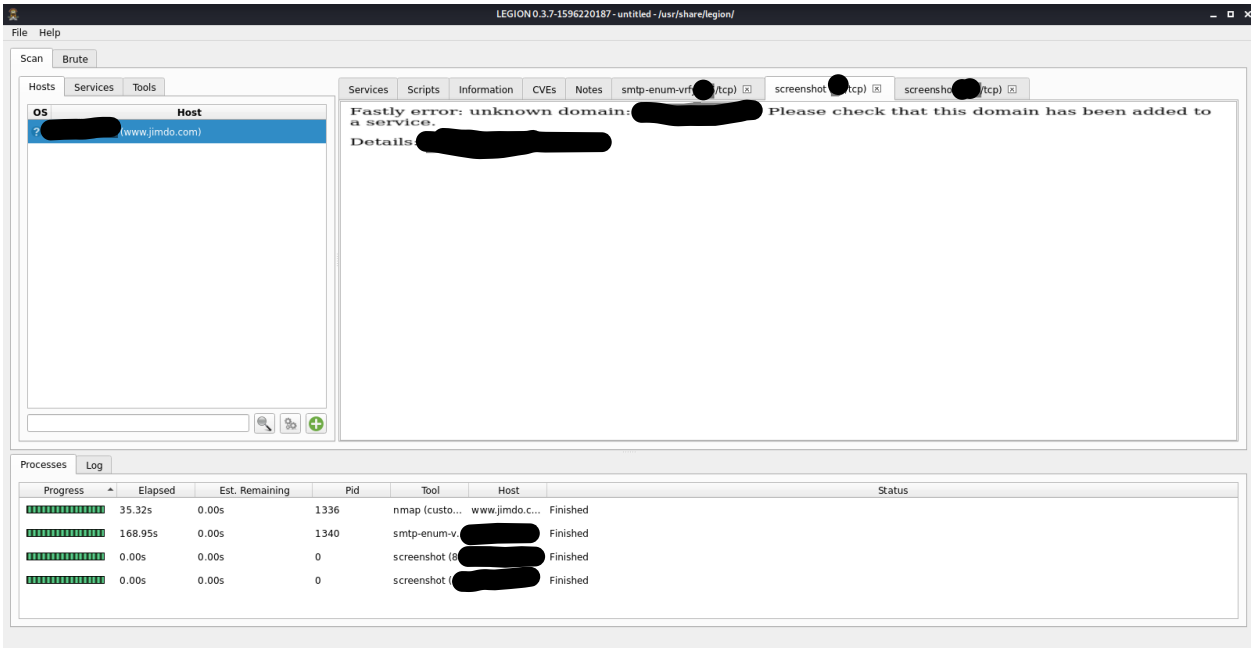


Fig. 52. Screenshot of the port 80

Other domains also give almost identical to the main domain result. And port 80 is an open port that is vulnerable to exploitation.

- **Nikto**

Nikto is a web vulnerability scanner that use to detect vulnerabilities on the targeted domain server. This tool actually detects that the server misconfiguration done by the developers. So, the Nikto tool can find misconfiguring ports in the targeted subdomain and output what type of vulnerabilities have in those subdomains.

- To get a better idea about the Nikto tool we can use the "nikto –help" command.

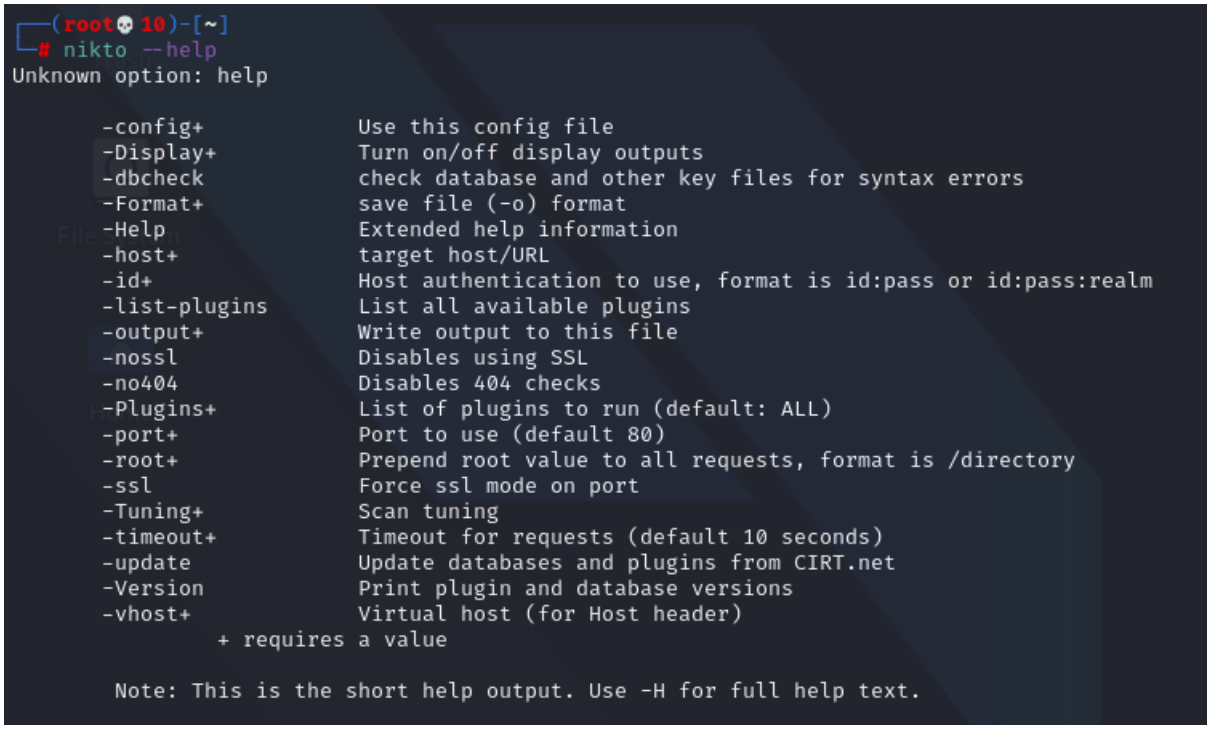


Fig. 53. Nikto --help

So, now we need open ports scan details that were collected during the information gathering stage using the Nmap tool to get the scan result of the Nikto tool. According to the Nmap scan results, I scan all open ports use in all the targeted subdomains. So, we can use to input the hostname to the Nikto tool "-h" command and input the port address "-p" command.

- Scan result of the www.jimdo.com using open port 80

Fig. 54. nikto -h www,Jimdo.com -p

- ✓ X-XSS-Protection is not defined. So, this protection is a must to have, and this website can be vulnerable to the Cross-Site Scripting (XSS) attack.
- ✓ And X-Content-Type-Option header also is not set. According to MDN Web Docs, “The X-Content-Type-Options response HTTP header is a marker used by the server to indicate that the MIME types advertised in the Content-Type headers should be followed and not be changed.” [8] . This also could have the risk of a Cross-Site Scripting (XSS) attack.

- Scan result of the cms.jimdo.com using open port 80. This Jimdo website login web page and the result of the scan are identical to the main domain results. And also, seme vulnerabilities can find in all the selected subdomains.

Fig. 55. nikto -h cms.jimdo.com -p

- Scan result of the cms.jimdo.com using open port 443. Extra protection is there because of the HTTPS use here. But same vulnerabilities can find also in this port.

Fig. 56. nikto -h cms.jimdo.com -p

- Using below commands can use to get the scanning report to a text file.

Fig. 57. nikto -h www.jimdo.com -p -o nikto_jimdo_main -F txt

➢ Arachni

Arachni tool is an open-source web application vulnerability detection tool, and it is a high-performance scanner as well as this is fully automated scanning tool. And the disadvantage is that this tool cannot do any customized way of scanning. Because of that scanning is a very time-consuming process.

Fuzzing, taint-analysis, differential analysis, and timing/delay attacks are the combination of techniques that the Arachni tool use to scan the vulnerabilities in the targeted system.

First, that the Arachni tool needs to install in our operating system. So, we can simply extract the files into our operating system.

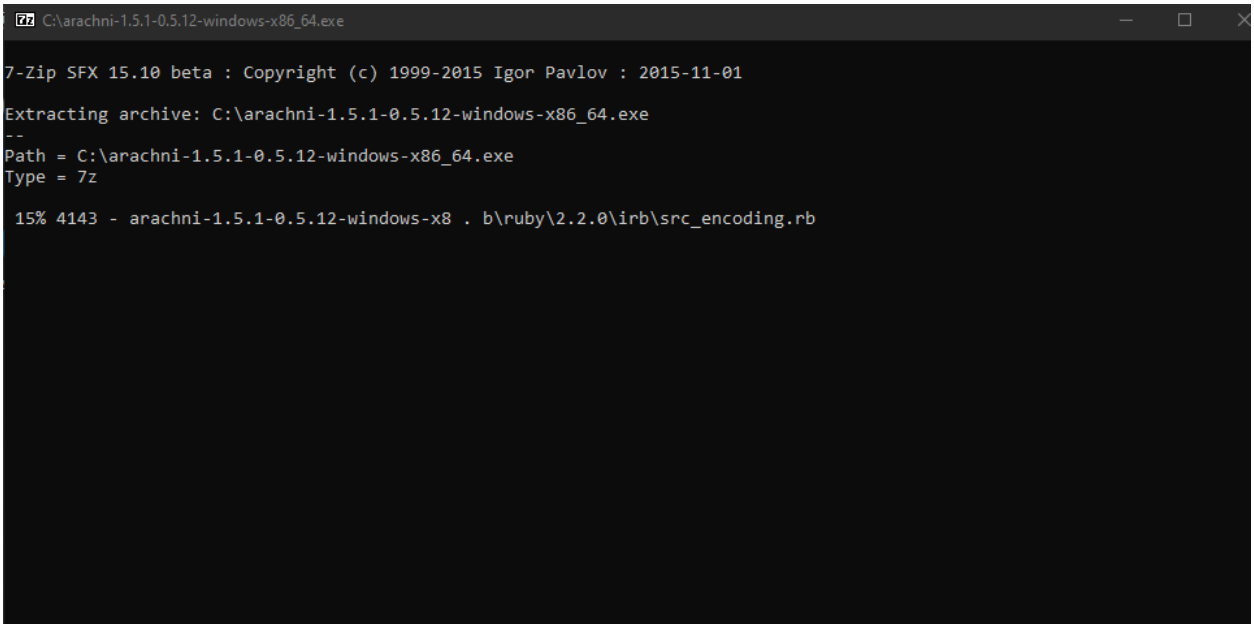


Fig. 58. Extract the files

And we can run this tool by going to the extract as well as move into the bin folder and after that, we need to open “arachni_web.bet” to access the web application link(//localhost:9292) of this tool and run this tool.

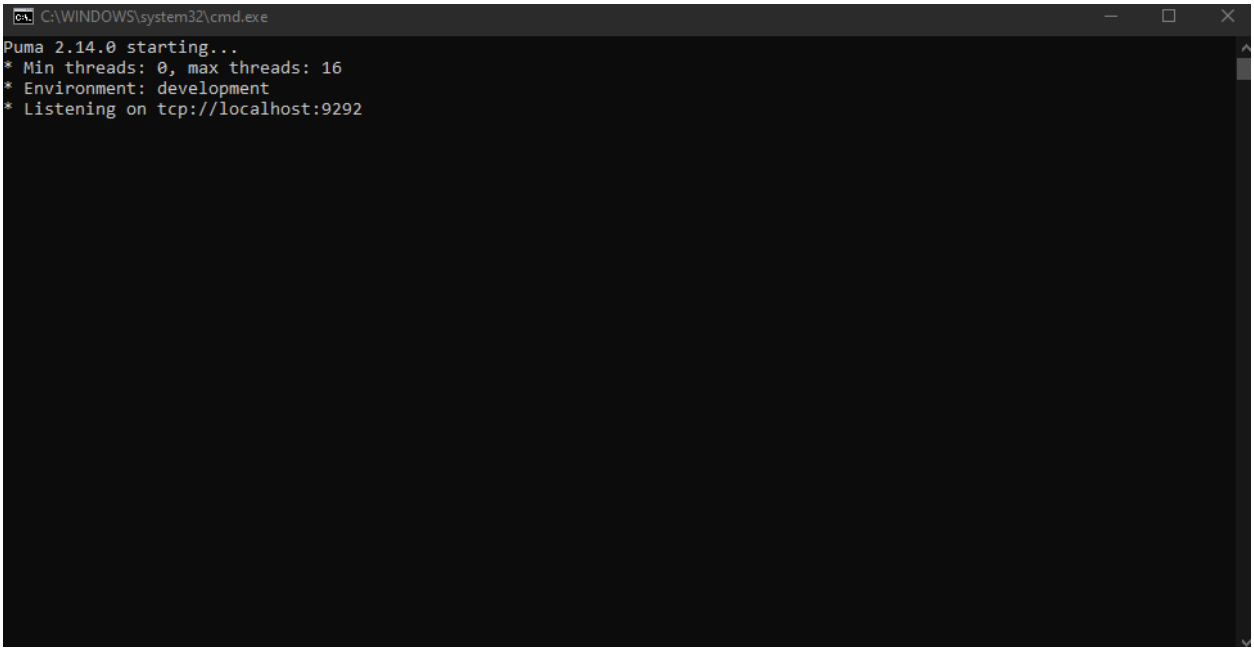


Fig. 59. Access the web application link(//localhost:9292)

After that log in to the Arachni tool by using the provided email(admin@admin.admin) and password(administrator).

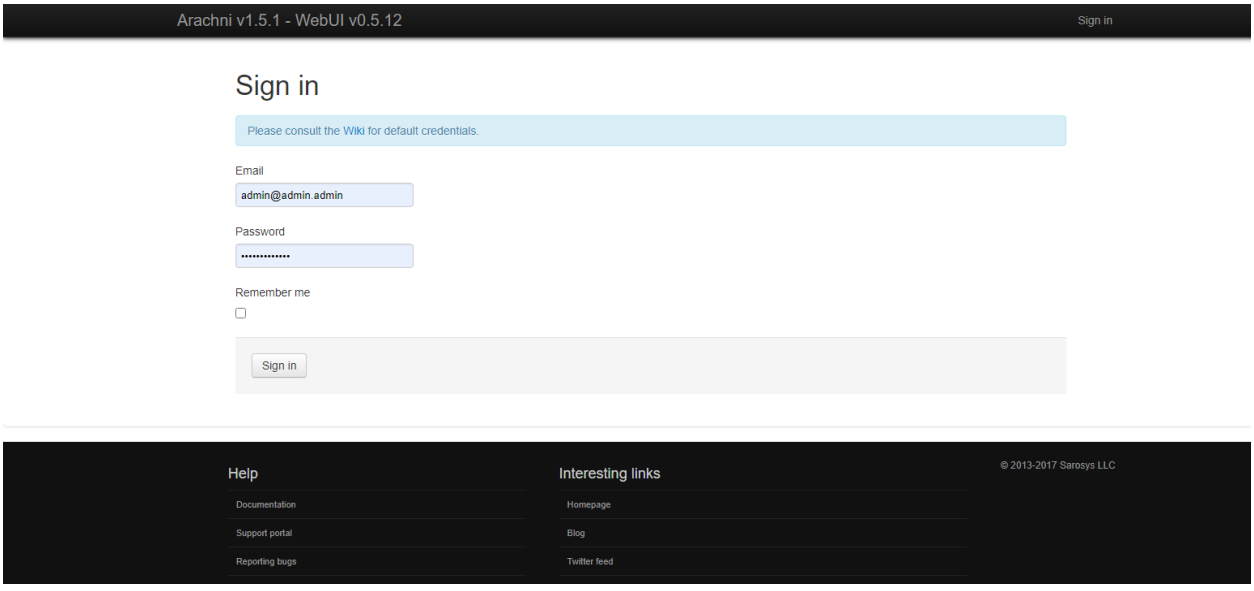


Fig. 60. Login to the Arachni tool

This is the dashboard of the Arachni tool, and they provide details about this tool.

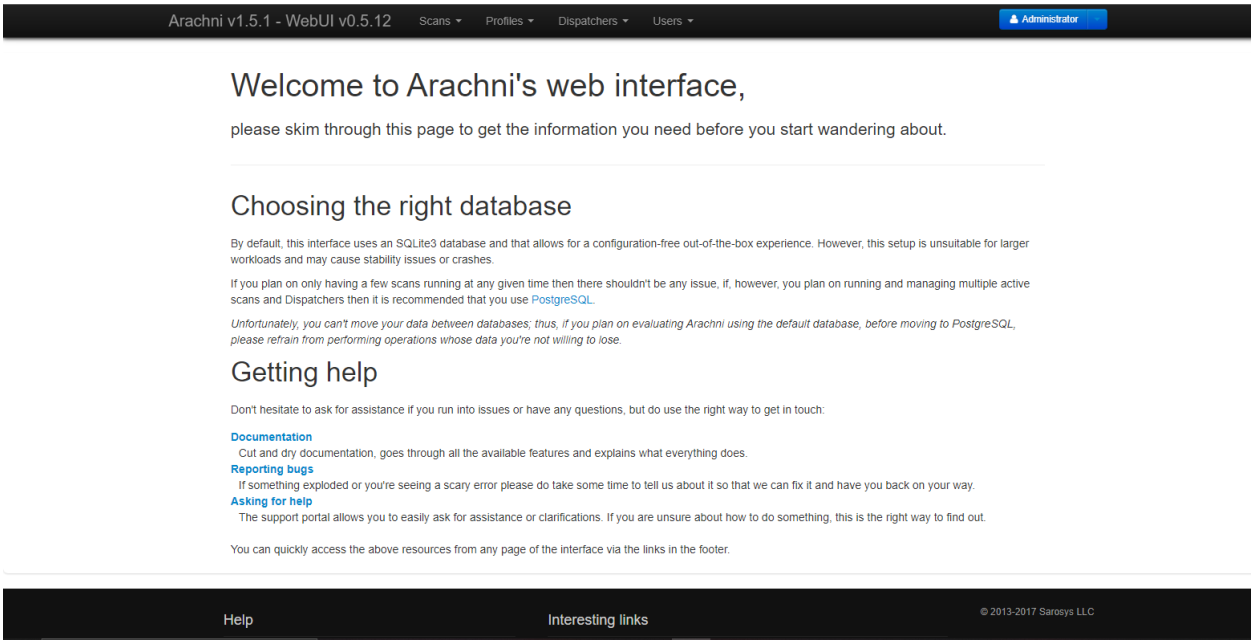


Fig. 61. Dashboard of Arachni tool

Using that the scans option in the status bar and start a new scan. We can provide the relevant domain URL to start the scan.

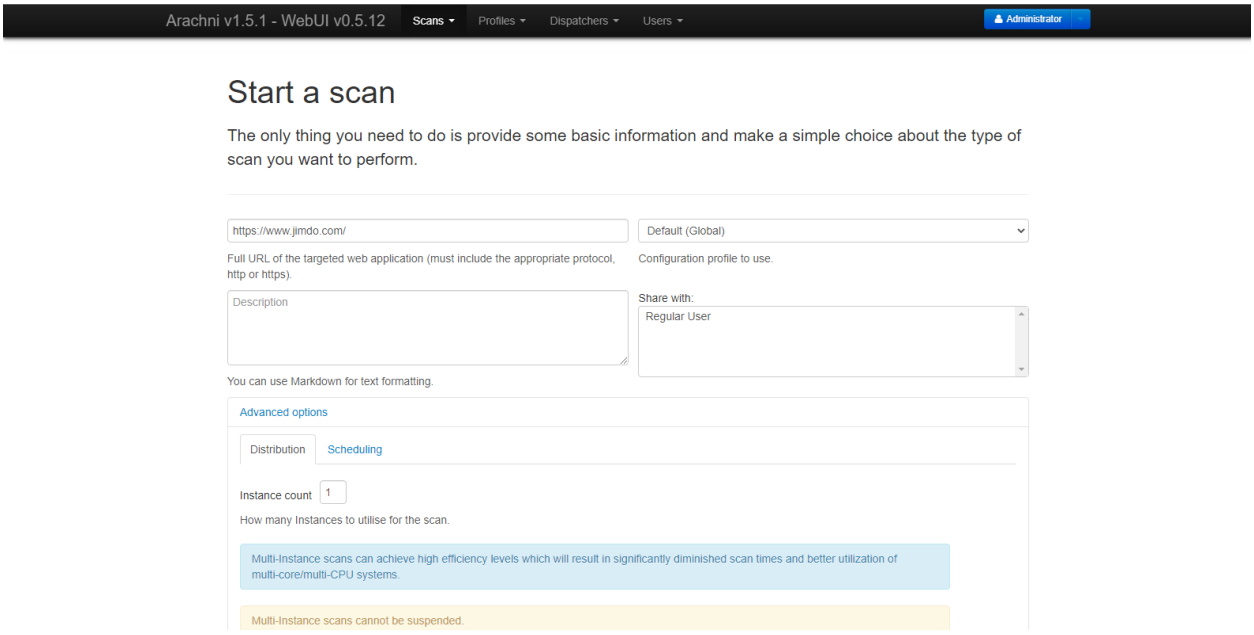


Fig. 62. Scan Dashboard

- Scan the result of the www.jimdo.com domain.

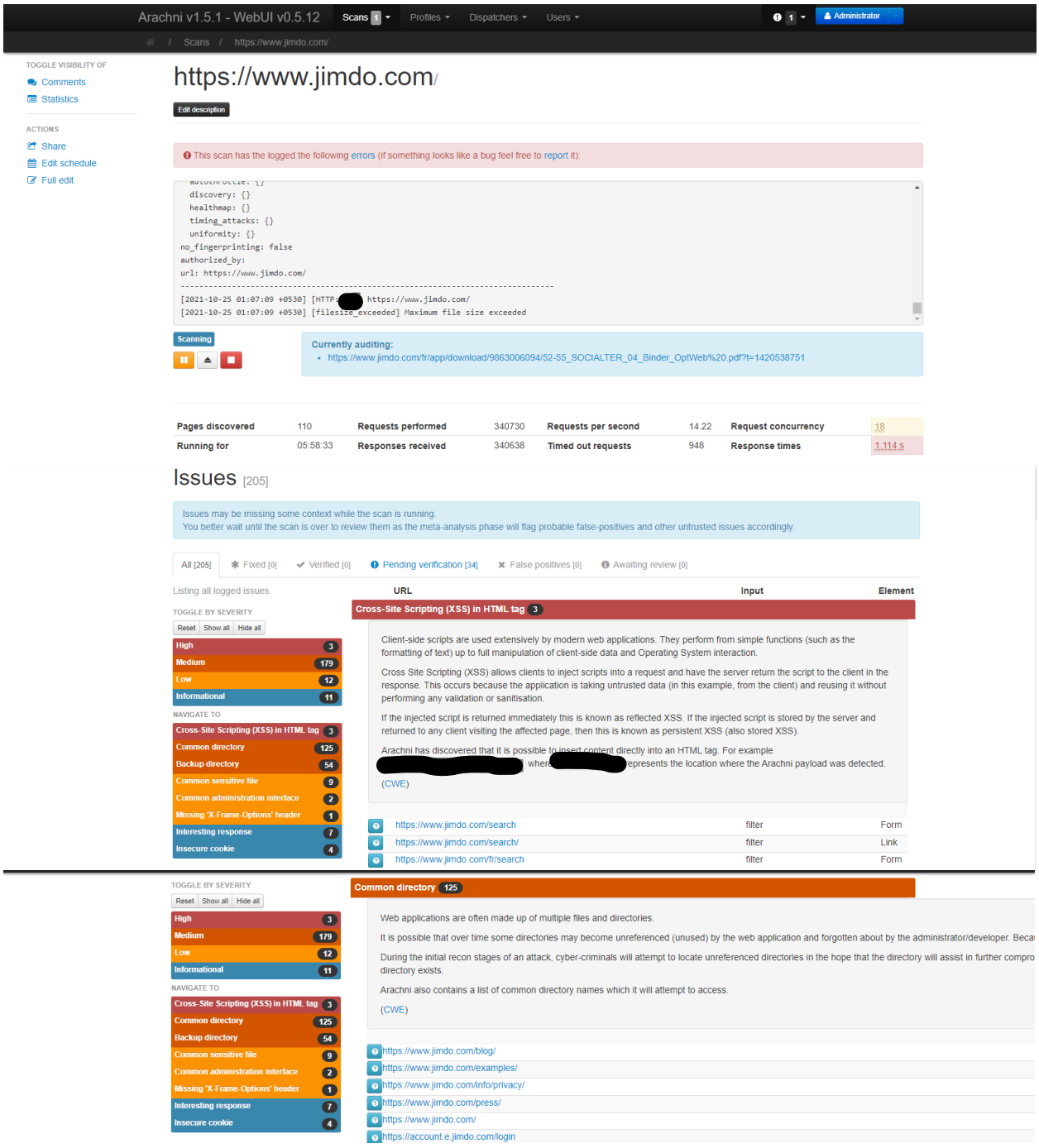


Fig. 63. Scan result of the www.jimdo.com

- ✓ As mentioned before in the Nikto scan result there are High stage vulnerabilities related to Cross-Site Scripting (XSS) attacks. According to the description targeted domain HTML injection type of Cross-Site Scripting (XSS) attack.

So, these are the subdomains we can use to exploit this vulnerability,

- https://www.jimdo[redacted]
- https://www.jimdo[redacted]
- https://www.jimdo[redacted]

○ Scan result of the cms.jimdo.com domain.

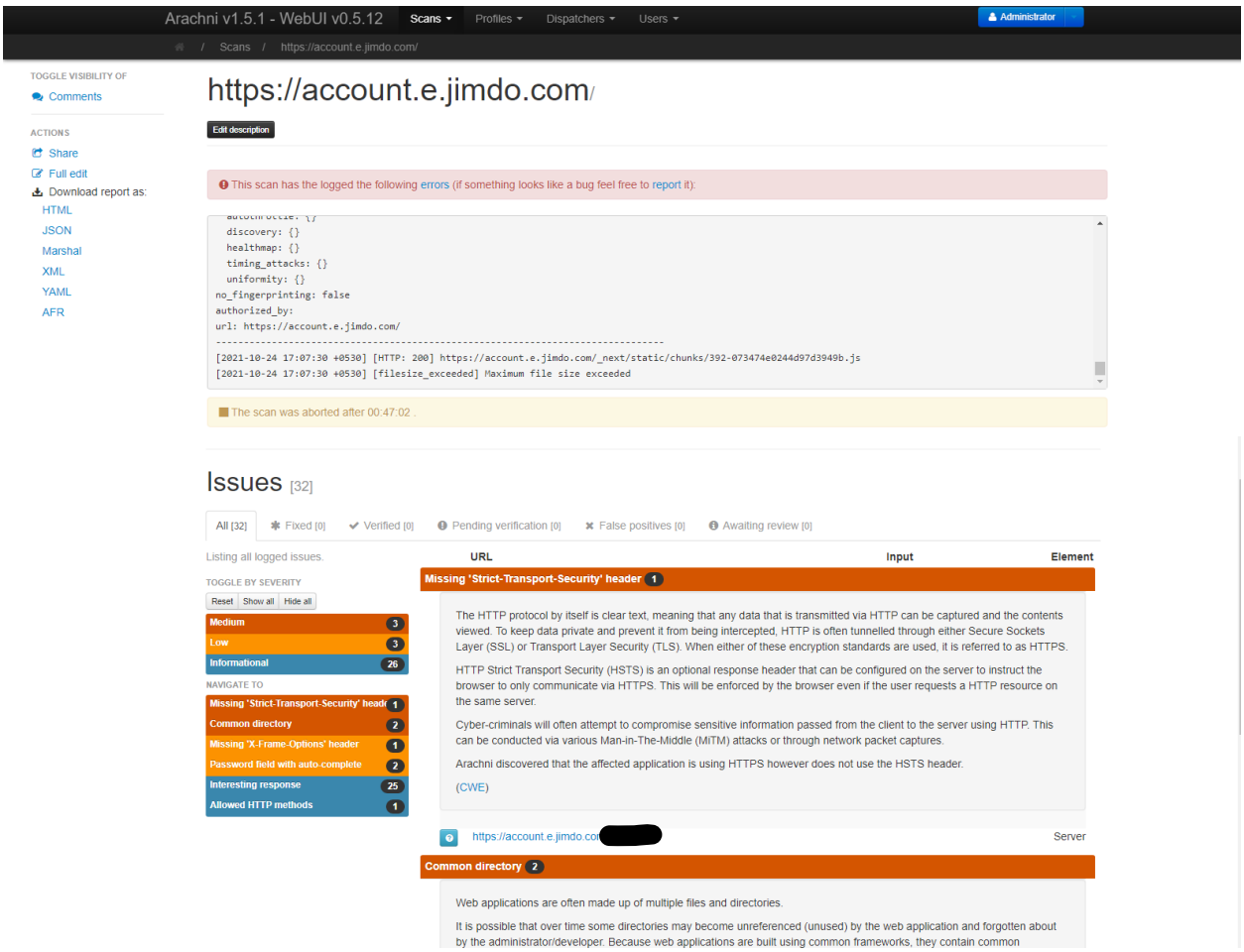


Fig. 64. Scan result of the cms.jimdo.com

✓ As previously mentioned in the Whatweb scan when happening in the information gathering stage. According to the status code, there is no Strict-Transport-Security in those domains, and this can be vulnerable to SSL Stripping attack. So, this is the login subdomain we can use to exploit this vulnerability,

- https://account.e.jimdo.com/en/signup

- Scan result of the jimdo.design domain.

Fig. 65. Scan result of the jimdo.design

✓ This domain and other selected subdomains output the almost same result. There are also X-Frame options in this domain. And that is low profile attack.

- This Arachni tool also provides finalized report to gather information about those vulnerabilities.

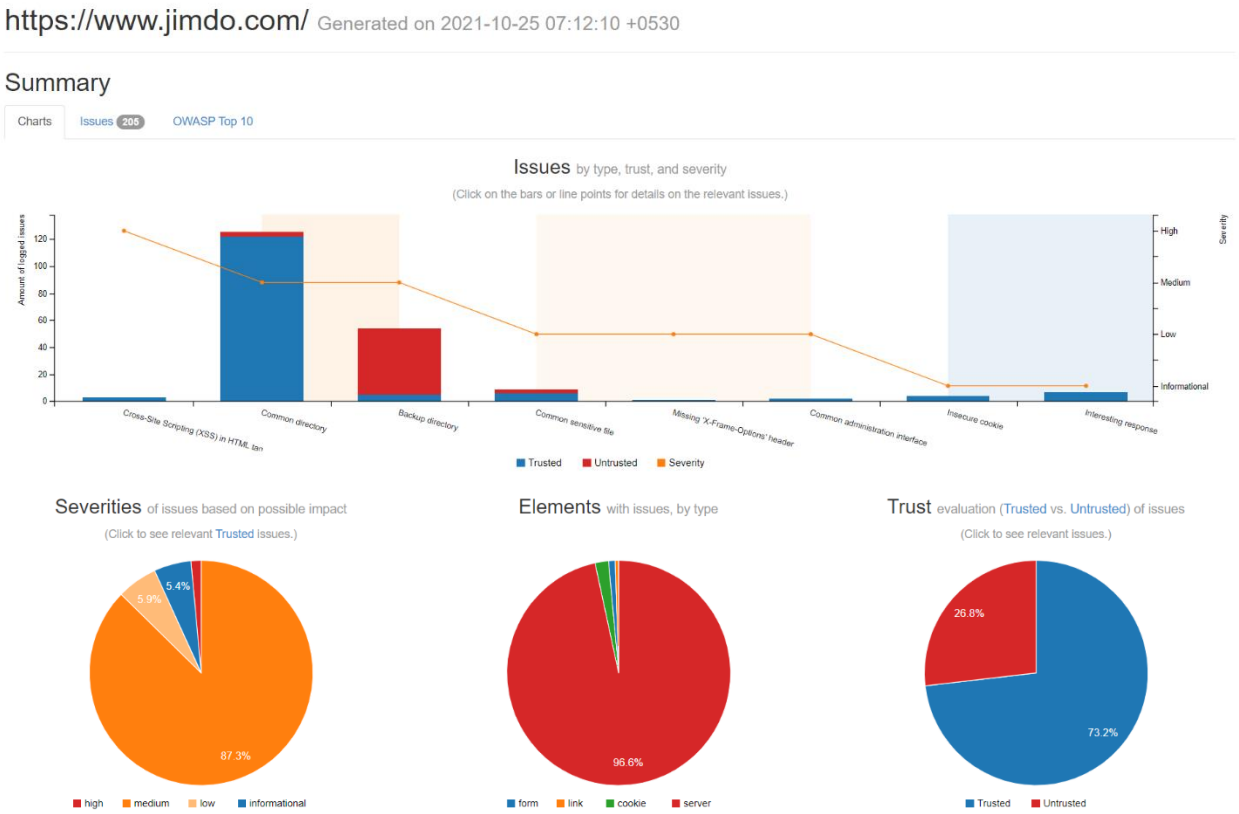


Fig. 66. Scan report of www.jimdo.com

So, according to this chart, we can analyze vulnerabilities perfectly. And we can get a good understanding of how much this domain is vulnerable to attackers.

➢ Uniscan

Uniscan is an open-source vulnerability detection tool that can be used to scan vulnerabilities in the targeted web application, such as, cross-site scripting(XSS), remote file inclusion, web shell vulnerabilities, SQL injection, blind SQL injection, and hidden backdoors. Also, the Uniscan tool is capable to do a Bing and Google search for finding domains on shared IP addresses.

So, this tool is inbuilt in the Kali Linux operating system, and we need to give root permission to access this tool. Uniscan tool can be manually configurable. So, this tool is suitable for the filtered way of scanning.

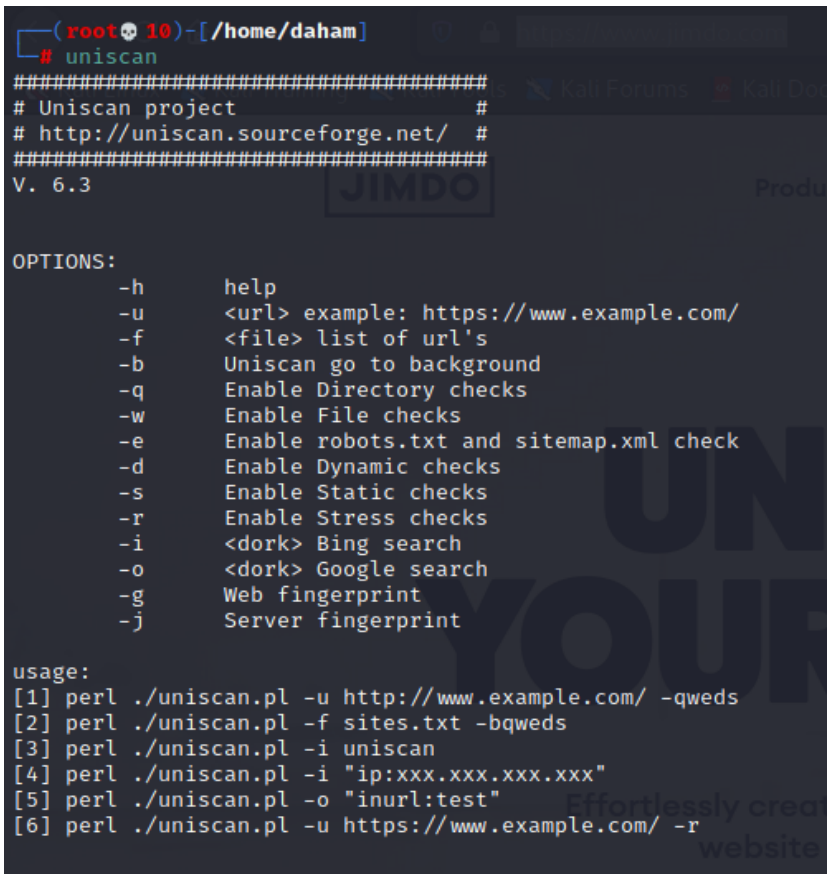
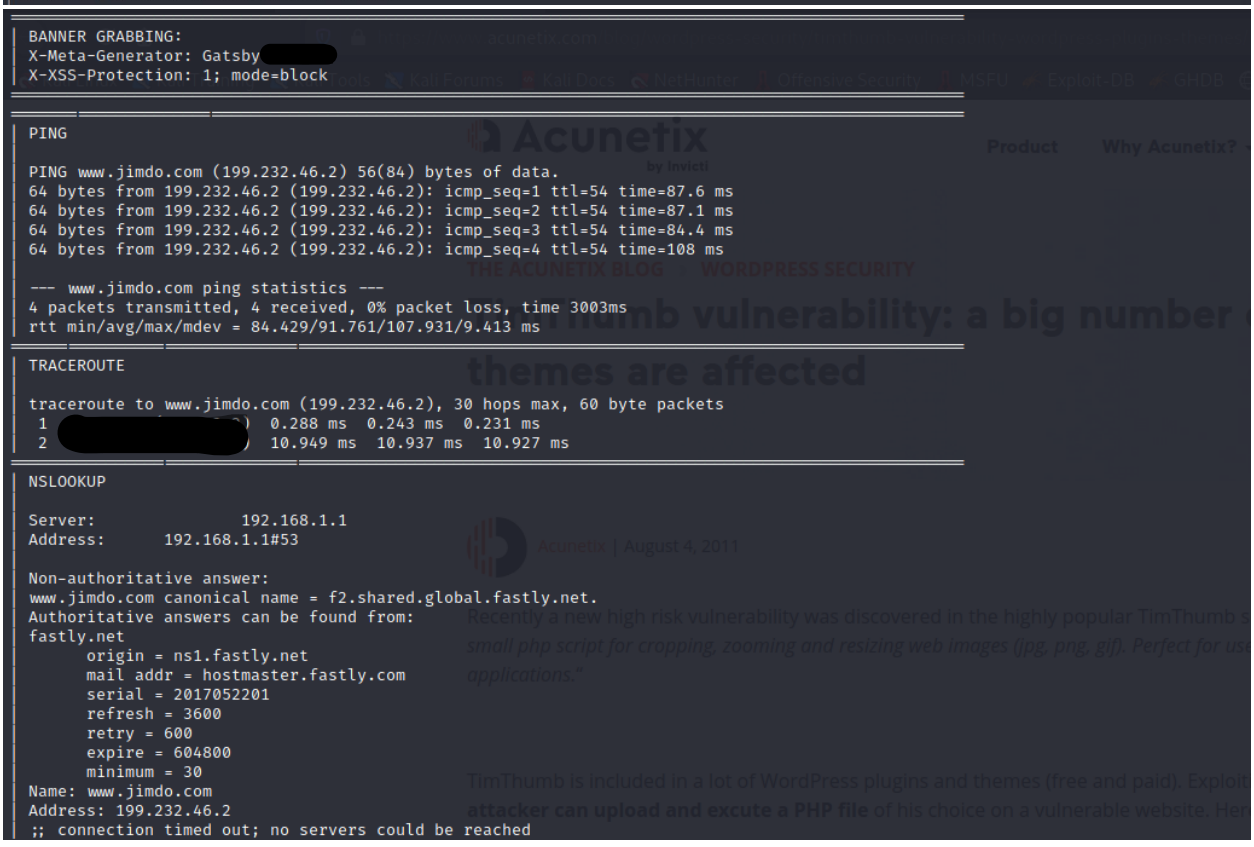


Fig. 67. Options that can use to filtered way of scanning

These are the commands used to filter the scanning results. Because of that the manual configuration efficiently gets the scan result.

- Scan the result of the www.jimdo.com domain.



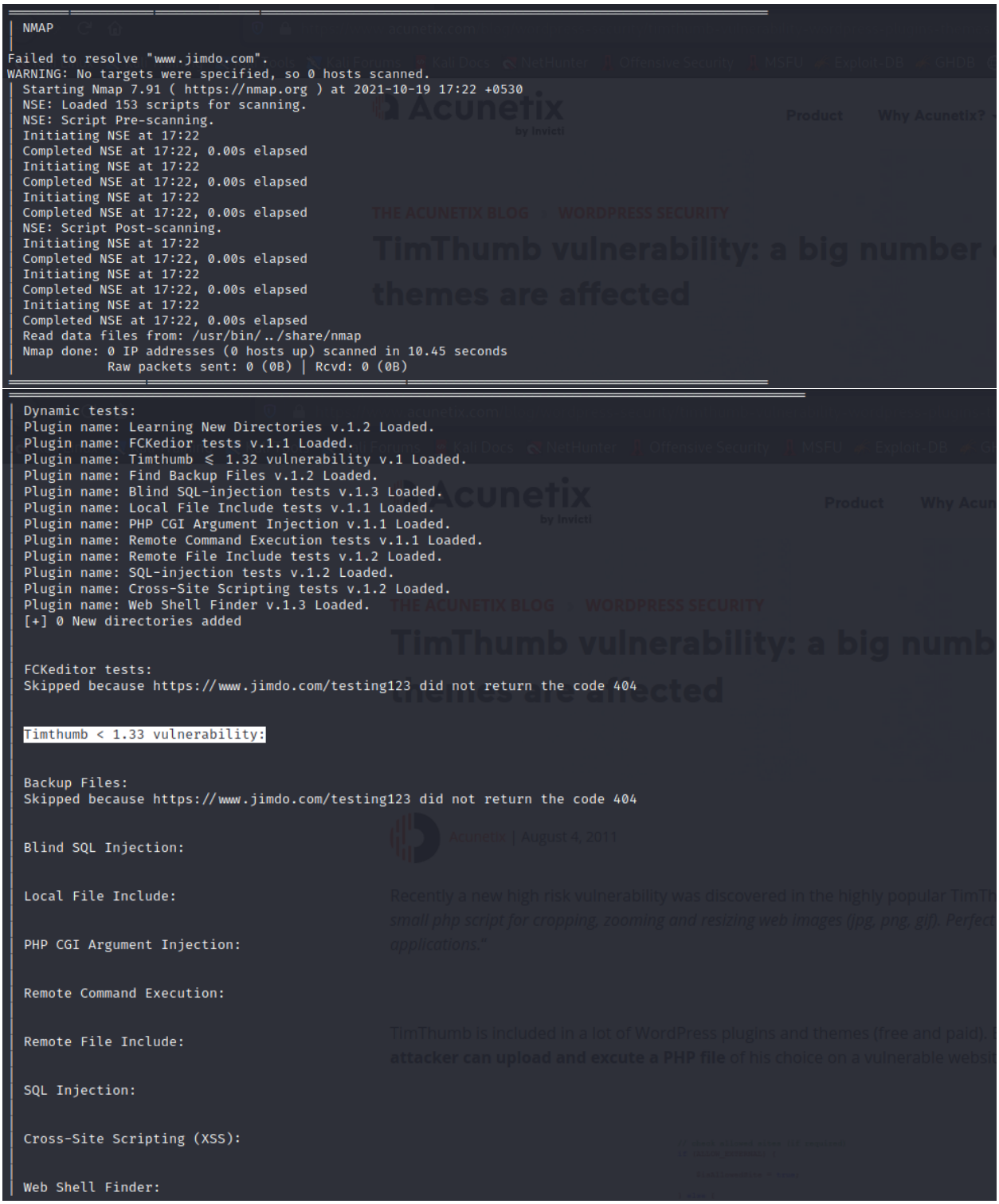


Fig. 68. Full scan report of www.jimdo.com

this is the scan result we can get from this tool. So, there are no vulnerabilities captured by this tool. But Nmap and other scan results are important to find vulnerabilities in the targeted system.

➢ Netsparker

Netsparker is a powerful vulnerability detection tool. According to Netsparker.com, “Netsparker is an automated, yet fully configurable, web application security scanner that enables you to scan websites, web applications, and web services, and identify security flaws.” [9]. This scanner is covered every unsecured area in the web application and consistently outputs every bit of weakness that the targeted domain has. But this process takes a lot of time and processing power to output the result.

Why because this tool goes through a lot of vulnerabilities in the targeted domain. Such as Remote Code Evaluation, SQL injection, Blind SQL injection, Boolean SQL injection, Remote File Inclusion (RFI), Command injection, Server-side Template Injection, Blind Command Injection, Injection via Local File Inclusion, and Local File Inclusion (LFI). Not only that Netsparker tool provides a full report of the exploitation method of those found vulnerabilities and the prevention method.

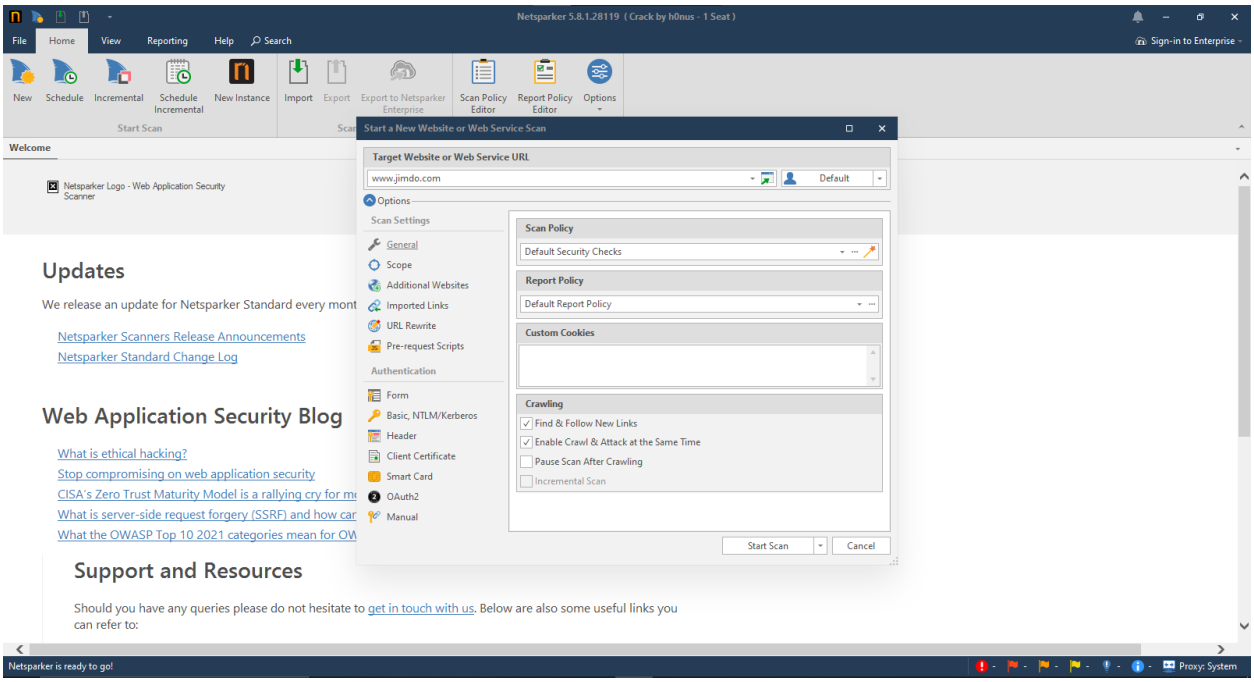


Fig. 69. Dashboard of the Netsparker Tool

So, this tool is automated, but we can customize technical details of the targeted domain and reduce the time taken to scan the target website.

- Customize the Web Server details of the targeted domain.

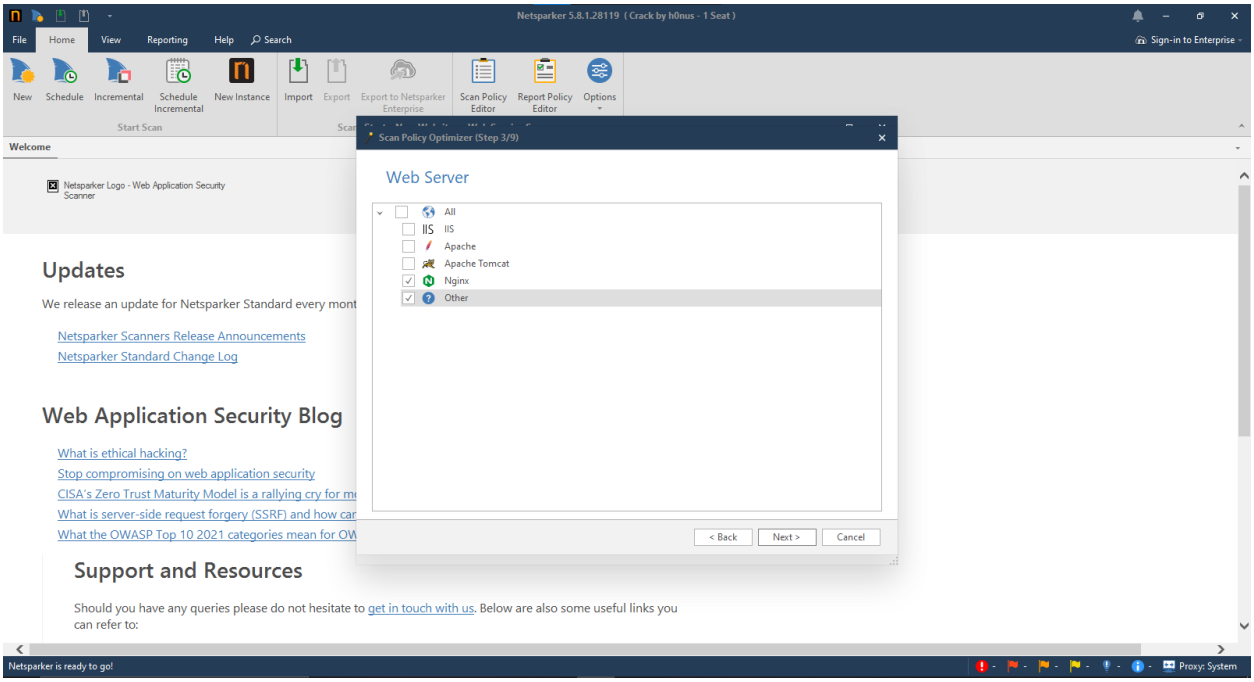


Fig. 70. Customize the Web Server details

- Customize the Application Server details of the targeted domain.

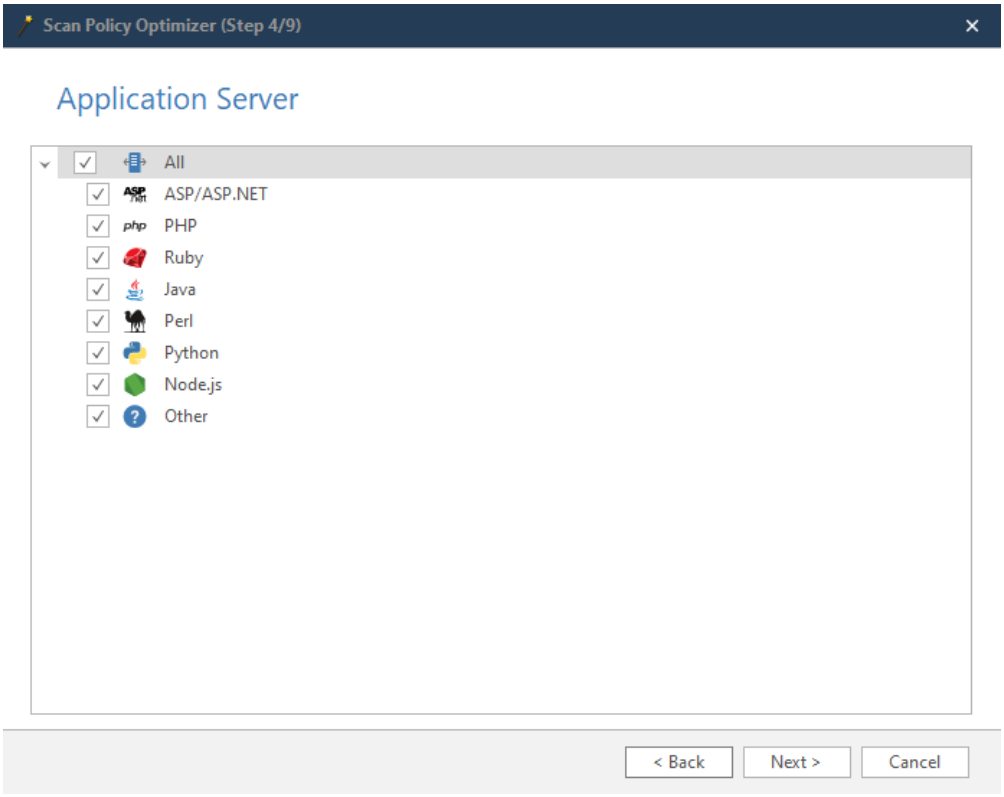


Fig. 71.Customize the Application Server details

- Customize the Database Server details of the targeted domain.

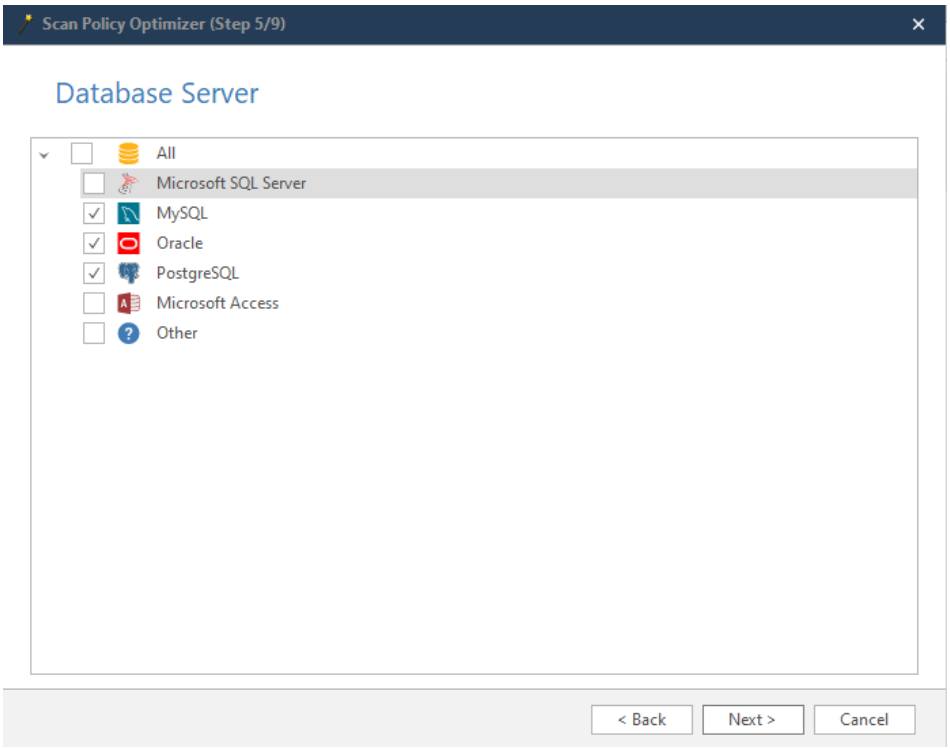


Fig. 72. Customize the Database Server details

- Targeted domain scanning process.

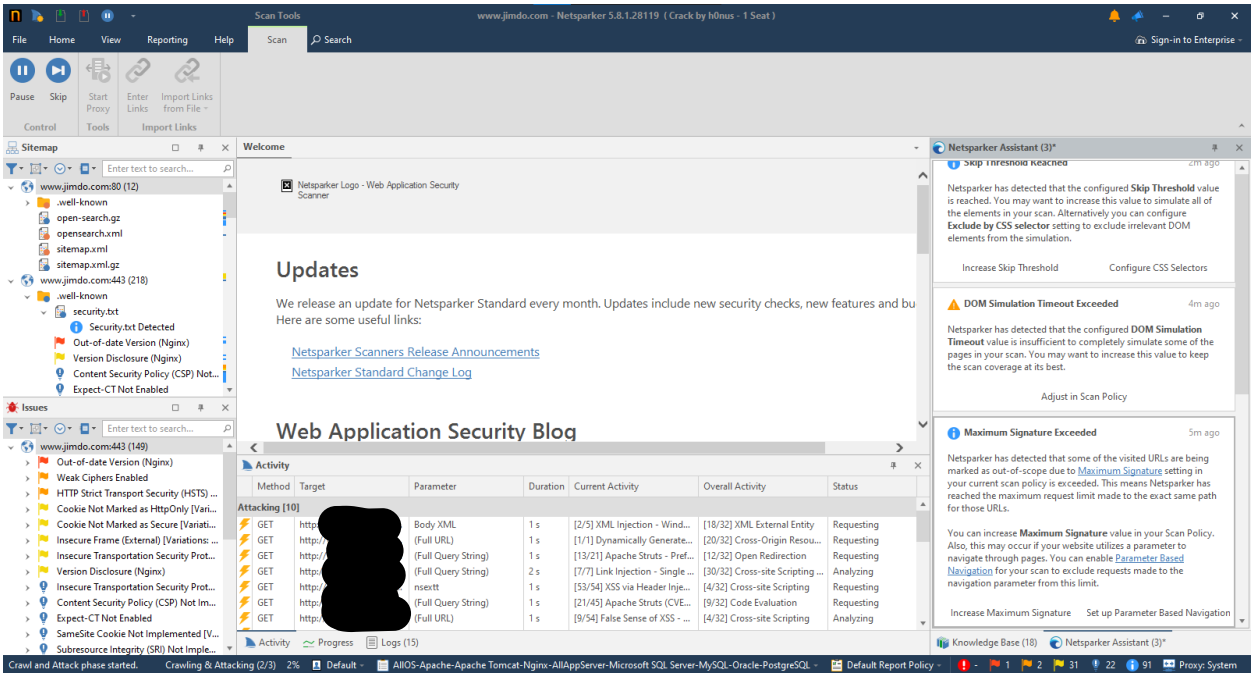


Fig. 73. Scanning process

After scanning the domain Netsparker tool output the vulnerability scanning report as a PDF file. So, these are the scanning results according to each domain.

- Scan report of www.jimdo.com.

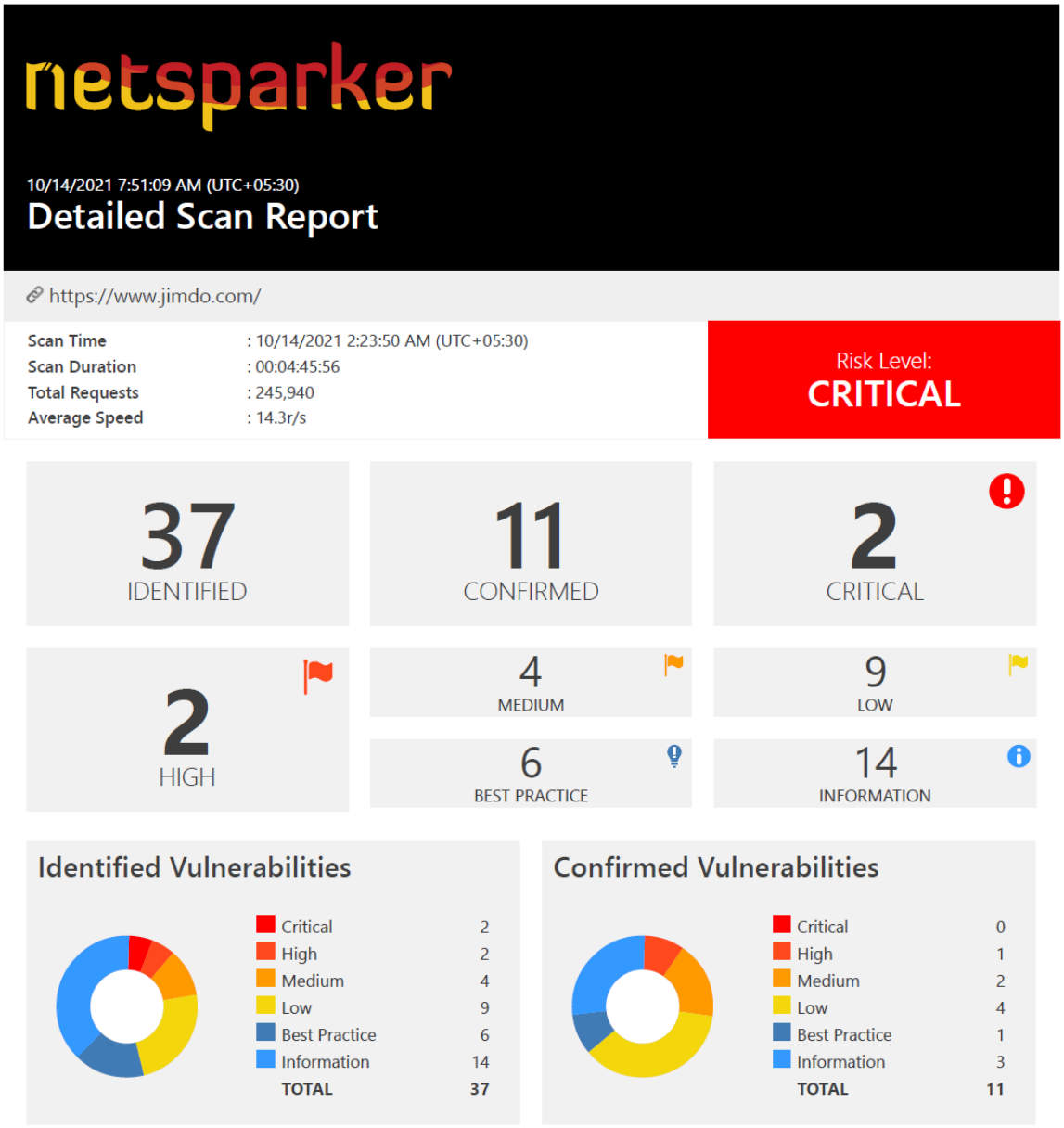


Fig. 74. Summary of the scan report

This is the report summary of the scanned vulnerabilities of the main domain. Netsparker tool identified two Critical and two High damageable vulnerabilities in the main domain.

Vulnerability Summary

CONFIRM	VULNERABILITY	METHOD	URL	PARAMETER
	Out-of-date Version (Lodash)	GET	https://ww[redacted]	
	Out-of-date Version (Nginx)	GET	https:[redacted]	
	Out-of-date Version (Modernizr)	GET	https://www.jimdo.co[redacted]	
	Session Cookie Not Marked as Secure	GET	https://www.jimdo.co[redacted]	
	HTTP Strict Transport Security (HSTS) Errors and Warnings	GET	https://ww[redacted]	
	Out-of-date Version (jQuery)	GET	https://www.jimdo.co[redacted]	
	Active Mixed Content over HTTPS	GET	https://www.jimdo.com/about-jimd[redacted]	
	Weak Ciphers Enabled	GET	https://ww[redacted]	

Fig. 75. Vulnerability Summary

These are the vulnerabilities that scan by the Netsparker tool and there are three confirmed vulnerabilities. One of these is High-level damageable vulnerability.

Jimdo's website uses out-of-date version servers and other services. Also, that the confirmed vulnerability is about the security issue of Session and Cookies. So, compared with other vulnerability detection tools Netsparker is a very efficient and more reliable tool.

Consider the first vulnerability captured by this tool, it is about the Lodash server-side software used in this website is the Out-of-date version. In the information gathering stage, we found older version software used to develop this targeted domain.

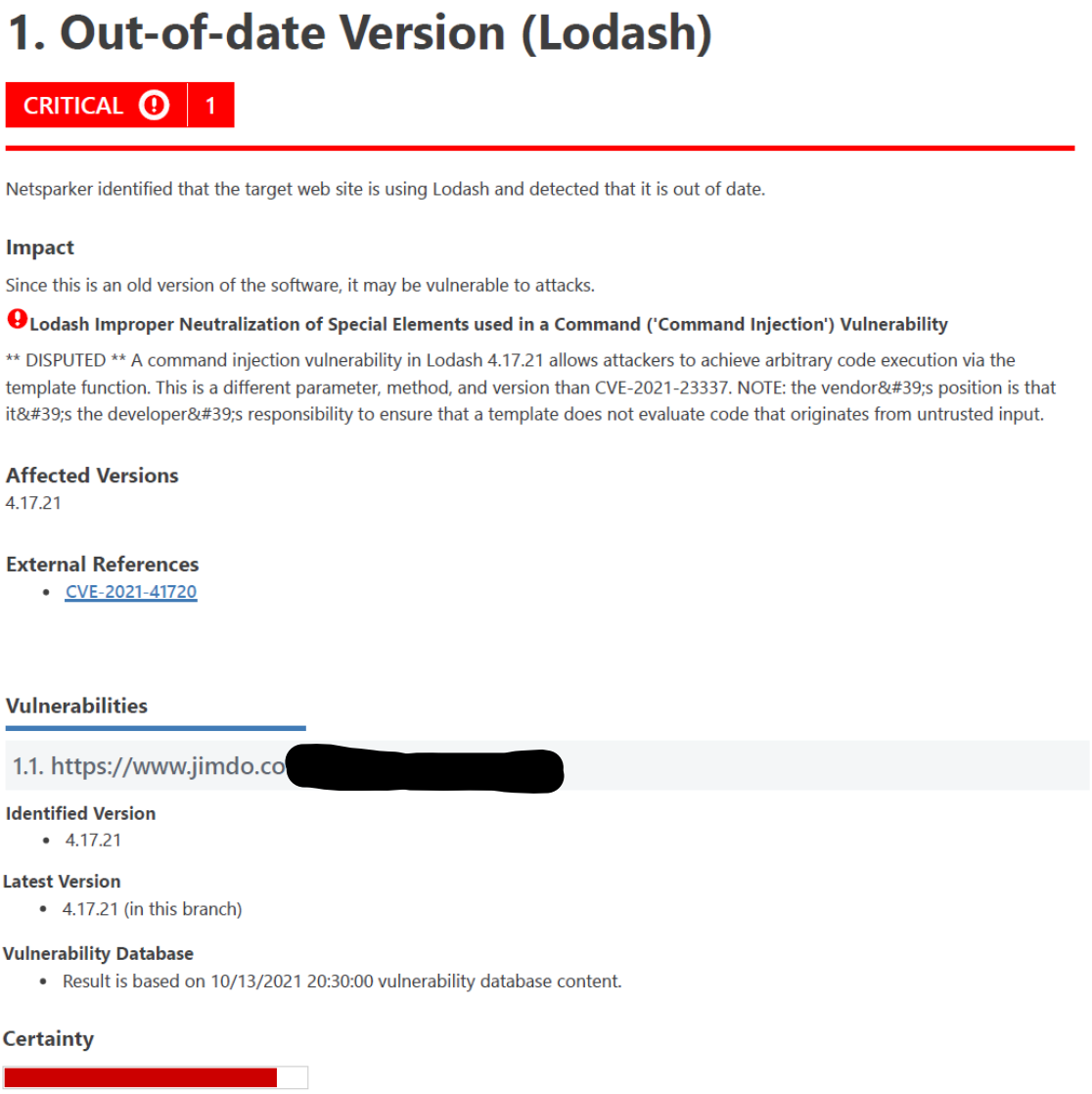


Fig. 76. Out-of-date Version (Lodash)

This older version of the software is vulnerable to the attacker because of the bugs have in the older versions. So, this software needs to be updated to a newer version or switch to the low risky anther newer version of the software which the same work done in Lodash software.

According to the Educative website, “Lodash is a JavaScript library that provides utility functions for common programming tasks using a functional programming paradigm; it builds upon the older underscore.js library.” [10]. So, this software also does not

updatable because this software is in the latest version. Instead of Lodash software, a newer version of the software can be used.

According to Netsparker, there is a command injection vulnerability that allows attackers to achieve arbitrary code execution via the template function and the targeted subdomain also mention in this report (https://www.jimdo.com[redacted]).

Other Critical vulnerability is also the Out-of-date Version vulnerability of the Nginx software.

2. Out-of-date Version (Nginx)

CRITICAL 1

Netsparker identified you are using an out-of-date version of Nginx.

Impact

Since this is an old version of the software, it may be vulnerable to attacks.

Nginx Allocation of Resources Without Limits or Throttling Vulnerability

Some HTTP/2 implementations are vulnerable to a header leak, potentially leading to a denial of service. The attacker sends a stream of headers with a 0-length header name and 0-length header value, optionally Huffman encoded into 1-byte or greater headers. Some implementations allocate memory for these headers and keep the allocation alive until the session dies. This can consume excess memory.

Affected Versions

1.9.5 to 1.16.0

External References

- CVE-2019-9516

Nginx Other Vulnerability

Some HTTP/2 implementations are vulnerable to resource loops, potentially leading to a denial of service. The attacker creates multiple request streams and continually shuffles the priority of the streams in a way that causes substantial churn to the priority tree. This can consume excess CPU.

Affected Versions

1.9.5 to 1.16.0

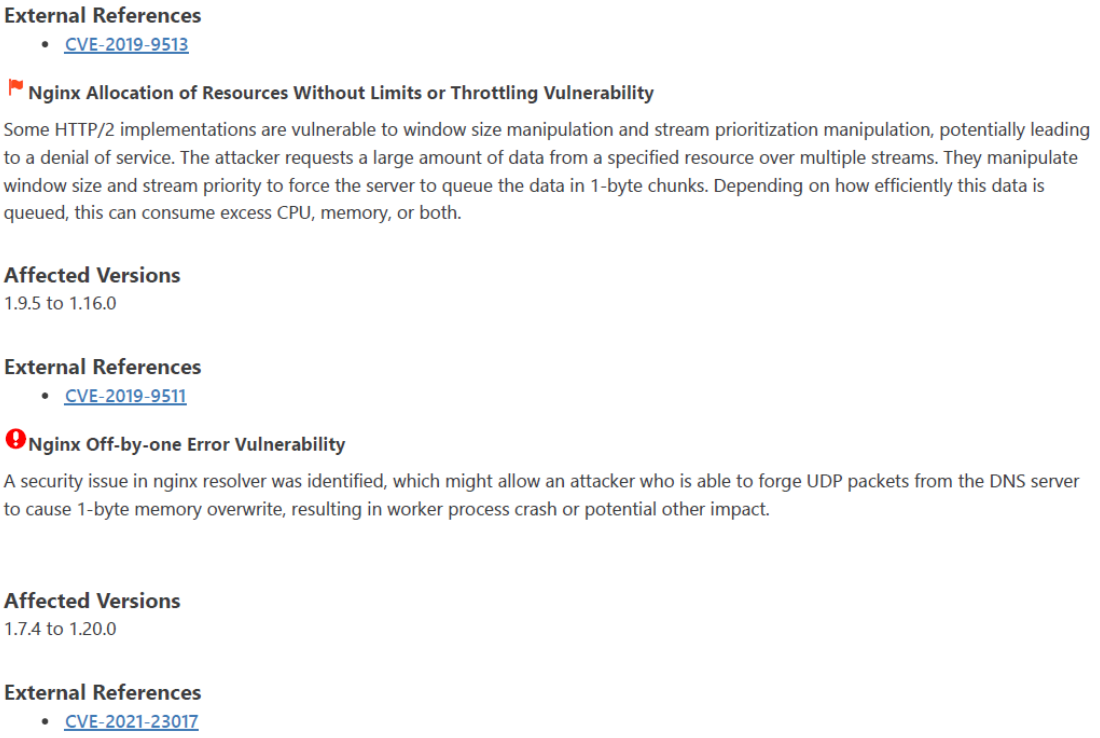


Fig 77. Out-of-date Version (Nginx)

According to the Nginx, “NGINX is open-source software for web serving, reverse proxying, caching, load balancing, media streaming, and more.” [11]. Because of that, this is critical software used in the targeted domain. That is why need to all the time update critical functioning software like Nginx.

So, this vulnerability is identified also in the information gathering stage. When using Netcraft and Whatweb tools it mentions that this domain uses an older version of Nginx software. So, that assumes as vulnerable to attack and it is confirmed by the Netsparker tool.

According to the Netsparker tool, this is HTTP/2 implementations are vulnerable to a header leak, resource loops, window size manipulation, and stream prioritization manipulation, potentially leading to denial of service attacks. And there is vulnerable to an attacker who can falsify DNS server UDP packets to trigger a 1-byte memory overwrite, resulting in worker process

crash or other possible effects. These attacks can be done directly to the main domain (https://www.jimdo.com/).

Other vulnerabilities are High and Medium level vulnerabilities. First, consider that there is a High level of confirmed vulnerability regarding Session and Cookies.

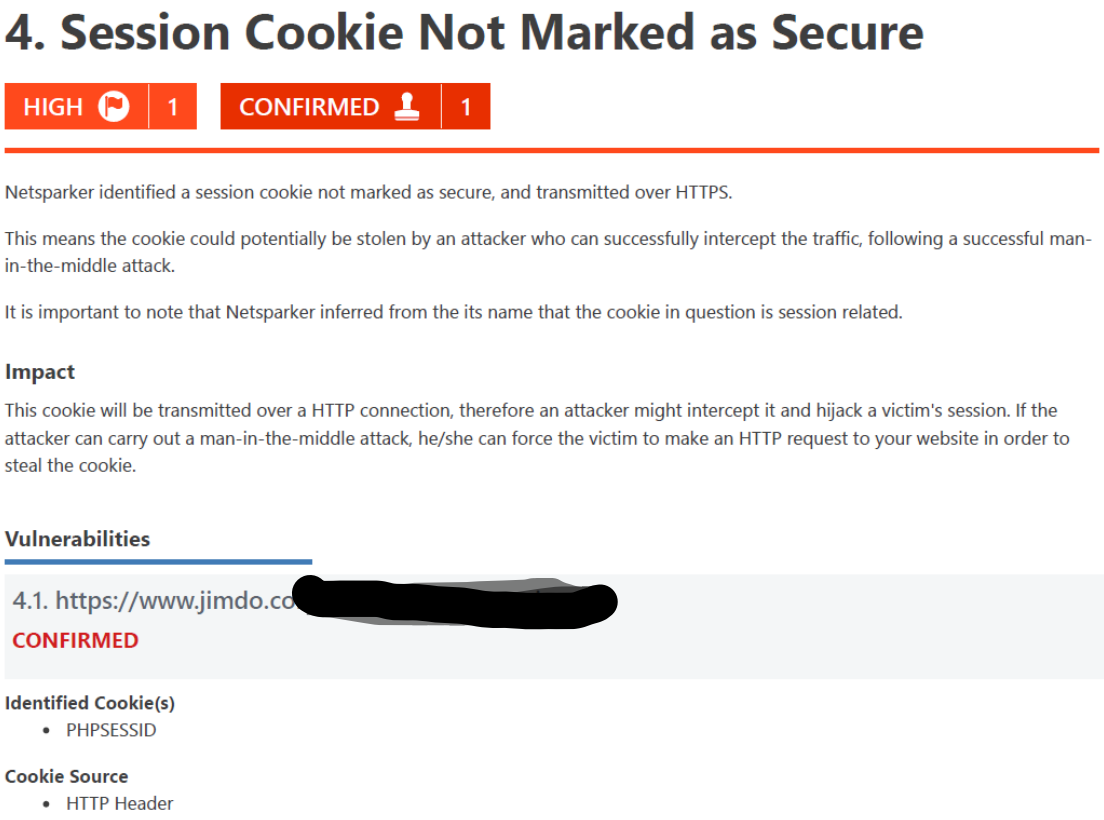


Fig. 78. Session and Cookie Not Marked as Secure

This vulnerability is that the Session and Cookies are not protected using protection methods. Consider the Session and Cookies are used to maintain the state of the web application. So, these types of vulnerabilities are fair enough to leak your account details to the attacker. Because of the man-in-the-middle attack possible to happen with those vulnerabilities in the targeted domain. And also targeted domain is mentioned and confirmed in the report (https://www.jimdo.com[redacted].

Next, consider the Medium level much important vulnerability that we need aware of. This vulnerability is related to HTTP Strict Transport Security (HSTS).

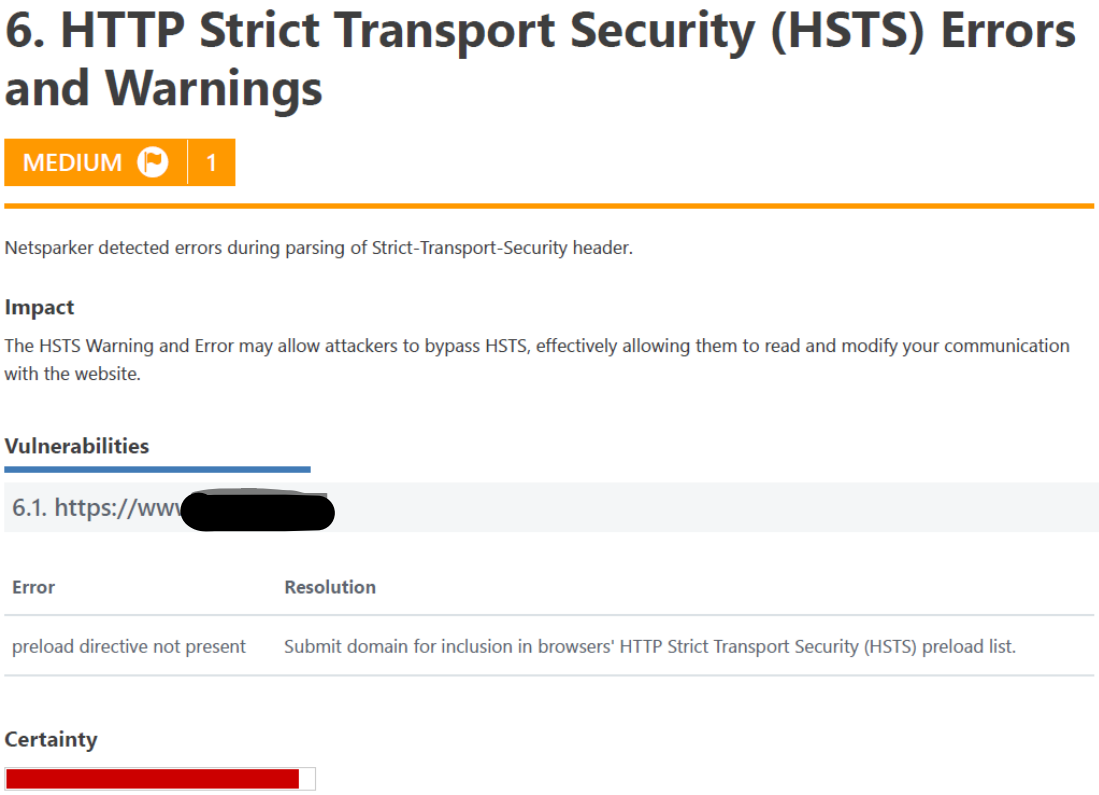


Fig. 79. HTTP Strict Transport Security (HSTS) Errors and Warnings

In the information gathering stage that the Whatweb tool used found the HTTP Strict Transport Security (HSTS) is not in this web application according to the status code used in targeted domains. Because of that, this website can be vulnerable to the SSL Stripping attack. Even this is a Medium level of vulnerability that the impact is in the critical stage. Not only this vulnerable domain is also the main domain of this web application (https://www.jimdo.com/).

- Scan report of jimdo.design

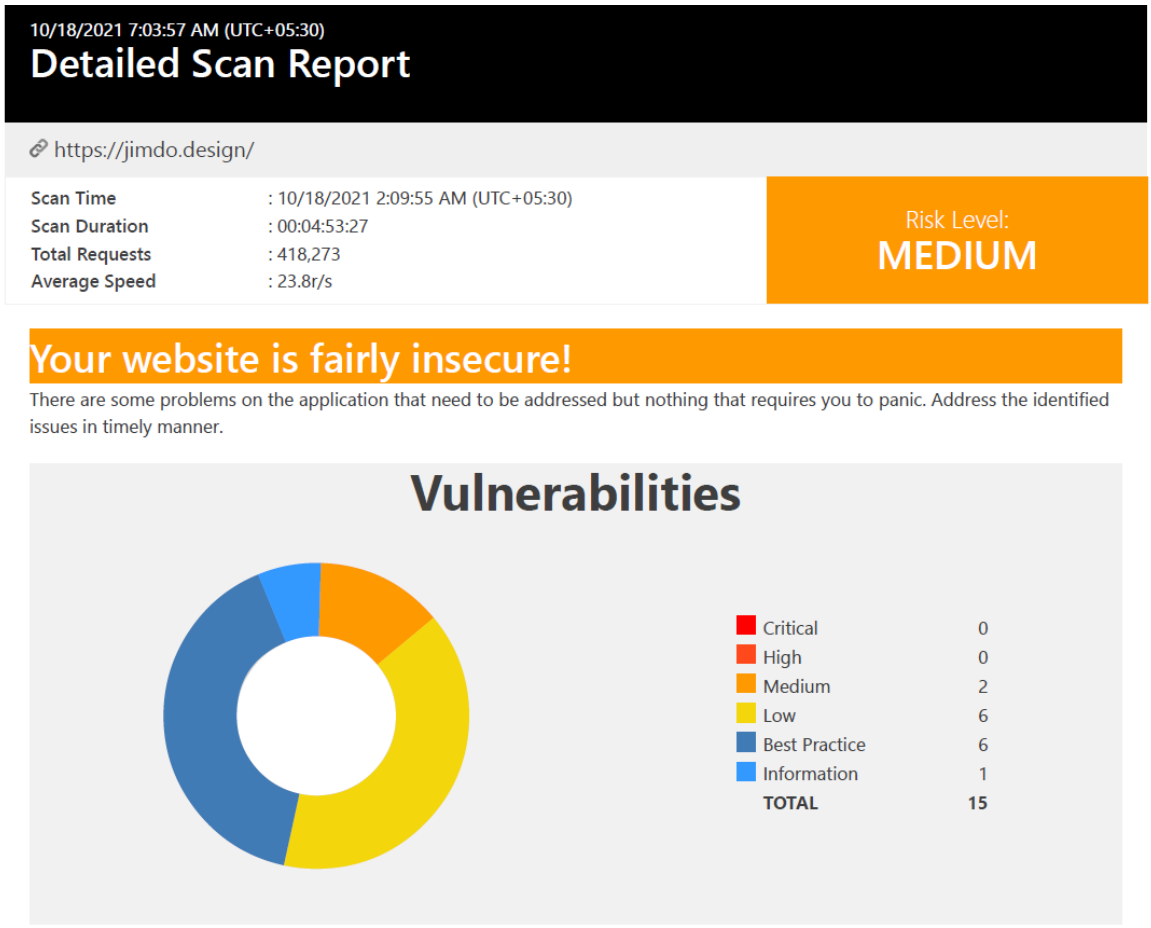


Fig. 80. Report of jimdo.design

- Scan report of halp.jimdo.com

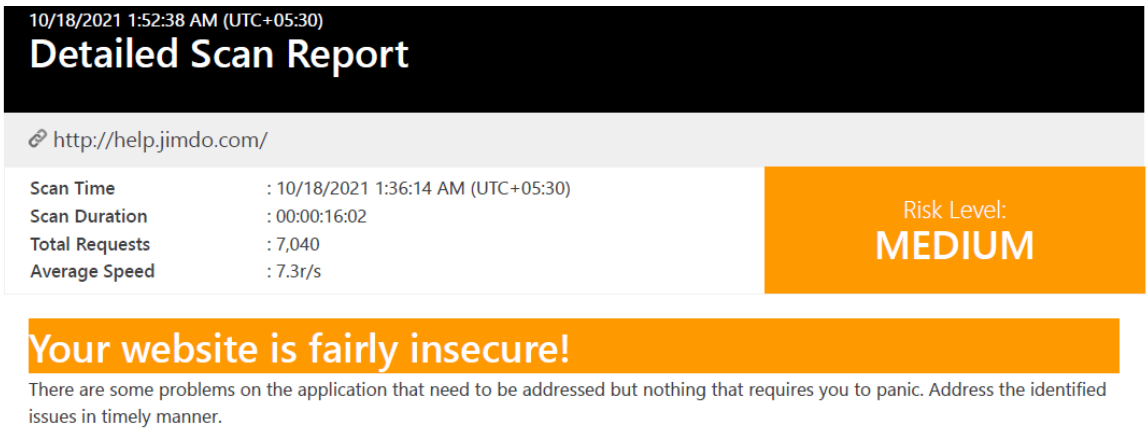


Fig. 81. Report of help.jimdo.com

Other scanned reports also do not capture any Critical or High level vulnerabilities.

- **Owasp ZAP**

The Open Web Application Security Project Zed Attack Proxy (OWASP ZAP) is an open-source web application vulnerability detection tool. This is one of the best vulnerability detection tool and efficient than compared with most other tools. Owasp ZAP is also can be used as web application professional penetration testers. This tool work according to the OWASP top 10 security threats. Such as Cross-site scripting (XSS), Broken access control, SQL injection, Broken authentication and session management, Security misconfiguration and other security threats.

Consider that how does Owasp ZAP work, according to Srijan's Framework and Libraries, "ZAP creates a proxy server and makes your website traffic pass through that server. It comprises of auto scanners that help you intercept the vulnerabilities in your website." [12]. There is an automated or manual scanning option and for this assignment choose that the automated scan method because the automated method filter and scan only the in-scope subdomains. The automated scan is also customizable and if it is customized well, we can reduce that time taken for scanning the targeted subdomain.

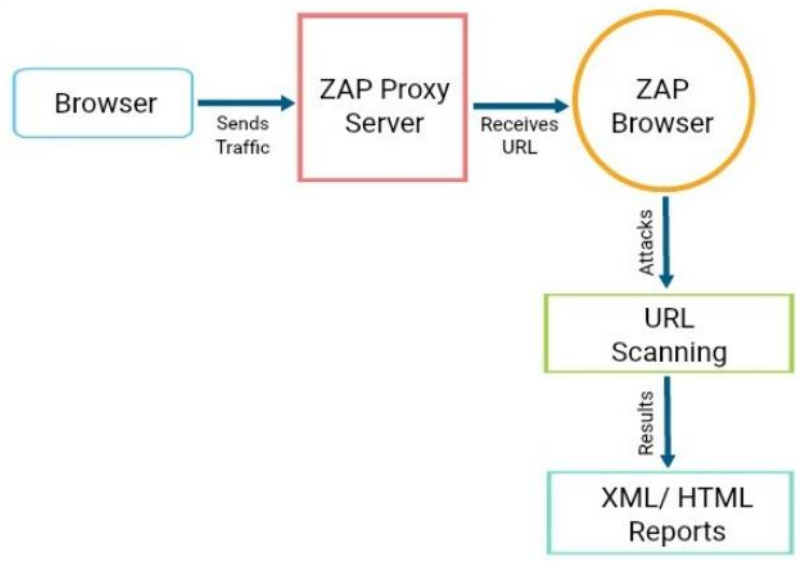


Fig. 82. How does Owasp ZAP work

- Customize the scan policy and add a new policy according to the targeted subdomain.

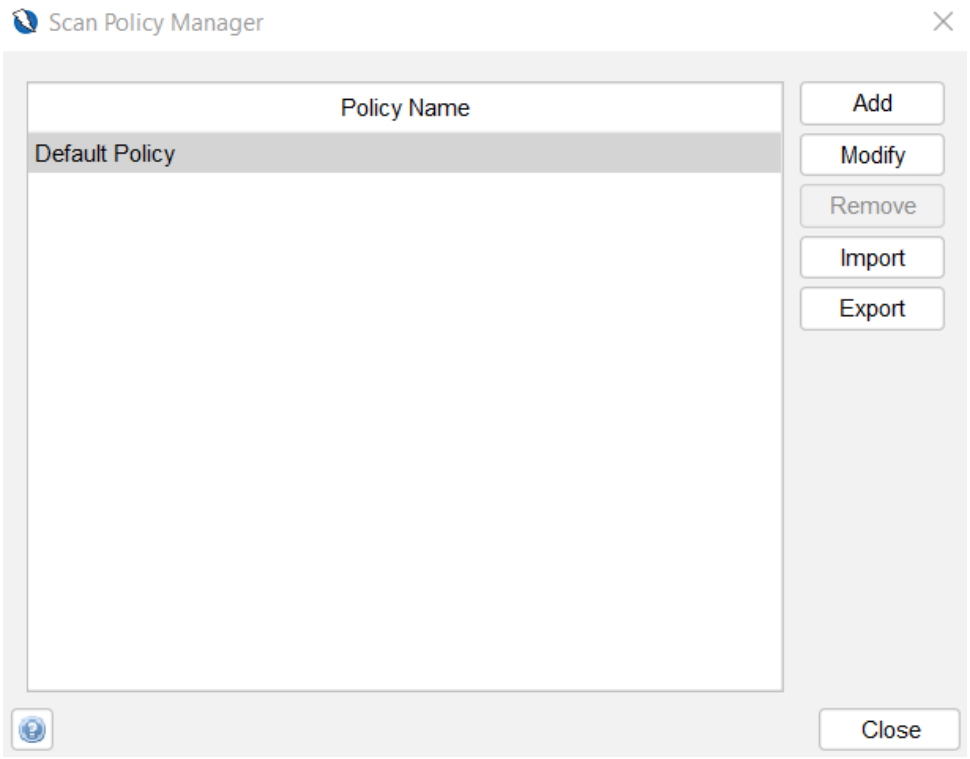


Fig. 83. Scan Policy Manager

- Change the scan policies according to the targeted domain.

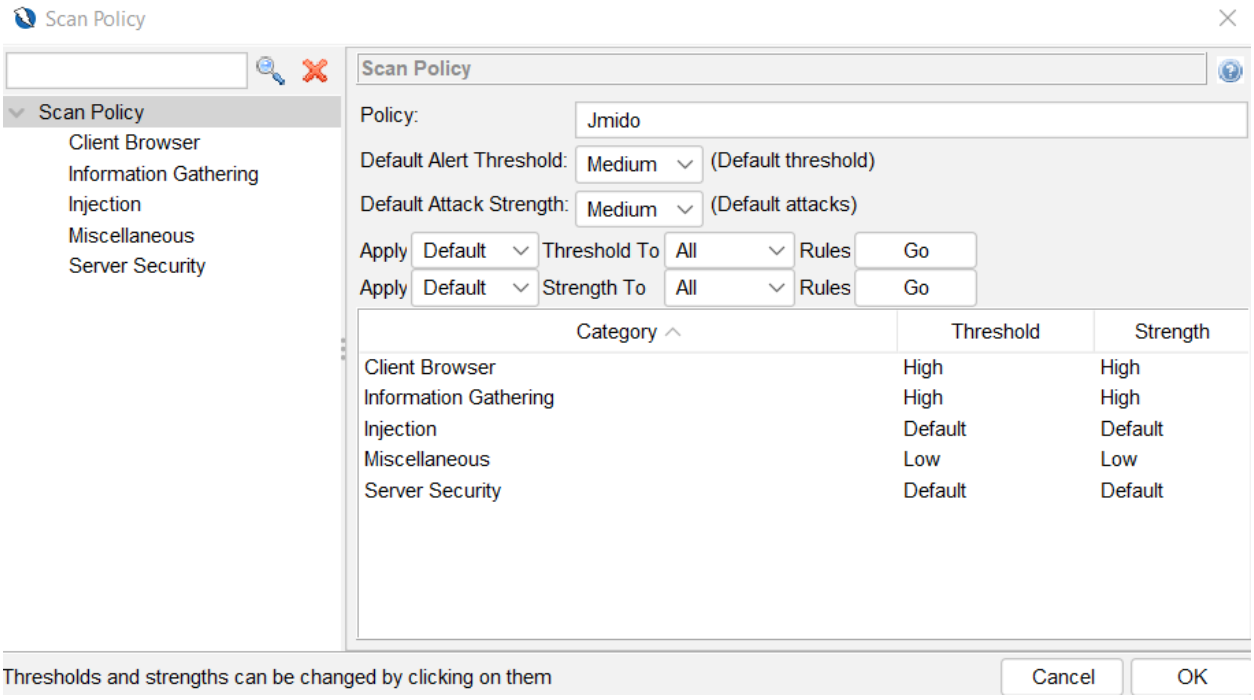


Fig. 84. Customize the new scan policy profile

- Scanning process of www.jimdo.com

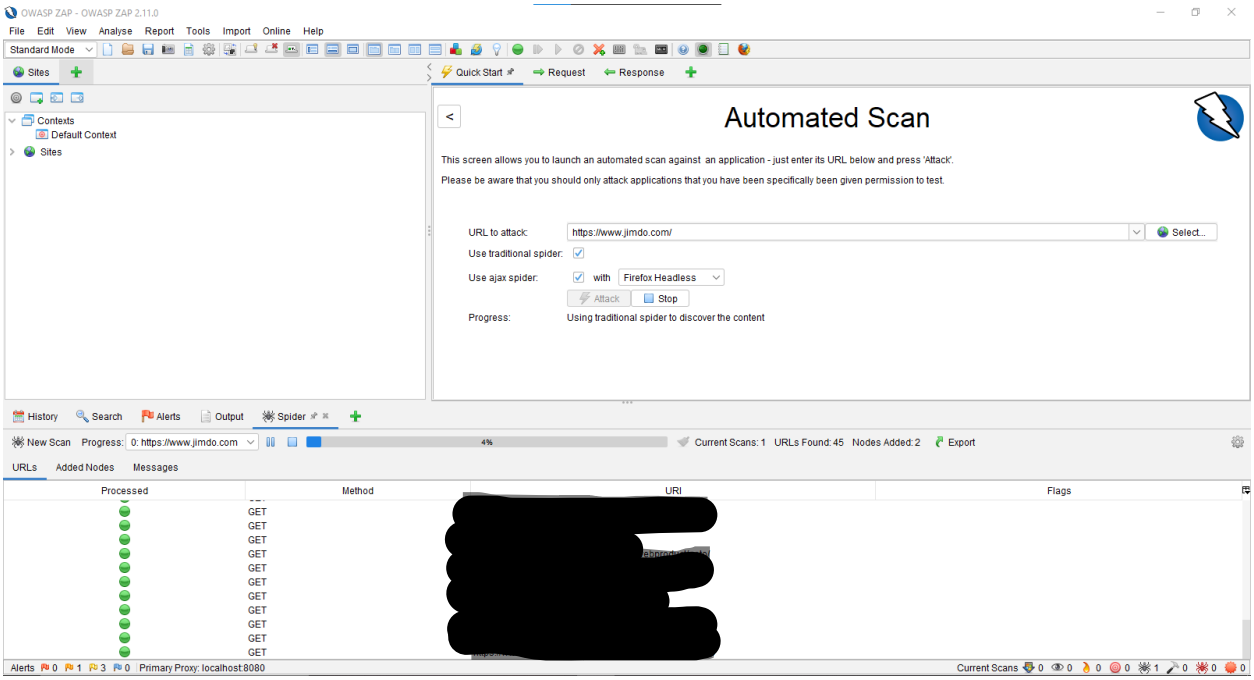
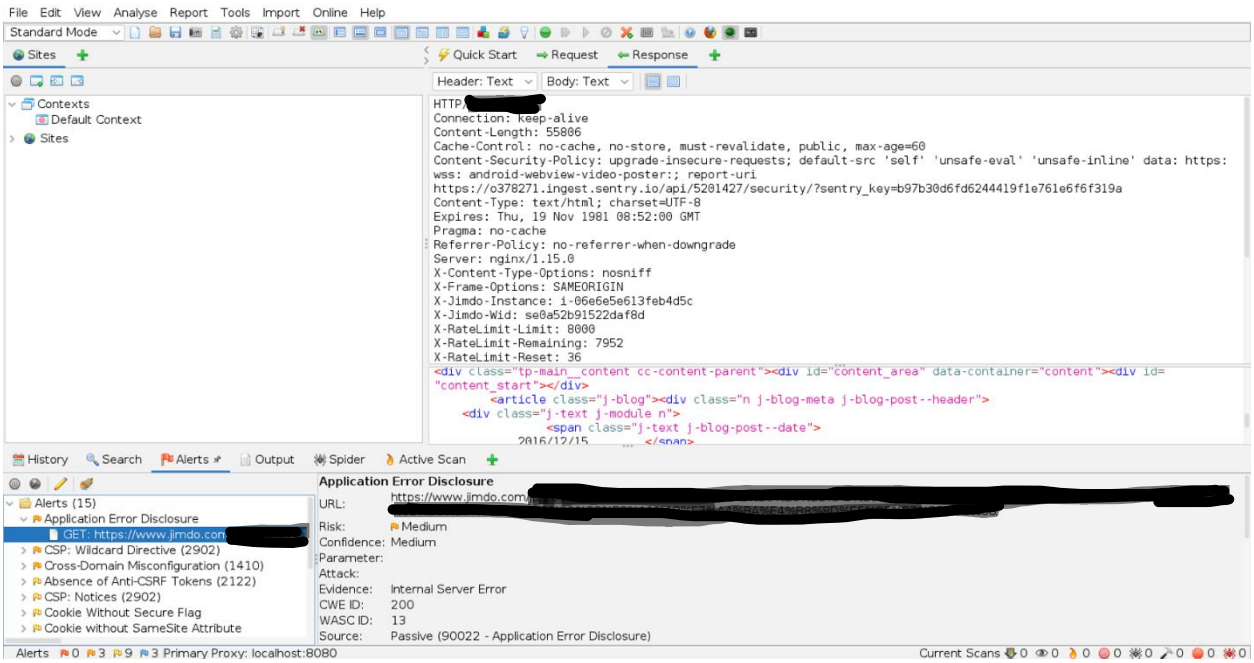


Fig. 85. Automated Scanning Process

- Scan results of www.jimdo.com



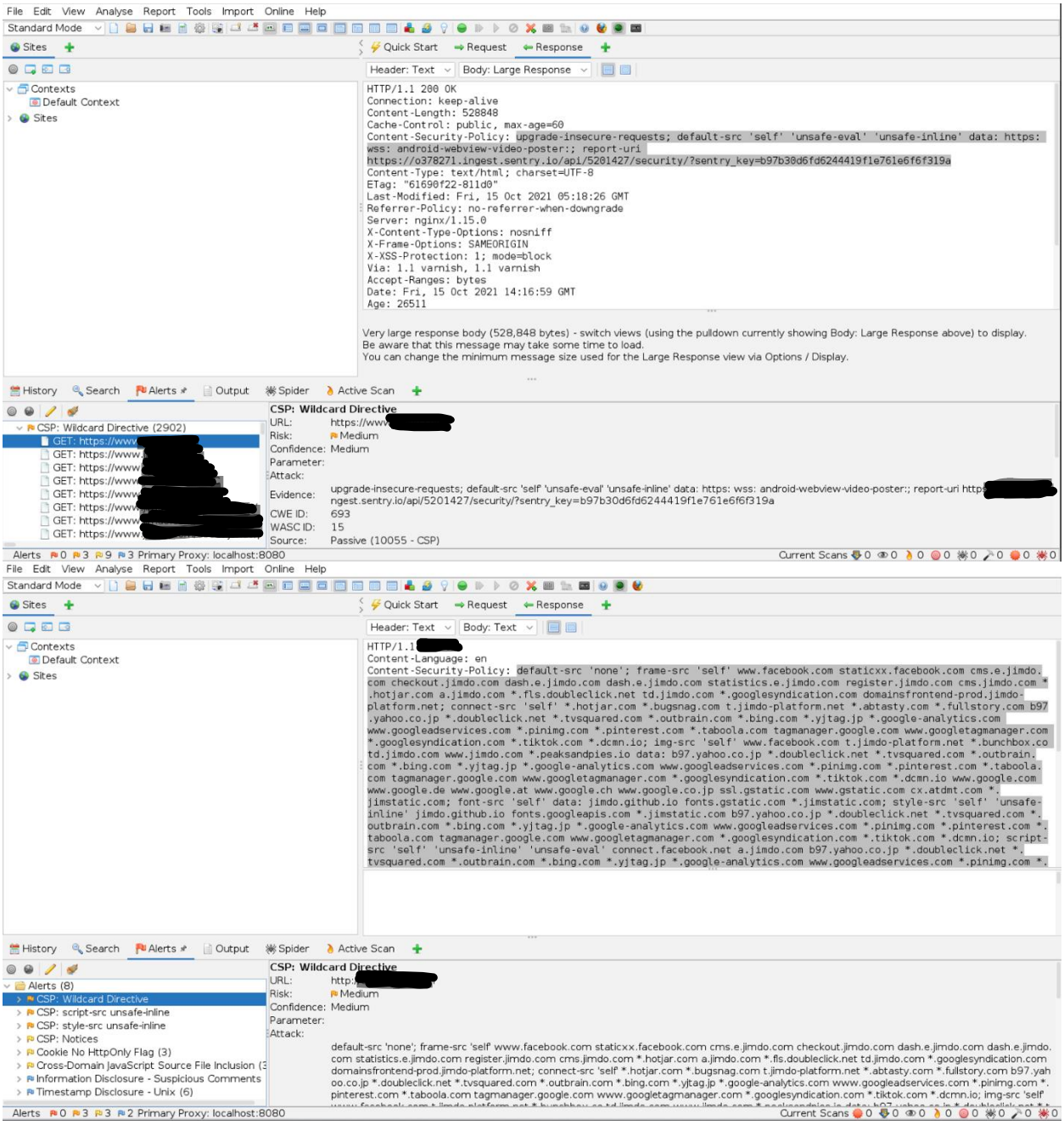


Fig. 86. Scan Results

Even this is a customized scan this tool consumes a lot of time to process the scan. Because of that, I had to abort the scan and get the result. So, these results are not the finalized result, and I already identified the vulnerabilities in the targeted system using the Netsparker and other tools.

Penetration Testing

Penetration Testing is a really important stage in Bug Bounty Assessment. Because in this stage test the scanned vulnerabilities found in the targeted subdomain. So, we can find out that vulnerabilities are actually exploitable or not. According to the National Institute of Standards and Technology (NIST), "Penetration Testing is a method of testing where testers target individual binary components or the application as a whole to determine whether intra or intercomponent vulnerabilities can be exploited to compromise the application, its data, or its environment resources." [13]. Also, this process is named ethical hacking or pen-testing. So, this process can help to confirm the vulnerabilities in the targeted system.

Penetration testing is a crucial aspect in the confirmation of data security in every aspect of the data is used today. The necessity of it is highlighted due to the benefits it gives. Penetration tests allow us to identify new bugs and loopholes in existing software, test new software for existing bugs, and whether the implemented security controls are sufficient to handle the latest security threats. It enables us or our company to be able to stay up to standard with recognized international standards like General Data Protection Regulation (EU GDPR), Data Protection Act (DPA), Payment Card Industry Data Security Standard (PSI DSS), fix the identified bugs and loopholes in security controls that have already been implemented to assure our clients and stakeholders that their data is secure.

After confirming those vulnerabilities, we need to report these vulnerabilities and the protection methods to the relevant company belong the targeted system. And this process needs to be done before attackers exploit the system.

In the vulnerability detection stage, there are identified Critical level and High level vulnerabilities. In the penetration testing stage check, these identified vulnerabilities are impacting the targeted domains and suggest that the protection techniques secure the target domains. So, mainly most successful scanning results are produced by the Arachni tool and the Netsparker tool.

➢ **Penetration testing according to the Arachni scan results**

The Arachni tool found a High-level vulnerability and three subdomains are impacts for this same Cross-Site Scripting (XSS) vulnerability in HTML tag.

- ✓ https://www.jimdo.com[redacted]
- ✓ https://www.jimdo.com[redacted]
- ✓ https://www.jimdo.com[redacted]

These are the three subdomains vulnerable to this attack. According to the Jimdo company policies, As this type of behavior is intended, each and every report which asks a user to place a Stored XSS payload on their own page to exploit anonymous users visiting the website, is taken as Out of Scope and will be closed as such [14]. So, we can not test those vulnerabilities as anonymous users and exploit those subdomains. Because of that, I do not attempt to exploit and see the results. But according to Arachini's scan report, I will suggest the prevention methods, what type of Cross-Site Scripting (XSS) vulnerability, and how it happens.

Modern-day web applications make extensive use of client-side scripting. Client-side scripts handle anything from basic text formatting to comprehensive data processing and operating system interactions on the client-side. So, clients can inject scripts into a request and have the server respond with scripts. This is known as cross-site scripting (XSS) and this occurs as a result of web applications acquiring untrustworthy data and reusing it without carrying out any validation or sanitization. There are three main types of cross-site scripting attacks,

- o Reflected cross-site scripting
  - ✓ where the malicious script comes from the current HTTP request [15].
- o Stored cross-site scripting
  - ✓ where the malicious script comes from the website's database [15].
- o DOM-based cross-site scripting
  - ✓ where the vulnerability exists in client-side code rather than server-side code [15].

So, the Reflected cross-site scripting and the Stored cross-site scripting attacks are mostly can be done with this HTML injection vulnerability. It has been discovered by Arachni that it is almost possible to directly insert content into an HTML tag. For example <INJECTION_HERE href=.....etc> where INJECTION_HERE shows the place where the payload was detected by Arachni.

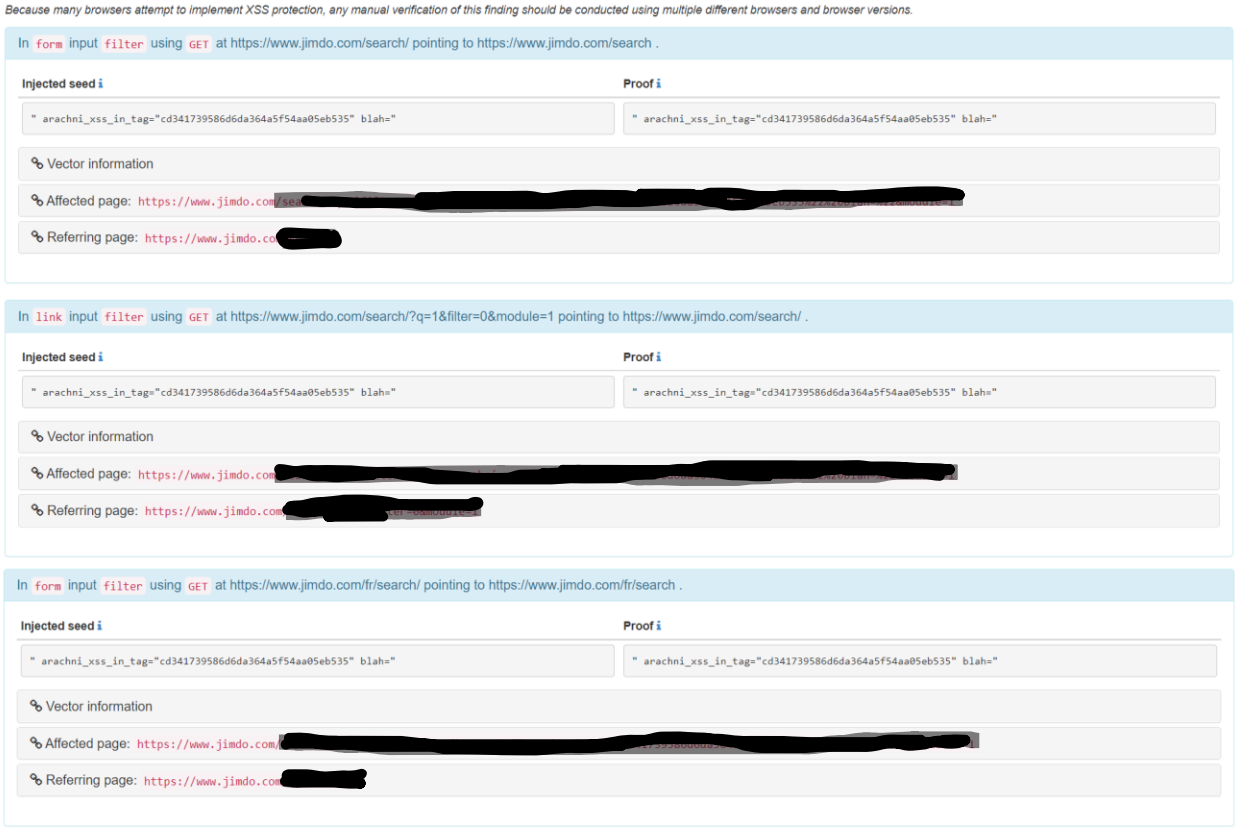


Fig. 87. Proof the Cross-Site Scripting (XSS) vulnerability in HTML tag

Consider how to protect from this vulnerability, It is critical to avoid using untrusted or unfiltered data in HTML page code. Untrusted data might come from a variety of sources, including the client, a third party, a previously submitted file, and so on. And also, Converting special characters to their HTML entity encoded counterparts is a common method of filtering untrustworthy material. As an example converting “>” to “>”. Despite the fact that unsafe input may be filtered, there are five locations on an HTML page where it should never be placed, and Each of these locations has its own method of escape and filtration. as an example, these are the location unsafe inputs can happen, inside an HTML comment, directly in a script, in

a tag name, in an attribute name, and so on. So, these are the prevention methods that can be used to protect against HTML injection attacks.

- **Penetration testing according to the Netsparker scan results**

The Netsparkrt tool found several Critical level vulnerabilities. So, in this assignment consider that penetration testing the two Critical vulnerabilities that can be a threat to the targeted system.

1. Out-of-date Version (Lodash) Critical vulnerability

The current version that the Lodash use in Jimdo's system is an outdated version 4.[redacted] and it is vulnerable to the command injection attack. The vendor's position is that it's the developer's responsibility to ensure that a template does not evaluate code that originates from untrusted input. So, these types of vulnerabilities allow attackers to use the template function to execute arbitrary code.

- ✓ https://www.jimdo.com/[redacted]

This is the subdomain vulnerable to this attack. Command injection vulnerability can impact the targeted domain when the system receives data from an untrustworthy source, the data is part of a string that the system interprets as a command, and the applications grants an attacker, a privilege or capability that they would not otherwise have by performing the command. According to the, "If a malicious user injects a character (such as a semi-colon) that delimits the end of one command and the beginning of another, it may be possible to then insert an entirely new and unrelated command that was not intended to be executed." [16].

So, the only way to protect from this command injection attack is to upgrade the installation of Lodash to the latest stable version.

Request

```
GET /info/terms-of-service/ HTTP/1.1
Host: www.jimdo.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
Cookie: shd=5d1776a2-4242-4622-9963-4a7b40b28b5a; ckies_functional=deny; ckies_marketing=deny; ckies_ne
cessary=allow; ckies_performance=deny; jLang=en; _dd_s=logs=1&id=89733f59-ce15-41a2-ab2f-b809b2b30caa&c
reated=1634158440124&expire=1634159340129&rum=1
Referer: https://www.jimdo.com/
User-Agent: Mozilla/5.0 (Windows NT 10.0; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.
77 Safari/537.36
X-Scanner: Netsparker
```

Response

Response Time (ms) : 288.0005 Total Bytes Received : 492805 Body Length : 491888 Is Compressed : No

```
HTTP/1.1 200 OK
X-Cache: HIT, HIT
Age: 33197
Cache-Control: public, max-age=60
ETag: W/"6166be3b-78170"
Strict-Transport-Security: max-age=31536000
Server: nginx/1.15.0
X-Content-Type-Options: nosniff
X-XSS-Protection: 1; mode=block
Connection: keep-alive
Referrer-Policy: no-referrer-when-downgrade
X-Frame-Options: SAMEORIGIN
Accept-Ranges: bytes
X-Cache-Hits: 3, 1
Content-Length: 90354
Vary: Accept-Encoding
Via: 1.1 varnish, 1.1 varnish
Last-Modified: Wed, 13 Oct 2021 11:08:43 GMT
Content-Type: text/html; charset=UTF-8
X-Served-By: cache-lcy19256-LCY, cache-qpg1227-QPG
Content-Security-Policy: upgrade-insecure-requests; default-src 'self' 'unsafe-eval' 'unsafe-inline' da
ta: https: wss: android-webview-video-poster:; report-uri https://o378271.ingest.sentry.io/api/5201427/
security/?sentry_key=b97b30d6fd6244419f1e761e6f6f319a
Date: Wed, 13 Oct 2021 20:54:22 GMT
Content-Encoding:
```

Fig. 88. Request and Response of the vulnerable domain

2. Out-of-date Version (Nginx) Critical vulnerability

In the information gathering and the vulnerability detection stages already concerned about this vulnerability and the prevention methods. https://www[redacted] is infected for this vulnerability. As the protection method suggests upgrading the installation of Nginx to the latest stable version.

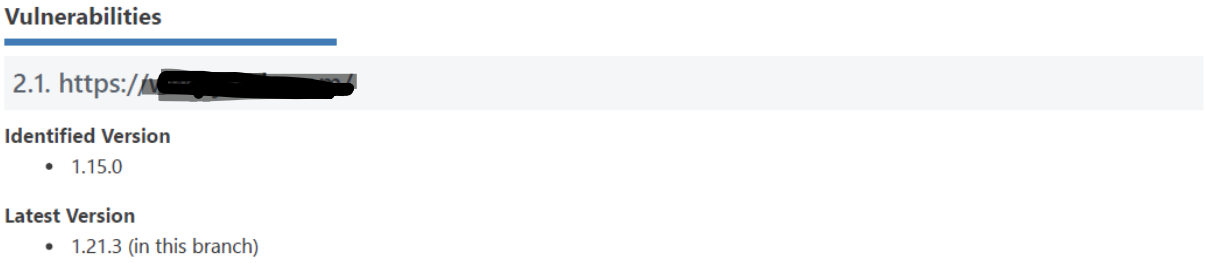


Fig. 89. Latest Version

So, in previous stages provide a lot of details about this vulnerability and I am not going to describe those things again and again here.

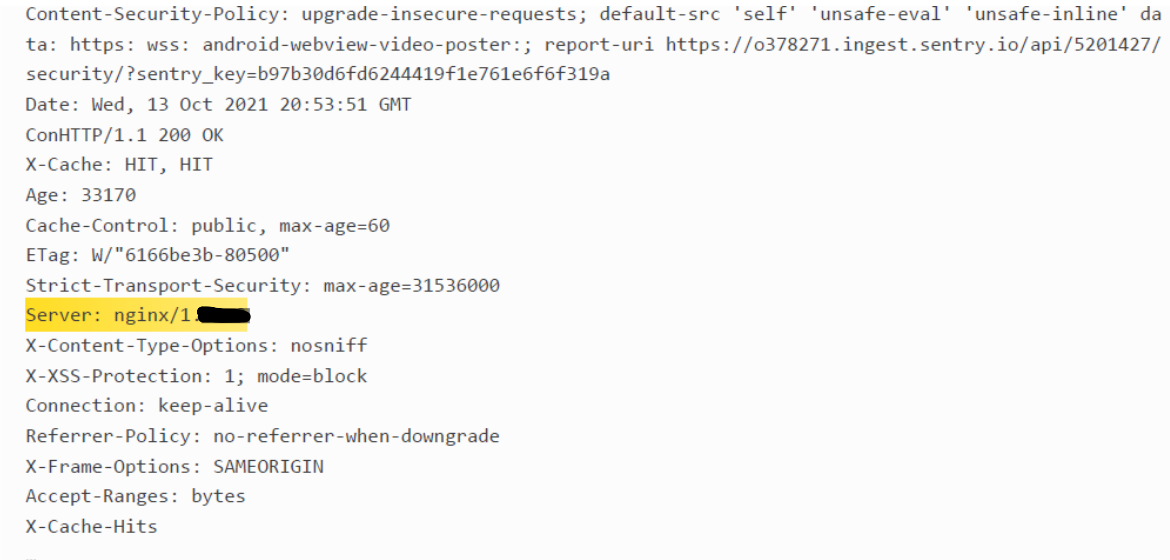


Fig. 90. Proof of this vulnerability

Conclusion

The purpose of this assignment was to assess the vulnerabilities of the web application. While engaging in this assignment I was able to identify a lot of Bug Bounty hunting platforms which helped to improve vulnerability assessing skills and knowledge about the penetration testing tool and how to use those tools. I decided to use the Hackerone platform because this website legally protects us to do Bug Bounty hunting for real-world web applications. The web audit reports give an excellent understanding of how to handle cybersecurity professional skills.

References

[1] Sucuri Guides' Coporate Auther, "Sucuri," 2021. [Online]. Available: https://sucuri.net/guides/owasp-top-10-security-vulnerabilities-2021/. [Accessed 11 10 2021].

[2] AAT TEAM, "All About Teating(AAT)," 21 9 2021. [Online]. Available: https://allabouttesting.org/information-gathering-techniques-for-penetration-testing/. [Accessed 13 10 2021].

[3] Github, "Sublist3r," Github, 29 6 2020. [Online]. Available: https://github.com/aboul3la/Sublist3r. [Accessed 13 10 2021].

[4] Kali linux, "WhatWeb," Kali Linux, [Online]. Available: https://www.kali.org/tools/whatweb/. [Accessed 14 10 2021].

[5] OWASP, "OWASP Cheat Sheet Series," OWASP, 2021. [Online]. Available: https://cheatsheetseries.owasp.org/cheatsheets/Clickjacking_Defense_Cheat_Sheet.html#defending-with-x-frame-options-response-headers. [Accessed 15 10 2021].

[6] MDN contributors, "MDN Web Docs," Mozilla, 13 7 2021. [Online]. Available: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-XSS-Protection. [Accessed 15 10 2021].

[7] Balbix Corporation, "Balbix," Balbix Corporation, 2020. [Online]. Available: https://www.balbix.com/insights/what-is-vulnerability-scanning/. [Accessed 17 10 2021].

[8] MDN contributors, "MDN Web Docs," Mozilla, 4 10 2021. [Online]. Available: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Content-Type-Options. [Accessed 24 10 2021].

[9] Netsparker Corporate, "What is Netsparker?," Netsparker, 2021. [Online]. Available: https://www.netsparker.com/support/what-is-netsparker/. [Accessed 27 10 2021].

[10] Edpresso Team, "What is Lodash?," Educative, [Online]. Available: https://www.educative.io/edpresso/what-is-lodash. [Accessed 25 10 2021].

[11] NGINX Corporation, "What is NGINX?," NGINX, [Online]. Available: https://www.nginx.com/resources/glossary/nginx/. [Accessed 26 10 2021].

[12] S. Gokte, "An intro to OWASP Zed Attack Proxy," Srijan, 15 12 2017. [Online]. Available: https://www.srijan.net/resources/blog/intro-owasp-zed-attack-proxy#gs.ekhbrk. [Accessed 25 10 2021].

[13] COMPUTER SECURITY RESOURCE CENTER, "Information Technology Laboratory," National Institute of Standards and Technology (NIST), [Online]. Available: https://csrc.nist.gov/glossary/term/penetration_testing. [Accessed 29 10 2021].

[14] Jimdo's corporation, "A Note About Stored XSS," Hackerone, 11 2020. [Online]. Available: https://hackerone.com/jimdo?type=team&view_policy=true. [Accessed 29 10 2021].

[15] PortSwigger, "Web Security Academy," PortSwigger, [Online]. Available: https://portswigger.net/web-security/cross-site-scripting. [Accessed 29 10 2021].

[16] Common Weaknesses Enumeration (CEW), "CWE-77: Improper Neutralization of Special

Elements used in a Command ('Command Injection')," Common Weaknesses Enumeration (CEW), 21 7 2021. [Online]. Available: http://cwe.mitre.org/data/definitions/77.html. [Accessed 30 10 2021].