UNIVERSITY OF WEST ATTICA

DEPARTMENT OF INFORMATICS AND COMPUTER ENGINEERING

CYBERSECURITY

MASTER'S THESIS

DEVELOPMENT OF A PLATFORM FOR MEASURING THE ATTACKING SURFACE OF AN ORGANIZATION, WITH OPEN SOURCE TOOLS

Sofoklis Charalampidis

I.D. 2024

Supervisor Professor
Dr. Panayotis Yannakopoulos

Master's Thesis submitted to the Department of Informatics and Computer Engineering

EGALEO, November 2022

UNIVERSITY OF WEST ATTICA

DEPARTMENT OF INFORMATICS AND COMPUTER ENGINEERING

CYBERSECURITY

The present Master's Thesis was presented

from

Sofoklis Charalampidis

I.D. 2024

on 21 Jan 2023

Members of the Committee of Inquiry including the Rapporteur

The master’s thesis was successfully examined by the following Examination Committee:

	NAME SURNAME	ACADEMIC RANK/POSITION	DIGITAL SIGNATURE
1	Panayotis Yannakopoulos	Professor Department of Informatics and Computer Engineering, University of West Attica	Panagiotis Giannakopoulos Digitally signed by Panagiotis Giannakopoulos Date: 2023.03.26 19:53:33 +03'00'
2	Spyridon Papageorgiou	Member of the Examination Committee	Spyridon Papageorgiou Signed by: Spyridon Papageorgiou Date: 2023-03-26 20:47:46 EEST
3	Stefanos Gritzalis	Professor Lab. of Systems Security, Dept. of Digital Systems, University of Piraeus	STEFANOS GKRITZALIS 26.03.2023 20:07

The approval of this Master’s Thesis doesn’t imply acceptance of the author’s opinions. The writing of this thesis complies with the principles of academic ethics.

MASTER'S THESIS AUTHOR'S DECLARATION

The undersigned Sofoklis Charalampidis of Konstantinos, with registration number 2024 student of the MSc in Cybersecurity, Department of Informatics and Computer Engineering, School of Engineering, University of West Attica, i declare responsibly that:
"I am the author of this thesis and every help I had for its preparation, is fully acknowledged and referred into the thesis. Also, any sources from which I used data, ideas or words, whether exact or paraphrased, are referred into fully, with complete reference to the authors, the publishing house or the magazine, including any sources used from the internet. I also certify that this work was written by me exclusively and is a copyrighted product of both my own and of the Institute. Violation of my above academic responsibility is a substantive reason for revocation of my degree."

The Declarant

Sofoklis Charalampidis

ABSTRACT

DEVELOPMENT OF A PLATFORM FOR MEASURING THE ATTACKING SURFACE OF AN ORGANIZATION, WITH OPEN SOURCE TOOLS.

SOFOKLIS CHARALAMPIDIS

Nowadays, organizations implement a great number of diverse web technologies and components in order to serve their needs. This fact broadens their attacking surface which is quite time consuming to be manually tested regularly. Thus, it is useful to have tools that automate these tests. This master's thesis presents the development of a platform for measuring the attacking surface of an organization with the use of open source tools. A python platform was developed that implements both passive and active information gathering tools and techniques. The aim of this platform is to imitate the reconnaissance and the scanning phase of the attacking procedure that takes place prior to every attack. As a result, the collected information can be used by an organization in order to calculate its exposure and take protection measures. The source code of the present platform is hosted on github "https://github.com/c65pt65in/reconmore.git" and can be downloaded and installed from there.

Keywords

attacking surface, reconnaissance, open source, protection measures

LIST OF FIGURES

TABLE OF CONTENTS

ACKNOWLEDGEMENTS

I would like to express my gratitude to professor Mr. Panagiotis Giannakopoulos who provided me the opportunity to participate in the present Master program and for all of his presence and assistance during the program. Furthermore, I would like to thank my supervisor Mr. Spyridon Papageorgiou for providing such an interesting project for research and knowledge growth and for all the expertise he brought during the lectures.

1. INTRODUCTION

The platform development followed the philosophy of creating a lightweight platform that would require the least computer resources and would complete its tests in a generally small amount of time. For these reasons, efforts were made for using a small number of programs and their resource consumption was taken into consideration. In addition, scripts, options or methods that potentially could cause damage to the target were excluded. Also, there is no need for any type of configuration supplied from users such as API keys or some interaction with the source code. As a result, the present platform completes its tests in approximately 50 minutes depending on the selected target and the kinds of tests.

1.1 PROBLEM DESCRIPTION

The attack surface [1.] is defined as the set of points on the boundary of a system, a system component, or an environment where an attacker can try to enter, cause an effect on or extract data from. In order to operate successfully, most of the organizations nowadays have public-facing infrastructure and applications that constitute organizations' external assets. Those assets are susceptible to attacks and so there is great need to protect them. Thus, organizations conduct attack surface analysis [2.] that helps them to: identify the assets that need to be reviewed or tested for security vulnerabilities, classify and prioritize the security risk and discover changes in the attack surface.

1.2 METHODOLOGY

The development of this master's thesis platform was done by utilizing tools and techniques from the open-source intelligence framework (OSINT) [3.], MITRE ATT&CK [4.] and methodology from the penetration testing model of the German agency Federal Office for Information Security

(BSI) [5.]. It functions as an unauthenticated user against an organization's domain without any prior knowledge about the organization's infrastructure (black-box information based). It is capable of implementing:

- Active reconnaissance [6.], which is a way of actively engaging with the target that does leave a footprint. Informations collected during this procedure have to do with the Operating System in use, the services, the ports, the software that is being used and its versions.
- Passive reconnaissance [6.], which is the gathering of information without engaging with the target and without alerting it. Thus, no possible security mechanisms are triggered.

1.3 PLATFORM SPECIFICATIONS

The platform is a python3 [7.] script named *reconmore,* which contains some bash scripts that execute different commands and tools. Its structure is as follows:

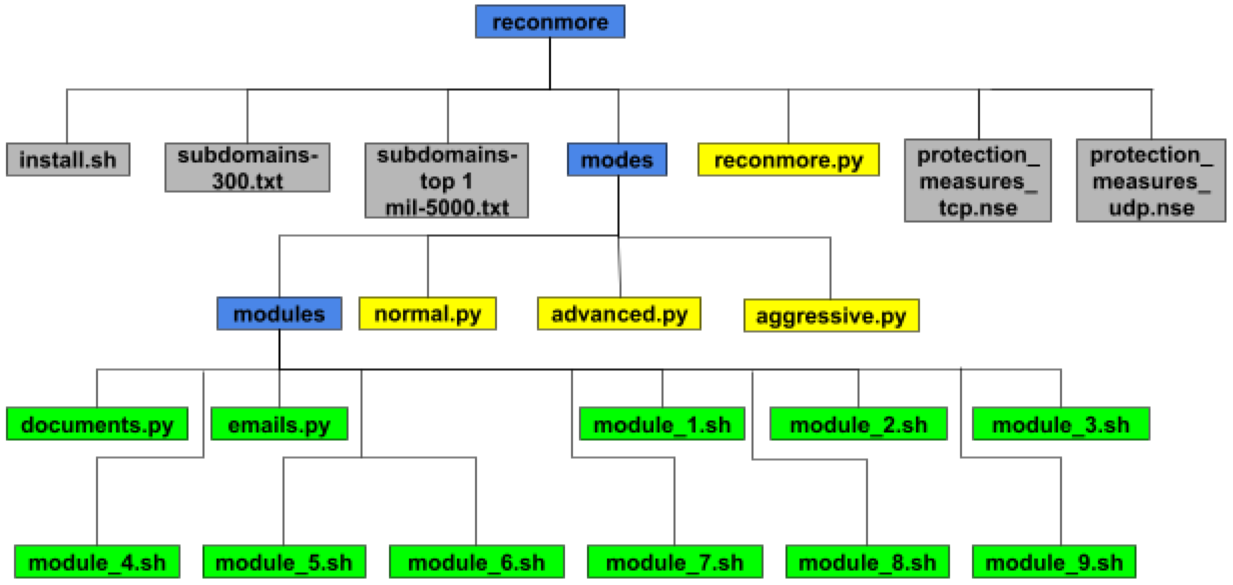


Figure 1. Platform structure.

As shown above, there are three directory levels denoted with blue colour. The first level directory, *reconmore* contains the following:

- *install.sh:* A bash script which is responsible for installing the platform.
- *subdomains-300.txt:* A text file containing 300 names for use as a wordlist during tests.
- *subdomains-top 1 mil-5000.txt:* A text file containing 5000 names for use as a wordlist during tests.
- *protection_measures_tcp.nse:* A Lua language script for use during scanning target IPs.
- *protection_measures_udp.nse:* A Lua language script for use during scanning target IPs.
- *reconmore.py:* A python script with which the users operate the platform. It is capable of executing the other python scripts *normal.py, advanced.py* and *aggressive.py* denoted with yellow colour and perform combinatorial tests. In addition, it is possible to directly execute the individual scripts *documents.py, emails.py* and *module_1* to *module_9* denoted with green colour.
- *modes:* A directory for structuring the platform hierarchy and distinguishing the basic combinatorial tests that can be performed from the individual tests.

The second level of directory, *modes* contains the following:

- *normal.py:* A python script which performs a predefined selection of tests from the *modules* directory.
- *advanced.py:* A python script which performs a predefined selection of tests from the *modules* directory.
- *aggressive.py:* A python script which performs a predefined selection of tests from the *modules* directory.
- *modules:* A directory containing all bash and python scripts for performing the tests.

The third level of directory, *modules* contains the following:

- *documents.py:* A python script that performs certain functionalities.
- *emails.py:* A python script that performs certain functionalities.
- *module_1.sh to module_9.sh:* Bash scripts that perform certain functionalities.

1.3.1 INSTALL.SH OPERATION

This bash script installs all required programs and platform dependencies, creates necessary directories, makes configurations and grants privileges. The platform is installed under the directory */usr/share/reconmore.* Every platform operation output is captured and saved in a report. All reports are saved under the directory */usr/share/reconmore/reports* in folders named after the target name and the current date and time.

1.3.2 INSTALL.SH SOURCE CODE

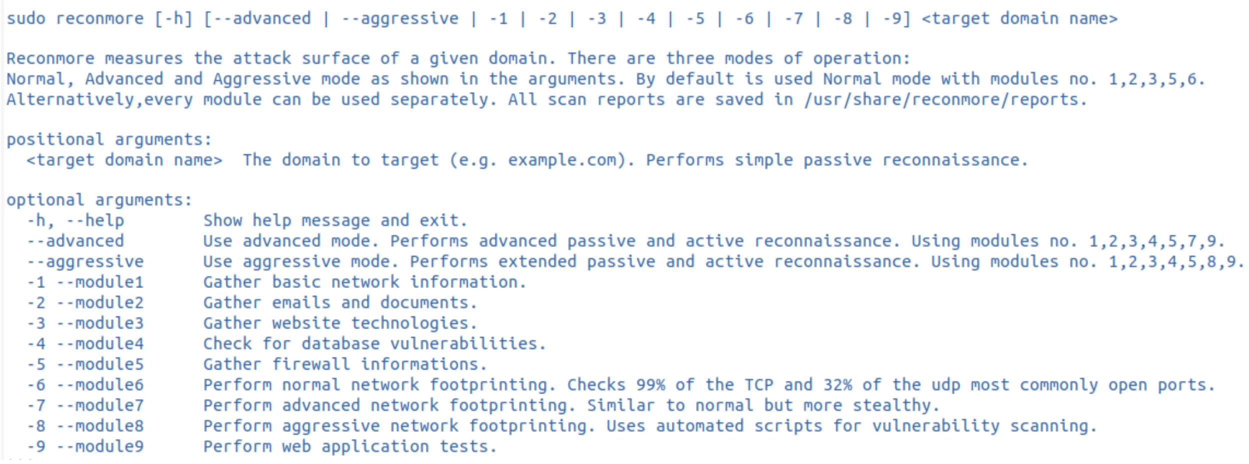
```
#!/bin/bash
if [ "$EUID" -ne 0 ]
then echo "This script must be run as root"
exit
fi
install_dir=/usr/share/reconmore
reports_dir=/usr/share/reconmore/reports
mkdir -p $install_dir 2> /dev/null
mkdir -p $reports_dir 2> /dev/null
chmod 755 -Rf $install_dir 2> /dev/null
cp -Rf * $install_dir 2> /dev/null
cd $install_dir
apt-get update -y
apt-get install git -y
pyversion=$(python3 --version)
if [[ ! "$pyversion" == *3.8* ]] || [[ ! "$pyversion" == *3.9* ]] || [[ ! "$pyversion" == *3.10* ]];
```

```
then
apt-get install python3.8 -y
fi
apt install python3-pip -y
apt install curl -y
pip3 install lxml
apt-get install dnsutils -y
apt-get install gawk -y
apt-get install whois -y
apt-get install python3-setuptools -y
git clone https://github.com/darkoperator/dnsrecon.git
cd dnsrecon
pip3 install -r requirements.txt
python3 setup.py install
cd ..
apt install gobuster -y
apt install libimage-exiftool-perl -y
apt install ruby ruby-dev -y
gem install bundler
git clone https://github.com/urbanadventurer/WhatWeb.git
cd WhatWeb
make install
cd ..
git clone https://github.com/jordanpotti/CloudScraper.git
cd CloudScraper
pip3 install -r requirements.txt
cd ..
```

```
sed -i 's/url not in base_urls:/url not in base_urls and len(base_urls) <= 100:/' /usr/share/reconmore/CloudScraper/CloudScraper.py
git clone --depth 1 https://github.com/sqlmapproject/sqlmap.git sqlmap-dev
git clone https://github.com/EnableSecurity/wafw00f.git
cd wafw00f
python3 setup.py install
cd ..
apt-get install nmap -y
wget https://github.com/Arachni/arachni/releases/download/v1.6.1.3/arachni-1.6.1.3-0.6.1.1-linux-x86_64 .tar.gz
tar -xzvf arachni-1.6.1.3-0.6.1.1-linux-x86_64.tar.gz
pip3 install --upgrade requests
chmod -R +x /usr/share/reconmore/*
ln -s /usr/share/reconmore/reconmore.py /usr/bin/reconmore
```

CHAPTER 1 - PLATFORM OPERATION

The picture below shows the help menu of the platform which indicates the necessary arguments and user input.



```
sudo reconmore [-h] [--advanced | --aggressive | -1 | -2 | -3 | -4 | -5 | -6 | -7 | -8 | -9] <target domain name>

Reconmore measures the attack surface of a given domain. There are three modes of operation:
Normal, Advanced and Aggressive mode as shown in the arguments. By default is used Normal mode with modules no. 1,2,3,5,6.
Alternatively,every module can be used separately. All scan reports are saved in /usr/share/reconmore/reports.

positional arguments:
  <target domain name>  The domain to target (e.g. example.com). Performs simple passive reconnaissance.

optional arguments:
  -h, --help        Show help message and exit.
  --advanced        Use advanced mode. Performs advanced passive and active reconnaissance. Using modules no. 1,2,3,4,5,7,9.
  --aggressive      Use aggressive mode. Performs extended passive and active reconnaissance. Using modules no. 1,2,3,4,5,8,9.
  -1 --module1      Gather basic network information.
  -2 --module2      Gather emails and documents.
  -3 --module3      Gather website technologies.
  -4 --module4      Check for database vulnerabilities.
  -5 --module5      Gather firewall informations.
  -6 --module6      Perform normal network footprinting. Checks 99% of the TCP and 32% of the udp most commonly open ports.
  -7 --module7      Perform advanced network footprinting. Similar to normal but more stealthy.
  -8 --module8      Perform aggressive network footprinting. Uses automated scripts for vulnerability scanning.
  -9 --module9      Perform web application tests.
```

Figure 2. Platform help menu.

The installation and use of the platform requires sudo privileges and python3.8+. It takes the target's domain name as a positional argument and if nothing else is specified, it operates in normal mode as described below. Before proceeding, it checks the python version installed, if the supplied domain is valid and if the domain exists. Many times security mechanisms are triggered due to heavy interaction with the servers and delay their responses, so it is possible to choose only one test module at a time to avoid such behaviour.

2.1 MODES OF OPERATION

Three modes of operation are defined, which are the following:

Normal Performs active and passive reconnaissance according to the OSINT model. It performs various checks and covers specific parts of the attacking surface. It doesn't perform checks in the website or web application. The modules used in this operation are no. 1,2,3,5,6.

Advanced Performs passive and active reconnaissance in an unobtrusive way. It performs scanning against the target regarding the most common configurations and takes into consideration possible firewalls or IDSs. Also, performs different checks against the website or web application. The modules used in this operation are no. 1,2,3,4,5,7,9.

Aggressive Performs passive and active reconnaissance in an obtrusive way. It performs scanning against the target trying every possible aspect including scanning the website or web application. It runs various scripts in order to gather information, identify vulnerabilities and possible exploits. The modules used in this operation are no.1,2,3,4,5,8,9.

2.2 RECONMORE.PY SOURCE CODE

```
#!/usr/bin/env python3
import os
import sys
import argparse
import re
import subprocess
from datetime import datetime as dt
import modes.normal
import modes.advanced
import modes.aggressive
if not os.geteuid()==0:
        sys.exit('This script must be run as root')
if sys.version_info[0]==3:
        if sys.version_info[1]==8 or sys.version_info[1]==9 or sys.version_info[1]==10:
        pass
else:
        sys.exit('This script must be run with python3.8+')
parser = argparse.ArgumentParser(description = '''Reconmore measures the attack surface of a given domain. There are three modes of operation: Normal, Advanced and Aggressive mode as shown in the arguments. By default is used Normal mode with modules no. 1,2,3,5,6. Alternatively, every module can be used separately. All scan reports are saved in /usr/share/reconmore/reports.''')
group = parser.add_mutually_exclusive_group(required=False)
parser.add_argument('domain', metavar = '<target domain name>', help = 'The domain to target (e.g. example.com). Performs simple passive reconnaissance.')
group.add_argument('--advanced', action='store_true', help = 'Use advanced mode. Performs advanced passive and active reconnaissance. Using modules no. 1,2,3,4,5,7,9.')
group.add_argument('--aggressive', action='store_true', help = 'Use aggressive mode. Performs extended passive and active reconnaissance. Using modules no. 1,2,3,4,5,8,9.')
group.add_argument('-1','--module1', action='store_true', help = 'Gather basic network information.')
group.add_argument('-2','--module2', action='store_true', help = 'Gather emails and documents.')
group.add_argument('-3','--module3', action='store_true', help = 'Gather website technologies.')
group.add_argument('-4','--module4', action='store_true', help = 'Check for database vulnerabilities.')
group.add_argument('-5','--module5', action='store_true', help = 'Gather firewall informations.')
group.add_argument('-6','--module6', action='store_true', help = 'Perform normal network footprinting. Checks 99%% of the TCP and 32%% of the udp most commonly open ports.')
group.add_argument('-7','--module7', action='store_true', help = 'Perform advanced network footprinting. Similar to normal but more stealthy.')
group.add_argument('-8','--module8', action='store_true', help = 'Perform aggressive network footprinting. Uses automated scripts for vulnerability scanning.')
group.add_argument('-9','--module9', action='store_true', help = 'Perform web application tests.')
```

```
args = parser.parse_args()
input_domain = args.domain
def operation():
    now = dt.now()
    recondatetime = dt.isoformat(now)
    filename = input_domain + recondatetime
    if args.advanced==True:
    modes.advanced.advanced_func(input_domain,filename)
    elif args.aggressive==True:
    modes.aggressive.aggressive_func(input_domain,filename)
    elif args.module1==True:
    subprocess.run(['/usr/share/reconmore/modes/modules/module_1.sh',input_domain,
filename])
    elif args.module2==True:
    subprocess.run(['/usr/share/reconmore/modes/modules/module_2.sh',input_domain,
filename])
    elif args.module3==True:
    subprocess.run(['/usr/share/reconmore/modes/modules/module_3.sh',input_domain,
filename])
    elif args.module4==True:
    subprocess.run(['/usr/share/reconmore/modes/modules/module_4.sh',input_domain,
filename])
    elif args.module5==True:
    subprocess.run(['/usr/share/reconmore/modes/modules/module_5.sh',input_domain,
filename])
    elif args.module6==True:
    subprocess.run(['/usr/share/reconmore/modes/modules/module_6.sh',input_domain,
filename])
    elif args.module7==True:
    subprocess.run(['/usr/share/reconmore/modes/modules/module_7.sh',input_domain,
filename])
    elif args.module8==True:
    subprocess.run(['/usr/share/reconmore/modes/modules/module_8.sh',input_domain,
filename])
    elif args.module9==True:
    subprocess.run(['/usr/share/reconmore/modes/modules/module_9.sh',input_domain,
filename])
    else:
    modes.normal.normal_func(input_domain,filename)
def domain_validation(input_domain):
    regex = "^((?!-)[A-Za-z0-9-]{1,63}(?<!-)\\.)+[A-Za-z]{2,6}"
    pattern = re.compile(regex)
    if(re.search(pattern, input_domain)):
      command = 'host {}'.format(input_domain)
      whois_completed_process=subprocess.run(command,stdout=subprocess.PIPE,
stderr=subprocess.PIPE, shell=True)
      if (whois_completed_process.returncode==0):
        operation()
      else:
        sys.exit('Domain doesn\'t exist.')
    else:
      sys.exit('Not valid domain.')
domain_validation(input_domain)
```

CHAPTER 2 - BASIC NETWORK INFORMATION (MODULE 1)

Starting by targeting a Domain Name, we can obtain information about associated IP addresses, IP ranges, autonomous system numbers (ASNs), geolocation data, mail addresses and phone numbers. Also, by searching records in the Domain Name System (DNS) which is a fundamental service of the IP networks it is possible to gather information about additional domains associated with the target, the use of spam filters, Sender Policy Framework (SPF) and cloud email services, zone transfers, recursion support and more. All this information may be used to carry out various attacks such as DoS attacks, data exfiltration, phishing, malware installation and network footprinting.

3.1 MODULE 1 TOOLS

The tools that are used in Module 1 are the following:

- Linux OS commands Whois [8.], Dig [9.]: The whois command uses the whois protocol and allows searching records of domain name registrars, registries and other hosts [10.,11.] in order to obtain information such as the owner of a domain, organization that registered the domain, country, registration and expiration dates, ASN numbers. The dig command is capable of querying Domain Name Servers and collecting data about all the DNS records (A,AAA,NS,CNAME,SOA,MX,TXT).
- Dnsrecon.py [12.]: It is a tool written in python that can enumerate all DNS records, check nameservers for zone transfers, enumerate service (SRV) and pointer (PTR) records, brute force host A and AAA records, perform reverse lookup in a range of IP addresses and check DNS cache for information.
- Gobuster [13.]: It is a tool written in Go that offers capability of enumerating domain directories, subdomains and more.

3.2 MODULE 1 PROTECTION MEASURES

The protection measures that Module 1 may suggest are the following:

- The procedure of gathering basic network information about a Domain cannot be easily mitigated as it takes place outside of the scope of the target's defenses and controls. Thus, the best practice should be to minimize the amount and sensitivity of the data that is available outside the target's environment.
- IP addresses and active domains should be monitored and all subdomains mapped out in order to prevent domain hijacking or subdomain takeover. More precisely, domain hijacking [14.] occurs when the registration of a domain name changes without permission of its original registrant and the subdomain takeover [15.] occurs when an attacker takes control over a subdomain due the fact that no host is providing content for it. Also, the WHOIS protection should be enabled so as to reduce the amount of sensitive data that are publicly available in the WHOIS database.
- Also, a security vulnerability occurs when allowing DNS recursion [16.], meaning the DNS server capability to search IP addresses in other DNS servers. This vulnerability may enable distributed denial-of-service (DDoS) and DNS cache poisoning attacks.
- Email security should be ensured by using the following authentication methods and protocols: Sender Policy Framework (SPF), DomainKeys Identified Mail (DKIM) and Domain-based Message Authentication, Reporting and Conformance (DMARC).
- Regarding DNS zone transfers [17.], which is the transaction of replicating DNS databases across DNS servers, it has to be allowed only between nameservers that are contained within each zone. It is a procedure that offers no authentication, so an unauthorized transfer may leak sensitive information such as server hostnames of a particular domain leading to an increased attack surface.
- The DNS security extensions protocol (DNSSEC) adds more security and protection to the DNS functionality but at the same time allows 'zone walking' which means getting

domain information and corresponding IP address from a domain. For this reason, suggesting the adoption of the DNSSEC protocol by a domain depends on the extent and sensitivity of the publicly available authoritative DNS.

3.3 MODULE 1 OPERATION

The module at first creates the directories necessary for saving the results report. Then by using *Dig* and *Whois* commands it gathers various network informations (target IPs, IP ranges, ASNs, other associated IPs). It also compares the target domain with the information gathered from associated IPs and determines which of them are highly related with the target. Then, with the use of *Dnsrecon.py* it checks the target's DNS servers for recursion and implementation of SPF and DMARC. After this, the nameservers are checked if zone transfers are allowed and then a reverse lookup brute force is performed by leveraging the pointer records (PTR). Then, in order to discover subdomains, a wordlist with common subdomains is provided and checked for their existence with the help of *Gobuster*. Also, it tries to discover subdomains by checking NSEC records if DNSSEC is implemented (zonewalking). Finally, all nameservers are queried with a subdomain list and if the answers are returned directly from their cache then hostnames related to the target are revealed (cache snooping).

3.4 MODULE 1 SOURCE CODE

```
#!/bin/bash
dir=/usr/share/reconmore/reports/$2
mkdir $dir
echo -e "\e[32mStarting Module 1-Gathering network information...\e[0m" | tee -a $dir/report
echo -e '\e[33m## BASIC NETWORK INFORMATION ##\e[0m' | tee -a $dir/report
domain_ip=$(dig +short $1)
```

```
echo -e '\e[33mIP Address\e[0m' | tee -a $dir/report
echo $domain_ip | tee -a $dir/report
echo -e '\e[33mGeneral Information\e[0m' | tee -a $dir/report
whois $domain_ip | tee -a $dir/report
echo -e '\e[33mIP Range\e[0m' | tee -a $dir/report
ip_range=$(whois -h asn.shadowserver.org origin $domain_ip | awk '{print $3}')
echo $ip_range | tee -a $dir/report
echo -e '\e[33mAutonomous System Number (ASN)\e[0m' | tee -a $dir/report
asn=$(whois -h asn.shadowserver.org origin $domain_ip | awk '{print $5}')
echo $asn | tee -a $dir/report
associated_ips=$(whois -h whois.radb.net -- "-i origin $asn" | grep -Eo "([0-9.]+){4}/[0-9]+" | head)
echo -e '\e[33mAssociated IPs\e[0m' | tee -a $dir/report
echo $associated_ips | tee -a $dir/report
echo -e '\e[33mHighly related IPs\e[0m' | tee -a $dir/report
name=$(echo $1 | cut -d '.' -f1)
for n in $(whois -h whois.radb.net -- "-i origin $asn" | grep -Eo "([0-9.]+){4}/[0-9]+" | head)
do
if [ ! -z "$(whois $n | grep $name)" ]
then
echo $n | tee -a $dir/report
fi
done
echo -e '\e[33m## DNS Information ##\e[0m' | tee -a $dir/report
if [ ! -z "$(dig $1 | grep 'rd ra')" ]
then
echo -e '\e[31mRecursion Allowed!\e[0m' | tee -a $dir/report
```

```
fi
dnsrecon -d $1 | tee -a $dir/report
if [ -z "$(cat $dir/report | grep -i 'v=spf')" ]
then
echo -e '\e[31mSender Policy Framework (SPF) protection is not implemented!\e[0m' | tee -a $dir/report
fi
if [ -z "$(cat $dir/report | grep -i 'v=dmarc')" ]
then
echo -e '\e[31mDomain-based Message Authentication, Reporting and Conformance (DMARC) protection is not implemented!\e[0m' | tee -a $dir/report
fi
echo -e '\e[33mChecking for DNS zone transfers\e[0m' | tee -a $dir/report
dnsrecon -d $1 -a | tee -a $dir/report
if cat $dir/report | grep -i 'zone transfer was successful'
then
echo -e '\e[31mZone transfers should be allowed only between name servers that are contained within each zone!\e[0m' | tee -a $dir/report
fi
echo -e '\e[33mPerforming reverse lookup brute force\e[0m' | tee -a $dir/report
dnsrecon -r $ip_range --threads 32 | tee -a $dir/report
echo -e '\e[33mSearching for subdomains\e[0m' | tee -a $dir/report
gobuster -m dns -u $1 -w /usr/share/reconmore/subdomains-top1mil-5000.txt -t 50 | tee -a $dir/report
echo -e '\e[33mTrying zonewalking\e[0m' | tee -a $dir/report
dnsrecon -d $1 -t zonewalk | tee -a $dir/report
if cat $dir/report | grep -i 'failed to answer the DNSSEC query'
```

```
then
echo -e '\e[31mConsider implementing DNSSEC protocol for more protection!\e[0m' | tee -a $dir/report
fi
echo -e '\e[33mPerforming cache snooping against all nameservers\e[0m' | tee -a $dir/report
for nameserver in $(dig -t ns $1 +noall +answer | awk '{print $5}')
do
nameserver_ip=$(dig +short $nameserver)
timeout -s 9 35s dnsrecon -t snoop --tcp -n $nameserver_ip -D /usr/share/reconmore/subdomains-300.txt 2>/dev/null | tee -a $dir/report
done
```

CHAPTER 3 - COLLECTING EMAILS AND DOCUMENTS (MODULE 2)

Valuable information can be extracted from searching user-specific data, documents and metadata from a website. Additional information can be obtained through phishing using discovered emails. Such findings may reveal usernames, login credentials, software and hardware that is used. All the above expand the attack surface, give attackers more choices and can be used to perform social engineering attacks.

4.1 MODULE 2 TOOLS

The tools that are used in Module 2 are the following:

- Emails.py [18.]: It is a custom made python web scraping script for retrieving email addresses regarding a certain domain with the help of a search engine and an OpenPGP keyserver.
- Documents.py [18.]: It is a custom made python web scraping script that collects using the Duckduckgo search engine, links to various types of files being publicly hosted on a web site.
- ExifTool [19.]: It is a tool written in Perl that is able to read metadata from different kinds of files.

4.2 MODULE 2 PROTECTION MEASURES

The protection measures that Module 2 may suggest are the following:

- A defence towards social engineering attacks could be the training and cybersecurity awareness of the organizations' staff. In addition, the protocols and methods that were mentioned in Module 1 about email security should be used.
- In every organization there should be a policy that regulates documents' publishing by requiring the erasing of their metadata.

4.3 MODULE 2 OPERATION

This module as also all the next ones, it first checks for the existence of the necessary directories and otherwise it creates them. Secondly, the script *Emails.py* is executed. This script uses the python’s *requests* module in order to perform http requests. This module visits "*https://html.duckduckgo.com/html/*" which is a free search engine and downloads email addresses with the email domain taken from the target. Also, through searching for PGP keys and signatures from *“https://keyserver.ubuntu.com/”* [20.] it recovers some more email addresses. Afterwards, the script *Documents.py* is executed. This script uses the same mechanism as *Emails.py* in order to visit "*https://html.duckduckgo.com/html/*" and by using advanced search operators, it searches documents with pdf,docx,xlsx,pptx extensions that are related to the target. Then, if such documents exist, they are downloaded. These files are passed to the exiftool so they are analyzed and their metadata are displayed.

4.4 MODULE 2 SOURCE CODE

```
#!/bin/bash
dir=/usr/share/reconmore/reports/$2
if [ ! -d $dir ]
then
mkdir $dir
fi
echo -e "\e[32mStarting Module 2-Gathering emails and documents...\e[0m" | tee -a $dir/report
mkdir -m 777 $dir/downloads 2>/dev/null
echo -e '\e[33mEmails\e[0m' | tee -a $dir/report
```

```
sudo -u $SUDO_USER python3 /usr/share/reconmore/modes/modules/emails.py $1 | tee -a $dir/report
echo -e '\e[33mDocuments\e[0m' | tee -a $dir/report
sudo -u $SUDO_USER python3 /usr/share/reconmore/modes/modules/documents.py $1 $2 | tee -a $dir/report
if [ ! -z "$(ls -A $dir/downloads)" ]
then
echo -e '\e[33mAnalysing downloaded files...\e[0m' | tee -a $dir/report
exiftool $dir/downloads | tee -a $dir/report
echo -e '\e[31mDocuments metadata should be inspected for sensitive information!\e[0m' | tee -a $dir/report
fi
```

4.4.1 EMAILS.PY SOURCE CODE

```
#!/usr/bin/env python3
import requests
import sys
import re
import random
from lxml import html
import time
def emails_func(input_domain):
    url = "https://html.duckduckgo.com/html/?"
    url_pgp = "http://keyserver.ubuntu.com/pks/lookup?"
    user_agent = "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:104.0) Gecko/20100101 Firefox/104.0"
    headers={"user-agent":user_agent,
"Accept-Language":"el-GR,el;q=0.8,en-US;q=0.5,en;q=0.3","Connection":"keep-alive", \
"Content-Type":"application/x-www-form-urlencoded","DNT":"1","Sec-Fetch-Dest":"document
","Sec-Fetch-Mode":"navigate","Sec-Fetch-Site":"same-origin", \
    "Sec-Fetch-User":"?1","TE":"trailers","Upgrade-Insecure-Requests":"1", "Host" :
"html.duckduckgo.com", "Origin" : "https://html.duckduckgo.com", \
    "Referer" : "https://html.duckduckgo.com/"}
    params = { "q": "@"+input_domain , "b" : "", "kl" : "", "df":""}
    response = requests.post(url, data=params, headers = headers)
    tree = html.fromstring(response.content)
    result_snippet = tree.xpath('//a[@class="result__snippet"]')
    emails = []
    i = 0
    while i <= 4:
    if len(result_snippet) > 0:
    results = tree.xpath('//a[@class="result__snippet"]')
    for r in results:
        results_text = r.text_content()
        regex = re.compile( '[a-zA-Z0-9.\-_+#~!$&\',;=:]+' + '@' + '[a-zA-Z0-9.-]*' +
input_domain)
        emails += regex.findall(results_text)
    form_check = tree.xpath('//div[@class="nav-link"]/form/input/@value')
    if "Previous" in form_check:
        form_names = tree.xpath('(//div[@class="nav-link"]/form)[2]/input/@name')
        form_values = tree.xpath('(//div[@class="nav-link"]/form)[2]/input/@value')
        form_values.pop(0)
    else:
        form_names = tree.xpath('//div[@class="nav-link"]/form/input/@name')
        form_values = tree.xpath('//div[@class="nav-link"]/form/input/@value')
        form_values.pop(0)
    if len(form_names) > 0:
        params = {}
        for j in range(len(form_names)):
        params.update({form_names[j]:form_values[j]})
        time.sleep(2)
        response = requests.post(url, params=params, headers = headers)
        tree = html.fromstring(response.content)
        result_snippet = tree.xpath('//a[@class="result__snippet"]')
    else:
        break
    i+=1
    params_pgp = { "op": "index", "search": "@"+input_domain }
    response_pgp = requests.get(url_pgp, params=params_pgp)
    tree_pgp = html.fromstring(response_pgp.content)
    results_pgp = tree_pgp.xpath('//span[@class="uid"]')
    if len(results_pgp) > 0:
    for r in results_pgp:
    results_text = r.text_content()
    regex = re.compile( '[a-zA-Z0-9.\-_+#~!$&\',;=:]+' + '@' + '[a-zA-Z0-9.-]*' +
input_domain)
    emails += regex.findall(results_text)
    if len(emails) > 0:
    for e in emails:
    print(e)
    else:
    print("No emails found!")
if __name__=="__main__":
    emails_func(sys.argv[1])
```

4.4.2 DOCUMENTS.PY SOURCE CODE

```
#!/usr/bin/env python3
import requests
import sys
import subprocess
import re
import random
from lxml import html
import time
def documents_func(input_domain,filename):
    download_dir="/usr/share/reconmore/reports/"+filename+"/downloads"
    url = "https://html.duckduckgo.com/html/?"
    user_agent = "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:104.0) Gecko/20100101
Firefox/104.0"
    headers={"user-agent":user_agent,
"Accept-Language":"el-GR,el;q=0.8,en-US;q=0.5,en;q=0.3","Connection":"keep-alive", \
"Content-Type":"application/x-www-form-urlencoded","DNT":"1","Sec-Fetch-Dest":"document
","Sec-Fetch-Mode":"navigate","Sec-Fetch-Site":"same-origin", \
    "Sec-Fetch-User":"?1","TE":"trailers","Upgrade-Insecure-Requests":"1","Host":
"html.duckduckgo.com", "Origin" : "https://html.duckduckgo.com", \
    "Referer" : "https://html.duckduckgo.com/"}
    links=[]
    for x in ['pdf','docx','xlsx','pptx']:
    params = { "q":  + " filetype:" + x + " site:" +    input_domain, "b" : "", "kl" : "",
"df":""}
    response = requests.post(url, data=params, headers = headers)
    tree = html.fromstring(response.content)
    result_snippet = tree.xpath('//a[@class="result__snippet"]')
    if len(result_snippet) > 0:
    results = tree.xpath('//a[@class="result__snippet"]/@href')
    links += results
    time.sleep(2)
    if len(links)>0:
    print("Documents found!", flush=True)
    print("Downloading files in " + download_dir, flush=True)
    i=0
    for link in links:
    print("Downloading file from " + link, flush=True)
    command='wget -P {} {}'.format(download_dir,link)
    subprocess.run(command, stdout=subprocess.PIPE, stderr=subprocess.PIPE, shell=True)
    i+=1
    print('Downloaded files:' + str(i), flush=True)
    else:
    print('No documents found!')

if __name__=="__main__":
    documents_func(sys.argv[1],sys.argv[2])
```

CHAPTER 4 - WEBSITE TECHNOLOGIES AND HTTP HEADERS (MODULE 3)

The reconnaissance of a domain's website can reveal the way it is built, the technologies used for that and possible weaknesses they might have. Such hints include content management systems (CMS), libraries, software version numbers, web framework modules and embedded devices. In addition, the HTTP security headers that are configured from the web server, may present vulnerabilities or leak information. By leveraging the above an attacker can better organize the attack process according to the used technologies, the possible Common Vulnerabilities and Exposures (CVEs), the cloud infrastructure and the HTTP headers. The most significant HTTP headers are the following [21.,22.].

Headers for protection against attacks:

HTTP Strict Transport Security (HSTS): Informs the web browsers to access a website only by using HTTPS instead of HTTP. In this way the communication is done via encrypted channels and man-in-the-middle attacks are prevented.

Content Security Policy (CSP): Controls the browser so as to load content and resources allowed by the policy. It prevents attacks such as cross-site scripting and other cross-site injections.

Cross-Origin-Resource-Policy (CORP): Enables browsers to block other domains from reading the response of a given resource. Helps mitigating side-channel attacks and cross-site script inclusion attacks.

X-Frame-Options (XFO): Indicates if a webpage is allowed to be rendered in a <frame>, <iframe>, <embed> or <object> in order to prevent clickjacking attacks.

X-Content-Type-Options: Forces browsers to use MIME types that are declared in the Content-Type headers. It prevents MIME type sniffing that can be taken advantage of by the attackers.

Cross-Origin-Embedder-Policy (COEP): Controls the fetching of cross-origin resources according to the permission granted.

Cross-Origin-Opener-Policy (COOP): Prevents other domains from accessing the global object of a top-level document.

Headers for leaking information:

Referrer-Policy: Controls what information is sent through the referer header.

Cache-control: Contains instructions for caching mechanisms in order to prevent exposure of information through the cache.

Clear-Site-Data: Allows clearing the browsing data such as cookies, storage or cache.

5.1 MODULE 3 TOOLS

The tools that are used in Module 3 are the following:

- Whatweb [23.]: It is a tool written in ruby that scans websites and identifies the technologies used.
- Linux OS command Curl [24.]: It is a command which supports many communication protocols and allows transferring data from or to a server.
- CloudScraper [25.]: It is a tool written in python that crawls domains in search of cloud resources.

5.2 MODULE 3 PROTECTION MEASURES

The protection measures that Module 3 may suggest are the following:

- Every piece of software and web technology that is being used should be updated to the latest version. Known vulnerabilities, bugs and security flaws are fixed by acquiring the latest software revisions as well as new features are being added.
- The HTTP security headers should be set in order to increase the web applications' security. Also, in case they are deprecated they should be substituted with the new ones.
- In case the domain uses cloud resources, they should be as well securely configured and audited. Defensive measures should be taken depending on the cloud environment that is used.

5.3 MODULE 3 OPERATION

This module uses the *Whatweb* tool and displays the web technologies and their versions with which the target's website is built and suggests checking for updates. The aggressive level (-a 3) has been set so as to make additional requests and identify more safely each finding. Then by using the *Curl* command it checks if the website's headers are in a list of specific HTTP headers that increase security and displays those that are not set. Finally, it uses the *Cloudscraper* tool, which crawls the website in search for links referring to cloud resources.

5.4 MODULE 3 SOURCE CODE

```
#!/bin/bash
dir=/usr/share/reconmore/reports/$2
if [ ! -d $dir ]
then
mkdir $dir
fi
echo -e "\e[32mStarting Module 3-Gathering website tehcnologies...\e[0m" | tee -a $dir/report
whatweb -a 3 -v $1 | tee -a $dir/report
echo -e '\e[31mSoftware versions should always be checked for newer!\e[0m' | tee -a $dir/report
allheaders=('HTTP-Strict-Transport-Security"Content-Security-Policy'
'Cross-Origin-Resource-Policy' 'X-Frame-Options' 'X-Content-Type-Options'
'Cross-Origin-Embedder-Policy' 'Cross-Origin-Opener-Policy' 'Referrer-Policy' 'Cache-Control'
'Clear-Site-Data')
echo -e '\e[33mGetting HTTP headers...\e[0m' | tee -a $dir/report
curl -s -L -I $1 | tee -a $dir/headers $dir/report
```

```
echo -e '\e[33mHTTP Headers not set:\e[0m' | tee -a $dir/report
for header in ${allheaders[@]}
do
if ! grep -iE $header $dir/headers >/dev/null
then
echo -e "\e[31m$header\e[0m" | tee -a $dir/report
fi
done
echo -e '\e[33mSearching for cloud resources...\e[0m' | tee -a $dir/report
url="https://${1}"
if ! curl -s -m 5 $url >/dev/null
then
url="http://${1}"
fi
python3 /usr/share/reconmore/CloudScraper/CloudScraper.py -u $url -v -p 4 --no-verify 2>/dev/null | tee -a $dir/report
```

CHAPTER 5 - DATABASE VULNERABILITIES (MODULE 4)

Injection attacks are one of the most frequent security risks with SQL and NoSQL injections being the most common types [26.,27.]. During these attacks, malicious statements are being inserted in the entry fields resulting in unauthorized data disclosure. In addition, these attacks could be combined with other attacks such as distributed denial-of-service (DDoS), DNS Hijacking or cross-site scripting (XSS) leading to more advanced and severe attacks.

6.1 MODULE 4 TOOLS

The tools that are used in Module 4 are the following:

- SQLmap [28.]: It is an automated tool in Python that discovers and exploits SQL injection flaws.
- Nmap [29.]: It is a tool for discovering hosts and services and exploring networks.

6.2 MODULE 4 PROTECTION MEASURES

The protection measures that Module 4 may suggest are shown below.

Protection measures against SQL injection [30.]:

- There should be input validation for every input value from users. According to this validation, the inputs are examined either client side or server side regarding their value type, length, format and other characteristics.
- Using prepared SQL statements (parameterized queries) allows the distinction between the SQL code and the supplied inputs. In this way the input data is always treated as plain text input and as a result, no code injection is possible. Also, such parameterized queries can be constructed with the use of stored procedures according to which, one or more SQL statements are grouped and built with parameters.

- Users when interacting with the databases should be granted only necessary privileges aiming to the minimum assigned privileges.

Protection measures against NoSQL injection [31.]:

- The main protection principles of mitigating SQL injection are familiar and apply as well in NoSQL. These include sanitizing user input with built in database tools and privilege isolation by using authentication and role based access control.
- The APIs should be protected by limiting the accepted requests' format.

6.3 MODULE 4 OPERATION

By using the *SQLmap* tool, the target is crawled in search of appropriate links (with GET parameters) and checks if they are vulnerable in different kinds of SQL injection. The tool's options that are set for speed and optimization are the crawl depth equal to 2 (--crawl=2), the amount of payloads and entry points (--level=3), the choice of least dangerous payloads (--risk=1), the use of persistent HTTP(s) connections (--keep-alive), the use of HEAD requests or range headers to save bandwidth (--null-connection) and the random selection of User-Agent (--random-agent). Afterwards, the *Nmap* tool is used in order to find open ports associated with NoSQL databases such as MongoDB (ports 27017, 27018, 27019, 28017), CouchDB (port 5984), Neo4J (ports 7473,7474), Redis (port 6379) and Riak(ports 8087, 8098).

6.4 MODULE 4 SOURCE CODE

```
#!/bin/bash
dir=/usr/share/reconmore/reports/$2
if [ ! -d $dir ]
then
mkdir $dir
```

```
fi
echo -e "\e[32mStarting Module 4-Checking for database vulnerabilities...\e[0m" | tee -a $dir/report
python3 /usr/share/reconmore/sqlmap-dev/sqlmap.py -u $1 --crawl=2 --batch --level=5 --risk=1 --threads=3 --keep-alive --null-connection --random-agent | tee -a $dir/report
echo -e '\e[33mChecking for open ports associated with NoSQL databases\e[0m' | tee -a $dir/report
sudo nmap -p 27017,27018,27019,28017,5984,7473,7474,6379,8087,8098 $1 -oN $dir/nosql_ports.nmap | tee -a $dir/report
var=$(cat $dir/nosql_ports.nmap | grep "open")
if [ ! -z $var ];
then
echo -e '\e[31mPorts associated with NoSQL databases discovered.\e[0m' | tee -a $dir/report
fi
```

CHAPTER 6 - FIREWALL RECONNAISSANCE (MODULE 5)

Firewalls are usually considered as the first line of defense. Their purpose is to filter and sometimes block a domain's incoming and outgoing network traffic. The way they are implemented defines if more security is achieved.

7.1 MODULE 5 TOOL

The tool that is used in Module 5 is the following:

- Wafw00f [32.]: It is a tool written in python that interacts with a domain by sending http requests and tries to identify what kind of web application firewall is implemented.

7.2 MODULE 5 PROTECTION MEASURES

The protection measures that Module 5 may suggest are the following:

- The lack of a firewall increases the risk for successful attacks. Thus, it is important that every domain has a properly configured and monitored web application firewall.

7.3 MODULE 5 OPERATION

This module uses the *Wafw00f* tool that sends different kinds of HTTP requests to the target and by analyzing the responses it determines if a web application firewall is implemented and which firewall it is.

7.4 MODULE 5 SOURCE CODE

```
#!/bin/bash
dir=/usr/share/reconmore/reports/$2
```

```
if [ ! -d $dir ]
then
mkdir $dir
fi
echo -e "\e[32mStarting Module 5-Checking for firewall...\e[0m" | tee -a $dir/report
url="https://${1}"
if ! curl -s -m 5 $url >/dev/null
then
url="http://${1}"
fi
wafw00f $url | tee -a $dir/report
```

CHAPTER 7 - NETWORK FOOTPRINTING (MODULE 6)

The security auditing and the discovery of the attacking surface includes the direct engagement with the systems. By scanning hosts in order to determine what ports are open and what services they offer, helps in determining if there are some of them unnecessary, forgotten or misconfigured. Open ports are a means of accepting connections from outside a network to the inside. Thus, they expose the outside world to the service that listens to the port inside the network.

8.1 MODULE 6 TOOL

The tool that is used in Module 6 is the Nmap [29.] as described in Module 4.

8.2 MODULE 6 PROTECTION MEASURES

More open ports lead to wider attacking surface so in order to lower security risks, their number should be the minimum and their services should be configured properly and updated. The protection measures that Module 6 may suggest regarding the most common TCP and UDP ports are the following [33.,34.,35.,36.,37.]:

- Ports 80 (HTTP), 443 (HTTPS): These ports handle the web services. It is advised to use HTTPS as it implements encryption. The web servers' versions should be the latest in order to avoid exposure to known vulnerabilities. Also, they should be configured properly so as to leak the least possible information through HTTP response headers and cookies. The web application that is served, should be set up in a secure way so that it is protected from various attacks as shown in MODULE 9.
- Port 23 (Telnet): This network protocol gives unsecure access to a network as it operates without encryption. So it is advised that this port remains closed and not used in order to avoid leaking information and avoid brute forcing attempts.

- Port 21 (FTP): It is a protocol that has many vulnerabilities such as anonymous authentication capabilities, abusing ftp server and interacting with other protocols, directory traversals and brute forcing. It is better not to be used and instead to use the SSH File Transfer Protocol (SFTP) in port 22.
- Port 22 (SSH): This protocol uses encryption and so the ssh hostkeys that are used should have adequate length and be created using strong cipher algorithms to avoid brute forcing. Also, root login should be disabled so an attacker cannot get administrative privileges directly.
- Ports 25, 465, 587 (SMTP) [38.]: Simple Mail Transfer Protocol is a protocol used for sending and receiving emails. In port 25 no encryption is supported whereas in port 587 there is the capability of upgrading connections to be secure using SSL/TLS. Thus, it is better to use port 25 for relaying (communication between mail servers), port 587 for submissions (communication between mail client and mail server) and port 465 should not be used. Moreover, the mail server should be updated so as to avoid CVE’s and configured properly to avoid information leakage. Proper configuration includes disallowing the execution of the commands EXPN, VRFY and RCPT TO and disabling NTLM authentication over HTTP if Microsoft IIS is used.
- Ports 110 (POP3), 143 (IMAP): These ports are non-encrypted so it is advised to use instead port 995(POP3S) and 993(IMAPS) respectively which operate under SSL/TLS.
- Port 3389 (RDP) [39.]: Remote Desktop Protocol provides a user with a graphical interface in order to connect to another computer through a network connection. Securing this functionality requires the enforcement of strong user passwords and setting up a lockout account policy to be safe from brute forcing attempts. Also, access to this port should be restricted by using a firewall or a VPN.
- Port 445 (Microsoft-DS) [40.,41.,42.]: The protocol that runs in this port offers SMB (Server Message Blocks) over IP meaning shared access to files, printers and ports between nodes on a network. Due to these capabilities it is not safe to be exposed

publicly. For this purpose it is advised to protect this port behind a firewall or VPN. Also, the SMB version 1 (SMBv1) should be disabled due to significant security vulnerabilities.

- Ports 137, 138, 139 (NetBIOS-SSN) [40.,41.,42.]: By using the NetBIOS protocol, these ports offer the same functionality as port 445. Additionally, considering that the NetBIOS protocol is less secure, the NetBIOS transport should be removed.
- Port 53 (DNS): The Domain Name System is generally an insecure system and its configuration is examined in Module 1.
- Port 135 (MSRPC) [43.]: Microsoft Remote Procedure Call is a protocol that provides a common interface between applications in order to allow client-server software communication. Safety measures for this port include restricting access with a firewall, updating to the latest protocol version and disallowing anonymous (Null Sessions) so that not anyone can query those interfaces and gather information.
- Port 3306 (MySQL): This port is used for communication with MySQL databases. Protection measures should focus on the configuration of the databases and specifically in granting privileges to users, requiring strong passwords and avoiding empty password for root or anonymous. Server versions should be updated to be safe against user enumeration attacks and other CVEs.
- Port 8080 (HTTP-Proxy): This port usually is used for HTTP Proxies or alternate port for web servers. Security measures for this port are the same as in port 80.
- Port 1723 (PPTP): Point-to-Point tunneling protocol can be used to implement VPNs but has a lot of known vulnerabilities and should not be used.
- Port 111 (RPCBind): It is a port that provides information between UNIX based systems and is used with NFS, NIS or any rpc-based service. Due to several vulnerabilities that have been discovered it is safer for the port to be closed if rpcs are not required or to limit its exposure with a firewall.
- Port 5900 (VNC): Virtual Network Computing provides a graphical desktop for remote controlling another computer. Depending on the versions of VNC there are some

vulnerabilities and also weak encryption implemented. To address these security issues it is advised to tunnel VNC over an SSH or VPN connection and to consider carefully which VNC application to use.

- Port 631 (IPP): The Internet Printing Protocol is a protocol that uses HTTP or HTTPS for communication between client devices and printers. Thus, it can be abused as a carrier in order to transfer malicious payloads so it is better not be exposed over the internet.
- Port 161 (SNMP): Simple Network Management Protocol is an application-layer protocol for exchanging management information between different devices in a network. This port is used by the SNMP agents to send notifications to SNMP managers. In order to stay secure, the SNMP version 3 should be preferred as it has many cryptographic security enhancements compared to previous versions and the SNMP-enabled devices should be checked for existing CVEs. Also, the SNMP agents should not use DHCP to reduce the chance of spoofing and be configured to use the SNMPv3 mode with the highest level of security.
- Port 123 (NTP): Network Time Protocol is a network protocol for time synchronization between computer systems. Most of the current NTP implementations are generally secure regarding vulnerabilities. Despite this, the information gathered from the NTP is often used for replay attacks, so it is important to implement tighter Access Control Lists (ACLs) in public facing assets or close port 123 if time synchronization is not required.
- Port 1433, 1434 (MS-SQL-S): The Microsoft SQL server is a database management system which uses port 1433 TCP as default connection port and 1434 UDP for providing information about available SQL Server instances. Protecting these ports requires setting strong passwords for users and administrators, avoiding empty passwords and implementing account lockout policies. Also, for authentication it is advised to use the mode that leverages Active Directory (AD) capabilities. If not needed, the service in port 1434 UDP should be turned off as it may inform attackers about available SQL Server instances. Finally, the SQL Server must always be kept up to date.

- Port 67 (DHCPS): The Dynamic Host Configuration Protocol Server is responsible for assigning IP addresses to devices when joining a network. During this procedure no authentication or authorization takes place making it susceptible to denial of service (DoS) attacks. To mitigate such attacks, it is recommended to enable DHCP MAC address filtering in order to provide IP addresses to a restricted list of devices or enable MAC address check so as to compare MAC addresses in DHCP requests and in frame headers.
- Port 500 (ISAKMP) [44.]: The Internet Security Association and Key Management Protocol provides a framework for authentication and key exchange. Securing the implementations of ISAKMP includes limiting access to port 500, avoiding default settings, disallowing weak cryptography and preferring the use of certificates over Pre-Shared Keys (PSK).
- Port 68 (DHCPC): This is the DHCP client port that receives the DHCP messages from the server. This client is mostly at risk from man in the middle attacks that can be mitigated by the security measures of DHCP server as mentioned above.
- Port 520 (RIP): The Routing Information Protocol is used to exchange route information between devices in a network. It is considered obsolete and in order to avoid attacks, access to port 520 should be blocked over the internet.
- Port 1900 (UPnP): Universal Plug and Play is a set of protocols that makes communication between devices easier by automating the process of device discovery and connectivity. The number of common security risks and the low necessity indicate that UpnP should be disabled.
- Port 4500 (ipsec-nat-t): Network address translation traversal is a technique that allows connections to continue working when passing through gateways with network address translation (NAT) implemented. This technique is used in IPsec VPNs due to the ISAKMP and this enforces the security measures of port 500.

- Port 514 (syslog): Syslog is a network event logging solution in UNIX-like systems according to which, syslog clients send logging events to syslog servers. If this service is exposed to the internet then it is vulnerable to possible network flooding attacks or exploits regarding the specific syslog server. Thus, this service is preferred to be used in a controlled local network environment and blocked for the open internet.
- Port 49152 (Various): It is the first port in the port range 49152-65535 which according to the Internet Assigned Numbers Authority (IANA) are used by applications as ephemeral ports. It is recommended to limit traffic in those ports with a firewall and allow open ports only for specific known services.
- Port 162 (SNMPTrap) [45.]: This port is used by the SNMP manager to receive notifications from the SNMP agents (port 161). All protection measures mentioned in port 161 should be followed in addition to proper administration, meaning the use of strong credentials and secure configuration of users, groups and privileges (principle of least privilege).
- Port 69 (TFTP): The Trivial File Transfer Protocol is a simple protocol for transferring files between clients and servers. It is generally considered not secure since it does not support any authentication or encryption. For these reasons, it is recommended not to be used in systems over the internet and instead to use more secure file transfer protocols.

8.3 MODULE 6 OPERATION

At first, this module uses the *Nmap* tool and scans the 3328 most common TCP ports (catches 99% of open TCP ports) and the 50 most common UDP ports (catches 32% of open UDP ports). In order to increase speed, it doesn’t perform ping tests to the host (-Pn) and reverse DNS resolution (-n) to IPs. The version detection (-sV) is enabled in UDP scans because it helps in determining open ports. Then, the *Nmap* Scripting Engine is used against the open ports that have been found, and runs some predefined automated scripts for getting more information on the services running in those ports. The same time, the scripts *protection_measures_tcp.nse* and *protection_measures_udp.nse* are run in order to propose protection measures for the most common ports found open.

8.4 MODULE 6 SOURCE CODE

```
#!/bin/bash
dir=/usr/share/reconmore/reports/$2
if [ ! -d $dir ]
then
mkdir $dir
fi
echo -e "\e[32mStarting Module 6-Network footprinting...\e[0m" | tee -a $dir/report
echo -e '\e[33mSearching for TCP ports\e[0m' | tee -a $dir/report
sudo nmap -n -Pn --top-ports 3328 -oN $dir/ports-tcp-3328.nmap $1 | tee -a $dir/report
echo -e '\e[33mSearching for UDP ports\e[0m' | tee -a $dir/report
sudo nmap -n -Pn -sU -sV --top-ports 50 -oN $dir/ports-udp-50.nmap $1 | tee -a $dir/report
all_tcp=$(cat $dir/ports-tcp-3328.nmap | awk -F'[/]' 'BEGIN{ORS=","} /open/{print $1}')
all_udp=$(cat $dir/ports-udp-50.nmap | awk -F'[/]' 'BEGIN{ORS=","} /open/{print $1}')
echo -e '\e[33mUsing automated scripts to discover common security issues in tcp ports\e[0m' | tee -a $dir/report
if [ ! -z "$all_tcp" ]
then
sudo nmap -n -Pn --script default,/usr/share/reconmore/protection_measures_tcp.nse -sV -p $all_tcp $1 | tee -a $dir/report
fi
```

```
echo -e '\e[33mUsing automated scripts to discover common security issues in udp ports\e[0m' |
tee -a $dir/report
if [ ! -z "$all_udp" ]
then
sudo nmap -n -Pn -sU --script default,/usr/share/reconmore/protection_measures_udp.nse -sV -p
$all_udp $1 | tee -a $dir/report
fi
echo -e "\e[31mReport saved at $dir.\e[0m"
```

8.5 LUA SCRIPTS FOR NMAP

The protection measures regarding the most common ports are presented after the scan with the use of two *Nmap* scripts (protection_measures_tcp.nse, protection_measures_udp.nse) written in Lua language [46.] with the help of Nmap Scripting Engine libraries.

8.5.1 PROTECTION_MEASURES_TCP.NSE SOURCE CODE

```
--HEAD--
description = [[This script displays the protection measures regarding the most common open tcp
ports.]]
author = "S.C."
local shortport = require "shortport"
--RULE--
portrule=
shortport.portnumber({80,443,23,21,22,25,465,587,110,143,3389,445,139,53,135,3306,8080,172
3,111,5900,1433,49152},"tcp")
--ACTION--
```

```
action = function (host,port)
    if port.number == 80 then
    print("\27[31mPORT 80: It is advised that port 443 (SSL/TLS encryption) is used instead, the web server software is updated and the website or web application is examined for common vulnerabilities! \27[0m")
    elseif port.number == 443 then
    print("\27[31mPORT 443: It is advised that the web server software is updated and the website or web application is examined for common vulnerabilities!\27[0m")
    elseif port.number == 23 then
    print("\27[31mPORT 23: It is advised that this port should be closed!\27[0m")
    elseif port.number == 21 then
    print("\27[31mPORT 21: It is advised that this port should be closed and instead port 22 (SFTP) is used!\27[0m")
    elseif port.number == 22 then
    print("\27[31mPORT 22: It is advised to use strong cipher algorithms and disable root login!\27[0m")
    elseif port.number == 25 then
    print("\27[31mPORT 25: It is advised that this port is used only for communication between mail servers!\27[0m")
    elseif port.number == 465 then
    print("\27[31mPORT 465: It is advised that this port is not used.\27[0m")
    elseif port.number == 587 then
    print("\27[31mPORT 587: It is advised that this port is only used for communication between  mail clients and mail servers!\27[0m")
    elseif port.number == 110 then
    print("\27[31mPORT 110: It is advised that this port should be closed and port 995(POP3S) is used instead.\27[0m")
```

```
elseif port.number == 143 then
print("\27[31mPORT 143: It is advised that this port should be closed and port 993(IMAPS) is used instead!\27[0m")
elseif port.number == 3389 then
print("\27[31mPORT 3389: It is advised that access to this port is restricted by a firewall or VPN, strong user passwords are used and a lockout account policy is set!\27[0m")
elseif port.number == 445 then
print("\27[31mPORT 445: It is advised that access to this port is restricted by a firewall or VPN and SMB version 1 is disabled!\27[0m")
elseif port.number == 139 then
print("\27[31mPORT 139: It is advised that this port should be closed and SMB in port 445 is used instead!\27[0m")
elseif port.number == 53 then
print("\27[31mPORT 53: It is advised to examine DNS configuration with appropriate tools!\27[0m")
elseif port.number == 135 then
print("\27[31mPORT 135: It is advised that access is restricted to this port by a firewall, the latest protocol version is used and anonymous (Null Sessions) are disallowed!\27[0m")
elseif port.number == 3306 then
print("\27[31mPORT 3306: It is advised that MySQL server version is updated and properly configured regarding granting privileges to users, requiring strong passwords and avoiding empty password for root or anonymous!\27[0m")
elseif port.number == 8080 then
print("\27[31mPORT 8080: It is advised that port 443(SSL/TLS encryption) is used instead!\27[0m")
elseif port.number == 1723 then
```

```
        print("\27[31mPORT 1723: It is advised that this port should be closed and not used for VPN implementation!\27[0m")
        elseif port.number == 111 then
        print("\27[31mPORT 111: It is advised that this port should be closed otherwise access should be restricted!\27[0m")
        elseif port.number == 5900 then
        print("\27[31mPORT 5900: It is advised to tunnel VNC over an SSH or VPN connection and be careful which VNC application is used!\27[0m")
        elseif port.number == 1433 then
        print("\27[31mPORT 1433: It is advised that access to this port is protected by setting strong passwords for users and administrators, avoiding empty passwords and implementing account lockout policy. Also, for authentication it is advised to use the mode that leverages Active Directory (AD) capabilities!\27[0m")
        elseif port.number >= 49152 and port.number <= 65535 then
        print("\27[31mPORT ".. port.number ..":".."It is advised that traffic in this port is limited with a firewall!\27[0m")
        end
        end
```

8.5.2 PROTECTION_MEASURES_UDP.NSE SOURCE CODE

```
--HEAD--
description = [[This script displays the protection measures regarding the most common open udp ports.]]
author = "S.C."
local shortport = require "shortport"
--RULE--
```

```
portrule = shortport.portnumber({137,138,631,161,123,1434,67,500,68,520,1900,4500,514,49152,162,69}, "udp")
--ACTION--
action = function (host,port)
        if port.number == 137 then
        print("\27[31mPORT 137: It is advised that this port should be closed and SMB in port 445 is used instead!\27[0m")
        elseif port.number == 138 then
        print("\27[31mPORT 138: It is advised that this port should be closed and SMB in port 445 is used instead!\27[0m")
        elseif port.number == 631 then
        print("\27[31mPORT 631: It is advised that this port is not exposed to the internet!\27[0m")
        elseif port.number == 161 then
        print("\27[31mPORT 161: It is advised that this port uses SNMPv3 with the highest level of security!\27[0m")
        elseif port.number == 123 then
        print("\27[31mPORT 123: It is advised that this port should be closed if time synchronization is not required otherwise access should be restricted!\27[0m")
        elseif port.number == 1434 then
        print("\27[31mPORT 1434: It is advised that this port should be closed if not needed for discovering SQL Server instances!\27[0m")
        elseif port.number == 67 then
        print("\27[31mPORT 67: It is advised to that DHCP MAC address filtering and MAC address check are enabled!\27[0m")
        elseif port.number == 500 then
```

```
print("\27[31mPORT 500: It is advised that access to this port is limited, the default settings are removed and certificates are used for authentication!\27[0m")
elseif port.number == 68 then
print("\27[31mPORT 68: It is advised to that DHCP MAC address filtering and MAC address check are enabled in the DHCP server!\27[0m")
elseif port.number == 520 then
print("\27[31mPORT 520: It is advised that this port should be closed!\27[0m")
elseif port.number == 1900 then
print("\27[31mPORT 1900: It is advised that this port should be closed!\27[0m")
elseif port.number == 4500 then
print("\27[31mPORT 4500: It is advised that access to this port is limited, the default settings are removed and certificates are used for authentication!\27[0m")
elseif port.number == 514 then
print("\27[31mPORT 514: It is advised that this port is open only in a controlled local network environment!\27[0m")
elseif port.number >= 49152 and port.number <= 65535 then
print("\27[31mPORT ".. port.number ..":".."It is advised that traffic in this port is limited with a firewall!\27[0m")
elseif port.number == 162 then
print("\27[31mPORT 162: It is advised that this port uses SNMPv3 with the highest level of security and also strong credentials and secure configuration of users, groups and privileges are required!\27[0m")
elseif port.number == 69 then
print("\27[31mPORT 69: It is advised that this port should be closed!\27[0m")
end
end
```

CHAPTER 8 - ADVANCED NETWORK FOOTPRINTING (MODULE 7)

It is quite common for companies to implement firewalls for restricting access and IDSs for monitoring traffic in order to protect their networks. By differentiating the behaviour of *Nmap* (Module 6) and making it more ‘quiet’ and less predictable, it is possible to test and verify if these mechanisms are fully functional.

9.1 MODULE 7 TOOL

The tool that is used in Module 7 is the Nmap [29.] as described in Module 4.

9.2 MODULE 7 PROTECTION MEASURES

The protection measures that Module 7 may suggest are the same as described in Module 6.

9.3 MODULE 7 OPERATION

The present module performs the same operations as Module 6 regarding the use of *Nmap* tool, the number of ports scanned and the automated scripts used. The only difference is that it performs the scanning phase for open ports more stealthy by fragmenting the IP packets in 16-byte fragments (--mtu 16) and also by appending 30-byte random data to the IP packets (--data-length 30) so as to be harder for those to be detected.

9.4 MODULE 7 SOURCE CODE

```
#!/bin/bash
dir=/usr/share/reconmore/reports/$2
if [ ! -d $dir ]
```

```
then
mkdir $dir
fi
echo -e "\e[32mStarting Module 7-Network footprinting...\e[0m" | tee -a $dir/report
echo -e '\e[33mSearching for TCP ports\e[0m' | tee -a $dir/report
sudo nmap --mtu 16 --data-length 30 -n -Pn --top-ports 3328 -oN $dir/ports-tcp-3328.nmap $1 | tee -a $dir/report
echo -e '\e[33mSearching for UDP ports\e[0m' | tee -a $dir/report
sudo nmap --mtu 16 --data-length 30 -n -Pn -sU -sV --top-ports 50 -oN $dir/ports-udp-50.nmap $1 | tee -a $dir/report
all_tcp=$(cat $dir/ports-tcp-3328.nmap | awk -F'[/]' 'BEGIN{ORS=","} /open/{print $1}')
all_udp=$(cat $dir/ports-udp-50.nmap | awk -F'[/]' 'BEGIN{ORS=","} /open/{print $1}')
echo -e '\e[33mUsing automated scripts to discover common security issues in tcp ports\e[0m' | tee -a $dir/report
if [ ! -z "$all_tcp" ]
then
sudo nmap -n -Pn --script default,/usr/share/reconmore/protection_measures_tcp.nse -sV -p $all_tcp $1 | tee -a $dir/report
fi
echo -e '\e[33mUsing automated scripts to discover common security issues in udp ports\e[0m' | tee -a $dir/report
if [ ! -z "$all_udp" ]
then
sudo nmap -n -Pn -sU --script default,/usr/share/reconmore/protection_measures_udp.nse -sV -p $all_udp $1 | tee -a $dir/report
fi
```

CHAPTER 9 - AGGRESSIVE NETWORK FOOTPRINTING (MODULE 8)

In addition to the network discovery process it is convenient and more efficient to test the findings for common security issues. This can be achieved by using automated scripts that are available in *Nmap* (Module 6) and are able to perform basic security auditing tasks. These scripts are classified in the following categories:

- auth: This category contains scripts that handle services which require authentication credentials. They attempt connection with default credentials, no credentials or try bypassing them.
- brute: These scripts perform brute forcing against several protocols using predefined wordlists.
- discovery: In this category the scripts actively retrieve information by querying the discovered services.
- exploit: These scripts try to exploit vulnerabilities if they exist.
- intrusive: These scripts perform their actions very aggressively to the extent of causing possible malfunction of the target system.
- malware: These scripts observe the behaviour of the target system and try to determine if it is infected by malware.
- safe: These scripts perform general network discovery and their operation is unlikely to cause undesired effects to the target.
- version: These scripts expand the capabilities of version detection in order to determine the versions of the discovered assets.
- vuln: In this category the scripts check the findings for known vulnerabilities.

10.1 MODULE 8 TOOL

The tool that is used in Module 8 is the Nmap [29.] as described in Module 4.

10.2 MODULE 8 PROTECTION MEASURES

The protection measures that Module 8 may suggest are the same as described in Module 6. Also, the output of the automated scripts may indicate situations which demand additional measures.

10.3 MODULE 8 OPERATION

This module uses the *Nmap* tool to scan all 65.535 TCP and UDP ports of the target and determine which are open. In order to reduce scan times, the wait time for non-responsive ports, due to rate limits in packet responses, is minimized (--defeat-rst-ratelimit, --defeat-icmp-ratelimit). Also, regarding UDP scans which take longer than TCP, the maximum retransmitting time for probes is set to 300 milliseconds (--max-rtt-timeout 300ms), the maximum times for retransmitting probes is 10 (--max-retries 10) and the maximum delay between probes is 1000 milliseconds (--max-scan-delay 1000). In addition, the version-intensity is set to 0 (--version-intensity 0) so as only the most effective probes are sent. After open ports have been found, the set of automated scripts that are above mentioned are run against those ports with maximum run time for each script the 10 minutes (--script-timeout 10m). The previous set of scripts excludes scripts that can cause damage to the target, increase scan time or are not so useful (--script "not dos and not broadcast and not external and not fuzzer").

10.4 MODULE 8 SOURCE CODE

```
#!/bin/bash
dir=/usr/share/reconmore/reports/$2
if [ ! -d $dir ]
```

```
then
mkdir $dir
fi
echo -e "\e[32mStarting Module 8-Network footprinting...\e[0m" | tee -a $dir/report
echo -e '\e[33mSearching for TCP ports\e[0m' | tee -a $dir/report
sudo nmap --script /usr/share/reconmore/protection_measures_tcp.nse -n -Pn -p-
--defeat-rst-ratelimit -oN all-ports-tcp.nmap $1 | tee -a $dir/report
echo -e '\e[33mSearching for UDP ports\e[0m' | tee -a $dir/report
sudo nmap --script /usr/share/reconmore/protection_measures_udp.nse -n -Pn -sU -sV -p-
--version-intensity 0 --max-rtt-timeout 300ms --max-retries 10 --max-scan-delay 1000ms
--defeat-icmp-ratelimit -oN all-ports-udp.nmap $1 | tee -a $dir/report
all_tcp=$(cat all-ports-tcp.nmap | awk -F'[/]' 'BEGIN{ORS=","} /open/{print $1}')
all_udp=$(cat all-ports-udp.nmap | awk -F'[/]' 'BEGIN{ORS=","} /open/{print $1}')
echo -e '\e[33mUsing automated scripts to discover common security issues in TCP ports\e[0m' |
tee -a $dir/report
if [ ! -z "$all_tcp" ]
then
sudo nmap --script "not dos and not broadcast and not external and not fuzzer" --script-timeout
10m -n -Pn -sV -p $all_tcp $1 | tee -a $dir/report
fi
echo -e '\e[33mUsing automated scripts to discover common security issues in UDP ports\e[0m' |
tee -a $dir/report
if [ ! -z "$all_udp" ]
then
sudo nmap --script "not dos and not broadcast and not external and not fuzzer" --script-timeout
10m --version-intensity 0 --max-rtt-timeout 300ms --max-retries 10 --max-scan-delay 10 -n -Pn
-sU -sV -p $all_udp $1 | tee -a $dir/report
```

fi

CHAPTER 10 - WEB APPLICATION SCANNING (MODULE 9)

The existence of a website or a web application hosted on a domain broadens the attack surface. Many attacks occur by leveraging the above's security flaws such as unauthorized access to files, input validation and generally the dynamic interaction with the users. The components that are mainly targeted by the attacks are various links, different kinds of forms, user-interface inputs, cookies, JSON and XML request data. The attacks can be described in the following categories [47.,48.]:

1. Cross-Site Request Forgery (CSRF): The attacker lures an authenticated user of a web application to engage with requests that include malicious URL parameters, cookies or other data. In case the validation mechanism of users' requests is weak, the web server executes and performs an unwanted action [49.,50.,51.].
2. Code injection: The attacker is allowed to inject untrusted code to the server and get it executed [52.,53.].
3. Blind code injection using timing: The attacker is able to inject code to the server that adds a delay in execution in order to determine if a certain vulnerability is present.
4. LDAP injection: The web application builds LDAP statements based on user inputs that are not properly sanitized and as a result the attacker manages to query or modify the LDAP tree [54.,55.].
5. Path traversal: The attacker is allowed to access files and directories outside the web applications' working directory, by changing the parameter values of a path to file that is being called by the server [56.].
6. Local File inclusion: The attacker manages to substitute the path value to files on the server and as a result read or execute them.
7. Http response splitting: The attacker inserts malicious data into the web application and those are included in the HTTP response header. As a result the response can be manipulated allowing other types of attacks [57.].

8. OS command injection: The attacker is allowed to execute arbitrary operating system (OS) commands gaining full control over the web application [58.,59.].
9. Blind OS command injection using timing: The attacker manages to detect an OS command injection vulnerability by executing code that causes time delay although its result is not returned in the application's responses.
10. Remote file inclusion: The attacker manages to substitute the path value to files on the server with the address of a remote source and as a result execute them on the server.
11. Unvalidated redirects: The attacker is able to modify the redirection address of a request and control the location of redirection [60.].
12. Xpath injection: Xpath is used to query data stored in XML. Thus, when user input is used to construct such queries, an attacker may supply malicious inputs and gain unauthorized access to sensitive data [61.].
13. Cross-site scripting (XSS): The attacker manages to inject malicious scripts to the web pages of a website or web application and those are returned to the victim user and lead to disclosure of session tokens, cookies and other sensitive data. There are different kinds of XSS attacks and they are distinguished by the fact that they are stored in the server or not (stored or reflected XSS) and if the server does not participate in the attack at all (DOM based XSS) [62.,63.,64.].
14. Source code disclosure: The server side code which must not be disclosed to users is often sent to them due to configuration errors or due to crafted requests by an attacker [65.].
15. XML External Entity (XXE): In case the web application exchanges data with the server using XML format, there is the external entity feature that represents items in the data located outside the XML document. Thus, an attacker can leverage this feature and modify the submitted XML performing malicious actions and different kinds of attacks if the XML parser is not configured properly [66.].

16. Back-up files: An attacker may retrieve valuable information from backup files that are accidentally left in the web server.
17. Back-up directories: An attacker may locate sensitive back-up files by searching back-up directories.
18. Common administration interfaces: The administrator interfaces provide a direct attack surface which if exploited may compromise the whole website or web application.
19. Common directories: Directories that are not used or are obsolete may assist an attacker during the information gathering phase.
20. HTTP PUT: The PUT method is used to upload data to a server and due to its capability to modify resources it is considered unsafe [67.].
21. Insufficient Transport Layer Protection for password forms: If the HTTP protocol is used without encryption (HTTPS) then an attacker can intercept packets and steal credentials.
22. WebDAVdetection: Web Distributed Authoring and Versioning allows basic file management from a client to a web server giving the opportunity to an attacker to extract information from reading files or upload malicious ones.
23. HTTP TRACE detection: The HTTP TRACE method allows a client to send a request to a server and then have this request back as a response. This functionality may be leveraged by an attacker in order to perform different kinds of attacks.
24. CVS/SVN user disclosure: Concurrent Version System (CVS) and Subversion (SVN) are version control systems. Attackers often search for CVS/SVN files in a website in order to gather as much information as possible about the target.
25. .htaccess Limit misconfiguration: In Apache’s ‘.htaccess’ file the ‘Limit’ directive indicates which HTTP methods to be blocked on the web server. Each method has a different level of risk so attackers search which methods are allowed.
26. Interesting responses: Responses that are not 200 (OK) or 404 (Not Found) may give the attacker useful information about the behaviour of a web application.

27. HTML object grepper: The HTML objects allow the execution of external sources offering ways for XSS attacks.
28. Mixed Resource/Scripting: If not all the accessed resources and the interaction with the web application are done through HTTPS then attackers can intercept data transferred via HTTP.
29. Insecure cookies: Cookies that do not have the ‘Secure’ attribute, are sent with unsecured HTTP and so the attackers can easily access them with man-in-the-middle attacks [68.].
30. HttpOnly cookies: Cookies with the ‘HTTPOnly’ attribute set, are inaccessible by client side scripts which acts as a precaution against XSS attacks [68.].
31. Auto-complete for password form fields: If autocomplete is enabled in HTML forms and especially for usernames and passwords, then the values that have been entered are cached in the browser. As a result, if an attacker has access to that certain computer then is able to use these credentials to interact with the website or web application.
32. Form-based upload: The capability of uploading files through HTML forms requires strict security controls because it can be leveraged in many ways by attackers [69.].
33. Cookie set for parent domain: The ‘Domain’ attribute in cookies specifies which hosts can receive the cookies and if it is set then less trusted subdomains will be included [70.].

11.1 MODULE 9 TOOL

The tool that is used in Module 9 is the following:

- Arachni [48.]: It is a security scanner for web applications which performs different kinds of checks.

11.2 MODULE 9 PROTECTION MEASURES

The protection measures that Module 9 may suggest are the following:

- User requests about very important actions should include CSRF tokens that are random, tied to the users' session and validated properly from the server. (Attack 1)
- Untrusted user inputs should never be processed by the server and in case this cannot be avoided then there should be strict validation by whitelisting specific values. (Attacks 2,3)
- Untrusted data in LDAP queries must be escaped, frameworks that protect from LDAP injection can be used, the principle of least privilege should be followed and there should be a list input validation prior to the execution of LDAP queries. (Attack 4)
- User input should not be used for file location calls, there should be a whitelist of permitted files and sensitive files should not be stored in the web root. (Attacks 5,6,10)
- Untrusted data should never be used to form the contents of a response header. (Attack 7)
- The direct execution of operating system commands from inside the web application must be avoided. If it cannot be avoided then this functionality should be performed by secure APIs with strong input validation. (Attacks 8,9)
- Redirection functions with inputs should be replaced by direct links and a list with allowed redirection URLs should be maintained and checked server-side. (Attack 11)
- User input should be validated and in most cases should include only alphanumeric strings. (Attack 12)
- Untrusted user inputs must not be placed directly to a script or in css, inside an HTML comment, an attribute name or a tag name. In specific cases where untrusted data is output in HTML or javascript contexts it should be encoded in order to avoid being interpreted as active content. Also, XSS attacks may be prevented by using appropriate HTTP response headers such as "Content-Type", "X-Content-Type-Options" and "Content-Security-Policy" (CSP). (Attacks 13,27)
- Sensitive application files must have proper permissions that prevent access from public users and they must not be placed inside the web root. (Attack 14)

- The XML features that are not intended to be used by the XML processor should be disabled and disable the support of external entities. (Attack 15)
- Backup files should not be kept inside the web root. (Attack 16)
- Unnecessary directories should not be available in a website or web application. (Attack 17)
- The administrator interfaces should be visited only by those in allowed access control lists. (Attack 18)
- Directories that are not used should be removed. (Attack 19)
- The HTTP PUT method should be disabled on the server. (Attack 20)
- The website or web application should function with the latest secure encryption protocols (SSL,TLS). (Attack 21)
- WebDAV provides functionalities that are too risky and not so secure and it should be disabled. (Attack 22)
- The HTTP TRACE method in most cases is not needed and should be disabled. (Attack 23)
- SVN/CVS files should be removed prior to development. (Attack 24)
- It is better to follow a whitelist approach with the ‘LimitExcept’directive in order not to forget which methods to disable by blacklisting. (Attack 25)
- Depending on which HTTP response is returned, they should leak the least information. (Attack 26)
- All pages and resources should use HTTPS. (Attack 28)
- Cookies that contain sensitive information should have the ‘Security’ attribute set in order to only be sent when HTTPS is used. (Attack 29)
- If cookies don’t need to be accessed by any client side scripts then the HTTPOnly attribute should be set. (Attack 30)
- The ‘autocomplete’ attribute in HTML forms should be disabled. (Attack 31)
- The files should be validated prior to the upload and if possible a framework is better to be used rather than a custom validation mechanism. They should be validated regarding

the extension, name and size and the preferred HTTP upload method should be ‘POST’. (Attack 32)

- In most situations it is recommended to omit the ‘Domain’ attribute in cookies. (Attack 33)

11.3 MODULE 9 OPERATION

The *Arachni* tool is used against the web application of the target and launches, wherever possible, the above-mentioned attacks. Every successful discovery of some issue is displayed (--output-only-positives) and then the related protection measures follow. The need for efficiency and speed demands that the attacks are limited until the fifth level of the directory depth (--scope-directory-depth-limit 5) and the maximum wait time for server response is 10000 milliseconds (--http-request-timeout 10000).

11.4 MODULE 9 SOURCE CODE

```
#!/bin/bash
dir=/usr/share/reconmore/reports/$2
if [ ! -d $dir ]
then
mkdir $dir
fi
echo -e "\e[32mStarting Module 9-Checking web application vulnerabilities...\e[0m" | tee -a $dir/report
path="/usr/share/reconmore/arachni-1.6.1.3-0.6.1.1/bin/arachni"
url="https://${1}"
if ! curl -s -m 5 $url >/dev/null
```

```
then
url="http://${1}"
fi
sudo -u $SUDO_USER $path --output-only-positives --browser-cluster-pool-size 6
--browser-cluster-ignore-images --report-save-path /tmp/arachni.afr $url
--checks=csrf,code-injection*,ldap_injection,path_traversal,
file_inclusion,response_splitting,os_cmd-injection*,rfi,unvalidated_redirect*,xpath_injection,xss
*,source_code_disclosure,xxe,backup_files,backup_directories,common_admin_interfaces,com
mon_directories,http_put,unencrypted_password_forms,webdav,xst,cvs_svn_users,htaccess_limit
,interesting_responses,html_objects,mixed_resource,insecure_cookies,http_only_cookies,passwo
rd_autocomplete,form_upload,cookie_set_for_parent_domain --scope-directory-depth-limit 5
2>/dev/null
sudo -u $SUDO_USER /usr/share/reconmore/arachni-1.6.1.3-0.6.1.1/bin/arachni_reporter
/tmp/arachni.afr >> $dir/report
sudo -u $SUDO_USER rm -f /tmp/arachni.afr
report=$(cat $dir/report)
if [[ $report == *"Cross-Site Request Forgery"* ]]
then
echo -e "\e[31mCross-Site Request Forgery: User requests about very important actions should
include CSRF tokens that are random, tied to the users session and validated properly from the
server!\e[0m" | tee -a $dir/report
fi
if [[ $report == *"Code injection"* ]] || [[ $report == *"Code injection (timing attack)"* ]]
then
echo -e "\e[31mCode injection: Untrusted user inputs should never be processed by the server
and in case this cannot be avoided then there should be strict validation by whitelisting specific
values!\e[0m" | tee -a $dir/report
```

```
fi
if [[ $report == *"LDAP Injection"* ]]
then
echo -e "\e[31mLDAP Injection: Untrusted data in LDAP queries must be escaped, frameworks that protect from LDAP injection can be used, the principle of least privilege should be followed and there should be a list input validation prior to the execution of LDAP query!\e[0m" | tee -a $dir/report
fi
if [[ $report == *"Path Traversal"* ]]
then
echo -e "\e[31mPath Traversal: User input should not be used for file location calls, there should be a whitelist of permitted files and sensitive files should not be stored in the web root!\e[0m" | tee -a $dir/report
fi
if [[ $report == *"File Inclusion"* ]]
then
echo -e "\e[31mFile Inclusion: User input should not be used for file location calls, there should be a whitelist of permitted files and sensitive files should not be stored in the web root!\e[0m" | tee -a $dir/report
fi
if [[ $report == *"Response Splitting"* ]]
then
echo -e "\e[31mResponse Splitting: Untrusted data should never be used to form the contents of a response header!\e[0m" | tee -a $dir/report
fi
if [[ $report == *"Operating system command injection"* ]] || [[ $report == *"Operating system command injection (timing attack)"* ]]
```

```
then
echo -e "\e[31mOperating system command injection: The direct execution of operating system commands from inside the web application must be avoided. If it cannot be avoided then this functionality should be performed by secure APIs with strong input validation!\e[0m" | tee -a $dir/report
fi
if [[ $report == *"Remote File Inclusion"* ]]
then
echo -e "\e[31mRemote File Inclusion: User input should not be used for file location calls, there should be a whitelist of permitted files and sensitive files should not be stored in the web root!\e[0m" | tee -a $dir/report
fi
if [[ $report == *"Unvalidated redirect"* ]] || [[ $report == *"Unvalidated DOM redirect"* ]]
then
echo -e "\e[31mUnvalidated redirect: Redirection functions with inputs should be replaced by direct links and a list with allowed redirection URLs should be maintained and checked server-side!\e[0m" | tee -a $dir/report
fi
if [[ $report == *"XPath Injection"* ]]
then
echo -e "\e[31mXPath Injection: User input should be validated and in most cases should include only alphanumeric strings!\e[0m" | tee -a $dir/report
fi
if [[ $report == *"Cross-Site Scripting (XSS)"* ]] || [[ $report == *"DOM-based Cross-Site Scripting (XSS)"* ]]|| [[ $report == *"DOM-based Cross-Site Scripting (XSS) in script context"* ]] || [[ $report == *"Cross-Site Scripting (XSS) in event tag of HTML element"* ]] || [[ $report
```

```
== *"Cross-Site Scripting (XSS) in path"* ]] || [[ $report == *"Cross-Site Scripting (XSS) in script context"* ]] || [[ $report == *"Cross-Site Scripting (XSS) in HTML tag"* ]]
then
echo -e "\e[31mCross-Site Scripting (XSS): Untrusted user inputs must not placed directly to a script or in css, inside an HTML comment, an attribute name or a tag name. In specific cases where untrusted data is output in HTML or javascript contexts it should be encoded in order to avoid being interpreted as active content. Also, XSS attacks may be prevented by using appropriate HTTP response headers such as “Content-Type”,“X-Content-Type-Options” and Content-Security-Policy (CSP)!\e[0m" | tee -a $dir/report
fi
if [[ $report == *"Source code disclosure"* ]]
then
echo -e "\e[31mSource code disclosure: Sensitive application files must have proper permissions that prevent access from public users and they must not placed inside the web root!\e[0m" | tee -a $dir/report
fi
if [[ $report == *"XML External Entity"* ]]
then
echo -e "\e[31mXML External Entity: The XML features that are not intended to be used by the XML processor should be disabled and disable the support of external entities!\e[0m" | tee -a $dir/report
fi
if [[ $report == *"Backup file"* ]]
then
echo -e "\e[31mBackup file: Backup files should not be kept inside the web root!\e[0m" | tee -a $dir/report
fi
```

```
if [[ $report == *"Backup directory"* ]]
then
echo -e "\e[31mBackup directory: Unnecessary directories should not be available in a website or web application!\e[0m" | tee -a $dir/report
fi
if [[ $report == *"Common administration interface"* ]]
then
echo -e "\e[31mCommon administration interface: The administrator interfaces should be visited only by those in allowed access control lists!\e[0m" | tee -a $dir/report
fi
if [[ $report == *"Common directory"* ]]
then
echo -e "\e[31mCommon directory: Directories that are not used should be removed!\e[0m" | tee -a $dir/report
fi
if [[ $report == *"Publicly writable directory"* ]]
then
echo -e "\e[31mPublicly writable directory: The HTTP PUT method should be disabled on the server!\e[0m" | tee -a $dir/report
fi
if [[ $report == *"Unencrypted password form"* ]]
then
echo -e "\e[31mUnencrypted password form: The website or web application should function with the latest secure encryption protocols (SSL,TLS)!\e[0m" | tee -a $dir/report
fi
if [[ $report == *"WebDAV"* ]]
then
```

```
echo -e "\e[31mWebDAV: WebDAV provides functionalities that are too risky and not so secure and it should be disabled!\e[0m" | tee -a $dir/report
fi
if [[ $report == *"HTTP TRACE"* ]]
then
echo -e "\e[31mHTTP TRACE: The HTTP TRACE method in most cases is not needed and should be disabled!\e[0m" | tee -a $dir/report
fi
if [[ $report == *"CVS/SVN user disclosure"* ]]
then
echo -e "\e[31mCVS/SVN user disclosure: SVN/CVS files should be removed prior to development!\e[0m" | tee -a $dir/report
fi
if [[ $report == *"Misconfiguration in LIMIT directive of .htaccess file"* ]]
then
echo -e "\e[31mMisconfiguration in LIMIT directive of .htaccess file: It is better to follow a whitelist approach with the 'LimitExcept' directive in order not to forget which methods to disable by blacklisting!\e[0m" | tee -a $dir/report
fi
if [[ $report == *"Interesting response"* ]]
then
echo -e "\e[31mInteresting response: HTTP responses should leak the least informations!\e[0m" | tee -a $dir/report
fi
if [[ $report == *"HTML object"* ]]
then
```

```
echo -e "\e[31mHTML object: Untrusted user inputs must not be placed directly to a script or in css, inside an HTML comment, an attribute name or a tag name. In specific cases where untrusted data is output in HTML or javascript contexts it should be encoded in order to avoid being interpreted as active content. Also, XSS attacks may be prevented by using appropriate HTTP response headers such as “Content-Type”,“X-Content-Type-Options” and Content-Security-Policy (CSP)!\e[0m" | tee -a $dir/report
fi
if [[ $report == *"Mixed Resource"* ]]
then
echo -e "\e[31mMixed Resource: All pages and resources should utilize HTTPS!\e[0m" | tee -a $dir/report
fi
if [[ $report == *"Insecure cookie"* ]]
then
echo -e "\e[31mInsecure cookie: Cookies that contain sensitive informations should have the 'Security' attribute set in order to only be sent when HTTPS is used!\e[0m" | tee -a $dir/report
fi
if [[ $report == *"HttpOnly cookie"* ]]
then
echo -e "\e[31mHttpOnly cookie: If cookies don’t need to be accessed by any client side scripts then the HTTPOnly attribute should be set!\e[0m" | tee -a $dir/report
fi
if [[ $report == *"Password field with auto-complete"* ]]
then
echo -e "\e[31mPassword field with auto-complete: The 'autocomplete' attribute in HTML forms should be disabled!\e[0m" | tee -a $dir/report
fi
```

```
if [[ $report == *"Form-based File Upload"* ]]
then
echo -e "\e[31mForm-based File Upload: The files should be validated prior to the upload and if possible a framework is better to be used rather than a custom validation mechanism. They should be validated regarding the extension, name and size and the preferred HTTP upload method  should be 'POST'!\e[0m" | tee -a $dir/report
fi
if [[ $report == *"Cookie set for parent domain"* ]]
then
echo -e "\e[31mCookie set for parent domain: In most situations it is recommended to omit the 'Domain' attribute in cookies!\e[0m" | tee -a $dir/report
fi
echo -e "\e[31mReport saved at $dir.\e[0m"
```

12. CONCLUSION

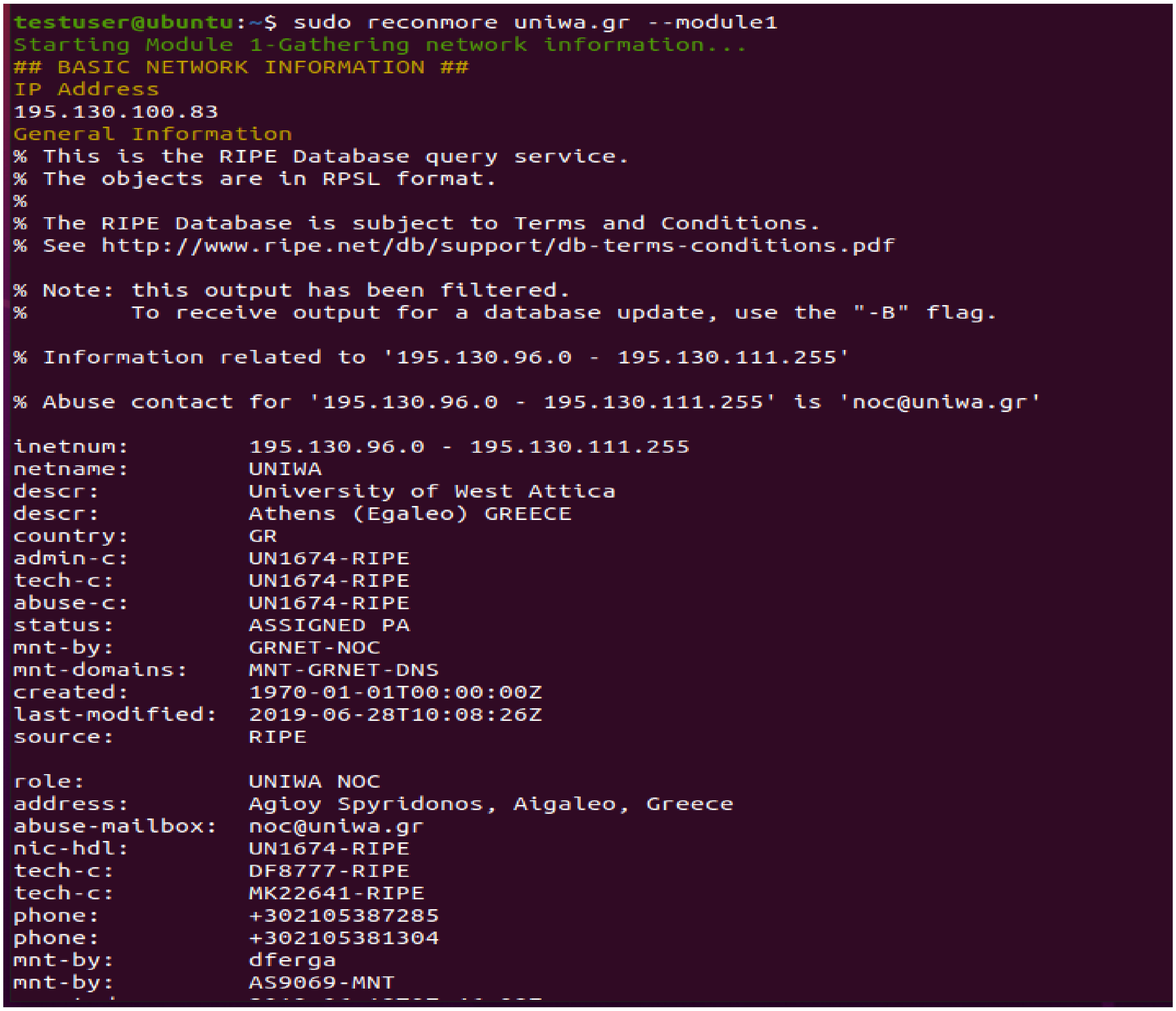
The aim of the present thesis was to develop a platform for measuring the attacking surface of an organization. It is statistically observed that many organizations don’t perform this task to the extent that they should and have limited awareness of their assets. During the development and the testing it became clear that calculating the exposure of the entire organization infrastructure is not an easy task. Performing such a task thoroughly takes many hours and has to be done quite often. In addition, the attacking surface continues to expand making it difficult for an organization to maintain an updated and detailed knowledge of it. For these reasons, automating the process of measuring the attacking surface is helpful. However, an organization in order to be fully aware and protected it should implement an entire attacking surface management strategy including continuous assets monitoring, prioritization and protection.

APPENDIX A

REPRESENTATION OF PLATFORM OPERATION

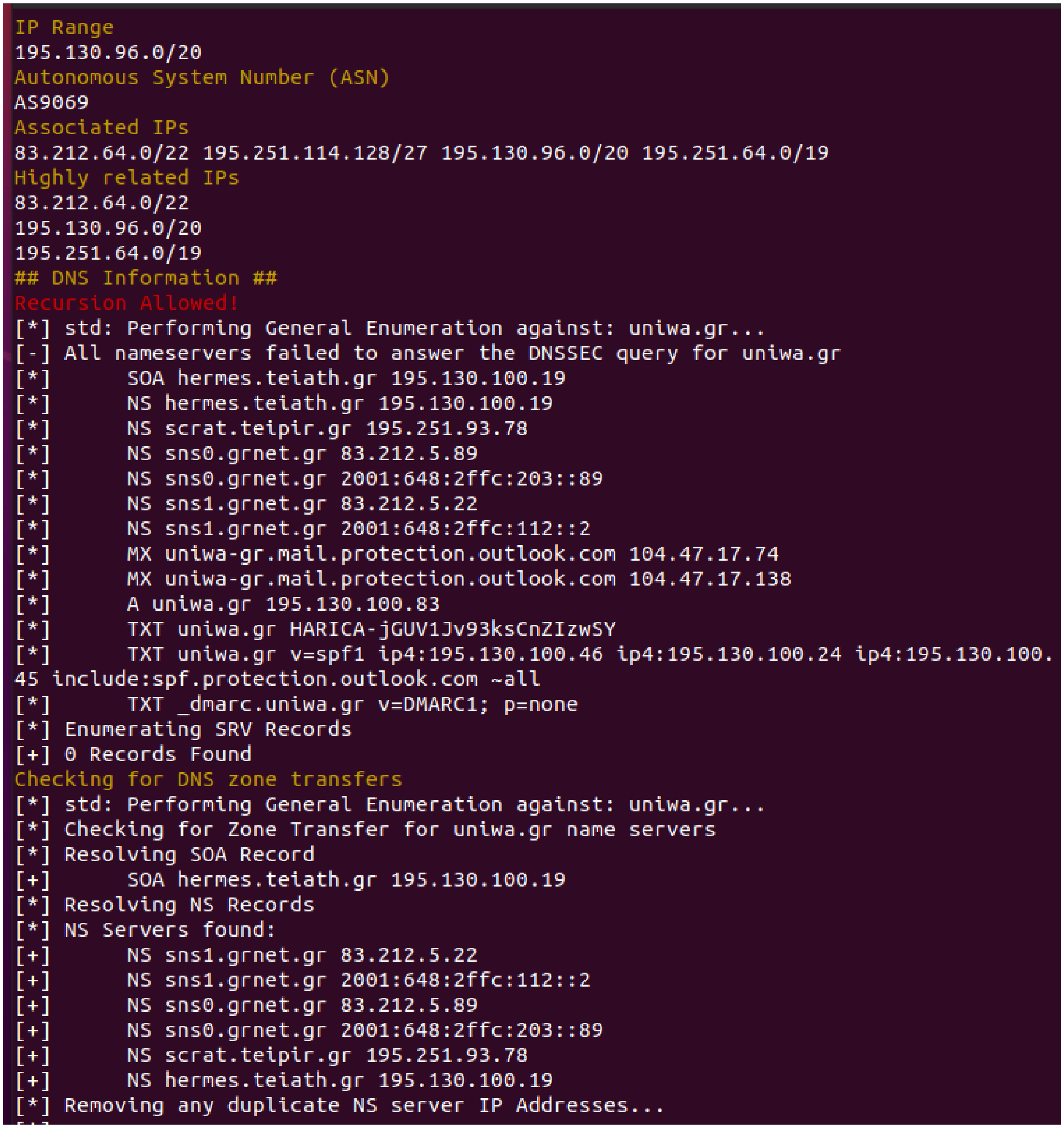
A.1 MODULE 1

The following figures show the platform operation with Module 1 against the domain *uniwa.gr*.



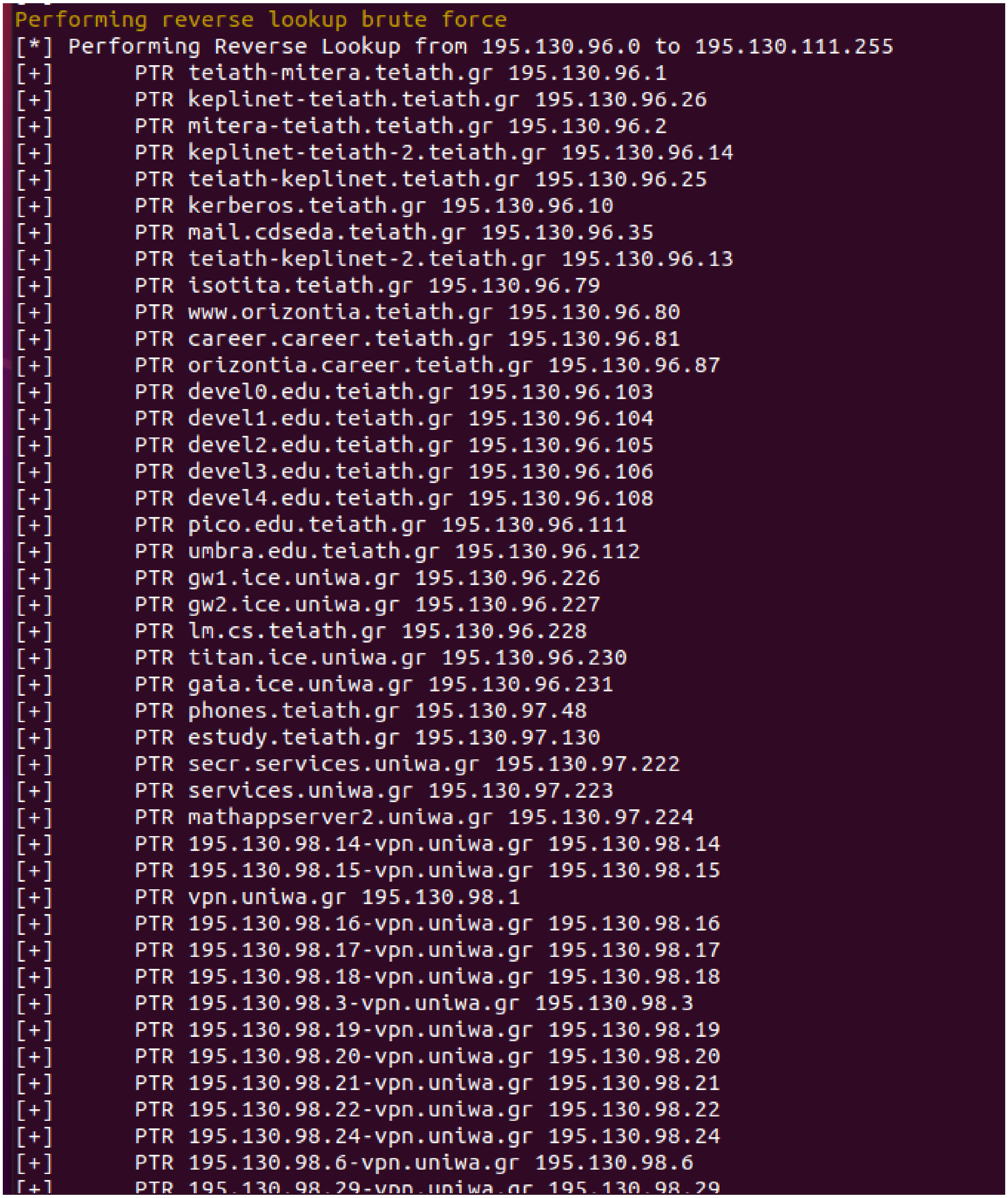
```
testuser@ubuntu:~$ sudo reconmore uniwa.gr --module1
Starting Module 1-Gathering network information...
## BASIC NETWORK INFORMATION ##
IP Address
195.130.100.83
General Information
% This is the RIPE Database query service.
% The objects are in RPSL format.
%
% The RIPE Database is subject to Terms and Conditions.
% See http://www.ripe.net/db/support/db-terms-conditions.pdf

% Note: this output has been filtered.
%       To receive output for a database update, use the "-B" flag.

% Information related to '195.130.96.0 - 195.130.111.255'

% Abuse contact for '195.130.96.0 - 195.130.111.255' is 'noc@uniwa.gr'

inetnum:        195.130.96.0 - 195.130.111.255
netname:        UNIWA
descr:          University of West Attica
descr:          Athens (Egaleo) GREECE
country:        GR
admin-c:        UN1674-RIPE
tech-c:         UN1674-RIPE
abuse-c:        UN1674-RIPE
status:         ASSIGNED PA
mnt-by:         GRNET-NOC
mnt-domains:    MNT-GRNET-DNS
created:        1970-01-01T00:00:00Z
last-modified:  2019-06-28T10:08:26Z
source:         RIPE

role:           UNIWA NOC
address:        Agioy Spyridonos, Aigaleo, Greece
abuse-mailbox:  noc@uniwa.gr
nic-hdl:        UN1674-RIPE
tech-c:         DF8777-RIPE
tech-c:         MK22641-RIPE
phone:          +302105387285
phone:          +302105381304
mnt-by:         dferga
mnt-by:         AS9069-MNT
```

Figure 3. Platform operation - Module 1: Gathering basic network information.



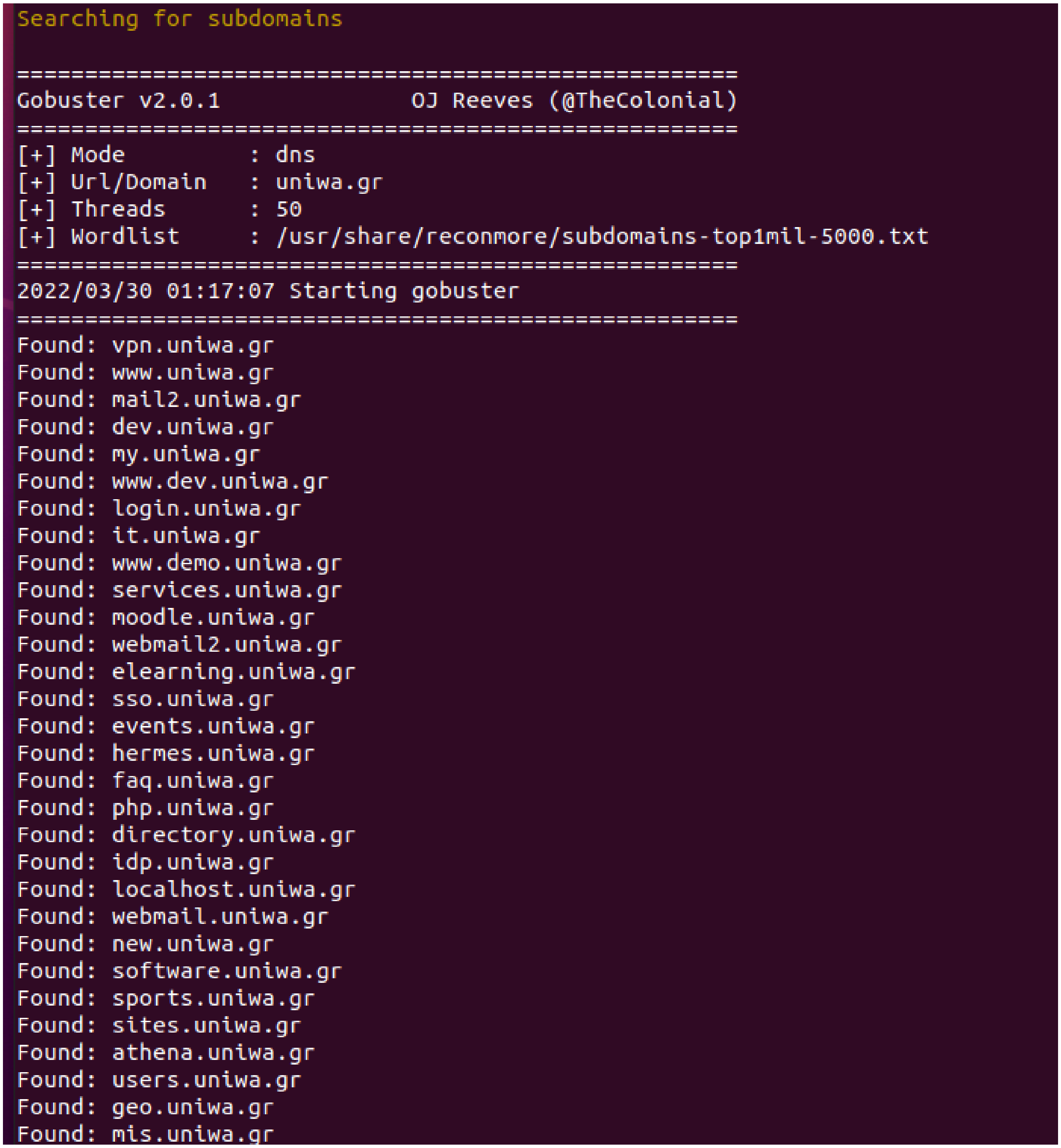
```
IP Range
195.130.96.0/20
Autonomous System Number (ASN)
AS9069
Associated IPs
83.212.64.0/22 195.251.114.128/27 195.130.96.0/20 195.251.64.0/19
Highly related IPs
83.212.64.0/22
195.130.96.0/20
195.251.64.0/19
## DNS Information ##
Recursion Allowed!
[*] std: Performing General Enumeration against: uniwa.gr...
[-] All nameservers failed to answer the DNSSEC query for uniwa.gr
[*]      SOA hermes.teiath.gr 195.130.100.19
[*]      NS hermes.teiath.gr 195.130.100.19
[*]      NS scrat.teipir.gr 195.251.93.78
[*]      NS sns0.grnet.gr 83.212.5.89
[*]      NS sns0.grnet.gr 2001:648:2ffc:203::89
[*]      NS sns1.grnet.gr 83.212.5.22
[*]      NS sns1.grnet.gr 2001:648:2ffc:112::2
[*]      MX uniwa-gr.mail.protection.outlook.com 104.47.17.74
[*]      MX uniwa-gr.mail.protection.outlook.com 104.47.17.138
[*]      A uniwa.gr 195.130.100.83
[*]      TXT uniwa.gr HARICA-jGUV1Jv93ksCnZIzwSY
[*]      TXT uniwa.gr v=spf1 ip4:195.130.100.46 ip4:195.130.100.24 ip4:195.130.100.
45 include:spf.protection.outlook.com ~all
[*]      TXT _dmarc.uniwa.gr v=DMARC1; p=none
[*] Enumerating SRV Records
[+] 0 Records Found
Checking for DNS zone transfers
[*] std: Performing General Enumeration against: uniwa.gr...
[*] Checking for Zone Transfer for uniwa.gr name servers
[*] Resolving SOA Record
[+]      SOA hermes.teiath.gr 195.130.100.19
[*] Resolving NS Records
[*] NS Servers found:
[+]      NS sns1.grnet.gr 83.212.5.22
[+]      NS sns1.grnet.gr 2001:648:2ffc:112::2
[+]      NS sns0.grnet.gr 83.212.5.89
[+]      NS sns0.grnet.gr 2001:648:2ffc:203::89
[+]      NS scrat.teipir.gr 195.251.93.78
[+]      NS hermes.teiath.gr 195.130.100.19
[*] Removing any duplicate NS server IP Addresses...
```

Figure 4. Platform operation - Module 1: Gathering DNS information.



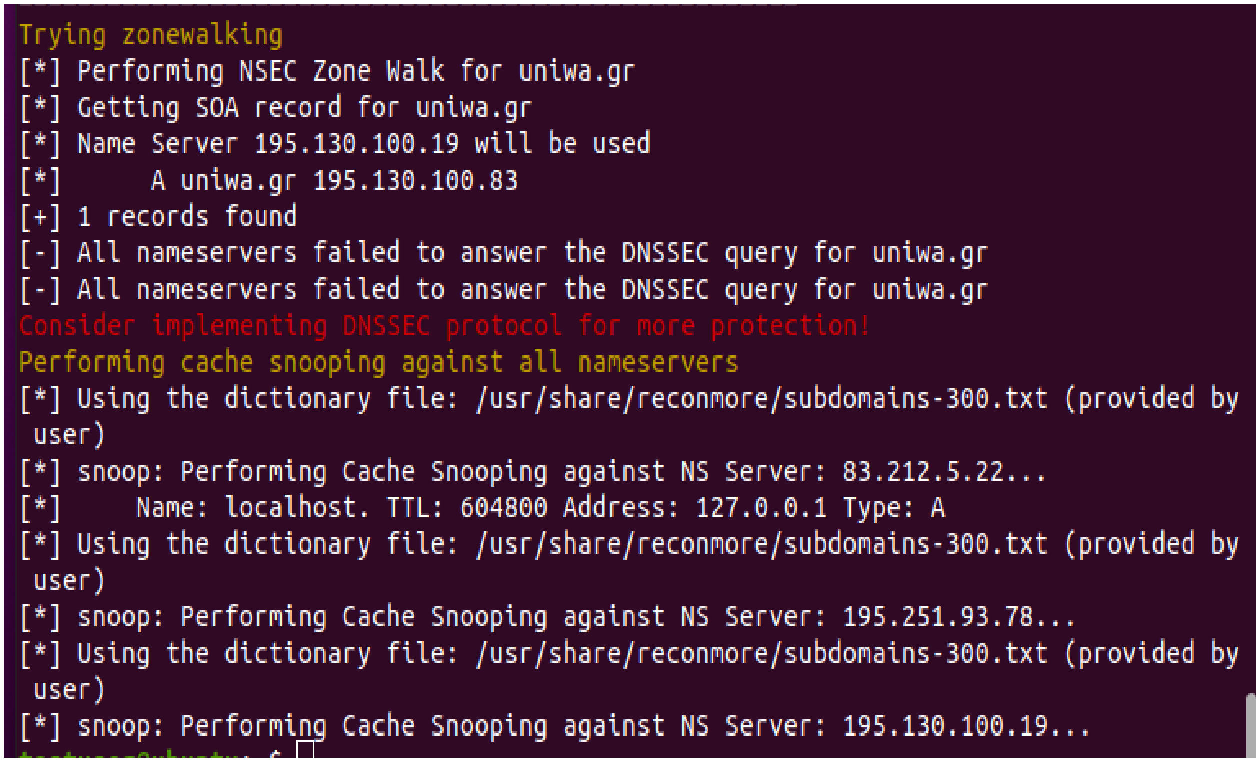
```
Performing reverse lookup brute force
[*] Performing Reverse Lookup from 195.130.96.0 to 195.130.111.255
[+]      PTR teiath-mitera.teiath.gr 195.130.96.1
[+]      PTR keplinet-teiath.teiath.gr 195.130.96.26
[+]      PTR mitera-teiath.teiath.gr 195.130.96.2
[+]      PTR keplinet-teiath-2.teiath.gr 195.130.96.14
[+]      PTR teiath-keplinet.teiath.gr 195.130.96.25
[+]      PTR kerberos.teiath.gr 195.130.96.10
[+]      PTR mail.cdseda.teiath.gr 195.130.96.35
[+]      PTR teiath-keplinet-2.teiath.gr 195.130.96.13
[+]      PTR isotita.teiath.gr 195.130.96.79
[+]      PTR www.orizontia.teiath.gr 195.130.96.80
[+]      PTR career.career.teiath.gr 195.130.96.81
[+]      PTR orizontia.career.teiath.gr 195.130.96.87
[+]      PTR devel0.edu.teiath.gr 195.130.96.103
[+]      PTR devel1.edu.teiath.gr 195.130.96.104
[+]      PTR devel2.edu.teiath.gr 195.130.96.105
[+]      PTR devel3.edu.teiath.gr 195.130.96.106
[+]      PTR devel4.edu.teiath.gr 195.130.96.108
[+]      PTR pico.edu.teiath.gr 195.130.96.111
[+]      PTR umbra.edu.teiath.gr 195.130.96.112
[+]      PTR gw1.ice.uniwa.gr 195.130.96.226
[+]      PTR gw2.ice.uniwa.gr 195.130.96.227
[+]      PTR lm.cs.teiath.gr 195.130.96.228
[+]      PTR titan.ice.uniwa.gr 195.130.96.230
[+]      PTR gaia.ice.uniwa.gr 195.130.96.231
[+]      PTR phones.teiath.gr 195.130.97.48
[+]      PTR estudy.teiath.gr 195.130.97.130
[+]      PTR secr.services.uniwa.gr 195.130.97.222
[+]      PTR services.uniwa.gr 195.130.97.223
[+]      PTR mathappserver2.uniwa.gr 195.130.97.224
[+]      PTR 195.130.98.14-vpn.uniwa.gr 195.130.98.14
[+]      PTR 195.130.98.15-vpn.uniwa.gr 195.130.98.15
[+]      PTR vpn.uniwa.gr 195.130.98.1
[+]      PTR 195.130.98.16-vpn.uniwa.gr 195.130.98.16
[+]      PTR 195.130.98.17-vpn.uniwa.gr 195.130.98.17
[+]      PTR 195.130.98.18-vpn.uniwa.gr 195.130.98.18
[+]      PTR 195.130.98.3-vpn.uniwa.gr 195.130.98.3
[+]      PTR 195.130.98.19-vpn.uniwa.gr 195.130.98.19
[+]      PTR 195.130.98.20-vpn.uniwa.gr 195.130.98.20
[+]      PTR 195.130.98.21-vpn.uniwa.gr 195.130.98.21
[+]      PTR 195.130.98.22-vpn.uniwa.gr 195.130.98.22
[+]      PTR 195.130.98.24-vpn.uniwa.gr 195.130.98.24
[+]      PTR 195.130.98.6-vpn.uniwa.gr 195.130.98.6
[+]      PTR 195.130.98.29-vpn.uniwa.gr 195.130.98.29
```

Figure 5. Platform operation - Module 1: Reverse lookup brute force.



```
Searching for subdomains

=====================================================
Gobuster v2.0.1              OJ Reeves (@TheColonial)
=====================================================
[+] Mode         : dns
[+] Url/Domain   : uniwa.gr
[+] Threads      : 50
[+] Wordlist     : /usr/share/reconmore/subdomains-top1mil-5000.txt
=====================================================
2022/03/30 01:17:07 Starting gobuster
=====================================================
Found: vpn.uniwa.gr
Found: www.uniwa.gr
Found: mail2.uniwa.gr
Found: dev.uniwa.gr
Found: my.uniwa.gr
Found: www.dev.uniwa.gr
Found: login.uniwa.gr
Found: it.uniwa.gr
Found: www.demo.uniwa.gr
Found: services.uniwa.gr
Found: moodle.uniwa.gr
Found: webmail2.uniwa.gr
Found: elearning.uniwa.gr
Found: sso.uniwa.gr
Found: events.uniwa.gr
Found: hermes.uniwa.gr
Found: faq.uniwa.gr
Found: php.uniwa.gr
Found: directory.uniwa.gr
Found: idp.uniwa.gr
Found: localhost.uniwa.gr
Found: webmail.uniwa.gr
Found: new.uniwa.gr
Found: software.uniwa.gr
Found: sports.uniwa.gr
Found: sites.uniwa.gr
Found: athena.uniwa.gr
Found: users.uniwa.gr
Found: geo.uniwa.gr
Found: mis.uniwa.gr
```

Figure 6. Platform operation - Module 1: Search for subdomains.

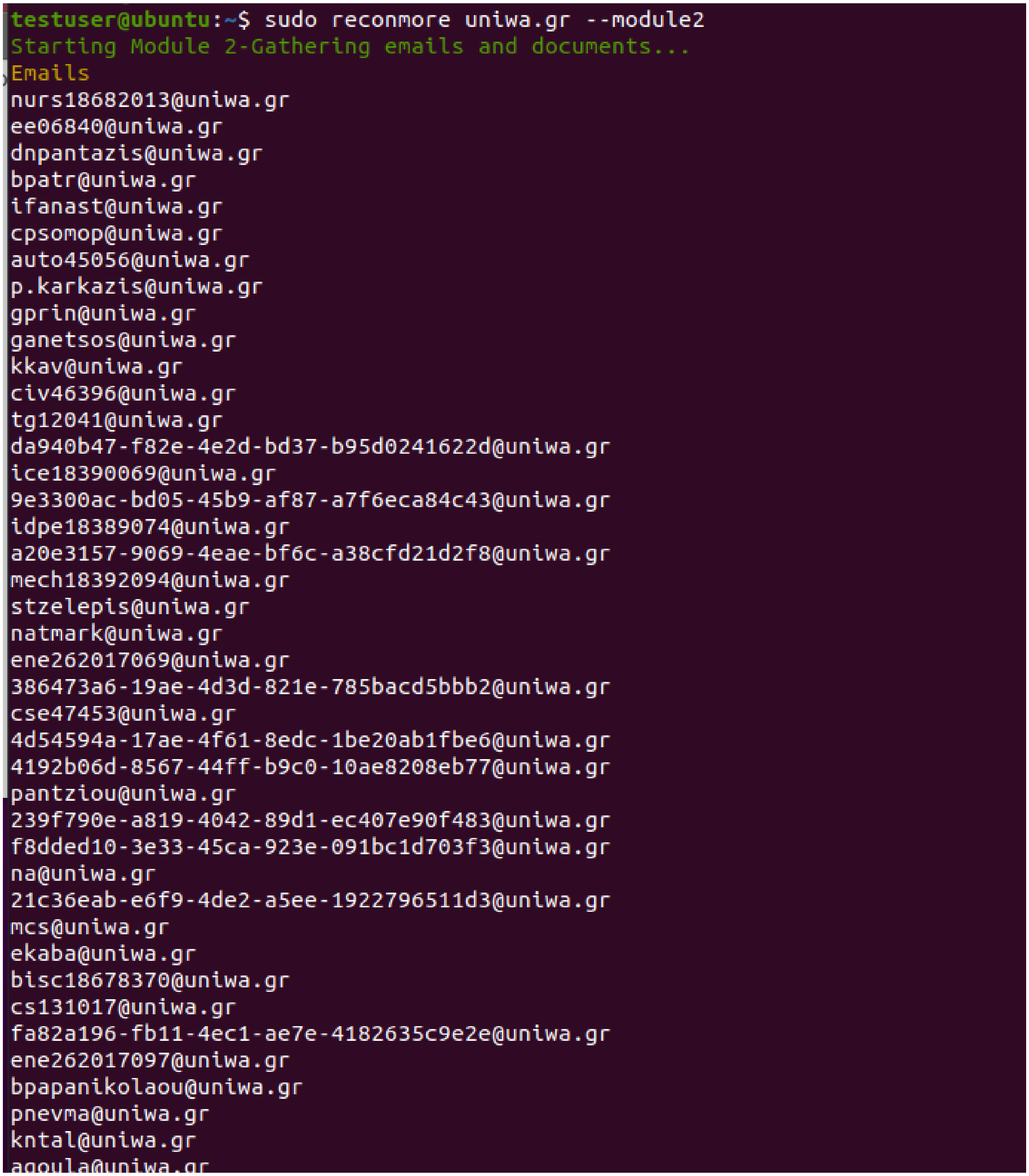


```
Trying zonewalking
[*] Performing NSEC Zone Walk for uniwa.gr
[*] Getting SOA record for uniwa.gr
[*] Name Server 195.130.100.19 will be used
[*]      A uniwa.gr 195.130.100.83
[+] 1 records found
[-] All nameservers failed to answer the DNSSEC query for uniwa.gr
[-] All nameservers failed to answer the DNSSEC query for uniwa.gr
Consider implementing DNSSEC protocol for more protection!
Performing cache snooping against all nameservers
[*] Using the dictionary file: /usr/share/reconmore/subdomains-300.txt (provided by
 user)
[*] snoop: Performing Cache Snooping against NS Server: 83.212.5.22...
[*]     Name: localhost. TTL: 604800 Address: 127.0.0.1 Type: A
[*] Using the dictionary file: /usr/share/reconmore/subdomains-300.txt (provided by
 user)
[*] snoop: Performing Cache Snooping against NS Server: 195.251.93.78...
[*] Using the dictionary file: /usr/share/reconmore/subdomains-300.txt (provided by
 user)
[*] snoop: Performing Cache Snooping against NS Server: 195.130.100.19...
```

Figure 7. Platform operation - Module 1: Zonewalking and cache snooping.

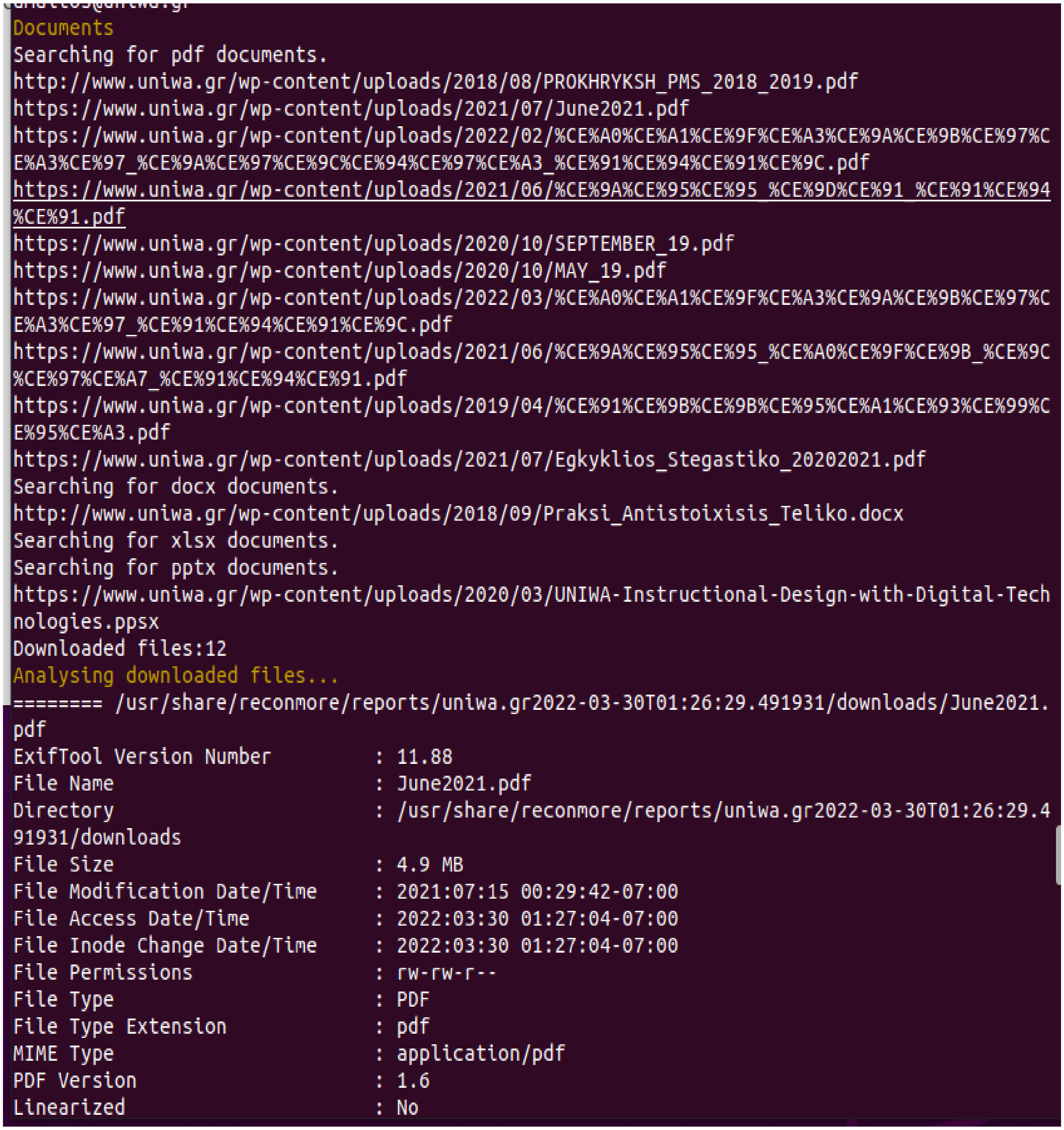
A.1 MODULE 2

The following figures show the platform operation with Module 2 against the domain *uniwa.gr*.



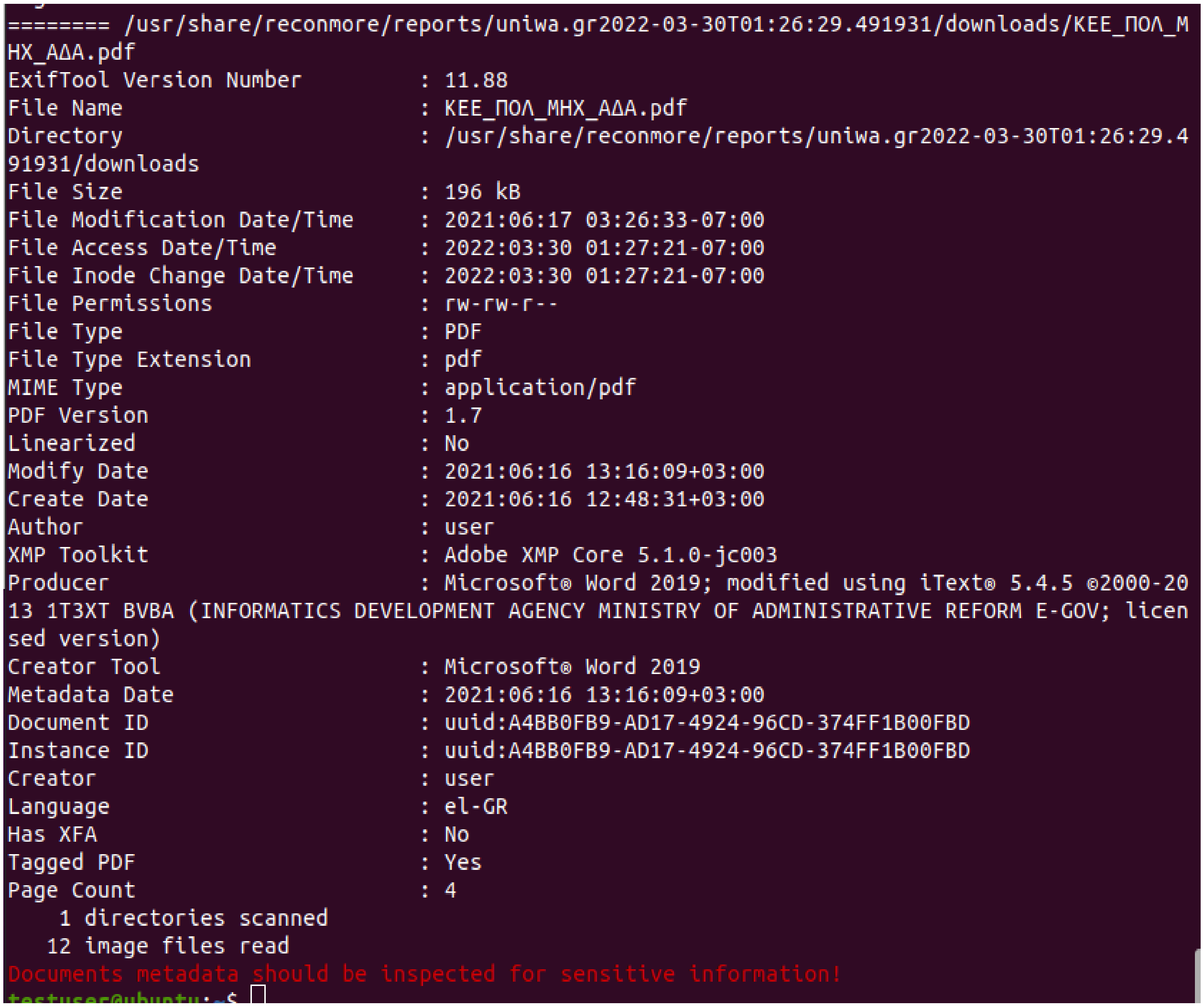
```
testuser@ubuntu:~$ sudo reconmore uniwa.gr --module2
Starting Module 2-Gathering emails and documents...
Emails
nurs18682013@uniwa.gr
ee06840@uniwa.gr
dnpantazis@uniwa.gr
bpatr@uniwa.gr
ifanast@uniwa.gr
cpsomop@uniwa.gr
auto45056@uniwa.gr
p.karkazis@uniwa.gr
gprin@uniwa.gr
ganetsos@uniwa.gr
kkav@uniwa.gr
civ46396@uniwa.gr
tg12041@uniwa.gr
da940b47-f82e-4e2d-bd37-b95d0241622d@uniwa.gr
ice18390069@uniwa.gr
9e3300ac-bd05-45b9-af87-a7f6eca84c43@uniwa.gr
idpe18389074@uniwa.gr
a20e3157-9069-4eae-bf6c-a38cfd21d2f8@uniwa.gr
mech18392094@uniwa.gr
stzelepis@uniwa.gr
natmark@uniwa.gr
ene262017069@uniwa.gr
386473a6-19ae-4d3d-821e-785bacd5bbb2@uniwa.gr
cse47453@uniwa.gr
4d54594a-17ae-4f61-8edc-1be20ab1fbe6@uniwa.gr
4192b06d-8567-44ff-b9c0-10ae8208eb77@uniwa.gr
pantziou@uniwa.gr
239f790e-a819-4042-89d1-ec407e90f483@uniwa.gr
f8dded10-3e33-45ca-923e-091bc1d703f3@uniwa.gr
na@uniwa.gr
21c36eab-e6f9-4de2-a5ee-1922796511d3@uniwa.gr
mcs@uniwa.gr
ekaba@uniwa.gr
bisc18678370@uniwa.gr
cs131017@uniwa.gr
fa82a196-fb11-4ec1-ae7e-4182635c9e2e@uniwa.gr
ene262017097@uniwa.gr
bpapanikolaou@uniwa.gr
pnevma@uniwa.gr
kntal@uniwa.gr
agoula@uniwa.gr
```

Figure 8. Platform operation - Module 2: Gathering emails.



```
Documents
Searching for pdf documents.
http://www.uniwa.gr/wp-content/uploads/2018/08/PROKHRYKSH_PMS_2018_2019.pdf
https://www.uniwa.gr/wp-content/uploads/2021/07/June2021.pdf
https://www.uniwa.gr/wp-content/uploads/2022/02/%CE%A0%CE%A1%CE%9F%CE%A3%CE%9A%CE%9B%CE%97%C
E%A3%CE%97_%CE%9A%CE%97%CE%9C%CE%94%CE%97%CE%A3_%CE%91%CE%94%CE%91%CE%9C.pdf
https://www.uniwa.gr/wp-content/uploads/2021/06/%CE%9A%CE%95%CE%95_%CE%9D%CE%91_%CE%91%CE%94
%CE%91.pdf
https://www.uniwa.gr/wp-content/uploads/2020/10/SEPTEMBER_19.pdf
https://www.uniwa.gr/wp-content/uploads/2020/10/MAY_19.pdf
https://www.uniwa.gr/wp-content/uploads/2022/03/%CE%A0%CE%A1%CE%9F%CE%A3%CE%9A%CE%9B%CE%97%C
E%A3%CE%97_%CE%91%CE%94%CE%91%CE%9C.pdf
https://www.uniwa.gr/wp-content/uploads/2021/06/%CE%9A%CE%95%CE%95_%CE%A0%CE%9F%CE%9B_%CE%9C
%CE%97%CE%A7_%CE%91%CE%94%CE%91.pdf
https://www.uniwa.gr/wp-content/uploads/2019/04/%CE%91%CE%9B%CE%9B%CE%95%CE%A1%CE%93%CE%99%C
E%95%CE%A3.pdf
https://www.uniwa.gr/wp-content/uploads/2021/07/Egkyklios_Stegastiko_20202021.pdf
Searching for docx documents.
http://www.uniwa.gr/wp-content/uploads/2018/09/Praksi_Antistoixisis_Teliko.docx
Searching for xlsx documents.
Searching for pptx documents.
https://www.uniwa.gr/wp-content/uploads/2020/03/UNIWA-Instructional-Design-with-Digital-Tech
nologies.ppsx
Downloaded files:12
Analysing downloaded files...
======== /usr/share/reconmore/reports/uniwa.gr2022-03-30T01:26:29.491931/downloads/June2021.
pdf
ExifTool Version Number         : 11.88
File Name                       : June2021.pdf
Directory                       : /usr/share/reconmore/reports/uniwa.gr2022-03-30T01:26:29.4
91931/downloads
File Size                       : 4.9 MB
File Modification Date/Time     : 2021:07:15 00:29:42-07:00
File Access Date/Time           : 2022:03:30 01:27:04-07:00
File Inode Change Date/Time     : 2022:03:30 01:27:04-07:00
File Permissions                : rw-rw-r--
File Type                       : PDF
File Type Extension             : pdf
MIME Type                       : application/pdf
PDF Version                     : 1.6
Linearized                      : No
```

Figure 9. Platform operation - Module 2: Gathering documents.

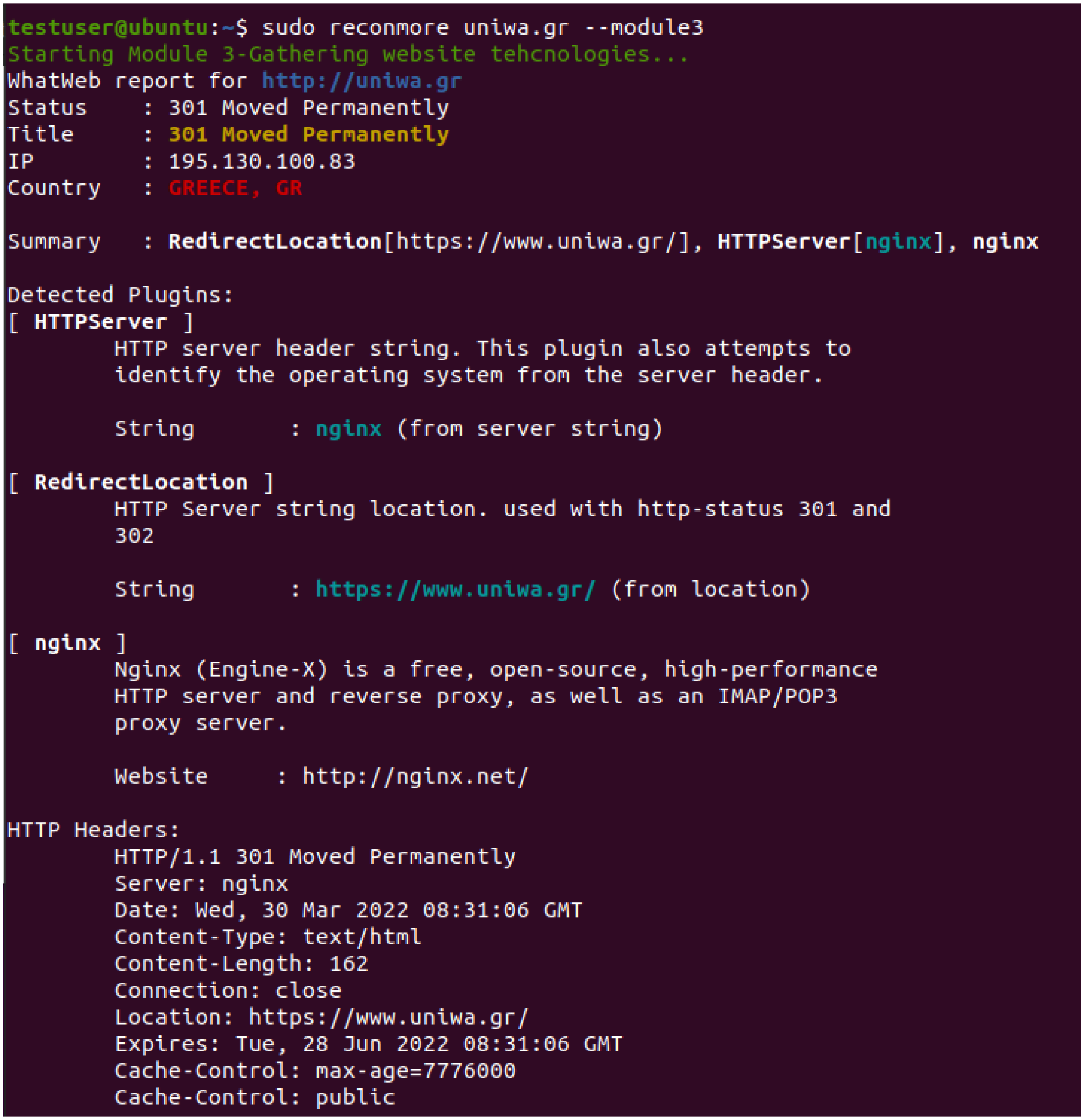


```
======== /usr/share/reconmore/reports/uniwa.gr2022-03-30T01:26:29.491931/downloads/KEE_ΠΟΛ_M
HX_AΔA.pdf
ExifTool Version Number         : 11.88
File Name                       : KEE_ΠΟΛ_MHX_AΔA.pdf
Directory                       : /usr/share/reconmore/reports/uniwa.gr2022-03-30T01:26:29.4
91931/downloads
File Size                       : 196 kB
File Modification Date/Time     : 2021:06:17 03:26:33-07:00
File Access Date/Time           : 2022:03:30 01:27:21-07:00
File Inode Change Date/Time     : 2022:03:30 01:27:21-07:00
File Permissions                : rw-rw-r--
File Type                       : PDF
File Type Extension             : pdf
MIME Type                       : application/pdf
PDF Version                     : 1.7
Linearized                      : No
Modify Date                     : 2021:06:16 13:16:09+03:00
Create Date                     : 2021:06:16 12:48:31+03:00
Author                          : user
XMP Toolkit                     : Adobe XMP Core 5.1.0-jc003
Producer                        : Microsoft® Word 2019; modified using iText® 5.4.5 ©2000-20
13 1T3XT BVBA (INFORMATICS DEVELOPMENT AGENCY MINISTRY OF ADMINISTRATIVE REFORM E-GOV; licen
sed version)
Creator Tool                    : Microsoft® Word 2019
Metadata Date                   : 2021:06:16 13:16:09+03:00
Document ID                     : uuid:A4BB0FB9-AD17-4924-96CD-374FF1B00FBD
Instance ID                     : uuid:A4BB0FB9-AD17-4924-96CD-374FF1B00FBD
Creator                         : user
Language                        : el-GR
Has XFA                         : No
Tagged PDF                      : Yes
Page Count                      : 4
    1 directories scanned
   12 image files read
Documents metadata should be inspected for sensitive information!
testuser@ubuntu:~$
```

Figure 10. Platform operation - Module 2: Documents analysis.

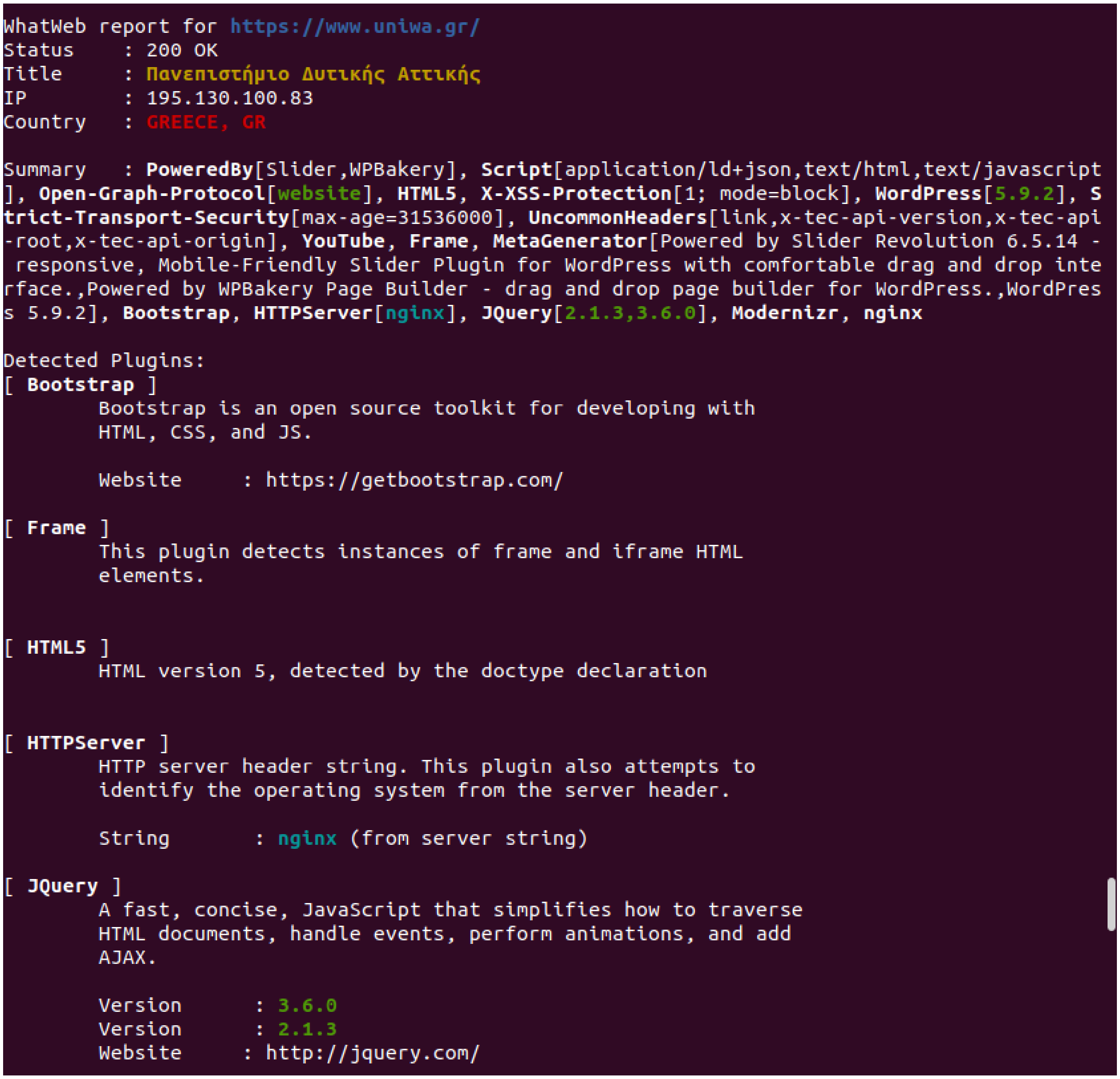
A.2 MODULE 3

The following figures show the platform operation with Module 3 against the domain *uniwa.gr*.



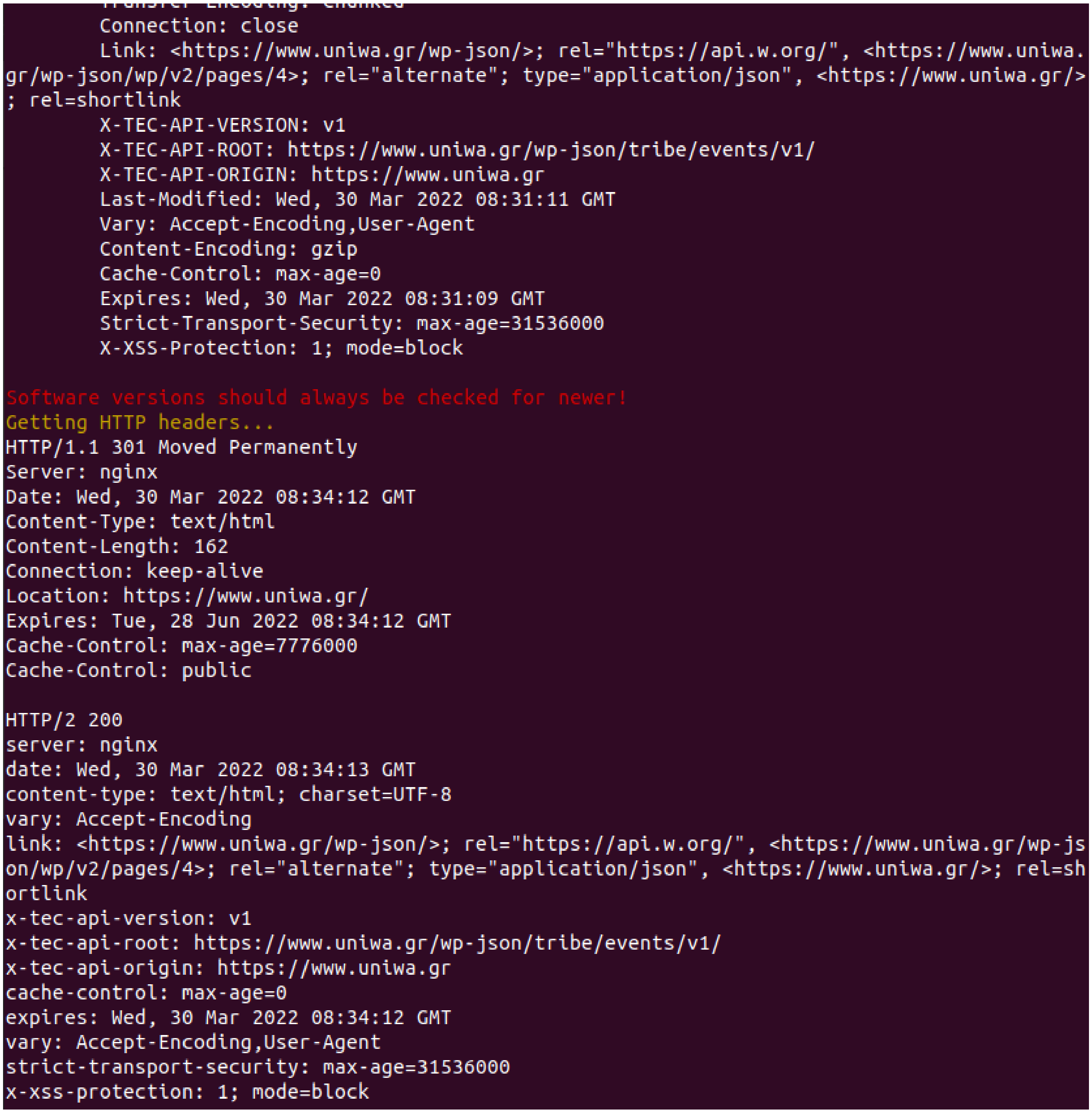
```
testuser@ubuntu:~$ sudo reconmore uniwa.gr --module3
Starting Module 3-Gathering website tehcnologies...
WhatWeb report for http://uniwa.gr
Status    : 301 Moved Permanently
Title     : 301 Moved Permanently
IP        : 195.130.100.83
Country   : GREECE, GR

Summary   : RedirectLocation[https://www.uniwa.gr/], HTTPServer[nginx], nginx

Detected Plugins:
[ HTTPServer ]
        HTTP server header string. This plugin also attempts to
        identify the operating system from the server header.

        String       : nginx (from server string)

[ RedirectLocation ]
        HTTP Server string location. used with http-status 301 and
        302

        String       : https://www.uniwa.gr/ (from location)

[ nginx ]
        Nginx (Engine-X) is a free, open-source, high-performance
        HTTP server and reverse proxy, as well as an IMAP/POP3
        proxy server.

        Website     : http://nginx.net/

HTTP Headers:
        HTTP/1.1 301 Moved Permanently
        Server: nginx
        Date: Wed, 30 Mar 2022 08:31:06 GMT
        Content-Type: text/html
        Content-Length: 162
        Connection: close
        Location: https://www.uniwa.gr/
        Expires: Tue, 28 Jun 2022 08:31:06 GMT
        Cache-Control: max-age=7776000
        Cache-Control: public
```

Figure 11. Platform operation - Module 3: Gathering website technologies.



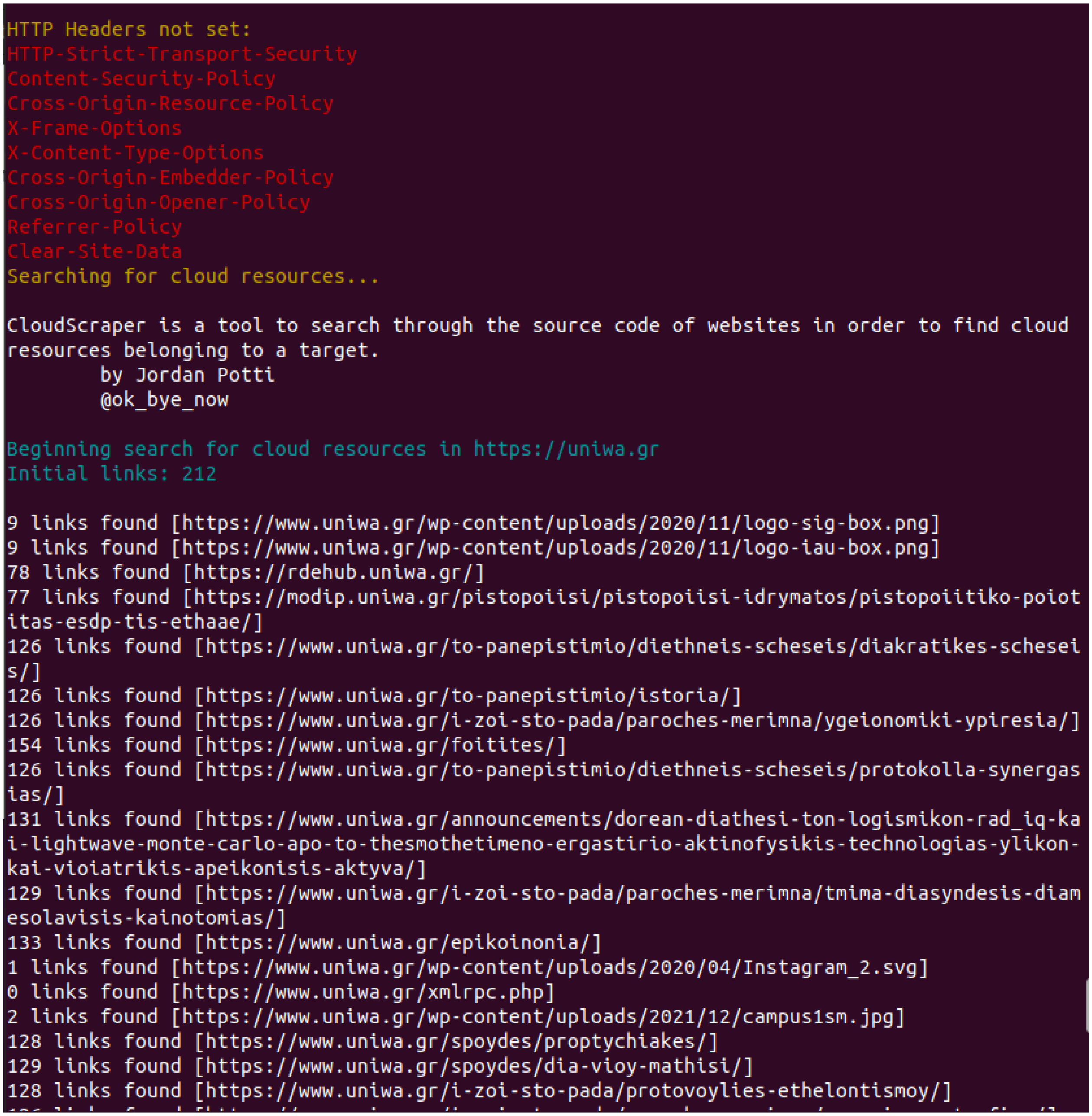
```
WhatWeb report for https://www.uniwa.gr/
Status    : 200 OK
Title     : Πανεπιστήμιο Δυτικής Αττικής
IP        : 195.130.100.83
Country   : GREECE, GR

Summary   : PoweredBy[Slider,WPBakery], Script[application/ld+json,text/html,text/javascript
], Open-Graph-Protocol[website], HTML5, X-XSS-Protection[1; mode=block], WordPress[5.9.2], S
trict-Transport-Security[max-age=31536000], UncommonHeaders[link,x-tec-api-version,x-tec-api
-root,x-tec-api-origin], YouTube, Frame, MetaGenerator[Powered by Slider Revolution 6.5.14 -
 responsive, Mobile-Friendly Slider Plugin for WordPress with comfortable drag and drop inte
rface.,Powered by WPBakery Page Builder - drag and drop page builder for WordPress.,WordPres
s 5.9.2], Bootstrap, HTTPServer[nginx], JQuery[2.1.3,3.6.0], Modernizr, nginx

Detected Plugins:
[ Bootstrap ]
        Bootstrap is an open source toolkit for developing with
        HTML, CSS, and JS.

        Website     : https://getbootstrap.com/

[ Frame ]
        This plugin detects instances of frame and iframe HTML
        elements.


[ HTML5 ]
        HTML version 5, detected by the doctype declaration


[ HTTPServer ]
        HTTP server header string. This plugin also attempts to
        identify the operating system from the server header.

        String       : nginx (from server string)

[ JQuery ]
        A fast, concise, JavaScript that simplifies how to traverse
        HTML documents, handle events, perform animations, and add
        AJAX.

        Version      : 3.6.0
        Version      : 2.1.3
        Website     : http://jquery.com/
```

Figure 12. Platform operation - Module 3: Gathering website technologies after redirection.



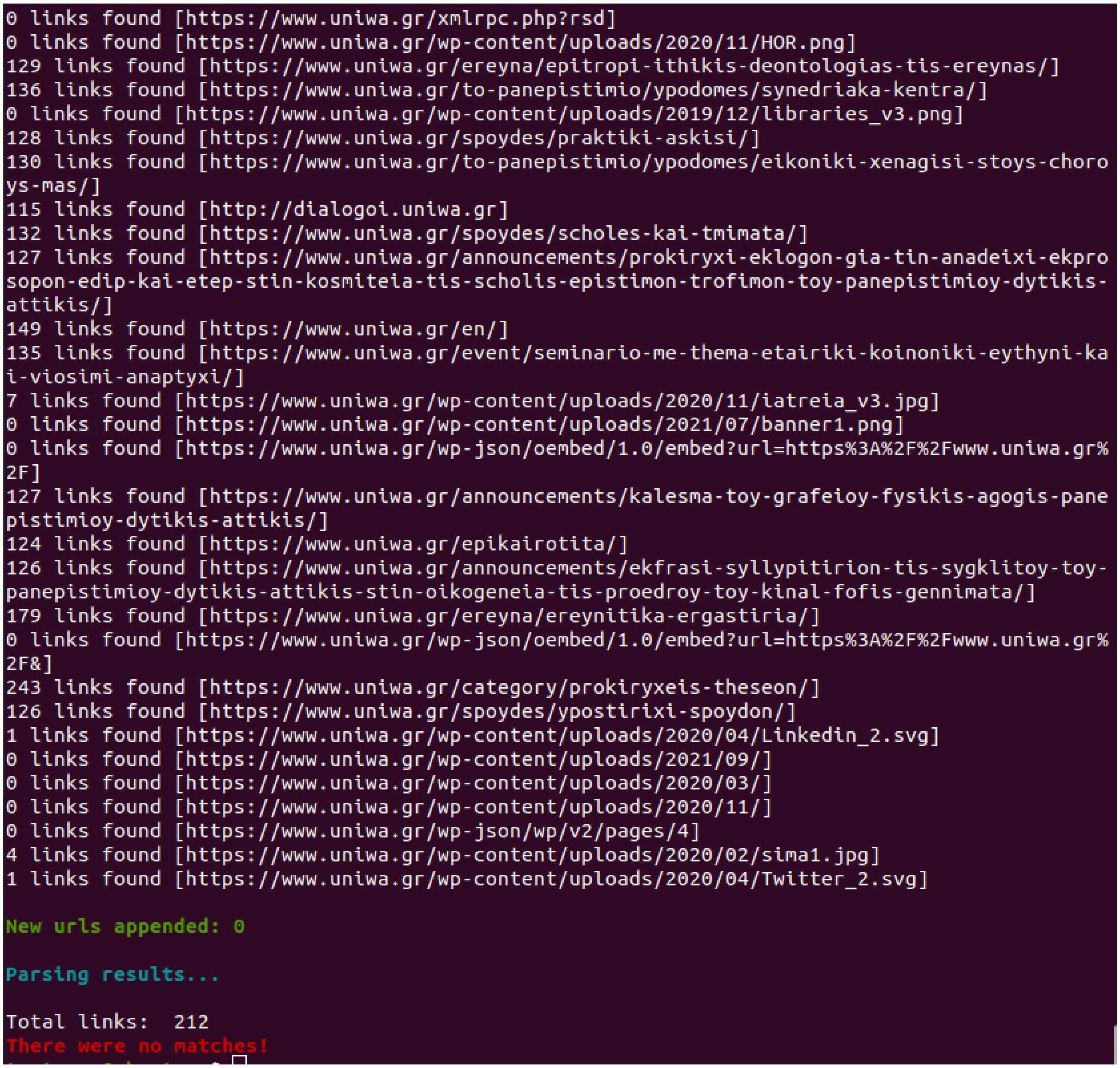
```
        Transfer-Encoding: chunked
        Connection: close
        Link: <https://www.uniwa.gr/wp-json/>; rel="https://api.w.org/", <https://www.uniwa.
gr/wp-json/wp/v2/pages/4>; rel="alternate"; type="application/json", <https://www.uniwa.gr/>
; rel=shortlink
        X-TEC-API-VERSION: v1
        X-TEC-API-ROOT: https://www.uniwa.gr/wp-json/tribe/events/v1/
        X-TEC-API-ORIGIN: https://www.uniwa.gr
        Last-Modified: Wed, 30 Mar 2022 08:31:11 GMT
        Vary: Accept-Encoding,User-Agent
        Content-Encoding: gzip
        Cache-Control: max-age=0
        Expires: Wed, 30 Mar 2022 08:31:09 GMT
        Strict-Transport-Security: max-age=31536000
        X-XSS-Protection: 1; mode=block

Software versions should always be checked for newer!
Getting HTTP headers...
HTTP/1.1 301 Moved Permanently
Server: nginx
Date: Wed, 30 Mar 2022 08:34:12 GMT
Content-Type: text/html
Content-Length: 162
Connection: keep-alive
Location: https://www.uniwa.gr/
Expires: Tue, 28 Jun 2022 08:34:12 GMT
Cache-Control: max-age=7776000
Cache-Control: public

HTTP/2 200
server: nginx
date: Wed, 30 Mar 2022 08:34:13 GMT
content-type: text/html; charset=UTF-8
vary: Accept-Encoding
link: <https://www.uniwa.gr/wp-json/>; rel="https://api.w.org/", <https://www.uniwa.gr/wp-js
on/wp/v2/pages/4>; rel="alternate"; type="application/json", <https://www.uniwa.gr/>; rel=sh
ortlink
x-tec-api-version: v1
x-tec-api-root: https://www.uniwa.gr/wp-json/tribe/events/v1/
x-tec-api-origin: https://www.uniwa.gr
cache-control: max-age=0
expires: Wed, 30 Mar 2022 08:34:12 GMT
vary: Accept-Encoding,User-Agent
strict-transport-security: max-age=31536000
x-xss-protection: 1; mode=block
```

Figure 13. Platform operation - Module 3: Getting HTTP headers.



```
HTTP Headers not set:
HTTP-Strict-Transport-Security
Content-Security-Policy
Cross-Origin-Resource-Policy
X-Frame-Options
X-Content-Type-Options
Cross-Origin-Embedder-Policy
Cross-Origin-Opener-Policy
Referrer-Policy
Clear-Site-Data
Searching for cloud resources...

CloudScraper is a tool to search through the source code of websites in order to find cloud
resources belonging to a target.
        by Jordan Potti
        @ok_bye_now

Beginning search for cloud resources in https://uniwa.gr
Initial links: 212

9 links found [https://www.uniwa.gr/wp-content/uploads/2020/11/logo-sig-box.png]
9 links found [https://www.uniwa.gr/wp-content/uploads/2020/11/logo-iau-box.png]
78 links found [https://rdehub.uniwa.gr/]
77 links found [https://modip.uniwa.gr/pistopoiisi/pistopoiisi-idrymatos/pistopoiitiko-poiot
itas-esdp-tis-ethaae/]
126 links found [https://www.uniwa.gr/to-panepistimio/diethneis-scheseis/diakratikes-schesei
s/]
126 links found [https://www.uniwa.gr/to-panepistimio/istoria/]
126 links found [https://www.uniwa.gr/i-zoi-sto-pada/paroches-merimna/ygeionomiki-ypiresia/]
154 links found [https://www.uniwa.gr/foitites/]
126 links found [https://www.uniwa.gr/to-panepistimio/diethneis-scheseis/protokolla-synergas
ias/]
131 links found [https://www.uniwa.gr/announcements/dorean-diathesi-ton-logismikon-rad_iq-ka
i-lightwave-monte-carlo-apo-to-thesmothetimeno-ergastirio-aktinofysikis-technologias-ylikon-
kai-vioiatrikis-apeikonisis-aktyva/]
129 links found [https://www.uniwa.gr/i-zoi-sto-pada/paroches-merimna/tmima-diasyndesis-diam
esolavisis-kainotomias/]
133 links found [https://www.uniwa.gr/epikoinonia/]
1 links found [https://www.uniwa.gr/wp-content/uploads/2020/04/Instagram_2.svg]
0 links found [https://www.uniwa.gr/xmlrpc.php]
2 links found [https://www.uniwa.gr/wp-content/uploads/2021/12/campus1sm.jpg]
128 links found [https://www.uniwa.gr/spoydes/proptychiakes/]
129 links found [https://www.uniwa.gr/spoydes/dia-vioy-mathisi/]
128 links found [https://www.uniwa.gr/i-zoi-sto-pada/protovoylies-ethelontismoy/]
```

Figure 14. Platform operation - Module 3: Checking HTTP headers.

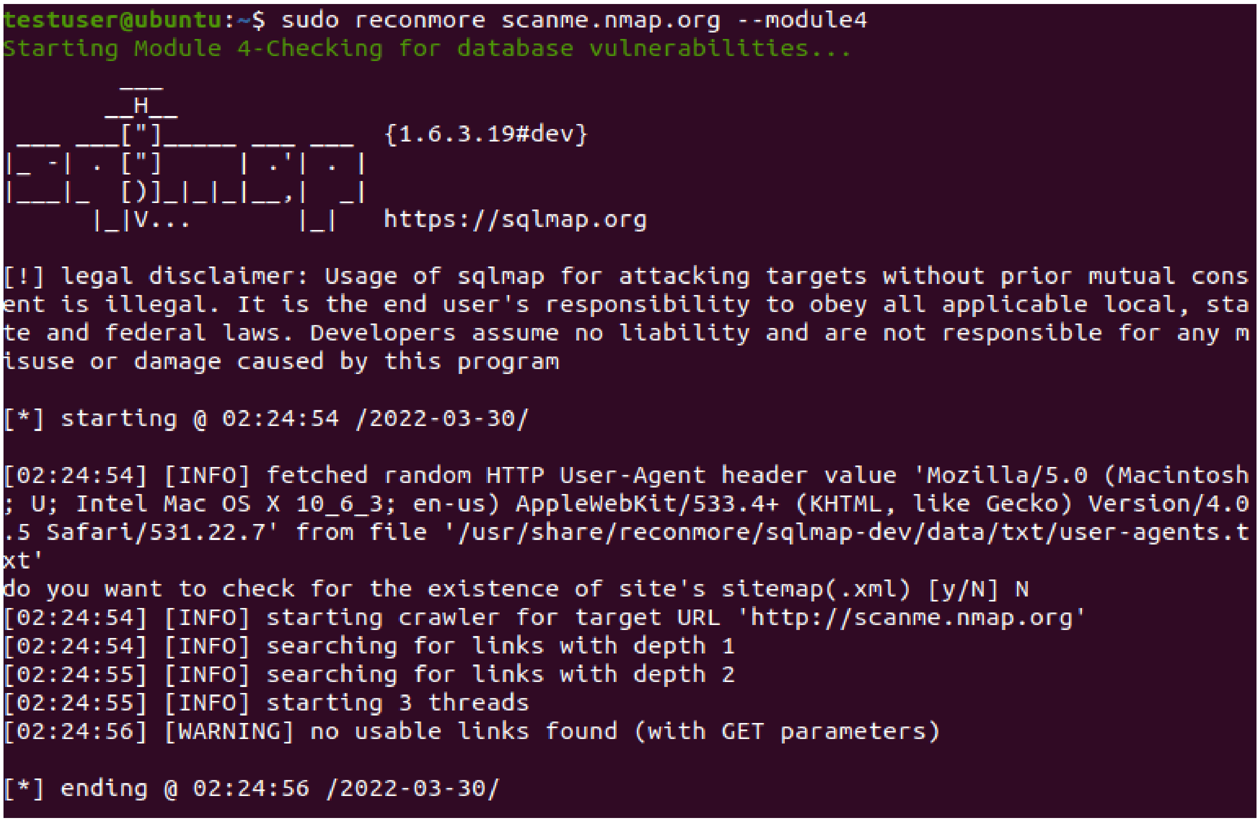


```
0 links found [https://www.uniwa.gr/xmlrpc.php?rsd]
0 links found [https://www.uniwa.gr/wp-content/uploads/2020/11/HOR.png]
129 links found [https://www.uniwa.gr/ereyna/epitropi-ithikis-deontologias-tis-ereynas/]
136 links found [https://www.uniwa.gr/to-panepistimio/ypodomes/synedriaka-kentra/]
0 links found [https://www.uniwa.gr/wp-content/uploads/2019/12/libraries_v3.png]
128 links found [https://www.uniwa.gr/spoydes/praktiki-askisi/]
130 links found [https://www.uniwa.gr/to-panepistimio/ypodomes/eikoniki-xenagisi-stoys-choro
ys-mas/]
115 links found [http://dialogoi.uniwa.gr]
132 links found [https://www.uniwa.gr/spoydes/scholes-kai-tmimata/]
127 links found [https://www.uniwa.gr/announcements/prokiryxi-eklogon-gia-tin-anadeixi-ekpro
sopon-edip-kai-etep-stin-kosmiteia-tis-scholis-epistimon-trofimon-toy-panepistimioy-dytikis-
attikis/]
149 links found [https://www.uniwa.gr/en/]
135 links found [https://www.uniwa.gr/event/seminario-me-thema-etairiki-koinoniki-eythyni-ka
i-viosimi-anaptyxi/]
7 links found [https://www.uniwa.gr/wp-content/uploads/2020/11/iatreia_v3.jpg]
0 links found [https://www.uniwa.gr/wp-content/uploads/2021/07/banner1.png]
0 links found [https://www.uniwa.gr/wp-json/oembed/1.0/embed?url=https%3A%2F%2Fwww.uniwa.gr%
2F]
127 links found [https://www.uniwa.gr/announcements/kalesma-toy-grafeioy-fysikis-agogis-pane
pistimioy-dytikis-attikis/]
124 links found [https://www.uniwa.gr/epikairotita/]
126 links found [https://www.uniwa.gr/announcements/ekfrasi-syllypitirion-tis-sygklitoy-toy-
panepistimioy-dytikis-attikis-stin-oikogeneia-tis-proedroy-toy-kinal-fofis-gennimata/]
179 links found [https://www.uniwa.gr/ereyna/ereynitika-ergastiria/]
0 links found [https://www.uniwa.gr/wp-json/oembed/1.0/embed?url=https%3A%2F%2Fwww.uniwa.gr%
2F&]
243 links found [https://www.uniwa.gr/category/prokiryxeis-theseon/]
126 links found [https://www.uniwa.gr/spoydes/ypostirixi-spoydon/]
1 links found [https://www.uniwa.gr/wp-content/uploads/2020/04/Linkedin_2.svg]
0 links found [https://www.uniwa.gr/wp-content/uploads/2021/09/]
0 links found [https://www.uniwa.gr/wp-content/uploads/2020/03/]
0 links found [https://www.uniwa.gr/wp-content/uploads/2020/11/]
0 links found [https://www.uniwa.gr/wp-json/wp/v2/pages/4]
4 links found [https://www.uniwa.gr/wp-content/uploads/2020/02/sima1.jpg]
1 links found [https://www.uniwa.gr/wp-content/uploads/2020/04/Twitter_2.svg]

New urls appended: 0

Parsing results...

Total links:  212
There were no matches!
```

Figure 15. Platform operation - Module 3: Searching for cloud resources.

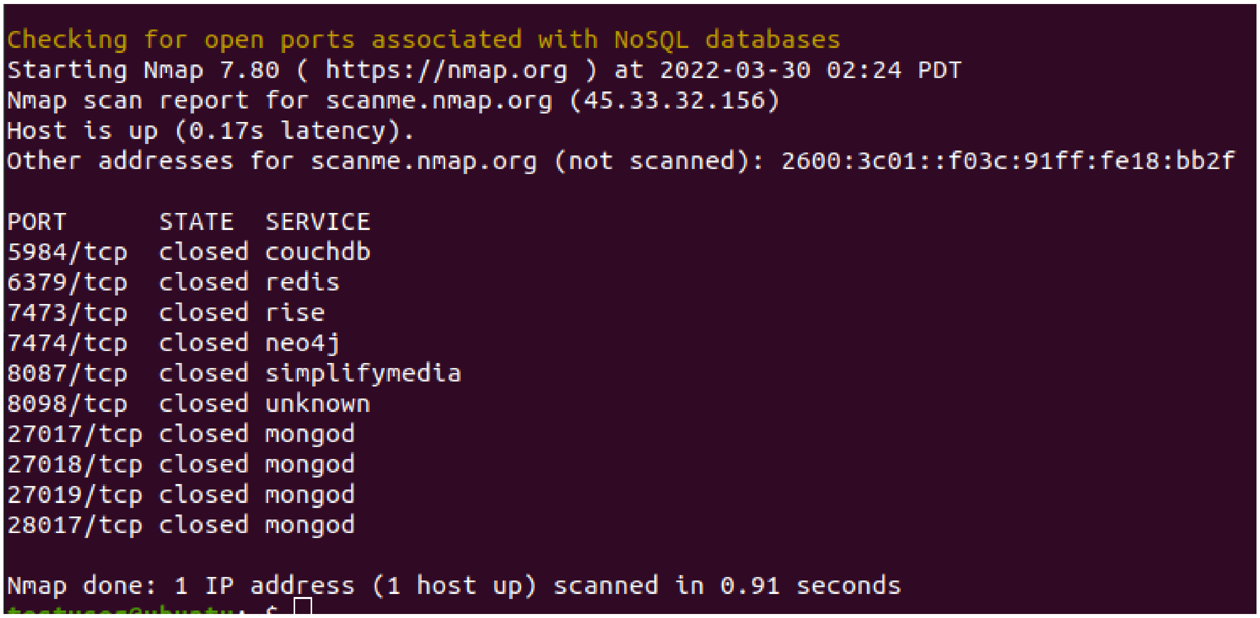
A.3 MODULE 4

The following figures show the platform operation with Module 4 against the domain *scanme.nmap.org*.



```
testuser@ubuntu:~$ sudo reconmore scanme.nmap.org --module4
Starting Module 4-Checking for database vulnerabilities...
        ___
       __H__
 ___ ___["]_____ ___ ___  {1.6.3.19#dev}
|_ -| . ["]     | .'| . |
|___|_  [)]_|_|_|__,|  _|
      |_|V...       |_|   https://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual cons
ent is illegal. It is the end user's responsibility to obey all applicable local, sta
te and federal laws. Developers assume no liability and are not responsible for any m
isuse or damage caused by this program

[*] starting @ 02:24:54 /2022-03-30/

[02:24:54] [INFO] fetched random HTTP User-Agent header value 'Mozilla/5.0 (Macintosh
; U; Intel Mac OS X 10_6_3; en-us) AppleWebKit/533.4+ (KHTML, like Gecko) Version/4.0
.5 Safari/531.22.7' from file '/usr/share/reconmore/sqlmap-dev/data/txt/user-agents.t
xt'
do you want to check for the existence of site's sitemap(.xml) [y/N] N
[02:24:54] [INFO] starting crawler for target URL 'http://scanme.nmap.org'
[02:24:54] [INFO] searching for links with depth 1
[02:24:55] [INFO] searching for links with depth 2
[02:24:55] [INFO] starting 3 threads
[02:24:56] [WARNING] no usable links found (with GET parameters)

[*] ending @ 02:24:56 /2022-03-30/
```

Figure 16. Platform operation - Module 4: Searching for SQL database vulnerabilities.

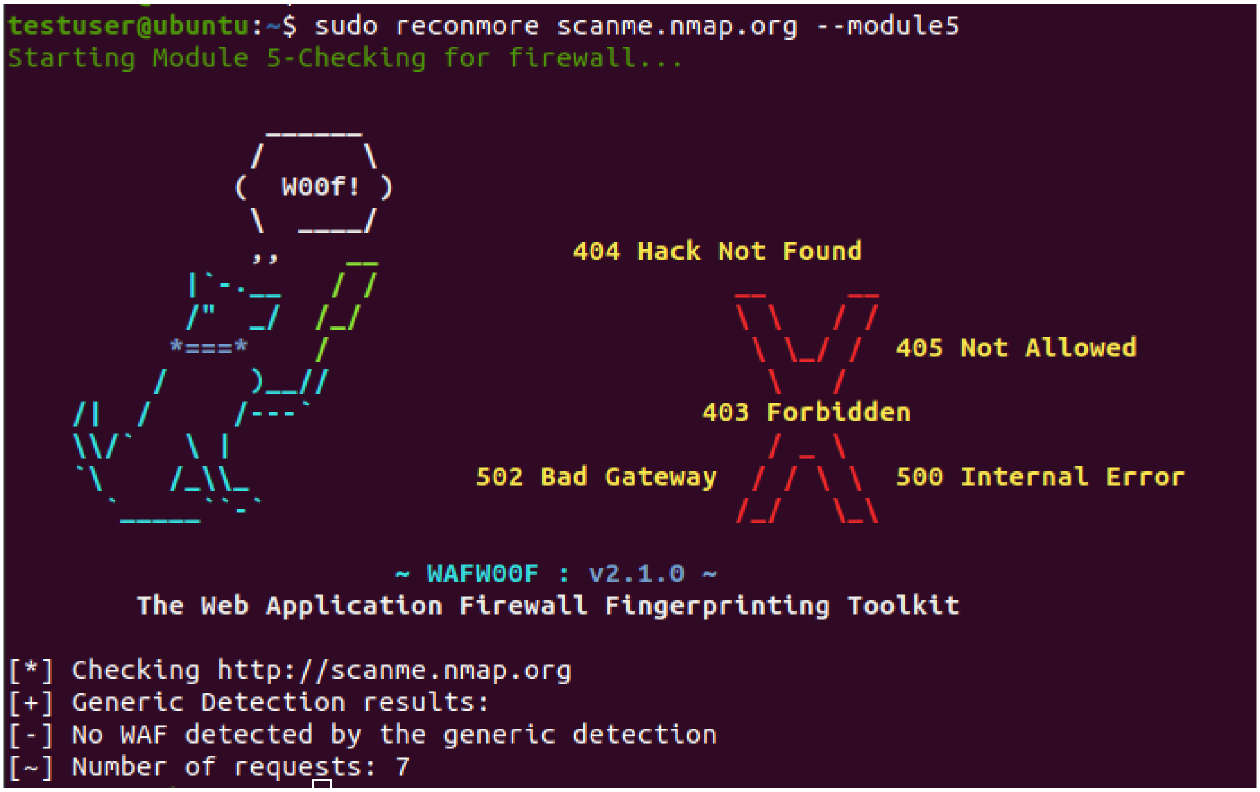


```
Checking for open ports associated with NoSQL databases
Starting Nmap 7.80 ( https://nmap.org ) at 2022-03-30 02:24 PDT
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.17s latency).
Other addresses for scanme.nmap.org (not scanned): 2600:3c01::f03c:91ff:fe18:bb2f

PORT      STATE  SERVICE
5984/tcp  closed couchdb
6379/tcp  closed redis
7473/tcp  closed rise
7474/tcp  closed neo4j
8087/tcp  closed simplifymedia
8098/tcp  closed unknown
27017/tcp closed mongod
27018/tcp closed mongod
27019/tcp closed mongod
28017/tcp closed mongod

Nmap done: 1 IP address (1 host up) scanned in 0.91 seconds
testuser@ubuntu:~$
```

Figure 17. Platform operation - Module 4: Searching for NoSQL database ports.

A.4 MODULE 5

The following figures show the platform operation with Module 5 against the domain *scanme.nmap.org*.

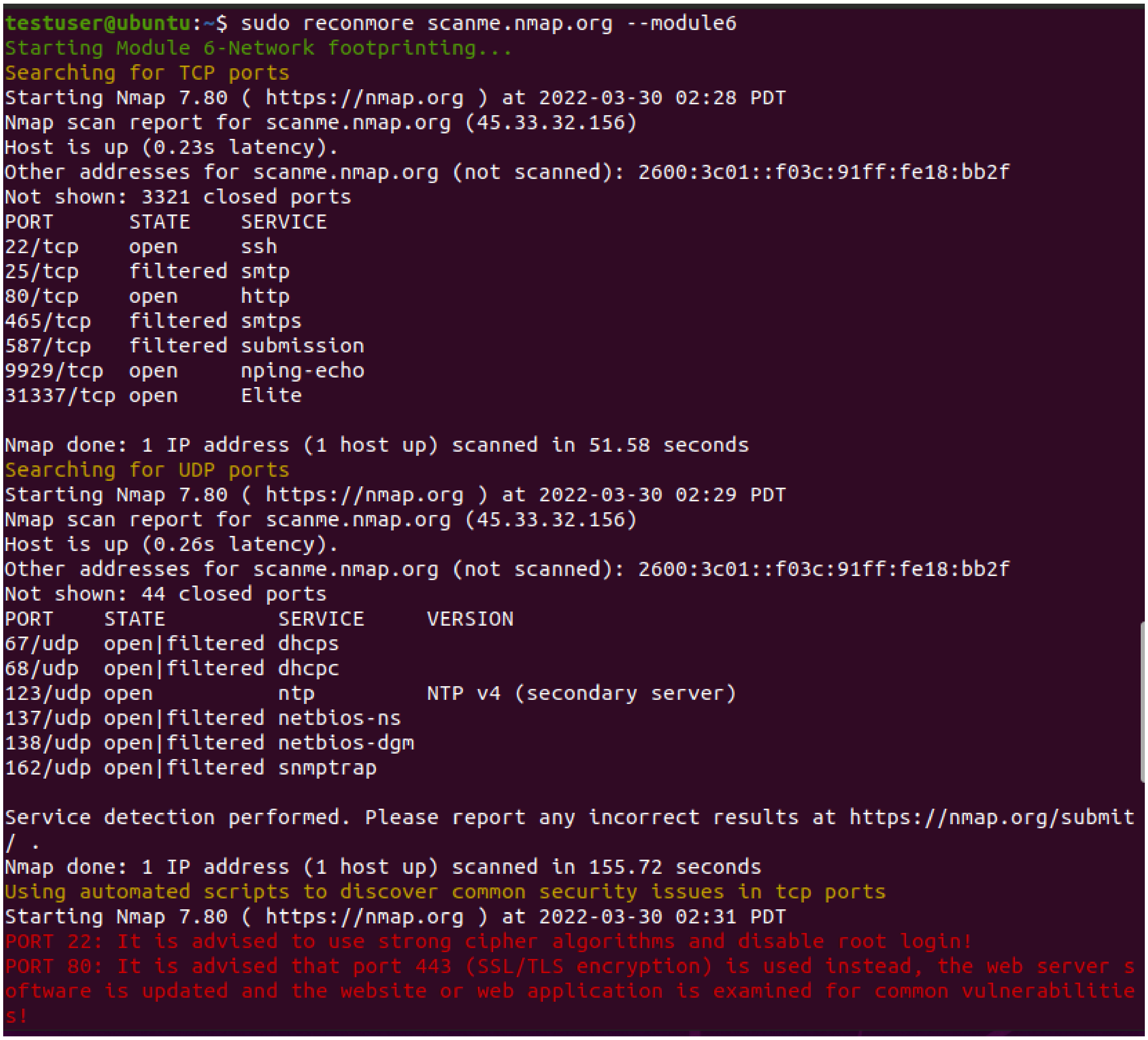


```
testuser@ubuntu:~$ sudo reconmore scanme.nmap.org --module5
Starting Module 5-Checking for firewall...

                ______
               /      \
              (  W00f! )
               \  ____/
               ,,    __            404 Hack Not Found
           |`-.__   / /                      __     __
           /"  _/  /_/                       \ \   / /
          *===*    /                          \ \_/ /  405 Not Allowed
         /     )__//                           \   /
    /|  /     /---`                        403 Forbidden
    \\/`   \ |                                 / _ \
    `\    /_\\_              502 Bad Gateway  / / \ \  500 Internal Error
      `_____``-`                             /_/   \_\

                        ~ WAFW00F : v2.1.0 ~
        The Web Application Firewall Fingerprinting Toolkit

[*] Checking http://scanme.nmap.org
[+] Generic Detection results:
[-] No WAF detected by the generic detection
[~] Number of requests: 7
```

Figure 18. Platform operation - Module 5: Searching for web application firewalls.

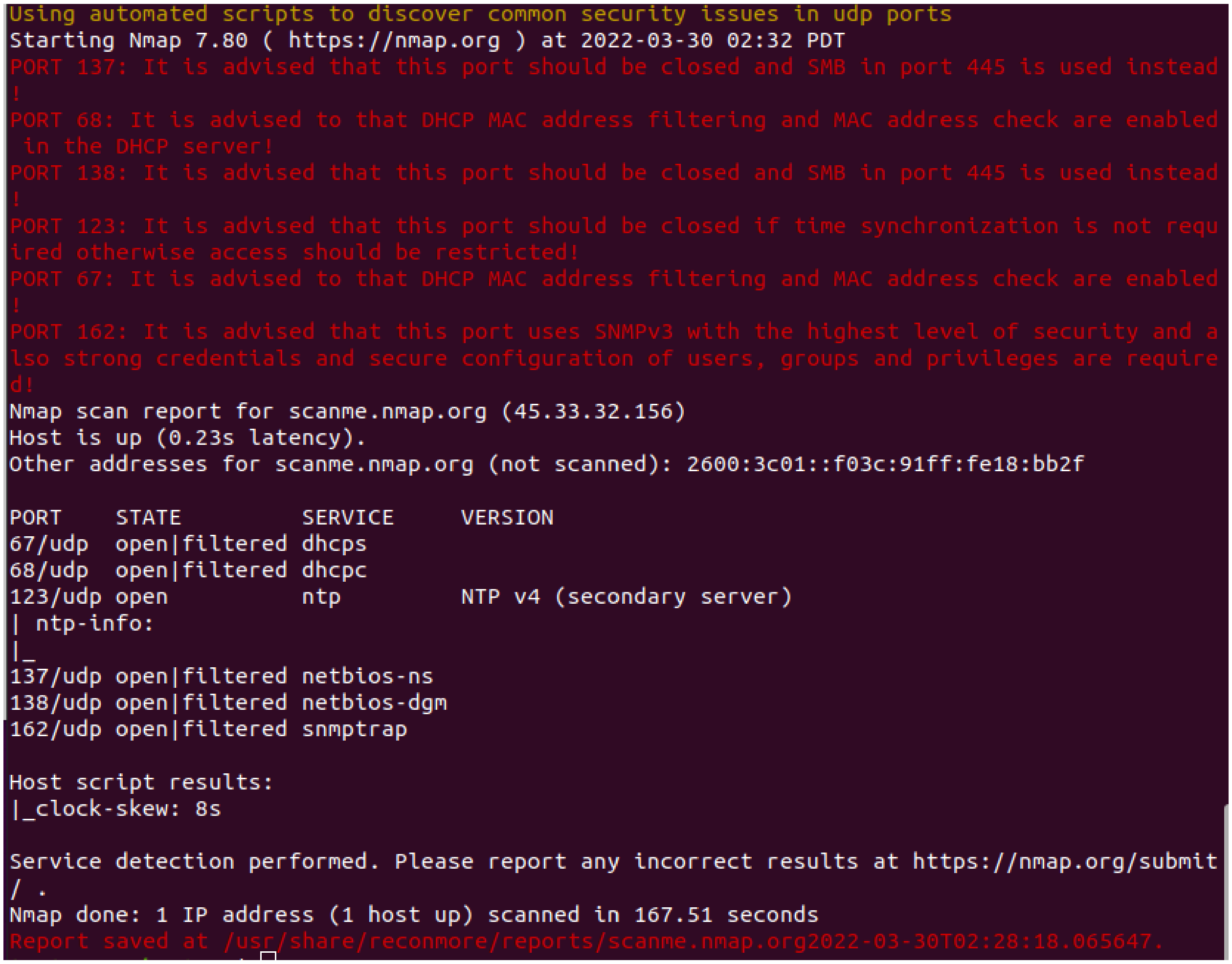
A.5 MODULE 6

The following figures show the platform operation with Module 6 against the domain *scanme.nmap.org*.



```
testuser@ubuntu:~$ sudo reconmore scanme.nmap.org --module6
Starting Module 6-Network footprinting...
Searching for TCP ports
Starting Nmap 7.80 ( https://nmap.org ) at 2022-03-30 02:28 PDT
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.23s latency).
Other addresses for scanme.nmap.org (not scanned): 2600:3c01::f03c:91ff:fe18:bb2f
Not shown: 3321 closed ports
PORT      STATE    SERVICE
22/tcp    open     ssh
25/tcp    filtered smtp
80/tcp    open     http
465/tcp   filtered smtps
587/tcp   filtered submission
9929/tcp  open     nping-echo
31337/tcp open     Elite

Nmap done: 1 IP address (1 host up) scanned in 51.58 seconds
Searching for UDP ports
Starting Nmap 7.80 ( https://nmap.org ) at 2022-03-30 02:29 PDT
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.26s latency).
Other addresses for scanme.nmap.org (not scanned): 2600:3c01::f03c:91ff:fe18:bb2f
Not shown: 44 closed ports
PORT    STATE         SERVICE     VERSION
67/udp  open|filtered dhcps
68/udp  open|filtered dhcpc
123/udp open          ntp         NTP v4 (secondary server)
137/udp open|filtered netbios-ns
138/udp open|filtered netbios-dgm
162/udp open|filtered snmptrap

Service detection performed. Please report any incorrect results at https://nmap.org/submit
/ .
Nmap done: 1 IP address (1 host up) scanned in 155.72 seconds
Using automated scripts to discover common security issues in tcp ports
Starting Nmap 7.80 ( https://nmap.org ) at 2022-03-30 02:31 PDT
PORT 22: It is advised to use strong cipher algorithms and disable root login!
PORT 80: It is advised that port 443 (SSL/TLS encryption) is used instead, the web server s
oftware is updated and the website or web application is examined for common vulnerabilitie
s!
```

Figure 19. Platform operation - Module 6: Searching for open ports and security issues.

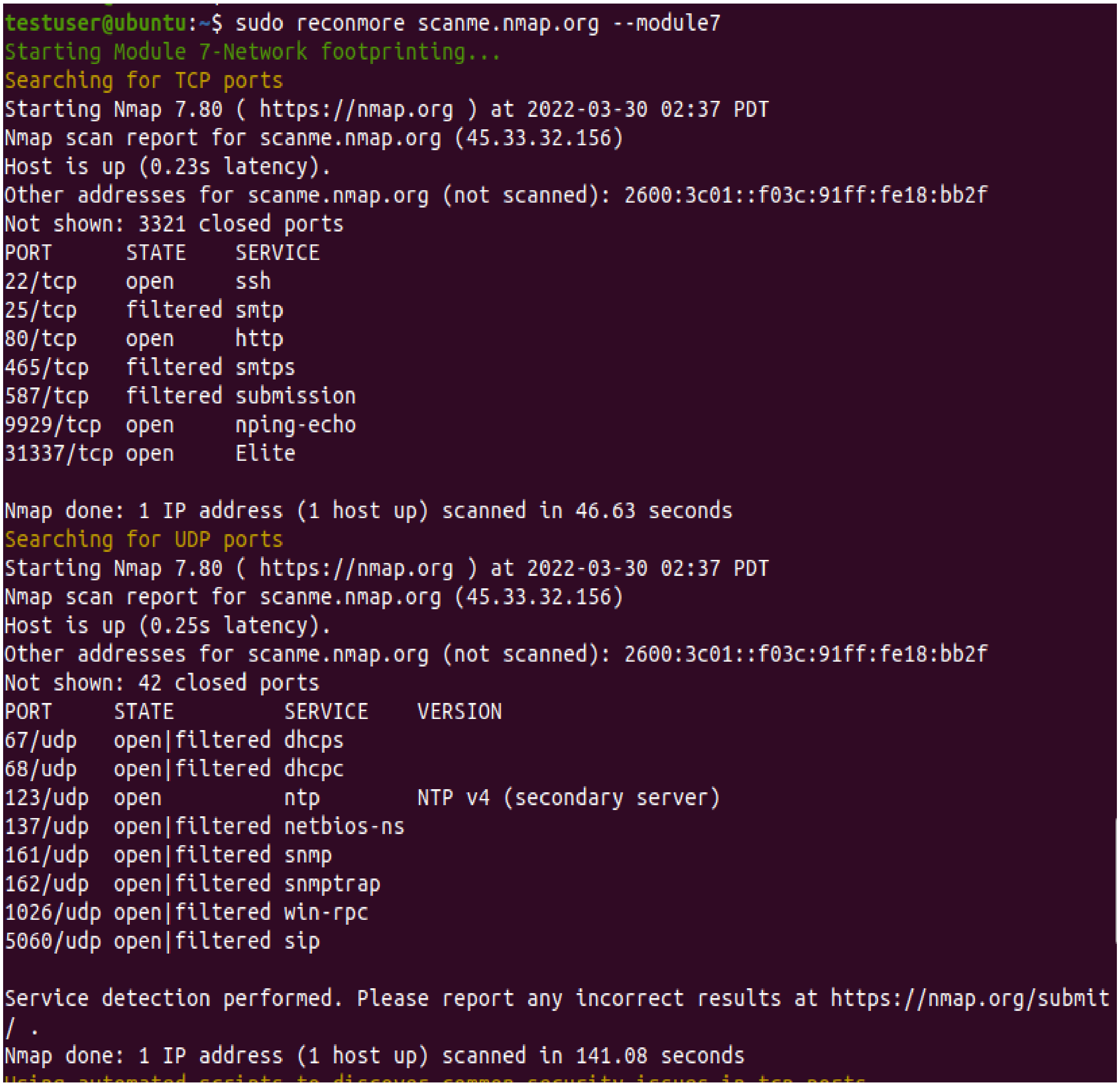


```
Using automated scripts to discover common security issues in udp ports
Starting Nmap 7.80 ( https://nmap.org ) at 2022-03-30 02:32 PDT
PORT 137: It is advised that this port should be closed and SMB in port 445 is used instead
!
PORT 68: It is advised to that DHCP MAC address filtering and MAC address check are enabled
 in the DHCP server!
PORT 138: It is advised that this port should be closed and SMB in port 445 is used instead
!
PORT 123: It is advised that this port should be closed if time synchronization is not requ
ired otherwise access should be restricted!
PORT 67: It is advised to that DHCP MAC address filtering and MAC address check are enabled
!
PORT 162: It is advised that this port uses SNMPv3 with the highest level of security and a
lso strong credentials and secure configuration of users, groups and privileges are require
d!
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.23s latency).
Other addresses for scanme.nmap.org (not scanned): 2600:3c01::f03c:91ff:fe18:bb2f

PORT    STATE         SERVICE     VERSION
67/udp  open|filtered dhcps
68/udp  open|filtered dhcpc
123/udp open          ntp         NTP v4 (secondary server)
| ntp-info:
|_
137/udp open|filtered netbios-ns
138/udp open|filtered netbios-dgm
162/udp open|filtered snmptrap

Host script results:
|_clock-skew: 8s

Service detection performed. Please report any incorrect results at https://nmap.org/submit
/ .
Nmap done: 1 IP address (1 host up) scanned in 167.51 seconds
Report saved at /usr/share/reconmore/reports/scanme.nmap.org2022-03-30T02:28:18.065647.
```

Figure 20. Platform operation - Module 6: Searching for security issues in UDP ports.

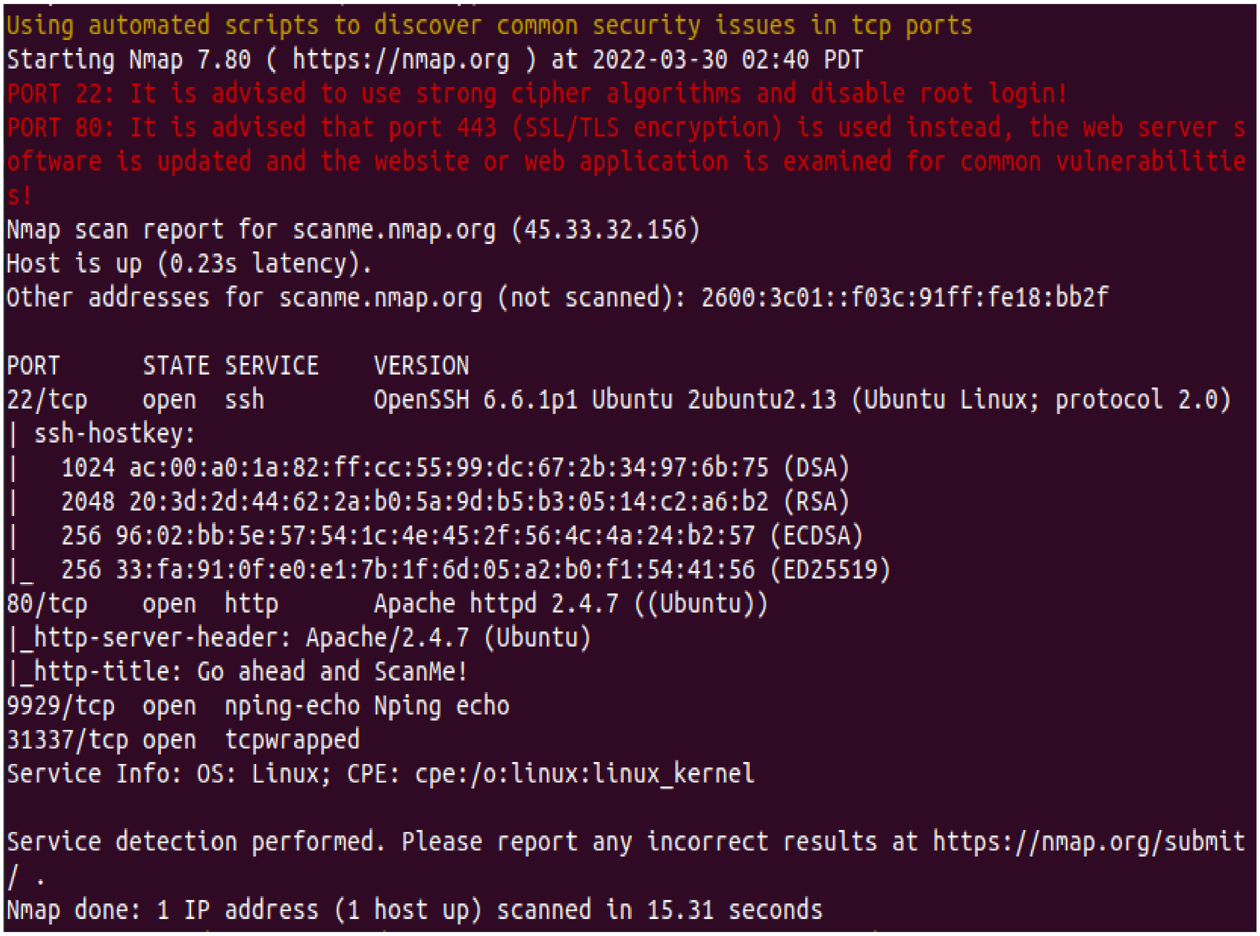
A.6 MODULE 7

The following figures show the platform operation with Module 7 against the domain *scanme.nmap.org*.



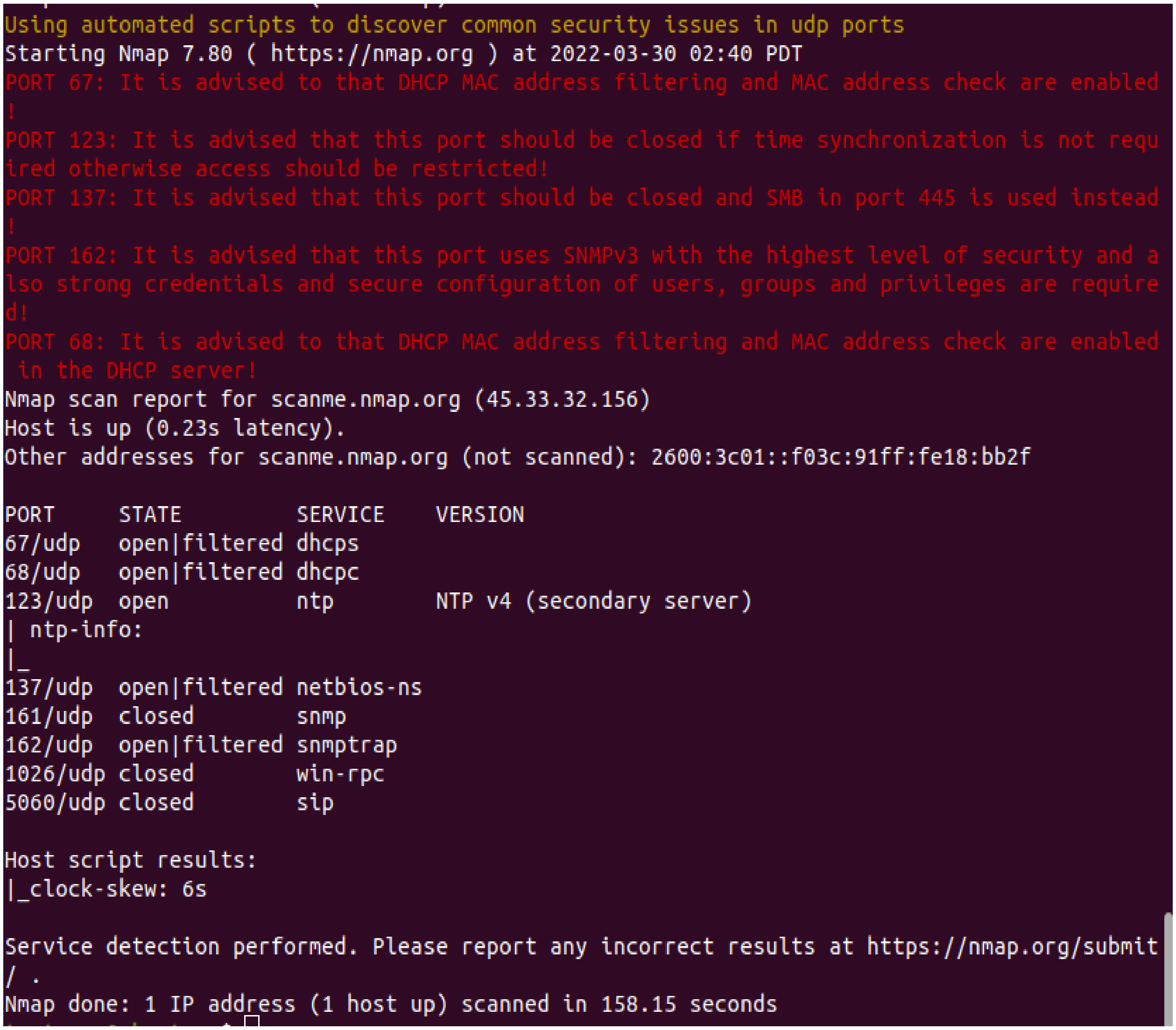
```
testuser@ubuntu:~$ sudo reconmore scanme.nmap.org --module7
Starting Module 7-Network footprinting...
Searching for TCP ports
Starting Nmap 7.80 ( https://nmap.org ) at 2022-03-30 02:37 PDT
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.23s latency).
Other addresses for scanme.nmap.org (not scanned): 2600:3c01::f03c:91ff:fe18:bb2f
Not shown: 3321 closed ports
PORT      STATE    SERVICE
22/tcp    open     ssh
25/tcp    filtered smtp
80/tcp    open     http
465/tcp   filtered smtps
587/tcp   filtered submission
9929/tcp  open     nping-echo
31337/tcp open     Elite

Nmap done: 1 IP address (1 host up) scanned in 46.63 seconds
Searching for UDP ports
Starting Nmap 7.80 ( https://nmap.org ) at 2022-03-30 02:37 PDT
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.25s latency).
Other addresses for scanme.nmap.org (not scanned): 2600:3c01::f03c:91ff:fe18:bb2f
Not shown: 42 closed ports
PORT     STATE         SERVICE    VERSION
67/udp   open|filtered dhcps
68/udp   open|filtered dhcpc
123/udp  open          ntp        NTP v4 (secondary server)
137/udp  open|filtered netbios-ns
161/udp  open|filtered snmp
162/udp  open|filtered snmptrap
1026/udp open|filtered win-rpc
5060/udp open|filtered sip

Service detection performed. Please report any incorrect results at https://nmap.org/submit
/ .
Nmap done: 1 IP address (1 host up) scanned in 141.08 seconds
Using automated scripts to discover common security issues in tcp ports
```

Figure 21. Platform operation - Module 7: Searching for open ports.



```
Using automated scripts to discover common security issues in tcp ports
Starting Nmap 7.80 ( https://nmap.org ) at 2022-03-30 02:40 PDT
PORT 22: It is advised to use strong cipher algorithms and disable root login!
PORT 80: It is advised that port 443 (SSL/TLS encryption) is used instead, the web server s
oftware is updated and the website or web application is examined for common vulnerabilitie
s!
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.23s latency).
Other addresses for scanme.nmap.org (not scanned): 2600:3c01::f03c:91ff:fe18:bb2f

PORT      STATE SERVICE    VERSION
22/tcp    open  ssh        OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.13 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   1024 ac:00:a0:1a:82:ff:cc:55:99:dc:67:2b:34:97:6b:75 (DSA)
|   2048 20:3d:2d:44:62:2a:b0:5a:9d:b5:b3:05:14:c2:a6:b2 (RSA)
|   256 96:02:bb:5e:57:54:1c:4e:45:2f:56:4c:4a:24:b2:57 (ECDSA)
|_  256 33:fa:91:0f:e0:e1:7b:1f:6d:05:a2:b0:f1:54:41:56 (ED25519)
80/tcp    open  http       Apache httpd 2.4.7 ((Ubuntu))
|_http-server-header: Apache/2.4.7 (Ubuntu)
|_http-title: Go ahead and ScanMe!
9929/tcp  open  nping-echo Nping echo
31337/tcp open  tcpwrapped
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit
/ .
Nmap done: 1 IP address (1 host up) scanned in 15.31 seconds
```

Figure 22. Platform operation - Module 7: Searching for security issues in TCP ports.

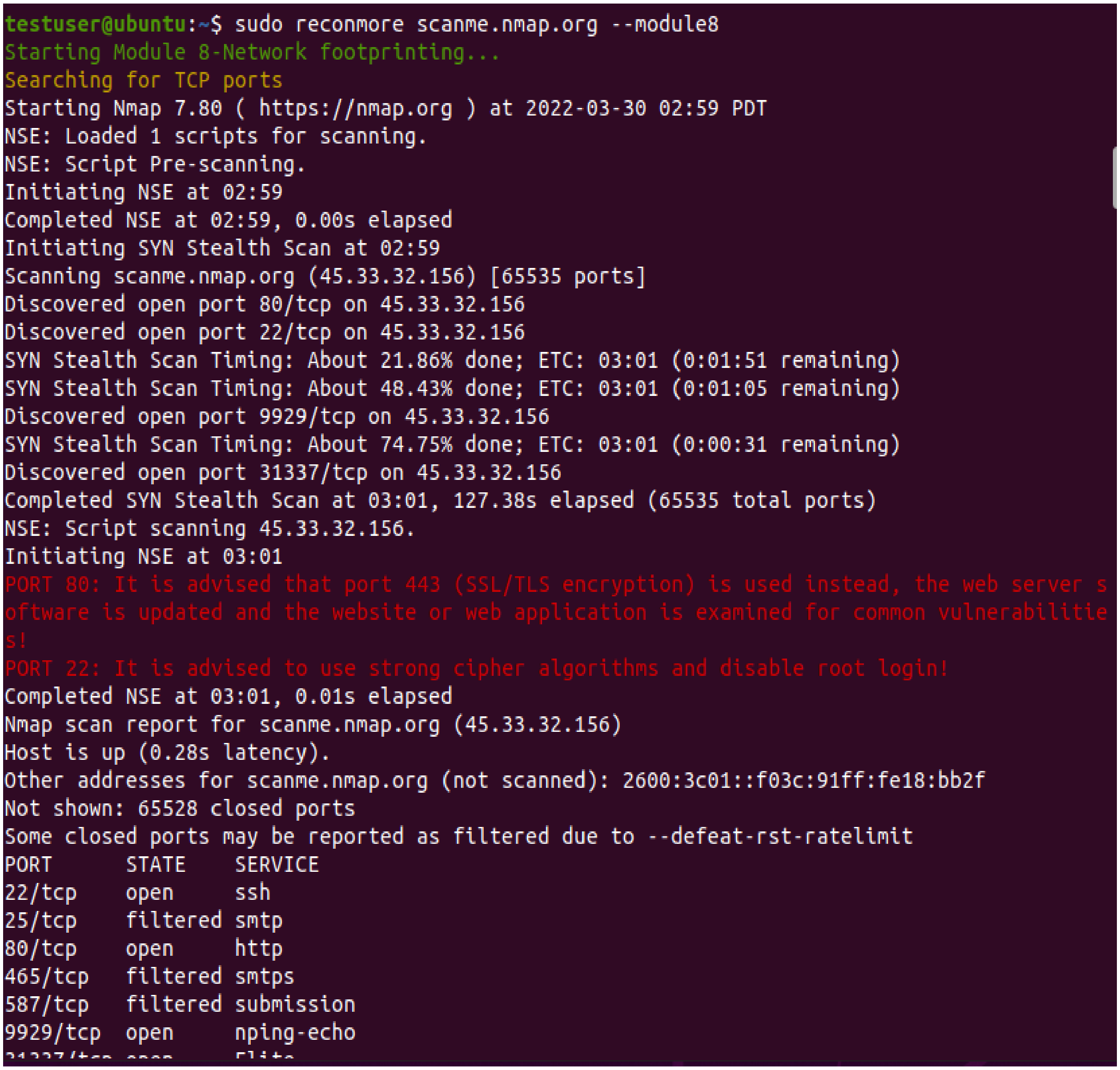


```
Using automated scripts to discover common security issues in udp ports
Starting Nmap 7.80 ( https://nmap.org ) at 2022-03-30 02:40 PDT
PORT 67: It is advised to that DHCP MAC address filtering and MAC address check are enabled
!
PORT 123: It is advised that this port should be closed if time synchronization is not requ
ired otherwise access should be restricted!
PORT 137: It is advised that this port should be closed and SMB in port 445 is used instead
!
PORT 162: It is advised that this port uses SNMPv3 with the highest level of security and a
lso strong credentials and secure configuration of users, groups and privileges are require
d!
PORT 68: It is advised to that DHCP MAC address filtering and MAC address check are enabled
 in the DHCP server!
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.23s latency).
Other addresses for scanme.nmap.org (not scanned): 2600:3c01::f03c:91ff:fe18:bb2f

PORT     STATE         SERVICE    VERSION
67/udp   open|filtered dhcps
68/udp   open|filtered dhcpc
123/udp  open          ntp        NTP v4 (secondary server)
| ntp-info:
|_
137/udp  open|filtered netbios-ns
161/udp  closed        snmp
162/udp  open|filtered snmptrap
1026/udp closed        win-rpc
5060/udp closed        sip

Host script results:
|_clock-skew: 6s

Service detection performed. Please report any incorrect results at https://nmap.org/submit
/ .
Nmap done: 1 IP address (1 host up) scanned in 158.15 seconds
```

Figure 23. Platform operation - Module 7: Searching for security issues in UDP ports.

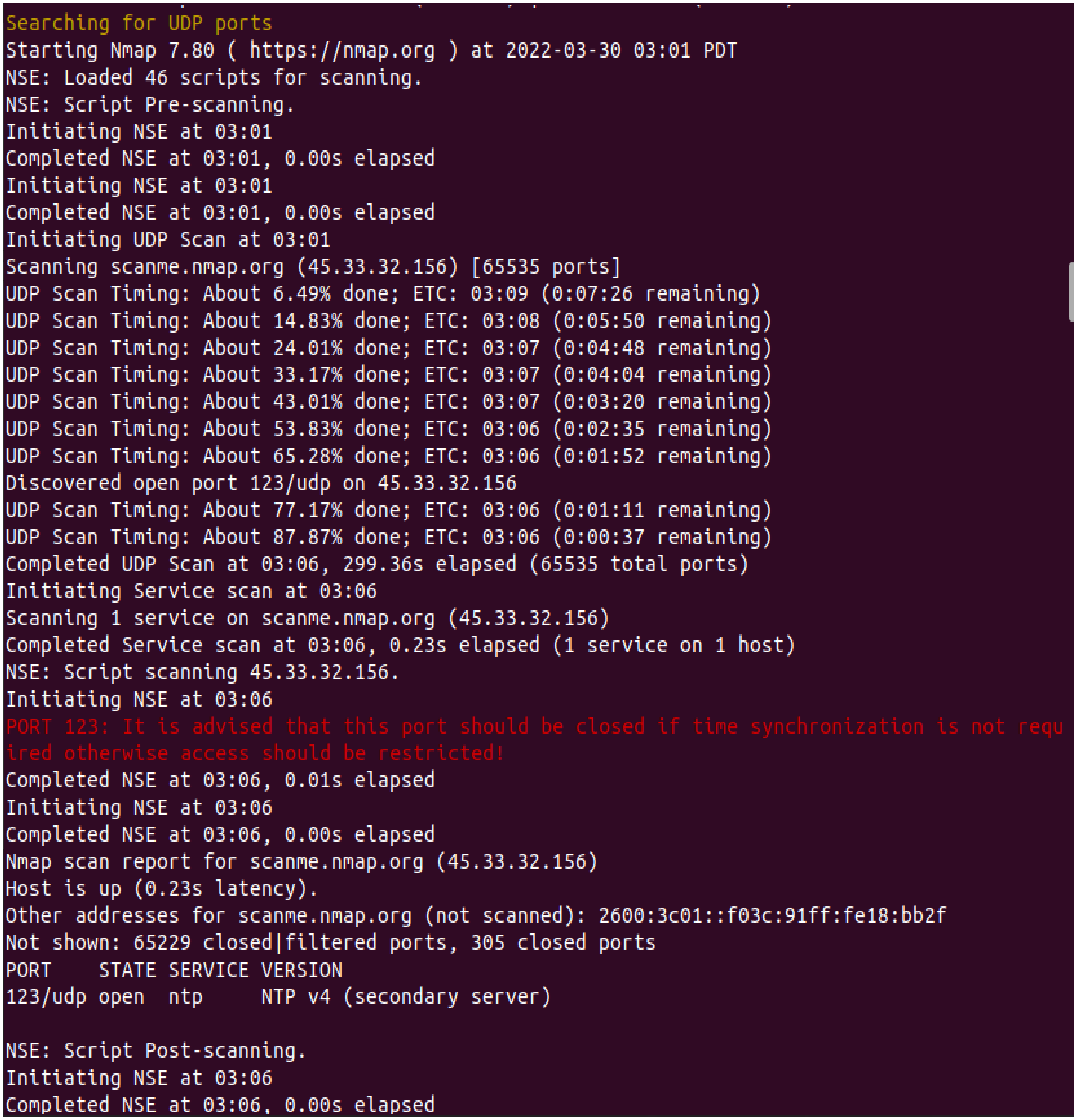
A.7 MODULE 8

The following figures show the platform operation with Module 8 against the domain *scanme.nmap.org*.



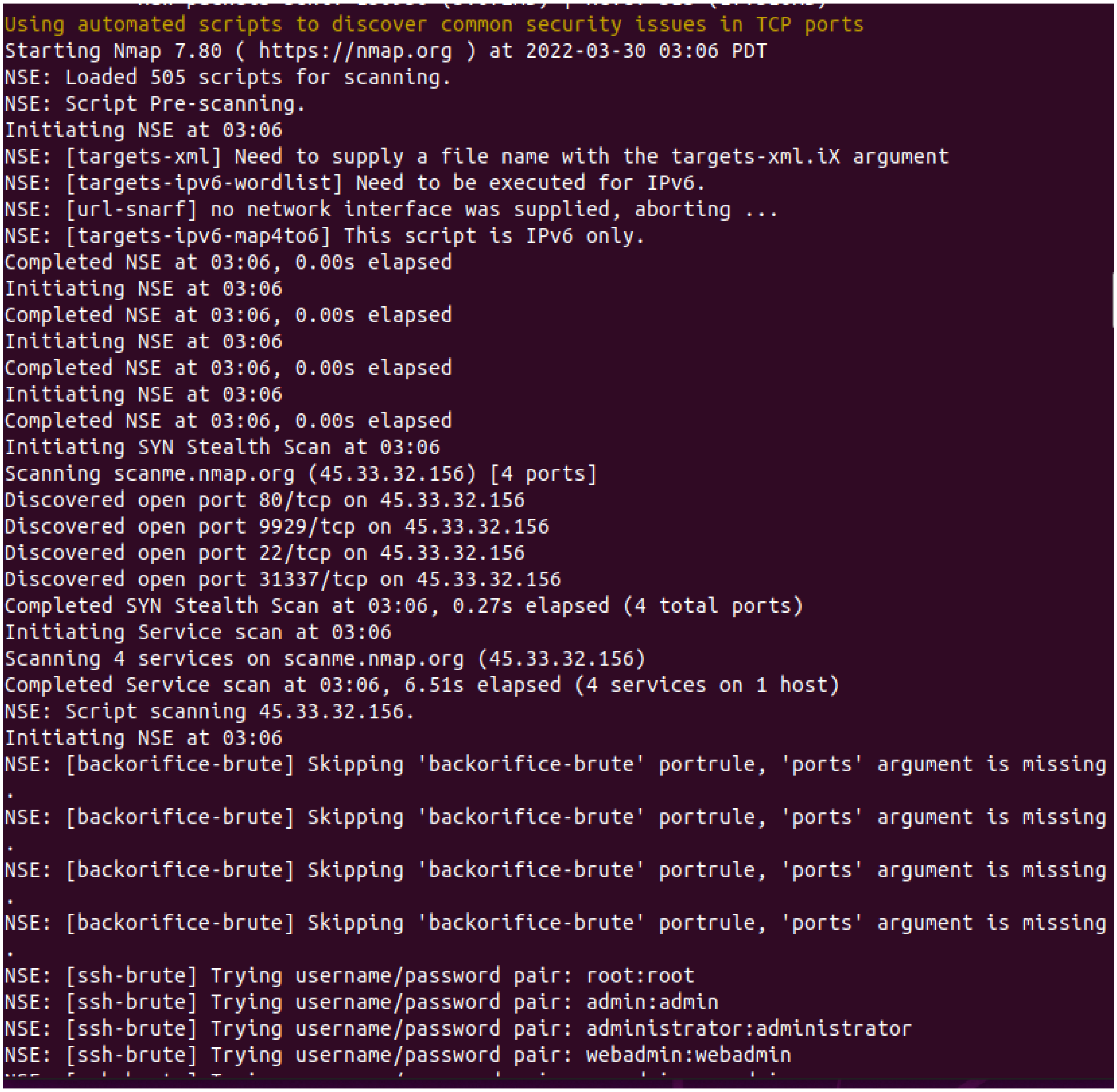
```
testuser@ubuntu:~$ sudo reconmore scanme.nmap.org --module8
Starting Module 8-Network footprinting...
Searching for TCP ports
Starting Nmap 7.80 ( https://nmap.org ) at 2022-03-30 02:59 PDT
NSE: Loaded 1 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 02:59
Completed NSE at 02:59, 0.00s elapsed
Initiating SYN Stealth Scan at 02:59
Scanning scanme.nmap.org (45.33.32.156) [65535 ports]
Discovered open port 80/tcp on 45.33.32.156
Discovered open port 22/tcp on 45.33.32.156
SYN Stealth Scan Timing: About 21.86% done; ETC: 03:01 (0:01:51 remaining)
SYN Stealth Scan Timing: About 48.43% done; ETC: 03:01 (0:01:05 remaining)
Discovered open port 9929/tcp on 45.33.32.156
SYN Stealth Scan Timing: About 74.75% done; ETC: 03:01 (0:00:31 remaining)
Discovered open port 31337/tcp on 45.33.32.156
Completed SYN Stealth Scan at 03:01, 127.38s elapsed (65535 total ports)
NSE: Script scanning 45.33.32.156.
Initiating NSE at 03:01
PORT 80: It is advised that port 443 (SSL/TLS encryption) is used instead, the web server s
oftware is updated and the website or web application is examined for common vulnerabilitie
s!
PORT 22: It is advised to use strong cipher algorithms and disable root login!
Completed NSE at 03:01, 0.01s elapsed
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.28s latency).
Other addresses for scanme.nmap.org (not scanned): 2600:3c01::f03c:91ff:fe18:bb2f
Not shown: 65528 closed ports
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT      STATE    SERVICE
22/tcp    open     ssh
25/tcp    filtered smtp
80/tcp    open     http
465/tcp   filtered smtps
587/tcp   filtered submission
9929/tcp  open     nping-echo
31337/tcp open     Elite
```

Figure 24. Platform operation - Module 8: Searching for open TCP ports.



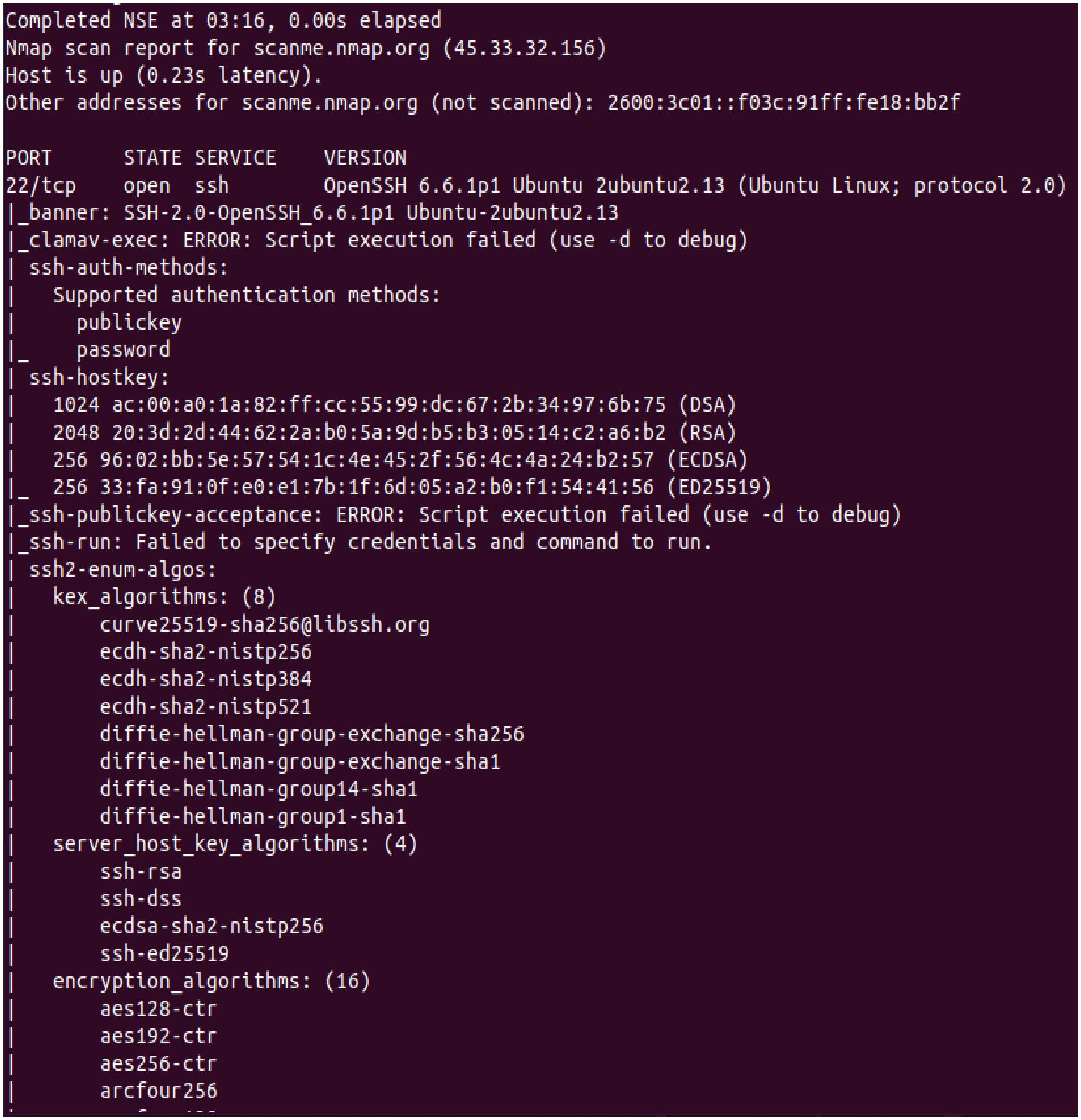
```
Searching for UDP ports
Starting Nmap 7.80 ( https://nmap.org ) at 2022-03-30 03:01 PDT
NSE: Loaded 46 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 03:01
Completed NSE at 03:01, 0.00s elapsed
Initiating NSE at 03:01
Completed NSE at 03:01, 0.00s elapsed
Initiating UDP Scan at 03:01
Scanning scanme.nmap.org (45.33.32.156) [65535 ports]
UDP Scan Timing: About 6.49% done; ETC: 03:09 (0:07:26 remaining)
UDP Scan Timing: About 14.83% done; ETC: 03:08 (0:05:50 remaining)
UDP Scan Timing: About 24.01% done; ETC: 03:07 (0:04:48 remaining)
UDP Scan Timing: About 33.17% done; ETC: 03:07 (0:04:04 remaining)
UDP Scan Timing: About 43.01% done; ETC: 03:07 (0:03:20 remaining)
UDP Scan Timing: About 53.83% done; ETC: 03:06 (0:02:35 remaining)
UDP Scan Timing: About 65.28% done; ETC: 03:06 (0:01:52 remaining)
Discovered open port 123/udp on 45.33.32.156
UDP Scan Timing: About 77.17% done; ETC: 03:06 (0:01:11 remaining)
UDP Scan Timing: About 87.87% done; ETC: 03:06 (0:00:37 remaining)
Completed UDP Scan at 03:06, 299.36s elapsed (65535 total ports)
Initiating Service scan at 03:06
Scanning 1 service on scanme.nmap.org (45.33.32.156)
Completed Service scan at 03:06, 0.23s elapsed (1 service on 1 host)
NSE: Script scanning 45.33.32.156.
Initiating NSE at 03:06
PORT 123: It is advised that this port should be closed if time synchronization is not requ
ired otherwise access should be restricted!
Completed NSE at 03:06, 0.01s elapsed
Initiating NSE at 03:06
Completed NSE at 03:06, 0.00s elapsed
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.23s latency).
Other addresses for scanme.nmap.org (not scanned): 2600:3c01::f03c:91ff:fe18:bb2f
Not shown: 65229 closed|filtered ports, 305 closed ports
PORT    STATE SERVICE VERSION
123/udp open  ntp     NTP v4 (secondary server)

NSE: Script Post-scanning.
Initiating NSE at 03:06
Completed NSE at 03:06, 0.00s elapsed
```

Figure 25. Platform operation - Module 8: Searching for open UDP ports.



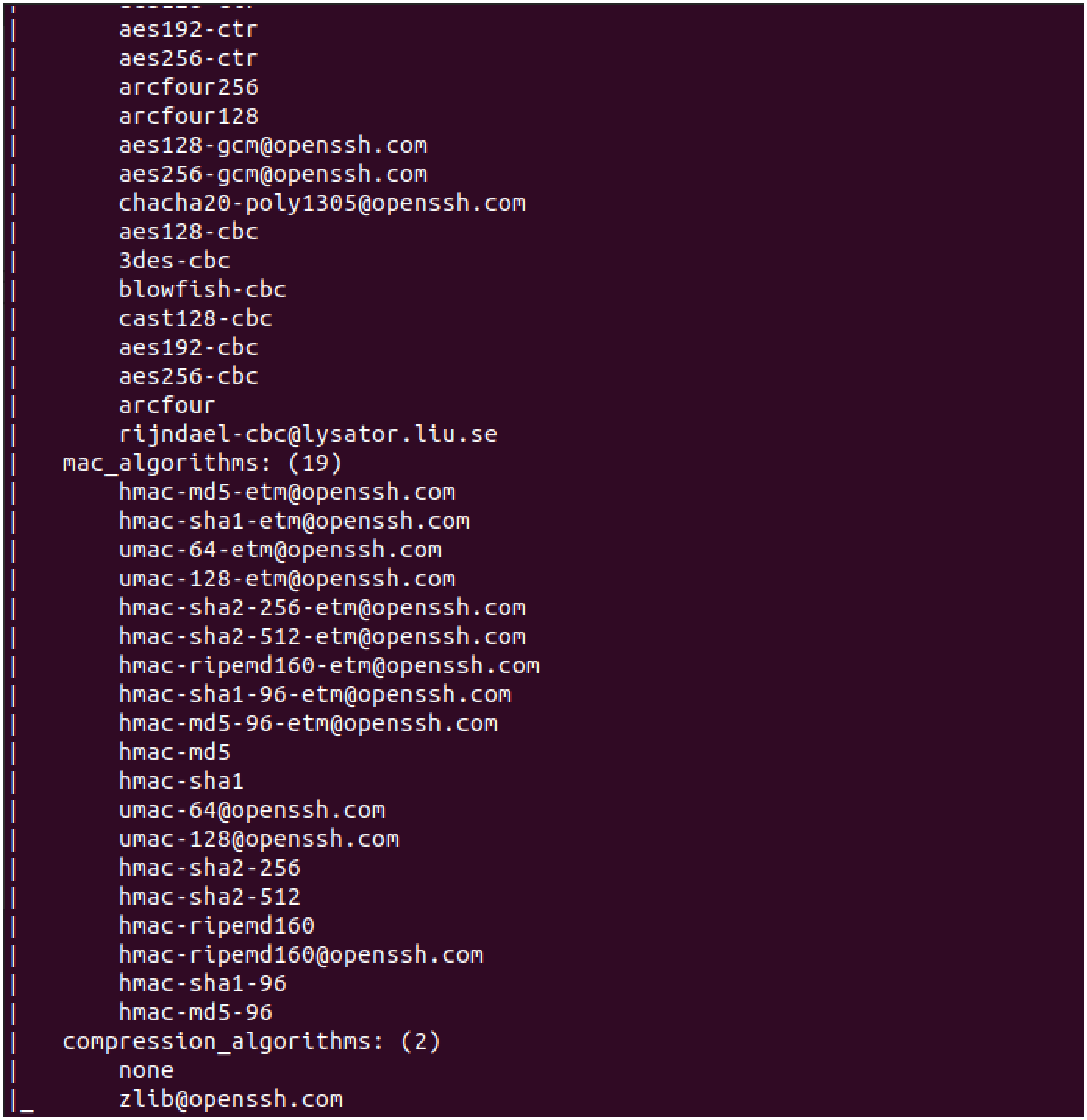
```
Using automated scripts to discover common security issues in TCP ports
Starting Nmap 7.80 ( https://nmap.org ) at 2022-03-30 03:06 PDT
NSE: Loaded 505 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 03:06
NSE: [targets-xml] Need to supply a file name with the targets-xml.iX argument
NSE: [targets-ipv6-wordlist] Need to be executed for IPv6.
NSE: [url-snarf] no network interface was supplied, aborting ...
NSE: [targets-ipv6-map4to6] This script is IPv6 only.
Completed NSE at 03:06, 0.00s elapsed
Initiating NSE at 03:06
Completed NSE at 03:06, 0.00s elapsed
Initiating NSE at 03:06
Completed NSE at 03:06, 0.00s elapsed
Initiating NSE at 03:06
Completed NSE at 03:06, 0.00s elapsed
Initiating SYN Stealth Scan at 03:06
Scanning scanme.nmap.org (45.33.32.156) [4 ports]
Discovered open port 80/tcp on 45.33.32.156
Discovered open port 9929/tcp on 45.33.32.156
Discovered open port 22/tcp on 45.33.32.156
Discovered open port 31337/tcp on 45.33.32.156
Completed SYN Stealth Scan at 03:06, 0.27s elapsed (4 total ports)
Initiating Service scan at 03:06
Scanning 4 services on scanme.nmap.org (45.33.32.156)
Completed Service scan at 03:06, 6.51s elapsed (4 services on 1 host)
NSE: Script scanning 45.33.32.156.
Initiating NSE at 03:06
NSE: [backorifice-brute] Skipping 'backorifice-brute' portrule, 'ports' argument is missing
.
NSE: [backorifice-brute] Skipping 'backorifice-brute' portrule, 'ports' argument is missing
.
NSE: [backorifice-brute] Skipping 'backorifice-brute' portrule, 'ports' argument is missing
.
NSE: [backorifice-brute] Skipping 'backorifice-brute' portrule, 'ports' argument is missing
.
NSE: [ssh-brute] Trying username/password pair: root:root
NSE: [ssh-brute] Trying username/password pair: admin:admin
NSE: [ssh-brute] Trying username/password pair: administrator:administrator
NSE: [ssh-brute] Trying username/password pair: webadmin:webadmin
```

Figure 26. Platform operation - Module 8: Searching for security issues in TCP ports.



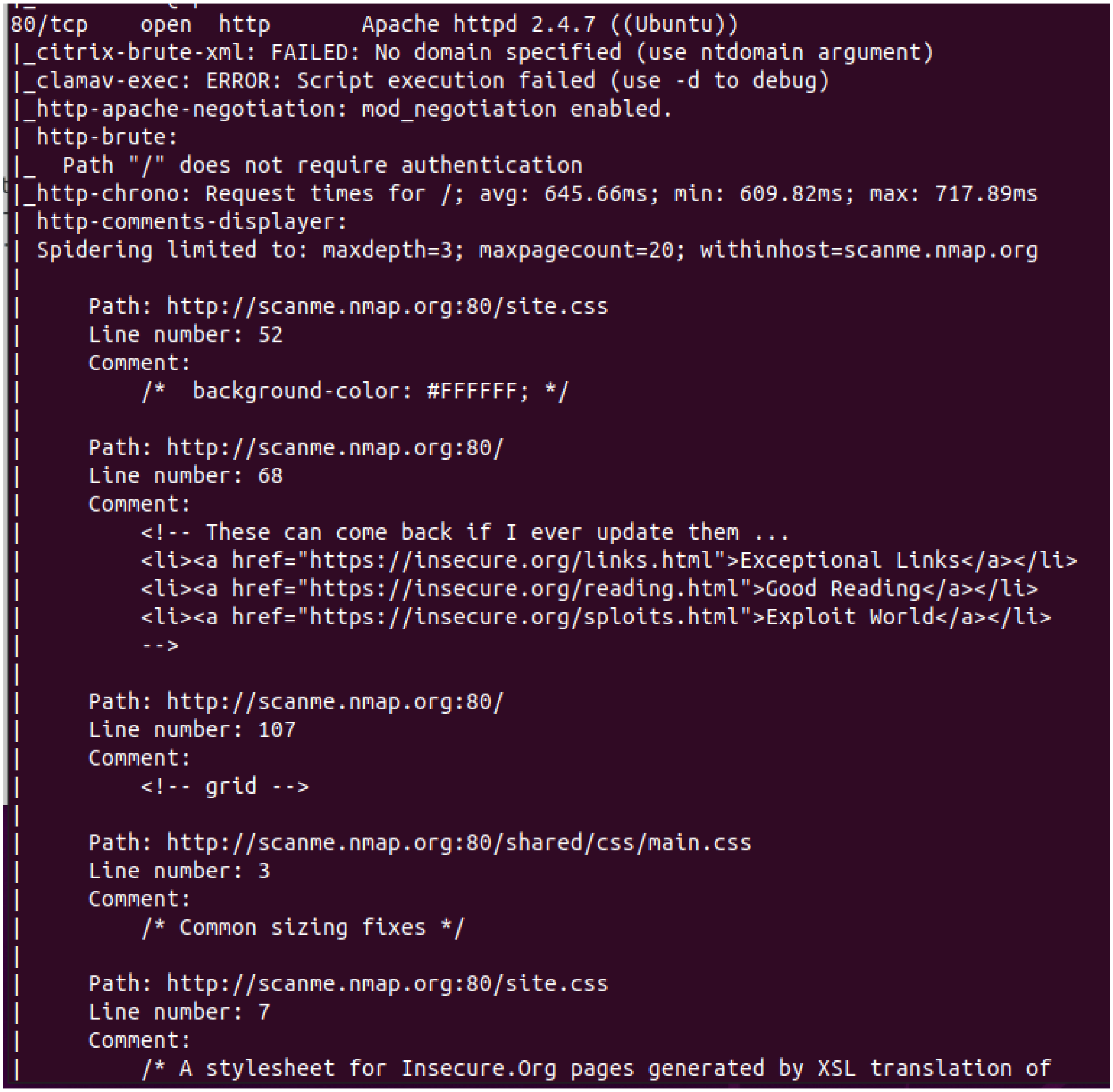
```
Completed NSE at 03:16, 0.00s elapsed
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.23s latency).
Other addresses for scanme.nmap.org (not scanned): 2600:3c01::f03c:91ff:fe18:bb2f

PORT     STATE SERVICE    VERSION
22/tcp   open  ssh        OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.13 (Ubuntu Linux; protocol 2.0)
|_banner: SSH-2.0-OpenSSH_6.6.1p1 Ubuntu-2ubuntu2.13
|_clamav-exec: ERROR: Script execution failed (use -d to debug)
| ssh-auth-methods:
|   Supported authentication methods:
|     publickey
|_    password
| ssh-hostkey:
|   1024 ac:00:a0:1a:82:ff:cc:55:99:dc:67:2b:34:97:6b:75 (DSA)
|   2048 20:3d:2d:44:62:2a:b0:5a:9d:b5:b3:05:14:c2:a6:b2 (RSA)
|   256 96:02:bb:5e:57:54:1c:4e:45:2f:56:4c:4a:24:b2:57 (ECDSA)
|_  256 33:fa:91:0f:e0:e1:7b:1f:6d:05:a2:b0:f1:54:41:56 (ED25519)
|_ssh-publickey-acceptance: ERROR: Script execution failed (use -d to debug)
|_ssh-run: Failed to specify credentials and command to run.
| ssh2-enum-algos:
|   kex_algorithms: (8)
|       curve25519-sha256@libssh.org
|       ecdh-sha2-nistp256
|       ecdh-sha2-nistp384
|       ecdh-sha2-nistp521
|       diffie-hellman-group-exchange-sha256
|       diffie-hellman-group-exchange-sha1
|       diffie-hellman-group14-sha1
|       diffie-hellman-group1-sha1
|   server_host_key_algorithms: (4)
|       ssh-rsa
|       ssh-dss
|       ecdsa-sha2-nistp256
|       ssh-ed25519
|   encryption_algorithms: (16)
|       aes128-ctr
|       aes192-ctr
|       aes256-ctr
|       arcfour256
```

Figure 27. Platform operation - Module 8: Results after scanning with automated scripts.



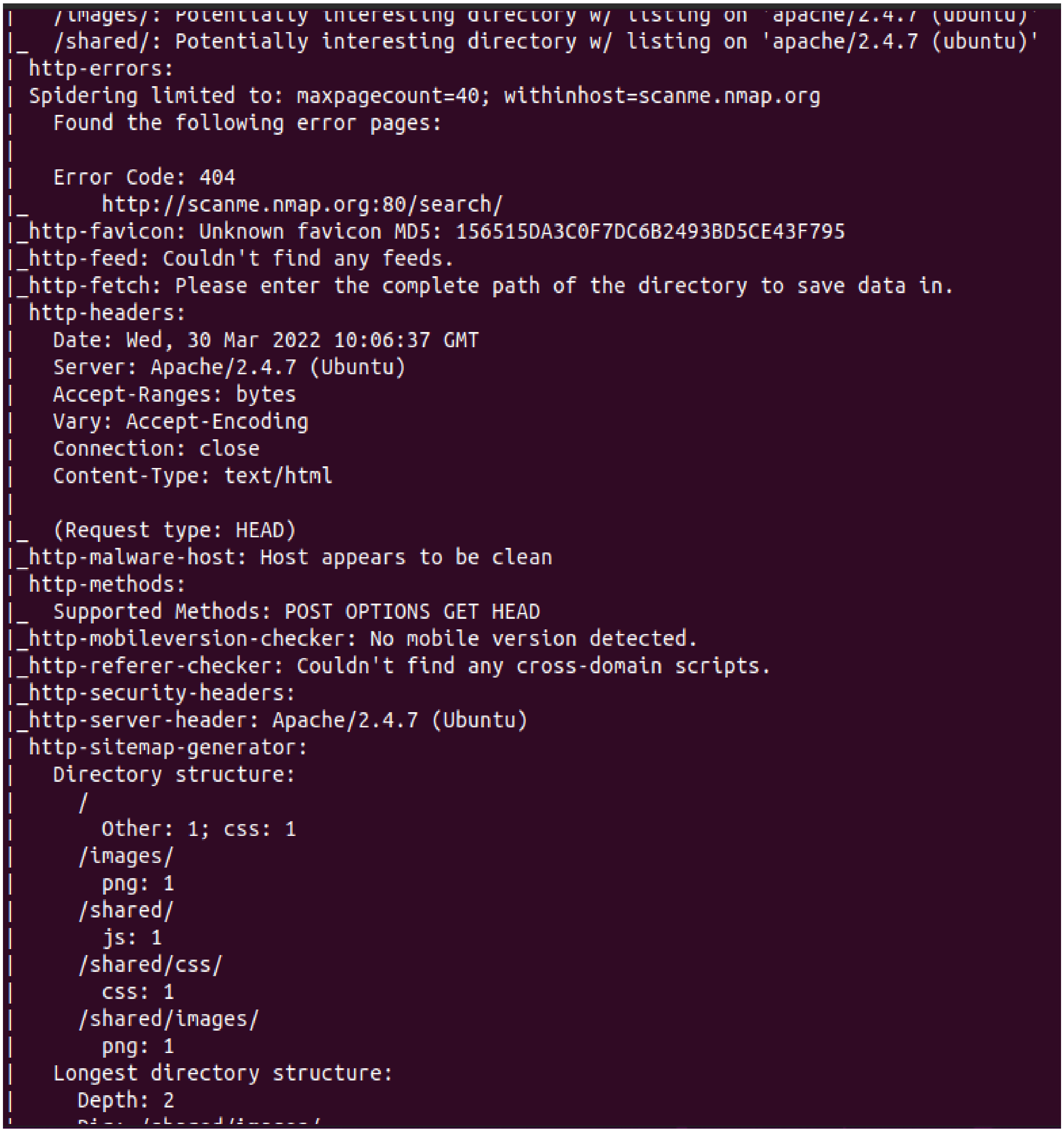
```
|       aes192-ctr
|       aes256-ctr
|       arcfour256
|       arcfour128
|       aes128-gcm@openssh.com
|       aes256-gcm@openssh.com
|       chacha20-poly1305@openssh.com
|       aes128-cbc
|       3des-cbc
|       blowfish-cbc
|       cast128-cbc
|       aes192-cbc
|       aes256-cbc
|       arcfour
|       rijndael-cbc@lysator.liu.se
|   mac_algorithms: (19)
|       hmac-md5-etm@openssh.com
|       hmac-sha1-etm@openssh.com
|       umac-64-etm@openssh.com
|       umac-128-etm@openssh.com
|       hmac-sha2-256-etm@openssh.com
|       hmac-sha2-512-etm@openssh.com
|       hmac-ripemd160-etm@openssh.com
|       hmac-sha1-96-etm@openssh.com
|       hmac-md5-96-etm@openssh.com
|       hmac-md5
|       hmac-sha1
|       umac-64@openssh.com
|       umac-128@openssh.com
|       hmac-sha2-256
|       hmac-sha2-512
|       hmac-ripemd160
|       hmac-ripemd160@openssh.com
|       hmac-sha1-96
|       hmac-md5-96
|   compression_algorithms: (2)
|       none
|_      zlib@openssh.com
```

Figure 28. Platform operation - Module 8: Results after scanning with automated scripts (part 2).



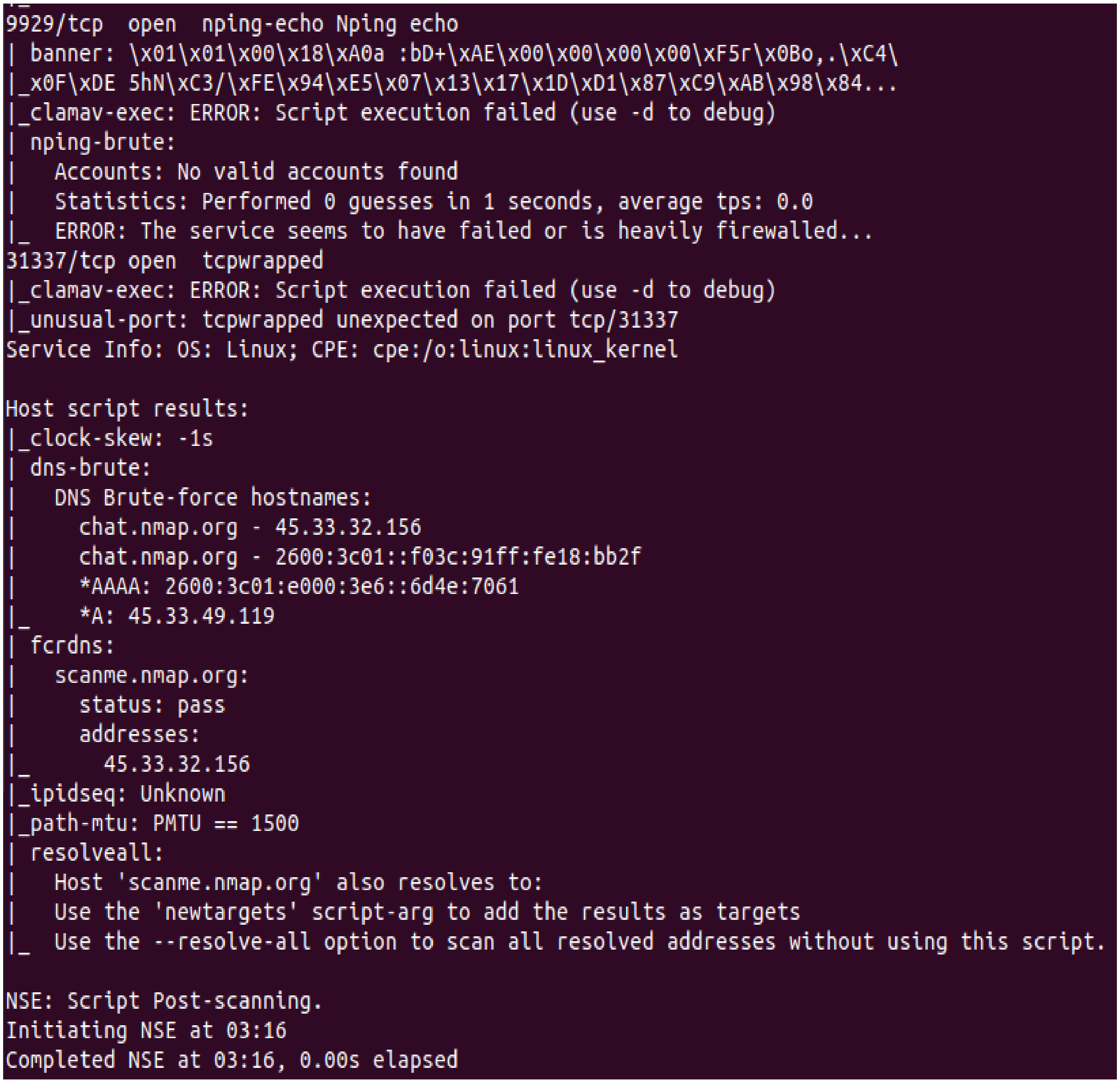
```
80/tcp    open  http       Apache httpd 2.4.7 ((Ubuntu))
|_citrix-brute-xml: FAILED: No domain specified (use ntdomain argument)
|_clamav-exec: ERROR: Script execution failed (use -d to debug)
|_http-apache-negotiation: mod_negotiation enabled.
| http-brute:
|_  Path "/" does not require authentication
|_http-chrono: Request times for /; avg: 645.66ms; min: 609.82ms; max: 717.89ms
| http-comments-displayer:
| Spidering limited to: maxdepth=3; maxpagecount=20; withinhost=scanme.nmap.org
|
|     Path: http://scanme.nmap.org:80/site.css
|     Line number: 52
|     Comment:
|         /*  background-color: #FFFFFF; */
|
|     Path: http://scanme.nmap.org:80/
|     Line number: 68
|     Comment:
|         <!-- These can come back if I ever update them ...
|         <li><a href="https://insecure.org/links.html">Exceptional Links</a></li>
|         <li><a href="https://insecure.org/reading.html">Good Reading</a></li>
|         <li><a href="https://insecure.org/sploits.html">Exploit World</a></li>
|         -->
|
|     Path: http://scanme.nmap.org:80/
|     Line number: 107
|     Comment:
|         <!-- grid -->
|
|     Path: http://scanme.nmap.org:80/shared/css/main.css
|     Line number: 3
|     Comment:
|         /* Common sizing fixes */
|
|     Path: http://scanme.nmap.org:80/site.css
|     Line number: 7
|     Comment:
|         /* A stylesheet for Insecure.Org pages generated by XSL translation of
```

Figure 29. Platform operation - Module 8: Results after scanning with automated scripts (part 3).



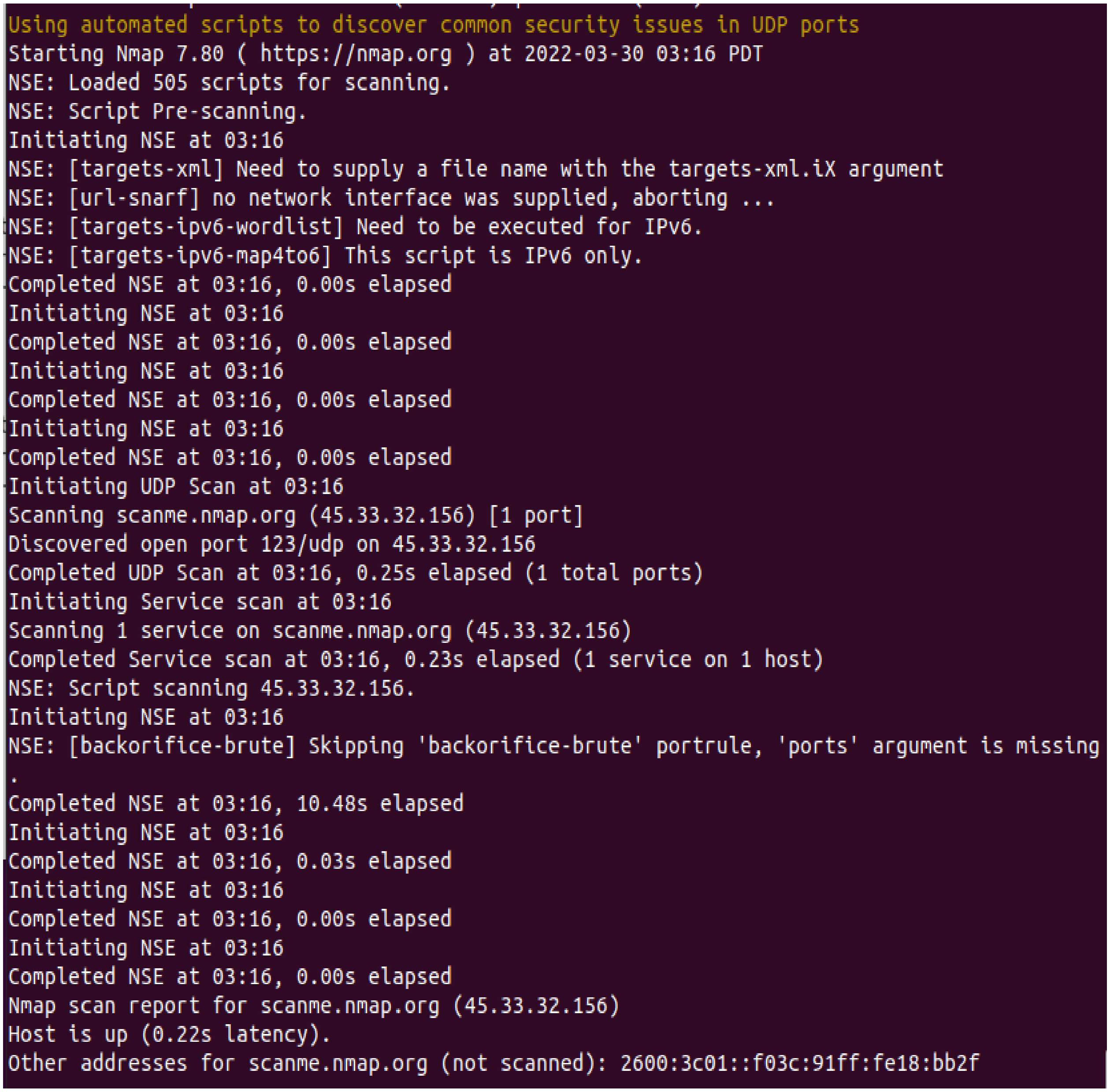
```
|   /images/: Potentially interesting directory w/ listing on 'apache/2.4.7 (ubuntu)'
|_  /shared/: Potentially interesting directory w/ listing on 'apache/2.4.7 (ubuntu)'
| http-errors:
| Spidering limited to: maxpagecount=40; withinhost=scanme.nmap.org
|   Found the following error pages:
|
|   Error Code: 404
|_      http://scanme.nmap.org:80/search/
|_http-favicon: Unknown favicon MD5: 156515DA3C0F7DC6B2493BD5CE43F795
|_http-feed: Couldn't find any feeds.
|_http-fetch: Please enter the complete path of the directory to save data in.
| http-headers:
|   Date: Wed, 30 Mar 2022 10:06:37 GMT
|   Server: Apache/2.4.7 (Ubuntu)
|   Accept-Ranges: bytes
|   Vary: Accept-Encoding
|   Connection: close
|   Content-Type: text/html
|
|_  (Request type: HEAD)
|_http-malware-host: Host appears to be clean
| http-methods:
|_  Supported Methods: POST OPTIONS GET HEAD
|_http-mobileversion-checker: No mobile version detected.
|_http-referer-checker: Couldn't find any cross-domain scripts.
|_http-security-headers:
|_http-server-header: Apache/2.4.7 (Ubuntu)
| http-sitemap-generator:
|   Directory structure:
|     /
|       Other: 1; css: 1
|     /images/
|       png: 1
|     /shared/
|       js: 1
|     /shared/css/
|       css: 1
|     /shared/images/
|       png: 1
|   Longest directory structure:
|     Depth: 2
|     Dir: /shared/images/
```

Figure 30. Platform operation - Module 8: Results after scanning with automated scripts (part 4).



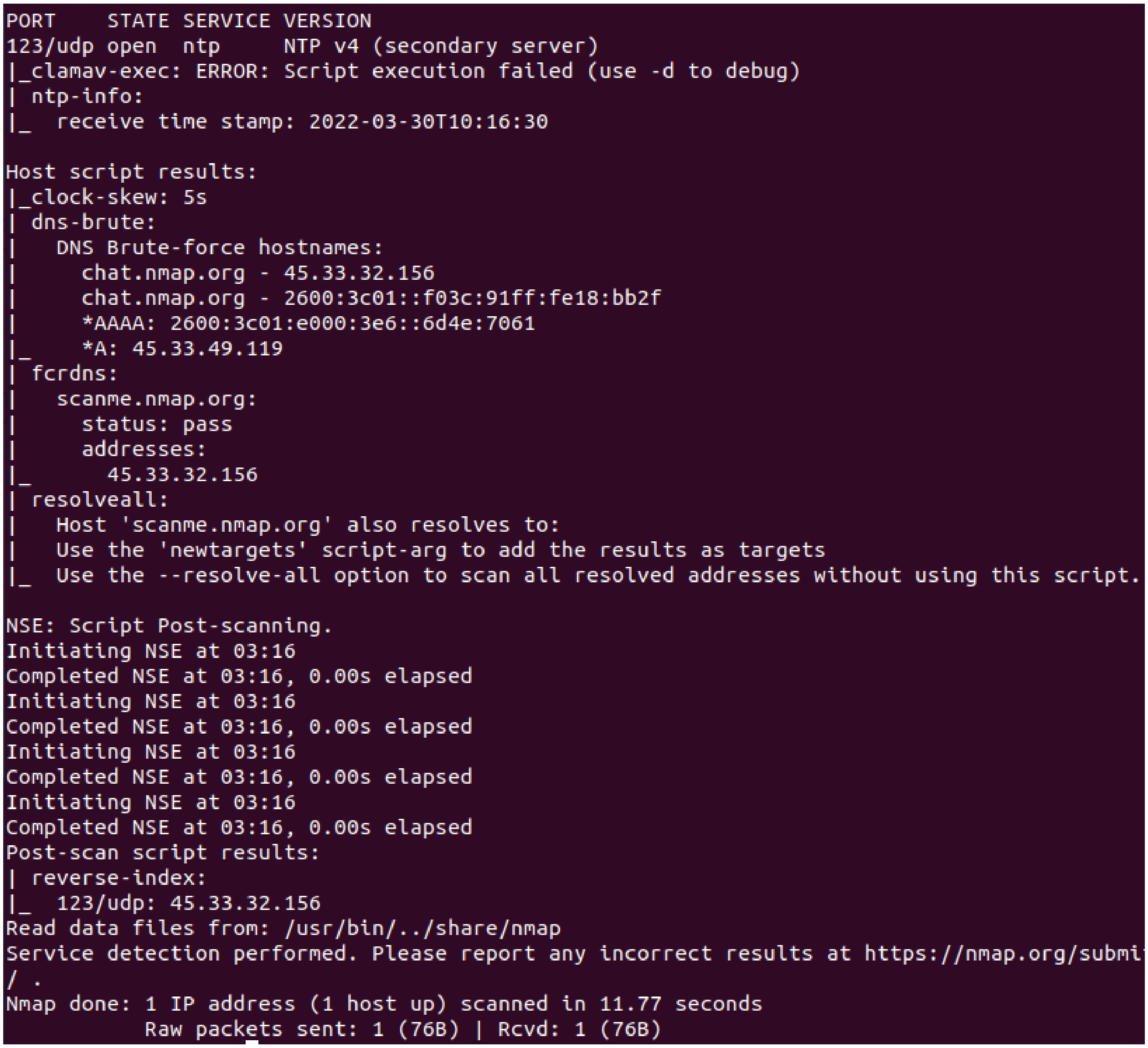
```
|_
9929/tcp  open  nping-echo Nping echo
| banner: \x01\x01\x00\x18\xA0a :bD+\xAE\x00\x00\x00\x00\xF5r\x0Bo,.\xC4\
|_x0F\xDE 5hN\xC3/\xFE\x94\xE5\x07\x13\x17\x1D\xD1\x87\xC9\xAB\x98\x84...
|_clamav-exec: ERROR: Script execution failed (use -d to debug)
| nping-brute:
|   Accounts: No valid accounts found
|   Statistics: Performed 0 guesses in 1 seconds, average tps: 0.0
|_  ERROR: The service seems to have failed or is heavily firewalled...
31337/tcp open  tcpwrapped
|_clamav-exec: ERROR: Script execution failed (use -d to debug)
|_unusual-port: tcpwrapped unexpected on port tcp/31337
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Host script results:
|_clock-skew: -1s
| dns-brute:
|   DNS Brute-force hostnames:
|     chat.nmap.org - 45.33.32.156
|     chat.nmap.org - 2600:3c01::f03c:91ff:fe18:bb2f
|     *AAAA: 2600:3c01:e000:3e6::6d4e:7061
|_    *A: 45.33.49.119
| fcrdns:
|   scanme.nmap.org:
|     status: pass
|     addresses:
|_      45.33.32.156
|_ipidseq: Unknown
|_path-mtu: PMTU == 1500
| resolveall:
|   Host 'scanme.nmap.org' also resolves to:
|   Use the 'newtargets' script-arg to add the results as targets
|_  Use the --resolve-all option to scan all resolved addresses without using this script.

NSE: Script Post-scanning.
Initiating NSE at 03:16
Completed NSE at 03:16, 0.00s elapsed
```

Figure 31. Platform operation - Module 8: Results after scanning with automated scripts (part 5).



```
Using automated scripts to discover common security issues in UDP ports
Starting Nmap 7.80 ( https://nmap.org ) at 2022-03-30 03:16 PDT
NSE: Loaded 505 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 03:16
NSE: [targets-xml] Need to supply a file name with the targets-xml.iX argument
NSE: [url-snarf] no network interface was supplied, aborting ...
NSE: [targets-ipv6-wordlist] Need to be executed for IPv6.
NSE: [targets-ipv6-map4to6] This script is IPv6 only.
Completed NSE at 03:16, 0.00s elapsed
Initiating NSE at 03:16
Completed NSE at 03:16, 0.00s elapsed
Initiating NSE at 03:16
Completed NSE at 03:16, 0.00s elapsed
Initiating NSE at 03:16
Completed NSE at 03:16, 0.00s elapsed
Initiating UDP Scan at 03:16
Scanning scanme.nmap.org (45.33.32.156) [1 port]
Discovered open port 123/udp on 45.33.32.156
Completed UDP Scan at 03:16, 0.25s elapsed (1 total ports)
Initiating Service scan at 03:16
Scanning 1 service on scanme.nmap.org (45.33.32.156)
Completed Service scan at 03:16, 0.23s elapsed (1 service on 1 host)
NSE: Script scanning 45.33.32.156.
Initiating NSE at 03:16
NSE: [backorifice-brute] Skipping 'backorifice-brute' portrule, 'ports' argument is missing
.
Completed NSE at 03:16, 10.48s elapsed
Initiating NSE at 03:16
Completed NSE at 03:16, 0.03s elapsed
Initiating NSE at 03:16
Completed NSE at 03:16, 0.00s elapsed
Initiating NSE at 03:16
Completed NSE at 03:16, 0.00s elapsed
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.22s latency).
Other addresses for scanme.nmap.org (not scanned): 2600:3c01::f03c:91ff:fe18:bb2f
```

Figure 32. Platform operation - Module 8: Searching for security issues in UDP ports.

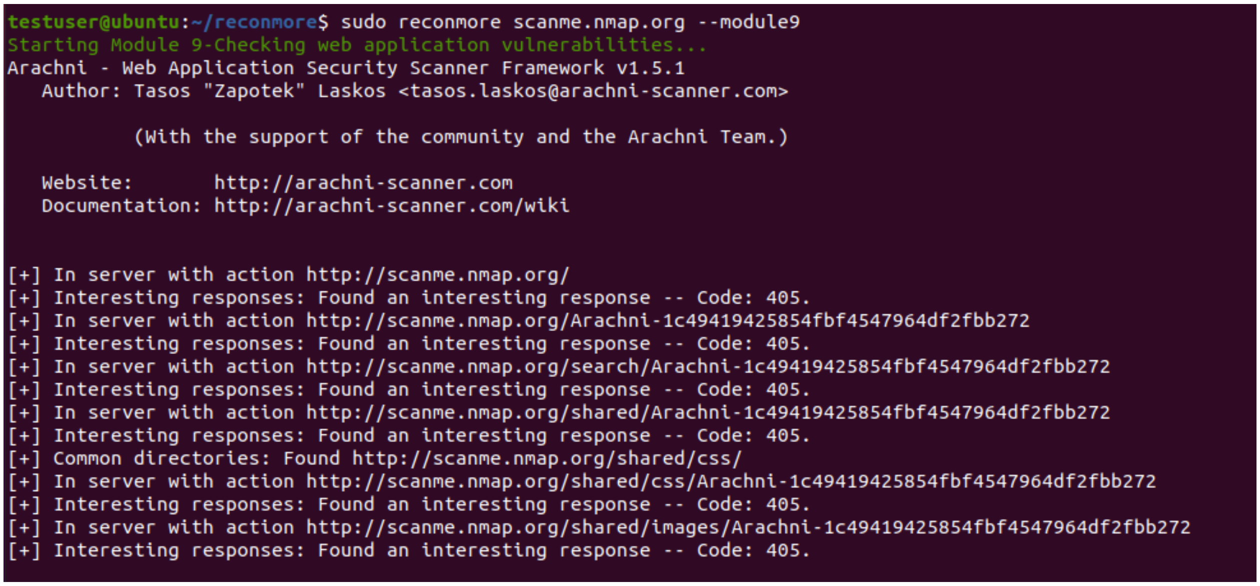


```
PORT    STATE SERVICE VERSION
123/udp open  ntp     NTP v4 (secondary server)
|_clamav-exec: ERROR: Script execution failed (use -d to debug)
| ntp-info:
|_  receive time stamp: 2022-03-30T10:16:30

Host script results:
|_clock-skew: 5s
| dns-brute:
|   DNS Brute-force hostnames:
|     chat.nmap.org - 45.33.32.156
|     chat.nmap.org - 2600:3c01::f03c:91ff:fe18:bb2f
|     *AAAA: 2600:3c01:e000:3e6::6d4e:7061
|_    *A: 45.33.49.119
| fcrdns:
|   scanme.nmap.org:
|     status: pass
|     addresses:
|_      45.33.32.156
| resolveall:
|   Host 'scanme.nmap.org' also resolves to:
|   Use the 'newtargets' script-arg to add the results as targets
|_  Use the --resolve-all option to scan all resolved addresses without using this script.

NSE: Script Post-scanning.
Initiating NSE at 03:16
Completed NSE at 03:16, 0.00s elapsed
Initiating NSE at 03:16
Completed NSE at 03:16, 0.00s elapsed
Initiating NSE at 03:16
Completed NSE at 03:16, 0.00s elapsed
Initiating NSE at 03:16
Completed NSE at 03:16, 0.00s elapsed
Post-scan script results:
| reverse-index:
|_  123/udp: 45.33.32.156
Read data files from: /usr/bin/../share/nmap
Service detection performed. Please report any incorrect results at https://nmap.org/submi
/ .
Nmap done: 1 IP address (1 host up) scanned in 11.77 seconds
           Raw packets sent: 1 (76B) | Rcvd: 1 (76B)
```

Figure 33. Platform operation - Module 8: Results after scanning with automated scripts.

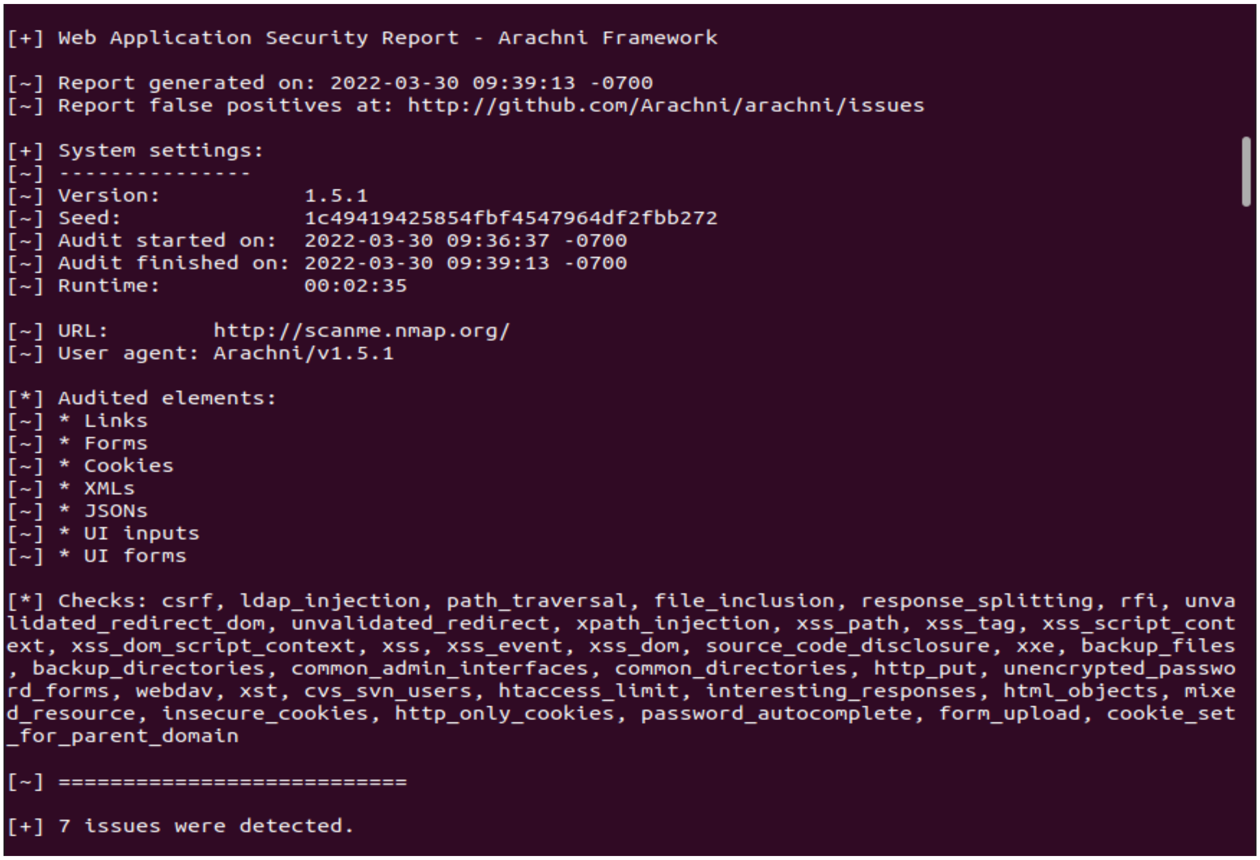
A.8 MODULE 9

The following figures show the platform operation with Module 9 against the domain *scanme.nmap.org*.



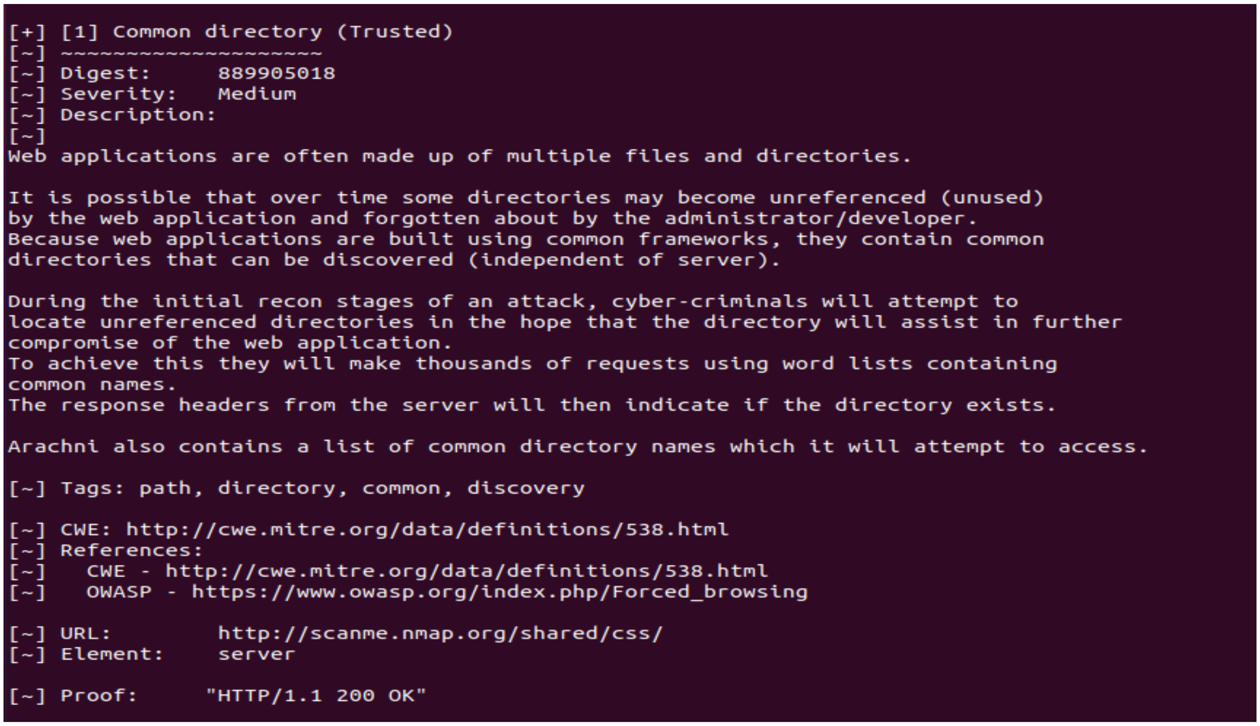
```
testuser@ubuntu:~/reconmore$ sudo reconmore scanme.nmap.org --module9
Starting Module 9-Checking web application vulnerabilities...
Arachni - Web Application Security Scanner Framework v1.5.1
   Author: Tasos "Zapotek" Laskos <tasos.laskos@arachni-scanner.com>

           (With the support of the community and the Arachni Team.)

   Website:       http://arachni-scanner.com
   Documentation: http://arachni-scanner.com/wiki


[+] In server with action http://scanme.nmap.org/
[+] Interesting responses: Found an interesting response -- Code: 405.
[+] In server with action http://scanme.nmap.org/Arachni-1c49419425854fbf4547964df2fbb272
[+] Interesting responses: Found an interesting response -- Code: 405.
[+] In server with action http://scanme.nmap.org/search/Arachni-1c49419425854fbf4547964df2fbb272
[+] Interesting responses: Found an interesting response -- Code: 405.
[+] In server with action http://scanme.nmap.org/shared/Arachni-1c49419425854fbf4547964df2fbb272
[+] Interesting responses: Found an interesting response -- Code: 405.
[+] Common directories: Found http://scanme.nmap.org/shared/css/
[+] In server with action http://scanme.nmap.org/shared/css/Arachni-1c49419425854fbf4547964df2fbb272
[+] Interesting responses: Found an interesting response -- Code: 405.
[+] In server with action http://scanme.nmap.org/shared/images/Arachni-1c49419425854fbf4547964df2fbb272
[+] Interesting responses: Found an interesting response -- Code: 405.
```

Figure 34. Platform operation - Module 9: Checking for web application vulnerabilities.



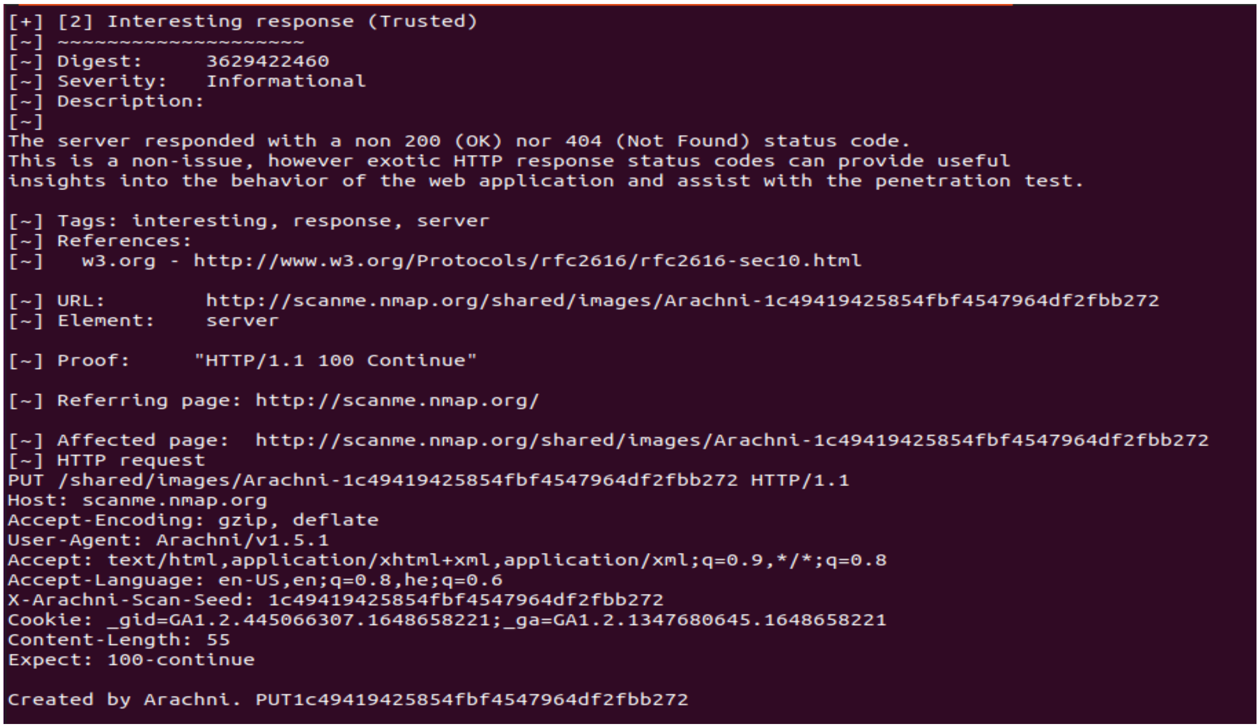
```
[+] Web Application Security Report - Arachni Framework

[~] Report generated on: 2022-03-30 09:39:13 -0700
[~] Report false positives at: http://github.com/Arachni/arachni/issues

[+] System settings:
[~] ---------------
[~] Version:           1.5.1
[~] Seed:              1c49419425854fbf4547964df2fbb272
[~] Audit started on:  2022-03-30 09:36:37 -0700
[~] Audit finished on: 2022-03-30 09:39:13 -0700
[~] Runtime:           00:02:35

[~] URL:        http://scanme.nmap.org/
[~] User agent: Arachni/v1.5.1

[*] Audited elements:
[~] * Links
[~] * Forms
[~] * Cookies
[~] * XMLs
[~] * JSONs
[~] * UI inputs
[~] * UI forms

[*] Checks: csrf, ldap_injection, path_traversal, file_inclusion, response_splitting, rfi, unva
lidated_redirect_dom, unvalidated_redirect, xpath_injection, xss_path, xss_tag, xss_script_cont
ext, xss_dom_script_context, xss, xss_event, xss_dom, source_code_disclosure, xxe, backup_files
, backup_directories, common_admin_interfaces, common_directories, http_put, unencrypted_passwo
rd_forms, webdav, xst, cvs_svn_users, htaccess_limit, interesting_responses, html_objects, mixe
d_resource, insecure_cookies, http_only_cookies, password_autocomplete, form_upload, cookie_set
_for_parent_domain

[~] ==========================

[+] 7 issues were detected.
```

Figure 35. Platform operation - Module 9: Web application security report.



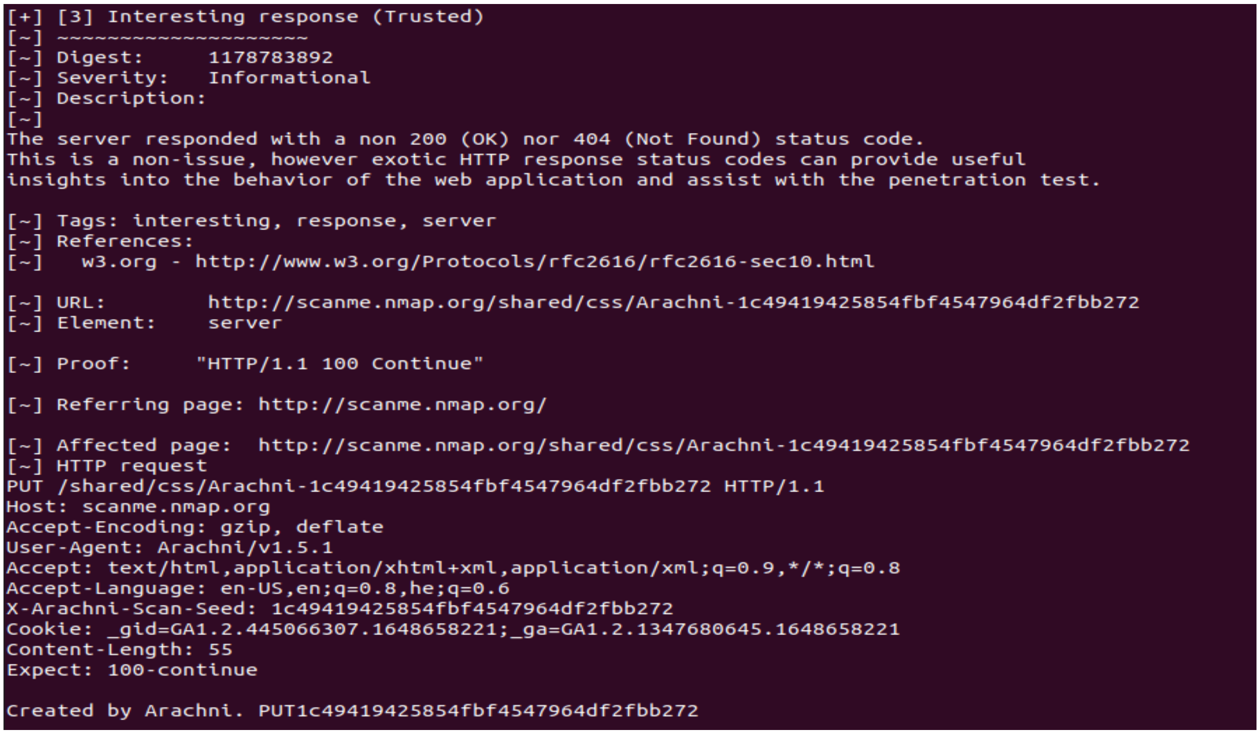
```
[+] [1] Common directory (Trusted)
[~] ~~~~~~~~~~~~~~~~~~~~
[~] Digest:     889905018
[~] Severity:   Medium
[~] Description:
[~]
Web applications are often made up of multiple files and directories.

It is possible that over time some directories may become unreferenced (unused)
by the web application and forgotten about by the administrator/developer.
Because web applications are built using common frameworks, they contain common
directories that can be discovered (independent of server).

During the initial recon stages of an attack, cyber-criminals will attempt to
locate unreferenced directories in the hope that the directory will assist in further
compromise of the web application.
To achieve this they will make thousands of requests using word lists containing
common names.
The response headers from the server will then indicate if the directory exists.

Arachni also contains a list of common directory names which it will attempt to access.

[~] Tags: path, directory, common, discovery

[~] CWE: http://cwe.mitre.org/data/definitions/538.html
[~] References:
[~]   CWE - http://cwe.mitre.org/data/definitions/538.html
[~]   OWASP - https://www.owasp.org/index.php/Forced_browsing

[~] URL:        http://scanme.nmap.org/shared/css/
[~] Element:    server

[~] Proof:     "HTTP/1.1 200 OK"
```

Figure 36. Platform operation - Module 9: Web application security report (part 2).



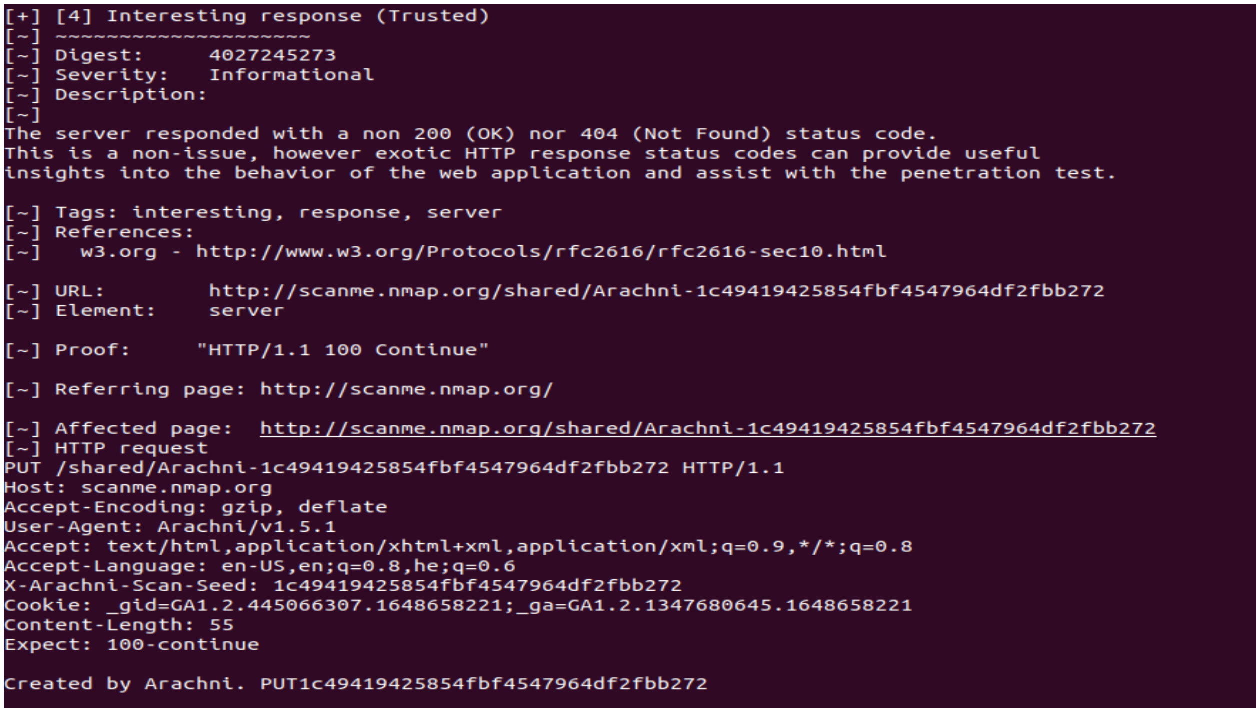
```
[+] [2] Interesting response (Trusted)
[~] ~~~~~~~~~~~~~~~~~~~~
[~] Digest:     3629422460
[~] Severity:   Informational
[~] Description:
[~]
The server responded with a non 200 (OK) nor 404 (Not Found) status code.
This is a non-issue, however exotic HTTP response status codes can provide useful
insights into the behavior of the web application and assist with the penetration test.

[~] Tags: interesting, response, server
[~] References:
[~]   w3.org - http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html

[~] URL:        http://scanme.nmap.org/shared/images/Arachni-1c49419425854fbf4547964df2fbb272
[~] Element:    server

[~] Proof:     "HTTP/1.1 100 Continue"

[~] Referring page: http://scanme.nmap.org/

[~] Affected page:  http://scanme.nmap.org/shared/images/Arachni-1c49419425854fbf4547964df2fbb272
[~] HTTP request
PUT /shared/images/Arachni-1c49419425854fbf4547964df2fbb272 HTTP/1.1
Host: scanme.nmap.org
Accept-Encoding: gzip, deflate
User-Agent: Arachni/v1.5.1
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.8,he;q=0.6
X-Arachni-Scan-Seed: 1c49419425854fbf4547964df2fbb272
Cookie: _gid=GA1.2.445066307.1648658221;_ga=GA1.2.1347680645.1648658221
Content-Length: 55
Expect: 100-continue

Created by Arachni. PUT1c49419425854fbf4547964df2fbb272
```

Figure 37. Platform operation - Module 9: Web application security report (part 3).



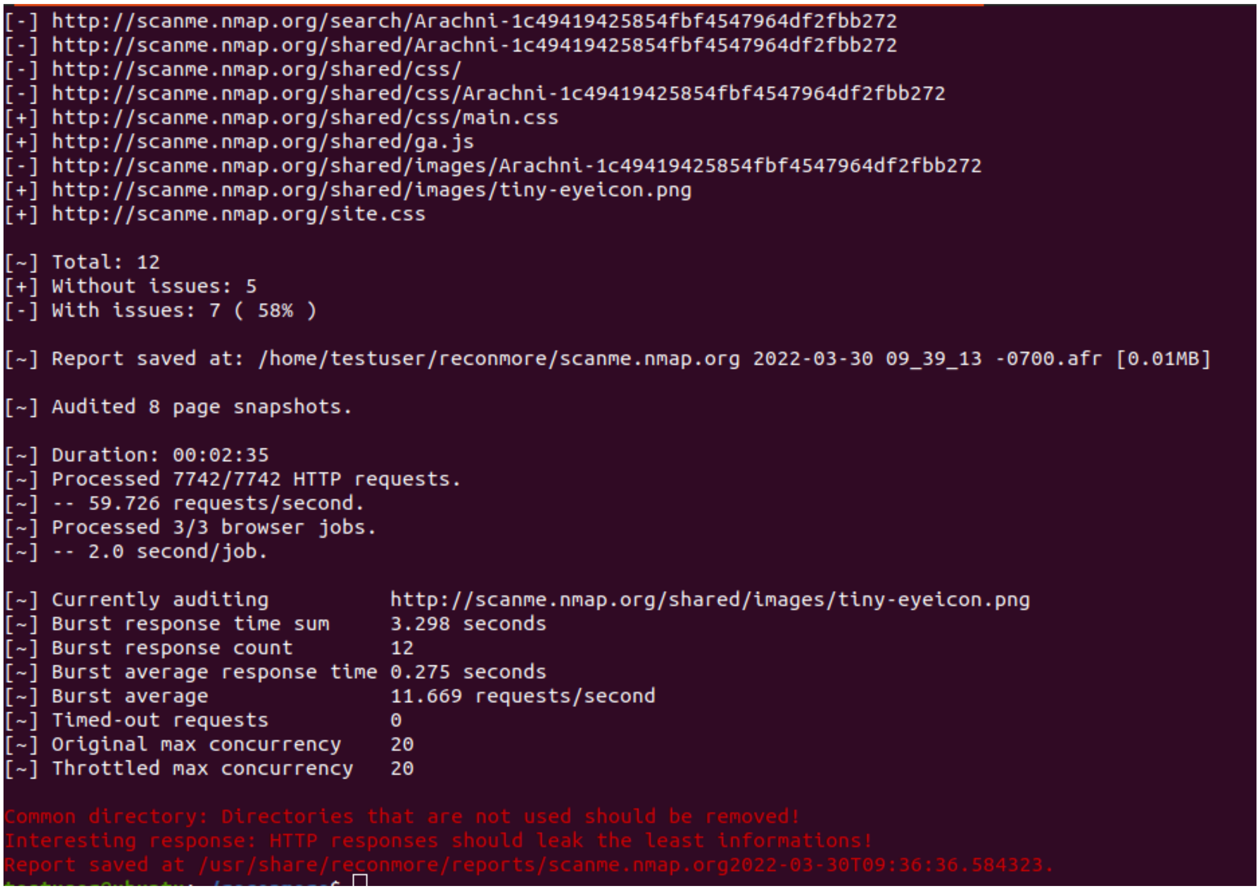
```
[+] [3] Interesting response (Trusted)
[~] ~~~~~~~~~~~~~~~~~~~
[~] Digest:     1178783892
[~] Severity:   Informational
[~] Description:
[~]
The server responded with a non 200 (OK) nor 404 (Not Found) status code.
This is a non-issue, however exotic HTTP response status codes can provide useful
insights into the behavior of the web application and assist with the penetration test.

[~] Tags: interesting, response, server
[~] References:
[~]   w3.org - http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html

[~] URL:        http://scanme.nmap.org/shared/css/Arachni-1c49419425854fbf4547964df2fbb272
[~] Element:    server

[~] Proof:     "HTTP/1.1 100 Continue"

[~] Referring page: http://scanme.nmap.org/

[~] Affected page:  http://scanme.nmap.org/shared/css/Arachni-1c49419425854fbf4547964df2fbb272
[~] HTTP request
PUT /shared/css/Arachni-1c49419425854fbf4547964df2fbb272 HTTP/1.1
Host: scanme.nmap.org
Accept-Encoding: gzip, deflate
User-Agent: Arachni/v1.5.1
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.8,he;q=0.6
X-Arachni-Scan-Seed: 1c49419425854fbf4547964df2fbb272
Cookie: _gid=GA1.2.445066307.1648658221;_ga=GA1.2.1347680645.1648658221
Content-Length: 55
Expect: 100-continue

Created by Arachni. PUT1c49419425854fbf4547964df2fbb272
```

Figure 38. Platform operation - Module 9: Web application security report (part 4).



```
[+] [4] Interesting response (Trusted)
[~] ~~~~~~~~~~~~~~~~~~~
[~] Digest:     4027245273
[~] Severity:   Informational
[~] Description:
[~]
The server responded with a non 200 (OK) nor 404 (Not Found) status code.
This is a non-issue, however exotic HTTP response status codes can provide useful
insights into the behavior of the web application and assist with the penetration test.

[~] Tags: interesting, response, server
[~] References:
[~]   w3.org - http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html

[~] URL:        http://scanme.nmap.org/shared/Arachni-1c49419425854fbf4547964df2fbb272
[~] Element:    server

[~] Proof:     "HTTP/1.1 100 Continue"

[~] Referring page: http://scanme.nmap.org/

[~] Affected page:  http://scanme.nmap.org/shared/Arachni-1c49419425854fbf4547964df2fbb272
[~] HTTP request
PUT /shared/Arachni-1c49419425854fbf4547964df2fbb272 HTTP/1.1
Host: scanme.nmap.org
Accept-Encoding: gzip, deflate
User-Agent: Arachni/v1.5.1
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.8,he;q=0.6
X-Arachni-Scan-Seed: 1c49419425854fbf4547964df2fbb272
Cookie: _gid=GA1.2.445066307.1648658221;_ga=GA1.2.1347680645.1648658221
Content-Length: 55
Expect: 100-continue

Created by Arachni. PUT1c49419425854fbf4547964df2fbb272
```

Figure 39. Platform operation - Module 9: Web application security report (part 5).



```
[-] http://scanme.nmap.org/search/Arachni-1c49419425854fbf4547964df2fbb272
[-] http://scanme.nmap.org/shared/Arachni-1c49419425854fbf4547964df2fbb272
[-] http://scanme.nmap.org/shared/css/
[-] http://scanme.nmap.org/shared/css/Arachni-1c49419425854fbf4547964df2fbb272
[+] http://scanme.nmap.org/shared/css/main.css
[+] http://scanme.nmap.org/shared/ga.js
[-] http://scanme.nmap.org/shared/images/Arachni-1c49419425854fbf4547964df2fbb272
[+] http://scanme.nmap.org/shared/images/tiny-eyeicon.png
[+] http://scanme.nmap.org/site.css

[~] Total: 12
[+] Without issues: 5
[-] With issues: 7 ( 58% )

[~] Report saved at: /home/testuser/reconmore/scanme.nmap.org 2022-03-30 09_39_13 -0700.afr [0.01MB]

[~] Audited 8 page snapshots.

[~] Duration: 00:02:35
[~] Processed 7742/7742 HTTP requests.
[~] -- 59.726 requests/second.
[~] Processed 3/3 browser jobs.
[~] -- 2.0 second/job.

[~] Currently auditing          http://scanme.nmap.org/shared/images/tiny-eyeicon.png
[~] Burst response time sum     3.298 seconds
[~] Burst response count        12
[~] Burst average response time 0.275 seconds
[~] Burst average               11.669 requests/second
[~] Timed-out requests          0
[~] Original max concurrency    20
[~] Throttled max concurrency   20

Common directory: Directories that are not used should be removed!
Interesting response: HTTP responses should leak the least informations!
Report saved at /usr/share/reconmore/reports/scanme.nmap.org2022-03-30T09:36:36.584323.
```

Figure 40. Platform operation - Module 9: Web application security report (part 6).

REFERENCES

1. National Institute of Standards and Technology, Attack surface definition, retrieved from https://csrc.nist.gov/glossary/term/attack_surface
2. OWASP CheatSheets Series Team, Attack Surface Analysis Cheat Sheet, retrieved from https://cheatsheetseries.owasp.org/cheatsheets/Attack_Surface_Analysis_Cheat_Sheet.html
3. Fale Association of Locksport Enthusiasts, Open Source Intelligence Framework, retrieved from https://osintframework.com/
4. MITRE ATT&CK Knowledge base, Reconnaissance tactics and techniques, retrieved from https://attack.mitre.org/
5. German agency Federal Office for Information Security (BSI), Study a penetration testing model, retrieved from https://www.bsi.bund.de/SharedDocs/Downloads/EN/BSI/Publications/Studies/Penetration/penetration_pdf.html
6. Vignesh Chandrasekaran, Let's recon webinar slides, retrieved from https://owasp.org/www-chapter-coimbatore/
7. Python Software Foundation, Python 3.8.13 documentation, retrieved from https://docs.python.org/3.8/
8. Marco d'Itri, Whois manual page, (2009, December 20) retrieved from https://manpages.debian.org/stretch/whois/whois.1.en.html

9. Internet Systems Consortium, Dig manual page, retrieved from https://manpages.debian.org/bullseye/bind9-dnsutils/dig.1.en.html
10. Merit Network Inc, The RADB whois server, retrieved from https://www.radb.net/query/help
11. The Shadowserver Foundation, ASN queries, retrieved from https://www.shadowserver.org/what-we-do/network-reporting/api-asn-and-network-queries/
12. Carlos Perez, DNSRecon, retrieved from https://github.com/darkoperator/dnsrecon
13. OJ Reeves, Gobuster, retrieved from https://github.com/OJ/gobuster
14. Wikipedia, Domain hijacking, (2021, November 25) retrieved from https://en.wikipedia.org/wiki/Domain_hijacking
15. MDN web docs, Subdomain takeovers, (2022, February 18), retrieved from https://developer.mozilla.org/en-US/docs/Web/Security/Subdomain_takeovers
16. Cloudflare, What is recursive DNS?, retrieved from https://www.cloudflare.com/learning/dns/what-is-recursive-dns/
17. Wikipedia, DNS zone transfer, (2021, November 20), retrieved from https://en.wikipedia.org/wiki/DNS_zone_transfer
18. Duckduckgo search engine, retrieved from https://html.duckduckgo.com/html/
19. Phil Harvey, ExifTool, retrieved from https://github.com/exiftool/exiftool
20. Free OpenPGP keyserver, retrieved from https://keyserver.ubuntu.com/
21. MDN web docs, HTTP Headers - Security, retrieved from https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers#security

22. Open Web Application Security Project, OWASP Secure Headers Project - Response Headers, retrieved from https://owasp.org/www-project-secure-headers
23. Andrew Horton, WhatWeb, retrieved from https://github.com/urbanadventurer/WhatWeb
24. Daniel Stenberg, Curl manual page, (2016, December 16) retrieved from https://manpages.debian.org/stretch/curl/curl.1.en.html
25. Jordan Potti, CloudScraper, retrieved from https://github.com/jordanpotti/CloudScraper
26. Open Web Application Security Project, Top 10:2021 List - A03 Injection, retrieved from https://owasp.org/Top10/A03_2021-Injection/
27. Wikipedia, SQL injection, (2022, February 25), retrieved from https://en.wikipedia.org/wiki/SQL_injection
28. Bernardo Damele A. G., Miroslav Stampar, SQLmap, retrieved from https://github.com/sqlmapproject/sqlmap
29. Gordon Lyon, Nmap, retrieved from https://nmap.org/
30. OWASP CheatSheets Series Team, SQL Injection Prevention Cheat Sheet, retrieved from https://cheatsheetseries.owasp.org/cheatsheets/SQL_Injection_Prevention_Cheat_Sheet.html
31. Aviv Ron, Alexandra Shulman-Peleg, Emanuel Bronshtein, (2015), No SQL, No Injection? Examining NoSQL Security, retrieved from https://www.researchgate.net/publication/278332363_No_SQL_No_Injection_Examining_NoSQL_Security
32. Enable Security, WAFW00F, retrieved from https://github.com/EnableSecurity/wafw00f

33. Gordon Lyon, Nmap Book Chapter 4 - What are the most popular ports?, retrieved from https://nmap.org/book/port-scanning.html#most-popular-ports

34. Carlos Polop, HackTricks, retrieved from https://book.hacktricks.xyz/

35. InfosecMatter, Nmap NSE library, (2021, April 7) retrieved from https://www.infosecmatter.com/nmap-nse-library/

36. Speed Guide Inc, Ports Database, retrieved from https://www.speedguide.net/ports.php

37. Gibson Research Corporation, GRC Port Authority - Interactive Internet Database, retrieved from https://www.grc.com/portdatahelp.htm

38. Fastmail, SSl, TLS and STARTTLS, retrieved from https://www.fastmail.help/hc/en-us/articles/360058753834

39. UC Berkeley - Information Security Office, Securing Remote Desktop (RDP) for System Administrators, retrieved from https://security.berkeley.edu/education-awareness/securing-remote-desktop-rdp-system-administrators

40. Microsoft Documentation, Direct Host SMB over TCP/IP, (2021, July 12) retrieved from https://docs.microsoft.com/en-US/troubleshoot/windows-server/networking/direct-hosting-of-smb-over-tcpip

41. Microsoft Documentation, How to detect, enable and disable SMBv1, SMBv2 and SMBv3 in Windows, (2021, May 11) retrieved from https://docs.microsoft.com/en-US/windows-server/storage/file-server/troubleshoot/detect-enable-and-disable-smbv1-v2-v3

42. Jeff Petters, What is an SMB Port + Ports 445 and 139 Explained, (2021, May 7), retrieved from https://www.varonis.com/blog/smb-port

43. Microsoft Documentation, How RPC Works, (2009, August 10), retrieved from https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2003/cc738291(v=ws.10)?redirectedfrom=MSDN

44. U.S. Department of Defence - National Security Agency, Configuring IPsec Virtual Private Networks, retrieved from https://media.defense.gov/2021/Sep/16/2002855928/-1/-1/0/CONFIGURING_IPSEC_VIRTUAL_PRIVATE_NETWORKS_2020_07_01_FINAL_RELEASE.PDF

45. Cybersecurity and Infrastructure Security Agency, Reducing the Risk of SNMP Abuse, (2017, June 5), retrieved from https://www.cisa.gov/uscert/ncas/alerts/TA17-156A

46. PUC-Rio, LUA 5.4 Reference Manual, retrieved from https://www.lua.org/manual/5.4/

47. Docmeta LLC, Arachni framework checks, retrieved from https://www.rdoc.info/github/Arachni/arachni/Arachni/Checks

48. Ecsypno, Arachni - Web Application Security Scanner Framework, retrieved from https://github.com/Arachni/arachni

49. PortSwigger, Cross-Site Request Forgery (CSRF), retrieved from https://portswigger.net/web-security/csrf

50. Open Web Application Security Project, Cross Site Request Forgery (CSRF), retrieved from https://owasp.org/www-community/attacks/csrf

51. Wikipedia, Cross site request forgery, (2022, March 19), retrieved from https://en.wikipedia.org/wiki/Cross-site_request_forgery
52. Open Web Application Security Project, Code Injection, retrieved from https://owasp.org/www-community/attacks/Code_Injection
53. PortSwigger, Unidentified code injection, retrieved from https://portswigger.net/kb/issues/00101000_unidentified-code-injection
54. Open Web Application Security Project, LDAP injection, retrieved from https://owasp.org/www-community/attacks/LDAP_Injection
55. OWASP CheatSheets Series Team, LDAP Injection Prevention Cheat Sheet, retrieved from https://cheatsheetseries.owasp.org/cheatsheets/LDAP_Injection_Prevention_Cheat_Sheet.html
56. Open Web Application Security Project, Path Traversal, retrieved from https://owasp.org/www-community/attacks/Path_Traversal
57. Open Web Application Security Project, HTTP Response Splitting, retrieved from https://owasp.org/www-community/attacks/HTTP_Response_Splitting
58. Open Web Application Security Project, Command Injection, retrieved from https://owasp.org/www-community/attacks/Command_Injection
59. PortSwigger, OS command injection, retrieved from https://portswigger.net/web-security/os-command-injection
60. OWASP CheatSheets Series Team, Unvalidated Redirects and Forwards Cheat Sheet, retrieved from

https://cheatsheetseries.owasp.org/cheatsheets/Unvalidated_Redirects_and_Forwards_Cheat_Sheet.html

61. PortSwigger, XPath injection, retrieved from https://portswigger.net/kb/issues/00100600_xpath-injection
62. Open Web Application Security Project, Cross Site Scripting (XSS), retrieved from https://owasp.org/www-community/attacks/xss/
63. PortSwigger, Cross-site scripting, retrieved from https://portswigger.net/web-security/cross-site-scripting
64. ENISA, Cross-site scripting (XSS), retrieved from https://www.enisa.europa.eu/topics/csirts-in-europe/glossary/cross-site-scripting-xss
65. PortSwigger, Source code disclosure, retrieved from https://portswigger.net/kb/issues/006000b0_source-code-disclosure
66. PortSwigger, XML external entity (XXE) injection, retrieved from https://portswigger.net/web-security/xxe
67. PortSwigger, HTTP PUT method is enabled, retrieved from https://portswigger.net/kb/issues/00100900_http-put-method-is-enabled
68. MDN Web Docs, Using HTTP cookies, (2022, January 20) retrieved from https://developer.mozilla.org/en-US/docs/Web/HTTP/Cookies
69. PortSwigger, File upload vulnerabilities, retrieved from, https://portswigger.net/web-security/file-upload

70. PortSwigger, Cookie scoped to parent domain, retrieved from https://portswigger.net/kb/issues/00500300_cookie-scoped-to-parent-domain