CS 455 – Computer Security Fundamentals

Dr. Chen-Yeou (Charles) Yu

System and Networks Security

- **Web application vulnerability**
  - The reason why we choose web applications is because it is the one most likely get hacked. It is commonly in these ways:
    - Replacements of web pages
    - Denial of service (DoS). The web server is active but end users cannot login to browse web pages
  - In the section, you will know how to analyze the vulnerability of a website by using some tools to scan it.
    - Nikto
    - Burp Suite (TBD, in the next time)
      - Web server pen. Testing
      - Very largely used by the bug bounty community

Web application vulnerability

- There are different kinds of vulnerability scanners, one is called web vulnerability scanners. However, generally speaking, there are some limitations.
    - It might gave us false-positive reports.
        - A real hacker will use different tools keep scanning what he/she wants.
    - Web vulnerability scanners cannot identify complex errors in business logic.
        - It can only detect if the software status is normal.

Web application vulnerability

- Here are the different kind of scanners
  - Scanners that extend the functionality of traditional vulnerability scanners to include websites and associated services (for example, the Metasploit framework and Websploit)
  - Scanners that extend the functionality of non-traditional applications, such as web browsers, to support web service vulnerability scanning (OWASP Mantra)
  - Scanners that are specifically developed to support reconnaissance and exploit detection in websites and web services (Arachni, Nikto, Skipfish, WPScan, joomscan, and so on)

Web application vulnerability

- Nikto
    - Nikto is one of the most utilized active web application scanners. It performs comprehensive tests against web servers
    - Nikto is a Perl-based open-source scanner
        - It used to be very popular before.
    - But It is beginning to show its age and is not as accurate as some of the more modern scanners.
    - Most (old-school) penetration testers start testing a website by using Nikto.
    - Yes! The purpose for Nikto is to do the vulnerability scanning! And most of the hackers like to use this specifically for web server / web applications vulnerabilities

Web application vulnerability

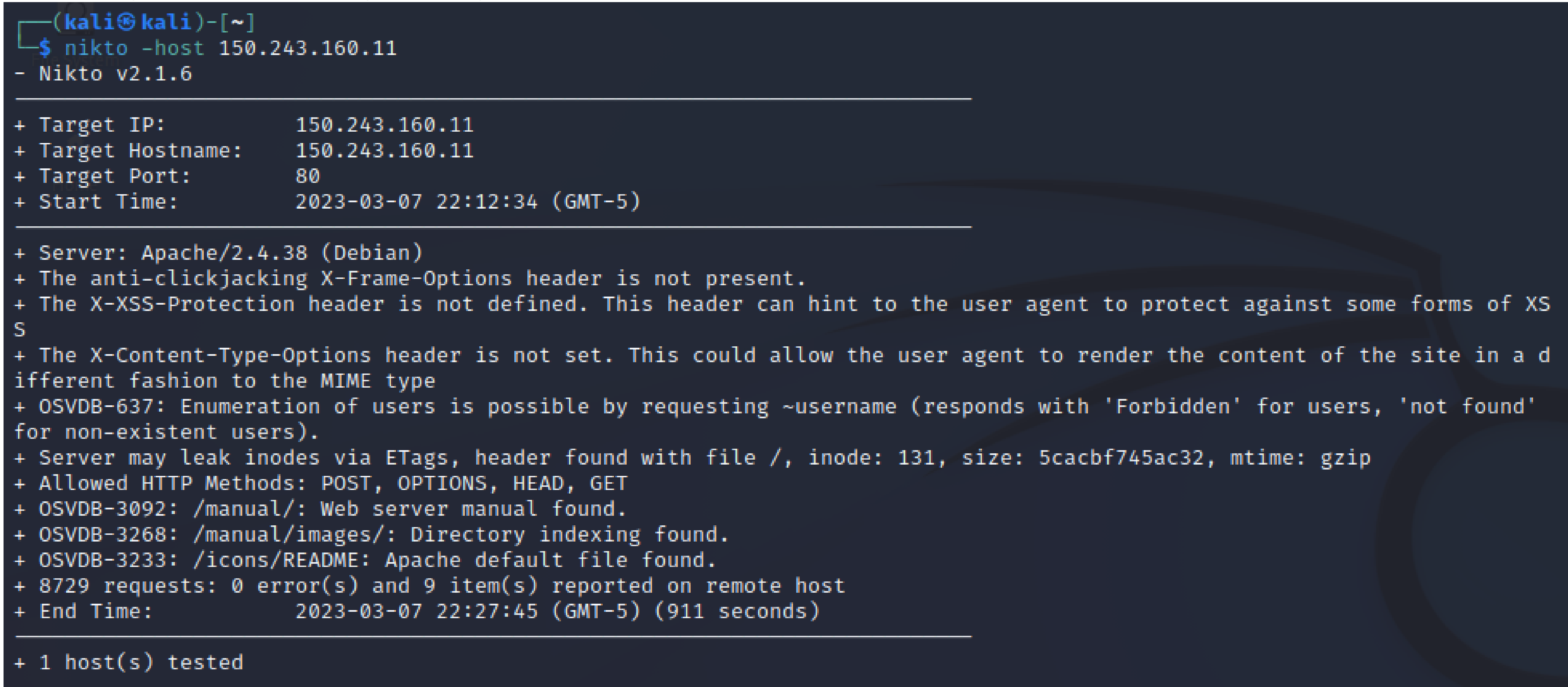
- nikto -H
  - To print out the “help” menu
- nikto -host
  - Try our “sand.truman.edu”
  - It has an interface facing to the internet: 150.243.160.11
  - It might take a while (3~5 minutes)
  - If we do not specify anything, the scan will be its web server (apache) by default. The port is 80.

Web application vulnerability

- A couple of info. Is collected.
    - Enumeration of user’s home folder is possible
    - IT supports several HTTP methods: POST, OPTIONS, HEAD, GET
    - A couple of Apache default manual or files are found
    - (Check the next page for detail)

Web application vulnerability

- For example, if we quickly scan this IP “150.243.160.11”



```
┌──(kali㉿kali)-[~]
└─$ nikto -host 150.243.160.11
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          150.243.160.11
+ Target Hostname:    150.243.160.11
+ Target Port:        80
+ Start Time:         2023-03-07 22:12:34 (GMT-5)
---------------------------------------------------------------------------
+ Server: Apache/2.4.38 (Debian)
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XS
S
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a d
ifferent fashion to the MIME type
+ OSVDB-637: Enumeration of users is possible by requesting ~username (responds with 'Forbidden' for users, 'not found'
for non-existent users).
+ Server may leak inodes via ETags, header found with file /, inode: 131, size: 5cacbf745ac32, mtime: gzip
+ Allowed HTTP Methods: POST, OPTIONS, HEAD, GET
+ OSVDB-3092: /manual/: Web server manual found.
+ OSVDB-3268: /manual/images/: Directory indexing found.
+ OSVDB-3233: /icons/README: Apache default file found.
+ 8729 requests: 0 error(s) and 9 item(s) reported on remote host
+ End Time:           2023-03-07 22:27:45 (GMT-5) (911 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
```

Web application vulnerability

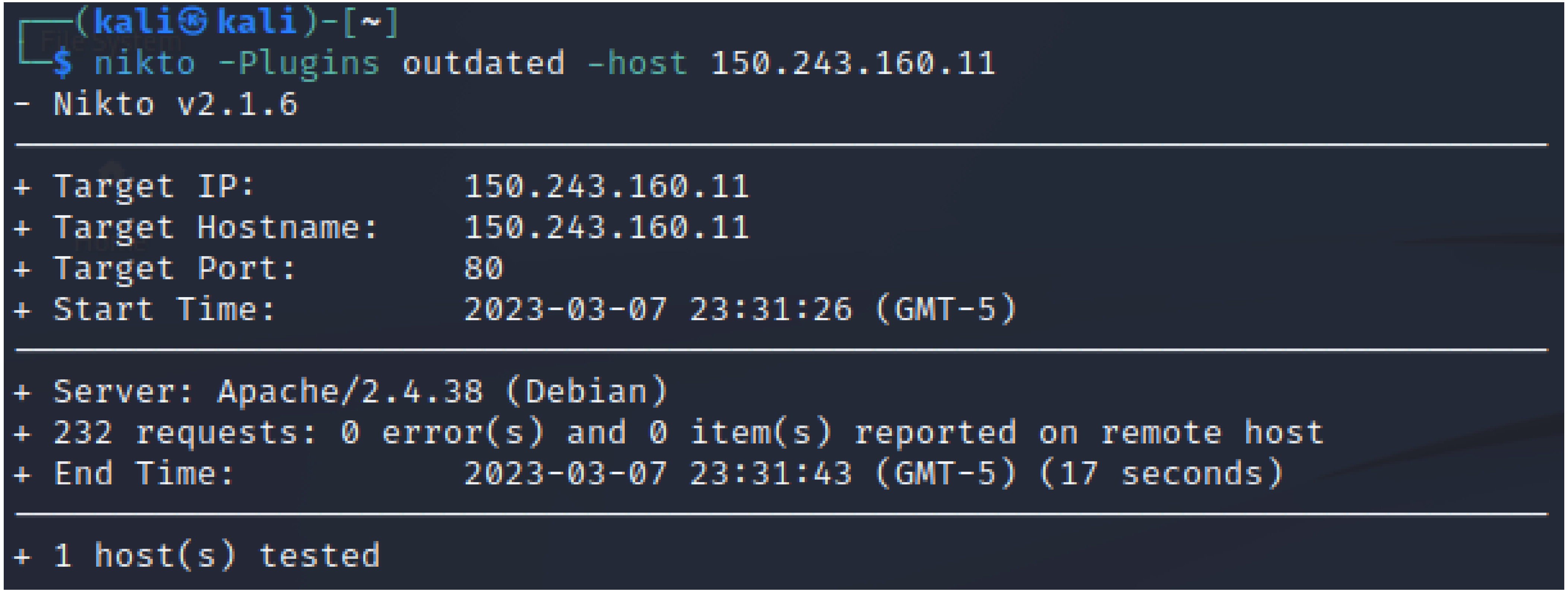
- Can we drill down to some of more specific scans? Yes!
  - We need plug-ins for nikto
  - nikto already has some preloaded plug-ins
  - nikto -list-plugins
    - To list all of its plug-ins

Web application vulnerability

- If we scanned everything, for example, the scans in the page #8, it would be very slow.
  - We can make it little bit faster if we can specify some modules or only one module --- a more customized scan. For example, the “outdated” software component or library
  - nikto -Plugins outdated -host 150.243.160.11

Web application vulnerability

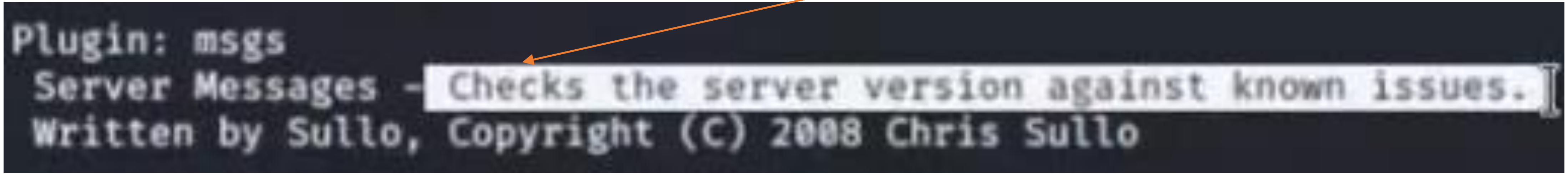
- We got nothing, since the admin is doing his job very well



```
┌──(kali㉿kali)-[~]
└─$ nikto -Plugins outdated -host 150.243.160.11
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          150.243.160.11
+ Target Hostname:    150.243.160.11
+ Target Port:        80
+ Start Time:         2023-03-07 23:31:26 (GMT-5)
---------------------------------------------------------------------------
+ Server: Apache/2.4.38 (Debian)
+ 232 requests: 0 error(s) and 0 item(s) reported on remote host
+ End Time:           2023-03-07 23:31:43 (GMT-5) (17 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
```

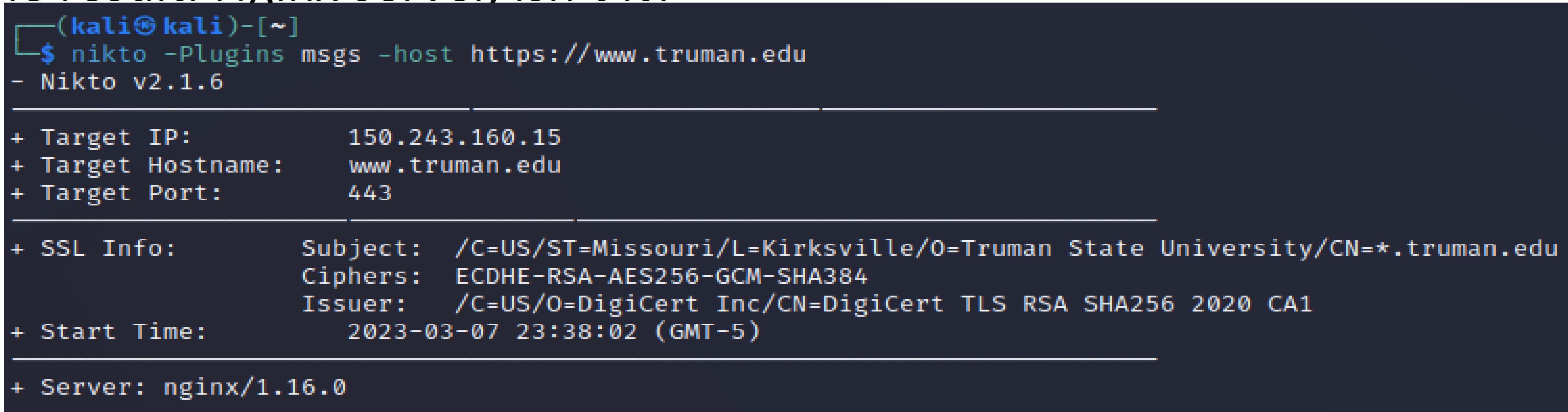
Web application vulnerability

- Let’s try the msgs module. This is the purpose of the module



```
Plugin: msgs
 Server Messages - Checks the server version against known issues.
 Written by Sullo, Copyright (C) 2008 Chris Sullo
```

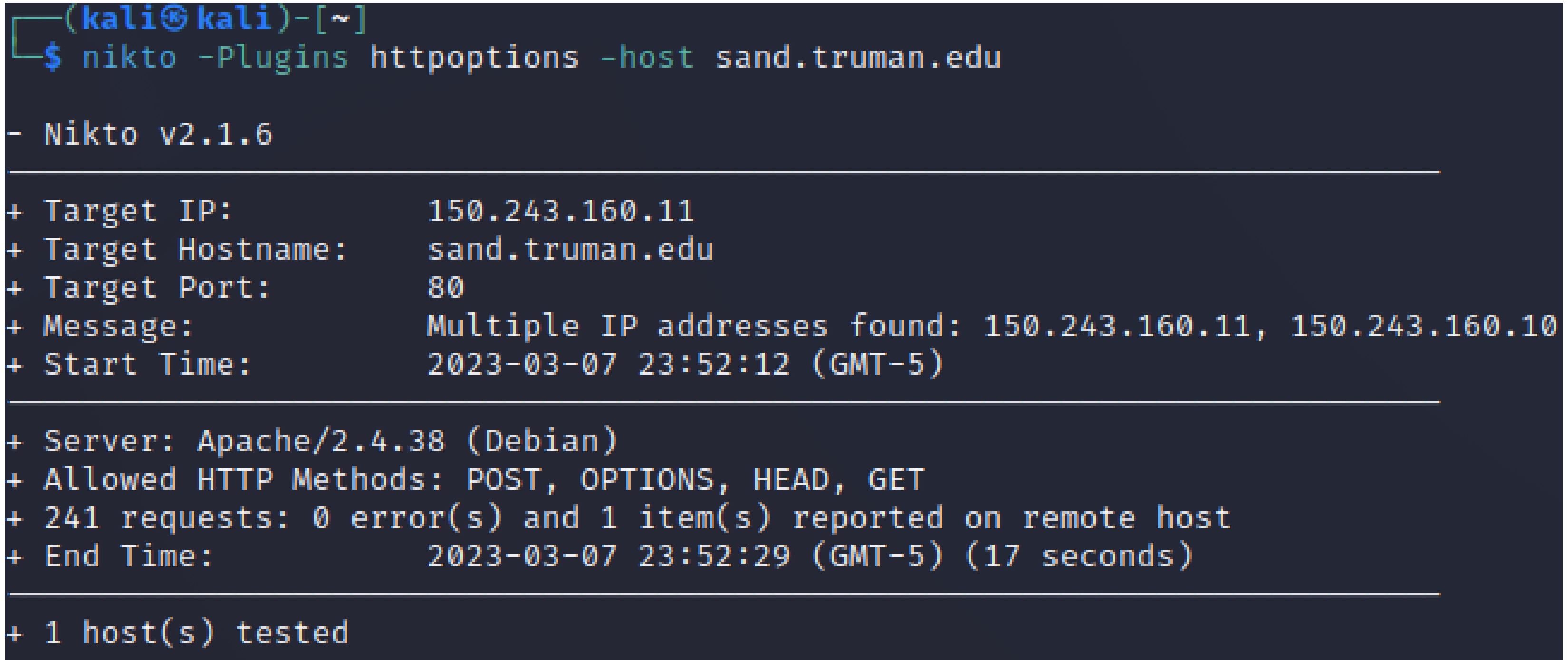
This is the result. Nginx server, isn’t it?



```
┌──(kali㉿kali)-[~]
└─$ nikto -Plugins msgs -host https://www.truman.edu
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          150.243.160.15
+ Target Hostname:    www.truman.edu
+ Target Port:        443
---------------------------------------------------------------------------
+ SSL Info:        Subject:  /C=US/ST=Missouri/L=Kirksville/O=Truman State University/CN=*.truman.edu
                   Ciphers:  ECDHE-RSA-AES256-GCM-SHA384
                   Issuer:   /C=US/O=DigiCert Inc/CN=DigiCert TLS RSA SHA256 2020 CA1
+ Start Time:         2023-03-07 23:38:02 (GMT-5)
---------------------------------------------------------------------------
+ Server: nginx/1.16.0
```

Web application vulnerability

- Let’s try “http options”
  - nikto -Plugins httpoptions -host http://sand.truman.edu
  - This is saying what kind of http methods does this server allow?
  - Everything looks good. The server admin is doing good job.

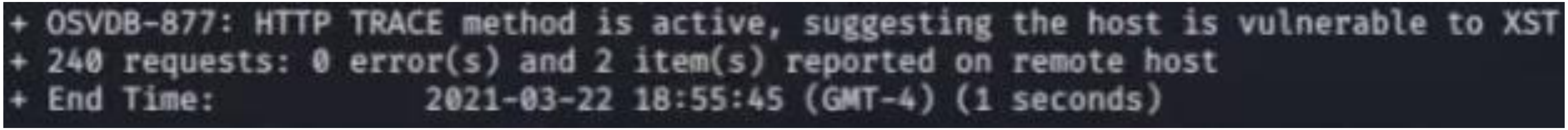


```
┌──(kali㉿kali)-[~]
└─$ nikto -Plugins httpoptions -host sand.truman.edu

- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          150.243.160.11
+ Target Hostname:    sand.truman.edu
+ Target Port:        80
+ Message:            Multiple IP addresses found: 150.243.160.11, 150.243.160.10
+ Start Time:         2023-03-07 23:52:12 (GMT-5)
---------------------------------------------------------------------------
+ Server: Apache/2.4.38 (Debian)
+ Allowed HTTP Methods: POST, OPTIONS, HEAD, GET
+ 241 requests: 0 error(s) and 1 item(s) reported on remote host
+ End Time:           2023-03-07 23:52:29 (GMT-5) (17 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
```

Web application vulnerability

- Sometimes, you might see this in some other web servers, in the scans of “httpoptions”

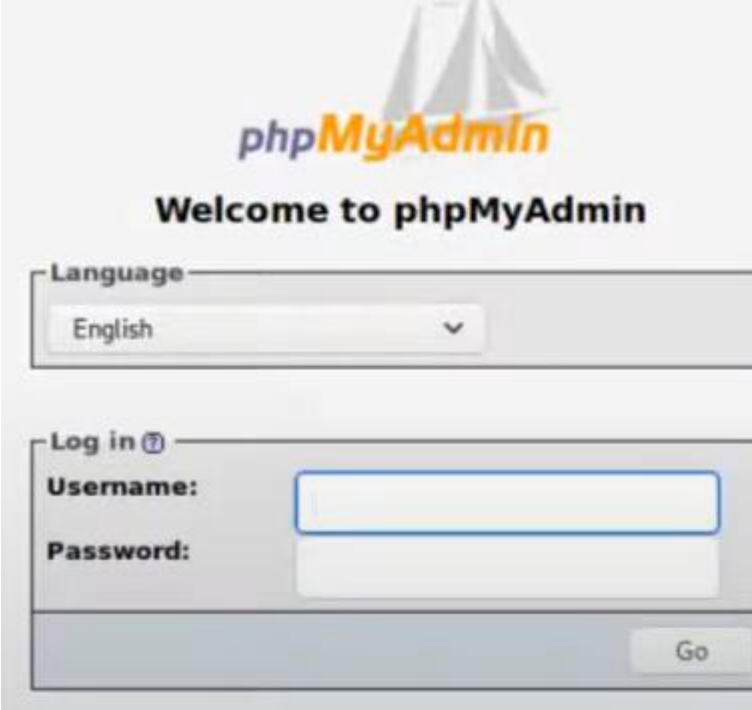


```
+ OSVDB-877: HTTP TRACE method is active, suggesting the host is vulnerable to XST
+ 240 requests: 0 error(s) and 2 item(s) reported on remote host
+ End Time:           2021-03-22 18:55:45 (GMT-4) (1 seconds)
```

- This is saying the host is vulnerable to “xst”
- xst stands for “cross site tracing”.
- We can use the trace method to retrieve http cookies, or potentially headers.
- This is something definitely to be disabled

Web application vulnerability

- One more thing, the address line is case sensitive. For example,
  - http://vh216602.truman.edu/agarvey/ ← Dr. Alan’s website
  - If you put “Agarvey”, it would be totally different for web servers
- A couple of hints in the “scanned result” from the web server (I won’t tell you how to hack someone’s server ^_^)
  - If you find “phpmyadmin” , you need to smile
  (for brute force hacks)
  - If you find a “browsable folder”, you need to smile
  (someone just forget to turn it off)



Welcome to phpMyAdmin

Language

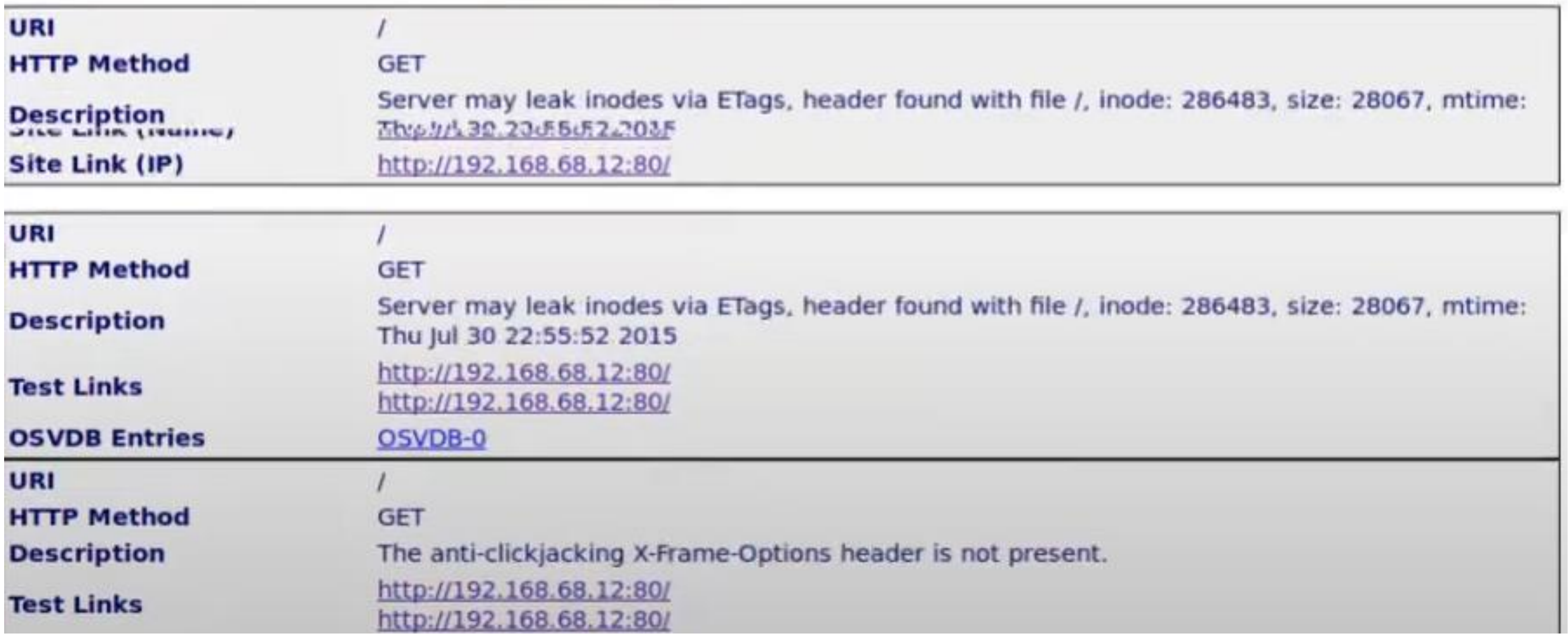
English

Web application vulnerability

- We can specify an output to a report, for example, report.html?
  - nikto -host xxx.xxx.xxx.xxx -output report.html
  - Basically, if you run this, it will re-run the full scan (because we do not specify the module (plugin), and send the result to the “report.html”

Web application vulnerability

- The report will be much human-readable (open the html file)



URI	/
HTTP Method	GET
Description	Server may leak inodes via ETags, header found with file /, inode: 286483, size: 28067, mtime: Thu Jul 30 22:55:52 2015
Site Link (Name)	
Site Link (IP)	http://192.168.68.12:80/

URI	/
HTTP Method	GET
Description	Server may leak inodes via ETags, header found with file /, inode: 286483, size: 28067, mtime: Thu Jul 30 22:55:52 2015
Test Links	http://192.168.68.12:80/ http://192.168.68.12:80/
OSVDB Entries	OSVDB-0

URI	/
HTTP Method	GET
Description	The anti-clickjacking X-Frame-Options header is not present.
Test Links	http://192.168.68.12:80/ http://192.168.68.12:80/