Default Password Scanner (default-http-login-hunter.sh)

infosecmatter.com/default-password-scanner-default-http-login-hunter-sh

April 7, 2020

Looking for default passwords during penetration tests is not always fun. It's actually hard work, especially when we have a large environment. How can we check all those different web interfaces? Is there a viable automation?

This article introduces a new and completely free scanner of default password logins (default-http-login-hunter) which can be useful for penetration testers, security auditors, but also for sysadmins and network administrators.

Introduction

Checking administrative interfaces for weak and default credentials is a vital part of every VAPT exercise. But doing it manually can quickly become exhausting.

The problem with web interfaces is that they are all different. And so to develop an universal automation that could do the job across multiple interfaces is very hard.

Although there are some solutions for this, they are mostly commercial and the functionality is not even that great.

Luckily there is a free and open source solution that can help us.

NNdefaccts alternate dataset

The NNdefaccts dataset made by nnposter is an alternate fingerprint dataset for the Nmap http-default-accounts.nse script.

The NNdefacts dataset can test more than 380 different web interfaces for default logins. For comparison, the latest Nmap 7.80 default dataset only supports 55.

Here are some examples of the supported devices and their web interfaces:

- Network devices (3Com, Asus, Cisco, D-Link, F5, Nortel..)
- Video cameras (AXIS, GeoVision, Hikvision, Sanyo..)
- Application servers (Apache Tomcat, JBoss EAP..)
- Monitoring software (Cacti, Nagios, OpenNMS..)
- Server management (Dell iDRAC, HP iLO..)
- Web servers (WebLogic, WebSphere..)
- Printers (Kyocera, Sharp, Xerox..)
- IP Phones (Cisco, Polycom..)
- Citrix, NAS4Free, ManageEngine, VMware..

See the following link for a full list of supported devices:
https://github.com/InfosecMatter/http-default-logins/blob/master/list.txt

The usage is quite simple – we simply run the Nmap script with the alternate dataset as a parameter. Like this:

```
nmap --script http-default-accounts --script-args http-default-
accounts.fingerprintfile=~/http-default-accounts-fingerprints-nndefaccts.lua -p 80
192.168.1.1
```

This is already pretty great as it is.

Nmap script limitations

Now the only caveat with this solution is that the http-default-accounts.nse script works only for web servers running on common web ports such as tcp/80, tcp/443 or similar.

This is because the script contains the following port rule which matches only common web ports:

```
license = "Same as Nmap--See https://nmap.org/book/man-legal.html"
categories = {"discovery", "auth", "intrusive"}

portrule = shortport.http

---
--validate_fingerprints(fingerprints)
--Returns an error string if there is something wrong with
```

So what if we find a web server running on a different port – say tcp/9999? Unfortunately the Nmap script will not run because of the port rule..

..unless we modify the port rule in the Nmap script to match our web server port! And that's exactly where our new tool comes handy.

Introducing default-http-login-hunter

The default-http-login-hunter tool, written in Bash, is essentially a wrapper around the aforementioned technologies to unlock their full potential and to make things easy for us.

The tool simply takes a URL as an argument:

```
default-http-login-hunter.sh <URL>
```

First it will make a local temporary copy of the http-default-accounts.nse script and it will modify the port rule so that it will match the web server port that we provided in the URL.

Then it will run the Nmap command for us and display the output nicely. Here's an example:

```
root@kali:~# ./default-http-login-hunter.sh 192.168.204.245:9999
Mon 05 Apr 2020 12:11:10 AM +04: trying default http logins on http://192.168.204.245:9999/
| 192.168.204.245.9999.http.45114354:
|   [Apache Tomcat Manager] at /manager/html/
|     admin:tomcat
root@kali:~#
```

From the above screenshot we can see that we found a default credentials for Apache Tomcat running on port tcp/9999. Now we could deploy a webshell on it and obtain RCE. But that's another story..

Additional features

List of URLs

The tool also accepts a list of URLs in a file. So, for instance, we could feed it with URLs found during Nessus scans extracted using our Nessus CSV parser.

The tool will go through all the URLs one by one and check for default logins. Like this:

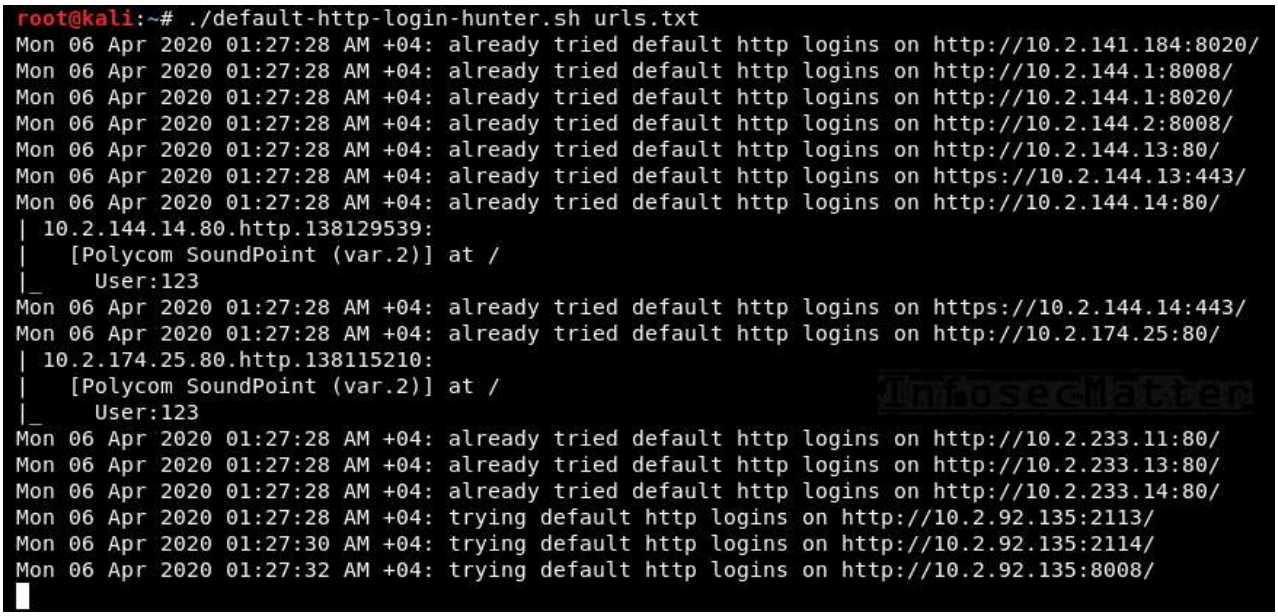
```
default-http-login-hunter.sh urls.txt
```

```
root@kali:~# ./default-http-login-hunter.sh urls.txt
Mon 06 Apr 2020 00:41:11 AM +04: trying default http logins on http://10.0.126.50:80/
Mon 06 Apr 2020 00:41:11 AM +04: trying default http logins on http://10.0.126.50:8008/
Mon 06 Apr 2020 00:41:12 AM +04: trying default http logins on http://10.0.126.50:8020/
Mon 06 Apr 2020 00:41:12 AM +04: trying default http logins on http://10.0.126.55:8008/
Mon 06 Apr 2020 00:41:12 AM +04: trying default http logins on http://10.0.126.55:8020/
Mon 06 Apr 2020 00:41:13 AM +04: trying default http logins on http://10.0.126.56:8008/
Mon 06 Apr 2020 00:41:13 AM +04: trying default http logins on http://10.0.126.56:8020/
Mon 06 Apr 2020 00:41:13 AM +04: trying default http logins on http://10.0.126.57:8008/
Mon 06 Apr 2020 00:41:14 AM +04: trying default http logins on http://10.0.126.57:8020/
Mon 06 Apr 2020 00:41:14 AM +04: trying default http logins on http://10.0.126.57:23479/
Mon 06 Apr 2020 00:41:15 AM +04: trying default http logins on http://10.0.126.63:8008/
Mon 06 Apr 2020 00:41:15 AM +04: trying default http logins on http://10.0.126.63:8020/
Mon 06 Apr 2020 00:41:16 AM +04: trying default http logins on http://10.0.126.68:80/
Mon 06 Apr 2020 00:41:16 AM +04: trying default http logins on http://10.0.126.68:8008/
Mon 06 Apr 2020 00:41:17 AM +04: trying default http logins on http://10.0.126.68:8020/
Mon 06 Apr 2020 00:41:17 AM +04: trying default http logins on https://10.0.126.69:9443/
| 10.0.126.69.9443.https.6480913382:
|   [Cisco IronPort] at /
|_    admin:ironport
Mon 06 Apr 2020 00:41:18 AM +04: trying default http logins on http://10.0.126.69:8008/
Mon 06 Apr 2020 00:41:18 AM +04: trying default http logins on http://10.0.126.69:8020/
```

Here the tool found a default login to the Cisco IronPort running on port https/9443.

Resume-friendly

Another useful feature is that it saves all the results in the current working directory. So if it gets accidentally interrupted, it will just continue where it stopped. Like in this example:



In this case we have found some Polycom IP phones logins.

Staying up-to-date

To make sure that we have the latest NNdefacts dataset, just run the update command:

```
default-http-login-hunter.sh update
```

And that's pretty much it. If you want to see more detailed output, use -v parameter in the command line.

You can find the tool in our InfosecMatter Github repository here.

Fingerprint contribution

I encourage you to check out the NNdefacts project and consider contributing with fingerprints that you find during your engagements.

Contribution is not hard – you can simply record the login procedure in the Fiddler, Burp or ZAP and send the session file to the author. Please see more information on the fingerprint contribution here.

You may find these links useful while hunting for default logins manually:

- https://cirt.net/passwords
- https://www.routerpasswords.com/

Conclusion

This tool can be of a great help not only while performing internal infrastructure penetration tests, but everywhere where we need to test a web interface for default credentials. Its simple design and smart features make it also very easy to use. Hope you will find it useful too!

If you like our tools and you would like more, please do subscribe to our mailing list and follow us on Twitter, Facebook or Github to get notified about new additions!

Thanks

Lastly, I want to thank nnposter for his awesome NNdefacts dataset without which this would not be possible and also for his contributions to the Nmap project. Thank younnposter!

SHARE THIS

TAGS | Credentials | HTTP | HTTPS | Login attack | Scanner | Scripting | Tool