

Security Hardening Proof of Concept (POC)

5 FEB 2022

[Overview 1](#_Toc1434719971)

[Introduction 1](#_Toc1119722555)

[Scope 1](#_Toc290189584)

[Network design 1](#_Toc802857651)

[Services 1](#_Toc1226518026)

[Setting Up Secure FTP 1](#_Toc109411254)

[Security 1](#_Toc1190907343)

[Features/Measures 1](#_Toc892286926)

[Setting GRUB Password 1](#_Toc180313293)

[Disable SSH Root Login 1](#_Toc868419551)

[SSH Brute Force Protection 1](#_Toc1690940499)

[Tightening Security For Services 1](#_Toc2026114850)

[Closing Unused Open Ports 1](#_Toc615093480)

[Remove banner grabbing for website 1](#_Toc1644986883)

[Vulnerability 1](#_Toc931758738)

[SUID binary – bash 1](#_Toc1339612434)

[Exploitation 1](#_Toc1821752024)

[Mitigation 1](#_Toc1270555319)

[World-readable /etc/shadow 1](#_Toc1802696876)

[Exploitation 1](#_Toc278171217)

[Mitigation 1](#_Toc1819700019)

[No password for switch user (su) 1](#_Toc1939689086)

[Exploitation 1](#_Toc1198953791)

[Mitigation 1](#_Toc706268490)

[Bad Practices 1](#_Toc1312247374)

[No password sudo 1](#_Toc1374204589)

[No functionality user 1](#_Toc1550254812)

[Conclusion 1](#_Toc2010864504)

[Agenda of Demo 1](#_Toc1681368545)

# Introduction

## Overview

This report is for the documentation of the conclusive proof of concept of the web application. FTPS was added to the POC system to enable authorized ABC staff to upload/update the price list content specifically through remote. There were also 6 security features configured to harden the system, and 3 vulnerabilities and 2 bad practices were found and patched. A potentially better design has also been added to the end as a proposal.

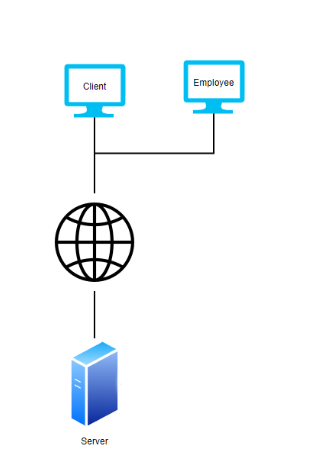
## Scope

* Enable the public to browse through a price list of the products that ACB is selling.
* Enable authorized ABC staff to upload/update price list content remotely.
* Design and implement security-related features/measures to enhance system security.
* Identify and apply appropriate fixes to potential security-related vulnerabilities.

## List of Procedure

* Find vulnerabilities & misconfigurations
* Patch the vulnerabilities & misconfigurations
* Implement FTPS to allow staff to upload/update price list
* Harden FTPS security by limiting permission to files
* Adding physical security by password locking BIOS and enabling GRUB
* Disable SSH root login
* Enable Fail2Ban to prevent SSH brute force
* Closing RPC port
* Remove python banner on website

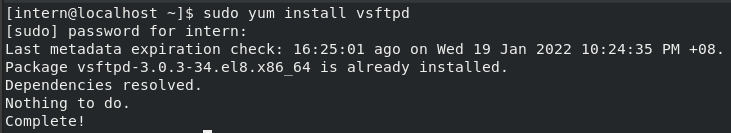
## Network design



# Services

## Setting Up Secure FTP

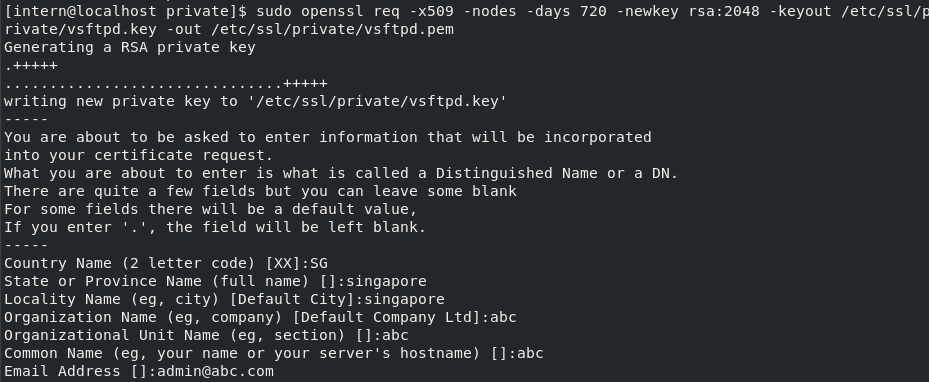
Ensure that vsftpd is installed:



To make vsftpd more secure we highly recommended using SSL from (<https://letsencrypt.org/>) or any other certificate provider however, the following will be using a self-signed certificate.

Generating self-signed certificate:

|  |
| --- |
| sudo mkdir /etc/ssl/private sudo openssl req -x509 -nodes -days 720 -newkey rsa:2048 -keyout /etc/ssl/private/vsftpd.key -out /etc/ssl/private/vsftpd.pem |



Add the following to the end of /etc/vsftpd/vsftpd.conf to configure vsftpd to use SSL:

|  |
| --- |
| # https://access.redhat.com/solutions/3436 ssl\_enable=YES    # To allow anonymous users to use SSL  allow\_anon\_ssl=YES    # To force anonymous users to use SSL  force\_anon\_data\_ssl=YES  force\_anon\_logins\_ssl=YES    # To force local users to use SSL  force\_local\_data\_ssl=YES  force\_local\_logins\_ssl=YES    # The following option depend of the authentication mode you require  # for TLS Version 1  ssl\_tlsv1=YES  # for SSL Version 2  ssl\_sslv2=YES  # for SSL Version 3  ssl\_sslv3=YES    # This values must be adjust according with you environment  rsa\_cert\_file=/etc/ssl/private/vsftpd.pem  rsa\_private\_key\_file=/etc/ssl/private/vsftpd.key |

Configure the following to prevent banner grabbing & allowing only a specific user to login by adding the following to the end of /etc/vsftpd/vsftpd.conf

|  |
| --- |
| ftpd\_banner=Ftp Server  chroot\_local\_user=YES  userlist\_deny=NO |

Run the following command to create a user with access to the pricelist.csv file:

|  |
| --- |
| sudo useradd –m –d /var/www/pricelist/data ftpwebadmin  sudo passwd ftpwebadmin  sudo chmod 777 /var/www/pricelist/data/pricelist.csv |

Comment out all the users in /etc/vsftpd/user\_list and add the newly created ftp user:

|  |
| --- |
| # vsftpd userlist  # If userlist\_deny=NO, only allow users in this file  # If userlist\_deny=YES (default), never allow users in this file, and  # do not even prompt for a password.  # Note that the default vsftpd pam config also checks /etc/vsftpd/ftpusers  # for users that are denied.  #root  #bin  #daemon  #adm  #lp  #sync  #shutdown  #halt  #mail  #news  #uucp  #operator  #games  #nobody  ftpwebadmin |

Allow full access to vsftpd

|  |
| --- |
| sudo setsebool -P allow\_ftpd\_full\_access 1 |

Restart vsftpd:

|  |
| --- |
| sudo systemctl restart vsftpd  sudo systemctl enable vsftpd |

# Security

## Features/Measures

### Setting GRUB Password

Setting a GRUB password is an effective way to secure a system physically with the addition of having the BIOS password locked. It prevents unauthorized users from accessing a root account by password protecting the grub boot loader so that they are unable to access the single user mode and grub console.

Set grub password in root account



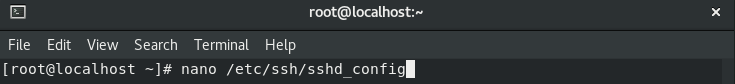
Enter desired grub password



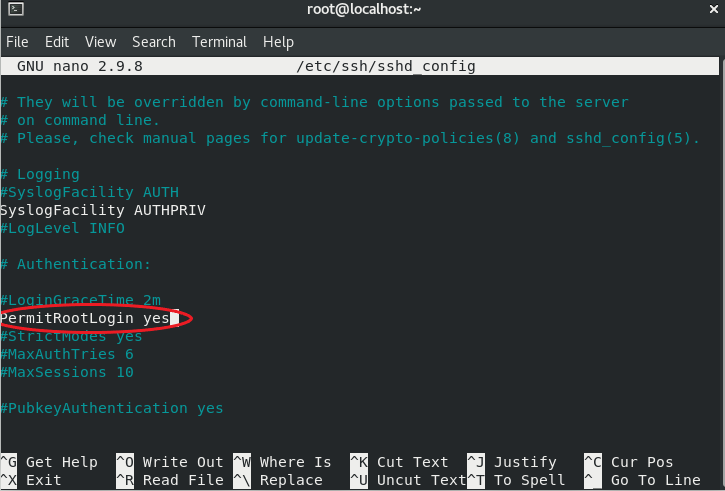
### Disable SSH Root Login

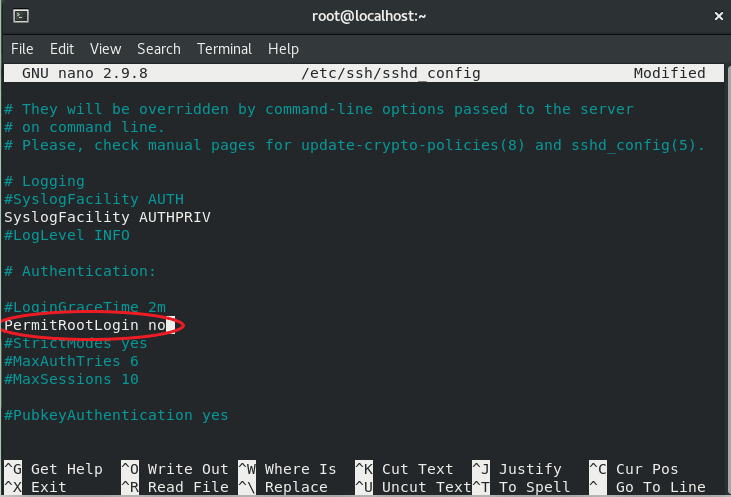
The root account holds a lot of access and permissions to a system; hence threat actors often try to gain remote access to these root accounts. Thus, SSH root login should be disabled.

Configure sshd\_config file



Change “PermitRootLogin yes” to “PermitRootLogin no”

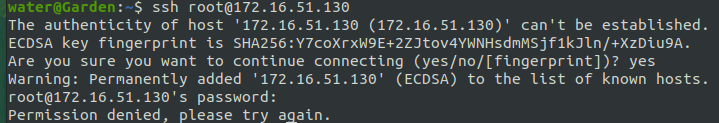


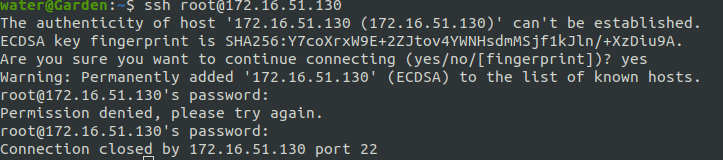


Restart the sshd service



You should no longer be able to login to root account through SSH



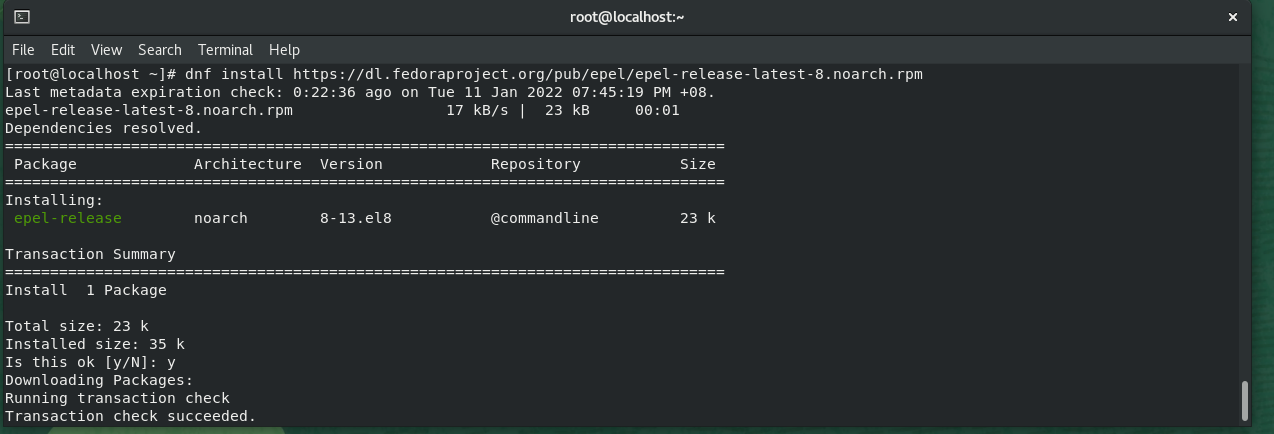


### SSH Brute Force Protection

By using Fail2Ban we can configure login security measures such as the amount of login failure on an account before it is temporarily locked, this would make brute force an unrealistic approach for threat actors as it would take a lot of time to guess the password

Install EPEL repository

https://dl.fedoraproject.org/pub/epel/epel-next-release-latest-8.noarch.rpm



Install fail2ban



Copy jail.conf file from the fail2ban directory to jail.local file

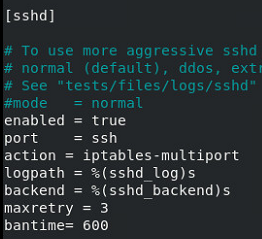
cp /etc/fail2ban/jail.conf /etc/fail2ban/jail.local



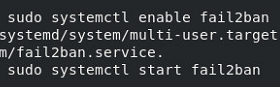
Edit the protection configuration setting through the jail.local file



You can configure multiple settings shown in the following picture:



Enable and start fail2ban:

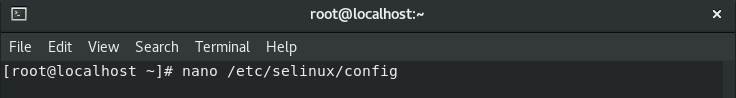


### Tightening Security For Services

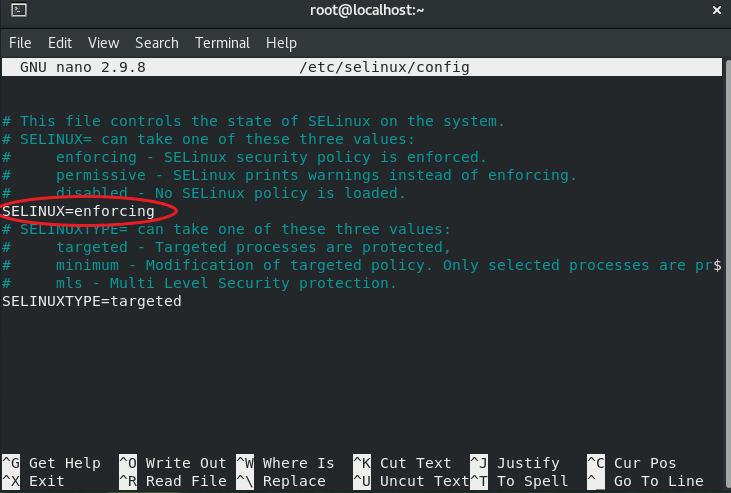
Ensure that selinux is set to enforcing

Security-enhanced Linux (SELinux) is a security module that allows administrators to control services that allows access to a system, SELinux has 3 modes; Enforcing, Permissive, Disabled. It is recommended to ensure that Enforcing mode is enabled, this is because Enforcing mode will allow access based on the policy rules of SELinux

Configure /etc/selinux/config



Enter “enforcing” at the end of “SELINUX=” line, if in the event you need to change selinux to another mode enter the mode name at the end of the “SELINUX=” line

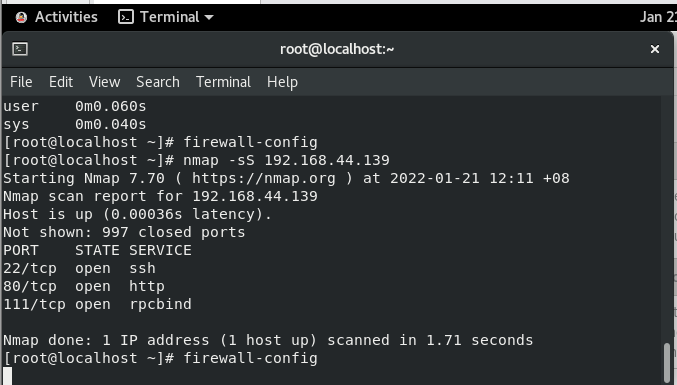


Using ACL to allow ftp user to only access single file to update pricelist of web server

|  |
| --- |
| sudo setfacl -m u:ftpwebadmin:000 /var/www/pricelist/data/\*  sudo setfacl -m user::rw-,group::rw-,other::rw-,user:ftpwebadmin:rw- pricelist.csv |

### Closing Unused Open Ports

Leaving unused ports open poses a security risk, as attackers can exploit these open ports to gain access to the system. As shown in the picture below, after running Nmap there are 3 open ports, however port 111 /service rpcbind is not used as the system is a stand-alone machine, hence it should be closed.

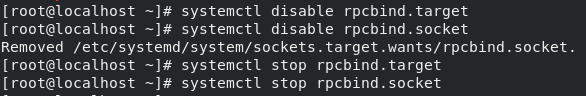


Enter the following commands to stop rpcbind and close the ports

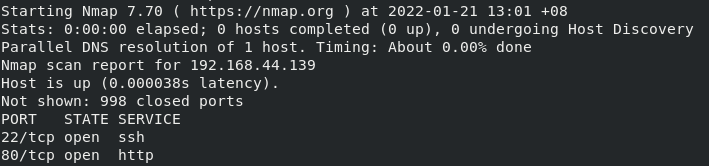
|  |
| --- |
| systemctl stop rpcbind.service  systemctl disable rpcbind.service  systemctl stop rpcbind.target  systemctl disable rpcbind.target  systemctl stop rpcbind.socket  systemctl disable rpcbind. socket |



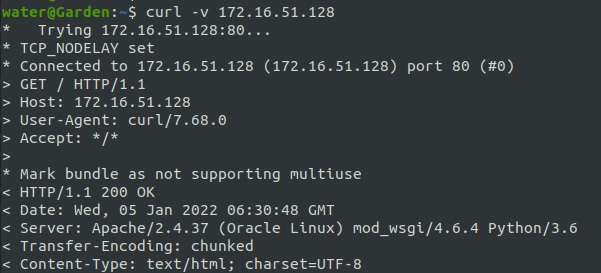


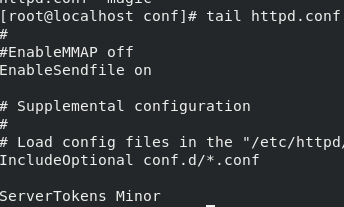


After running Nmap, port 111 /service rpcbind is now closed



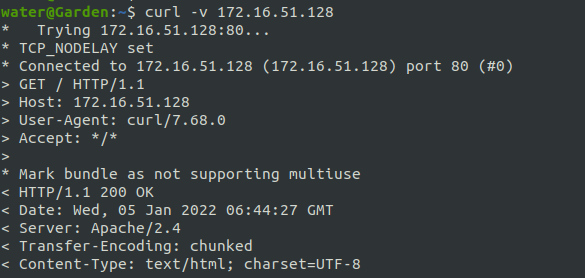
### Remove banner grabbing for website



Add ServerTokens Minor to your /etc/httpd/conf/httpd.conf :  


Restart httpd service:

|  |
| --- |
| service httpd restart |



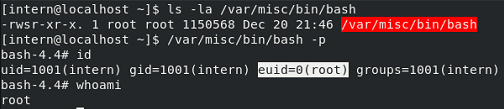
# Vulnerability

## SUID binary – bash

### Exploitation

Finding /var/misc/bin/bash as a world executable suid binary allowing anyone to run it and privilege escalate to root.

Using –p spawn a bash shell with the effective user id of root which allows anyone to execute commands with root access.



### Mitigation

To mitigate this potential vulnerability, it is recommended to delete the binary as it is bad practice to have a shell SUID binary as owned by root. However, if the binary is necessary, it is recommended to change the permissions to (4750) and add the group which needs the binary.



## World-readable /etc/shadow

### Exploitation

Finding /etc/shadow is world readable; this allows everyone to be able to read the hashed password of every user on the system.

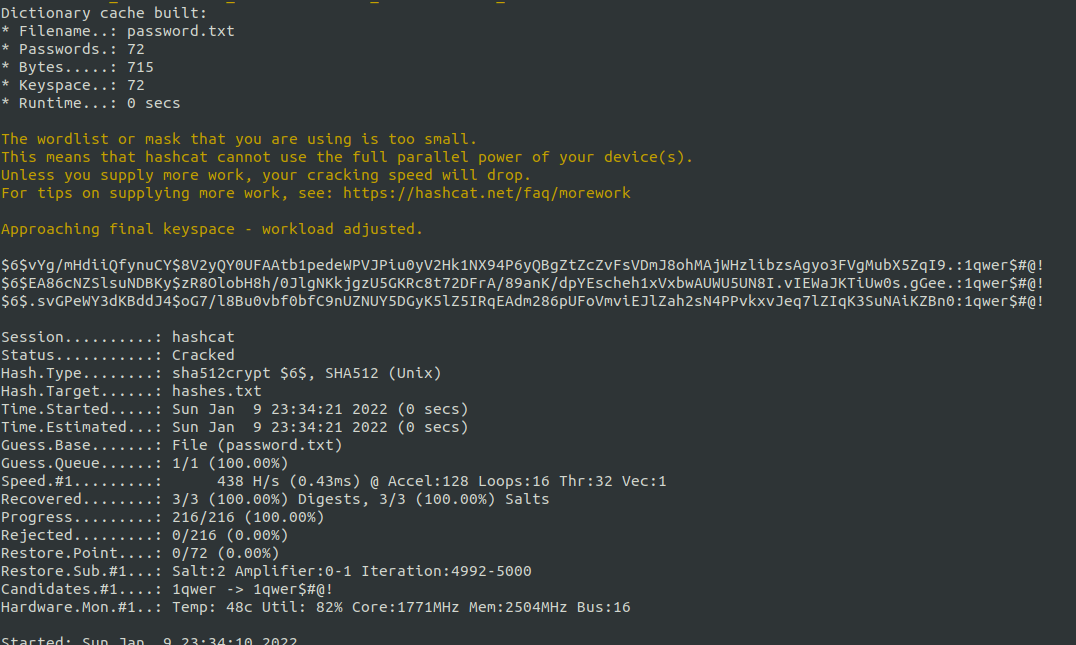
Attackers may use the hashed password by cracking the hash to obtain the password.





Password is 1qwer$#@!





### Mitigation

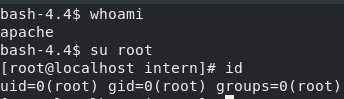
To fix this all that needs to be done is to change the permission back to the default which is 540:



## No password for switch user (su)

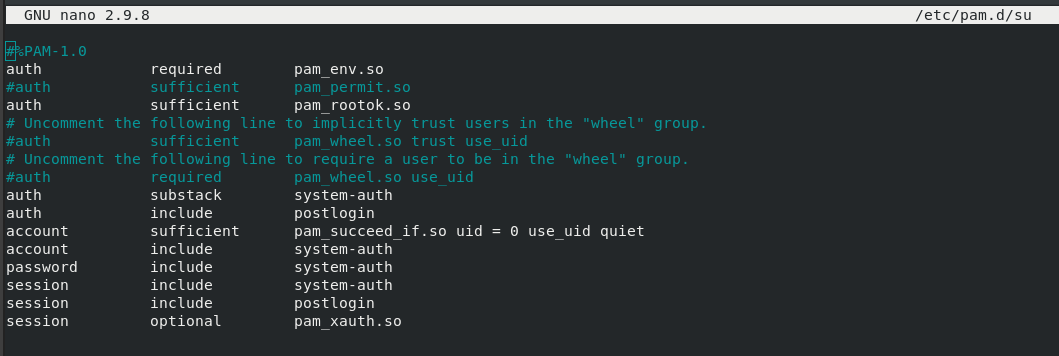
### Exploitation

Any user that can run the “su” command is able to switch to any user without having to authenticate with a password



### Mitigation

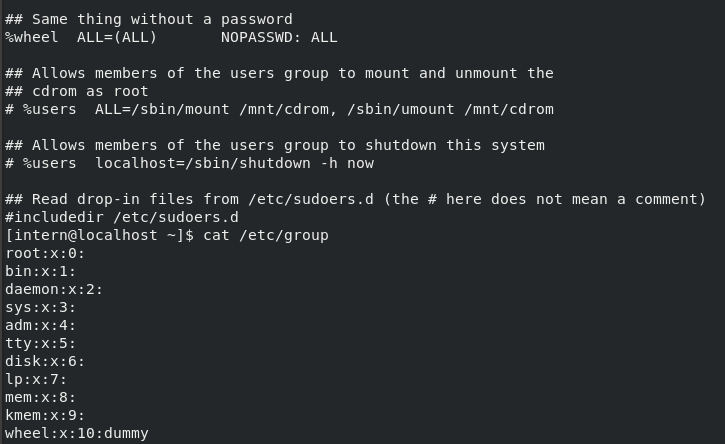
To fix this misconfiguration, the /etc/pam.d/su file will need to be edited, on the third line which contains “pam\_permit.so” should be commented out. This will re-enable the need for a password to switch to another user.



# Bad Practices

## No password sudo

No password sudo for wheel group was configured on the system and the user dummy was in said wheel group. It is generally good practice to not enable no password for sudo and when necessary, it should be for a service account with a specific binary with measures taken to prevent privilege escalation



## No functionality user

The user dummy has no function in the system, it is recommended to just delete the user and its home directory /var/misc as the user does not have any use.

# Conclusion

## Agenda of Demo

We would like to demonstrate our changes made to the POC system to ensure that everything is correctly set up and why it is a necessity for the production system.

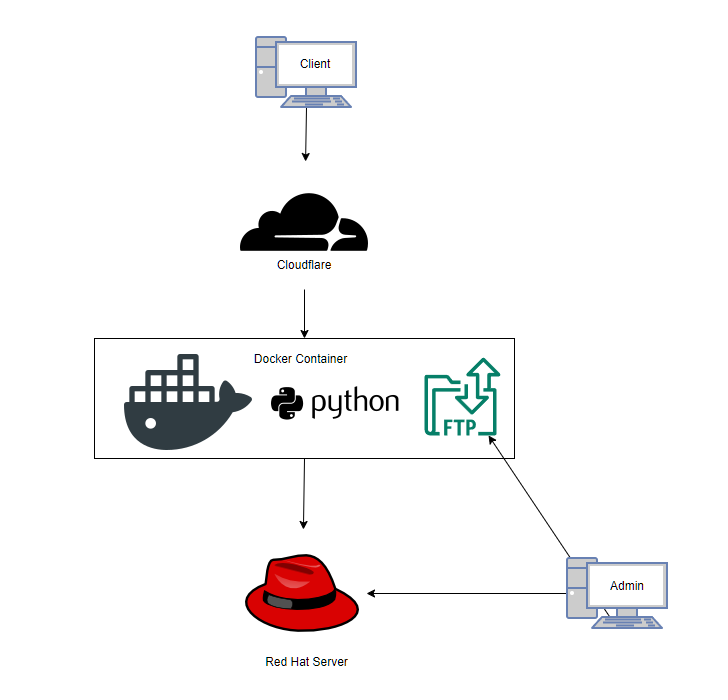
* Demonstration of FTPS
* The need of SSL certificate from a CA
* Using FileZilla client to connect to FTPS
* Showing that the FTP user can only write to pricelist.csv
* Demonstration of Fail2Ban
* Show the potential of Fail2Ban and why we need it
* Fail2Ban can be used for SSH, FTP and HTTP
* Demonstration of enumerating the system
* NMAP & website no longer gives information that can potential be used to identify vulnerabilities

6 examples were given to secure the system, however there are plenty more things to do to fully secure the system, the following link: <https://security.utexas.edu/os-hardening-checklist/linux-7> can be used to ensure that most of the security measures are taken. They are not mandatory, as it highly depends on the application itself.

Another security feature would be implementing docker to isolate the application itself, this will add another security layer if the application has an RCE vulnerability, the attacker must escape the docker to do actual damage to the system.

To protect clients, a WAF can be used to block unsophisticated attacks. Using Cloudflare will do the trick.

Ideally a best set up would look something like this:



# References

security.utexas.edu. (2015). *Red Hat Enterprise Linux 7 Hardening Checklist*. [online] Available at: <https://security.utexas.edu/os-hardening-checklist/linux-7>.

khess (n.d.). *Linux security: Protect your systems with fail2ban*. [online] Enable Sysadmin. Available at: <https://www.redhat.com/sysadmin/protect-systems-fail2ban> [Accessed 5 Feb. 2022].

Red Hat Customer Portal. (n.d.). *How to configure vsftpd with SSL/TLS on Red Hat Enterprise Linux ?* [online] Available at: <https://access.redhat.com/solutions/3436> [Accessed 5 Feb. 2022].

gtfobins.github.io. (n.d.). *GTFOBins*. [online] Available at: <https://gtfobins.github.io/>.

‌

‌