

# Linux Project: Creating Functional Mail Servers in a Virtual Sandbox

DOCUMENTATION

STEVEN FONSECA

# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

## Contents

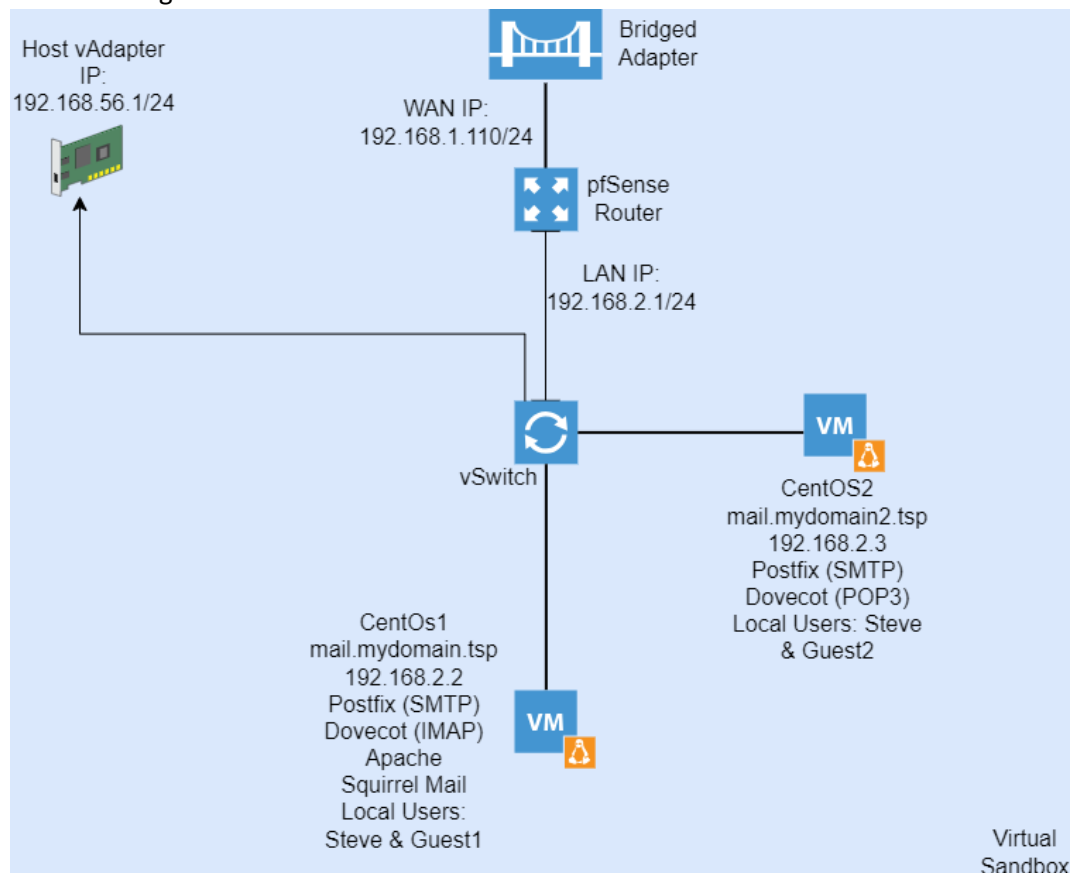
Create the Environment: .....	2
Disable DHCP: .....	3
Install CentOS and pfSense: .....	3
Log into CentOS: .....	4
Enabling Network Connectivity on CentOS: .....	7
Updating and Installing Required Packages: .....	12
Updating pfSense: .....	14
Disabling the Firewall in CentOS: .....	14
Configuring pfSense router: .....	15
Disable SELINUX on CentOS: .....	15
Configure MX records for your router: .....	16
Verifying Network Internet Access: .....	18
Questions: .....	18
Installing Postfix: .....	19
Configuring POSTFIX: .....	20
Creating a new user in CentOS: .....	23
Create a Mail Directory for a user: .....	24
Create a Maildir for a user: .....	24
Verify the Mail/Maildir Structures: .....	24
Enabling Postfix on runlevels 2,3 and 5: .....	25
Using Telnet: .....	26
Configuring POP/IMAP Server: .....	29
Install Dovecot: .....	29
Configuring Dovecot: .....	30
Configure the location of the Mailbox: .....	30
Configure service settings for Dovecot: .....	31
Configure authentication mechanisms for connecting clients: .....	31
Enable Dovecot to start its services: .....	32
Testing Connectivity using Telnet: .....	34
Command-Line Testing: .....	34

# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

Testing Dovecot for POP3: .....	35
Testing Dovecot for POP3 using MUA (Mail User Agent): .....	36
Testing Dovecot for IMAP: .....	40
Testing Dovecot for IMAP using MUA (Mail User Agent): .....	41
Configuring Webmail: .....	46
Installing SquirrelMail: .....	46
Configuring SquirrelMail: .....	46
Installing Apache: .....	47
Configuring Apache: .....	49
References: .....	53
Using the checklist, click the box next to each completed task: .....	54

## Create the Environment:

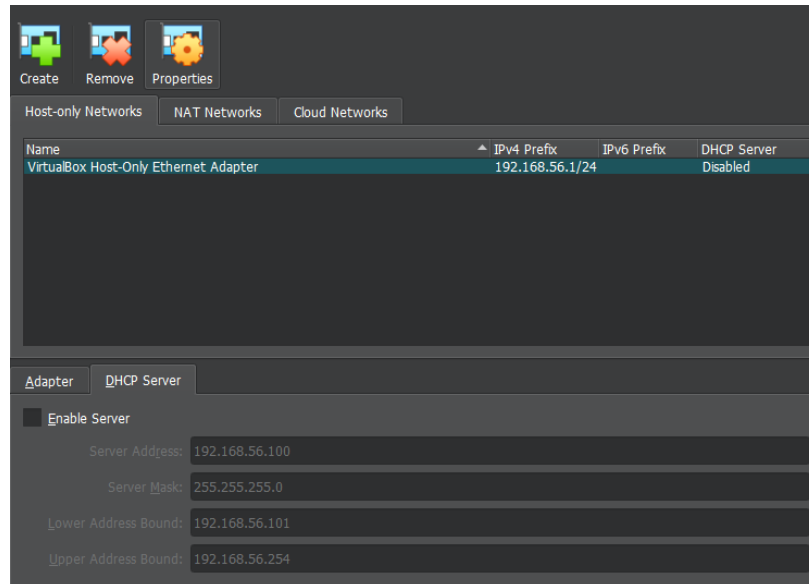
### - Network Diagram:



# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

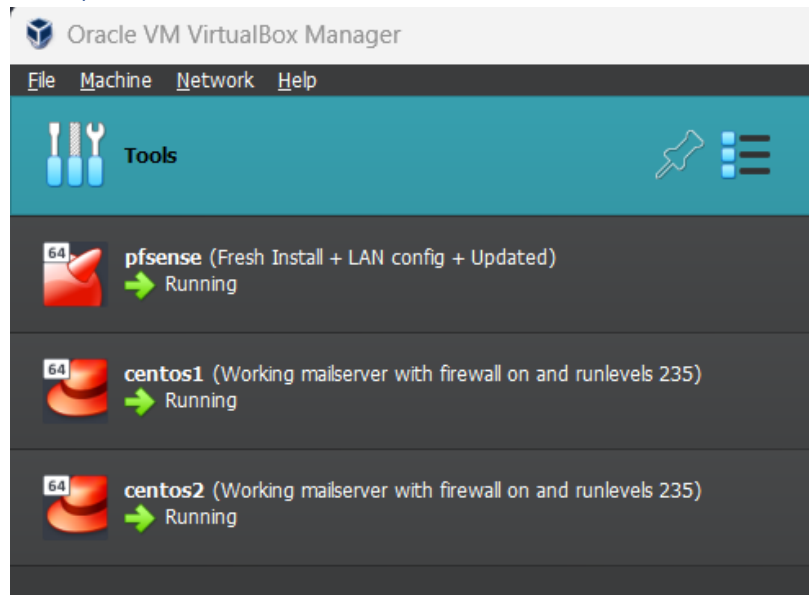
- Install all necessary software: **Virtual Box**, **pfSense**, **CentOS**
  - pfSense 2.7.1 - CentOS 7 - Virtual Box 7

Disable DHCP:



- We want to navigate to **Tools > Network > Host-Only Networks**
- **Uncheck** Enable Server

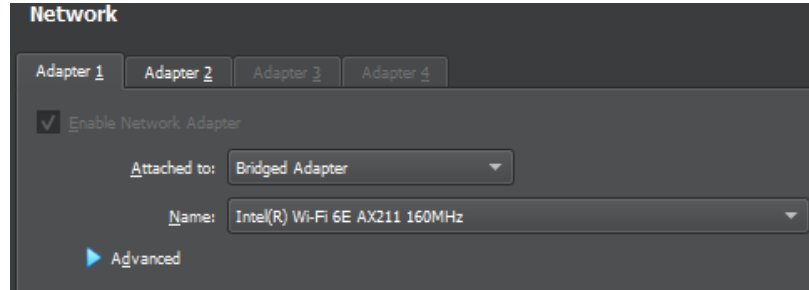
Install CentOS and pfSense:



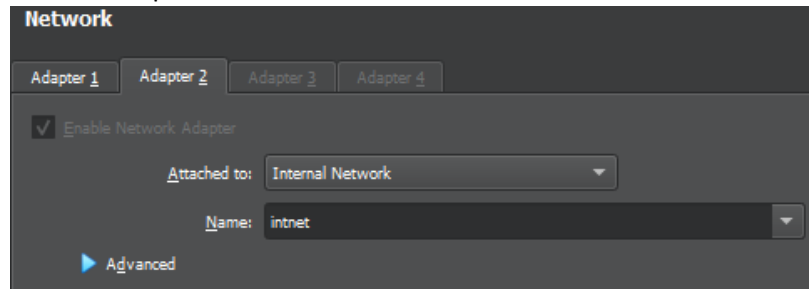
- Configure Network Configuration for the VM's:
  - In Virtual Box, click on intended **VM** and click **Settings > Network**

# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

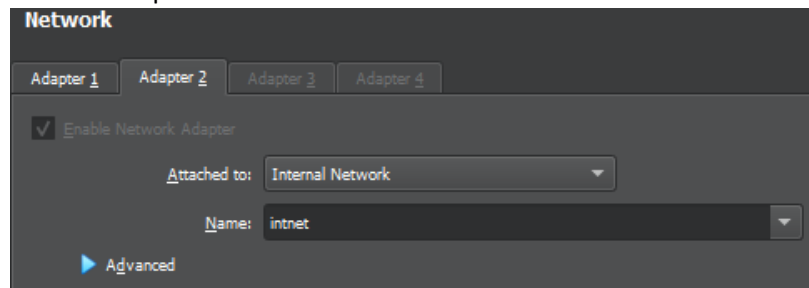
- In pfSense, click on intended **VM** and click **Settings > Network**
  - Adapter 1: **Bridged Network**



- Adapter 2: **Internal Network**



- In CentOS1, click on intended VM and click Settings > Network
  - Adapter 1: **Internal Network**



- Any subsequent CentOS will also connect to *the Internal Network* on Adapter 1

## Log into CentOS:

- Accessing Sudo Privileges:
  - Log into a CentOS account that has administrative privileges or has access to the root user.
  - If you're logged in as a user without **sudo** privileges, try executing commands with su (switch user) to access the root account. You'll be prompted to enter the root password.
  - **su -**
    - Once you're logged in as the **root** user, you can modify the **sudoers** file using the "**visudo**" command:

# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

```
root@localhost:~  
File Edit View Search Terminal Help  
[steve@localhost ~]$ su -  
Password:  
Last login: Tue Nov 28 21:04:11 PST 2023 on pts/0  
[root@localhost ~]# visudo  
[root@localhost ~]# █
```

## - Visudo:

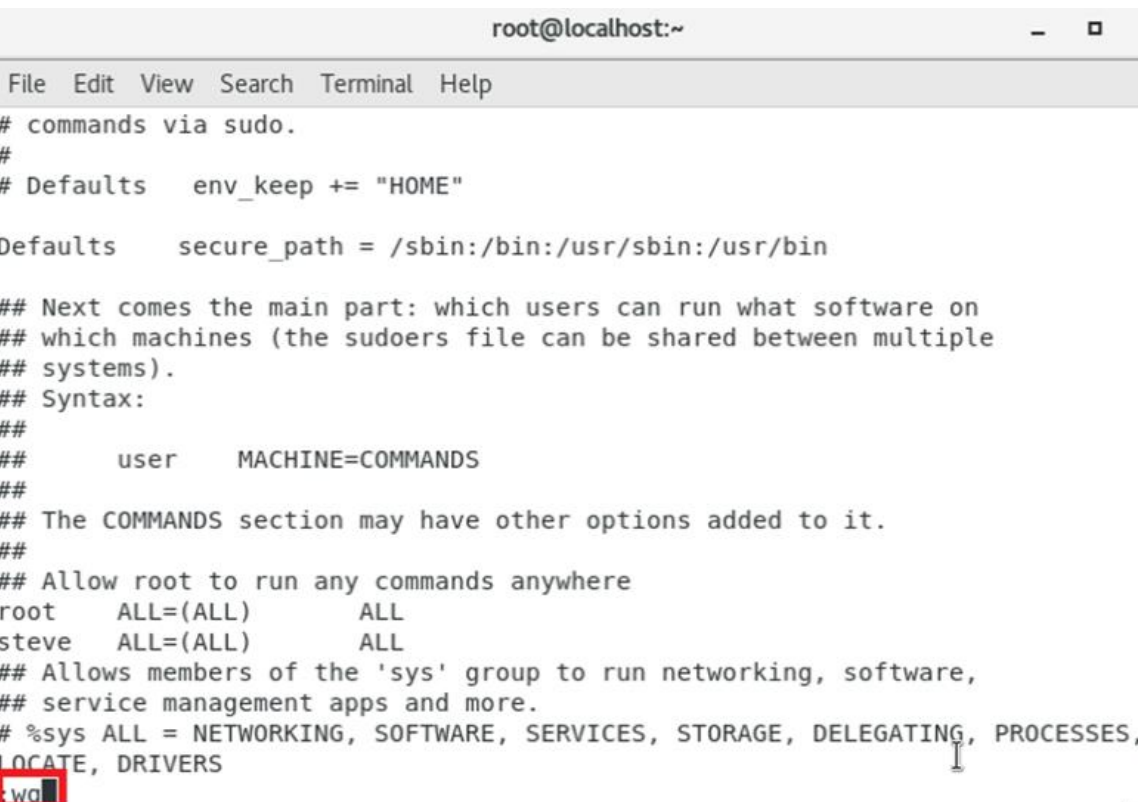
- This command opens the **sudoers** file in a secure manner. Look for the following line in the **sudoers** file:
- **## Allow root to run any commands anywhere**
  - root ALL=(ALL) ALL
  - Below the root entry, add the following line to grant sudo privileges to your username:
  - **steve ALL=(ALL) ALL**
  - Pay close attention to the syntax

```
root@localhost:~  
File Edit View Search Terminal Help  
# Defaults env_keep += "HOME"  
  
Defaults secure_path = /sbin:/bin:/usr/sbin:/usr/bin  
  
## Next comes the main part: which users can run what software on  
## which machines (the sudoers file can be shared between multiple  
## systems).  
## Syntax:  
##  
##      user    MACHINE=COMMANDS  
##  
## The COMMANDS section may have other options added to it.  
##  
## Allow root to run any commands anywhere  
root    ALL=(ALL)    ALL  
steve   ALL=(ALL)    ALL  
## Allows members of the 'sys' group to run networking, software,  
## service management apps and more.  
# %sys ALL = NETWORKING, SOFTWARE, SERVICES, STORAGE, DELEGATING, PROCESSES,  
LOCATE, DRIVERS  
  
## Allows people in group wheel to run all commands  
-- INSERT --
```

- Replace *username* with your *actual* username.

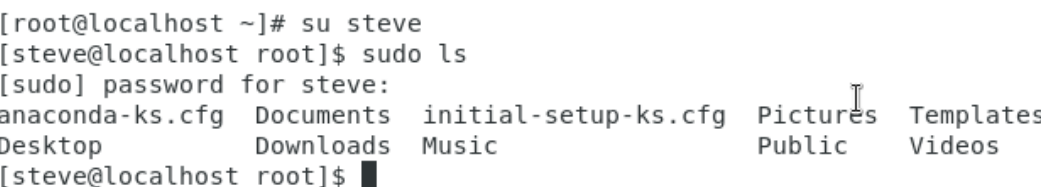
# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

- After making the changes, to save and exit the file using the **vi text** editor:
  - Press **Esc** to ensure you're not in insert mode.
  - Type **:wq** (colon followed by **w** for write and **q** for quit)
  - Press **Enter**.



```
root@localhost:~  
File Edit View Search Terminal Help  
# commands via sudo.  
#  
# Defaults    env_keep += "HOME"  
  
Defaults      secure_path = /sbin:/bin:/usr/sbin:/usr/bin  
  
## Next comes the main part: which users can run what software on  
## which machines (the sudoers file can be shared between multiple  
## systems).  
## Syntax:  
##  
##          user      MACHINE=COMMANDS  
##  
## The COMMANDS section may have other options added to it.  
##  
## Allow root to run any commands anywhere  
root    ALL=(ALL)      ALL  
steve   ALL=(ALL)      ALL  
## Allows members of the 'sys' group to run networking, software,  
## service management apps and more.  
# %sys ALL = NETWORKING, SOFTWARE, SERVICES, STORAGE, DELEGATING, PROCESSES,  
# LOCATE, DRIVERS  
:wq
```

- Now, log out of **root** account and log back in with the same account where you made the changes. You should now have sudo privileges.
- Test **sudo** by running a command with elevated privileges: "**sudo ls**"

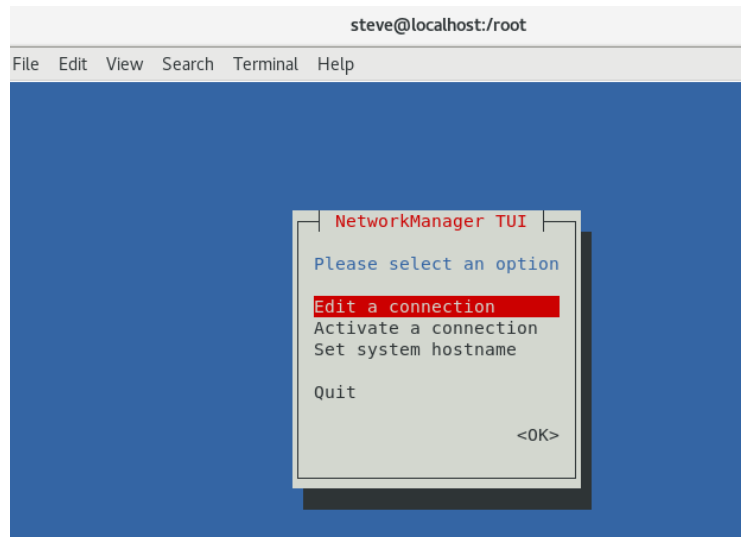


```
[root@localhost ~]# su steve  
[steve@localhost root]$ sudo ls  
[sudo] password for steve:  
anaconda-ks.cfg  Documents  initial-setup-ks.cfg  Pictures  Templates  
Desktop          Downloads  Music                 Public    Videos  
[steve@localhost root]$
```

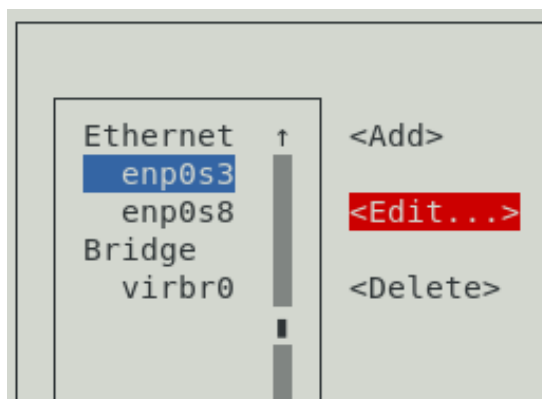
# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

## Enabling Network Connectivity on CentOS:

- Edit Network Configuration: CentOS typically uses **NetworkManager** or network scripts for network configuration. For **NetworkManager**:
- Use **nmtui** (*Network Manager Text User Interface*) command in the terminal.

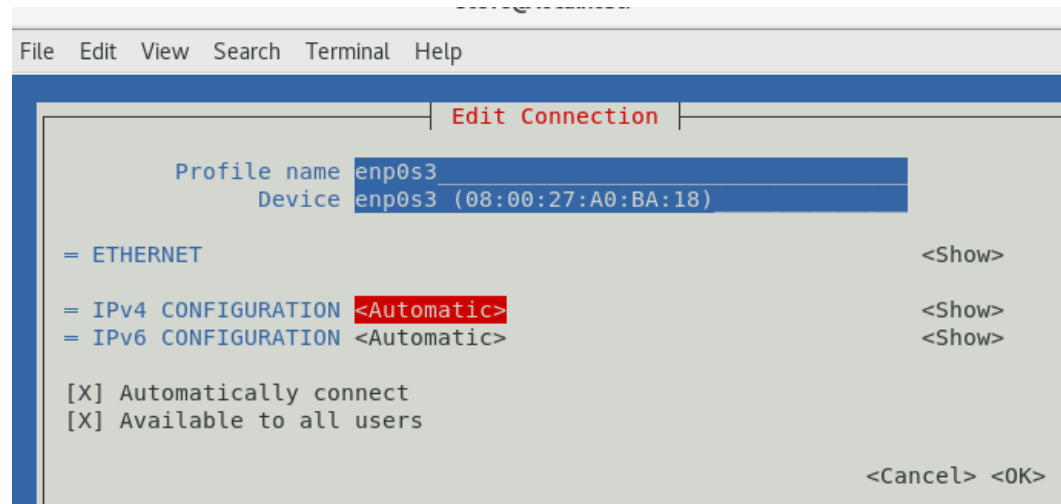


- Edit a connection > Edit **enp0s3** > **Wi-Fi**
  - **This will activate a connection**
  - **Set it to automatically connect**





# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX



- Check network configuration, obtain an IP:
  - **ip a**
- Test connectivity by opening a browser and using
  - **ping**
    - **ping [www.google.ca](http://www.google.ca)**

```
steve@localhost:~  
File Edit View Search Terminal Help  
[steve@localhost ~]$ ping www.google.ca -c 3  
PING www.google.ca (142.251.33.67) 56(84) bytes of data.  
64 bytes from sea09s28-in-f3.1e100.net (142.251.33.67): icmp_seq=1 ttl=121 time=7.76 ms  
64 bytes from sea09s28-in-f3.1e100.net (142.251.33.67): icmp_seq=2 ttl=121 time=8.37 ms  
64 bytes from sea09s28-in-f3.1e100.net (142.251.33.67): icmp_seq=3 ttl=121 time=8.40 ms  
  
--- www.google.ca ping statistics ---  
3 packets transmitted, 3 received, 0% packet loss, time 2004ms  
rtt min/avg/max/mdev = 7.769/8.183/8.406/0.311 ms  
[steve@localhost ~]$
```

- Now that we have connectivity, we need to change the pfSense router configuration for the LAN IP, so it doesn't conflict with my host machine's default gateway of **192.168.1.1** since we are using bridged network configuration to access the external network.
- In pfSense console we will now configure the router accordingly:
  - Select option 2 to set interface(s) IP address

# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

```
0) Logout (SSH only)          9) pfTop
1) Assign Interfaces          10) Filter Logs
2) Set interface(s) IP address 11) Restart webConfigurator
3) Reset webConfigurator password 12) PHP shell + pfSense tools
4) Reset to factory defaults  13) Update from console
5) Reboot system              14) Enable Secure Shell (sshd)
6) Halt system                15) Restore recent configuration
7) Ping host                  16) Restart PHP-FPM
8) Shell

Enter an option: 2

Available interfaces:

1 - WAN (em0 - dhcp, dhcp6)
2 - LAN (em1 - static)

Enter the number of the interface you wish to configure: █
```

- Configure the IPv4 address LAN interface to use a static IP address, this address will correlate directly with the DNS Server which will translate the IP address to a name server and the default gateway which will allow our configuration access to the external network in the cloud.
- Following in line with a /24 CIDR notation we will assign a static class C address of **192.168.2.1**

```
Enter the number of the interface you wish to configure: 2

Configure IPv4 address LAN interface via DHCP? (y/n) n

Enter the new LAN IPv4 address. Press <ENTER> for none:
> 192.168.2.1

Subnet masks are entered as bit counts (as in CIDR notation) in pfSense.
e.g. 255.255.255.0 = 24
     255.255.0.0   = 16
     255.0.0.0     = 8

Enter the new LAN IPv4 subnet bit count (1 to 32):
> 24
```

- Configure the interface, press <ENTER> for none for a LAN
- **No DHCP6 for IPv6 required here so we say N for No**
- We will want to enable the DHCP server on LAN connected with this new address we just inputted so say Y for Yes
- We will want to enter a start address of the IPv4 client address range:
  - **192.168.2.2 – 192.168.2.20**

```
For a WAN, enter the new LAN IPv4 upstream gateway address.
For a LAN, press <ENTER> for none:
>

Configure IPv6 address LAN interface via DHCP6? (y/n) n

Enter the new LAN IPv6 address. Press <ENTER> for none:
>

Do you want to enable the DHCP server on LAN? (y/n) y
Enter the start address of the IPv4 client address range: 192.168.2.2
Enter the end address of the IPv4 client address range: 192.168.2.20
Disabling IPv6 DHCPD...

Do you want to revert to HTTP as the webConfigurator protocol? (y/n) █
```

## LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

- Do you want to revert to HTTP as the webConfigurator protocol? N for No
- Wait while the changes are saved to the LAN...

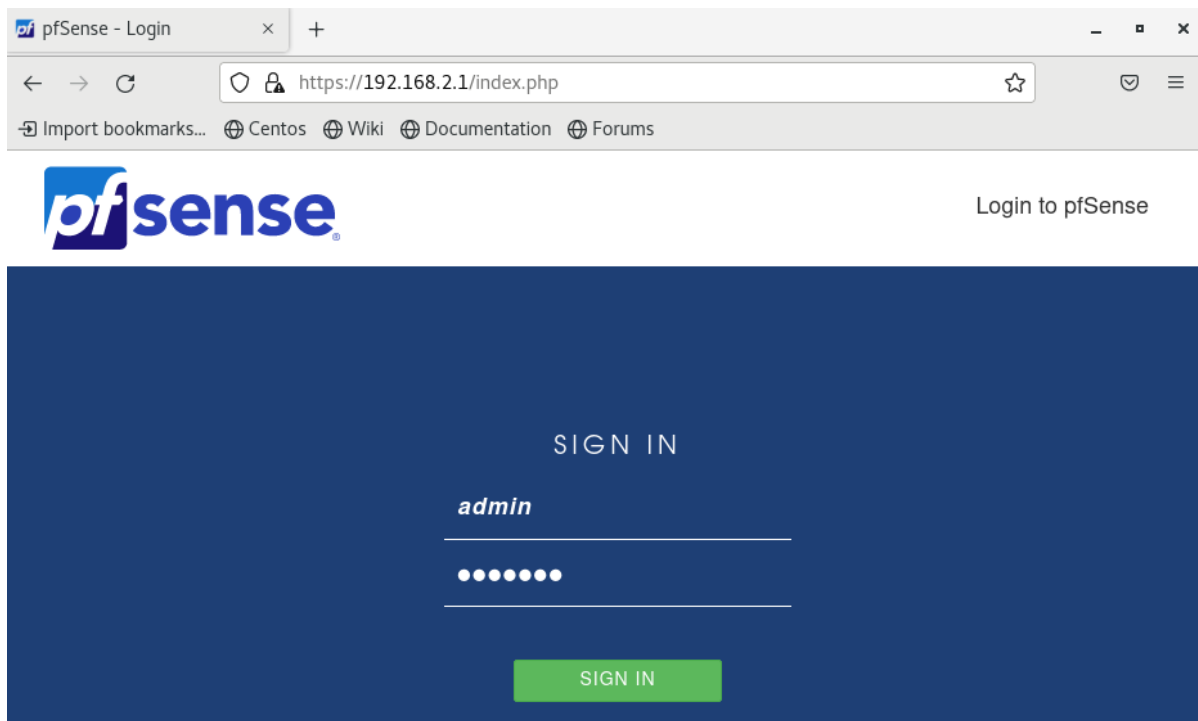
```
Please wait while the changes are saved to LAN...
Reloading filter...
Reloading routing configuration...
DHCPD...

The IPv4 LAN address has been set to 192.168.2.1/24
You can now access the webConfigurator by opening the following URL in your web
browser:

    https://192.168.2.1/

Press <ENTER> to continue.█
```

- Using a web browser such as pfSense, navigate to the router settings page to access the dashboard and configuration settings: <https://192.168.2.1/index.php>



- Navigate the pfSense dashboard to **Status > Interfaces**
  - It displays detailed information about the status of the network interfaces configured on the system

# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

WAN Interface (wan, em0)	
Status	up
DHCP	up  Release WAN <input type="checkbox"/> Relinquish Lease
MAC Address	08:00:27:58:6e:2b
IPv4 Address	192.168.1.110
Subnet mask IPv4	255.255.255.0
Gateway IPv4	192.168.1.1
IPv6 Link Local	fe80::a00:27ff:fe58:6e2b%em0
DNS servers	192.168.1.1
MTU	1500
Media	1000baseT <full-duplex>
In/out packets	35161/23047 (27.91 MiB/4.07 MiB)
In/out packets (pass)	35161/23047 (27.91 MiB/4.07 MiB)
In/out packets (block)	48/0 (16 KiB/0 B)
In/out errors	0/0
Collisions	0
Interrupts	23094 (12/s)

LAN Interface (lan, em1)	
Status	up
MAC Address	08:00:27:ff:4f:4a
IPv4 Address	192.168.2.1
Subnet mask IPv4	255.255.255.0
IPv6 Link Local	fe80::a00:27ff:feff:4f4a%em1
MTU	1500
Media	1000baseT <full-duplex>
In/out packets	16684/29096 (3.76 MiB/26.88 MiB)
In/out packets (pass)	16684/29096 (3.76 MiB/26.88 MiB)
In/out packets (block)	0/0 (0 B/0 B)

- We will renew the network configuration to reflect these changes in CentOS:
  - **Ifconfig** to identify the interface name that is connected to LAN interface, in this case **enp0s3**
  - We will use commands: **sudo ifdown / sudo ifup**

- This will essentially disconnect and reconnect the interface with the new address to update the network configuration in CentOS

```
[steve@localhost ~]$ sudo ifdown enp0s3
[sudo] password for steve:
Device 'enp0s3' successfully disconnected.
[steve@localhost ~]$ sudo ifup enp0s3
Connection successfully activated (D-Bus active path: /org/freedesktop/NetworkManager/ActiveConnection/4)
[steve@localhost ~]$
```

- Run **Ifconfig** to confirm that interface enp0s3 has a new IP as per given in the selected range between **192.168.2.2** and **192.168.2.20**

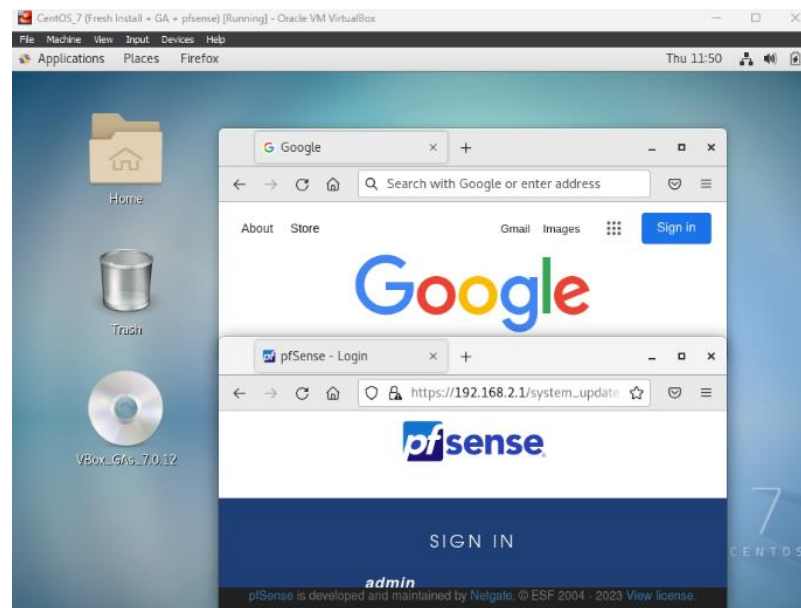
```
[steve@localhost ~]$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.2.2 netmask 255.255.255.0 broadcast 192.168.2.255
    inet6 fe80::e288:e68:eee4:c02c prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:a0:ba:18 txqueuelen 1000 (Ethernet)
    RX packets 320 bytes 246022 (240.2 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 324 bytes 27757 (27.1 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

## LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

- For our designated router to act as the default gateway, we will need to confirm that the static IP of 192.168.2.1 is configured as such along with the DNS server designation as well. This will ensure that we have internet connectivity in CentOS and can access the pfSense dashboard GUI as well
  - We will use command: “**cat /etc/resolv.conf**” to view DNS servers
  - We will use command: “**ip route**” to view the Gateway address

```
[steve@localhost ~]$ cat /etc/resolv.conf
# Generated by NetworkManager
search home.arpa
nameserver 192.168.2.1
[steve@localhost ~]$ ip route
default via 192.168.2.1 dev enp0s3 proto dhcp metric 100
192.168.2.0/24 dev enp0s3 proto kernel scope link src 192.168.2.2 metric 100
192.168.122.0/24 dev virbr0 proto kernel scope link src 192.168.122.1
```

- We have retrieved the proper information now we can try to access these specific areas within CentOS:
- In Firefox: [www.google.ca](http://www.google.ca) and the pfSense dashboard: <https://192.168.2.1>



### Updating and Installing Required Packages:

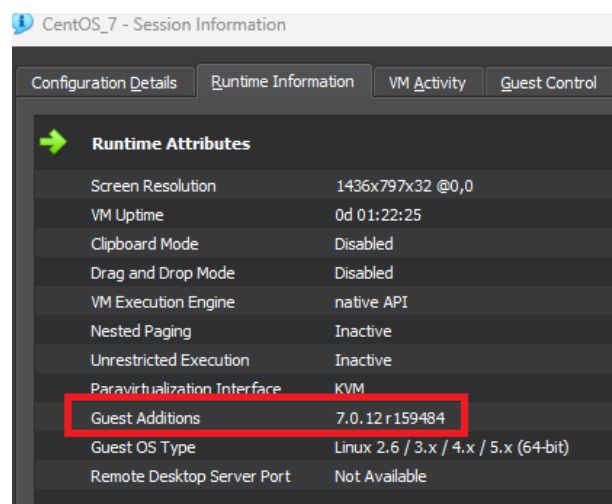
- Prepare CentOS:
  - Ensure OS is running and up to date
- Insert Guest Additions CD Image:
  - In the VirtualBox menu, go to **Devices -> Insert Guest Additions CD Image**
- Open **Terminal** in CentOS

# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

- Install Required Packages:
  - **sudo yum update**
  - **sudo yum install kernel-devel gcc make perl bzip2**
- Mount the Guest Additions ISO:
  - **sudo mkdir /mnt/cdrom**
  - **sudo mount /dev/cdrom /mnt/cdrom**
- Navigate to the Mounted Directory:
  - **cd /mnt/cdrom**
- Run the Installation Script:
  - **Sudo ./VboxLinuxAdditions.run**
- Reboot the CentOS Machine:
  - **sudo reboot**
- Verify Guest Additions are installed:
  - **lsmod | grep vboxguest**

```
steve@localhost:~  
File Edit View Search Terminal Help  
[steve@localhost ~]$ lsmod | grep vboxguest  
vboxguest 375317 6  
[steve@localhost ~]$
```

- Open the Session Information window:
  - Click on **Machine** at the top of the menu in CentOS VM > **Session Information**
  - Select the **Runtime Information** tab.
- Look at the **Guest Additions** version number to verify:



# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

## Updating pfSense:

- In the console running option 13 to update was detecting that there was no package database installed, indicating an issue with the package manager or database
- The steps to fix:
  - o 1. Run option 14, **enable secure shell (sshd)**
  - o 2. Reinstall package database
    - **Pkg-static bootstrap -f**
  - o 3. Update Package Repository Configuration:
    - **Cat /usr/local/etc/pkg/repos/pfSense.conf**
  - o 4. Reset Package Manager Settings:
    - **rm -rf /var/db/pkg/repo-\***
    - **pkg update**
  - o Read the console to verify the package has been successfully updated-

```
Updating pfSense repository catalogue...
Fetching meta.conf: . done
Fetching packagesite.pkg: ... done
Processing entries: ..... done
pfSense repository update completed. 549 packages processed.
All repositories are up to date.
Your packages are up to date
VirtualBox Virtual Machine - Netgate Device ID: d6b7eaba4518943f1f34

*** Welcome to pfSense 2.7.1-RELEASE (amd64) on pfSense ***
```

## Disabling the Firewall in CentOS:

- Check the status of the firewall by running the following command in the terminal:
  - o **sudo firewall-cmd --state**
- If the firewall is running, stop and disable it with the following commands:
  - o **sudo systemctl stop firewalld**
  - o **sudo systemctl disable firewalld**
  - o Verify that the firewall is disabled by checking its status again:
  - o **sudo firewall-cmd --state**

```
steve@localhost:~
File Edit View Search Terminal Help
[steve@localhost ~]$ sudo firewall-cmd --state
[sudo] password for steve:
running
[steve@localhost ~]$ sudo systemctl stop firewalld
[steve@localhost ~]$ sudo systemctl disable firewalld
Removed symlink /etc/systemd/system/multi-user.target.wants/firewalld.service.
Removed symlink /etc/systemd/system/dbus-org.fedoraproject.FirewallD1.service.
[steve@localhost ~]$ sudo firewall-cmd --state
not running
[steve@localhost ~]$
```

# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

## Configuring pfSense router:

- Configure your router (pfSense) to “Register DHCP leases in the DNS Resolver”
  - o In the pfSense dashboard navigate to Services > DNS Resolver

<b>DHCP Registration</b>	<input checked="" type="checkbox"/> Register DHCP leases in the DNS Resolver
	If this option is set, then machines that specify their hostname when requesting an IPv4 DHCP lease will be registered in the DNS Resolver so that their name can be resolved. Note that this will cause the Resolver to reload and flush its resolution cache whenever a DHCP lease is issued. The domain in <a href="#">System &gt; General Setup</a> should also be set to the proper value.

- o We will check to see if the domain in System > General Setup is set to the proper value:

DNS Server Settings	
<b>DNS Servers</b>	<input type="text" value="192.168.2.1"/>
Address	DNS Hostname
Enter IP addresses to be used by the system for DNS resolution. These are also used for the DHCP service, DNS Forwarder and DNS Resolver when it has DNS Query Forwarding enabled.	Hostname
	Enter the DNS Server Hostname for TLS Verification in the DNS Resolver (optional).

## Disable SELINUX on CentOS:

- First, we'll check the SELinux status:
  - o In Terminal, run the command: **sestatus**  

```
[steve@localhost ~]$ sestatus
```

SELinux status:	enabled
SELinuxfs mount:	/sys/fs/selinux
SELinux root directory:	/etc/selinux
Loaded policy name:	targeted
Current mode:	enforcing
Mode from config file:	enforcing
Policy MLS status:	enabled
Policy deny_unknown status:	allowed
Max kernel policy version:	31
- We can see that it's currently **enabled**
  - o To **disable**, in Terminal run command: **sudo nano /etc/selinux/config**
  - o Find the line that starts with '**SELINUX=**' and change its value to '**disabled**'
    - **CTRL-O > Enter**
    - **CTRL-X > Exit**



# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

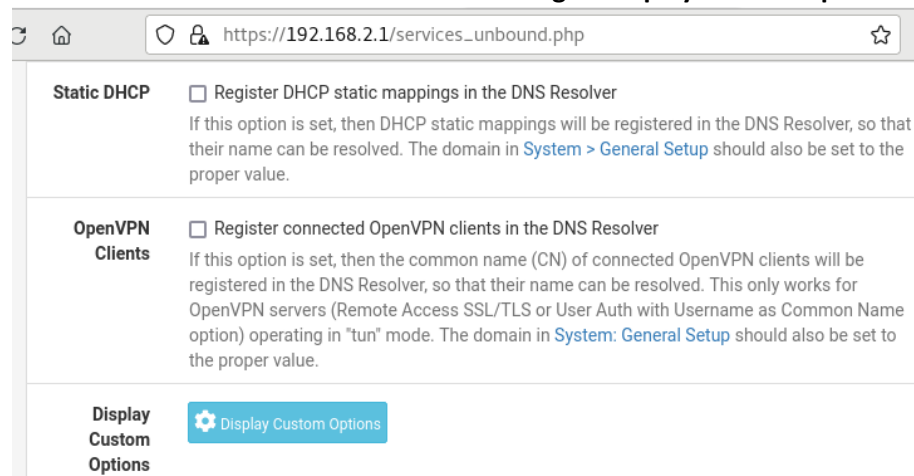
```
steve@localhost:~  
File Edit View Search Terminal Help  
GNU nano 2.3.1 File: /etc/selinux/config Modified  
  
# This file controls the state of SELinux on the system.  
# SELINUX= can take one of these three values:  
#   enforcing - SELinux security policy is enforced.  
#   permissive - SELinux prints warnings instead of enforcing.  
#   disabled - No SELinux policy is loaded.  
SELINUX=disabled  
# SELINUXTYPE= can take one of three values:  
#   targeted - Targeted processes are protected,  
#   minimum - Modification of targeted policy. Only selected p$  
#   mls - Multi Level Security protection.  
SELINUXTYPE=targeted
```

- In terminal, run the command “**sudo reboot**” after making the change
- Then, run the command “**sestatus**” to verify SELinux is now in disabled mode

```
steve@localhost:~  
File Edit View Search Terminal Help  
[steve@localhost ~]$ sestatus  
SELinux status: disabled  
[steve@localhost ~]$
```

## Configure MX records for your router:

- Browse to my pfSense Dashboard (192.168.2.1)
- Navigate to **Services > DNS Resolver > General Settings > Display Custom Options**



- In DNS Resolver we want to check-mark ‘*Register DHCP leases in the DNS Resolver*’:

# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

## DHCP Registration

### ☒ Register DHCP leases in the DNS Resolver

If this option is set, then machines that specify their hostname when requesting an IPv4 DHCP lease will be registered in the DNS Resolver so that their name can be resolved. Note that this will cause the Resolver to reload and flush its resolution cache whenever a DHCP lease is issued. The domain in [System > General Setup](#) should also be set to the proper value.

- **"local-data: "mydomain2.tsp. IN MX 10 mail.mydomain2.tsp"** sets the MX record for the domain **"mydomain2."** with a priority of **10** pointing to the mail server **"mail.mydomain2.tsp"**
- **Mail Routing:** MX records specify the mail servers responsible for receiving email on behalf of your domain
- **PTR Records:** PTR records, also known as Reverse DNS records, map IP addresses to domain names. They allow reverse DNS lookups, which resolve IP addresses to domain names, providing a way to verify that the IP address corresponds to a valid domain
- We will need to create an MX table for both domains. **"mydomain"** and **"mydomain2"**
  - o Server for mydomain.tsp:

## Custom options

```
server:
local-zone: "mydomain.tsp." static
local-data: "mydomain.tsp. IN MX 10 mail.mydomain.tsp."
local-data: "mail.mydomain.tsp. IN A 192.168.2.2"
local-data-ptr: "192.168.2.2 mail.mydomain.tsp"
```

- o Server for mydomain2.tsp:

## Custom options

```
server:
local-zone: "mydomain2.tsp." static
local-data: "mydomain2.tsp. IN MX 10 mail.mydomain2.tsp."
local-data: "mail.mydomain2.tsp. IN A 192.168.2.3"
local-data-ptr: "192.168.2.3 mail.mydomain2.tsp"
```

- This is also a **FQDN (fully qualified domain name)** which assists in the functioning of the email infrastructure, preventing confusion and mail routing issues
- **Save** and **apply** the changes
- Navigate to **System > General Setup**
  - o Type the IP address for the DNS Server used to resolve DHCP:

# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

**DNS Server Settings**

<b>DNS Servers</b>	<input type="text" value="192.168.2.1"/>	<b>DNS Hostname</b>	<input type="text"/>
	Address		Hostname
	Enter IP addresses to be used by the system for DNS resolution. These are also used for the DHCP service, DNS Forwarder and DNS Resolver when it has DNS Query Forwarding enabled.		Enter the DNS Server Hostname for TLS Verification in the DNS Resolver (optional).

**Add DNS Server** + Add DNS Server

## Verifying Network Internet Access:

- Ping google.ca from the router:

```
[2.7.1-RELEASE][root@pfSense.home.arpal/root: ping google.ca
PING google.ca (172.217.14.227): 56 data bytes
64 bytes from 172.217.14.227: icmp_seq=0 ttl=121 time=7.534 ms
64 bytes from 172.217.14.227: icmp_seq=1 ttl=121 time=8.063 ms
64 bytes from 172.217.14.227: icmp_seq=2 ttl=121 time=8.156 ms
64 bytes from 172.217.14.227: icmp_seq=3 ttl=121 time=8.605 ms
^Z
Suspended
[2.7.1-RELEASE][root@pfSense.home.arpal/root: █
```

- Ping google.ca from CentOS:

```
[steve@localhost ~]$ ping google.ca
PING google.ca (172.217.14.227) 56(84) bytes of data.
64 bytes from sea30s02-in-f3.1e100.net (172.217.14.227): icmp_seq=1 ttl=120 time=9.18 ms
64 bytes from sea30s02-in-f3.1e100.net (172.217.14.227): icmp_seq=2 ttl=120 time=9.39 ms
64 bytes from sea30s02-in-f3.1e100.net (172.217.14.227): icmp_seq=3 ttl=120 time=9.29 ms
64 bytes from sea30s02-in-f3.1e100.net (172.217.14.227): icmp_seq=4 ttl=120 time=8.86 ms
64 bytes from sea30s02-in-f3.1e100.net (172.217.14.227): icmp_seq=5 ttl=120 time=8.51 ms
^Z
[1]+  Stopped                  ping google.ca
[steve@localhost ~]$
```

- You can enable SELINUX on CentOS after configuration is done
- You can enable the firewall on CentOS after the configuration is done

## Questions:

1. Which DNS server resolves the names for your private network?
  - a. For a private network such as this one, the DNS resolution for domain names within that network is typically handled by a DNS server set up specifically for internal or private use. In our case it's designated as (192.168.2.1). We can check this by using the command **cat /etc/resolv.conf**. This contains the DNS server that the system is currently using for DNS resolution.

## LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

2. What is the address of the router that passes traffic directly to the school network?
  - a. The gateway to access the internet for the VM will typically be the IP address of the pfSense router's interface facing the external network (Internet)
    - i. 192.168.2.1
3. What is the MAC address of the machine the CentOS box is communicating with?
  - a. 08:00:27:a0:ba:18
4. When you edited the SELINUX config file, what did you change?
  - a. Changed the SELINUX inside the text editor from enabled to disabled.
5. How did you disable the firewall on CentOS?
  - a. I used the command
    - **sudo systemctl stop firewalld**
    - **sudo systemctl disable firewalld**
6. Did you document what went right and what went wrong?
  - a. Yes
7. Could you rebuild this setup from your notes?
  - a. Yes

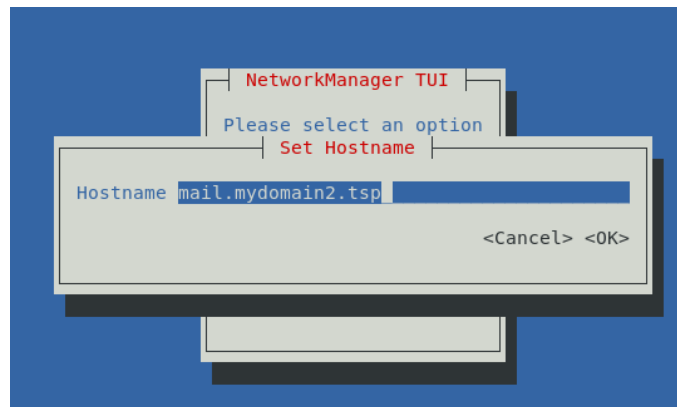
### Installing Postfix:

- Install a secondary CentOS VM: Have it fully updated with Guest Additions installed
- Postfix Installation:
  - o Install Postfix on both CentOS VMs
  - o Use this command:
    - **sudo yum install postfix**

```
steve@localhost:~  
File Edit View Search Terminal Help  
[steve@localhost ~]$ sudo yum install postfix  
[sudo] password for steve:  
Loaded plugins: fastestmirror, langpacks  
Loading mirror speeds from cached hostfile  
* base: lesnet.mm.fcix.net  
* extras: lesnet.mm.fcix.net  
* updates: lesnet.mm.fcix.net  
Package 2:postfix-2.10.1-9.el7.x86_64 already installed and latest version  
Nothing to do  
[steve@localhost ~]$ █
```

- Here we will modify the '**myhostname**' parameter according to my server's hostname
- **Network Manger Text User Interface** for each VM, this will help with our networking purposes.
  - o Use command: **sudo nmtui**
  - o **Set system hostname**
    - In this example, I'll show what was derived for VM2

# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX



- Enter the command: **“hostnamectl status”** in Terminal to reply with your current hostname

```
guest2@mail:~  
File Edit View Search Terminal Help  
[guest2@mail ~]$ nmtui  
[guest2@mail ~]$ hostnamectl status  
Static hostname: mail.mydomain2.tsp  
Icon name: computer-vm  
Chassis: vm  
Machine ID: 54387a04e779544f973a9708a6113ec0  
Boot ID: 76bfba22e1ed4aaa929d5927764a9608  
Virtualization: kvm  
Operating System: CentOS Linux 7 (Core)  
CPE OS Name: cpe:/o:centos:centos:7  
Kernel: Linux 3.10.0-1160.102.1.el7.x86_64  
Architecture: x86-64  
[guest2@mail ~]$
```

- Repeat these steps for VM1. It will read: **“mail.mydomain.tsp”**
- Now we will proceed with the next phase, with the hostnames clearly established for each VM will be able to distinguish each domain clearly when we edit the Postfix configuration file so it can communicate as a SMTP server for the purposes of sending a message from one mail directory to another.

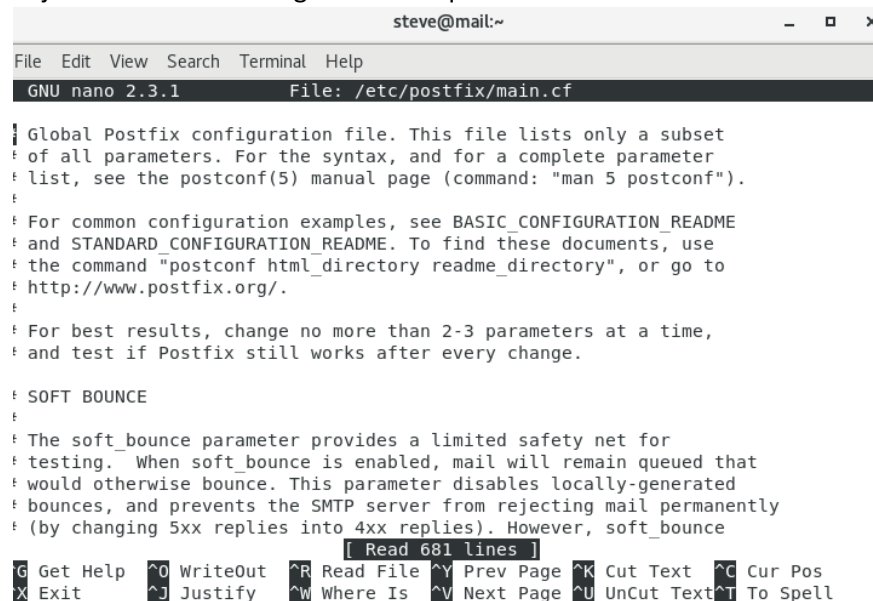
## Configuring POSTFIX:

- Locate the field and modify if necessary:
  - “myhostname”
  - “mydomain”
  - “myorigin”
  - “mydestination”
  - “inet\_interfaces”
  - “mynetworks”
  - “home\_mailbox”

# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

- To enter basic configuration, we need to edit postfix in the editor on *both VM's*
- Keep in mind that some of the information entered will be specific to the VM hostname and domain and must be adjusted accordingly:

- We run the command: **sudo nano /etc/postfix/main.cf**
- Here we will be introduced to the area where we will find the information needed to adjust the Postfix settings for our requirements:



```
steve@mail:~
File Edit View Search Terminal Help
GNU nano 2.3.1 File: /etc/postfix/main.cf

Global Postfix configuration file. This file lists only a subset
of all parameters. For the syntax, and for a complete parameter
list, see the postconf(5) manual page (command: "man 5 postconf").

For common configuration examples, see BASIC_CONFIGURATION_README
and STANDARD_CONFIGURATION_README. To find these documents, use
the command "postconf html_directory readme_directory", or go to
http://www.postfix.org/.

For best results, change no more than 2-3 parameters at a time,
and test if Postfix still works after every change.

SOFT BOUNCE

The soft_bounce parameter provides a limited safety net for
testing. When soft_bounce is enabled, mail will remain queued that
would otherwise bounce. This parameter disables locally-generated
bounces, and prevents the SMTP server from rejecting mail permanently
(by changing 5xx replies into 4xx replies). However, soft_bounce

[ Read 681 lines ]
G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell
```

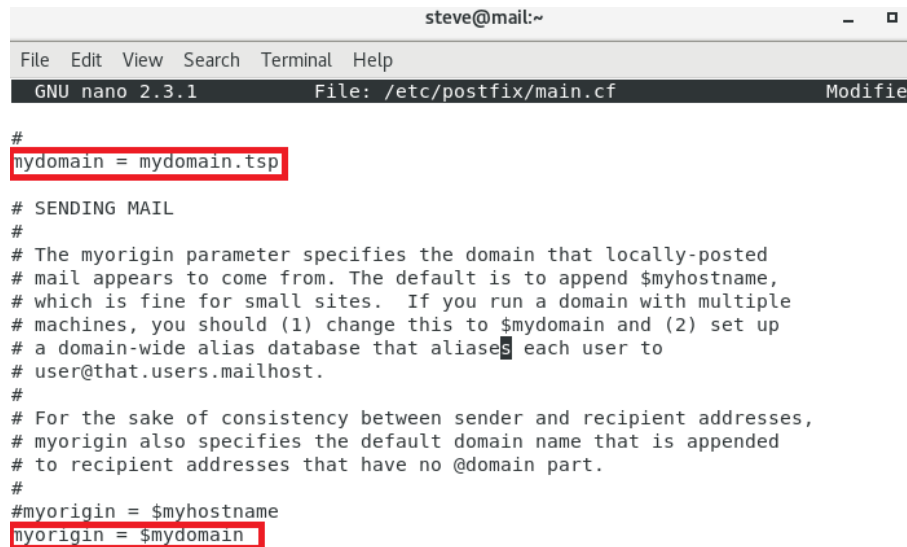
- To navigate this editing tool, we will use **CTRL-W** and search for the parameter:
  - **"myhostname"**
    - This will take us to the corresponding area to make the changes
    - This will help us establish the fundamental identify and behavior of the mail sever

```
..
myhostname = mail.mydomain.tsp
#myhostname = virtual.domain.tld
```

- *Lines in the configuration file that begin with a hashtag are considered comments, comments are ignored by the mail server*
- *Uncommented lines without a hashtag are considered active settings or directives in the configuration file. They contain the actual configuration parameters*
- When editing the file, type **"I"** for insert, **"ESC"** to leave insert mode
- When it's time to save and exit the editing tool, press ESC and type:
  - **":x!"**

# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

- After adjusting myhostname for VM1 we will scroll down and adjust the other settings necessary to make the mail server work according to our specifications:
- **“mydomain”**
  - More identifiers
- **“myorigin”**
  - Determines the domain name that is appended to locally posted email
  - **“\$”** in front of a parameter signifies a variable reference. When Postfix read the configuration file, it interprets variables predicted with **“\$”** and substitutes them with their defined values



```
steve@mail:~
File Edit View Search Terminal Help
GNU nano 2.3.1 File: /etc/postfix/main.cf Modified

#
mydomain = mydomain.tsp

# SENDING MAIL
#
# The myorigin parameter specifies the domain that locally-posted
# mail appears to come from. The default is to append $myhostname,
# which is fine for small sites. If you run a domain with multiple
# machines, you should (1) change this to $mydomain and (2) set up
# a domain-wide alias database that aliases each user to
# user@that.users.mailhost.
#
# For the sake of consistency between sender and recipient addresses,
# myorigin also specifies the default domain name that is appended
# to recipient addresses that have no @domain part.
#
myorigin = $myhostname
myorigin = $mydomain
```

- **“mydestination”**
    - Specifies how incoming mail should be handled and delivered. It also defines the list of domains for which the server will accept mail
- ```
#mydestination = $myhostname, localhost.$mydomain, localhost
mydestination = $myhostname, localhost.$mydomain, localhost, $mydomain
#mydestination = $myhostname, localhost.$mydomain, localhost, $mydomain,
# mail.$mydomain, www.$mydomain, ftp.$mydomain
```
- By adding a hashtag to the first line and uncommenting the second line, it adds \$mydomain. This addition means that the mail server will also accept emails addressed to its down domain explicitly. This helps if you expect emails to be addressed directly to the mail server’s domain itself, ensuring that they are accepted and processed locally
- **“inet\_interfaces”**
    - Network related setting that defines the network interface on which the Postfix mail system should listen for incoming connections

# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

```
#  
inet_interfaces = all  
#inet_interfaces = $myhostname  
#inet_interfaces = $myhostname, localhost  
#inet_interfaces = localhost
```

- Uncommented to allow for all, Postfix will listen on all available network interfaces on the server including loopback interface. This allows for ease of setup

- “mynetworks”

- Specifies the trusted network addresses or IP ranges that are allowed to relay or send emails through the Postfix mail server without authentication
- By default, it includes the loopback address (**127.0.0.0/8**)

```
mynetworks = 192.168.2.0/24, 127.0.0.0/8  
#mynetworks = $config_directory/mynetworks  
#mynetworks = hash:/etc/postfix/network_table
```

- Since my pfSense router oversees defining DNS and supplying DHCP within a subnet, the ‘**0**’ represents the network address of that subnet
  - DNS: 192.168.2.1
  - VM1: 192.168.2.2
  - VM2: 192.168.2.3
- The DHCP server on pfSense usually assigns IP addresses within the defined range of the subnet but avoids assigning ‘**0**’ and ‘**.255**’ addresses to devices, as they are reserved for the network address and broadcast address

- “home\_mailbox”

- Determines the default location for storing user’s mail. It sets the path to the default mailbox file for local delivery of incoming emails

```
#home_mailbox = Mailbox  
home_mailbox = Maildir/
```

- The ‘**Maildir**’ format is a common mailbox format that stores each email as a separate file within various subdirectories (**cur, new, tmp**)

- This concludes the configuration part of Postfix, once you are comfortable with the changes, press the ESC key and type :x! to save and quit the editing tool
- I configured VM1, VM2 will be similar but the values will change for:
  - **myhostname**: mail.mydomain2.tsp
  - **mydomain**: mydomain2.tsp

## Creating a new user in CentOS:

- Open Terminal: Run the command: “**sudo add user guest2**”
  - To set a password for the new user type:
    - “**sudo passwd guest2**”



# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

```
steve@mail:~  
File Edit View Search Terminal Help  
[steve@mail ~]$ sudo add user guest2
```

- Enter the username for which you want to set the password
  - You will be prompted to enter and confirm the new password
  - Press Enter, then re-enter the same password when prompted
- To verify the new user:
    - o Run the command: **"getent passwd | grep guest2"** to display details:

```
steve@mail:~  
File Edit View Search Terminal Help  
[steve@mail ~]$ getent passwd | grep guest2  
guest2:x:1001:1001::/home/guest2:/bin/bash  
[steve@mail ~]$ █
```

- To switch user:
  - o Run the command: **"su - guest2"**

## Create a Mail Directory for a user:

- Once the user is created, you can create a mail directory for them by running the command:
  - o **"sudo mkdir -p /home/guest2/mail"**
- We'll want to ensure that the user guest2 has proper ownership and permissions for their mail directory by running the command:
  - o **"sudo chown -R guest2:guest2 /home/guest2/mail"**

## Create a Maildir for a user:

- Maildir structure typically consists of three subdirectories: 'cur', 'new', and 'tmp'
- To create this structure run the command:
  - o **"sudo -u guest2 mkdir -p /home/guest2/Maildir/{cur,new,tmp}"**
- Set permissions:
  - o **"sudo chown -R guest2:guest2 /home/guest2/Maildir"**

## Verify the Mail/Maildir Structures:

- Use the **"sudo -u guest2 ls -la /home/guest2/Maildir"** command at terminal to display the contents of the **'Maildir'** directory. The Maildir structure is commonly used for storing emails in a format that's suitable for IMAP (Internet Message Access Protocol) servers and clients.
- To simply check the directory creation run the command:  
**"ls -l /home/guest2"**

## LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

- The differences between Mail and Maildir:
  - o **Mail Format:** In the traditional "Mail" format (also known as mbox), all messages for a particular mailbox are stored within a single file
  - o **Maildir Format:** In the Maildir format, each email message is stored as a separate file within a directory structure. Each email message is saved in its own file with a unique name, usually stored in a directory hierarchy (e.g., new, cur, tmp) for organization

```
guest2@mail:~  
File Edit View Search Terminal Help  
[guest2@mail ~]$ ls -l /home/guest2  
total 0  
drwxr-xr-x. 2 guest2 guest2 6 Dec 6 11:26 Desktop  
drwxr-xr-x. 2 guest2 guest2 6 Dec 3 14:11 Documents  
drwxr-xr-x. 2 guest2 guest2 6 Dec 3 14:11 Downloads  
drwx-----. 2 guest2 guest2 6 Dec 5 12:14 Mail  
drwx-----. 5 guest2 guest2 201 Dec 6 12:56 Maildir  
drwxr-xr-x. 2 guest2 guest2 6 Dec 3 14:11 Music  
drwxr-xr-x. 2 guest2 guest2 6 Dec 3 14:11 Pictures  
drwxr-xr-x. 2 guest2 guest2 6 Dec 3 14:11 Public  
drwxr-xr-x. 2 guest2 guest2 6 Dec 3 14:11 Templates  
drwxr-xr-x. 2 guest2 guest2 6 Dec 3 14:11 Videos  
[guest2@mail ~]$ sudo -u guest2 ls -la /home/guest2/Maildir  
total 32  
drwx-----. 5 guest2 guest2 201 Dec 6 12:56 .  
drwx-----. 18 guest2 guest2 4096 Dec 6 12:45 ..  
drwx-----. 2 guest2 guest2 4096 Dec 6 12:56 cur  
-rw-----. 1 guest2 guest2 5296 Dec 6 12:49 dovecot.index.cache  
-rw-----. 1 guest2 guest2 2316 Dec 6 12:56 dovecot.index.log  
-rw-----. 1 guest2 guest2 6 Dec 5 12:16 dovecot-keywords  
-rw-----. 1 guest2 guest2 803 Dec 6 12:53 dovecot-uidlist  
-rw-----. 1 guest2 guest2 8 Dec 5 12:10 dovecot-uidvalidity  
-r--r--r--. 1 guest2 guest2 0 Dec 5 12:10 dovecot-uidvalidity.656f83b6  
drwx-----. 2 guest2 guest2 6 Dec 6 12:49 new  
drwx-----. 2 guest2 guest2 6 Dec 5 19:31 tmp  
[guest2@mail ~]$
```

### Enabling Postfix on runlevels 2,3 and 5:

- The purpose is to configure the system so that the Postfix mail service automatically starts when the Linux system enters these runlevels. Each runlevel determines which services or processes should be active

## LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

- Runlevel 2: Multi-user mode without networking
- Runlevel 3: Multi-user mode with networking
- Runlevel 5: Multi-user mode with networking and GUI
- Open a terminal
- Type: “**sudo chkconfig postfix on**” to enable Postfix for the default runlevels, including 2, 3 and 5
- To verify these entries are enabled type: “**chkconfig --list**”  

```
[steve@mail ~]$ sudo chkconfig postfix on  
[sudo] password for steve:  
Note: Forwarding request to 'systemctl enable postfix.service'.  
[steve@mail ~]$ chkconfig --list
```

Note: This output shows SysV services only and does not include native systemd services. SysV configuration data might be overridden by native systemd configuration.

If you want to list systemd services use 'systemctl list-unit-files'.  
To see services enabled on particular target use 'systemctl list-dependencies [target]'.

|            |       |       |       |       |       |       |       |
|------------|-------|-------|-------|-------|-------|-------|-------|
| netconsole | 0:off | 1:off | 2:off | 3:off | 4:off | 5:off | 6:off |
| network    | 0:off | 1:off | 2:on  | 3:on  | 4:on  | 5:on  | 6:off |
- *In some distributions of Unix systems, runlevel 4 is unused and has no specific functionality assigned to it*

### Using Telnet:

- Ensure Postfix is running on both VM's and configured to allow connections from each other
  - Editing 'mynetworks' and 'inet\_interfaces' will ensure this
- Check firewall settings, ensure that the VM's can communicate over the network and that the port used by Postfix (usually 25 for SMTP) is open
  - Since my firewall is on and active, I created a rule which allows port 25 for incoming connections. Open terminal and type:  

```
sudo firewall-cmd --zone=public --add-port=25/tcp --permanent
```
  - Reload the firewall:  

```
sudo firewall-cmd --reload
```
- Additionally, you can create a rule to allow Telnet traffic (port 23) though the firewall on CentOS.
- Check which ports are allowed in zones:  

```
[steve@mail ~]$ sudo firewall-cmd --zone=public --list-ports  
23/tcp
```
- Use Telnet to manually connect to Postfix and send an email from one VM to the other. By default, Postfix listens on Port 25 for SMTP connections with the following syntax:

## LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

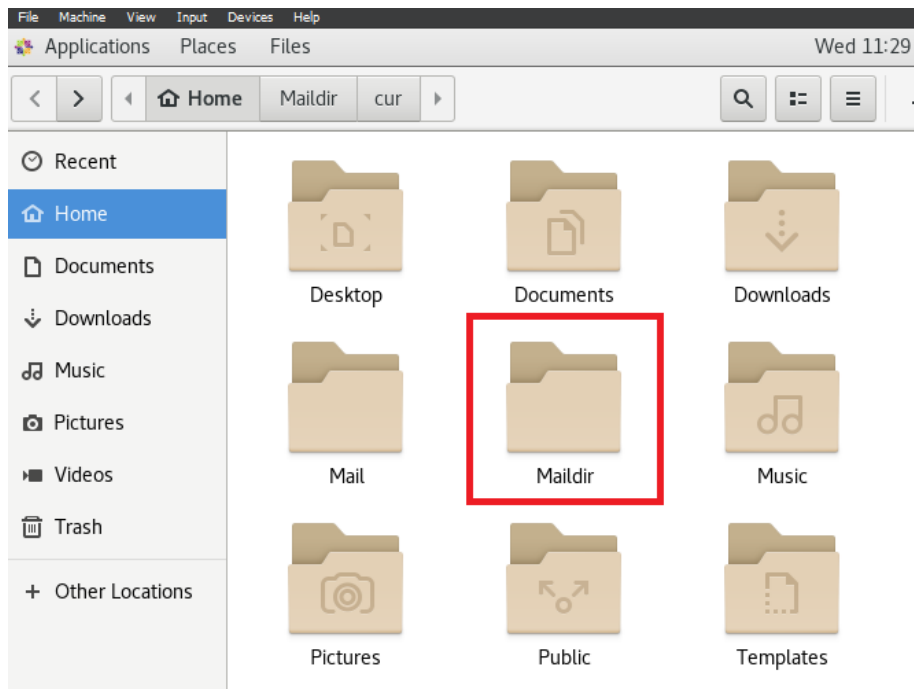
- At terminal, type **"telnet 192.168.2.3 25"**
  - o Establish connection
- Enter "EHLO **"mydomain.tsp"**"
- MAIL FROM: [steve@mydomain.tsp](mailto:steve@mydomain.tsp)
- RCPT TO: [guest2@mydomain2.tsp](mailto:guest2@mydomain2.tsp)
- DATA
  - o Enter subject
  - o The body of the message
- End the data section with a single period on a line by itself
- QUIT
  - o This closes the connection

```
[steve@mail ~]$ telnet 192.168.2.3 25
Trying 192.168.2.3...
Connected to 192.168.2.3.
Escape character is '^]'.
220 mail.mydomain2.tsp ESMTP Postfix
EHLO mydomain.tsp
250-mail.mydomain2.tsp
250-PIPELINING
250-SIZE 10240000
250-VERFY
250-ETRN
250-ENHANCEDSTATUSCODES
250-8BITMIME
250 DSN
MAIL FROM: steve@mydomain.tsp
250 2.1.0 Ok
RCPT TO: guest2@mydomain2.tsp
250 2.1.5 Ok
DATA
354 End data with <CR><LF>.<CR><LF>
*Warning* This is a test email only!

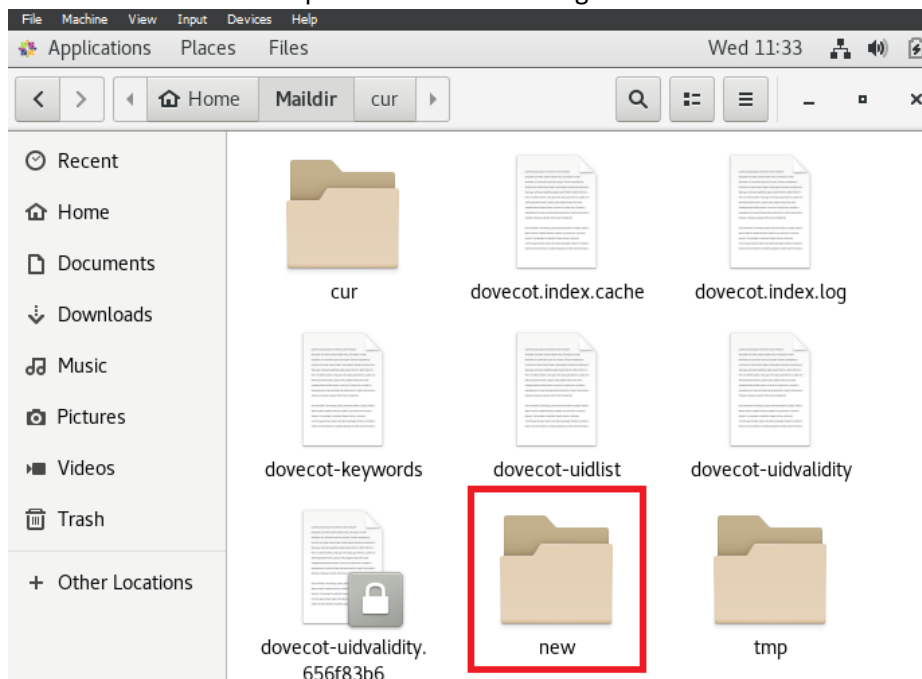
.
250 2.0.0 Ok: queued as 49FCF212BCF5
quit
```

## LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

- On VM2 (mail.mydomain2.tsp) navigate to the *Home* folder on the desktop and click on the *Maildir*



- Click the *new* folder to open the latest mail for guest2



## LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

- Open the newest file:

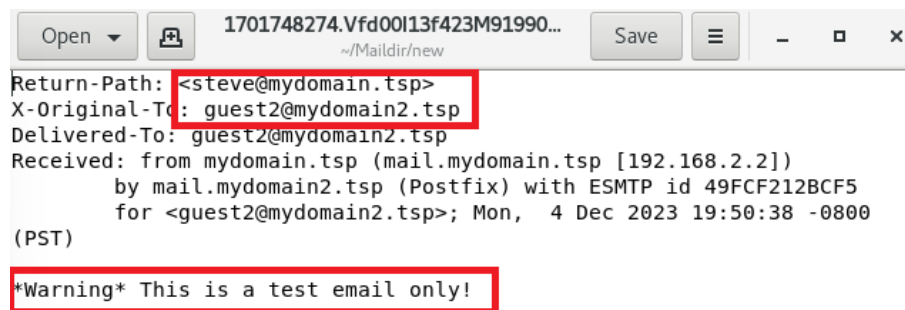


1701736195.

Vfd00I13f422M4416.

-- mail.mvdomain2.tsp:--

- Confirm it's been received:



- *Note: \*Its advised that using Telnet is typically a method to test the mail system and not used for regular mail transfer, which should be done through email clients or applications using the SMTP protocol securely\**
- *Note: \* Since May 2022, Google has prohibited access to Google accounts for less secure apps and won't accept connections from telnet\**

### Configuring POP/IMAP Server:

- To install Dovecot, CentOS 7.0 and Postfix should be installed
- Dovecot is an open-source IMAP and POP3 server
- Postfix should be properly configured on one VM to handle outgoing email (SMTP) while the second VM is configured with Dovecot to handle incoming email (IMAP/POP3)

### Install Dovecot:

- Download and install the Dovecot package by running the following command:
  - o **"sudo yum install dovecot -y"**
    - We will install and configure Dovecot on VM2, here it will handle incoming email in the form of IMAP and POP3

# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

```
guest2@mail:~
File Edit View Search Terminal Help
=====
Package Arch Version Repository Size
=====
Installing:
dovecot x86_64 1:2.2.36-8.el7 base 4.4 M
Installing for dependencies:
clucene-core x86_64 2.3.3.4-11.el7 base 528 k
portreserve x86_64 0.0.5-11.el7 base 26 k
=====
Transaction Summary
=====
Install 1 Package (+2 Dependent packages)

Total download size: 4.9 M
Installed size: 16 M
Downloading packages:
(1/3): clucene-core-2.3.3.4-11.el7.x86_64.rpm | 528 kB 00:00
(2/3): portreserve-0.0.5-11.el7.x86_64.rpm | 26 kB 00:00
(3/3): dovecot-2.2.36-8.el7.x86_64.rpm | 4.4 MB 00:03
-----
Total 1.2 MB/s | 4.9 MB 00:04
```

## Configuring Dovecot:

- Once the installation is finished, you need to configure the services in the configuration file at terminal with the command: **"sudo nano /etc/dovecot/dovecot.conf"**
- Like Postfix, '#' character and everything after it is treated as comments
- We will uncomment the following line in the file:
  - o **"protocols = imap pop3 lmtp"**

```
guest2@mail:~
File Edit View Search Terminal Help
GNU nano 2.3.1 File: /etc/dovecot/dovecot.conf

# --sysconfdir=/etc --localstatedir=/var

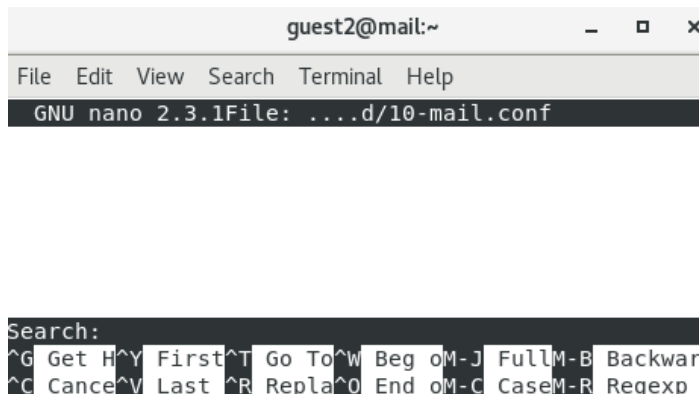
# Protocols we want to be serving.
protocols = imap pop3 lmtp
```

- **CTRL-O to + Enter** to save the modification
- **CTRL-X** to exit the editor

## Configure the location of the Mailbox:

- Use the following command at terminal: **"sudo nano /etc/dovecot/conf.d/10-mail.conf"**
- Search using **CTRL-W**

# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

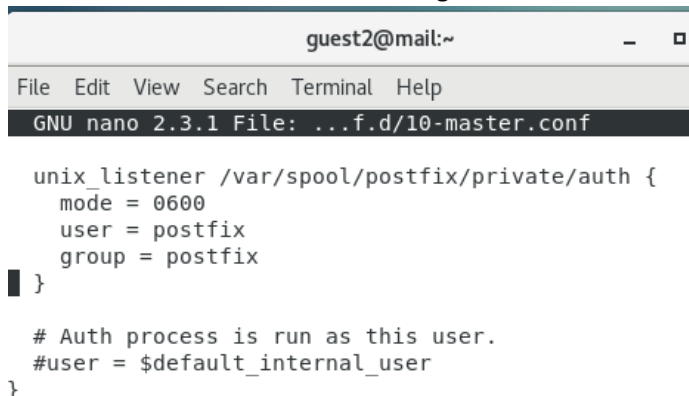


```
guest2@mail:~  
File Edit View Search Terminal Help  
GNU nano 2.3.1File: ...d/10-mail.conf  
  
Search:  
^G Get H^Y First^T Go To^W Beg oM-J FullM-B Backwar  
^C Cance^V Last ^R Repla^O End oM-C CaseM-R Regexp
```

- Uncomment the following line:
  - o **"mail\_location = maildir:~/Maildir"**  
mail\_location = maildir:~/Maildir
- **CTRL-O + Enter** to save
- **CTRL-X** to exit

## Configure service settings for Dovecot:

- Use the following command at terminal: **"sudo nano /etc/dovecot/conf.d/10-master.conf"**
- Search for **"smtp-auth"**
  - o Edit the following lines:
    - Mode changes to **'0600'** ensures that only authorized processes can communicate through this socket



```
guest2@mail:~  
File Edit View Search Terminal Help  
GNU nano 2.3.1 File: ...f.d/10-master.conf  
  
unix_listener /var/spool/postfix/private/auth {  
    mode = 0600  
    user = postfix  
    group = postfix  
}  
  
# Auth process is run as this user.  
#user = $default_internal_user  
}
```

- **CTRL-O + Enter** to save
- **CTRL-X** to exit

## Configure authentication mechanisms for connecting clients:

- Use the following command at terminal: **"sudo nano /etc/dovecot/conf.d/10-auth.conf"**
- Uncomment **"disable\_plaintext\_auth = yes"**
- Add the word *login* proceeding **"auth\_mechanisms = plain"**
  - o **"auto\_mechanisms = plain login"**



## LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

- **CTRL-W** to search `auth_mechanisms`

```
File Edit View Search Terminal Help
GNU nano 2.3.1 File: /etc/d

## Password and user databases
##

#
# Password database is used to v
# You can have multiple passdbs
# allow both system users (/etc/
# duplicating the system users i
#
# <doc/wiki/PasswordDatabase.txt
Search: auth_mechanisms
```

- Edit the line:

```
guest2@mail:~
File Edit View Search Terminal Help
GNU nano 2.3.1 File: /etc/dovecot/conf.d/10-auth.conf

# Space separated list of wanted authentication mechanisms:
# plain login digest-md5 cram-md5 ntlm rpa apop anonymous
# gss-spnego
# NOTE: See also disable_plaintext_auth setting.
auth_mechanisms = plain login
```

- *'plain'* and *'login'* are standard authentication mechanisms used by many email clients to authenticate with email servers over protocols like IMAP and POP3

### Enable Dovecot to start its services:

- At terminal enter the command `“sudo systemctl enable dovecot”`
- After enabling the service, start the Dovecot service using the following command:
  - o `“sudo systemctl start dovecot”`
- Check the status of Dovecot:
  - o `“sudo systemctl status dovecot”`

# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

```
guest2@mail:~  
File Edit View Search Terminal Help  
[guest2@mail ~]$ sudo systemctl enable dovecot  
Created symlink from /etc/systemd/system/multi-user.target.wants/dovecot.service  
to /usr/lib/systemd/system/dovecot.service.  
[guest2@mail ~]$ sudo systemctl start dovecot  
[guest2@mail ~]$ sudo systemctl status dovecot  
● dovecot.service - Dovecot IMAP/POP3 email server  
   Loaded: loaded (/usr/lib/systemd/system/dovecot.service; enabled; vendor pres  
et: disabled)  
   Active: active (running) since Mon 2023-12-04 21:04:43 PST; 13s ago  
     Docs: man:dovecot(1)
```

- We can verify Dovecot's listening ports as well:

- o At terminal run the netstat command:

- **sudo netstat -tuln | grep -E "143|993"**

```
Dec 04 21:04:43 mail.mydomain2.tsp systemd[1]: Starting Dovecot IMAP/POP3 em...  
Dec 04 21:04:43 mail.mydomain2.tsp systemd[1]: Can't open PID file /var/run/...  
Dec 04 21:04:43 mail.mydomain2.tsp dovecot[3882]: master: Dovecot v2.2.36 (1f...  
Dec 04 21:04:43 mail.mydomain2.tsp systemd[1]: Started Dovecot IMAP/POP3 ema...  
Hint: Some lines were ellipsized, use -l to show in full.  
[guest2@mail ~]$ sudo netstat -tuln | grep -E "143|993"  
tcp        0      0 0.0.0.0:143          0.0.0.0:*           LISTEN  
tcp        0      0 0.0.0.0:993          0.0.0.0:*           LISTEN  
tcp6       0      0 :::143               :::*                 LISTEN  
tcp6       0      0 :::993               :::*                 LISTEN  
[guest2@mail ~]$
```

- The command will display a list of open ports, and if Dovecot is running and configured properly, you should see an entry for port 143(IMAP) or 993(IMAPS) indicating that Dovecot is listening for connections on these ports
- **Port 143** is used for unencrypted communication between email clients and servers
- **Port 993** is used for encrypted communication between email clients and servers

```
[guest2@mail ~]$ sudo netstat -tuln | grep -E "110"  
tcp        0      0 0.0.0.0:110          0.0.0.0:*           LISTEN  
tcp6       0      0 :::110               :::*                 LISTEN  
[guest2@mail ~]$
```

- **sudo netstat -tuln | grep -E "110"**
  - Checks for open network connections or listening ports related to the POP3 service

# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

## Testing Connectivity using Telnet:

- **"telnet 192.168.2.3 143"** - This is the actual IP address of VM2 mail server running Dovecot, if Dovecot is properly configured and running, you should see a response indicating that it's ready to accept IMAP connections:

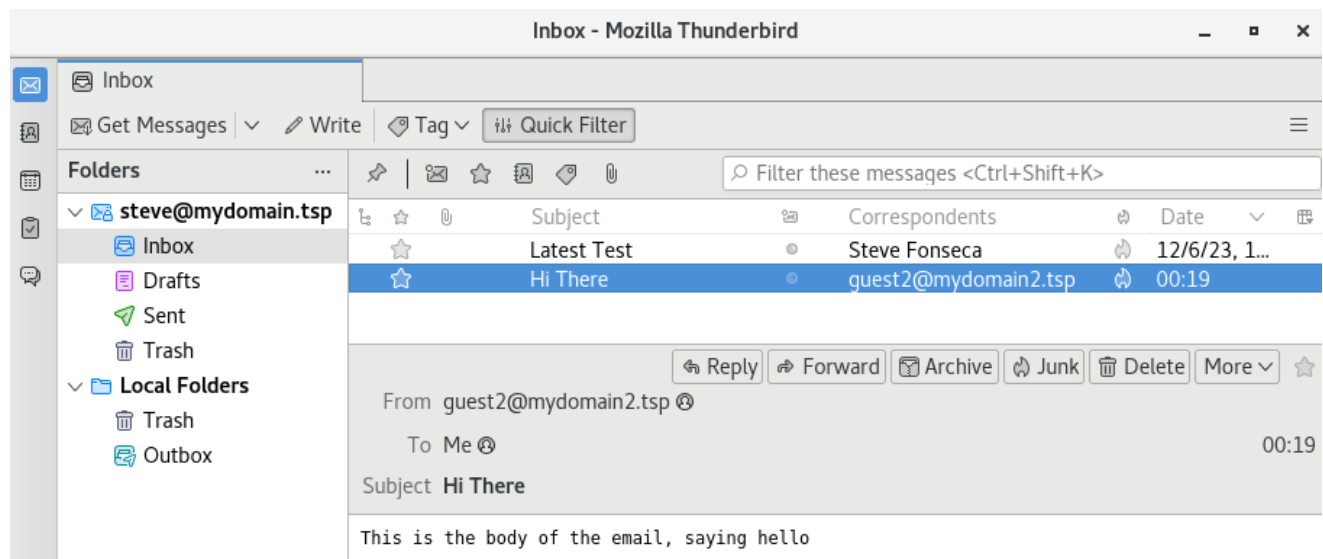
```
[guest2@mail ~]$ telnet 192.168.2.3 143
Trying 192.168.2.3...
Connected to 192.168.2.3.
Escape character is '^'.
* OK [CAPABILITY IMAP4rev1 LITERAL+ SASL-IR LOGIN-REFERRALS ID ENABLE IDLE
TH=PLAIN] Dovecot ready.
```

## Command-Line Testing:

- The command **echo "body of email" | mail -s "subject of email"** is used in Unix-based systems to send an email via the command line using the mail command. This one-liner sends an email with the specified subject and body text

```
[guest2@mail ~]$ echo "This is the body of the email, saying hello" | mail -s "Hi There" steve@mydomain.tsp
[guest2@mail ~]$
```

- Since this was sent from VM2 to VM1 (**mydomain2.tsp to mydomain.tsp**) using webmail the email was successfully received:



# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

## Testing Dovecot for POP3:

- At terminal prompt run command: **"telnet mail pop3"**
  - o This command establishes a Telnet connection to the POP3 (Post Office Protocol Version 3) server running on your local machine on the default port (port 110)
- **"user guest2"**
- **"pass nicole"**
- **"retr 8"**
  - o This command requests the 8<sup>th</sup> message in the mailbox. The POP3 server will respond with the contents of the first message:
- **"quit"**
  - o This logs you out and will read: Connection closed by foreign host

```
guest2@mail:~  
File Edit View Search Terminal Help  
[guest2@mail ~]$ telnet mail pop3  
Trying 192.168.2.3...  
Connected to mail.  
Escape character is '^]'.  
+OK Dovecot ready.  
user guest2  
+OK  
pass nicole  
+OK Logged in.  
retr 8  
+OK 818 octets  
Return-Path: <steve@mydomain.tsp>  
X-Original-To: guest2@mydomain2.tsp  
Delivered-To: guest2@mydomain2.tsp  
Received: from mail.mydomain.tsp (mail.mydomain.tsp [192.168.2.2])  
by mail.mydomain2.tsp (Postfix) with ESMTP id ABAEB212BE34  
for <guest2@mydomain2.tsp>; Mon, 4 Dec 2023 12:34:40 -0800 (PST)  
Received: by mail.mydomain.tsp (Postfix, from userid 1000)  
id A8708223C0FA; Mon, 4 Dec 2023 12:34:40 -0800 (PST)  
Date: Mon, 04 Dec 2023 12:34:40 -0800  
To: guest2@mydomain2.tsp  
Subject: Testing to Guest2  
User-Agent: Heirloom mailx 12.5 7/5/10  
MIME-Version: 1.0  
Content-Type: text/plain; charset=us-ascii  
Content-Transfer-Encoding: 7bit  
Message-Id: <20231204203440.A8708223C0FA@mail.mydomain.tsp>  
From: steve@mydomain.tsp (Steve Fonseca)  
  
This is another test to see if this is functioning correctly  
.
```

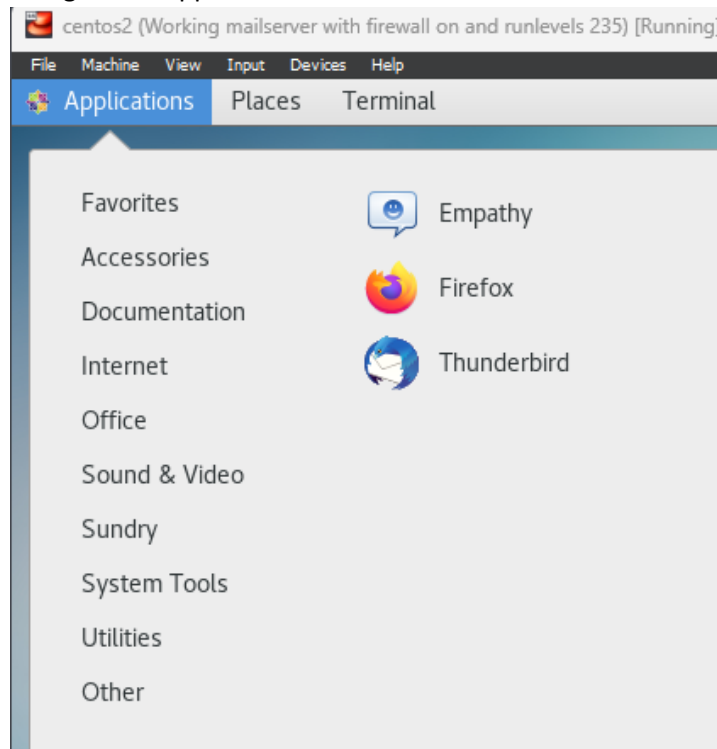
# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

Testing Dovecot for POP3 using MUA (Mail User Agent):

- Open your preferred email client
- In my case, I chose Thunderbird
  - o To download via terminal, run command:
    - **“sudo yum install thunderbird -y”**
    - Enter password when prompted

```
[guest2@mail ~]$ sudo yum install thunderbird -y
[sudo] password for guest2:
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
epel/x86_64/metalink | 27 kB 00:00
* base: lesnet.mm.fcix.net
* epel: lesnet.mm.fcix.net
* extras: lesnet.mm.fcix.net
* updates: lesnet.mm.fcix.net
base | 3.6 kB 00:00
extras | 2.9 kB 00:00
updates | 2.9 kB 00:00
Package thunderbird-102.15.0-1.el7.centos.x86_64 already installed and latest versio
n
Nothing to do
[guest2@mail ~]$
```

- Navigate to: Applications > Internet



# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

- Add a new Email account:
  - o Go to the email client's settings or account setup section

**Set Up Your Existing Email Address**

To use your current email address fill in your credentials.  
Thunderbird will automatically search for a working and recommended server configuration.

Your full name  
Steve Fonseca

Email address  
steve@mydomain2.tsp

Password  
•••••

☒ Remember password

[Configure manually](#) [Cancel](#) [Continue](#)

Your credentials will only be stored locally on your computer.

- Add a new email account and choose the option to set it up manually or using POP3 settings:

✓ Configuration found by trying common server names.

**Available configurations**

☐ IMAP  
Keep your folders and emails synced on your server

☒ POP3  
Keep your folders and emails on your computer

☒ Incoming POP3 STARTTLS  
mail.mydomain2.tsp

☒ Outgoing SMTP NO ENCRYPTION  
mail.mydomain2.tsp

☒ Username  
steve

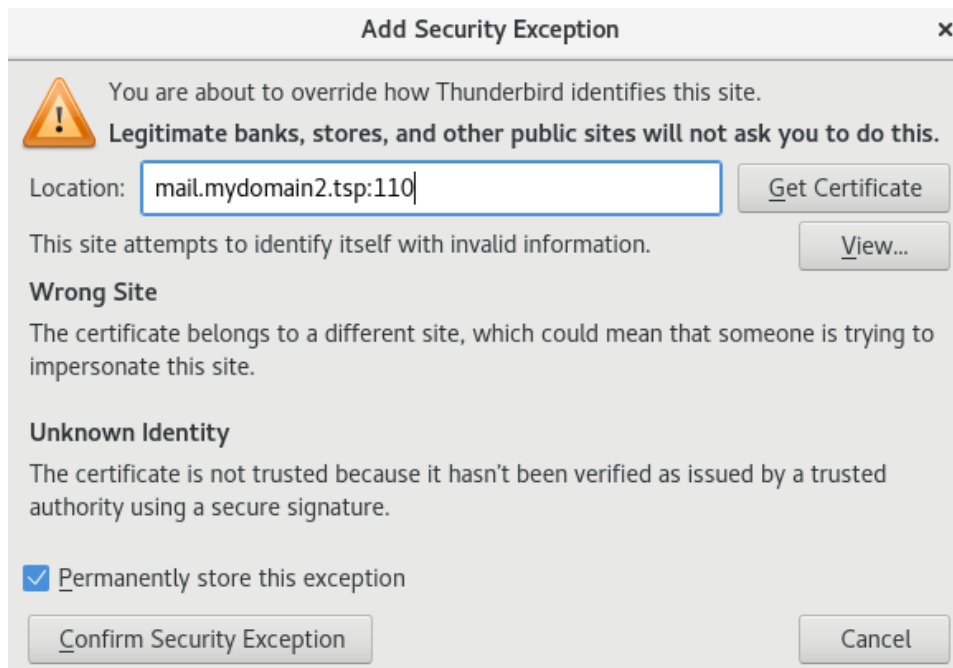
[Configure manually](#) [Cancel](#) [Done](#)

## LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

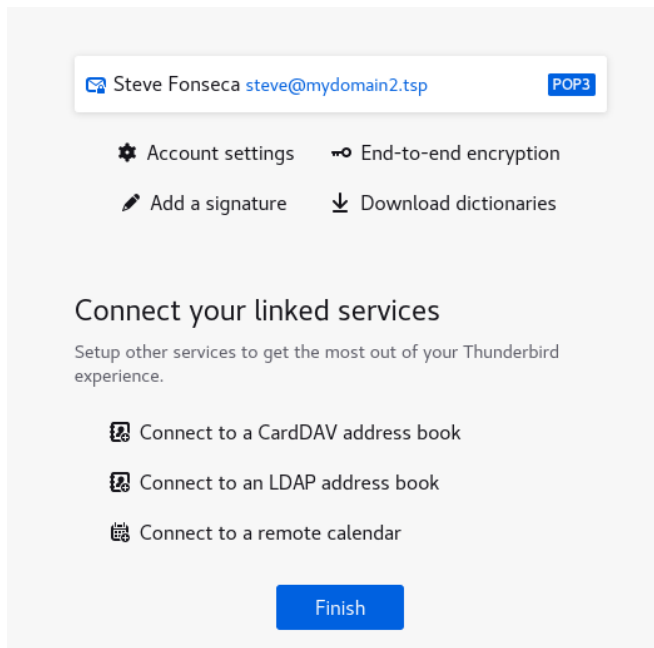
- You'll be prompted to enter POP3 server settings, which typically include:
  - o Incoming Mail Server (POP3): Provide the hostname or IP address of the POP3 server



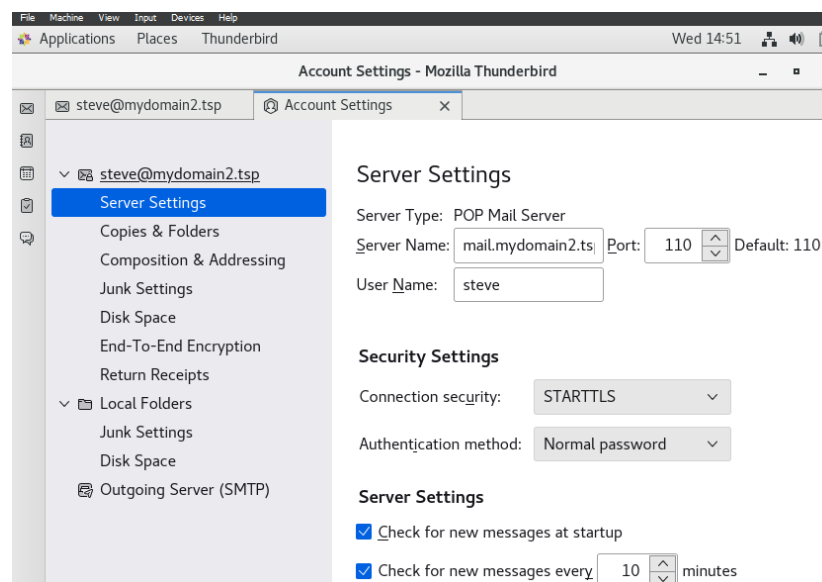
- Checkmark '*I understand the risks*'



# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX



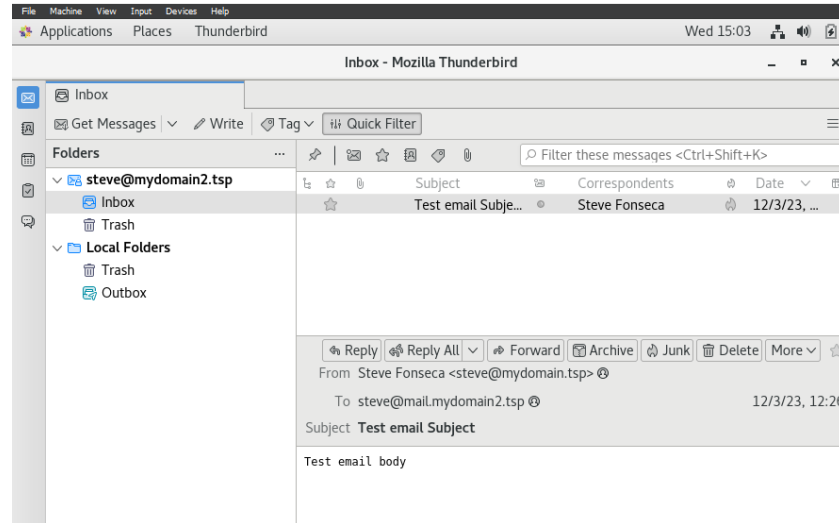
- Account has been successful created, click 'Finish' to return to inbox
- Choose Connection Type:
  - o **STARTTLS** is a command used with various email protocols, including POP3, to enable encryption during an active session
  - o When using **STARTTLS**, the **POP3** session initially starts in plain text on the standard port (110). However, after the client issues the STARTTLS command, the connection is upgraded to a secure, encrypted connection:





## LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

- Test the connection:
  - o Once you've entered the required settings, try to complete the setup. The email client will attempt to connect to the POP3 server using the provided settings
- Retrieve Emails:
  - o After successfully setting up the account, the email client should connect to the POP3 server and retrieve emails from the specified mailbox



### Testing Dovecot for IMAP:

- Use Telnet to connect to the IMAP port (default port is 143 for unencrypted IMAP or 993 for IMAPS - IMAP over SSL/TLS)
- You can test using the actual IP address or hostname of your Dovecot server:
- At terminal run command:
  - o **"telnet mail.mydomain.tsp 143"**
- Handshake with Dovecot:
  - o **"01 LOGIN username password"**

```
steve@mail:~  
File Edit View Search Terminal Help  
[steve@mail ~]$ telnet mail.mydomain.tsp 143  
Trying 192.168.2.2...  
Connected to mail.mydomain.tsp.  
Escape character is '^]'.  
* OK [CAPABILITY IMAP4rev1 LITERAL+ SASL-IR LOGIN-REFERRALS ID ENABLE IDLE STARTTLS AUTH=PLAIN AUTH=LOGIN] Dovecot ready.  
01 LOGIN steve nicole  
01 OK [CAPABILITY IMAP4rev1 LITERAL+ SASL-IR LOGIN-REFERRALS ID ENABLE IDLE SORT SORT=DISPLAY THREAD=REFERENCES THREAD=REFS THREAD=ORDEREDSUBJECT MULTIAPPEND URL-PARTIAL CATENATE UNSELECT CHILDREN NAMESPACE UIDPLUS LIST-EXTENDED I18NLEVEL=1 CONDSTORE QRESYNC ESEARCH ESORT SEARCHRES WITHIN CONTEXT=SEARCH LIST-STATUS BINARY MOVE SNIPPET=FUZZY SPECIAL-USE] Logged in
```

## LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

- Issue IMAP commands to interact with the server:

- o "02 LIST "" \*"
02 LIST "" \*
\* LIST (\HasNoChildren \Sent) "." Sent
\* LIST (\HasNoChildren \Trash) "." Trash
\* LIST (\HasNoChildren \Drafts) "." Drafts
\* LIST (\HasNoChildren) "." INBOX
02 OK List completed (0.001 + 0.000 secs).
o "03 SELECT INBOX"
o "01 FETCH 1:\* (BODY[HEADER.FIELDS (From To Subject Date)])"

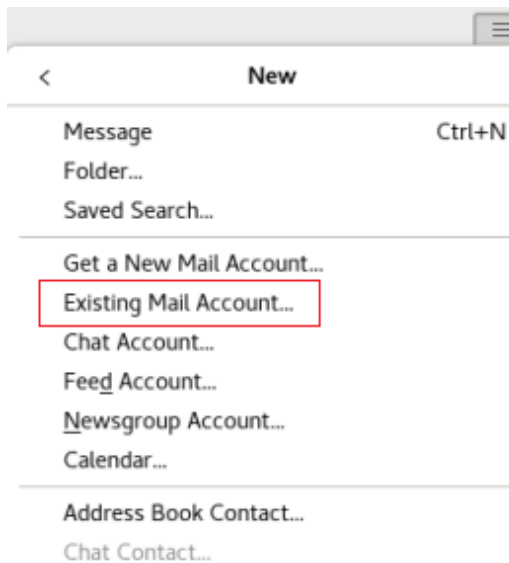
03 SELECT INBOX
\* FLAGS (\Answered \Flagged \Deleted \Seen \Draft Old)
\* OK [PERMANENTFLAGS (\Answered \Flagged \Deleted \Seen \Draft Old \*)] Flags permitted.
\* 1 EXISTS
\* 0 RECENT
\* OK [UIDVALIDITY 1701831842] UIDs valid
\* OK [UIDNEXT 12] Predicted next UID
03 OK [READ-WRITE] Select completed (0.001 + 0.000 secs).
01 FETCH 1:\* (BODY[HEADER.FIELDS (From To Subject Date)])
\* 1 FETCH (BODY[HEADER.FIELDS (FROM TO SUBJECT DATE)] {129}
Date: Wed, 6 Dec 2023 15:22:29 -0800
To: steve@mydomain.tsp
From: Steve Fonseca <steve@mydomain2.tsp>
Subject: Latest Test
)
01 OK Fetch completed (0.001 + 0.000 secs).

- This basic Telnet test allows you to simulate an IMAP session with Dovecot
- To exit: enter **CTRL-]**
- You'll see the **telnet>** prompt. Type **Quit** to close the connection

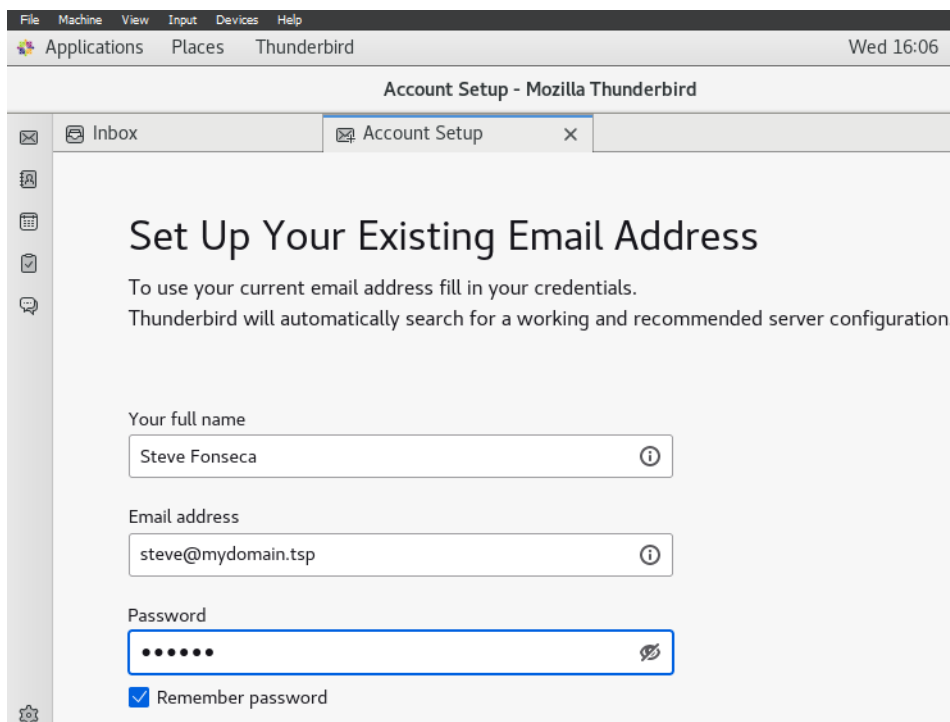
### Testing Dovecot for IMAP using MUA (Mail User Agent):

- Open your preferred email client
- In my case, I chose Thunderbird
  - o To download via terminal, run command if it's not already installed:
    - "sudo yum install thunderbird -y"
- Navigate to: *Applications > Internet > Thunderbird*
- Configure an Email Account:
  - o Open Thunderbird and go to *File > New > Existing Mail Account*:

# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

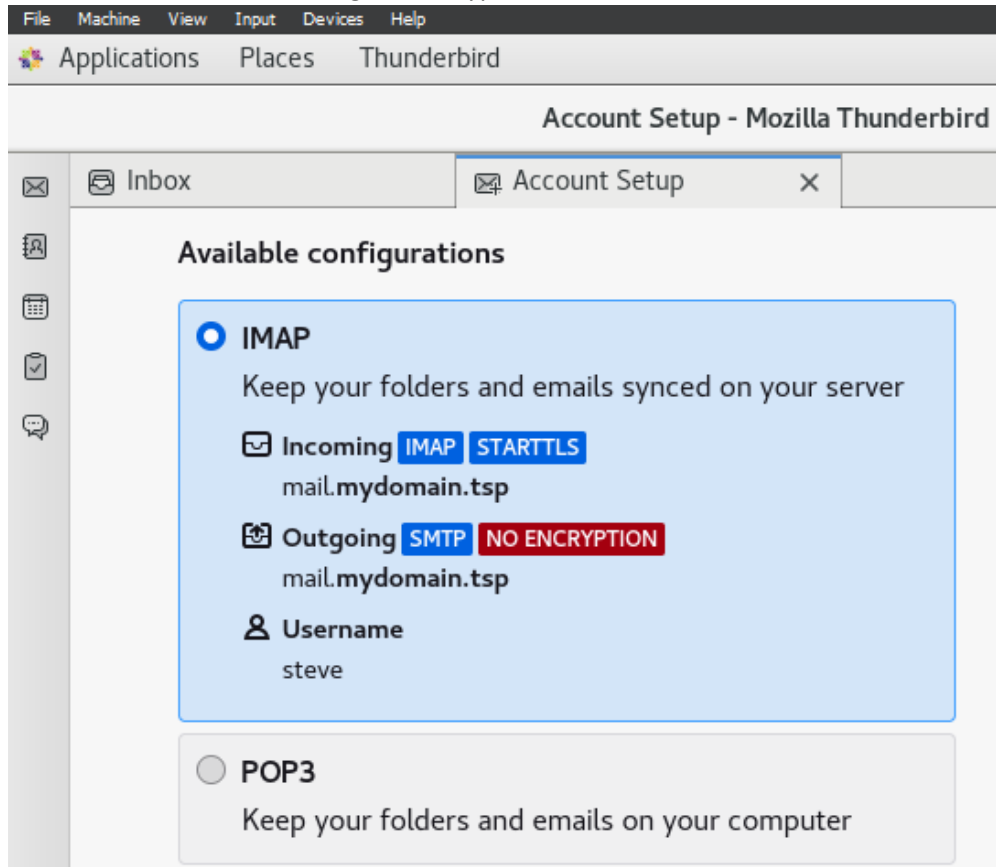


- Enter User Details:
  - o Enter the user details for the account you want to test. This includes:
    - *Your name*
    - *Email address*
    - *Password*




# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

- Choose IMAP as the Incoming Server Type:



- Checkmark 'I understand the risks'

 **Warning!**

**Outgoing settings:**

**mail.mydomain.tsp** does not use encryption.

Insecure mail servers do not use encrypted connections to protect your passwords and private information. By connecting to this server you could expose your password and private information.

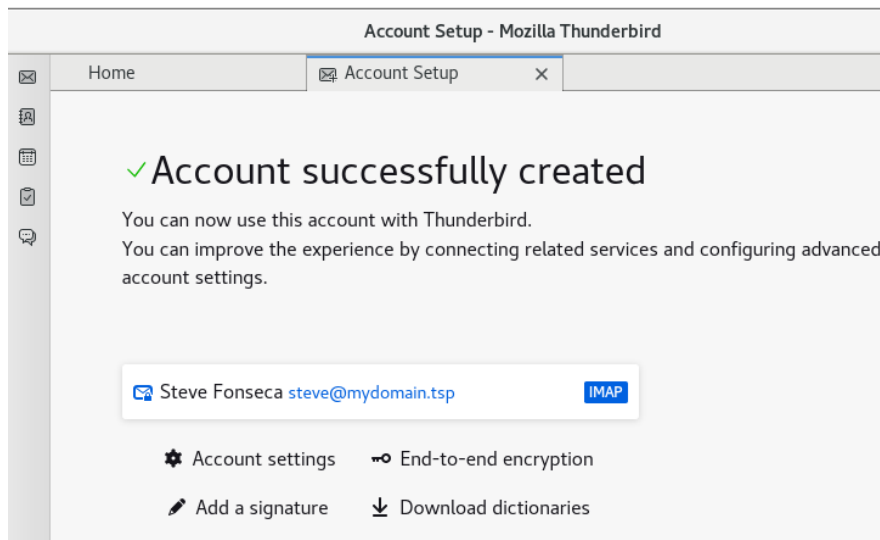
Thunderbird can allow you to get to your mail using the provided configurations. However, you should contact your administrator or email provider regarding these improper connections. See the [Thunderbird FAQ](#) for more information.

☒ I understand the risks

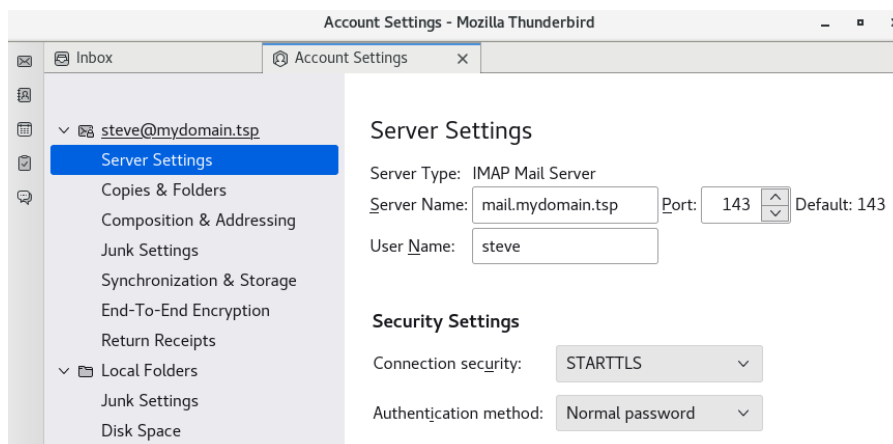
Change Settings

Confirm

# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

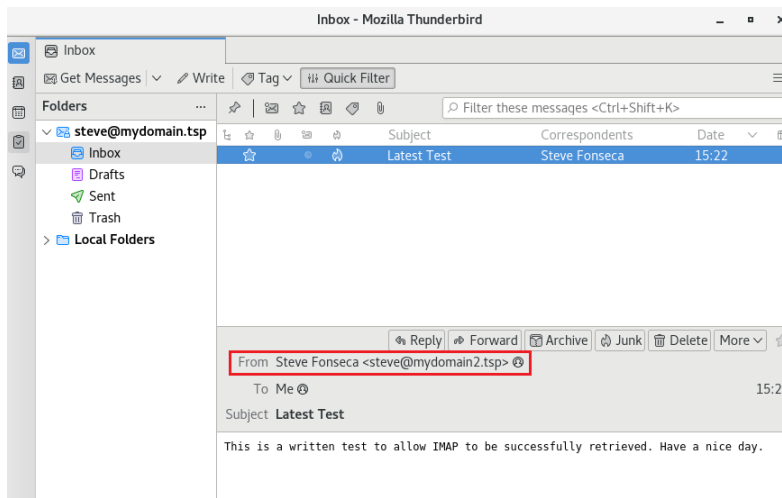


- Click 'Finish' to return to inbox
- Next, we'll look at the Server Settings page
  - o When a client initially connects to an **IMAP** server, the communication typically starts in plain text over the default IMAP port (usually **143** for unencrypted IMAP or **993** for IMAPS - IMAP over SSL/TLS)
  - o We can toggle the Connection security type with the menu option:
  - o For our purposes we'll keep STARTTLS
    - This command tells the server to switch the current connection to a secure encrypted one.

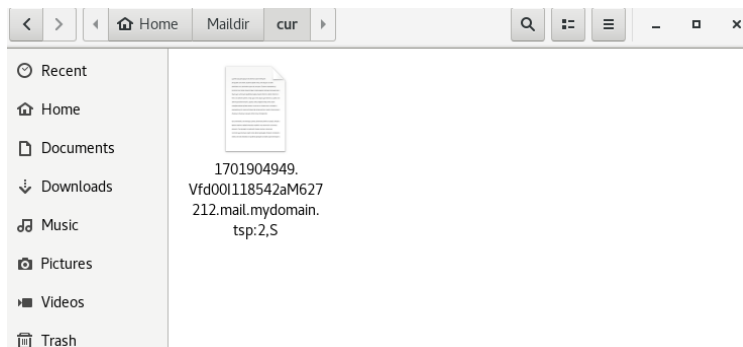


- Test the Connection:
  - o Thunderbird will attempt to connect to the Dovecot server and synchronize folders. You should be able to see your email folders and messages if the connection is successful.
  - o Send a test email to the configured email address to verify if Thunderbird retrieves the email via IMAP

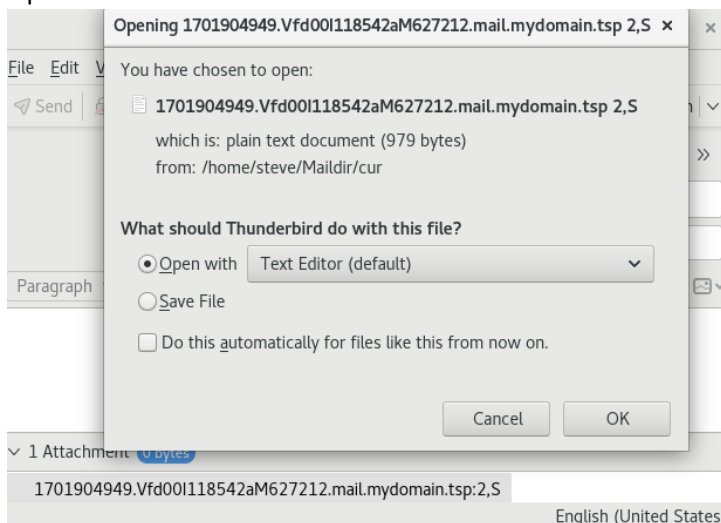
# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX



- It appears that an email from VM2 was successfully transmitted to VM1 in Thunderbird
- We can also verify that the message was delivered to the Maildir by navigating to:
  - o *Home > Maildir > cur > Latest File*

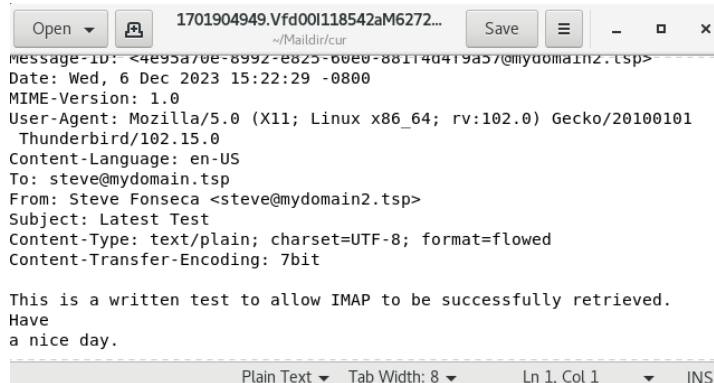


- Open the attachment:



# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

- Read the .txt file of the received email:



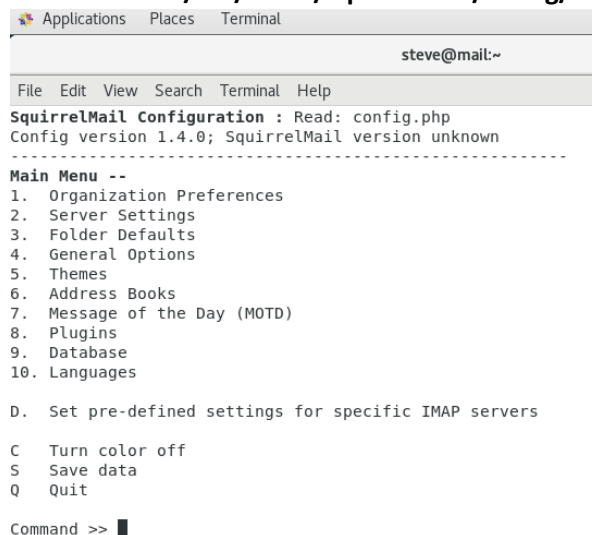
## Configuring Webmail:

### Installing SquirrelMail:

- **SquirrelMail** provides a web-based interface that allows users to access their emails through a browser, providing an alternative to desktop-based email clients
- SquirrelMail package is not available in the official CentOS 7 repositories
- Some web sources determined we must enable EPEL repository with the following command:
  - o "yum -y install epel-release"
- Install SquirrelMail:
  - o "sudo yum install squirrelmail"

### Configuring SquirrelMail:

- Open Terminal and run command:
  - o "sudo /usr/share/squirrelmail/config/conf.pl"



## LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

- At the menu options screen, select (2) for server settings
  - o Change **Domain** to your own domain which in this case is:
    - **mydomain.tsp**
- Setting the domain in SquirrelMail configuration allows the webmail interface to work with emails associated with a specific domain
- Next, we will select (3) for Sendmail or SMTP
  - o We will want to select option 2 for **SMTP**
- SMTP is a standard protocol used for sending emails across networks
- This will ensure emails are routed through the appropriate SMTP server for proper delivery

```
steve@mail:~
File Edit View Search Terminal Help
SquirrelMail Configuration : Read: config.php
Config version 1.4.0; SquirrelMail version unknown
-----
Server Settings

General
-----
1. Domain : mydomain.tsp
2. Invert Time : false
3. Sendmail or SMTP : SMTP

A. Update IMAP Settings : localhost:143 (uw)
B. Update SMTP Settings : localhost:25

R Return to Main Menu
C Turn color off
S Save data
Q Quit

Command >> █
```

- Press **S** to Save Data
- **Q** to Quit

### Installing Apache:

- To install **Apache**:
  - o When a user accesses SquirrelMail through a web browser, Apache acts as the web server interacting with the SquirrelMail web interface, allowing users to log in and access their emails via a web-based interface
  - o “**sudo yum install httpd**”



## LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

```
[steve@mail ~]$ sudo yum install httpd
[sudo] password for steve:
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
epel/x86_64/metalink | 26 kB 00:00:00
* base: lesnet.mm.fcix.net
* epel: ziplly.mm.fcix.net
* extras: lesnet.mm.fcix.net
* updates: lesnet.mm.fcix.net
base | 3.6 kB 00:00:00
extras | 2.9 kB 00:00:00
updates | 2.9 kB 00:00:00
Package httpd-2.4.6-99.el7.centos.1.x86_64 already installed and latest version
Nothing to do
```

- Next the following commands will start and enable Apache to start on boot:
  - o **"sudo systemctl start httpd"**
  - o **"sudo systemctl enable httpd"**
- To check if Apache is running on CentOS, the output should indicate if the service is active and running: **"sudo systemctl status httpd"**

```
[steve@mail ~]$ sudo yum install httpd
[sudo] password for steve:
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
epel/x86_64/metalink | 26 kB 00:00:00
* base: lesnet.mm.fcix.net
* epel: ziplly.mm.fcix.net
* extras: lesnet.mm.fcix.net
* updates: lesnet.mm.fcix.net
base | 3.6 kB 00:00:00
extras | 2.9 kB 00:00:00
updates | 2.9 kB 00:00:00
Package httpd-2.4.6-99.el7.centos.1.x86_64 already installed and latest version
Nothing to do
[steve@mail ~]$ sudo systemctl start httpd
[steve@mail ~]$ sudo systemctl enable httpd
Created symlink from /etc/systemd/system/multi-user.target.wants/httpd.service to /usr/lib/systemd/system/httpd.service.
[steve@mail ~]$ sudo system status httpd
sudo: system: command not found
[steve@mail ~]$ sudo systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; vendor preset: disabled)
   Active: active (running) since Tue 2023-12-05 20:09:06 PST; 23h ago
     Docs: man:httpd(8)
           man:apachectl(8)
   Main PID: 10456 (httpd)
   Status: "Total requests: 309; Current requests/sec: 0.1; Current traffic: 409 B/sec"
```

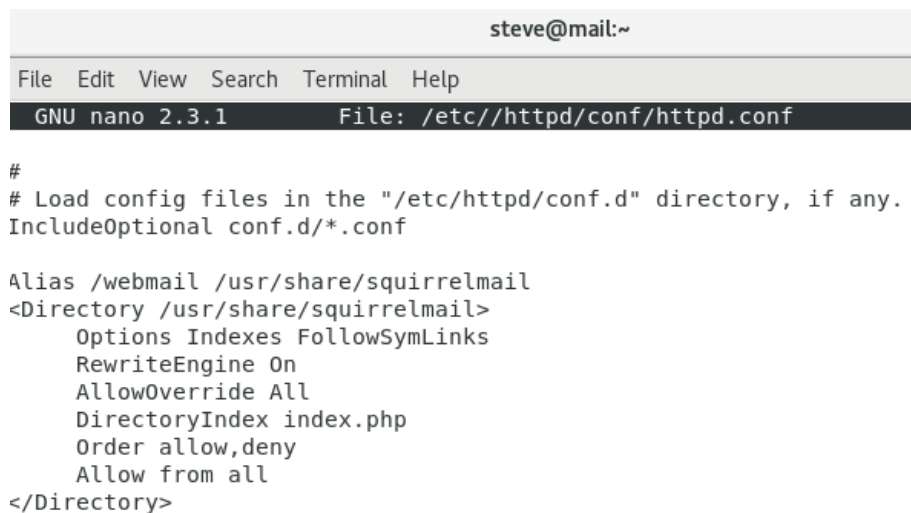
- To configure Firewall to allow HTTP traffic we can make a rule:

# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

- **"sudo firewall-cmd --permanent --add-service=http"**
- **"sudo firewall-cmd --reload"**

## Configuring Apache:

- Enter the configuration file with the following command:
  - **"sudo nano /etc/httpd/conf/httpd.conf"**
    - Enter password when prompted
- Input this line of code at the end of the configuration file and apply:
  - **Alias /webmail /usr/share/squirrelmail**  
**<Directory /usr/share/squirrelmail>**  
**Options Indexes FollowSymLinks**  
**RewriteEngine On**  
**AllowOverride All**  
**DirectoryIndex index.php**  
**Order allow,deny**  
**Allow from all**  
**</Directory>**



```
steve@mail:~
File Edit View Search Terminal Help
GNU nano 2.3.1 File: /etc/httpd/conf/httpd.conf

#
# Load config files in the "/etc/httpd/conf.d" directory, if any.
IncludeOptional conf.d/*.conf

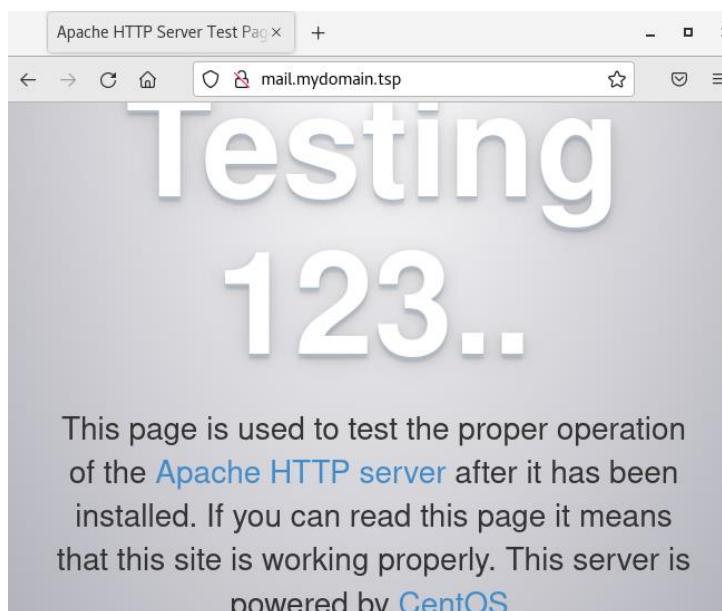
Alias /webmail /usr/share/squirrelmail
<Directory /usr/share/squirrelmail>
    Options Indexes FollowSymLinks
    RewriteEngine On
    AllowOverride All
    DirectoryIndex index.php
    Order allow,deny
    Allow from all
</Directory>
```

- Restart Apache: **"sudo systemctl restart httpd"**
- The code block provided is an Apache configuration snippet that creates an alias and sets up a directory configuration for serving SquirrelMail via the web server.
- **Alias /webmail /usr/share/squirrelmail:**
  - This directive creates an alias in Apache configuration. It maps the **URL path /webmail** to the physical directory **/usr/share/squirrelmail**. So, when a user accesses'

## LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

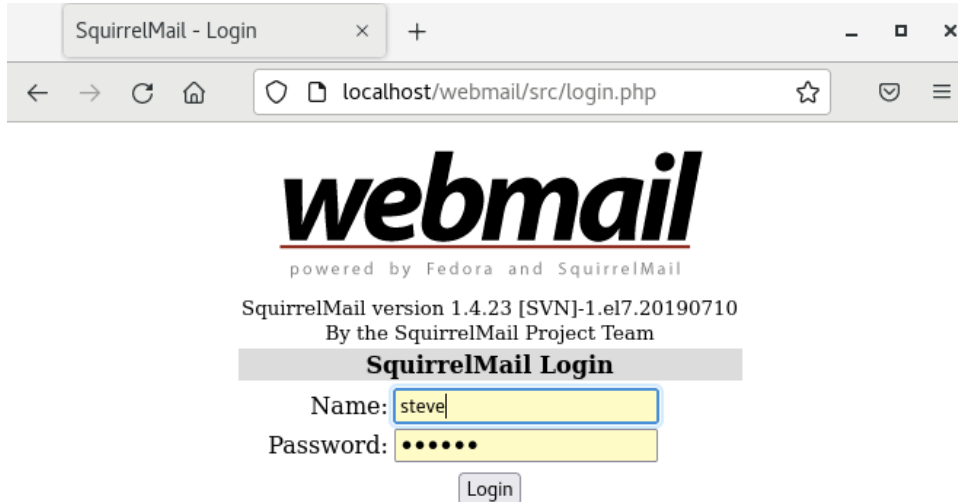
**`http://your_server/webmail`**, Apache will serve content **from `/usr/share/squirrelmail`** directory

- **<Directory>**:
  - o This block specifies settings that apply to the directory `/usr/share/squirrelmail` and its subdirectories
- **Options Indexes FollowSymLinks**:
  - o Specifies which server features are available in that directory. Indexes enables directory indexing, and FollowSymLinks allows Apache to follow symbolic links
- **RewriteEngine On**:
  - o Enables Apache's URL rewriting engine, which allows for flexible URL manipulation using rewrite rules
- **AllowOverride All**:
  - o Permits the use of `.htaccess` files in that directory to override Apache configuration directives
- **DirectoryIndex index.php**:
  - o Sets the default index file to `index.php` within the directory if not specified in the URL.
- **Order allow,deny and Allow from all**:
  - o Specifies access control settings. Order allow,deny sets the order in which Apache evaluates access control directives, and Allow from all allows unrestricted access to the directory from all IP addresses
- To Test the functionality of the Apache HTTP server, in a browser type the address:
  - o <http://mail.mydomain.tsp>

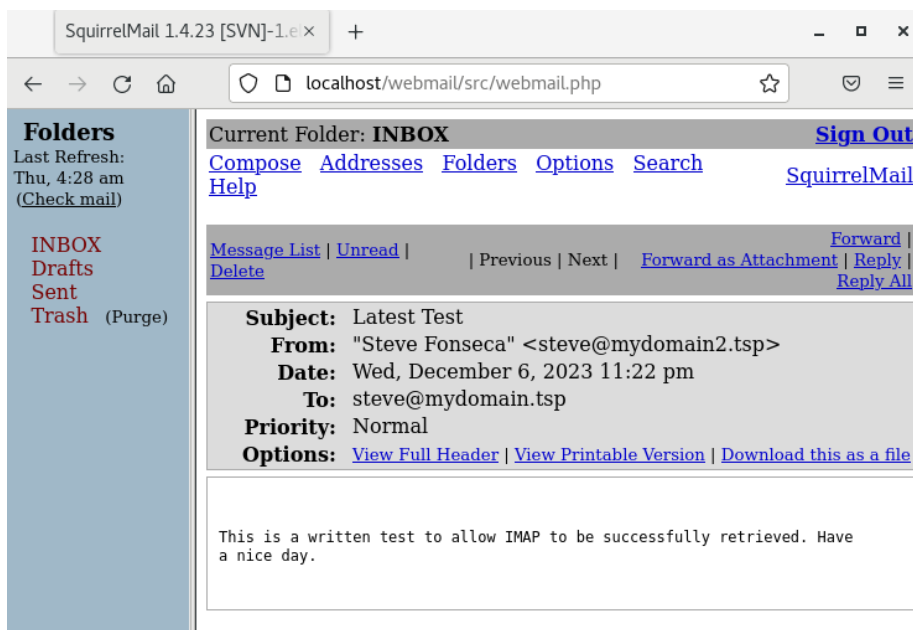


## LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

- This configuration makes it easier for users to access SquirrelMail by providing a cleaner URL: (<http://localhost/webmail>)

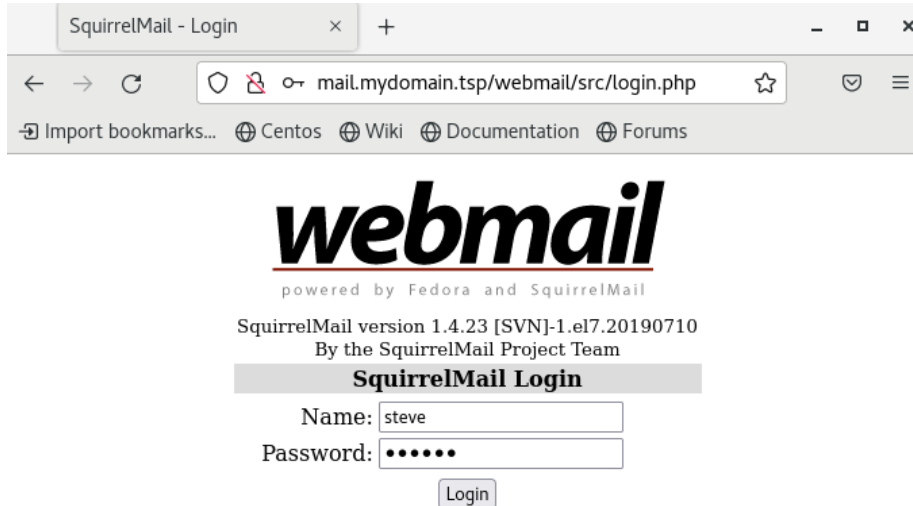


- Log in using the credentials of a user who has access to the configured IMAP server. SquirrelMail should display a web-based interface to access and manage emails

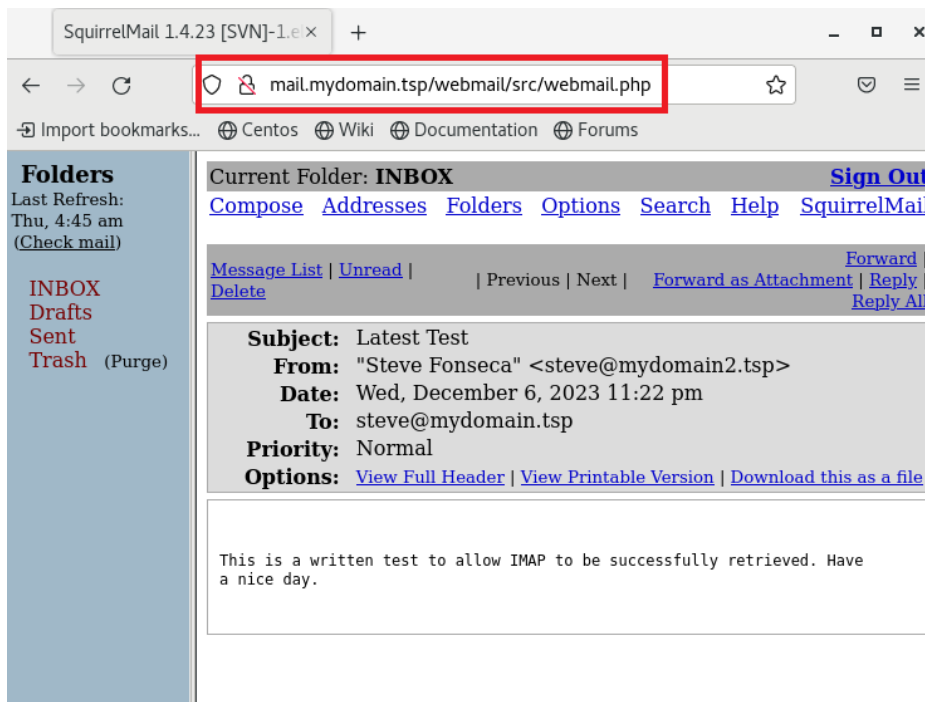


# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

- Alternatively, you can also log in with: (<http://your.fully.qualified.domain/webmail>)
  - o (<http://mail.mydomain.tsp/webmail>)



- There will appear a prompt to save the login for mydomain.tsp
- We can log in and view our inbox with the convenience of webmail



# LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

## References:

- <https://www.pfsense.org/download/>
- <https://www.pfsense.org/download/>
- <https://blog.paranoidpenguin.net/2017/07/pfsense-how-to-add-a-mx-record-to-a-local-zone/>
- <https://askubuntu.com/questions/169024/how-can-i-tell-if-the-virtualbox-guest-additions-were-installed-on-an-ubuntu-vm>
- <https://superuser.com/questions/878135/what-is-the-difference-between-centos-dvd-vs-everything-isos>
- <https://superuser.com/questions/878135/what-is-the-difference-between-centos-dvd-vs-everything-isos>
- <https://www.centos.org/centos-linux/>
- <https://www.cyberciti.biz/faq/howto-linux-renew-dhcp-client-ip-address/>
- <https://redmine.pfsense.org/issues/14137>
- [https://docs.gitlab.com/ee/administration/reply\\_by\\_email\\_postfix\\_setup.html](https://docs.gitlab.com/ee/administration/reply_by_email_postfix_setup.html)
- <https://www.linkedin.com/learning/linux-system-engineer-mail-systems-using-postfix/send-email-from-a-remote-client?autoSkip=true&resume=false&u=2097252>
- [https://medium.com/@pepi\\_post/how-to-install-postfix-on-centos-7-32ecd94caf8c](https://medium.com/@pepi_post/how-to-install-postfix-on-centos-7-32ecd94caf8c)
- <https://www.rosehosting.com/blog/how-to-install-squirrelmail-on-centos-7/>
- <https://docs.rackspace.com/docs/dovecot-installation-and-configuration-on-centos>
- ChatGPT
- <https://gist.github.com/daubac402/8f8473ab531bc988970e2788a71e8eea>
- <https://www.linuxpathfinder.com/install-and-configure-mail-server-using-postfix-dovecot-and-squirrelmail-centos-6>

## \*\*Project Afterthoughts\*\*

- When creating the environment, I ran into a connectivity issue, but it was due to the LAN interface not properly configured to accept a scope of IPs for the DNS server to deliver, this was solved by configuring the LAN via the pfSense console with option 2, statically set with DHCP enabled
- Postfix had a problem with authenticating and the culprit was “mynetworks” in terms of syntax it was fine, but the address was not, it was pointing to one of my local VM IP’s the subnet was wrong, and I had to change the last octet to zero 192.168.2.0 to encompass the addresses previously set by me in the created environment
- Lastly, DNS Resolver was a challenge. I went through a few iterations of how to write in the custom options area and I discovered I needed a functioning MX and PTR record written with the proper syntax and identifiers clearly defined for each server
- Issue sending a composed mail message in Squirrel Mail. I was able to send email finally by editing the Squirrel Mail config file. Switch from SMTP to Sendmail in:  
**/etc/squirrelmail/config.php** change **\$useSendmail = true;** then restart Apache: **systemctl restart httpd**

## LINUX PROJECT: CREATING FUNCTIONAL MAIL SERVERS IN A VIRTUAL SANDBOX

Using the checklist, click the box next to each completed task:

- Only do so when ALL tasks have been completed OR you've done all you can. **Do not submit individual tasks.**

| TASK                                                                                       | COMPLETED                           |
|--------------------------------------------------------------------------------------------|-------------------------------------|
| Create The Environment<br>(You must use <a href="#">pfsense</a> as the NAT Router/Gateway) | <input checked="" type="checkbox"/> |
| Install and Configure Postfix                                                              | <input checked="" type="checkbox"/> |
| Configure POP/IMAP                                                                         | <input checked="" type="checkbox"/> |
| Configure Webmail                                                                          | <input checked="" type="checkbox"/> |