To complete Task-01, follow these steps:

**Network Topology Setup**

1. **Install Virtualization Software:**
   * Download and install a virtualization software like VMware or VirtualBox.
2. **Create and Configure MikroTik Router VM:**
   * Create a new virtual machine for MikroTik RouterOS.
   * Install MikroTik RouterOS on this VM.
3. **MikroTik Router Configuration:**
   * **IP Information:**
     + **WAN:** Configure as per your choice in VirtualBox/VMware.
     + **LAN:** Set IP range to 10.10.10.0/24.
     + **Username:** admin
     + **Password:** cefalo
     + **DNS:** 8.8.8.8, 9.9.9.9
   * **DHCP Server:**
     + Set lease time to 2 days.
     + Configure IP Pool from 10.10.10.50 to 10.10.10.200.
   * **Webfig Port:**
     + Change the Webfig port to 8088.
   * **Admin User:**
     + Create another admin user named sysadmin with password Sysadmin.
4. **Client VM Configuration:**
   * Create a new virtual machine and install Windows 10 or 11.
   * This machine will be used for further testing.
5. **Server VM Configuration:**
   * Create a new virtual machine and install a preferred Linux distribution (Disk: 50GB, RAM: 4GB, CPU: 2 Cores).
   * **Web Server:**
     + Install and configure a web server (e.g., Apache or Nginx) to be accessible from the client machine at webserver.cefalo-bd.com.
   * **SSH Server:**
     + Enable SSH on port 2512 with root access disabled.
   * **File Transfer:**
     + Create a folder named "copy-me" on the client's desktop and transfer it to the Linux web server's root folder.
6. **Partition the Client’s C Drive:**
   * Split the C drive into two partitions.
7. **Create a Linked Clone of the Client:**
   * Create a linked clone of the client VM and name it CLIENT-CLONE.
   * Configure the clone to be accessible via Remote Desktop Protocol (RDP) on port 3339.

**Bonus Tasks**

**VM01:**

1. **Install Docker and Docker Compose:**
   * Connect to VM01 via SSH: ssh -p 23232 recruit@103.143.91.240
   * Install Docker-CE:

bash

Copy code

sudo apt-get update

sudo apt-get install -y docker-ce docker-ce-cli containerd.io

* + Install Docker Compose:

bash

Copy code

sudo curl -L "https://github.com/docker/compose/releases/download/1.29.2/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose

sudo chmod +x /usr/local/bin/docker-compose

1. **Install Web Proxy Server:**
   * Choose a proxy server like Nginx or HAProxy and install it.
   * Configure it for balancing HTTP/HTTPS requests and SSL termination.
2. **Generate Self-Signed Certificate:**
   * Generate a wildcard certificate for \*.cefalo-bd.com:

bash

Copy code

openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/ssl/private/cefalo-bd.key -out /etc/ssl/certs/cefalo-bd.crt -subj "/CN=\*.cefalo-bd.com"

1. **Run Hello-World Containers:**
   * Run the Nginx Hello-World containers:

bash

Copy code

docker run -d -p 80:80 --name hello-world nginxdemos/nginx-hello

1. **Configure Load Balancer:**
   * Configure the load balancer to handle HTTPS for recruit.cefalo-bd.com.

**VM02:**

1. **Setup and Configure Graylog:**
   * Connect to VM02 via SSH: ssh -p 23233 recruit@103.143.91.240
   * Install and configure Graylog using Docker:

bash

Copy code

docker run --name mongo -d mongo:4.2

docker run --name elasticsearch -d -e "http.host=0.0.0.0" -e "discovery.type=single-node" docker.elastic.co/elasticsearch/elasticsearch:7.10.2

docker run --name graylog -d -p 9000:9000 -p 12201:12201/udp -e "GRAYLOG\_PASSWORD\_SECRET=somepasswordpepper" -e "GRAYLOG\_ROOT\_PASSWORD\_SHA2=<SHA-256-Hash>" -e "GRAYLOG\_HTTP\_EXTERNAL\_URI=http://127.0.0.1:9000/" --link mongo --link elasticsearch graylog/graylog:3.3

1. **Collect Logs from Load Balancer:**
   * Configure load balancer to send logs to the Graylog instance.
2. **Configure Domain for Graylog:**
   * Ensure Graylog is reachable at graylog.cefalo-bd.com.

This setup will provide you with a fully functional network lab as described in the task document.

4o

You tell me the pdf file. Need solution

ChatGPT

To complete the tasks outlined in the PDF, follow these detailed steps:

**Task-01**

**1. Install Virtualization Software**

* Install VMware or VirtualBox on your host machine.

**2. Create a VM and Install MikroTik RouterOS**

* **Create a VM for MikroTik RouterOS:**
  + Open your virtualization software (VMware or VirtualBox).
  + Create a new VM with appropriate resources (1 CPU, 512MB RAM, 1GB HDD).
  + Attach the MikroTik RouterOS ISO image to the VM.
  + Install MikroTik RouterOS following the on-screen instructions

4o

Stop

ChatGPT

Let's focus on providing the solution directly based on the steps provided in the PDF.

**Task-01**

**Step 1: Install Virtualization Software**

* **Virtualization Software:** Install either VMware or VirtualBox.

**Step 2: Create and Configure MikroTik Router VM**

1. **Create VM:**
   * In VirtualBox/VMware, create a new VM with suitable specifications.
   * Install MikroTik RouterOS.
2. **Configure MikroTik Router:**
   * **IP Information:**
     + **WAN:** Configure as desired.
     + **LAN:** 10.10.10.0/24
     + **Username:** admin
     + **Password:** cefalo
     + **DNS Servers:** 8.8.8.8, 9.9.9.9
   * **DHCP Server:**
     + **Lease Time:** 2 days
     + **IP Pool:** 10.10.10.50 to 10.10.10.200
   * **Webfig Port:** Change to 8088
   * **Create Admin User:**
     + Username: sysadmin

4o

Tell me the total mikrotik configuration which they asked in the pdf

ChatGPT

Here is the complete MikroTik configuration as per the requirements outlined in the PDF:

**MikroTik Configuration**

**1. IP Information**

* **WAN:** Configure as per your choice in VirtualBox/VMware.
* **LAN Network:** 10.10.10.0/24
* **Username:** admin
* **Password:** cefalo
* **DNS Servers:** 8.8.8.8, 9.9.9.9

**2. DHCP Server Configuration**

* **Lease Time:** 2 days
* **IP Pool:** 10.10.10.50 to 10.10.10.200

**3. Webfig Port Change**

* **New Port:** 8088

**4. Create Another Admin User**

* **Username:** sysadmin
* **Password:** Sysadmin

**Steps to Configure MikroTik RouterOS**

1. **Login to MikroTik RouterOS:**
   * Open WinBox or connect via SSH to your MikroTik Router.
2. **Set IP Address on LAN Interface:**

shell

Copy code

/ip address add address=10.10.10.1/24 interface=ether2

1. **Set DNS Servers:**

shell

Copy code

/ip dns set servers=8.8.8.8,9.9.9.9

1. **Configure DHCP Server:**
   * Create a DHCP Pool:

shell

Copy code

/ip pool add name=dhcp\_pool ranges=10.10.10.50-10.10.10.200

* + Create a DHCP Network:

shell

Copy code

/ip dhcp-server network add address=10.10.10.0/24 gateway=10.10.10.1 dns-server=8.8.8.8,9.9.9.9

* + Add a DHCP Server:

shell

Copy code

/ip dhcp-server add address-pool=dhcp\_pool interface=ether2 lease-time=2d name=dhcp1

1. **Change Webfig Port:**

shell

Copy code

/ip service set www-ssl port=8088

1. **Create Another Admin User:**

shell

Copy code

/user add name=sysadmin group=full password=Sysadmin

1. **Remove Default Admin User for Security (Optional):**

shell

Copy code

/user remove admin

**Verifying the Configuration**

1. **Check IP Configuration:**

shell

Copy code

/ip address print

1. **Check DNS Configuration:**

shell

Copy code

/ip dns print

1. **Check DHCP Server:**

shell

Copy code

/ip dhcp-server print

1. **Check Webfig Port:**

shell

Copy code

/ip service print

1. **Check User Accounts:**

shell

Copy code

/user print

This configuration ensures that your MikroTik Router is set up with the specified IP range, DHCP server, DNS settings, and user accounts, along with the modified Webfig port.

4o

Now tell the config of server which they asked

ChatGPT

Here's the detailed configuration for the Linux server as per the requirements outlined in the PDF:

**Server Configuration**

**Step 1: Create and Configure Linux Server VM**

* **Disk:** 50GB
* **RAM:** 4GB
* **CPU:** 2 Cores

**Step 2: Install a Preferred Linux Distribution**

* Install a Linux distribution (e.g., Ubuntu, CentOS, Debian).

**Step 3: Install Web Server**

* Install and configure a web server (Apache or Nginx).

**Step 4: Configure the Server to be Accessible from the Client**

* The web server should be accessible from the Windows client at webserver.cefalo-bd.com.

**Step 5: Enable SSH Server on Port 2512**

* Enable SSH on the server and configure it to use port 2512 with root access disabled.

**Step 6: Create a Folder on the Client Machine's Desktop and Transfer it to the Server**

**Steps to Configure the Linux Server**

1. **Install the Linux Distribution:**
   * Create a new VM in your virtualization software with the specified resources.
   * Install your preferred Linux distribution.
2. **Install Web Server:**
   * For Apache:

shell

Copy code

sudo apt update

sudo apt install apache2

sudo systemctl start apache2

sudo systemctl enable apache2

* + For Nginx:

shell

Copy code

sudo apt update

sudo apt install nginx

sudo systemctl start nginx

sudo systemctl enable nginx

1. **Configure Web Server:**
   * Ensure the web server is accessible at webserver.cefalo-bd.com by editing the server configuration file.
   * For Apache:

shell

Copy code

sudo nano /etc/apache2/sites-available/000-default.conf

Add the following lines:

apache

Copy code

ServerName webserver.cefalo-bd.com

Enable the site and restart Apache:

shell

Copy code

sudo a2ensite 000-default.conf

sudo systemctl reload apache2

* + For Nginx:

shell

Copy code

sudo nano /etc/nginx/sites-available/default

Modify the server block to include:

nginx

Copy code

server\_name webserver.cefalo-bd.com;

Test the configuration and reload Nginx:

shell

Copy code

sudo nginx -t

sudo systemctl reload nginx

1. **Configure SSH on Port 2512:**
   * Edit the SSH configuration file:

shell

Copy code

sudo nano /etc/ssh/sshd\_config

Change the port and disable root login:

plaintext

Copy code

Port 2512

PermitRootLogin no

Restart the SSH service:

shell

Copy code

sudo systemctl restart ssh

1. **Transfer Folder from Client to Server:**
   * On the Windows client, create a folder named copy-me on the desktop.
   * Transfer the folder to the web server's root directory (e.g., /var/www/html):
     + Use SCP from the Windows machine (use a tool like WinSCP or command line):

shell

Copy code

scp -P 2512 -r ~/Desktop/copy-me username@webserver.cefalo-bd.com:/var/www/html/

**Verifying the Configuration**

1. **Check Web Server Accessibility:**
   * From the Windows client, open a web browser and navigate to http://webserver.cefalo-bd.com.
2. **Check SSH Access:**
   * Try connecting to the server using SSH on port 2512:

shell

Copy code

ssh -p 2512 username@webserver.cefalo-bd.com

This setup ensures that your Linux server is configured with a web server, SSH on a non-default port, and the specified folder is transferred from the Windows client.

4o

Virtual switch configure in vmware workstation pro

ChatGPT

To configure a virtual switch in VMware Workstation Pro, follow these steps:

**Configuring a Virtual Switch in VMware Workstation Pro**

**Step 1: Open VMware Workstation Pro**

1. Launch VMware Workstation Pro.

**Step 2: Open Virtual Network Editor**

1. Go to Edit > Virtual Network Editor.

**Step 3: Configure a New Virtual Network**

1. **Add a New Network:**
   * Click on Add Network.
   * Select an unused network (e.g., VMnet2) and click OK.
2. **Configure the Virtual Network:**
   * Select the newly created VMnet2.
   * Choose Host-only or NAT depending on your network needs. For this lab, we will use Host-only.
   * Check the option Connect a host virtual adapter to this network to allow the host machine to communicate with the VMs.
3. **Set IP Configuration:**
   * Click on DHCP Settings if you want to configure DHCP for this network. You can disable DHCP if you prefer static IP configuration.
   * Configure the subnet IP for the virtual network. For example, set the subnet to 10.10.10.0 with a subnet mask of 255.255.255.0.

**Step 4: Connect VMs to the Virtual Network**

1. **MikroTik Router VM:**
   * Select the MikroTik Router VM in VMware Workstation.
   * Go to VM > Settings.
   * Select Network Adapter.
   * Choose Custom: Specific virtual network.
   * Select the newly created virtual network (e.g., VMnet2).
2. **Windows Client VM:**
   * Select the Windows Client VM in VMware Workstation.
   * Go to VM > Settings.
   * Select Network Adapter.
   * Choose Custom: Specific virtual network.
   * Select the same virtual network (e.g., VMnet2).
3. **Linux Server VM:**
   * Select the Linux Server VM in VMware Workstation.
   * Go to VM > Settings.
   * Select Network Adapter.
   * Choose Custom: Specific virtual network.
   * Select the same virtual network (e.g., VMnet2).

**Verification**

1. **Verify Network Connectivity:**
   * Start all the VMs.
   * Ensure that each VM has an IP address in the 10.10.10.0/24 subnet.
2. **Ping Test:**
   * Open a terminal or command prompt on each VM.
   * Ping the other VMs to ensure network connectivity.

**Example IP Configuration**

* **MikroTik Router:** 10.10.10.1 (LAN Interface)
* **Windows Client:** 10.10.10.2
* **Linux Server:** 10.10.10.3

shell

Copy code

# On Windows Client:

ping 10.10.10.1 # Should get a response from MikroTik Router

ping 10.10.10.3 # Should get a response from Linux Server

# On Linux Server:

ping 10.10.10.1 # Should get a response from MikroTik Router

ping 10.10.10.2 # Should get a response from Windows Client

By following these steps, you will have a virtual switch configured in VMware Workstation Pro, enabling network communication between your VMs.

4o

webserver configure in centos7

ChatGPT

To configure a web server on CentOS 7, you can use either Apache (httpd) or Nginx. Below are the steps to set up a web server using both options.

**Option 1: Configure Apache (httpd) on CentOS 7**

**Step 1: Update the System**

bash

Copy code

sudo yum update -y

**Step 2: Install Apache**

bash

Copy code

sudo yum install httpd -y

**Step 3: Start and Enable Apache**

bash

Copy code

sudo systemctl start httpd

sudo systemctl enable httpd

**Step 4: Adjust the Firewall to Allow HTTP and HTTPS Traffic**

bash

Copy code

sudo firewall-cmd --permanent --add-service=http

sudo firewall-cmd --permanent --add-service=https

sudo firewall-cmd --reload

**Step 5: Configure Apache**

1. **Create a Virtual Host Configuration File:**

bash

Copy code

sudo nano /etc/httpd/conf.d/webserver.conf

Add the following configuration:

apache

Copy code

<VirtualHost \*:80>

ServerAdmin webmaster@cefalo-bd.com

DocumentRoot /var/www/html

ServerName webserver.cefalo-bd.com

ErrorLog /var/log/httpd/webserver-error.log

CustomLog /var/log/httpd/webserver-access.log combined

</VirtualHost>

1. **Restart Apache:**

bash

Copy code

sudo systemctl restart httpd

**Step 6: Test the Configuration**

* Open a web browser and navigate to http://webserver.cefalo-bd.com. You should see the default Apache test page.

**Option 2: Configure Nginx on CentOS 7**

**Step 1: Update the System**

bash

Copy code

sudo yum update -y

**Step 2: Install Nginx**

bash

Copy code

sudo yum install epel-release -y

sudo yum install nginx -y

**Step 3: Start and Enable Nginx**

bash

Copy code

sudo systemctl start nginx

sudo systemctl enable nginx

**Step 4: Adjust the Firewall to Allow HTTP and HTTPS Traffic**

bash

Copy code

sudo firewall-cmd --permanent --add-service=http

sudo firewall-cmd --permanent --add-service=https

sudo firewall-cmd --reload

**Step 5: Configure Nginx**

1. **Create a Server Block Configuration File:**

bash

Copy code

sudo nano /etc/nginx/conf.d/webserver.conf

Add the following configuration:

nginx

Copy code

server {

listen 80;

server\_name webserver.cefalo-bd.com;

root /usr/share/nginx/html;

index index.html;

access\_log /var/log/nginx/webserver-access.log;

error\_log /var/log/nginx/webserver-error.log;

location / {

try\_files $uri $uri/ =404;

}

}

1. **Restart Nginx:**

bash

Copy code

sudo systemctl restart nginx

**Step 6: Test the Configuration**

* Open a web browser and navigate to http://webserver.cefalo-bd.com. You should see the default Nginx test page.

**Additional Steps**

**Create a Sample Index Page**

1. **Apache (httpd):**

bash

Copy code

echo "<h1>Welcome to webserver.cefalo-bd.com</h1>" | sudo tee /var/www/html/index.html

1. **Nginx:**

bash

Copy code

echo "<h1>Welcome to webserver.cefalo-bd.com</h1>" | sudo tee /usr/share/nginx/html/index.html

**Enable and Configure SSH on Port 2512**

1. **Install and Configure SSH:**

bash

Copy code

sudo yum install openssh-server -y

sudo systemctl start sshd

sudo systemctl enable sshd

1. **Edit SSH Configuration:**

bash

Copy code

sudo nano /etc/ssh/sshd\_config

* + Change the port and disable root login:

plaintext

Copy code

Port 2512

PermitRootLogin no

* + Save and close the file.

1. **Restart SSH Service:**

bash

Copy code

sudo systemctl restart sshd

**Transfer Folder from Windows Client to Server**

1. **Create Folder on Windows Desktop:**
   * Create a folder named copy-me on the Windows desktop.
2. **Transfer Folder Using SCP:**
   * Use an SCP tool (like WinSCP or command line):

bash

Copy code

scp -P 2512 -r C:\Users\<YourUsername>\Desktop\copy-me username@webserver.cefalo-bd.com:/var/www/html/

By following these steps, you'll have a fully configured web server on CentOS 7, accessible via webserver.cefalo-bd.com, with SSH enabled on port 2512, and a folder transferred from the Windows client.

4o

now make webserver as per the pdf requirement

ChatGPT

Based on the requirements outlined in the PDF, here are the detailed steps to configure the web server on CentOS 7.

**Web Server Configuration on CentOS 7**

**Prerequisites:**

* A virtual machine running CentOS 7 with 50GB disk, 4GB RAM, and 2 CPU cores.
* The MikroTik router has been configured to provide network connectivity.

**Step-by-Step Configuration**

**Step 1: Update the System**

bash

Copy code

sudo yum update -y

**Step 2: Install Apache Web Server**

bash

Copy code

sudo yum install httpd -y

**Step 3: Start and Enable Apache**

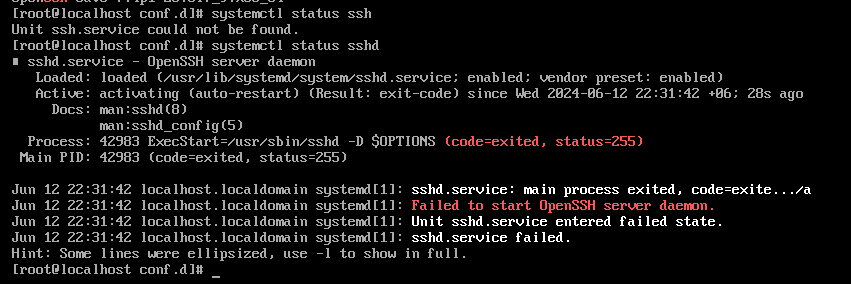
bash

Copy code

sudo systemctl start httpd

sudo systemctl enable httpd

4o



ChatGPT

The error message indicates that the SSH service failed to start. This could be due to various reasons, such as configuration errors or conflicts. Let's troubleshoot and fix the SSH service issue on CentOS 7.

**Troubleshooting SSH Service**

**Step 1: Check the SSH Configuration File**

1. Open the SSH configuration file for editing:

bash

Copy code

sudo nano /etc/ssh/sshd\_config

1. Ensure the configuration is correct. Check for any syntax errors or invalid directives. Here are some key points to verify:
   * Ensure the Port directive is set to 2512.
   * Make sure PermitRootLogin is set to no.
   * Verify other directives do not have syntax errors.

Example:

plaintext

Copy code

Port 2512

PermitRootLogin no

PasswordAuthentication yes

1. Save and close the file.

**Step 2: Check for Syntax Errors in the Configuration**

1. Verify the configuration file syntax:

bash

Copy code

sudo sshd -t

If there are any errors, correct them as indicated by the output.

**Step 3: Restart SSH Service**

1. Try to restart the SSH service again:

bash

Copy code

sudo systemctl restart sshd

**Step 4: Check Logs for Detailed Error Messages**

1. Check the system journal for SSH-related messages:

bash

Copy code

sudo journalctl -xe

Look for any specific error messages related to sshd and address them accordingly.

**Step 5: Ensure OpenSSH is Installed Correctly**

1. If the SSH service still fails to start, ensure OpenSSH is installed correctly:

bash

Copy code

sudo yum install openssh-server -y

1. Try starting the service again:

bash

Copy code

sudo systemctl start sshd

**Reconfigure the SSH Service if Necessary**

If the issue persists, you might need to reset the SSH configuration to default and then reapply your changes.

1. **Reset SSH Configuration:**

bash

Copy code

sudo mv /etc/ssh/sshd\_config /etc/ssh/sshd\_config.bak

sudo yum reinstall openssh-server

1. **Reapply Custom Configuration:**

bash

Copy code

sudo nano /etc/ssh/sshd\_config

Add your custom configurations again:

plaintext

Copy code

Port 2512

PermitRootLogin no

PasswordAuthentication yes

1. **Restart SSH Service:**

bash

Copy code

sudo systemctl restart sshd

**Verify SSH Service**

After making these adjustments, verify the SSH service is running:

bash

Copy code

sudo systemctl status sshd

If the service starts successfully, you should see it running without errors.

**Completing the Web Server Configuration**

Once the SSH service is running, proceed with the web server configuration steps as previously outlined:

**Step 4: Adjust the Firewall to Allow HTTP and HTTPS Traffic**

bash

Copy code

sudo firewall-cmd --permanent --add-service=http

sudo firewall-cmd --permanent --add-service=https

sudo firewall-cmd --reload

**Step 5: Configure Apache or Nginx**

1. **Apache:**

bash

Copy code

sudo nano /etc/httpd/conf.d/webserver.conf

Add the configuration:

apache

Copy code

<VirtualHost \*:80>

ServerAdmin webmaster@cefalo-bd.com

DocumentRoot /var/www/html

ServerName webserver.cefalo-bd.com

ErrorLog /var/log/httpd/webserver-error.log

CustomLog /var/log/httpd/webserver-access.log combined

</VirtualHost>

Restart Apache:

bash

Copy code

sudo systemctl restart httpd

1. **Nginx:**

bash

Copy code

sudo nano /etc/nginx/conf.d/webserver.conf

Add the configuration:

nginx

Copy code

server {

listen 80;

server\_name webserver.cefalo-bd.com;

root /usr/share/nginx/html;

index index.html;

access\_log /var/log/nginx/webserver-access.log;

error\_log /var/log/nginx/webserver-error.log;

location / {

try\_files $uri $uri/ =404;

}

}

Restart Nginx:

bash

Copy code

sudo systemctl restart nginx

**Final Steps**

**Create a Sample Index Page**

1. **Apache:**

bash

Copy code

echo "<h1>Welcome to webserver.cefalo-bd.com</h1>" | sudo tee /var/www/html/index.html

1. **Nginx:**

bash

Copy code

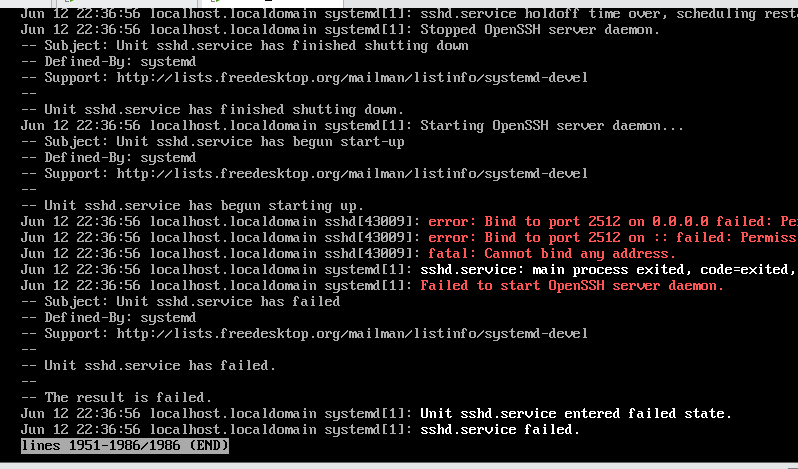
echo "<h1>Welcome to webserver.cefalo-bd.com</h1>" | sudo tee /usr/share/nginx/html/index.html

**Verify Web Server**

* Open a web browser and navigate to http://webserver.cefalo-bd.com.

This should complete the web server configuration on CentOS 7 according to the requirements outlined in the PDF.

4o



ChatGPT

The error message indicates that the SSH daemon (sshd) is unable to bind to port 2512 because of permission issues. This can happen if SELinux policies are enforcing restrictions on port bindings.

To resolve this, we'll need to configure SELinux to allow SSH to bind to the specified port.

**Steps to Resolve SSH Port Binding Issue**

**Step 1: Check the Current SELinux Status**

First, check the current status of SELinux:

bash

Copy code

sestatus

If SELinux is enforcing, you need to allow SSH to bind to the new port.

**Step 2: Allow SSH Port in SELinux**

1. **Add the New Port to SELinux Configuration:**

bash

Copy code

sudo semanage port -a -t ssh\_port\_t -p tcp 2512

If the semanage command is not available, install the required package:

bash

Copy code

sudo yum install policycoreutils-python -y

1. **Verify the Port has been Added:**

bash

Copy code

sudo semanage port -l | grep ssh

**Step 3: Restart SSH Service**

Now, try restarting the SSH service again:

bash

Copy code

sudo systemctl restart sshd

**Step 4: Adjust Firewall Rules**

Ensure that the firewall is allowing traffic on the new SSH port (2512):

bash

Copy code

sudo firewall-cmd --permanent --add-port=2512/tcp

sudo firewall-cmd --reload

**Step 5: Verify SSH Service Status**

Check the status of the SSH service to ensure it is running without issues:

bash

Copy code

sudo systemctl status sshd

**Complete the Web Server Configuration**

Now that SSH should be working, continue with the web server configuration as previously outlined:

**Install and Configure Apache or Nginx**

1. **Install Apache:**

bash

Copy code

sudo yum install httpd -y

sudo systemctl start httpd

sudo systemctl enable httpd

Or, **Install Nginx:**

bash

Copy code

sudo yum install epel-release -y

sudo yum install nginx -y

sudo systemctl start nginx

sudo systemctl enable nginx

**Configure Firewall for HTTP and HTTPS**

bash

Copy code

sudo firewall-cmd --permanent --add-service=http

sudo firewall-cmd --permanent --add-service=https

sudo firewall-cmd --reload

**Create and Configure Virtual Host**

1. **For Apache:**

bash

Copy code

sudo nano /etc/httpd/conf.d/webserver.conf

Add the following configuration:

apache

Copy code

<VirtualHost \*:80>

ServerAdmin webmaster@cefalo-bd.com

DocumentRoot /var/www/html

ServerName webserver.cefalo-bd.com

ErrorLog /var/log/httpd/webserver-error.log

CustomLog /var/log/httpd/webserver-access.log combined

</VirtualHost>

Restart Apache:

bash

Copy code

sudo systemctl restart httpd

1. **For Nginx:**

bash

Copy code

sudo nano /etc/nginx/conf.d/webserver.conf

Add the following configuration:

nginx

Copy code

server {

listen 80;

server\_name webserver.cefalo-bd.com;

root /usr/share/nginx/html;

index index.html;

access\_log /var/log/nginx/webserver-access.log;

error\_log /var/log/nginx/webserver-error.log;

location / {

try\_files $uri $uri/ =404;

}

}

Restart Nginx:

bash

Copy code

sudo systemctl restart nginx

**Create a Sample Index Page**

1. **Apache:**

bash

Copy code

echo "<h1>Welcome to webserver.cefalo-bd.com</h1>" | sudo tee /var/www/html/index.html

1. **Nginx:**

bash

Copy code

echo "<h1>Welcome to webserver.cefalo-bd.com</h1>" | sudo tee /usr/share/nginx/html/index.html

**Final Verification**

1. **Verify SSH Access:** Use an SSH client to connect to the server:

bash

Copy code

ssh -p 2512 user@webserver.cefalo-bd.com

1. **Verify Web Server:** Open a web browser and navigate to http://webserver.cefalo-bd.com. You should see the sample index page.

By following these steps, you should have a fully functional web server on CentOS 7 with SSH configured on a custom port (2512) and SELinux adjustments made to allow the port binding.

4o

Install your preferred web server on the Linux machine, and it should be accessible from the Windows-10/11 Client Machine with the address "webserver.cefalo-bd.com".

ChatGPT

To install a web server on a CentOS 7 machine and make it accessible from a Windows 10/11 client machine using the address "webserver.cefalo-bd.com", follow these steps. We'll use Apache as our preferred web server.

**Steps to Install Apache Web Server on CentOS 7**

**Step 1: Update the System**

bash

Copy code

sudo yum update -y

**Step 2: Install Apache**

bash

Copy code

sudo yum install httpd -y

**Step 3: Start and Enable Apache**

bash

Copy code

sudo systemctl start httpd

sudo systemctl enable httpd

**Step 4: Configure Firewall to Allow HTTP and HTTPS Traffic**

bash

Copy code

sudo firewall-cmd --permanent --add-service=http

sudo firewall-cmd --permanent --add-service=https

sudo firewall-cmd --reload

**Step 5: Set Up Virtual Host for "webserver.cefalo-bd.com"**

1. **Create Virtual Host Configuration File:**

bash

Copy code

sudo nano /etc/httpd/conf.d/webserver.conf

1. **Add the Following Configuration:**

apache

Copy code

<VirtualHost \*:80>

ServerAdmin webmaster@cefalo-bd.com

DocumentRoot /var/www/html

ServerName webserver.cefalo-bd.com

ErrorLog /var/log/httpd/webserver-error.log

CustomLog /var/log/httpd/webserver-access.log combined

</VirtualHost>

1. **Restart Apache to Apply Changes:**

bash

Copy code

sudo systemctl restart httpd

**Step 6: Create a Sample Index Page**

bash

Copy code

echo "<h1>Welcome to webserver.cefalo-bd.com</h1>" | sudo tee /var/www/html/index.html

**Steps to Access the Web Server from Windows 10/11 Client**

**Step 1: Update the Hosts File on Windows**

1. **Open Notepad as Administrator:**
   * Press Windows Key, type Notepad, right-click on Notepad, and select Run as administrator.
2. **Open the Hosts File:**
   * In Notepad, click on File -> Open.
   * Navigate to C:\Windows\System32\drivers\etc.
   * Select All Files from the dropdown menu to view the hosts file.
   * Open the hosts file.
3. **Add an Entry for the Web Server:**
   * Add the following line at the end of the file:

arduino

Copy code

<Your-Server-IP> webserver.cefalo-bd.com

* + Replace <Your-Server-IP> with the actual IP address of your CentOS server.

1. **Save the Hosts File:**
   * Save the file and close Notepad.

**Step 2: Verify the Web Server from Windows Client**

1. **Open a Web Browser:**
   * Open your preferred web browser on the Windows client.
2. **Access the Web Server:**
   * Enter the URL http://webserver.cefalo-bd.com.
   * You should see the sample index page displaying "Welcome to webserver.cefalo-bd.com".

**Verify Everything Works**

* **Apache Service Status:**

bash

Copy code

sudo systemctl status httpd

* **Firewall Rules:**

bash

Copy code

sudo firewall-cmd --list-all

By following these steps, you will have a functioning Apache web server on your CentOS 7 machine that is accessible from a Windows 10/11 client using the address "webserver.cefalo-bd.com".

4o

Install your preferred web server on the Linux machine, and it should be accessible from the Windows-10/11 Client Machine with the address "webserver.cefalo-bd.com".

ChatGPT

To set up an Apache web server on a CentOS 7 machine and ensure it is accessible from a Windows 10/11 client machine using the address "webserver.cefalo-bd.com", follow the detailed steps below.

**Step-by-Step Guide**

**Step 1: Update the CentOS System**

Ensure your CentOS 7 system is up to date.

bash

Copy code

sudo yum update -y

**Step 2: Install Apache Web Server**

Install the Apache web server package.

bash

Copy code

sudo yum install httpd -y

**Step 3: Start and Enable Apache Service**

Start the Apache service and enable it to start at boot.

bash

Copy code

sudo systemctl start httpd

sudo systemctl enable httpd

**Step 4: Configure Firewall to Allow HTTP and HTTPS Traffic**

Adjust the firewall settings to allow web traffic.

bash

Copy code

sudo firewall-cmd --permanent --add-service=http

sudo firewall-cmd --permanent --add-service=https

sudo firewall-cmd --reload

**Step 5: Configure Apache Virtual Host**

1. **Create a Virtual Host Configuration File:**

bash

Copy code

sudo nano /etc/httpd/conf.d/webserver.conf

1. **Add the Following Configuration:**

apache

Copy code

<VirtualHost \*:80>

ServerAdmin webmaster@cefalo-bd.com

DocumentRoot /var/www/html

ServerName webserver.cefalo-bd.com

ErrorLog /var/log/httpd/webserver-error.log

CustomLog /var/log/httpd/webserver-access.log combined

</VirtualHost>

1. **Restart Apache to Apply Changes:**

bash

Copy code

sudo systemctl restart httpd

**Step 6: Create a Sample Index Page**

Add a simple HTML page to verify the server is working.

bash

Copy code

echo "<h1>Welcome to webserver.cefalo-bd.com</h1>" | sudo tee /var/www/html/index.html

**Configure Windows Client to Access the Web Server**

**Step 1: Edit Hosts File on Windows**

1. **Open Notepad as Administrator:**
   * Press Windows Key, type Notepad, right-click on Notepad, and select Run as administrator.
2. **Open the Hosts File:**
   * In Notepad, click on File -> Open.
   * Navigate to C:\Windows\System32\drivers\etc.
   * Select All Files from the dropdown menu to view the hosts file.
   * Open the hosts file.
3. **Add an Entry for the Web Server:**
   * Add the following line at the end of the file:

arduino

Copy code

<Your-Server-IP> webserver.cefalo-bd.com

* + Replace <Your-Server-IP> with the actual IP address of your CentOS server.

1. **Save the Hosts File:**
   * Save the file and close Notepad.

**Step 2: Verify Access from Windows Client**

1. **Open a Web Browser:**
   * Open your preferred web browser on the Windows client.
2. **Access the Web Server:**
   * Enter the URL http://webserver.cefalo-bd.com.
   * You should see the sample index page displaying "Welcome to webserver.cefalo-bd.com".

**Verify Everything Works**

* **Apache Service Status:**

bash

Copy code

sudo systemctl status httpd

* **Firewall Rules:**

bash

Copy code

sudo firewall-cmd --list-all

By following these steps, you will have a fully functional Apache web server on your CentOS 7 machine, accessible from a Windows 10/11 client using the address "webserver.cefalo-bd.com".

4o

http://webserver.cefalo-bd.com by this url, nothing is shown in client end

ChatGPT

If "<http://webserver.cefalo-bd.com>" is not showing anything on the client end, we need to troubleshoot a few things to ensure everything is set up correctly.

**Troubleshooting Steps**

**1. Verify Apache is Running on the Server**

Make sure Apache is running:

bash

Copy code

sudo systemctl status httpd

You should see that the service is active (running). If not, start it:

bash

Copy code

sudo systemctl start httpd

**2. Check Firewall Rules**

Ensure that the firewall is allowing HTTP and HTTPS traffic:

bash

Copy code

sudo firewall-cmd --permanent --add-service=http

sudo firewall-cmd --permanent --add-service=https

sudo firewall-cmd --reload

**3. Verify Apache Configuration**

Check the Apache configuration file for any syntax errors:

bash

Copy code

sudo apachectl configtest

If there are errors, correct them and restart Apache:

bash

Copy code

sudo systemctl restart httpd

**4. Ensure the Virtual Host is Configured Correctly**

Make sure the virtual host configuration is correct:

bash

Copy code

sudo nano /etc/httpd/conf.d/webserver.conf

It should contain:

apache

Copy code

<VirtualHost \*:80>

ServerAdmin webmaster@cefalo-bd.com

DocumentRoot /var/www/html

ServerName webserver.cefalo-bd.com

ErrorLog /var/log/httpd/webserver-error.log

CustomLog /var/log/httpd/webserver-access.log combined

</VirtualHost>

Save any changes and restart Apache:

bash

Copy code

sudo systemctl restart httpd

**5. Create a Sample HTML Page**

Ensure there is content in the DocumentRoot:

bash

Copy code

echo "<h1>Welcome to webserver.cefalo-bd.com</h1>" | sudo tee /var/www/html/index.html

**6. Verify DNS Resolution from Client Machine**

Make sure the client machine can resolve "webserver.cefalo-bd.com" to the correct IP address.

On the Windows client machine, open Command Prompt and run:

cmd

Copy code

nslookup webserver.cefalo-bd.com

The output should show the IP address of your CentOS server.

**7. Check Hosts File on Client Machine**

Ensure that the hosts file on the Windows client machine has the correct entry:

1. Open Notepad as Administrator.
2. Open C:\Windows\System32\drivers\etc\hosts.
3. Verify the line:

arduino

Copy code

<Your-Server-IP> webserver.cefalo-bd.com

1. Save and close the file.

**8. Check Network Connectivity**

Make sure the client machine can reach the server IP:

cmd

Copy code

ping <Your-Server-IP>

**Additional Steps**

**Restart Network Service on Server**

Sometimes network services might need a restart:

bash

Copy code

sudo systemctl restart network

**Check SELinux Status**

If SELinux is enforcing, it might block the web server. You can set it to permissive mode:

bash

Copy code

sudo setenforce 0

To make it permanent:

bash

Copy code

sudo nano /etc/selinux/config

Set SELINUX=permissive.

**Verify IP Address Binding**

Ensure Apache is binding to the correct IP:

bash

Copy code

sudo nano /etc/httpd/conf/httpd.conf

Make sure the Listen directive is set to 80:

apache

Copy code

Listen 80

Save any changes and restart Apache:

bash

Copy code

sudo systemctl restart httpd

**Verify Web Server Access**

1. **On the Client Machine:**
   * Open a web browser and enter http://webserver.cefalo-bd.com.

If you have followed all these steps and are still facing issues, there might be a more specific problem with the network configuration or DNS resolution that needs to be addressed.

4o

Create a folder named “copy-me on” the Windows-10/11 CLIENT NODE's desktop that needs to be transferred to the Linux machine's web server root folder.

ChatGPT

To transfer a folder named "copy-me" from the Windows 10/11 client node's desktop to the Linux machine's web server root folder, you can follow these steps:

**Step-by-Step Guide**

**Step 1: Create the Folder on Windows Desktop**

1. **Create the Folder:**
   * On your Windows desktop, right-click, select New, then Folder.
   * Name the folder "copy-me".
2. **Add Files to the Folder:**
   * Add any files you want to include in the "copy-me" folder.

**Step 2: Install and Configure SSH on the Linux Machine**

Ensure that the SSH service is installed and running on the Linux server.

1. **Install SSH:**

bash

Copy code

sudo yum install openssh-server -y

1. **Start and Enable SSH:**

bash

Copy code

sudo systemctl start sshd

sudo systemctl enable sshd

1. **Check SSH Status:**

bash

Copy code

sudo systemctl status sshd

**Step 3: Transfer the Folder from Windows to Linux**

Use an SCP (Secure Copy Protocol) tool to transfer the folder. You can use WinSCP or any other SCP client.

**Using WinSCP:**

1. **Download and Install WinSCP:**
   * Download WinSCP from here and install it on your Windows machine.
2. **Open WinSCP and Configure Connection:**
   * Open WinSCP.
   * Enter the Linux server’s IP address in the Host name field.
   * Enter the Port number (default is 22 for SSH).
   * Enter your Username and Password.
   * Click Login.
3. **Navigate to the Desktop on Windows:**
   * In the left panel, navigate to Desktop and find the "copy-me" folder.
4. **Navigate to the Web Server Root Folder on Linux:**
   * In the right panel, navigate to the web server's root directory, typically /var/www/html.
5. **Transfer the Folder:**
   * Drag the "copy-me" folder from the left panel (Windows Desktop) to the right panel (Linux web server root directory).

**Step 4: Verify the Transfer on Linux**

Log into your Linux machine and verify the folder transfer:

1. **Check the Web Server Root Directory:**

bash

Copy code

ls /var/www/html

You should see the "copy-me" folder listed.

1. **Check the Contents of the Folder:**

bash

Copy code

ls /var/www/html/copy-me

**Using Command Line (Optional)**

Alternatively, if you prefer using command-line tools, you can use scp from a Windows command prompt if you have OpenSSH client installed.

1. **Install OpenSSH Client on Windows:**
   * Go to Settings -> Apps -> Optional features -> Add a feature.
   * Find and install OpenSSH Client.
2. **Open Command Prompt and Use scp:**

cmd

Copy code

scp -r C:\Users\<YourUsername>\Desktop\copy-me <YourLinuxUsername>@<YourServerIP>:/var/www/html

Replace <YourUsername> with your actual Windows username, <YourLinuxUsername> with your Linux username, and <YourServerIP> with the IP address of your Linux server.

**Verify Web Access**

1. **Open a Web Browser:**
   * Navigate to http://webserver.cefalo-bd.com/copy-me.
2. **You should see the contents of the "copy-me" folder displayed if it contains HTML files or be able to access individual files if they are directly referenced.**

By following these steps, you should successfully transfer the "copy-me" folder from the Windows client to the Linux web server's root directory.

4o

but the ssh port 2512

ChatGPT

Since the SSH port is set to 2512 instead of the default port 22, you need to specify this port during your file transfer. Here's how you can adjust the steps accordingly.

**Step-by-Step Guide with SSH Port 2512**

**Step 1: Create the Folder on Windows Desktop**

1. **Create the Folder:**
   * On your Windows desktop, right-click, select New, then Folder.
   * Name the folder "copy-me".
2. **Add Files to the Folder:**
   * Add any files you want to include in the "copy-me" folder.

**Step 2: Install and Configure SSH on the Linux Machine**

Ensure that the SSH service is installed and running on the Linux server with port 2512.

1. **Install SSH:**

bash

Copy code

sudo yum install openssh-server -y

1. **Configure SSH to Use Port 2512:**

bash

Copy code

sudo nano /etc/ssh/sshd\_config

Find the line that says #Port 22 and change it to:

plaintext

Copy code

Port 2512

Save and exit the editor.

1. **Start and Enable SSH:**

bash

Copy code

sudo systemctl start sshd

sudo systemctl enable sshd

1. **Restart SSH Service:**

bash

Copy code

sudo systemctl restart sshd

1. **Check SSH Status:**

bash

Copy code

sudo systemctl status sshd

**Step 3: Transfer the Folder from Windows to Linux**

Use an SCP (Secure Copy Protocol) tool to transfer the folder, specifying the custom port 2512. You can use WinSCP or any other SCP client.

**Using WinSCP:**

1. **Download and Install WinSCP:**
   * Download WinSCP from here and install it on your Windows machine.
2. **Open WinSCP and Configure Connection:**
   * Open WinSCP.
   * Enter the Linux server’s IP address in the Host name field.
   * Enter 2512 in the Port number field.
   * Enter your Username and Password.
   * Click Login.
3. **Navigate to the Desktop on Windows:**
   * In the left panel, navigate to Desktop and find the "copy-me" folder.
4. **Navigate to the Web Server Root Folder on Linux:**
   * In the right panel, navigate to the web server's root directory, typically /var/www/html.
5. **Transfer the Folder:**
   * Drag the "copy-me" folder from the left panel (Windows Desktop) to the right panel (Linux web server root directory).

**Step 4: Verify the Transfer on Linux**

Log into your Linux machine and verify the folder transfer:

1. **Check the Web Server Root Directory:**

bash

Copy code

ls /var/www/html

You should see the "copy-me" folder listed.

1. **Check the Contents of the Folder:**

bash

Copy code

ls /var/www/html/copy-me

**Using Command Line with Custom SSH Port (Optional)**

Alternatively, if you prefer using command-line tools, you can use scp from a Windows command prompt if you have OpenSSH client installed.

1. **Install OpenSSH Client on Windows:**
   * Go to Settings -> Apps -> Optional features -> Add a feature.
   * Find and install OpenSSH Client.
2. **Open Command Prompt and Use scp with Port 2512:**

cmd

Copy code

scp -P 2512 -r C:\Users\<YourUsername>\Desktop\copy-me <YourLinuxUsername>@<YourServerIP>:/var/www/html

Replace <YourUsername> with your actual Windows username, <YourLinuxUsername> with your Linux username, and <YourServerIP> with the IP address of your Linux server.

**Verify Web Access**

1. **Open a Web Browser:**
   * Navigate to http://webserver.cefalo-bd.com/copy-me.
2. **You should see the contents of the "copy-me" folder displayed if it contains HTML files or be able to access individual files if they are directly referenced.**

By following these steps, you should successfully transfer the "copy-me" folder from the Windows client to the Linux web server's root directory, even with the custom SSH port 2512.

4o

Create a linked-clone of the CLIENT, and name it CLIENT-CLONE. This Windows machine should be accessible remotely through Remote Desktop Protocol (RDP) with port 3339.

ChatGPT

To create a linked-clone of the CLIENT virtual machine and configure it to be accessible via Remote Desktop Protocol (RDP) on port 3339, follow these steps. We'll use VMware Workstation for creating the linked-clone and Windows built-in features for RDP configuration.

**Step-by-Step Guide**

**Step 1: Create a Linked-Clone of the CLIENT VM**

1. **Open VMware Workstation:**
   * Start VMware Workstation on your host machine.
2. **Select the CLIENT VM:**
   * In the left pane, select the CLIENT VM that you want to clone.
3. **Create Linked Clone:**
   * Right-click on the CLIENT VM and select Manage -> Clone.
   * The Clone Virtual Machine Wizard will open. Click Next.
4. **Specify Clone Type:**
   * Select Create a linked clone and click Next.
5. **Select Snapshot:**
   * Select an existing snapshot to clone or create a new snapshot and then select it. Click Next.
6. **Name the Clone:**
   * Enter CLIENT-CLONE as the name for the new virtual machine.
   * Specify a location for the clone files and click Finish.

**Step 2: Configure RDP on CLIENT-CLONE to Use Port 3339**

1. **Start CLIENT-CLONE:**
   * Power on the CLIENT-CLONE VM in VMware Workstation.
2. **Enable Remote Desktop:**
   * Right-click This PC and select Properties.
   * Click Remote settings in the left pane.
   * Under Remote Desktop, select Allow remote connections to this computer.
   * Click OK.
3. **Configure RDP to Use Port 3339:**
   * Open Registry Editor by pressing Win + R, typing regedit, and pressing Enter.
   * Navigate to the following key:

arduino

Copy code

HKEY\_LOCAL\_MACHINE\System\CurrentControlSet\Control\Terminal Server\WinStations\RDP-Tcp

* + Find the PortNumber entry. If it doesn't exist, create it as a DWORD (32-bit) Value.
  + Right-click PortNumber, select Modify, choose Decimal, and enter 3339.
  + Click OK.

1. **Allow RDP Through Windows Firewall:**
   * Open Control Panel, go to System and Security -> Windows Firewall -> Advanced settings.
   * In the left pane, click Inbound Rules, and then in the right pane, click New Rule.
   * Select Port and click Next.
   * Select TCP and specify port 3339. Click Next.
   * Allow the connection, click Next.
   * Apply the rule to Domain, Private, and Public profiles, and click Next.
   * Name the rule RDP Port 3339 and click Finish.

**Step 3: Verify RDP Configuration**

1. **Restart the CLIENT-CLONE VM:**
   * Restart the CLIENT-CLONE VM to apply the changes.
2. **Verify Listening Port:**
   * Open Command Prompt and run:

cmd

Copy code

netstat -an | find "3339"

* + You should see the port 3339 listening.

1. **Test RDP Connection:**
   * From another machine, open Remote Desktop Connection.
   * Enter the IP address of the CLIENT-CLONE VM followed by :3339 (e.g., 192.168.1.100:3339).
   * Click Connect and enter the credentials to verify the connection.

**Troubleshooting Tips**

* **Ensure Network Connectivity:**
  + Make sure CLIENT-CLONE VM is connected to the network and its IP address is reachable.
* **Check Windows Firewall:**
  + Verify that the firewall rule for port 3339 is correctly configured and enabled.
* **Confirm RDP Service Status:**
  + Ensure the RDP service is running by checking Services (services.msc), and verify that Remote Desktop Services is set to Automatic and is running.

Following these steps, you will successfully create a linked clone of the CLIENT VM and configure it to be accessible via RDP on port 3339.