Setting Up, Hosting, and Securing an SFTP Server on Debian 12 with Access from Windows 11

This guide provides comprehensive instructions on how to create, host, and secure an SFTP (Secure File Transfer Protocol) server on a Debian 12 machine and access it from a Windows 11 machine. The steps include installation, configuration, security enhancements, and accessing the server using a Windows client.

Table of Contents

Prepared by:
Jon Tweeton

Last Modified12/18/2024

Prerequisites

Before starting, ensure you have the following:

- Debian 12 Server:
    - A machine running Debian 12 with administrative (root) access.
    - A static IP address or a resolvable hostname.
- Windows 11 Client:
    - A Windows 11 machine with internet/network access to the Debian server.
    - Administrative rights to install software.
- Basic Knowledge:
    - Familiarity with the Linux command line.
    - Understanding of user and permission management in Linux.

Prepared by:
Jon Tweeton
Last Modified12/18/2024

Installing OpenSSH Server (Debian Machine)

OpenSSH provides the necessary tools to set up an SFTP server.

1. **Update Package Lists:**

```
sudo apt update
```

2. **Install OpenSSH Server:**

```
sudo apt install openssh-server
```

3. **Verify Installation:**

Ensure the SSH service is running.

```
sudo systemctl status ssh
```

You should see an active (running) status.

4. **Enable SSH to Start on Boot:**

```
sudo systemctl enable ssh
```

Configuring the SFTP Server

Configure OpenSSH to support SFTP and enhance security.

1. **Backup Original Configuration:**

   ```
   sudo cp /etc/ssh/sshd_config /etc/ssh/sshd_config.backup
   ```

2. **Edit SSH Configuration:**

   Open the SSH daemon configuration file.

   ```
   sudo nano /etc/ssh/sshd_config
   ```

3. **Modify Configuration for SFTP:**

   ○ **Change the SSH Port (Optional but Recommended):**

   To reduce unauthorized access attempts, you can change the default SSH port (22) to a non-standard port (e.g., 2222).

   ```
   Port 2222
   ```

   Note: Ensure this port is allowed through the firewall (covered in Managing Firewall Settings).

   ○ **Disable Root Login:**

   ```
   PermitRootLogin no
   ```

   ○ **Disable Password Authentication (Optional for Enhanced Security):**

   If you plan to use SSH keys.

   ```
   PasswordAuthentication no
   ```

   ○ **Configure SFTP Subsystem:**

   Ensure the SFTP subsystem is properly configured.

   ```
   Subsystem sftp internal-sftp
   ```

   ○ **Add SFTP Group Configuration:**

   At the end of the file, add:

   ```
   Match Group sftpusers
     ChrootDirectory /home/%u
     ForceCommand internal-sftp
     X11Forwarding no
     AllowTcpForwarding no
   ```

   This configuration ensures that users in the sftpusers group are restricted to their home directories and can only use SFTP.

4. **Save and Exit:**

   Press *CTRL* + *X*, then *CTRL* + *S* to save changes.

5. **Restart SSH Service:**

   ```
   sudo systemctl restart ssh
   ```

Prepared by:
Jon Tweeton
Last Modified12/18/2024

Creating SFTP Users and Directories

Create dedicated users for SFTP access and set appropriate directory permissions.

1. **Create a New Group for SFTP Users:**

   ```
   sudo groupadd sftpusers
   ```

2. **Create a New User:**

   Replace username with the desired username.

   ```
   sudo useradd -m -G sftpusers -s /sbin/nologin username
   ```

   - -m: Creates a home directory.
   - -G sftpusers: Adds the user to the sftpusers group.
   - -s /sbin/nologin: Prevents SSH shell access.

3. **Set User Password:**

   ```
   sudo passwd username
   ```

4. **Set Directory Permissions:**

   The user's home directory must be owned by root and not writable by any other user.

   ```
   sudo chown root:root /home/username
   sudo chmod 755 /home/username
   ```

5. **Create an Upload Directory:**

   Allow users to upload files to a subdirectory.

   ```
   sudo mkdir /home/username/uploads
   sudo chown username:sftpusers /home/username/uploads
   sudo chmod 755 /home/username/uploads
   ```

6. **Repeat for Additional Users:**

   Follow steps 2-5 for each additional SFTP user.

Securing the SFTP Server

Implement security best practices to protect your SFTP server.

1. **Use SSH Keys for Authentication (Optional but Recommended):**
   - **Generate SSH Keys on Windows:**
     Use a tool like PuTTYgen or the built-in ssh-keygen if using Windows Subsystem for Linux (WSL).
   - **Copy Public Key to Server:**
     On the Debian server:

     ```
     sudo mkdir /home/username/.ssh
     sudo nano /home/username/.ssh/authorized_keys
     ```

     Paste the public key into authorized_keys, save, and exit.
   - **Set Permissions:**

     ```
     sudo chmod 700 /home/username/.ssh
     sudo chmod 600 /home/username/.ssh/authorized_keys
     sudo chown -R username:sftpusers /home/username/.ssh
     ```

   - **Disable Password Authentication:**
     As configured earlier in *sshd_config:*

     ```
     PasswordAuthentication no
     ```

     Note: Ensure SSH keys are correctly set up before disabling password authentication to prevent lockout.

2. **Install and Configure Fail2Ban:**
   Protect against brute-force attacks.

   ```
   sudo apt install fail2ban
   ```

   Create a local configuration file:

   ```
   sudo cp /etc/fail2ban/jail.conf /etc/fail2ban/jail.local
   sudo nano /etc/fail2ban/jail.local
   ```

   Add the following under [sshd]:

   ```
   [sshd]
   enabled = true
   port = 2222
   filter = sshd
   logpath = /var/log/auth.log
   maxretry = 5
   bantime = 600
   ```

   Save and exit, then restart Fail2Ban:

   ```
   sudo systemctl restart fail2ban
   ```

Prepared by:
Jon Tweeton
Last Modified12/18/2024

3. **Enable UFW Firewall and Allow Necessary Ports:**

```
sudo apt install ufw
sudo ufw allow
2222/tcp
sudo ufw allow
OpenSSH
sudo ufw enable
sudo ufw status
```

4. **Regularly Update the System:**

   Keep your server updated to patch vulnerabilities.

```
sudo apt update && sudo apt upgrade -y
```

Prepared by:
Jon Tweeton
Last Modified12/18/2024

Managing Firewall Settings

Ensure that the firewall allows SFTP connections and restricts unnecessary access.

1. **Check UFW Status:**
   ```
   sudo ufw status
   ```
2. **Allow SFTP Port:**

   If you changed the SSH port to 2222, allow it:
   ```
   sudo ufw allow 2222/tcp
   ```
   If using the default port 22:
   ```
   sudo ufw allow 22/tcp
   ```
3. **Enable UFW:**

   If not already enabled.
   ```
   sudo ufw enable
   ```
4. **Deny Unnecessary Ports:**

   Ensure that only required ports are open.
   ```
   sudo ufw deny <port_number>
   ```
   Replace <port_number> with the specific port to deny.
5. **Reload UFW to Apply Changes:**
   ```
   sudo ufw reload
   ```

Accessing the SFTP Server from Windows 11

Use an SFTP client to connect to the Debian server. This guide uses WinSCP, a popular free SFTP client for Windows.

1. **Download and Install WinSCP:**
   - Visit WinSCP Download Page.
   - Download and install the latest version.
2. **Launch WinSCP and Create a New Session:**
   - **File Protocol**: SFTP
   - **Host Name**: <Debian_Server_IP> or <Hostname>
   - **Port Number**: 2222 (or your configured SSH port)
   - **User Name**: username (SFTP user)
   - **Password**: password (if using password authentication)
3. **If using SSH keys:**
   - Private Key File: Browse to your private key file (e.g., .ppk for PuTTY).
4. **Save the Session (Optional):**
   Click Save to store the session details for future use.
5. **Connect:**
   Click Login to connect to the SFTP server.

Prepared by:
Jon Tweeton
Last Modified12/18/2024

Testing the SFTP Connection

After setting up, verify that the connection works correctly.

1. **Open WinSCP and Connect:**
   Use the session details to connect.
2. **Navigate Directories:**
   Upon successful connection, you should see the uploads directory.
3. **Transfer Files:**
   - Upload: Drag and drop files from Windows to the uploads directory.
   - Download: Drag and drop files from the server to Windows.
4. **Verify Permissions:**
   Ensure that users can upload and download files but cannot navigate outside their designated directories.

Prepared by:
Jon Tweeton
Last Modified12/18/2024

Maintaining and Monitoring the SFTP Server

Regular maintenance and monitoring are essential for security and performance.

1. **Monitor Logs:**
   Check SSH and Fail2Ban logs for any suspicious activity.
   ```
   sudo tail -f /var/log/auth.log
   sudo fail2ban-client status
   ```
2. **Regular Backups:**
   Backup user data and server configurations regularly.
   ```
   sudo rsync -av /home/username/uploads /path/to/backup/
   ```
3. **Update Software:**
   Keep the system and all software up to date.
   ```
   sudo apt update && sudo apt upgrade -y
   ```
4. **Audit User Access:**
   Regularly review user accounts and permissions.
   ```
   getent group sftpusers
   ```

Prepared by:
Jon Tweeton
Last Modified12/18/2024

Troubleshooting

If you encounter issues, follow these steps:

1. Cannot Connect:
    - Verify that the SSH service is running.

      ```
      sudo systemctl status ssh
      ```

    - Check firewall settings to ensure the SFTP port is open.

      ```
      sudo ufw status
      ```

    - Ensure you are using the correct IP/hostname and port.
2. Permission Denied Errors:
    - Verify user permissions and directory ownership.

      ```
      ls -ld /home/username
      ls -ld /home/username/uploads
      ```

    - Ensure the SSH configuration is correctly set for the sftpusers group.
3. Failed Login Attempts:
    - Check Fail2Ban status to see if the IP is banned.

      ```
      sudo fail2ban-client status sshd
      ```

    - Unban an IP if necessary.

      ```
      sudo fail2ban-client set sshd unbanip <IP_ADDRESS>
      ```

4. Cannot Transfer Files:
    - Ensure sufficient disk space on the server.

      ```
      df -h
      ```

    - Check file permissions in the uploads directory.

      ```
      sudo sshd -t
      ```

5. SSH Service Fails to Restart:
    - Review the SSH configuration for syntax errors.

      ```
      command template
      ```

    - Correct any reported issues before restarting.

Prepared by:
Jon Tweeton
Last Modified12/18/2024

Sources

The following resources were used to compile this documentation:

1. **OpenSSH Server Installation and Configuration:**
   - OpenSSH Official Documentation
   - DigitalOcean: How To Set Up an SFTP Server with OpenSSH
2. **Debian 12 Specific Instructions:**
   - Debian 12 Official Release Notes
3. **Securing SSH/SFTP:**
   - Fail2Ban Documentation
   - UFW - Uncomplicated Firewall Documentation
4. **Windows 11 SFTP Client:**
   - WinSCP Official Website
5. **General Linux User and Permission Management:**
   - Linuxize: How to Manage Users and Groups in Linux
6. **Best Practices for SSH Security:**
   - SSH Best Practices
7. **Troubleshooting SSH:**
   - SSH Troubleshooting Guide

Prepared by:
Jon Tweeton
Last Modified12/18/2024

Appendix

Prepared by:
Jon Tweeton
Last Modified12/18/2024