Server Sicherheit erhöhen

> Um die Sicherheit auf einem Server zu erhöhen sollten ein paar Vorkehrungen getroffen werden.

Youtube Viedos

Wichtige erste Schritte auf einem Server

SSH Zugang anpassen

Den standard-mäßigen Zugang als root-User auf Linux deaktivieren. Durch die Deaktivierung wird schon mal die erste Hürde eines Hackers aufgebaut, dass er nicht weiß, wie der Admin-User heißt.

Neuen User anlegen, der die Admin Rolle übernehmen soll
Wir arbeiten als root user

`adduser BENUTZERNAME`

Passwort vergeben und Daten eintragen wenn gewünscht

Neuen User der Gruppe sudo hinzufügen

`usermod -aG sudo BENUTZERNAME`

Home-Verzeichnis Rechte setzen (Client und Server)

`:~$ sudo chmod 755 /home/<Benutzer>`

Schlüsselpaar auf dem Client erstellen
Ich habe mich für einen 2048 Bit langen RSA-Schlüssel entschieden. Ihr könnt die folgenden Eingaben mit Enter bestätigen. Die Eingabe einer Passphrase für den Key ist empfohlen, doch für unser Vorhaben nicht praktikabel.

`:~$ ssh-keygen`

Public Key vom Client auf den Server transferieren
Bei diesem Schritt ist die Eingabe des Passworts ein letztes Mal notwendig, um den Key auf den Server zu transferieren.

`:~$ ssh-copy-id <Benutzer>@192.168.0.130`

Public Key vom Server aus vom Client holen

die id_rsa.pub datei per scp auf den Server kopieren und mit `cat id_rsa.pub >> /home/hermann/.ssh/authorizied_keys` anhängen

> oder

Der Key kann auch manuell auf den Server übertragen werden indem man auf dem Server im Homeverzeichnis des Benutzers mit dem man sich verbinden will unter /home/benutzer/.ssh/authorizised_keys den text aus der id_rsa.pub Datei rein kopiert.

Wurde die Verbindung erfolgreich hergestellt, ist in der Datei /home/<Benutzer>/.ssh/authorized_keys auf dem Server der Public-Key vom Client erfolgreich eingetragen. Prüfen könnt ihr das wie folgt:

```
:~$ cat /home/<Benutzer>/.ssh/authorized_keys
```

SSH-Key Verbindung testen – Client zu Server
So könnt ihr testen, ob die Einrichtung erfolgreich abgeschlossen wurde – die Passworteingabe ist ab jetzt nicht mehr erforderlich.

```
:~$ ssh <Benutzer>@192.168.0.130
```

Hat man die Konfiguration abgeschlossen und der SSH Zugriff mit Key funktioniert, so ist die Deaktivierung des Anmeldeverfahrens mit Passwort möglich. Dazu einfach diese Zeile in der systemweiten Konfiguration ändern.

```
:~$ sudo nano /etc/ssh/ssh_config
```

SSH-Config anpassen

```
# Datei befindet sich /etc/ssh/sshd_config


PermitRootLogin no
```

```
PasswordAuthentication no
```

Authorized_Key kopieren

```
# Erst einen Ordner für den Key anlegen
mkdir /home/BENUTZERNAME/.ssh

# der Authorized Key befindet sich unter /root/.ssh
cp /root/.ssh/authorized_key /home/BENUTZERNAME/.ssh
```

SSH Neustarten

`sudo systemctl restart sshd`

Firewall

Installieren

`apt install ufw`

```
# Applikationen anzeigen
ufw app list

# Alles blockieren
ufw default deny incoming

# SSH erlauben
ufw allow 22/tcp

# Firewall aktivieren
ufw enable

# Firewall Status anzeigen
ufw status

# Port für eine IP-Ardesse frei geben
sudo ufw allow from 192.168.1.100 to any port 80
```

```
# Ping verbieten
sudo ufw deny proto icmp
```

Systemcheck durchführen

```
git clone https://github.com/CISOfy/lynis

cd lynis

sudo chown -R 0:0 *

sudo ./lynis audit system
```

Die Warnungen mit 'System-analyze security' kann man dabei ignorieren, denn dabei geht es nur um optionale Sandbox-Funktionen von Systemd.

Logwatch: Über Angriffe Informiert

```
sudo apt install logwatch

# Starten
sudo logwatch
```

Revision #8
Created 21 March 2023 07:57:55 by Hermann
Updated 9 April 2024 10:27:51 by Hermann