Exercício orientado - Personalização da configuração do serviço OpeSSH

Instruções

1. Na `workstation`, abra uma sessão de SSH para a máquina `serverb` como o usuário `student`.

```
[student@workstation ~]$ssh student@serverb
[student@serverb ~]$
```

2. Use o comando `su` para alternar para o usuário `operator2` na máquina `serverb`. Use `redhat` como a senha do usuário `operator2`.

```
[student@serverb ~]$su - operator2
Password:redhat
[operator2@serverb ~]$
```

3. Use o comando `ssh-keygen` para gerar chaves SSH. Não insira senhas para as chaves.

```
[operator2@serverb ~]$ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/operator2/.ss
h/id_rsa):Enter
Created directory '/home/operator2/.ssh'.
Enter passphrase (empty for no passphrase):Enter
Enter same passphrase again:Enter
Your identification has been saved in /home/operator2/.s
sh/id_rsa.
Your public key has been saved in /home/operator2/.ssh/i
d_rsa.pub.
```

```
The key fingerprint is:
SHA256:LN5x1irX0OWxgyd/qhATNgZWOtLUj16EZkM1JHkCR+I opera
tor2@serverb.lab.example.com
The key's randomart image is:
+---[RSA 3072]----+
|*=+    |
|       = =O.o.   |
|      . Eo=B  o  |
|       o +.=o+ o |
|      . S..= =  |
|     . o +. + . |
|      . o + . . .|
|         o .   o |
|            ...  |
+----[SHA256]-----+
```

4. Use o comando `ssh-copy-id` para enviar a chave pública do par de chaves SSH para o usuário `operator2` na máquina `servera`. Use `redhat` como a senha do usuário `operator2` no `servera`.

```
[operator2@serverb ~]$ssh-copy-id operator2@servera
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be insta
lled: "/home/operator2/.ssh/id_rsa.pub"
The authenticity of host 'servera (172.25.250.10)' can't
be established.
ED25519 key fingerprint is SHA256:h/hEJa/anxp6AP7BmB5azI
PVbPNqieh0oKi4KWOTK80.
Are you sure you want to continue connecting (yes/no)?ye
s
/usr/bin/ssh-copy-id: INFO: attempting to log in with th
e new key(s), to filter out any that are already install
ed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be instal
led -- if you are prompted now it is to install the new
keys
operator2@servera's password:redhat
Number of key(s) added: 1
Now try logging into the machine, with:   "ssh 'operator
```

```
2@servera'"
and check to make sure that only the key(s) you wanted w
ere added.
```

5. Confirme se você consegue fazer login com êxito na máquina `servera` como o usuário `operator2` com as chaves SSH.

   a. Abra uma sessão de SSH para a máquina `servera` como o usuário `operator2`.

   ```
   [operator2@serverb ~]$ssh operator2@servera...output
   omitted...
   [operator2@servera ~]$
   ```

   O comando `ssh` anterior usou chaves SSH para autenticação.

   b. Faça o logout da máquina `servera`.

   ```
   [operator2@servera ~]$exit
   logout
   Connection to servera closed.
   ```

6. Confirme se você consegue fazer login com êxito na máquina `servera` como o usuário `root` com `redhat` como a senha.

   a. Abra uma sessão de SSH para a máquina `servera` com o usuário `root` usando a senha `redhat`.

   ```
   [operator2@serverb ~]$ssh root@servera
   root@servera's password:redhat...output omitted...
   [root@servera ~]#
   ```

   O comando `ssh` anterior usou a senha do superusuário para autenticação, porque as chaves SSH não existem para o superusuário.

   b. Faça o logout da máquina `servera`.

   ```
   [root@servera ~]#exit
   logout
   ```

```
Connection to servera closed.
[operator2@serverb ~]$
```

7. Confirme se você consegue fazer login com êxito na máquina `servera` como o usuário `operator3` com `redhat` como a senha.

   a. Abra uma sessão de SSH para a máquina `servera` com o usuário `operator3` usando a senha `redhat`.

   ```
   [operator2@serverb ~]$ssh operator3@servera
   operator3@servera's password:redhat...output omitte
   d...
   [operator3@servera ~]$
   ```

   O comando `ssh` anterior usou a senha do usuário `operator3` para autenticação, porque as chaves SSH não existem para o usuário `operator3`.

   b. Faça o logout da máquina `servera`.

   ```
   [operator3@servera ~]$exit
   logout
   Connection to servera closed.
   [operator2@serverb ~]$
   ```

8. Configure o serviço `sshd` na máquina `servera` para impedir que os usuários façam login como o usuário `root`. Use `redhat` como a senha do superusuário quando necessário.

   a. Abra uma sessão de SSH para a máquina `servera` como o usuário `operator2` com as chaves SSH.

   ```
   [operator2@serverb ~]$ssh operator2@servera...output
   omitted...
   [operator2@servera ~]$
   ```

   b. Na máquina `servera`, alterne para o usuário `root`. Use `redhat` como a senha do usuário `root`.

```
[operator2@servera ~]$su -
Password:redhat
[root@servera ~]#
```

c. Defina `PermitRootLogin` como `no` no arquivo `/etc/ssh/sshd_config` e recarregue o serviço `sshd`. Você pode usar o comando `vim /etc/ssh/sshd_config` para editar o arquivo de configuração do serviço `sshd`.

```
...output omitted...
PermitRootLogin no
...output omitted...
[root@servera ~]#systemctl reload sshd
```

d. Abra outro terminal na `workstation` e abra uma sessão de SSH na máquina `serverb` com o usuário `operator2`. Na máquina `serverb`, tente fazer login na máquina `servera` como o usuário `root`. Esse comando deve falhar, porque você desativou o login de usuário `root` no SSH na etapa anterior.

Nota

Para sua conveniência, o login sem senha já está configurado entre a `workstation` e o `serverb` no ambiente de sala de aula.

```
[student@workstation ~]$ssh operator2@serverb...outpu
t omitted...
[operator2@serverb ~]$ssh root@servera
root@servera's password:redhat
Permission denied, please try again.
root@servera's password:redhat
Permission denied, please try again.
root@servera's password:redhat
root@servera: Permission denied (publickey,gssapi-key
ex,gssapi-with-mic,password).
```

Por padrão, o comando `ssh` tenta autenticar com a autenticação baseada em chave primeiro e, depois, se isso falhar, ele tenta com a

autenticação baseada em senha.

9. Configure o serviço `sshd` na máquina `servera` para permitir que os usuários autentiquem com somente chaves SSH, em vez de com suas senhas.

   a. Volte para o primeiro terminal com o shell ativo do usuário `root` na máquina `servera`. Defina o parâmetro `PasswordAuthentication` como `no` no arquivo `/etc/ssh/sshd_config` e recarregue o serviço `sshd`. Você pode usar o comando `vim /etc/ssh/sshd_config` para editar o arquivo de configuração do serviço `sshd`.

   ```
   ...output omitted...
   PasswordAuthentication no
   ...output omitted...
   [root@servera ~]#systemctl reload sshd
   ```

   b. Acesse o segundo terminal com o shell ativo do usuário `operator2` na máquina `serverb` e tente fazer login na máquina `servera` como o usuário `operator3`. Esse comando deve falhar, porque as chaves SSH não estão configuradas para o usuário `operator3`, e o serviço `sshd` na máquina `servera` não permite o uso de senhas para autenticação.

   ```
   [operator2@serverb ~]$ssh operator3@servera
   operator3@servera: Permission denied (publickey,gssap
   i-keyex,gssapi-with-mic).
   ```

   ### Nota

   Para mais granularidade, você pode usar as opções explícitas `-o PubkeyAuthentication=no` e `-o PasswordAuthentication=yes` com o comando `ssh`. Você pode então substituir os padrões do comando `ssh` e determine com segurança que o comando anterior falhe com base nas configurações ajustadas no arquivo `/etc/ssh/sshd_config` na etapa anterior.

   c. Volte para o primeiro terminal com o shell ativo do usuário `root` na máquina `servera`. Verifique se `PubkeyAuthentication` está ativado no arquivo `/etc/ssh/sshd_config`. Você pode usar o comando `vim /etc/ssh/sshd_config` para ver o arquivo de configuração do serviço `sshd`.

```
...output omitted...
#PubkeyAuthentication yes
...output omitted...
```

A linha `PubkeyAuthentication` está comentada. As linhas comentadas indicam os valores padrão de um parâmetro. A autenticação de chave pública do SSH está ativa por padrão, como a linha comentada indica.

d. Retorne ao segundo terminal com o shell ativo do usuário `operator2` na máquina `serverb` e tente fazer login na máquina `servera` com o usuário `operator2`. Esse comando deve ser bem-sucedido, porque as chaves SSH estão configuradas para o usuário `operator2` fazer login na máquina `servera` a partir da máquina `serverb`.

```
[operator2@serverb ~]$ssh operator2@servera...output
omitted...
[operator2@servera ~]$
```

e. Do segundo terminal, saia do shell do usuário `operator2` nas máquinas `servera` e `serverb`.

```
[operator2@servera ~]$exit
logout
Connection to servera closed.
[operator2@serverb ~]$exit
logout
Connection to serverb closed.
[student@workstation ~]$
```

f. Feche o segundo terminal na máquina `workstation`.

```
[student@workstation ~]$exit
```

g. No primeiro terminal, saia do shell do usuário `root` na máquina `servera`.

```
[root@servera ~]#exit
logout
```

h. Do primeiro terminal, saia do shell do usuário `operator2` nas máquinas `servera` e `serverb`.

```
[operator2@servera ~]$exit
logout
Connection to servera closed.
[operator2@serverb ~]$exit
logout
[student@serverb ~]$
```

i. Faça o logout do `serverb` e volte para o shell do usuário `student` na `workstation`.

```
[student@serverb ~]$exit
logout
Connection to serverb closed.
[student@workstation ~]$
```

Encerramento

Na máquina `workstation`, altere para o diretório pessoal do usuário `student` e use o comando `lab` para concluir este exercício. Essa etapa é importante para garantir que recursos de exercícios anteriores não afetem exercícios futuros.

```
[student@workstation ~]$lab finish ssh-customize
```

Isso conclui a seção.