

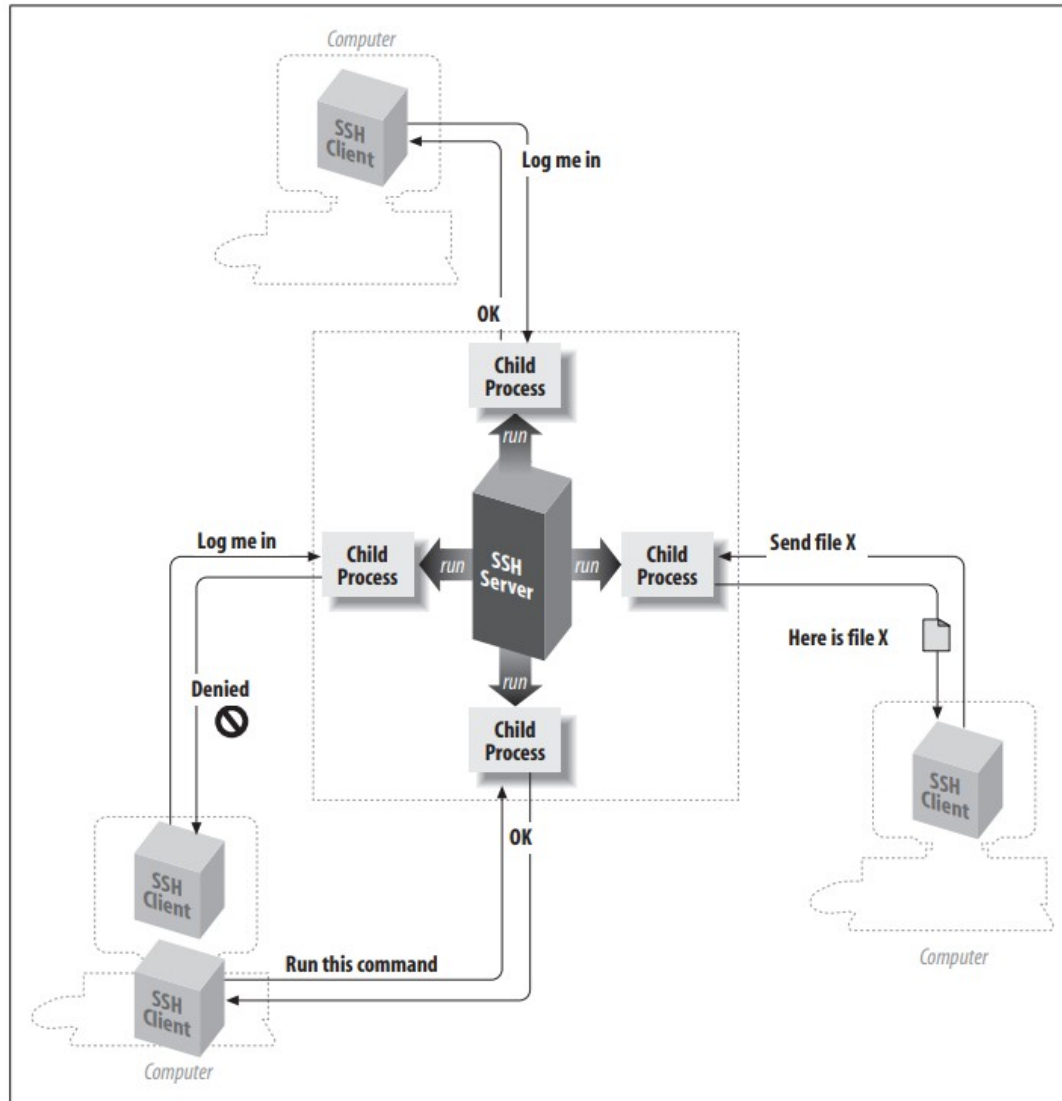
Entendendo o SSH (Secure Shell)

**Uma introdução ao protocolo
SSH para alunos de Sistemas de
Informação**

O que é SSH(Secure Shell)?

- Protocolo de rede criptográfico para operação segura;
- Ele criptografa automaticamente os dados enviados pela rede, garantindo que, ao chegar ao destinatário, sejam automaticamente descriptografados.
- A arquitetura do SSH é baseada no modelo cliente/servidor. Um programa de servidor SSH, geralmente configurado por um administrador de sistema, aceita ou rejeita conexões de entrada.

Arquitetura Cliente-Servidor



O que é SSH(Secure Shell)?

- SSH é um protocolo, não um produto. É uma especificação de como conduzir comunicação segura em uma rede.
- O protocolo SSH abrange **autenticação, criptografia e integridade de dados** transmitidos por uma rede.

O que é SSH(Secure Shell)?

Autenticação

Determina de forma confiável a identidade de alguém. Se você tentar fazer login em uma conta em um computador remoto, o SSH solicitará uma prova digital de sua identidade.

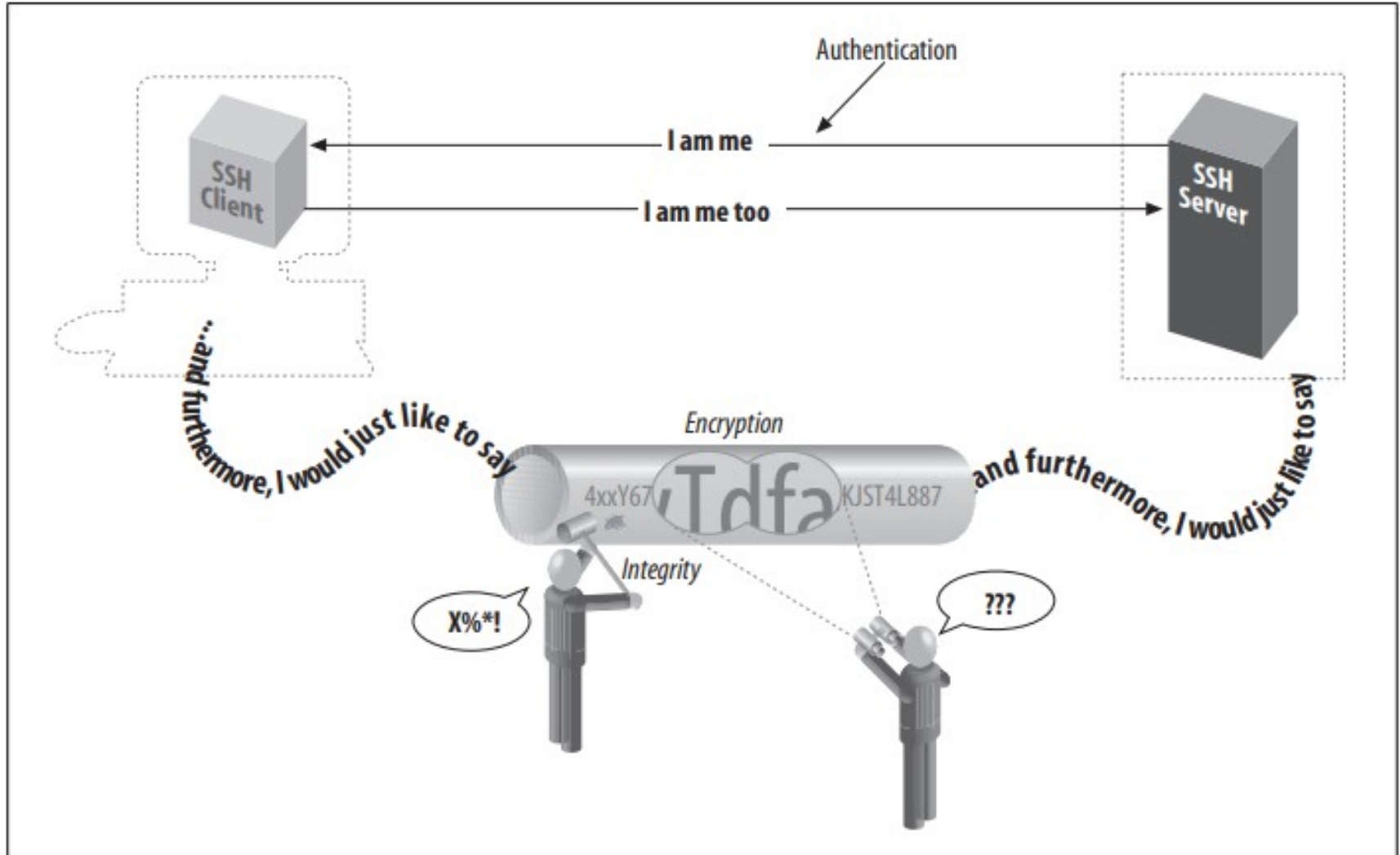
Criptografia

Embaralha os dados para que fiquem ineligíveis, exceto para os destinatários pretendidos. Isso protege seus dados à medida que passam pela rede.

Integridade

Garante que os dados que viajam pela rede cheguem inalterados. Se um terceiro capturar e modificar seus dados em trânsito, o SSH detectará esse fato.

PROTOCOLLO SSH



Instalação do OpenSSH

Linux

sudo apt-get install openssh-server

```
root@debian:~# apt install openssh-server
Lendo listas de pacotes... Pronto
Construindo árvore de dependências... Pronto
Lendo informação de estado... Pronto
openssh-server is already the newest version (1:8.4p1-5+deb11u3).
openssh-server configurado para instalar manualmente.
0 pacotes atualizados, 0 pacotes novos instalados, 0 a serem removidos e 0 não atualizados.
root@debian:~#
```

Configuração Básica do SSH

Arquivo de configuração: /etc/ssh/sshd_config

```
GNU nano 5.4                                sshd_config
# possible, but leave them commented.  Uncommented options override the
# default value.

Include /etc/ssh/sshd_config.d/*.conf

#Port 22
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
#PermitRootLogin prohibit-password
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

#PubkeyAuthentication yes

# Expect .ssh/authorized_keys2 to be disregarded by default in future.
#AuthorizedKeysFile .ssh/authorized_keys .ssh/authorized_keys2
```


Métodos de Autenticação

Servidor

```
root@debian:~# service ssh status
• ssh.service - OpenBSD Secure Shell server
  Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enabled)
  Active: inactive (dead) since Sun 2024-08-04 19:52:28 -03; 9s ago
    Docs: man:sshd(8)
           man:sshd_config(5)
  Process: 392 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
  Process: 439 ExecStart=/usr/sbin/sshd -D $SSHD_OPTS (code=exited, status=0/SUCCESS)
 Main PID: 439 (code=exited, status=0/SUCCESS)
   CPU: 19ms

ago 04 19:18:25 debian systemd[1]: Starting OpenBSD Secure Shell server...
ago 04 19:18:25 debian sshd[439]: Server listening on 0.0.0.0 port 22.
ago 04 19:18:25 debian sshd[439]: Server listening on :: port 22.
ago 04 19:18:25 debian systemd[1]: Started OpenBSD Secure Shell server.
ago 04 19:52:28 debian sshd[439]: Received signal 15; terminating.
ago 04 19:52:28 debian systemd[1]: Stopping OpenBSD Secure Shell server...
ago 04 19:52:28 debian systemd[1]: ssh.service: Succeeded.
ago 04 19:52:28 debian systemd[1]: Stopped OpenBSD Secure Shell server.
root@debian:~#
```

Ver o serviço do ssh está desativado no servidor

```
root@debian:~# ss -nltp
State      Recv-Q    Send-Q      Local Address:Port      Peer Address:Port      Process
root@debian:~# _
```

Ver se a porta padrão do protocolo ssh está aberta.

Métodos de Autenticação

Servidor

```
root@debian:~# service ssh start
root@debian:~# service ssh status
• ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enabled)
   Active: active (running) since Sun 2024-08-04 19:58:29 -03; 15s ago
     Docs: man:sshd(8)
           man:sshd_config(5)
   Process: 898 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
  Main PID: 899 (sshd)
    Tasks: 1 (limit: 1115)
   Memory: 1.1M
      CPU: 8ms
   CGroup: /system.slice/ssh.service
           └─899 sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups

ago 04 19:58:29 debian systemd[1]: Starting OpenBSD Secure Shell server...
ago 04 19:58:29 debian systemd[1]: Started OpenBSD Secure Shell server.
ago 04 19:58:29 debian sshd[899]: Server listening on 0.0.0.0 port 22.
ago 04 19:58:29 debian sshd[899]: Server listening on :: port 22.
root@debian:~# _
```

inicia o servidor ssh

ver o status está ativado

```
root@debian:~# ss -nltp
State  Recv-Q  Send-Q  Local Address:Port  Peer Address:Port  Process
LISTEN  0        128      0.0.0.0:22          0.0.0.0:*          users:(("sshd",pid=439,fd
LISTEN  0        128      [::]:22           [::]:*             users:(("sshd",pid=439,fd
root@debian:~#
```

Porta 22 aberta no servidor

Ver se a porta padrão do protocolo ssh está aberta.

Métodos de Autenticação

Senha: Simples

ssh user@host

SERVIDOR

```
Arquivo  Máquina  Visualizar  Entrada  Dispositivos  Ajuda
aluno@debian:~$ ip -br -c a
lo                UNKNOWN        127.0.0.1/8 ::1/128
enp0s3            UP             10.0.2.15/24 fe80::a00:27ff:febf:37e0/64
enp0s8            UP             192.168.10.30/24 fe80::a00:27ff:fe13:5d5c/64
aluno@debian:~$
```

Ip do SERVIDOR na placa de rede enp0s8

CLIENTE

```
File Actions Edit View Help
(jacinp@kali)-[~]
$ ip -br -c a
lo                UNKNOWN        127.0.0.1/8 ::1/128
eth0              UP             [REDACTED]/24 2804:14c:60:93db:a00:27ff:fe75:b42f/64
eth1              UP             192.168.10.20/24 fe80::a00:27ff:fe47:363a/64
eth2              DOWN
```

Ip do CLIENTE na placa de rede enp0s8

Métodos de Autenticação

Senha: Simples

SERVIDOR

ssh user@host

```
aluno@debian:~$ ls
documentos foto.png hash1 hash2 passwd.txt pastaSSH senha.txt teste.txt video.avi
aluno@debian:~$
```

CLIENTE

Documentos no usuário **aluno**

```
(jacinp@kali)-[~/Documents]
$ ssh aluno@192.168.10.30
The authenticity of host '192.168.10.30 (192.168.10.30)' can't be established.
ED25519 key fingerprint is SHA256:HXhrA8QYF6+aAdX6DXLn9JPGa5w1kiz18cEZmmWGgGQ.
This host key is known by the following other names/addresses:
  ~/.ssh/known_hosts:2: [hashed name]
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.10.30' (ED25519) to the list of known hosts.
aluno@192.168.10.30's password:
Linux debian 5.10.0-28-amd64 #1 SMP Debian 5.10.209-2 (2024-01-31) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sat Aug 3 17:07:58 2024 from 192.168.10.20
aluno@debian:~$ ls
documentos foto.png hash1 hash2 passwd.txt pastaSSH senha.txt teste.txt video.avi
aluno@debian:~$
```

usuário do cliente

conexão ssh

resposta ao ssh

solicitação da senha

Estamos dentro servidor

Arquivo que iremos transferir para o servidor via ssh

Métodos de Transferência de arquivos via ssh

scp <arquivo> user@host

SERVIDOR

```
aluno@debian ~/pastaSSH$
```

← pasta vazia

Usuario aluno do SERVIDOR com a pastaSSH vazia

CLIENTE

```
(jacinp@kali)-[~/Documents]
$ echo "arquivo transferido" > arquivoTransferido.txt

(jacinp@kali)-[~/Documents]
$ ls
arquivoTransferido.txt

(jacinp@kali)-[~/Documents]
$ cat arquivoTransferido.txt
arquivo transferido
```

← arquivo criado

← conteúdo do arquivo

Arquivo que iremos transferir para o servidor via ssh

Métodos de Transferência de arquivos via ssh

scp <arquivo> user@host

SERVIDOR

```
root@debian:/home/aluno/pastaSSH# ls  
arquivoTransferido.txt
```

Usuario aluno do SERVIDOR com a pastaSSH

CLIENTE

```
(jacinp@kali)-[~/Documents]  
$ scp arquivoTransferido.txt aluno@192.168.10.30:/home/aluno/pastaSSH/  
arquivoTransferido.txt
```

Arquivo que iremos transferir para o servidor via ssh

Métodos de Autenticação

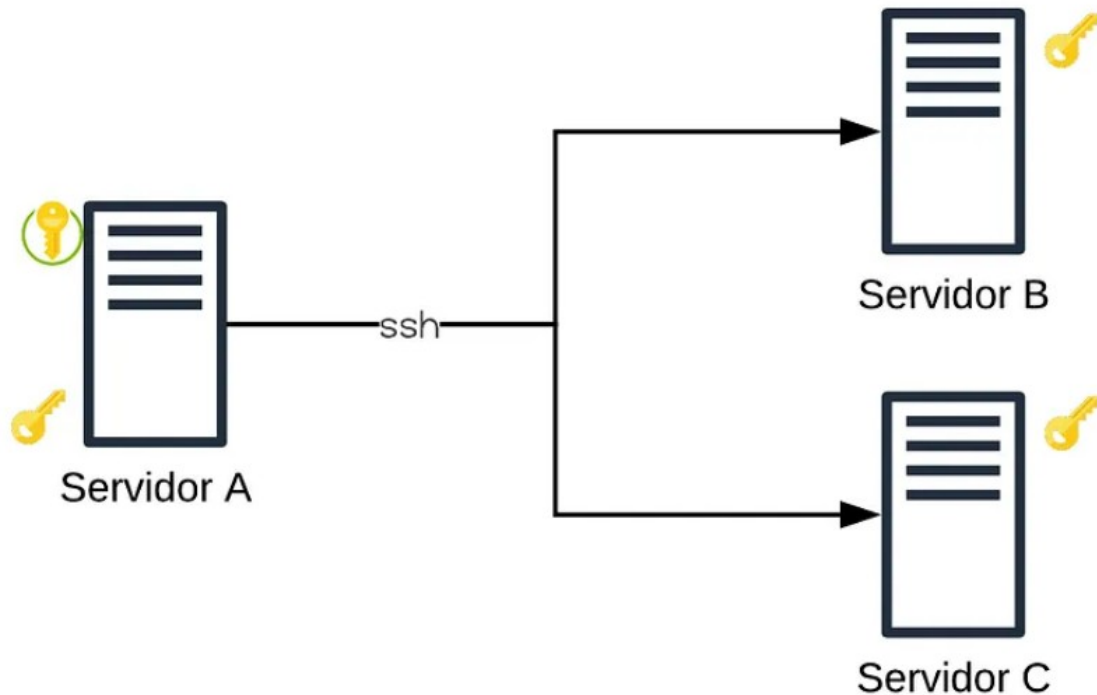
Chave Pública/Privada

Mais seguro

Métodos de Autenticação

chave privada
`~/.ssh/id_dsa`

chave pública
`~/.ssh/id_dsa.pub`



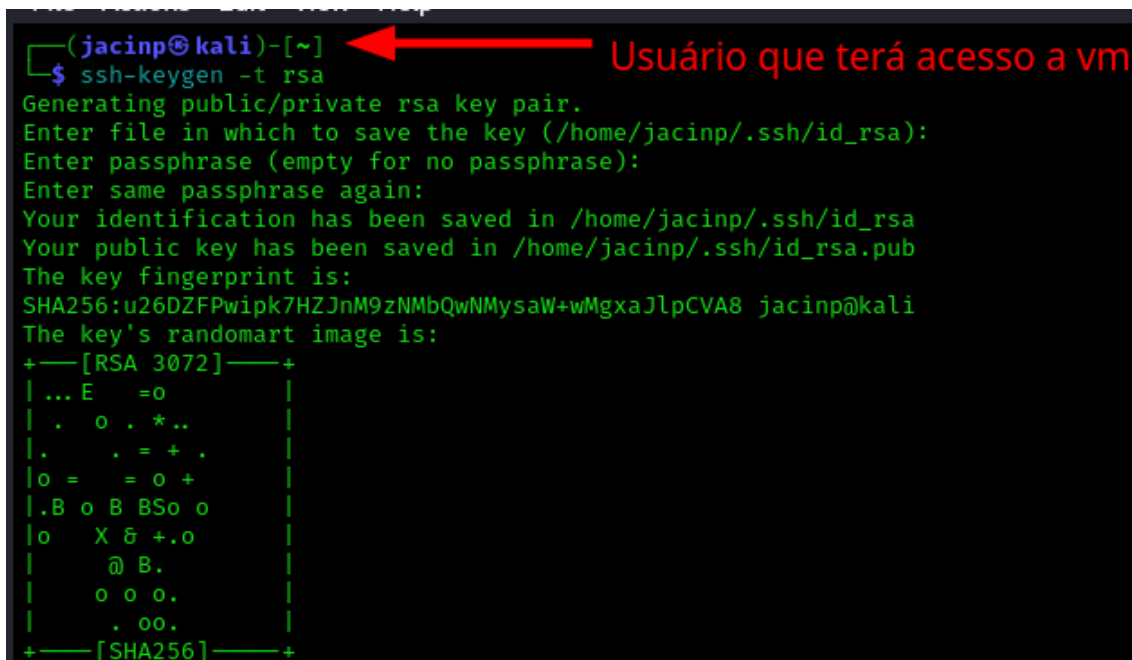
Métodos de Autenticação

Chave Pública/Privada

Gerar chave pública (SEM SENHA)

Na máquina onde será executado o comando remoto (ou scripts)

Utilize o **usuário** que terá acesso na máquina remota.



```
(jacinp@kali)-[~]  
$ ssh-keygen -t rsa  
Generating public/private rsa key pair.  
Enter file in which to save the key (/home/jacinp/.ssh/id_rsa):  
Enter passphrase (empty for no passphrase):  
Enter same passphrase again:  
Your identification has been saved in /home/jacinp/.ssh/id_rsa  
Your public key has been saved in /home/jacinp/.ssh/id_rsa.pub  
The key fingerprint is:  
SHA256:u26DZFPwipk7HZJnM9zNMbQwNMysaW+wMgxaJlpCVA8 jacinp@kali  
The key's randomart image is:  
+--[RSA 3072]--+  
| ... E    =0 |  
| .  o  . *.. |  
| .    . = + . |  
| o =    = o + |  
|.B o B BSo o |  
|o  X & +.o   |  
|    @ B.     |  
|    o o o.    |  
|    . oo.     |  
+--[SHA256]--+
```

Usuário que terá acesso a vm

Métodos de Autenticação

Chave Pública

```
(jacinp@kali)-[~/ssh]
$ ls
authorized_keys  id_rsa  id_rsa.pub
```

```
(jacinp@kali)-[~/ssh]
$ cat id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQgQCbuTryQgsNcrYx0T++a4jQf1W/2MMQTGITARSB/eMcvccA63k0t/HwudvaF9sa0iF3P91dS3HMMBli
3pwf94sy5NDpWHD7NIqU3b9zn8yPODCvW+JZ0jn9x0gV0CNkEIYp7WpsJRuwtu3D1J0VKoEAetVYy/9j9UWsuMzYCMWU5xRogUU9b7rm8+XL5o4dEUB
aMf1bWaV6BqmYaSp0r0EvvidBtCUkJ1sou59RCMDlBr08W7lEJsJVIONmP10dJd1W0vBEfwsYE7mNok=jacinp@kali
```

Copiar a chave pública do cliente para o SERVER

CLIENTE

```
(jacinp@kali)-[~/ssh]
$ ssh-copy-id aluno@192.168.10.30
/usr/bin/ssh-copy-id: INFO: source of key(s) to be installed: "/home/jacinp/.ssh/id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
aluno@192.168.10.30's password:

Number of key(s) added: 1
Now try logging into the machine, with: "ssh 'aluno@192.168.10.30'"
and check to make sure that only the key(s) you wanted were added.
```

Copiando a chave pública para o SERVER

Chave pública adicionada

Métodos de Autenticação

Chave Pública

Copiar a chave pública do cliente para o SERVER

CLIENTE

```
(jacinp@kali)-[~/ssh]
└─$ ssh-copy-id aluno@192.168.10.30
```

Copiando a chave pública para o SERVER

```
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/jacinp/.ssh/id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
aluno@192.168.10.30's password:

Number of key(s) added: 1
```

Chave pública adicionada

```
Now try logging into the machine, with:  "ssh 'aluno@192.168.10.30'"
and check to make sure that only the key(s) you wanted were added.
```

SERVER

```
root@debian:/home/aluno/.ssh# ls
authorized_keys
```

Chave pública copiada no SERVER

```
root@debian:/home/aluno/.ssh# cat authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQGCbUtRyQgsNcrYx0T++a4jQf1W/2MMQTGITARSB/eMcvccA63k0t/HwudvaF9
0iF3P91dS3HMMB1iaEQXE1fMfNrYCpxnCblRlgzmH01h58Nt5REe0ypTY6V8q9CHYDI/NK5jc8Ywxyb1/CRPFFwn9BjfHykID
CFx8ldxr7/VCI17ivrLjinkD+AIWzA8T8mVG3pwf94sy5NDpWHD7NIqU3b9zn8yPODCvW+JZ0jN9x0gVOCNkEIYp7WpsJRuwtu
1J0VKoEAEtVYy/9j9UXoxM+cXGMXU5xRogUU9b7rm8+XL5o4dEUBiQ0e21L1bPjs6aMDu9bhom752p9fMpU7J5dabj7qNNSjb4
r3ZViTFGGMONovS201QJc2YM+uZS1imQJrekza1Gn4EpF+e0eJGgp5m2+0PuA4YxLPfWeWMraMf1bWaV6BqmYaSp0r0EvvidBt
kJ1sou59RCdM1Br08W71EJsJVIONmP10dJd1W0vBEfwsYE7mNok=jacinp@kali
```

```
root@debian:/home/aluno/.ssh# _
```

Métodos de Autenticação

Chave Pública

Altera a permissão de usuário

SERVER

```
root@debian:/home/aluno/.ssh# chmod -v 600 authorized_keys  
modo de 'authorized_keys' mantido como 0600 (rw-----)  
root@debian:/home/aluno/.ssh#
```

O primeiro 6 dá ao proprietário do arquivo (o usuário) permissão de leitura e escrita (4+2=6).

O segundo 0 remove todas as permissões do grupo.

O terceiro 0 remove todas as permissões de outros usuários.

Métodos de Autenticação

Chave Pública

CLIENTE

ssh aluno@192.168.10.30

```
(jacinp@kali)-[~/.ssh]
$ ssh aluno@192.168.10.30
Linux debian 5.10.0-28-amd64 #1 SMP Debian 5.10.209-2 (2024-01-31) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Aug  6 21:26:29 2024 from 192.168.10.20
aluno@debian:~$
```

ACESSO SSH SEM CREDENCIAIS COM SUCESSO!

Práticas recomendadas para proteger o acesso SSH

A importância de proteger o serviço SSH e o papel do arquivo **sshd_config** na segurança do servidor.

/etc/ssh/sshd_config

Práticas recomendadas para proteger o acesso SSH

Proibir Logins de Root

- **Configuração:** *PermitRootLogin no*
- **Benefício:** Impede logins diretos como root, reduzindo o risco de comprometer a conta superusuária.

```
# Authentication:
#PasswordAuthentication no
#LoginGraceTime 2m
#PermitRootLogin prohibit-password
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10
```

nano /etc/ssh/sshd_config

Práticas recomendadas para proteger o acesso SSH

Restringir Autenticação por Senha

Configuração: *PasswordAuthentication no*

Benefício: Força o uso de chaves SSH, oferecendo maior segurança.

```
# Authentication:
PasswordAuthentication no
#LoginGraceTime 2m
#PermitRootLogin prohibit-password
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10
```

nano /etc/ssh/sshd_config

Práticas recomendadas para proteger o acesso SSH

Desabilitar Autenticação por Host-Based

Configuração: *HostbasedAuthentication no*

Benefício: Elimina a autenticação baseada no nome do host.

nano /etc/ssh/sshd_config

```
# For this to work you will also need host keys in /etc/ssh/ssh_known_hosts
HostbasedAuthentication no ←
# Change to yes if you don't trust ~/.ssh/known_hosts for
# HostbasedAuthentication
#IgnoreUserKnownHosts no
# Don't read the user's ~/.rhosts and ~/.shosts files
#IgnoreRhosts yes
```

Práticas recomendadas para proteger o acesso SSH

Desabilitar Autenticação por Host-Based

Configuração: *HostbasedAuthentication no*

Benefício: Elimina a autenticação baseada no nome do host.

nano /etc/ssh/sshd_config

```
# For this to work you will also need host keys in /etc/ssh/ssh_known_hosts
HostbasedAuthentication no ←
# Change to yes if you don't trust ~/.ssh/known_hosts for
# HostbasedAuthentication
#IgnoreUserKnownHosts no
# Don't read the user's ~/.rhosts and ~/.shosts files
#IgnoreRhosts yes
```

Práticas recomendadas para proteger o acesso SSH

Alterar a Porta SSH Padrão

Configuração: *Port 2222*

Benefício: Reduz a exposição a ataques automatizados.

```
Port 2222
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::
```

`nano /etc/ssh/sshd_config`

Práticas recomendadas para proteger o acesso SSH

Desativar o Forwarding de X11

Configuração: *X11Forwarding no*

Benefício: Previne o forwarding de sessões X11

```
# Example of overriding settings on a per-user basis
#Match User anoncvs
-       X11Forwarding no
#       AllowTcpForwarding no
#       PermitTTY no
#       ForceCommand cvs server
```

nano /etc/ssh/sshd_config


Práticas recomendadas para proteger o acesso SSH

Limitar Tentativas de Login

Configuração: *MaxAuthTries 3*

Benefício: Ajuda a prevenir ataques de força bruta.

```
# Authentication:
#PasswordAuthentication no
#LoginGraceTime 2m
#PermitRootLogin prohibit-password
#StrictModes yes
MaxAuthTries 3
#MaxSessions 10
```



nano /etc/ssh/sshd_config

Práticas recomendadas para proteger o acesso SSH

Configurar Timeouts de Sessão

Configuração: ClientAliveInterval 300
ClientAliveCountMax 0

Benefício: Desconecta sessões ociosas para reduzir riscos.

```
#Compression delayed  
ClientAliveInterval 300  
ClientAliveCountMax 3
```

nano /etc/ssh/sshd_config