

INSTITUTO FEDERAL

Norte de Minas Gerais

Campus Januária

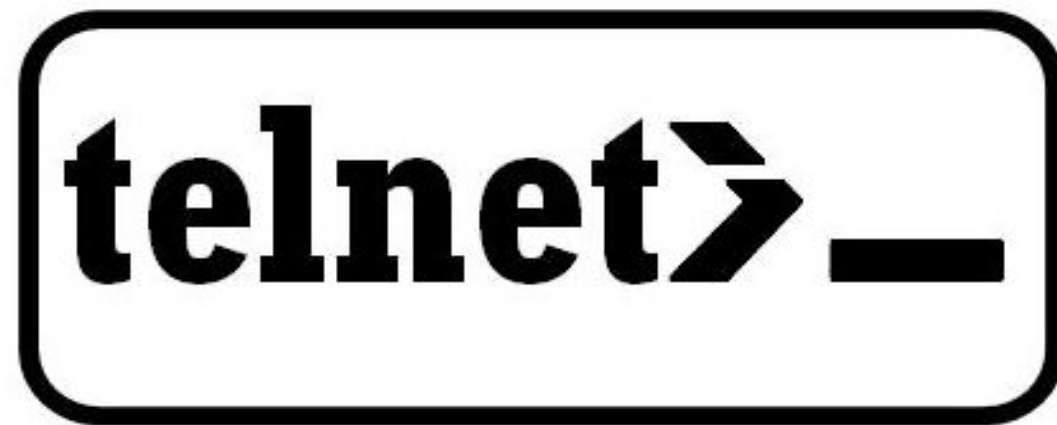
Admin. Serviços de Redes

- Acesso Remoto -



INSTITUTO FEDERAL
Norte de Minas Gerais
Campus Januária

Acesso Remoto





INSTITUTO FEDERAL
Norte de Minas Gerais
Campus Januária

Telnet

TELNET



Telnet

- **Telnet** é um dos protocolos padrões da Internet para acesso remoto a hosts (p.ex. servidores).
- Acesso remoto permite que um usuário efetue comandos e altere configurações em *hosts* distantes, através da **visualização do terminal remoto em sua própria estação**.
- Entretanto, o **TELNET não utiliza criptografia** na comunicação entre a máquina local e remota, o que pode causar um grave problema de segurança.



Telnet

- Por padrão, o serviço Telnet baseia-se em conexões TCP através da Porta 23... Ou outra porta definida em:

```
# /etc/services
```

- Devido às suas limitações de segurança, **é usado somente em casos muito específicos.**





Telnet

■ Instalação

```
# apt-get install telnetd
```

■ Configuração

```
# /etc/inetd.conf
```

■ Ativação do Servidor

```
# service openbsd-inetd start
```



Segurança de Ambiente

ATENÇÃO

***Por questões de segurança
NUNCA
faça um acesso remoto
diretamente para o usuário root.***



Gestão de Usuários

- Crie um novo usuário no Server:

```
# adduser nome_usuario
```

- Conceder permissões de root ao usuário (se necessário)

```
# usermod -aG sudo nome_usuario
```

- Trocar para novo usuário (*Switch User*)

```
# su nome_usuario
```

- Trocar para usuário root

```
# su -
```




Laboratório 08-1

- Crie um novo usuário no Server:

```
# adduser nome_usuario
```

- A partir da VM Cliente, acesse o Server remotamente.

```
# telnet 200.0.0.1
```

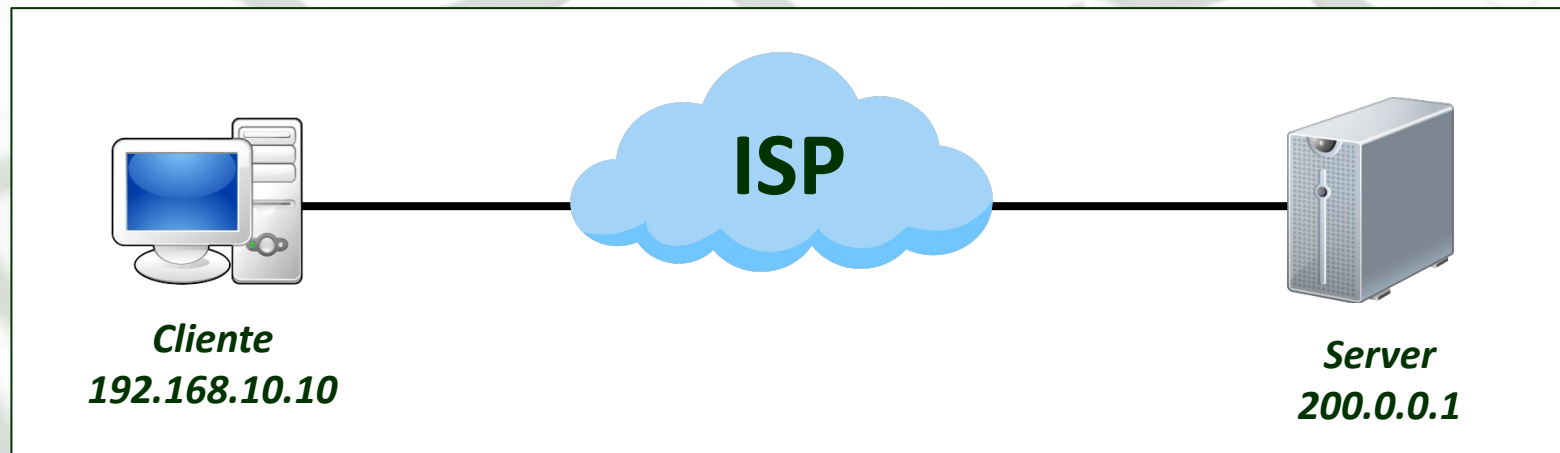




Laboratório 08-1

- No ISP, utilize um *Sniffer* + Analisador de Pacotes para inspecionar como as credenciais de autenticação são transmitidas entre o Cliente e o Server.

```
# tcpdump -w escutaSSH.pcap
```

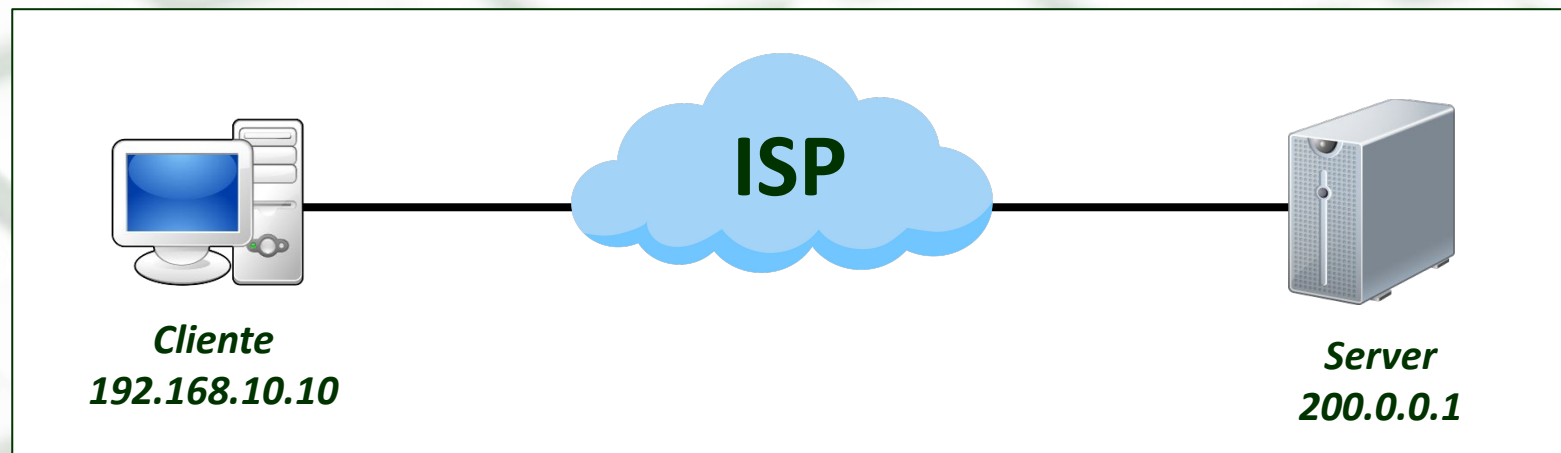




Laboratório 08-1

- Façamos outro teste...
 - Inicie o servidor HTTP (Apache) do server...

```
# /etc/init.d/apache2 start
```



- Inicie uma conexão TELNET para a porta 80, e verifique...

```
# telnet 200.0.0.1 80  
> GET /
```



Laboratório 08-1

- Pelo navegador, acesse o site “**neverssl.com**”

NeverSSL

What?

This website is for when you try to open Facebook, Google, Amazon, etc on a wifi network, and nothing happens. Type “<http://neverssl.com>” into your browser’s url bar, and you’ll be able to log on.

How?

neverssl.com will never use SSL (also known as TLS). No encryption, no strong authentication, no [HSTS](#), no HTTP/2.0, just plain old unencrypted HTTP and forever stuck in the dark ages of internet security.

Why?

Normally, that’s a bad idea. You should always use SSL and secure encryption when possible. In fact, it’s such a bad idea that most websites are now using https by default.

And that’s great, but it also means that if you’re relying on poorly-behaved wifi networks, it can be hard to get online. Secure browsers and websites using https make it impossible for those wifi networks to send you to a login or payment page. Basically, those networks can’t tap into your connection just like attackers can’t. Modern browsers are so good that they can remember when a website supports encryption and even if you type in the website name, they’ll use https.



Laboratório 08-1

- Agora, observe pelo Telnet...

```
telnet neverssl.com 80
Arquivo  Editar  Ver  Pesquisar  Terminal  Ajuda
$> telnet neverssl.com 80
Trying 34.223.124.45...
Connected to neverssl.com.
Escape character is '^]'.
GET /
```




Laboratório 08-1

- Agora, observe pelo Telnet...

```
adriano@adriano-pc:~  
Arquivo  Editar  Ver  Pesquisar  Terminal  Ajuda  
$> telnet neverssl.com 80  
Trying 34.223.124.45...  
Connected to neverssl.com.  
Escape character is '^]'.  
GET /  
<html>  
    <head>  
        <title>NeverSSL - Connecting ... </title>  
        <style>  
            body {  
                font-family: Montserrat, helvetica, arial, sans-serif;  
                font-size: 16x;  
                color: #444444;  
                margin: 0;  
            }  
            h2 {  
                font-weight: 700;  
                font-size: 1.6em;  
                margin-top: 30px;  
            }  
        </style>  
    </head>  
</html>
```




Serviços sem Criptografia

Por razões óbvias, **protocolos inseguros** como o **Telnet** que oferece serviço de acesso remoto, e outros serviços, como **HTTP** (serviço WEB), **FTP** (transferência de arquivos), e **DNS** (resolução de nomes), estão caindo em desuso e sendo substituídos por versões correspondentes, que **adotam algum sistema de criptografia moderna** (**SSH, HTTPS, SCP/SFTP e DNS-Sec, respectivamente**), garantindo segurança ao tráfego gerado por essas aplicações.



INSTITUTO FEDERAL
Norte de Minas Gerais
Campus Januária

Fundamentos de Criptografia



Alice

Olá Bob!



Bob



INSTITUTO FEDERAL
Norte de Minas Gerais
Campus Januária

Fundamentos de Criptografia



Alice

Olá Bob!

Algoritmo de Criptografia: Cada letra da mensagem deve avançar N posições à frente...



Bob



INSTITUTO FEDERAL
Norte de Minas Gerais
Campus Januária

Fundamentos de Criptografia



Alice

Olá Bob!

Algoritmo de Criptografia: Cada letra da mensagem deve avançar N posições à frente...

Chave da Criptografia: $N = 3$



Bob



Fundamentos de Criptografia



Alice

Olá Bob!

Algoritmo de Criptografia: Cada letra da mensagem deve avançar N posições à frente...

Chave da Criptografia: $N = 3$

Rod Ere!



Bob



INSTITUTO FEDERAL
Norte de Minas Gerais
Campus Januária

Fundamentos de Criptografia



Alice

Olá Bob!

Algoritmo de Criptografia: Cada letra da mensagem deve avançar N posições à frente...

Chave da Criptografia: $N = 3$

Rod Ere!



Bob

O que Bob precisa saber para conseguir ler a mensagem de Alice?



Fundamentos de Criptografia



Alice

Olá Bob!

Algoritmo de Criptografia: Cada letra da mensagem deve avançar N posições à frente...

Chave da Criptografia: $N = 3$

Como Alice informa a chave para Bob SEM que Darth também a veja?

Rod Ere!



Bob



Fundamentos de Criptografia

- Existem dois modelos básicos de criptografia...

- **Criptografia Simétrica**

- Como mostrado no exemplo anterior...
- A chave usada para criptografar deve ser a mesma para descriptografar (*simetria*).

- **Criptografia Assimétrica**

- Arquitetura de Chaves Públicas (e Privadas)
- Cada ente possui um par de chaves inter-relacionadas matematicamente.



INSTITUTO FEDERAL
Norte de Minas Gerais
Campus Januária

Fundamentos de Criptografia

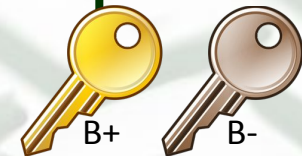


Alice



Olá Bob!

Chave Pública de Bob

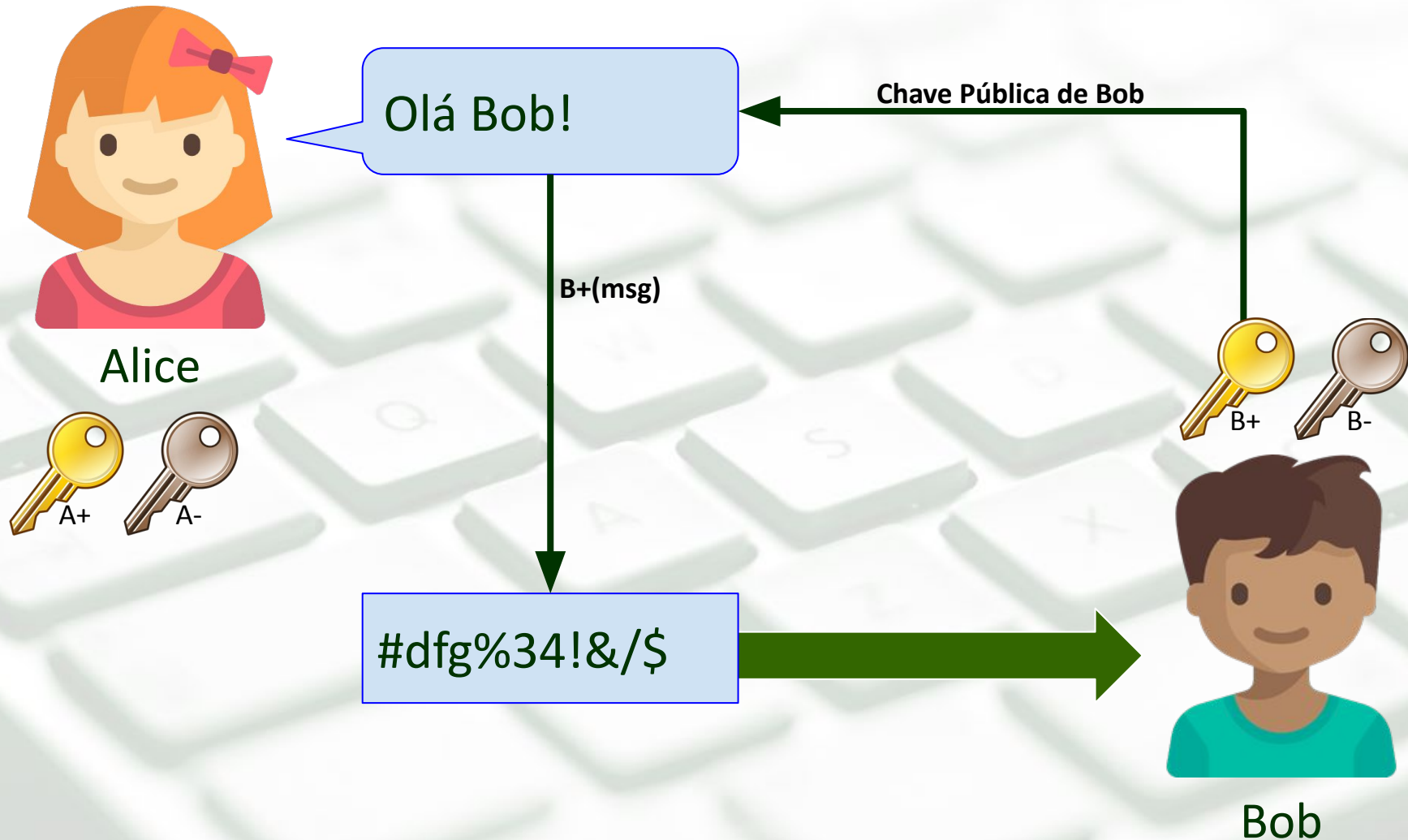


Bob



INSTITUTO FEDERAL
Norte de Minas Gerais
Campus Januária

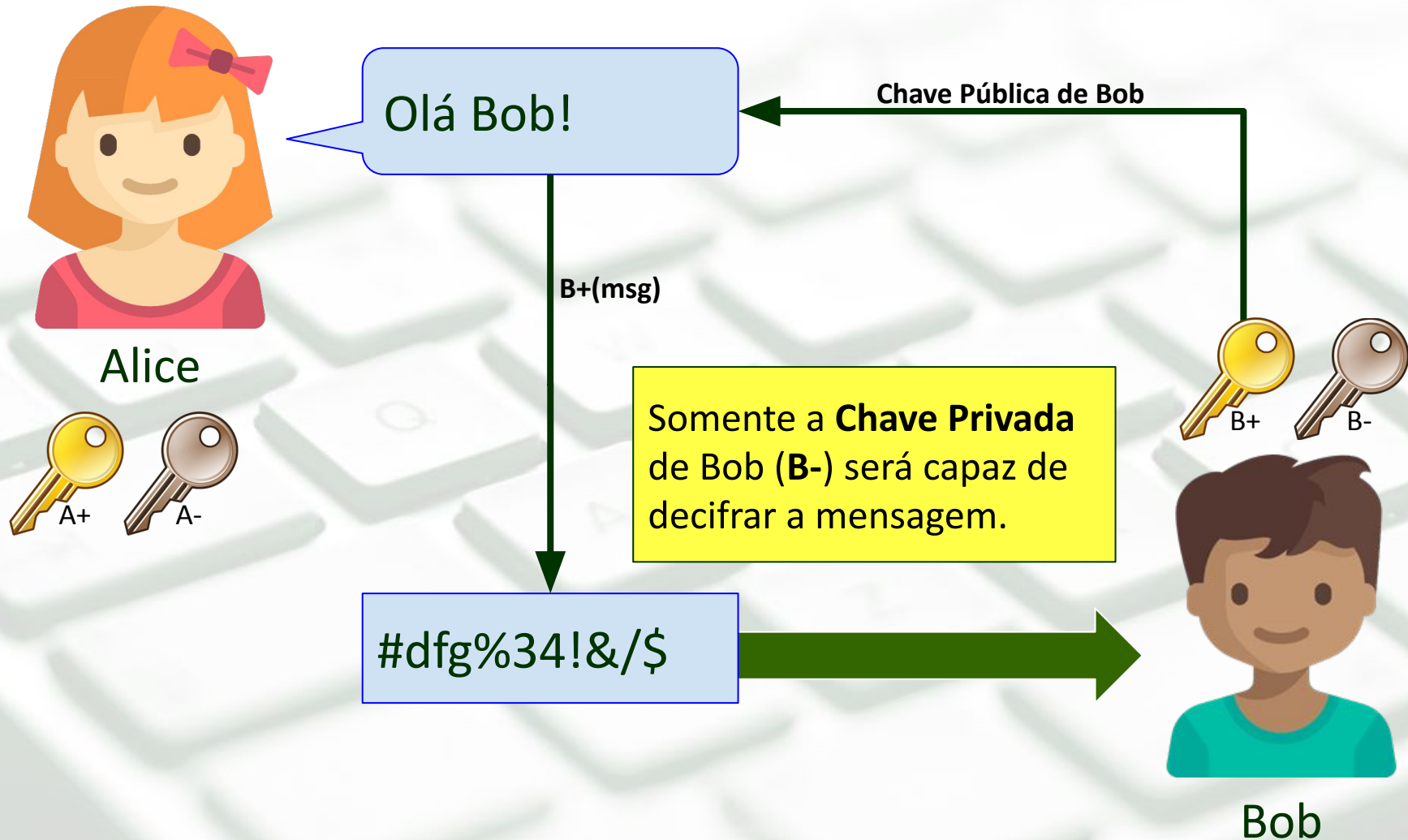
Fundamentos de Criptografia





INSTITUTO FEDERAL
Norte de Minas Gerais
Campus Januária

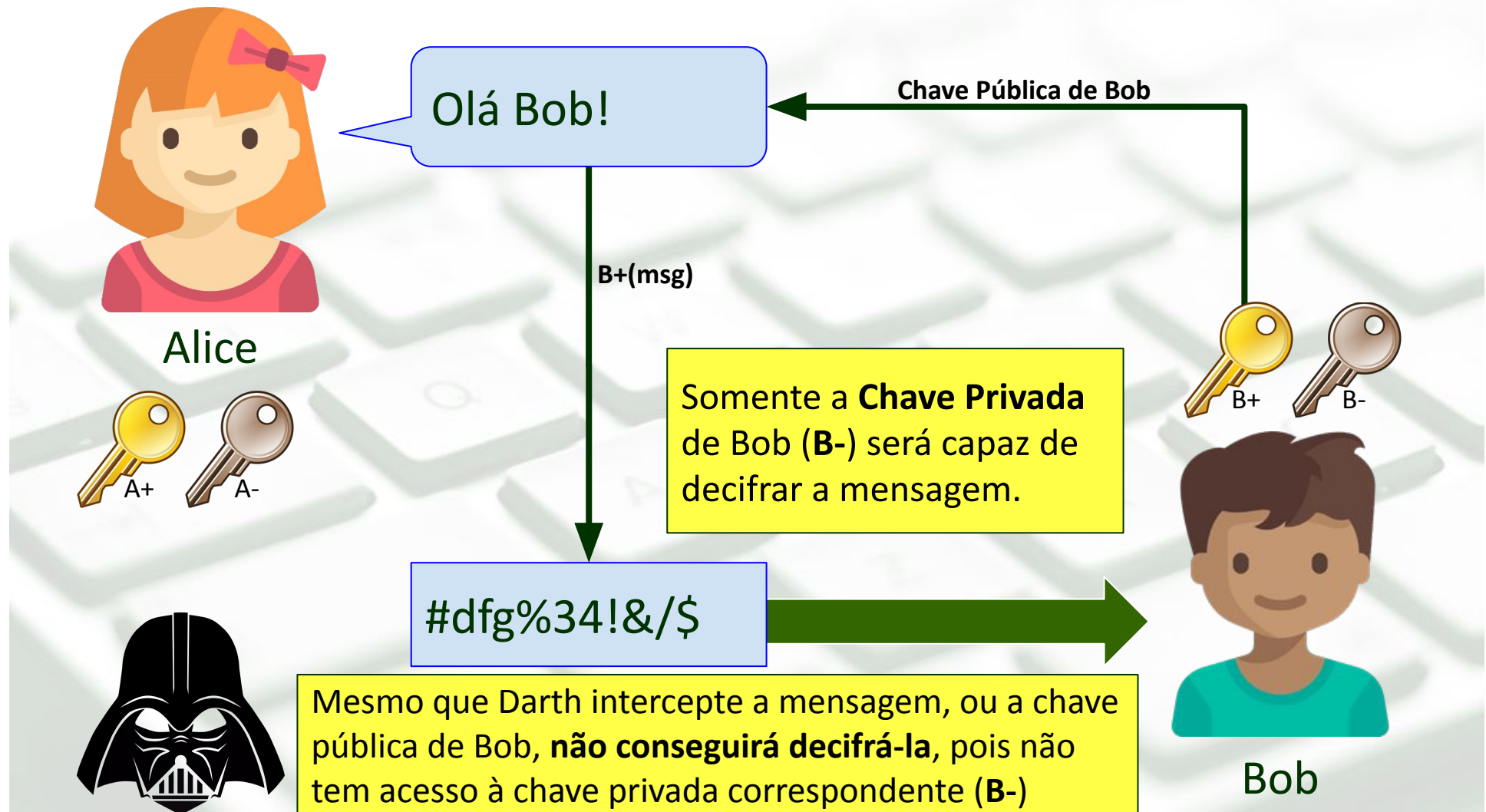
Fundamentos de Criptografia





INSTITUTO FEDERAL
Norte de Minas Gerais
Campus Januária

Fundamentos de Criptografia





Secure Shell

SSH



SSH

- **SSH (Secure SHell)** também é um **protocolo padrão** da arquitetura TCP/IP para acesso remoto a *hosts*.
- Ao contrário do Telnet, o SSH implementa **comunicação criptografada** entre o cliente e o servidor remoto.
- A autenticação é baseada em **Criptografia Assimétrica: Algoritmo RSA** (*Rivest, Shamir e Adleman*).

Maior Segurança



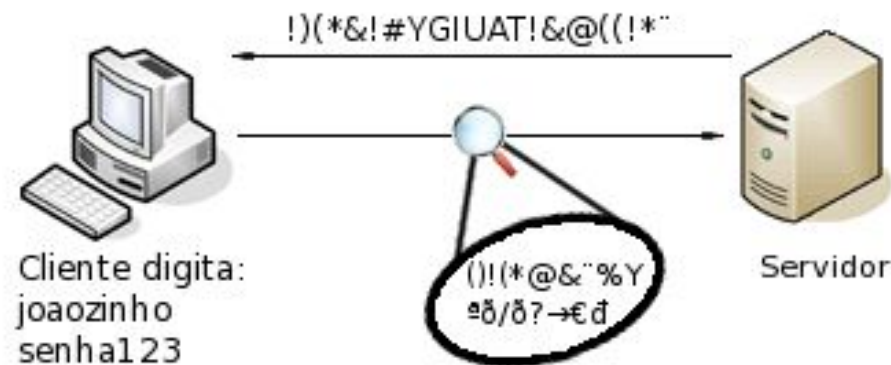


Telnet vs. SSH

Sessão de login sem criptografia como no telnet



Sessão de login criptografada como no SSH





INSTITUTO FEDERAL
Norte de Minas Gerais
Campus Januária

Autenticação SSH



Chave Privada C-

**Cliente
SSH**

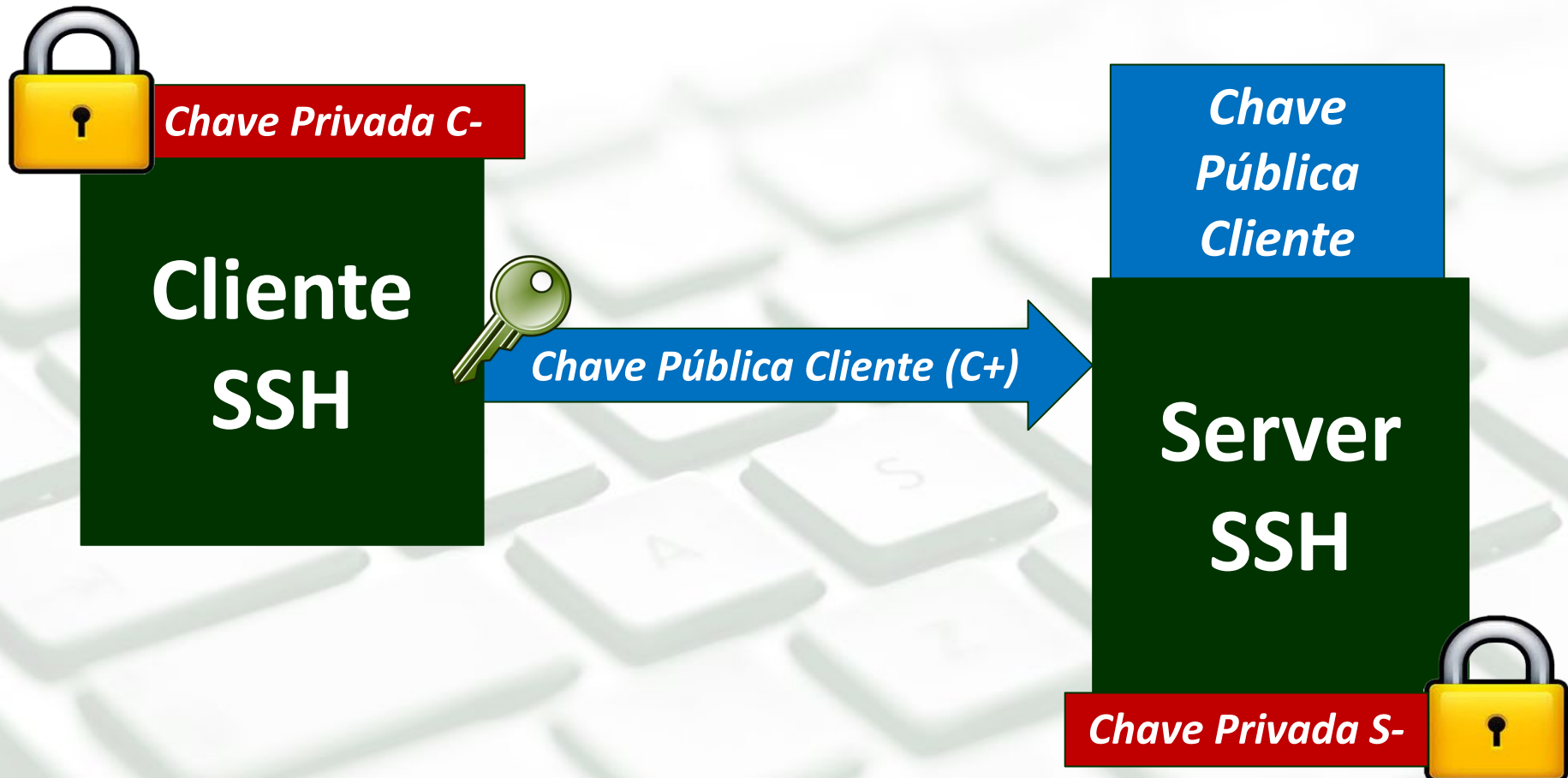
**Server
SSH**

Chave Privada S-



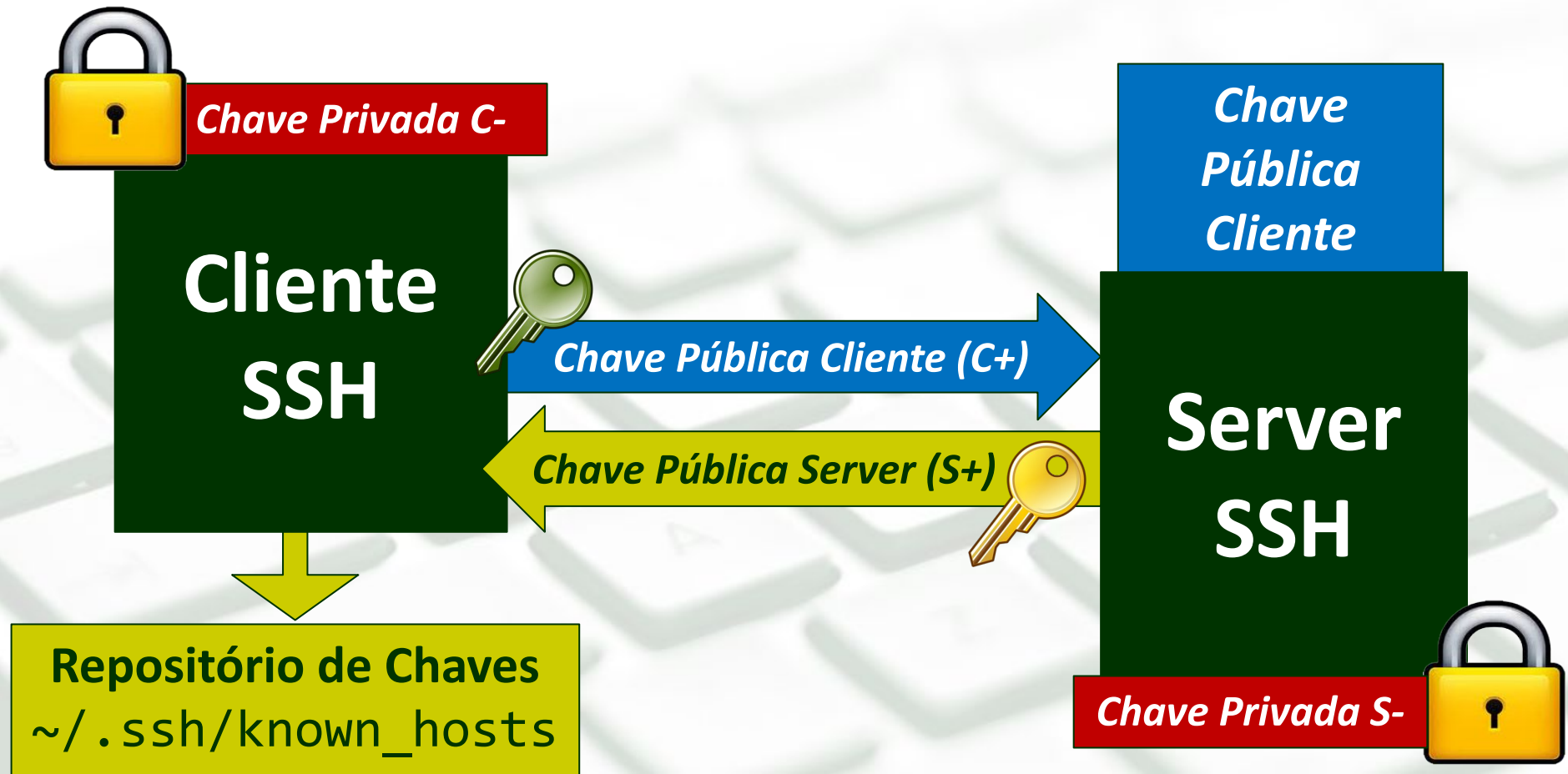


Autenticação SSH



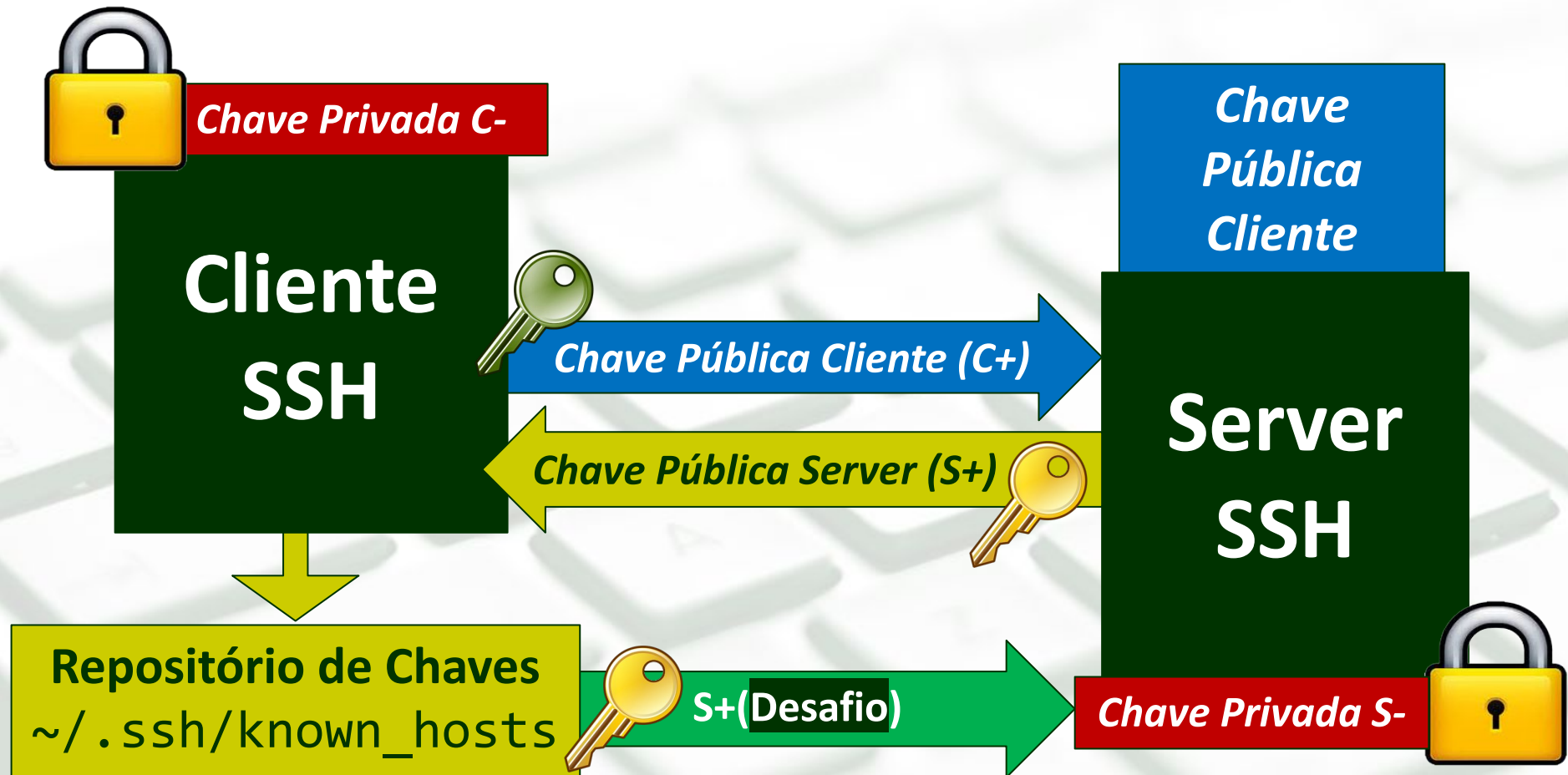


Autenticação SSH



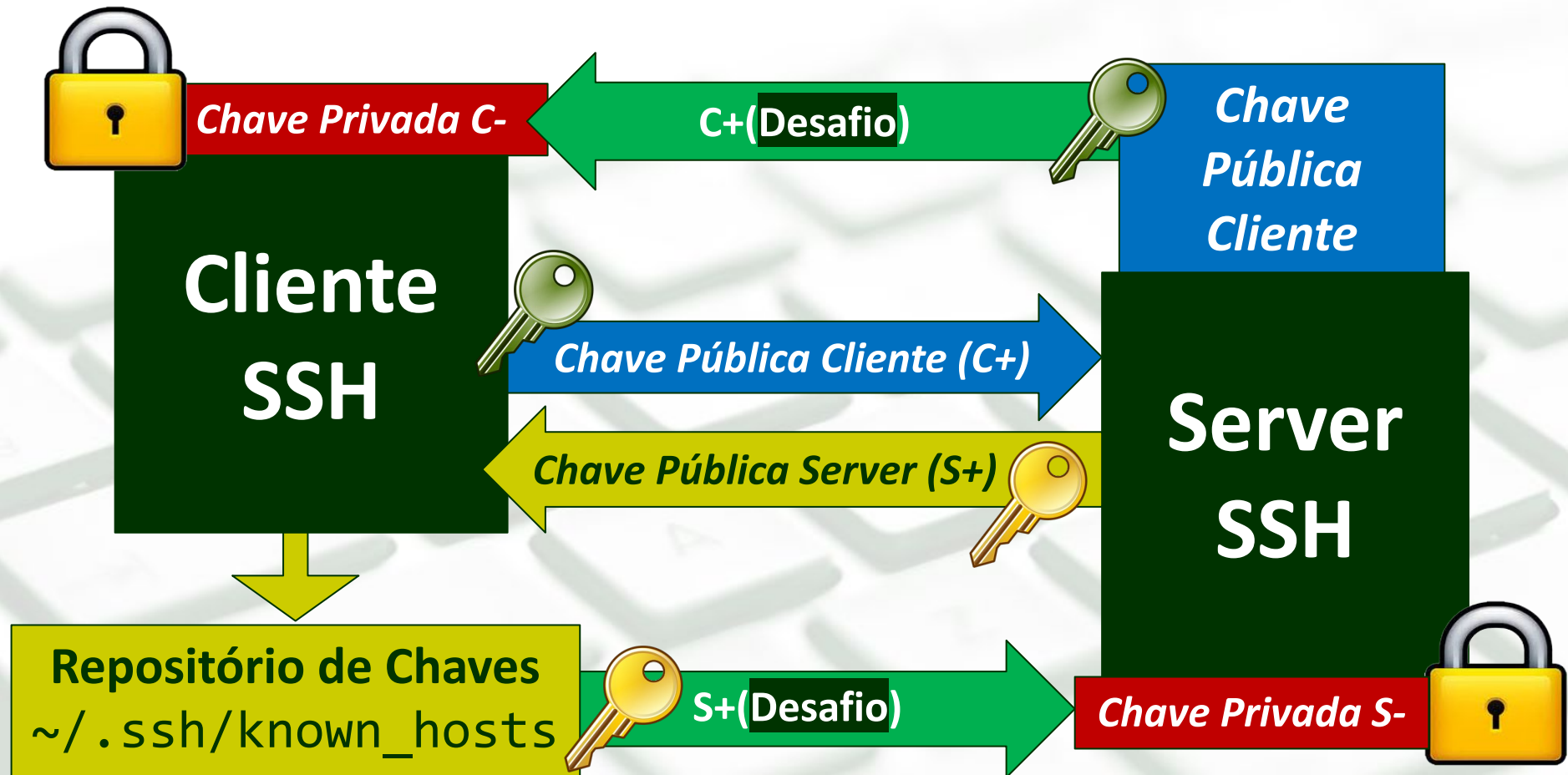


Autenticação SSH



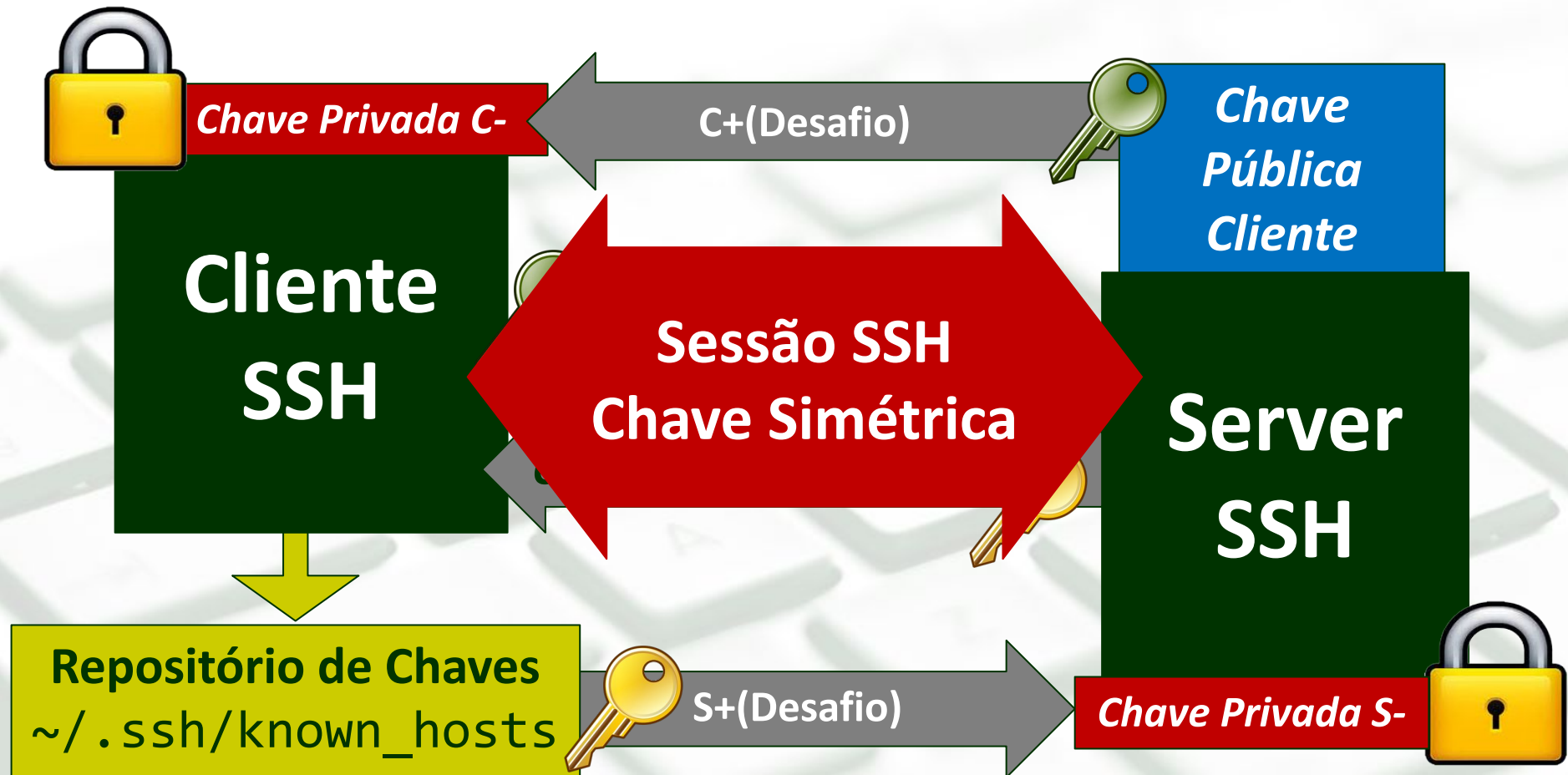


Autenticação SSH



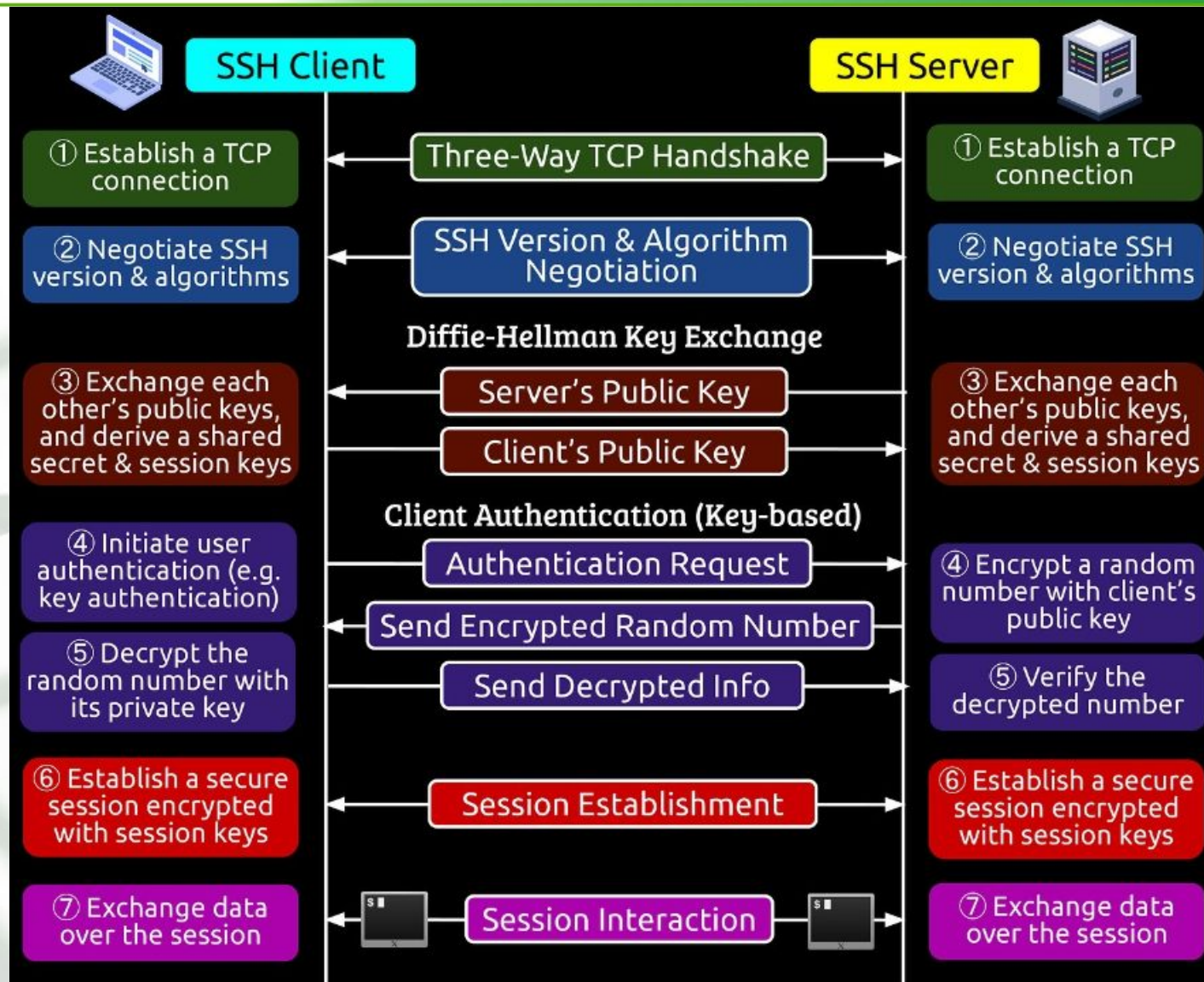


Autenticação SSH





Autenticação SSH





SSH

■ Instalação:

```
# apt-get install openssh-server  
# apt-get install openssh-client
```

■ Configuração:

```
# /etc/ssh/sshd_config (server)  
# /etc/ssh/ssh_config (client)
```

■ Ativação

```
# /etc/init.d/ssh start
```



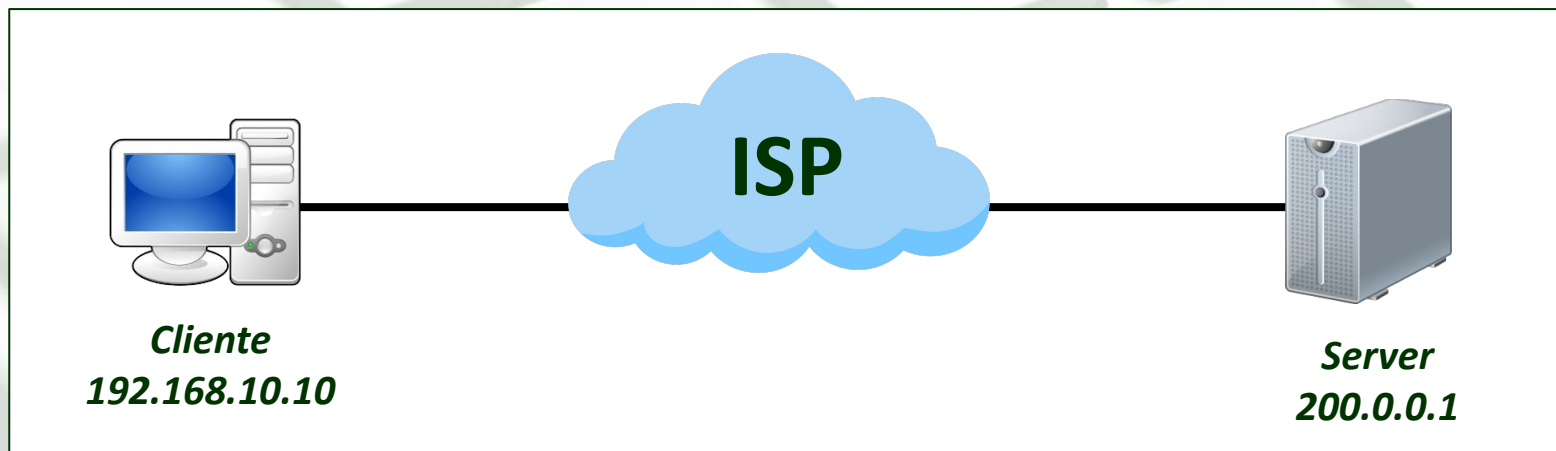

Laboratório 08-2

- Crie um usuário no Server.

```
# adduser nome_usuario
```

- Acesse remotamente o Server.

```
# ssh nome_usuario@200.0.0.1  
# su -
```





Chaves de Autenticação

- Abra o arquivo abaixo no **Cliente** e veja a identificação da chave pública do **Server**:

```
# nano ~/.ssh/known_hosts
```

- Exclua uma chave pública do repositório do Cliente:

```
# ssh-keygen -R 200.0.0.1
```

- As chaves dos **hosts** estão localizadas em:

```
# /etc/ssh/ssh_host_rsa_key ←  
# /etc/ssh/ssh_host_rsa_key.pub
```

Sempre mantenha as chaves privadas bem protegidas!



Autenticação por Chaves

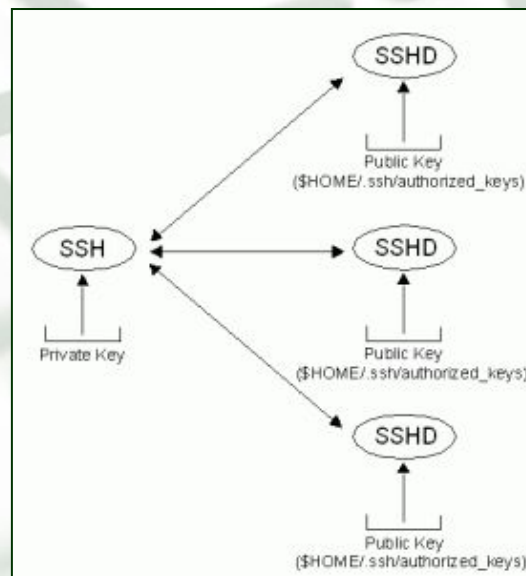
- **Autenticação por Chaves** ou **Autenticação de Duas Vias** é um método ainda mais seguro para fazer a autenticação entre duas máquinas remotas.
- Nesse método, a autenticação é feita através de **chaves assimétricas geradas pelo usuário** - ao invés de usar a sua própria senha de acesso.
 - *Evita roubo de senhas por “olhudos” de plantão e Ataques de brute-force.*
- A **chave pública** gerada pelo usuário deve ser instalada no servidor, e a **chave privada** (armazenada localmente) pode ser (ou não) protegida por uma ***passphrase***.



Autenticação por Chaves

- Porque a autenticação por chaves é mais segura?

Para que um invasor consiga ter acesso indevido a um servidor é necessário que ele roube a chave privada do usuário, e ainda conheça a passphrase que a decodifica.





Chaves de Autenticação

- Para gerar um par de chaves utilize o comando:

```
# ssh-keygen
```

- As chaves serão salvas no diretório “home” do usuário:

```
# ~/.ssh/id_rsa
```

```
# ~/.ssh/id_rsa.pub
```

- Instale a chave pública no servidor:

```
# ssh-copy-id -i ~/.ssh/id_rsa.pub login@server
```




Chaves de Autenticação

- Recomenda-se (por simplificação) que o nome do usuário no servidor remoto seja o mesmo nome de usuário do cliente.
- As chaves públicas autorizadas a acessar uma determinada conta de usuário no servidor, são instaladas no arquivo:

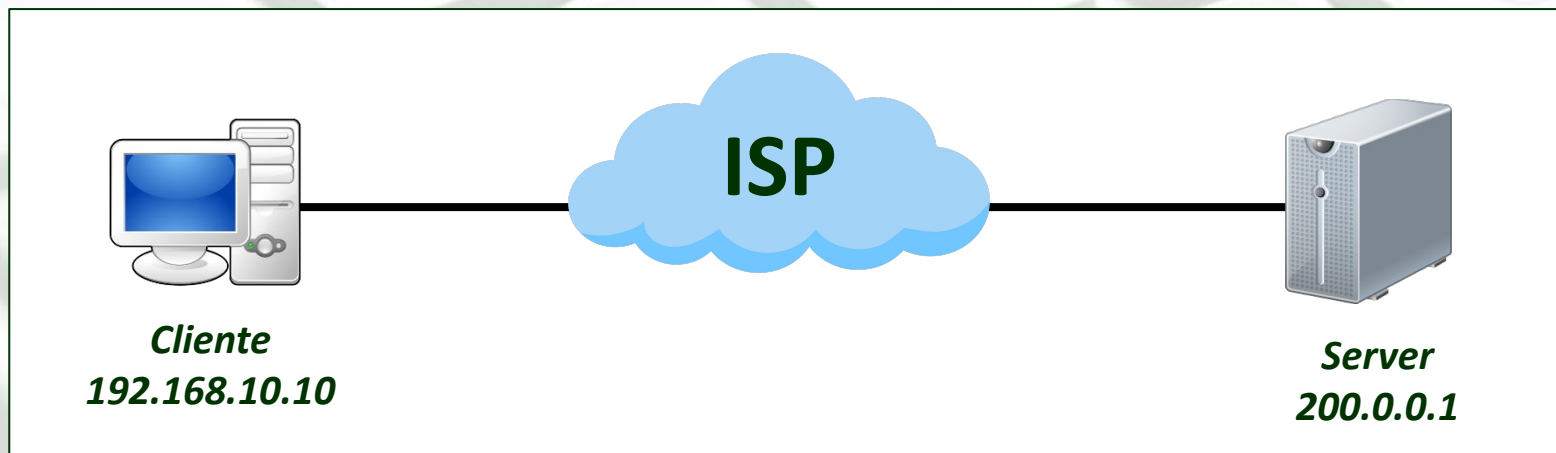
```
# ~/.ssh/authorized_keys
```

** Também é possível instalar (copiar) a chave pública do cliente diretamente no arquivo `authorized_keys`, ao invés de usar o comando `ssh-copy-id`.*



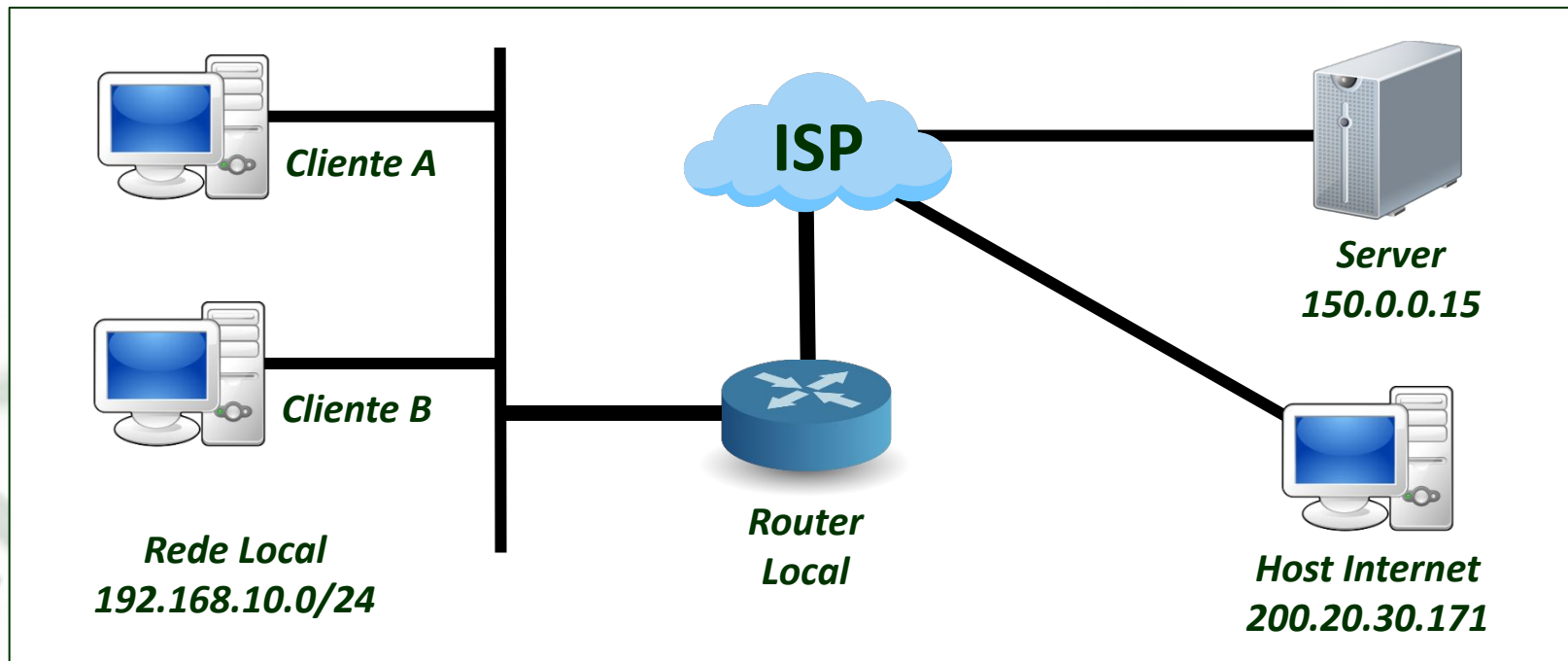
Laboratório 08-3

- Crie um par de chaves de autenticação SSH para o usuário “Admin”.
- Quando este usuário tentar acessar remotamente sua conta no Server, não deverá ser solicitada nenhuma senha (a autenticação será via chaves assimétricas).





Laboratório 08-4



- A partir do “Cliente A”, use o SSH (com autenticação via chaves) para subir a porta 80 em modo escuta no Server (use o netcat).
- A partir de um “Host qualquer na Internet”, use o SSH (login+senha) para “invadir” o “Router Local” e criar um sniffer (captura de tráfego) para gerar e salvar em arquivo todos os pacotes que trafegam ali.
- A partir do “Cliente B”, conecte-se a porta 80 do server, enviando e recebendo informações.
- Através do Wireshark, inspecione o arquivo gerado pelo *sniffer* do Router Local.



Boas Práticas SSH

```
# nano /etc/ssh/sshd_config
```

```
# Alterar a porta padrão (22) do serviço
```

```
Port 1025
```

```
# Endereço de escuta da conexão
```

```
ListenAddress 192.168.10.1
```

```
# Desabilite login do usuário root (apenas por chaves já é padrão!)
```

```
PermitRootLogin prohibit-password || no
```

```
# Desabilite login de usuários por senha (força que a autenticação aconteça apenas por chaves)
```

```
PasswordAuthentication no
```

```
# Nº Tentativas de conexão sem sucesso (5), % de recusa de chamadas após as 5 iniciais, Nº Máximo Total até bloqueio total.
```

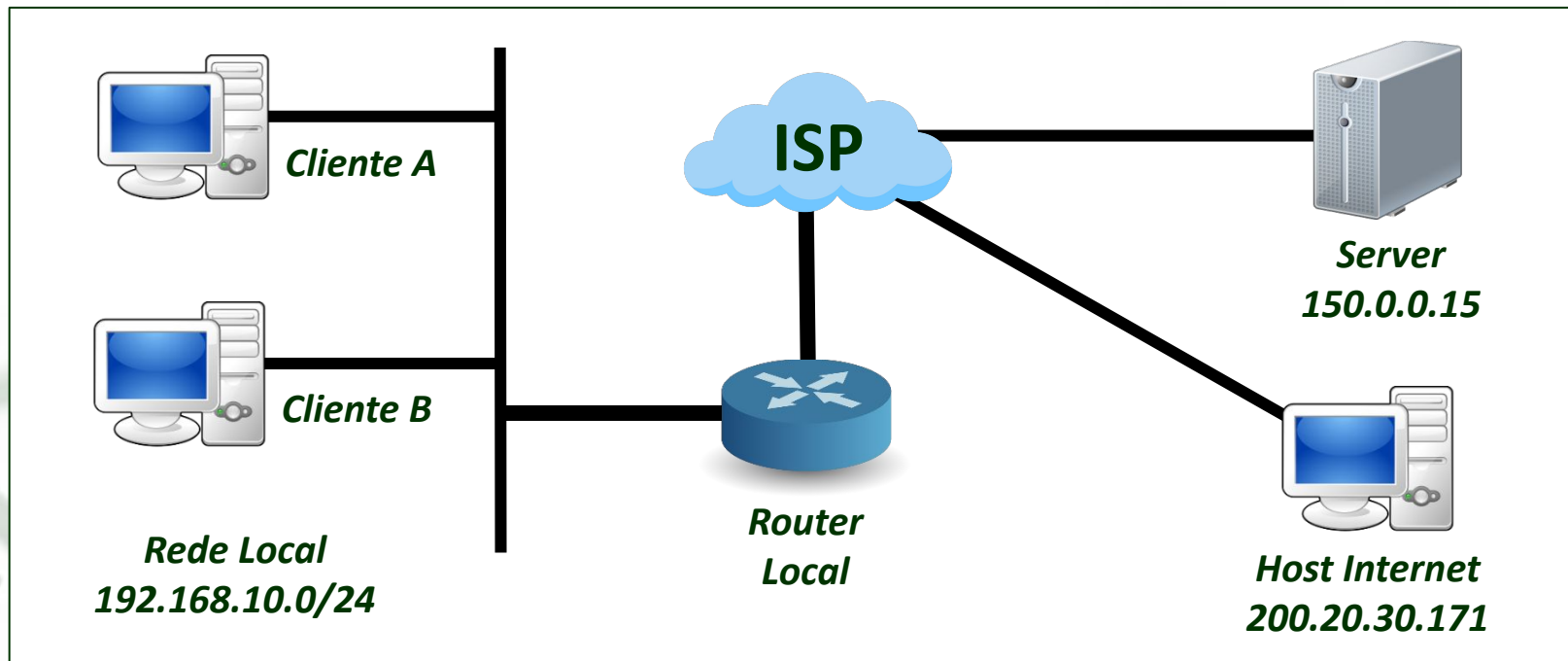
```
MaxStartups 5:70:8
```

```
# Permite apenas conexões para os usuários explicitamente indicados.
```

```
AllowUsers adminIfnmg
```



Laboratório 08-5



- Aprimore a segurança do Lab08-4:
 - Conexões SSH para o “Router Local” devem ser aceitas APENAS provenientes da Rede Local, e específicas para um usuário bem definido (p.ex.:adminIfnmg)
 - Conexões SSH para o “Server” devem ser feitas EXCLUSIVAMENTE via chaves assimétricas, porta padrão deve ser a 5001, e sendo 5 tentativas no máximo.