

**Relatório Assist**

**Disciplina:** ASSIST

**Trabalho Elaborado por:** José Nuno Mota Teixeira 1200941

Ano letivo

2023/2024

**Índice**

[**1 - US 910** 2](#_Toc154509484)

[**1.1 - Objetivo Final – Conexão a servidor via SSH sem recurso a credenciais de Login** 2](#_Toc154509485)

[**1º passo - Gerar um Par de Chaves RSA** 2](#_Toc154509486)

[**2º Passo - Copiar a Chave Pública para o Servidor** 2](#_Toc154509487)

[**3 º Passo - Confirmar ficheiro de configuração sshd** 3](#_Toc154509488)

[**4º Passo - Conexão** 3](#_Toc154509489)

[**2 - US 930** 4](#_Toc154509490)

[**2.1 - Objetivo Final - Reposição de Backup e Confirmação** 4](#_Toc154509491)

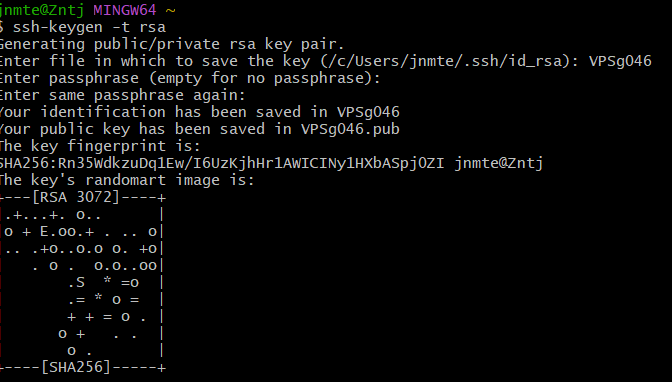
[**1º Passo – Criação de Script** 4](#_Toc154509492)

# **1 - US 910**

## **1.1 - Objetivo Final – Conexão a servidor via SSH sem recurso a credenciais de Login**

### **1º passo - Gerar um Par de Chaves RSA**

Usar comando “ssh-keygen -t rsa”.



**Explicação do comando** “ssh-keygen -t rsa”

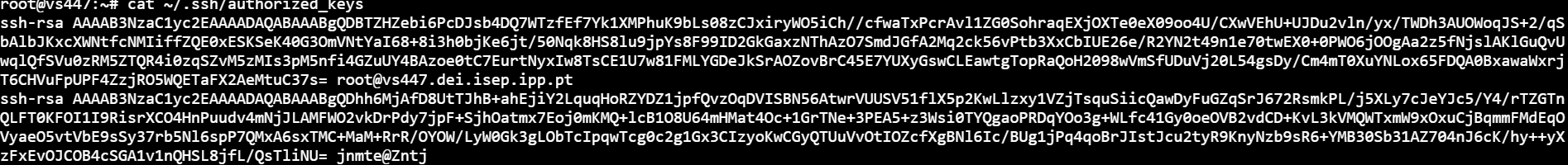
* ssh-keygen: Este é o programa principal usado para criar chaves de autenticação para SSH. Está incluído na maioria dos sistemas Unix e Linux como parte do pacote OpenSSH.
* -t rsa: Este argumento especifica o tipo de chave a ser gerada. No caso, rsa refere-se ao algoritmo RSA (Rivest-Shamir-Adleman), um dos algoritmos de criptografia mais comuns usados para a criação de pares de chaves públicas e privadas.

Se não se especificar um local, este comando usará um local padrão (geralmente ~/.ssh/id\_rsa para a chave privada e ~/.ssh/id\_rsa.pub para a chave pública).

### **2º Passo - Copiar a Chave Pública para o Servidor**

Copiar a chave pública gerada, para o servidor ao qual deseja conectar-se. Este passo pode ser feito de várias formas.

Decidi utilizar uma forma mais manual. Conectei-me ao servidor por ssh com password, colei, ainda, no ficheiro “~/.ssh/authorized\_keys” a chave pública criada anteriormente:

Todo este processo seria possível executar com o comando:

ssh-copy-id [root@vs447.dei.isep.ipp.pt](mailto:root@vs447.dei.isep.ipp.pt)

Não optei por esta solução devido a estar a utilizar uma máquina Windows, a qual cria conflito com a localização da chave pública gerada, por isso decidi utilizar a prática acima indicada.

### **3 º Passo - Confirmar ficheiro de configuração sshd**

Confirmar, se a linha PermitRootLogin no ficheiro de configuração “etc/ssh/sshd\_config” está em “yes” e para uma maior segurança desabilitar a password:

**Uma imagem com texto, Tipo de letra, captura de ecrã, Gráficos

Descrição gerada automaticamente**

### **4º Passo - Conexão**

**Comando:**

ssh -v -i C:/Users/jnmte/VPSg046 [root@vs447.dei.isep.ipp.pt](mailto:root@vs447.dei.isep.ipp.pt)

**Explicação do comando**

-v: Esta opção ativa o modo de depuração (verbose mode) no SSH. Quando usado, o SSH exibe informações detalhadas sobre o processo de conexão. Isso é útil para diagnosticar problemas de conexão, autenticação e configuração.

-i C:/Users/jnmte/VPSg046:

* -i: Esta opção é usada para especificar o caminho da chave privada que será usada para a autenticação com o servidor.

root@vs447.dei.isep.ipp.pt:

* root: Este é o nome de usuário que você está tentando usar para fazer login no servidor remoto. Neste caso, é o utilizador 'root', que é o superusuário em sistemas baseados em Unix/Linux.
* vs447.dei.isep.ipp.pt: Este é o endereço do servidor.

Uma imagem com texto, captura de ecrã

Descrição gerada automaticamente

# **2 - US 930**

## **2.1 - Objetivo Final - Reposição de Backup e Confirmação**

### **1º Passo – Criação de Script**

Para obter a reposição de Backup e a sua devida confirmação, foi criado um script em que os utilizadores administradores têm permissão de execução.

**Script:** O objetivo deste código é automatizar a restauração de backups de um banco de dados MongoDB e verificar a integridade do backup restaurado. O código permite ao usuário escolher um backup disponível.



O script fornece feedback ao usuário em cada etapa e relata qualquer erro encontrado durante o processo de restauração e verificação. Isso ajuda a garantir que os backups sejam restaurados com sucesso. Para a confirmação do processo de restauro, verifica-se a presença e consistência da coleção "buildings". É também possível ser verificado o número total de coleções repostas.



**Segurança**: Em todos os momentos de utilização direta com a base de dados os dados de login estão guardados no ficheiro config.sh onde as permissões são extremamente restritas de forma a preservar a **confidencialidade, usando “chmod 600 config.sh”**

Existem duas “Strings Connections” para a mesma base de dados, isto deve-se ao “mongorestore” e ao “mongoexport” utilizarem uma sintaxe diferente.

**Explicação detalhada das etapas no script**

* Listagem de Backups:

O script começa por listar todos os backups disponíveis no diretório de backup especificado.

* Seleção de Backup:

O usuário é solicitado a escolher um backup digitando o número correspondente ao backup desejado.

* Restauração do Backup:

O script utiliza o utilitário mongorestore para restaurar o backup selecionado no banco de dados MongoDB especificado nas configurações.

* Verificação de Restauração Bem-Sucedida:

É verificado se a restauração foi concluída com sucesso, verificando o código de saída do comando mongorestore. Se a restauração falhar, o script sai com uma mensagem de erro.

* Verificação da Existência da Coleção "buildings":

O script utiliza o mongoexport para verificar se a coleção "buildings" existe no banco de dados restaurado após a restauração.