# Configure Private Key Authentication with ISE

## Contents

## Introduction

This document describes how to create a private secure shell (SSH) key for authentication to the CLI on Identity Secure Engine (ISE).

## Prerequisites

### Requirements

Cisco recommends that you have knowledge of these topics:

- Repository in ISE.
- Certificate authentication.

### Components Used

The information in this document is based on these software and hardware versions:

- ISE 3.3 patch 3
- Windows 10
- MacOS X
- SSH client Putty

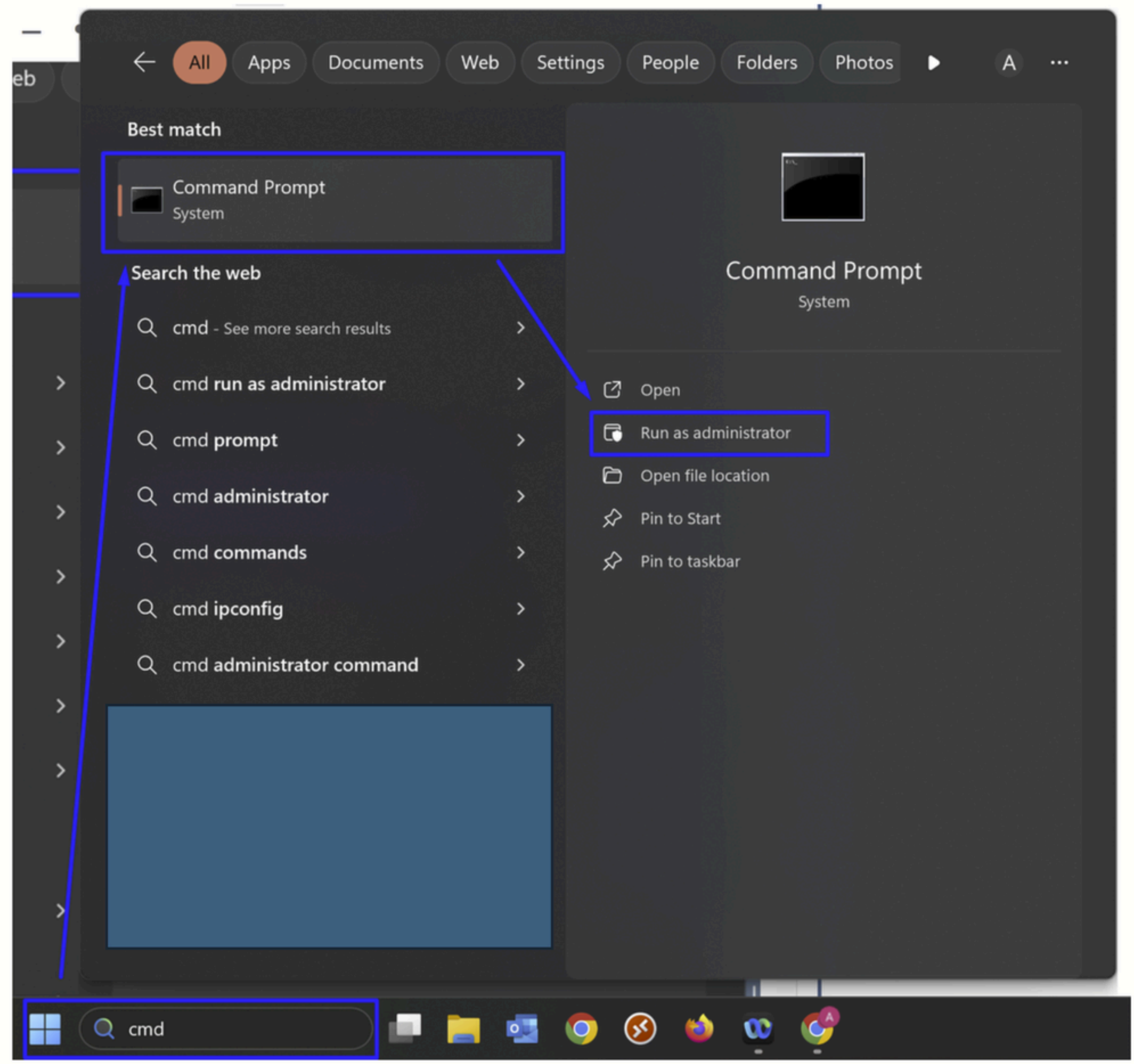
The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

# Configure

## Create the private and public keys in Windows

Click on the Search icon located on the taskbar:

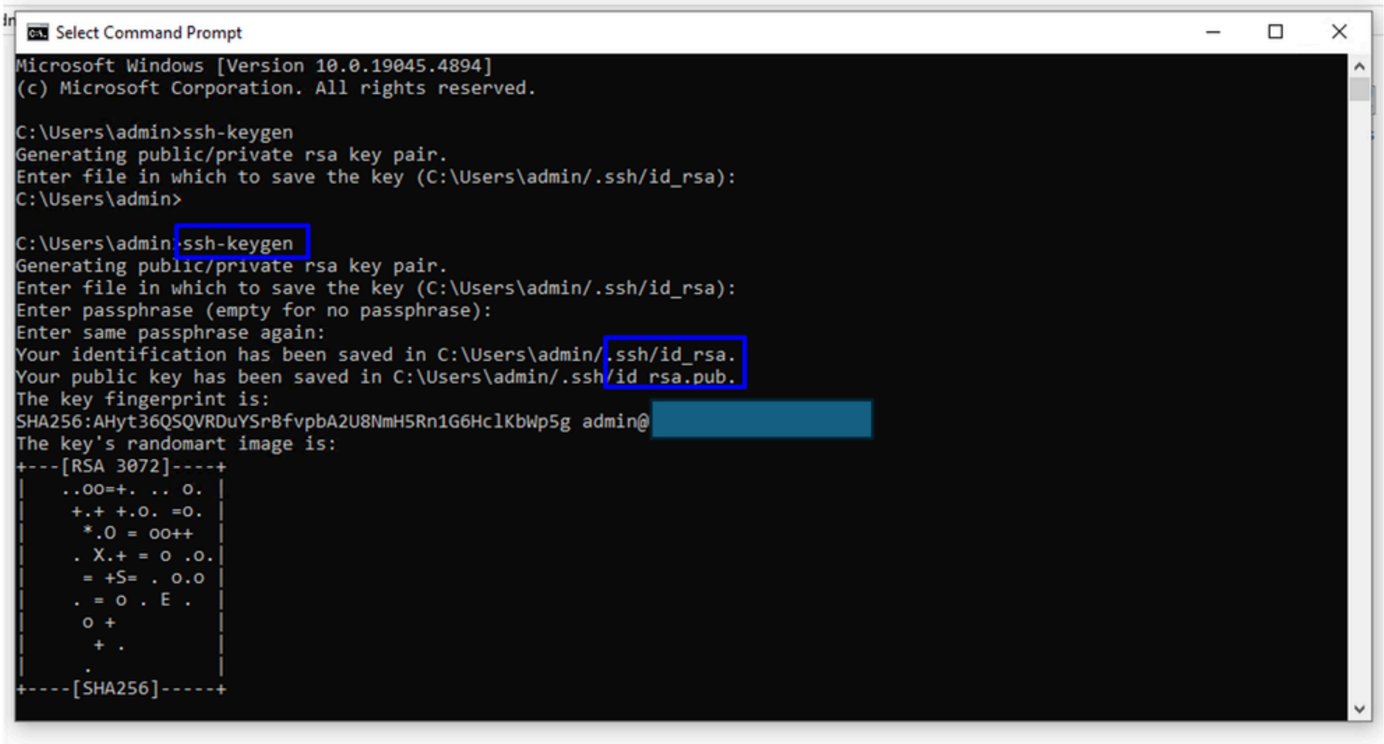
- Type cmd into the search bar
- In the search results, right-click on Command Prompt and select Run as administrator. This ensures you have the necessary permissions to execute commands



- Execute the next command:

```
ssh-keygen
```

- This prompts you to enter the encryption key twice. Please save it, because this is for being authenticated against the ISE as the new password. After that, this results in the creation of two files, the private (id_rsa) and the public (id_rsa.pub) keys, then. Save the files in one directory. For example, the default one was used

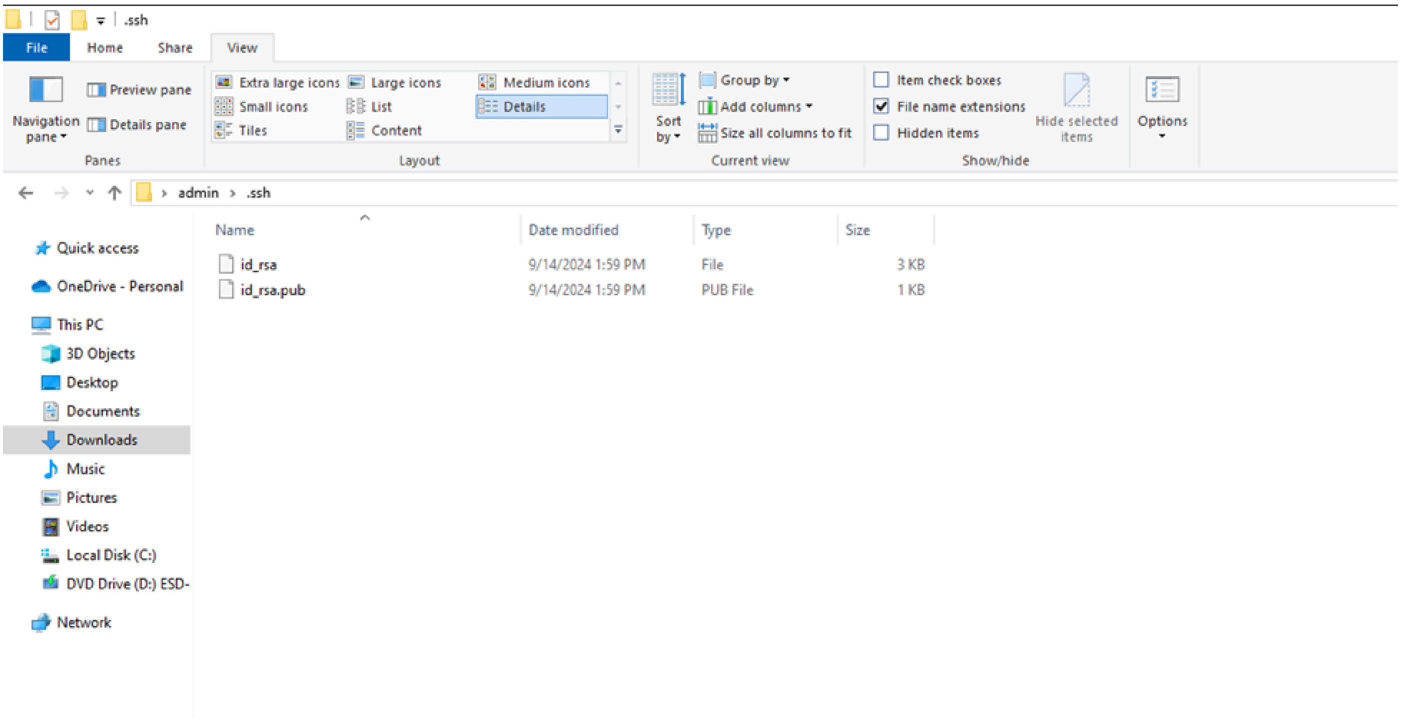


```
Microsoft Windows [Version 10.0.19045.4894]
(c) Microsoft Corporation. All rights reserved.

C:\Users\admin>ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (C:\Users\admin/.ssh/id_rsa):
C:\Users\admin>

C:\Users\admin>ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (C:\Users\admin/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in C:\Users\admin/.ssh/id_rsa.
Your public key has been saved in C:\Users\admin/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:AHyt36QSQVRDuYSrBfvpbA2U8NmH5Rn1G6HclKbWp5g admin@[redacted]
The key's randomart image is:
+---[RSA 3072]----+
|   ..oo=+. .. o. |
|    +.+ +.o. =o. |
|     *.O = oo++  |
|    . X.+ = o .o.|
|     = +S= . o.o |
|    . = o . E .  |
|     o +         |
|      + .        |
|     .           |
+----[SHA256]-----+
```

- Verify where the files are stored



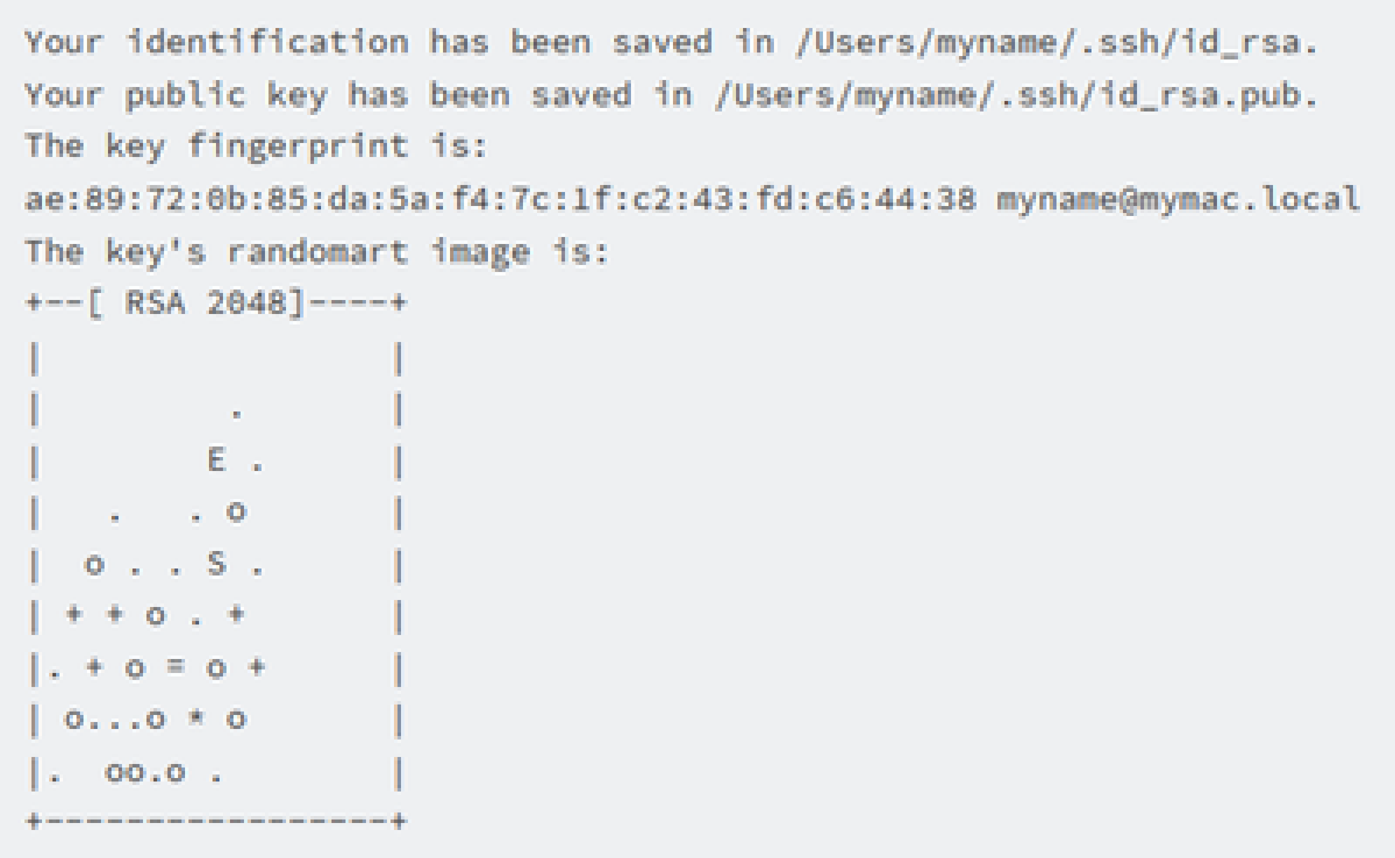
| Name | Date modified | Type | Size |
|---|---|---|---|
| id_rsa | 9/14/2024 1:59 PM | File | 3 KB |
| id_rsa.pub | 9/14/2024 1:59 PM | PUB File | 1 KB |

Transfer the public key **(id_rsa.pub)** in the file repository folder that is configured on ISE.

## Create the private and public keys via in MacOS

Click on the **Finder** icon located in the Dock
- Navigate to the **Applications folder**
- Within the **Applications folder**, locate and open the Utilities folder
- In the Utilities list, find **Terminal**
- Double-click on **Terminal** to open it
- In the **Terminal** window, type “**ssh-keygen -t rsa**“ and press the enter key to execute it
- Write the encryption key twice and **save it**
- Go to files location

Transfer the public key **(id_rsa.pub)** in the file repository folder that is configured on ISE.

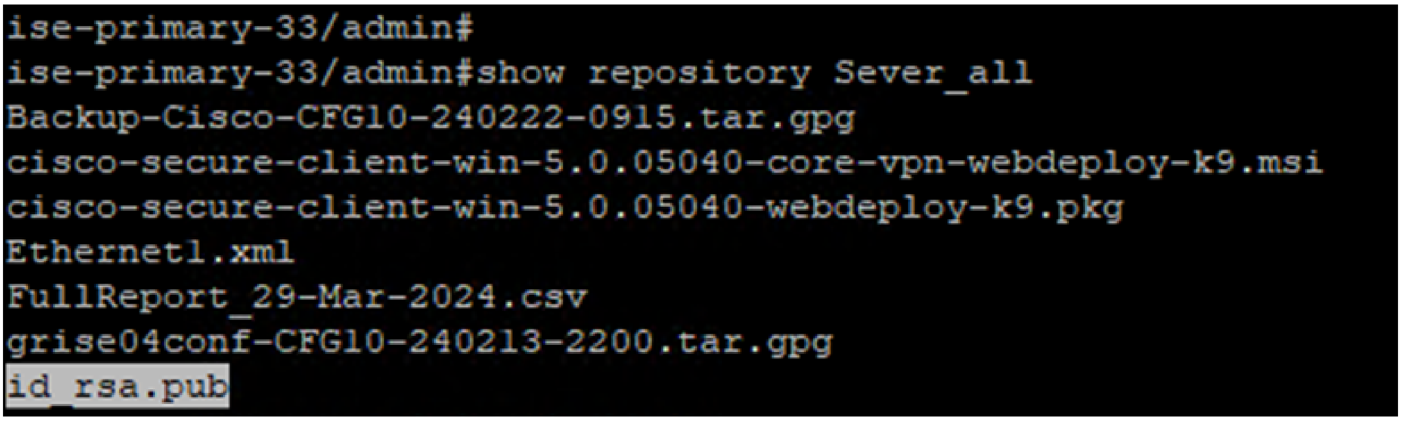


```
Your identification has been saved in /Users/myname/.ssh/id_rsa.
Your public key has been saved in /Users/myname/.ssh/id_rsa.pub.
The key fingerprint is:
ae:89:72:0b:85:da:5a:f4:7c:1f:c2:43:fd:c6:44:38 myname@mymac.local
The key's randomart image is:
+--[ RSA 2048]----+
|                 |
|         .       |
|        E .      |
|   .   . o       |
|  o . . S .      |
| + + o . +       |
|. + o = o +      |
| o...o * o       |
|.  oo.o .        |
+-----------------+
```

## Configure the certificate to login in ISE

Corroborate if the public file is under the repository by using the next command:

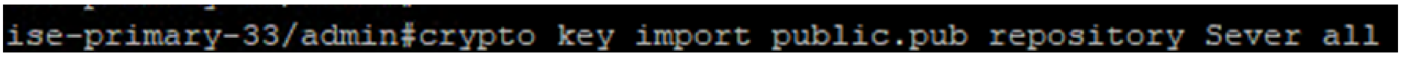
```
show repository <name of the repository>
```



```
ise-primary-33/admin#
ise-primary-33/admin#show repository Sever_all
Backup-Cisco-CFG10-240222-0915.tar.gpg
cisco-secure-client-win-5.0.05040-core-vpn-webdeploy-k9.msi
cisco-secure-client-win-5.0.05040-webdeploy-k9.pkg
Ethernet1.xml
FullReport_29-Mar-2024.csv
grise04conf-CFG10-240213-2200.tar.gpg
id_rsa.pub
```

- Import the public key file **(id_rsa.pub)** by using the command in the privilege mode:

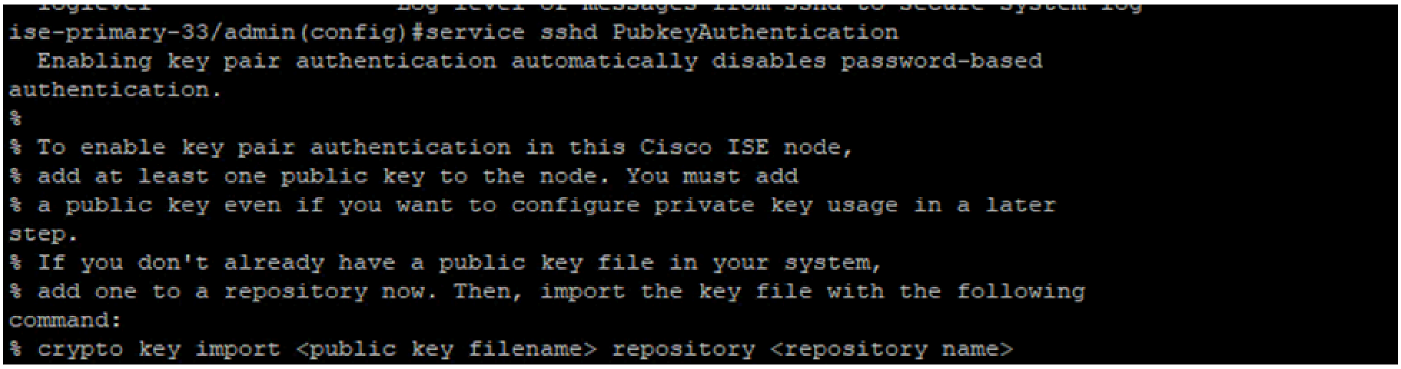
```
crypto key import <public key filename> repository <repository name>
```



```
ise-primary-33/admin#crypto key import public.pub repository Sever all
```

- Enter to the global configuration mode and use the command:

```
service sshd PubkeyAuthentication
```



```
ise-primary-33/admin(config)#service sshd PubkeyAuthentication
  Enabling key pair authentication automatically disables password-based
authentication.
%
% To enable key pair authentication in this Cisco ISE node,
% add at least one public key to the node. You must add
% a public key even if you want to configure private key usage in a later
step.
% If you don't already have a public key file in your system,
% add one to a repository now. Then, import the key file with the following
command:
% crypto key import <public key filename> repository <repository name>
```

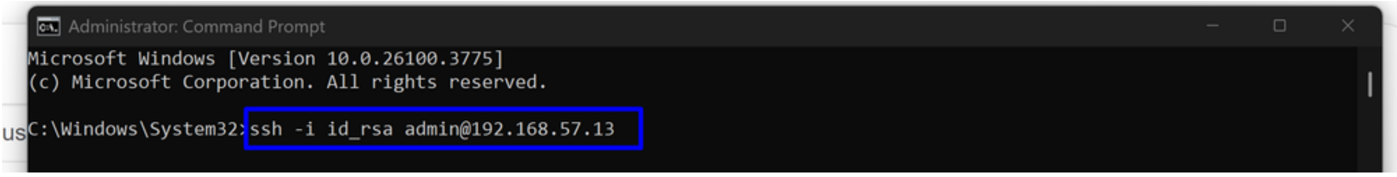
Please use the comand in order to verifying that you do not get any errors meanwhile importing the public key. It is suggested to proceed with this via the console port to avoid losing the access to the ISE.

# Verify

## Login in Windows

Try to access to the ISE via **cmd** by using the command:

```
ssh -i <private_key_file> <ISE_CLI_USER>@<IP_OR_FQDN_OF_ISE>
EXAMPLE:
ssh -i id_rsa admin@192.168.57.13
```



```
Microsoft Windows [Version 10.0.26100.3775]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\System32>ssh -i id_rsa admin@192.168.57.13
```

Use the encryption key configured in the step Create the private and public keys in Windows in order to authenticate.

## Login in MacOS

Enter this command in the terminal:

```
ssh -i <private_key_file> <ISE_CLI_USER>@<IP_OR_FQDN_OF_ISE>
EXAMPLE:
ssh -i id_rsa admin@192.168.57.13
```

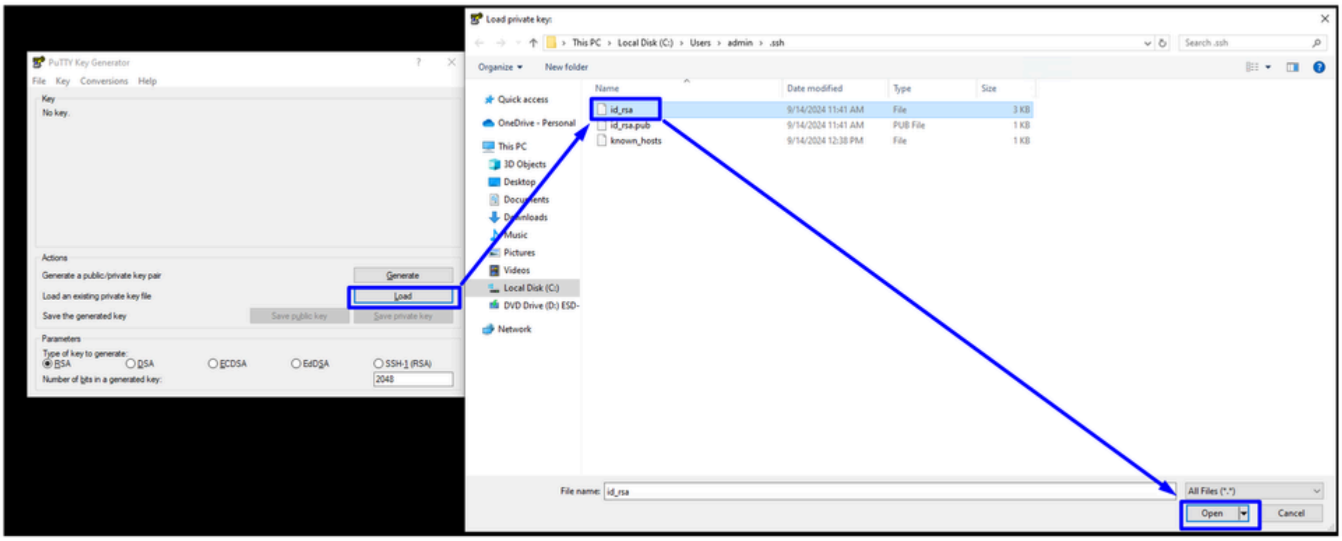
or

```
ssh -i ~/.ssh/ <private_key_file> <ISE_CLI_USER>@<IP_OR_FQDN_OF_ISE>
EXAMPLE:
ssh -i ~/.ssh/id_rsa admin@192.168.57.13
```

Use the encryption key configured in the step  Create the private and public keys via in MacOS in order to authenticate.

## Login in Putty

Open PuTTy key generator  (search by PuttyGen in the start search bar), click on Load, select all files, and open the private key generated from cmd (Windows) or terminal (MacOS):

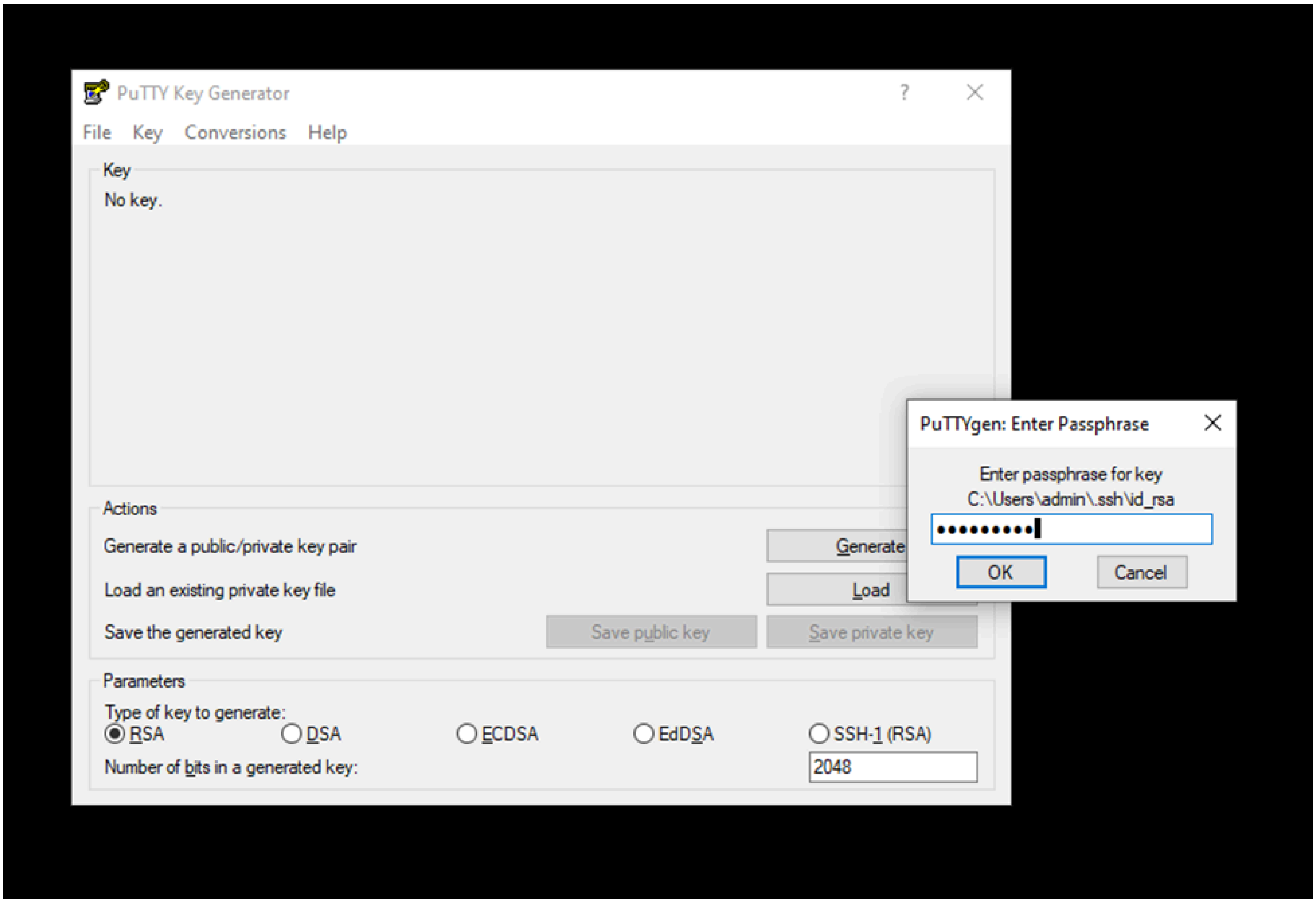


This PC > Local Disk (C:) > Users > admin > .ssh

| Name | Date modified | Type | Size |
|---|---|---|---|
| id_rsa | 9/14/2024 11:41 AM | File | 3 KB |
| id_rsa.pub | 9/14/2024 11:41 AM | PUB File | 1 KB |
| known_hosts | 9/14/2024 12:38 PM | File | 1 KB |

File name: id_rsa

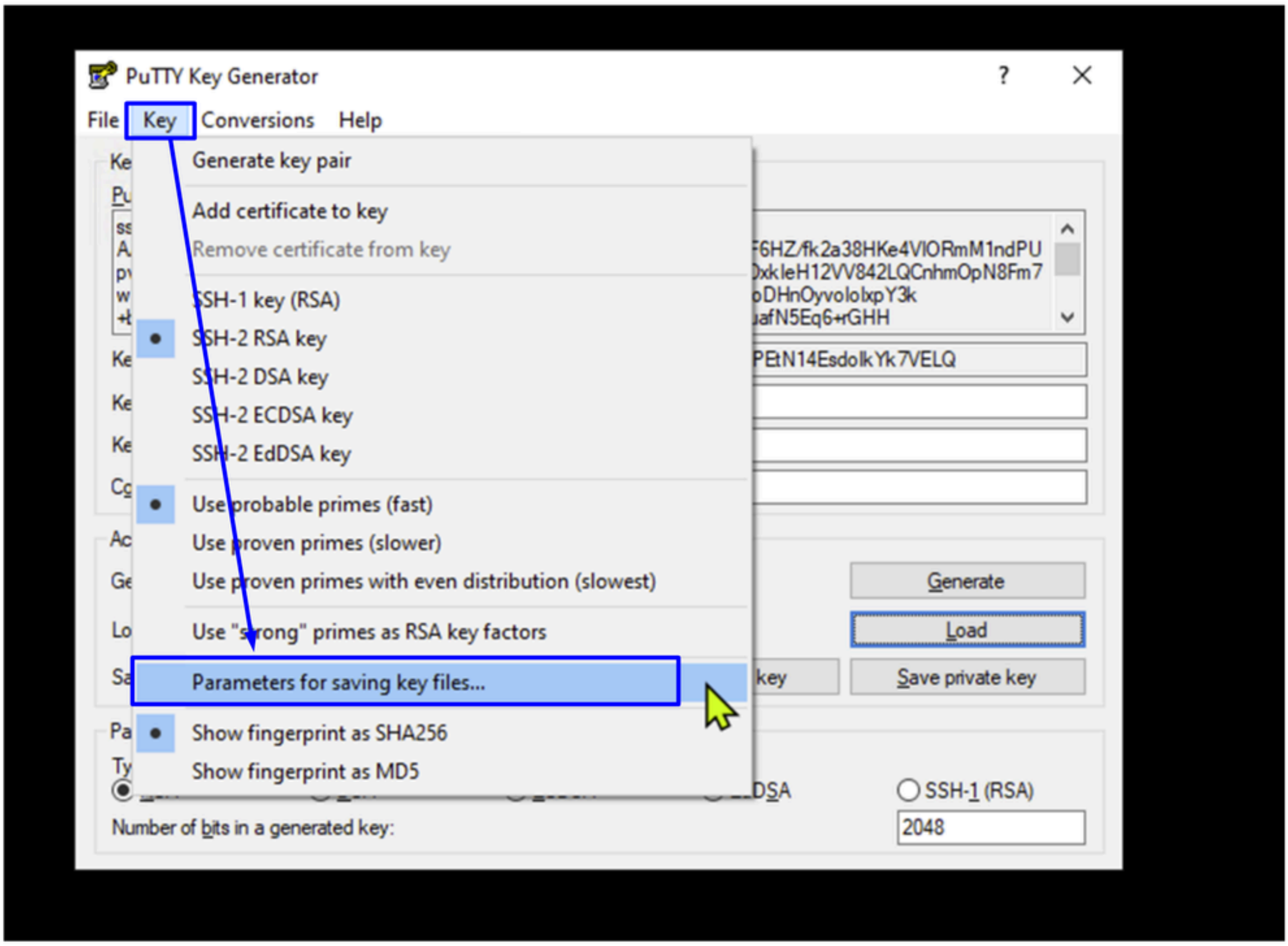
- Write the encryption key previously used in the cmd or terminal

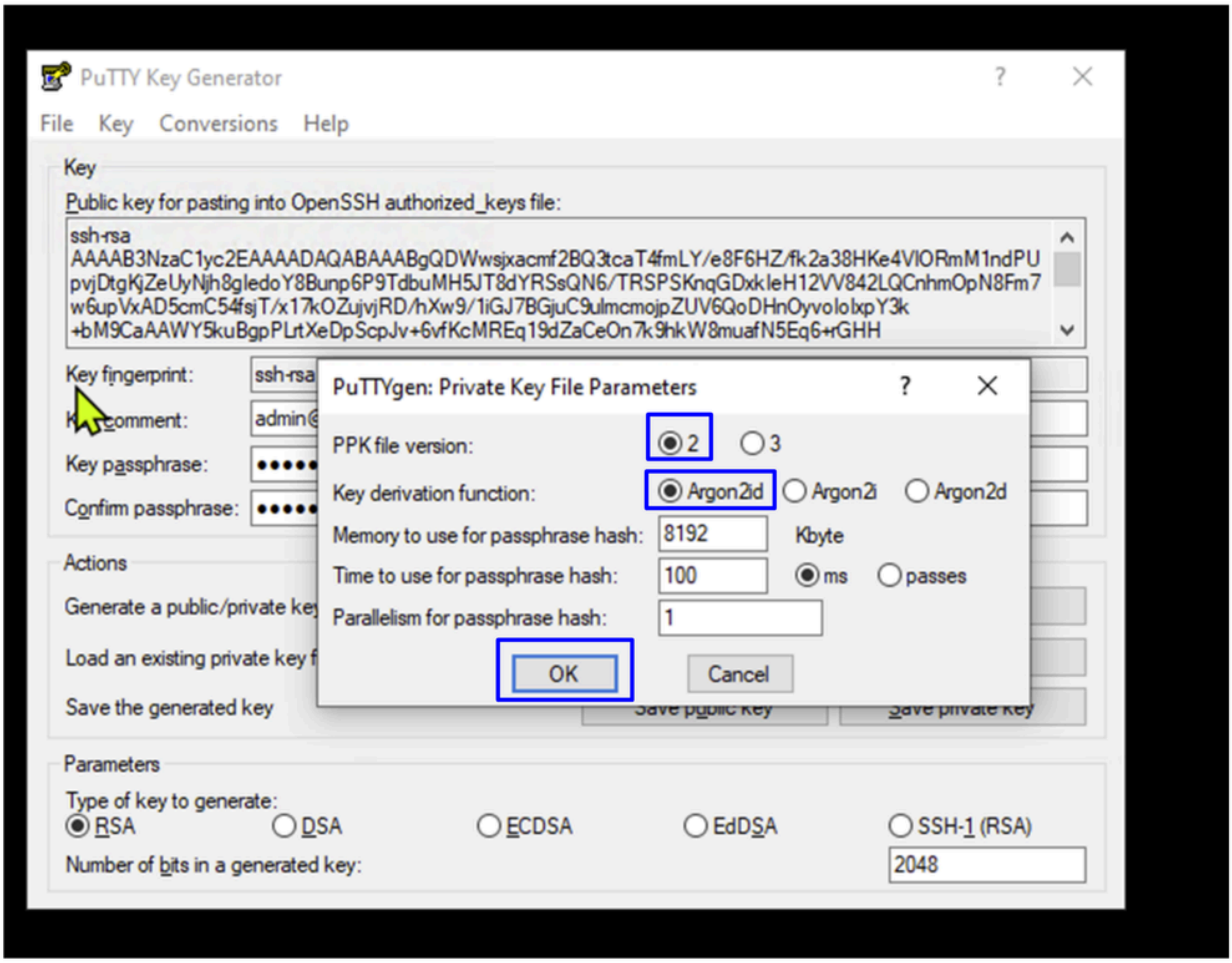

PuTTY Key Generator
File Key Conversions Help
Key
No key.
Actions
Generate a public/private key pair
Generate
Load an existing private key file
Load
Save the generated key
Save public key
Save private key
Parameters
Type of key to generate:
● RSA ○ DSA ○ ECDSA ○ EdDSA ○ SSH-1 (RSA)
Number of bits in a generated key:
2048
PuTTYgen: Enter Passphrase
OK
Cancel

Enter passphrase for key
C:\Users\admin\.ssh\id_rsa
••••••••••

Convert this file to a Putty version compatible by executing the next steps:

- Click on Key > Parameters for saving key files


PuTTY Key Generator
?
File
Key
Conversions
Help
Generate key pair
Add certificate to key
Remove certificate from key
SSH-1 key (RSA)
● SSH-2 RSA key
SSH-2 DSA key
SSH-2 ECDSA key
SSH-2 EdDSA key
● Use probable primes (fast)
Use proven primes (slower)
Use proven primes with even distribution (slowest)
Use "strong" primes as RSA key factors
Parameters for saving key files...
● Show fingerprint as SHA256
Show fingerprint as MD5
F6HZ/fk2a38HKe4VIORmM1ndPU
0xkleH12VV842LQCnhmOpN8Fm7
oDHnOyvololxpY3k
uafN5Eq6+rGHH
PEtN14EsdolkYk7VELQ
Generate
Load
key
Save private key
○ SSH-1 (RSA)
Number of bits in a generated key:
2048



Key

Public key for pasting into OpenSSH authorized_keys file:

```
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABgQDWwsjxacmf2BQ3tcaT4fmLY/e8F6HZ/fk2a38HKe4VIORmM1ndPU
pvjDtgKjZeUyNjh8gledoY8Bunp6P9TdbuMH5JT8dYRSsQN6/TRSPSKnqGDxkleH12VV842LQCnhmOpN8Fm7
w6upVxAD5cmC54fsjT/x17kOZujvjRD/hXw9/1iGJ7BGjuC9ulmcmojpZUV6QoDHnOyvololxpY3k
+bM9CaAAWY5kuBgpPLrtXeDpScpJv+6vfKcMREq19dZaCeOn7k9hkW8muafN5Eq6+rGHH
```

Key fingerprint: ssh-rsa

Key comment: admin@

Key passphrase: ●●●●●

Confirm passphrase: ●●●●●

PuTTYgen: Private Key File Parameters

PPK file version: ● 2 ○ 3

Key derivation function: ● Argon2id ○ Argon2i ○ Argon2d

Memory to use for passphrase hash: 8192 Kbyte

Time to use for passphrase hash: 100 ● ms ○ passes

Parallelism for passphrase hash: 1

Parameters

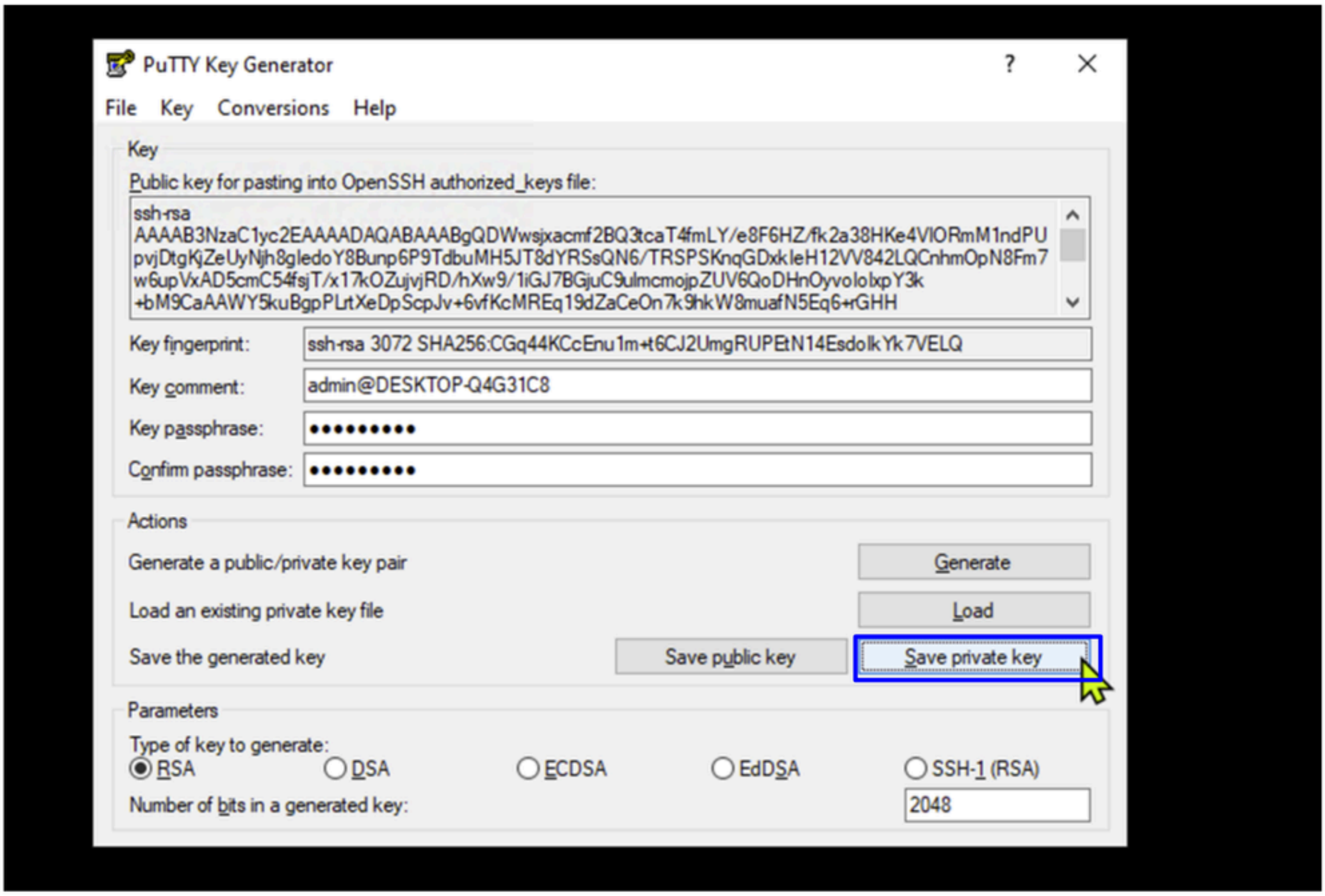
Type of key to generate:
● RSA ○ DSA ○ ECDSA ○ EdDSA ○ SSH-1 (RSA)

Number of bits in a generated key: 2048

- PPK file version : Chose 2
- **Key derivation function:** Choose Argon2id

**Note**: For the rest of parameters use the default values.

- Click Ok



PuTTY Key Generator

Public key for pasting into OpenSSH authorized_keys file:

```
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABgQDWwsjxacmf2BQ3tcaT4fmLY/e8F6HZ/fk2a38HKe4VIORmM1ndPU
pvjDtgKjZeUyNjh8gledoY8Bunp6P9TdbuMH5JT8dYRSsQN6/TRSPSKnqGDxkleH12VV842LQCnhmOpN8Fm7
w6upVxAD5cmC54fsjT/x17kOZujvjRD/hXw9/1iGJ7BGjuC9ulmcmojpZUV6QoDHnOyvolollxpY3k
+bM9CaAAWY5kuBgpPLrtXeDpScpJv+6vfKcMREq19dZaCeOn7k9hkW8muafN5Eq6+rGHH
```

Key fingerprint: ssh-rsa 3072 SHA256:CGq44KCcEnu1m+t6CJ2UmgRUPEtN14EsdolkYk7VELQ

Key comment: admin@DESKTOP-Q4G31C8

Key passphrase: ••••••••••

Confirm passphrase: ••••••••••

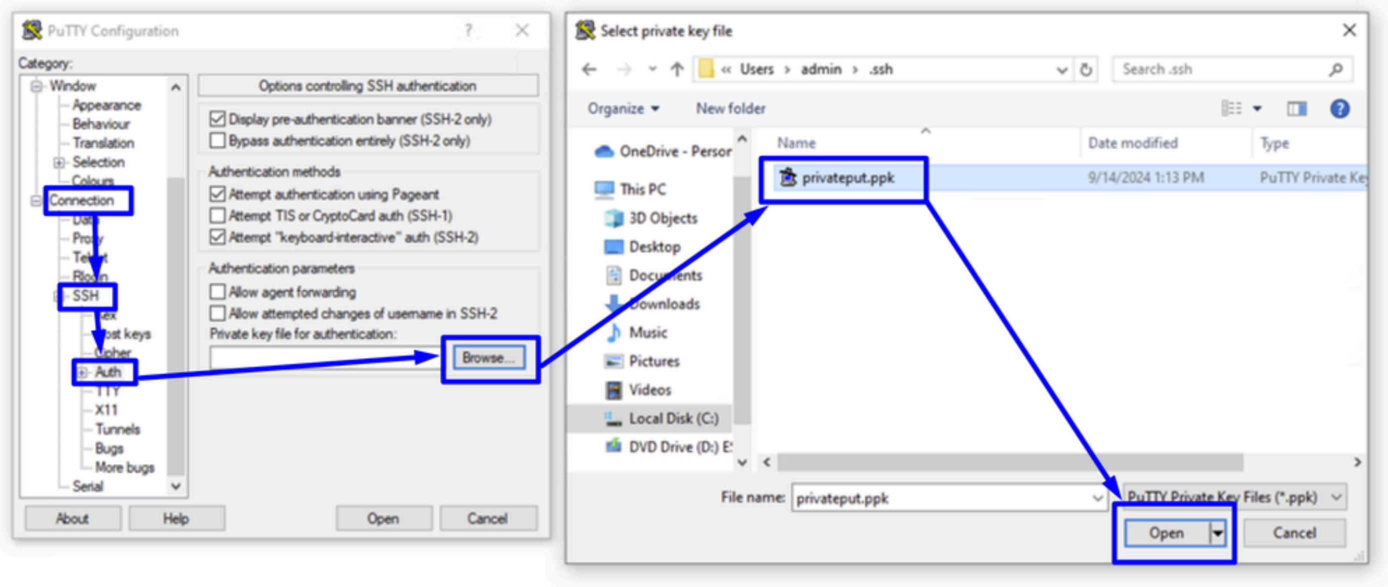
Type of key to generate: ● RSA ○ DSA ○ ECDSA ○ EdDSA ○ SSH-1 (RSA)

Number of bits in a generated key: 2048

- Click on Save private Key

After you save the key on your computer, you are ready to use it by referring to the next examples:

- Open Putty
- Click on **Connection** > **SSH** > **Auth** > **Browse**
- Select your private key and click Open



PuTTY Configuration

Options controlling SSH authentication

☑ Display pre-authentication banner (SSH-2 only)
☐ Bypass authentication entirely (SSH-2 only)

Authentication methods

☑ Attempt authentication using Pageant
☐ Attempt TIS or CryptoCard auth (SSH-1)
☑ Attempt "keyboard-interactive" auth (SSH-2)

Authentication parameters

☐ Allow agent forwarding
☐ Allow attempted changes of username in SSH-2

Private key file for authentication:

Select private key file

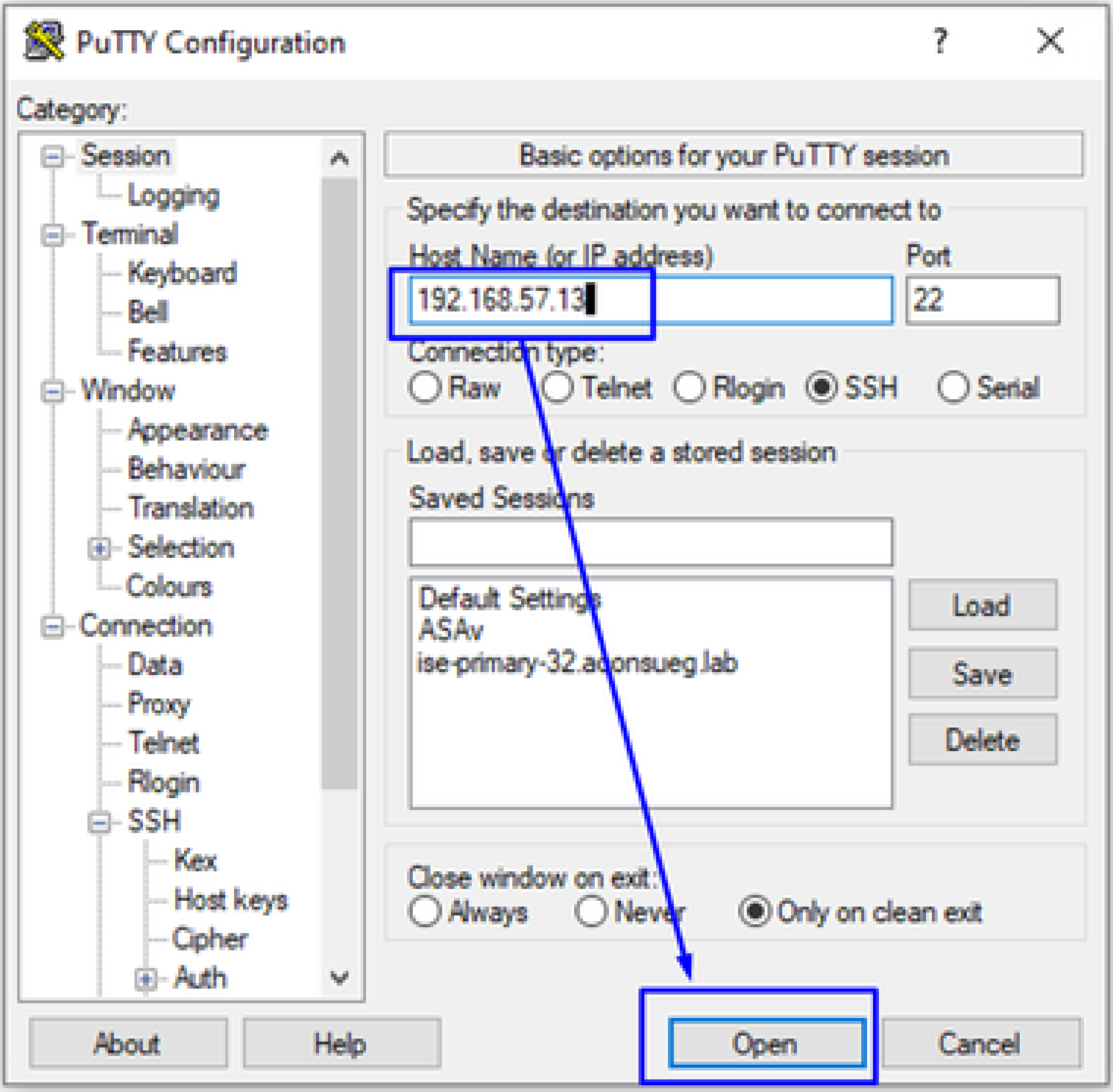
Users > admin > .ssh

| Name | Date modified | Type |
|---|---|---|
| privateput.ppk | 9/14/2024 1:13 PM | PuTTY Private Key |

File name: privateput.ppk

PuTTY Private Key Files (*.ppk)

- Go back to Session, set the IP address or hostname (FQDN) of the ISE
- Click Open



PuTTY Configuration

Basic options for your PuTTY session

Specify the destination you want to connect to

Host Name (or IP address): 192.168.57.13

Port: 22

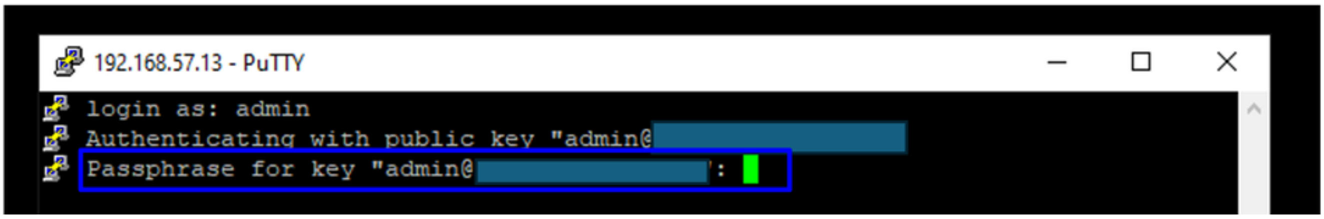
Connection type: ○ Raw ○ Telnet ○ Rlogin ● SSH ○ Serial

Load, save or delete a stored session

Saved Sessions

Default Settings
ASAv
ise-primary-32.aaronsueg.lab

Close window on exit: ○ Always ○ Never ● Only on clean exit



192.168.57.13 - PuTTY

```
login as: admin
Authenticating with public key "admin@[redacted]
Passphrase for key "admin@[redacted]":
```

Use the encryption key configured in the step Create the private and public keys via in MacOS or Create the private and public keys in Windows in order to authenticate.

## Troubleshoot

Check out error messages from the endpoint site adding in the ssh connection the flag **-v**

```
Example for Windows:
ssh -v -i id_rsa admin@192.168.57.13
```

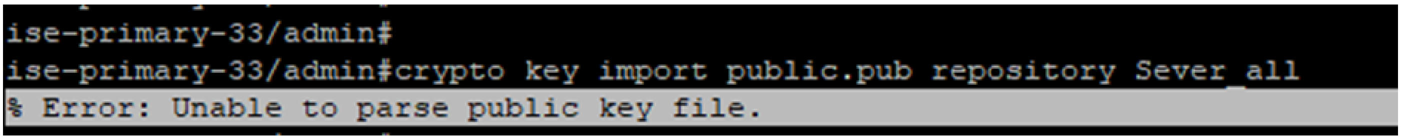
```
Example for MacOS:
ssh -v -i id_rsa admin@192.168.57.13
```

or

```
ssh -v -i ~/.ssh/id_rsa admin@192.168.57.13
```

## Error importing the public key

% Error: Unable to parse public key file.



```
ise-primary-33/admin#
ise-primary-33/admin#crypto key import public.pub repository Sever_all
% Error: Unable to parse public key file.
```

If you face any inconvenience importing more than one public key, please contact Cisco support.