LAB: ssh

Lab Environment

The workshop WiFi:

- SSID: `workshop`
- PASS: `iij/2497`

Hosts - Virtual machines (Ubuntu 18.04LTS/LXC)

- Hostname: `nsXX.workshop`
- IPv6: `fd00:2497:1::X`
- IPv4: `10.0.0.X`

Where `X` and `XX` is yoru group ID. For group 1, hostname is `ns01.workshop`, IPv6 address is `fd00:2497:1::1`, and IPv4 is `10.0.0.1`.

Download ssh client (older Windows)

Visit the PuTTY download site.

```
https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html
```

Download `putty-0.70-installer.msi` or `putty-64bit-0.70-installer.msi` based on your Windows edition and install the package.

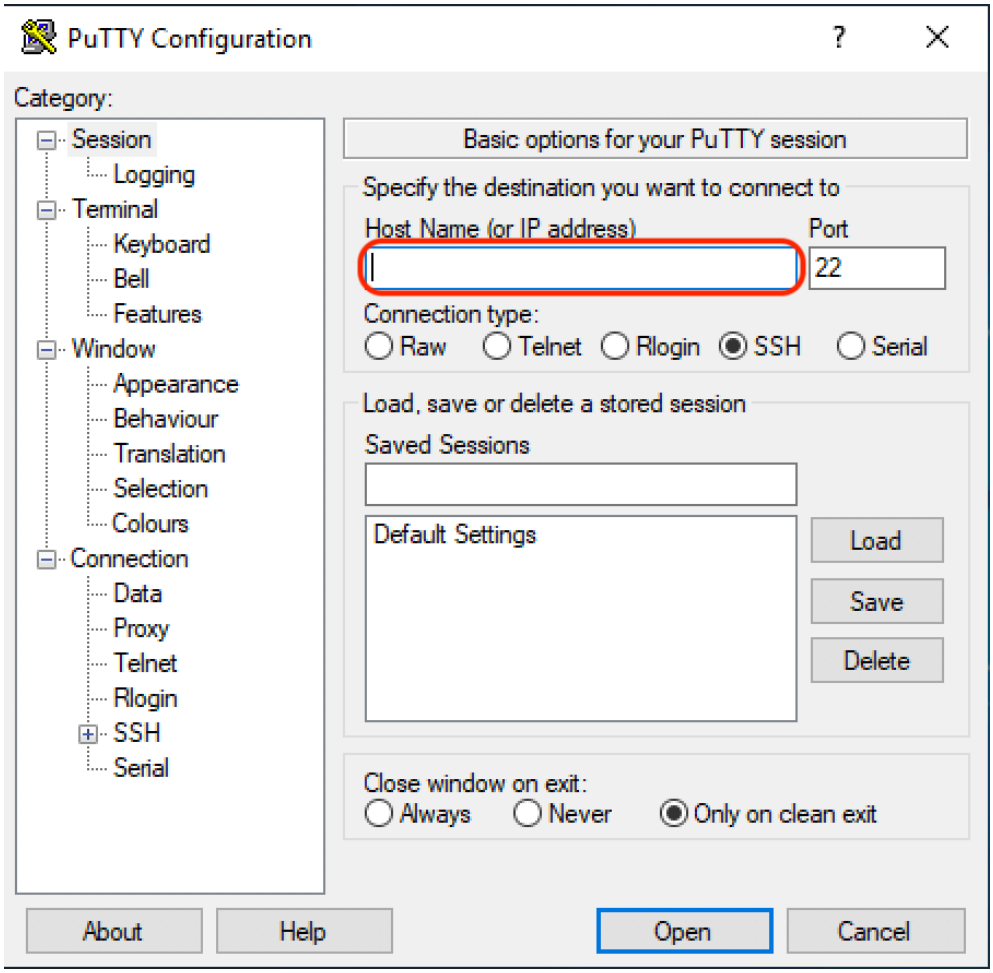
Or you can download client software individually if you like.

- `putty.exe` (ssh/telnet remote login client)
- `puttygen.exe` (ssh key generator)
- `pagent.exe` (ssh agent)
- `pscp.exe` (ssh file copy tool)

> Note: Latest Windows10 (1803 or later) has builtin ssh client software

Exercise 1: ssh and password (PuTTY/Windows)

Run `putty.exe`



Input `workshop@ns0X.workshop` in the Host Name field. If the hostname doesn't work, use IP address (e.g. `10.0.0.X`) instead.

Click `Open` to connect to your virtual server. The password is `iij/2497`.

Type `exit` in the remote terminal when you finish remote login.

Excerise 1: ssh and password (Latest Windows10)

Run a command prompt application.

Login to your virtual server.

```
$ ssh workshop@nsXX.workshop
```

If hostname doesn't work in your environment, use IP address instead.

```
$ ssh workshop@10.0.0.X
```

Password is `iij/2497`.

Excerise 1: ssh and password (UNIX/Mac)

Run a terminal application (e.g. `Terminal.app` if you use Mac).

Login to your virtual server.

```
$ ssh workshop@nsXX.workshop
```

If hostname doesn't work in your environment, use IP address instead.

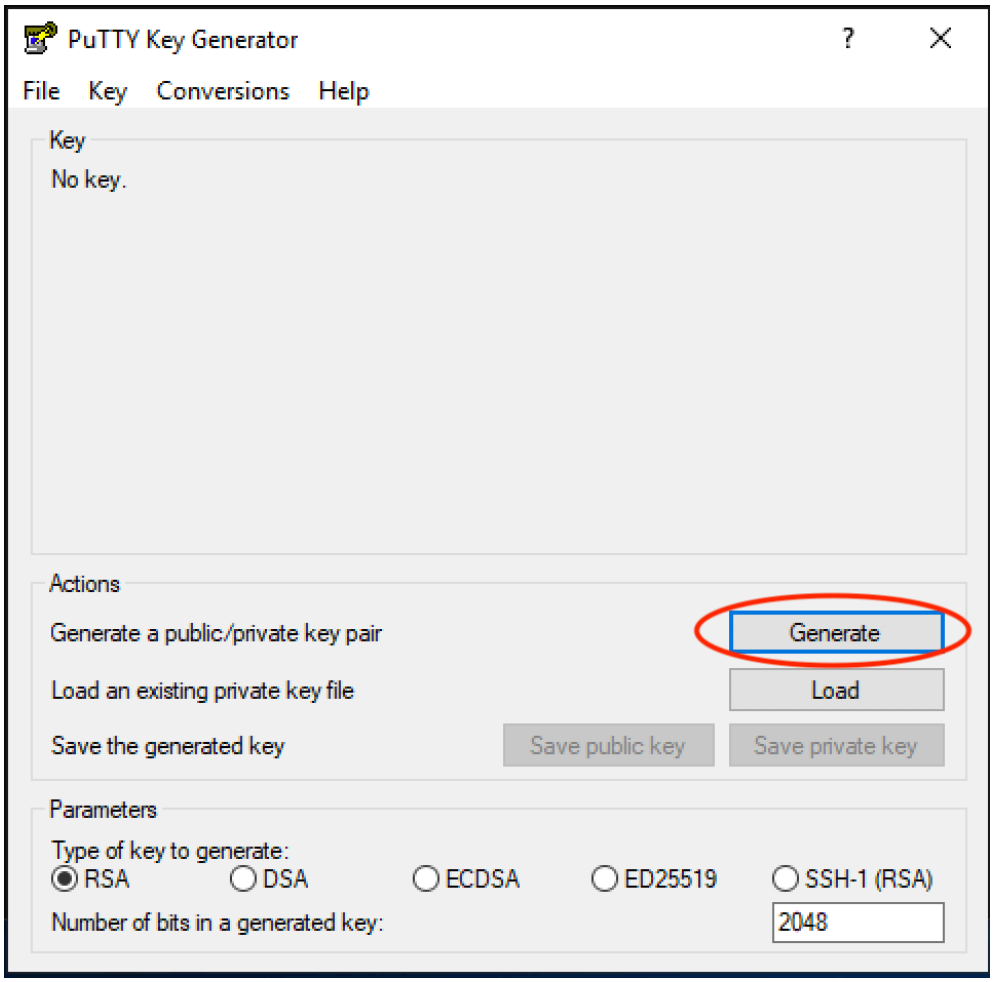
```
$ ssh workshop@10.0.0.X
```

Password is `iij/2497`.

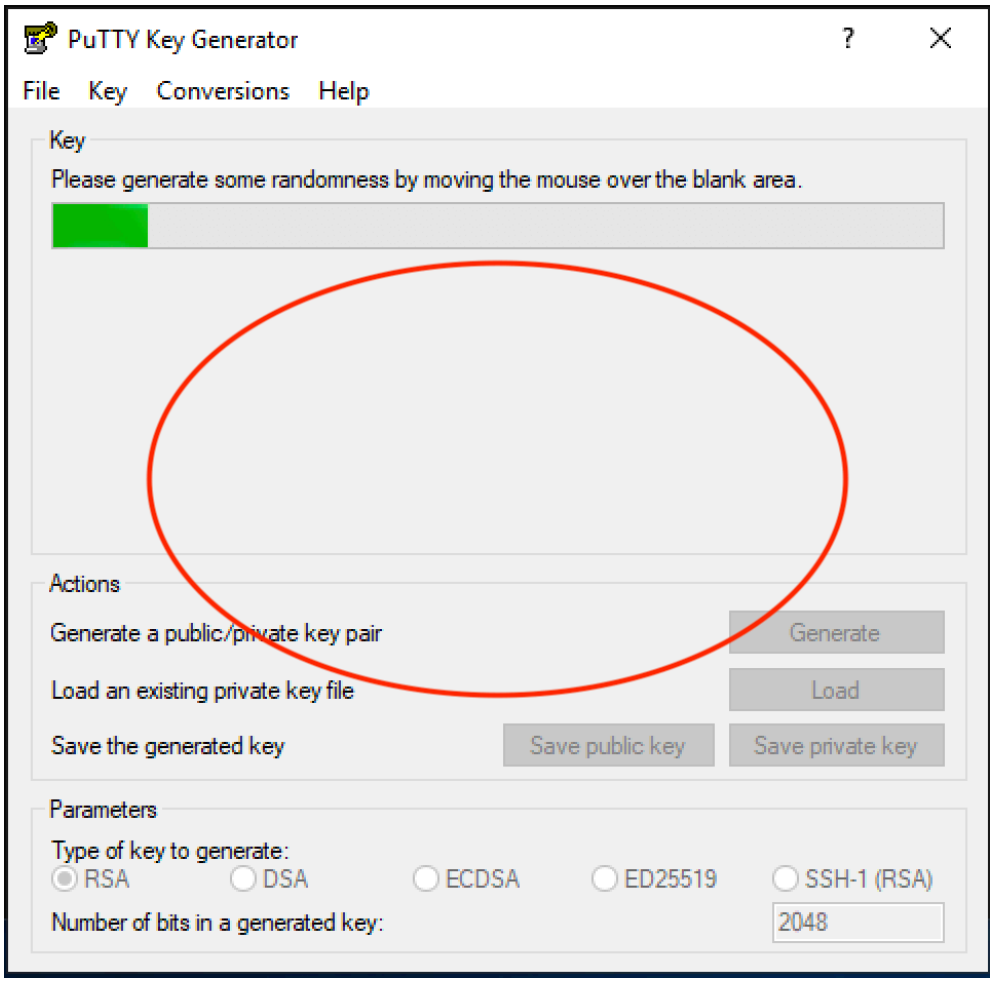
Exercise 2: ssh and key (PuTTY/Windows)

Key generation

Run `puttygen.exe` to generate your key pair and save them.

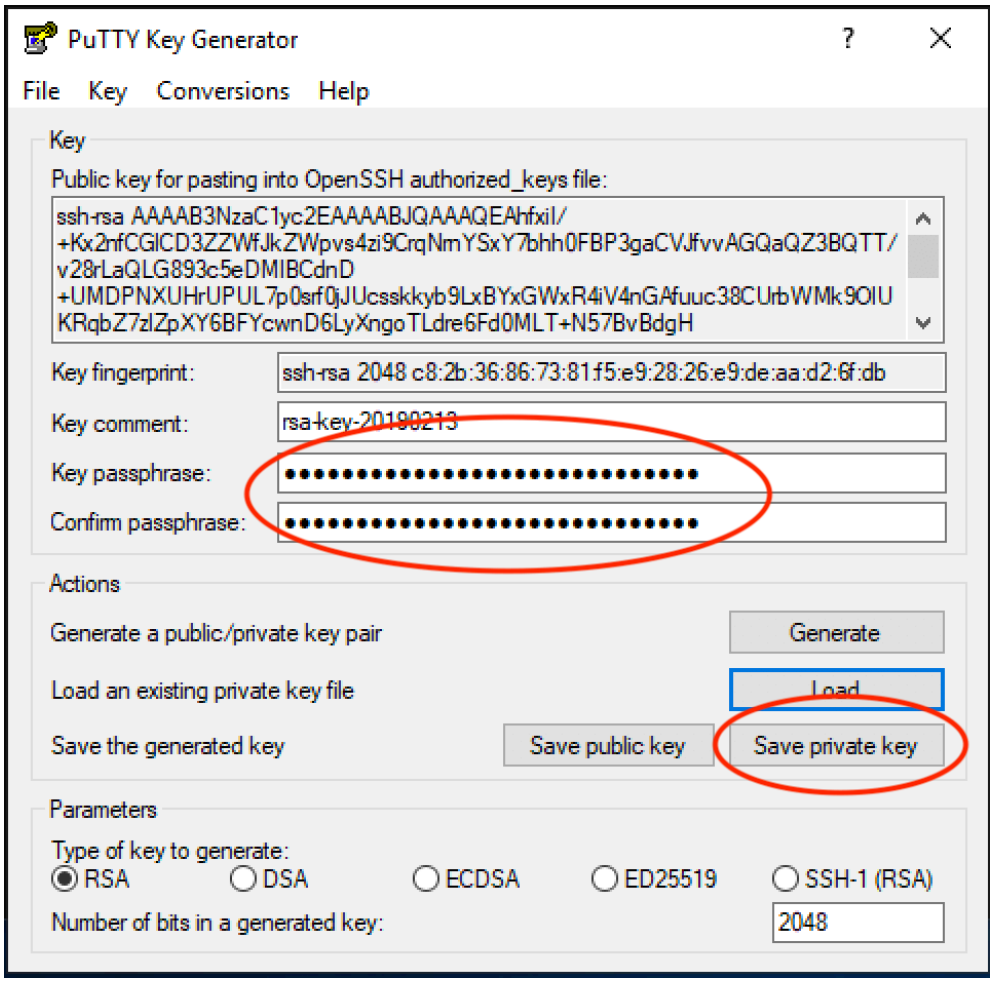


Choose parameters (default RSA/2048 is a good choice) and click `Generate`.



While key is being generated, move your mouse to provide a random source.

Once a keypair is generated, set your passphrase to the key and save your private key.



Your public key is shown in the text box on top. Copy the entire string and setup the remote server.

Remote server setup

Login to your virtual server using `ssh` and password, and create a file to keep your public keys.

```
(first, login to your virtual server using password)
$ mkdir -p ~/.ssh
$ chmod 0700 ~/.ssh
$ vi ~/.ssh/authorized_keys
```

Once you open the file, follow the procedures below.

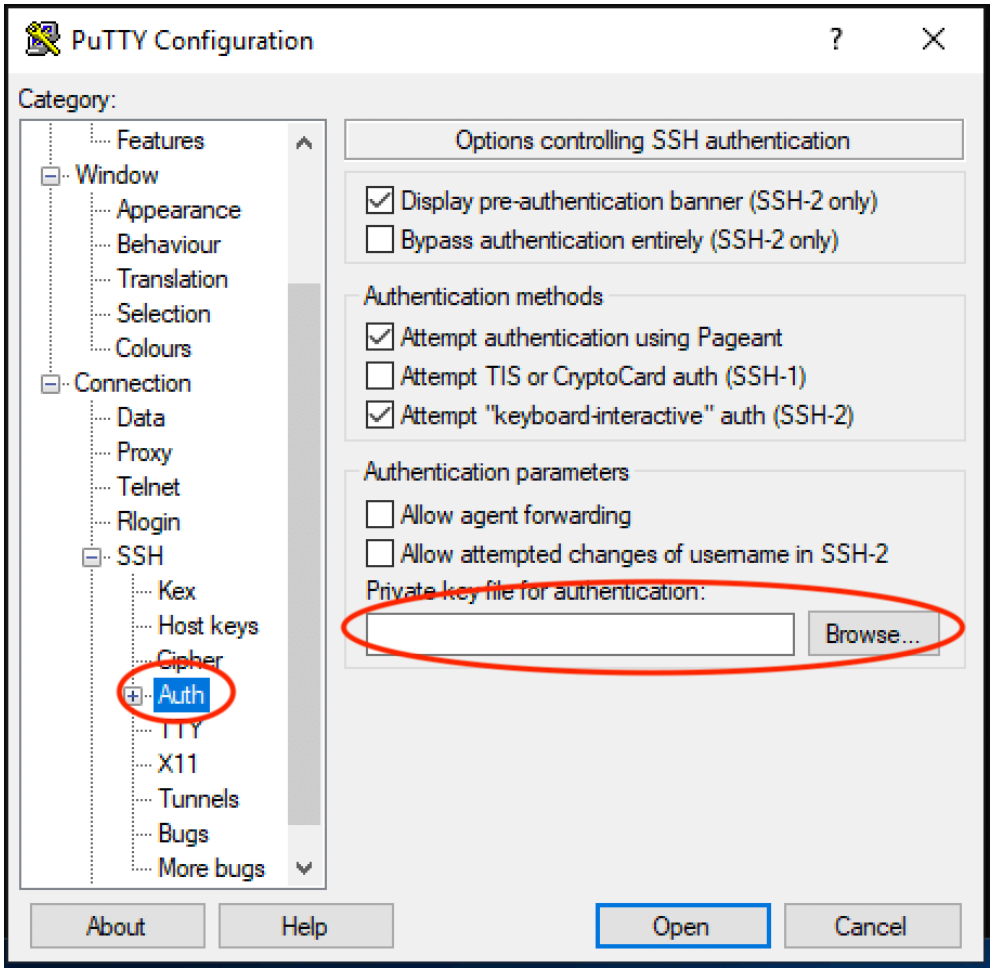
- Copy the public key string on your Windows
- Type `i` key on the remote server (in the `vi` editor)
- Right click on the editor to paste the public key string
- Press `ESC` key and type `:wq` to save the `authorized_keys` file and exit the `vi` editor

> Note: Do not fold the public key string (by inserting CR/LF). The key information must be in one line.

Login with key

Open the Putty application and input your virtual server name (or IP address)

Then open the `Connection/SSH/Auth` menu and place your private key. You will be required to input your passphrase of your private key.



Click `Open` to connect to your remote server.

Exercise 2: ssh and key (Latest Windows10)

Key generation

Run the `ssh-keygen` command to generate a new key pair.

```
C:\Users\Keiichi Shima>ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (C:\Users\Keiichi Shima/.ssh/id_rs
a):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in C:\Users\Keiichi Shima/.ssh/id_r
sa.
Your public key has been saved in C:\Users\Keiichi Shima/.ssh/id_rsa.p
ub.
The key fingerprint is:
SHA256:Ub+vvv3YH+ddsQGAKUgAdh80CKCUS3sZZe1kWeeFPRQ keiichi shima@DESKT
OP-7BKMDS3
The key's randomart image is:
+---[RSA 2048]----+
|o=+o*Bo o.+o+E.  |
|+o.oo.o* +oooo   |
|o o o.+ o  ....  |
| o o   . .   ..  |
|  .     S   .  o |
|             .  +|
|              .oo|
|             o +=|
|           .+.o.B|
+----[SHA256]-----+
```

You will be asked the location of your key pair (the default location is `~/.ssh/id_rsa` for a private key and `~/.ssh/id_rsa.pub` for a public key) and a passphrase.

Remote server setup

Login to your virtual server using `ssh` and password, and create a file to keep your public keys.

```
(first, login to your virtual server)
$ mkdir -p ~/.ssh
$ chmod 0700 ~/.ssh
$ vi ~/.ssh/authorized_keys
```

Once you open the file, follow the procedures below.

- Copy the contents of `~/.ssh/id_rsa.pub` on your Windows
- Type `i` key on the remote server (in the `vi` editor)
- Right click on the editor to paste the public key string
- Press `ESC` key and type `:wq` to save the `authorized_keys` file and exit the `vi` editor

> Note: Do not fold the public key string (by inserting CR/LF). The key information must be in one line.

Login with key

Now you can login using your key pair.

```
$ ssh workshop@nsXX.workshop
```

You will be asked your passphrase of your private key.

Exercise 2: ssh and key (UNIX/Mac)

Key generation

Run the `ssh-keygen` command to generate a new key pair.

```
$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/shima/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/shima/.ssh/id_rsa.
Your public key has been saved in /home/shima/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:lsrq65wLUlcCzYE7cIDTV6RnOxCf/82HbzkYSVLt+Q0 shima@mymachine
The key's randomart image is:
+---[RSA 2048]----+
|oo.+o+o      .   |
|+ +.++ .    . .  |
| + o+ *    . . . |
|  o  * o .. . E  |
|  ... o S  o . o.|
| . . . + . oo.  o|
|. .   o   . +o.. |
| . o o      .o+  |
|   oOo       ... |
+----[SHA256]-----+
```

You will be asked the location of your key pair (the default location is `~/.ssh/id_rsa` for a private key and `~/.ssh/id_rsa.pub` for a public key) and a passphrase.

Remote server setup

Copy your key to your remote server.

```
$ ssh-copy-id workshop@nsXX.workshop
```

Login with key

Now you can login using your key pair.

```
$ ssh workshop@nsXX.workshop
```

You will be asked your passphrase of your private key.

Exercise 3: Disable password login

Login your virtual server and edit `/etc/ssh/sshd_config`. You will find the following lines.

```
# To disable tunneled clear text passwords, change to no here!
PasswordAuthentication yes
```

Change the value from `yes` to `no` using an editor.

Restart the ssh service.

```
$ sudo systemctl restart ssh
```

> Note: the sudo password is `iij/2497`.

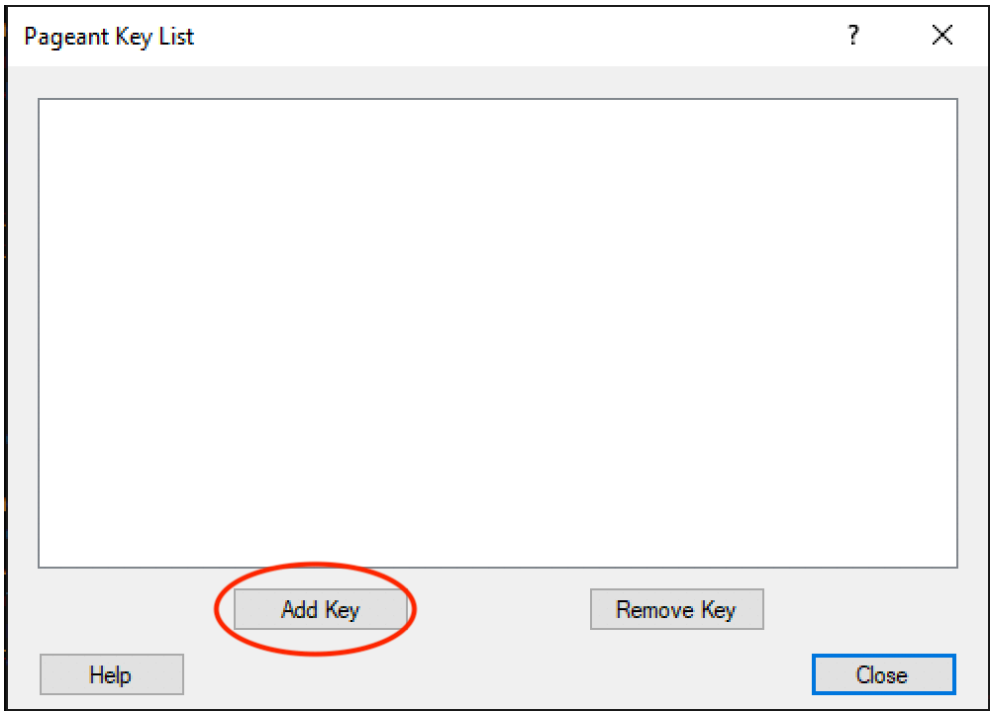
Check whether the password authentication is properly disabled.

```
OnYourLocalPC> ssh foobar@nsXX.workshop
foobar@nsXX.workshop: Permission denied (publickey).
```

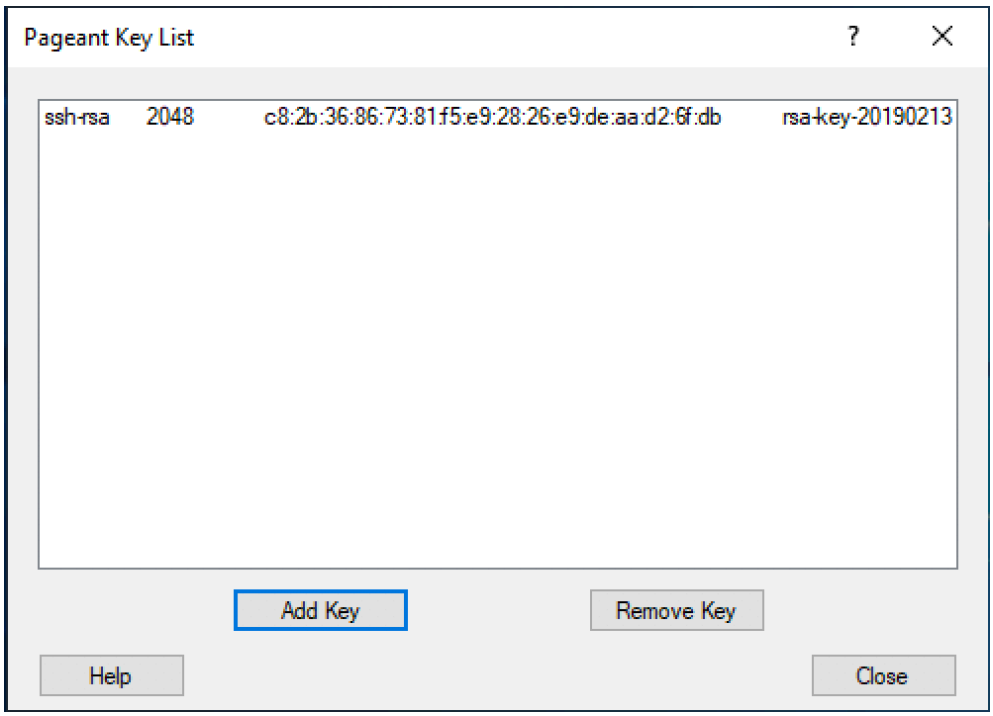
Exercise 4: Using ssh agent (PuTTY/Windows)

Run `pagent.exe`. You will find a task tray icon similar to the below.

Right click the icon and select `View Keys`.



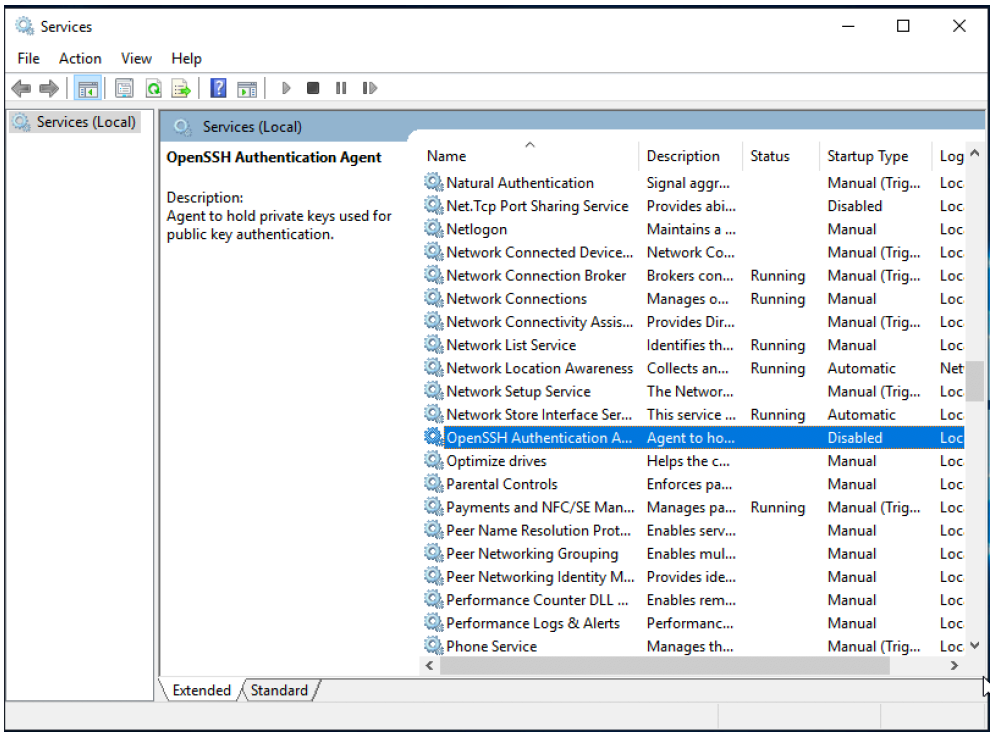
Click `Add Key` and select your private key file. You will be asked your passphrase of your private key.



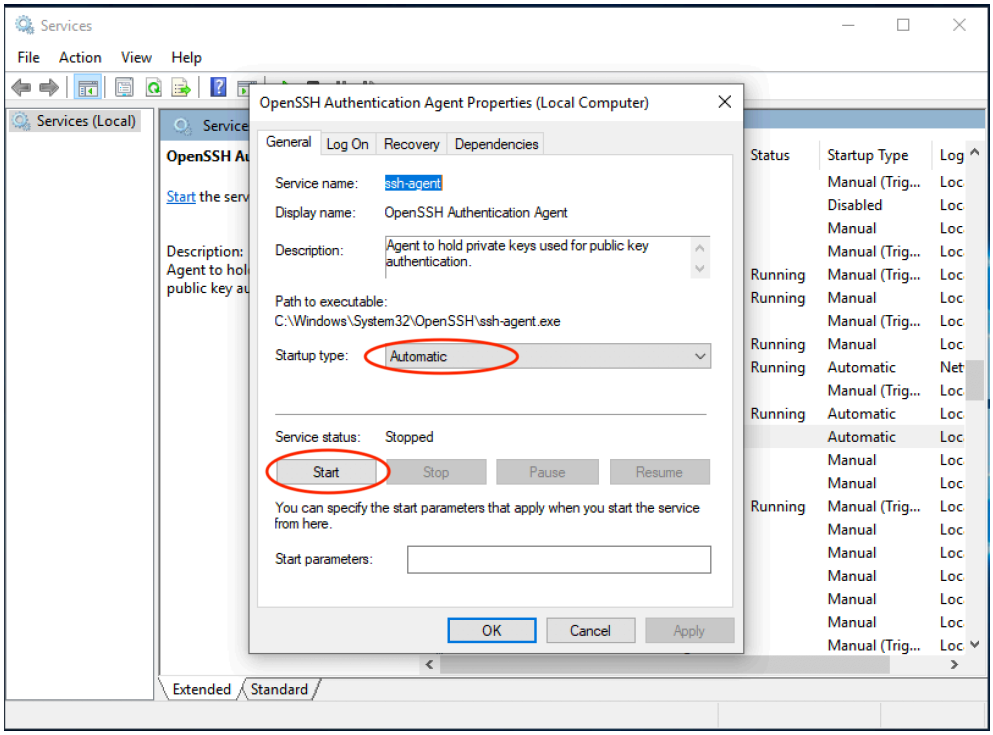
Once your key is registered to the agent, you can bypass your passphrase when you login to remote server using the registered private key.

Exercise 4: Using ssh agent (Latest Windows10)

Run the `Services` application under the `Windows Administration Tools` group.



Find the OpenSSH Authentication Agent service and open the property.



Change the value of the startup type from `Disabled` to `Automatic`, and click `Start` to start the ssh agent service.

Open the command terminal application and register your private key to the agent.

```
C:\Users/Keiichi Shima>ssh-add
Enter passphrase for C:\Users/Keiichi Shima/.ssh/id_rsa:
```

Once your key is registered to the agent, you can bypass your passphrase when you login to remote server using the registered private key.

> Note: In some Windows10 environment, `ssh-add` fails with the error message something like below.
>
> `Can't add keys to ssh-agent, communication with agent failed`
>
> In that case type the following command.
>
> `sc.exe create sshd binPath=C:\Windows\System32\OpenSSH\ssh.exe`
>
> There is a web discussion of this issue at `https://github.com/PowerShell/Win32-OpenSSH/issues/1234`.

Exercise 4: Using ssh agent (UNIX)

Run the `ssh-agent` program.

```
$ ssh-agent bash
```

> Note: You may want to use your favorite shell other than `bash`.

Add your private key.

```
$ ssh-add
Enter passphrase for /home/shima/.ssh/id_rsa:
```

Once your key is registered to the agent, you can bypass your passphrase when you login to remote server using the registered private key.

Exercise 4: Using ssh agent (Mac)

Add your private key.

```
$ ssh-add
Enter passphrase for /home/shima/.ssh/id_rsa:
```

Once your key is registered to the agent, you can bypass your passphrase when you login to remote server using the registered private key.

To check if your key is registered to the agent, you can check the registered keys with the following command.

```
$ ssh-add -l
ssh-add -l
2048 SHA256:HNOg7sCYJcHY5rOlWO7tQcTAkCTcmntYVb96jRg2B7c /Users/keiichi
/.ssh/id_rsa (RSA)
```

Exercise 5: Copy file (PuTTY/Windows)

Run the command application.

Open the folder where the `pscp.exe` command is installed. Usually, the location is `C:\Program Files\PuTTY\`. Drag the `pscp.exe` to the command window.

Type the filename you want to copy and remote server name in the command line.

```
C:\Users/Keiichi Shima>"C:\Program Files\PuTTY\pscp.exe" SOME_FILE.txt
 workshop@nsxX.workshop:/home/workshop/
```

> Note: If hostname doesn't work in your environment, use IP address instead.

Exercise 5: Copy file (Other OSes)

Use the `scp` command to copy files to a remote server.

```
$ scp SOME_FILE.txt workshop@ns0X.workshop:/home/workshop/
```

Exercise 6: Allow other users

Get your neighbor's public key and add it to your virtual server's `authorized_keys` file.

- You can ask to send the key somehow (e.g. via an email)
- It is a good exercise to think about a safe procedure to get others' keys, like pgp signed message

> Note: The `authorized_keys` file can contain multiple keys, one line per key.

Ask your neighbor to login to your virtual server.