

FEPS

**SSH Access
with
Two-Factor Authentication**

RSA Key-pairs

access.eps.surrey.ac.uk



**UNIVERSITY OF
SURREY**

Contents:

Introduction	-	3
RSA Key-pairs	-	3
Where can I use my RSA Key-Pair?	-	3
Step 1 – Prepare to generate your RSA Key-Pair	-	4
Step 2 – Setting up the .ssh folder	-	5
Step 3 – Creating your RSA Key-Pair	-	6
Step 4 – Deploying your Keys	-	8
Step 5 – Using your RSA Key-Pair	-	9
FAQ	-	10

Server Details:

Server Address:	access.eps.surrey.ac.uk
IP Address:	131.227.81.254
Operating System:	Ubuntu Linux 16.04

Help and Assistance:

For Help and Support for SSH Two-Factor Authentication, please contact:

Web:	https://www.surrey.ac.uk/fepsit/contact
Email:	itservicedesk@surrey.ac.uk
Tel (external):	01483 68 9826
Tel (internal):	9826

Introduction

To improve account security at the University of Surrey, two-factor authentication is becoming a requirement for all users who wish to access services from off-campus locations.

Two-Factor Authentication is an extra step used to verify your identity when you log in. Traditional authentication is based on your **Username** (*something you are*) and **Password** (*something you know*). Two-Factor authentication uses both of these two methods but adds as third method, in this case, the **RSA Key-pair** (*something you have*).

FEPS IT has launched a new SSH service, **access.eps.surrey.ac.uk**. All users who access SSH Services from computers external to the University will be required to switch to using **access.eps.surrey.ac.uk**.

RSA Key-pairs

RSA Key-pairs work on the basis that a username or password can be stolen (without your knowledge), but a Token is something you have, and thus harder to steal. The RSA Key-pair consists of two elements, both of which are files stored on computers:

Public Key – This is what you distribute to the systems you wish to log in to. This part of the key is considered public, and thus can be stolen, transferred, or copied without any fears.

Private Key – This part of the key-pair is stored by you on the system you wish to connect from. If it is stolen then it is considered Compromised, and both the **Private Key** and **Public Key**'s must be replaced. To help protect the theft / use of a Private Key, it should always be protected by a pass phrase

Once the RSA Key-pair is in place, logging in will prompt you for your username. Once entered, you will be prompted for the Passphrase from your **RSA Private Key** (stored on your local computer). Once entered, you will be successfully logged in

Where can I use my RSA Key-Pair?

SSH Access to FEPS SSH - Access through the SSH gateway 'access.eps.surrey.ac.uk' from outside the campus network will **require** a two-factor authentication method. Internal access through this server **will not require** two-factor, however SSH will always prefer to use the RSA key-pair over the regular account password.

SSH Access to other Linux machines - Your RSA Key-Pair can be used in place of a password on almost any system that allows an SSH connection.

GITLab – Your RSA Key-Pair can be used with GitLab (gitlab.com or gitlab.eps.surrey.ac.uk)

IMPORTANT: If you are a user of GITLab and already use SSH to upload code, you will already have an RSA Key-Pair, so you may wish to skip to Step 5. It is highly recommended that your RSA Key-Pair is protected by a passphrase. If this is not the case, please consider regenerating your RSA Key-Pair for added security.

Step 1 – Prepare to generate your RSA Key-Pair

You will need to be on campus at the University of Surrey and logged onto a FEPS Linux machine running Ubuntu 16.04 (Xenial) or CentOS 7.3. This process will require the use of the Linux Command Line Interface (CLI).

FEPS Linux Machine → Start here:

- I Open a new Terminal Window. You are able to run the commands from your local machine.
- II. Now continue to the ‘Step 2 – Setting up the .ssh folder’ section on

University of Surrey Windows Machine → Start here:

- I. In a web browser, go to Surrey Software (<https://surreysoftware.surrey.ac.uk>) and sign in with your University of Surrey credentials. Choose PuTTY from the software list.
- II. Once PuTTY is installed, run the ‘PuTTY’ application from the Start Menu.
- III. Under ‘Host Name’ enter ‘access.eps.surrey.ac.uk’ and make sure the port is ‘22’. Now click open. Log in with your *Username* and *Password*
- III. Now continue to the ‘Step 2 – Setting up the .ssh folder’ section

Non-University of Surrey machines, or laptops on Wireless

You will be unable to set up your two-factor authentication by connecting to access.eps.surrey.ac.uk from a computer which is not on University of Surrey wired network, as doing so will require two-factor authentication.

In order to set up your two-factor authentication you will need to use a University of Surrey managed desktop machine (see above) or establish a VPN connection using <https://remote.surrey.ac.uk> (Staff only).

Step 2 – Setting up the .ssh folder

First we need to check if the .ssh directory exists in your home directory. Your Linux home path is denoted by the ~ (tilda) symbol. Please be sure to type all commands exactly and double check before you press Enter.

From the command line or terminal, run the following command to create your SSH directory and press Enter (take special note of the . ahead of ssh):

```
mkdir ~/.ssh
```

If you receive the error: mkdir: cannot create directory '/user/HS104/<username>/.ssh': File exists then your .ssh directory already exists.

Now run the following command to change directory (cd) into the newly created (or existing) .ssh directory

```
cd ~/.ssh
```

Now that you should be in your .ssh directory, we can check the contents of this to see if any existing key pairs exist. Run the following command to print a file list:

```
ls -lah
```

If your .ssh directory already existed, you should see output similar to the following:

```
[1:23pm] <computername>: > ls -lah
total 112K
drwx----- 1 <user> itsstaff 296 Oct 23 11:34 .
drwx----- 1 <user> itsstaff 8.0K Nov 7 12:19 ..
-rw----- 1 <user> itsstaff 394 Apr 7 2017 authorized_keys
-rw----- 1 <user> itsstaff 1.4K Oct 23 11:34 config
-rw----- 1 <user> itsstaff 1.8K Mar 1 2017 id_rsa
-rw----- 1 <user> itsstaff 394 Mar 1 2017 id_rsa.pub
-rw-r--r-- 1 <user> itsstaff 16K Nov 7 12:19 known_hosts
```

In this example we can see the .ssh directory has existed for some time and contains files. We can also see the existence of the *id_rsa* and *id_rsa.pub* files. If you have not previously set up an RSA Key-pair then these two files will not exist yet. If you already see them, you already have an RSA key-pair created.

id_rsa is your *private key*, while *id_rsa.pub* is your *public key*.

If you have *id_dsa* and *id_dsa.pub*, it is highly recommended that you delete these keys and recreate your keypair using RSA due to insecurities in DSA and its deprecation in later SSH versions.

Step 3 – Creating your RSA Key-Pair

Now that the relevant .ssh folder exists and we've confirmed that it does not contain existing RSA keys, it is time to generate a new RSA Key-pair.

WARNING: The following commands will delete any existing RSA keys.

From the command line, run the following command to begin your RSA Key-pair generation.

```
ssh-keygen
```

The following prompt will be displayed:

```
[1:38pm] <computername>: > ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key
(/user/<path>/<user>/.ssh/id_rsa):
```

The value described in brackets (/user/<path>/<user>/.ssh/id_rsa) is the default path. Press Enter to accept this path. Moving this to a different path may render the key-pair unusable.

Now you will be prompted to enter a passphrase. This secures your [private key](#):

```
Enter passphrase (empty for no passphrase):
```

WARNING: Do not leave this blank otherwise if your private key is stolen it could be used by the malicious party to gain entry to your account or any other computer containing your public key.

Please note the terminology 'passphrase' instead of 'password'. Using a passphrase increases the security of your key in the event it is copied or stolen from your account.

Please enter your passphrase now, then press enter.

Now you will be prompted to enter the passphrase again.

```
Enter same passphrase again:
```

Once you press Enter your key will be generated and the following information displayed to you:

```
Your identification has been saved in
/user/<path>/<user>/ssh/id_rsa.
Your public key has been saved in
/user/<path>/<user>/ssh/id_rsa.pub.
The key fingerprint is:
SHA256:9J5U1UjIYX+TL1gEvXkrKCCynM61aIM5qpDZ3AuR8T0
<user>@<computer>
The key's randomart image is:
+---[RSA 2048]---+
|                 .+*+o |
|                 .o+o o|
|      .      .      . o=.|
|      = o o . . . oo.+|
|      + = E S o o . .o|
|      = * . . + o . o |
|+ o + .      +      . |
|.+ B o                  |
|= o o                    |
+-----[SHA256]-----+
```

Your keys have now been successfully generated.

Under the directory `~/ssh` or `/user/<path>/<user>/ssh` you will have created the following files

- **id_rsa**
Your [private key](#), protected by your passphrase. This is stored on the machine you are connecting from. This must be kept confidential / private!
 - **id_rsa.pub**
Your [public key](#). This is stored in the account you are connecting to.
-

It is now important to ensure the permissions are correctly set for the `.ssh` folder that you created.

To do this, run the following commands:

```
chmod 700 ~/.ssh

chmod -R 600 ~/.ssh/*
```

The first command will set the correct permissions for the `.ssh` directory, while the second will set the correct permissions for all files in the directory.

If you attempt to use an RSA Private Key with an insecure `.ssh` directory you will receive an error stating that the key is not secure.

Step 4 – Deploying your Keys

As you have completed the previous steps on a University of Surrey machine, both your private key and public key will be stored in the correct area (`~/.ssh` or `/user/<path>/<user>/.ssh`).

In order to use this key-pair, you will need to move the [private key](#) to a computer you wish to connect from, and the [public key](#) must be located on the computer you wish to connect to in the `'authorized_keys'` file.

Private Key

WARNING: It is highly recommended you do not use an insecure service such as Email to transfer your key, but instead transfer the key using an offline method, such as a memory stick. If you have SSH set up on the machine you wish to connect from, then you are able to SCP to securely copy the key onto that machine.

On **Linux or Mac machines**, your private key ([id_rsa](#)) needs to be stored in the correct folder so that it can be accessed by the operating system. By default, this is the `~/.ssh` directory (`~/.ssh/id_rsa`), just like on the FEPS Linux machines. If you have previously used SSH on that account, the `.ssh` directory should already exist, but if it doesn't you can follow the steps in this guide to create it.

On **Windows Machines** - Your private key file ([id_rsa](#)) can be stored anywhere on the system, but usually somewhere inside of your Home Directory. When establishing an SSH connection using [PuTTY*](#), [WinSCP*](#) or [FileZilla](#) you can specify the path to your `id_rsa` file. For PuTTY / WinSCP you will be required to convert your keys to the `.ppk` format using PuTTYgen (part of the [PuTTY suite](#)).

Public Key

NOTE: It is safe to transfer your public key via email or any form of unencrypted communication. The public key is only used on the client machine and therefore would not be of any use to a malicious party. If you have SSH set up on the machine you wish to connect from, then you are able to SCP to securely copy the key onto that machine.

Your public key ([id_rsa.pub](#)) can be deployed to any account on any system that you wish to establish an SSH connection to. As SSH is exclusively a Unix/Linux service (including MacOS), the location for the public key should always be inside the `~/.ssh` directory, in the `'authorized_keys'` file (`~/.ssh/authorized_keys`). To add your key to this file, run the following command to append the key to the end of an existing `authorized_keys` file, or to generate a new file if it does not already exist

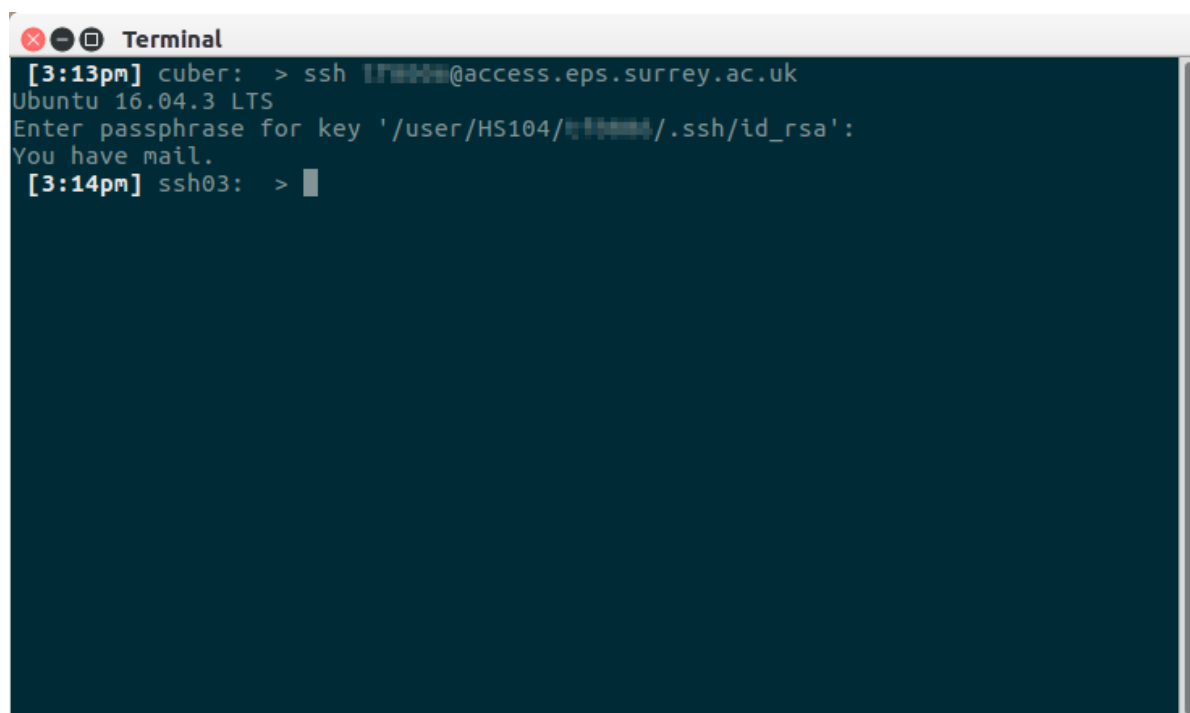
```
cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
```


Step 5 - Using your RSA Key-pair

You can test if your RSA Key-pair is working by establishing an SSH connection to **access.eps.surrey.ac.uk** from any computer containing your [RSA Private key](#). As your [RSA Public key](#) is located in the .ssh directory your FEPS Home area, logging onto any FEPS Linux machine will use the RSA Key pair over Username and Password.

During login, you should be prompted for:

- Username
- RSA [Private Key](#) passphrase



```
Terminal
[3:13pm] cuber: > ssh [REDACTED]@access.eps.surrey.ac.uk
Ubuntu 16.04.3 LTS
Enter passphrase for key '/user/HS104/[REDACTED]/.ssh/id_rsa':
You have mail.
[3:14pm] ssh03: > █
```

Now that you have successfully tested your RSA Key Pair, you are able to use the [Public key](#) on any system that you wish to SSH into, while the [Private key](#) must be on the machine you wish to connect from.

Fallback

SSH will always prefer to use Key-Pairs over Username and Password as it is considered more secure, but should your RSA Key-Pair not be available on the system it will fall back to password authentication.

Please note that fallback to username and password on access.eps.surrey.ac.uk is not available externally. If a Key-Pair is not available, then the system will fall back to Username, Password and Two-Factor Authentication token. It is recommended that all Key-Pair users still set up a two-factor authentication token.

Please follow the '[SSH Access with Two-Factor Authentication: Two-Factor Token](#)' guide to generate your token

FAQ

Q: I have replaced / reinstalled / lost my RSA Key-Pair and can no longer authenticate.

A: While on campus, please log in to a FEPS IT Linux machine or create an SSH connection to `access.eps.surrey.ac.uk` . Now follow the instructions from Step 2 to replace your token. You will need to ensure your new Private key is transferred to your laptop if you wish to authenticate from it.

If you are off-campus and require this to be reset, please check the FEPS IT website for the procedure.

Q: Occasionally I get disconnected when using SSH, resulting in the program I am running to stop working. Is there a way around this?

A: Yes. Using the command ‘screen’ followed by the command you wish to run will detach the command from your active session. This means that if your connection is lost, the screen / session containing your application is able to be restored. For a quick guide to using screen, visit here:

<https://www.mattcutts.com/blog/a-quick-tutorial-on-screen/>

Q: Can I log into `access.eps.surrey.ac.uk` and use it to run application X, Y, or Z?

A: FEPS IT have built the new SSH service with the most commonly used apps. The exceptions to this are CPU/Memory Intensive packages as the SSH servers should not be used for anything computational. Packages that are currently excluded from this list are:

- Mathematica
- Matlab
- Eclipse
- Maple
- R

If you wish to run computational applications, please establish your connection to `access.eps.surrey.ac.uk` before then creating another SSH connection from there to the machine you wish to run the application on (i.e. your own desktop or dedicated application or departmental server).

Please remember that the SSH servers are a shared resource open to every member of the FEPS faculty.

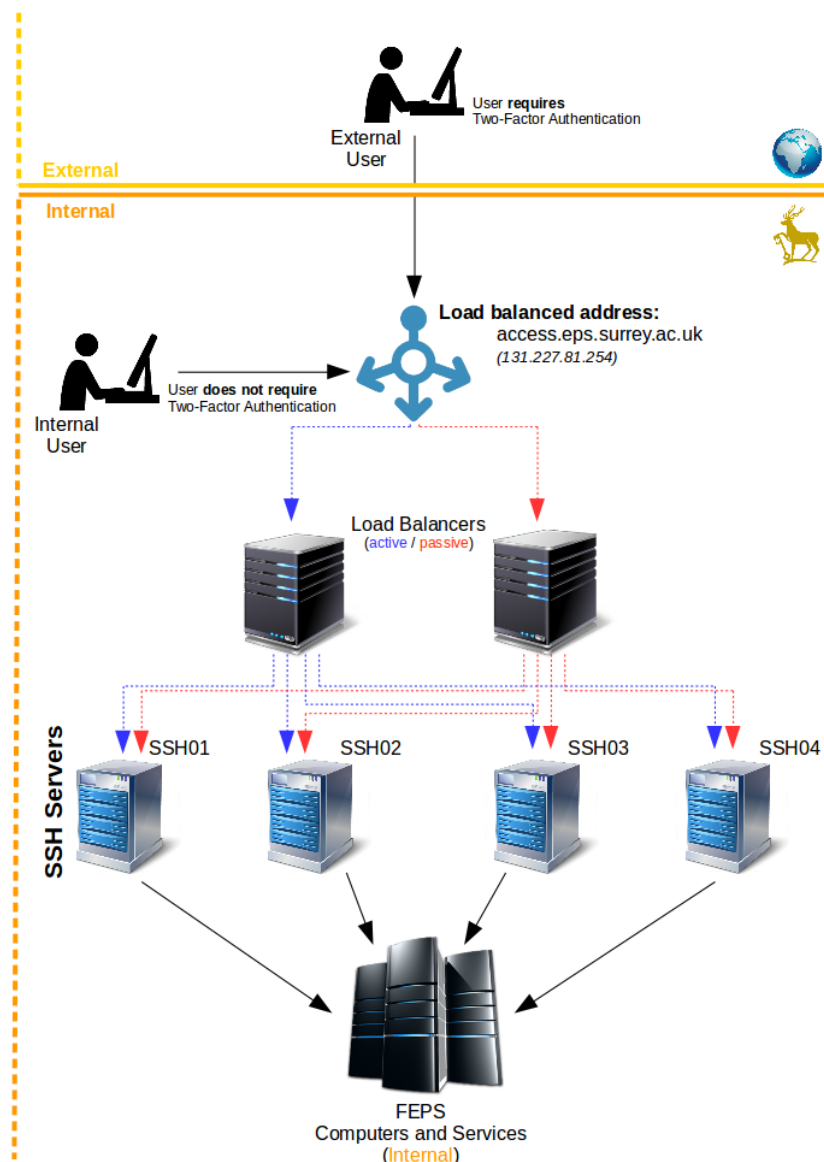
Q: Sometimes when I log in I get a different machine name. Is this normal behaviour?

A: In order to improve the reliability of our services, FEPS IT has implemented a small cluster of SSH servers. This allows us to share the load between servers and minimise any downtime caused by a single machine failing.

There are currently four SSH Servers running the access.eps.surrey.ac.uk service:

- SSH01
- SSH02
- SSH03
- SSH04

These servers sit behind a load-balancer which routes traffic accordingly, and also monitors the state of the servers, not routing traffic to any server which is offline.



If you are accessing from an external machine, you will be directed to **SSH01** or **SSH02**.

If you are accessing from an internal machine, you will be directed to **SSH03** or **SSH04**.

FEPS IT can add more SSH Servers in the future without any disruption to the service.

