SSH Key Generation and Login (PuTTY version)

Last update: Jul 9, 2025, tested with PuTTY 0.83 and minor fixes.

Introduction

The aim of this page is to explain how to login to RCSS supercomputer using PuTTY and PuTTYgen.

Install PuTTY

You can download PuTTY at its official site https://www.chiark.greenend.org.uk/~sgtatham/putty/). MSI installer version of the latest stable release available in this page maybe the most standard way to install PuTTY. If you already have PuTTY but not PuTTYgen, you can install PuTTYgen (puttygen.exe) from the "Alternative binary files" section of the download page.

SSH key generation (PuTTYgen)

Launch PuTTYgen



PuTTYgen can be launched from Windows start menu.

Key generation (1) - Key type

PuTTY Key Generator			?	×
File Key Conversions Help				
Key No key.				
Actions			3	
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				1
Type of key to generate: ○ RSA ○ DSA	⊖ ecdsa	• EdDSA	O SSH-1 (RSA)	_
Curve to use for generating this key:		2	Ed25519 (255 bits)	\sim

You may see PuTTYgen window like above.

1. Choose Key Type

Ed25519, ECDSA (256, 384 bits), and RSA 4096 bits of keys are recommended in RCCS. Please choose Ed25519 if you have no preference.

Notices

- Ed448 is not yet available at RCCS login servers. Please don't choose Ed448.
- ECDSA 521 bits keys are disabled due to the issue on PuTTY 0.68-0.80.
- To use RSA keys, PuTTY 0.75 or later is required.

2. Choose Key Length (ECDSA or RSA case)

In case of ECDSA or RSA type, you may find optional item at the position specified by "2". Please choose/input value there.

3. Start Generation of Key

Once you click the "Generate" button, the key generation will begin. You need to move mouse cursor around to proceed the key generation after clicking this button.

Key generation(2) - Set passphrase, then save the keys

PuTTY Key Generator ? × File Key Conversions Help						
Key Public kev for pasting into OpenSSH authorized kevs file: ssh-ed25519 eddsa-key- 20210524 1 Key fingerprint: esh-ed25519 255 SHA256: Key comment: eddsa-key-20210524 Key passphrase: Image: Confirm passphrase: Confirm passphrase: Image: Confirm passphrase: Generate a public/private key pair Generate	聲 PuTTY Key Genera	tor			?	\times
Public key for pasting into OpenSSH authorized keys file: ssh-ed25519 20210524 1 Key fingerprint: ssh-ed25519 255 SHA256: Key comment: eddsa-key-20210524 Key passphrase: Confirm passphrase: Generate a public/private key pair	File Key Conversion	s Help				
ssh-ed25519 eddsa-key- 20210524 1 Key fingerprint: ssh-ed25519 255 SHA256: Key comment: eddsa-key-20210524 Key passphrase: 000000000000000000000000000000000000	Key					
20210524 1 Key fingerprint: ssh=ed25519 255 SHA256: Key comment: eddsa=key=20210524 Key passphrase: Confirm passphrase: Confirm passphrase: Confirm passphrase: Generate a public/private key pair Generate		into OpenSSH authorize	d kevs file:			_
Key comment: eddsa-key-20210524 Key passphrase: Confirm passphrase: Confirm passphrase: Generate a public/private key pair					eddsa-key-	^
Key comment: eddsa-key-20210524 Key passphrase: Confirm passphrase: Confirm passphrase: Generate a public/private key pair			1			
Key comment: eddsa-key-20210524 Key passphrase: Confirm passphrase: Confirm passphrase: Generate a public/private key pair						\sim
Key passphrase: Image: Confirm passphrase: Confirm passphrase: Image: Confirm passphrase: Actions Image: Confirm passphrase: Generate a public/private key pair Image: Confirm passphrase:	Key fingerprint:	ssh-ed25519 255 SHA2	56:			
Confirm passphrase: 2 Actions Generate a public/private key pair Generate	Key comment:	eddsa-key-20210524				
Actions Generate a public/private key pair Generate	Key passphrase:	•••••		2		
Generate a public/private key pair Generate	Confirm passphrase:	•••••		2		
	Actions					
	Generate a public/priv	vate kev pair			Generate	
Load an existing private key file Load						
	Load an existing priva	te key file			Load	_
Save the generated key Save private key Save private key	Save the generated k	эу		Save public key	Save private key	
Parameters 3	Parameters				3	
Type of key to generate: ○ RSA ○ DSA ○ ECDSA ④ EdDSA ○ SSH-1 (RSA)			C ECDSA	() EdDSA	⊖ ssh-1 (RSA)	
Curve to use for generating this key: Ed25519 (255 bits)	Curve to use for sene	rating this key:		F	d25519 (255 bits)	\sim

Once the key generation completed, the appearance of the window will change like above.

1. OpenSSH type public key

The public key shown as a string in this field is what we need. Extract all the contents in this filed into notepad or others, and then save it! (Do not miss ssh-/ecdsa- part in the beginning!) Note: you don't need public key from "Save public key" button; we need only OpenSSH format one.

You can rebuild public keys via "Load" button or "Conversion" menu if you still have private key. (If you lost the private key, you need to generate a new key.)

2. Set passphrase for private key

You can set passphrase for private key here. We, RCCS, recommend passphrase of 10 or more characters contaning 4 types of characters - "lower-case", "upper-case", "number", and "symbol".

3. Save the private key

After setting passphrase, click "Save private key" button to save the key. Easy-to-understand name such as "rccs.ppk" or "ccfep.ppk" may be a good choice. (NOTE: THE PRIVATE KEY FILE MUST BE KEPT SECRET!)

Register Public Key (Common)

You need to register the public key before login. Please register your public key according to the instructions in https://ccportal.ims.ac.jp/en/account.

Please note that the public key you need is an OpenSSH type one (see above), not the one from "Save

Login (PuTTY)

Launch PuTTY and set destination

🔀 PuTTY Configuration	×
Category: Session - Logging - Terminal - Keyboard - Bell - Features - Window - Appearance - Behaviour - Translation - Colours - Colours - Colours - Colours - Colours - Colours - Serial - Telnet - Rlogin - SUPDUP	Basic options for your PuTTY session Specify the destination you want to connect to Host Name (or IP address) Connection type: (•) SSH Other: Telnet Load, save or delete a stored session Saved Sessions Default Settings Load Save Delete
About	Open Cancel

Launch PuTTY, input ccfep.ims.ac.jp in "Host Name (or IP address)" text box. (Do not push "Open" now!)

Specify user name

😹 PuTTY Configuration			×
Category: Session Logging Terminal Keyboard Bell Features Window Appearance Behaviour Translation Selection Colours Consection Data Proxy SSH Serial Telnet Rlogin SUPDUP	Login details Auto-login username When username is n	ot specified: Ise system username (
	Environment variable Variable [Value [S	Add Remove
About		Open	Cancel

Move to "Data" in "Connection" item. You can set username in "Auto-login username" textbox. Please inputRCCS user id (three-letter ID given by RCCS).

You can skip this step. You need to input username upon authentication in this case.

Specify Private Key File

😹 PuTTY Configuratio	n		\times
Category:	^	Credentials to authenticate with	
- Keyboard - Bell - Features Window		Public-key authentication Private key file for authentication: 	
Appearance Behaviour Translation		Certificate to use with the private key: Browse	
Election Colours Connection		Plugin to provide authentication responses Plugin command to run	
— Proxy ⊟ SSH — Kex			
Host keys Cipher Cipher			
-Credentials GSSAPI TTY X11			
About	~	Open Cancel	

Move to "Credentials" menu item in "Connection"=>"SSH"=>"Auth". Then, click "Browse" button and choose the generated private key file (.ppk).

If "Credentials" menu item does not exist, the private key file field may be found in "Connection"=>"SSH"=>"Auth" menu item.

Save Configuration 😹 PuTTY Configuration \times Category Session Basic options for your PuTTY session ^ Logging Specify the destination you want to connect to 📄 Terminal Host Name (or IP address) Port Kevboard 22 Bell ccfep.ims.ac.jp Features Connection type: 📄 Window Appearance ● SSH ── Serial ── Other: Telnet Behaviour Translation -Load, save or delete a stored session 🗄 Selection Colours RCCS - Connection Data Default Settings 2 .oac Proxy IRCOS 📄 SSH Save -Kex Host kevs Delete Cipher 📄 Auth -Credentials Close window on exit: GSSAPI ⊖ Never) Always • Only on clean exit TTY -X11 3 Open

Although you can connect to RCCS servers by clicking "Open" button, you should save the configuration before trying to connect.

Cancel

Move back to "Session", and give a name to this connection in textbox ("1"), then click "Save" button to save it ("2"). The saved connection name would appear in the list. Finally, click "Open" to begin connection.

The second and subsequent connections can be completed without changing the settings by loading the saved settings. In the case of the window above, click "RCCS" in the larger box (underlined with green), then click the "Load" button, and then click the "Open" button.

Alert message upon first connection

About



Dialog like above would be shown upon first connection.Please check the fingerprint (1); this must match with either of the fingerprint listed below.

- ad:de:79:30:81:b0:b1:6a:17:f5:6f:ea:f4:b4:3b:de (MD5)
- e3:a9:bb:6f:e4:a9:37:fd:98:14:54:16:3c:81:2d:aa (MD5)
- 07:7e:df:7b:16:47:a8:f5:7c:48:b0:a3:d3:86:71:6a (MD5)
- wnEM30z4AxyDJ9XI/DdGr2PINeoivFRR8v5krXHEmdU (SHA256)
- 0KL38Yn/kBee1pAuxyKwenEwXjtPxr9ZElolfVqXvbI (SHA256)
- Nhg+9Lgj3XeuW//A/j7jqgUJllxWehryCtStlp1Dir (SHA256)

Click Ok button if the fingerprint is a valid one.

Window like above would be shown if connection to server is succeeded. You then need to input passphrase to use your private key.

(Note: if you register the key to Pageant beforehand, you are able to skip this step.)

Login Completed!



If your key settings are OK, you can login to our login server like above.