

BETELGEUSE 2FA (TWO FACTOR AUTHENTICATION)

The 2FA is a secure manner to connect to a remote ssh service. It combine or requires something that you known (password) and a thing that you own (smarthphone or computer).

In order to adapt our access portal 'betelgeuse' to the ICIQ security access policy, have been enabled a 2FA system. So from now onwards you must to use it if you are connecting through betelgeuse from a public or shared computer. The private/public key is also allowed in your personal computers but is mandatory to provide a passphrase when you generate them.

You can follow the next steps and advices in order to configure the 2FA access mode.

METHOD-1 : TIME-BASED ONE TIME PASSWORD

STEP-1 – INSTALLING AN AUTHENTICATOR IN YOUR DEVICE

SMARTPHONE

- Install the Google Authenticator app (you can use other) in your smartphone:
Google Authenticator in [Play Store](#) or [Apple Store](#).

COMPUTERS

In the computer:

LINUX OS

Open a new terminal and install 'oathtool':

DEB based distros:

```
$ sudo apt-get install oathtool
```

RPM based distros:

```
$ sudo zypper in oath-toolkit
```

or

```
$ sudo yum install oathtool
```

WINDOWS OS

Install [WinAuth](#) in your Windows OS ([instructions](#))

MAC-OSX OS

Install [G2FA Google Authenticator](#) in your OSX

LIKE A BROWSER EXTENSION

CHROME BROWSER EXTENSION

Install the chrome extension [GAuth Authenticator](#) extension in the Chrome browser ([instructions](#)).

FIREFOX

Install the firefox [Authenticator](#) extension in the Firefox browser.

STEP-2 – GENERATING THE AUTH CODES AND KEY

Open a new terminal in ‘betelgeuse’ portal and execute the next command:

```
$ google-authenticator
```

the command will return an URL (link to a QR image) and some key/codes. Save this information in a secure place. In addition it ask you some questions, reply ‘y’ all of them. Finally a new hidden file ‘.google_authenticator’ will be created in your home directory, do not change content, name nor delete it.

```
mante@betelgeuse:~ $ google-authenticator
Do you want authentication tokens to be time-based (y/n) y
https://www.google.com/chart?chs=200x200&chld=M|0&cht=qr&chl=otpauth://totp/mante@betelgeuse%3Fsecret_key=FZFSIALSDOLZPOT6&issuer=3Dbetelgeuse
Your new secret key is: FZFSIALSDOLZPOT6
Your verification code is 578820
Your emergency scratch codes are:
92203957
73671260
85484466
14641792
85632082
```

STEP-3 – SETTING A NEW ACCOUNT IN THE AUTHENTICATOR

Open the authenticator application installed in the Step-1.
Add a new account using the QR code (copy paste URL in a browser) or using the SECRET_KEY.
A new account will be created and will be shown the VALIDATION CODE (it change each 30 seconds but remains valid 1½ minutes).

Note: for linux command line authenticator, use the next command:

2FA on betelgeuse

```
$ oathtool --totp -b 'SECRET_KEY'
```

be aware that it will be readable in your `.bash_history` file.

STEP-4 – TRYING TO VALIDATE IN ‘betelgeuse’

Open a ssh connection to betelgeuse and it ask you for ‘VALIDATION CODE’ in the first time and your user password after. You are completed the 2FA mechanism.

NOTE: the AUTHENTICATOR time based (TOTP - Time-based One-Time Password) use the time for validating, so the ‘SECRET_KEY’ (or QR) provided by betelgeuse is used to install a new account in the Authenticator device allowing to be synchronized.

USING EMERGENCY CODES

If you are connecting without possibility of generating the VERIFICATION_CODES (with an Authenticator) you can use the one-time-use EMERGENCY_CODES yet. When use any of them it will be disabled and automatically will disappear from the `$HOME/.google_authenticator` file. You easilly can generate more Emergency codes using the command `google-authenticator` (without continuig the process answering ‘no’ or using Ctrl+C) in a betelgeuse terminal and copying them at bottom of the `.google_authenticator` file.

```
mante@betelgeuse:/usr/lib $ google-authenticator
Do you want authentication tokens to be time-based (y/n) y
https://www.google.com/chart?chs=200x200&schld=Mj0&cht=qr&chl=otpauth://totp/mante@betelgeuse%3Fsecret%3DEXCDJOIST2WGJQH0%26issuer%3Dbetelgeuse
Your new secret key is: EXCDJOIST2WGJQH0
Your verification code is 233321
Your emergency scratch codes are:
47533791
25351975
54384421
56499966
80768964
Do you want me to update your "/home/mante/.google_authenticator" file (y/n) n
```

← Copy to `.google_authenticator` file

← IMPORTANT: stop processing

RECOVERING THE 2FA METHOD

If you have lost your device or deleted the betelgeuse account in your Authenticator, you can easily recover it using the original QR code or SECRET_KEY. You can also contact with the betelgeuse manager by e-mail (`mgumbau@iciq.cat`).

METHOD-2 : PRIVATE AND PUBLIC KEYS

(FOR PERSONAL/PRIVATED COMPUTERS)

Key files method is based on Private and Public key files plus an additional Passphrase.

2FA on betelgeuse

The PRIVATE_KEY should be stored in a a secure place (Personal Computer) and the PUBLIC_KEY will be copied in the remote computer where you want to connect to.

LINUX OS

Open a new terminal in your personal computer.

Execute the command:

```
$ ssh-keygen -t rsa
```

it ask you the location for the key files. Default location is right (\$HOME/.ssh).

Now you must enter two times a secure and unforgettable passphrase.

That's all in the client side.

Now copy the public key (\$HOME/.ssh/id_rsa.pub) file content to the remote server 'betelgeuse':

```
$ ssh-copy-id -p2004 -i ~/.ssh/id_rsa.pub yourusername@betelgeuse.iciq.es
```

WINDOWS

- With MobaXTerm: [Link1](#) or [Link2](#)
- With Putty: [Link1](#)

MAC-OSX

- Similar to linux OS: [Link1](#)

Note: if you are connecting to 'betelgeuse' from a personal computer where you have stored the PRIVATE_KEY, betelgeuse will ask you for the passphrase you have generated. In other case 'betelgeuse' will ask you for the VERIFICATION_CODE and user password.

Questions to: mgumbau@iciq.cat