# SSH

use **id_ed25519** over rsa (toolshelf.tech)

## ssh public key auth

- generate key: `ssh-keygen`
- copy ssh key to remote:
  - windows: `type $env:{userProfile}\.ssh\id_ed25519.pub | ssh -p{port} {user@serverAddress} "cat >> .ssh/authorized_keys"`
  - linux: `ssh-copy-id -i {userHome}/.ssh/id_ed25519.pub -p{port} {user@serverAddress}`
- try login: `ssh -p{port} {user@serverAddress}`
- the public key should now be on in the file `~/.ssh/authorized_keys` on the server

# workflow

# add your key to a single server

- create keys
  - on linux use `ssh-keygen -C "{name or mail}"`
  - create a public/private ed25519 key
  - use a passphrase
  - identification / private key: `~/.ssh/id_ed25519`
  - public key: `~/.ssh/id_ed25519.pub`
- copy ssh key to remote:
  - windows: `type $env:{userProfile}\.ssh\id_ed25519.pub | ssh -p{port} {user@serverAddress} "cat >> .ssh/authorized_keys"`
  - linux: `ssh-copy-id -i {userHome}/.ssh/id_ed25519.pub -p{port} {user@serverAddress}`
- try login: `ssh -p{port} {user@serverAddress}`

# ssh basics

## keys

## generating using openSSH

- `ssh-keygen  -C "{name or mail}"`
  - generates by default a id_ed25519
  - `-C "{name or mail}`
- save your public key, private key and passphrase on a save place

## upload key

- `ssh-copy-id -i ~/.ssh/id_ed25519 -p222 user@host`
  - `-i ~/.ssh/id_ed25519`
  - `-p222`
  - `user@host`
- windows: `type $env:USERPROFILE\.ssh\id_ed25519.pub | ssh {IP-ADDRESS-OR-FQDN} "cat >> .ssh/authorized_keys"`

## restore keys (on a new pc)

- copy `id_ed25519` and `id_ed25519.pub` to `~/.ssh/`
- set correect permissions `sudo chmod 400 ~/.ssh/id_ed25519*`

# files

## server side

### config file

parts of `/etc/ssh/sshd_config`:

```
Port = 22   # self explaining
PermitRootLogin = no   # should be 'no'
PasswordAuthentication no   # disallow Username-Password login
ClientAliveInterval 300   # inactivity time period after which the server send an alive message
ClientAliveCountMax 3   # number of attempts the server will make
```

### auth keys `authorized_keys`

https://www.ssh.com/academy/ssh/authorized-keys-file
https://www.ssh.com/academy/ssh/authorized-keys-openssh

- server side
- list of pub keys

# client side

## config (client side)

- before client config: `ssh john@dev.example.com -p 2322`
- edit client config `~/.ssh/config`:

```
Host devNetcup
    HostName dev.example.com
    User john

Host *Netcup
    Port 222

Host *
    ForwardAgent yes
    ServerAliveInterval 300
```

- after client config: `ssh dev`

# ssh agent

```
eval "$(ssh-agent -a "$HOME/.ssh/agent.sock")"
ssh-add ~/.ssh/id_ed25519
ssh-add -l
```

- `ssh-agent -a "$HOME/.ssh/agent.sock"` - sets the agent socket file to a dedicated file
- `ssh-add -l` shows the added keys

---

Created 2022-10-29 22:29:10 UTC by Daniel Raab
Updated 2026-05-12 07:54:01 UTC by Daniel Raab