NHR FAU

FAU Friedrich-Alexander-Universität Erlangen-Nürnberg

Secure system access with SSH

HPC Café

2024-04-09

SSH overview

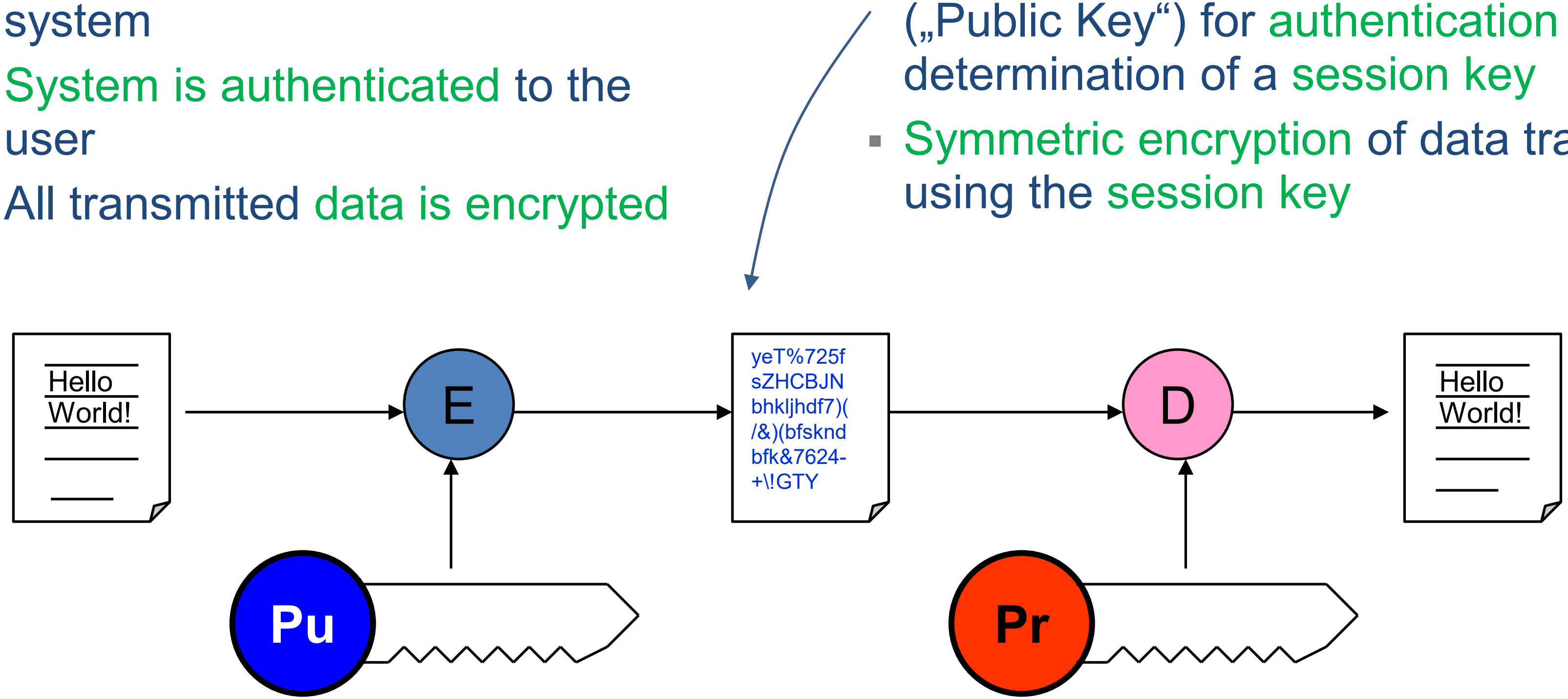
SSH is a cryptographic network protocol

- Designed in 1995 by Tatu Ylönen (researcher at Helsinki University of Technology, Finland)
- SSH is standardized by an Internet Engineering Task Force (IETF) working group
- OpenSSH (an OpenBSD project) is the most common Open Source implementation
  - Available on Linux, Windows, MacOS
- Recommended Windows client:
  MobaXterm Home Edition
  `https://mobaxterm.mobatek.net/download-home-edition.html`

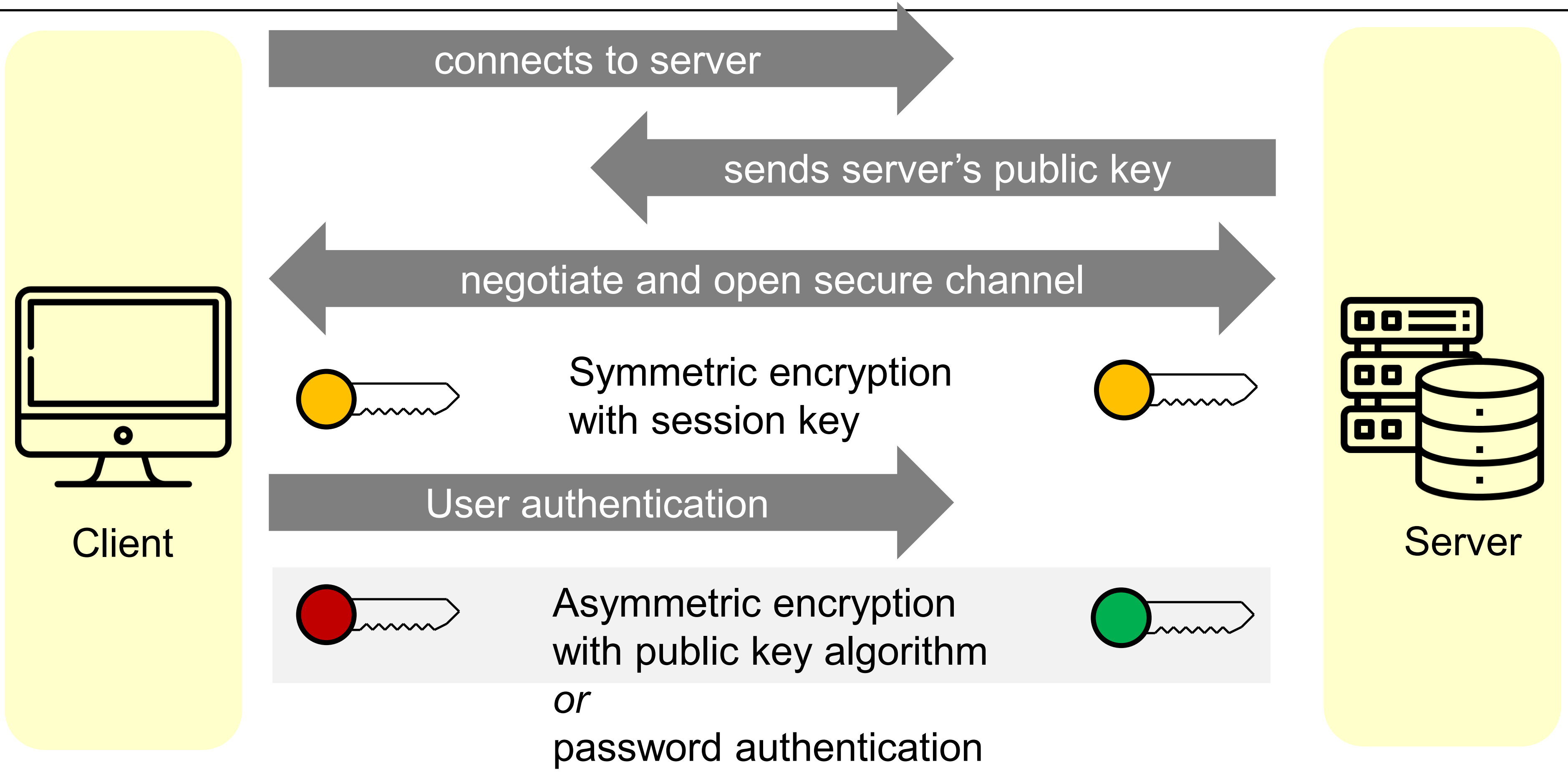
SSH: How does it work?

- “Secure” means
  - User is authenticated to the system
  - System is authenticated to the user
  - All transmitted data is encrypted

- Technology
  - Asymmetric encryption algorithm („Public Key“) for authentication and determination of a session key
  - Symmetric encryption of data transfer using the session key



SSH: How does it work?



Client

connects to server

sends server’s public key

negotiate and open secure channel

Symmetric encryption
with session key

User authentication

Asymmetric encryption
with public key algorithm
or
password authentication

Server

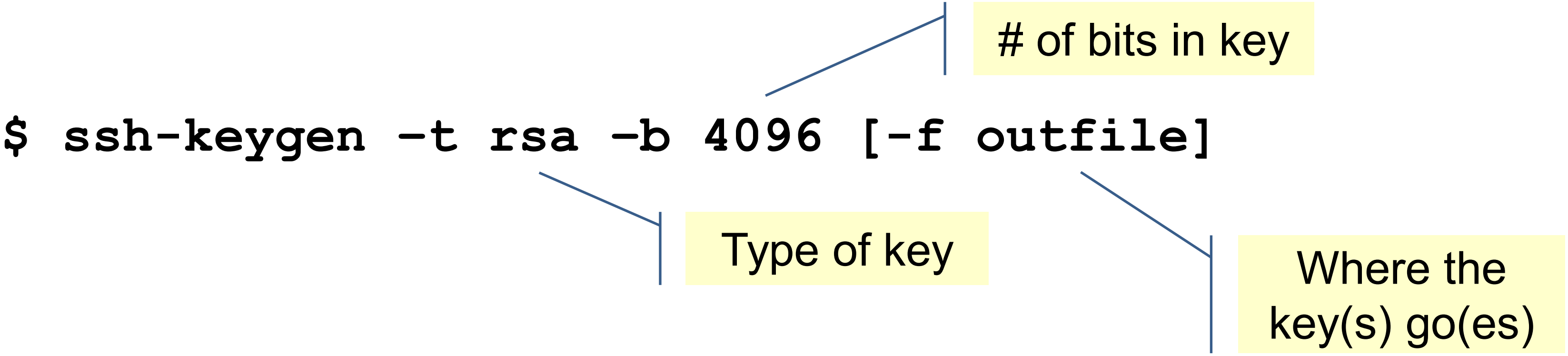
Questions (& quick answers)

- How do I make a public/private key pair?
  - **`ssh-keygen`**
- How do I store the private key?
  - Encrypted, on your local system
- How many key pairs do I need?
  - One for each client computer
- Where do I put the public key?
  - In an appropriate place on the remote system
- Can I transfer files, too?
  - Yes, with **`scp`** (and other tools)
- How do I connect from A to C via B with least hassle?
  - Via “proxy jump”

How to make a key pair

Accepted key formats (by us):
RSA (at least 4096 bit)
ECDSA (at least 512 bit)
ED25519

- Recommended options for key generation:



```
$ ssh-keygen -t rsa -b 4096 [-f outfile]
```

# of bits in key

Type of key

Where the key(s) go(es)

- Standard files and location:

```
$ ls -l ~/.ssh
-rw------- 1 unrz55 unrz  1766 Apr 10  2023 id_rsa
-rw-r--r-- 1 unrz55 unrz   395 Apr 10  2023 id_rsa.pub
```

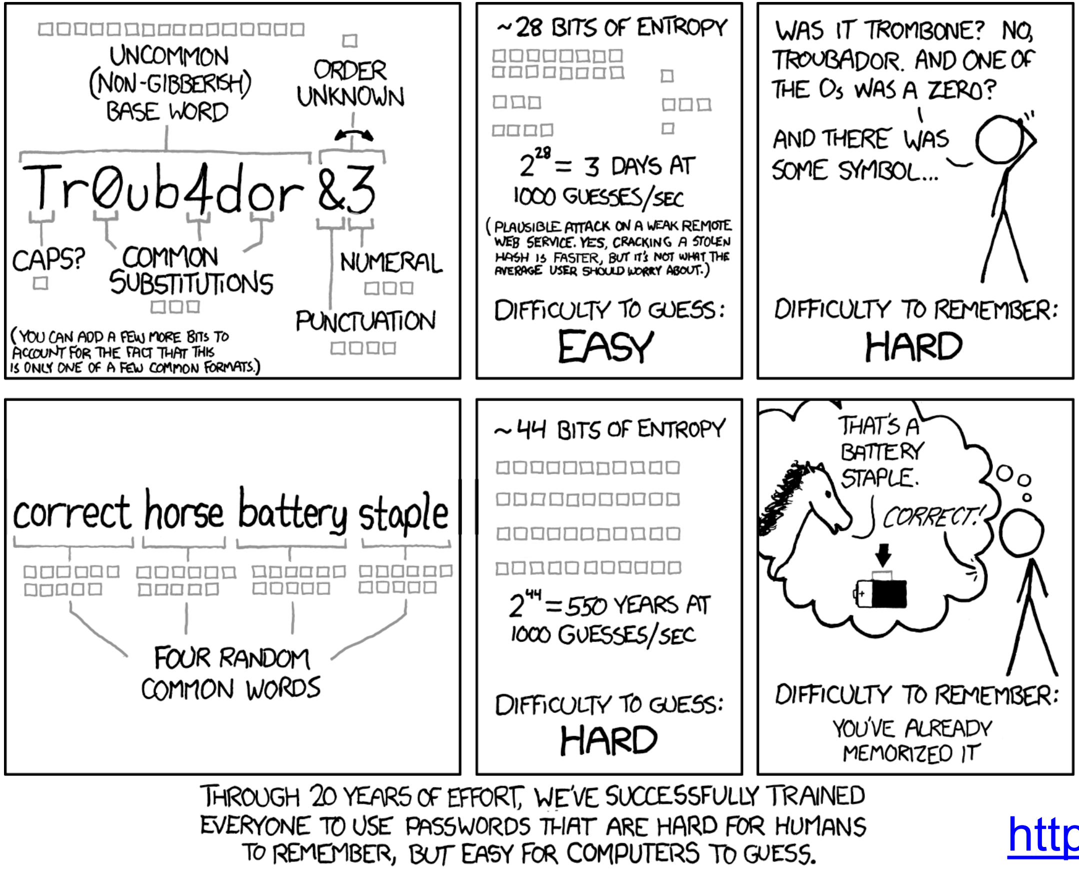
Handling the private key(s)

- The private key is secret!
  Anyone who has your private key can log in as you

- When generating the key pair, you are asked for a passphrase
  - This is how your private key is protected (encrypted)
  - It is still a good idea to protect the key from others

- If a client is compromised, assume that the private key is exposed
  - Use one key pair per distinct client system

Choosing a passphrase (or any password)


UNCOMMON (NON-GIBBERISH) BASE WORD
ORDER UNKNOWN
Tr0ub4dor&3
CAPS?
COMMON SUBSTITUTIONS
NUMERAL
PUNCTUATION
(YOU CAN ADD A FEW MORE BITS TO ACCOUNT FOR THE FACT THAT THIS IS ONLY ONE OF A FEW COMMON FORMATS.)
~28 BITS OF ENTROPY
2^28 = 3 DAYS AT 1000 GUESSES/SEC
(PLAUSIBLE ATTACK ON A WEAK REMOTE WEB SERVICE. YES, CRACKING A STOLEN HASH IS FASTER, BUT IT'S NOT WHAT THE AVERAGE USER SHOULD WORRY ABOUT.)
DIFFICULTY TO GUESS: EASY
WAS IT TROMBONE? NO, TROUBADOR. AND ONE OF THE Os WAS A ZERO?
AND THERE WAS SOME SYMBOL...
DIFFICULTY TO REMEMBER: HARD
correct horse battery staple
FOUR RANDOM COMMON WORDS
~44 BITS OF ENTROPY
2^44 = 550 YEARS AT 1000 GUESSES/SEC
DIFFICULTY TO GUESS: HARD
THAT'S A BATTERY STAPLE.
CORRECT!
DIFFICULTY TO REMEMBER: YOU'VE ALREADY MEMORIZED IT
THROUGH 20 YEARS OF EFFORT, WE'VE SUCCESSFULLY TRAINED EVERYONE TO USE PASSWORDS THAT ARE HARD FOR HUMANS TO REMEMBER, BUT EASY FOR COMPUTERS TO GUESS.

https://xkcd.com/936/

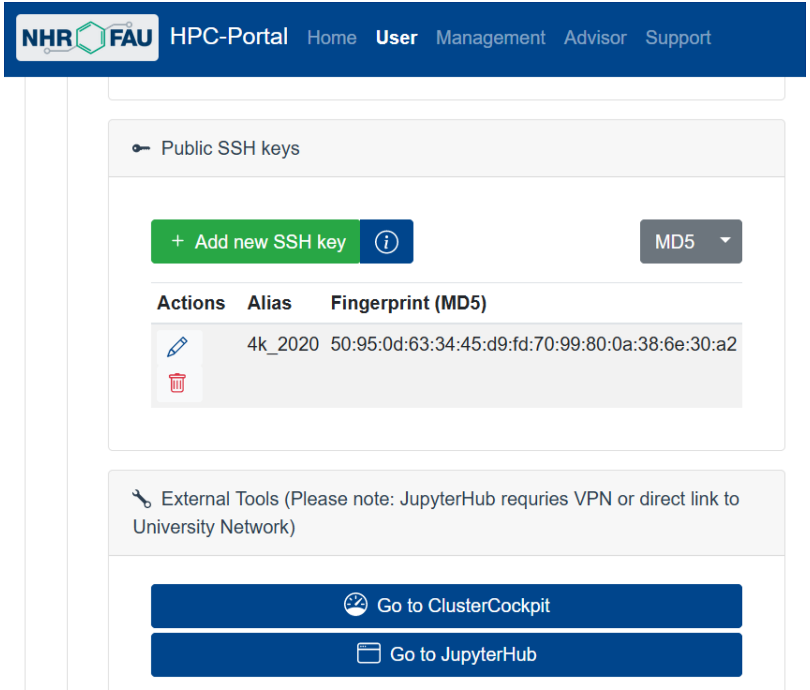
Where to put the public key, and how

- Standard environment: transfer public key to server

```
$ ssh-copy-id -i ~/.ssh/mykey.pub user@host
```

  - Public key added to `~/.ssh/authorized_keys`

- NHR@FAU: Upload SSH key to HPC Portal
  https://doc.nhr.fau.de/hpc-portal/#the-user-tab
  - Give it two hours to distribute to all systems



Actions	Alias	Fingerprint (MD5)
	4k_2020	50:95:0d:63:34:45:d9:fd:70:99:80:0a:38:6e:30:a2

Actually logging in with a private key

- Example:

```
mylaptop$ ssh unrz55@fritz.nhr.fau.de
Enter passphrase for key '/home/unrz55/.ssh/id_rsa':
[...massive babble...]
unrz55@fritz3:~ $
```

- Can add explicit private key with "`-i <keyfile>`"
- Use option "`-X`" for X11 forwarding
  - Display GUI through SSH connection
  - Probably painful over DSL/Cable
  - Better use remote desktop options, especially if you are outside FAU
    https://doc.nhr.fau.de/access/nx/ (phasing out)
    https://doc.nhr.fau.de/access/xrdp/ (the new sh!t)

Transferring files

- Built-in command: `scp`
  - To target:
    ```
    $ scp [-r] [-p] <local_source> target_host:[path]
    ```
  - From target:
    ```
    $ scp [-r] [-p] target_host:[path] <local_target>
    ```
  - Example:
    ```
    $ scp -r -p iwst345h@fritz.nhr.fau.de:work/\*.dat ~/data
    ```

    Recurse into subdirs

    Preserve mod. times and permissions

    Remote wildcard
- GUI frontends
  - WinSCP
  - MobaXterm file browser
  - …

Making your life easier: SSH agent

- SSH agent is a service daemon that remembers your private keys
- Usually started by desktop environment
- Adding a specific key to the agent (lifetime 1 day):
  ```
  $ ssh-add -t 86400 ~/.ssh/id_rsa_laptop
  ```
- In the following, no passphrase is necessary for login

- Agent forwarding enables passphrase-free logins out from the remote host
  - No need to deploy private keys remotely (only public keys necessary)
- Caveat: Authentication can be hijacked and is forwarded to a potentially untrusted remote environment

Our advice: Do not use agent forwarding!

Making your life easier: the ssh config file

- Location: `~/.ssh/config`
- Allows to create shortcuts to hosts and adjust ssh settings per host
  Documentation: `$ man ssh_config`

- Example entry:

```
Host csnhr
  ForwardAgent no
  ForwardX11 no
  HostName csnhr.nhr.fau.de
  User unrz55
  IdentityFile /home/unrz55/.ssh/id_rsa_laptop
```

Proxy Jump

- Proxy jump enables login through a “jump host”
- The connection is tunneled through the jump host but the connection to final target host is made by the initial client
- Necessary for logins to NHR@FAU systems from outside FAU if no VPN or IPv6 is available

- Basic use:

Jump host

Target host

```
$ ssh -J unrz55@csnhr.nhr.fau.de unrz55@fritz.nhr.fau.de
Enter passphrase for key '/home/pi/.ssh/id_rsa':
Enter passphrase for key '/home/pi/.ssh/id_rsa’:
unrz55@fritz:~ $
```

Suggested SSH config for Proxy Jump

- Full config: https://doc.nhr.fau.de/access/ssh-command-line/?h=proxy+jump#template-for-connecting-to-hpc-systems

```
Host csnhr.nhr.fau.de
  HostName csnhr.nhr.fau.de
  User <HPC account>
  IdentityFile ~/.ssh/<your_private_key>
  IdentitiesOnly yes
  PasswordAuthentication no
  PreferredAuthentications publickey
  ForwardX11 no
  ForwardX11Trusted no
```

```
Host alex.nhr.fau.de
  HostName alex.nhr.fau.de
  User <HPC account>
  ProxyJump csnhr.nhr.fau.de
  IdentityFile ~/.ssh/<your_private_key>
  IdentitiesOnly yes
  PasswordAuthentication no
  PreferredAuthentications publickey
  ForwardX11 no
  ForwardX11Trusted no
```

If this is all Greek to you, use VPN
(or a GUI session on csnhr)!

Security hints for SSH clients

- Keep the private key files secret!
  - If possible, put private keys only on trusted hosts

- Use a “long enough” key protected by a passphrase
- Use a strong passphrase (at least 15 characters long)

- Use a separate key for every client
- Disable Agent Forwarding and X11 Forwarding in config
- Do not leave open external logins in running tmux/screen
- Keep your SSH client installation up to date

Some hints for NHR@FAU HPC systems

- Your home directory is an NFS share
  - Special care w.r.t. private keys required
  - Take care of proper permission settings

- Recommendation: Use a dedicated key pair for NHR@FAU-internal logins
  - Mostly hostkey based anyway
  - Do not use this key to login anywhere else

What we have left out

- Managing passphrases and passwords
  - Password managers: KeePass, Pass, gopass, …
- General port forwarding through SSH connections
  - “Poor man’s VPN”
  - `$ ssh -L 8888:proxy:80 unrz55@cshpc`
- SSHFS
  - Mounting remote file systems over SSH connections
  - `$ sshfs -o <options> cshpc:/home/vault/unrz/unrz55 ~/vault`
- Graphical remote desktops
  - XRDP via csnhr
  - https://doc.nhr.fau.de/access/xrdp/