RMOUG

ROCKY MOUNTAIN ORACLE USERS GROUP

Training Days Conference

February 15 - 17, 2023

Be Informed and Inspired to Initiate new Ideas to Improve your work lives.

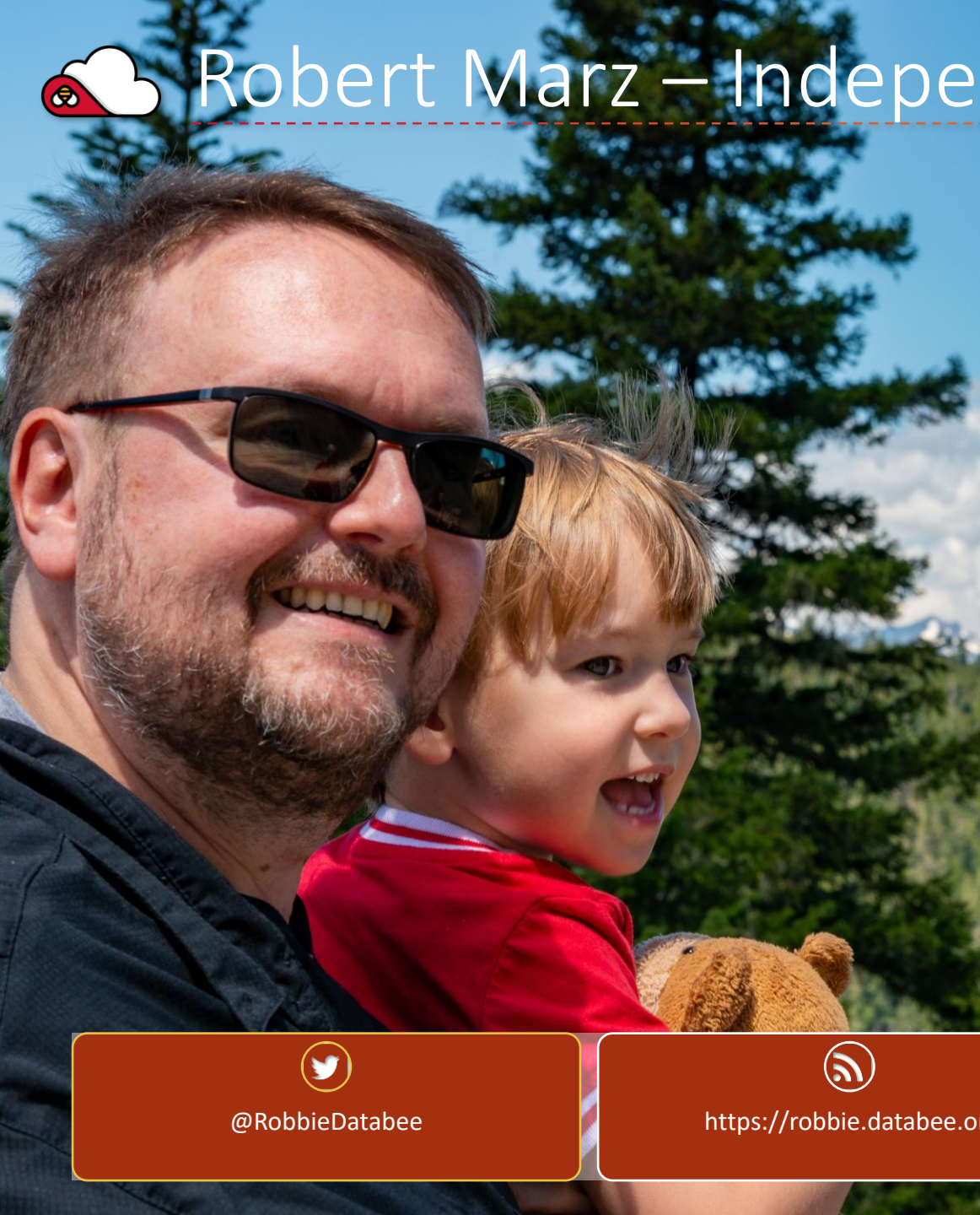
Tools for DBAs and Cloud User: The Secure Shell (ssh)

Robert Marz

DATABEE

Die IT-Architekten

Robert Marz – Independent Consultant



Primary Role

Senior Technical Architect
with database centric view of the world

DOAG (German Oracle User Group)

Active Member of Database Community
Responsible for Cloud Topics

Databees.

DATABEE
Die IT-Architekten

Oracle ACE Pro

@RobbieDatabee

https://robbie.databee.org

robert.marz@databee.org

Oracle ACE Program

500+ technical experts helping peers globally

The **Oracle ACE Program** recognizes and rewards community members for their technical and community contributions to the Oracle community

3 membership tiers

Oracle ACE Director | Oracle ACE Pro | Oracle ACE Associate

Oracle ACE

Nominate

yourself or someone you know:

ace.oracle.com/nominate

For more details on Oracle ACE Program:
ace.oracle.com

Connect: aceprogram_ww@oracle.com Facebook.com/OracleACEs @oracleace

ssh – the secure shell

ssh – the secure shell

Network Protocol

encrypted

Client-Server

Client: ssh
Server: sshd

Open Source

OpenSSH

Incompatible Protocols

SSH-1 : Original back in1996
SSH-2: IETF Standard

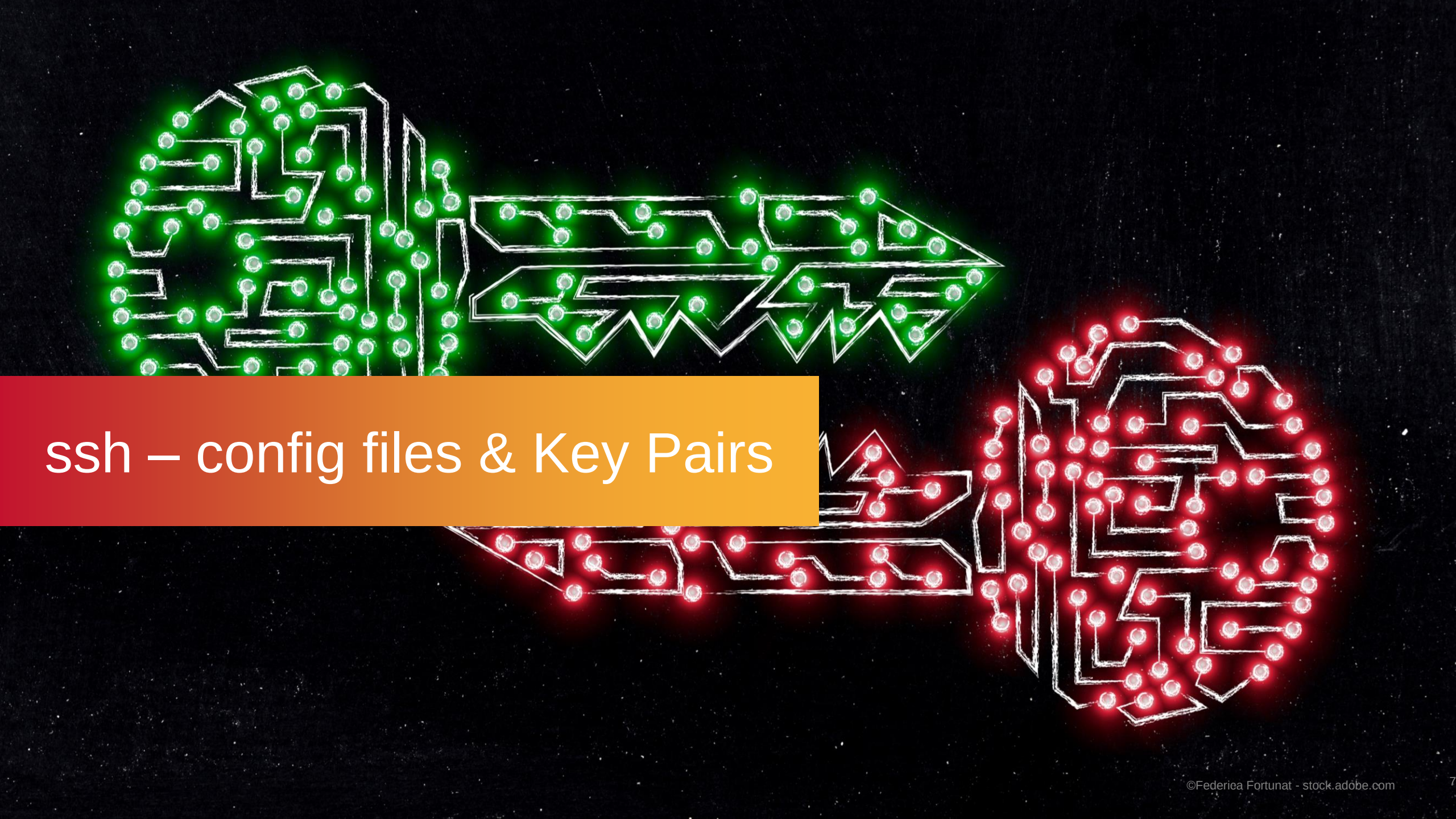
ssh suite – a mighty Tools Suite

Defacto standard for console connections

- Replaces telnet, rlogin, rsh, ftp, etc.
- Machine to Machine
- SSL encrypted

There is way more:

- Automated Login
- Network tunnel
- File transfer
- Remote Filesystem
- Tunnel chains
- Jump Hosts



ssh – config files & Key Pairs

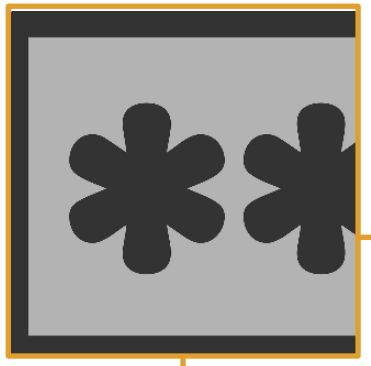
Configuration Files

Global in /etc, per User in ~/.ssh

Permissions r-- or rx--

- Server (sshd)
  - sshd_config
  - authorized_keys*
  - sshrc
- Client (ssh)
  - config*
  - known_hosts*
  - Keys*
    - Private*
      - identity
      - id_dsa
      - id_rsa
      - ...
    - Public*
      - identity.pub
      - id_dsa.pub
      - id_rsa.pub
      - ...
  - sshrc
  - environment

Passwords or Key Pairs?



Passwords

- Remember
- short
- Keylogger
- Migrate (plain text or hash)
- Simple usage

Key Pair

- Files
- Complex
- Private Key only local
- A little complicated

Config Files on the client: known_hosts

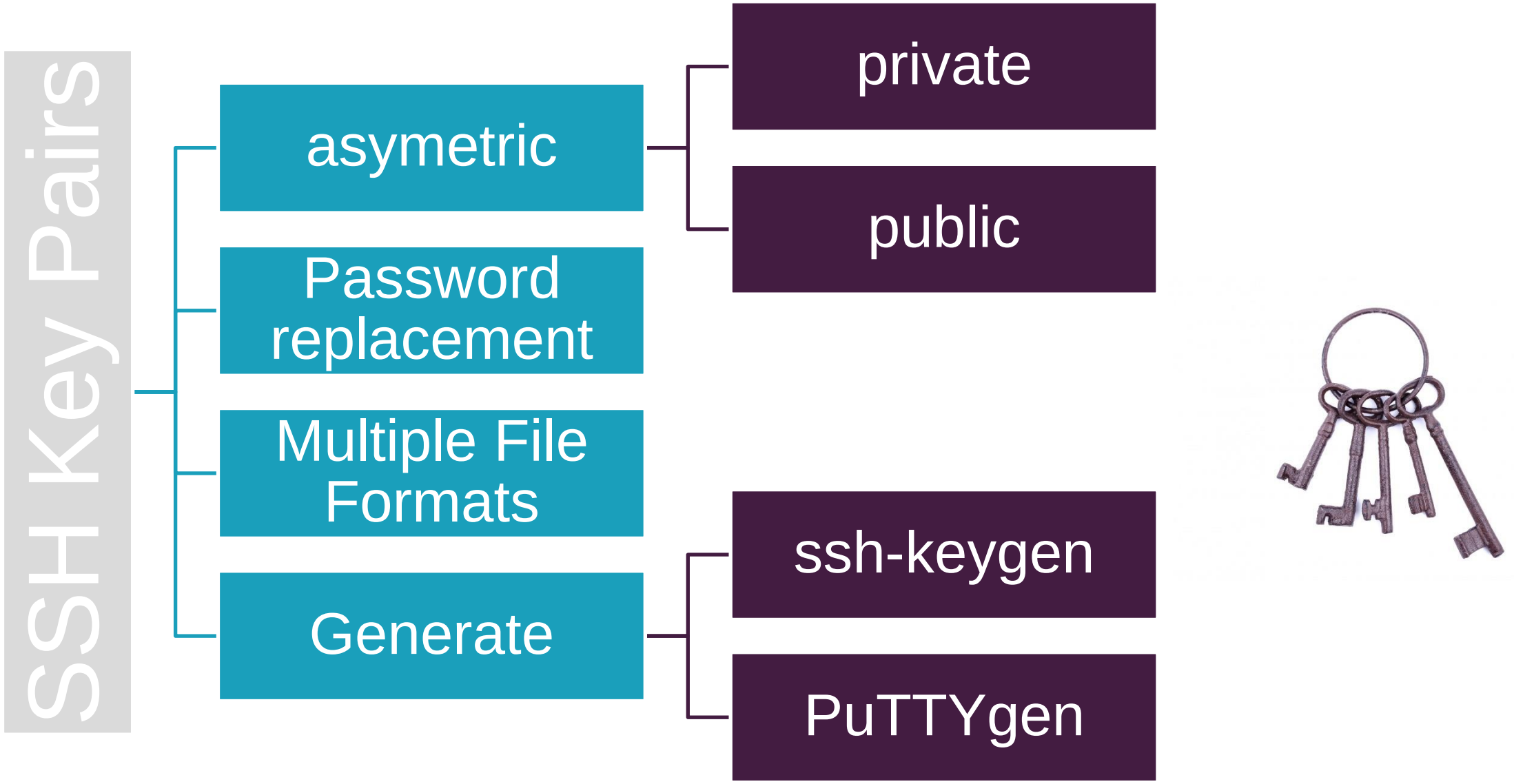
known_hosts (Client)

```
[rmarz@niteowl ~]$ ssh no-web
The authenticity of host 'no-web (10.145.176.14)' can't be established.
RSA key fingerprint is 49:29:a4:ac:5b:f0:6e:5c:83:40:68:a8:77:bc:32:e3.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'no-web' (RSA) to the list of known hosts.
```

Watch list of known servers
position of trust (Attention!)
Password transmission in clear text
Known exploits

```
Warning: the RSA host key for 'no-web' differs from the key for the IP address '10.145.176.14'
Offending key for IP in /home/rmarz/.ssh/known_hosts:17
```

SSH Key Pairs



Config files on the server: The Public Key

Disrtibute

ssh-copy-id
E-Mail
Cloud Web-UI

File Names

identity.pub
id_dsa.pub
id_rsa.pub
arbitrary

Config Files on the client: The Private Key

File names (default)

identity
id_dsa
id_rsa

Arbitrary File name

ssh –i /path/to/private/key

NEVER leaves your machine voluntarily

Private and Secret

Encrypt

- Optional with password

SSH Key Variants

openssh

IETF Standard
Public & Private Keys encrypted
RFC 4716

Putty

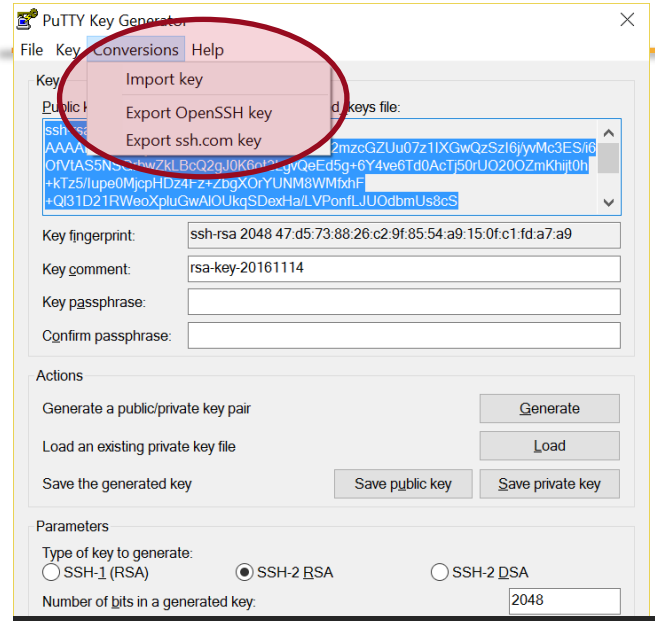
Proprietary format
Public Key unencrypted
Multiple Lines per Key

ssh.com

Commercial variant
Relativly rare

Convert

using tool or text editor




```
PuTTY Format:

---- BEGIN SSH2 PUBLIC KEY ----
Comment: "rsa-key-20161114"
AAAAB3NzaC1yc2EAAAABJQAAAQEAgOH2mzcGZUu07z1IXGwQzSzI6j/yvMc3ES/i
6OfVtAS5NSGrbwZkLBcQ2gJ0K6ot2LgvQeEd5g+6Y4ve6Td0AcTj50rUO20OZmKh
ijt0h+kTz5/Iupe0MjcpHDz4Fz+ZbgXOrYUNM8WMfxhF+Ql31D21RWeoXpluGwAl
OUkqSDexHa/LVPonfLJUOdbmUs8cS+e3Cdd6MDRJeXAoArxFZAfveRXK9RT0DBOc
gcIEv3/s9Vtyp5bg/NHsV40am/jUemDlSQIL/1ZBI4dKmGt4EV43tqSaZg3ylbEn
knAgDpCgBCs0VHpbRLbLJCR/9umS1TNklalkuqiYI2xEmJqOXQ==
---- END SSH2 PUBLIC KEY ----

######################################################################

openSSH-Format:

ssh-rsa AAAAB3NzaC1yc2EAAAABJQAAAQEAgOH2mzcGZUu07z1IXGwQzSzI6j/yvMc3ES/
i6OfVtAS5NSGrbwZkLBcQ2gJ0K6ot2LgvQeEd5g+6Y4ve6Td0AcTj50rUO20OZmKhijt0h+kTz5/
Iupe0MjcpHDz4Fz+ZbgXOrYUNM8WMfxhF+Ql31D21RWeoXpluGwAlOUkqSDexHa/
LVPonfLJUOdbmUs8cS+e3Cdd6MDRJeXAoArxFZAfveRXK9RT0DBOcgcIEv3/s9Vtyp5bg/NHsV40am/
jUemDlSQIL/1ZBI4dKmGt4EV43tqSaZg3ylbEnknAgDpCgBCs0VHpbRLbLJCR/9umS1TNklalkuqiYI
2xEmJqOXQ== rsa-key-20161114
```

Config files on the server: authorized_keys

authorized_keys (Server)

Grants login access	To users on the server
	List of public keys
Options (excerpt)	command
	Environment
	From
	no-pty
	restrict (denies everything)
	grant arbitrary options

```
# Comments allowed at start of line
ssh-rsa AAAAB3Nza...LiPk== user@example.net

from="*.sales.example.net,!pc.sales.example.net" ssh-rsa
AAAAB2...19Q== john@example.net

command="dump /home",no-pty,no-port-forwarding ssh-dss
AAAAC3...51R== example.net

permitopen="192.0.2.1:80",permitopen="192.0.2.2:25" ssh-dss
AAAAB5...21S==

tunnel="0",command="sh /etc/netstart tun0" ssh-rsa AAAA...==
jane@example.net

restrict,command="uptime" ssh-rsa AAAA1C8...32Tv==
user@example.net

restrict,pty,command="nethack" ssh-rsa AAAA1f8...IrrC5==
user@example.net
```

Config Files on the client: config

Parameter Hierarchy

- Command-line → user config → global config
  - Attention: The first obtained value per parameter will be used
- User config: ~/.ssh/config
- Global config: /etc/ssh/ssh_config

Contains Stanzas (parameter sets)

- Stanza starts with Keyword Host or match host followed by Nicknames
- multiple Nicknames per Stanza, can contain wildcards
- stanzas can be nested

Parameter Exampes

- Hostname
- User
- IdentityFile
- LocalForward
- Remote Forward
- ProxyJump
- ProxyCommand
- LogLevel

```
# every stanza (section) starts with the Host Keypword
Host ssh.git.tech.rz.acme.de gitlab.acme gitlab
    Hostname ssh.git.tech.rz.acme.de
    User git
# Multiple Hostnames per Stanza possible, wildcards are fine
Host tu*
    User stred_tu

Host pu*
    User stred_pu

# When working from Home, use a jump host
Match host tu* pu* !exec "ifconfig en0 | grep 192.168"
    ProxyJump r1-102345@jump.server.acme.com

Host pu-internet
    HostName pu.internet.comp.acme.de
# Different hostnames for intranet
Host pu*
    HostName pu-dbtier.inet2.comp.acme.de

Host tu* au*
    HostName au-dbtier.inet2-test.comp.acme.de

Host spatial-bug-testcase bug
    User robertmarz2
    LocalForward 1913 127.0.0.1:6780
    LocalForward 1916 127.0.0.1:6778
    HostName testcase-sandbox.inet2-test.comp.acme.de
Host *
    User stred_au
    ServerAliveInterval 120
    ServerAliveCountMax 3
    IdentityFile ~/.ssh/RobertMarzPAN
```

Config Files on the client: config (example)

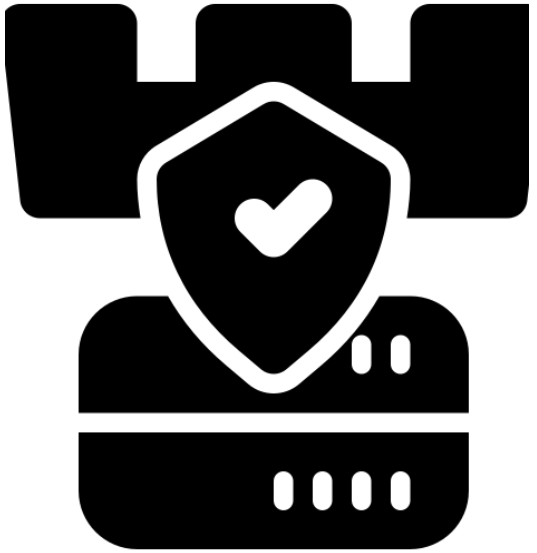
```
# every stanza (section) starts with
# the keyword host or matchhost
Host ssh.git.tech.rz.acme.de gitlab.acme gitlab
    Hostname ssh.git.tech.rz.acme.de
    User git
# Multiple Hostnames per Stanza possible,
# wildcards are fine
Host tu*
    User stred_tu

Host pu*
    User stred_pu

# When working from Home, use a jump host
Match host tu* pu* !exec "ifconfig en0 | grep
192.168"
    ProxyJump r1-102345@jump.server.acme.com

Host pu-internet
    HostName pu.internet.comp.acme.de
```

```
# Different hostnames for intranet
Host pu*
    HostName pu-dbtier.inet2.comp.acme.de

Host tu* au*
    HostName au-dbtier.inet2-test.comp.acme.de

Host spatial-bug-testcase bug
    User robertmarz2
    LocalForward 1913 127.0.0.1:6780
    LocalForward 1916 127.0.0.1:6778
    HostName testcase-sandbox.inet2-
test.comp.acme.de
Host *
    User stred_au
    ServerAliveInterval 120
    ServerAliveCountMax 3
    IdentityFile ~/.ssh/RobertMarzPAN
```

ssh: selected features

Jump Hosts

Jump Hosts / Bastion Hosts	When direct connection impossible
	ssh user@Bastion; then ssh user2@target
ProxyJump	new (openSSH 7.5) and easy
	Portforwarding must be supported by Jump-Hosts
	`ssh -J <jump server> <remote server>`
ProxyCommand	more flexible aka complicated
	allows execution of any command, e.g. unencrypted forward by nc (netcat)
	`ssh -o ProxyCommand="ssh -W %h:%p <jump server>" <remote server>`



Network Tunnel – Port forwards

Mini VPN

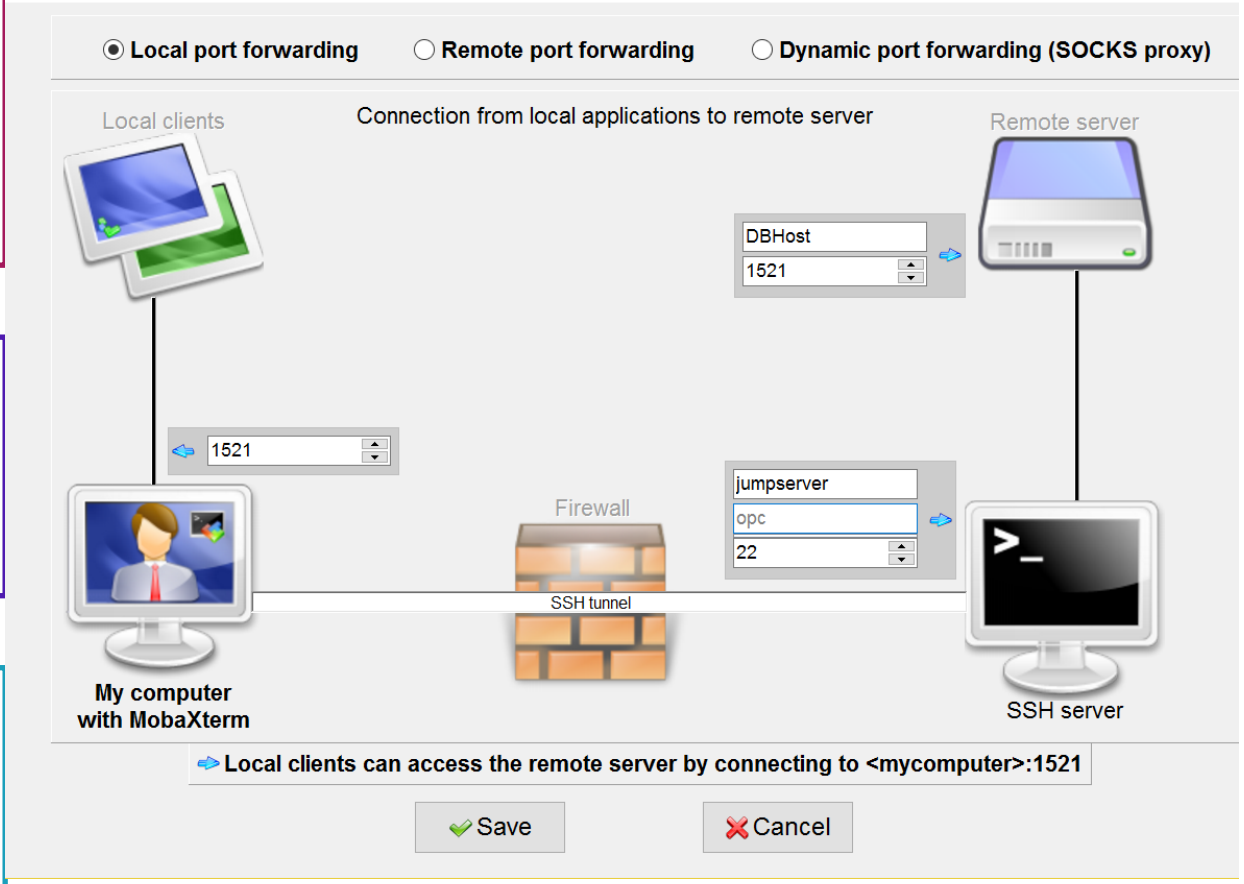
- Parameter
  - -L lport:target-machine:dport
  - -L 1521:127.0.0.1:1521
  - Tunnel chaining possible

Tools

- SQL Developer
- SQLcl
- MobaXterm

Specials:

- X11 Forwarding
- SOCKS5-Tunnel
  - https-Proxy
  - Parameter **DynamicForward or** -D



Secure File Transfer

scp

- Secure Copy

sftp

- Secure FTP

rsync -e ssh

- Remote Sync

tar

- ssh user@remote.host "tar czpf - /some/important/data" \
  | tar xzpf - -C /new/root/directory
- tar cpf - /some/important/data | ssh user@destination-machine \
  "tar xpf - -C /some/directory/"

Remote File Systems: sshfs

Remote File System using SFTP

Local mount

Persistant or per session

Integrated in Client Tools e.g.

- Midnight Commander
- MobaXTerm

The ssh agent

Stores Keys

- In memory (only RAM)
- One-shot password entry for Key

Takes over key-operations

- for clients on same machine & user
- Keys won‘t be transfered

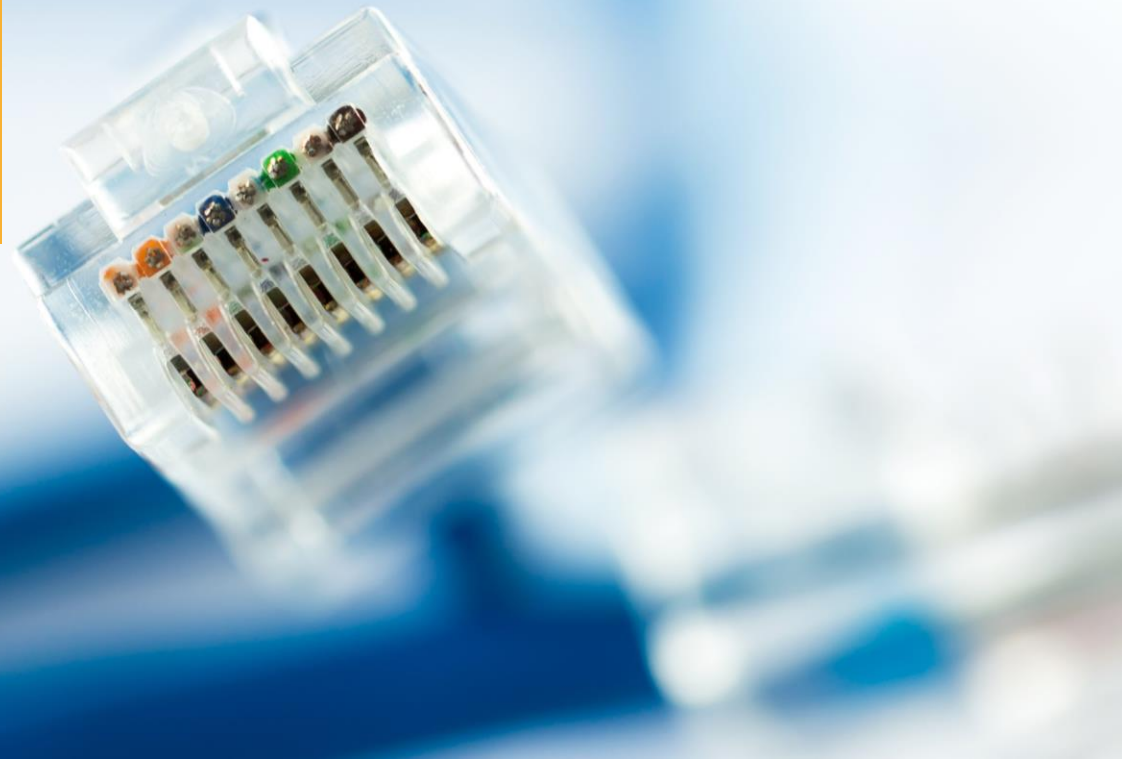
Linux:

- ssh-agent # starts agent
  - ssh-add # adds key

Windows:

- PuTTY-Agent (pagent.exe)

Conclusion



ssh Demo: Connecting to the Oracle Cloud OCI

SSH Implementierungen

Unix / Linux

- Native

OSX (Mac)

- Native

Windows

- PuTTY
- MobaXterm
- mRemoteNG
- Filezilla
- PowerShell/Win32-OpenSSH von MS
  - Client und Server

The Secure Shell (ssh)

Prefer

- Keys over passwords

The client config – file

- makes your Life a lot easier

ssh is a Tool Suite

- There is more than just „ssh“