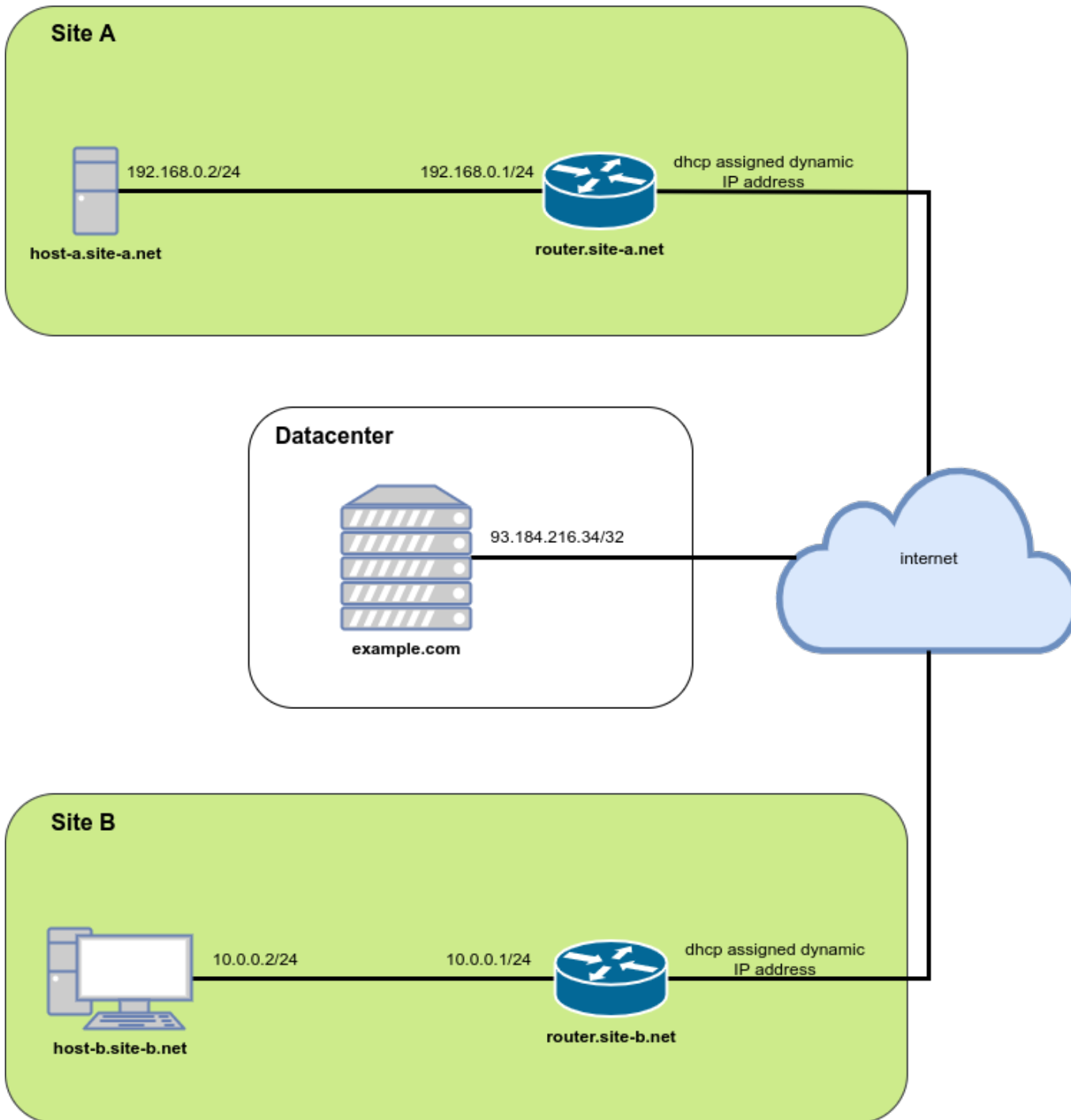



[remotessh](#), [ssh](#), [autossh](#), [tunnel](#)

Remote SSH Tunnel with autossh

If you need to access a remote system behind a firewall and an unknown dynamic IP address, this might be useful. This configuration enables **host-b.site-b.net** to connect to **host-a.site-a.net** via *jump host* **example.com**. *autossh* is used to establish a *remote ssh tunnel* from **host-a.site-a.net** to **example.com**. Through this *remote ssh tunnel* **host-b.site-b.net** will be able to connect to **host-a.site-a.net**.



 In this example I am using:

- `example.com` as the *hostname* of the *jump host*.
- `host-a.site-a.net` as the *hostname* of the system from which the remote ssh tunnel will be originating



- remotesitea as the *username* on **example.com**.

Please replace all used hostnames and usernames according to your environment.

on example.com

Create a user for the *remote tunnel* at the *jump host*:

```
useradd -m -s /bin/bash remotesitea
passwd remotesitea
```

host-a.site-a.net

Install *autossh* and generate a *SSH key* **without a pass phrase**.

```
apt update && apt -y upgrade && apt -y full-upgrade && apt -y autoremove
apt -y install autossh
ssh-keygen -o -a 100 -t ed25519 -f ~/.ssh/autossh-key -C "autossh@host-a.site-a.net"
```

Copy the *autossh* pub key to the jump host.

```
ssh-copy-id -i /root/.ssh/autossh-key remotesitea@example.com
```

Create a unit file.

```
vim /etc/systemd/system/sshtunnel.service
```

```
[Unit]
Description=Remote SSH tunnel to 'example.com'
After=network-online.target ssh.service

[Service]
User=root
Environment="AUTOSSH_PORT=0"
Environment="AUTOSSH_GATETIME=0"
RestartSec=30
Restart=always

ExecStart=/usr/bin/autossh -NT -o "ExitOnForwardFailure=yes" -R 16000:127.0.0.1:22 -p 22 -l remotesitea
example.com -i /root/.ssh/autossh-key
ExecStop=/usr/bin/killall -s KILL autossh
TimeoutStopSec=10

[Install]
WantedBy=multi-user.target
```

Enable and start the *autossh* remote tunnel as a service

```
systemctl enable sshtunnel.service
systemctl start sshtunnel.service
```

host-b.site-b.net

Now you can connect from **host-b.site-b.net** to **host-a.site-a.net** via the jump host:

```
# ssh to example.com and log in
ssh <youruser>@example.com
# when you are logged in on example.com, ssh through the remote tunnel to host-a.site-a.net
ssh -p 16000 <A-User-Account-On-host-a.site-a.net>@localhost
```

From:

<https://wiki.nanoscopic.de/> - **nanoscopic wiki**

Permanent link:

<https://wiki.nanoscopic.de/doku.php/pages/howtos/ssh/remote-ssh-tunnel-with-autossh?rev=1612986744>

Last update: **2021/02/10 19:52**

