

Connecting to the clusters

- [Prerequisites to connect to SCITAS clusters:](#)
- [Step-by-step guide](#)
- [Errors](#)
 - **WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!**
- [Related articles](#)
- [Note](#)

Prerequisites to connect to SCITAS clusters:

- A valid EPFL [GASPAR account](#)
- An approved [SCITAS account](#)
- The client computer needs to be inside the [EPFL network](#), or the user should establish a [VPN](#) connection



Our clusters

The clusters are accessible via SSH at the following addresses:

```
fidis.epfl.ch
helvetios.epfl.ch
izar.epfl.ch
```

Step-by-step guide

1. Log to a cluster using your GASPAR username and password as follows:

```
ssh -X <myusername>@<mypreferredcluster>.epfl.ch
```



SSH clients

SSH clients are available by default on MacOS X and Linux.

For windows, we recommend [Git Bash](#), [PuTTY](#) or [MobaXterm](#).

2. Once logged in, you have access to your home directory at `/home/<username>`.

Your home directory is the same on all SCITAS clusters, and is limited by a quota of 100GB per user. Home files are backed up every night. This means that important files like source code, consolidated simulation results, etc should go here.

3. On each cluster, you have also access to a private scratch directory at `/scratch/<username>`.

Scratch file systems are local to each cluster and are fast. This is typically where raw results, checkpoints, large temporary datasets will go. Scratch files are not backed up and files older than two weeks are subject to automatic deletion by a daily cleanup procedure.



Please note that from outside the EPFL network you will need to connect to the VPN

Errors

WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!

A SSH server is identified by a key which is used to derive the security of the connection. A fingerprint of the key is kept on the SSH client at the first login and compared at every consequent logins. If the key changes after some time, the SSH client refuses to connect and you see this error. It's required from you that you check that the server you're connecting to is legit and you're not being hacked. Usually it means contacting the server administrator to ask for the new key fingerprint. You can then proceed by removing the old offending key from your `~/.ssh/known_hosts` file and connect again. You'll be asked to accept the new key fingerprint.

Check the current fingerprint, against the list you can find bellow:

```
ssh-keygen -F fidis.epfl.ch -l
```

If the fingerprint doesn't match, remove the offending key (in this example for fidis.epfl.ch):

```
ssh-keygen -R fidis.epfl.ch
```

Here's the list of fingerprints for our cluster frontends:

deneb1.epfl.ch

```
MD5:95:e9:2b:d2:d7:9a:fd:d4:b8:b4:08:7a:54:9a:7c:52 (DSA) SHA256:jYIQmefcjLsVbMp43XQ0lPXqk91vLQT7mr0xDJlju3g (DSA)
MD5:79:15:67:34:a8:1b:a4:32:a9:29:eb:e1:3b:52:39:c8 (ECDSA) SHA256:7xCS2WRw/fItCaOfUEE1GjYpweX4uFDKwGhcfAcexSA (ECDSA)
MD5:d5:01:e1:17:8c:30:6f:45:dd:69:28:34:08:b8:a2:6f (ED25519) SHA256:rI8wRfGfqtnYWmiFWtfuIizqhtfjLQB /b9UVmLD8H6w (ED25519)
MD5:bd:de:e7:97:d3:4d:a7:11:11:2d:7c:b7:5b:59:ec:65 (RSA) SHA256:0k20RduYq+7PZv64ua3G1UxcLpBcDK7S6tsreppSimA (RSA)
```

deneb2.epfl.ch

```
MD5:95:e9:2b:d2:d7:9a:fd:d4:b8:b4:08:7a:54:9a:7c:52 (DSA) SHA256:
jYIQmefcjLsVbMp43XQ0lPXqk91vLQT7mr0xDJlju3g (DSA)
MD5:79:15:67:34:a8:1b:a4:32:a9:29:eb:e1:3b:52:39:c8 (ECDSA) SHA256:7xCS2WRw
/fItCaOfUEE1GjYpweX4uFDKwGhcfAcexSA (ECDSA)
MD5:62:25:c8:a7:02:69:3b:72:77:8b:4d:21:32:9a:ed:04 (ED25519) SHA256:B7ZnGnmNf+mV7i7NiR9KtnjF1Ylnw
/794kwZY2sTfBk (ED25519)
MD5:bd:de:e7:97:d3:4d:a7:11:11:2d:7c:b7:5b:59:ec:65 (RSA) SHA256:
0k20RduYq+7PZv64ua3G1UxcLpBcDK7S6tsreppSimA (RSA)
```

fidis.epfl.ch

```
MD5:a2:e5:62:c6:c6:4c:ff:75:71:b1:a9:c9:0a:96:e4:48 (DSA) SHA256:
WcLqkZl7xsvE5FSU5Lt0RpUh4t0SVIHZirFRSg+ZH7c (DSA)
MD5:cb:cb:2e:82:b3:70:c0:94:3f:9f:f5:95:21:ba:07:d5 (ECDSA) SHA256:
fYSeHbrIju5yfoPCN3qMwkWxwr8iq8h6GbbLIbBb7kc (ECDSA)
MD5:2a:41:ce:c9:97:a9:c5:5d:db:7d:c7:a1:ee:37:5b:49 (ED25519) SHA256:4x3/0GOMYVb3iG
/o6u7ZeI0KSDVmg9mj8VNEYSqBo78 (ED25519)
MD5:4f:37:ec:03:1e:05:6c:db:26:07:fa:68:b1:a6:bc:51 (RSA) SHA256:vND0UKeNW
/3wTA2FE70gGT4sdF1+2uG0gQ6FBfyVz08 (RSA)
```

fdata1.epfl.ch

```
MD5:b7:77:41:d9:eb:f6:5d:ad:eb:ee:47:cf:96:26:be:2d (ECDSA) SHA256:
P9C8BlrIlhhSapWlj8toQhrOef6tYT2Ec7WctgdYLLY (ECDSA)
MD5:03:04:17:4e:9f:23:0f:ae:4c:42:3e:12:e4:70:c3:31 (ED25519) SHA256:
aECkiZoSfo+FapnmKGXb4nclyDnMmpwBGMyc8D0ld0Y (ED25519)
MD5:d7:fb:55:82:58:35:1b:cc:dd:4b:2b:18:c3:8d:d4:83 (RSA) SHA256:T4whsuNmxE/uz2qTx16SNOP040Q/Nevm/03oN1xh
/QM (RSA)
```

helvetios.epfl.ch

```
MD5:89:7c:b9:5a:8e:e9:9d:fb:a3:18:1f:dc:6d:23:42:b0 (DSA) SHA256:  
w+nE4SpZNaK6l9FR7ztBnP9PbMSqgANDKO3WHIfLlVY (DSA)  
MD5:4a:e7:5c:7d:da:80:e8:e6:e3:ba:c8:58:4d:c3:79:07 (ECDSA) SHA256:SAw3eUuhAAqDGB098/5Q9fS17+4Ck8u  
/te8GWjjOtrM (ECDSA)  
MD5:00:7a:58:10:8e:72:65:27:32:b3:6b:10:ec:02:76:fd (ED25519) SHA256:  
bjJ7bylq58jnlE+KCov89tLVFFxoTwBatw3gnJsWMB4 (ED25519)  
MD5:d9:f0:9d:0e:55:52:1a:da:42:8f:36:86:95:3f:68:c9 (RSA) SHA256:  
9ZUcgDL54YHLwNIgXPf2rPQMm3DZtgYxDy4oEkJOBGc (RSA)
```

izar.epfl.ch

```
MD5:ec:fc:bb:82:4a:9a:ba:a1:fd:02:45:cb:64:44:3e:c7 (DSA) SHA256:  
MHpCKnb34lxKdzGDJ4kb896fIczbs5c38WxsPF78Aw0 (DSA)  
MD5:2d:3a:92:38:47:15:61:ba:c1:62:57:cc:02:f6:4e:eb (ECDSA) SHA256:uNGVKhqr2YPYNG5YT0DatwyXHKsOqTE5fwux  
/TjZXXc (ECDSA)  
MD5:f8:54:0b:b9:26:11:17:98:1c:97:bd:a5:67:22:49:3a (ED25519) SHA256:  
xZ+QSy3IizOrZ0h9lZcsBKQPP5a9WkjE5z+kagilAw4 (ED25519)  
MD5:f1:55:22:c6:f0:5a:5f:f5:b5:9a:1c:d7:9a:ab:aa:1f (RSA) SHA256:  
j6Ql40elkHG3Sq6pXGAIWLAKySfaakdonXloRKC3HRc (RSA)  
MD5:02:64:72:07:42:09:13:7c:7f:bf:19:4c:2d:a2:a1:5f (RSA) SHA256:ylv3VU  
/KiCwWJ4FOBaSGpP3RnmYe3KN6PMSloQV4fzg (RSA)
```

Related articles

- [Using the clusters](#)
- [How to configure PuTTY SSH to transfer files to Cluster from Windows](#)
- [Connecting to the clusters](#)
- [FAQ](#)
- [First steps on the clusters](#)

Note

If you are already connected to a machine, you can check the host keys using this (you need to be root):

```
for i in $(ls /etc/ssh/ssh_host*_key); do  
    ssh-keygen -l -E MD5 -f $i | tr '\n' ' '  
    ssh-keygen -l -E SHA256 -f $i  
done | awk '{print $2,$4,$6,$8}' | column -t
```