Restoring after Windows has been reinstalled

Русский · User guide

The program and the chosen DATA must survive outside the disk being formatted. Every machine uses its own DATA. The private SSHD keys in this folder identify the server; the client keys give the right to sign in to it.

Before formatting C:

On every machine update the program, choose its DATA and run **SETUP → SSHD** once, confirming UAC. This step saves the live pairs `ProgramData\ssh\ssh_host_*` into `DATA\Tunnel\SSHD` together with a manifest. It also checks the pairs, tightens the ACL and sets up the incoming CLIENTS keys. Make sure the step finished successfully, then save the whole of DATA.

Having only CLIENTS, Export Keys or `known_hosts` does not replace this step: a former private host key cannot be restored from a fingerprint or the public part. An incomplete `SSHD.pending-*` does not count as a saved identity.

After the reinstall

1. Bring the program and DATA back, choose **DATA FOLDER**.
2. Choose **SETUP → RESTORE → Restore this machine**, confirm UAC. If the updated aliases have to be carried over from DATA, choose the option **Restore and replace SSH config**. It keeps one former copy, `~/.ssh/config.before-restore`; the ordinary option keeps the existing config.
3. Wait for the SSH/SFTP table. Restart the window so that it reads the profile again.

RESTORE installs a missing cloudflared when such routes exist, imports the keys, repairs lost paths and starts SSHD. The saved host keys come back **before the component is installed and the service first starts**. If the live key and the saved one differ, the step stops: both versions stay. Without a saved

identity the first install creates a new one; the old trust in the former Windows will not start working by itself.

Console SSH and ProxyJump get a `UserKnownHostsFile` with the file `DATA/config/rclone/known_hosts` and the ordinary files of the profile. Someone else's explicit `UserKnownHostsFile` settings for a particular Host keep priority. SSH-KEYSCAN also refreshes this binding after its work. RESTORE does not scan and does not accept unknown or changed keys automatically.

Only an explicitly enabled (`Persistent=true`, `AutoStart=true`) SSH TUN of this machine is restored. The name is taken from `Tunnel/beacon-startup.json` (the explicit choice made when the channel was enabled), and for former data sets from the SSHD identity, so renaming Windows does not pick a random channel out of the neighbours' backup copies. The Startup shortcut comes back and the existing channel is started; the key is not registered on the VPS again. Disabled channels and other machines' channels stay disabled.

The rclone paths to the keys and to `known_hosts` are repaired for the chosen DATA and the new profile. The set of remotes and the cloud credentials are not created anew. If rclone.conf is missing, it has to be built with the RCLONE.CONF button; the empty sample in DATA is not a backup copy of a working machine.

The check reads the metadata of the SFTP root through rclone and runs `echo` over SSH for the configured machines/VPS. It uses strict host-key checking, is limited in time, prints no directory contents and opens no RDP/terminals. It is a check of the saved SSH routes; it does not claim that every alternative route of AUTO, every disk and every cloud is reachable. Routers with password sign-in are not checked.

The prerequisites remain: the former addresses/accounts in targets.json must be current; the matching Windows account must exist and have the rights it needs. SSHD sets up the keys of administrators. The WG routes of the routers, access to the VPS and the OpenVPN driver are not created by RESTORE. For WORK, first install the driver and connect the chosen OpenVPN, if access goes through it.

Clean from DATA (MIRROR)

SETUP → RESTORE → "Clean from DATA (MIRROR)" is the main option after a reinstall. The ordinary RESTORE adds what is missing; MIRROR makes C: exactly what DATA describes, like robocopy /MIR:

- `~/.ssh` is built anew as a whole: the keys from Export/SERVERS, the config from DATA, the trust from the `known_hosts` of DATA. The old keys, config and `known_hosts` of the profile go.

- `administrators_authorized_keys` — exactly the CLIENTS keys; keys brought by SSH-COPY-ID past CLIENTS are taken off. An empty CLIENTS does not touch the file.
- The SSHD host keys — from `Tunnel\SSHD`, even if the machine already has others.
- Removed are `rclone.conf` outside DATA (`%APPDATA%\rclone`, `~\.config\rclone`), the shortcuts of REVERSE SSH TUN (an enabled channel comes back), the `Audion Backup Sync` task.
- No rclone or cloudflared in the program — they are downloaded.

Nothing is deleted silently: everything removed lands in `DATA\Recovery\mirror-<time>` with rights for the owner, SYSTEM and administrators only; `restore.log` and `sshd.log` are there too. If the import of the keys failed, the former `~/.ssh` is put back in place.

The steps that need a person wait with no timeouts: a declined UAC is asked again; for every route through cloudflared the sign-in to Cloudflare Access is checked and a browser opens if there is none; for routers with an address that answer from here and whose password is not in DATA, the password is asked — it is saved only after a real sign-in, an empty Enter skips. Whether all routers share one password is asked once and remembered in `targets.json` (`"_routers": { "sharedPassword": true }`). Only on "yes" is the checked password tried on the other routers and saved for the routers of other sites that cannot be reached from here; on "no" or without an answer every router has its own password. MIRROR does not touch the Access tokens in `~/.cloudflared`: that is a sign-in, not data. MIRROR does not run SSH-KEYSCAN — the trust is taken from DATA only.

CLI

```
.\Tunnel\RemoteDesktop\restore-machine.cmd -Root "D:\Data\This machine" -NoPause
.\Tunnel\RemoteDesktop\restore-machine.cmd -Root "D:\Data\This machine" -Check -
NoPause
.\Tunnel\RemoteDesktop\restore-machine.cmd -Root "D:\Data\This machine" -
RestoreConfig -NoPause
.\Tunnel\RemoteDesktop\restore-machine.cmd -Root "D:\Data\This machine" -Mirror
.\Tunnel\RemoteDesktop\restore-machine.cmd -Root "D:\Data\This machine" -Mirror -
Check -NoPause
```

`-Mirror -Check` shows what will leave C: and changes nothing. `-Mirror -NoPrompt` asks about nothing: UAC is requested once, the sign-ins are only listed.

`-Check` reads the data set, does not change the services/profile and does not touch the network. `-SkipProbe` does the local restore without the network check. Code 0 means the chosen mode finished successfully; 1 — an error of a local step; 2 — the local steps are done, but the connection

check did not pass in full. To replace the routes separately: `Runs\import-keys.cmd -Root ... -RestoreConfig`. `-Force` still refers to differing key files, not to the SSH config.

Checking the implementation

`Tests/Test-CleanRestore.ps1` uses a real synthetic ED25519 key, empty profiles/ProgramData/Startup in a temporary folder and stubs for the service/network. It checks that the identity is kept, the ACL of the backup copy, the refusal on damage and on a conflict, the import and the move of paths, the SSH/ProxyJump trust, Startup and the CLI order. `Test-SshServerInstall.ps1` separately checks the order before the service is installed/started and the readiness errors. These are isolated checks, not a trial on a physically reinstalled Windows.