

Server - Installation mit Distributionspaketen (*:deb, *.rpm)

Mit Hilfe fertiger Installationspakete ersparen wir dem Anwender das Installieren von Kompilierungswerkzeugen und Bibliotheken und reduzieren dadurch erheblich den Ressourcen- und Zeitverbrauch. Alternativ zu den Installationspaketen gibt es auch Beschreibungen für das [Selbstkompilieren](#) sowie das [Deployment mit Containern](#).

Im Paket `fabaccess-bffh` enthalten sind der BFFH Server, sowie das [FabFire Provisioning Tool](#). Die zwei Binaries, die installiert werden, sind damit:

- `/usr/bin/bffhd`
- `/usr/bin/fabfire_provision`

Im optionalen Paket `fabaccess-bffh-dbgsym` finden sich außerdem Debug Symbols für das [Fehlerrückmeldung](#), z.B. durch `gdb`, `valgrind` oder `strace`. Beim Installieren werden hierbei kryptische binäre Dateien in `/usr/lib/debug/.build-id/` abgelegt.

Anforderungen und Kompatibilität

Mindestanforderungen und kompatible Hardware finden sich [hier](#).

Downloads

Fertige Pakete für Debian / Ubuntu / Kubuntu (*.deb) und Fedora / CentOS / RHEL / Rocky Linux (*.rpm) finden sich in unter folgenden Quellen.

Direkter Download

<https://downloads.fab-access.org/?dir=bffh-packages> - auch direkt über CLI auflist- und abrufbar:

```
cd /tmp/ && wget --spider -r -np https://downloads.fab-access.org/bffh-packages/development/2>&1 | grep "\--" | grep fabaccess-bffh | awk -F ' ' '{print $3}' && cd -
```

Wir suchen das passende Paket aus (kann mit dem Kommando `arch` oder `uname -a` überprüft werden) und laden es ins Verzeichnis `/tmp/` herunter (im Beispiel `_amd64.deb`):

```
wget https://downloads.fab-access.org/bffh-packages/development/fabaccess-bffh_0.4.4_amd64.deb
-P /tmp/
```

Und optional:

```
wget https://downloads.fab-access.org/bffh-packages/development/fabaccess-bffh-
dbgsym_0.4.4_amd64.deb -P /tmp/
```

GitLab Releases

<https://gitlab.com/fabinfra/fabaccess/bffh/-/releases> (**Achtung:** es gibt hier **noch** keine fertigen Pakete. Dies ändert sich mit den nächsten Versionen!)

Signaturen prüfen

Der öffentliche Teil des **PGP-Schlüssels**, der für die Paketerzeugung verwendet wird, findet sich hier: <https://gitlab.com/fabinfra/fabaccess/bffh/-/blob/feature/deb+rpm/debian/public.pgp> bzw. unter

<https://keys.openpgp.org/search?q=D793B5E8A754285F9F776D22C4505B2062532D6E>

Dieser kann verwendet werden, um die Signaturen vor dem Installieren zu überprüfen.

```
gpg --keyserver keys.openpgp.org --recv-keys D793B5E8A754285F9F776D22C4505B2062532D6E
```

Debian / Ubuntu / Kubuntu:

Für das Verifizieren mit gpg benötigen wir noch die *.sig oder *.sig.asc Datei (nur für *.deb notwendig):

```
gpg --verify /tmp/fabaccess-bffh_0.4.4_amd64.sig.asc /tmp/fabaccess-bffh_0.4.4_amd64.deb
gpg --verify /tmp/fabaccess-bffh_0.4.4_amd64.sig.asc /tmp/fabaccess-bffh-
dbgsym_0.4.4_amd64.deb #optional
```

Ergebnisse:

```
gpg: Signatur vom Mi 19 Mär 2025 01:33:43 CET
```

```
gpg:           mittels EDDSA-Schlüssel D793B5E8A754285F9F776D22C4505B2062532D6E
```

```
gpg: Korrekte Signatur von "FabInfra <info@fab-access.org>" [ultimativ]
```

```
gpg: Signatur vom Mi 19 Mär 2025 01:33:45 CET
```

```
gpg:           mittels EDDSA-Schlüssel D793B5E8A754285F9F776D22C4505B2062532D6E
```

```
gpg: Korrekte Signatur von "FabInfra <info@fab-access.org>" [ultimativ]
```

Fedora / CentOS / RHEL / Rocky Linux:

```
KEY=$(gpg --export --armor D793B5E8A754285F9F776D22C4505B2062532D6E) && echo $KEY > rpm --import -

rpm --verbose --checksig /tmp/fabaccess-bffh-0.4.4-1.x86_64.rpm
rpm --verbose --checksig /tmp/fabaccess-bffh-dbgsym-0.4.4-1.x86_64.rpm #optional

# Ergebnisse:
/tmp/fabaccess-bffh-0.4.4-1.x86_64.rpm:
    Header SHA256 digest: OK
    Header SHA1 digest: OK
    Payload SHA256 digest: OK
    MD5 digest: OK
/tmp/fabaccess-bffh-dbgsym-0.4.4-1.x86_64.rpm:
    Header SHA256 digest: OK
    Header SHA1 digest: OK
    Payload SHA256 digest: OK
    MD5 digest: OK
```

Paket per Dry-run testen

Debian / Ubuntu / Kubuntu

Wer nicht gleich installieren will, kann vorher testen und Infos vom Paket abrufen:

```
dpkg-deb --info /tmp/fabaccess-bffh_0.4.4_amd64.deb

sudo apt install --simulate /tmp/fabaccess-bffh_0.4.4_amd64.deb
```

Installation

Achtung: BFFH benötigt zwingend einen laufenden Eclipse Mosquitto Server. Dieser kann sich entweder auf dem gleichen Server befinden wie BFFH, oder auf einer anderen Maschine. Deshalb wird `mosquitto` standardmäßig nicht installiert!

Debian / Ubuntu / Kubuntu

Das Paket schlägt beim Installieren die Installation weiterer optional empfohlener (recommended) und abhängiger (dependent) Pakete vor. Diese Infos lassen über `dpkg-deb --info fabaccess-bffh_0.4.4_amd64.deb | grep "Depends:\|Recommends:"` abrufen oder sie werden angezeigt, wenn nicht mit `dpkg`, sondern mit `apt` installiert wird. Ein empfehlenswertes Paket ist z.B. `dhall`.

Variante 1: Installation über apt (bevorzugt):

```
sudo apt install --no-install-recommends /tmp/fabaccess-bffh_0.4.4_amd64.deb
sudo apt install /tmp/fabaccess-bffh-dbgsym_0.4.4_amd64.deb #optional
```

Wer die empfohlenen Zusatzpakete installieren will, bekommt diese über `apt` angezeigt und kann auch wie folgt installieren, indem der Parameter `--no-install-recommends` weggelassen wird:

```
sudo apt install /tmp/fabaccess-bffh_0.4.4_amd64.deb
sudo apt install /tmp/fabaccess-bffh-dbgsym_0.4.4_amd64.deb #optional
```

Variante 2: Installation über `dpkg`:

```
sudo dpkg -i /tmp/fabaccess-bffh-0.4.4_amd64.deb
sudo dpkg -i /tmp/fabaccess-bffh-dbgsym-0.4.4_amd64.deb #optional
```

Fedora / CentOS / RHEL / Rocky Linux

```
sudo rpm -i /tmp/fabaccess-bffh-0.4.4-1.x86_64.rpm
sudo rpm -i /tmp/fabaccess-bffh-dbgsym-0.4.4-1.x86_64.rpm #optional
```

Installation prüfen:

```
sudo dpkg -l | grep fabaccess-bffh
```

BFFH als Service aktivieren und Logs prüfen

```
sudo systemctl enable bffh --now
sudo journalctl -f -u bffh.service
```

BFFH startet standardmäßig auf dem TCP-Port `59661`, wenn nicht geändert. Außerdem wird im Hintergrund auf dem TCP-Port `49289` eine [Debugger Console](#)

gestartet.

BFFH Hilfe anzeigen

Es gibt eine minimale [Manpage](#) für BFFH. Diese kann angezeigt werden:

```
man bffhd
man fabfire_provision
```

BFFH Konfiguration

Nach der Installation muss der Server passend konfiguriert werden. Das ist kein Bestandteil des Installers und ist komplett individuell, je nach Gebrauchszweck. Alle wichtige Infos finden sich im Kapitel [FabAccess Konfiguration](#).

Metainformationen anzeigen

Das ist rein informativ und nicht notwendig!

Ein paar Kommandos für verschiedene Zwecke (z.B. nach Tags suchen, Infos zu Paket anzeigen):

```
sudo apt install debtags

debtags update
debtags show fabaccess-bffh

apt-cache search fabaccess
apt-cache search bffh
apt search fabaccess
apt search bffh
apt search difluoroborane

aptitude search '?tag(network::server)'
aptitude search fabaccess
aptitude search bffh
aptitude search difluoroborane
```

Deinstallation

Die Deinstallation von BFFH ist relativ rückstandsfrei, bis auf Konfigurationsdateien, Datenbanken und Logs. Bitte ggf. die Verzeichnisse manuell danach überprüfen:

- `/var/log/bffh/`
- `/etc/bffh/`
- `/var/lib/bffh/`

Debian / Ubuntu / Kubuntu

```
sudo apt remove fabaccess-bffh
sudo apt remove fabaccess-bffh-dbgsym
```

oder:

```
sudo apt purge fabaccess-bffh
sudo apt purge fabaccess-bffh-dbgsym
```

Fedora / CentOS / RHEL / Rocky Linux

`dnf` hat **keinen** `purge` Befehl (bei apt kann man deinstallieren oder deinstallieren + restlos bereinigen = purge). Bei `dnf` gibt nur remove!

```
sudo dnf remove fabaccess-bffh
```

Benennung von Debian Paketen

Diese Entscheidungen sind grundlegend, deshalb finden sie hier Erwähnung.

`fabaccess-bffh` - BFFH Server

Der Server hat mehrere Namen: FabAccess Server, BFFH, bffh, bffhd, Diflouroborane - welcher ist der beste? Wir entscheiden uns für den Paketname `fabaccess-bffh`, weil

- der Paketname relativ klar ist und erlaubt, später weitere Pakete im Zusammenhang `fabaccess-*` zu publizieren
- mit dem Nachteil, dass es etwas länger ist als einfach `bffh` und der systemd Service `bffh` heißt. Das müssen wir im Setup also klar heraus stellen!

