systemd-socket-proxyd

- Description
- Options
- Exit status
- Examples
  - Simple Example
  - Namespace Example
  - PROXY protocol example with nginx
- See Also

systemd

systemd-socket-proxyd

8

systemd-socket-proxyd

Bidirectionally proxy local sockets to another (possibly remote) socket

systemd-socket-proxyd

OPTIONS

HOST

:

PORT

systemd-socket-proxyd

OPTIONS

UNIX-DOMAIN-SOCKET-PATH

Description

`systemd-socket-proxyd` is a generic socket-activated network socket forwarder proxy daemon for IPv4, IPv6 and UNIX stream sockets. It may be used to bi-directionally forward traffic from a local listening socket to a local or remote destination socket.

One use of this tool is to provide socket activation support for services that do not natively support socket activation. On behalf of the service to activate, the proxy inherits the socket from systemd, accepts each client connection, opens a connection to a configured server for each client, and then bidirectionally forwards data between the two.

This utility's behavior is similar to `socat(1)`. The main differences for `systemd-socket-proxyd` are support for socket activation with `Accept=no` and an event-driven design that scales better with the number of connections.

Note that `systemd-socket-proxyd` will not forward socket side channel information, i.e. will not forward `SCM_RIGHTS`, `SCM_CREDENTIALS`, `SCM_SECURITY`, `SO_PEERCRED`, `SO_PEERPIDFD`, `SO_PEERSEC`, `SO_PEERGROUPS` and similar.

Options

The following options are understood:

`--connections-max=`; `-c`
Sets the maximum number of simultaneous connections, defaults to 256. If the limit of concurrent connections is reached further connections will be refused.

`--exit-idle-time=`
Sets the time before exiting when there are no connections, defaults to `infinity`. Takes a unit-less value in seconds, or a time span value such as `5min 20s`.

`--proxy-protocol=`
Uses the PROXY protocol to communicate with the server. This allows an appropriately configured server to know the real client IP address. Takes the version of the PROXY protocol used, and only supports `v1` for now. Default is not to use a PROXY protocol.

Exit status

On success, 0 is returned, a non-zero failure code otherwise.

Examples

Simple Example

Use two services with a dependency and no namespace isolation.

proxy-to-nginx.socket

```
[Socket]
ListenStream=80

[Install]
WantedBy=sockets.target
```

proxy-to-nginx.service

```
[Unit]
Requires=nginx.service
After=nginx.service
Requires=proxy-to-nginx.socket
After=proxy-to-nginx.socket

[Service]
Type=notify
ExecStart=/usr/lib/systemd/systemd-socket-proxyd /run/nginx/socket
PrivateTmp=yes
PrivateNetwork=yes
```

nginx.conf

```
[…]
server {
    listen       unix:/run/nginx/socket;
    […]
```

Enabling the proxy

```
# systemctl enable --now proxy-to-nginx.socket
$ curl http://localhost:80/
```

If `nginx.service` has `StopWhenUnneeded=` set, then passing `--exit-idle-time=` to `systemd-socket-proxyd` allows both services to stop during idle periods.

Namespace Example

Similar as above, but runs the socket proxy and the main service in the same private namespace, assuming that `nginx.service` has `PrivateTmp=` and `PrivateNetwork=` set, too.

proxy-to-nginx.socket

```
[Socket]
ListenStream=80

[Install]
WantedBy=sockets.target
```

proxy-to-nginx.service

```
[Unit]
Requires=nginx.service
After=nginx.service
Requires=proxy-to-nginx.socket
After=proxy-to-nginx.socket
JoinsNamespaceOf=nginx.service

[Service]
Type=notify
ExecStart=/usr/lib/systemd/systemd-socket-proxyd 127.0.0.1:8080
PrivateTmp=yes
PrivateNetwork=yes
```

nginx.conf

```
[…]
server {
    listen       8080;
    […]
```

Enabling the proxy

```
# systemctl enable --now proxy-to-nginx.socket
$ curl http://localhost:80/
```

PROXY protocol example with nginx

systemd-socket-proxyd and nginx using the PROXY protocol

proxy-to-nginx.socket

```
[Socket]
ListenStream=80

[Install]
WantedBy=sockets.target
```

proxy-to-nginx.service

```
[Unit]
Requires=nginx.service
After=nginx.service
Requires=proxy-to-nginx.socket
After=proxy-to-nginx.socket

[Service]
Type=notify
ExecStart=/usr/lib/systemd/systemd-socket-proxyd --proxy-protocol=v1 /run/nginx/socket
PrivateTmp=yes
PrivateNetwork=yes
```

nginx.conf

```
[…]
server {
    listen         unix:/run/nginx/socket proxy_protocol;
    real_ip_header proxy_protocol;
    […]
```

Enabling the proxy

```
# systemctl enable --now proxy-to-nginx.socket
$ curl http://localhost:80/
```

See Also

`systemd(1)`, `systemd.socket(5)`, `systemd.service(5)`, `systemctl(1)`, `socat(1)`, `nginx(1)`, `curl(1)`, PROXY protocol specification, nginx: Accepting the PROXY Protocol

---

□□□ #1
□ □□□ □□□ 2026-10-31 15:01:53 UTC
□ □□□ □□□ 2026-10-31 15:01:53 UTC