
HevSocks5Tunnel

pipeline **passed**

A tunnel over Socks5 proxy (tun2socks) for Unix.

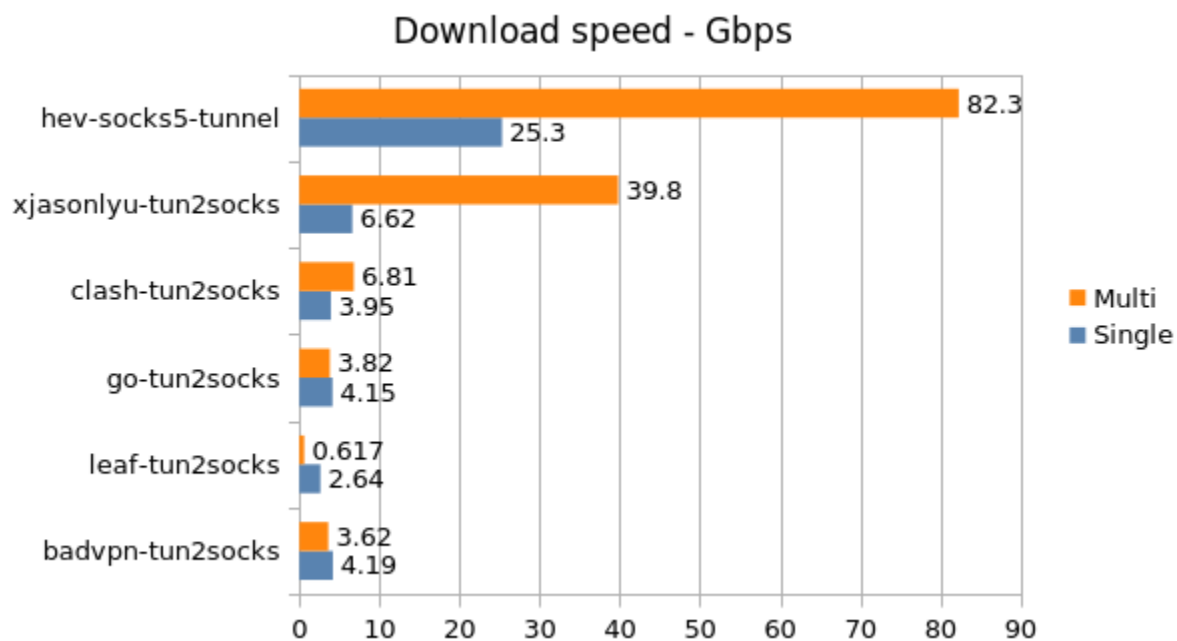
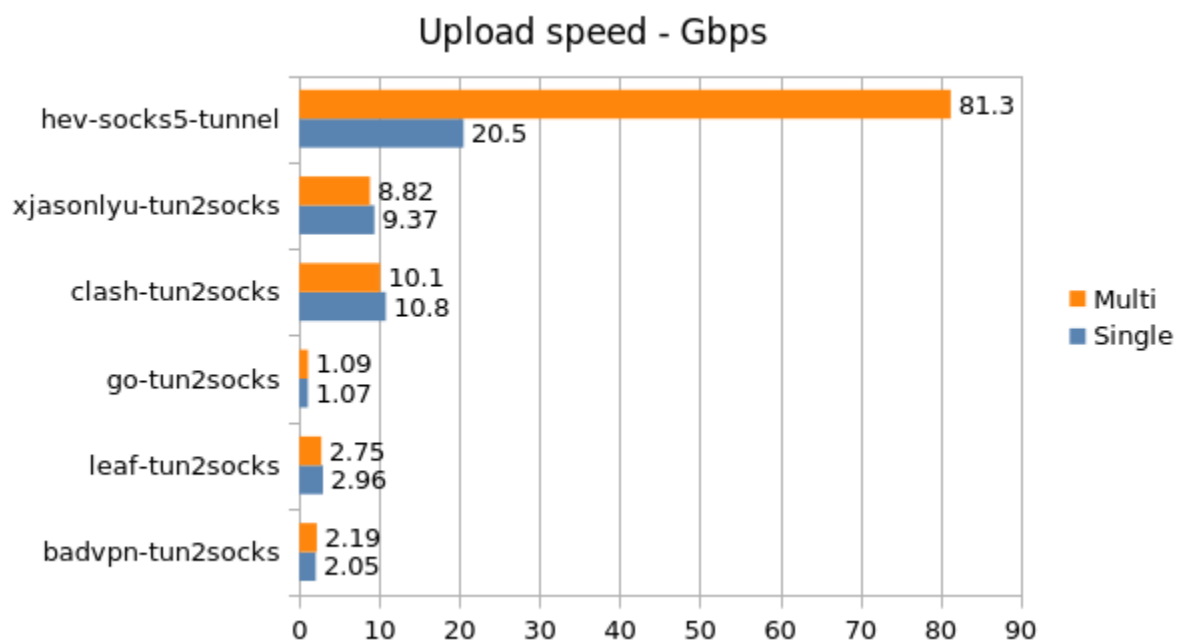
Features

- IPv4/IPv6. (dual stack)
- Redirect TCP connections.
- Redirect UDP packets. (Fullcone NAT, UDP in UDP/TCP)
- Linux/Android/FreeBSD/macOS/iOS/WSL2.

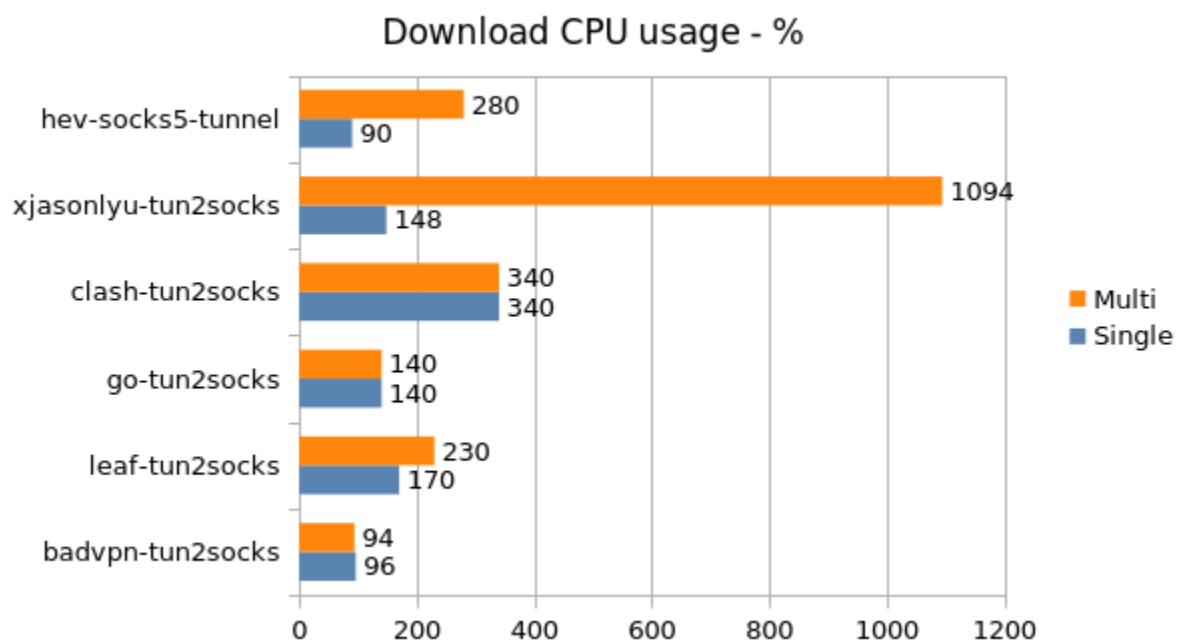
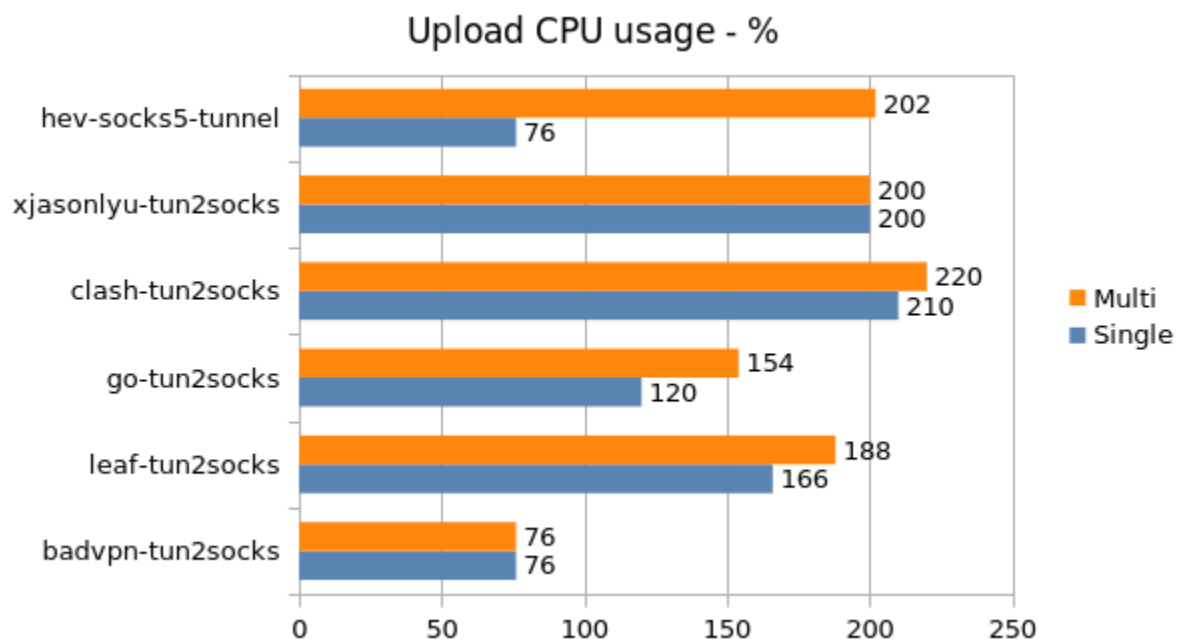
Benchmarks

See [here](#) for more details.

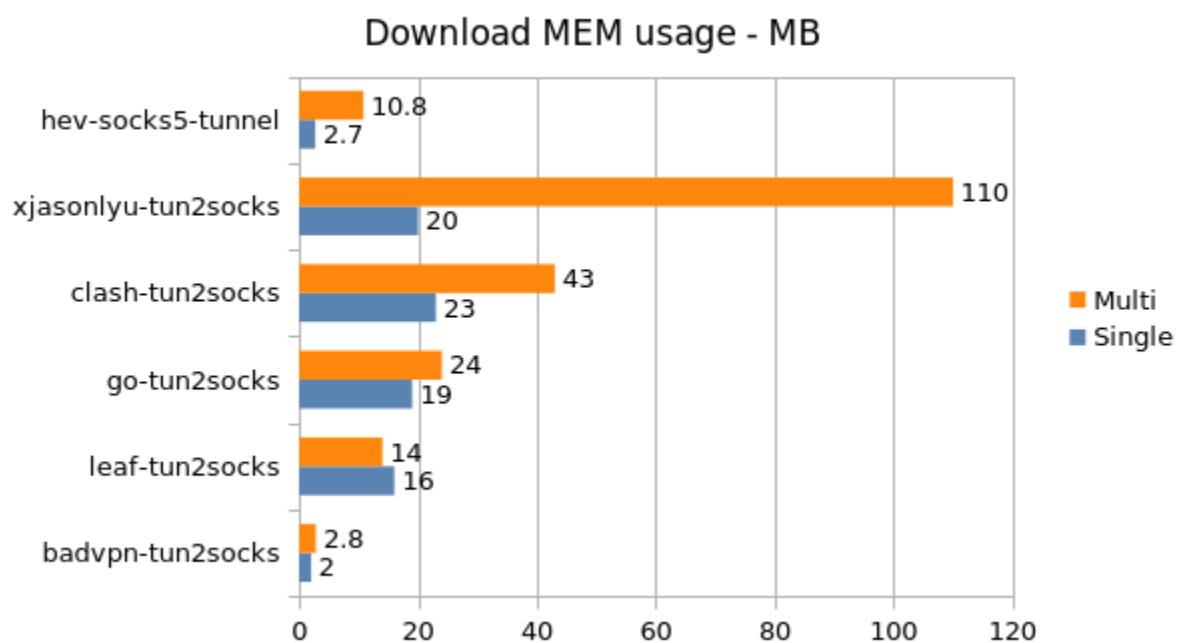
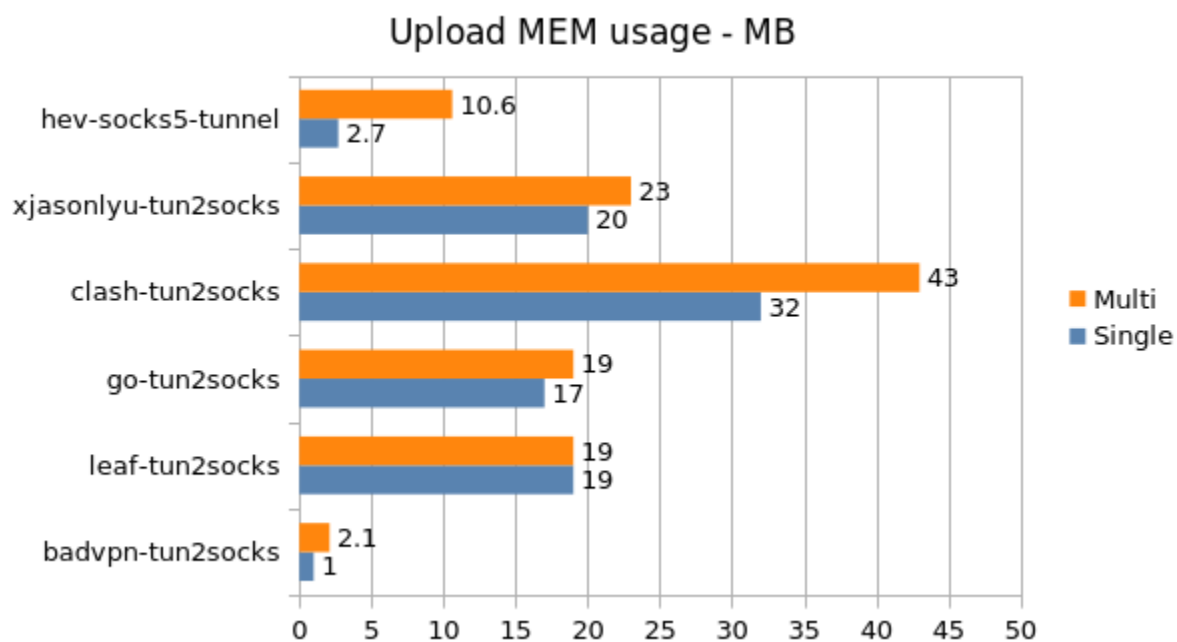
Speed



CPU usage



Memory usage



How to Build

Unix

```
1 git clone --recursive https://github.com/heihher/hev-socks5-tunnel
2 cd hev-socks5-tunnel
3 make
```

Android

```
1 mkdir hev-socks5-tunnel
2 cd hev-socks5-tunnel
3 git clone --recursive https://github.com/heihher/hev-socks5-tunnel jni
4 ndk-build
```

iOS and MacOS

```
1 git clone --recursive https://github.com/heihher/hev-socks5-tunnel
2 cd hev-socks5-tunnel
3 # will generate HevSocks5Tunnel.xcframework
4 ./build.sh
```

Library

```
1 git clone --recursive https://github.com/heihher/hev-socks5-tunnel
2 cd hev-socks5-tunnel
3
4 # Static library
5 make static
6
7 # Shared library
8 make shared
9
10 # Static library for iOS
11 make PP="xcrun --sdk iphoneos --toolchain iphoneos clang" \
12      CC="xcrun --sdk iphoneos --toolchain iphoneos clang" \
13      CFLAGS="-arch arm64 -mios-version-min=12.0" \
14      LFLAGS="-arch arm64 -mios-version-min=12.0 -Wl,-Bsymbolic-
15             functions" static
16
16 libtool -static -o libhev-socks5-tunnel.a \
17      bin/libhev-socks5-tunnel.a \
18      third-part/lwip/bin/liblwip.a \
19      third-part/yaml/bin/libyaml.a \
20      third-part/hev-task-system/bin/libhev-task-system.a
```

How to Use

Config

```
1 tunnel:
2   # Interface name
3   name: tun0
4   # Interface MTU
5   mtu: 8500
6   # Multi-queue
7   multi-queue: false
8   # IPv4 address
9   ipv4: 198.18.0.1
10  # IPv6 address
11  ipv6: 'fc00::1'
12  # Post up script
13  # post-up-script: up.sh
14  # Pre down script
15  # pre-down-script: down.sh
16
17 socks5:
18  # Socks5 server port
19  port: 1080
20  # Socks5 server address (ipv4/ipv6)
21  address: 127.0.0.1
22  # Socks5 UDP relay mode (tcp|udp)
23  udp: 'udp'
24  # Socks5 handshake using pipeline mode
25  # pipeline: false
26  # Socks5 server username
27  # username: 'username'
28  # Socks5 server password
29  # password: 'password'
30  # Socket mark
31  # mark: 0
32
33 #misc:
34  # task stack size (bytes)
35  # task-stack-size: 20480
36  # connect timeout (ms)
37  # connect-timeout: 5000
38  # read-write timeout (ms)
39  # read-write-timeout: 60000
40  # stdout, stderr or file-path
41  # log-file: stderr
42  # debug, info, warn or error
43  # log-level: warn
44  # If present, run as a daemon with this pid file
45  # pid-file: /run/hev-socks5-tunnel.pid
46  # If present, set rlimit nofile; else use default value
```

```
47 # limit-nofile: 65535
```

Run

Linux

```
1 # Set socks5.mark = 438
2 bin/hev-socks5-tunnel conf/main.yml
3
4 # Bypass upstream socks5 server
5 sudo ip rule add fwmark 0x438 lookup main pref 10
6 sudo ip -6 rule add fwmark 0x438 lookup main pref 10
7
8 # Route others
9 sudo ip route add default dev tun0 table 20
10 sudo ip rule add lookup 20 pref 20
11 sudo ip -6 route add default dev tun0 table 20
12 sudo ip -6 rule add lookup 20 pref 20
```

FreeBSD/macOS

```
1 # Bypass upstream socks5 server
2 # 10.0.0.1: socks5 server
3 # 10.0.2.2: default gateway
4 sudo route add -net 10.0.0.1/32 10.0.2.2
5
6 # Route others
7 sudo route change -inet default -interface utun99
8 sudo route change -inet6 default -interface utun99
```

API

```
1 /**
2  * hev_socks5_tunnel_main:
3  * @config_path: config file path
4  * @tun_fd: tunnel file descriptor
5  *
6  * Start and run the socks5 tunnel, this function will blocks until the
7  * hev_socks5_tunnel_quit is called or an error occurs.
8  *
9  * Alias of hev_socks5_tunnel_main_from_file
10 *
11 * Returns: returns zero on successful, otherwise returns -1.
12 *
13 * Since: 2.4.6
14 */
15 int hev_socks5_tunnel_main (const char *config_path, int tun_fd);
16
17 /**
```

```

18  * hev_socks5_tunnel_main_from_file:
19  * @config_path: config file path
20  * @tun_fd: tunnel file descriptor
21  *
22  * Start and run the socks5 tunnel, this function will blocks until the
23  * hev_socks5_tunnel_quit is called or an error occurs.
24  *
25  * Returns: returns zero on successful, otherwise returns -1.
26  *
27  * Since: 2.6.7
28  */
29  int hev_socks5_tunnel_main_from_file (const char *config_path, int
    tun_fd);
30
31  /**
32  * hev_socks5_tunnel_main_from_str:
33  * @config_str: string config
34  * @config_len: the byte length of string config
35  * @tun_fd: tunnel file descriptor
36  *
37  * Start and run the socks5 tunnel, this function will blocks until the
38  * hev_socks5_tunnel_quit is called or an error occurs.
39  *
40  * Returns: returns zero on successful, otherwise returns -1.
41  *
42  * Since: 2.6.7
43  */
44  int hev_socks5_tunnel_main_from_str (const unsigned char *config_str,
    unsigned int config_len, int
    tun_fd);
45
46
47  /**
48  * hev_socks5_tunnel_quit:
49  *
50  * Stop the socks5 tunnel.
51  *
52  * Since: 2.4.6
53  */
54  void hev_socks5_tunnel_quit (void);
55
56  /**
57  * hev_socks5_tunnel_stats:
58  * @tx_packets (out): transmitted packets
59  * @tx_bytes (out): transmitted bytes
60  * @rx_packets (out): received packets
61  * @rx_bytes (out): received bytes
62  *
63  * Retrieve tunnel interface traffic statistics.
64  *
65  * Since: 2.6.5
66  */

```

```
67 void hev_socks5_tunnel_stats (size_t *tx_packets, size_t *tx_bytes,  
68                               size_t *rx_packets, size_t *rx_bytes);
```

Use Cases

Android VPN

- SocksTun

iOS

- Tun2SocksKit

Contributors

- **arrior** - <https://github.com/arrior>
- **EbrahimTahernejad** - <https://github.com/EbrahimTahernejad>
- **heiby** - <https://github.com/heiby>
- **hev** - <https://hev.cc>
- **pronebird** - <https://github.com/pronebird>

License

MIT