Rust-based
MASQUE-Proxying for Lower OSI-Layer Protocol Traffic

Existing proxying technology comes with drawbacks. SOCKS is unencrypted and HTTP CONNECT is currently limited to proxying of TCP data. The MASQUE working group of IETF [1] plans to extend the current HTTP CONNECT with capabilities for proxying UDP and even IP-layer traffic. They focus on HTTP/3 which runs on top of the new transport protocol QUIC, which offers improved performance, embedded security and multiplexing. This technology has already attracted the attention of research [2], but is still relatively unknown and has especially seen only very few implementations. Its biggest use case is Apple’s iCloud Private Relay [3], their code is however closed source. The main goal of this thesis is now to use one of the existing, Rust based QUIC/HTTP3-libraries ([4] or others) and extend them with the functionality UDP and IP-layer data proxying.

Motivation

In order to familiarize you with QUIC, MASQUE RFCs and drafts and your chosen implementation, you need to:

- Implement support for lower-layer proxying in it
- Test and benchmark your implementation

Your Task

References

[1] https://datatracker.ietf.org/wg/masque/about/

Requirements

Good programming skills in Rust (or willingness to acquire them), familiarity with GNU/Linux and network protocols.

Contact

Lion Steger stegerl@net.in.tum.de
Richard von Seck seck@net.in.tum.de