MASQUE IP Proxying Requirements

Use Cases & Requirements

draft-ietf-masque-ip-proxy-reqs

Recap: Unchanged requirements

Use Cases

Point-to-Point

Create a private, encrypted, authenticated network between two endpoints.

Useful for container networking to provide a virtual (overlay) network with addressing separate from the physical transport.

Point-to-Network

More traditional remote access "VPN" use case.

Frequently used when a user needs to connect to a different network (such as an enterprise network) for access to resources that are not exposed to the public Internet.

Network-to-Network

a.k.a Site-to-Site VPN. Like Point-to-Network, connect to a network that is not exposed publicly. The entire networks are connected to each other and route packets transparently without a VPN client installed on each network endpoint.

Useful when devices cannot easily be made to run VPN clients.

Security Requirements

- **Transport** - Must run over a protocol that provides mutual authentication, confidentiality and integrity. e.g., QUIC, TLS
- **Indistinguishability** - A passive network observer should not be able to distinguish an IP session from regular encrypted HTTPS Web traffic. This requirement is only about network-visible unencrypted bytes, traffic analysis is out of scope.
- **Authentication** - In addition to authentication provided by transport, must have the ability to mutually authenticate during the establishment of an IP session by way of e.g., OAuth token or vendor-specific auth.
- **Identity** - Endpoints should use a cryptographically authenticated identifier to determine the identity of their peer.

Session Requirements

- **Establishment** - Allow a client to request an IP session along with configuration options. Server may accept/deny client's request at its discretion.
- **Proxy IP Packets** - The protocol will establish Data Transports (either using QUIC DATAGRAM or STREAM frames), which will be able to forward IP packets, in their unmodified entirety. Will support both IPv6 and IPv4.
- **Multiplexing** - Multiple independent IP sessions should be supported over a single HTTP connection.
- **Support HTTP/2 & HTTP/3** - Should strongly prefer H3 (using QUIC DATAGRAM frames), but allow support for H2 in cases where UDP is unavailable.

Session Requirements (cont.)

- **Reliable Transmission** - While generally desirable to transmit IP packets unreliably, the protocol will provide a mechanism to allow forwarding some packets reliably.
- **Flow Control** - The protocol will allow the ability to proxy IP packets without flow control, at least when HTTP/3 is in use.
- **Load Balancing** - If using streams, both client & server should be permitted to create additional streams -- allowing multi-threaded servers to support increasing throughput.

Out of Scope

- **Addressing Architecture** - This protocol will only consider proxying of IP packets. It will have no opinions about how the IPs assigned are determined or managed.

- **Translation** - Network Address Translation (NAT) or any other modification to packets they forwarded using this protocol is out of scope.

- **IP Packet Extraction** - How packets are forwarded between the IP proxying connection and the physical network is out of scope. This is deliberately not specified and will be left to individual implementations.

Clarifications

Configuration Requirements

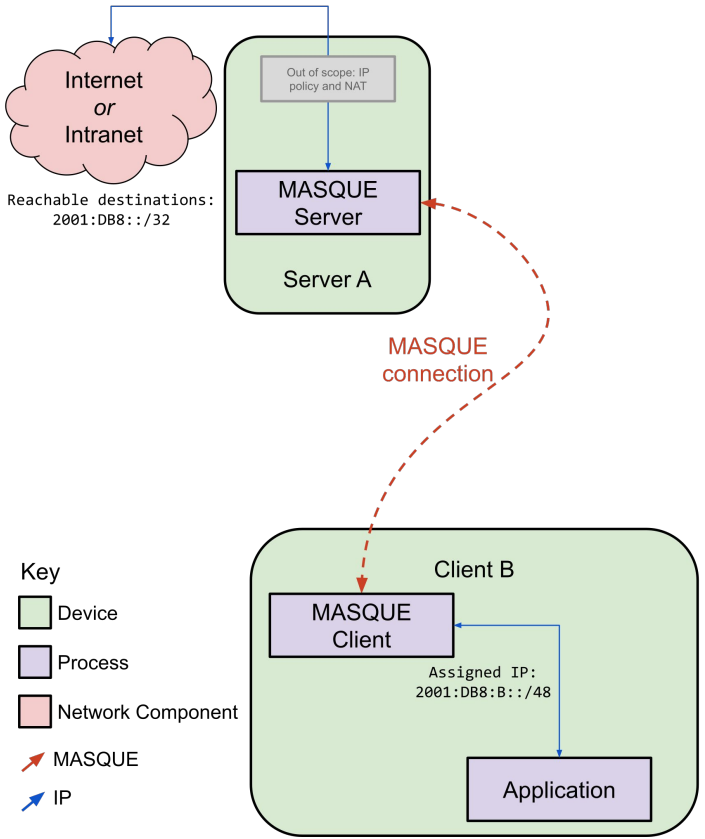
- **MTU Negotiation** - Endpoints will be allowed to negotiate the MTU in use over the IP session.
- **IP Assignment** - Either client or server may request to be assigned an IP address range. In response, the peer will respond with an IP address range of its choosing.
- **Reachable Destinations** - At any point in an IP session, both client and server may inform its peer of a set of reachable IP addresses that it is willing to forward traffic to.
- **Extensibility** - The protocol will provide a mechanism by which clients and servers can add extension information to the exchange that establishes the IP session, and convey extension information at any point in the lifetime of the IP session.

Addressing Example - Point to Network

- (These addresses are internal to the tunnel)

- Client: can I have address range please?
- Server: 2001:db8:b::/64 is yours!
  - Inside the tunnel, only packets to/from that range are allowed – client drops received tunneled packets to other destinations, server drops received tunneled packets from other sources

- Server: I can let you reach 2001:db8::/32
  - Inside the tunnel, only packets to/from that range are allowed – server drops received tunneled packets to other destinations, client drops received tunneled packets from other sources

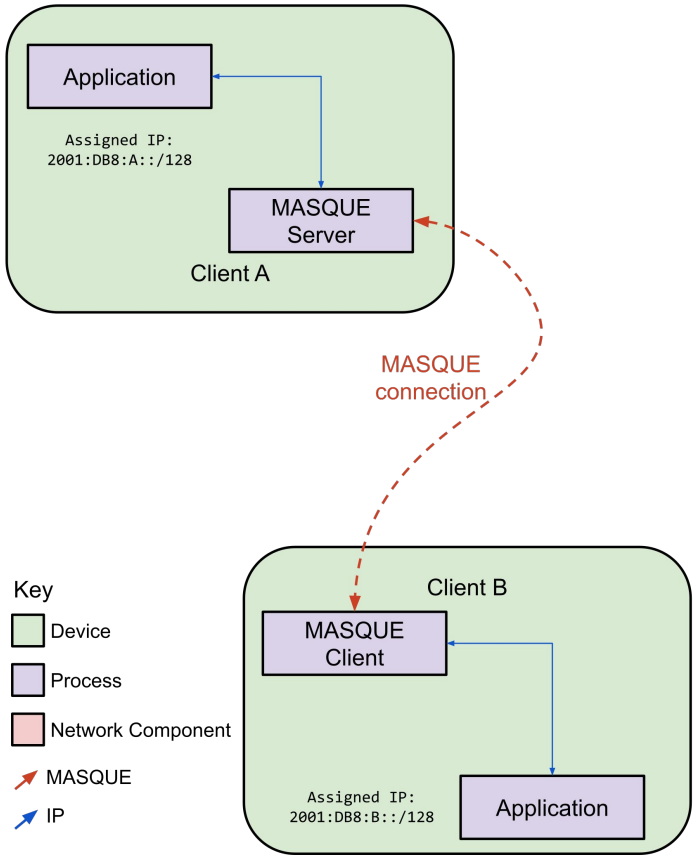


Addressing Example - Point to Point

- Point to point is a special case of point to network.

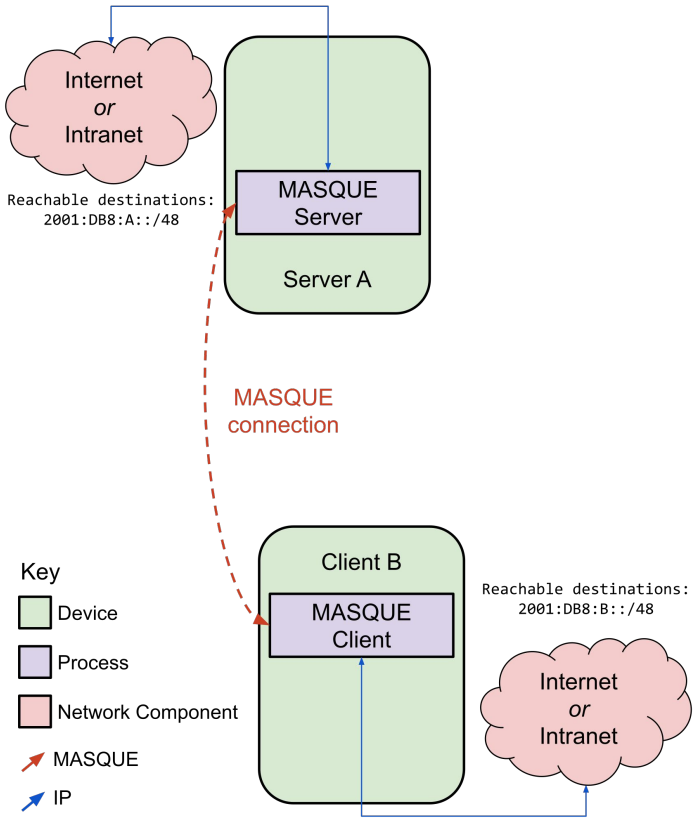
- Client: can I have address range please?
- Server: 2001:db8::b/128 is yours!

- Server: I can let you reach 2001:db8::a/128



Addressing Example - Network to Network

- Server: I can let you reach 2001:db8:a::/48
- Client: I can let you reach 2001:db8:b::/48



MASQUE IP Proxying Requirements

Use Cases & Requirements

draft-ietf-masque-ip-proxy-reqs