July 24th, 2025

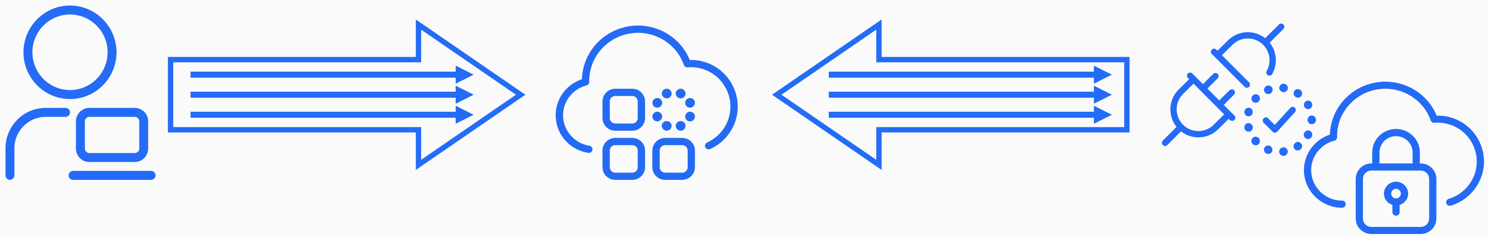
Reverse HTTP for CONNECT

Yaroslav Rosomakho

Traditional remote access VPN architecture

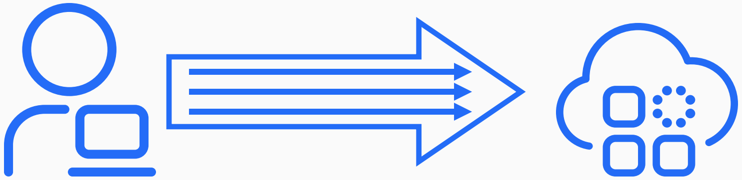
- Client connects directly to VPN concentrator
- Client obtains IP address for tunnel interface
- Transport carries IP packets
- Encapsulated connectivity is bidirectional
- IKEv2/CONNECT-IP/Wireguard/VPNs over TLS

ZTNA architecture



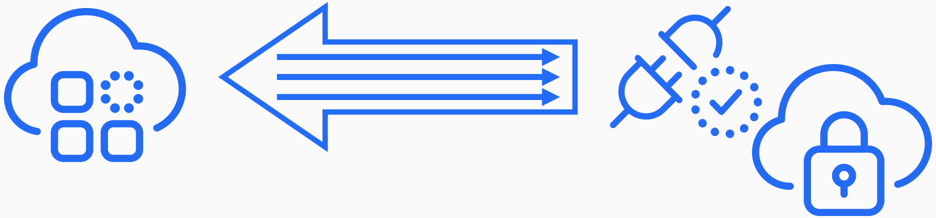
- Client and worker establish outbound connections to proxy
- Client and worker do not use “internal” IP addresses
- Client sends flows to proxy which forwards flows to connector
- Transport carries L4 flows

Client to proxy connectivity



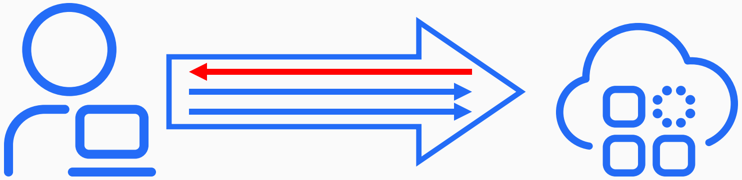
- HTTP CONNECT, templated CONNECT-TCP, CONNECT-UDP are a great fit and can offer cross-vendor interoperability

Standard Proxy to Worker connectivity options available today



- **CONNECT-IP**: requires IP addresses and routing. Performs badly over TCP. MTU limitations...
- **draft-masque-connect-udp-listen**: UDP only, extra overhead, requires IP addresses to distinguish clients
- **Reversing roles after transport is ready**: outbound connectivity is still useful (authentication, status reporting, bi-directional connectivity)

Proxy to client connectivity



- RDP, SSH, VNC, SCCM for remote administration or user assist

Desired properties of solution for these use cases

- Server-initiated requests over HTTP/3 and HTTP/2
- Flow encapsulation similar to CONNECT/CONNECT-UDP
- Regular and reverse requests over the same HTTP transport

- draft-rosomakho-masque-reverse-connect is a potential solution for this use case

Questions?