WIKIPEDIA

HTTP tunnel

HTTP tunneling is used to create a network link between two computers in conditions of restricted network connectivity including firewalls, NATs and ACLs, among other restrictions. The tunnel is created by an intermediary called a proxy server which is usually located in a DMZ.

Tunneling can also allow communication using a protocol that normally wouldn’t be supported on the restricted network.

Contents

HTTP CONNECT method

The most common form of HTTP tunneling is the standardized HTTP CONNECT method.[1][2] In this mechanism, the client asks an HTTP proxy server to forward the TCP connection to the desired destination. The server then proceeds to make the connection on behalf of the client. Once the connection has been established by the server, the proxy server continues to proxy the TCP stream to and from the client. Only the initial connection request is HTTP - after that, the server simply proxies the established TCP connection.

This mechanism is how a client behind an HTTP proxy can access websites using SSL or TLS (i.e. HTTPS). Proxy servers may also limit connections by only allowing connections to the default HTTPS port 443, whitelisting hosts, or blocking traffic which doesn't appear to be SSL.

Example negotiation

The client connects to the proxy server and requests tunneling by specifying the port and the host computer to which it would like to connect. The port is used to indicate the protocol being requested.[3]

```
CONNECT streamline.t-mobile.com:22 HTTP/1.1
Proxy-Authorization: Basic encoded-credentials
```

If the connection was allowed and the proxy has connected to the specified host then the proxy will return a 2XX success response.[3]

```
HTTP/1.1 200 OK
```

The client is now being proxied to the remote host. Any data sent to the proxy server is now forwarded, unmodified, to the remote host[3] and the client can communicate using any protocol accepted by the remote host. In the example below, the client is starting SSH communications, as hinted at by the port number in the initial CONNECT request.

```
SSH-2.0-OpenSSH_4.3\r\n
...
```

HTTP tunneling without using CONNECT

A HTTP tunnel can also be implemented using only the usual HTTP methods as POST, GET, PUT and DELETE. This is similar to the approach used in Bidirectional-streams Over Synchronous HTTP (BOSH).

In this proof-of-concept program (https://github.com/luizluca/bridge), a special HTTP server runs outside the protected network and a client program is run on a computer inside the protected network. Whenever any network traffic is passed from the client, the client repackages the traffic data as a HTTP request and relays the data to the outside server, which extracts and executes the original network request for the client. The response to the request, sent to the server, is then repackaged as an HTTP response and relayed back to the client. Since all traffic is encapsulated inside normal GET and POST requests and responses, this approach works through most proxies and firewalls.

See also

- ICMP tunnel
- Pseudo-wire
- Tunnel broker
- Virtual private network (VPN)
- Virtual extensible LAN
- Network virtualization using generic routing encapsulation

References

1. Fielding, R. (June 1999). "Method Definitions, CONNECT" (https://datatracker.ietf.org/doc/html/rfc2616#section-9.9). *Hypertext Transfer Protocol -- HTTP/1.1* (https://datatracker.ietf.org/doc/html/rfc2616). IETF. p. 56. sec. 9.9. doi:10.17487/RFC2616 (https://doi.org/10.17487%2FRFC2616). RFC 2616 (https://datatracker.ietf.org/doc/html/rfc2616). Retrieved 2010-07-09.
2. "Upgrading to TLS Within HTTP/1.1 (RFC 2817)" (https://tools.ietf.org/html/rfc2817). RFC 2817 (https://tools.ietf.org/html/rfc2817). Retrieved 3 July 2011.
3. "CONNECT" (https://datatracker.ietf.org/doc/html/rfc7231#section-4.3.6). *HTTP/1.1 Semantics and Content* (https://datatracker.ietf.org/doc/html/rfc7231). IETF. June 2014. p. 30. sec. 4.3.6. doi:10.17487/RFC7231 (https://doi.org/10.17487%2FRFC7231). RFC 7231 (https://datatracker.ietf.org/doc/html/rfc7231). Retrieved 4 November 2017.

Retrieved from "https://en.wikipedia.org/w/index.php?title=HTTP_tunnel&oldid=1063937717"