VPNless for the Cloud

Patrick Mourot

ALE International

Follow this and additional works at: http://www.tdcommons.org/dpubs_series

Recommended Citation
Mourot, Patrick, "VPNless for the Cloud", Technical Disclosure Commons, (February 08, 2017)
http://www.tdcommons.org/dpubs_series/389

This work is licensed under a Creative Commons Attribution 4.0 License.
This Article is brought to you for free and open access by Technical Disclosure Commons. It has been accepted for inclusion in Defensive Publications Series by an authorized administrator of Technical Disclosure Commons.
VPNLess for the cloud

Contributor: Patrick Mourot – ALE International

A solution, not based on VPN but on WebPush, is proposed for the delivery of call control features (basic or enhanced) to users when connected to a corporate LAN, or working at home when a call server is in the cloud, i.e., when Signaling and Media are delivered in an IP environment. Signaling is often proprietary (i.e.: not standardized) and Media uses UDP which is often blocked by NAT/Firewalls.

The WebPush protocol is defined in IETF (Web-Based Push Notifications working group) whereas W3C (Push API group) defines usages for notifications. As shown Figure 1, a client includes a WebPush Client/Server software as defined by ongoing standards (harmonization of cloud messaging at Google, Push Notification Service at Apple, Notification Service for Windows at Microsoft…) and also the known techniques to travel media through NAT/Firewalls (ICE, STUN client/server).

Figure 1

Signaling (black arrow) uses Http/https to vehicle signaling messages (i.e.: proprietary session messages) and the media path is established as is known with a STUN server (blue line) and the Media itself (red lines) goes to the destination either via a TURN/STUN relay or a media gateway if needed.

WebPush server at the client allows for the IP-PBX (Call Server) to register the device and be granted services. Once done an Object containing parameters of the session, call and URL of the service is pushed back to the IP-PBX. When the device wants to send Signaling messages to the IP-PBX (Like on-hook, line…) it uses the push services (Push Object) to transport its proprietary messages.

The IP-PBX acts also as a Webpush server to the device when it wants to send signaling message or responses or notifications. In order to accept/understand messages the device also includes a Webpush client (as a reminder: the WebPush protocol is well suited for headless browser (browsers with no user interface) that be configured as a TSR (terminate and stay resident) application to receive Server Sent events from the application server)).

As the mechanism is based on HTTP/2 (HTTP/2.0), the NAT/Firewall is to be transparent for the deployment.
Figure 2 provides a detailed view of the exchanges between Push Client/Server:

When an application has requested permission to a Push client to receive messages, the Push Client registers with the server and once confirmed messages can be delivered from the server to the Applications through the Client. Both the IP-PBX and the device act as a client or a server.