Fast Wireguard Mesh
VPP + wgsd + wg = ❤

Interconnect your services with taste

Benoît Ganne, bganne@cisco.com
Wireguard Mesh

How to dynamically and securely interconnect services running at different locations?

- Must be secure
  - Services run at different locations (crossing internet) and/or in public clouds
  - All communications must be encrypted

- Must be efficient
  - Mesh: direct peer-to-peer connections between services
  - Fast: crypto is CPU intensive

- Must be automated
  - How to do service discovery?
  - How to automate interconnects?

- Must be simple to configure and operate

⇒ Wireguard DNS Service Discovery + VPP to the rescue!
WireGuard

• WireGuard is a new VPN protocol and tools
• Popular thanks to its ease-of-use vs OpenVPN and IPsec/IKEv2
• UDP-based

→ Interesting solution to easily interconnect services in uncontrolled network environments, eg. multi-cloud
VPP

• Opensource userspace dataplane project: https://fd.io
• Fastest userspace dataplane running on general-purpose CPUs (x86, ARM)
• Used to interconnect services locally in a server or services themselves
  • vSwitch, vRouter, services load-balancer, etc.
  • Firewall-as-a-Service, Load-Balancer-as-a-Service, etc.
• Interoperable with Linux netstack
• Very fast crypto
  • IPsec support since a long time
  • WireGuard added recently
Wireguard DNS Service Discovery

• New opensource project: wgsd “Wireguard DNS Service Discovery” by Jordan Whited
  • https://www.jordanwhited.com/posts/wireguard-endpoint-discovery-nat-traversal/
  • https://github.com/jwhited/wgsd

• Use DNS-SD (RFC6763) to publish Wireguard peers
  • All peers connect to the “registry” through Wireguard
  • The registry serves all of its peers through DNS-SD (SRV records)
  • Any peer can request the configuration of another service to the registry and then connect to it directly

• Wireguard is used as a gatekeeper, database and even for NAT traversal
  • Gatekeeper: all mesh participants must be able to connect to the registry node (so must know the secret)
  • Database: the wgsd service does not keep track of the mesh participants, it relies on the Wireguard peer database
  • NAT traversal: other mesh participants connect to the destination IP and UDP port allocated when connecting to the registry (NAT punch-holing)
Wireguard Mesh

Cloud A

Cloud B

Cloud C

Host

VPP

Service 1

... Service n

Host

Linux

Service 1

... Service n

Host

VPP

Service 1

... Service n

Host

Linux

Service 1

... Service n

FOSDEM 2021

7 Feb 2021
Demo!
Status

• This is a work-in-progress
  • Several modifications are merged/implemented in upstream wgsd
  • Others are in-progress
  • VPP Wireguard implementation is still young
  • VPP and wgsd integration is still a bit rough 😊

• Feel free to try it out!
  • https://github.com/bganne/wgsd/blob/master/vagrant/README

Thank you!