# Auto Discovery VPN Protocol

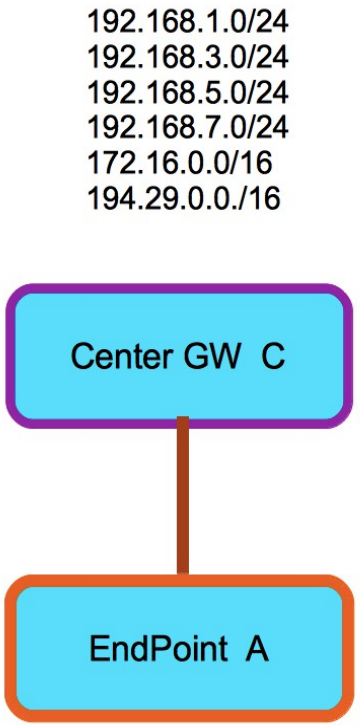
draft-sathyanarayan-ipsecme-advpn-00

# Auto Discovery VPN Protocol

- A solution proposal for the AD-VPN problem statement.
- -00 version submitted 5-July
- 46 pages
- Based on “shortcuts”:
  - If gateway C decrypts traffic from A, re-encrypts it and sends it to B, then C can tell A and B to communicate directly.

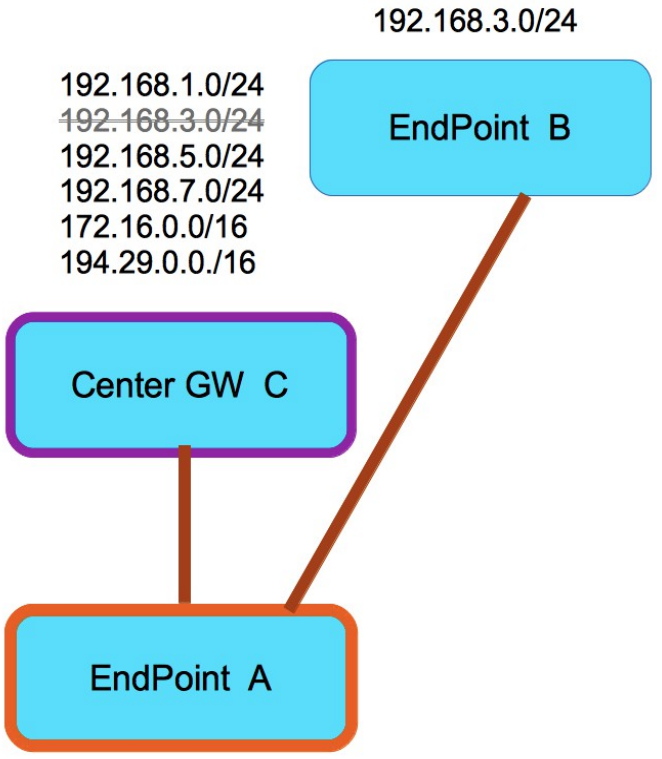
# Auto Discovery VPN example

- These slides are mostly from the viewpoint of EndPoint  A.
- EndPoint A knows about only one gateway – C, and there's lots of subnets behind that.
- There's some traffic going through A to host 192.168.3.7.

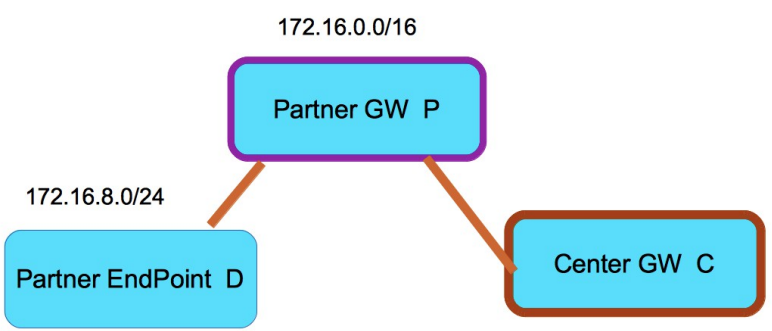


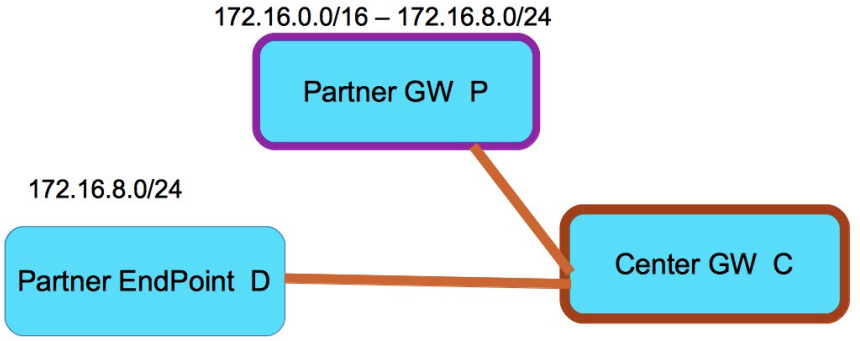
# Auto Discovery VPN example

- At some point GW C sends SHORTCUT messages to A and B, introducing them and updating their SPDs and PADs.
- Endpoint A then sets up a tunnel with EndPoint B, and the traffic now goes directly rather than through GW C.



# AD-VPN example – Cross Domain

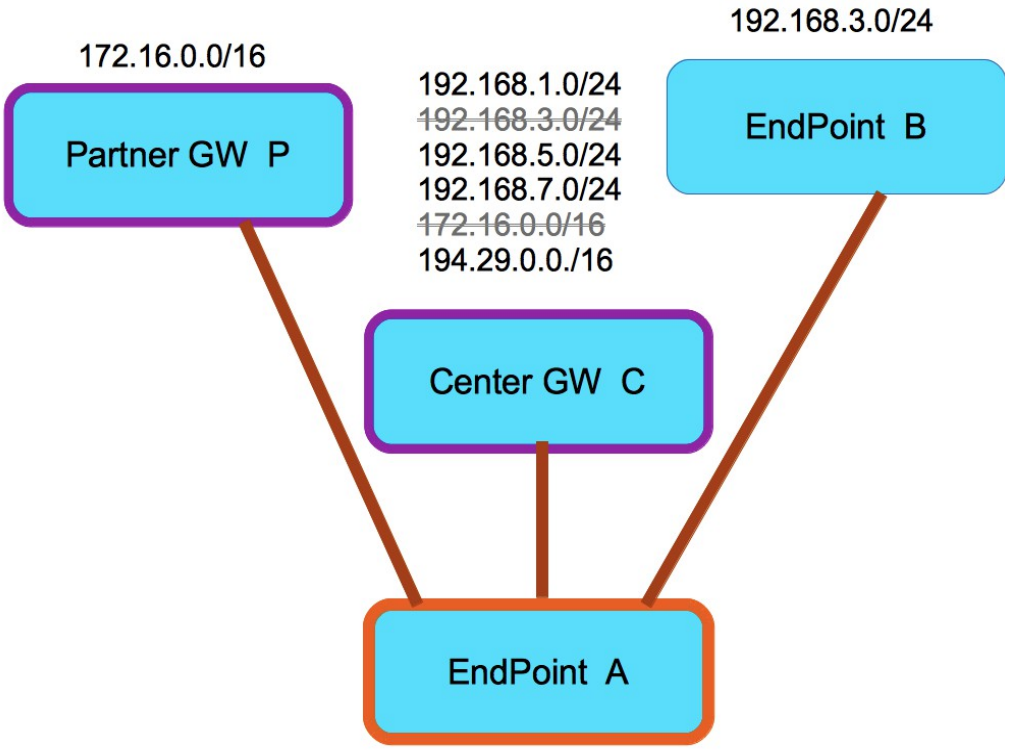



- A (not depicted) doesn't know this, but traffic to the 172.16.0.0/16 subnet is routed to partner gateway P, which also routes traffic to endpoint D.
- SHORTCUTS work in cross domain too. Likely with PSK.

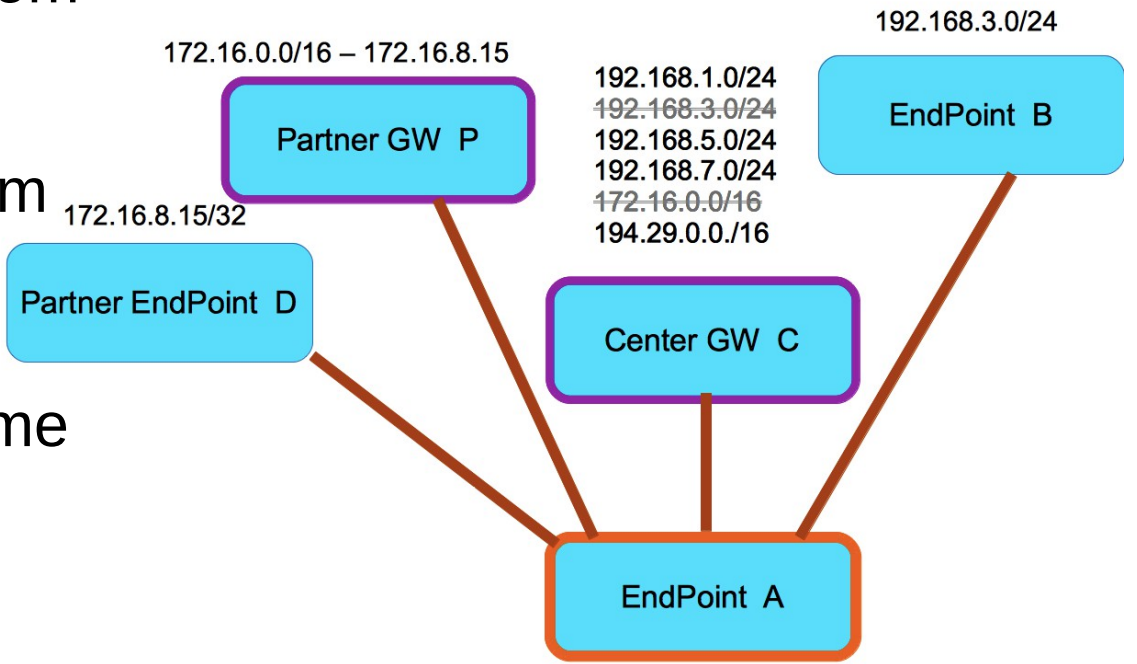
# Auto Discovery VPN example

- When traffic starts going through A to 172.16.8.15 GW C tries to delegate the traffic straight to EndPoint D (not depicted).
- However, policy on D rejects this, because it doesn't accept SHORTCUTS from partners.
- Instead, GW C delegates the traffic to Partner GW P.



# Auto Discovery VPN example

- When a suggestion comes from Partner GW P, Endpoint D accepts it. Now traffic to 172.16.8.15 flows directly from EndPoint A to EndPoint D.
- All these SHORTCUTs are time-limited, so after some time traffic reverts to configured policy, unless the shortcut is renewed.



# Auto Discovery VPN Principles

- If C tells A and B to shortcut, C is called the “suggester”.
- A only accepts shortcuts for things that its current policy says should be sent to C. C cannot delegate traffic it doesn't “own”.
- The suggester provides credentials, identities, and traffic selectors (see section 3.4).
- C decides which of A and B is the Initiator.
  - B first suggests the shortcut to the Responder.
  - If the Responder does not reject, it suggests to the Initiator.

# Auto Discovery VPN Benefits

- Improved network performance through reduced latency.
- Better scalability in two ways:
  - CPU and bandwidth load on the center gateways, preventing it from becoming a bottleneck. Also reduces total load.
  - A single tunnel (of limited bandwidth) between administrative domains is replaced with multiple tunnels.
- Better reliability: some traffic continues even if a center gateway or a cross-domain tunnel endpoint fails.

# Auto Discovery VPN Benefits

- Easier management / maintenance
  - Less configuration
  - Configuration is automatically harmonized
- Local shortcut decisions lead to quick convergence.
- Cross domain support
  - Suggester-generated PSKs makes credential-provisioning easy.

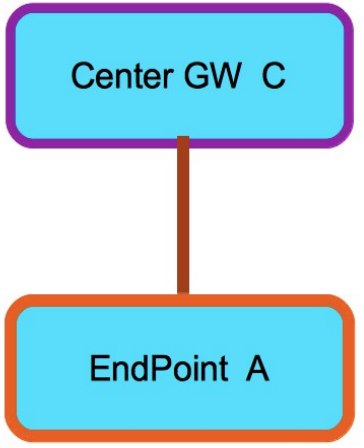
# Missing Pieces

- “Initial” or “Basic” configuration
- Non-GW suggesters (?)
- Shortcut Deletion
- NAT Traversal
- “Drop” and “Bypass” shortcuts

# Missing Pieces – Basic Config

- In the example on the right, there are many EndPoint gateways, each hiding a network like 192.168.x.0/24.
- If we add a new gateway, Endpoint E with subnet 192.168.247.0/24, what do we need to do?
  - We need to configure Center GW C to know about E and its protected network. There's no avoiding that.
  - We need to configure EndPoint E to know about C and all of the networks that it protects.
  - We need to configure all endpoints like A so that their SPD will show that 192.168.247.0/24 is behind Center GW C.
- But the PS requires that configuration be “minimal”

192.168.1.0/24
192.168.3.0/24
192.168.5.0/24
192.168.7.0/24
172.16.0.0/16
194.29.0.0./16



# Missing Piece – Basic Config

- There are two ways to overcome this:
  - Introduce a “trusted suggester”, who ***is*** allowed to make suggestions for any traffic selectors. The center gateway is an obvious candidate for this role, as is an external non-gw entity.
    - Since the “full picture” might be very complex, we might need to allow a hash-and-URL type shortcut.
    - We also might need to not call it a “shortcut”.
  - Begin with some very generic configuration, such as routing all traffic through the center gateway.

# Missing Piece – Off-Path Suggester

- Yaron has suggested allowing a non-GW to be a suggester.
- This would allow shortcuts by some device that has the “big picture” rather than a view of a particular flow.
- OTOH most of the use cases we considered involved responding to actual flows, so this off-path suggester would need to get real-time reports from gateways, and to be trusted by endpoints.
- If it's needed, I guess it should be a trusted suggester.

# Missing Piece – Shortcut Deletion

- Suppose the suggester would like to delete the shortcut.
  - Maybe the responder agreed, but the initiator rejected, so we want to tell the responder that the shortcut tunnel isn't coming.
  - Maybe policy has changed.
- This would probably require some unique identifier.
  - Maybe send it also in the exchange between the peers?

# Missing Piece – NAT Traversal

- One of the benefits of a hub and spoke topology is that all spokes can be behind NAT.
  - IKEv2 supports peers behind NAT only as initiators.
- The current spec doesn't help two NAT-ted spokes to create a shortcut.
- However, we know that VoIP applications manage to work even when both endpoints are NAT-ted.
- We would like to have a similar mechanism for AD-VPN.

# Missing Piece – Drop and Bypass

- Occasionally, a configuration might happen where certain flows come through a VPN tunnel, and are then either forwarded to the Internet, or simply dropped.
  - Rather than being forwarded to the protected domain or to another peer.
- Sometimes, this is by design: Center GW C has some Next Generation Firewall capabilities that are missing in the spokes, but other times it's just a coincidence of some simplified initial configuration such as route-all-traffic.

# Missing Piece – Drop and Bypass

- Especially for the dropped traffic, this is a waste of resources.
- The SHORTCUT in the current draft adds a PROTECT entry to the SPD. We would like to add a variation that adds BYPASS or DROP entries into the SPD.
  - Maybe also a BYPASS-with-NAT, although that should probably be a local decision.
- If we do that, the name “shortcut” becomes less appropriate.

# Questions?

Funny, Unencumbered Question Mark Goes Here

# Extra Slides

# Compared with Requirements (1/6)

- Minimal changes when adding or deleting a gateway / endpoint
  - Only the ones connected to the new endpoint need change. The rest learn through shortcuts.
- No configuration changes when setting up new tunnels
  - Shortcuts modify the SPD and PAD without configuration changes.

# Compared with Requirements (2/6)

- Work with tunneling and routing protocols
  - Yes. The traffic selectors can indicate a GRE or L2TP tunnel, and routing protocols can run through that.
- Allow both meshes and stars (don't force stars to become meshes)
  - This document is silent about policy. We do not say when and what a suggester should suggest, nor what suggestions should be accepted. If spokes are configured not to make shortcuts with other spokes, they won't,

# Compared with Requirements (3/6)

- No long term credentials for others. Compromised nodes cannot attack the rest of the VPN
  - By limiting configuration changes to delegation, compromised spokes can only affect traffic that already flows through them.
  - PSKs provisioned by this protocol expire after a set time.
- Seamless handoff when roaming
  - A roaming endpoint has to establish tunnels or update existing ones through a mechanism like mobike. If setting up a tunnel with the shortcut peer fails, the roaming endpoint can revert.

# Compared with Requirements (4/6)

- Easy handoff to other gateways
  - New tunnels do not close the old tunnels, so traffic continues through the old one until the new one is ready.
- Work with some endpoints behind NAT
  - Suggester knows whether each of the peers is behind NAT
  - If only one, it becomes the initiator
  - If both, we need something better than NAT-T
    - STUN or STUN-like?

# Compared with Requirements (5/6)

- Reportable and Manageable, but no MIB
  - We have several events that are reportable:
    - Gateway makes a suggestion
    - Peer accepts or rejects
    - Shortcut times out or reverts because of failure
  - We did not create a MIB (success!!!)
- Allow cross-domain
  - Yes, through generated PSKs

# Compared with Requirements (6/6)

- Administrator can configure star, mesh or partial
  - We do not specify policy, but policies can do this.
- Monitoring, logging, reporting
  - All can be supported (but we did not define a MIB)
- Multicast and L3VPN
  - This all depends on the protocols and ports in the TS
- Allow per-peer QoS
  - It's allowed, but this protocol does not communicate QoS policy.