Contents

[End State Patterns: 2](#_Toc195278981)

[Single Vnet (Hub and Spoke) with Azure VPN GW Solution: 2](#_Toc195278982)

[Architecture Diagram: 5](#_Toc195278983)

[Pros: 5](#_Toc195278984)

[Cons: 5](#_Toc195278985)

[Transit/Spoke VNET for VPN: 6](#_Toc195278986)

[Solution Description: 6](#_Toc195278987)

[Diagram: 7](#_Toc195278988)

[Pros: 8](#_Toc195278989)

[Cons: 8](#_Toc195278990)

[vWAN based Solution: 8](#_Toc195278991)

[Solution Description: 8](#_Toc195278992)

[Architecture Diagram: 9](#_Toc195278993)

[Pros: 9](#_Toc195278994)

[Cons: 9](#_Toc195278995)

[Single Vnet (Hub and Spoke) with Palo VPN GW Solution: 9](#_Toc195278996)

[Solutions Details: 9](#_Toc195278997)

[Architecture diagram: 11](#_Toc195278998)

[Pros: 11](#_Toc195278999)

[Cons: 11](#_Toc195279000)

[End-State Pattern Selection Criteria: 11](#_Toc195279001)

[Phased deployment to End State: 12](#_Toc195279002)

[Phase 1: Extend Layer 2 12](#_Toc195279003)

[Solution Description: 12](#_Toc195279004)

[Diagram: 12](#_Toc195279005)

[CutOver: 12](#_Toc195279006)

[Phase 2: Move VPN Tunnels, one at a time from on-prem to Azure 12](#_Toc195279007)

[Solution Description: 12](#_Toc195279008)

[Diagram: 13](#_Toc195279009)

[Cutover: 13](#_Toc195279010)

[Phase 3: Move Default GW to AVS: 13](#_Toc195279011)

[Solution Description: 13](#_Toc195279012)

[Diagram: 13](#_Toc195279013)

[Cutover: 13](#_Toc195279014)

[Phase 4: Move Edge capabilities to Azure: 13](#_Toc195279015)

[Solution Description: 13](#_Toc195279016)

[Diagram: 13](#_Toc195279017)

[Cutover: 14](#_Toc195279018)

# End State Patterns:

## Single Vnet (Hub and Spoke) with Azure VPN GW Solution:

* Other spokes of the Hub and Spoke solution are dotted lines because we don't know there is a use case yet for needing the spokes.
* Every single VPN is considered East West traffic and will be filtered through the Palo inside interface. The next hop will be the ILB. Then it will forward to one of Palo Altos, and then traffic leaves the same interface to another destination that is explicitly allowed.
  + The Azure VPN gateway will forward to a LB
* VPN gateway in the hub is connected to AVS via the ER circuit and ER GW in the hub.
* With the ER circuit connecting the hub ER GW and AVS, the Hub learns the AVS prefixes
* In a hub and spoke design, ER GW learnt routes will leak to the end customers via the ARS acting as a route reflector between ER Gateway and VPN Gateway.
* NATing:
  + PAN FW will need to SNAT the end customer's traffic to AVS
  + Tunnels will terminate on the VPN GW in the Hub Vnet with destination as PAN. When end customers want to communicate with the app hosted on AVS, the PAN will DNAT the traffic and SNAT it.
* Once AVS is connected to the ER GW in the Hub, T0 in AVS will be responsible for advertising whatever it knows in AVS via the ER GW (in the hub Vnet) to
  + VNets in Azure
  + On-prem locations
  + Learning:
    - AVS is learning Vnets connected to the Hub
    - VNets are learning AVS routes
* We want everything to go through the Palo for outbound (internet or elsewhere)
* ARS is needed to reflect/exchange the routes between the ER GW and VPN GW because by default VPN GW is not learning the routes from ER GW (and everything connected to the ER GW like AVS).
  + It will all be static routing from Palo to the next hop of something in AVS
  + After traffic is inspected by Palo, it will go to its destination.
  + Palo is learning AVS routes and learning spoke routes because of peering
    - You cannot turn off the Palo from learning AVS and spoke routes via BGP
    - The ER GW will always send everything it knows to the Palo
  + Can you create a route table on the NIC that connected to the private/inside subnet and not propagate its routes ?
  + Palos would want to learn the BGP routes from AVS (via NSXT) on a separate route table with propagation enabled, so you can see it as the next hop coming from the ER GW.
  + The NIC that forwards left-right traffic needs to know about BGP which is whatever the ER GW is learning, so it can send traffic there.
  + Next hop on the Palo won't have EGP, instead have a static route towards the AVS inside subnet .1 address. Then the inside subnet will look at its route table and see the BGP learnt routes from the NSXT to route to the destination.
* Why do we need ARS?
  + ARS provides a way to
    - push 0/0 down from Palo to
      * AVS
      * Client/customers connecting via VPN GW
    - Push VPN learnt routes (from the clients/customers) to on-prem, AVS and VNets because AVS acts as a route reflector between VPN GW and ER GW
  + Without ARS, VPN connected clients/customers will only learn the routes of the VNets, not the routes learnt by the ER gateway (ie on-prem and AVS routes)
    - We don't provide transitivity between ER GW and VPN GW connected routes by default
    - VPN clients won't learn the routes learned through ER GW even though they are injected into the Vnet, but is not advertised.
  + Without ARS, VPN
    - won't learn the routes learnt by the ER GW (even though they are injected into the Vnet, but its not advertised).
      * The ER GW is learning the AVS Routes and the on-prem routes
    - will learn only the connected VNets to the ER GW.
  + Without ARS,
    - the VPN GW will learn the routes of the connected clients (customers) and send it to the Vnet
    - The ER GW will learn the route of its connected entities (AVS and on-prem) and sent it to the VNET
    - But they don’t send their learnt routes to each other: the VPN GW does not send its learnt routes to the ER GW. The ER GW does not send its learnt routes to the VPN GW
  + Although the ER GW and the VPN GW are in the same subnet (called GW subnet), they share each other's connected routes with the VNet, but not each other's learnt routes amongst themselves unless there is an ARS with its "enable branch to branch" setting is turned on.
  + ARS does this:
    - Takes routes learned by ER GW send it to VPN GW
    - Takes VPN routes learnt from the customers and send it to ER GW
  + How does ARS connect to VPN GW?
    - Under the hood, once you enable "Branch to Branch" setting on ARS, that will make
      * The Palo have iBGP connectivity to ARS
      * the VPN GW exchange routes though iBGP connectivity.
    - Customers don't see the iBGP connectivity on the portal or anywhere.
  + Palo can advertise 0/0, but it needs something to send 0/0 to the ER GW. Palo cannot be peered directly to the ER GW.
    - How do you generate a default route to send through to an ER GW?
      * You need ARS
  + ARS Acts as a route reflector between the VPN GW and the ER GW. ARS is like another GW besides VPN GW and ER GW
  + The OS on Palo will BGP peer with the two instances of the route server. Palo will originate the 0/0 route and send to ARS. ARS will send it to the ER GW. ER GW will leak it to the AVS and to on-prem
  + Here is the flow for 0/0 traffic:
    - Palo advertises 0/0 to ARS through BGP with next hop ILB (if running in Active-Active)
      * Peer Palo and tell ARS that ILB is the next hop
    - Thus 0/0 (with next hop ILB) gets advertised to AVS and the ER GW
* Source IP preservation: default route required if the VM in AVS needs to see the source IP.
  + Can always do SNAT on the Palo so it looks like its coming from the inside interface on the Palo, but that may not be the only requirement, may need to have source IP preservation. Need to support both: source IP preservation and SNT on the Palo inside interface.
  + Not using SNAT, ARS has an option called "Branch to Branch" which is where Palo can start learning through the ARS. That makes the ER GW send a route to the VPN GW. Now all your clients will be learning all the routes.
  + With SNAT on the on the Palo, you can have a UDR ont eh VPN GW saying if you want to go to AVS, first go to Palo. But that does not solve your clients from learning everything even on-prem.
    - We want the clients to only learn about AVS, not on-prem.
  + There are multiple design options to prevent clients from learning all routes or a simpler design is vWAN. In the next call will go over the multiple design options.

### Architecture Diagram:

### 

A diagram of a diagram

AI-generated content may be incorrect.

### Pros:



### Cons:

* Complex because clients/customers accessing VPN will know all routes to get to everything on-prem and everything in AVS:
  + With ARS acting as route reflector between ER GW (connecting on-prem and AVS) and VPN GW (connecting clients/customers), the Clients/customers connected to VPN learn all the routing to get to
    - All on-prem
      * Not necessary for clients/customers to learn on-prem routes unless the VMs are split between AVS and Austin
    - All AVS
      * Fine if Clients/customers only need to get to all AVS
  + To prevent the customer/clients from receiving all on-prem routes, build static tunnels won't work
* Routing is Comple: Even if we allow client/customers to learn all on-prem routes, because the traffic needs to be inspected, we will need a UDR on the GW subnet (same subnet hosts both ER and VPN GW) to force the traffic to go to the ILB before it goes to the VPN GW
  + Otherwise, traffic on AVS <-> ER GW <-> VPN GW <-> client happens without inspection
  + UDR on GW subnet should be:
    - AVS Prefixes --Next Hop--> Palo ILB
      * Nightmare for VPN->AVS traffic as AVS Prefixes will grow over time and UDR has to be exact AVS prefixes. It cannot be one single big prefix, it has to be the exact prefix, otherwise the VPN traffic to AVS will bypass the FW.
    - VPN Prefixes --Next Hop--> Palo ILB
      * Nightmare for AVS-> VPN traffic as VPN Prefixes will grow over time and UDR has to be exact VPN prefixes. It cannot be one single big prefix, it has to be the exact prefix, otherwise the AVS traffic to VPN will bypass the FW.

* Higher cost

## Transit/Spoke VNET for VPN:

### Solution Description:

* VPN Connections are terminated in a VPN GW in a separate VNet (not separate subnet).
* The VNet hosting the VPN GW will have:
  + Route Server
  + Palo
* You need the ARS in the HubVNet to advertise 0/0 down to AVS and on-prem
* VPN GW in a separate Vnet provides control over what the VPN GW is learning
  + The Palos in the Hub VNet will BGP peer with the ARS in the Spoke VPN Vnet and send only those prefixes that you want the clients/customers to know.
  + The ARS in the Spoke VPN VNet also has a BGP peering with the VPN GW and sends the routes learnt above to the clients/customers.
* From the perspective of the ER GW in the Hub VNet, you can't have ARS (in the VPN Net) as the next hop, nor can you have the VPN GW as the next hop. We need something to do next hop:
  + We need something to do next hop because:
    - AVS will know about VPN because the Palo (in the Hub VNet) through its BGP peering with the ARS (in the Hub VNet) is sending 0/0 + VPN routes (can be summary or 1by 1)
    - AVS -> VPN client traffic flow: AVS -> ER GW -> ILB -> a Palo in the Hub VNet. The Palo in the Hub VNet needs a next hop.
      * The Palo in the Hub VNet has learnt the VPN routes through its BGP peering with the ARS in the VPN Subnet.
      * In the OS of PAN, you will see the VPN GW as the next hop, but from Azure's perspective, you cannot have:
        + next hop to ARS in the VPN VNet. ARS is just a route reflector.
        + Next hop to VPN GW IP. A UDR cannot point to the VPN GW.

VPN GW does not have an IP. It just builds tunnels. You cannot point to the VPN GW. Its not like a NVA.

One option is to set "use a remote GW" on the peering between the Hub and the Spoke/VPN peering connection.

Because we have Gateways in the Hub and the Spoke VNets, you cannot set "use remote GW" on the peering between the Hub and Spoke.

* + That something to point the return traffic is the NVA/PAN in the VPN VNet.
    - Can use Az FW if you don't want to worry about deploying HA pairs and use Az FW more for routing than inspecting.
    - Or can be any flavor of NVA like PAN
  + On the egress interface of the Palo in the Hub VNet, you need a UDR that says:
    - If you want to go to the VPN summary routes, next hop go through the FW in the VPN VNet.
    - UDR: on palo egress send to AzFW back to client
    - The FW in the VPN VNet is already leaning the VPN routes from the VPN GW
  + Return traffic from the client/customers hits VPN GW:
    - The VPN GW is learning (through the ARS in the VPN VNet) that next hop for AVS is the Palo in the Hub VNet because Palo in the Hub VNet is advertising AVS routes to the ARS in the VPN VNet
    - But you cannot point the return traffic hitting the VPN GW to the ARS (in the VPN VNet) to Palos in the hub. Because return traffic is going through the FW in the VPN VNet
    - Hence you need a UDR on the VPN Gateway subnet: AVS ---> Az FW
      * UDR says to go to AVS (ie AVS summary routes), go to the FW in the VPN VNet first.
      * The FW in the VPN VNet is already leaning the routes to get to AVS because the ARS in the VPN VNet is advertising that the next hop is Palo (in the Hub VNet). Hence the traffic goes to the Palo ILB. Then to AVS.

### Diagram:

A diagram of a network

AI-generated content may be incorrect.

### Pros:

* More control over the routes received by the clients/customers

### Cons:

* Complex routing setup
* Requires a duplication of NVA/Palo & ARS in the VPN Subnet

## vWAN based Solution:

### Solution Description:

* Needs Palo inside the VHub
  + Today you cannot have Palo in a spoke
    - If Palo were in a spoke, you can't transit from ER GW to VPN GW via Palo in the spoke in vWAN
  + Requires Palo SaaS licensing different from Palo on VM licensing
    - Not all features of Palo on VM are supported in the Palo SaaS license.
* Simpler design because you don't need UDRs by default:
  + Traffic: AVS -> to anywhere (say Future spokes) that need to be inspected don't need a UDR.
* The issue of the VPN clients/customers learning all routes in AVS and on-prem can be resolved with routeMaps in vWAN.
  + With route Maps, the connection properties of a specific client's VPN tunnel, can specify to drop the on-prem routes and only leave AVS routes.
  + RouteMap option to drop routes on a VPN tunnel does not exist in Hub and Spoke. The RouteMap option is only available in vWAN.
  + With Hub and Spoke the only option to control the on-prem and AVS routes advertised to clients/customers was to setup a Transit VPN GW VNet (as shown above) to control routes being sent by the Palo in the hub VNet (above) to the ARS in the VPN VNet to send to the VPN GW and clients/customers as shown above.



### Architecture Diagram:

A diagram of a network

AI-generated content may be incorrect.

### Pros:

* Makes the design easier

### Cons:

* Requires Palo SaaS licensing different from Palo on VM licensing
  + Not all features of Palo on VM are supported in the Palo SaaS license.

## Single Vnet (Hub and Spoke) with Palo VPN GW Solution:

### Solutions Details:

* Leverages the Palos to terminating the client/customer VPN tunnels
* If we want to terminate the VPN on the Palos, we can use their static public IPs and bypass the public Load Balancer for IPSec packets.
  + The customer side would have an active-standby configuration (like on Cisco ASA) where you can put multiple peer IPs.
* First deployment of Eagle in Azure leveraged Active-Active Palos peered to ARS. ARS handled all the VPN networks. The customer could determine which Palo they peered to, based off the learnt routes the Palo would advertise to ARS. The VMs would know which Palo has the active tunnel because it would re-advertise that to ARS.
  + The downside was that it was that tunnel building was co-managed. When customers has to build out tunnels to both Palo 1 and Palo 2, but customers failed to build out tunnels on Palo 2. Hence it did not work out in practice.
    - This problem can be address if the client configuration to Palo 1 and Palo 2 were built out using automation.
  + With Cisco ASA, when you can set peer IP1 IP2, makes the client will try IP1, and if it can't it will try IP2.
  + You have to set the secondary tunnel as passive mode because if both tunnels are down and the VM (onprem or AVS) initiates a reach out connection to the customer (like for printing with Bistrack), the VM connection enters passive mode so the customer has to bring up the tunnel by initiating traffic.
    - To keep the tunnel up, customers will have to trigger on their side by setting up IP SLA.
    - Customers may not have the technical ability to bring up the secondary tunnel or do IP SLA.
    - IN AWS, you can download the VPN config which sets up the config such that you can trigger the IP SLA from the client side.
    - P21 has 45 customers that need VPN, not using the web version. But even if using web, they could still use VPN for printing.
    - Customers could build a Cron job on a Linux machine with a loop that pings the other side.
  + Why not clients have tunnels up to both primary and secondary Palos and setup Local Preference?
    - Tunnels are not BGP tunnels, tunnels are static tunnels that are policy based.
  + We have to assume that the client at a minimum knows how to setup a tunnel, but not know how to manage it beyond dynamic routing etc.

### Architecture diagram:

A diagram of a computer network

AI-generated content may be incorrect.

### Pros:

* Removes the complexity introduced by the clients/customers receiving all on-prem and AVS routes

### Cons:

* Redundancy and resiliency goes down because of no failover as the Palo's cannot do VPN when frontended by a Load Blancer for redundancy and failover. Microsoft LB does not support IPSec packets required for the VPN tunnels to Palos.

# End-State Pattern Selection Criteria:

* Its not favorable for the client to learn all the on-prem route which introduces complexity, but can we compromise and reduce complexity by letting the clients know about the on-prem routes?
* The return traffic from the clients is going to Palo, so you can drop connections there as option.
  + Client Traffic -> VPN GW in Hub -> ILB -> Palo -> ER GW -> On-prem. So you can drop in the Palos if destined on-prem.
* Per Andrew: best solution option is to eliminate the Azure VPN GW based solution and ternminate the VPN tunnels off Palo.
  + This allows for manipulate the traffic because there is overlapping client network tunnels that Palo can handle, but Azure VPN GW has some limitations.
  + Can do NAT on Azure VPN GW, but you can setup the Palos such the customer doesn't have to adjust their side.
* The way Eagle is set up today with ARS and the VPNs will work if executed properly.
* Make sure to drop 0/0 onprem
  + because 0/0 will leak via 2 paths:
    - ER GW -> MSEE -> OnPrem
    - ER GW -> AVS DMSEE -> MSEE -> OnPrem
  + Whenever we peer on the ER, we'll apply route maps (on on-prem routers) for inbound and outbound.
  + The default route
    - will not go: on-prem -> hub in Azure
    - Will go: ER GW in Azure -> on-prem
* Overlapping IP considerations:
  + Some apps have overlapping IP addresses. P21 does not have overlapping IP addresses.

# Phased deployment to End State:

## Phase 1: Extend Layer 2

### Solution Description:

* All the VMs move to AVS, while networking remains on-prem
* AVS will route Internet traffic through on-prem while migrating:
  + AVS -> DMSEE -> MSEE -> On-Prem -> internet
* The VNet in Azure will not exist initially, AVS will be just someone else's datacenter.
* All traffic still flows back through on-prem.
* On-prem will retain all the network equipment, only VMs are migrating from on-prem to AVS.
  + On-Prem will retain all the network equipment: core switches, next layer switches, routes, FW, Internet WAN circuits etc.

### Diagram:



### CutOver:



## Phase 2: Move VPN Tunnels, one at a time from on-prem to Azure

### Solution Description:

* Build out the hub in Azure
* Move the VPN Tunnels, one at a time from on-prem to Azure
* The VPN Tunnel will be to a different IP to the Palos in the Hub
* The VMs in AVS, need to be able to talk to the VPN Tunnels and client instead of using the Global Reach to on-prem out to on-prem VPN.
* There are 1000s of tunnels that will need to be moved one at a time.
* After a customer's VPN tunnel cuts over from on-prem to Azure, they should have routes to get to AVS. And AVS should have a single route to get back to the customer's VPN tunnel.

### Diagram:

### Cutover:

* After a customer's VPN tunnel cuts over from on-prem to Azure, they should have routes to get to AVS.

## Phase 3: Move Default GW to AVS:

### Solution Description:

* Moves some traffic by moving the default GW to AVS, but internet traffic flows through on-prem
* When you cut over default GW to AVS, you are cutting over everything. This will take a long time because of the VPN tunnels.
  + Cannot change the VPN peer IPs for 1000s of customers

### Diagram:

### Cutover:

* When you cut over default GW to AVS, you are cutting over everything.

## Phase 4: Move Edge capabilities to Azure:

### Solution Description:

* Move VPN Traffic one at a time from

### Diagram:



### Cutover: