WHITE PAPER

Using Fortinet and AWS To Provide Fault-tolerant RingCentral Access

Using Classical Features of Fortinet Firewalls

Background

Fortinet FortiGate next-generation firewalls are used as the firewall standard by many companies globally. Many of these companies have a distributed global presence and would benefit from the RingCentral unified communications solution. Communications services, particularly their voice service, is critical to their operation and they require high levels of fault tolerance.

Please note that this solution, while architected for RingCentral traffic, can be altered to support many other services so long as that service is based upon delivery of traffic to fixed public IP addresses or address blocks.

RingCentral has multiple data centers globally where Internet peering/ingress/egress occurs for connection to their services. Many of these data centers are adjacent to Amazon AWS data centers. Implementation of FortiGate VM units in these AWS data centers can be utilized to achieve fault tolerance and depending on the FortiOS firmware version chosen, to compensate for degraded connections exhibiting packet loss.

This document describes the implementation of a highly fault tolerant active/backup VPN overlay network to support transport of a customer's voice traffic. This design is based upon classical FortiGate features and provides support for customer sites with dual WAN links. It offers:

- Failover from a primary to a backup WAN link at the same RingCentral data center is extremely fast and almost transparent. Testing shows approximately 1 second of audio silence on failure of the ISP primary circuit. Transition back to the primary ISP circuit on recovery is seamless. SIP registrations are unaffected by the transition.
- Failover from a primary RingCentral data center to a secondary RingCentral data center occurs in less than 1 second but the change in NAT source address results in all active connections and registrations being dropped/reset. Polycom phones can dial out immediately after the transition. Active phone calls will drop after 10-20 seconds of silence and may be redialed immediately.
- Packet loss remediation – Calls maintain toll quality even with over 15% packet loss on the active WAN link due to Fortinet's FEC over VPN feature.
    - *Requires Fortinet firmware version 6.2.3 or later*.
    - Even a site with a single WAN link can be set up so that it gains the packet loss remediation feature and failover from the primary RingCentral data center to the secondary RingCentral data center.

This configuration utilizes the BFD protocol to monitor integrity of the VPN pathways. The iBGP routing protocol is used to reroute traffic on failure of the active VPN pathway. During testing, pathway failure was detected and rerouting/reconvergence occurred within a range of 0.5-1.25 seconds.

Warning: The OSPF routing protocol does not provide a seamless failover, **do not attempt to use it**. During the period of route transition RingCentral traffic will be sent over the default route directly through a WAN link and its associated NAT subsystem. The firewall's NAT subsystem will lock that connection's traffic to the WAN link regardless of subsequent routing updates. RingCentral traffic must never go out the WAN link unless there is no other option, as in both AWS FortiGate appliances becoming unreachable.

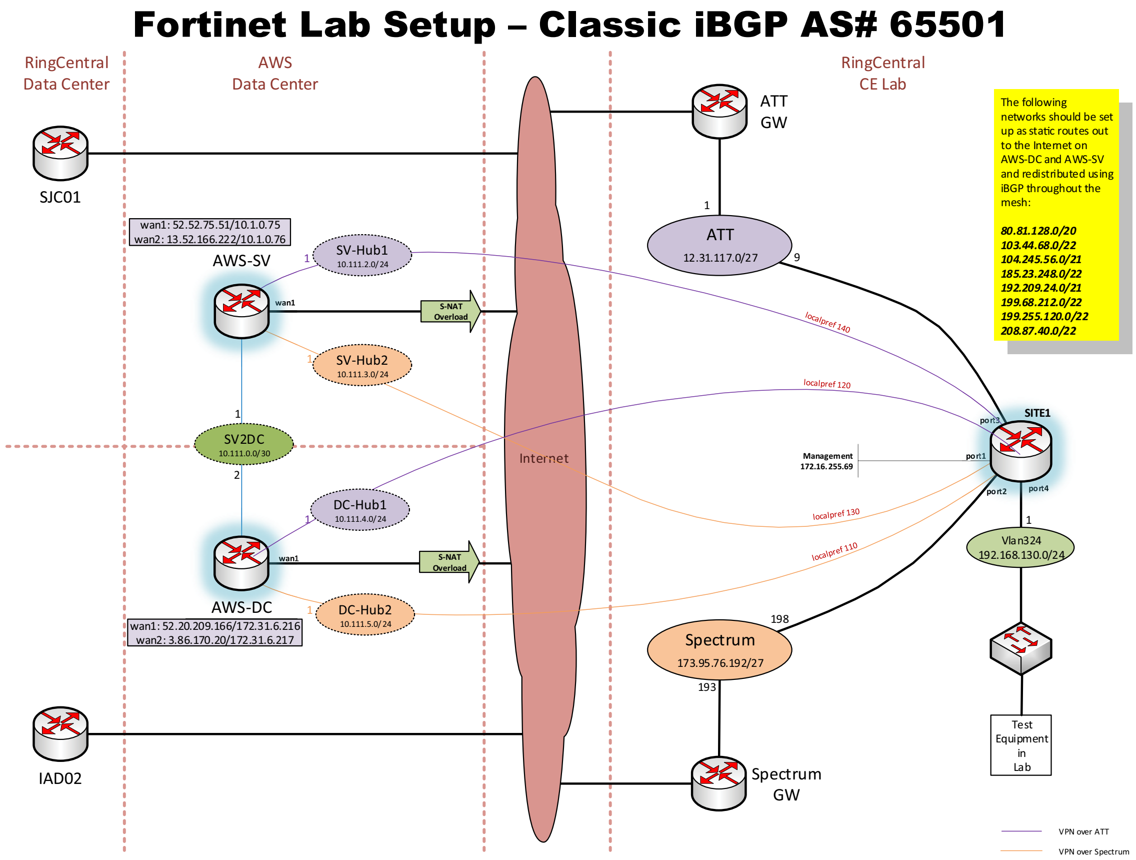
The configuration architecture shown in this paper supports over 240 customer sites and can be easily expanded within the architectural/performance limits of the FortiGate firmware.

Assumptions

- FortiGate appliances must be running 6.2.x or higher firmware to support some of the features utilized in this configuration. Prior versions may be usable with corrections for changed command syntax. Usable packet loss remediation (FEC) is not available prior to version 6.2.2. Testing was performed using firmware versions 6.2.2 and 6.2.3. Functionality has been confirmed with version 6.4.0.
- The customer will turn up Fortinet virtual FortiGate appliances in multiple AWS locations. Each customer site will select two of the selected AWS locations for connections. One of these AWS locations, usually the closest, should be designated as the primary hub site. In North America those locations should be **Northern Virginia (Ashburn)** and **Northern California (Silicon Valley/San Jose)**. *[For the purposes of this document these two locations will be used and referred to as AWS-DC and AWS-SV, respectively.]* Fortinet System Engineers should be consulted to determine proper sizing of the appliances based upon your projected simultaneous call/video-conference volume.

  Note that you may elect to use a single AWS site. The use of two sites simply provides additional levels of redundancy.
- This configuration does not support the RingCentral Engage product at this time. It will be added in a future revision.

Architecture



The following networks should be set up as static routes out to the Internet on AWS-DC and AWS-SV and redistributed using iBGP throughout the mesh:

80.81.128.0/20
103.44.68.0/22
104.245.56.0/21
185.23.248.0/22
192.209.24.0/21
199.68.212.0/22
199.255.120.0/22
208.87.40.0/22

There are two elements to the architecture, AWS hub sites and customer sites. Please note that while this document describes fault tolerance for RingCentral services, it may easily be expanded to support other services by alteration of the routing configuration. These elements exchange routing information using iBGP. ***(See earlier warning note regarding issue with OSPF.)***

Note that all iBGP routing is performed across the ***VPN TUNNEL*** interfaces, not the physical interfaces.

Five (5) subnets must be allocated from the customer's interior address space to support this configuration (note that the 4 *pools* must be the same size):

Use	Size	Description
InterHub	/30	Routing link between the AWS-SV and AWS-DC Hub sites.
AWS-SV-Hub1	/24 or larger	Dialup VPN addressing pool to support site ISP #1 on AWS-SV.
AWS-SV-Hub2	/24 or larger	Dialup VPN addressing pool to support site ISP #2 on AWS-SV.
AWS-DC-Hub1	/24 or larger	Dialup VPN addressing pool to support site ISP #1 on AWS-DC.
AWS-DC-Hub2	/24 or larger	Dialup VPN addressing pool to support site ISP #2 on AWS-DC.

Customer Sites

Each site with dual WAN links will configure 4 VPN tunnels as follows:

To Hub	ISP	Interface	Gateway	Tunnel IP
AWS-SV-Hub1	ISP1	wan1	ISP1 Gateway	assigned by hub
AWS-SV-Hub2	ISP2	wan2	ISP2 Gateway	assigned by hub
AWS-DC-Hub1	ISP1	wan1	ISP1 Gateway	assigned by hub
AWS-DC-Hub2	ISP2	wan2	ISP2 Gateway	assigned by hub

Each tunnel supports BFD outage detection and may support bi-directional FEC error correction. Tunnel IP endpoint addresses on the site side are dynamic and are provided by the hub from an mash specific address pool. Control of tunnel priority is implemented by each site's configuration.

AWS Hub Sites

Two AWS Hub sites must be established, each near a RingCentral Data Center, preferably geographically diverse RingCentral Data Centers. Each AWS virtual FortiGate appliance must be configured with a primary and a secondary IP address on the single WAN port. Both addresses must be mapped to discrete Elastic IP public addresses. Two mesh VPN ***Hub*** services will be established, each configured to utilize one of these addresses. Each of the two different ISP connections at each customer site will connect to these ***Hub*** services as detailed in the previous table. *[These mesh VPN Hubs are set up like dialup/ADVPN VPN services.]*

The security group inbound rules for each AWS site must include the following additional rules to allow Fortinet VPN setup/traffic:

Type	Protocol	PortRange	Source	
Custom Protocol	ESP (50)	all	Custom	0.0.0.0/0
Custom UDP	UDP	500	Custom	0.0.0.0/0
Custom UDP	UDP	4500	Custom	0.0.0.0/0

AWS/Site Interactions

Every customer site will define the 4 VPNs such that they have a specifically defined egress/gateway. This will ensure that the connection of AWS-SV Hub1 goes out ISP1's gateway, AWS-SV Hub2 goes out ISP2's gateway, AWS-DC Hub1 goes out ISP1's gateway, and AWS-DC-Hub2 goes out ISP2's gateway.

Each AWS Data Center site will define 8 static routes encompassing the RingCentral assigned IP address space. These static routes will be advertised out iBGP to the customer sites. Link monitoring is utilized to ensure that the routes are automatically disabled and withdrawn from iBGP if they fail at one or both AWS sites. Preference of these routes is controlled by the site configuration and will be described later.

Each tunnel supports bi-directional FEC error correction and BFD outage detection. Tunnel IP endpoint addresses for customer site client VPNs are dynamic and are provided by the hub.

Tunnel Preference Configuration

Control of tunnel priority is implemented in and controlled solely by each ***site's*** configuration. Each VPN tunnel interface is assigned an iBGP localpref value. The shortest route with the highest iBGP localpref value will be utilized for corresponding traffic.

A route-map is applied to all routes being advertised from the Hubs to the site. This inbound route-map assigns an iBGP localpref value to all routes advertised by that Hub.

A corresponding outbound route-map is applied to all routes being advertised from the site to each Hub.

In this example configuration, we utilize iBGP localpref values as follows:

iBGP localpref value	Meaning
140	**PrimaryPath** – Most favored pathway, use this ISP link to this AWS Data Center. Usually this will be your favored ISP going to the closer (primary) AWS Data Center.
130	**SecondaryPath** – Next most favored pathway. Usually this will be your backup ISP going to the closer (primary) AWS Data Center. This usually indicates that your favored ISP link is down.
120	**TertiaryPath** – Use this pathway if no pathway to the closer (primary) AWS Data Center is available. It will usually use your favored ISP.
110	**QuaternaryPath** – Use this pathway if no pathway to the closer (primary) AWS Data Center is available and your favored ISP link is down.
100	This is the default iBGP localpref value and should never be seen on any RingCentral routes seen in the site iBGP routing table on the Customer site devices. It will be seen in the iBGP routing table of the AWS Data Center machines.

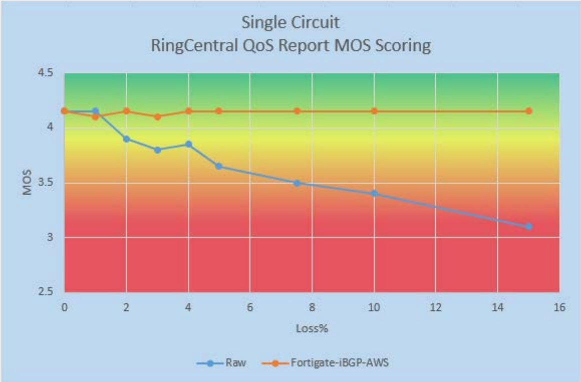
Test Results

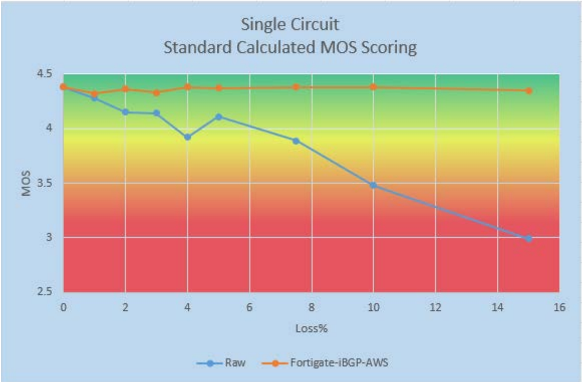
These test results assume that the Primary and Secondary routes both go to the same Data Center and the Tertiary and Quaternary routes both go to the alternate Data Center.

Failure of the Primary route (most likely ISP failure) results in a rapid transition to the Secondary route using the alternate ISP. Outage time is between 0.75 and 1.25 seconds. Failback from Secondary to Primary route may result in a momentary, almost unnoticeable 'blip'. Any ongoing telephone calls will remain intact and suffer only a 0.75 - 1.25 second period of silence before the audio resumes. The source address of the NATted traffic does not change.

Failure of the actual Data Center (both Primary and Secondary routes) results in a rapid transition to the Tertiary route using the alternate ISP. Outage time is between 0.75 and 1.25 seconds. Failback to the Primary Data Center may result in a momentary, almost unnoticeable outage. Any ongoing telephone calls will fail and suffer a 20 - 30 second period of silence before the phone times out, drops the call, and re-registers. All phones lose registration and re-register within 30 seconds. The source address of the NATted traffic changes.

The FEC option dramatically improves voice quality over lossy (or lousy) ISP connections.





The line graph on the left represents the MOS voice quality of a 60 second test call as reported by RingCentral's media server equipment and the telephone set. The line graph on the right represents a standardized MOS quality score as determined mathematically using the standard MOS formula. The orange lines represent test results from a call going through the FortiGate units with bidirectional FEC enabled. The blue lines represent a standard phone call using the Internet.

Configurations

VDOM Configuration Notes/Caveats

FortiGate firewalls provide for a ‘virtual domain’ (VDOM) functionality which allows creation of multiple independent logical firewalls on a single platform. This feature, when enabled, changes the configuration syntax slightly.

There are two categories of settings, global settings which apply to all VDOM instances and VDOM specific settings. If you have enabled VDOM features, you will need to enter the extra configuration commands shown in **GREEN** color. Do not enter these confirmation commands if you have not enabled VDOM features. You will also need to change the “root” VDOM name to the name of the VDOM in which you want to enter the configuration. (The name “root” refers to the default VDOM.)

Warning: The configuration BACKUP command outputs a script that is read by the configuration RESTORE command. The BACKUP function reverses the order of the ‘`set tos-mask`’ and ‘set tos’ configuration elements. The ‘`set tos-mask`’ **MUST** come first, else the ‘`set tos`’ element will be silently ignored. These elements are found in many places in the configuration and are critical to the operation of this system. You must edit the BACKUP files and correct the order prior to performing any RESTORE operations. Failure to do so will severely impact call quality as it will render QoS inoperative. This is the correct order:

```
set tos-mask 0xfc

set tos 0xb8
```

Special Note about FEC Settings on Hardware FortiGate Units (Only on firmware versions that support FEC)

Please note that the FEC options will be ***silently*** ignored on many ***hardware*** based FortiGate appliances unless NPU offload is enabled on each of the 4 Hub Tunnel Phase1 Interfaces. The FortiOS GUI interface will not notify you of this issue and will silently ignore your settings. The commands to enable NPU offload are shown using green highlighted type in the ‘`config vpn ipsec phase1-interface`’ sections. This is not necessary and may be omitted when configuring virtual instances, thus this command will not be needed on AWS instances, but is shown in the Hub configurations just for those that wish to modify this setup using Customer locations with hardware appliances.

AWS-SV (Silicon Valley <N. California> Hub site)

Global Settings

There are several ‘global’ settings that should be configured. The parameter ‘`reset-sessionless-tcp`’ is used to force transmission of a TCP RST (reset) packet when a packet destined for a non-existent session is received. This will force a phone running the TCP/TLS variants of SIP to immediately reregister if the connection has been dropped.

Additionally, the FortiGate unit defaults to QoS using the ToS settings. You must switch it to use DSCP and to default all unclassified traffic to the low-priority queue.

```
config global
config system global
 # Automatically return TCP RST packet on transmission to invalid session
 set reset-sessionless-tcp enable
 # Set system to use DSCP in lieu of TOS
 set traffic-priority dscp
 # Traffic defaults to low priority unless overridden
 set traffic-priority-level low
end
# Configure DSCP to priority mapping table
config system dscp-based-priority
 edit 46
  set ds 46
  set priority high
 next
```

```
  edit 34
   set ds 34
   set priority medium
  next
  edit 26
   set ds 26
   set priority high
  next
end
end
```

VDOM Specific Settings

```
config vdom
edit “root”
config system settings
 # Enable BFD for rapid outage detection
 set bfd enable
 set bfd-desired-min-tx 300
 set bfd-required-min-rx 600
 # Set up logging of VPN data
 set vpn-stats-log ipsec
 set vpn-stats-period 60
 # Set up rapid detection of VPN IKE outages
 set ike-quick-crash-detect enable
end
# Configure Physical ports for WAN links
config system interface
 # ISP #1 & ISP #2 are both serviced on port1 (wan1) by using a secondary IP
 # IP address from the same subnet as the primary IP address
 #
 # ISP #1 uses:
 # 10.1.0.75/24 - AWS assigned subnet address for this interface
 # 52.52.75.51 – AWS Elastic IP address associated with 10.1.0.75
 #
 # ISP #2 uses:
 # 10.1.0.76/24 - AWS secondary address for this interface
 # 13.52.166.222 – AWS Elastic IP address associated with 10.1.0.76
 #
 edit “port1”
  # Primary IP address
  set ip 10.1.0.75 255.255.255.0
  set allowaccess ping https ssh
  set bfd enable
  set bfd-desired-min-tx 300
  set bfd-required-min-rx 600
  set type physical
  set inbandwidth 1000000
  set outbandwidth 1000000
  set alias “wan1”
  set role wan
  set mtu-override enable
  set mtu 9001
  # Secondary IP address
  set secondary-IP enable
  config secondaryip
   edit 0
    set ip 10.1.0.76 255.255.255.0
    set allowaccess ping https ssh
   next
  end
 next
end
# Configure VPN Interfaces
config vpn ipsec phase1-interface
 # PtP link between Hub sites
 edit “VPN-SV2DC”
  set interface “port1”
  # Use the primary IP address
  set local-gw 10.1.0.75
  set ike-version 2
  set peertype any
  set net-device enable
  set proposal aes256-sha512
  set dhgrp 32
  # localid not used, but convenient for troubleshooting
  set localid “SV-Hub-IH”
  set dpd on-idle
  #
  #########################################################################
  # ON CERTAIN HARDWARE FORTIGATES MUST ENABLE NPU OFFLOAD TO ENABLE FEC #
  #########################################################################
  #
  set npu-offload enable
  # Enable FEC in both directions
  set fec-egress enable
  set fec-ingress enable
  set psksecret “InterHubPtP”
  set dpd-retrycount 2
  set dpd-retryinterval 5
  set remote-gw 52.20.209.166
 next
 # Establish Dialup (Mesh) VPN server for SV-Hub1
 # Note the use of ‘set local-gw x.x.x.x’ to override the routing table. This
 # is MANDATORY!!!!
 edit “VPN-SV-Hub1”
  set type dynamic
  set interface “port1”
  # Use the primary IP address
  set local-gw 10.1.0.75
  set mode aggressive
  set peertype one
  set net-device disable
  set mode-cfg enable
  set proposal aes256-sha512
  set add-route disable
  # Outbound connections will be identified as being originated by ‘SV-Hub1’
  set localid “SV-Hub1”
  set dpd on-idle
  set dhgrp 32
  set auto-discovery-sender enable
  #
  #########################################################################
  # ON CERTAIN HARDWARE FORTIGATES MUST ENABLE NPU OFFLOAD TO ENABLE FEC #
  #########################################################################
  #
  set npu-offload enable
  # Enable FEC in both directions
  set fec-egress enable
  set fec-ingress enable
  # Inbound connections must have a localid of ‘To-SV-Hub1’
  set peerid “To-SV-Hub1”
  set tunnel-search nexthop
  # Define the IP Pool block used to allocate addressing to customer site VPN
  # clients. Note that this configuration allows up to 245 clients – this may
  # be expanded by using a larger netmask / allocation.
  set ipv4-start-ip 10.111.2.10
  set ipv4-end-ip 10.111.2.254
  set ipv4-netmask 255.255.255.0
  set unity-support disable
  set psksecret “PSK-SV-Wan1”
  set dpd-retrycount 2
  set dpd-retryinterval 5
 next
 # Establish Dialup (Mesh) VPN server for SV-Hub2
 # Note the use of ‘set local-gw x.x.x.x’ to override the routing table. This
 # is MANDATORY!!!!
 edit “VPN-SV-Hub2”
  set type dynamic
  set interface “port1”
  # Use the secondary IP address
  set local-gw 10.1.0.76
  set mode aggressive
  set peertype one
  set net-device disable
  set mode-cfg enable
  set proposal aes256-sha512
  set add-route disable
  # Outbound connections will be identified as being originated by ‘SV-Hub2’
  set localid “SV-Hub2”
  set dpd on-idle
  set dhgrp 32
  set auto-discovery-sender enable
  #
  #########################################################################
  # ON CERTAIN HARDWARE FORTIGATES MUST ENABLE NPU OFFLOAD TO ENABLE FEC #
  #########################################################################
  #
  set npu-offload enable
```

```
  # Enable FEC in both directions
  set fec-egress enable
  set fec-ingress enable
  # Inbound connections must have a localid of ‘To-SV-Hub2’
  set peerid “To-SV-Hub2”
  set tunnel-search nexthop
  # Define the IP Pool block used to allocate addressing to customer site VPN
  # clients. Note that this configuration allows up to 245 clients – this may
  # be expanded by using a larger netmask / allocation.
  set ipv4-start-ip 10.111.3.10
  set ipv4-end-ip 10.111.3.254
  set ipv4-netmask 255.255.255.0
  set unity-support disable
  set psksecret “PSK-SV-Wan2”
  set dpd-retrycount 2
  set dpd-retryinterval 5
 next
end
config vpn ipsec phase2-interface
 edit “VPN-SV2DC”
  set phase1name “VPN-SV2DC”
  set proposal aes256gcm
  set dhgrp 32
  set auto-negotiate enable
 next
 edit “VPN-SV-Hub1”
  set phase1name “VPN-SV-Hub1”
  set proposal aes256gcm
  set dhgrp 32
  set keepalive enable
 next
 edit “VPN-SV-Hub2”
  set phase1name “VPN-SV-Hub2”
  set proposal aes256gcm
  set dhgrp 32
  set keepalive enable
 next
end
# Complete configuration of VPN interfaces adding addresses and BFD
config system interface
 # HUB to HUB tunnel
 edit “VPN-SV2DC”
  set ip 10.111.0.1 255.255.255.255
  set allowaccess ping https ssh
  set bfd enable
  set bfd-desired-min-tx 300
  set bfd-required-min-rx 600
  set remote-ip 10.111.0.2 255.255.255.255
 next
 # Dialup Mesh VPN server #1
 edit “VPN-SV-Hub1”
  # Address should be .1 from same block as Phase1 IP Pool.
  # Netmask MUST be /32.
  set ip 10.111.2.1 255.255.255.255
  set allowaccess ping https ssh
  set bfd enable
  set bfd-desired-min-tx 300
  set bfd-required-min-rx 600
  # Dummy address should be .2 from same block as Phase1 IP Pool.
  # Netmask MUST be identical to Phase1 IP Pool netmask.
  set remote-ip 10.111.2.2 255.255.255.0
 next
 # Dialup Mesh VPN server #2
 edit “VPN-SV-Hub2”
  # Address should be .1 from same block as Phase1 IP Pool.
  # Netmask MUST be /32.
  set ip 10.111.3.1 255.255.255.255
  set allowaccess ping https ssh
  set bfd enable
  set bfd-desired-min-tx 300
  set bfd-required-min-rx 600
  # Dummy address should be .2 from same block as Phase1 IP Pool.
  # Netmask MUST be identical to Phase1 IP Pool netmask.
  set remote-ip 10.111.3.2 255.255.255.0
 next
end
# Group interfaces into zones for ease of policy creation
config system zone
 edit “ZN_Mesh”
  set intrazone allow
  set interface “VPN-SV2DC” “VPN-SV-Hub1” “VPN-SV-Hub2”
 next
 edit “ZN_Wan”
  set intrazone allow
  set interface “port1”
 next
end
# Establish traffic shapers for each category of traffic
#
# NOTE: These values must be adjusted to match reality. They are expressed in
#       Kilobits per second (kbps).
config firewall shaper traffic-shaper
 # Shaping to apply to tunnels.
 #
 # This sample allows (guaranteed/allowed):
 #  100000/100000Kbps of voice RealTime traffic (1000 calls)
 #  300000/300000Kbps of video RealTime traffic (450 video calls)
 #  1000/3000Kbps of signaling traffic
 #
 edit “TS_DSCP_EF”
  set guaranteed-bandwidth 100000
  set maximum-bandwidth 100000
 next
 edit “TS_DSCP_AF41”
  set guaranteed-bandwidth 300000
  set maximum-bandwidth 300000
  set priority medium
 next
 edit “TS_DSCP_AF31”
  set guaranteed-bandwidth 1000
  set maximum-bandwidth 3000
 next
 # Shaping to apply to WAN ports.
 #
 # Note that the tunnels flow through the WAN ports, so the minimum values
 # used here should be a minimum of the amount for the tunnels plus overhead.
 # Higher values may be used if non-RingCentral traffic is required to be
 # QoS matched
 edit “TS_W_DSCP_EF”
  set guaranteed-bandwidth 110000
  set maximum-bandwidth 110000
 next
 edit “TS_W_DSCP_AF41”
  set guaranteed-bandwidth 330000
  set maximum-bandwidth 330000
  set priority medium
 next
 edit “TS_W_DSCP_AF31”
  set guaranteed-bandwidth 1100
  set maximum-bandwidth 3300
 next
end
#---------------------------------------------------------------------------------------
# Set up the firewall traffic shaping policy. It overrides all other policies for
# QoS and traffic shaping. Force remarking of return traffic with diffserv-reverse.
config firewall shaping-policy
 # The first 4 rules are for traffic egressing via the MESH VPNs.
 edit 0
  set name “TSP_H_DSCP_EF”
  set service “ALL”
  set dstintf “ZN_Mesh”
  set tos-mask 0xfc
  set tos 0xb8
  set traffic-shaper “TS_DSCP_EF”
  set traffic-shaper-reverse “TS_DSCP_EF”
  set diffserv-reverse enable
  set srcaddr “all”
  set dstaddr “all”
  set diffservcode-rev 101110
 next
 edit 0
  set name “TSP_H_DSCP_AF41”
  set service “ALL”
  set dstintf “ZN_Mesh”
  set tos-mask 0xfc
  set tos 0x88
  set traffic-shaper “TS_DSCP_AF41”
  set traffic-shaper-reverse “TS_DSCP_AF41”
  set diffserv-reverse enable
  set srcaddr “all”
  set dstaddr "all"
  set diffservcode-rev 100010
 next
 edit 0
  set name "TSP_H_DSCP_AF31"
  set service "ALL"
  set dstintf "ZN_Mesh"
  set tos-mask 0xfc
  set tos 0x68
  set traffic-shaper "TS_DSCP_AF31"
  set traffic-shaper-reverse "TS_DSCP_AF31"
  set diffserv-reverse enable
  set srcaddr "all"
  set dstaddr "all"
  set diffservcode-rev 011010
 next
 edit 0
  set name "TSP_H_DSCP_CS3"
  set service "ALL"
  set dstintf "ZN_Mesh"
  set tos-mask 0xfc
  set tos 0x60
  set traffic-shaper "TS_DSCP_AF31"
  set traffic-shaper-reverse "TS_DSCP_AF31"
  set diffserv-forward enable
  set diffserv-reverse enable
  set srcaddr "all"
  set dstaddr "all"
  set diffservcode-forward 011010
  set diffservcode-rev 011010
 next
 #
 # The last 4 rules are for traffic egressing via the last-ditch WAN pathway.
 #
 # Note that the tunnels to the AWS sites egress via these rules… This requires
 # that the values of the traffic shapers must be a minimum of what is allowed
 # via the tunnel rules above.
 #
 edit 0
  set name "TSP_W_DSCP_EF"
  set service "ALL"
  set dstintf "ZN_Wan"
  set tos-mask 0xfc
  set tos 0xb8
  set traffic-shaper "TS_W_DSCP_EF"
  set traffic-shaper-reverse "TS_W_DSCP_EF"
  set diffserv-reverse enable
  set srcaddr "all"
  set dstaddr "all"
  set diffservcode-rev 101110
 next
 edit 0
  set name "TSP_W_DSCP_AF41"
  set service "ALL"
  set dstintf "ZN_Wan"
```

```
  set tos-mask 0xfc
  set tos 0x88
  set traffic-shaper “TS_W_DSCP_AF41”
  set traffic-shaper-reverse “TS_W_DSCP_AF41”
  set diffserv-reverse enable
  set srcaddr “all”
  set dstaddr “all”
  set diffservcode-rev 100010
 next
 edit 0
  set name “TSP_W_DSCP_AF31”
  set service “ALL”
  set dstintf “ZN_Wan”
  set tos-mask 0xfc
  set tos 0x68
  set traffic-shaper “TS_W_DSCP_AF31”
  set traffic-shaper-reverse “TS_W_DSCP_AF31”
  set diffserv-reverse enable
  set srcaddr “all”
  set dstaddr “all”
  set diffservcode-rev 011010
 next
 edit 0
  set name “TSP_W_DSCP_CS3”
  set service “ALL”
  set dstintf “ZN_Wan”
  set tos-mask 0xfc
  set tos 0x60
  set traffic-shaper “TS_W_DSCP_AF31”
  set traffic-shaper-reverse “TS_W_DSCP_AF31”
  set diffserv-forward enable
  set diffserv-reverse enable
  set srcaddr “all”
  set dstaddr “all”
  set diffservcode-forward 011010
  set diffservcode-rev 011010
 next
end
# Establish basic policies to allow mesh to mesh and mesh to wan traffic.
end
config firewall policy
 # Mesh to Mesh
 edit 0
  set name “POL_Hub2Hub”
  set srcintf “ZN_Mesh”
  set dstintf “ZN_Mesh”
  set srcaddr “all”
  set dstaddr “all”
  set action accept
  set schedule “always”
  set service “ALL”
 next
 # Mesh to Wan
 edit 0
  set name “POL_Hub2Wan”
  set srcintf “ZN_Mesh”
  set dstintf “ZN_Wan”
  set srcaddr “all”
  set dstaddr “all”
  set action accept
  set schedule “always”
  set service “ALL”
  set nat enable
 next
end
# Create access list to match all RingCentral public address space (8 CIDR blocks)
config router access-list
 edit “ACL-RC-All”
  config rule
   edit 0
    set prefix 66.81.240.0 255.255.240.0
    set exact-match enable
   next
   edit 0
    set prefix 80.81.128.0 255.255.240.0
    set exact-match enable
   next
   edit 0
    set prefix 103.44.68.0 255.255.252.0
    set exact-match enable
   next
   edit 0
    set prefix 104.245.56.0 255.255.248.0
    set exact-match enable
   next
   edit 0
    set prefix 185.23.248.0 255.255.252.0
    set exact-match enable
   next
   edit 0
    set prefix 192.209.24.0 255.255.248.0
    set exact-match enable
   next
   edit 0
    set prefix 199.68.212.0 255.255.252.0
    set exact-match enable
   next
   edit 0
    set prefix 199.255.120.0 255.255.252.0
    set exact-match enable
   next
   edit 0
    set prefix 208.87.40.0 255.255.252.0
    set exact-match enable
   next
  end
 next
end
# Configure route-maps needed for iBGP functionality.
```

```
config router route-map
 # Identify route as ingressing from SV-Hub1 (community 65501:1) and force
 # the next-hop ip address to the address of this tunnel.
 edit “RM-Out-Set-SV1”
  set comments “SV Hub1 VPN”
  config rule
   edit 0
    set set-community “65501:1”
    # Be sure to set the following address to the address of the
    # SV-Hub1 VPN tunnel.
    set set-ip-nexthop 10.111.2.1
   next
  end
 next
 # Identify traffic as ingressing from SV-Hub2 (community 65501:2) and force
 # the next-hop ip address to the address of this tunnel.
 edit “RM-Out-Set-SV2”
  set comments “SV Hub2 VPN”
  config rule
   edit 0
    set set-community “65501:2”
    # Be sure to set the following address to the address of the
    # SV-Hub2 VPN tunnel.
    set set-ip-nexthop 10.111.3.1
   next
  end
 next
 # Allow ONLY the 8 RingCentral CIDR blocks to redistribute from static to BGP routes
 edit “RM-Redis-Static-2-Bgp”
  config rule
   edit 0
    set match-ip-address “ACL-RC-All”
   next
  end
 next
end
# Create static routes and set up route monitoring.
config router static
 # Default route out port 1, never withdrawn.
 edit 0
  set gateway 10.1.0.1
  set device “port1”
  set link-monitor-exempt enable
 next
 # RingCentral CIDR blocks; disable and withdraw on failure of test.
 edit 0
  set dst 66.81.240.0 255.255.240.0
  set gateway 10.1.0.1
  set device “port1”
 next
 edit 0
  set dst 80.81.128.0 255.255.240.0
  set gateway 10.1.0.1
  set device “port1”
 next
 edit 0
  set dst 103.44.68.0 255.255.252.0
  set gateway 10.1.0.1
  set device “port1”
 next
 edit 0
  set dst 104.245.56.0 255.255.248.0
  set gateway 10.1.0.1
  set device “port1”
 next
 edit 0
  set dst 185.23.248.0 255.255.252.0
  set gateway 10.1.0.1
  set device “port1”
 next
 edit 0
  set dst 192.209.24.0 255.255.248.0
  set gateway 10.1.0.1
  set device “port1”
 next
 edit 0
  set dst 199.68.212.0 255.255.252.0
  set gateway 10.1.0.1
  set device “port1”
 next
 edit 0
  set dst 199.255.120.0 255.255.252.0
  set gateway 10.1.0.1
  set device “port1”
 next
 edit 0
  set dst 208.87.40.0 255.255.252.0
  set gateway 10.1.0.1
  set device “port1”
 next
end
# Create route monitor using 199.255.120.129 test address.
config system link-monitor
 edit “1”
  set srcintf “port1”
  set server “199.255.120.129”
  set interval 10000
  set failtime 3
  set recoverytime 2
 next
end
# Set up iBGP using private AS 65501.
config router bgp
 set as 65501
 set router-id 10.111.0.1
 config neighbor
  edit “10.111.0.2”
   set bfd enable
   set link-down-failover enable
   set next-hop-self enable
   set soft-reconfiguration enable
   set interface “VPN-SV2DC”
   set remote-as 65501
   set update-source “VPN-SV2DC”
  next
 end
 # Define a neighbor-group (template) for each dialup Hub member.
 config neighbor-group
  edit “clients-hub1”
   set bfd enable
   set link-down-failover enable
   set next-hop-self enable
   set soft-reconfiguration enable
   set interface “VPN-SV-Hub1”
   set remote-as 65501
   # Tag outbound routes with a community indicating this Hub and
   # force the nexthop to be the origin IP address of this Hub.
   set route-map-out “RM-Out-Set-SV1”
   set update-source “VPN-SV-Hub1”
   set route-reflector-client enable
  next
  edit “clients-hub2”
   set bfd enable
   set link-down-failover enable
   set next-hop-self enable
   set soft-reconfiguration enable
   set interface “VPN-SV-Hub2”
   set remote-as 65501
   # Tag outbound routes with a community indicating this Hub and
   # force the nexthop to be the origin IP address of this Hub.
   set route-map-out “RM-Out-Set-SV2”
   set update-source “VPN-SV-Hub2”
   set route-reflector-client enable
  next
 end
 # Define IP ranges for each Neighbor group. These must match the IP Pools
 # defined in the Phase1 definitions.
 config neighbor-range
  edit 0
   set prefix 10.111.2.0 255.255.255.0
   set neighbor-group “clients-hub1”
  next
  edit 0
   set prefix 10.111.3.0 255.255.255.0
   set neighbor-group “clients-hub2”
  next
 end
 # Redistribute static routes matching the route-map to all clients.
 config redistribute “static”
  set status enable
  set route-map “RM-Redis-Static-2-Bgp”
 end
end
end
```

AWS-DC (Ashburn <N. Virginia> Hub site)

Global Settings

The description of global settings from the AWS-SV configuration also apply here.

```
config global
config system global
 # Automatically return TCP RST packet on transmission to invalid session
 set reset-sessionless-tcp enable
 # Set system to use DSCP in lieu of TOS
 set traffic-priority dscp
 # Traffic defaults to low priority unless overridden
 set traffic-priority-level low
end
# Configure DSCP to priority mapping table
config system dscp-based-priority
 edit 46
  set ds 46
  set priority high
 next
 edit 34
  set ds 34
  set priority medium
 next
 edit 26
  set ds 26
  set priority high
 next
end
end
```

VDOM Specific Settings

```
config vdom
edit “root”
config system settings
 # Enable BFD for rapid outage detection
 set bfd enable
 set bfd-desired-min-tx 300
 set bfd-required-min-rx 600
 # Set up logging of VPN data
 set vpn-stats-log ipsec
 set vpn-stats-period 60
 # Set up rapid detection of VPN IKE outages
 set ike-quick-crash-detect enable
end
# Configure Physical ports for WAN links
config system interface
 # ISP #1 & ISP #2 are both serviced on port1 (wan1) by using a secondary IP
 # IP address from the same subnet as the primary IP address
 #
 # ISP #1 uses:
 #  172.31.6.216/20 - AWS assigned subnet address for this interface
 #  52.20.209.166 - AWS Elastic IP address associated with 172.31.6.216
 #
 # ISP #2 uses:
 #  172.31.6.217/20 - AWS secondary address for this interface
 #  3.86.170.20 – AWS Elastic IP address associated with 172.31.6.217
 #
 edit “port1”
  # Primary IP address
  set ip 172.31.6.216 255.255.240.0
  set allowaccess ping https ssh
  set bfd enable
  set bfd-desired-min-tx 300
  set bfd-required-min-rx 600
  set type physical
  set inbandwidth 1000000
  set outbandwidth 1000000
  set alias “wan1”
  set role wan
  set mtu-override enable
  set mtu 9001
  # Secondary IP address
  set secondary-IP enable
  config secondaryip
   edit 0
    set ip 172.31.6.217 255.255.255.0
    set allowaccess ping https ssh
   next
  end
 next
end
# Configure VPN Interfaces
config vpn ipsec phase1-interface
 # PtP link between Hub sites
 edit “VPN-DC2SV”
  set interface “port1”
  # Use the primary IP address
  set local-gw 172.31.6.216
  set ike-version 2
  set peertype any
  set net-device enable
  set proposal aes256-sha512
  set dhgrp 32
  # localid not used, but convenient for troubleshooting
  set localid “DC-Hub-IH”
  set dpd on-idle
  #
  #########################################################################
  # ON CERTAIN HARDWARE FORTIGATES MUST ENABLE NPU OFFLOAD TO ENABLE FEC #
  #########################################################################
  #
  set npu-offload enable
  # Enable FEC in both directions
  set fec-egress enable
  set fec-ingress enable
  set psksecret “InterHubPtP”
  set dpd-retrycount 2
  set dpd-retryinterval 5
  set remote-gw 52.52.75.51
 next
 # Establish Dialup (Mesh) VPN server for DC-Hub1
 # Note the use of 'set local-gw x.x.x.x' to override the routing table. This
 # is MANDATORY!!!!
 edit "VPN-DC-Hub1"
  set type dynamic
  set interface "port1"
  # Use the primary IP address
  set local-gw 172.31.6.216
  set mode aggressive
  set peertype one
  set net-device disable
  set mode-cfg enable
  set proposal aes256-sha512
  set add-route disable
  # Outbound connections will be identified as being originated by 'DC-Hub1'
  set localid "DC-Hub1"
  set dpd on-idle
  set dhgrp 32
  set auto-discovery-sender enable
  #
  #########################################################################
  # ON CERTAIN HARDWARE FORTIGATES MUST ENABLE NPU OFFLOAD TO ENABLE FEC #
  #########################################################################
  #
  set npu-offload enable
  # Enable FEC in both directions
  set fec-egress enable
  set fec-ingress enable
  # Inbound connections must have a localid of 'To-DC-Hub1'
  set peerid "To-DC-Hub1"
  set tunnel-search nexthop
  # Define the IP Pool block used to allocate addressing to customer site VPN
  # clients. Note that this configuration allows up to 245 clients - this may
  # be expanded by using a larger netmask / allocation.
  set ipv4-start-ip 10.111.4.10
  set ipv4-end-ip 10.111.4.254
  set ipv4-netmask 255.255.255.0
  set unity-support disable
  set psksecret "PSK-DC-Wan1"
  set dpd-retrycount 2
  set dpd-retryinterval 5
 next
 # Establish Dialup (Mesh) VPN server for DC-Hub2
 # Note the use of 'set local-gw x.x.x.x' to override the routing table. This
 # is MANDATORY!!!!
 edit "VPN-DC-Hub2"
  set type dynamic
  set interface "port1"
  # Use the secondary IP address
  set local-gw 172.31.6.217
  set mode aggressive
  set peertype one
  set net-device disable
```

```
  set mode-cfg enable
  set proposal aes256-sha512
  set add-route disable
  # Outbound connections will be identified as being originated by 'DC-Hub2'
  set localid "DC-Hub2"
  set dpd on-idle
  set dhgrp 32
  set auto-discovery-sender enable
  #
  #########################################################################
  # ON CERTAIN HARDWARE FORTIGATES MUST ENABLE NPU OFFLOAD TO ENABLE FEC #
  #########################################################################
  #
  set npu-offload enable
  # Enable FEC in both directions
  set fec-egress enable
  set fec-ingress enable
  # Inbound connections must have a localid of 'To-DC-Hub2'
  set peerid "To-DC-Hub2"
  set tunnel-search nexthop
  # Define the IP Pool block used to allocate addressing to customer site VPN
  # clients. Note that this configuration allows up to 245 clients - this may
  # be expanded by using a larger netmask / allocation.
  set ipv4-start-ip 10.111.5.10
  set ipv4-end-ip 10.111.5.254
  set ipv4-netmask 255.255.255.0
  set unity-support disable
  set psksecret "PSK-DC-Wan2"
  set dpd-retrycount 2
  set dpd-retryinterval 5
 next
end
config vpn ipsec phase2-interface
 edit "VPN-DC2SV"
  set phase1name "VPN-DC2SV"
  set proposal aes256gcm
  set dhgrp 32
  set auto-negotiate enable
 next
 edit "VPN-DC-Hub1"
  set phase1name "VPN-DC-Hub1"
  set proposal aes256gcm
  set dhgrp 32
  set keepalive enable
 next
 edit "VPN-DC-Hub2"
  set phase1name "VPN-DC-Hub2"
  set proposal aes256gcm
  set dhgrp 32
  set keepalive enable
 next
end
```

```
# Complete configuration of VPN interfaces adding addresses and BFD
config system interface
 # HUB to HUB tunnel
 edit “VPN-DC2SV”
  set ip 10.111.0.2 255.255.255.255
  set allowaccess ping https ssh
  set bfd enable
  set bfd-desired-min-tx 300
  set bfd-required-min-rx 600
  set remote-ip 10.111.0.1 255.255.255.255
 next
 # Dialup Mesh VPN server #1
 edit “VPN-DC-Hub1”
  # Address should be .1 from same block as Phase1 IP Pool.
  # Netmask MUST be /32.
  set ip 10.111.4.1 255.255.255.255
  set allowaccess ping https ssh
  set bfd enable
  set bfd-desired-min-tx 300
  set bfd-required-min-rx 600
  # Dummy address should be .2 from same block as Phase1 IP Pool.
  # Netmask MUST be identical to Phase1 IP Pool netmask.
  set remote-ip 10.111.4.2 255.255.255.0
 next
 # Dialup Mesh VPN server #2
 edit “VPN-DC-Hub2”
  # Address should be .1 from same block as Phase1 IP Pool.
  # Netmask MUST be /32.
  set ip 10.111.5.1 255.255.255.255
  set allowaccess ping https ssh
  set bfd enable
  set bfd-desired-min-tx 300
  set bfd-required-min-rx 600
  # Dummy address should be .2 from same block as Phase1 IP Pool.
  # Netmask MUST be identical to Phase1 IP Pool netmask.
  set remote-ip 10.111.5.2 255.255.255.0
 next
end
# Group interfaces into zones for ease of policy creation
 config system zone
  edit “ZN_Mesh”
   set intrazone allow
   set interface “VPN-DC2SV” “VPN-DC-Hub1” “VPN-DC-Hub2”
  next
  edit “ZN_Wan”
   set intrazone allow
   set interface “port1”
  next
 end
 config firewall shaper traffic-shaper
  # Shaping to apply to tunnels.
  #
  # This sample allows (guaranteed/allowed):
  #  100000/100000Kbps of voice RealTime traffic (1000 calls)
  #  300000/300000Kbps of video RealTime traffic (450 video calls)
  #  1000/3000Kbps of signaling traffic
  #
  edit “TS_DSCP_EF”
   set guaranteed-bandwidth 100000
   set maximum-bandwidth 100000
  next
  edit “TS_DSCP_AF41”
   set guaranteed-bandwidth 300000
   set maximum-bandwidth 300000
   set priority medium
  next
  edit “TS_DSCP_AF31”
   set guaranteed-bandwidth 1000
   set maximum-bandwidth 3000
  next
  # Shaping to apply to WAN ports.
  #
  # Note that the tunnels flow through the WAN ports, so the minimum values
  # used here should be a minimum of the amount for the tunnels plus overhead.
  # Higher values may be used if non-RingCentral traffic is required to be
  # QoS matched.
  edit “TS_W_DSCP_EF”
   set guaranteed-bandwidth 110000
   set maximum-bandwidth 110000
  next
  edit “TS_W_DSCP_AF41”
   set guaranteed-bandwidth 330000
   set maximum-bandwidth 330000
   set priority medium
  next
  edit “TS_W_DSCP_AF31”
   set guaranteed-bandwidth 1100
   set maximum-bandwidth 3300
  next
 end
 #------------------------------------------------------------------------------------
 # Set up the firewall traffic shaping policy. It overrides all other policies for
 # QoS and traffic shaping. Force remarking of return traffic with diffserv-reverse.
 config firewall shaping-policy
 # The first 4 rules are for traffic egressing via the MESH VPNs.
  edit 0
   set name “TSP_H_DSCP_EF”
   set service “ALL”
   set dstintf “ZN_Mesh”
   set tos-mask 0xfc
   set tos 0xb8
   set traffic-shaper “TS_DSCP_EF”
   set traffic-shaper-reverse “TS_DSCP_EF”
   set diffserv-reverse enable
   set srcaddr “all”
   set dstaddr “all”
   set diffservcode-rev 101110
  next
```

```
edit 0
 set name “TSP_H_DSCP_AF41”
 set service “ALL”
 set dstintf “ZN_Mesh”
 set tos-mask 0xfc
 set tos 0x88
 set traffic-shaper “TS_DSCP_AF41”
 set traffic-shaper-reverse “TS_DSCP_AF41”
 set diffserv-reverse enable
 set srcaddr “all”
 set dstaddr “all”
 set diffservcode-rev 100010
next
edit 0
 set name “TSP_H_DSCP_AF31”
 set service “ALL”
 set dstintf “ZN_Mesh”
 set tos-mask 0xfc
 set tos 0x68
 set traffic-shaper “TS_DSCP_AF31”
 set traffic-shaper-reverse “TS_DSCP_AF31”
 set diffserv-reverse enable
 set srcaddr “all”
 set dstaddr “all”
 set diffservcode-rev 011010
next
edit 0
 set name “TSP_H_DSCP_CS3”
 set service “ALL”
 set dstintf “ZN_Mesh”
 set tos-mask 0xfc
 set tos 0x60
 set traffic-shaper “TS_DSCP_AF31”
 set traffic-shaper-reverse “TS_DSCP_AF31”
 set diffserv-forward enable
 set diffserv-reverse enable
 set srcaddr “all”
 set dstaddr “all”
 set diffservcode-forward 011010
 set diffservcode-rev 011010
next
#
# The last 4 rules are for traffic egressing via the last-ditch WAN pathway.
#
# Note that the tunnels to the AWS sites egress via these rules… This requires
# that the values of the traffic shapers must be a minimum of what is allowed
# via the tunnel rules above.
#
edit 0
 set name “TSP_W_DSCP_EF”
 set service “ALL”
 set dstintf “ZN_Wan”
 set tos-mask 0xfc
 set tos 0xb8
 set traffic-shaper “TS_W_DSCP_EF”
```

```
  set traffic-shaper-reverse “TS_W_DSCP_EF”
  set diffserv-reverse enable
  set srcaddr “all”
  set dstaddr “all”
  set diffservcode-rev 101110
 next
 edit 0
  set name “TSP_W_DSCP_AF41”
  set service “ALL”
  set dstintf “ZN_Wan”
  set tos-mask 0xfc
  set tos 0x88
  set traffic-shaper “TS_W_DSCP_AF41”
  set traffic-shaper-reverse “TS_W_DSCP_AF41”
  set diffserv-reverse enable
  set srcaddr “all”
  set dstaddr “all”
  set diffservcode-rev 100010
 next
 edit 0
  set name “TSP_W_DSCP_AF31”
  set service “ALL”
  set dstintf “ZN_Wan”
  set tos-mask 0xfc
  set tos 0x68
  set traffic-shaper “TS_W_DSCP_AF31”
  set traffic-shaper-reverse “TS_W_DSCP_AF31”
  set diffserv-reverse enable
  set srcaddr “all”
  set dstaddr “all”
  set diffservcode-rev 011010
 next
 edit 0
  set name “TSP_W_DSCP_CS3”
  set service “ALL”
  set dstintf “ZN_Wan”
  set tos-mask 0xfc
  set tos 0x60
  set traffic-shaper “TS_W_DSCP_AF31”
  set traffic-shaper-reverse “TS_W_DSCP_AF31”
  set diffserv-forward enable
  set diffserv-reverse enable
  set srcaddr “all”
  set dstaddr “all”
  set diffservcode-forward 011010
  set diffservcode-rev 011010
 next
end
# Establish basic policies to allow mesh to mesh and mesh to wan traffic.
config firewall policy
 # Mesh to Mesh
 edit 0
  set name “POL_Hub2Hub”
  set srcintf “ZN_Mesh”
  set dstintf “ZN_Mesh”
  set srcaddr “all”
  set dstaddr “all”
  set action accept
  set schedule “always”
  set service “ALL”
 next
 # Mesh to Wan
 edit 0
  set name “POL_Hub2Wan”
  set srcintf “ZN_Mesh”
  set dstintf “ZN_Wan”
  set srcaddr “all”
  set dstaddr “all”
  set action accept
  set schedule “always”
  set service “ALL”
  set nat enable
 next
end
# Create access list to match all RingCentral public address space (8 CIDR blocks)
config router access-list
 edit “ACL-RC-All”
  config rule
   edit 0
    set prefix 66.81.240.0 255.255.240.0
    set exact-match enable
   next
   edit 0
    set prefix 80.81.128.0 255.255.240.0
    set exact-match enable
   next
   edit 0
    set prefix 103.44.68.0 255.255.252.0
    set exact-match enable
   next
   edit 0
    set prefix 104.245.56.0 255.255.248.0
    set exact-match enable
   next
   edit 0
    set prefix 185.23.248.0 255.255.252.0
    set exact-match enable
   next
   edit 0
    set prefix 192.209.24.0 255.255.248.0
    set exact-match enable
   next
   edit 0
    set prefix 199.68.212.0 255.255.252.0
    set exact-match enable
   next
   edit 0
    set prefix 199.255.120.0 255.255.252.0
    set exact-match enable
   next
   edit 0
    set prefix 208.87.40.0 255.255.252.0
    set exact-match enable
   next
  end
 next
end
# Establish named communities for use in route-map matching.
config router community-list
 edit “CommunitySetLP140”
  config rule
   edit 0
    set action permit
    set match “65501:140”
   next
  end
 next
 edit “CommunitySetLP130”
  config rule
   edit 0
    set action permit
    set match “65501:130”
   next
  end
 next
 edit “CommunitySetLP120”
  config rule
   edit 0
    set action permit
    set match “65501:120”
   next
  end
 next
 edit “CommunitySetLP110”
  config rule
   edit 0
    set action permit
    set match “65501:110”
   next
  end
 next
end
# Configure route-maps needed for iBGP functionality.
config router route-map
 # Identify route as originating from DC-Hub1 (community 65501:3) and force
 # the next-hop ip address to the address of this tunnel.
 edit “RM-Out-Set-DC1”
 set comments “DC Hub1 VPN”
 config rule
  edit 0
   set set-community “65501:3”
   # Be sure to set the following address to the address of the
   # DC-Hub1 VPN tunnel.
   set set-ip-nexthop 10.111.4.1
  next
  end
 next
 # Identify traffic as originating from DC-Hub2 (community 65501:4) and force
 # the next-hop ip address to the address of this tunnel.
 edit “RM-Out-Set-DC2”
  set comments “DC Hub2 VPN”
  config rule
   edit 0
    set set-community “65501:4”
    # Be sure to set the following address to the address of the
    # DC-Hub2 VPN tunnel.
    set set-ip-nexthop 10.111.5.1
   next
  end
 next
 # Allow ONLY the 8 RingCentral CIDR blocks to redistribute from static to BGP routes
 edit “RM-Redis-Static-2-Bgp”
  config rule
   edit 0
    set match-ip-address “ACL-RC-All”
   next
  end
 next
end
# Create static routes and set up route monitoring.
config router static
 # Default route out port 1, never withdrawn.
 edit 0
  set gateway 172.31.0.1
  set device “port1”
  set link-monitor-exempt enable
 next
 # RingCentral CIDR blocks; disable and withdraw on failure of test.
 edit 0
  set dst 66.81.240.0 255.255.240.0
  set gateway 172.31.0.1
  set device “port1”
 next
 edit 0
  set dst 80.81.128.0 255.255.240.0
  set gateway 172.31.0.1
  set device “port1”
 next
 edit 0
  set dst 103.44.68.0 255.255.252.0
  set gateway 172.31.0.1
  set device “port1”
 next
 edit 0
  set dst 104.245.56.0 255.255.248.0
  set gateway 172.31.0.1
  set device “port1”
 next
 edit 0
  set dst 185.23.248.0 255.255.252.0
  set gateway 172.31.0.1
  set device “port1”
 next
 edit 0
  set dst 192.209.24.0 255.255.248.0
  set gateway 172.31.0.1
  set device “port1”
 next
 edit 0
  set dst 199.68.212.0 255.255.252.0
  set gateway 172.31.0.1
  set device “port1”
 next
 edit 0
  set dst 199.255.120.0 255.255.252.0
  set gateway 172.31.0.1
  set device “port1”
 next
 edit 0
  set dst 208.87.40.0 255.255.252.0
  set gateway 172.31.0.1
  set device “port1”
 next
end
# Create route monitor using 199.255.120.129 test address.
config system link-monitor
 edit “1”
  set srcintf “port1”
  set server “199.255.120.129”
  set interval 15000
  set failtime 3
  set recoverytime 2
 next
end
# Set up iBGP using private AS 65501.
config router bgp
 set as 65501
 set router-id 10.111.0.2
 config neighbor
  edit “10.111.0.1”
   set bfd enable
   set link-down-failover enable
   set next-hop-self enable
   set soft-reconfiguration enable
   set interface “VPN-DC2SV”
   set remote-as 65501
   set update-source “VPN-DC2SV”
  next
 end
 # Define a neighbor-group (template) for each dialup Hub member.
 config neighbor-group
  edit “clients-hub1”
   set bfd enable
   set link-down-failover enable
   set next-hop-self enable
   set soft-reconfiguration enable
   set interface “VPN-DC-Hub1”
   set remote-as 65501
   # Tag outbound routes with a community indicating this Hub and
   # force the nexthop to be the origin IP address of this Hub.
   set route-map-out “RM-Out-Set-DC1”
   set update-source “VPN-DC-Hub1”
   set route-reflector-client enable
  next
  edit “clients-hub2”
   set bfd enable
   set link-down-failover enable
   set next-hop-self enable
   set soft-reconfiguration enable
   set interface “VPN-DC-Hub2”
   set remote-as 65501
   # Tag outbound routes with a community indicating this Hub and
   # force the nexthop to be the origin IP address of this Hub.
   set route-map-out “RM-Out-Set-DC2”
   set update-source “VPN-DC-Hub2”
   set route-reflector-client enable
  next
 end
 # Define IP ranges for each Neighbor group. These must match the IP Pools
 # defined in the Phase1 definitions.
 config neighbor-range
  edit 0
   set prefix 10.111.4.0 255.255.255.0
   set neighbor-group “clients-hub1”
  next
  edit 0
   set prefix 10.111.5.0 255.255.255.0
   set neighbor-group “clients-hub2”
  next
 end
 # Redistribute static routes matching the route-map to all clients.
 config redistribute “static”
  set status enable
  set route-map “RM-Redis-Static-2-Bgp”
 end
end
```

Customer Site #1

Global Settings

The description of global settings from the AWS-SV configuration also apply here.

```
config global
config system global
 # Automatically return TCP RST packet on transmission to invalid session
 set reset-sessionless-tcp enable
 # Set system to use DSCP in lieu of TOS
 set traffic-priority dscp
 # Traffic defaults to low priority unless overridden
 set traffic-priority-level low
end
# Configure DSCP to priority mapping table
```

```
config system dscp-based-priority
 edit 46
  set ds 46
  set priority high
 next
 edit 34
  set ds 34
  set priority medium
 next
 edit 26
  set ds 26
  set priority high
 next
end
end
```

VDOM Specific Settings

```
config vdom
edit “root”
config system settings
 # Enable BFD for rapid outage detection
 set bfd enable
 set bfd-desired-min-tx 300
 set bfd-required-min-rx 600
 # Set up logging of VPN data
 set vpn-stats-log ipsec
 set vpn-stats-period 60
 # Set up rapid detection of VPN IKE outages
 set ike-quick-crash-detect enable
end
# Configure Physical ports
config system interface
 # Management access is on port1 (this will vary by site)
 edit “port1”
  set ip 172.16.255.69 255.255.255.0
  set allowaccess ping https ssh
  set alias “Management”
  set role lan
 next
 # LAN access is on port4 (this will vary by site)
 edit “port4”
  set ip 192.168.130.1 255.255.255.0
  set allowaccess ping https ssh
  set alias “lan1”
  set role lan
 next
 # ISP #1 is on port2 (wan1) (this will vary by site)
 #
 #  173.95.76.198/27 - Public Spectrum Cable address
 #   In this lab configuration, this is the backup interface.
 #
 edit “port2”
  set ip 173.95.76.198 255.255.255.224
  set allowaccess ping https ssh
  # Set in and out bandwidth in Kbps (Must be correct!!!)
  set inbandwidth 100000
  set outbandwidth 5000
  set description “Spectrum”
  set alias “wan1”
  set role wan
 next
 # ISP #2 in on port3 (wan2) (this will vary by site)
 #
 #  12.31.117.9/27 – Public ATT Internet Access address
 #  In this lab configuration, this is the preferred interface.
 #
 edit “port3”
  set ip 12.31.117.9 255.255.255.224
  set allowaccess ping https ssh
  # Set in and out bandwidth in Kbps (Must be correct!!!)
  set inbandwidth 100000
  set outbandwidth 100000
  set description “ATT”
  set alias “wan2”
  set role wan
 next
end
# Configure VPN Interfaces
config vpn ipsec phase1-interface
 # Establish Dialup (Mesh) VPN client connecting to AWS-SV-Hub1 using ATT link
 edit “VPN-SV-Att”
  set interface “port3”
  # Force next-hop to ATT interface/gateway and ignore routing table
  set local-gw 12.31.117.9
  set mode aggressive
  set peertype one
  set net-device disable
  set mode-cfg enable
  set proposal aes256-sha512
  set dhgrp 32
  set add-route disable
  set auto-discovery-receiver enable
  # Set ID of this tunnel to match expected peer in AWS-SV-Hub1 configuration.
  set localid “To-SV-Hub1”
  # Require PeerID of remote side to be SV-Hub1
  set peerid “SV-Hub1”
  #
  #########################################################################
  # ON CERTAIN HARDWARE FORTIGATES MUST ENABLE NPU OFFLOAD TO ENABLE FEC #
  #########################################################################
  #
  set npu-offload enableddd
  # Enable FEC in both directions
  set fec-egress enable
  set fec-ingress enable
  set remote-gw 52.52.75.51
  set tunnel-search nexthop
  set dpd on-idle
  set dpd-retrycount 2
  set dpd-retryinterval 5
  set psksecret “PSK-SV-Wan1”
 next
 # Enable FEC in both directions
  set fec-egress enable
  set fec-ingress enable
  set remote-gw 52.52.75.51
  set tunnel-search nexthop
  set dpd on-idle
  set dpd-retrycount 2
  set dpd-retryinterval 5
  set psksecret “PSK-SV-Wan1”
 next
 # Establish Dialup (Mesh) VPN client connecting to AWS-SV-Hub2 using Spectrum link
 edit “VPN-SV-Spec”
  set interface “port2”
  # Force next-hop to Spectrum interface/gateway and ignore routing table
  set local-gw 173.95.76.198
  set mode aggressive
  set peertype one
  set net-device disable
  set mode-cfg enable
  set proposal aes256-sha512
  set dhgrp 32
  set add-route disable
  set auto-discovery-receiver enable
  # Set ID of this tunnel to match expected peer in AWS-SV-Hub2 configuration.
  set localid “To-SV-Hub2”
  # Require PeerID of remote side to be SV-Hub2
  set peerid “SV-Hub2”
  #
  #########################################################################
  # ON CERTAIN HARDWARE FORTIGATES MUST ENABLE NPU OFFLOAD TO ENABLE FEC #
  #########################################################################
  #
  set npu-offload enable
  # Enable FEC in both directions
  set fec-egress enable
  set fec-ingress enable
  set remote-gw 13.52.166.222
  set tunnel-search nexthop
  set dpd on-idle
  set dpd-retrycount 2
  set dpd-retryinterval 5
  set psksecret “PSK-SV-Wan2”
 next
 # Establish Dialup (Mesh) VPN client connecting to AWS-DC-Hub1 using ATT link
 edit “VPN-DC-Att”
  set interface “port3”
  # Force next-hop to ATT interface/gateway and ignore routing table
  set local-gw 12.31.117.9
  set mode aggressive
  set peertype one
  set net-device disable
  set mode-cfg enable
  set proposal aes256-sha512
  set dhgrp 32
  set add-route disable
  set auto-discovery-receiver enable
  # Set ID of this tunnel to match expected peer in AWS-DC-Hub1 configuration.
  set localid “To-DC-Hub1”
  # Require PeerID of remote side to be DC-Hub1
  set peerid “DC-Hub1”
  #
  #########################################################################
  # ON CERTAIN HARDWARE FORTIGATES MUST ENABLE NPU OFFLOAD TO ENABLE FEC #
  #########################################################################
  #
  set npu-offload enable
  # Enable FEC in both directions
  set fec-egress enable
  set fec-ingress enable
  set remote-gw 52.20.209.166
  set tunnel-search nexthop
  set dpd on-idle
  set dpd-retrycount 2
  set dpd-retryinterval 5
  set psksecret “PSK-DC-Wan1”
 next
 # Establish Dialup (Mesh) VPN client connecting to AWS-DC-Hub2 using Spectrum link
 edit “VPN-DC-Spec”
  set interface “port2”
  # Force next-hop to Spectrum interface/gateway and ignore routing table
  set local-gw 173.95.76.198
  set mode aggressive
  set peertype one
  set net-device disable
  set mode-cfg enable
  set proposal aes256-sha512
  set dhgrp 32
  set add-route disable
  set auto-discovery-receiver enable
  # Set ID of this tunnel to match expected peer in AWS-DC-Hub2 configuration.
  set localid “To-DC-Hub2”
  # Require PeerID of remote side to be DC-Hub1
  set peerid “DC-Hub2”
  #
  #########################################################################
  # ON CERTAIN HARDWARE FORTIGATES MUST ENABLE NPU OFFLOAD TO ENABLE FEC #
  #########################################################################
  #
  set npu-offload enable
  # Enable FEC in both directions
  set fec-egress enable
  set fec-ingress enable
  set remote-gw 3.86.170.20
  set tunnel-search nexthop
  set dpd on-idle
  set dpd-retrycount 2
  set dpd-retryinterval 5
```

```
  set psksecret “PSK-DC-Wan2”
 next
end
config vpn ipsec phase2-interface
 edit “VPN-DC-Att”
  set phase1name “VPN-DC-Att”
  set proposal aes256gcm
  set dhgrp 32
  set auto-negotiate enable
 next
 edit “VPN-DC-Spec”
  set phase1name “VPN-DC-Spec”
  set proposal aes256gcm
  set dhgrp 32
  set auto-negotiate enable
 next
 edit “VPN-SV-Spec”
  set phase1name “VPN-SV-Spec”
  set proposal aes256gcm
  set dhgrp 32
  set auto-negotiate enable
 next
 edit “VPN-SV-Att”
  set phase1name “VPN-SV-Att”
  set proposal aes256gcm
  set dhgrp 32
  set auto-negotiate enable
 next
end
# Complete configuration of VPN interfaces adding addresses and BFD
config system interface
 edit “VPN-SV-Att”
  set ip 0.0.0.0 255.255.255.255
  set bfd enable
  set bfd-desired-min-tx 300
  set bfd-required-min-rx 600
  set type tunnel
  set inbandwidth 100000
  set outbandwidth 100000
  set role wan
  set interface “port3”
 next
 edit “VPN-SV-Spec”
  set ip 0.0.0.0 255.255.255.255
  set bfd enable
  set bfd-desired-min-tx 300
  set bfd-required-min-rx 600
  set type tunnel
  set inbandwidth 100000
  set outbandwidth 5000
  set role wan
  set interface “port2”
 next
 edit “VPN-DC-Att”
  set ip 0.0.0.0 255.255.255.255
  set bfd enable
  set bfd-desired-min-tx 300
  set bfd-required-min-rx 600
  set type tunnel
  set inbandwidth 100000
  set outbandwidth 100000
  set role wan
  set interface “port3”
 next
 edit “VPN-DC-Spec”
  set ip 0.0.0.0 255.255.255.255
  set bfd enable
  set bfd-desired-min-tx 300
  set bfd-required-min-rx 600
  set type tunnel
  set inbandwidth 100000
  set outbandwidth 5000
  set role wan
  set interface “port2”
 next
end
# Group interfaces into zones for ease of use
config system zone
 edit “ZN_Wan”
  set interface “port2” “port3”
 next
 edit “ZN_Lan”
  set interface “port4”
 next
 edit “ZN_Mgmt”
  set interface “port1”
 next
 edit “ZN_Mesh”
  set intrazone allow
  set interface “VPN-DC-Att” “VPN-DC-Spec” “VPN-SV-Att” “VPN-SV-Spec”
 next
end
# Establish traffic shapers for each category of traffic
#
# NOTE: These values must be adjusted to match reality. They are expressed in
#       Kilobits per second (kbps).
config firewall shaper traffic-shaper
 # Shaping to apply to tunnels.
 #
 # This sample allows (guaranteed/allowed):
 # 800/800Kbps of voice RealTime traffic (10 calls)
 # 1200/2000Kbps of video RealTime traffic (2 video calls, 3 max)
 # 64/128Kbps of signaling traffic
 #
 edit “TS_DSCP_EF”
  set guaranteed-bandwidth 800
  set maximum-bandwidth 800
 next
 edit “TS_DSCP_AF41”
  set guaranteed-bandwidth 1200
  set maximum-bandwidth 2000
  set priority medium
 next
 edit “TS_DSCP_AF31”
  set guaranteed-bandwidth 64
  set maximum-bandwidth 128
 next
 # Shaping to apply to WAN ports.
 #
 # Note that the tunnels flow through the WAN ports, so the minimum values
 # used here should be a minimum of the amount for the tunnels plus overhead.
 # Higher values may be used if non-RingCentral traffic is required to be
 # QoS matched.
 edit “TS_W_DSCP_EF”
  set guaranteed-bandwidth 900
  set maximum-bandwidth 900
 next
 edit “TS_W_DSCP_AF41”
  set guaranteed-bandwidth 1400
  set maximum-bandwidth 2200
  set priority medium
 next
 edit “TS_W_DSCP_AF31”
  set guaranteed-bandwidth 84
  set maximum-bandwidth 168
 next
end
#------------------------------------------------------------------------------------
# Set up the firewall traffic shaping policy. It overrides all other policies for
# QoS and traffic shaping. Force remarking of return traffic with diffserv-reverse.
config firewall shaping-policy
 # The first 4 rules are for traffic egressing via the MESH VPNs.
 edit 0
  set name “TSP_H_DSCP_EF”
  set service “ALL”
  set dstintf “ZN_Mesh”
  set tos-mask 0xfc
  set tos 0xb8
  set traffic-shaper “TS_DSCP_EF”
  set traffic-shaper-reverse “TS_DSCP_EF”
  set diffserv-reverse enable
  set srcaddr “all”
  set dstaddr “all”
  set diffservcode-rev 101110
 next
 edit 0
  set name “TSP_H_DSCP_AF41”
  set service “ALL”
  set dstintf “ZN_Mesh”
  set tos-mask 0xfc
  set tos 0x88
  set traffic-shaper “TS_DSCP_AF41”
  set traffic-shaper-reverse “TS_DSCP_AF41”
  set diffserv-reverse enable
  set srcaddr “all”
  set dstaddr “all”
  set diffservcode-rev 100010
 next
 edit 0
  set name “TSP_H_DSCP_AF31”
  set service “ALL”
  set dstintf “ZN_Mesh”
  set tos-mask 0xfc
  set tos 0x68
  set traffic-shaper “TS_DSCP_AF31”
  set traffic-shaper-reverse “TS_DSCP_AF31”
  set diffserv-reverse enable
  set srcaddr “all”
  set dstaddr “all”
  set diffservcode-rev 011010
 next
 edit 0
  set name “TSP_H_DSCP_CS3”
  set service “ALL”
  set dstintf “ZN_Mesh”
  set tos-mask 0xfc
  set tos 0x60
  set traffic-shaper “TS_DSCP_AF31”
  set traffic-shaper-reverse “TS_DSCP_AF31”
  set diffserv-forward enable
  set diffserv-reverse enable
  set srcaddr “all”
  set dstaddr “all”
  set diffservcode-forward 011010
  set diffservcode-rev 011010
 next
 #
 # The last 4 rules are for traffic egressing via the last-ditch WAN pathway.
 #
 # Note that the tunnels to the AWS sites egress via these rules… This requires
 # that the values of the traffic shapers must be a minimum of what is allowed
 # via the tunnel rules above.
 #
 edit 0
  set name “TSP_W_DSCP_EF”
  set service “ALL”
  set dstintf “ZN_Wan”
  set tos-mask 0xfc
  set tos 0xb8
  set traffic-shaper “TS_W_DSCP_EF”
  set traffic-shaper-reverse “TS_W_DSCP_EF”
  set diffserv-reverse enable
  set srcaddr “all”
  set dstaddr “all”
  set diffservcode-rev 101110
 next
edit 0
  set name “TSP_W_DSCP_AF41”
  set service “ALL”
  set dstintf “ZN_Wan”
  set tos-mask 0xfc
  set tos 0x88
  set traffic-shaper “TS_W_DSCP_AF41”
  set traffic-shaper-reverse “TS_W_DSCP_AF41”
  set diffserv-reverse enable
  set srcaddr “all”
  set dstaddr “all”
  set diffservcode-rev 100010
 next
 edit 0
  set name “TSP_W_DSCP_AF31”
  set service “ALL”
  set dstintf “ZN_Wan”
  set tos-mask 0xfc
  set tos 0x68
  set traffic-shaper “TS_W_DSCP_AF31”
  set traffic-shaper-reverse “TS_W_DSCP_AF31”
  set diffserv-reverse enable
  set srcaddr “all”
  set dstaddr “all”
  set diffservcode-rev 011010
 next
 edit 0
  set name “TSP_W_DSCP_CS3”
  set service “ALL”
  set dstintf “ZN_Wan”
  set tos-mask 0xfc
  set tos 0x60
  set traffic-shaper “TS_W_DSCP_AF31”
  set traffic-shaper-reverse “TS_W_DSCP_AF31”
  set diffserv-forward enable
  set diffserv-reverse enable
  set srcaddr “all”
  set dstaddr “all”
  set diffservcode-forward 011010
  set diffservcode-rev 011010
 next
end
#------------------------------------------------------------------------------------
# Establish Addresses, Address-Groups, and Services that can be used in policies to
# identify specific RingCentral traffic.
#
config firewall address
 edit “ADR_RC_1”
  set subnet 103.44.68.0 255.255.252.0
 next
 edit “ADR_RC_2”
  set subnet 104.245.56.0 255.255.248.0
 next
 edit “ADR_RC_3”
  set subnet 185.23.248.0 255.255.252.0
 next
 edit “ADR_RC_4”
  set subnet 192.209.24.0 255.255.248.0
 next
 edit “ADR_RC_5”
  set subnet 199.255.120.0 255.255.252.0
 next
 edit “ADR_RC_6”
  set subnet 199.68.212.0 255.255.252.0
 next
 edit “ADR_RC_7”
 next
 edit “ADR_RC_8”
  set subnet 80.81.128.0 255.255.240.0
 next
 edit “ADR_RC_9”
  set subnet 66.81.240.0 255.255.240.0
 next
 edit “ADR_RC_11”
  set type fqdn
  set fqdn “ringcentral.com”
 next
 edit “ADR_RC_Prov_1”
  set type fqdn
  set fqdn “pp.ringcentral.com”
 next
 edit “ADR_RC_Prov_2”
  set type fqdn
  set fqdn “cp.ringcentral.com”
 next
 edit “ADR_RC_Prov_3”
  set type fqdn
  set fqdn “yp.ringcentral.com”
 next
 edit “ADR_RC_FwUp_1”
  set type fqdn
  set fqdn “pp.s3.ringcentral.com”
 next
 edit “ADR_RC_API_1”
  set type fqdn
  set fqdn “platform.ringcentral.com”
 next
 edit “ADR_RC_API_2”
  set type fqdn
  set fqdn “platform.devtest.ringcentral.com”
 next
end
config firewall addrgrp
 edit “AG_RingCentral”
  set member “ADR_RC_1” “ADR_RC_2” “ADR_RC_3” “ADR_RC_4” “ADR_RC_5” “ADR_RC_6”
“ADR_RC_7” “ADR_RC_8” “ADR_RC_9”
 next
 edit “AG_RC_Prov”
  set member “ADR_RC_Prov_1” “ADR_RC_Prov_2” “ADR_RC_Prov_3”
 next
edit “AG_RC_FwUp”
```

```
  set member “ADR_RC_FwUp_1”
 next
 edit “AG_RC_API”
  set member “ADR_RC_API_1” “ADR_RC_API_2”
 next
end
config firewall service custom
 edit “SVC_RC_SIP”
  set category “VoIP, Messaging & Other Applications”
  set tcp-portrange 5090 5091 5093 5094 5096 5097 5099 8083
  set udp-portrange 5090
 next
 edit “SVC_RC_RTP”
  set category “VoIP, Messaging & Other Applications”
  set udp-portrange 20000-64999
 next
 edit “SVC_RC_Prov”
  set category “VoIP, Messaging & Other Applications”
  set tcp-portrange 443
 next
 edit “SVC_RC_FwUp”
  set category “VoIP, Messaging & Other Applications”
  set tcp-portrange 443
 next
 edit “SVC_RC_Pres”
  set category “VoIP, Messaging & Other Applications”
  set tcp-portrange 80 443
 next
 edit “SVC_RC_API”
  set category “VoIP, Messaging & Other Applications”
  set tcp-portrange 443
 next
 edit “SVC_RC_Video”
  set category “VoIP, Messaging & Other Applications”
  set tcp-portrange 8801-8802 3000-4000
  set udp-portrange 3000-4000 3478-3479 8801-8802 8810-8829 9000-9999 10000-19999
 next
end
#------------------------------------------------------------------------------------
# Establish firewall policies. These policies enforce QoS as well as allow traffic
# to flow
#
#####################################################################################
# Make SURE that you move these policies to a position BEFORE any existing policies #
# already defined in your firewall for each interface-pair. #
#####################################################################################
config firewall policy
 # Allow all LAN <=> Mgmt traffic flows
 edit 0
  set name “POL_Lan2Mgmt”
  set srcintf “ZN_Lan”
  set dstintf “ZN_Mgmt”
  set srcaddr “all”
  set dstaddr “all”
  set action accept
  set schedule “always”
  set service “ALL”
 next
 # Allow all Mgmt <=> LAN traffic flows
 edit 0
  set name “POL_Mgmt2Lan”
  set srcintf “ZN_Mgmt”
  set dstintf “ZN_Lan”
  set srcaddr “all”
  set dstaddr “all”
  set action accept
  set schedule “always”
  set service “ALL”
 next
 # Allow all Mesh/Hub <=> Mgmt traffic flows
 edit 0
  set name “POL_Mesh2Mgmt”
  set srcintf “ZN_Mesh”
  set dstintf “ZN_Mgmt”
  set srcaddr “all”
  set dstaddr “all”
  set action accept
  set schedule “always”
  set service “ALL”
 next
 # Allow all Mgmt <=> Mesh/Hub traffic flows
 edit 0
  set name “POL_Mgmt2Mesh”
  set srcintf “ZN_Mgmt”
  set dstintf “ZN_Mesh”
  set srcaddr “all”
  set dstaddr “all”
  set action accept
  set schedule “always”
  set service “ALL”
 next
 # Allow all Mgmt <=> WAN traffic flows (most common configuration)
 edit 0
  set name “POL_Mgmt2Wan”
  set srcintf “ZN_Mgmt”
  set dstintf “ZN_Wan”
  set srcaddr “all”
  set dstaddr “all”
  set action accept
  set schedule “always”
  set service “ALL”
  set nat enable
 next
 # Allow all Mesh/Hub <=> Mesh/Hub traffic flows (inter-client)
 edit 0
  set name “POL_HUB”
  set srcintf “ZN_Mesh”
  set dstintf “ZN_Mesh”
  set srcaddr “all”
  set dstaddr “all”
  set action accept
  set schedule “always”
  set service “ALL”
 next
 # Allow all Mesh/Hub <=> WAN traffic flows (this won’t occur unless you add routing
 # allowing spokes to use each other’s WAN links for outbound traffic)
 edit 0
  set name “POL_Mesh2Wan”
  set srcintf “ZN_Mesh”
  set dstintf “ZN_Wan”
  set srcaddr “all”
  set dstaddr “all”
  set action accept
  set schedule “always”
  set service “ALL”
  set nat enable
 next
 #=========================================================================
 # Allow all Mesh/Hub <=> LAN traffic flows (this is the return traffic for
 # all RingCentral bound traffic)
 #
 # All DSCP marking should have already been accomplished by the HUB sites.
 #
 edit 0
  set name “POL_Mesh2Lan”
  set srcintf “ZN_Mesh”
  set dstintf “ZN_Lan”
  set srcaddr “all”
  set dstaddr “all”
  set action accept
  set schedule “always”
  set service “ALL”
 next
 #=========================================================================
 # LAN <=> Mesh/Hub traffic flows (this is the desired flow for RingCentral
 # bound traffic)
 #
 # Real-time Audio
 edit 0
  set name “POL_RC_H_RTP”
  set srcintf “ZN_Lan”
  set dstintf “ZN_Mesh”
  set srcaddr “all”
  set dstaddr “AG_RingCentral”
  set action accept
  set schedule “always”
  set service “SVC_RC_RTP”
  set vlan-cos-fwd 5
  set vlan-cos-rev 5
  set diffserv-forward enable
  set diffserv-reverse enable
  set diffservcode-forward 101110
  set diffservcode-rev 101110
  set timeout-send-rst enable
  set traffic-shaper “TS_DSCP_EF”
  set traffic-shaper-reverse “TS_DSCP_EF”
 next
 # Real-time Video
 edit 0
  set name “POL_RC_H_Meeting”
  set srcintf “ZN_Lan”
  set dstintf “ZN_Mesh”
  set srcaddr “all”
  set dstaddr “AG_RingCentral”
  set action accept
  set schedule “always”
  set service “SVC_RC_Video”
  set vlan-cos-fwd 4
  set vlan-cos-rev 4
  set diffserv-forward enable
  set diffserv-reverse enable
  set diffservcode-forward 100010
  set diffservcode-rev 100010
  set timeout-send-rst enable
  set traffic-shaper “TS_DSCP_AF41”
  set traffic-shaper-reverse “TS_DSCP_AF41”
 next
 # Meetings (P2P) traffic (must already be marked DSCP AF41!!!)
 edit 0
  set name “POL_RC_H_Meetings_P2P_Mkd”
  set srcintf “ZN_Lan”
  set dstintf “ZN_Mesh”
  set srcaddr “all”
  set dstaddr “all”
  set action accept
  set schedule “always”
  set service “ALL_UDP”
  set tos-mask 0xfc
  set tos 0x88
  set diffserv-reverse enable
  set diffservcode-rev 100010
  set vlan-cos-fwd 4
  set vlan-cos-rev 4
  set comments “RC Meeting already marked Peer 2 Peer”
  set traffic-shaper “TS_DSCP_AF41”
  set traffic-shaper-reverse “TS_DSCP_AF41”
 next
 # Signaling traffic
 edit 0
  set name “POL_RC_H_SIP”
  set srcintf “ZN_Lan”
  set dstintf “ZN_Mesh”
  set srcaddr “all”
  set dstaddr “AG_RingCentral”
  set action accept
  set schedule “always”
  set service “SVC_RC_SIP”
  set vlan-cos-fwd 3
  set vlan-cos-rev 3
  set diffserv-forward enable
  set diffserv-reverse enable
  set diffservcode-forward 011010
  set diffservcode-rev 011010
  set timeout-send-rst enable
  set traffic-shaper “TS_DSCP_AF31”
  set traffic-shaper-reverse “TS_DSCP_AF31”
 next
 # Provisioning traffic
 edit 0
  set name “POL_RC_H_Prov”
  set srcintf “ZN_Lan”
  set dstintf “ZN_Mesh”
  set srcaddr “all”
  set dstaddr “AG_RC_Prov”
  set action accept
  set schedule “always”
  set service “SVC_RC_Prov”
  set timeout-send-rst enable
 next
 # Firmware Update traffic
 edit 0
  set name “POL_RC_H_FW_Update”
  set srcintf “ZN_Lan”
  set dstintf “ZN_Mesh”
  set srcaddr “all”
  set dstaddr “AG_RC_FwUp”
  set action accept
  set schedule “always”
  set service “SVC_RC_FwUp”
  set timeout-send-rst enable
 next
 # API traffic
 edit 0
  set name “POL_RC_H_API”
  set srcintf “ZN_Lan”
  set dstintf “ZN_Mesh”
  set srcaddr “all”
  set dstaddr “AG_RC_API”
  set action accept
  set schedule “always”
  set service “SVC_RC_API”
  set timeout-send-rst enable
 next
 # Default for any RC traffic not already matched
 edit 0
  set name “POL_Lan2Mesh”
  set srcintf “ZN_Lan”
  set dstintf “ZN_Mesh”
  set srcaddr “all”
  set dstaddr “AG_RingCentral”
  set action accept
  set schedule “always”
  set service “ALL”
 next
```

```
#=========================================================================
# LAN <=> WAN traffic flows (this is the last-ditch failover route used
# only if none of the Hubs are reachable)
#
# Real-time Audio
edit 0
 set name “POL_RC_Meeting”
 set srcintf “ZN_Lan”
 set dstintf “ZN_Wan”
 set srcaddr “all”
 set dstaddr “AG_RingCentral”
 set action accept
 set schedule “always”
 set service “SVC_RC_RTP”
 set vlan-cos-fwd 5
 set vlan-cos-rev 5
 set diffserv-forward enable
 set diffserv-reverse enable
 set diffservcode-forward 101110
 set diffservcode-rev 101110
 set traffic-shaper “TS_DSCP_EF”
 set traffic-shaper-reverse “TS_DSCP_EF”
 set nat enable
next
# Real-time Video
edit 0
 set name “POL_RC_Meeting”
 set srcintf “ZN_Lan”
 set dstintf “ZN_Wan”
 set srcaddr “all”
 set dstaddr “AG_RingCentral”
 set action accept
 set schedule “always”
 set service “SVC_RC_Video”
 set vlan-cos-fwd 4
 set vlan-cos-rev 4
 set diffserv-forward enable
 set diffserv-reverse enable
 set diffservcode-forward 100010
 set diffservcode-rev 100010
 set timeout-send-rst enable
 set traffic-shaper “TS_DSCP_AF41”
 set traffic-shaper-reverse “TS_DSCP_AF41”
 set nat enable
next
# Meetings (P2P) traffic (must already be marked DSCP AF41!!!)
edit 0
 set name “POL_RC_Meetings_P2P_Marked”
 set srcintf “ZN_Lan”
 set dstintf “ZN_Wan”
 set srcaddr “all”
 set dstaddr “all”
 set action accept
 set schedule “always”
 set service “ALL_UDP”
 set tos-mask 0xfc
 set tos 0x88
 set vlan-cos-fwd 4
 set vlan-cos-rev 4
 set comments “RC Meeting already marked Peer 2 Peer”
 set nat enable
next
# Signaling traffic
edit 0
 set name “POL_RC_SIP”
 set srcintf “ZN_Lan”
 set dstintf “ZN_Wan”
 set srcaddr “all”
 set dstaddr “AG_RingCentral”
 set action accept
 set schedule “always”
 set service “SVC_RC_SIP”
 set vlan-cos-fwd 3
 set vlan-cos-rev 3
 set diffserv-forward enable
 set diffserv-reverse enable
 set diffservcode-forward 011010
 set diffservcode-rev 011010
 set timeout-send-rst enable
 set traffic-shaper “TS_DSCP_AF31”
 set traffic-shaper-reverse “TS_DSCP_AF31”
 set nat enable
next
# Provisioning traffic
edit 0
 set name “POL_RC_Prov”
 set srcintf “ZN_Lan”
 set dstintf “ZN_Wan”
 set srcaddr “all”
 set dstaddr “AG_RC_Prov”
 set action accept
 set schedule “always”
 set service “SVC_RC_Prov”
 set timeout-send-rst enable
 set nat enable
next
# Firmware Update traffic
edit 0
 set name “POL_RC_FW_Update”
 set srcintf “ZN_Lan”
 set dstintf “ZN_Wan”
 set srcaddr “all”
 set dstaddr “AG_RC_FwUp”
 set action accept
 set schedule “always”
 set service “SVC_RC_FwUp”
 set timeout-send-rst enable
 set nat enable
next
# API traffic
```

```
 edit 0
  set name “POL_RC_API”
  set srcintf “ZN_Lan”
  set dstintf “ZN_Wan”
  set srcaddr “all”
  set dstaddr “AG_RC_API”
  set action accept
  set schedule “always”
  set service “SVC_RC_API”
  set timeout-send-rst enable
  set nat enable
 next
 # Default for ALL OTHER traffic not already matched
 edit 0
  set name “POL_Lan2Wan”
  set srcintf “ZN_Lan”
  set dstintf “ZN_Wan”
  set srcaddr “all”
  set dstaddr “all”
  set action accept
  set schedule “always”
  set service “ALL”
  set ssl-ssh-profile “certificate-inspection”
  set nat enable
 next
end
#------------------------------------------------------------------------------------
# Set up an access-list for matching all routes destined to the RingCentral
# owned address space.
config router access-list
 edit “ACL-RC-All”
  config rule
   edit 0
    set prefix 66.81.240.0 255.255.240.0
    set exact-match enable
   next
   edit 0
    set prefix 80.81.128.0 255.255.240.0
    set exact-match enable
   next
   edit 0
    set prefix 103.44.68.0 255.255.252.0
    set exact-match enable
   next
   edit 0
    set prefix 104.245.56.0 255.255.248.0
    set exact-match enable
   next
   edit 0
    set prefix 185.23.248.0 255.255.252.0
    set exact-match enable
   next
   edit 0
    set prefix 192.209.24.0 255.255.248.0
    set exact-match enable
   next
   edit 0
    set prefix 199.68.212.0 255.255.252.0
    set exact-match enable
   next
   edit 0
    set prefix 199.255.120.0 255.255.252.0
    set exact-match enable
   next
   edit 0
    set prefix 208.87.40.0 255.255.252.0
    set exact-match enable
   next
  end
 next
end
#------------------------------------------------------------------------------------
# Set up route-maps to control routing.
config router route-map
 # Mark routes as primary path
 edit “RM-PrimaryPath”
  config rule
   edit 0
    set set-local-preference 140
   next
  end
 next
 # Mark routes as secondary path
edit “RM-SecondaryPath”
  config rule
   edit 0
    set set-local-preference 130
   next
  end
 next
 # Mark routes as tertiary path
 edit “RM-TertiaryPath”
  config rule
   edit 0
    set set-local-preference 120
   next
  end
 next
 # Mark routes as quarternary path
 edit “RM-QuarternaryPath”
  config rule
   edit 0
    set set-local-preference 110
   next
  end
 next
end
#------------------------------------------------------------------------------------
# Set up static routes
config router static
```

```
 # Management routes.
 edit 0
  set dst 172.16.0.0 255.240.0.0
  set gateway 172.16.255.1
  set device “port1”
 next
 # Preferred ISP default route.
 edit 0
  set gateway 12.31.117.1
  set device “port3”
 next
 # Backup ISP default route.
 edit 0
  set gateway 173.95.76.193
  set distance 20
  set device “port2”
 next
 # Hard route to SV-Hub1 over ATT on port3. Do not withdraw.
 edit 0
  set dst 52.52.75.51 255.255.255.255
  set gateway 12.31.117.1
  set device “port3”
  set link-monitor-exempt ena
 next
 # Hard route to DC-Hub1 over ATT on port3. Do not withdraw.
 edit 0
  set dst 52.50.209.166 255.255.255.255
  set gateway 12.31.117.1
  set device “port3”
  set link-monitor-exempt ena
 next
 # Hard route to SV-Hub2 over Spectrum on port2. Do not withdraw.
 edit 0
  set dst 13.52.166.222 255.255.255.255
  set gateway 173.95.76.193
  set device “port2”
  set link-monitor-exempt ena
 next
 # Hard route to DC-Hub2 over Spectrum on port2. Do not withdraw.
 edit 0
  set dst 3.86.170.20 255.255.255.255
  set gateway 173.95.76.193
  set device “port2”
  set link-monitor-exempt ena
 next
end
#------------------------------------------------------------------------------------
# Monitor default route and switch to secondary no response in 45 seconds
config system link-monitor
 edit “1”
  set srcintf “port3”
  set server “8.8.8.8”
  set interval 15000
  set failtime 3
  set recoverytime 2
 next
end
```

```
 next
end
#-----------------------------------------------------------------------------------------
# Set up iBGP router
config router bgp
 set as 65501
 config neighbor
  # AWS-SV-Hub1 neighbor
  edit “10.111.2.1”
   set advertisement-interval 1
   set bfd enable
   set link-down-failover enable
   set next-hop-self enable
   set soft-reconfiguration enable
   set remote-as 65501
   # All neighbors use the same route-map-in
   set route-map-in “RM-PrimaryPath”
   # Each neighbor uses the route-map-out with the localpref number that
   # matches the version of route-map-in being used for that neighbor
   set route-map-out “RM-PrimaryPath”
  next
  edit “10.111.3.1”
   set advertisement-interval 1
   set bfd enable
   set link-down-failover enable
   set next-hop-self enable
   set soft-reconfiguration enable
   set remote-as 65501
   set route-map-in “RM-SecondaryPath”
   set route-map-out “RM-SecondaryPath”
  next
  edit “10.111.4.1”
   set advertisement-interval 1
   set bfd enable
   set link-down-failover enable
   set next-hop-self enable
   set soft-reconfiguration enable
   set remote-as 65501
   set route-map-in “RM-TertiaryPath”
   set route-map-out “RM-TertiaryPath”
  next
  edit “10.111.5.1”
   set advertisement-interval 1
   set bfd enable
   set link-down-failover enable
   set next-hop-self enable
   set soft-reconfiguration enable
   set remote-as 65501
   set route-map-in “RM-QuarternaryPath”
   set route-map-out “RM-QuarternaryPath”
  next
 end
 # Advertise the LAN network out iBGP. You can redistribute connected or
 # an interior routing protocol if needed. Observe great care!!!
 config network
  edit 0
   set prefix 192.168.130.0 255.255.255.0
  next
 end
end
end
```

September 24, 2020 11:00 PM