Deployment Guide
Feb-2018 rev. a

APV Series Deployment Guide for SSL Interception and Security Device Load Balancing

Table of Contents

1 Introduction 5
1.1 Distribution Mode 5
1.2 Deployment Layer 5
1.3 Network Topology 6
1.4 SSL Interception License 6
2 Inbound SSL Interception Deployment 7
2.1 Inbound SSL Interception Deployment in a Reverse Proxy 7
2.1.1 Integrated Mode: One L3 APV + Two L3 AWFs 7
2.1.2 Distributed Mode: Two L3 APVs + Two L3 AWFs 9
2.1.3 Integrated Mode: One L3 APV + Two L2 AWFs 12
2.1.4 Distributed Mode: Two L3 APVs + Two L2 AWFs 14
2.1.5 Integrated Mode: One L2 APV +One L2 AWF 18
2.1.6 Distributed Mode: Two L2 APVs + One L2 AWF 20
2.2 Inbound SSL Interception Deployment in a Forward Proxy 24
2.2.1 Integrated Mode: One L3 APV + Two L3 AWFs 24
2.2.2 Distributed Mode: Two L3 APVs + Two L3 AWFs 27
2.2.3 Integrated Mode: One L3 APV + Two L2 AWFs 30
2.2.4 Distributed Mode: Two L3 APVs + Two L2 AWFs 34
2.2.5 Integrated Mode: One L2 APV + One L2 AWF 37
2.2.6 Distributed Mode: Two L2 APVs + One L2 AWF 40
3 Outbound SSL Interception Deployment 44
3.1 Deployment on APV Working in L3 Mode 44
3.1.1 Integrated Mode: One L3 APV + Two L3 Firewalls 44
3.1.2 Distributed Mode: Two L3 APVs + Two L3 Firewalls 48
3.1.3 Integrated Mode: One L3 APV + Two L2 Firewalls 51
3.1.4 Distributed Mode: Two L3 APVs + Two L2 Firewalls 54
3.2 Deployment on APV Working in L2 Mode 58
3.2.1 Integrated Mode: One L2 APV + One L2 Firewall (With VLAN) 58
3.2.2 Integrated Mode: One L2 APV + One L2 Firewall (Without VLAN) 61
3.2.3 Distributed Mode: Two L2 APVs + One L2 Firewall (With VLAN) 63

3.2.4 Distributed Mode: Two L2 APVs + One L2 Firewall (Without VLAN) 67
4 SSL Interception Integrated with Webagent 71
4.1 Distributed Mode: Three L3 APVs + Two L3 Firewalls 71
4.1.1 Configuring the Webagent 72
4.1.2 Configuring the Ingress Node 72
4.1.3 Configuring the Egress Node 74
4.2 Distributed Mode: Two L3 APVs + Two L3 Firewalls 75
4.2.1 Configuring the Ingress Node 75
4.2.2 Configuring the Egress Node 77
4.3 Integrated Mode: One L3 APV + Two L3 Firewalls 78
4.3.1 Address and Route Settings 78
4.3.2 Load Balance Settings 79
4.3.3 Webagent Service Settings 80
4.3.4 SSL Interception Settings 80
4.4 Distributed Mode: Three L3 APVs +Two L2 Firewalls 81
4.4.1 Configuring the Webagent 81
4.4.2 Configuring the Ingress Node 82
4.4.3 Configuring the Egress Node 83
4.5 Distributed Mode: Two L3 APVs + Two L2 Firewalls 84
4.5.1 Configuring the Ingress Node 84
4.5.2 Configuring the Egress Node 86
4.6 Integrated Mode: One L3 APV + Two L2 Firewalls 87
4.6.1 Address and Route Settings 88
4.6.2 Load Balance Settings 89
4.6.3 Webagent Service Settings 90
4.6.4 SSL Interception Settings 90
5 Dynamic Port Interception (DPI) 91
5.1 Introduction 91
5.2 Configuration Example 91
5.2.1 Integrated Mode: One L3 APV +Two L3 Firewalls 91
5.2.2 Integrated Mode: One L3 APV +Two L2 Firewalls 95
5.2.3 Distributed Mode: Two L3 APVs + Two L3 Firewalls 98
5.2.4 Distributed Mode: Two L3 APVs + Two L2 Firewalls 102

5.2.5 Integrated Mode: One L2 APV + One L2 Firewall (With VLAN) 105
5.2.6 Integrated Mode: One L2 APV + One L2 Firewall (Without VLAN) 108
5.2.7 Distributed Mode: Two L2 APVs + One L2 Firewall (With VLAN) 110
5.2.8 Distributed Mode: Two L2 APVs + One L2 Firewall (Without VLAN) 114
6 WebRoot Website Classification 118
6.1 Filtering Policy 118
6.2 License 119
6.3 Configuration Example 119
6.3.1 Configuring the Ingress Node 120
6.3.2 Configuring the Egress Node 122
7 SPAN Port 124
7.1 Introduction 124
7.2 Configuration Example 124
7.2.1 Integrated Mode: One L3 APV + Four L2 Firewalls (Hybrid) 124
7.2.2 Integrated Mode: One L3 APV + Two L2 Firewalls 129
7.2.3 Distributed Mode: Two L3 APVs + Two L2 Firewalls 131
7.2.4 Integrated Mode: One L2 APV + Two L2 Firewalls 136
7.2.5 Distributed Mode: Two L2 APVs + Two L2 Firewalls 139

1 Introduction

The Array APV Series' SSL interception feature decrypts SSL traffic and then sends the cleartext data to security devices for inspection, such as a firewall, IPS and IDS. This helps prevent attacks, intrusion and data exfiltration caused by insecure information encrypted within SSL data.

In the SSL interception process, the APV appliance plays the role of an ingress node to intercept and decrypt SSL traffic, and the role of an egress node to forward inspected traffic to the real servers. When there are two or more security devices deployed, the APV appliance supports load balancing of decrypted traffic to the security devices.

SSL interception supports a variety of deployment mode combinations based on the interception device's distribution mode, deployment layer and network topology.

1.1 Distribution Mode

In terms of the distribution mode, the ingress and egress nodes can be deployed in the integrated mode or the distributed mode.

- Integrated mode

The ingress and egress nodes are integrated on one APV appliance. That is, one APV plays the role of both the ingress and egress nodes.

- Distributed mode

SSL interception is implemented on two APV appliances. One plays the role of the ingress node, and the other plays the role of the egress node.

1.2 Deployment Layer

The APV appliance can work in L2 or L3 mode when implementing SSL interception.

- L2 mode

L2 mode is also called bridge mode. With a L2 bridge, the APV appliance can function as a L2 device to bridge SSL traffic, transfer it to the upper layer for decryption and then forward the decrypted traffic to the security device for inspection. It forwards packets that are not destined for any of its MAC addresses by looking up its MAC address table.

When the APV appliance works in L2 mode, it can cooperate with one security device working in L2 inline mode to implement SSL interception. If multiple security devices, either of the same type or of different types, need to be deployed for a load balancing or other purpose, the security devices must be deployed in bypass mode and the SPAN Port feature should be configured on the APV appliance. For SPAN Port deployment in implementing SSL interception, please refer to "Chapter 7 SPAN Port".

- L3 mode

L3 mode is also called routing mode. The APV appliance working in L3 mode forwards packets that are not destined for any of its IP addresses by looking up its routing table.

When the APV appliance works in L3 mode, it can cooperate with two or more security devices working in L2 or L3 mode to implement SSL interception. The security devices can be deployed either in bypass mode with SPAN Port deployed as well, or in inline mode.

1.3 Network Topology

Based on networking modes, SSL interception supports inbound networking and outbound networking.

- Inbound

In inbound networking, the APV appliance receives clients' SSL traffic coming from the internet on behalf of the real server. If the APV appliance serves as a reverse proxy server in an inbound networking, it supports both the load balancing of security devices and the load balancing of real application servers while implementing SSL interception.

- Outbound

In an outbound network topology, SSL traffic coming from clients is sent out to the internet via the APV appliance.

1.4 SSL Interception License

Beginning with ArrayOS APV 8.6.1.33, license control is added for the SSL interception feature. To enable SSL interception on ArrayOS APV 8.6.1.33 or later versions, a new APV license with the SSL interception feature must be loaded into the system.
To obtain an APV license enabled with the SSL interception feature, please contact Array Networks Customer Support at support@arraynetworks.com.

2 Inbound SSL Interception Deployment

This chapter provides SSL configuration examples in inbound network topologies.

- 2.1 Inbound SSL Interception Deployment in a Reverse Proxy
  - 2.1.1 Integrated Mode: One L3 APV + Two L3 AWFs
  - 2.1.2 Distributed Mode: Two L3 APVs + Two L3 AWFs
  - 2.1.3 Integrated Mode: One L3 APV + Two L2 AWFs
  - 2.1.4 Distributed Mode: Two L3 APVs + Two L2 AWFs
  - 2.1.5 Integrated Mode: One L2 APV +One L2 AWFs
  - 2.1.6 Distributed Mode: Two L2 APVs + One L2 AWFs
- 2.2 Inbound SSL Interception Deployment in a Forward Proxy
  - 2.2.1 Integrated Mode: One L3 APV + Two L3 AWFs
  - 2.2.2 Distributed Mode: Two L3 APVs + Two L3 AWFs
  - 2.2.3 Integrated Mode: One L3 APV + Two L2 AWFs
  - 2.2.4 Distributed Mode: Two L3 APVs + Two L2 AWFs
  - 2.2.5 Integrated Mode: One L2 APV + One L2 AWFs
  - 2.2.6 Distributed Mode: Two L2 APVs + One L2 AWFs

2.1 Inbound SSL Interception Deployment in a Reverse Proxy

2.1.1 Integrated Mode: One L3 APV + Two L3 AWFs

In this deployment mode, both the APV appliance and the Array AWF web application firewall appliances are set up in L3 mode and the APV appliance serves as both the ingress and egress nodes. The interface and route configurations on the AWF appliances are as shown in the following figure.

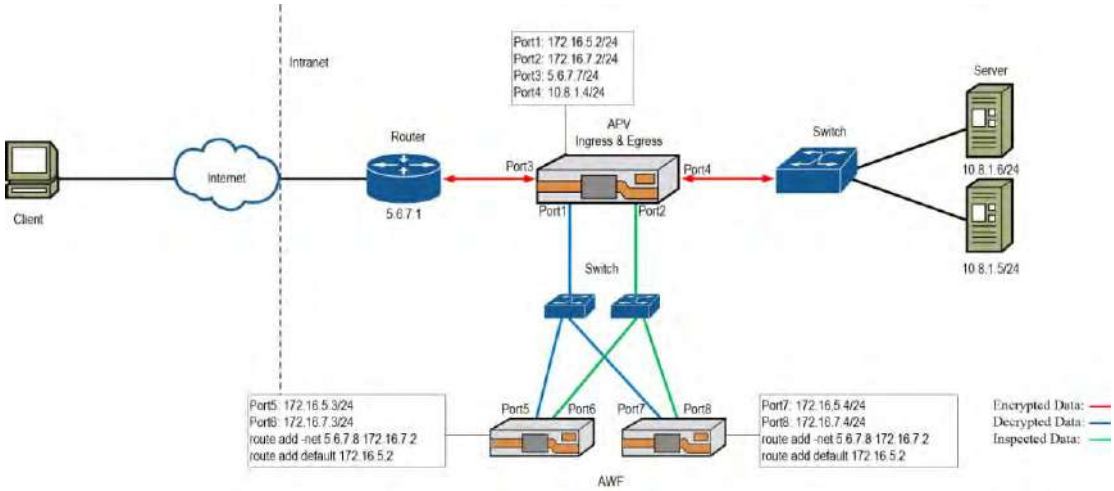

Figure 2–1 Integrated Mode: One L3 APV + Two L3 AWFs

2.1.1.1 Address and Route Settings

1. Set the IP addresses of Port1, Port2, Port3 and Port4.

```
AN(config)#ip address port1 172.16.5.2 24
AN(config)#ip address port2 172.16.7.2 24
AN(config)#ip address port3 5.6.7.7 24
AN(config)#ip address port4 10.8.1.4 24
```

2. Set the default route.

```
AN(config)#ip route default 5.6.7.1
```

2.1.1.2 Load Balance Settings

➢ **Load Balancing of Traffic Received from Clients**

1. Create FWDIP real services.

```
AN(config)#slb real fwdip rs1 172.16.5.3 8080
AN(config)#slb real fwdip rs2 172.16.5.4 8080
```

2. Create a real service group using the chi method and add “rs1” and “rs2” to this group.

```
AN(config)#slb group method chi_group1 chi
AN(config)#slb group member chi_group1 rs1
AN(config)#slb group member chi_group1 rs2
```

3. Create a TCPS virtual service and set Port3 as its serving interface.

```
AN(config)#slb virtual tcps vs1 5.6.7.8 443
AN(config)#slb virtual settings interface vs1 port3
```

4. Configure a default policy to associate “vs1” with “chi_group1”.

```
AN(config)#slb policy default vs1 chi_group1
```

5. Configure health check for “rs1” and “rs2” to ensure that Port2 is accessible (a health check reflector is needed on the egress node).

```
AN(config)#slb real health a1 rs1 172.16.7.2 56789 tcp
AN(config)#slb real health a2 rs2 172.16.7.2 56789 tcp
AN(config)#health ipreflect aa 172.16.7.2 56789 tcp
```

6. Configure health check for “rs1” and “rs2” to check the health status of security devices.

```
AN(config)#slb real health hc_os_h1 rs1 172.16.5.3 0 icmp 3 3
AN(config)#slb real health hc_os_h2 rs2 172.16.5.4 0 icmp 3 3
```

7. Set the relationship among health checks of “rs1” and “rs2” to “and”.

```
AN(config)#health relation rs1 and
AN(config)#health relation rs2 and
```

➢ **Load Balancing of Inspected Traffic to the Real Servers**

1. Create TCPS real services.

```
AN(config)#slb real tcps rs3 10.8.1.5 443 icmp
AN(config)#slb real tcps rs4 10.8.1.6 443 icmp
```

2. Create a real service group using the rr method and add “rs3” and “rs4” to this group.

```
AN(config)#slb group method rr_group2 rr
AN(config)#slb group member rr_group2 rs3
AN(config)#slb group member rr_group2 rs4
```

3. Create a TCP virtual service, enable RTS for it and set Port2 as its serving interface.

```
AN(config)#slb virtual tcp vs2 5.6.7.8 8080 noarp
AN(config)#slb virtual settings rts vs2
AN(config)#slb virtual settings interface vs2 port2
```

4. Configure a default policy to associate “vs2” with “rr_group2”.

```
AN(config)#slb policy default vs2 rr_group2
```

2.1.1.3 SSL Settings

1. Create an SSL virtual host and associate it with “vs1”.

```
AN(config)#ssl host virtual vhost1 vs1
```

2. Import a CA certificate and the private key for “vhost1”, activate the certificate and enable “vhost1”.

```
AN(config)#ssl import key vhost1 1
AN(config)#ssl import certificate vhost1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

3. Create SSL real host “rhost1”, associate it with “rs3” and enable “rhost1”.

```
AN(config)#ssl host real rhost1 rs3
AN(config)#ssl start rhost1
```

2.1.2 Distributed Mode: Two L3 APVs + Two L3 AWFs

In this deployment mode, both the APV appliances and the AWF appliances are set up in L3 mode, and the two APV appliances play the role of the ingress and egress nodes respectively. The interface and route configurations on the AWF appliances are as shown in the following figure.

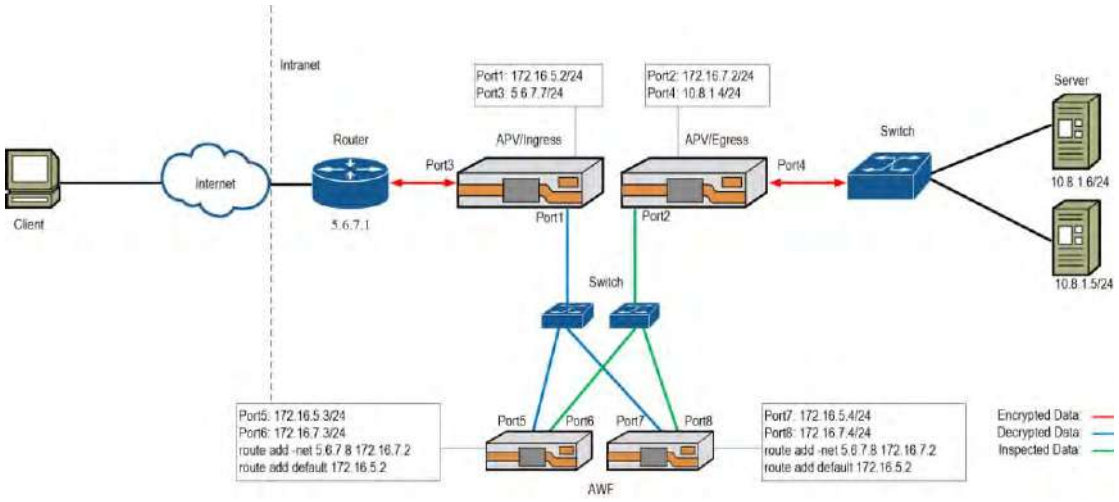

Figure 2–2 Distributed Mode: Two L3 APVs + Two L3 AWFs

2.1.2.1 Configuring the Ingress Node

2.1.2.1.1 Address and Route Settings

1. Set the IP addresses of Port1 and Port3.

```
AN(config)#ip address port1 172.16.5.2 24
AN(config)#ip address port3 5.6.7.7 24
```

2. Set the default route.

```
AN(config)#ip route default 5.6.7.1
```

2.1.2.1.2 Load Balance Settings

1. Create FWDIP real services.

```
AN(config)#slb real fwdip rs1 172.16.5.3 8080
AN(config)#slb real fwdip rs2 172.16.5.4 8080
```

2. Create a real service group using the chi method and add “rs1” and “rs2” to this group.

```
AN(config)#slb group method chi_group chi
AN(config)#slb group member chi_group rs1
AN(config)#slb group member chi_group rs2
```

3. Create a TCPS virtual service.

```
AN(config)#slb virtual tcps vs1 5.6.7.8 443
```

4. Configure a default policy to associate “vs1” with “chi_group”.

```
AN(config)#slb policy default vs1 chi_group
```

5. Configure health check for “rs1” and “rs2” to ensure that Port2 on the egress node is accessible (a health check reflector is needed on the egress node).

```
AN(config)#slb real health a1 rs1 172.16.7.2 56789 tcp
AN(config)#slb real health a2 rs2 172.16.7.2 56789 tcp
```

6. Configure health check for “rs1” and “rs2” to check the health status of security devices.

```
AN(config)#slb real health hc_os_h1 rs1 172.16.5.3 0 icmp 3 3
AN(config)#slb real health hc_os_h2 rs2 172.16.5.4 0 icmp 3 3
```

7. Set the relationship among health checks of “rs1” and “rs2” to “and”.

```
AN(config)#health relation rs1 and
AN(config)#health relation rs2 and
```

2.1.2.1.3 SSL Settings

1. Create an SSL virtual host and associate it with “vs1”.

```
AN(config)#ssl host virtual vhost1 vs1
```

2. Import a CA certificate and the private key for “vhost1”, activate the certificate and enable “vhost1”.

```
AN(config)#ssl import key vhost1 1
AN(config)#ssl import certificate vhost1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

2.1.2.2 Configuring the Egress Node

2.1.2.2.1 Address and Route Settings

1. Set the IP addresses of Port2 and Port4.

```
AN(config)#ip address port2 172.16.7.2 24
AN(config)#ip address port4 10.8.1.4 24
```

2.1.2.2.2 Load Balance Settings

1. Create TCPS real services.

```
AN(config)#slb real tcps rs3 10.8.1.5 443 icmp
AN(config)#slb real tcps rs4 10.8.1.6 443 icmp
```

2. Create a real service group using the rr method and add “rs3” and “rs4” to this group.

```
AN(config)#slb group method rr_group2 rr
```

```
AN(config)#slb group member rr_group2 rs3
AN(config)#slb group member rr_group2 rs4
```

3. Create a TCP virtual service and enable RTS for it.

```
AN(config)#slb virtual tcp vs2 5.6.7.8 8080 noarp
AN(config)#slb virtual settings rts vs2
```

4. Configure a default policy to associate “vs2” with “rr_group2”.

```
AN(config)#slb policy default vs2 rr_group2
```

5. Configure a health check reflector.

```
AN(config)#health ipreflect aa 172.16.7.2 56789 tcp
```

2.1.2.2.3 SSL Settings

1. Create SSL real host “rhost1”, associate it with “rs3” and enable “rhost1”.

```
AN(config)#ssl host real rhost1 rs3
AN(config)#ssl start rhost1
```

2.1.3 Integrated Mode: One L3 APV + Two L2 AWFs

In this deployment mode, the APV appliance is set up in L3 mode, and the AWF appliances are set up in L2 mode. The APV appliance serves as both the ingress and egress nodes. The interface and route configurations on the AWF appliances are as shown in the following figure.

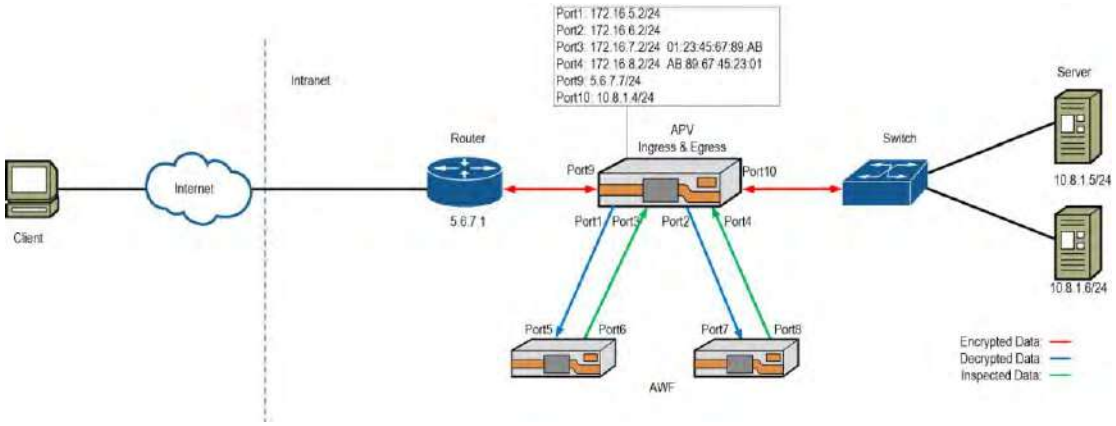

Figure 2–3 Integrated Mode: One L3 APV + Two L2 AWFs

2.1.3.1 Address and Route Settings

1. Set the IP addresses of Port1, Port2, Port3, Port4, Port9 and Port10.

```
AN(config)#ip address port1 172.16.5.2 24
AN(config)#ip address port2 172.16.6.2 24
AN(config)#ip address port3 172.16.7.2 24
```

```
AN(config)#ip address port4 172.16.8.2 24
AN(config)#ip address port9 5.6.7.7 24
AN(config)#ip address port10 10.8.1.4 24
```

2. Set the MAC addresses.

```
AN(config)#interface mac port3 01:23:45:67:89:AB
AN(config)#interface mac port4 AB:89:67:45:23:01
```

3. Set the default route.

```
AN(config)#ip route default 5.6.7.1
```

2.1.3.2 Load Balance Settings

- **Load Balancing of Traffic Received from Clients**

1. Create FWDMAC real services.

```
AN(config)#slb real fwdmac rs1 port1 01:23:45:67:89:AB 8080
AN(config)#slb real fwdmac rs2 port2 AB:89:67:45:23:01 8080
```

2. Create a real service group using the chi method and add “rs1” and “rs2” to this group.

```
AN(config)#slb group method chi_group chi
AN(config)#slb group member chi_group rs1
AN(config)#slb group member chi_group rs2
```

3. Create a TCPS virtual service.

```
AN(config)#slb virtual tcps vs1 5.6.7.8 443
```

4. Configure a default policy to associate “vs1” with “chi_group”.

```
AN(config)#slb policy default vs1 chi_group
```

5. Configure health checks for “rs1” and “rs2 to ensure that Port3 and Port4 are accessible.

```
AN(config)#slb real health a1 rs1 172.16.7.2 56789 tcp
AN(config)#slb real health a2 rs2 172.16.8.2 56789 tcp
AN(config)#health ipreflect aa 0.0.0.0 56789 tcp
```

- **Load Balancing of Inspected Traffic to the Real Servers**

1. Create TCPS real services.

```
AN(config)#slb real tcps rs3 10.8.1.5 443 icmp
AN(config)#slb real tcps rs4 10.8.1.6 443 icmp
```

2. Create a real service group using the rr method and add “rs3” and “rs4” to this group.

```
AN(config)#slb group method rr_group2 rr
AN(config)#slb group member rr_group2 rs3
```

```
AN(config)#slb group member rr_group2 rs4
```

3. Create a TCP virtual service, enable RTS for it and set Port3 and Port4 as its serving interfaces.

```
AN(config)#slb virtual tcp vs2 5.6.7.8 8080 noarp
AN(config)#slb virtual settings rts vs2
AN(config)#slb virtual settings interface vs2 port3
AN(config)#slb virtual settings interface vs2 port4
```

4. Configure a default policy to associate “vs2” with “rr_group2”.

```
AN(config)#slb policy default vs2 rr_group2
```

2.1.3.3 SSL Settings

1. Create an SSL virtual host and associate it with “vs1”.

```
AN(config)#ssl host virtual vhost1 vs1
```

2. Import a CA certificate and the private key for “vhost1”, activate the certificate and enable “vhost1”.

```
AN(config)#ssl import key vhost1 1
AN(config)#ssl import certificate vhost1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

3. Create an SSL real host “rhost1”, associate it with “rs3” and enable “rhost1”.

```
AN(config)#ssl host real rhost1 rs3
AN(config)#ssl start rhost1
```

2.1.4 Distributed Mode: Two L3 APVs + Two L2 AWFs

In this deployment mode, the APV appliances are set up in L3 mode, and the AWF appliances are set up in L2 mode. The two APV appliances play the role of the ingress and egress nodes respectively. The interface and route configurations on the AWF appliances are as shown in the following figure.

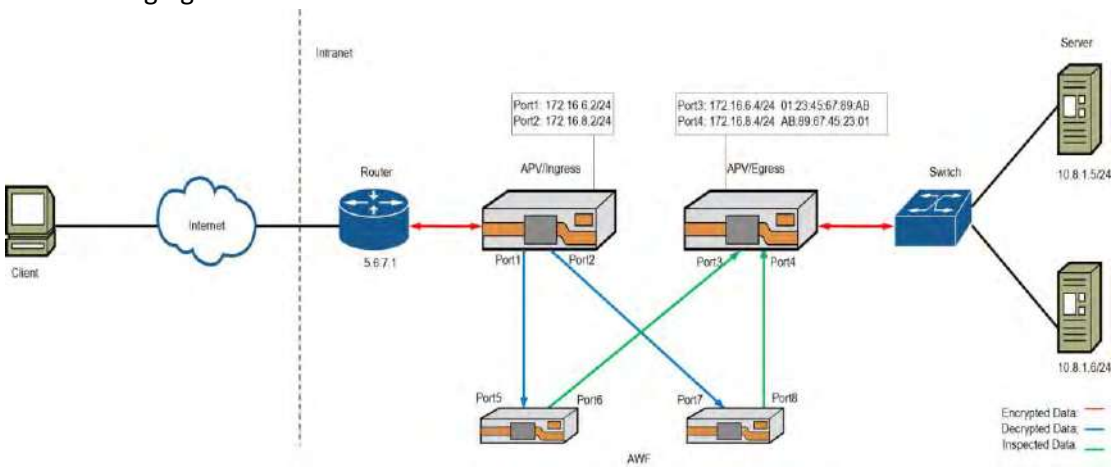

Figure 2–4 Distributed Mode: Two L3 APVs + Two L2 AWFs

2.1.4.1 Configuring the Ingress Node

2.1.4.1.1 Address and Route Settings

1. Set the IP addresses of Port1 and Port2.

```
AN(config)#ip address port1 172.16.6.2 24
AN(config)#ip address port2 172.16.8.2 24
```

2. Set the MAC addresses.

```
AN(config)#interface mac port3 01:23:45:67:89:AB
AN(config)#interface mac port4 AB:89:67:45:23:01
```

3. Set the default route.

```
AN(config)#ip route default 5.6.7.1
```

2.1.4.1.2 Load Balance Settings

1. Create FWDMAC real services.

```
AN(config)#slb real fwdmac rs1 port1 01:23:45:67:89:AB 8080
AN(config)#slb real fwdmac rs2 port2 AB:89:67:45:23:01 8080
```

2. Create a real service group using the chi method and add “rs1” and “rs2” to this group.

```
AN(config)#slb group method chi_group1 chi
AN(config)#slb group member chi_group1 rs1
AN(config)#slb group member chi_group1 rs2
```

3. Create a TCPS virtual service.

```
AN(config)#slb virtual tcps vs1 5.6.7.8 443
```

4. Configure a default policy to associate “vs1” with “chi_group1”.

```
AN(config)#slb policy default vs1 chi_group1
```

5. Configure health checks for “rs1” and “rs2 to ensure that Port3 and Port4 are accessible (a health check reflector is needed on the egress node).

```
AN(config)#slb real health a1 rs1 172.16.6.4 56789 tcp
AN(config)#slb real health a2 rs2 172.16.8.4 56789 tcp
```

2.1.4.1.3 SSL Settings

1. Create an SSL virtual host and associate it with “vs1”.

```
AN(config)#ssl host virtual vhost1 vs1
```

2. Import a CA certificate and the private key for “vhost1”, activate the certificate and enable “vhost1”.

```
AN(config)#ssl import key vhost1 1
AN(config)#ssl import certificate vhost1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

2.1.4.2 Configuring the Egress Node

2.1.4.2.1 Address and Route Settings

1. Set the IP addresses of Port3 and Port4.

```
AN(config)#ip address port3 172.16.6.4 24
AN(config)#ip address port4 172.16.8.4 24
```

2. Set the MAC addresses.

```
AN(config)#interface mac port3 01:23:45:67:89:AB
AN(config)#interface mac port4 AB:89:67:45:23:01
```

2.1.4.2.2 Load Balance Settings

1. Create TCPS real services.

```
AN(config)#slb real tcps rs3 10.8.1.5 443 icmp
AN(config)#slb real tcps rs4 10.8.1.6 443 icmp
```

2. Create a real service group using the rr method and add “rs3” and “rs4” to this group.

```
AN(config)#slb group method rr_group2 rr
AN(config)#slb group member rr_group2 rs3
AN(config)#slb group member rr_group2 rs4
```

3. Create a TCP virtual service and enable RTS for it.

```
AN(config)#slb virtual tcp vs2 5.6.7.8 8080 noarp
AN(config)#slb virtual settings rts vs2
```

4. Configure a default policy to associate “vs2” with “rr_group2”.

```
AN(config)#slb policy default vs2 rr_group2
```

5. Configure a health check reflector.

```
AN(config)#health ipreflect aa 0.0.0.0 56789 tcp
```

2.1.4.2.3 SSL Settings

1. Create SSL real host “rhost1”, associate it with “rs3” and enable “rhost1”.

```
AN(config)#ssl host real rhost1 rs3
AN(config)#ssl start rhost1
```

2.1.5 Integrated Mode: One L2 APV +One L2 AWF

In this deployment mode, the APV appliance and the AWF appliance are set up in L2 mode, and the APV appliance serves as both the ingress and egress nodes. Two bridge instances are configured on the APV appliance:

- One bridge is used to transfer SSL traffic to the SSL interception module for decryption and then forward the decrypted traffic to the AWF appliance.
- The other bridge is used to receive the inspected traffic from the AWF appliance and re-encrypt the traffic before sending it on to the servers.

The network topology and interface configurations are as shown in the following figure.

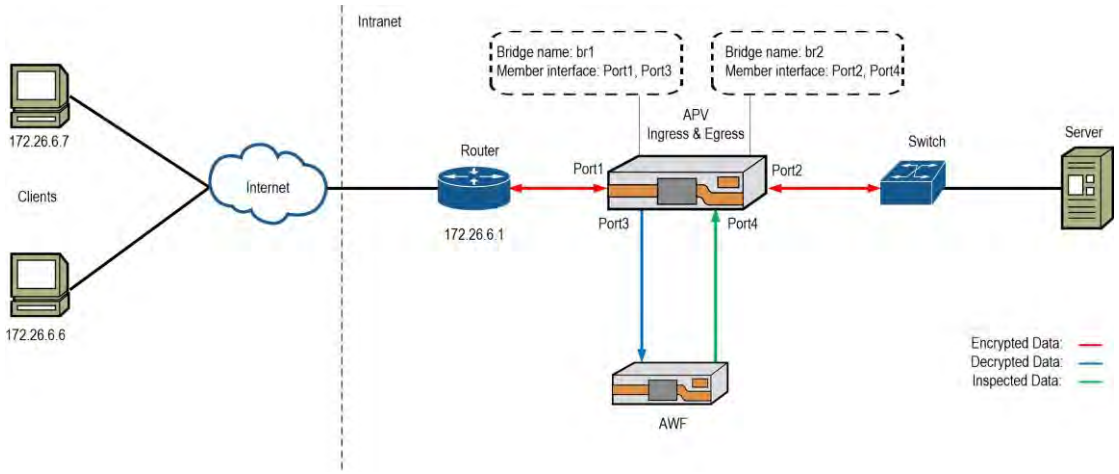

Figure 2–5 Integrated Mode: One L2 APV +One L2 AWF

2.1.5.1 Bridge Settings

1. Create two bridge instances.

```
AN(config)#bridge name br1
AN(config)#bridge name br2
```

2. Add members to the created bridge instances.

```
AN(config)#bridge member br1 port1 yes
AN(config)#bridge member br1 port3 yes
AN(config)#bridge member br2 port2 yes
AN(config)#bridge member br2 port4 yes
```

3. Create filter rules to bypass returned SSL traffic and to acquire server certificates.

```
AN(config)#bridge apprule br1 0.0.0.0 443 0.0.0.0 0 tcp
```

4. Create filter rules to forward all SSL traffic (including encrypted and cleartext traffic) to the SSL interception module.

```
AN(config)#bridge apprule br1 0.0.0.0 0 0.0.0.0 443 tcp
AN(config)#bridge apprule br1 0.0.0.0 8443 0.0.0.0 0 tcp
```

```
AN(config)#bridge apprule br2 0.0.0.0 443 0.0.0.0 0 tcp
AN(config)#bridge apprule br2 0.0.0.0 0 0.0.0.0 8443 tcp
```

2.1.5.2 SSL Settings

2.1.5.2.1 Forwarding of Received SSL Traffic to the Security Device

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Create a FWDMAC real service.

   Note that “AB:89:67:45:23:01” does not represent any port. It can be replaced with an arbitrary MAC address, but it must be set.

```
AN(config)#slb real fwdmac rs1 port3 AB:89:67:45:23:01 8443
```

3. Create a real service group using the chi method and add “rs1” to this group.

```
AN(config)#slb group method chi_group chi
AN(config)#slb group member chi_group rs1
```

4. Create a TCPS virtual service.

```
AN(config)#slb virtual tcps vs1 0.0.0.0 443 noarp 0
```

5. Configure a default policy to associate “vs1” with “chi_group”.

```
AN(config)#slb policy default vs1 chi_group
```

6. Create an SSL virtual host and associate it with “vs1”.

```
AN(config)#ssl host virtual vhost1 vs1
```

2.1.5.2.2 Forwarding of Inspected SSL Traffic to the Real Service

1. Create a TCPS real service, and configure it to keep destination IP addresses unchanged when forwarding packets.

   Note that “172.26.6.1” can be replaced with an arbitrary IP address.

```
AN(config)#slb real tcps rs2 172.26.6.1 443 none
AN(config)#slb real settings keepdip rs2
```

2. Create a TCP virtual service.

```
AN(config)#slb virtual tcp vs2 0.0.0.0 8443 noarp 0
```

3. Configure a static policy to associate “vs2” with “rs2”.

```
AN(config)#slb policy static vs2 rs2
```

4. Disable real service health check.

```
AN(config)#health off
```

5. Create an SSL real host “rhost1” and associate it with “rs2”.

```
AN(config)#ssl host real rhost1 rs2
```

2.1.5.3 SSL Interception Settings

1. Enable SSL interception for “vhost1”.

```
AN(config)#ssli on vhost1 1
```

2. Generate SSL interception certificates for “vhost1”, activate them and enable “vhost1”.

Note: The generated CA certificates must also be imported into the trusted CA list of the trusted CA list of clients’ browsers.

```
AN(config)#ssli cacert rsa vhost1 2048 1 1
AN(config)#ssli cacert ecc vhost1 prime256v1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

Note: Currently, elliptic curve secp521r1 is not widely supported by mainstream browsers. It is recommended to set prime256v1 or secp384r1 to avoid connection failures.

3. Enable SSL interception for “rhost1” and enable “rhost1”.

```
AN(config)#ssli on rhost1 1
AN(config)#ssl start rhost1
```

Note: If the SSL server uses a testing certificate or a self-signed certificate, both the ingress and egress nodes must have its certificate chain (including the root CA certificate) imported. For testing-only purposes, administrators can choose not to import the certificate chain, but must execute the “**ssl globals verifycert off**” command to disable the server authentication function.

2.1.6 Distributed Mode: Two L2 APVs + One L2 AWF

In this deployment mode, both the APV appliances and the AWF appliance are set up in L2 mode, and the two APV appliances play the role of the ingress and egress nodes respectively. The ingress node and the egress node each have a bridge instance configured:

- The bridge on the ingress node is used to transfer SSL traffic to the SSL interception module for decryption and then forward the decrypted traffic to the AWF appliance.
- The bridge on the egress node is used to receive the inspected traffic from the AWF appliance and re-encrypt the traffic before sending it on to the servers.

The network topology and interface are as shown in the following figure.

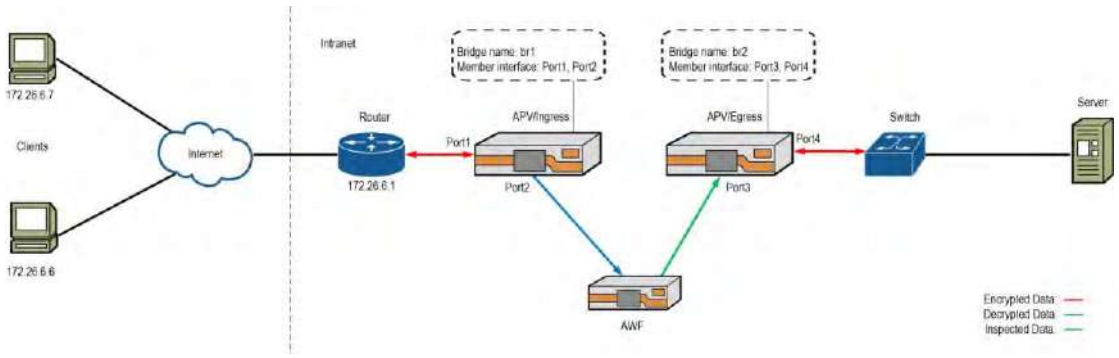

Figure 2–6 Distributed Model: Two L2 APVs + One L2 AWF

2.1.6.1 Configuring the Ingress Node

2.1.6.1.1 Bridge Settings

1. Create a bridge instance.

```
AN(config)#bridge name br1
```

2. Add members to the created bridge instance.

```
AN(config)#bridge member br1 port1 yes
AN(config)#bridge member br1 port2 yes
```

3. Create filter rules to bypass returned SSL traffic and to acquire server certificates.

```
AN(config)#bridge apprule br1 0.0.0.0 443 0.0.0.0 0 tcp
```

4. Create filter rules to forward clients' encrypted SSL traffic and servers' cleartext SSL traffic to the SSL interception module.

```
AN(config)#bridge apprule br1 0.0.0.0 0 0.0.0.0 443 tcp
AN(config)#bridge apprule br1 0.0.0.0 8443 0.0.0.0 0 tcp
```

2.1.6.1.2 SSL Settings

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Create a TCPS virtual service.

```
AN(config)#slb virtual tcps vs1 0.0.0.0 443 noarp 0
```

3. Create a FWDMAC real service.

   Note that "AB:89:67:45:23:01" does not represent any port. It can be replaced with an arbitrary MAC address, but it must be set.

```
AN(config)#slb real fwdmac rs1 port2 AB:89:67:45:23:01 8443
```

4. Create an L2 real service group using the chi method and add “rs1” to this group.

```
AN(config)#slb group method chi_group chi
AN(config)#slb group member chi_group rs1
```

5. Configure a default policy to associate “vs1” with “chi_group”.

```
AN(config)#slb policy default vs1 chi_group
```

6. Create an SSL virtual host and associate it with “vs1”.

```
AN(config)#ssl host virtual vhost1 vs1
```

2.1.6.1.3 SSL Interception Settings

1. Enable SSL interception for “vhost1”.

```
AN(config)#ssli on vhost1 0
```

2. Generate SSL interception certificates for “vhost1”, activate them and enable “vhost1”.

```
AN(config)#ssli cacert rsa vhost1 2048 1 1
AN(config)#ssli cacert ecc vhost1 prime256v1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

Note:

1. The generated CA certificates must also be imported into the trusted CA list of clients’ browsers.
2. If the SSL server uses a testing certificate or a self-signed certificate, both the ingress and egress nodes must have its certificate chain (including the root CA certificate) imported. For testing-only purposes, administrators can choose not to import the certificate chain, but must execute the “**ssl globals verifycert off**” command to disable the server authentication function.
3. Currently, elliptic curve secp521r1 is not widely supported by mainstream browsers. It is recommended to set prime256v1 or secp384r1 to avoid connection failures.

2.1.6.2 Configuring the Egress Node

2.1.6.2.1 Bridge Settings

1. Create a bridge instance.

```
AN(config)#bridge name br2
```

2. Add members to the created bridge instance.

```
AN(config)#bridge member br2 port3 yes
```

```
AN(config)#bridge member br2 port4 yes
```

3. Create filter rules to forward servers' encrypted SSL traffic and clients' cleartext SSL traffic to the SSL interception module.

```
AN(config)#bridge apprule br2 0.0.0.0 443 0.0.0.0 0 tcp
AN(config)#bridge apprule br2 0.0.0.0 0 0.0.0.0 8443 tcp
```

2.1.6.2.2 SSL Settings

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Configure a TCPS real service, and configure it to keep destination IP addresses unchanged when forwarding packets.

   Note that "172.26.6.1" can be replaced with an arbitrary IP address.

```
AN(config)#slb real tcps rs2 172.26.6.1 443 none
AN(config)#slb real settings keepdip rs2
```

3. Configure a TCP virtual service.

```
AN(config)#slb virtual tcp vs2 0.0.0.0 8443 noarp 0
```

4. Configure a static policy to associate "vs2" with "rs2".

```
AN(config)#slb policy static vs2 rs2
```

5. Disable the real service health check.

```
AN(config)#health off
```

6. Create an SSL real host "rhost1" and associate it with "rs2".

```
AN(config)#ssl host real rhost1 rs2
```

2.1.6.2.3 SSL Interception Settings

7. Enable SSL interception for "rhost1" and enable "rhost1".

```
AN(config)#ssli on rhost1 0
AN(config)#ssl start rhost
```

Note: If the SSL server uses a testing certificate or a self-signed certificate, both the ingress and egress nodes must have its certificate chain (including the root CA certificate) imported. For testing-only purposes, administrators can choose not to import the certificate chain, but must execute the "**ssl globals verifycert off**" command to disable the server authentication function.

2.2 Inbound SSL Interception Deployment in a Forward Proxy

2.2.1 Integrated Mode: One L3 APV + Two L3 AWFs

In this deployment mode, the AWF appliances and the APV appliance are set up in L3 mode, and the APV appliance serves as both the ingress and egress nodes. The interface and route configurations on the AWF appliances are as shown in the following figure.

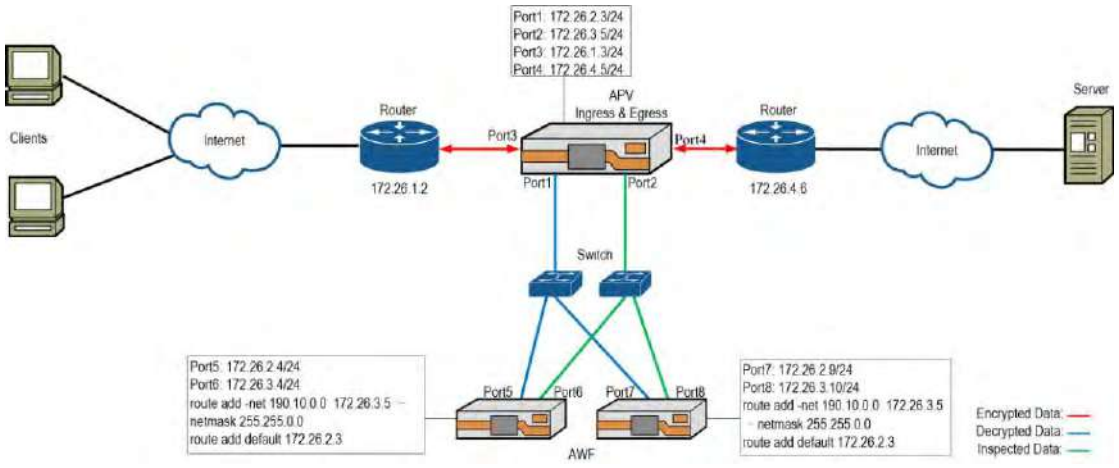

Figure 2–7 Integrated Mode: One L3 APV + Two L3 AWF

2.2.1.1 Address and Route Settings

1. Set the IP addresses of Port1, Port2, Port3 and Port4.

```
AN(config)#ip address port1 172.26.2.3 24
AN(config)#ip address port2 172.26.3.5 24
AN(config)#ip address port3 172.26.1.3 24
AN(config)#ip address port4 172.26.4.5 24
```

2. Set the default route.

```
AN(config)#ip route default 172.26.1.2
```

3. Define an Eroute.

```
AN(config)#ip eroute er1 1900 0.0.0.0 0.0.0.0 0 190.10.0.0 255.255.0.0 0 any 172.26.4.6
```

2.2.1.2 Load Balance Settings

➢ **Load Balancing of SSL Traffic Received from Clients**

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Create FWDIP real services.

```
AN(config)#slb real fwdip rs1 172.26.2.4 8443
AN(config)#slb real fwdip rs2 172.26.2.9 8443
```

3. Create a real service group using the hi method and add “rs1” and “rs2” to this group.

```
AN(config)#slb group method hi_group hi
AN(config)#slb group member hi_group rs1
AN(config)#slb group member hi_group rs2
```

4. Create a TCPS virtual service.

```
AN(config)#slb virtual tcps vs1 0.0.0.0 443 noarp 0
```

5. Configure a default policy to associate “vs1” with “hi_group”.

```
AN(config)#slb policy default vs1 hi_group
```

6. Configure health checks for “rs1” and “rs2” to ensure that Port2 is accessible.

```
AN(config)#slb real health a1 rs1 172.26.3.5 56789 tcp 3 3
AN(config)#slb real health a2 rs2 172.26.3.5 56789 tcp 3 3
AN(config)#health ipreflect aa 172.26.3.5 56789 tcp
```

7. Configure health checks for “rs1” and “rs2” to check the health status of the security devices.

```
AN(config)#slb real health hc_os_h1 rs1 172.26.2.4 0 icmp 3 3
AN(config)#slb real health hc_os_h2 rs2 172.26.2.9 0 icmp 3 3
```

8. Set the relationship among health checks of “rs1” and “rs2” to “and”.

```
AN(config)#health relation rs1 and
AN(config)#health relation rs2 and
```

➢ **Load Balancing of Non-SSL Traffic Received from Clients**

1. Create L2IP real services.

```
AN(config)#slb real l2ip rs4 172.26.2.4
AN(config)#slb real l2ip rs5 172.26.2.9
```

2. Create an L2 real service group using the chi method, set the route mode to “direct” and add “rs4” and “rs5” to this group.

```
AN(config)#slb group method chi_group1 chi direct
AN(config)#slb group member chi_group1 rs4
AN(config)#slb group member chi_group1 rs5
```

3. Create an L2IP virtual service.

```
AN(config)#slb virtual l2ip l2ip_vs1 172.26.1.3 172.26.1.2
```

4. Configure a default policy to associate “l2ip_vs1” with “chi_group1”.

```
AN(config)#slb policy default l2ip_vs1 chi_group1
```

5. Configure two port ranges for “l2ip_vs1”.

```
AN(config)#slb virtual portrange l2ip_vs1 0 442 all dst
AN(config)#slb virtual portrange l2ip_vs1 444 65535 all dst
```

6. Configure two port ranges for “chi_group1”.

```
AN(config)#slb group option portrange chi_group1 0 8442 all src
AN(config)#slb group option portrange chi_group1 8444 65535 all src
```

➢ **Forwarding of Inspected Traffic to the Real Server**

1. Create a TCP virtual service and enable RTS for it.

```
AN(config)#slb virtual tcp vs2 0.0.0.0 8443 noarp 0
AN(config)#slb virtual settings rts vs2
```

2. Create a TCPS real service and configure it to keep destination IP addresses unchanged when forwarding packets.

```
AN(config)#slb real tcps rs3 172.26.4.6 443 icmp
AN(config)#slb real settings keepdip rs3
```

3. Configure a static policy to associate “vs2” with “rs3”.

```
AN(config)#slb policy static vs2 rs3
```

➢ **Load Balancing of Non-SSL Traffic Returned from the Real Server**

1. Create L2IP real services.

```
AN(config)#slb real l2ip rs6 172.26.3.4
AN(config)#slb real l2ip rs7 172.26.3.10
```

2. Create an L2 real service group using the chi method, and add “rs6” and “rs7” to this group.

```
AN(config)#slb group method chi_group2 chi route
AN(config)#slb group member chi_group2 rs6
AN(config)#slb group member chi_group2 rs7
```

3. Create an L2IP virtual service.

```
AN(config)#slb virtual l2ip l2ip_vs2 172.26.4.5 172.26.4.6
```

4. Configure a default policy to associate “l2ip_vs2” with “chi_group2”.

```
AN(config)#slb policy default l2ip_vs2 chi_group2
```

5. Configure two port ranges for “l2ip_vs2”.

```
AN(config)#slb virtual portrange l2ip_vs2 0 442 all src
AN(config)#slb virtual portrange l2ip_vs2 444 65535 all src
```

6. Configure two port ranges for “chi_group2”.

```
AN(config)#slb group option portrange chi_group2 0 8442 all dst
```

```
AN(config)#slb group option portrange chi_group2 8444 65535 all dst
```

2.2.1.3 SSL Interception Settings

1. Create an SSL virtual host and associate it with “vs1”.

```
AN(config)#ssl host virtual vhost1 vs1
```

2. Enable SSL interception for “vhost1”.

```
AN(config)#ssli on vhost1 1
```

3. Generate SSL interception certificates for “vhost1”, activate them, and enable “vhost1”.

Note: The generated CA certificates must also be imported into the trusted CA list of clients’ browsers.

```
AN(config)#ssli cacert rsa vhost1 2048 1 1
AN(config)#ssli cacert ecc vhost1 prime256v1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

Note: Currently, elliptic curve secp521r1 is not widely supported by mainstream browsers. It is recommended to set prime256v1 or secp384r1 to avoid connection failures.

4. Create an SSL real host “rhost1” and associate it with “rs3”.

```
AN(config)#ssl host real rhost1 rs3
```

5. Enable SSL interception for “rhost1” and enable “rhost1”.

```
AN(config)#ssli on rhost1 1
AN(config)#ssl start rhost1
```

2.2.2 Distributed Mode: Two L3 APVs + Two L3 AWFs

In this deployment mode, the AWF appliances and APV appliances are set up in L3 mode, and the two APV appliances play the role of the ingress and egress nodes respectively. The interface and route configurations on the AWF appliances are as shown in the following figure.

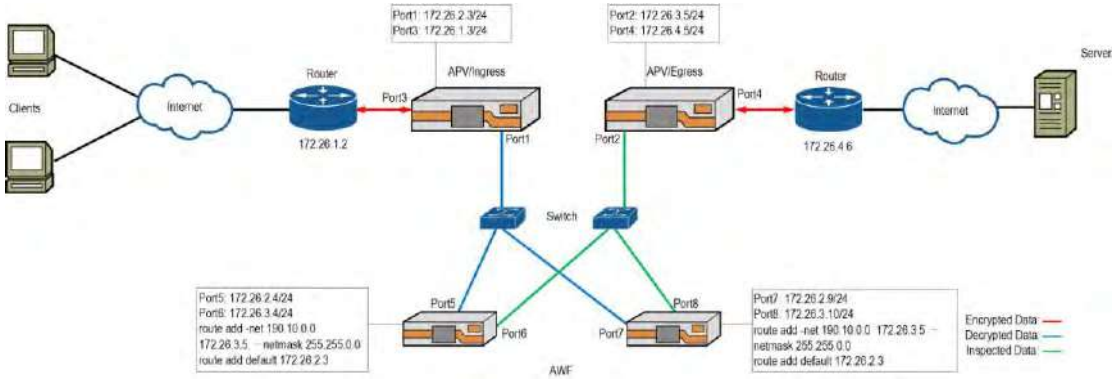

Figure 2–8 Distributed Mode: Two L3 APVs + Two L3 AWFs

2.2.2.1 Configuring the Ingress Node

2.2.2.1.1 Address and Route Settings

1. Set the IP addresses of Port1 and Port3.

```
AN(config)#ip address port1 172.26.2.3 24
AN(config)#ip address port3 172.26.1.3 24
```

2. Set the default route.

```
AN(config)#ip route default 172.26.1.2
```

3. Define Eroutes.

```
AN(config)#ip eroute er1 1900 190.10.0.0 255.255.0.0 0 0.0.0.0 0.0.0.0 172.26.2.4
AN(config)#ip eroute er2 1900 190.10.0.0 255.255.0.0 0 0.0.0.0 0.0.0.0 172.26.2.9
```

4. Enable the IPflow function.

```
AN(config)#ip ipflow on
```

2.2.2.1.2 Load Balance Settings

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Create FWDIP real services.

```
AN(config)#slb real fwdip rs1 172.26.2.4 8443
AN(config)#slb real fwdip rs2 172.26.2.9 8443
```

3. Create a real service group using the chi method and add FWDIP real services to this group.

```
AN(config)#slb group method chi_group chi
AN(config)#slb group member chi_group rs1
AN(config)#slb group member chi_group rs2
```

4. Create a TCPS virtual service.

```
AN(config)#slb virtual tcps vs1 0.0.0.0 443 noarp 0
```

5. Configure a default policy to associate “vs1” with “chi_group”.

```
AN(config)#slb policy default vs1 chi_group
```

6. Configure health checks for “rs1” and “rs2” to ensure that Port2 on the egress node is accessible (a health check reflector is needed on the egress node).

```
AN(config)#slb real health a1 rs1 172.26.3.5 56789 tcp 3 3
AN(config)#slb real health a2 rs2 172.26.3.5 56789 tcp 3 3
```

7. Configure health checks for “rs1” and “rs2” to check the health status of security devices.

```
AN(config)#slb real health hc_os_h1 rs1 172.26.2.4 0 icmp 3 3
AN(config)#slb real health hc_os_h2 rs2 172.26.2.9 0 icmp 3 3
```

8. Set the relationship among health checks of “rs1” and “rs2” to “and”.

```
AN(config)#health relation rs1 and
AN(config)#health relation rs2 and
```

2.2.2.1.3 SSL Interception Settings

1. Create an SSL virtual host and associate it with “vs1”.

```
AN(config)#ssl host virtual vhost1 vs1
```

2. Enable SSL interception for “vhost1”.

```
AN(config)#ssli on vhost1 0
```

3. Generate SSL interception certificates for “vhost1”, activate them, and enable “vhost1”.

Note: The generated CA certificates must also be imported into the trusted CA list of clients’ browsers.

```
AN(config)#ssli cacert rsa vhost1 2048 1 1
AN(config)#ssli cacert ecc vhost1 prime256v1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

Note: Currently, elliptic curve secp521r1 is not widely supported by mainstream browsers. It is recommended to set prime256v1 or secp384r1 to avoid connection failures.

2.2.2.2 Configuring the Egress Node

2.2.2.2.1 Address and Route Settings

1. Set the IP addresses of Port2 and Port4.

```
AN(config)#ip address port2 172.26.3.5 24
AN(config)#ip address port4 172.26.4.5 24
```

2. Define eroutes to the server.

```
AN(config)#ip eroute er1 1900 0.0.0.0 0.0.0.0 0 190.10.0.0 255.255.0.0 0 172.26.4.6
```

3. Enable RTS.

```
AN(config)#ip rts on
```

4. Define Eroutes to the clients.

```
AN(config)#ip eroute er2 1900 190.10.0.0 255.255.0.0 0 0.0.0.0 0.0.0.0 0 172.26.3.4
AN(config)#ip eroute er3 1900 190.10.0.0 255.255.0.0 0 0.0.0.0 0.0.0.0 0 172.26.3.10
```

2.2.2.2.2 Load Balance Settings

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Configure a TCPS real service, and configure it to keep destination IP addresses unchanged when forwarding packets.

```
AN(config)#slb real tcps rs3 172.26.4.6 443 icmp
AN(config)#slb real settings keepdip rs3
```

3. Configure a TCP virtual service and enable RTS for it.

```
AN(config)#slb virtual tcp vs2 0.0.0.0 8443 noarp 0
AN(config)#slb virtual settings rts vs2
```

4. Configure a static policy to associate “vs2” with “rs3”.

```
AN(config)#slb policy static vs2 rs3
```

5. Create a health check reflector “reflector1”.

```
AN(config)#health ipreflect reflector1 172.26.3.5 56789 tcp
```

2.2.2.2.3 SSL Interception Settings

1. Create an SSL real host “rhost1” and associate it with “rs3”.

```
AN(config)#ssl host real rhost1 rs3
```

2. Enable SSL interception for “rhost1” and enable “rhost1”.

```
AN(config)#ssli on rhost1 0
AN(config)#ssl start rhost1
```

2.2.3 Integrated Mode: One L3 APV + Two L2 AWFs

In this deployment mode, the APV appliance is set up in L3 mode, AWF appliances are set up in L2 mode, and the APV appliance serves as both the ingress and egress nodes. The interface and route configurations on the AWF appliances are as shown in the following figure.

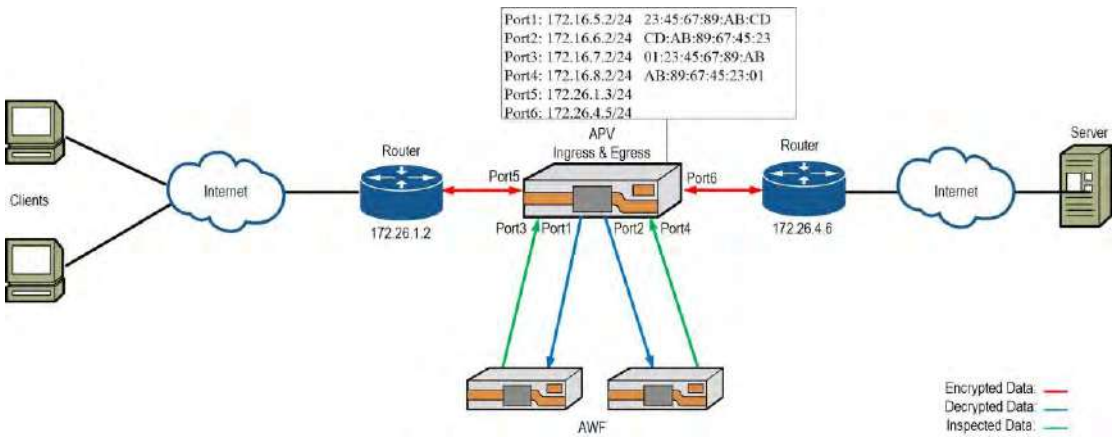

Figure 2–9 Integrated Mode: One L3 APV + Two L2 AWFs

2.2.3.1 Address and Route Settings

1. Set the IP addresses of Port1, Port2, Port3, Port4, Port5 and Port6.

```
AN(config)#ip address port1 172.16.5.2 24
AN(config)#ip address port2 172.16.6.2 24
AN(config)#ip address port3 172.16.7.2 24
AN(config)#ip address port4 172.16.8.2 24
AN(config)#ip address port5 172.26.1.3 24
AN(config)#ip address port6 172.26.4.5 24
```

2. Set the default route.

```
AN(config)#ip route default 172.26.1.2
```

3. Define an Eroute.

```
AN(config)#ip eroute er1 1900 0.0.0.0 0.0.0.0 0 190.10.0.0 255.255.0.0 0 any 172.26.4.6
```

2.2.3.2 Load Balance Settings

- **Load Balancing of SSL Traffic Received from Clients**

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Create FWDMAC real services.

```
AN(config)#slb real fwdmac rs1 port1 01:23:45:67:89:AB 8443
AN(config)#slb real fwdmac rs2 port2 AB:89:67:45:23:01 8443
```

3. Create a real service group using the chi method and add “rs1” and “rs2” to this group.

```
AN(config)#slb group method chi_group chi
AN(config)#slb group member chi_group rs1
AN(config)#slb group member chi_group rs2
```

4. Create a TCPS virtual service.

```
AN(config)#slb virtual tcps vs1 0.0.0.0 443 noarp 0
```

5. Configure a default policy to associate “vs1” with “chi_group”.

```
AN(config)#slb policy default vs1 chi_group
```

6. Configure health checks for “rs1” and “rs2” to ensure that Port3 and Port4 are accessible.

```
AN(config)#slb real health a1 rs1 172.16.7.2 56789 tcp
AN(config)#slb real health a2 rs2 172.16.8.2 56789 tcp
AN(config)#health ipreflect aa 0.0.0.0 56789 tcp
```

➢ **Load Balancing of Non-SSL Traffic Received from Clients**

1. Create L2mac real services.

```
AN(config)#slb real l2mac rs4 01:23:45:67:89:AB port1
AN(config)#slb real l2mac rs5 AB:89:67:45:23:01 port2
```

2. Create an L2 real service group using the chi method, set the route mode to “direct” and add “rs4” and “rs5” to this group.

```
AN(config)#slb group method chi_group1 chi direct
AN(config)#slb group member chi_group1 rs4
AN(config)#slb group member chi_group1 rs5
```

3. Create an L2IP virtual service.

```
AN(config)#slb virtual l2ip l2ip_vs 172.26.1.3 172.26.1.2
```

4. Configure a default policy to associate “l2ip_vs1” with “chi_group1”.

```
AN(config)#slb policy default l2ip_vs1 chi_group1
```

5. Configure two port ranges for “l2ip_vs1”.

```
AN(config)#slb virtual portrange l2ip_vs1 0 442 all dst
AN(config)#slb virtual portrange l2ip_vs1 444 65535 all dst
```

6. Configure two port ranges for “chi_group1”.

```
AN(config)#slb group option portrange chi_group1 0 8442 all src
AN(config)#slb group option portrange chi_group1 8444 65535 all src
```

➢ **Forwarding of Inspected Traffic to the Real Server**

1. Create a TCPS real service, and configure it to keep destination IP addresses unchanged when forwarding packets.

```
AN(config)#slb real tcps rs3 172.26.4.6 443 icmp
AN(config)#slb real settings keepdip rs3
```

2. Create a TCP virtual service and enable RTS for it.

```
AN(config)#slb virtual tcp vs2 0.0.0.0 8443 noarp 0
AN(config)#slb virtual settings rts vs2
```

3. Configure a static policy to associate “vs2” with “rs3”.

```
AN(config)#slb policy static vs2 rs3
```

➢ **Load Balancing of Non-SSL Traffic Returned from the Real Server**

1. Create L2mac real services.

```
AN(config)#slb real l2mac rs6 23:45:67:89:AB:CD port3
AN(config)#slb real l2mac rs7 CD:AB:89:67:45:23 port4
```

2. Create an L2 real service group using the chi method, set the route mode to “route” and add “rs6” and “rs7” to this group.

```
AN(config)#slb group method chi_group2 chi route
AN(config)#slb group member chi_group2 rs6
AN(config)#slb group member chi_group2 rs7
```

3. Create an L2IP virtual service.

```
AN(config)#slb virtual l2ip l2ip_vs2 172.16.4.5
```

4. Configure a default policy to associate “l2ip_vs2” with “chi_group2”.

```
AN(config)#slb policy default l2ip_vs2 chi_group2
```

5. Configure two port ranges for “l2ip_vs2”.

```
AN(config)#slb virtual portrange l2ip_vs2 0 442 all src
AN(config)#slb virtual portrange l2ip_vs2 444 65535 all src
```

6. Configure two port ranges for “chi_group2”.

```
AN(config)#slb group option portrange chi_group2 0 8442 all dst
AN(config)#slb group option portrange chi_group2 8444 65535 all dst
```

2.2.3.3 SSL Interception Settings

1. Create an SSL virtual host and associate it with “vs1”.

```
AN(config)#ssl host virtual vhost1 vs1
```

2. Enable SSL interception for “vhost1”.

```
AN(config)#ssli on vhost1 1
```

3. Generate SSL interception certificates for “vhost1”, activate them and enable “vhost1”.

Note: The generated CA certificates must also be imported into the trusted CA list of clients' browsers.

```
AN(config)#ssli cacert rsa vhost1 2048 1 1
AN(config)#ssli cacert ecc prime256v1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

Note: Currently, elliptic curve secp521r1 is not widely supported by mainstream browsers. It is recommended to set prime256v1 or secp384r1 to avoid connection failures.

4. Create an SSL real host "rhost1" and associate it with "rs3".

```
AN(config)#ssl host real rhost1 rs3
```

5. Enable SSL interception for "rhost1" and enable "rhost1".

```
AN(config)#ssli on rhost1 1
AN(config)#ssl start rhost1
```

2.2.4 Distributed Mode: Two L3 APVs + Two L2 AWFs

In this deployment mode, the APV appliances are set up in L3 mode, the AWF appliances are set up in L2 mode and the two APV appliances play the role of the ingress and egress nodes respectively. The interface and route configurations on the AWF appliances are as shown in the following figure.

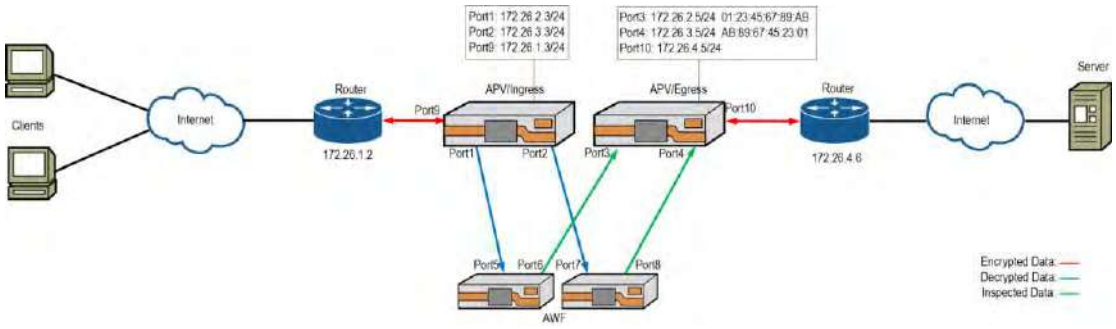

Figure 2–10 Distributed Mode: Two L3 APVs + Two L2 AWFs

2.2.4.1 Configuring the Ingress Node

2.2.4.1.1 Address and Route Settings

1. Set the IP addresses of Port1, Port2, and Port9.

```
AN(config)#ip address port1 172.26.2.3 24
AN(config)#ip address port2 172.26.3.3 24
AN(config)#ip address port9 172.26.1.3 24
```

2. Set the default route.

```
AN(config)#ip route default 172.26.1.2
```

3. Enable the IPflow function.

```
AN(config)#ip ipflow on
```

4. Define Eroutes.

```
AN(config)#ip eroute er1 1900 0.0.0.0 0.0.0.0 0 190.10.0.0 255.255.0.0 0 172.26.2.5
AN(config)#ip eroute er2 1900 0.0.0.0 0.0.0.0 0 190.10.0.0 255.255.0.0 0 172.26.3.5
```

2.2.4.1.2 Load Balance Settings

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Create a TCPS virtual service.

```
AN(config)#slb virtual tcps vs1 0.0.0.0 443 noarp 0
```

3. Create FWDMAC real services.

```
AN(config)#slb real fwdmac port1 01:23:45:67:89:AB 8443
AN(config)#slb real fwdmac port2 AB:89:67:45:23:01 8443
```

4. Create an L2 real service group using the chi method and add “rs1” and “rs2” to this group.

```
AN(config)#slb group method chi_group chi
AN(config)#slb group member chi_group rs1
AN(config)#slb group member chi_group rs2
```

5. Configure a default policy to associate “vs1” with “chi_group”.

```
AN(config)#slb policy default vs1 chi_group
```

6. Configure health checks for “rs1” and “rs2” to ensure that Port3 and Port4 on the egress node are accessible (a health check reflector is needed on the egress node).

```
AN(config)#slb real health a1 rs1 172.26.2.5 56789 tcp 3 3
AN(config)#slb real health a2 rs2 172.26.3.5 56789 tcp 3 3
```

2.2.4.1.3 SSL Interception Settings

1. Create an SSL virtual host and associate it with “vs1”.

```
AN(config)#ssl host virtual vhost1 vs1
```

2. Enable SSL interception for “vhost1”.

```
AN(config)#ssli on vhost1 0
```

3. Generate SSL interception certificates for “vhost1”, activate them and enable “vhost1”.

Note: The generated CA certificates must also be imported into the trusted CA list of clients’ browsers.

```
AN(config)#ssli cacert rsa vhost1 2048 1 1
AN(config)#ssli cacert ecc vhost1 prime256v1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

Note: Currently, elliptic curve secp521r1 is not widely supported by mainstream browsers. It is recommended to set prime256v1 or secp384r1 to avoid connection failures.

2.2.4.2 Configuring the Egress Node

2.2.4.2.1 Address and Route Settings

1. Set the IP addresses of Port3, Port4 and Port10.

```
AN(config)#ip address port3 172.26.2.5 24
AN(config)#ip address port4 172.26.3.5 24
AN(config)#ip address port10 172.26.4.5 24
```

2. Define an Eroute to the server.

```
AN(config)#ip eroute er2 1900 0.0.0.0 0.0.0.0 0 190.10.0.0 255.255.0.0 0 any 172.26.4.6
```

3. Enable RTS.

```
AN(config)#ip rts on
```

4. Define Eroutes to the clients.

```
AN(config)#ip eroute er3 1900 190.10.0.0 255.255.0.0 0 0.0.0.0 0.0.0.0 0 any 172.26.2.3
AN(config)#ip eroute er4 1900 190.10.0.0 255.255.0.0 0 0.0.0.0 0.0.0.0 0 any 172.26.3.3
```

2.2.4.2.2 Load Balance Settings

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Configure a TCPS real service, and configure it to keep destination IP addresses unchanged when forwarding packets.

```
AN(config)#slb real tcps rs3 172.26.4.6 443 icmp
AN(config)#slb real settings keepdip rs3
```

3. Configure a TCP virtual service and enable RTS for it.

```
AN(config)#slb virtual tcp vs2 0.0.0.0 8443 noarp 0
AN(config)#slb virtual settings rts vs2
```

4. Configure a static policy to associate “vs2” with “rs3”.

```
AN(config)#slb policy static vs2 rs3
```

5. Create a health check reflector “reflector1”.

```
AN(config)#health ipreflect reflector1 0.0.0.0 56789 tcp
```

2.2.4.2.3 SSL Interception Settings

1. Create an SSL real host “rhost1” and associate it with “rs3”.

```
AN(config)#ssl host real rhost1 rs3
```

2. Enable SSL interception for “rhost1” and enable “rhost1”.

```
AN(config)#ssli on rhost1 0
AN(config)#ssl start rhost1
```

2.2.5 Integrated Mode: One L2 APV + One L2 AWF

In this deployment mode, the APV appliance is set up in L2 mode, the AWF appliance is set up in L2 mode. The APV appliance serves as both the ingress and egress nodes. Two bridge instances are configured on the APV appliance:

- One bridge is used to transfer SSL traffic to the SSL interception module for decryption and then forward the decrypted traffic to the AWF appliance.
- The other bridge is used to receive the inspected traffic from the AWF appliance and re-encrypt the traffic before sending it on to the servers.

The network topology and interface configurations are as shown in the following figure.

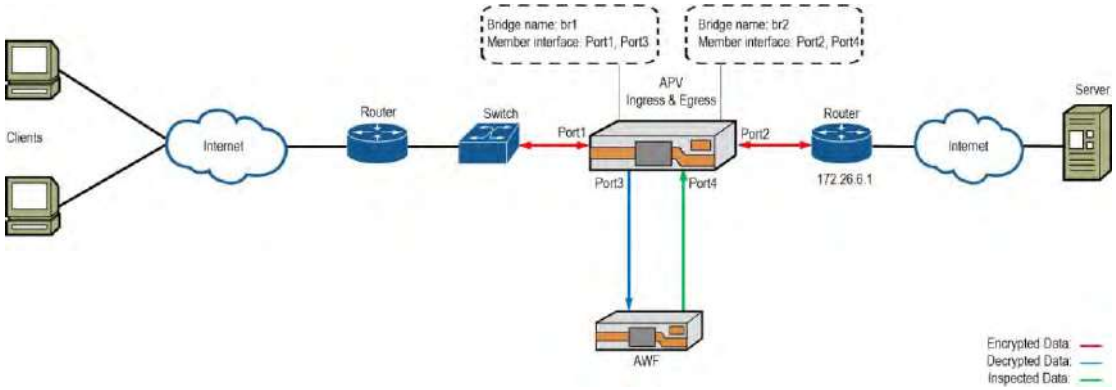

Figure 2–11 Integrated Mode: One L2 APV + One L2 AWF

2.2.5.1 Bridge Settings

1. Create two bridge instances.

```
AN(config)#bridge name br1
AN(config)#bridge name br2
```

2. Add members to the created bridge instances.

```
AN(config)#bridge member br1 port1 yes
AN(config)#bridge member br1 port3 yes
AN(config)#bridge member br2 port2 yes
AN(config)#bridge member br2 port4 yes
```

3. Create filter rules to bypass returned SSL traffic and to acquire server certificates.

```
AN(config)#bridge apprule br1 0.0.0.0 443 0.0.0.0 0 tcp
```

4. Create filter rules to forward all SSL traffic (including encrypted and cleartext traffic) to the SSL interception module.

```
AN(config)#bridge apprule br1 0.0.0.0 0 0.0.0.0 443 tcp
AN(config)#bridge apprule br1 0.0.0.0 8443 0.0.0.0 0 tcp
AN(config)#bridge apprule br2 0.0.0.0 443 0.0.0.0 0 tcp
AN(config)#bridge apprule br2 0.0.0.0 0 0.0.0.0 8443 tcp
```

2.2.5.2 SSL Settings

2.2.5.2.1 Forwarding of Received SSL Traffic to the Security Device

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Create a FWDMAC real service.

   Note that “AB:89:67:45:23:01” does not represent any port. It can be replaced with an arbitrary MAC address, but it must be set.

```
AN(config)#slb real fwdmac rs1 port3 AB:89:67:45:23:01 8443
```

3. Create a real service group using the chi method and add “rs1” to this group.

```
AN(config)#slb group method chi_group chi
AN(config)#slb group member chi_group rs1
```

4. Create a TCPS virtual service.

```
AN(config)#slb virtual tcps vs1 0.0.0.0 443 noarp 0
```

5. Configure a default policy to associate “vs1” with “chi_group”.

```
AN(config)#slb policy default vs1 chi_group
```

6. Create an SSL virtual host and associate it with “vs1”.

```
AN(config)#ssl host virtual vhost1 vs1
```

2.2.5.2.2 Forwarding of Inspected SSL Traffic to the Real Service

1. Create a TCPS real service, and configure it to keep destination IP addresses unchanged when forwarding packets.

   Note that “172.26.6.1” can be replaced with an arbitrary IP address.

```
AN(config)#slb real tcps rs2 172.26.6.1 443 none
AN(config)#slb real settings keepdip rs2
```

2. Create a TCP virtual service.

```
AN(config)#slb virtual tcp vs2 0.0.0.0 8443 noarp 0
```

3. Configure a static policy to associate “vs2” with “rs2”.

```
AN(config)#slb policy static vs2 rs2
```

4. Disable the real service health check.

```
AN(config)#health off
```

5. Create an SSL real host “rhost1” and associate it with “rs2”.

```
AN(config)#ssl host real rhost1 rs2
```

2.2.5.3 SSL Interception Settings

1. Enable SSL interception for “vhost1”.

```
AN(config)#ssli on vhost1 1
```

2. Generate SSL interception certificates for “vhost1”, activate them and enable “vhost1”.

Note: The generated CA certificates must also be imported into the trusted CA list of clients’ browsers.

```
AN(config)#ssli cacert rsa vhost1 2048 1 1
AN(config)#ssli cacert ecc vhost1 prime256v1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

Note: Currently, elliptic curve secp521r1 is not widely supported by mainstream browsers. It is recommended to set prime256v1 or secp384r1 to avoid connection failures.

3. Enable SSL interception for “rhost1” and enable “rhost1”.

```
AN(config)#ssli on rhost1 1
AN(config)#ssl start rhost1
```

Note: If the SSL server uses a testing certificate or a self-signed certificate, both the ingress and egress nodes must have its certificate chain (including the root CA certificate) imported. For testing-only purposes, administrators can choose not to import the certificate chain, but must execute the "**ssl globals verifycert off**" command to disable the server authentication function.

2.2.6 Distributed Mode: Two L2 APVs + One L2 AWF

In this deployment mode, both the APV appliances and the AWF appliance are set up in L2 mode, and the two APV appliances play the role of the ingress and egress nodes respectively. The ingress node and the egress node each have a bridge instance configured:

- The bridge on the ingress node is used to transfer SSL traffic to the SSL interception module for decryption and then forward the decrypted traffic to the AWF appliance.
- The bridge on the egress node is used to receive the inspected traffic from the AWF appliance and re-encrypt the traffic.

The network topology and interface are as shown in the following figure.

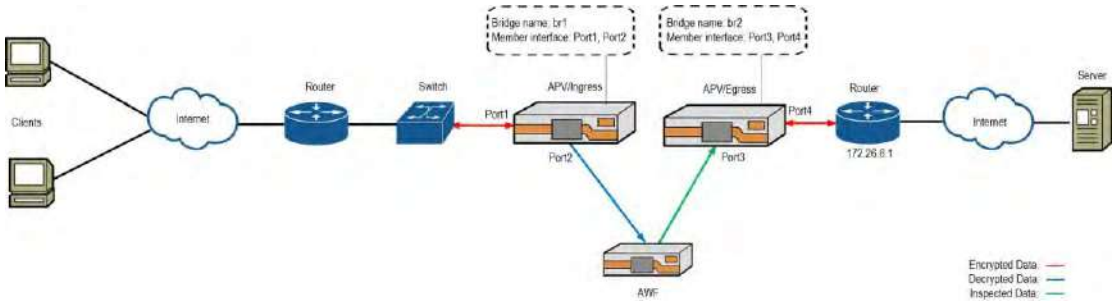

Figure 2–12 Distributed Mode: Two L2 APVs + One L2 AWF

2.2.6.1 Configuring the Ingress Node

2.2.6.1.1 Bridge Settings

1. Create a bridge instance.

```
AN(config)#bridge name br1
```

2. Add members to the created bridge instance.

```
AN(config)#bridge member br1 port1 yes
AN(config)#bridge member br1 port2 yes
```

3. Create filter rules to bypass returned SSL traffic and to acquire server certificates.

```
AN(config)#bridge apprule br1 0.0.0.0 443 0.0.0.0 0 tcp
```

4. Create filter rules to forward clients' encrypted SSL traffic and servers' cleartext SSL traffic to the SSL interception module.

```
AN(config)#bridge apprule br1 0.0.0.0 0 0.0.0.0 443 tcp
```

```
AN(config)#bridge apprule br1 0.0.0.0 8443 0.0.0.0 0 tcp
```

2.2.6.1.2 SSL Settings

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Create a TCPS virtual service.

```
AN(config)#slb virtual tcps vs1 0.0.0.0 443 noarp 0
```

3. Create a FWDMAC real service.

   Note that “AB:89:67:45:23:01” does not represent any port. It can be replaced with an arbitrary MAC address, but it must be set.

```
AN(config)#slb real fwdmac rs1 port2 AB:89:67:45:23:01 8443
```

4. Create an L2 real service group using the chi method and add “rs1” to this group.

```
AN(config)#slb group method chi_group chi
AN(config)#slb group member chi_group rs1
```

5. Configure a default policy to associate “vs1” with “chi_group”.

```
AN(config)#slb policy default vs1 chi_group
```

6. Create an SSL virtual host and associate it with “vs1”.

```
AN(config)#ssl host virtual vhost1 vs1
```

2.2.6.1.3 SSL Interception Settings

1. Enable SSL interception for “vhost1”.

```
AN(config)#ssli on vhost1 0
```

2. Generate SSL interception certificates for “vhost1”, activate them and enable “vhost1”.

```
AN(config)#ssli cacert rsa vhost1 2048 1 1
AN(config)#ssli cacert ecc vhost1 prime256v1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

Note:

1. The generated CA certificates must also be imported into the trusted CA list of clients’ browsers.
2. If the SSL server uses a testing certificate or a self-signed certificate, both the ingress and egress nodes must have its certificate chain (including the root CA certificate) imported. For testing-only purposes, administrators can choose not to

import the certificate chain, but must execute the "**ssl globals verifycert off**" command to disable the server authentication function.

3. Currently, elliptic curve secp521r1 is not widely supported by mainstream browsers. It is recommended to set prime256v1 or secp384r1 to avoid connection failures.

2.2.6.2 Configuring the Egress Node

2.2.6.2.1 Bridge Settings

1. Create a bridge instance.

```
AN(config)#bridge name br2
```

2. Add members to the created bridge instance.

```
AN(config)#bridge member br2 port3 yes
AN(config)#bridge member br2 port4 yes
```

3. Create filter rules to forward servers' encrypted SSL traffic and clients' cleartext SSL traffic to the SSL interception module.

```
AN(config)#bridge apprule br2 0.0.0.0 443 0.0.0.0 0 tcp
AN(config)#bridge apprule br2 0.0.0.0 0 0.0.0.0 8443 tcp
```

2.2.6.2.2 SSL Settings

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Configure a TCPS real service, and configure it to keep destination IP addresses unchanged when forwarding packets.

   Note that "172.26.6.1" can be replaced with an arbitrary IP address.

```
AN(config)#slb real tcps rs2 172.26.6.1 443 none
AN(config)#slb real settings keepdip rs2
```

3. Configure a TCP virtual service.

```
AN(config)#slb virtual tcp vs2 0.0.0.0 8443 noarp 0
```

4. Configure a static policy to associate "vs2" with "rs2".

```
AN(config)#slb policy static vs2 rs2
```

5. Disable the real service health check.

```
AN(config)#health off
```

6. Create an SSL real host "rhost1" and associate it with "rs2".

```
AN(config)#ssl host real rhost1 rs2
```

2.2.6.2.3 SSL Interception Settings

7. Enable SSL interception for “rhost1” and enable “rhost1”.

```
AN(config)#ssli on rhost1 0
AN(config)#ssl start rhost
```

Note: If the SSL server uses a testing certificate or a self-signed certificate, both the ingress and egress nodes must have its certificate chain (including the root CA certificate) imported. For testing-only purposes, administrators can choose not to import the certificate chain, but must execute the “**ssl globals verifycert off**” command to disable the server authentication function.

3 Outbound SSL Interception Deployment

This chapter provides SSL configuration examples in outbound network topologies.

- 3.1 Deployment on APV Working in L3 Mode
  - 3.1.1 Integrated Mode: One L3 APV + Two L3 Firewalls
  - 3.1.2 Distributed Mode: Two L3 APVs + Two L3 Firewalls
  - 3.1.3 Integrated Mode: One L3 APV + Two L2 Firewalls
  - 3.1.4 Distributed Mode: Two L3 APVs + Two L2 Firewalls
- 3.2 Deployment on APV Working in L2 Mode
  - 3.2.1 Integrated Mode: One L2 APV + One L2 Firewall (With VLAN)
  - 3.2.2 Integrated Mode: One L2 APV + One L2 Firewall (Without VLAN)
  - 3.2.3 Distributed Mode: Two L2 APVs + One L2 Firewall (With VLAN)
  - 3.2.4 Distributed Mode: Two L2 APVs + One L2 Firewall (Without VLAN)

Note: In the integrated mode, a global root CA certificate needs to be imported. In the distributed mode, a global root CA certificate is needed on both the ingress and egress nodes. The APV will use this global CA certificate to verify the validity of server certificates.

3.1 Deployment on APV Working in L3 Mode

3.1.1 Integrated Mode: One L3 APV + Two L3 Firewalls

In this deployment mode, both the APV appliance and the firewalls are set up in L3 mode. The APV appliance serves as both the ingress and egress nodes. The interface and route configurations on the firewalls are as shown in the following figure.

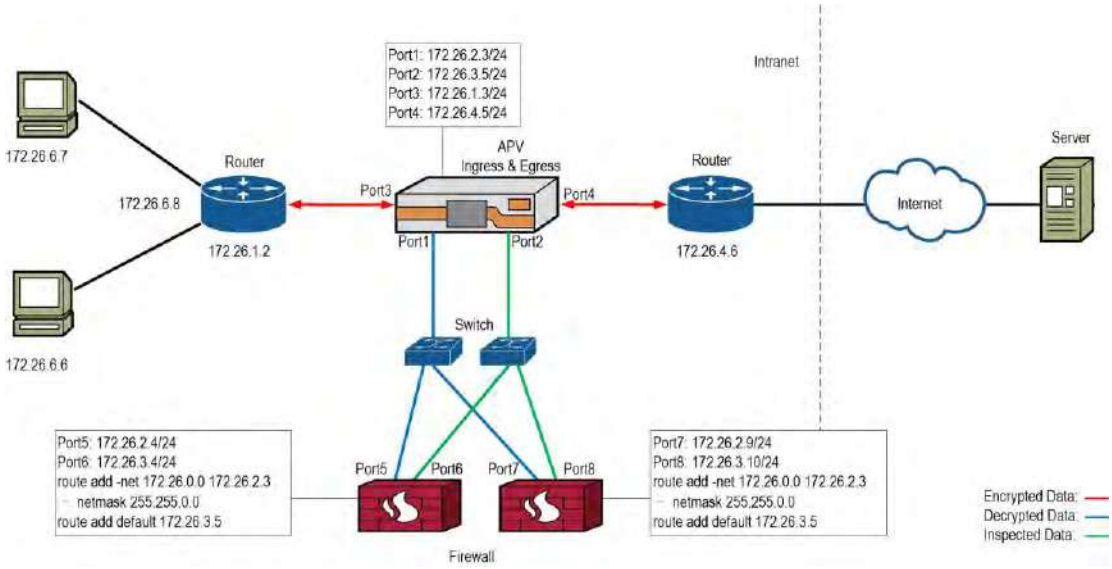

Figure 3–1 Integrated Mode: One L3 APV + Two L3 Firewalls

3.1.1.1 Address and Route Settings

1. Set the IP addresses of Port1, Port2, Port3 and Port4.

```
AN(config)#ip address port1 172.26.2.3 24
AN(config)#ip address port2 172.26.3.5 24
AN(config)#ip address port3 172.26.1.3 24
AN(config)#ip address port4 172.26.4.5 24
```

2. Set the default route.

```
AN(config)#ip route default 172.26.4.6
```

3. Define an Eroute.

```
AN(config)#ip eroute er1 1900 0.0.0.0 0.0.0.0 0 172.26.0.0 255.255.0.0 0 any 172.26.1.2
```

3.1.1.2 Load Balance Settings

- **Load Balancing of SSL Traffic Received from Clients**

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Create FWDIP real services.

```
AN(config)#slb real fwdip rs1 172.26.2.4 8443
AN(config)#slb real fwdip rs2 172.26.2.9 8443
```

3. Create a real service group using the hi method and add “rs1” and “rs2” to this group.

```
AN(config)#slb group method hi_group hi
AN(config)#slb group member hi_group rs1
AN(config)#slb group member hi_group rs2
```

4. Create a TCPS virtual service.

```
AN(config)#slb virtual tcps vs1 0.0.0.0 443 noarp 0
```

5. Configure a default policy to associate “vs1” with “hi_group”.

```
AN(config)#slb policy default vs1 hi_group
```

6. Configure health check for “rs1” and “rs2” to ensure that Port2 is accessible.

```
AN(config)#slb real health a1 rs1 172.26.3.5 56789 tcp 3 3
AN(config)#slb real health a2 rs2 172.26.3.5 56789 tcp 3 3
AN(config)#health ipreflect aa 172.26.3.5 56789 tcp
```

7. Configure health check for “rs1” and “rs2” to check the health status of security devices.

```
AN(config)#slb real health hc_os_h1 rs1 172.26.2.4 0 icmp 3 3
AN(config)#slb real health hc_os_h2 rs2 172.26.2.9 0 icmp 3 3
```

8. Set the relationship among health checks of “rs1” and “rs2” to “and”.

```
AN(config)#health relation rs1 and
AN(config)#health relation rs2 and
```

➢ **Load Balancing of Non-SSL Traffic Received from Clients**

1. Create L2IP real services.

```
AN(config)#slb real l2ip rs4 172.26.2.4
AN(config)#slb real l2ip rs5 172.26.2.9
```

2. Create an L2 real service group using the chi method, set the route mode to “direct” and add “rs4” and “rs5” to this group.

```
AN(config)#slb group method chi_group1 chi direct
AN(config)#slb group member chi_group1 rs4
AN(config)#slb group member chi_group1 rs5
```

3. Create an L2IP virtual service.

```
AN(config)#slb virtual l2ip l2ip_vs1 172.26.1.3 172.26.1.2
```

4. Configure a default policy to associate “l2ip_vs1” with “chi_group1”.

```
AN(config)#slb policy default l2ip_vs1 chi_group1
```

5. Configure two port ranges for “l2ip_vs1”.

```
AN(config)#slb virtual portrange l2ip_vs1 0 442 all dst
AN(config)#slb virtual portrange l2ip_vs1 444 65535 all dst
```

6. Configure two port ranges for “chi_group1”.

```
AN(config)#slb group option portrange chi_group1 0 8442 all src
AN(config)#slb group option portrange chi_group1 8444 65535 all src
```

➢ **Forwarding of Inspected Traffic to the Real Server**

1. Create a TCP virtual service and enable RTS for it.

```
AN(config)#slb virtual tcp vs2 0.0.0.0 8443 noarp 0
AN(config)#slb virtual settings rts vs2
```

2. Create a TCPS real service and configure it to keep destination IP addresses unchanged when forwarding packets.

```
AN(config)#slb real tcps rs3 172.26.4.6 443 icmp
AN(config)#slb real settings keepdip rs3
```

3. Configure a static policy to associate “vs2” with “rs3”.

```
AN(config)#slb policy static vs2 rs3
```

- **Load Balancing of Non-SSL Traffic Returned from the Real Server**

1. Create L2IP real services.

```
AN(config)#slb real l2ip rs6 172.26.3.4
AN(config)#slb real l2ip rs7 172.26.3.10
```

2. Create an L2 real service group using the chi method, and add “rs6” and “rs7” to this group.

```
AN(config)#slb group method chi_group2 chi route
AN(config)#slb group member chi_group2 rs6
AN(config)#slb group member chi_group2 rs7
```

3. Create an L2IP virtual service.

```
AN(config)#slb virtual l2ip l2ip_vs2 172.26.4.5 172.26.4.6
```

4. Configure a default policy to associate “l2ip_vs2” with “chi_group2”.

```
AN(config)#slb policy default l2ip_vs2 chi_group2
```

5. Configure two port ranges for “l2ip_vs2”.

```
AN(config)#slb virtual portrange l2ip_vs2 0 442 all src
AN(config)#slb virtual portrange l2ip_vs2 444 65535 all src
```

6. Configure two port ranges for “chi_group2”.

```
AN(config)#slb group option portrange chi_group2 0 8442 all dst
AN(config)#slb group option portrange chi_group2 8444 65535 all dst
```

3.1.1.3 SSL Interception Settings

1. Create an SSL virtual host and associate it with “vs1”.

```
AN(config)#ssl host virtual vhost1 vs1
```

2. Enable SSL interception for “vhost1”.

```
AN(config)#ssli on vhost1 1
```

3. Generate SSL interception certificates for “vhost1”, activate them, and enable “vhost1”.

Note: The generated CA certificates must also be imported into the trusted CA list of clients’ browsers.

```
AN(config)#ssli cacert rsa vhost1 2048 1 1
AN(config)#ssli cacert ecc vhost1 prime256v1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

Note: Currently, elliptic curve secp521r1 is not widely supported by mainstream browsers. It is recommended to set prime256v1 or secp384r1 to avoid connection failures.

4. Create an SSL real host “rhost1” and associate it with “rs3”.

```
AN(config)#ssl host real rhost1 rs3
```

5. Enable SSL interception for “rhost1” and enable “rhost1”.

```
AN(config)#ssli on rhost1 1
AN(config)#ssl start rhost1
```

3.1.2 Distributed Mode: Two L3 APVs + Two L3 Firewalls

In this deployment mode, the APV appliances and firewalls are set up in L3 mode and the two APV appliances play the role of the ingress and egress nodes respectively. The interface and route configurations on the firewalls are as shown in the following figure.

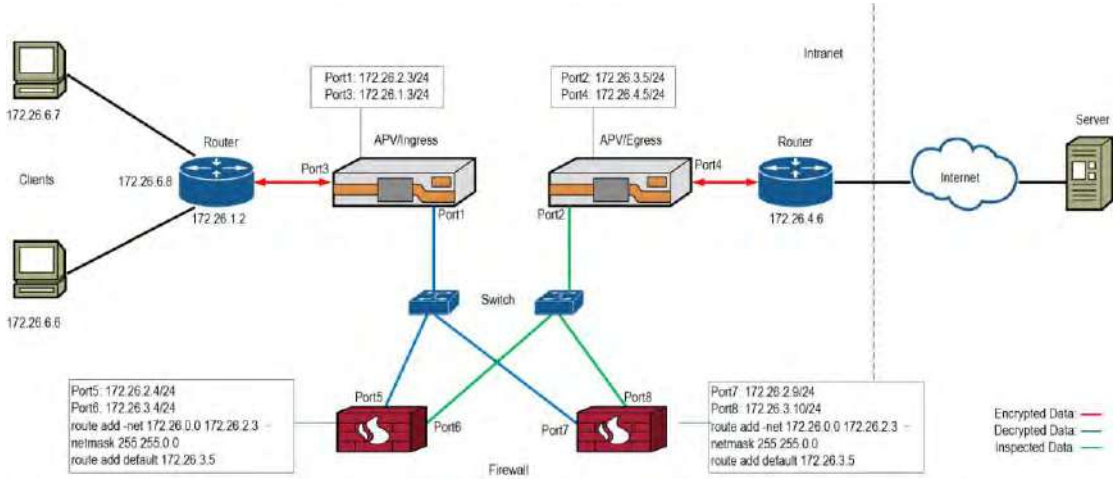

Figure 3–2 Distributed Mode: Two L3 APVs + Two L3 Firewalls

3.1.2.1 Configuring the Ingress Node

3.1.2.1.1 Address and Route Settings

1. Set the IP addresses of Port1 and Port3.

```
AN(config)#ip address port1 172.26.2.3 24
AN(config)#ip address port3 172.26.1.3 24
```

2. Set the default route.

```
AN(config)#ip route default 172.26.1.2
```

3. Define Eroutes.

```
AN(config)#ip eroute er1 1900 172.26.0.0 255.255.0.0 0 0.0.0.0 0.0.0.0 172.26.2.4
AN(config)#ip eroute er2 1900 172.26.0.0 255.255.0.0 0 0.0.0.0 0.0.0.0 172.26.2.9
```

4. Enable the IPflow function.

```
AN(config)#ip ipflow on
```

3.1.2.1.2 Load Balance Settings

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Create FWDIP real services.

```
AN(config)#slb real fwdip rs1 172.26.2.4 8443
AN(config)#slb real fwdip rs2 172.26.2.9 8443
```

3. Create a real service group using the chi method and add the FWDIP real services to this group.

```
AN(config)#slb group method chi_group chi
AN(config)#slb group member chi_group rs1
AN(config)#slb group member chi_group rs2
```

4. Create a TCPS virtual service.

```
AN(config)#slb virtual tcps vs1 0.0.0.0 443 noarp 0
```

5. Configure a default policy to associate “vs1” with “chi_group”.

```
AN(config)#slb policy default vs1 chi_group
```

6. Configure health checks for “rs1” and “rs2” to ensure that Port2 on the egress node is accessible (a health check reflector is needed on the egress node).

```
AN(config)#slb real health a1 rs1 172.26.3.5 56789 tcp 3 3
AN(config)#slb real health a2 rs2 172.26.3.5 56789 tcp 3 3
```

7. Configure health checks for “rs1” and “rs2” to check the health status of security devices.

```
AN(config)#slb real health hc_os_h1 rs1 172.26.2.4 0 icmp 3 3
AN(config)#slb real health hc_os_h2 rs2 172.26.2.9 0 icmp 3 3
```

8. Set the relationship among health checks of “rs1” and “rs2” to “and”.

```
AN(config)#health relation rs1 and
AN(config)#health relation rs2 and
```

3.1.2.1.3 SSL Interception Settings

1. Create an SSL virtual host and associate it with “vs1”.

```
AN(config)#ssl host virtual vhost1 vs1
```

2. Enable SSL interception for “vhost1”.

```
AN(config)#ssli on vhost1 0
```

3. Generate SSL interception certificates for “vhost1”, activate them, and enable “vhost1”.

Note: The generated CA certificates must also be imported into the trusted CA list of clients’ browsers.

```
AN(config)#ssli cacert rsa vhost1 2048 1 1
AN(config)#ssli cacert ecc vhost1 prime256v1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

Note: Currently, elliptic curve secp521r1 is not widely supported by mainstream browsers. It is recommended to set prime256v1 or secp384r1 to avoid connection failures.

3.1.2.2 Configuring the Egress Node

3.1.2.2.1 Address and Route Settings

1. Set the IP addresses of Port2 and Port4.

```
AN(config)#ip address port2 172.26.3.5 24
AN(config)#ip address port4 172.26.4.5 24
```

2. Set the default route.

```
AN(config)#ip route default 172.26.4.6
```

3. Enable RTS.

```
AN(config)#ip rts on
```

4. Define Eroutes.

```
AN(config)#ip eroute er1 1900 0.0.0.0 0.0.0.0 0 172.26.0.0 255.255.255.0 0 172.26.3.4
AN(config)#ip eroute er2 1900 0.0.0.0 0.0.0.0 0 172.26.0.0 255.255.255.0 0 172.26.3.10
```

3.1.2.2.2 Load Balance Settings

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Configure a TCPS real service, and configure it to keep destination IP addresses unchanged when forwarding packets.

```
AN(config)#slb real tcps rs3 172.26.4.6 443 icmp
AN(config)#slb real settings keepdip rs3
```

3. Configure a TCP virtual service and enable RTS for it.

```
AN(config)#slb virtual tcp vs2 0.0.0.0 8443 noarp 0
AN(config)#slb virtual settings rts vs2
```

4. Configure a static policy to associate “vs2” with “rs3”.

```
AN(config)#slb policy static vs2 rs3
```

5. Create a health check reflector “reflector1”.

```
AN(config)#health ipreflect reflector1 172.26.3.5 56789 tcp
```

3.1.2.2.3 SSL Interception Settings

1. Create an SSL real host “rhost1” and associate it with “rs3”.

```
AN(config)#ssl host real rhost1 rs3
```

2. Enable SSL interception for “rhost1” and enable “rhost1”.

```
AN(config)#ssli on rhost1 0
AN(config)#ssl start rhost1
```

3.1.3 Integrated Mode: One L3 APV + Two L2 Firewalls

In this deployment mode, the APV appliance is set up in L3 mode, firewalls are set up in L2 mode, and the APV appliance serves as both the ingress and egress nodes. The interface and route configurations on the firewalls are as shown in the following figure.

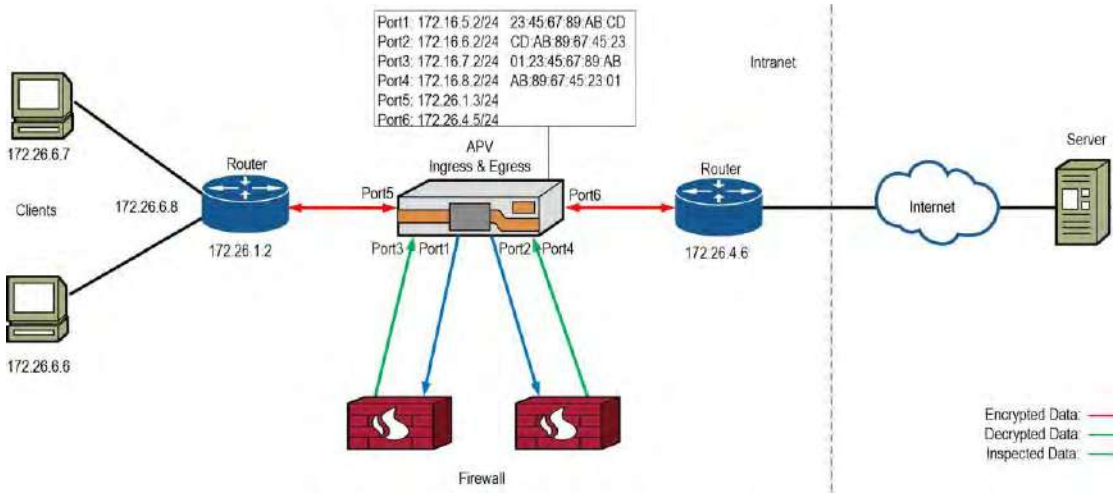

Figure 3–3 Integrated Mode: One L3 APV + Two L2 Firewalls

3.1.3.1 Address and Route Settings

1. Set the IP addresses of Port1, Port2, Port3, Port4, Port5 and Port6.

```
AN(config)#ip address port1 172.16.5.2 24
AN(config)#ip address port2 172.16.6.2 24
AN(config)#ip address port3 172.16.7.2 24
```

```
AN(config)#ip address port4 172.16.8.2 24
AN(config)#ip address port5 172.26.1.3 24
AN(config)#ip address port6 172.26.4.5 24
```

2. Set the default route.

```
AN(config)#ip route default 172.26.4.6
```

3. Define an Eroute.

```
AN(config)#ip eroute er1 1900 0.0.0.0 0.0.0.0 0 172.26.0.0 255.255.0.0 0 any 172.26.1.2
```

3.1.3.2 Load Balance Settings

- **Load Balancing of SSL Traffic Received from Clients**

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Create FWDMAC real services.

```
AN(config)#slb real fwdmac rs1 port1 01:23:45:67:89:AB 8443
AN(config)#slb real fwdmac rs2 port2 AB:89:67:45:23:01 8443
```

3. Create a real service group using the chi method and add “rs1” and “rs2” to this group.

```
AN(config)#slb group method chi_group chi
AN(config)#slb group member chi_group rs1
AN(config)#slb group member chi_group rs2
```

4. Create a TCPS virtual service.

```
AN(config)#slb virtual tcps vs1 0.0.0.0 443 noarp 0
```

5. Configure a default policy to associate “vs1” with “chi_group”.

```
AN(config)#slb policy default vs1 chi_group
```

6. Configure health checks for “rs1” and “rs2” to ensure that Port3 and Port4 are accessible.

```
AN(config)#slb real health a1 rs1 172.16.7.2 56789 tcp
AN(config)#slb real health a2 rs2 172.16.8.2 56789 tcp
AN(config)#health ipreflect aa 0.0.0.0 56789 tcp
```

- **Load Balancing of Non-SSL Traffic Received from Clients**

1. Create L2mac real services.

```
AN(config)#slb real l2mac rs4 01:23:45:67:89:AB port1
AN(config)#slb real l2mac rs5 AB:89:67:45:23:01 port2
```

2. Create an L2 real service group using the chi method, set the route mode to “direct” and add “rs4” and “rs5” to this group.

```
AN(config)#slb group method chi_group1 chi direct
AN(config)#slb group member chi_group1 rs4
AN(config)#slb group member chi_group1 rs5
```

3. Create an L2IP virtual service.

```
AN(config)#slb virtual l2ip l2ip_vs 172.26.1.3 172.26.1.2
```

4. Configure a default policy to associate “l2ip_vs1” with “chi_group1”.

```
AN(config)#slb policy default l2ip_vs1 chi_group1
```

5. Configure two port ranges for “l2ip_vs1”.

```
AN(config)#slb virtual portrange l2ip_vs1 0 442 all dst
AN(config)#slb virtual portrange l2ip_vs1 444 65535 all dst
```

6. Configure two port ranges for “chi_group1”.

```
AN(config)#slb group option portrange chi_group1 0 8442 all src
AN(config)#slb group option portrange chi_group1 8444 65535 all src
```

➢ **Forwarding of Inspected Traffic to the Real Server**

1. Create a TCPS real service, and configure it to keep destination IP addresses unchanged when forwarding packets.

```
AN(config)#slb real tcps rs3 172.26.4.6 443 icmp
AN(config)#slb real settings keepdip rs3
```

2. Create a TCP virtual service and enable RTS for it.

```
AN(config)#slb virtual tcp vs2 0.0.0.0 8443 noarp 0
AN(config)#slb virtual settings rts vs2
```

3. Configure a static policy to associate “vs2” with “rs3”.

```
AN(config)#slb policy static vs2 rs3
```

➢ **Load Balancing of Non-SSL Traffic Returned from the Real Server**

1. Create L2mac real services.

```
AN(config)#slb real l2mac rs6 23:45:67:89:AB:CD port3
AN(config)#slb real l2mac rs7 CD:AB:89:67:45:23 port4
```

2. Create an L2 real service group using the chi method, set the route mode to “route” and add “rs6” and “rs7” to this group.

```
AN(config)#slb group method chi_group2 chi route
AN(config)#slb group member chi_group2 rs6
AN(config)#slb group member chi_group2 rs7
```

3. Create an L2IP virtual service.

```
AN(config)#slb virtual l2ip l2ip_vs2 172.16.4.5
```

4. Configure a default policy to associate “l2ip_vs2” with “chi_group2”.

```
AN(config)#slb policy default l2ip_vs2 chi_group2
```

5. Configure two port ranges for “l2ip_vs2”.

```
AN(config)#slb virtual portrange l2ip_vs2 0 442 all src
AN(config)#slb virtual portrange l2ip_vs2 444 65535 all src
```

6. Configure two port ranges for “chi_group2”.

```
AN(config)#slb group option portrange chi_group2 0 8442 all dst
AN(config)#slb group option portrange chi_group2 8444 65535 all dst
```

3.1.3.3 SSL Interception Settings

1. Create an SSL virtual host and associate it with “vs1”.

```
AN(config)#ssl host virtual vhost1 vs1
```

2. Enable SSL interception for “vhost1”.

```
AN(config)#ssli on vhost1 1
```

3. Generate SSL interception certificates for “vhost1”, activate them and enable “vhost1”.

Note: The generated CA certificates must also be imported into the trusted CA list of clients’ browsers.

```
AN(config)#ssli cacert rsa vhost1 2048 1 1
AN(config)#ssli cacert ecc prime256v1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

Note: Currently, elliptic curve secp521r1 is not widely supported by mainstream browsers. It is recommended to set prime256v1 or secp384r1 to avoid connection failures.

4. Create an SSL real host “rhost1” and associate it with “rs3”.

```
AN(config)#ssl host real rhost1 rs3
```

5. Enable SSL interception for “rhost1” and enable “rhost1”.

```
AN(config)#ssli on rhost1 1
AN(config)#ssl start rhost1
```

3.1.4 Distributed Mode: Two L3 APVs + Two L2 Firewalls

In this deployment mode, the APV appliances are set up in L3 mode, firewalls are set up in L2 mode, and the two APV appliances play the role of the ingress and egress nodes respectively. The interface and route configurations on the firewalls are as shown in the following figure.

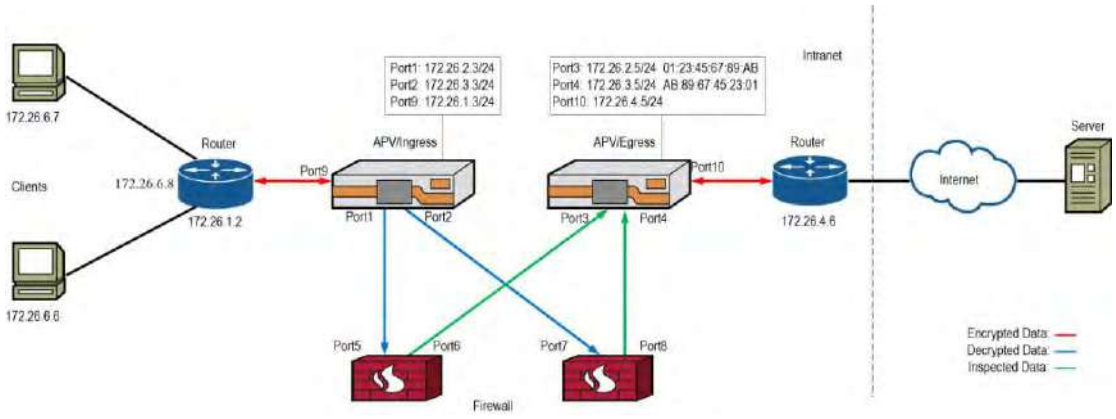

Figure 3–4 Distributed Mode: Two L3 APVs + Two L2 Firewalls

3.1.4.1 Configuring the Ingress Node

3.1.4.1.1 Address and Route Settings

1. Set the IP addresses of Port1, Port2, and Port9.

```
AN(config)#ip address port1 172.26.2.3 24
AN(config)#ip address port2 172.26.3.3 24
AN(config)#ip address port9 172.26.1.3 24
```

2. Set the default route.

```
AN(config)#ip route default 172.26.1.2
```

3. Enable the IPflow function.

```
AN(config)#ip ipflow on
```

4. Define Eroutes.

```
AN(config)#ip eroute er1 1900 172.26.0.0 255.255.0.0 0 0.0.0.0 0.0.0.0 0 172.26.2.5
AN(config)#ip eroute er2 1900 172.26.0.0 255.255.0.0 0 0.0.0.0 0.0.0.0 0 172.26.3.5
```

3.1.4.1.2 Load Balance Settings

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Create a TCPS virtual service.

```
AN(config)#slb virtual tcps vs1 0.0.0.0 443 noarp 0
```

3. Create FWDMAC real services.

```
AN(config)#slb real fwdmac port1 01:23:45:67:89:AB 8443
AN(config)#slb real fwdmac port2 AB:89:67:45:23:01 8443
```

4. Create an L2 real service group using the chi method and add “rs1” and “rs2” to this group.

```
AN(config)#slb group method chi_group chi
AN(config)#slb group member chi_group rs1
AN(config)#slb group member chi_group rs2
```

5. Configure a default policy to associate “vs1” with “chi_group”.

```
AN(config)#slb policy default vs1 chi_group
```

6. Configure health checks for “rs1” and “rs2” to ensure that Port3 and Port4 on the egress node are accessible (a health check reflector is needed on the egress node).

```
AN(config)#slb real health a1 rs1 172.26.2.5 56789 tcp 3 3
AN(config)#slb real health a2 rs2 172.26.3.5 56789 tcp 3 3
```

3.1.4.1.3 SSL Interception Settings

1. Create an SSL virtual host and associate it with “vs1”.

```
AN(config)#ssl host virtual vhost1 vs1
```

2. Enable SSL interception for “vhost1”.

```
AN(config)#ssli on vhost1 0
```

3. Generate SSL interception certificates for “vhost1”, activate them and enable “vhost1”.

Note: The generated CA certificates must also be imported into the trusted CA list of clients’ browsers.

```
AN(config)#ssli cacert rsa vhost1 2048 1 1
AN(config)#ssli cacert ecc vhost1 prime256v1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

Note: Currently, elliptic curve secp521r1 is not widely supported by mainstream browsers. It is recommended to set prime256v1 or secp384r1 to avoid connection failures.

3.1.4.2 Configuring the Egress Node

3.1.4.2.1 Address and Route Settings

1. Set the IP addresses of Port3, Port4 and Port10.

```
AN(config)#ip address port3 172.26.2.5 24
AN(config)#ip address port4 172.26.3.5 24
AN(config)#ip address port10 172.26.4.5 24
```

2. Set the default route

```
AN(config)#ip route default 172.26.4.6
```

3. Enable RTS.

```
AN(config)#ip rts on
```

4. Define Eroutes.

```
AN(config)#ip eroute er3 1900 0.0.0.0 0.0.0.0 0 172.26.0.0 255.255.0.0 0 any 172.26.2.3
AN(config)#ip eroute er4 1900 0.0.0.0 0.0.0.0 0 172.26.0.0 255.255.0.0 0 any 172.26.3.3
```

3.1.4.2.2 Load Balance Settings

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Configure a TCPS real service, and configure it to keep destination IP addresses unchanged when forwarding packets.

```
AN(config)#slb real tcps rs3 172.26.4.6 443 icmp
AN(config)#slb real settings keepdip rs3
```

3. Configure a TCP virtual service and enable RTS for it.

```
AN(config)#slb virtual tcp vs2 0.0.0.0 8443 noarp 0
AN(config)#slb virtual settings rts vs2
```

4. Configure a static policy to associate “vs2” with “rs3”.

```
AN(config)#slb policy static vs2 rs3
```

5. Create a health check reflector “reflector1”.

```
AN(config)#health ipreflect reflector1 0.0.0.0 56789 tcp
```

3.1.4.2.3 SSL Interception Settings

1. Create an SSL real host “rhost1” and associate it with “rs3”.

```
AN(config)#ssl host real rhost1 rs3
```

2. Enable SSL interception for “rhost1” and enable “rhost1”.

```
AN(config)#ssli on rhost1 0
AN(config)#ssl start rhost1
```

3.2 Deployment on APV Working in L2 Mode

3.2.1 Integrated Mode: One L2 APV + One L2 Firewall (With VLAN)

In this deployment mode, both the APV appliance and the firewall are set up in L2 mode, and the APV appliance serves as both the ingress and egress nodes. Two bridge instances are configured on the APV appliance:

- One bridge is used to transfer SSL traffic to the SSL interception module for decryption and then forward the decrypted traffic to the firewall.
- The other bridge is used to receive the inspected traffic from the firewall, re-encrypt the traffic and then forward it to the server.
- Port1, Port2, Port3 and Port4 can receive and send packets carrying VLAN tags 1000 and 1001.

The network topology and interface configurations are as shown in the following figure.

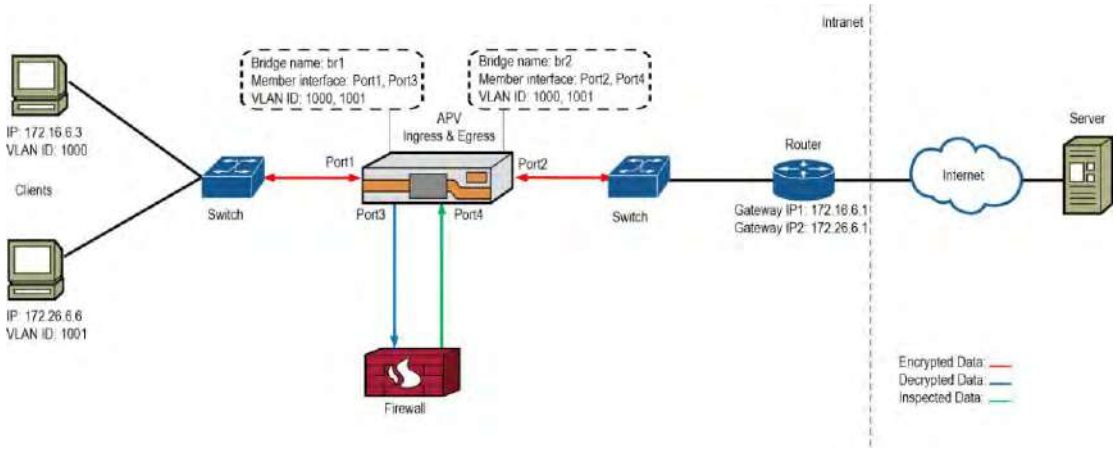

Figure 3–5 Integrated Mode: One L2 APV + One L2 Firewall

3.2.1.1 Bridge Settings

1. Create two bridge instances.

```
AN(config)#bridge name br1
AN(config)#bridge name br2
```

2. Add members to the created bridge instances.

```
AN(config)#bridge member br1 port1 yes
AN(config)#bridge member br1 port3 yes
AN(config)#bridge member br2 port2 yes
AN(config)#bridge member br2 port4 yes
```

3. Set VLAN tags for the bridge member interfaces.

If packets passing through a member interface carry VLAN tags, you need to set the corresponding VLAN IDs in order for the interface to receive and send tagged packets.

```
AN(config)#bridge vlan br1 port1 1000
```

```
AN(config)#bridge vlan br1 port3 1000
AN(config)#bridge vlan br2 port2 1000
AN(config)#bridge vlan br2 port4 1000
AN(config)#bridge vlan br1 port1 1001
AN(config)#bridge vlan br1 port3 1001
AN(config)#bridge vlan br2 port2 1001
AN(config)#bridge vlan br2 port4 1001
```

4. Create filter rules to bypass returned SSL traffic and to acquire server certificates.

```
AN(config)#bridge apprule br1 0.0.0.0 443 0.0.0.0 0 tcp
```

5. Create filter rules to forward all SSL traffic (including encrypted and cleartext traffic) to the SSL interception module.

```
AN(config)#bridge apprule br1 0.0.0.0 0 0.0.0.0 443 tcp
AN(config)#bridge apprule br1 0.0.0.0 8443 0.0.0.0 0 tcp
AN(config)#bridge apprule br2 0.0.0.0 443 0.0.0.0 0 tcp
AN(config)#bridge apprule br2 0.0.0.0 0 0.0.0.0 8443 tcp
```

3.2.1.2 SSL Settings

3.2.1.2.1 Forwarding of Received SSL Traffic to the Security Device

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Create a FWDMAC real service.

   Note that “AB:89:67:45:23:01” does not represent any port. It can be replaced with an arbitrary MAC address, but it must be set.

```
AN(config)#slb real fwdmac rs1 port3 AB:89:67:45:23:01 8443
```

3. Create a real service group using the chi method and add “rs1” to this group.

```
AN(config)#slb group method chi_group chi
AN(config)#slb group member chi_group rs1
```

4. Create a TCPS virtual service.

```
AN(config)#slb virtual tcps vs1 0.0.0.0 443 noarp 0
```

5. Configure a default policy to associate “vs1” with “chi_group”.

```
AN(config)#slb policy default vs1 chi_group
```

6. Create an SSL virtual host and associate it with “vs1”.

```
AN(config)#ssl host virtual vhost1 vs1
```

3.2.1.2.2 Forwarding of Inspected SSL Traffic to the Real Service

1. Create a TCPS real service, and configure it to keep destination IP addresses unchanged when forwarding packets.

   Note that “172.26.6.1” can be replaced with an arbitrary IP address.

```
AN(config)#slb real tcps rs2 172.26.6.1 443 none
AN(config)#slb real settings keepdip rs2
```

2. Create a TCP virtual service.

```
AN(config)#slb virtual tcp vs2 0.0.0.0 8443 noarp 0
```

3. Configure a static policy to associate “vs2” with “rs2”.

```
AN(config)#slb policy static vs2 rs2
```

4. Disable the real service health check.

```
AN(config)#health off
```

5. Create an SSL real host “rhost1” and associate it with “rs2”.

```
AN(config)#ssl host real rhost1 rs2
```

3.2.1.3 SSL Interception Settings

1. Enable SSL interception for “vhost1”.

```
AN(config)#ssli on vhost1 1
```

2. Generate SSL interception certificates for “vhost1”, activate them and enable “vhost1”.

Note: The generated CA certificates must also be imported into the trusted CA list of clients’ browsers.

```
AN(config)#ssli cacert rsa vhost1 2048 1 1
AN(config)#ssli cacert ecc vhost1 prime256v1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

Note: Currently, elliptic curve secp521r1 is not widely supported by mainstream browsers. It is recommended to set prime256v1 or secp384r1 to avoid connection failures.

3. Enable SSL interception for “rhost1” and enable “rhost1”.

```
AN(config)#ssli on rhost1 1
AN(config)#ssl start rhost1
```

Note: If the SSL server uses a testing certificate or a self-signed certificate, both the ingress and egress nodes must have its certificate chain (including the root CA certificate) imported. For testing-only purposes, administrators can choose not to import the certificate chain, but must execute the "**ssl globals verifycert off**" command to disable the server authentication function.

3.2.2 Integrated Mode: One L2 APV + One L2 Firewall (Without VLAN)

In this deployment mode, both the APV appliance and the firewall are set up in L2 mode, and the APV appliance serves as both the ingress and egress nodes. Two bridge instances are configured on the APV appliance:

- One bridge is used to transfer SSL traffic to the SSL interception module for decryption and then forward the decrypted traffic to the firewall.
- The other bridge is used to receive the inspected traffic from the firewall, re-encrypt the traffic and then send it to the server.

The network topology and interface configurations are as shown in the following figure.

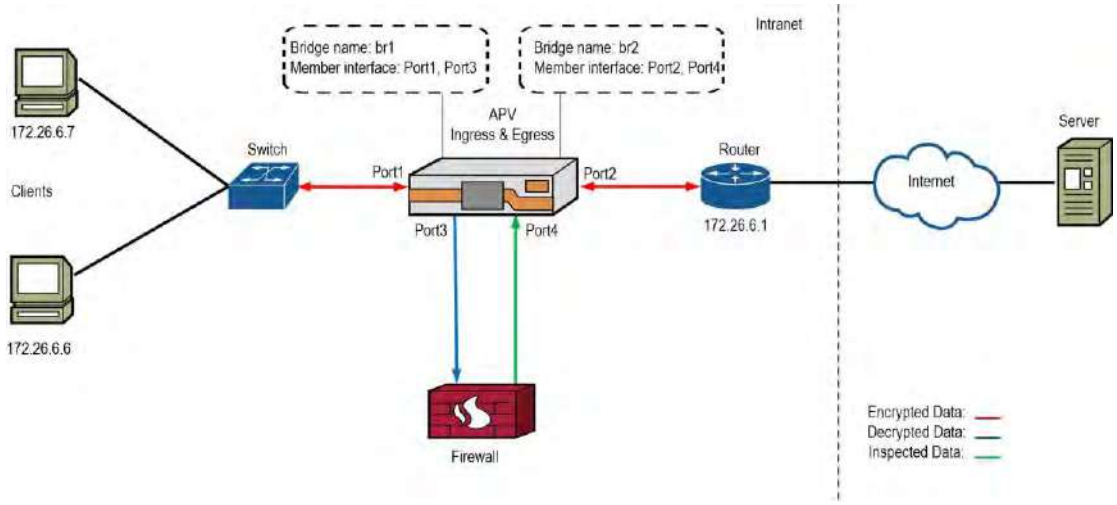

Figure 3–6 Integrated Mode: One L2 APV + One L2 Firewall

3.2.2.1 Bridge Settings

1. Create two bridge instances.

```
AN(config)#bridge name br1
AN(config)#bridge name br2
```

2. Add members to the created bridge instances.

```
AN(config)#bridge member br1 port1 yes
AN(config)#bridge member br1 port3 yes
AN(config)#bridge member br2 port2 yes
AN(config)#bridge member br2 port4 yes
```

3. Create filter rules to bypass returned SSL traffic and to acquire server certificates.

```
AN(config)#bridge apprule br1 0.0.0.0 443 0.0.0.0 0 tcp
```

4. Create filter rules to forward all SSL traffic (including encrypted and cleartext traffic) to the SSL interception module.

```
AN(config)#bridge apprule br1 0.0.0.0 0 0.0.0.0 443 tcp
AN(config)#bridge apprule br1 0.0.0.0 8443 0.0.0.0 0 tcp
AN(config)#bridge apprule br2 0.0.0.0 443 0.0.0.0 0 tcp
AN(config)#bridge apprule br2 0.0.0.0 0 0.0.0.0 8443 tcp
```

3.2.2.2 SSL Settings

3.2.2.2.1 Forwarding of Received SSL Traffic to the Security Device

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Create a FWDMAC real service.

   Note that “AB:89:67:45:23:01” does not represent any port. It can be replaced with an arbitrary MAC address, but it must be set.

```
AN(config)#slb real fwdmac rs1 port3 AB:89:67:45:23:01 8443
```

3. Create a real service group using the chi method and add “rs1” to this group.

```
AN(config)#slb group method chi_group chi
AN(config)#slb group member chi_group rs1
```

4. Create a TCPS virtual service.

```
AN(config)#slb virtual tcps vs1 0.0.0.0 443 noarp 0
```

5. Configure a default policy to associate “vs1” with “chi_group”.

```
AN(config)#slb policy default vs1 chi_group
```

6. Create an SSL virtual host and associate it with “vs1”.

```
AN(config)#ssl host virtual vhost1 vs1
```

3.2.2.2.2 Forwarding of Inspected SSL Traffic to the Real Service

1. Create a TCPS real service, and configure it to keep destination IP addresses unchanged when forwarding packets.

   Note that “172.26.6.1” can be replaced with an arbitrary IP address.

```
AN(config)#slb real tcps rs2 172.26.6.1 443 none
AN(config)#slb real settings keepdip rs2
```

2. Create a TCP virtual service.

```
AN(config)#slb virtual tcp vs2 0.0.0.0 8443 noarp 0
```

3. Configure a static policy to associate “vs2” with “rs2”.

```
AN(config)#slb policy static vs2 rs2
```

4. Disable real service health check.

```
AN(config)#health off
```

5. Create an SSL real host “rhost1” and associate it with “rs2”.

```
AN(config)#ssl host real rhost1 rs2
```

3.2.2.3 SSL Interception Settings

1. Enable SSL interception for “vhost1”.

```
AN(config)#ssli on vhost1 1
```

2. Generate SSL interception certificates for “vhost1”, activate them and enable “vhost1”.

Note: The generated CA certificates must also be imported into the trusted CA list of clients’ browsers.

```
AN(config)#ssli cacert rsa vhost1 2048 1 1
AN(config)#ssli cacert ecc vhost1 prime256v1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

Note: Currently, elliptic curve secp521r1 is not widely supported by mainstream browsers. It is recommended to set prime256v1 or secp384r1 to avoid connection failures.

3. Enable SSL interception for “rhost1” and enable “rhost1”.

```
AN(config)#ssli on rhost1 1
AN(config)#ssl start rhost1
```

Note: If the SSL server uses a testing certificate or a self-signed certificate, both the ingress and egress nodes must have its certificate chain (including the root CA certificate) imported. For testing-only purposes, administrators can choose not to import the certificate chain, but must execute the “**ssl globals verifycert off**” command to disable the server authentication function.

3.2.3 Distributed Mode: Two L2 APVs + One L2 Firewall (With VLAN)

In this deployment mode, both the APV appliances and the firewall work in L2 mode, and the two APV appliances play the role of the ingress and egress nodes respectively. The ingress node and the egress node each have a bridge instance configured:

- The bridge on the ingress node is used to transfer SSL traffic to the SSL interception module for decryption and then forward the decrypted traffic to the firewall.

- The bridge on the egress node is used to receive the inspected traffic from the firewall, re-encrypt the traffic and then send it to the server.
- Port1, Port2, Port3 and Port4 can receive and send packets carrying VLAN tags 1000 and 1001.

The network topology and interface are as shown in the following figure.

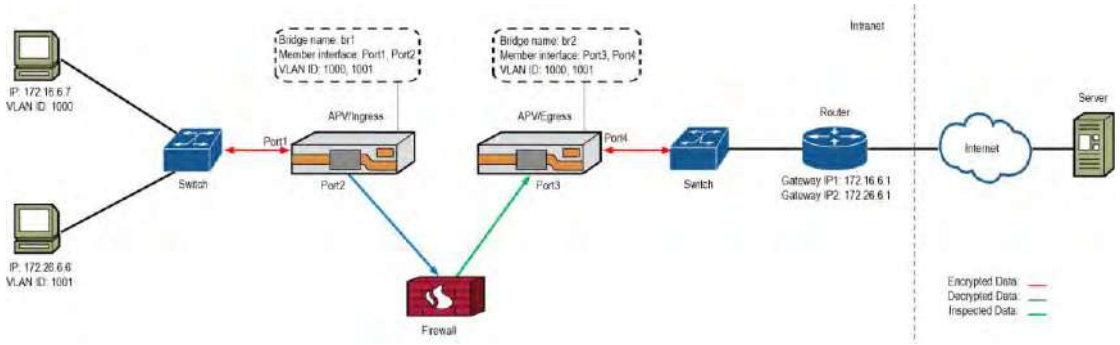

Figure 3–7 Distributed Mode: Two L2 APVs + One L2 Firewall

3.2.3.1 Configuring the Ingress Node

3.2.3.1.1 Bridge Settings

1. Create a bridge instance.

```
AN(config)#bridge name br1
```

2. Add members to the created bridge instance.

```
AN(config)#bridge member br1 port1 yes
AN(config)#bridge member br1 port2 yes
```

3. Set VLAN tags for the bridge member interfaces.

If packets passing through a member interface carry VLAN tags, you need to set the corresponding VLAN IDs in order for the interface to receive and send tagged packets.

```
AN(config)#bridge vlan br1 port1 1000
AN(config)#bridge vlan br1 port1 1001
AN(config)#bridge vlan br1 port2 1000
AN(config)#bridge vlan br1 port2 1001
```

4. Create filter rules to bypass returned SSL traffic and to acquire server certificates.

```
AN(config)#bridge apprule br1 0.0.0.0 443 0.0.0.0 0 tcp
```

5. Create filter rules to forward clients' encrypted SSL traffic and servers' cleartext SSL traffic to the SSL interception module.

```
AN(config)#bridge apprule br1 0.0.0.0 0 0.0.0.0 443 tcp
AN(config)#bridge apprule br1 0.0.0.0 8443 0.0.0.0 0 tcp
```

3.2.3.1.2 SSL Settings

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Create a TCPS virtual service.

```
AN(config)#slb virtual tcps vs1 0.0.0.0 443 noarp 0
```

3. Create a FWDMAC real service.

   Note that “AB:89:67:45:23:01” does not represent any port. It can be replaced with an arbitrary MAC address, but it must be set.

```
AN(config)#slb real fwdmac rs1 port2 AB:89:67:45:23:01 8443
```

4. Create an L2 real service group using the chi method and add “rs1” to this group.

```
AN(config)#slb group method chi_group chi
AN(config)#slb group member chi_group rs1
```

5. Configure a default policy to associate “vs1” with “chi_group”.

```
AN(config)#slb policy default vs1 chi_group
```

6. Create an SSL virtual host and associate it with “vs1”.

```
AN(config)#ssl host virtual vhost1 vs1
```

3.2.3.1.3 SSL Interception Settings

1. Enable SSL interception for “vhost1”.

```
AN(config)#ssli on vhost1 0
```

2. Generate SSL interception certificates for “vhost1”, activate them and enable “vhost1”.

```
AN(config)#ssli cacert rsa vhost1 2048 1 1
AN(config)#ssli cacert ecc vhost1 prime256v1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

Note:

1. The generated CA certificates must also be imported into the trusted CA list of clients’ browsers.
2. If the SSL server uses a testing certificate or a self-signed certificate, both the ingress and egress nodes must have its certificate chain (including the root CA certificate) imported. For testing-only purposes, administrators can choose not to import the certificate chain, but must execute the “**ssl globals verifycert off**” command to disable the server authentication function.
3. Currently, elliptic curve secp521r1 is not widely supported by mainstream

browsers. It is recommended to set prime256v1 or secp384r1 to avoid connection failures.

3.2.3.2 Configuring the Egress Node

3.2.3.2.1 Bridge Settings

1. Create a bridge instance.

```
AN(config)#bridge name br2
```

2. Add members to the created bridge instance.

```
AN(config)#bridge member br2 port3 yes
AN(config)#bridge member br2 port4 yes
```

3. Set VLAN tags for the bridge member interfaces.

If packets passing through a member interface carry VLAN tags, you need to set the corresponding VLAN IDs for it to allow the interface to receive and send tagged packets.

```
AN(config)#bridge vlan br2 port3 1000
AN(config)#bridge vlan br2 port3 1001
AN(config)#bridge vlan br2 port4 1000
AN(config)#bridge vlan br2 port4 1001
```

4. Create filter rules to forward servers' encrypted SSL traffic and clients' cleartext SSL traffic to the SSL interception module.

```
AN(config)#bridge apprule br2 0.0.0.0 443 0.0.0.0 0 tcp
AN(config)#bridge apprule br2 0.0.0.0 0 0.0.0.0 8443 tcp
```

3.2.3.2.2 SSL Settings

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Configure a TCPS real service, and configure it to keep destination IP addresses unchanged when forwarding packets.

   Note that "172.26.6.1" can be replaced with an arbitrary IP address.

```
AN(config)#slb real tcps rs2 172.26.6.1 443 none
AN(config)#slb real settings keepdip rs2
```

3. Configure a TCP virtual service.

```
AN(config)#slb virtual tcp vs2 0.0.0.0 8443 noarp 0
```

4. Configure a static policy to associate "vs2" with "rs2".

```
AN(config)#slb policy static vs2 rs2
```

5. Disable the real service health check.

```
AN(config)#health off
```

6. Create an SSL real host “rhost1” and associate it with “rs2”.

```
AN(config)#ssl host real rhost1 rs2
```

3.2.3.2.3 SSL Interception Settings

7. Enable SSL interception for “rhost1” and enable “rhost1”.

```
AN(config)#ssli on rhost1 0
AN(config)#ssl start rhost
```

Note: If the SSL server uses a testing certificate or a self-signed certificate, both the ingress and egress nodes must have its certificate chain (including the root CA certificate) imported. For testing-only purposes, administrators can choose not to import the certificate chain, but must execute the “**ssl globals verifycert off**” command to disable the server authentication function.

3.2.4 Distributed Mode: Two L2 APVs + One L2 Firewall (Without VLAN)

In this deployment mode, both the APV appliances and the firewall are set up in L2 mode, and the two APV appliances play the role of the ingress and egress nodes respectively. The ingress node and the egress node each have a bridge instance configured:

- The bridge on the ingress node is used to transfer SSL traffic to the SSL interception module for decryption and forward the decrypted traffic to the firewall.
- The bridge on the egress node is used to receive the inspected traffic from the firewall, re-encrypt the traffic and then send it to the server.

The network topology and interface are as shown in the following figure.

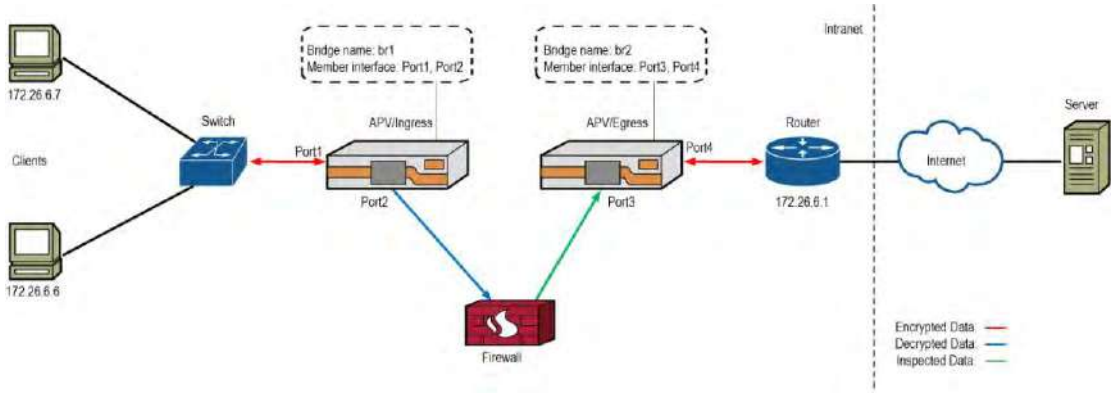

Figure 3–8 Distributed Mode: Two L2 APVs + One L2 Firewall (Without VLAN)

3.2.4.1 Configuring the Ingress Node

3.2.4.1.1 Bridge Settings

1. Create a bridge instance.

```
AN(config)#bridge name br1
```

2. Add members to the created bridge instance.

```
AN(config)#bridge member br1 port1 yes
AN(config)#bridge member br1 port2 yes
```

3. Create filter rules to bypass returned SSL traffic and to acquire server certificates.

```
AN(config)#bridge apprule br1 0.0.0.0 443 0.0.0.0 0 tcp
```

4. Create filter rules to forward clients' encrypted SSL traffic and servers' cleartext SSL traffic to the SSL interception module.

```
AN(config)#bridge apprule br1 0.0.0.0 0 0.0.0.0 443 tcp
AN(config)#bridge apprule br1 0.0.0.0 8443 0.0.0.0 0 tcp
```

3.2.4.1.2 SSL Settings

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Create a TCPS virtual service.

```
AN(config)#slb virtual tcps vs1 0.0.0.0 443 noarp 0
```

3. Create a FWDMAC real service.

   Note that "AB:89:67:45:23:01" does not represent any port. It can be replaced with an arbitrary MAC address, but it must be set.

```
AN(config)#slb real fwdmac rs1 port2 AB:89:67:45:23:01 8443
```

4. Create an L2 real service group using the chi method and add "rs1" to this group.

```
AN(config)#slb group method chi_group chi
AN(config)#slb group member chi_group rs1
```

5. Configure a default policy to associate "vs1" with "chi_group".

```
AN(config)#slb policy default vs1 chi_group
```

6. Create an SSL virtual host and associate it with "vs1".

```
AN(config)#ssl host virtual vhost1 vs1
```

3.2.4.1.3 SSL Interception Settings

1. Enable SSL interception for “vhost1”.

```
AN(config)#ssli on vhost1 0
```

2. Generate SSL interception certificates for “vhost1”, activate them and enable “vhost1”.

```
AN(config)#ssli cacert rsa vhost1 2048 1 1
AN(config)#ssli cacert ecc vhost1 prime256v1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

Note:

1. The generated CA certificates must also be imported into the trusted CA list of clients’ browsers.
2. If the SSL server uses a testing certificate or a self-signed certificate, both the ingress and egress nodes must have its certificate chain (including the root CA certificate) imported. For testing-only purposes, administrators can choose not to import the certificate chain, but must execute the “**ssl globals verifycert off**” command to disable the server authentication function.
3. Currently, elliptic curve secp521r1 is not widely supported by mainstream browsers. It is recommended to set prime256v1 or secp384r1 to avoid connection failures.

3.2.4.2 Configuring the Egress Node

3.2.4.2.1 Bridge Settings

1. Create a bridge instance.

```
AN(config)#bridge name br2
```

2. Add members to the created bridge instance.

```
AN(config)#bridge member br2 port3 yes
AN(config)#bridge member br2 port4 yes
```

3. Create filter rules to forward servers’ encrypted SSL traffic and clients’ cleartext SSL traffic to the SSL interception module.

```
AN(config)#bridge apprule br2 0.0.0.0 443 0.0.0.0 0 tcp
AN(config)#bridge apprule br2 0.0.0.0 0 0.0.0.0 8443 tcp
```

3.2.4.2.2 SSL Settings

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Configure a TCPS real service, and configure it to keep destination IP addresses unchanged when forwarding packets.

   Note that “172.26.6.1” can be replaced with an arbitrary IP address.

```
AN(config)#slb real tcps rs2 172.26.6.1 443 none
AN(config)#slb real settings keepdip rs2
```

3. Configure a TCP virtual service.

```
AN(config)#slb virtual tcp vs2 0.0.0.0 8443 noarp 0
```

4. Configure a static policy to associate “vs2” with “rs2”.

```
AN(config)#slb policy static vs2 rs2
```

5. Disable the real service health check.

```
AN(config)#health off
```

6. Create an SSL real host “rhost1” and associate it with “rs2”.

```
AN(config)#ssl host real rhost1 rs2
```

3.2.4.2.3 SSL Interception Settings

7. Enable SSL interception for “rhost1” and enable “rhost1”.

```
AN(config)#ssli on rhost1 0
AN(config)#ssl start rhost
```

Note: If the SSL server uses a testing certificate or a self-signed certificate, both the ingress and egress nodes must have its certificate chain (including the root CA certificate) imported. For testing-only purposes, administrators can choose not to import the certificate chain, but must execute the “**ssl globals verifycert off**” command to disable the server authentication function.

4 SSL Interception Integrated with Webagent

SSL interception can also be deployed in scenarios where an APV functions as a Webagent service to implement explicit forward proxy. In such scenarios, SSL interception also supports both the integrated mode and the distributed mode. The deployment of SSL interception combined with Webagent can be completed on one, two or three APV appliances:

- The Webagent service can be deployed on an independent APV, so this APV only plays the role of an explicit forward proxy. SSL interception can be integrated on another independent APV or distributed on two independent APVs.
- The Webagent service can integrate with the ingress node, so one APV can play both the role of an explicit forward proxy and an ingress node, and another APV can play the role of an egress node.
- The Webagent service can also integrate with both the ingress node and the egress node on a single APV.

This chapter provides examples for configuring SSL interception in explicit forward proxies.

- 4.1 Distributed Mode: Three L3 APVs + Two L3 Firewalls
- 4.2 Distributed Mode: Two L3 APVs + Two L3 Firewalls
- 4.3 Integrated Mode: One L3 APV + Two L3 Firewalls
- 4.4 Distributed Mode: Three L3 APVs +Two L2 Firewalls
- 4.5 Distributed Mode: Two L3 APVs + Two L2 Firewalls
- 4.6 Integrated Mode: One L3 APV + Two L2 Firewalls

4.1 Distributed Mode: Three L3 APVs + Two L3 Firewalls

In this deployment mode, both the APV appliances and the firewalls are set up in L3 mode, and two of the APV appliances play the role of the ingress and egress nodes respectively. Another APV appliance plays the role of a Webagent service. The clients should set the APV appliance that is serving as the Webagent service as the proxy server. The interface and route configurations on the firewalls are as shown in the following figure.

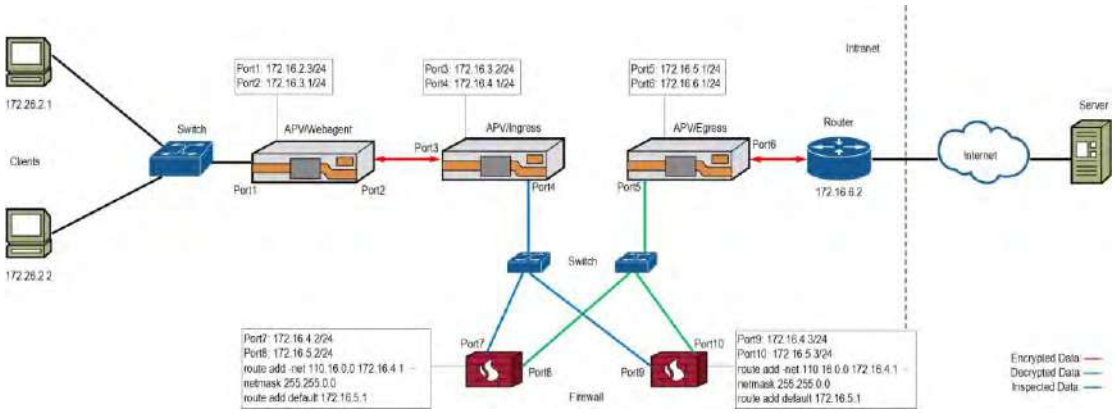

Figure 4–1 Distributed Mode: Three L3 APVs + Two L3 Firewalls

4.1.1 Configuring the Webagent

4.1.1.1 Address and Route Settings

1. Set the IP addresses of Port1 and Port2.

```
AN(config)#ip address port1 172.16.2.3 24
AN(config)#ip address port2 172.16.3.1 24
```

2. Configure an MNET.

```
AN(config)#mnet port2 mport2
AN(config)#ip address mport2 110.16.10.1
```

3. Configure an IP pool

```
AN(config)#ip pool p1 110.16.10.1
AN(config)#slb proxyip global p1
```

4. Set the default route.

```
AN(config)#ip route default 172.16.3.2
```

4.1.1.2 Webagent Service Settings

1. Configure a DNS server.

```
AN(config)#ip nameserver 10.8.80.10
```

2. Configure a Webagent service.

```
AN(config)#webagent service w1 172.16.2.4 8000
```

4.1.2 Configuring the Ingress Node

4.1.2.1 Address and Route Settings

1. Set the IP addresses of Port3 and Port4.

```
AN(config)#ip address port3 172.16.3.2 24
```

```
AN(config)#ip address port4 172.16.4.1 24
```

2. Define Eroutes.

```
AN(config)#ip eroute er1 1900 110.16.0.0 255.255.0.0 0 0.0.0.0 0.0.0.0 0 any 172.16.4.2
AN(config)#ip eroute er2 1900 110.16.0.0 255.255.0.0 0 0.0.0.0 0.0.0.0 0 any 172.16.4.3
```

3. Enable the IPflow function.

```
AN(config)#ip ipflow on
```

4. Add a static route.

```
AN(config)#ip route static 110.16.0.0 255.255.0.0 172.16.3.1
```

4.1.2.2 Load Balance Settings

1. Set the system mode to reverse.

```
AN(config)#system mode reverse
```

2. Create FWDIP real services and add them to a real service group using the chi method.

```
AN(config)#slb real fwdip rs1 172.16.4.2 8443
AN(config)#slb real fwdip rs2 172.16.4.3 8443
AN(config)#slb group method g1 chi
AN(config)#slb group member g1 rs1
AN(config)#slb group member g1 rs2
```

3. Configure a TCPS virtual service.

```
AN(config)#slb virtual tcps vs1 0.0.0.0 443 noarp
```

4. Configure a default policy to associate “vs1” with “g1”.

```
AN(config)#slb policy default vs1 g1
```

4.1.2.3 SSL Interception Settings

1. Create an SSL virtual host and associate it with “vs1”.

```
AN(config)#ssl host virtual vhost1 vs1
```

2. Enable SSL interception for “vhost1”.

```
AN(config)#ssli on vhost1 0
```

3. Generate SSL interception certificates for “vhost1”, activate them and enable “vhost1”.

Note: The generated CA certificates must also be imported into the trusted CA list of clients’ browsers.

```
AN(config)#ssli cacert rsa vhost1 2048 1 1
AN(config)#ssli cacert ecc vhost1 prime256v1 1
```

```
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

Note: Currently, elliptic curve secp521r1 is not widely supported by mainstream browsers. It is recommended to set prime256v1 or secp384r1 to avoid connection failures.

4.1.3 Configuring the Egress Node

4.1.3.1 Address and Route Settings

1. Set the IP addresses of Port5 and Port6.

```
AN(config)#ip address port5 172.16.5.1 24
AN(config)#ip address port6 172.16.6.1 24
```

2. Define Eroutes.

```
AN(config)#ip eroute er1 1900 0.0.0.0 0.0.0.0 0 110.16.0.0 255.255.255.0 0 any 172.16.5.2
AN(config)#ip eroute er1 1900 0.0.0.0 0.0.0.0 0 110.16.0.0 255.255.255.0 0 any 172.16.5.3
```

3. Enable RTS.

```
AN(config)#ip rts on
```

4.1.3.2 Load Balance Settings

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Create a TCPS real service and configure it to keep destination IP addresses unchanged when forwarding packets.

```
AN(config)#slb real tcps rs1 172.16.6.2 443 icmp
AN(config)#slb real settings keepdip rs1
```

3. Configure a TCP virtual service.

```
AN(config)#slb virtual tcp vs1 0.0.0.0 8443 noarp
```

4. Configure a static policy to associate “vs1” with “rs1”.

```
AN(config)#slb policy default vs1 rs1
```

4.1.3.3 SSL Interception Settings

1. Create an SSL real host “rhost1” and associate it with “rs1”.

```
AN(config)#ssl host real rhost1 rs1
```

2. Enable SSL interception for “rhost1” and enable “rhost1”.

```
AN(config)#ssli on rhost1 0
```

```
AN(config)#ssl start rhost1
```

4.2 Distributed Mode: Two L3 APVs + Two L3 Firewalls

In this deployment mode, both the APV appliances and firewalls are set up in L3 mode, and the two APV appliances play the role of the ingress and egress nodes respectively. The ingress node also plays the role of the Webagent service. The clients should set the APV appliance as the proxy server. The interface and route configurations on the firewalls are as shown in the following figure.

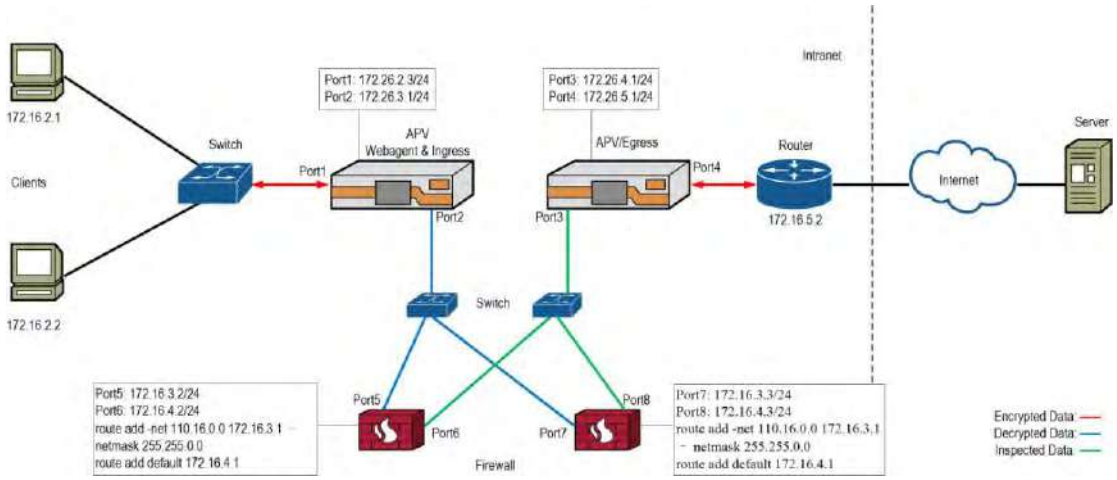

Figure 4–2 Distributed Mode: Two L3 APVs + Two L3 Firewalls

4.2.1 Configuring the Ingress Node

4.2.1.1 Address and Route Settings

1. Set the IP addresses of Port1 and Port2.

```
AN(config)#ip address port1 172.16.2.3 24
AN(config)#ip address port2 172.16.3.1 24
```

2. Configure an MNET.

```
AN(config)#mnet port2 mport2
AN(config)#mport2 110.16.10.1
```

3. Configure an IP pool

```
AN(config)#ip pool p1 110.16.10.1
AN(config)#slb proxyip global p1
```

4.2.1.2 Load Balance Settings

1. Set the system mode to reverse.

```
AN(config)#system mode reverse
```

2. Create FWDIP real services and add them to two real service groups using the rr method. The group “g-ssl” is used to distribute SSL traffic to the security devices. The group “g-clear” is used to distribute non-SSL traffic to the security devices.

```
AN(config)#slb real fwdip rs-ssl1 172.16.3.2 8443
AN(config)#slb real fwdip rs-ssl2 172.16.3.3 8443
AN(config)#slb real fwdip rs-clear1 172.16.3.2 8090
AN(config)#slb real fwdip rs-clear2 172.16.3.3 8090
AN(config)#slb group method g-ssl rr
AN(config)#slb group member g-ssl rs-ssl1
AN(config)#slb group member g-ssl rs-ssl2
AN(config)#slb group method g-clear rr
AN(config)#slb group member g-clear rs-clear1
AN(config)#slb group member g-clear rs-clear2
```

3. Create a TCP and a TCPS virtual service.

```
AN(config)#slb virtual tcp vs-clear 0.0.0.0 80 noarp
AN(config)#slb virtual tcps vs-ssl 0.0.0.0 443 noarp
```

4. Configure two default policies.

```
AN(config)#slb policy default vs-clear g-clear
AN(config)#slb policy default vs-ssl g-ssl
```

4.2.1.3 Webagent Service Settings

1. Configure a DNS server.

```
AN(config)#ip nameserver 10.8.80.10
```

2. Configure the Webagent service and create two Webagent links to associate it with “vs-clear” and “vs-ssl”.

```
AN(config)#webagent service w1 172.16.2.4 8000
AN(config)#webagent link w1 vs-clear
AN(config)#webagent link w1 vs-ssl
```

4.2.1.4 SSL Interception Settings

1. Create an SSL virtual host and associate it with “vs-ssl”.

```
AN(config)#ssl host virtual vhost1 vs-ssl
```

2. Enable SSL interception for “vhost1”.

```
AN(config)#ssli on vhost1 0
```

3. Generate SSL interception certificates for “vhost1”, activate them and enable “vhost1”.

Note: The generated CA certificates must also be imported into the trusted CA list of clients' browsers.

```
AN(config)#ssli cacert rsa vhost1 2048 1 1
AN(config)#ssli cacert ecc vhost1 prime256v1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

Note: Currently, elliptic curve secp521r1 is not widely supported by mainstream browsers. It is recommended to set prime256v1 or secp384r1 to avoid connection failures.

4.2.2 Configuring the Egress Node

4.2.2.1 Address and Route Settings

1. Set the IP addresses of Port3 and Port4.

```
AN(config)#ip address port3 172.16.4.1 24
AN(config)#ip address port4 172.16.5.1 24
```

2. Configure the default route.

```
AN(config)#ip route default 172.16.5.2
```

3. Define Eroutes.

```
AN(config)#ip eroute er1 1900 0.0.0.0 0.0.0.0 0 110.16.0.0 255.255.255.0 0 any 172.16.4.2
AN(config)#ip eroute er2 1900 0.0.0.0 0.0.0.0 0 110.16.0.0 255.255.255.0 0 any 172.16.4.3
```

4. Enable RTS.

```
AN(config)#ip rts on
```

4.2.2.2 Load Balance Settings

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Create a TCP and a TCPS real service and configure them to keep destination IP unchanged when forwarding packets.

```
AN(config)#slb real tcps rs-ssl 172.16.5.2 443 icmp
AN(config)#slb real tcp rs-clear 172.16.5.2 80 icmp
AN(config)#slb real settings keepdip rs-ssl
AN(config)#slb real settings keepdip rs-clear
```

3. Create a TCP and a TCPS virtual service, and enable RTS for them.

```
AN(config)#slb virtual tcp vs-ssl 0.0.0.0 8443 noarp
AN(config)#slb virtual tcp vs-clear 0.0.0.0 8090 noarp
```

```
AN(config)#slb virtual setting rts vs-ssl
AN(config)#slb virtual setting rts vs-clear
```

4. Configure two static policies.

```
AN(config)#slb policy static vs-ssl rs-ssl
AN(config)#slb policy static vs-clear rs-clear
```

4.2.2.3 SSL Interception Settings

1. Create an SSL real host “rhost1” and associate it with “rs-ssl”.

```
AN(config)#ssl host real rhost1 rs-ssl
```

2. Enable SSL interception for “rhost1” and enable “rhost1”.

```
AN(config)#ssli on rhost1 0
AN(config)#ssl start rhost1
```

4.3 Integrated Mode: One L3 APV + Two L3 Firewalls

In this deployment mode, both the APV appliance and the firewalls are set up in L3 mode, and the APV serves as the ingress node, the egress node and the Webagent service. The clients should set the APV appliance as the proxy server. The interface and route configurations on the firewalls are as shown in the following figure.

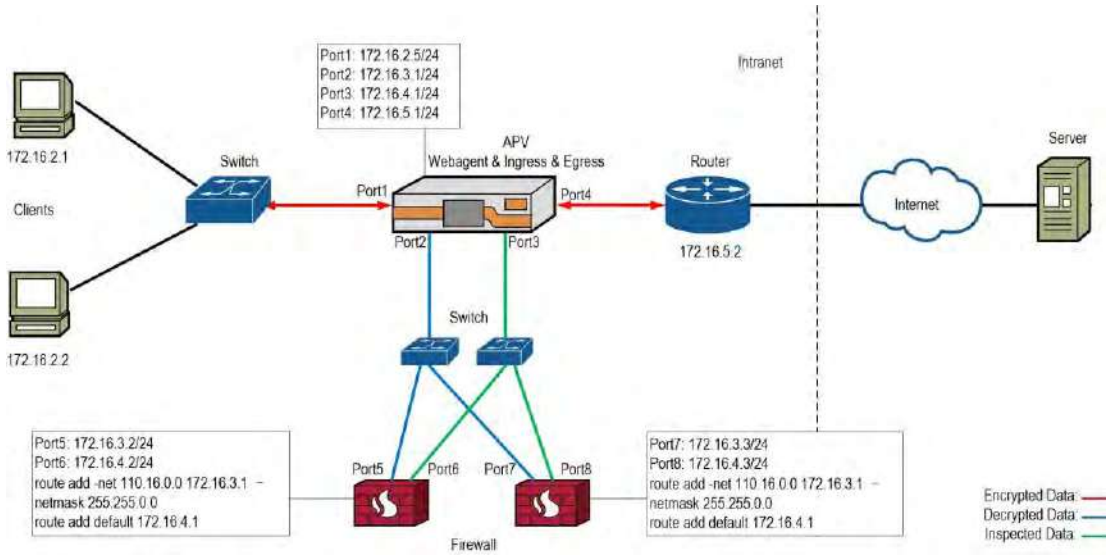

Figure 4–3 Integrated Mode: One L3 APV + Two L3 Firewalls

In this deployment mode, all configurations are performed on the APV appliance, which is functioning as the Webagent, ingress node and egress node at the same time.

4.3.1 Address and Route Settings

1. Set the IP addresses of Port1, Port2, Port3 and Port4.

```
AN(config)#ip address port1 172.16.2.5 24
AN(config)#ip address port2 172.16.3.1 24
```

```
AN(config)#ip address port3 172.16.4.1 24
AN(config)#ip address port4 172.16.5.1 24
```

2. Configure an MNET.

```
AN(config)#mnet port2 mport2
AN(config)#mport2 110.16.10.1
```

3. Configure an IP pool.

```
AN(config)#ip pool p1 110.16.10.1
AN(config)#slb proxyip global p1
```

4. Configure the default route.

```
AN(config)#ip route default 172.16.5.2
```

4.3.2 Load Balance Settings

1. Set the system mode to reverse.

```
AN(config)#system mode reverse
```

2. Create FWDIP real services and add them to two real service groups using the rr method. The group “g-ssl” is used to distribute SSL traffic to the security devices. The group “g-clear” is used to distribute non-SSL traffic to the security devices.

```
AN(config)#slb real fwdip rs-ssl1 172.16.3.2 8443
AN(config)#slb real fwdip rs-ssl2 172.16.3.3 8443
AN(config)#slb real fwdip rs-clear1 172.16.3.2 8090
AN(config)#slb real fwdip rs-clear2 172.16.3.3 8090
AN(config)#slb group method g-ssl rr
AN(config)#slb group method g-clear rr
AN(config)#slb group member g-ssl rs-ssl1
AN(config)#slb group member g-ssl rs-ssl2
AN(config)#slb group member g-clear rs-clear1
AN(config)#slb group member g-clear rs-clear2
```

3. Create a TCPS and a TCP virtual service.

```
AN(config)#slb virtual tcps vs-ssl1 0.0.0.0 443 noarp
AN(config)#slb virtual tcp vs-clear1 0.0.0.0 80 noarp
```

4. Configure two default policies.

```
AN(config)#slb policy default vs-ssl1 g-ssl
AN(config)#slb policy default vs-clear1 g-clear
```

5. Create a TCP and a TCPS real service, and configure them to keep the destination IP unchanged when forwarding packets.

```
AN(config)#slb real tcps rs-ssl3 172.16.5.2 443 icmp
```

```
AN(config)#slb real tcp rs-clear3 172.16.5.2 80 icmp
AN(config)#slb real settings keepdip rs-ssl3
AN(config)#slb real settings keepdip rs-clear3
```

6. Create two TCP virtual services, and enable RTS for them.

```
AN(config)#slb virtual tcp vs-ssl2 0.0.0.0 8443 noarp
AN(config)#slb virtual tcp vs-clear2 0.0.0.0 8090 noarp
AN(config)#slb virtual settings rts vs-ssl2
AN(config)#slb virtual settings rts vs-clear2
```

7. Configure two static policies.

```
AN(config)#slb policy static vs-ssl2 rs-ssl3
AN(config)#slb policy static vs-clear2 rs-clear3
```

4.3.3 Webagent Service Settings

1. Configure a DNS server.

```
AN(config)#ip nameserver 10.8.80.10
```

2. Configure the Webagent service.

```
AN(config)#webagent service w1 172.16.2.5 8000
```

4.3.4 SSL Interception Settings

1. Create an SSL virtual host and associate it with “vs-ssl1”.

```
AN(config)#ssl host virtual vhost1 vs-ssl1
```

2. Enable SSL interception for “vhost1”.

```
AN(config)#ssli on vhost1 1
```

3. Generate SSL interception certificates for “vhost1”, activate them and enable “vhost1”.

Note: The generated CA certificates must also be imported into the trusted CA list of clients’ browsers.

```
AN(config)#ssli cacert rsa vhost1 2048 1 1
AN(config)#ssli cacert ecc vhost1 prime256v1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

Note: Currently, elliptic curve secp521r1 is not widely supported by mainstream browsers. It is recommended to set prime256v1 or secp384r1 to avoid connection failures.

4. Create an SSL real host “rhost1” and associate it with “rs-ssl3”.

```
AN(config)#ssl host real rhost1 rs-ssl3
```

5. Enable SSL interception for “rhost1” and enable “rhost1”.

```
AN(config)#ssli on rhost1 1
AN(config)#ssl start rhost1
```

4.4 Distributed Mode: Three L3 APVs +Two L2 Firewalls

In this deployment mode, the APV appliancess are set up in L3 mode, the firewall are set up in L2 mode, and the two APV appliances play the role of the ingress and egress nodes respectively. Another APV appliance plays the role of a Webagent service. The clients should set the APV appliance that is serving as the Webagent service as the proxy server. The interface and route configurations on the firewalls are as shown in the following figure.

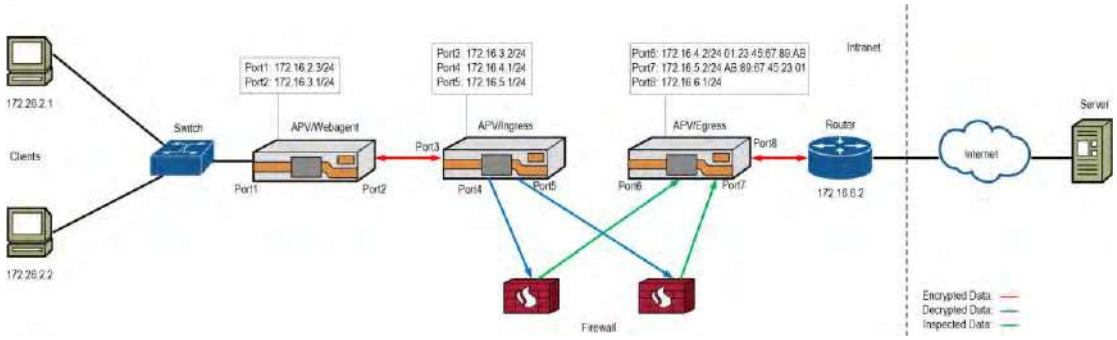

Figure 4–4 Distributed Mode: Three L3 APVs +Two L2 Firewalls

4.4.1 Configuring the Webagent

4.4.1.1 Address and Route Settings

1. Set the IP addresses of Port1 and Port2.

```
AN(config)#ip address port1 172.16.2.3 24
AN(config)#ip address port2 172.16.3.1 24
```

2. Configure an MNET.

```
AN(config)#mnet port2 mport2
AN(config)#mport2 110.16.10.1
```

3. Configure an IP pool.

```
AN(config)#ip pool p1 110.16.10.1
AN(config)#slb proxyip global p1
```

4. Configure a default route.

```
AN(config)#route default 172.16.3.2
```

4.4.1.2 Webagent Service Settings

1. Configure a DNS server.

```
AN(config)#ip nameserver 10.8.80.10
```

2. Configure a Webagent service.

```
AN(config)#webagent service w1 172.16.2.4 8000
```

4.4.2 Configuring the Ingress Node

4.4.2.1 Address and Route Settings

1. Set the IP addresses of Port3, Port4 and port 5.

```
AN(config)#ip address port3 172.16.3.2 24
AN(config)#ip address port4 172.16.4.1 24
AN(config)#ip address port5 172.16.5.1 24
```

2. Define Eroutes

```
AN(config)#ip eroute er1 1900 110.16.0.0 255.255.0.0 0 0.0.0.0 0.0.0.0 0 any 172.16.4.2
AN(config)#ip eroute er2 1900 110.16.0.0 255.255.0.0 0 0.0.0.0 0.0.0.0 0 any 172.26.5.2
```

3. Enable the IPflow function.

```
AN(config)#ip ipflow on
```

4.4.2.2 Load Balance Settings

1. Set the system mode to reverse.

```
AN(config)#system mode reverse
```

2. Create FWDMAC real services and add them to a real service group using the RR method.

```
AN(config)#slb real fwdmac rs1 port4 01:23:45:67:89:AB 8443
AN(config)#slb real fwdmac rs2 port5 AB:89:67:45:23:01 8443
AN(config)#slb group method g1 rr
AN(config)#slb group member g1 rs1
AN(config)#slb group member g1 rs2
```

3. Create a TCPS virtual service.

```
AN(config)#slb virtual tcps vs1 0.0.0.0 443 noarp
```

4. Configure a default policy.

```
AN(config)#slb policy default vs1 g1
```

4.4.2.3 SSL Interception Settings

1. Create an SSL virtual host and associate it with “vs1”.

```
AN(config)#ssl host virtual vhost1 vs1
```

2. Enable SSL interception for “vhost1”.

```
AN(config)#ssli on vhost1 0
```

3. Generate SSL interception certificates for “vhost1”, activate them and enable “vhost1”.

Note: The generated CA certificates must also be imported into the trusted CA list of clients’ browsers.

```
AN(config)#ssli cacert rsa vhost1 2048 1 1
AN(config)#ssli cacert ecc vhost1 prime256v1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

Note: Currently, elliptic curve secp521r1 is not widely supported by mainstream browsers. It is recommended to set prime256v1 or secp384r1 to avoid connection failures.

4.4.3 Configuring the Egress Node

4.4.3.1 Address and Route Settings

1. Se the IP address of Port6, Port7 and Port8.

```
AN(config)#ip address port6 172.16.4.2 24
AN(config)#ip address port7 172.16.5.2 24
AN(config)#ip address port8 172.16.6.1 24
```

2. Set the MAC address of Port6 and Port7.

```
AN(config)#interface mac port6 01:23:45:67:89:AB
AN(config)#interface mac port7 AB:89:67:45:23:01
```

3. Configure the default route.

```
AN(config)#ip route default 172.16.6.2
```

4. Define Eroutes.

```
AN(config)#ip eroute er3 1900 0.0.0.0 0.0.0.0 0 110.16.0.0 255.255.0.0 0 any 172.16.4.1
AN(config)#ip eroute er4 1900 0.0.0.0 0.0.0.0 0 110.16.0.0 255.255.0.0 0 any 172.16.5.1
```

5. Enable RTS.

```
AN(config)#ip rts on
```

4.4.3.2 Load Balance Settings

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Create a TCPS real service and configure it to keep destination IP unchanged when forwarding packets.

```
AN(config)#slb real tcps rs1 172.16.6.2 443 icmp
AN(config)#slb real settings keepdip rs1
```

3. Create a TCP virtual service and enable RTS for it.

```
AN(config)#slb virtual tcp vs1 0.0.0.0 8443 noarp
AN(config)#slb virtual settings rts vs1
```

4. Configure a static policy.

```
AN(config)#slb policy static vs1 rs1
```

4.4.3.3 SSL Interception Settings

1. Create an SSL real host “rhost1” and associate it with “rs1”.

```
AN(config)#ssl host real rhost1 rs1
```

2. Enable SSL interception for “rhost1” and enable “rhost1”.

```
AN(config)#ssli on rhost1 0
AN(config)#ssl start rhost1
```

4.5 Distributed Mode: Two L3 APVs + Two L2 Firewalls

In this deployment mode, the APV appliances work in L3 mode, firewalls are set up in L2 mode, and the two APV appliances play the role of the ingress and egress nodes respectively. The ingress node also plays the role of the Webagent service. The clients should set the ingress node as the proxy server.

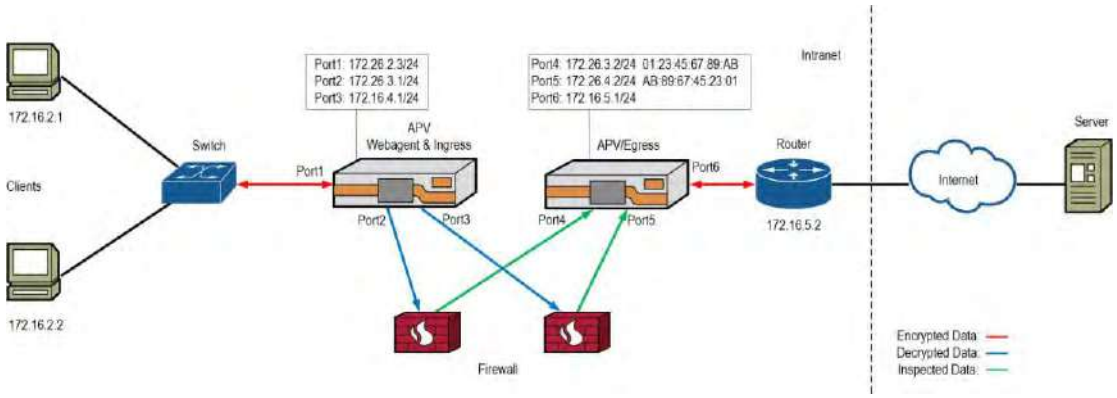

Figure 4–5 Distributed Mode: Two L3 APVs + Two L2 Firewalls

4.5.1 Configuring the Ingress Node

4.5.1.1 Address and Route Settings

1. Set the IP addresses of Port1, Port2 and Port3.

```
AN(config)#ip address port1 172.16.2.3 24
AN(config)#ip address port2 172.16.3.1 24
AN(config)#ip address port3 172.16.4.1 24
```

2. Configure an MNET.

```
AN(config)#mnet port2 mport2
```

```
AN(config)#mport2 110.16.10.1
```

3. Configure an IP pool.

```
AN(config)#ip pool p1 110.16.10.1
AN(config)#slb proxyip global p1
```

4.5.1.2 Load Balance Settings

1. Set the system mode to reverse.

```
AN(config)#system mode reverse
```

2. Create FWDMAC real services and add them to two real service groups using the rr method. The group “g-ssl” is used to distribute SSL traffic to the security devices. The group “g-clear” is used to distribute non-SSL traffic to the security devices.

```
AN(config)#slb real fwdmac rs-clear1 port2 01:23:45:67:89:AB 8090
AN(config)#slb real fwdmac rs-clear2 port3 AB:89:67:45:23:01 8090
AN(config)#slb real fwdmac rs-ssl1 port2 01:23:45:67:89:AB 8443
AN(config)#slb real fwdmac rs-ssl2 port3 AB:89:67:45:23:01 8443
AN(config)#slb group method g-ssl rr
AN(config)#slb group method g-clear rr
AN(config)#slb group member g-ssl rs-ssl1
AN(config)#slb group member g-ssl rs-ssl2
AN(config)#slb group member g-clear rs-clear1
AN(config)#slb group member g-clear rs-clear2
```

3. Create a TCP and a TCPS virtual service.

```
AN(config)#slb virtual tcp vs-clear 0.0.0.0 80 noarp
AN(config)#slb virtual tcps vs-ssl 0.0.0.0 443 noarp
```

4. Configure two default policies.

```
AN(config)#slb policy default vs-clear g-clear
AN(config)#slb policy default vs-ssl g-ssl
```

4.5.1.3 Webagent Service Settings

1. Configure a DNS server.

```
AN(config)#ip nameserver 10.8.80.10
```

2. Configure a Webagent service, and create two links to associate it with “vs-clear” and “vs-ssl”.

```
AN(config)#webagent service w1 172.16.2.4 8000
AN(config)#webagent link w1 vs-clear
AN(config)#webagent link w1 vs-ssl
```

4.5.1.4 SSL Interception Settings

1. Create an SSL virtual host and associate it with “vs-ssl”.

```
AN(config)#ssl host virtual vhost1 vs-ssl
```

2. Enable SSL interception for “vhost1”.

```
AN(config)#ssli on vhost1 0
```

3. Generate SSL interception certificates for “vhost1”, activate them and enable “vhost1”.

Note: The generated CA certificates must also be imported into the trusted CA list of clients’ browsers.

```
AN(config)#ssli cacert rsa vhost1 2048 1 1
AN(config)#ssli cacert ecc vhost1 prime256v1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

Note: Currently, elliptic curve secp521r1 is not widely supported by mainstream browsers. It is recommended to set prime256v1 or secp384r1 to avoid connection failures.

4.5.2 Configuring the Egress Node

4.5.2.1 Address and Route Settings

1. Set the IP addresses of Port4, Port5 and Port6.

```
AN(config)#ip address port4 172.16.3.2 24
AN(config)#ip address port5 172.16.4.2 24
AN(config)#ip address port6 172.16.5.1 24
```

2. Set the MAC address of Port4 and Port5.

```
AN(config)#interface mac port4 01:23:45:67:89:AB
AN(config)#interface mac port5 AB:89:67:45:23:01
```

3. Configure the default route.

```
AN(config)#ip route default 172.16.5.2
```

4. Define Eroutes

```
AN(config)#ip eroute er1 1900 0.0.0.0 0.0.0.0 0 110.16.0.0 255.255.255.0 0 any 172.16.3.1
AN(config)#ip eroute er2 1900 0.0.0.0 0.0.0.0 0 110.16.0.0 255.255.255.0 0 any 172.16.4.1
```

5. Enable RTS.

```
AN(config)#ip rts on
```

4.5.2.2 Load Balance Settings

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Create a TCP and TCPS real service, and configure them to keep destination IP unchanged when forwarding packets.

```
AN(config)#slb real tcps rs1 172.16.5.2 443 icmp
AN(config)#slb real tcp rs2 172.16.5.2 80 icmp
AN(config)#slb real settings keepdip rs1
AN(config)#slb real settings keepdip rs2
```

3. Configure two TCP virtual services and enable RTS for them.

```
AN(config)#slb virtual tcp vs1 0.0.0.0 8443 noarp
AN(config)#slb virtual tcp vs2 0.0.0.0 8090 noarp
AN(config)#slb virtual settings rts vs1
AN(config)#slb virtual settings rts vs2
```

4. Configure two static policies.

```
AN(config)#slb policy static vs1 rs1
AN(config)#slb policy static vs2 rs2
```

4.5.2.3 SSL Interception Settings

1. Create an SSL real host “rhost1” and associate it with “rs1”.

```
AN(config)#ssl host real rhost1 rs1
```

2. Enable SSL interception for “rhost1” and enable “rhost1”.

```
AN(config)#ssli on rhost1 0
AN(config)#ssl start rhost1
```

4.6 Integrated Mode: One L3 APV + Two L2 Firewalls

In this deployment mode, the APV appliance works in L3 mode, firewalls are set up in L2 mode, and one APV integrates the ingress node, the egress node and the Webagent service. The clients should set the APV appliance as the proxy server.

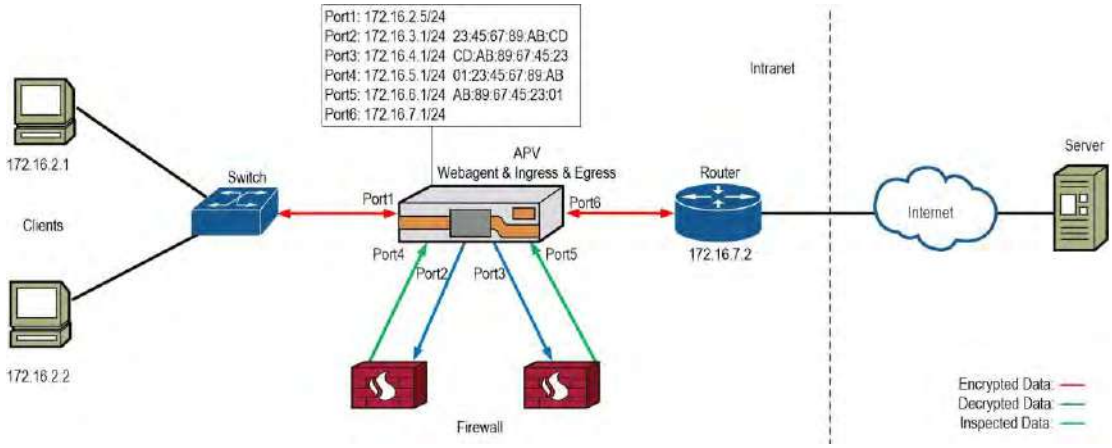

Figure 4–6 Integrated Mode: One L3 APV + Two L2 Firewalls

In this deployment mode, all configurations are performed on the APV appliance, which functions as the Webagent, ingress node and egress node at the same time.

4.6.1 Address and Route Settings

1. Set the IP addresses of Port1, Port2, Port3, Port4, Port5 and Port6.

```
AN(config)#ip address port1 172.16.2.5 24
AN(config)#ip address port2 172.16.3.1 24
AN(config)#ip address port3 172.16.4.1 24
AN(config)#ip address port4 172.16.5.1 24
AN(config)#ip address port5 172.16.6.1 24
AN(config)#ip address port6 172.16.7.1 24
```

2. Set the MAC address of Port2, Port3, Port4 and Port5.

```
AN(config)#interface mac port2 23:45:67:89:AB:CD
AN(config)#interface mac port3 CD:AB:89:67:45:23
AN(config)#interface mac port4 01:23:45:67:89:AB
AN(config)#interface mac port5 AB:89:67:45:23:01
```

3. Configure an MNET

```
AN(config)#mnet port2 mport2
AN(config)#mport2 110.16.10.1
```

4. Configure an IP pool.

```
AN(config)#ip pool p1 110.16.10.1
AN(config)#slb proxyip global p1
```

5. Configure the default route.

```
AN(config)#ip route default 172.16.7.2
```

4.6.2 Load Balance Settings

1. Set the system mode to reverse.

```
AN(config)#system mode reverse
```

2. Create FWDMAC real services and add them to two real service groups using the rr method. The group “g-ssl” is used to distribute SSL traffic to the security devices. The group “g-clear” is used to distribute non-SSL traffic to the security devices.

```
AN(config)#slb real fwdmac rs-ssl1 port2 01:23:45:67:89:AB 8443
AN(config)#slb real fwdmac rs-ssl2 port3 AB:89:67:45:23:01 8443
AN(config)#slb real fwdmac rs-clear1 port2 01:23:45:67:89:AB 8090
AN(config)#slb real fwdmac rs-clear2 port3 AB:89:67:45:23:01 8090
AN(config)#slb group method g-ssl rr
AN(config)#slb group method g-clear rr
AN(config)#slb group member g-ssl rs-ssl1
AN(config)#slb group member g-ssl rs-ssl2
AN(config)#slb group member g-clear rs-clear1
AN(config)#slb group member g-clear rs-clear2
```

3. Configure a TCPS and a TCP virtual service.

```
AN(config)#slb virtual tcps vs-ssl1 0.0.0.0 443 noarp
AN(config)#slb virtual tcp vs-clear1 0.0.0.0 80 noarp
```

4. Configure two default policies.

```
AN(config)#slb policy default vs-ssl1 g-ssl
AN(config)#slb policy default vs-clear1 g-clear
```

5. Create a TCP and a TCPS real service and configure them to keep the destination IP unchanged when forwarding packets.

```
AN(config)#slb real tcps rs-ssl3 172.16.7.2 443 icmp
AN(config)#slb real tcp rs-clear3 172.16.7.2 80 icmp
AN(config)#slb real settings keepdip rs-ssl3
AN(config)#slb real settings keepdip rs-clear3
```

6. Create two TCP virtual services and enable RTS for them.

```
AN(config)#slb virtual tcp vs-ssl2 0.0.0.0 8443 noarp
AN(config)#slb virtual tcp vs-clear2 0.0.0.0 8090 noarp
AN(config)#slb virtual settings rts vs-ssl2
AN(config)#slb virtual settings rts vs-clear2
```

7. Configure two static policies.

```
AN(config)#slb policy static vs-ssl2 rs-ssl3
AN(config)#slb policy static vs-clear2 rs-clear3
```

4.6.3 Webagent Service Settings

1. Configure the DNS server.

```
AN(config)#ip nameserver 10.8.80.10
```

2. Configure the Webagent service, and create two links to associate it with “vs-ssl1” and “vs-clear1”.

```
AN(config)#webagent service w1 172.16.2.5 8000
AN(config)#webagent link w1 vs-ssl1
AN(config)#webagent link w1 vs-clear1
```

4.6.4 SSL Interception Settings

1. Create an SSL virtual host and associate it with “vs-ssl”.

```
AN(config)#ssl host virtual vhost1 vs-ssl
```

2. Enable SSL interception for “vhost1”.

```
AN(config)#ssli on vhost1 1
```

3. Generate SSL interception certificates for “vhost1”, activate them and enable “vhost1”.

Note: The generated CA certificates must also be imported into the trusted CA list of clients’ browsers.

```
AN(config)#ssli cacert rsa vhost1 2048 1 1
AN(config)#ssli cacert ecc vhost1 prime256v1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

Note: Currently, elliptic curve secp521r1 is not widely supported by mainstream browsers. It is recommended to set prime256v1 or secp384r1 to avoid connection failures.

4. Create an SSL real host “rhost1” and associate it with “rs-ssl3”.

```
AN(config)#ssl host real rhost1 rs-ssl3
```

5. Enable SSL interception for “rhost1” and enable “rhost1”.

```
AN(config)#ssli on rhost1 1
AN(config)#ssl start rhost1
```

5 Dynamic Port Interception (DPI)

5.1 Introduction

With the Dynamic Port Interception (DPI) feature, the APV appliance can intercept all SSL traffic, no matter whether the SSL traffic is bound for the default 443 port of HTTPS or another port.

Currently, DPI is supported for SSL interception deployed in both the integrated mode and the distributed mode.

When the APV appliance works in L3 mode, implementing DPI requires the following changes to the original SSL interception configurations:

- On the ingress node, change the port number from “443” to “0” when defining the TCPS virtual service; and remove the L2IP virtual service’s port range settings, which are used for load balancing of non-SSL traffic received from the clients.
- On the egress node, change the port number from “443” to “0” when defining the TCPS real service; and remove the L2IP virtual service’s port range settings, which are used for load balancing of non-SSL traffic received from the real servers.

When the APV appliance works in L2 mode, implementing DPI requires the following changes to the original SSL interception configurations:

- On the ingress node, change the port number from “443” to “0” when defining the TCPS virtual service; remove the filter rule used to bypass returned SSL traffic and to acquire server certificates; remove the filter rule used to forward servers’ cleartext SSL traffic to the SSL interception module; use port number “0” for all bridge filter rules.
- On the egress node, change the port number from “443” to “0” when defining the TCPS real service; and remove the filter rule used to forward clients’ cleartext SSL traffic to the SSL interception module; use port number “0” for all bridge filter rules.

When a Webagent link is deployed in an SSL interception scenario, DPI is not supported.

5.2 Configuration Example

5.2.1 Integrated Mode: One L3 APV +Two L3 Firewalls

In this deployment mode, the APV appliance and the firewalls are set up in L3 mode. The APV appliance serves as both the ingress and egress nodes. The interface and route configurations on the firewalls are as shown in the following figure.

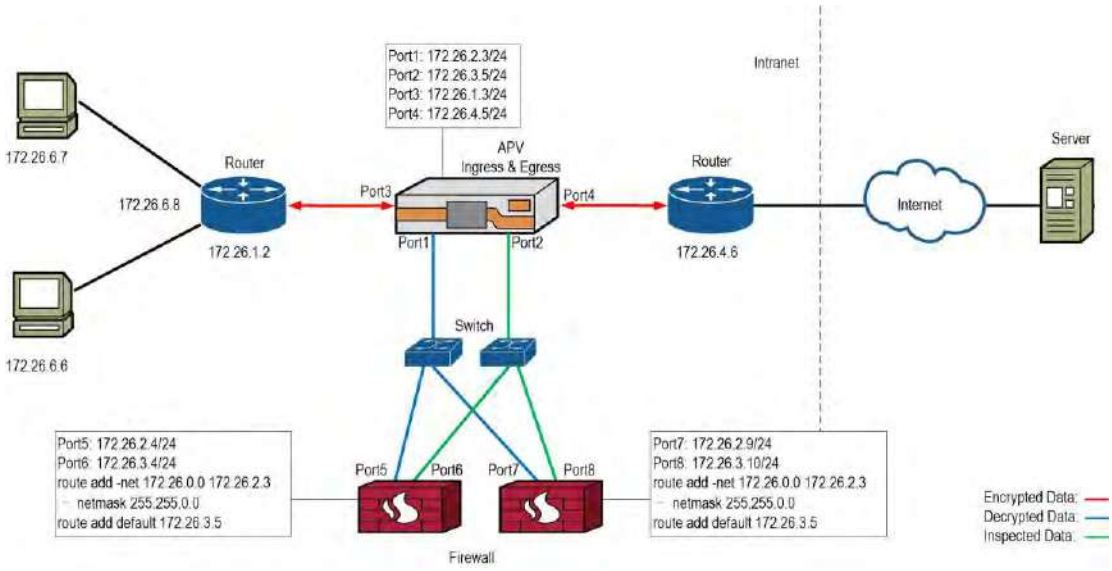

Figure 5–1 Integrated Mode: One L3 APV + Two L3 Firewalls

5.2.1.1 Address and Route Settings

1. Set the IP addresses of Port1, Port2, Port3 and Port4.

```
AN(config)#ip address port1 172.26.2.3 24
AN(config)#ip address port2 172.26.3.5 24
AN(config)#ip address port3 172.26.1.3 24
AN(config)#ip address port4 172.26.4.5 24
```

2. Set the default route.

```
AN(config)#ip route default 172.26.4.6
```

3. Define an Eroute.

```
AN(config)#ip eroute er1 1900 0.0.0.0 0.0.0.0 0 172.26.0.0 255.255.0.0 0 any 172.26.1.2
```

5.2.1.2 Load Balance Settings

- **Load Balancing of SSL Traffic Received from Clients**

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Create FWDIP real services.

```
AN(config)#slb real fwdip rs1 172.26.2.4 8443
AN(config)#slb real fwdip rs2 172.26.2.9 8443
```

3. Create a real service group using the hi method and add “rs1” and “rs2” to this group.

```
AN(config)#slb group method hi_group hi
AN(config)#slb group member hi_group rs1
```

```
AN(config)#slb group member hi_group rs2
```

4. Create a TCPS virtual service.

```
AN(config)#slb virtual tcps vs1 0.0.0.0 0 noarp 0
```

5. Configure a default policy to associate “vs1” with “hi_group”.

```
AN(config)#slb policy default vs1 hi_group
```

6. Configure health check for “rs1” and “rs2” to ensure that Port2 is accessible.

```
AN(config)#slb real health a1 rs1 172.26.3.5 56789 tcp 3 3
AN(config)#slb real health a2 rs2 172.26.3.5 56789 tcp 3 3
AN(config)#health ipreflect aa 172.26.3.5 56789 tcp
```

7. Configure health check for “rs1” and “rs2” to check the health status of security devices.

```
AN(config)#slb real health hc_os_h1 rs1 172.26.2.4 0 icmp 3 3
AN(config)#slb real health hc_os_h2 rs2 172.26.2.9 0 icmp 3 3
```

8. Set the relationship among health checks of “rs1” and “rs2” to “and”.

```
AN(config)#health relation rs1 and
AN(config)#health relation rs2 and
```

- **Load Balancing of Non-SSL Traffic Received from Clients**

1. Create L2Ip real services.

```
AN(config)#slb real l2ip rs4 172.26.2.4
AN(config)#slb real l2ip rs5 172.26.2.9
```

2. Create an L2 real service group using the chi method, set the route mode to “direct” and add “rs4” and “rs5” to this group.

```
AN(config)#slb group method chi_group1 chi direct
AN(config)#slb group member chi_group1 rs4
AN(config)#slb group member chi_group1 rs5
```

3. Create an L2IP virtual service.

```
AN(config)#slb virtual l2ip l2ip_vs1 172.26.1.3 172.26.1.2
```

4. Configure a default policy to associate “l2ip_vs1” with “chi_group1”.

```
AN(config)#slb policy default l2ip_vs1 chi_group1
```

5. Configure two port ranges for “chi_group1”.

```
AN(config)#slb group option portrange chi_group1 0 8442 all src
AN(config)#slb group option portrange chi_group1 8444 65535 all src
```

- **Forwarding of Inspected Traffic to the Real Server**

1. Create a TCP virtual service and enable RTS for it.

```
AN(config)#slb virtual tcp vs2 0.0.0.0 8443 noarp 0
AN(config)#slb virtual settings rts vs2
```

2. Create a TCPS real service and configure it to keep destination IP addresses unchanged when forwarding packets.

```
AN(config)#slb real tcps rs3 172.26.4.6 0 icmp
AN(config)#slb real settings keepdip rs3
```

3. Configure a static policy to associate “vs2” with “rs3”.

```
AN(config)#slb policy static vs2 rs3
```

➢ **Load Balancing of Non-SSL Traffic Returned from the Real Server**

1. Create L2IP real services.

```
AN(config)#slb real l2ip rs6 172.26.3.4
AN(config)#slb real l2ip rs7 172.26.3.10
```

2. Create an L2 real service group using the chi method, and add “rs6” and “rs7” to this group.

```
AN(config)#slb group method chi_group2 chi route
AN(config)#slb group member chi_group2 rs6
AN(config)#slb group member chi_group2 rs7
```

3. Create an L2IP virtual service.

```
AN(config)#slb virtual l2ip l2ip_vs2 172.26.4.5 172.26.4.6
```

4. Configure a default policy to associate “l2ip_vs2” with “chi_group2”.

```
AN(config)#slb policy default l2ip_vs2 chi_group2
```

5. Configure two port ranges for “chi_group2”.

```
AN(config)#slb group option portrange chi_group2 0 8442 all dst
AN(config)#slb group option portrange chi_group2 8444 65535 all dst
```

5.2.1.3 SSL Interception Settings

1. Create an SSL virtual host and associate it with “vs1”.

```
AN(config)#ssl host virtual vhost1 vs1
```

2. Enable SSL interception for “vhost1”.

```
AN(config)#ssli on vhost1 1
```

3. Generate SSL interception certificates for “vhost1”, activate them, and enable “vhost1”.

Note: The generated CA certificates must also be imported into the trusted CA list of clients' browsers.

```
AN(config)#ssli cacert rsa vhost1 2048 1 1
AN(config)#ssli cacert ecc vhost1 prime256v1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

Note: Currently, elliptic curve secp521r1 is not widely supported by mainstream browsers. It is recommended to set prime256v1 or secp384r1 to avoid connection failures.

4. Create an SSL real host "rhost1" and associate it with "rs3".

```
AN(config)#ssl host real rhost1 rs3
```

5. Enable SSL interception for "rhost1" and enable "rhost1".

```
AN(config)#ssli on rhost1 1
AN(config)#ssl start rhost1
```

5.2.2 Integrated Mode: One L3 APV +Two L2 Firewalls

In this deployment mode, the APV appliance is set up in L3 mode, and firewalls are set up in L2 mode. The APV appliance serves as both the ingress and egress nodes.

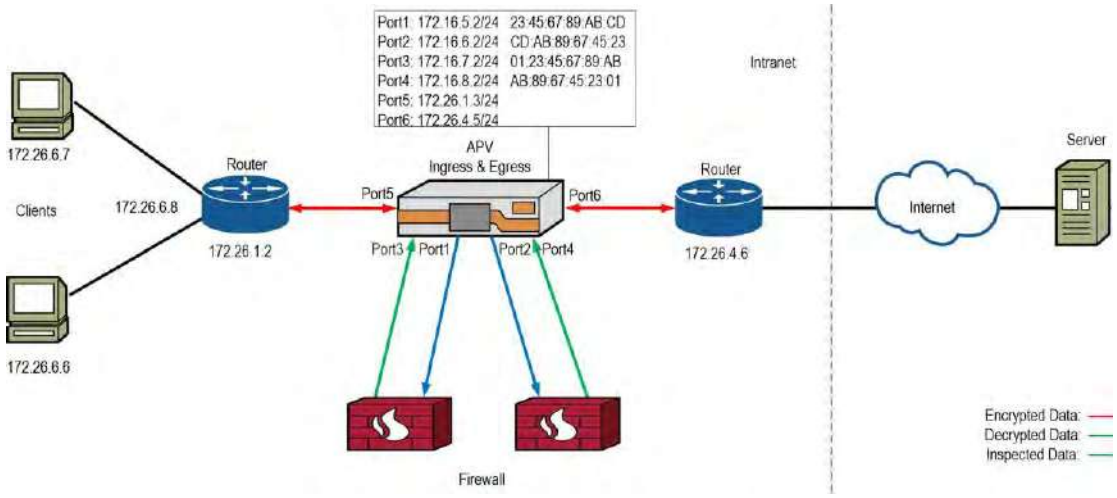

Figure 5–2 Integrated Mode: One L3 APV + Two L2 Firewalls

5.2.2.1 Address and Route Settings

1. Set the IP addresses of Port1, Port2, Port3, Port4, Port5 and Port6.

```
AN(config)#ip address port1 172.16.5.2 24
AN(config)#ip address port2 172.16.6.2 24
AN(config)#ip address port3 172.16.7.2 24
AN(config)#ip address port4 172.16.8.2 24
AN(config)#ip address port5 172.26.1.3 24
AN(config)#ip address port6 172.26.4.5 24
```

2. Set the default route.

```
AN(config)#ip route default 172.26.4.6
```

3. Define an Eroute.

```
AN(config)#ip eroute er1 1900 0.0.0.0 0.0.0.0 0 172.26.0.0 255.255.0.0 0 any 172.26.1.2
```

5.2.2.2 Load Balance Settings

➢ **Load Balancing of SSL Traffic Received from Clients**

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Create FWDMAC real services.

```
AN(config)#slb real fwdmac rs1 port1 01:23:45:67:89:AB 8443
AN(config)#slb real fwdmac rs2 port2 AB:89:67:45:23:01 8443
```

3. Create a real service group using the chi method and add “rs1” and “rs2” to this group.

```
AN(config)#slb group method chi_group chi
AN(config)#slb group member chi_group rs1
AN(config)#slb group member chi_group rs2
```

4. Create a TCPS virtual service.

```
AN(config)#slb virtual tcps vs1 0.0.0.0 0 noarp 0
```

5. Configure a default policy to associate “vs1” with “chi_group”.

```
AN(config)#slb policy default vs1 chi_group
```

6. Configure health checks for “rs1” and “rs2” to ensure that Port3 and Port4 are accessible.

```
AN(config)#slb real health a1 rs1 172.16.7.2 56789 tcp
AN(config)#slb real health a2 rs2 172.16.8.2 56789 tcp
AN(config)#health ipreflect aa 0.0.0.0 56789 tcp
```

➢ **Load Balancing of Non-SSL Traffic Received from Clients**

1. Create L2mac real services.

```
AN(config)#slb real l2mac rs4 01:23:45:67:89:AB port1
AN(config)#slb real l2mac rs5 AB:89:67:45:23:01 port2
```

2. Create an L2 real service group using the chi method, set the route mode to “direct” and add “rs4” and “rs5” to this group.

```
AN(config)#slb group method chi_group1 chi direct
AN(config)#slb group member chi_group1 rs4
AN(config)#slb group member chi_group1 rs5
```

3. Create an L2IP virtual service.

```
AN(config)#slb virtual l2ip l2ip_vs 172.26.1.3 172.26.1.2
```

4. Configure a default policy to associate “l2ip_vs1” with “chi_group1”.

```
AN(config)#slb policy default l2ip_vs1 chi_group1
```

5. Configure two port ranges for “chi_group1”.

```
AN(config)#slb group option portrange chi_group1 0 8442 all src
AN(config)#slb group option portrange chi_group1 8444 65535 all src
```

➢ **Forwarding of Inspected Traffic to the Real Server**

1. Create a TCPS real service, and configure it to keep destination IP addresses unchanged when forwarding packets.

```
AN(config)#slb real tcps rs3 172.26.4.6 0 icmp
AN(config)#slb real settings keepdip rs3
```

2. Create a TCP virtual service and enable RTS for it.

```
AN(config)#slb virtual tcp vs2 0.0.0.0 8443 noarp 0
AN(config)#slb virtual settings rts vs2
```

3. Configure a static policy to associate “vs2” with “rs3”.

```
AN(config)#slb policy static vs2 rs3
```

➢ **Load Balancing of Non-SSL Traffic Returned from the Real Server**

1. Create L2mac real services.

```
AN(config)#slb real l2mac rs6 23:45:67:89:AB:CD port3
AN(config)#slb real l2mac rs7 CD:AB:89:67:45:23 port4
```

2. Create an L2 real service group using the chi method, set the route mode to “route” and add “rs6” and “rs7” to this group.

```
AN(config)#slb group method chi_group2 chi route
AN(config)#slb group member chi_group2 rs6
AN(config)#slb group member chi_group2 rs7
```

3. Create an L2IP virtual service.

```
AN(config)#slb virtual l2ip l2ip_vs2 172.16.4.5
```

4. Configure a default policy to associate “l2ip_vs2” with “chi_group2”.

```
AN(config)#slb policy default l2ip_vs2 chi_group2
```

5. Configure two port ranges for “chi_group2”.

```
AN(config)#slb group option portrange chi_group2 0 8442 all dst
AN(config)#slb group option portrange chi_group2 8444 65535 all dst
```

5.2.2.3 SSL Interception Settings

1. Create an SSL virtual host and associate it with “vs1”.

```
AN(config)#ssl host virtual vhost1 vs1
```

2. Enable SSL interception for “vhost1”.

```
AN(config)#ssli on vhost1 1
```

3. Generate SSL interception certificates for “vhost1”, activate them and enable “vhost1”.

Note: The generated CA certificates must also be imported into the trusted CA list of clients’ browsers.

```
AN(config)#ssli cacert rsa vhost1 2048 1 1
AN(config)#ssli cacert ecc prime256v1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

Note: Currently, elliptic curve secp521r1 is not widely supported by mainstream browsers. It is recommended to set prime256v1 or secp384r1 to avoid connection failures.

4. Create an SSL real host “rhost1” and associate it with “rs3”.

```
AN(config)#ssl host real rhost1 rs3
```

5. Enable SSL interception for “rhost1” and enable “rhost1”.

```
AN(config)#ssli on rhost1 1
AN(config)#ssl start rhost1
```

5.2.3 Distributed Mode: Two L3 APVs + Two L3 Firewalls

In this deployment mode, the APV appliances and firewalls are set up in L3 mode and the two APV appliances serve as the ingress and egress nodes respectively.

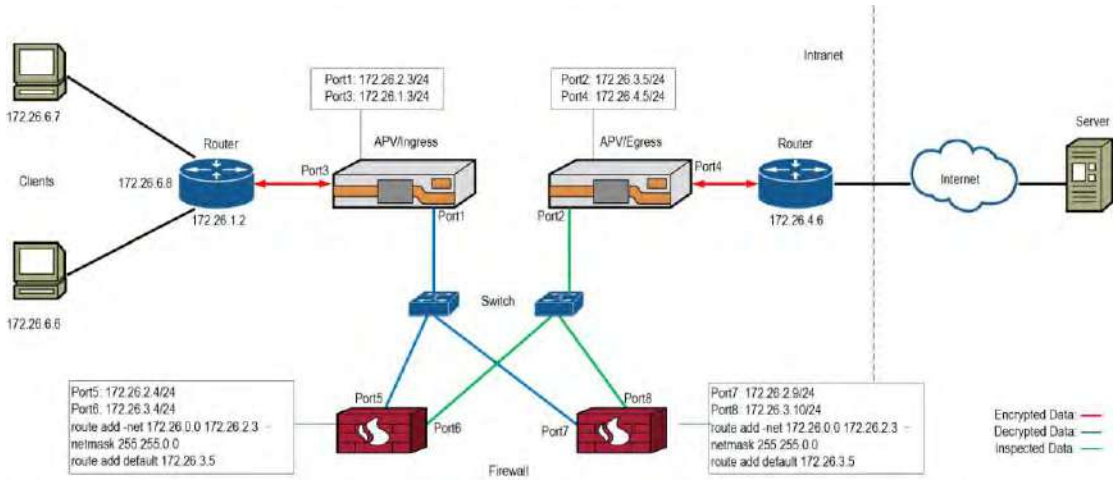

Figure 5–3 Distributed Mode: Two L3 APVs + Two L3 Firewalls

5.2.3.1 Configuring the Ingress Node

5.2.3.1.1 Address and Route Settings

1. Set the IP addresses of Port1 and Port3.

```
AN(config)#ip address port1 172.26.2.3 24
AN(config)#ip address port3 172.26.1.3 24
```

2. Set the default route.

```
AN(config)#ip route default 172.26.1.2
```

3. Define Eroutes.

```
AN(config)#ip eroute er1 1900 172.26.0.0 255.255.0.0 0 0.0.0.0 0.0.0.0 172.26.2.4
AN(config)#ip eroute er2 1900 172.26.0.0 255.255.0.0 0 0.0.0.0 0.0.0.0 172.26.2.9
```

4. Enable the IPflow function.

```
AN(config)#ip ipflow on
```

5.2.3.1.2 Load Balance Settings

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Create FWDIP real services.

```
AN(config)#slb real fwdip rs1 172.26.2.4 8443
AN(config)#slb real fwdip rs2 172.26.2.9 8443
```

3. Create a real service group using the chi method and add the FWDIP real services to this group.

```
AN(config)#slb group method chi_group chi
AN(config)#slb group member chi_group rs1
AN(config)#slb group member chi_group rs2
```

4. Create a TCPS virtual service.

```
AN(config)#slb virtual tcps vs1 0.0.0.0 0 noarp 0
```

5. Configure a default policy to associate “vs1” with “chi_group”.

```
AN(config)#slb policy default vs1 chi_group
```

6. Configure health checks for “rs1” and “rs2” to ensure that Port2 on the egress node is accessible (a health check reflector is needed on the egress node).

```
AN(config)#slb real health a1 rs1 172.26.3.5 56789 tcp 3 3
AN(config)#slb real health a2 rs2 172.26.3.5 56789 tcp 3 3
```

7. Configure health checks for “rs1” and “rs2” to check the health status of security devices.

```
AN(config)#slb real health hc_os_h1 rs1 172.26.2.4 0 icmp 3 3
AN(config)#slb real health hc_os_h2 rs2 172.26.2.9 0 icmp 3 3
```

8. Set the relationship among health checks of “rs1” and “rs2” to “and”.

```
AN(config)#health relation rs1 and
AN(config)#health relation rs2 and
```

5.2.3.1.3 SSL Interception Settings

1. Create an SSL virtual host and associate it with “vs1”.

```
AN(config)#ssl host virtual vhost1 vs1
```

2. Enable SSL interception for “vhost1”.

```
AN(config)#ssli on vhost1 0
```

3. Generate SSL interception certificates for “vhost1”, activate them, and enable “vhost1”.

Note: The generated CA certificates must also be imported into the trusted CA list of clients’ browsers.

```
AN(config)#ssli cacert rsa vhost1 2048 1 1
AN(config)#ssli cacert ecc vhost1 prime256v1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

Note: Currently, elliptic curve secp521r1 is not widely supported by mainstream browsers. It is recommended to set prime256v1 or secp384r1 to avoid connection failures.

5.2.3.2 Configuring the Egress Node

5.2.3.2.1 Address and Route Settings

1. Set the IP addresses of Port2 and Port4.

```
AN(config)#ip address port2 172.26.3.5 24
AN(config)#ip address port4 172.26.4.5 24
```

2. Set the default route.

```
AN(config)#ip route default 172.26.4.6
```

3. Enable RTS.

```
AN(config)#ip rts on
```

4. Define Eroutes.

```
AN(config)#ip eroute er1 1900 0.0.0.0 0.0.0.0 0 172.26.0.0 255.255.255.0 0 172.26.3.4
AN(config)#ip eroute er2 1900 0.0.0.0 0.0.0.0 0 172.26.0.0 255.255.255.0 0 172.26.3.10
```

5.2.3.2.2 Load Balance Settings

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Configure a TCPS real service, and configure it to keep destination IP addresses unchanged when forwarding packets.

```
AN(config)#slb real tcps rs3 172.26.4.6 0 icmp
AN(config)#slb real settings keepdip rs3
```

3. Configure a TCP virtual service and enable RTS for it.

```
AN(config)#slb virtual tcp vs2 0.0.0.0 8443 noarp 0
AN(config)#slb virtual settings rts vs2
```

4. Configure a static policy to associate “vs2” with “rs3”.

```
AN(config)#slb policy static vs2 rs3
```

5. Create a health check reflector “reflector1”.

```
AN(config)#health ipreflect reflector1 172.26.3.5 56789 tcp
```

5.2.3.2.3 SSL Interception Settings

1. Create an SSL real host “rhost1” and associate it with “rs3”.

```
AN(config)#ssl host real rhost1 rs3
```

2. Enable SSL interception for “rhost1” and enable “rhost1”.

```
AN(config)#ssli on rhost1 0
AN(config)#ssl start rhost1
```

5.2.4 Distributed Mode: Two L3 APVs + Two L2 Firewalls

In this deployment mode, the APV appliances are set up in L3 mode, firewalls are set up in L2 mode, and the two APV appliances serve as the ingress and egress nodes respectively.

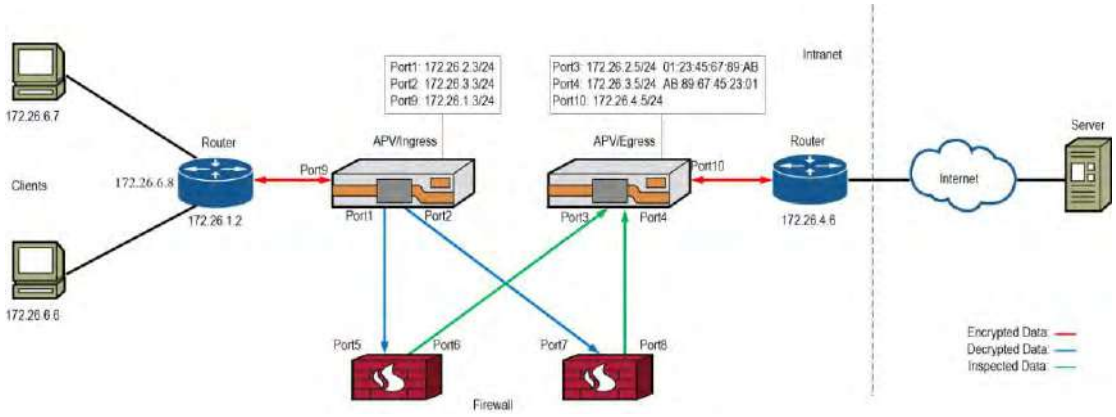

Figure 5–4 Distributed Mode: Two L3 APVs + Two L2 Firewalls

5.2.4.1 Configuring the Ingress Node

5.2.4.1.1 Address and Route Settings

1. Set the IP addresses of Port1, Port2, and Port9.

```
AN(config)#ip address port1 172.26.2.3 24
AN(config)#ip address port2 172.26.3.3 24
AN(config)#ip address port9 172.26.1.3 24
```

2. Set the default route.

```
AN(config)#ip route default 172.26.1.2
```

3. Enable the IPflow function.

```
AN(config)#ip ipflow on
```

4. Define Eroutes.

```
AN(config)#ip eroute er1 1900 172.26.0.0 255.255.0.0 0 0.0.0.0 0.0.0.0 0 172.26.2.5
AN(config)#ip eroute er2 1900 172.26.0.0 255.255.0.0 0 0.0.0.0 0.0.0.0 0 172.26.3.5
```

5.2.4.1.2 Load Balance Settings

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Create a TCPS virtual service.

```
AN(config)#slb virtual tcps vs1 0.0.0.0 0 noarp 0
```

3. Create FWDMAC real services.

```
AN(config)#slb real fwdmac port1 01:23:45:67:89:AB 8443
AN(config)#slb real fwdmac port2 AB:89:67:45:23:01 8443
```

4. Create an L2 real service group using the chi method and add “rs1” and “rs2” to this group.

```
AN(config)#slb group method chi_group chi
AN(config)#slb group member chi_group rs1
AN(config)#slb group member chi_group rs2
```

5. Configure a default policy to associate “vs1” with “chi_group”.

```
AN(config)#slb policy default vs1 chi_group
```

6. Configure health checks for “rs1” and “rs2” to ensure that Port3 and Port4 on the egress node are accessible (a health check reflector is needed on the egress node).

```
AN(config)#slb real health a1 rs1 172.26.2.5 56789 tcp 3 3
AN(config)#slb real health a2 rs2 172.26.3.5 56789 tcp 3 3
```

5.2.4.1.3 SSL Interception Settings

1. Create an SSL virtual host and associate it with “vs1”.

```
AN(config)#ssl host virtual vhost1 vs1
```

2. Enable SSL interception for “vhost1”.

```
AN(config)#ssli on vhost1 0
```

3. Generate SSL interception certificates for “vhost1”, activate them and enable “vhost1”.

Note: The generated CA certificates must also be imported into the trusted CA list of clients’ browsers.

```
AN(config)#ssli cacert rsa vhost1 2048 1 1
AN(config)#ssli cacert ecc vhost1 prime256v1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

Note: Currently, elliptic curve secp521r1 is not widely supported by mainstream browsers. It is recommended to set prime256v1 or secp384r1 to avoid connection failures.

5.2.4.2 Configuring the Egress Node

5.2.4.2.1 Address and Route Settings

1. Set the IP addresses of Port3, Port4 and Port10.

```
AN(config)#ip address port3 172.26.2.5 24
AN(config)#ip address port4 172.26.3.5 24
AN(config)#ip address port10 172.26.4.5 24
```

2. Set the default route

```
AN(config)#ip route default 172.26.4.6
```

3. Enable RTS.

```
AN(config)#ip rts on
```

4. Define Eroutes.

```
AN(config)#ip eroute er3 1900 0.0.0.0 0.0.0.0 0 172.26.0.0 255.255.0.0 0 any 172.26.2.3
AN(config)#ip eroute er4 1900 0.0.0.0 0.0.0.0 0 172.26.0.0 255.255.0.0 0 any 172.26.3.3
```

5.2.4.2.2 Load Balance Settings

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Configure a TCPS real service, and configure it to keep destination IP addresses unchanged when forwarding packets.

```
AN(config)#slb real tcps rs3 172.26.4.6 0 icmp
AN(config)#slb real settings keepdip rs3
```

3. Configure a TCP virtual service and enable RTS for it.

```
AN(config)#slb virtual tcp vs2 0.0.0.0 8443 noarp 0
AN(config)#slb virtual settings rts vs2
```

4. Configure a static policy to associate “vs2” with “rs3”.

```
AN(config)#slb policy static vs2 rs3
```

5. Create a health check reflector “reflector1”.

```
AN(config)#health ipreflect reflector1 0.0.0.0 56789 tcp
```

5.2.4.2.3 SSL Interception Settings

1. Create an SSL real host “rhost1” and associate it with “rs3”.

```
AN(config)#ssl host real rhost1 rs3
```

2. Enable SSL interception for “rhost1” and enable “rhost1”.

```
AN(config)#ssli on rhost1 0
AN(config)#ssl start rhost1
```

5.2.5 Integrated Mode: One L2 APV + One L2 Firewall (With VLAN)

In this deployment mode, both the APV appliance and the firewall are set up in L2 mode, and the APV appliance serves as both the ingress and egress nodes. Two bridge instances are configured on the APV appliance:

- One bridge is used to transfer SSL traffic to the SSL interception module for decryption and then forward the decrypted traffic to the firewall.
- The other bridge is used to receive the inspected traffic from the firewall, re-encrypt the traffic and then forward it to the server.
- Port1, Port2, Port3 and Port4 can receive and send packets carrying VLAN tags 1000 and 1001.

The network topology and interface configurations are as shown in the following figure.

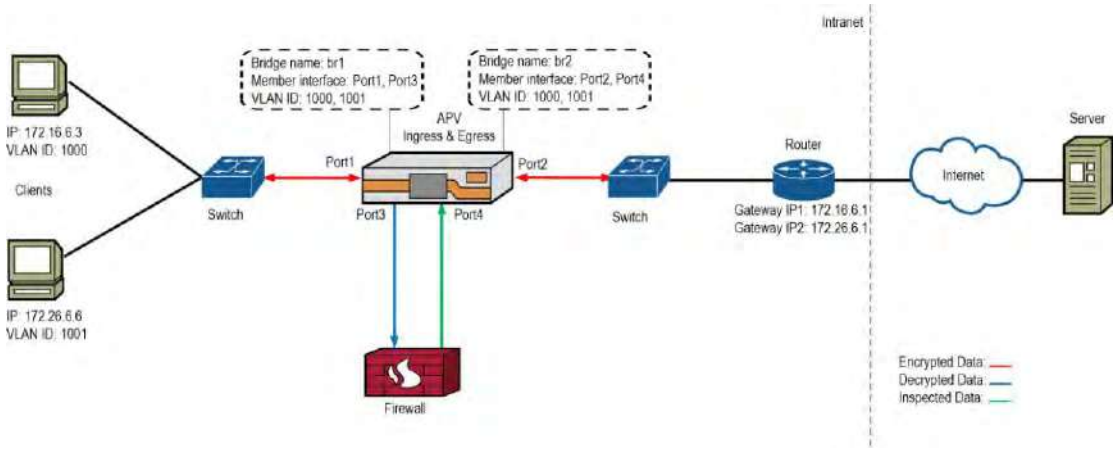

Figure 5–5 Integrated Mode: One L2 APV + One L2 Firewall

5.2.5.1 Bridge Settings

1. Create two bridge instances.

```
AN(config)#bridge name br1
AN(config)#bridge name br2
```

2. Add members to the created bridge instances.

```
AN(config)#bridge member br1 port1 yes
AN(config)#bridge member br1 port3 yes
AN(config)#bridge member br2 port2 yes
AN(config)#bridge member br2 port4 yes
```

3. Set VLAN tags for the bridge member interfaces.

If packets passing through a member interface carry VLAN tags, you need to set the corresponding VLAN IDs in order for the interface to receive and send tagged packets.

```
AN(config)#bridge vlan br1 port1 1000
AN(config)#bridge vlan br1 port3 1000
AN(config)#bridge vlan br2 port2 1000
AN(config)#bridge vlan br2 port4 1000
AN(config)#bridge vlan br1 port1 1001
AN(config)#bridge vlan br1 port3 1001
AN(config)#bridge vlan br2 port2 1001
AN(config)#bridge vlan br2 port4 1001
```

4. Create filter rules to forward all traffic to the SSL interception module.

```
AN(config)#bridge apprule br1 0.0.0.0 0 0.0.0.0 0 tcp
AN(config)#bridge apprule br2 0.0.0.0 0 0.0.0.0 0 tcp
```

5.2.5.2 SSL Settings

5.2.5.2.1 Forwarding of Received SSL Traffic to the Security Device

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Create a FWDMAC real service.

   Note that “AB:89:67:45:23:01” does not represent any port. It can be replaced with an arbitrary MAC address, but it must be set.

```
AN(config)#slb real fwdmac rs1 port3 AB:89:67:45:23:01 8443
```

3. Create a real service group using the chi method and add “rs1” to this group.

```
AN(config)#slb group method chi_group chi
AN(config)#slb group member chi_group rs1
```

4. Create a TCPS virtual service.

```
AN(config)#slb virtual tcps vs1 0.0.0.0 0 noarp 0
```

5. Configure a default policy to associate “vs1” with “chi_group”.

```
AN(config)#slb policy default vs1 chi_group
```

6. Create an SSL virtual host and associate it with “vs1”.

```
AN(config)#ssl host virtual vhost1 vs1
```

5.2.5.2.2 Forwarding of Inspected SSL Traffic to the Real Service

1. Create a TCPS real service, and configure it to keep destination IP addresses unchanged when forwarding packets.

   Note that “172.26.6.1” can be replaced with an arbitrary IP address.

```
AN(config)#slb real tcps rs2 172.26.6.1 0 none
AN(config)#slb real settings keepdip rs2
```

2. Create a TCP virtual service.

```
AN(config)#slb virtual tcp vs2 0.0.0.0 8443 noarp 0
```

3. Configure a static policy to associate “vs2” with “rs2”.

```
AN(config)#slb policy static vs2 rs2
```

4. Disable the real service health check.

```
AN(config)#health off
```

5. Create an SSL real host “rhost1” and associate it with “rs2”.

```
AN(config)#ssl host real rhost1 rs2
```

5.2.5.3 SSL Interception Settings

1. Enable SSL interception for “vhost1”.

```
AN(config)#ssli on vhost1 1
```

2. Generate SSL interception certificates for “vhost1”, activate them and enable “vhost1”.

Note: The generated CA certificates must also be imported into the trusted CA list of clients’ browsers.

```
AN(config)#ssli cacert rsa vhost1 2048 1 1
AN(config)#ssli cacert ecc vhost1 prime256v1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

Note: Currently, elliptic curve secp521r1 is not widely supported by mainstream browsers. It is recommended to set prime256v1 or secp384r1 to avoid connection failures.

3. Enable SSL interception for “rhost1” and enable “rhost1”.

```
AN(config)#ssli on rhost1 1
```

```
AN(config)#ssl start rhost1
```

Note: If the SSL server uses a testing certificate or a self-signed certificate, both the ingress and egress nodes must have its certificate chain (including the root CA certificate) imported. For testing-only purposes, administrators can choose not to import the certificate chain, but must execute the "**ssl globals verifycert off**" command to disable the server authentication function.

5.2.6 Integrated Mode: One L2 APV + One L2 Firewall (Without VLAN)

In this deployment mode, both the APV appliance and the firewall are set up in L2 mode, and the APV appliance serves as both the ingress and egress nodes. Two bridge instances are configured on the APV appliance:

- One bridge is used to transfer SSL traffic to the SSL interception module for decryption and then forward the decrypted traffic to the firewall.
- The other bridge is used to receive the inspected traffic from the firewall, re-encrypt the traffic and then send it to the server.

The network topology and interface configurations are as shown in the following figure.

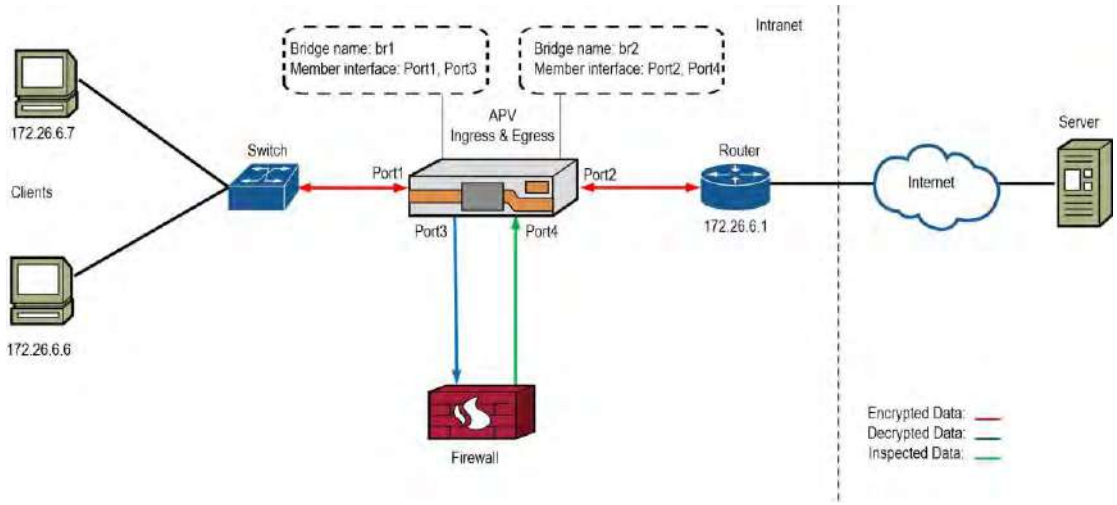

Figure 5–6 Integrated Mode: One L2 APV + One L2 Firewall

5.2.6.1 Bridge Settings

1. Create two bridge instances.

```
AN(config)#bridge name br1
AN(config)#bridge name br2
```

2. Add members to the created bridge instances.

```
AN(config)#bridge member br1 port1 yes
AN(config)#bridge member br1 port3 yes
AN(config)#bridge member br2 port2 yes
AN(config)#bridge member br2 port4 yes
```

3. Create filter rules to forward all traffic to the SSL interception module.

```
AN(config)#bridge apprule br1 0.0.0.0 0 0.0.0.0 0 tcp
AN(config)#bridge apprule br2 0.0.0.0 0 0.0.0.0 0 tcp
```

5.2.6.2 SSL Settings

5.2.6.2.1 Forwarding of Received SSL Traffic to the Security Device

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Create a FWDMAC real service.

   Note that “AB:89:67:45:23:01” does not represent any port. It can be replaced with an arbitrary MAC address, but it must be set.

```
AN(config)#slb real fwdmac rs1 port3 AB:89:67:45:23:01 8443
```

3. Create a real service group using the chi method and add “rs1” to this group.

```
AN(config)#slb group method chi_group chi
AN(config)#slb group member chi_group rs1
```

4. Create a TCPS virtual service.

```
AN(config)#slb virtual tcps vs1 0.0.0.0 0 noarp 0
```

5. Configure a default policy to associate “vs1” with “chi_group”.

```
AN(config)#slb policy default vs1 chi_group
```

6. Create an SSL virtual host and associate it with “vs1”.

```
AN(config)#ssl host virtual vhost1 vs1
```

5.2.6.2.2 Forwarding of Inspected SSL Traffic to the Real Service

1. Create a TCPS real service, and configure it to keep destination IP addresses unchanged when forwarding packets.

   Note that “172.26.6.1” can be replaced with an arbitrary IP address.

```
AN(config)#slb real tcps rs2 172.26.6.1 0 none
AN(config)#slb real settings keepdip rs2
```

2. Create a TCP virtual service.

```
AN(config)#slb virtual tcp vs2 0.0.0.0 8443 noarp 0
```

3. Configure a static policy to associate “vs2” with “rs2”.

```
AN(config)#slb policy static vs2 rs2
```

4. Disable real service health check.

```
AN(config)#health off
```

5. Create an SSL real host “rhost1” and associate it with “rs2”.

```
AN(config)#ssl host real rhost1 rs2
```

5.2.6.3 SSL Interception Settings

1. Enable SSL interception for “vhost1”.

```
AN(config)#ssli on vhost1 1
```

2. Generate SSL interception certificates for “vhost1”, activate them and enable “vhost1”.

Note: The generated CA certificates must also be imported into the trusted CA list of clients’ browsers.

```
AN(config)#ssli cacert rsa vhost1 2048 1 1
AN(config)#ssli cacert ecc vhost1 prime256v1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

Note: Currently, elliptic curve secp521r1 is not widely supported by mainstream browsers. It is recommended to set prime256v1 or secp384r1 to avoid connection failures.

3. Enable SSL interception for “rhost1” and enable “rhost1”.

```
AN(config)#ssli on rhost1 1
AN(config)#ssl start rhost1
```

Note: If the SSL server uses a testing certificate or a self-signed certificate, both the ingress and egress nodes must have its certificate chain (including the root CA certificate) imported. For testing-only purposes, administrators can choose not to import the certificate chain, but must execute the “**ssl globals verifycert off**” command to disable the server authentication function.

5.2.7 Distributed Mode: Two L2 APVs + One L2 Firewall (With VLAN)

In this deployment mode, both the APV appliances and the firewall work in L2 mode, and the two APV appliances play the role of the ingress and egress nodes respectively. The ingress node and the egress node each have a bridge instance configured:

- The bridge on the ingress node is used to transfer SSL traffic to the SSL interception module for decryption and then forward the decrypted traffic to the firewall.
- The bridge on the egress node is used to receive the inspected traffic from the firewall, re-encrypt the traffic and then send it to the server.
- Port1, Port2, Port3 and Port4 can receive and send packets carrying VLAN tags 1000 and 1001.

The network topology and interface are as shown in the following figure.

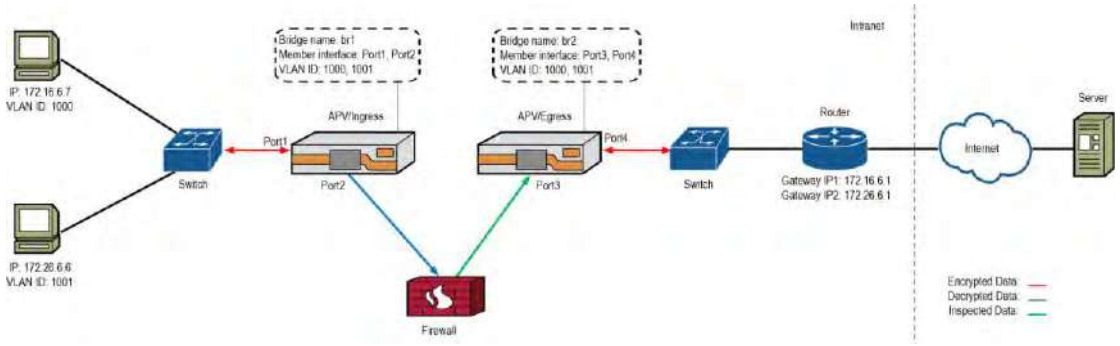

Figure 5–7 Distributed Mode: Two L2 APVs + One L2 Firewall

5.2.7.1 Configuring the Ingress Node

5.2.7.1.1 Bridge Settings

1. Create a bridge instance.

```
AN(config)#bridge name br1
```

2. Add members to the created bridge instance.

```
AN(config)#bridge member br1 port1 yes
AN(config)#bridge member br1 port2 yes
```

3. Set VLAN tags for the bridge member interfaces.

If packets passing through a member interface carry VLAN tags, you need to set the corresponding VLAN IDs in order for the interface to receive and send tagged packets.

```
AN(config)#bridge vlan br1 port1 1000
AN(config)#bridge vlan br1 port1 1001
AN(config)#bridge vlan br1 port2 1000
AN(config)#bridge vlan br1 port2 1001
```

4. Create filter rules to forward all traffic from the server to the SSL interception module.

```
AN(config)#bridge apprule br1 0.0.0.0 0 0.0.0.0 0 tcp
```

5.2.7.1.2 SSL Settings

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Create a TCPS virtual service.

```
AN(config)#slb virtual tcps vs1 0.0.0.0 0 noarp 0
```

3. Create a FWDMAC real service.

   Note that “AB:89:67:45:23:01” does not represent any port. It can be replaced with an arbitrary MAC address, but it must be set.

```
AN(config)#slb real fwdmac rs1 port2 AB:89:67:45:23:01 8443
```

4. Create an L2 real service group using the chi method and add “rs1” to this group.

```
AN(config)#slb group method chi_group chi
AN(config)#slb group member chi_group rs1
```

5. Configure a default policy to associate “vs1” with “chi_group”.

```
AN(config)#slb policy default vs1 chi_group
```

6. Create an SSL virtual host and associate it with “vs1”.

```
AN(config)#ssl host virtual vhost1 vs1
```

5.2.7.1.3 SSL Interception Settings

1. Enable SSL interception for “vhost1”.

```
AN(config)#ssli on vhost1 0
```

2. Generate SSL interception certificates for “vhost1”, activate them and enable “vhost1”.

```
AN(config)#ssli cacert rsa vhost1 2048 1 1
AN(config)#ssli cacert ecc vhost1 prime256v1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

Note:

1. The generated CA certificates must also be imported into the trusted CA list of clients’ browsers.
2. If the SSL server uses a testing certificate or a self-signed certificate, both the ingress and egress nodes must have its certificate chain (including the root CA certificate) imported. For testing-only purposes, administrators can choose not to import the certificate chain, but must execute the “**ssl globals verifycert off**” command to disable the server authentication function.
3. Currently, elliptic curve secp521r1 is not widely supported by mainstream browsers. It is recommended to set prime256v1 or secp384r1 to avoid connection failures.

5.2.7.2 Configuring the Egress Node

5.2.7.2.1 Bridge Settings

1. Create a bridge instance.

```
AN(config)#bridge name br2
```

2. Add members to the created bridge instance.

```
AN(config)#bridge member br2 port3 yes
AN(config)#bridge member br2 port4 yes
```

3. Set VLAN tags for the bridge member interfaces.

If packets passing through a member interface carry VLAN tags, you need to set the corresponding VLAN IDs for it to allow the interface to receive and send tagged packets.

```
AN(config)#bridge vlan br2 port3 1000
AN(config)#bridge vlan br2 port3 1001
AN(config)#bridge vlan br2 port4 1000
AN(config)#bridge vlan br2 port4 1001
```

4. Create filter rules to forward traffic from the client to the SSL interception module.

```
AN(config)#bridge apprule br2 0.0.0.0 0 0.0.0.0 0 tcp
```

5.2.7.2.2 SSL Settings

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Configure a TCPS real service, and configure it to keep destination IP addresses unchanged when forwarding packets.

   Note that “172.26.6.1” can be replaced with an arbitrary IP address.

```
AN(config)#slb real tcps rs2 172.26.6.1 0 none
AN(config)#slb real settings keepdip rs2
```

3. Configure a TCP virtual service.

```
AN(config)#slb virtual tcp vs2 0.0.0.0 8443 noarp 0
```

4. Configure a static policy to associate “vs2” with “rs2”.

```
AN(config)#slb policy static vs2 rs2
```

5. Disable the real service health check.

```
AN(config)#health off
```

6. Create an SSL real host “rhost1” and associate it with “rs2”.

```
AN(config)#ssl host real rhost1 rs2
```

5.2.7.2.3 SSL Interception Settings

1. Enable SSL interception for “rhost1” and enable “rhost1”.

```
AN(config)#ssli on rhost1 0
AN(config)#ssl start rhost
```

Note: If the SSL server uses a testing certificate or a self-signed certificate, both the ingress and egress nodes must have its certificate chain (including the root CA certificate) imported. For testing-only purposes, administrators can choose not to import the certificate chain, but must execute the “**ssl globals verifycert off**” command to disable the server authentication function.

5.2.8 Distributed Mode: Two L2 APVs + One L2 Firewall (Without VLAN)

In this deployment mode, both the APV appliances and the firewall are set up in L2 mode, and the two APV appliances play the role of the ingress and egress nodes respectively. The ingress node and the egress node each have a bridge instance configured:

- The bridge on the ingress node is used to transfer SSL traffic to the SSL interception module for decryption and forward the decrypted traffic to the firewall.
- The bridge on the egress node is used to receive the inspected traffic from the firewall, re-encrypt the traffic and then send it to the server.

The network topology and interface are as shown in the following figure.

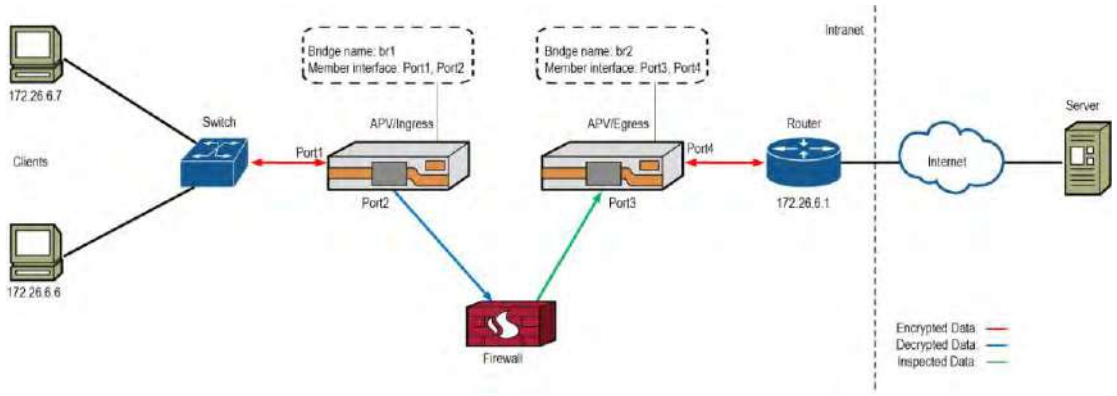

Figure 5–8 Distributed Mode: Two L2 APVs + One L2 Firewall (Without VLAN)

5.2.8.1 Configuring the Ingress Node

5.2.8.1.1 Bridge Settings

1. Create a bridge instance.

```
AN(config)#bridge name br1
```

2. Add members to the created bridge instance.

```
AN(config)#bridge member br1 port1 yes
AN(config)#bridge member br1 port2 yes
```

3. Create filter rules to forward all traffic from the server to the SSL interception module.

```
AN(config)#bridge apprule br1 0.0.0.0 0 0.0.0.0 0 tcp
```

5.2.8.1.2 SSL Settings

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Create a TCPS virtual service.

```
AN(config)#slb virtual tcps vs1 0.0.0.0 0 noarp 0
```

3. Create a FWDMAC real service.

   Note that “AB:89:67:45:23:01” does not represent any port. It can be replaced with an arbitrary MAC address, but it must be set.

```
AN(config)#slb real fwdmac rs1 port2 AB:89:67:45:23:01 8443
```

4. Create an L2 real service group using the chi method and add “rs1” to this group.

```
AN(config)#slb group method chi_group chi
AN(config)#slb group member chi_group rs1
```

5. Configure a default policy to associate “vs1” with “chi_group”.

```
AN(config)#slb policy default vs1 chi_group
```

6. Create an SSL virtual host and associate it with “vs1”.

```
AN(config)#ssl host virtual vhost1 vs1
```

5.2.8.1.3 SSL Interception Settings

1. Enable SSL interception for “vhost1”.

```
AN(config)#ssli on vhost1 0
```

2. Generate SSL interception certificates for “vhost1”, activate them and enable “vhost1”.

```
AN(config)#ssli cacert rsa vhost1 2048 1 1
AN(config)#ssli cacert ecc vhost1 prime256v1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

Note:

1. The generated CA certificates must also be imported into the trusted CA list of clients’ browsers.
2. If the SSL server uses a testing certificate or a self-signed certificate, both the ingress and egress nodes must have its certificate chain (including the root CA certificate) imported. For testing-only purposes, administrators can choose not to

import the certificate chain, but must execute the "**ssl globals verifycert off**" command to disable the server authentication function.

3. Currently, elliptic curve secp521r1 is not widely supported by mainstream browsers. It is recommended to set prime256v1 or secp384r1 to avoid connection failures.

5.2.8.2 Configuring the Egress Node

5.2.8.2.1 Bridge Settings

1. Create a bridge instance.

```
AN(config)#bridge name br2
```

2. Add members to the created bridge instance.

```
AN(config)#bridge member br2 port3 yes
AN(config)#bridge member br2 port4 yes
```

3. Create filter rules to forward all traffic from the client to the SSL interception module.

```
AN(config)#bridge apprule br2 0.0.0.0 0 0.0.0.0 0 tcp
```

5.2.8.2.2 SSL Settings

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Configure a TCPS real service, and configure it to keep destination IP addresses unchanged when forwarding packets.

   Note that "172.26.6.1" can be replaced with an arbitrary IP address.

```
AN(config)#slb real tcps rs2 172.26.6.1 0 none
AN(config)#slb real settings keepdip rs2
```

3. Configure a TCP virtual service.

```
AN(config)#slb virtual tcp vs2 0.0.0.0 8443 noarp 0
```

4. Configure a static policy to associate "vs2" with "rs2".

```
AN(config)#slb policy static vs2 rs2
```

5. Disable the real service health check.

```
AN(config)#health off
```

6. Create an SSL real host "rhost1" and associate it with "rs2".

```
AN(config)#ssl host real rhost1 rs2
```

5.2.8.2.3 SSL Interception Settings

1. Enable SSL interception for “rhost1” and enable “rhost1”.

```
AN(config)#ssli on rhost1 0
AN(config)#ssl start rhost
```

Note: If the SSL server uses a testing certificate or a self-signed certificate, both the ingress and egress nodes must have its certificate chain (including the root CA certificate) imported. For testing-only purposes, administrators can choose not to import the certificate chain, but must execute the “**ssl globals verifycert off**” command to disable the server authentication function.

6 WebRoot Website Classification

Website classification is a dynamic website category recognition function that the APV appliance provides through a subscription to the Webroot BrightCloud URL Classification service. This function allows the APV appliance to look up the category of a website via the local cache, local database and online connection to the Webroot server. Administrators can configure the APV's processing modes for traffic accessing specific website categories.

The SSL interception module supports manual configurations of interception SNI lists or bypass SNI lists to define SSL traffic to be intercepted or bypassed. This method applies if clients access only a few types of websites and the manual configuration workload is light. If the application scenario accommodates accesses to a wide variety of types of websites, it is recommended to purchase and configure the website classification function to achieve intelligent URL filtering on the SSL interception module. Via Webroot website classification, the APV appliance supports recognition of 82 website categories. For details about these categories, please refer to: http://www.brightcloud.com/tools/change-request-url-categorization.php.

The website classification function should be configured on the ingress node (in distributed deployment mode). To enable online lookup ("**webclassify cloud on**"), the ingress node must be able to access the external network.

6.1 Filtering Policy

Website classification supports two types of policies for filtering website categories:

- Interception policy: all traffic accessing websites belonging to the configured category(s) will be decrypted and then sent to security devices for inspection. Traffic accessing website categories that are not defined by the policy will be forwarded transparently.
- Bypass policy: all traffic accessing websites belonging to the configured category(s) will be transparently forwarded without being decrypted and inspected by security devices. Traffic accessing website categories that are not defined by the policy will be intercepted and then sent to security devices for inspection.

On the same virtual host, interception policies and bypass policies cannot be configured simultaneously, that is, each virtual host can be configured with only one type of filtering policy. When a virtual host receives a request or response:

- The virtual host will preferentially match the SNI lists configured by the SNI interception function.
  - If a matching entry is found, it will process the request or response based on the control type (intercept or bypass) of the SNI list.
  - If no matching entry is found in the SNI lists, it will search for the website category using the website classification function.

- When trying to determine the website category using the website classification function, the virtual host will first search in the local cache and database.
  - If the local cache or database has category information for the website, the virtual host will process the request or response based on the control type (intercept or bypass) of the filtering policy.
  - If no category information is available in the local cache and database, the system will connect to the Webroot server to query the website category online and save the acquired category information to the local cache.
  - If the Webroot server cannot recognize the website category, the system will intercept the request or response by default.

6.2 License

To use the website classification function for intelligent URL filtering, please contact Array Networks Customer Support and provide the device model and serial number to obtain a license. This function supports two types of licenses:

- Free trial license: 30-day free trial
- Formal license: 365-day validity period

After the license expires, the website classification function will be unavailable. The Webroot server will deny the APV appliance's access and the APV appliance will stop sending website category lookup queries to the server, and the local cache and database are also unavailable for use.

6.3 Configuration Example

All website classification configurations should be performed on the SSL interception virtual host on the ingress node. In this example, the APV appliance will be configured to bypass traffic bound for healthcare and financial services sites to protect user privacy. With the exception of these website categories, traffic accessing all other websites will be intercepted. This configuration example is based on the network topology as shown in the following figure. In other deployment scenarios, you can repeat these steps to complete website classification configurations for SSL interception.

- Both the APV appliances and the firewalls are set up in L3 mode.
- Two APV appliances play the role of the ingress and egress nodes respectively.
- The interface and route configurations on the firewalls are as shown in the following figure.

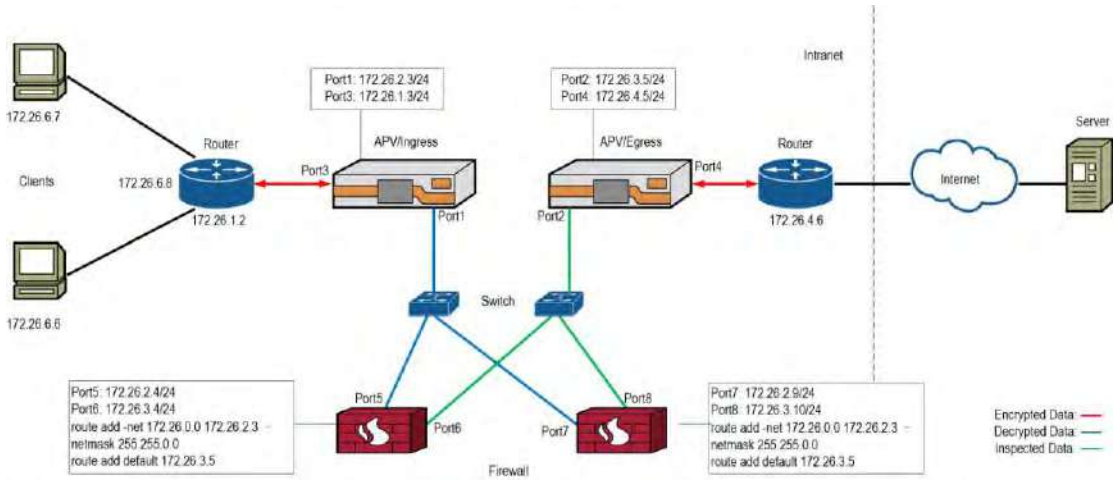

Figure 6–1 Distributed Mode: Two L3 APVs + Two L3 Firewalls

6.3.1 Configuring the Ingress Node

6.3.1.1 Address and Route Settings

1. Set the IP addresses of Port1 and Port3.

```
AN(config)#ip address port1 172.26.2.3 24
AN(config)#ip address port3 172.26.1.3 24
```

2. Set the default route.

```
AN(config)#ip route default 172.26.1.2
```

3. Define Eroutes.

```
AN(config)#ip eroute er1 1900 172.26.0.0 255.255.0.0 0 0.0.0.0 0.0.0.0 172.26.2.4
AN(config)#ip eroute er2 1900 172.26.0.0 255.255.0.0 0 0.0.0.0 0.0.0.0 172.26.2.9
```

4. Enable the IPflow function.

```
AN(config)#ip ipflow on
```

6.3.1.2 Load Balance Settings

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Create FWDIP real services.

```
AN(config)#slb real fwdip rs1 172.26.2.4 8443
AN(config)#slb real fwdip rs2 172.26.2.9 8443
```

3. Create a real service group using the chi method and add FWDIP real services to this group.

```
AN(config)#slb group method chi_group chi
AN(config)#slb group member chi_group rs1
```

```
AN(config)#slb group member chi_group rs2
```

4. Create a TCPS virtual service.

```
AN(config)#slb virtual tcps vs1 0.0.0.0 443 noarp 0
```

5. Configure a default policy to associate “vs1” with “chi_group”.

```
AN(config)#slb policy default vs1 chi_group
```

6. Configure health checks for “rs1” and “rs2” to ensure that Port2 on the egress node is accessible (a health check reflector is needed on the egress node).

```
AN(config)#slb real health a1 rs1 172.26.3.5 56789 tcp 3 3
AN(config)#slb real health a2 rs2 172.26.3.5 56789 tcp 3 3
```

7. Configure health check for “rs1” and “rs2” to check the health status of security devices.

```
AN(config)#slb real health hc_os_h1 rs1 172.26.2.4 0 icmp 3 3
AN(config)#slb real health hc_os_h2 rs2 172.26.2.9 0 icmp 3 3
```

8. Set the relationship among health checks of “rs1” and “rs2” to “and”.

```
AN(config)#health relation rs1 and
AN(config)#health relation rs2 and
```

6.3.1.3 Website Classification Settings

1. Import the website classification function license (license number given is for example purposes only).

```
AN(config)#webclassify license 41ce070c-4baa4482-02bc5237-f62e21b8-b50b177c-00000000-00000001-20171130-20180129
```

2. Enable the global website classification function.

```
AN(config)#webclassify on
```

3. Enable the online website classification lookup function.

```
AN(config)#webclassify cloud on
```

4. Configure the default action for traffic accessing unrecognized websites on the SSL interception module.

```
AN(config)#ssli webclassify defaction vhost 0
```

5. Define healthcare and financial service sites as bypass URL categories. Category names must match the Webroot class(es) exactly. Use the link given previously to ensure the name(s) are exact.

```
AN(config)#ssli web url bypass vhost "Health & Medicine"
AN(config)#ssli web url bypass vhost "Financial Services"
```

6. Enable the URL classification function for the SSL interception virtual host.

```
AN(config)#ssli webclassify on vhost
```

6.3.1.4 SSL Interception Settings

1. Create an SSL virtual host and associate it with “vs1”.

```
AN(config)#ssl host virtual vhost1 vs1
```

2. Enable SSL interception for “vhost1”.

```
AN(config)#ssli on vhost1 0
```

3. Generate SSL interception certificates for “vhost1”, activate them, and enable “vhost1”.

Note: The generated CA certificates must also be imported into the trusted CA list of clients’ browsers.

```
AN(config)#ssli cacert rsa vhost1 2048 1 1
AN(config)#ssli cacert ecc vhost1 prime256v1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

Note: Currently, elliptic curve secp521r1 is not widely supported by mainstream browsers. It is recommended to set prime256v1 or secp384r1 to avoid connection failures.

6.3.2 Configuring the Egress Node

6.3.2.1 Address and Route Settings

1. Set the IP addresses of Port2 and Port4.

```
AN(config)#ip address port2 172.26.3.5 24
AN(config)#ip address port4 172.26.4.5 24
```

2. Set the default route.

```
AN(config)#ip route default 172.26.4.6
```

3. Enable RTS.

```
AN(config)#ip rts on
```

4. Define Eroutes.

```
AN(config)#ip eroute er1 1900 0.0.0.0 0.0.0.0 0 172.26.0.0 255.255.255.0 0 172.26.3.4
AN(config)#ip eroute er2 1900 0.0.0.0 0.0.0.0 0 172.26.0.0 255.255.255.0 0 172.26.3.10
```

6.3.2.2 Load Balance Settings

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Configure a TCPS real service, and configure it to keep destination IP addresses unchanged when forwarding packets.

```
AN(config)#slb real tcps rs3 172.26.4.6 443 icmp
AN(config)#slb real settings keepdip rs3
```

3. Configure a TCP virtual service and enable RTS for it.

```
AN(config)#slb virtual tcp vs2 0.0.0.0 8443 noarp 0
AN(config)#slb virtual settings rts vs2
```

4. Configure a static policy to associate “vs2” with “rs3”.

```
AN(config)#slb policy static vs2 rs3
```

5. Create a health check reflector “reflector1”.

```
AN(config)#health ipreflect reflector1 172.26.3.5 56789 tcp
```

6.3.2.3 SSL Interception Settings

1. Create an SSL real host “rhost1” and associate it with “rs3”.

```
AN(config)#ssl host real rhost1 rs3
```

2. Enable SSL interception for “rhost1” and enable “rhost1”.

```
AN(config)#ssli on rhost1 0
AN(config)#ssl start rhost1
```

7 SPAN Port

Beginning with ArrayOS APV 8.6.1.40, the system supports SPAN Port to capture packets from a source port to a destination port and then to a security device such as a dedicated IDS or a sniffer device. This is often used for the purpose of troubleshooting, debugging and traffic analysis.

7.1 Introduction

The SPAN Port feature employs filter lists to filter out traffic to be captured. With the filter lists, the system can be configured to capture traffic with specific source IPs, source ports, destination IPs and destination ports. In addition, it allows definition of packets to be captured that are flowing in the inbound direction, the outbound direction or both (bidirectional). It also allows self-definition of the transport protocol (TCP, UDP or both) of traffic to be captured. For traffic that does not match any filter list, the system will forward them via the original routing interface.

Using the SPAN Port feature, the system (either one or two APV appliances) can send captured packets to multiple security devices, either of the same type or of different types. If the security devices are of the same type, the system can be configured to send the captured packets to the security devices in a load balancing manner based on the hash value of the packets' source IP and destination IP addresses. Packets with the same source IP and destination IP addresses will be persistently sent to the same security device. Otherwise, the system can send a copy of the captured packets to each of the security device, that is, every security device will get the same packets. If the security devices are of different types, each of them will get the same packets too.

7.2 Configuration Example

7.2.1 Integrated Mode: One L3 APV + Four L2 Firewalls (Hybrid)

In this deployment mode, the APV appliance works in L3 mode and firewalls work in L2 mode. Two firewalls are inline deployed, and another two firewalls are bypass deployed. The APV appliance serves as both the ingress and egress nodes. The network topology and interface configurations are as shown in the following figure.

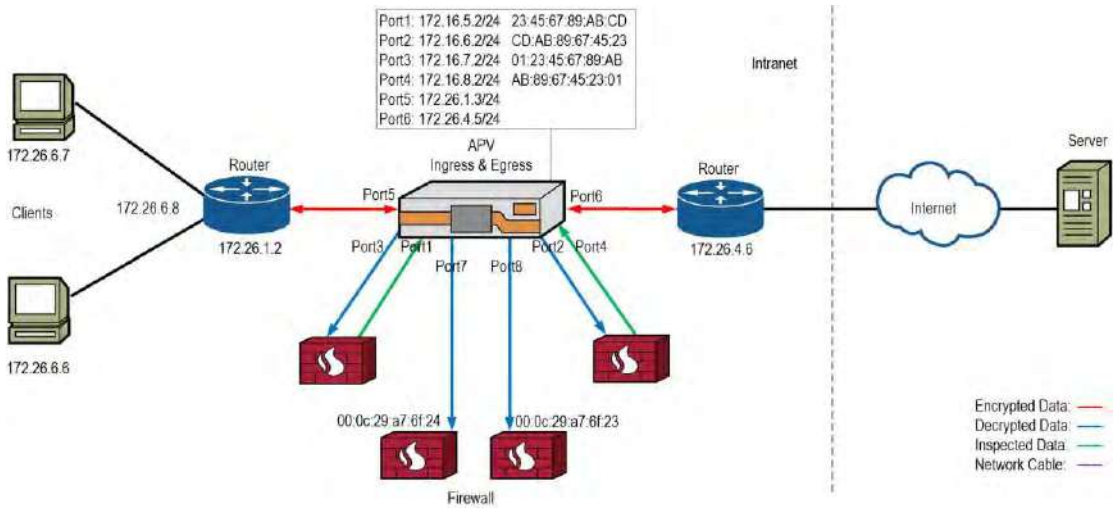

Figure 7–1 Integrated Mode: One L3 APV + Four L2 Firewalls (Hybrid)

7.2.1.1 Address and Route Settings

1. Set the IP addresses of Port1, Port2, Port3, Port4, Port5 and Port6.

```
AN(config)#ip address port1 172.16.5.2 24
AN(config)#ip address port2 172.16.6.2 24
AN(config)#ip address port3 172.16.7.2 24
AN(config)#ip address port4 172.16.8.2 24
AN(config)#ip address port5 172.26.1.3 24
AN(config)#ip address port6 172.26.4.5 24
```

2. Set the default route.

```
AN(config)#ip route default 172.26.4.6
```

3. Define an Eroute.

```
AN(config)#ip eroute er1 1900 0.0.0.0 0.0.0.0 0 172.26.0.0 255.255.0.0 0 any 172.26.1.2
```

7.2.1.2 Load Balance Settings

➢ **Load Balancing of SSL Traffic Received from Clients**

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Create FWDMAC real services.

```
AN(config)#slb real fwdmac rs1 port3 23:45:67:89:AB:CD 8443
AN(config)#slb real fwdmac rs2 port2 AB:89:67:45:23:01 8443
```

3. Create a real service group using the chi method and add “rs1” and “rs2” to this group.

```
AN(config)#slb group method chi_group chi
```

```
AN(config)#slb group member chi_group rs1
AN(config)#slb group member chi_group rs2
```

4. Create a TCPS virtual service.

```
AN(config)#slb virtual tcps vs1 0.0.0.0 443 noarp 0
```

5. Configure a default policy to associate “vs1” with “chi_group”.

```
AN(config)#slb policy default vs1 chi_group
```

6. Configure health checks for “rs1” and “rs2” to ensure that Port3 and Port4 are accessible.

```
AN(config)#slb real health a1 rs1 172.16.7.2 56789 tcp
AN(config)#slb real health a2 rs2 172.16.8.2 56789 tcp
AN(config)#health ipreflect aa 0.0.0.0 56789 tcp
```

➢ **Load Balancing of Non-SSL Traffic Received from Clients**

1. Create L2mac real services.

```
AN(config)#slb real l2mac rs4 23:45:67:89:AB:CD port3
AN(config)#slb real l2mac rs5 AB:89:67:45:23:01 port2
```

2. Create an L2 real service group using the chi method, set the route mode to “direct” and add “rs4” and “rs5” to this group.

```
AN(config)#slb group method chi_group1 chi direct
AN(config)#slb group member chi_group1 rs4
AN(config)#slb group member chi_group1 rs5
```

3. Create an L2IP virtual service.

```
AN(config)#slb virtual l2ip l2ip_vs 172.26.1.3 172.26.1.2
```

4. Configure a default policy to associate “l2ip_vs1” with “chi_group1”.

```
AN(config)#slb policy default l2ip_vs1 chi_group1
```

5. Configure two port ranges for “l2ip_vs1”.

```
AN(config)#slb virtual portrange l2ip_vs1 0 442 all dst
AN(config)#slb virtual portrange l2ip_vs1 444 65535 all dst
```

6. Configure two port ranges for “chi_group1”.

```
AN(config)#slb group option portrange chi_group1 0 8442 all src
AN(config)#slb group option portrange chi_group1 8444 65535 all src
```

➢ **Forwarding of Inspected Traffic to the Real Server**

1. Create a TCPS real service, and configure it to keep destination IP addresses unchanged when forwarding packets.

```
AN(config)#slb real tcps rs3 172.26.4.6 443 icmp
AN(config)#slb real settings keepdip rs3
```

2. Create a TCP virtual service and enable RTS for it.

```
AN(config)#slb virtual tcp vs2 0.0.0.0 8443 noarp 0
AN(config)#slb virtual settings rts vs2
```

3. Configure a static policy to associate “vs2” with “rs3”.

```
AN(config)#slb policy static vs2 rs3
```

➢ **Load Balancing of Non-SSL Traffic Returned from the Real Server**

1. Create L2mac real services.

```
AN(config)#slb real l2mac rs6 01:23:45:67:89:AB port1
AN(config)#slb real l2mac rs7 CD:AB:89:67:45:23 port4
```

2. Create an L2 real service group using the chi method, set the route mode to “route” and add “rs6” and “rs7” to this group.

```
AN(config)#slb group method chi_group2 chi route
AN(config)#slb group member chi_group2 rs6
AN(config)#slb group member chi_group2 rs7
```

3. Create an L2IP virtual service.

```
AN(config)#slb virtual l2ip l2ip_vs2 172.16.4.5
```

4. Configure a default policy to associate “l2ip_vs2” with “chi_group2”.

```
AN(config)#slb policy default l2ip_vs2 chi_group2
```

5. Configure two port ranges for “l2ip_vs2”.

```
AN(config)#slb virtual portrange l2ip_vs2 0 442 all src
AN(config)#slb virtual portrange l2ip_vs2 444 65535 all src
```

6. Configure two port ranges for “chi_group2”.

```
AN(config)#slb group option portrange chi_group2 0 8442 all dst
AN(config)#slb group option portrange chi_group2 8444 65535 all dst
```

7.2.1.3 SSL Interception Settings

1. Create an SSL virtual host and associate it with “vs1”.

```
AN(config)#ssl host virtual vhost1 vs1
```

2. Enable SSL interception for “vhost1”.

```
AN(config)#ssli on vhost1 1
```

3. Generate SSL interception certificates for “vhost1”, activate them and enable “vhost1”.

Note: The generated CA certificates must also be imported into the trusted CA list of clients’ browsers.

```
AN(config)#ssli cacert rsa vhost1 2048 1 1
AN(config)#ssli cacert ecc prime256v1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

Note: Currently, elliptic curve secp521r1 is not widely supported by mainstream browsers. It is recommended to set prime256v1 or secp384r1 to avoid connection failures.

4. Create an SSL real host “rhost1” and associate it with “rs3”.

```
AN(config)#ssl host real rhost1 rs3
```

5. Enable SSL interception for “rhost1” and enable “rhost1”.

```
AN(config)#ssli on rhost1 1
AN(config)#ssl start rhost1
```

7.2.1.4 SPAN Port Settings

1. Configure a filter list.

```
AN(config)#spanport filterlist name f1
```

2. Add filter rules to define the SSL traffic (including ciphertext and cleartext) to be captured and then sent to the security devices.

```
AN(config)#spanport filterlist member f1 port3 0.0.0.0 8443 0.0.0.0 0 tcp inbound
AN(config)#spanport filterlist member f1 port3 0.0.0.0 0 0.0.0.0 8443 tcp
outboundAN(config)#spanport filterlist member f1 port2 0.0.0.0 8443 0.0.0.0 0 tcp inbound
AN(config)#spanport filterlist member f1 port2 0.0.0.0 0 0.0.0.0 8443 tcp outbound
```

3. Configure a security device group.

```
AN(config)#spanport devicegroup name d1 lb
```

4. Add the security devices as members to the group.

```
AN(config)#spanport devicegroup member d1 port7 00:0c:29:a7:6f:24 abc
AN(config)#spanport devicegroup member d1 port8 00:0c:29:a7:6f:23 abd
```

5. Configure a SPAN port policy to associate the specified filter list with the specified security device group

```
AN(config)#spanport policy p1 f1 d1
```

7.2.2 Integrated Mode: One L3 APV + Two L2 Firewalls

In this deployment mode, the APV appliance works in L3 mode, firewalls work in L2 mode and the APV appliance serves as both the ingress and egress nodes. The firewalls are deployed in bypass mode. The interface and route configurations on the firewalls are as shown in the following figure. Compared with section 7.2.1, the orginal two firewalls that were deployed in inline mode are removed, so decrypted SSL traffic will be transparently transferred inside the APV appliance.

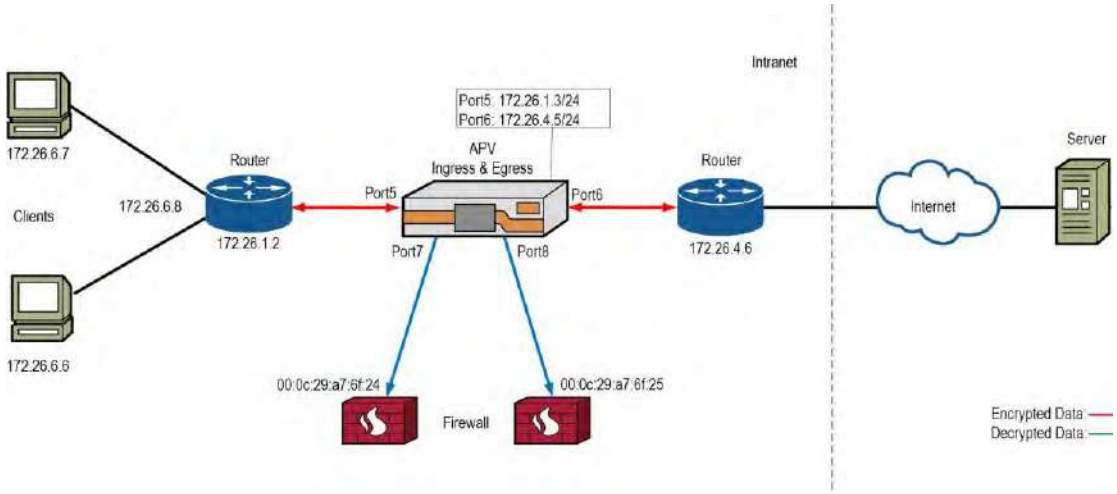

Figure 7–2 Integrated Mode: One L3 APV + Two L2 Firewalls

7.2.2.1 Address and Route Settings

1. Set the IP addresses of Port5 and Port6.

```
AN(config)#ip address port5 172.26.1.3 24
AN(config)#ip address port6 172.26.4.5 24
```

2. Set the default route.

```
AN(config)#ip route default 172.26.4.6
```

3. Define an Eroute.

```
AN(config)#ip eroute er1 2001 172.26.4.6 255.255.255.255 8443 0.0.0.0 0.0.0.0 0 tcp 172.26.1.3
```

7.2.2.2 Load Balance Settings

- **Forwarding of SSL Traffic Received from Clients**

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Create a FWDIP real service.

```
AN(config)#slb real fwdip rs1 172.26.1.3 8443
```

3. Create a real service group using the chi method and add “rs1” to this group.

```
AN(config)#slb group method chi_group chi
AN(config)#slb group member chi_group rs1
```

4. Create a TCPS virtual service.

```
AN(config)#slb virtual tcps vs1 0.0.0.0 443 noarp 0
```

5. Configure a default policy to associate “vs1” with “chi_group”.

```
AN(config)#slb policy default vs1 chi_group
```

➢ **Forwarding of Decrypted Traffic to the Real Server**

1. Create a TCPS real service, and configure it to keep destination IP addresses unchanged when forwarding packets.

```
AN(config)#slb real tcps rs3 172.26.4.6 443 icmp
AN(config)#slb real settings keepdip rs3
```

2. Create a TCP virtual service.

```
AN(config)#slb virtual tcp vs2 0.0.0.0 8443 noarp 0
AN(config)#slb virtual settings rts vs2
```

3. Configure a static policy to associate “vs2” with “rs3”.

```
AN(config)#slb policy static vs2 rs3
```

7.2.2.3 SSL Interception Settings

1. Create an SSL virtual host and associate it with “vs1”.

```
AN(config)#ssl host virtual vhost1 vs1
```

2. Enable SSL interception for “vhost1”.

```
AN(config)#ssli on vhost1 1
```

3. Generate SSL interception certificates for “vhost1”, activate them and enable “vhost1”.

Note: The generated CA certificates must also be imported into the trusted CA list of clients’ browsers.

```
AN(config)#ssli cacert rsa vhost1 2048 1 1
AN(config)#ssli cacert ecc prime256v1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

Note: Currently, elliptic curve secp521r1 is not widely supported by mainstream browsers. It is recommended to set prime256v1 or secp384r1 to avoid connection failures.

4. Create an SSL real host “rhost1” and associate it with “rs3”.

```
AN(config)#ssl host real rhost1 rs3
```

5. Enable SSL interception for “rhost1” and enable “rhost1”.

```
AN(config)#ssli on rhost1 1
AN(config)#ssl start rhost1
```

7.2.2.4 SPAN Port Settings

1. Configure a filter list.

```
AN(config)#spanport filterlist name f1
```

2. Add filter rules to define the SSL traffic (including ciphertext and cleartext) to be captured and then sent to the security devices.

```
AN(config)#spanport filterlist member f1 loopback 0.0.0.0 8443 0.0.0.0 0 tcp inbound
AN(config)#spanport filterlist member f1 loopback 0.0.0.0 0 0.0.0.0 8443 tcp outbound
```

3. Configure a security device group.

```
AN(config)#spanport devicegroup name d1 lb
```

4. Add the security devices as members to the group.

```
AN(config)#spanport devicegroup member d1 port7 00:0c:29:a7:6f:24 abc
AN(config)#spanport devicegroup member d1 port8 00:0c:29:a7:6f:25 abd
```

5. Configure a SPAN port policy to associate the specified filter list with the specified security device group

```
AN(config)#spanport policy p1 f1 d1
```

7.2.3 Distributed Mode: Two L3 APVs + Two L2 Firewalls

In this deployment mode, the APV appliances work in L3 mode, firewalls work in L2 mode and the two APV appliances play the role of the ingress node and egress node respectively. The firewalls are deployed in bypass mode. A network cable should be deployed to connect the two APV appliances. The interface and route configurations on the firewalls are as shown in the following figure.

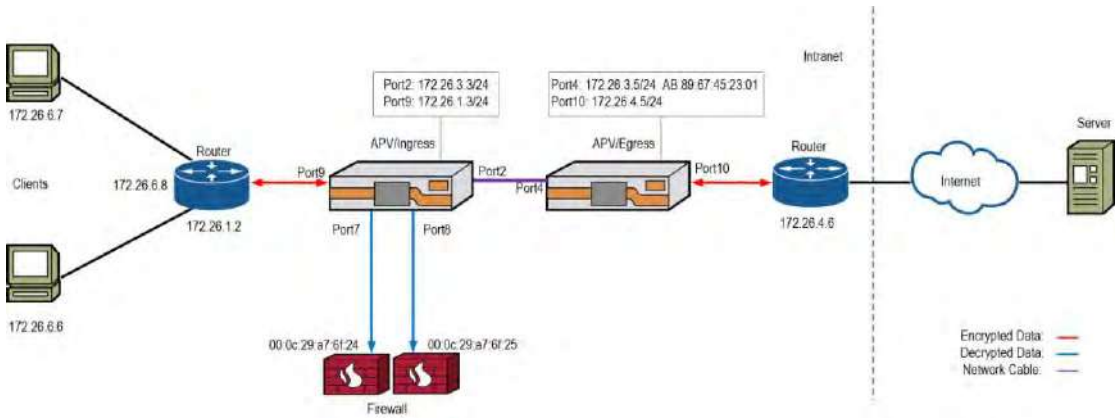

Figure 7–3 Distributed Mode: Two L3 APVs + Two L2 Firewalls

7.2.3.1 Configuring the Ingress Node

7.2.3.1.1 Address and Route Settings

1. Set the IP addresses of Port2, and Port9.

```
AN(config)#ip address port2 172.26.3.3 24
AN(config)#ip address port9 172.26.1.3 24
```

2. Set the default route.

```
AN(config)#ip route default 172.26.1.2
```

3. Enable the IPflow function.

```
AN(config)#ip ipflow on
```

4. Define Eroutes.

```
AN(config)#ip eroute er1 1900 172.26.0.0 255.255.0.0 0 0.0.0.0 0.0.0.0 0 172.26.2.5
AN(config)#ip eroute er2 1900 172.26.0.0 255.255.0.0 0 0.0.0.0 0.0.0.0 0 172.26.3.5
```

7.2.3.1.2 Load Balance Settings

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Create a TCPS virtual service.

```
AN(config)#slb virtual tcps vs1 0.0.0.0 443 noarp 0
```

3. Create FWDMAC real services.

```
AN(config)#slb real fwdmac rs2 port2 AB:89:67:45:23:01 8443
```

4. Create an L2 real service group using the chi method and add “rs2” to this group.

```
AN(config)#slb group method chi_group chi
AN(config)#slb group member chi_group rs2
```

5. Configure a default policy to associate “vs1” with “chi_group”.

```
AN(config)#slb policy default vs1 chi_group
```

6. Configure health check for “rs2” to ensure that Port4 on the egress node is accessible (a health check reflector is needed on the egress node).

```
AN(config)#slb real health a2 rs2 172.26.3.5 56789 tcp 3 3
```

7.2.3.1.3 SSL Interception Settings

1. Create an SSL virtual host and associate it with “vs1”.

```
AN(config)#ssl host virtual vhost1 vs1
```

2. Enable SSL interception for “vhost1”.

```
AN(config)#ssli on vhost1 0
```

3. Generate SSL interception certificates for “vhost1”, activate them and enable “vhost1”.

Note: The generated CA certificates must also be imported into the trusted CA list of clients’ browsers.

```
AN(config)#ssli cacert rsa vhost1 2048 1 1
AN(config)#ssli cacert ecc vhost1 prime256v1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

Note: Currently, elliptic curve secp521r1 is not widely supported by mainstream browsers. It is recommended to set prime256v1 or secp384r1 to avoid connection failures.

7.2.3.1.4 SPAN Port Settings

1. Configure a filter list.

```
AN(config)#spanport filterlist name f1
```

2. Add filter rules to define the SSL traffic (including ciphertext and cleartext) to be captured and then sent to the security devices.

```
AN(config)#spanport filterlist member f1 port2 0.0.0.0 8443 0.0.0.0 0 tcp inbound
AN(config)#spanport filterlist member f1 port2 0.0.0.0 0 0.0.0.0 8443 tcp outbound
```

3. Configure a security device group.

```
AN(config)#spanport devicegroup name d1 lb
```

4. Add the security device ports as members to the group.

```
AN(config)#spanport devicegroup member d1 port7 00:0c:29:a7:6f:24 abc
AN(config)#spanport devicegroup member d1 port8 00:0c:29:a7:6f:25 abd
```

5. Configure a SPAN port policy to associate the specified filter list with the specified security device group

```
AN(config)#spanport policy p1 f1 d1
```

7.2.3.2 Configuring the Egress Node

7.2.3.2.1 Address and Route Settings

1. Set the IP addresses of Port4 and Port10.

```
AN(config)#ip address port4 172.26.3.5 24
AN(config)#ip address port10 172.26.4.5 24
```

2. Set the default route

```
AN(config)#ip route default 172.26.4.6
```

3. Enable RTS.

```
AN(config)#ip rts on
```

4. Define Eroutes.

```
AN(config)#ip eroute er3 1900 0.0.0.0 0.0.0.0 0 172.26.0.0 255.255.0.0 0 any 172.26.2.3
AN(config)#ip eroute er4 1900 0.0.0.0 0.0.0.0 0 172.26.0.0 255.255.0.0 0 any 172.26.3.3
```

7.2.3.2.2 Load Balance Settings

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Configure a TCPS real service, and configure it to keep destination IP addresses unchanged when forwarding packets.

```
AN(config)#slb real tcps rs3 172.26.4.6 443 icmp
AN(config)#slb real settings keepdip rs3
```

3. Configure a TCP virtual service and enable RTS for it.

```
AN(config)#slb virtual tcp vs2 0.0.0.0 8443 noarp 0
AN(config)#slb virtual settings rts vs2
```

4. Configure a static policy to associate “vs2” with “rs3”.

```
AN(config)#slb policy static vs2 rs3
```

5. Create a health check reflector “reflector1”.

```
AN(config)#health ipreflect reflector1 0.0.0.0 56789 tcp
```

7.2.3.2.3 SSL Interception Settings

1. Create an SSL real host “rhost1” and associate it with “rs3”.

```
AN(config)#ssl host real rhost1 rs3
```

2. Enable SSL interception for “rhost1” and enable “rhost1”.

```
AN(config)#ssli on rhost1 0

AN(config)#ssl start rhost1
```

7.2.4 Integrated Mode: One L2 APV + Two L2 Firewalls

In this deployment mode, the APV appliance and the firewalls work in L2 mode, and the APV appliance serves as both the ingress and egress nodes. A network cable should be deployed to connect Port3 to Port4. Two bridge instances are configured on the APV appliance:

- One bridge is used to transfer SSL traffic to the SSL interception module for decryption and then forward the decrypted traffic to the firewall.
- The other bridge is used to receive the inspected traffic from the firewall, then re-encrypt the traffic before sending it on to the server.

The network topology and interface configurations are as shown in the following figure.

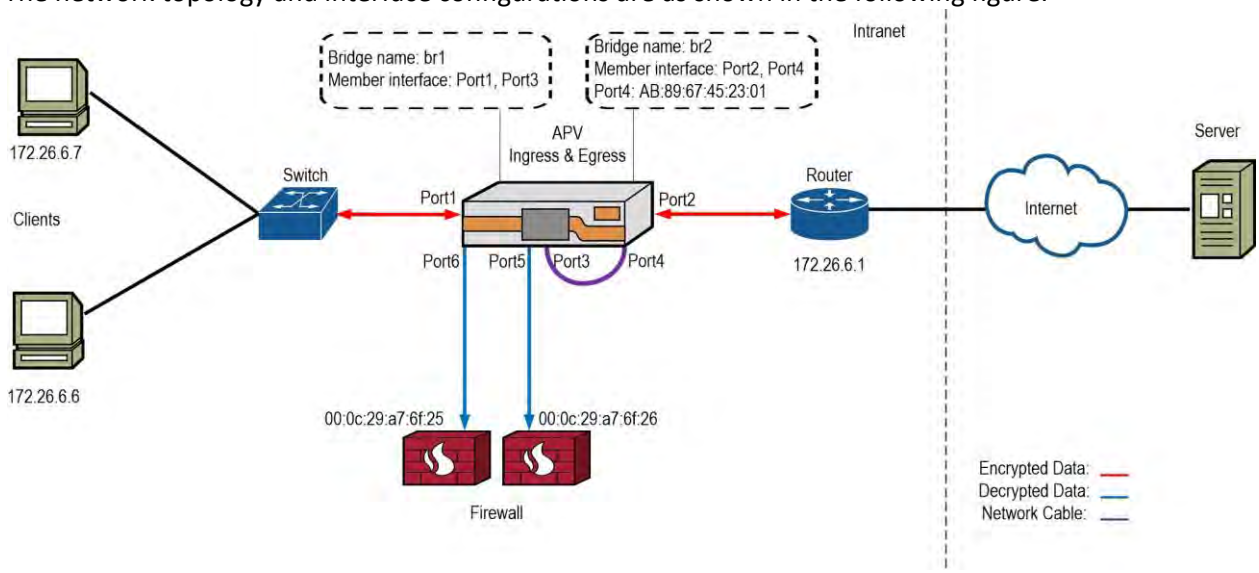

Figure 7–4 Integrated Mode: One L2 APV + Two L2 Firewalls

7.2.4.1 Bridge Settings

1. Create two bridge instances.

```
AN(config)#bridge name br1
AN(config)#bridge name br2
```

2. Add members to the created bridge instances.

```
AN(config)#bridge member br1 port1 yes
AN(config)#bridge member br1 port3 yes
AN(config)#bridge member br2 port2 yes
AN(config)#bridge member br2 port4 yes
```

3. Create filter rules to bypass returned SSL traffic and to acquire server certificates.

```
AN(config)#bridge apprule br1 0.0.0.0 443 0.0.0.0 0 tcp
```

4. Create filter rules to forward all SSL traffic (including encrypted and cleartext traffic) to the SSL interception module.

```
AN(config)#bridge apprule br1 0.0.0.0 0 0.0.0.0 443 tcp
AN(config)#bridge apprule br1 0.0.0.0 8443 0.0.0.0 0 tcp
AN(config)#bridge apprule br2 0.0.0.0 443 0.0.0.0 0 tcp
```

```
AN(config)#bridge apprule br2 0.0.0.0 0 0.0.0.0 8443 tcp
```

7.2.4.2 SSL Settings

7.2.4.2.1 Forwarding of Received SSL Traffic to the Security Device

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Create a FWDMAC real service.

   Note that “AB:89:67:45:23:01” does not represent any port. It can be replaced with an arbitrary MAC address, but it must be set.

```
AN(config)#slb real fwdmac rs1 port3 AB:89:67:45:23:01 8443
```

3. Create a real service group using the chi method and add “rs1” to this group.

```
AN(config)#slb group method chi_group chi
AN(config)#slb group member chi_group rs1
```

4. Create a TCPS virtual service.

```
AN(config)#slb virtual tcps vs1 0.0.0.0 443 noarp 0
```

5. Configure a default policy to associate “vs1” with “chi_group”.

```
AN(config)#slb policy default vs1 chi_group
```

6. Create an SSL virtual host and associate it with “vs1”.

```
AN(config)#ssl host virtual vhost1 vs1
```

7.2.4.2.2 Forwarding of Inspected SSL Traffic to the Real Service

1. Create a TCPS real service, and configure it to keep destination IP addresses unchanged when forwarding packets.

   Note that “172.26.6.1” can be replaced with an arbitrary IP address.

```
AN(config)#slb real tcps rs2 172.26.6.1 443 none
AN(config)#slb real settings keepdip rs2
```

2. Create a TCP virtual service.

```
AN(config)#slb virtual tcp vs2 0.0.0.0 8443 noarp 0
```

3. Configure a static policy to associate “vs2” with “rs2”.

```
AN(config)#slb policy static vs2 rs2
```

4. Disable real service health check.

```
AN(config)#health off
```

5. Create an SSL real host “rhost1” and associate it with “rs2”.

```
AN(config)#ssl host real rhost1 rs2
```

7.2.4.3 SSL Interception Settings

1. Enable SSL interception for “vhost1”.

```
AN(config)#ssli on vhost1 1
```

2. Generate SSL interception certificates for “vhost1”, activate them and enable “vhost1”.

Note: The generated CA certificates must also be imported into the trusted CA list of clients’ browsers.

```
AN(config)#ssli cacert rsa vhost1 2048 1 1
AN(config)#ssli cacert ecc vhost1 prime256v1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

Note: Currently, elliptic curve secp521r1 is not widely supported by mainstream browsers. It is recommended to set prime256v1 or secp384r1 to avoid connection failures.

3. Enable SSL interception for “rhost1” and enable “rhost1”.

```
AN(config)#ssli on rhost1 1
AN(config)#ssl start rhost1
```

Note: If the SSL server uses a testing certificate or a self-signed certificate, both the ingress and egress nodes must have its certificate chain (including the root CA certificate) imported. For testing-only purposes, administrators can choose not to import the certificate chain, but must execute the “**ssl globals verifycert off**” command to disable the server authentication function.

7.2.4.4 SPAN Port Settings

1. Configure a filter list.

```
AN(config)#spanport filterlist name f1
```

2. Add filter rules to define the ciphertext SSL traffic to be captured and then sent to the security devices.

```
AN(config)#spanport filterlist member f1 port3 0.0.0.0 8443 0.0.0.0 0 tcp inbound
AN(config)#spanport filterlist member f1 port3 0.0.0.0 0 0.0.0.0 8443 tcp outbound
```

3. Configure a security device group.

```
AN(config)#spanport devicegroup name d1 lb
```

4. Add the security devices as members to the group.

```
AN(config)#spanport devicegroup member d1 port5 00:0c:29:a7:6f:26 abc
AN(config)#spanport devicegroup member d1 port6 00:0c:29:a7:6f:25 abd
```

5. Configure a SPAN port policy to associate the specified filter list with the specified security device group

```
AN(config)#spanport policy p1 f1 d1
```

7.2.5 Distributed Mode: Two L2 APVs + Two L2 Firewalls

In this deployment mode, the APV appliances and the firewalls work in L2 mode, and the two APV appliances play the role of the ingress and egress nodes respectively. The ingress node and the egress node each have a bridge instance configured:

- The bridge on the ingress node is used to transfer SSL traffic to the SSL interception module for decryption and forward the decrypted traffic to the firewall.
- The bridge on the egress node is used to receive the inspected traffic from the firewall and re-encrypt the traffic before sending it to the server.

The network topology and interfaces are as shown in the following figure.

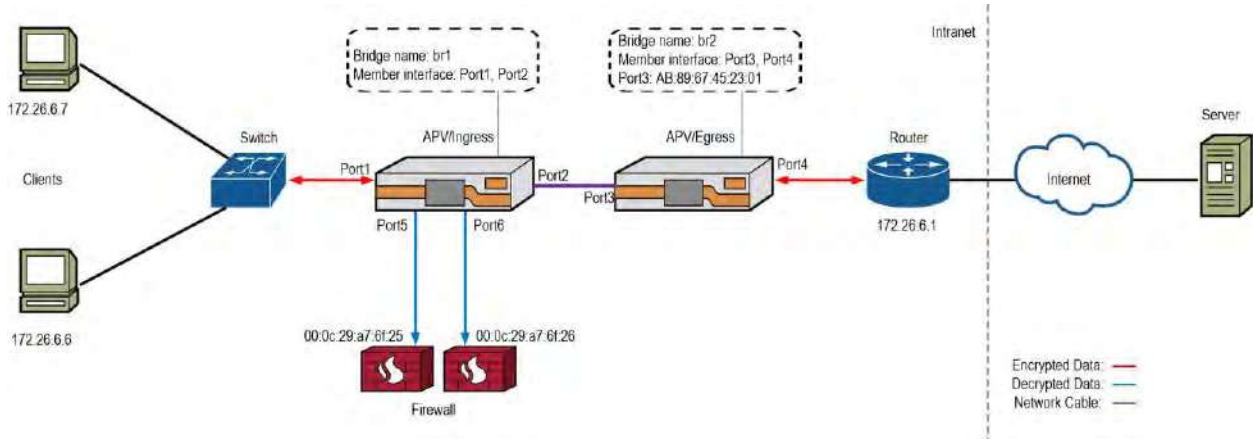

Figure 7–5 Distributed Mode: Two L2 APVs + Two L2 Firewalls

7.2.5.1 Configuring the Ingress Node

7.2.5.1.1 Bridge Settings

1. Create a bridge instance.

```
AN(config)#bridge name br1
```

2. Add members to the created bridge instance.

```
AN(config)#bridge member br1 port1 yes
AN(config)#bridge member br1 port2 yes
```

3. Create filter rules to bypass returned SSL traffic and to acquire server certificates.

```
AN(config)#bridge apprule br1 0.0.0.0 443 0.0.0.0 0 tcp
```

4. Create filter rules to forward clients' encrypted SSL traffic and servers' cleartext SSL traffic to the SSL interception module.

```
AN(config)#bridge apprule br1 0.0.0.0 0 0.0.0.0 443 tcp
AN(config)#bridge apprule br1 0.0.0.0 8443 0.0.0.0 0 tcp
```

7.2.5.1.2 SSL Settings

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Create a TCPS virtual service.

```
AN(config)#slb virtual tcps vs1 0.0.0.0 443 noarp 0
```

3. Create a FWDMAC real service.

   Note that “AB:89:67:45:23:01” does not represent any port. It can be replaced with an arbitrary MAC address, but it must be set.

```
AN(config)#slb real fwdmac rs1 port2 AB:89:67:45:23:01 8443
```

4. Create an L2 real service group using the chi method and add “rs1” to this group.

```
AN(config)#slb group method chi_group chi
AN(config)#slb group member chi_group rs1
```

5. Configure a default policy to associate “vs1” with “chi_group”.

```
AN(config)#slb policy default vs1 chi_group
```

6. Create an SSL virtual host and associate it with “vs1”.

```
AN(config)#ssl host virtual vhost1 vs1
```

7.2.5.1.3 SSL Interception Settings

1. Enable SSL interception for “vhost1”.

```
AN(config)#ssli on vhost1 0
```

2. Generate SSL interception certificates for “vhost1”, activate them and enable “vhost1”.

```
AN(config)#ssli cacert rsa vhost1 2048 1 1
AN(config)#ssli cacert ecc vhost1 prime256v1 1
AN(config)#ssl activate certificate vhost1 1
AN(config)#ssl start vhost1
```

Note:

1. The generated CA certificates must also be imported into the trusted CA list of clients’ browsers.
2. If the SSL server uses a testing certificate or a self-signed certificate, both the ingress and egress nodes must have its certificate chain (including the root CA certificate) imported. For testing-only purposes, administrators can choose not to import the certificate chain, but must execute the “**ssl globals verifycert off**” command to disable the server authentication function.
3. Currently, elliptic curve secp521r1 is not widely supported by mainstream browsers. It is recommended to set prime256v1 or secp384r1 to avoid connection failures.

7.2.5.1.4 SPAN Port Settings

1. Configure a filter list.

```
AN(config)#spanport filterlist name f1
```

2. Add filter rules to define the ciphertext SSL traffic to be captured and then sent to the security devices.

```
AN(config)#spanport filterlist member f1 port2 0.0.0.0 8443 0.0.0.0 0 tcp inbound
AN(config)#spanport filterlist member f1 port2 0.0.0.0 0 0.0.0.0 8443 tcp outbound
```

3. Configure a security device group.

```
AN(config)#spanport devicegroup name d1 lb
```

4. Add the security device ports as members to the group.

```
AN(config)#spanport devicegroup member d1 port5 00:0c:29:a7:6f:25 abc
AN(config)#spanport devicegroup member d1 port6 00:0c:29:a7:6f:26 abd
```

5. Configure a SPAN port policy to associate the specified filter list with the specified security device group

```
AN(config)#spanport policy p1 f1 d1
```

7.2.5.2 Configuring the Egress Node

7.2.5.2.1 Bridge Settings

1. Create a bridge instance.

```
AN(config)#bridge name br2
```

2. Add members to the created bridge instance.

```
AN(config)#bridge member br2 port3 yes
AN(config)#bridge member br2 port4 yes
```

3. Create filter rules to forward servers' encrypted SSL traffic and clients' cleartext SSL traffic to the SSL interception module.

```
AN(config)#bridge apprule br2 0.0.0.0 443 0.0.0.0 0 tcp
AN(config)#bridge apprule br2 0.0.0.0 0 0.0.0.0 8443 tcp
```

7.2.5.2.2 SSL Settings

1. Set the system mode to transparent.

```
AN(config)#system mode transparent
```

2. Configure a TCPS real service, and configure it to keep destination IP addresses unchanged when forwarding packets.

   Note that "172.26.6.1" can be replaced with an arbitrary IP address.

```
AN(config)#slb real tcps rs2 172.26.6.1 443 none
AN(config)#slb real settings keepdip rs2
```

3. Configure a TCP virtual service.

```
AN(config)#slb virtual tcp vs2 0.0.0.0 8443 noarp 0
```

4. Configure a static policy to associate "vs2" with "rs2".

```
AN(config)#slb policy static vs2 rs2
```

5. Disable real service health check.

```
AN(config)#health off
```

6. Create an SSL real host “rhost1” and associate it with “rs2”.

```
AN(config)#ssl host real rhost1 rs2
```

7.2.5.2.3 SSL Interception Settings

1. Enable SSL interception for “rhost1” and enable “rhost1”.

```
AN(config)#ssli on rhost1 0
AN(config)#ssl start rhost1
```

Note: If the SSL server uses a testing certificate or a self-signed certificate, both the ingress and egress nodes must have its certificate chain (including the root CA certificate) imported. For testing-only purposes, administrators can choose not to import the certificate chain, but must execute the “**ssl globals verifycert off**” command to disable the server authentication function.