This article describes how to troubleshoot an AppNav deployment.

Guide Contents
Main Article Understanding the WAAS Architecture and Traffic Flow Preliminary WAAS Troubleshooting Troubleshooting Optimization Troubleshooting Application Acceleration Troubleshooting the CIFS AO Troubleshooting the HTTP AO Troubleshooting the EPM AO Troubleshooting the MAPI AO Troubleshooting the NFS AO Troubleshooting the SSL AO Troubleshooting the Video AO Troubleshooting the Generic AO Troubleshooting Overload Conditions Troubleshooting WCCP **Troubleshooting AppNav** Troubleshooting Disk and Hardware Problems Troubleshooting Serial Inline Clusters Troubleshooting vWAAS Troubleshooting WAAS Express Troubleshooting NAM Integration

Contents

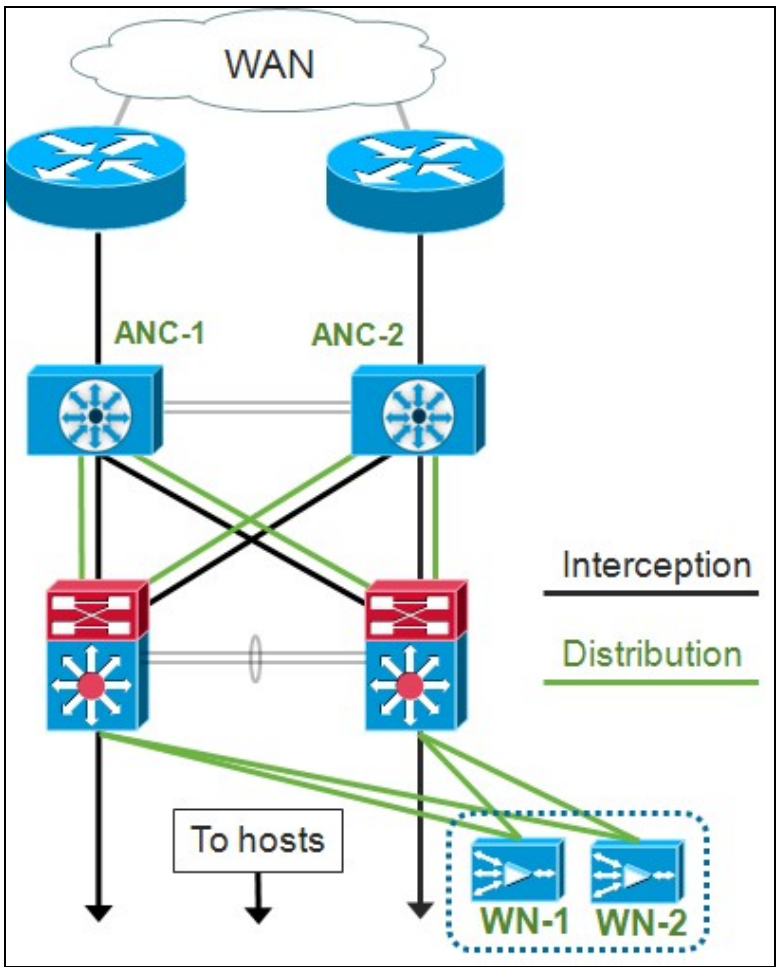
AppNav Troubleshooting

Cisco WAAS AppNav simplifies network integration of WAN optimization and greatly reduces dependency on the intercepting switch or router by using AppNav Controllers (ANCs) to distribute traffic among WAAS Nodes (WNs) for optimization using a powerful class and policy mechanism. You can use WAAS nodes (WNs) to optimize traffic based on sites and/or applications. This article describes how to troubleshoot AppNav.

NOTE: The AppNav feature was introduced in WAAS version 5.0.1. This section is not applicable to earlier WAAS versions.

In-Path (Inline) Interception

In inline mode, ANCs are positioned in the path of network traffic where they intercept packets and distribute them to WNs.



The interface configuration for an inline deployment assigns the interception and distribution roles to separate interfaces on the Cisco AppNav Controller Interface Module. A bridge-group interface is required for interception and consists of two or more physical or port-channel interfaces or one of each. The bridge-group interface does not have fail to wire capability; that is, it fails open and traffic is not mechanically bridged after a device failure or loss of power. AppNav uses clustering to provide high availability if the AppNav Controller Interface Module, the link path, or connectivity to the AppNav Controller Interface Module is lost or there is a power failure.

Note: Bridge interfaces do not block bridge protocol data unit (BPDU) packets and in the case of redundant interfaces that create loops, one of the interfaces is blocked by the Spanning Tree Protocol.

Troubleshooting inline interception consists of these steps:

- Verify correct inline placement of the ANC by checking the network design. If necessary, use basic tools like ping and traceroute, or Layer 7 tools or applications to confirm that the network traffic path is as expected. Check the physical cabling of the ANC.
- Verify that the ANC is set to inline interception mode.
- Verify that the bridge-group interface is configured correctly.

The last two steps can be performed either in Central Manager or at the command line, though the Central Manager is the preferred method and is described first.

In the Central Manager, choose **Devices >** *AppNavController*, then choose **Configure > Interception > Interception Configuration**. Verify that the Interception Method is set to Inline.

In the same window, verify that a bridge interface is configured. If a bridge interface is needed, click **Create Bridge** to create it. You can assign up to two member interfaces to the bridge group. You can use the VLAN Calculator to define the VLAN entries based on include or exclude operations. Note that the bridge interface is not assigned an IP address.

Use the Alarm panel or the **show alarm** exec command to check if any bridge related alarms are raised on the device. A bridge_down alarm indicates that one or more member interfaces in the bridge are down.

From the CLI, follow these steps to configure inline operation:

1. Set the interception method to inline:

```
wave# config
wave(config)# interception-method inline
```

2. Create the bridge-group interface:

```
wave(config)# bridge 1 protocol interception
```

3. (Optional) Specify the list of VLANs to intercept, if needed:

```
wave(config)# bridge 1 intercept vlan-id all
```

4. Add two logical/physical interfaces to the bridge-group interface:

```
wave(config)# interface GigabitEthernet 1/0
wave(config-if)# bridge-group 1
wave(config-if)# exit
wave(config)# interface GigabitEthernet 1/1
wave(config-if)# bridge-group 1
```

```
wave(config-if)# exit
```

You can use the **show bridge** exec command to verify the bridge interface operational status and see statistics for the bridge.

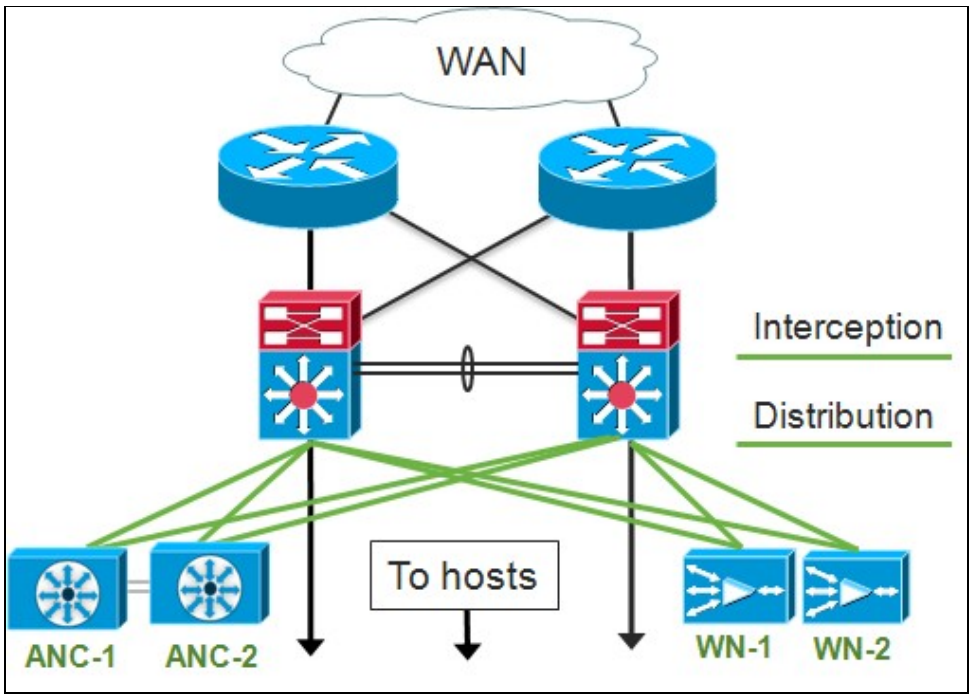
```
wave# show bridge 1
lsp: Link State Propagation
flow sync: AppNav Controller is in the process of flow sync
Member Interfaces:
 GigabitEthernet 1/0
 GigabitEthernet 1/1
Link state propagation: Enabled
VLAN interception:
 intercept vlan-id all                                                  <<< VLANs to intercept

Interception Statistics:
                              GigabitEthernet 1/0    GigabitEthernet 1/1
Operation State                  :   Down           Down(lsp)            <<< Down due to LSP
Input Packets Forwarded/Bridged  :   16188          7845
Input Packets Redirected         :   5068           0
Input Packets Punted             :   1208           605
Input Packets Dropped            :   0              0
Output Packets Forwarded/Bridged :   7843           21256
Output Packets Injected          :   301            301
Output Packets Dropped           :   2              0
```

In the example above, the Gig 1/0 interface is down and the Gig 1/1 interface is also down due to link state propagation (LSP). You might also see Down(flow sync), which means that the ANC is joining a cluster and synchronizing flow information with other ANCs in the cluster. It keeps the interception path (bridge interface) shut for about two minutes until all ANCs are synchronized so that existing flows can be distributed correctly.

The lower part of the output shows traffic statistics for the member interfaces.

Off-Path (WCCP) Interception

In WCCP mode, WCCP routers are positioned in the path of network traffic where they intercept packets and redirect them to ANCs, which are located off-path. Since AppNav handles the interception processing, the intelligent flow distribution, and load consideration between WAAS accelerators, the WCCP configuration on the routers is simplified significantly.



In the interface configuration for an off-path deployment, the interception and distribution roles can share the same interfaces on the Cisco AppNav Controller Interface Module, but it is not required.

Troubleshooting off-path interception consists of these steps:

- Verify correct placement of the WCCP routers to ensure they are in the path of traffic going to and from the optimized hosts. You can use the **show run** or **show wccp** commands to verify that these are the same routers that are configured for WCCP. If necessary, use basic tools like ping and traceroute, or Layer 7 tools or applications to confirm that all traffic needing optimization passes through the WCCP routers.
- Verify the WCCP configuration on the WAAS ANCs, using either the Central Manager (preferred) or the CLI.
- Verify the WCCP configuration on the redirecting routers, using the router CLI.

To verify the WCCP configuration on the ANCs, in the Central Manager, choose **Devices >** *AppNavController*, then choose **Configure > Interception > Interception Configuration**.

- Verify that the Interception Method is set to WCCP.
- Verify that the Enable WCCP Service check box is checked.
- Verify that the Use Default Gateway as WCCP Router check box is checked or that the WCCP router IP addresses are listed in the WCCP Router field.
- Verify that the other settings such as the load balancing mask and redirect method are configured properly for your deployment.

Check for any WCCP related alarms on the ANCs that are part of the router WCCP farm. On the Central Manager, click the Alarms panel at the bottom of the screen or use the **show alarm** command on each device to view alarms. Correct any alarm conditions by changing the configuration on the ANC or router, as needed.

From the CLI, follow these steps to configure WCCP operation:

1. Set the interception method to wccp.

```
wave# config
```

```
wave(config)# interception-method wccp
```

2. Configure the WCCP router list, which contains the IP addresses of the routers participating in the WCCP farm.

```
wave(config)# wccp router-list 1 10.10.10.21 10.10.10.22
```

3. Configure the WCCP service ID. A single service ID is preferred for AppNav, though two service IDs are supported.

```
wave(config)# wccp tcp-promiscuous 61
```

4. Associate the configured router list with the WCCP service.

```
wave(config-wccp-service)# router-list-num 1
```

5. Configure the WCCP assignment method (only the mask method is supported on an ANC). If you do not specify the dst-ip-mask or src-ip-mask options, the default source IP mask is set to f and the destination IP mask is set to 0.

```
wave(config-wccp-service)# assignment-method mask
```

6. Configure the WCCP redirect method (the egress and return methods are set automatically to match the redirect method and are not configurable for an ANC). You can choose L2 (the default) or GRE. L2 requires that the ANC has a Layer 2 connection with the router and the router is also configured for Layer 2 redirection.

```
wave(config-wccp-service)# redirect-method gre
```

7. Enable the WCCP service.

```
wave(config-wccp-service)# enable
```

Verify WCCP interception on each ANC by using the **show running-config** command. The two examples below show the running config output for L2 redirect and GRE redirect.

Show running-config wccp (for L2 redirect):

```
wave# sh run wccp
wccp router-list 1 10.10.10.21 10.10.10.22
wccp tcp-promiscuous service-pair 61
  router-list-num 1
  enable                                          <<< L2 redirect is default so is not shown in r
  exit
```

Show running-config wccp (for GRE):

```
wave# sh run wccp
wccp router-list 1 10.10.10.21 10.10.10.22
wccp tcp-promiscuous service-pair 61
 router-list-num 1
 redirect-method gre                             <<< GRE redirect method is configured
 enable
 exit
```

Verify the WCCP status on each ANC by using the **show wccp status** command.

```
wave# show wccp routers
WCCP Interception :
Configured State :  Enabled                        <<< Shows Disabled if WCCP is not configured
Operational State : Enabled                        <<< Shows Disabled if WCCP is not enabled
 Services Enabled on this WAE:
        TCP Promiscuous 61                         <<< Shows NONE if no service groups are configur
```

Verify the routers that have responded to keep-alive messages in the WCCP farm by using the **show wccp routers** command.

```
wave# show wccp routers
Router Information for Service Id: 61

       Routers Seeing this Wide Area Engine(2)
       Router Id       Sent To                     <<< List of routers seen by this ANC
       192.168.1.1     10.10.10.21
       192.168.1.2     10.10.10.22
       Routers not Seeing this Wide Area Engine    <<< List of routers not seen by this ANC
               -NONE-
       Routers Notified of from other WAE's        <<< List of routers notified of but not configur
               -NONE-
```

Verify each ANCs' view of the other ANCs in the WCCP farm and the routers reachable by each of them by using the **show wccp clients** command.

```
wave# show wccp clients
Wide Area Engine List for Service: 61
Number of WAE's in the Cache farm: 2                          <<< Number of ANCs in the farm
       IP address = 10.10.10.31   Lead WAE = NO   Weight = 0   <<< Entry for each ANC in the farm
       Routers seeing this Wide Area Engine(2)
               192.168.1.1                                    <<< List of routers seeing this ANC
               192.168.1.2

       IP address = 10.10.10.32   Lead WAE = YES  Weight = 0   <<< YES indicates ANC is serving as
       Routers seeing this Wide Area Engine(2)
               192.168.1.1                                    <<< List of routers seeing this ANC
               192.168.1.2
```

Verify that packets are being received by each ANC from the routers in the farm by using the **show statistics wccp** command. Statistics for traffic that is received from, passed through, and sent to each router are shown. Cumulative statistics for all routers in the farm are shown at the bottom. A similar command is **show wccp statistics**. Note that "OE" refers to the ANC devices here.

```
wave# sh statistics wccp

WCCP Stats for Router              : 10.10.10.21
 Packets Received from Router      : 1101954
 Bytes Received from Router        : 103682392
 Packets Transmitted to Router     : 1751072
 Bytes Transmitted to Router       : 2518114618
 Pass-thru Packets sent to Router  : 0
 Pass-thru Bytes sent to Router    : 0
 Redirect Packets sent to OE       : 1101954
 Redirect Bytes sent to OE         : 103682392

WCCP Stats for Router              : 10.10.10.22
 Packets Received from Router      : 75264
 Bytes Received from Router        : 10732204
 Packets Transmitted to Router     : 405193
```

```
 Bytes Transmitted to Router       : 597227459
 Pass-thru Packets sent to Router  : 0
 Pass-thru Bytes sent to Router    : 0
 Redirect Packets sent to OE       : 75264
 Redirect Bytes sent to OE         : 10732204

Cummulative WCCP Stats:
 Total Packets Received from all Routers : 1177218
 Total Bytes Received from all Routers : 114414596
 Total Packets Transmitted to all Routers : 2156265
 Total Bytes Transmitted to all Routers : 3115342077
 Total Pass-thru Packets sent to all Routers : 0
 Total Pass-thru Bytes sent to all Routers : 0
 Total Redirect Packets sent to OE : 1177218
 Total Redirect Bytes sent to OE   : 114414596
```

Configuring and Verifying WCCP Interception on the Router

To configure WCCP interception on each router in the WCCP farm, follow these steps.

1. Configure the WCCP service on the router by using the **ip wccp** router command.

```
Core-Router1 configure terminal
Core-Router1(config)# ip wccp 61
```

2. Configure WCCP interception on the router LAN and WAN interfaces. You can configure the same service ID on both interfaces if you are using a single service ID on the ANCs.

```
Core-Router1(config)# interface GigabitEthernet0/0
Core-Router1(config-subif)# ip address 10.20.1.1 255.255.255.0
Core-Router1(config-subif)# ip wccp 61 redirect in
Core-Router1(config-subif)# ip router isis inline_wccp_pod
Core-Router1(config-subif)# exit

Core-Router1(config)# interface GigabitEthernet0/1
Core-Router1(config-subif)# ip address 10.19.1.1 255.255.255.0
Core-Router1(config-subif)# ip wccp 61 redirect in
Core-Router1(config-subif)# ip router isis inline_wccp_pod
Core-Router1(config-subif)# glbp 701 ip 10.19.1.254
Core-Router1(config-subif)# duplex auto
Core-Router1(config-subif)# speed auto
Core-Router1(config-subif)# media-type rj45
Core-Router1(config-subif)# exit
```

3. (Optional) Configure a tunnel interface if you are using generic GRE egress (only if you chose GRE for the ANC WCCP redirect method).

```
Core-Router1(config)# interface Tunnel1
Core-Router1(config-subif)# ip address 192.168.1.1 255.255.255.0
Core-Router1(config-subif)# no ip redirects
Core-Router1(config-subif)# tunnel source GigabitEthernet0/0.3702
Core-Router1(config-subif)# tunnel mode gre multipoint
```

Verify the WCCP configuration on each router in the farm by using the **show wccp** command.

```
Core-Router1 sh ip wccp 61 detail
WCCP Client information:
       WCCP Client ID:          10.10.10.31            <<< ANC IP address
       Protocol Version:        2.00
       State:                   Usable
```

```
Redirection:             GRE                    <<< Negotiated WCCP parameters
Packet Return:           GRE                    <<<
Assignment:              MASK                   <<<
Connect Time:            00:31:27
Redirected Packets:
  Process:               0
  CEF:                   0
GRE Bypassed Packets:
  Process:               0
  CEF:                   0
Mask Allotment:          16 of 16 (100.00%)
Assigned masks/values:   1/16

Mask  SrcAddr    DstAddr    SrcPort DstPort
----  -------    -------    ------- -------
0000: 0x0000000F 0x00000000 0x0000  0x0000      <<< Configured mask

Value SrcAddr    DstAddr    SrcPort DstPort
----- -------    -------    ------- -------
0000: 0x00000000 0x00000000 0x0000  0x0000      <<< Mask assignments
0001: 0x00000001 0x00000000 0x0000  0x0000
0002: 0x00000002 0x00000000 0x0000  0x0000
0003: 0x00000003 0x00000000 0x0000  0x0000
0004: 0x00000004 0x00000000 0x0000  0x0000
0005: 0x00000005 0x00000000 0x0000  0x0000
0006: 0x00000006 0x00000000 0x0000  0x0000
0007: 0x00000007 0x00000000 0x0000  0x0000
0008: 0x00000008 0x00000000 0x0000  0x0000
0009: 0x00000009 0x00000000 0x0000  0x0000
0010: 0x0000000A 0x00000000 0x0000  0x0000
0011: 0x0000000B 0x00000000 0x0000  0x0000
0012: 0x0000000C 0x00000000 0x0000  0x0000
0013: 0x0000000D 0x00000000 0x0000  0x0000
0014: 0x0000000E 0x00000000 0x0000  0x0000
0015: 0x0000000F 0x00000000 0x0000  0x0000
```

Additional Information

For additional information, see these documents:

- WCCP Network Integration with Cisco Catalyst 6500: Best Practice Recommendations for Successful Deployments
- Cisco Wide Area Application Services Web Cache Communication Protocol Redirection: Cisco Router Platform Support
- Configuring Advanced WCCP Features on Routers, from the *Cisco Wide Area Application Services Configuration Guide*
- Configuring WCCP on WAEs, from the *Cisco Wide Area Application Services Configuration Guide*

Network Connectivity Troubleshooting

When troubleshooting WAAS, it may be helpful to determine how the network is behaving with WAAS disabled. This is helpful when traffic is not only failing to be optimized, but failing to get through at all. In these cases, it may turn out that the problem is not WAAS related. Even in cases where traffic is getting through, this technique may help determine which WAAS devices require troubleshooting.

Before testing Layer 3 connectivity, verify that the AppNav Controller Interface Module is connected to proper switch ports. If the connected switch supports and has Cisco Discovery Protocol (CDP) enabled, run the command **show cdp neighbors detail** to verify proper connectivity to the network switch.

Disabling WAAS may not be applicable in all cases. If some traffic is being optimized and some is not, it may be unacceptable to disable WAAS, thereby disrupting the traffic that is being optimized successfully. In such a case, the interception ACL or the AppNav policy can be used to pass through the specific type of traffic that is experiencing problems. For details, see the section Passing Through Specific Traffic.

To disable WAAS, different steps are performed for inline mode than for off-path mode:

- Inline mode requires putting the interception bridge in the pass-through state. For details, see the section Disabling an Inline ANC.
- Off-path mode requires disabling the WCCP protocol. For details, see the section Disabling an Off-Path ANC.

In AppNav environments, only the ANCs need to be disabled. WNs need not be disabled, since they do not participate in interception.

Once WAAS is disabled, check network connectivity using standard methods.

- Check Layer 3 connectivity using tools like ping and traceroute.
- Check application behavior to determine upper layer connectivity
- If the network is experiencing the same connectivity problems that it was with WAAS enabled, the problem is most likely non-WAAS related.
- If the network is working fine with WAAS disabled, but had connection problems with WAAS enabled, then there are probably one or more WAAS devices requiring attention. The next step is to isolate the problem to specific WAAS devices.
- If the network has connectivity with and without WAAS enabled, but there is no optimization, then there are probably one or more WAAS devices requiring attention. The next step is to isolate the problem to specific WAAS devices.

To check network behavior with WAAS enabled, follow these steps:

1. Reenable the WAAS functionality on the WAAS ANCs and, if applicable, the WCCP routers.

2. If you have determined that there is a WAAS-related problem, enable each AppNav cluster, and/or ANC individually, to isolate it as a potential cause of the observed problem.

3. As each ANC is enabled, perform the same basic network connectivity tests as in earlier steps and note whether this specific ANC seems to be operating correctly. Do not be concerned with individual WNs at this stage. The goal at this stage is to determine which clusters, and which specific ANCs, are experiencing desired or undesired behavior.

4. As each ANC is enabled and tested, disable it again so that the next one can be enabled. Enabling and testing each ANC in turn allows you to determine which ones require further troubleshooting.

This troubleshooting technique is most applicable in situations where the WAAS configuration appears to be not only failing to optimize, but also causing problems with normal network connectivity.

Passing Through Specific Traffic

You can pass through specific traffic either by using an interception ACL or by configuring the AppNav policy for pass through.

- Create an ACL that denies the specific traffic to be passed through and permits everything else. In this example, we want to pass through HTTP traffic (dest port 80). Set the ANC interception access

list to the defined ACL. Connections destined for port 80 are passed through. You can use the **show statistics pass-through type appnav** command to verify that pass-through is happening by checking that the PT Intercept ACL counters are incrementing.

```
anc# config
anc(config)# ip access-list extended pt_http
anc(config-ext-nacl)# deny tcp any any eq 80
anc(config-ext-nacl)# permit ip any any
anc(config-ext-nacl)# exit
anc(config)# interception appnav-controller access-list pt_http
```

- Configure the ANC policy to pass through traffic matching specific classes.

```
class-map type appnav HTTP
   match tcp dest port 80

policy-map type appnav my_policy
.
.
.
class HTTP
   pass-through
```

Disabling an Inline ANC

There are several ways to disable an inline ANC by putting it in pass-through state:

- Set the interception bridge VLAN list to none. In the Central Manager, choose an ANC device, then choose **Configure > Interception > Interception Configuration**. Select the bridge interface and click the **Edit** taskbar icon. Set the VLANs field to the value "none".
- Disable the service context containing the ANC. In the Central Manager, choose a cluster, then click the AppNav Controllers tab, select an ANC, and click the **Disable** taskbar icon.
- Apply an interception ACL with "deny ALL" criteria. This method is preferred. (The first two methods disrupt existing optimized connections.) Define an ACL with the deny ALL criteria. In the Central Manager, choose an ANC device, then choose **Configure > Interception > Interception Access List**, and choose the deny ALL access list in the AppNav Controller Interception Access List drop-down list.

To disable interception with an ACL from the CLI, use the following commands:

```
anc# config
anc(config)# ip access-list standard deny
anc(config-std-nacl)# deny any
anc(config-std-nacl)# exit
anc(config)# interception appnav-controller access-list deny
```

Putting an ANC in pass-through state:

- Disables WAAS interception, not the interfaces.
- Disables all WAAS optimization.
- Causes all traffic to pass through unaffected.

Disabling an Off-Path ANC

To disable an ANC that is running in off-path mode, disable the WCCP protocol for the ANC. You can do this action on the ANC or on the redirecting router or both. On the ANC, you can disable or delete the

WCCP services, or you can remove the interception method or change it from WCCP to another method.

To disable WCCP interception, in the Central Manager, choose an ANC device, then choose **Configure > Interception > Interception Configuration**. Uncheck the Enable WCCP Service check box or click the Remove Settings taskbar icon to remove WCCP interception settings completely (they will be lost).

To disable WCCP interception from the CLI, use the following commands:

```
anc# config
anc(config)# wccp tcp-promiscuous service-pair 61
anc(config-wccp-service)# no enable
```

In some cases, there may be multiple ANCs receiving redirected traffic from the same router. For convenience, you may choose to disable WCCP at the router, rather than the ANCs. The advantage is that you can remove multiple ANCs from a WCCP farm in a single step. The disadvantage is that you cannot do this from the WAAS Central Manager.

To disable WCCP at the router, use the following syntax:

```
RTR1(config)# no ip wccp 61
RTR1(config)# no ip wccp 62    <<< Only needed if you are using two WCCP service IDs
```

To reenable WCCP at the router, use the following syntax:

```
RTR1(config)# ip wccp 61
RTR1(config)# ip wccp 62       <<< Only needed if you are using two WCCP service IDs
```

At each WCCP router, verify that the ANCs you chose to disable are not showing up as WCCP clients. The following output is displayed when the WCCP services have been deleted on the router.

```
RTR1# show ip wccp 61
The WCCP service specified is not active.
```

AppNav Cluster Troubleshooting

To troubleshoot an AppNav Cluster, you can use the following tools:

- AppNav Alarms
- Central Manager Monitoring
- AppNav CLI Commands for Monitoring Cluster and Device Status
- AppNav CLI Commands for Monitoring Flow Distribution Statistics
- Connection Tracing
- AppNav Debug Logging

AppNav Alarms

The Cluster Membership Manager (CMM) raises the following the alarms due to error conditions:

- Degraded Cluster (Critical)--Partial visibility among ANCs. ANC will pass through new connections.
- Convergence Failed (Critical)--ANC failed to converge on a stable view of ANCs and WNs. ANC will pass through new connections.
- ANC Join Failed (Critical)--ANC failed to join an existing cluster due to potential degradation of the cluster with the ANC in it.

- ANC Mixed Farm (Minor)--ANCs in the cluster are running different but compatible versions of the cluster protocol.
- ANC Unreachable (Major)--A configured ANC is unreachable.
- WN Unreachable (Major)--A configured WN is unreachable. This WN is not used for traffic redirection.
- WN Excluded (Major)--A configured WN is reachable but excluded because one or more other ANCs cannot see it. This WN is not used for traffic redirection (new connections).

You can see alarms in the Central Manager Alarms panel or by using the **show alarms** EXEC command on a device.

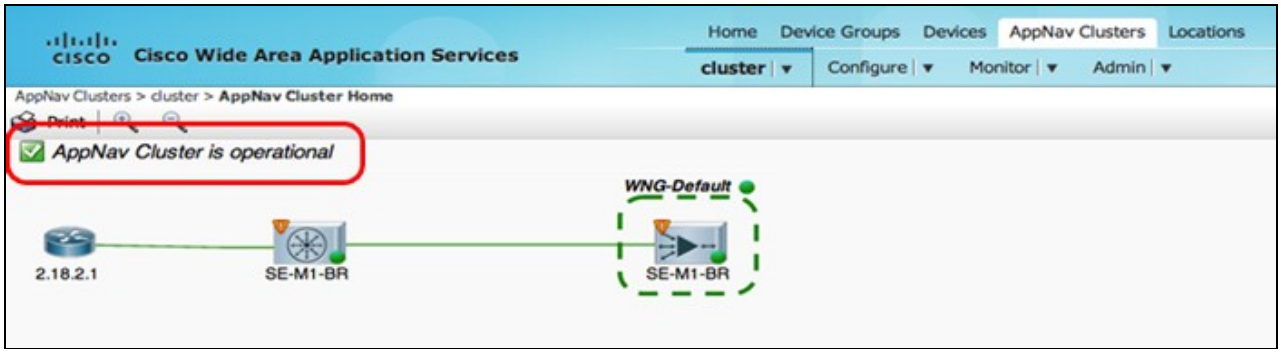
Note: The CMM is an internal AppNav component that manages the grouping of ANCs and WNs into an AppNav cluster associated with a service context.

Central Manager Monitoring

You can use the Central Manager to verify, monitor, and troubleshoot AppNav clusters. The Central Manager has a global view of all registered WAAS devices in your network and can quickly help you locate most AppNav issues.

From the Central Manager menu, choose **AppNav Clusters >** *cluster-name*. The cluster home window displays the cluster topology (including WCCP and gateway routers), overall cluster status, device status, device group status, and link status.

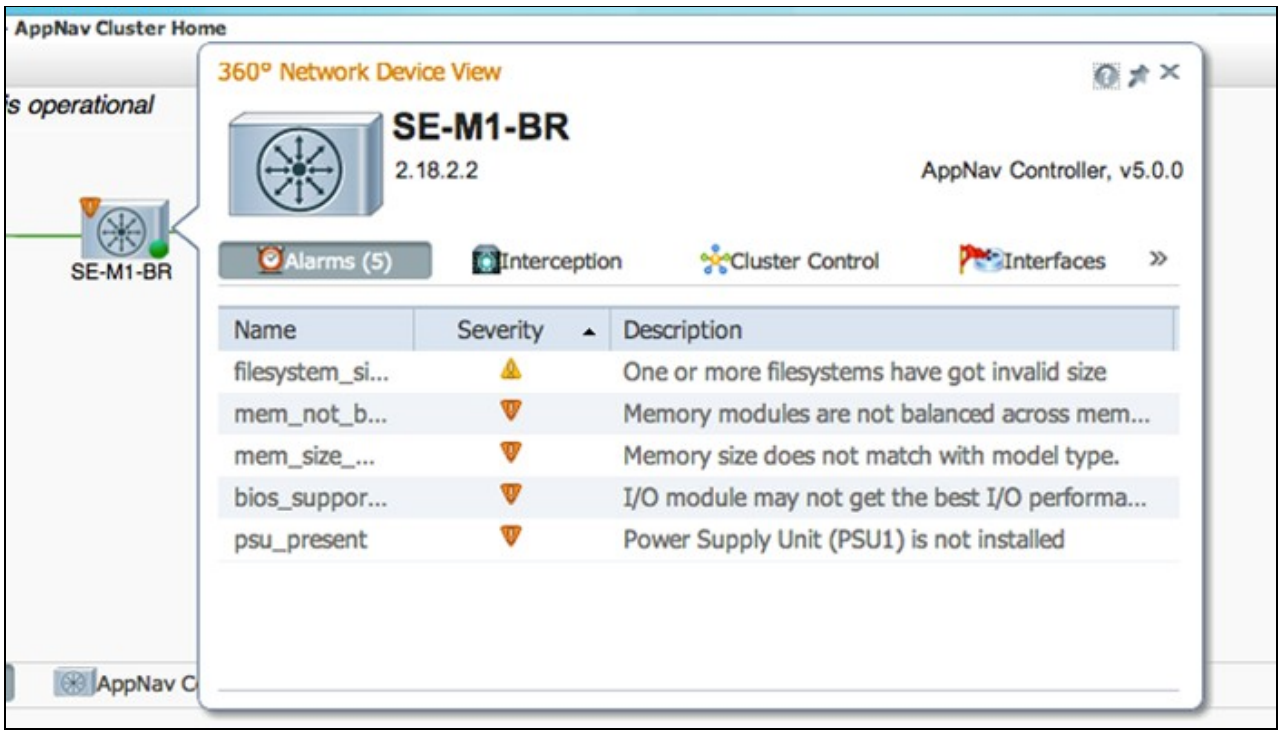
First, verify that the overall cluster status is operational.



Note that the ANC and WN icons shown in this diagram have the same device name because they reside on the same device. On an ANC that is also optimizing traffic as a WN, these two functions are shown as separate icons on the topology diagram.

An orange triangle warning indicator is shown on any device for which the Central Manager may not have current information because the device has not responded within the last 30 seconds (the device could be offline or unreachable).

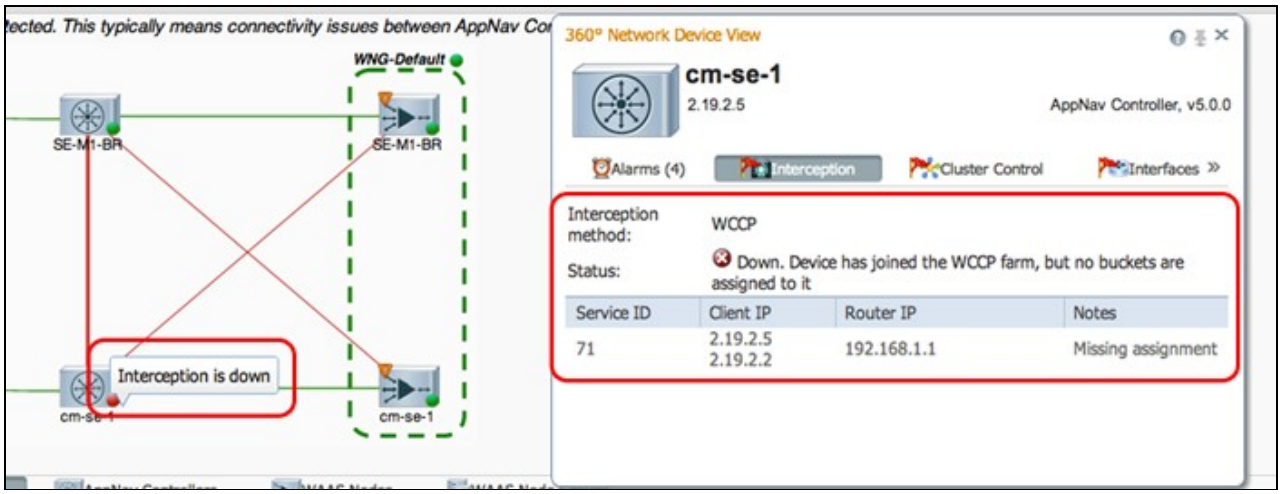
You can get a detailed 360 degree status view of any ANC or WN device by hovering your cursor over the device icon. The first tab displays alarms on the device. You should resolve any alarms that are inhibiting proper cluster operation.



Click the Interception tab to verify the device interception method on each ANC.

File:Waast-an-interceptiontab.png

If interception is down, the status appears as follows:



Click the Cluster Control tab to see the IP address and status of each device in the cluster that this ANC can see. Each ANC in the cluster should have the same list of devices. If not, it indicates a configuration or network issue.

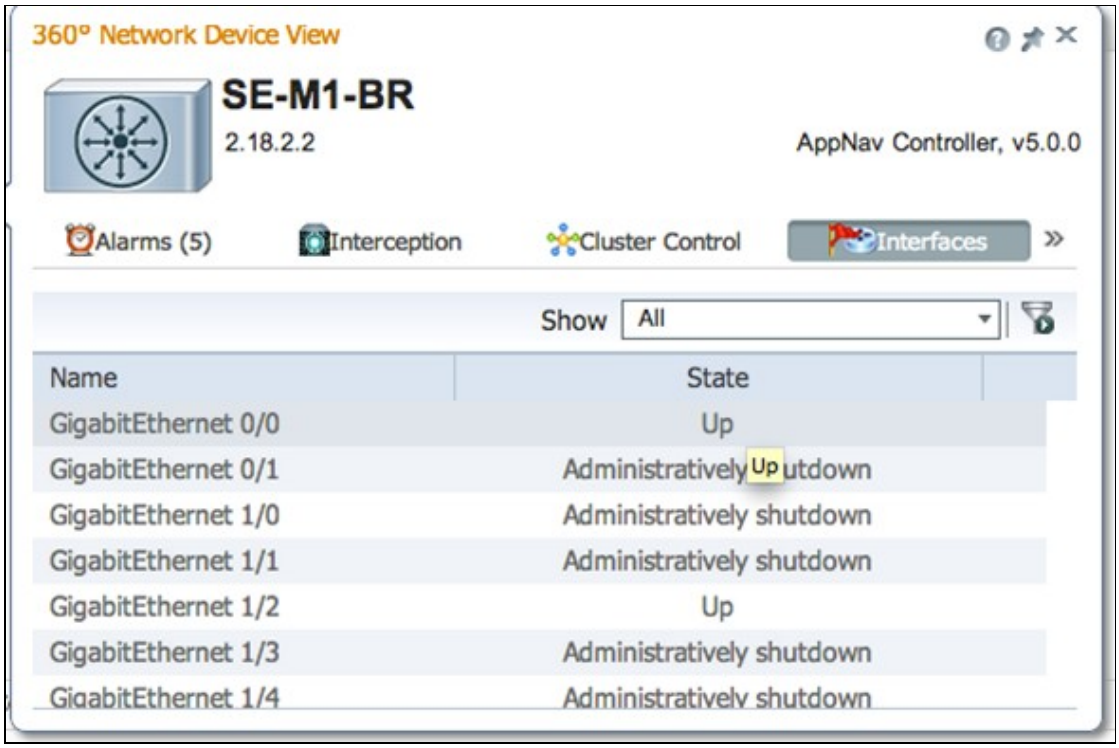
File:Waast-an-clustercontrol.png

If all ANCs cannot see each other, the cluster is nonoperational and all traffic is passed through due to the cluster's inability to synchronize flows.

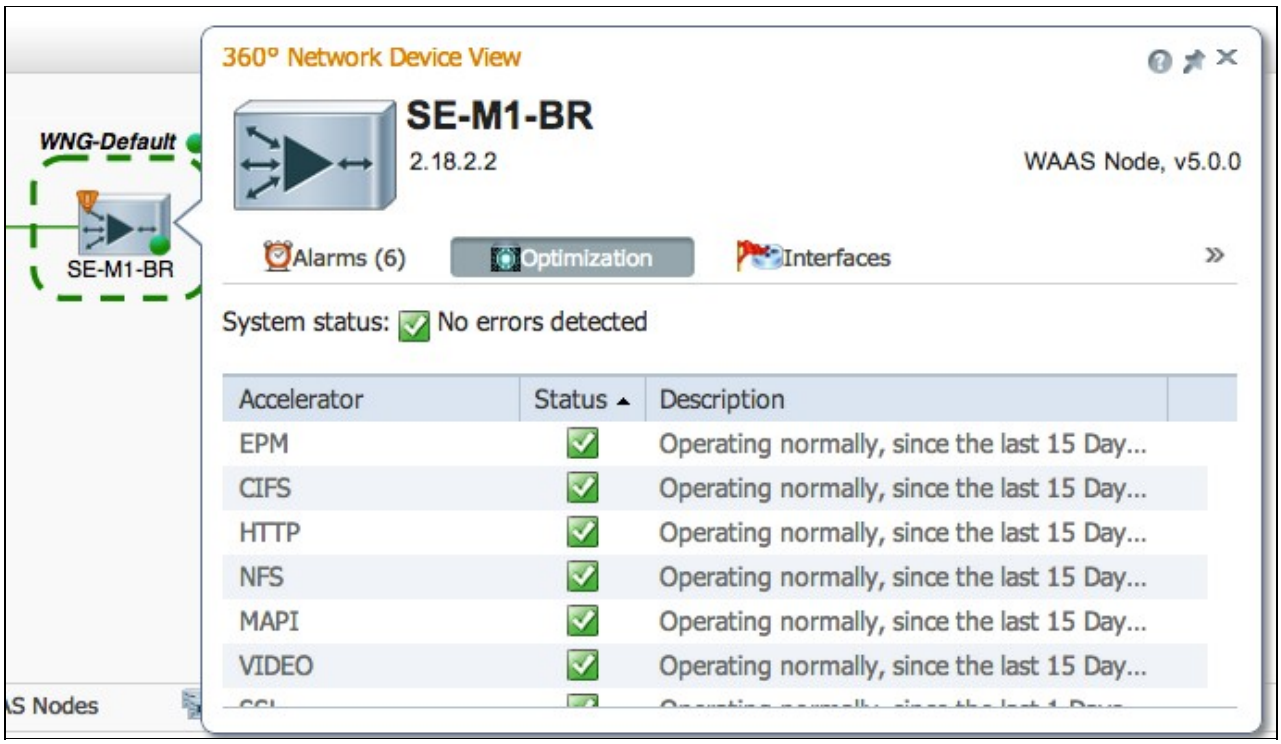
If all ANCs are connected but have different views of the WNs, the cluster is in a degraded state. Traffic is still distributed, but only to the WNs that are seen by all ANCs.

Any WNs not seen by all ANCs are excluded.

Click the Interfaces tab to verify the state of the physical and logical interfaces on the ANC.

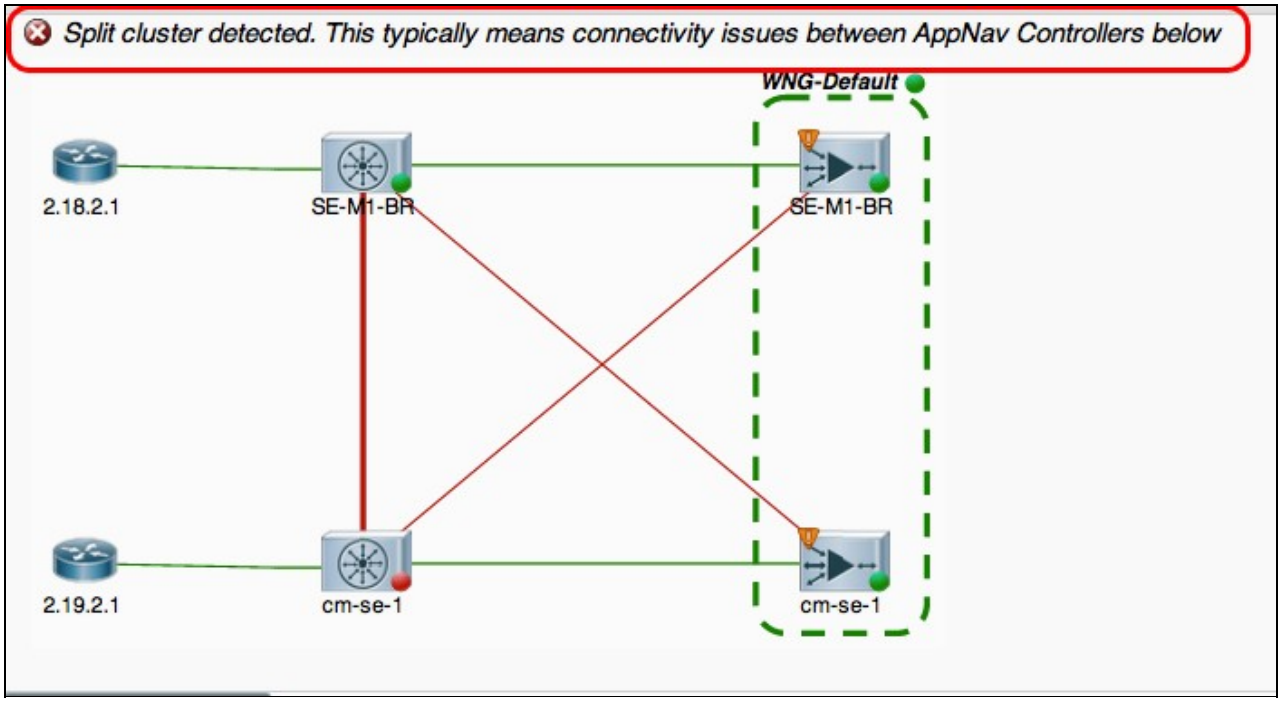


Look at the 360 degree view on each WN in the cluster and verify the green status of all accelerators in the Optimization tab. A yellow status for an accelerator means that the accelerator is running but is unable to service new connections, for example because it is overloaded or because its license has been removed. A red status indicates that the accelerator is not running. If any accelerators are yellow or red, you must separately troubleshoot those accelerators. If the Enterprise license is missing, the description reads System license has been revoked. Install the Enterprise license in the **Admin > History > License Management** device page.



A split cluster results from connectivity issues between ANCs in the cluster. If the Central Manager can communicate with all ANCs, it can detect a split cluster, however, if it cannot communicate with some ANCs, it cannot detect the split. The "Management status is offline" alarm is raised if the Central Manager loses connectivity with any device and the device is shown as offline in the Central Manager.

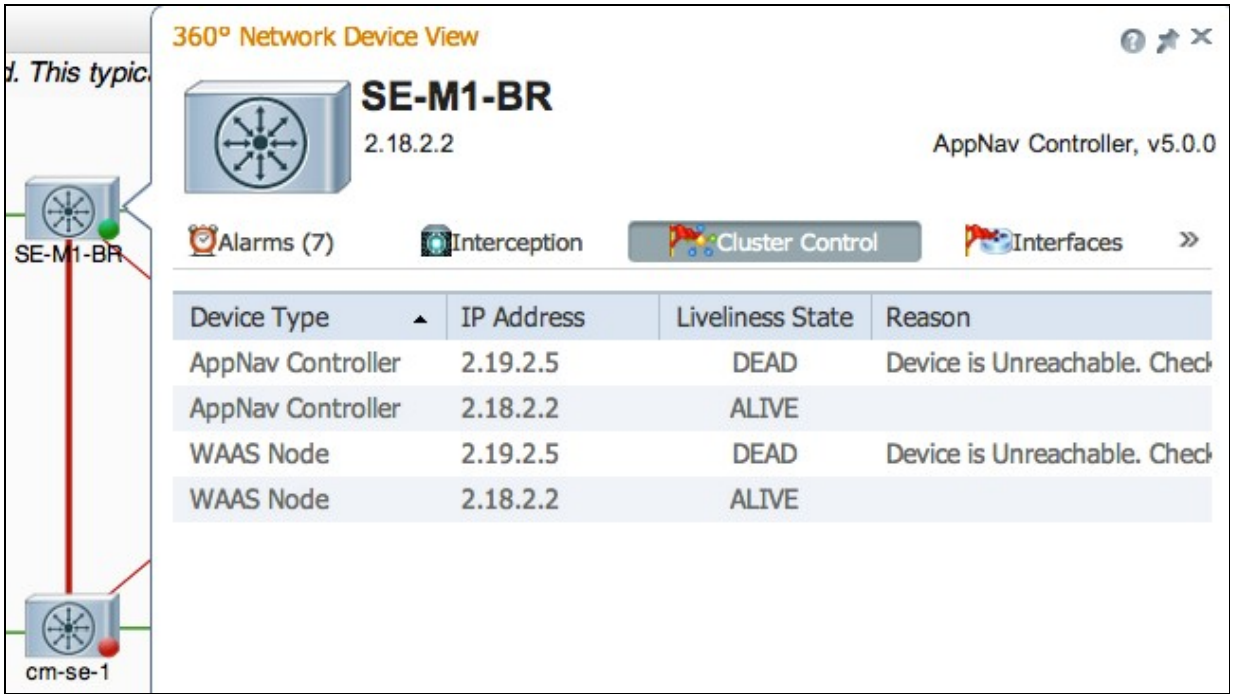
It is best to separate the management interfaces from the data interfaces to maintain management connectivity even if a data link is down.



In a split cluster, each subcluster of ANCs independently distributes flows to the WNGs that it can see, but since flows between the subclusters are not coordinated, it can cause reset connections and the overall cluster

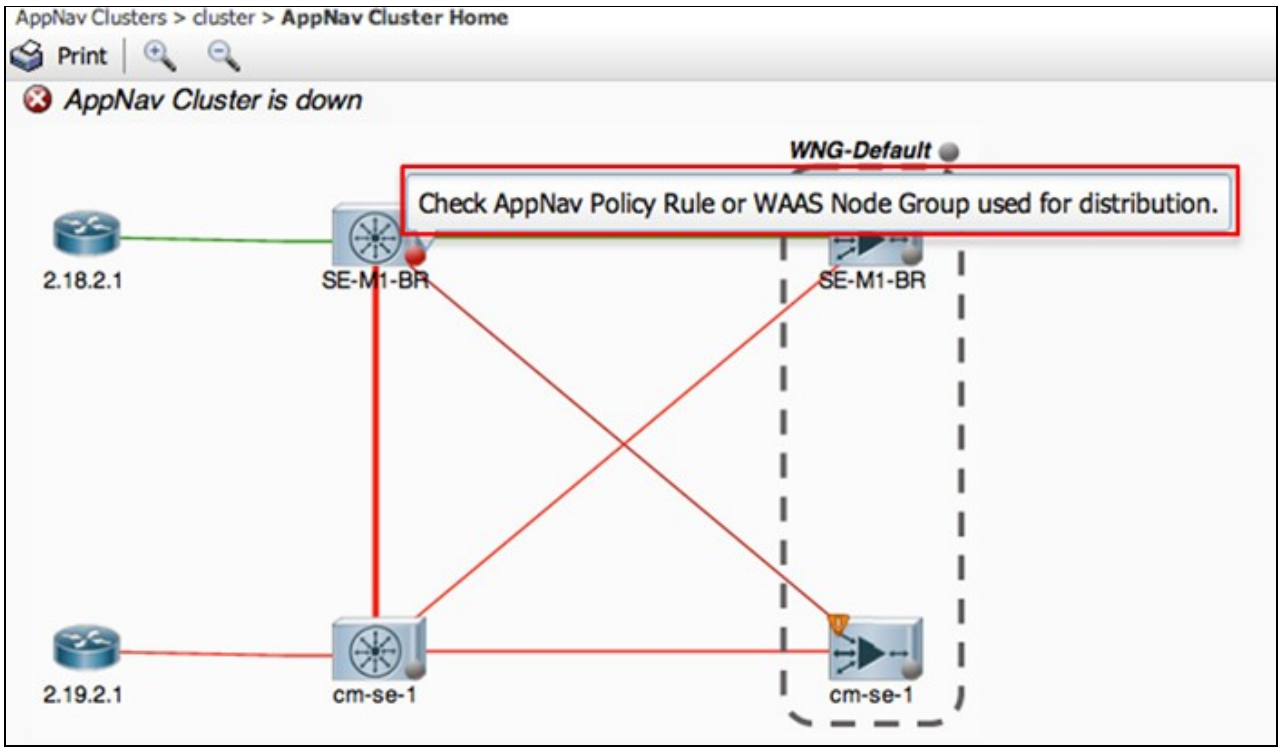
performance is degraded.

Check the Cluster Control tab of each ANC to see if one or more ANCs are unreachable. The "Service controller is unreachable" alarm is raised if two ANCs that once could communicate with each other lose connectivity between themselves but this situation is not the only cause of a split cluster so it is best to check the Cluster Control tab of each ANC.

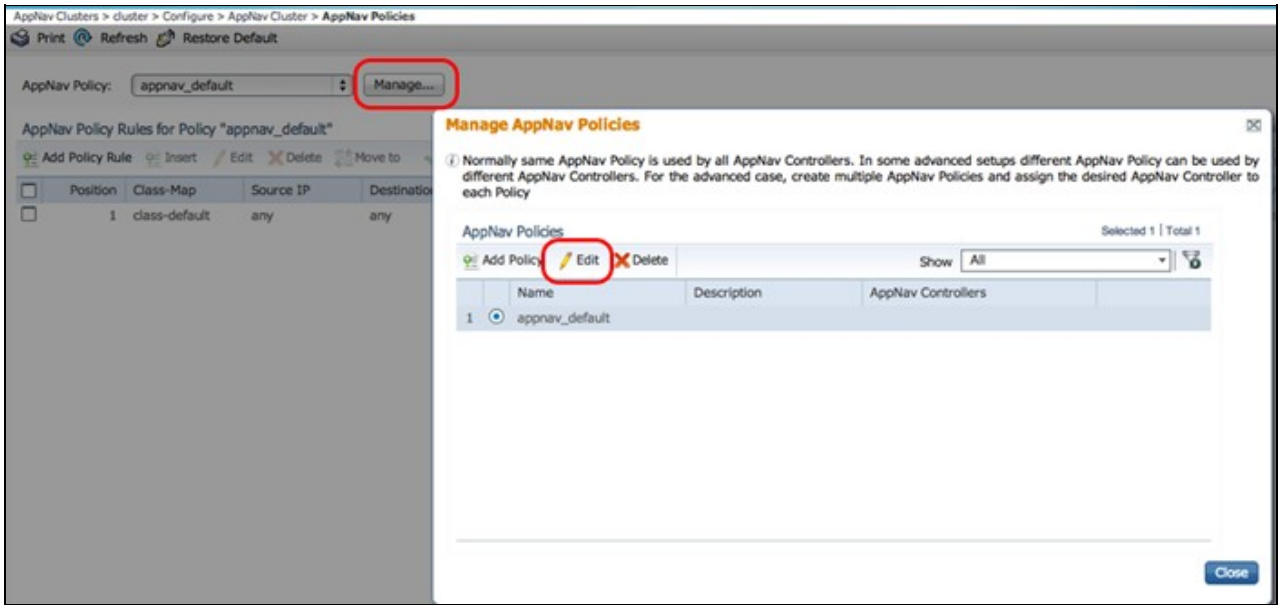


If an ANC has a gray status light, it might be disabled. Check that all ANCs are enabled by clicking the AppNav Controllers tab below the topology diagram. If an ANC is not enabled, its Enabled status is No. You can click the **Enable** taskbar icon to enable an ANC.

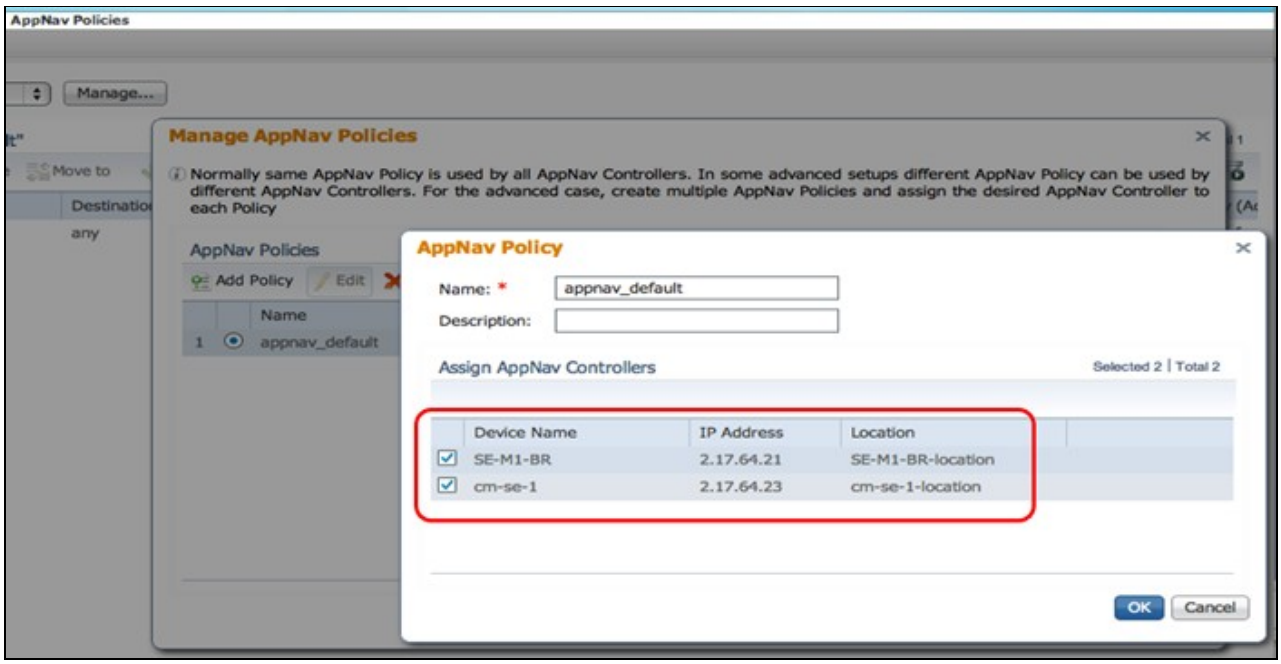
Check the AppNav policy on each ANC that has anything other than a green status light. If you hover your cursor over the status light on a device, a tool tip tells you the status or problem, if one is detected.



To check the defined policies, from the Central Manager menu, choose **Configure > AppNav Policies** and then click the **Manage** button.

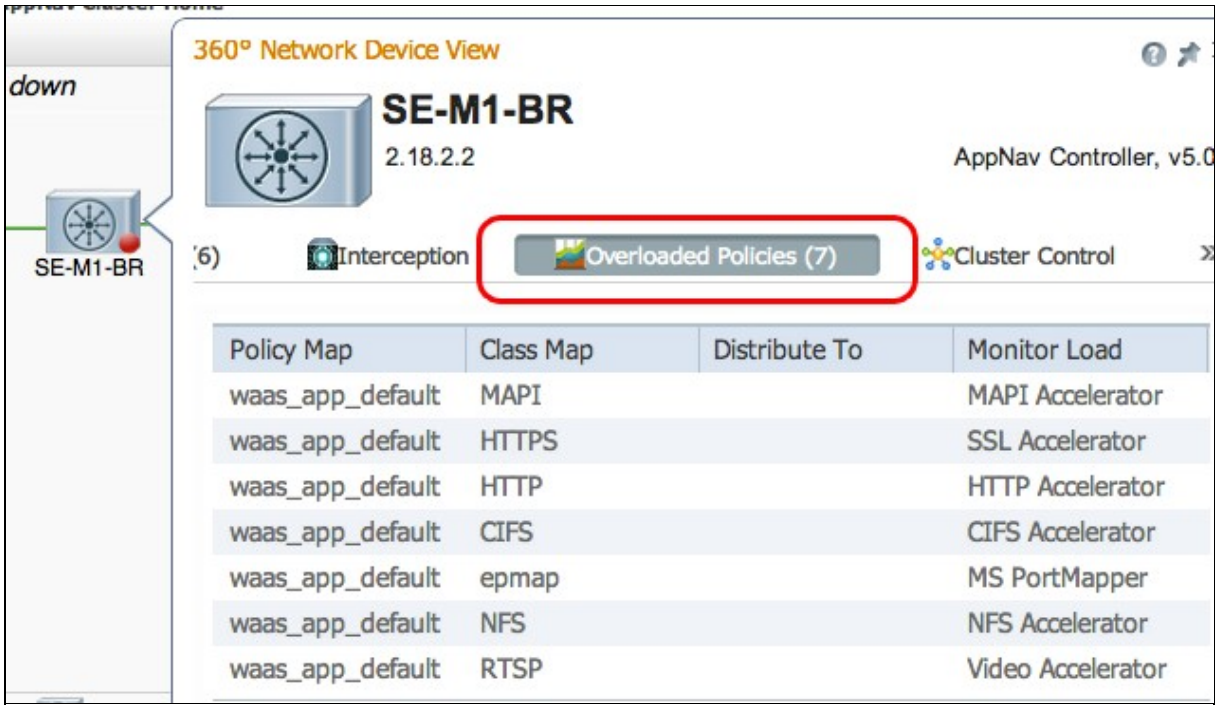


There should generally be a single policy assigned to all ANCs in the cluster. The default policy is named appnav_default. Select the radio button next to a policy and click the **Edit** taskbar icon. The AppNav Policy pane shows you the ANCs to which the selected policy is applied. If all ANCs are not shown with a checkmark, click the checkbox next to each unchecked ANC to assign the policy to it. Click **OK** to save the changes.

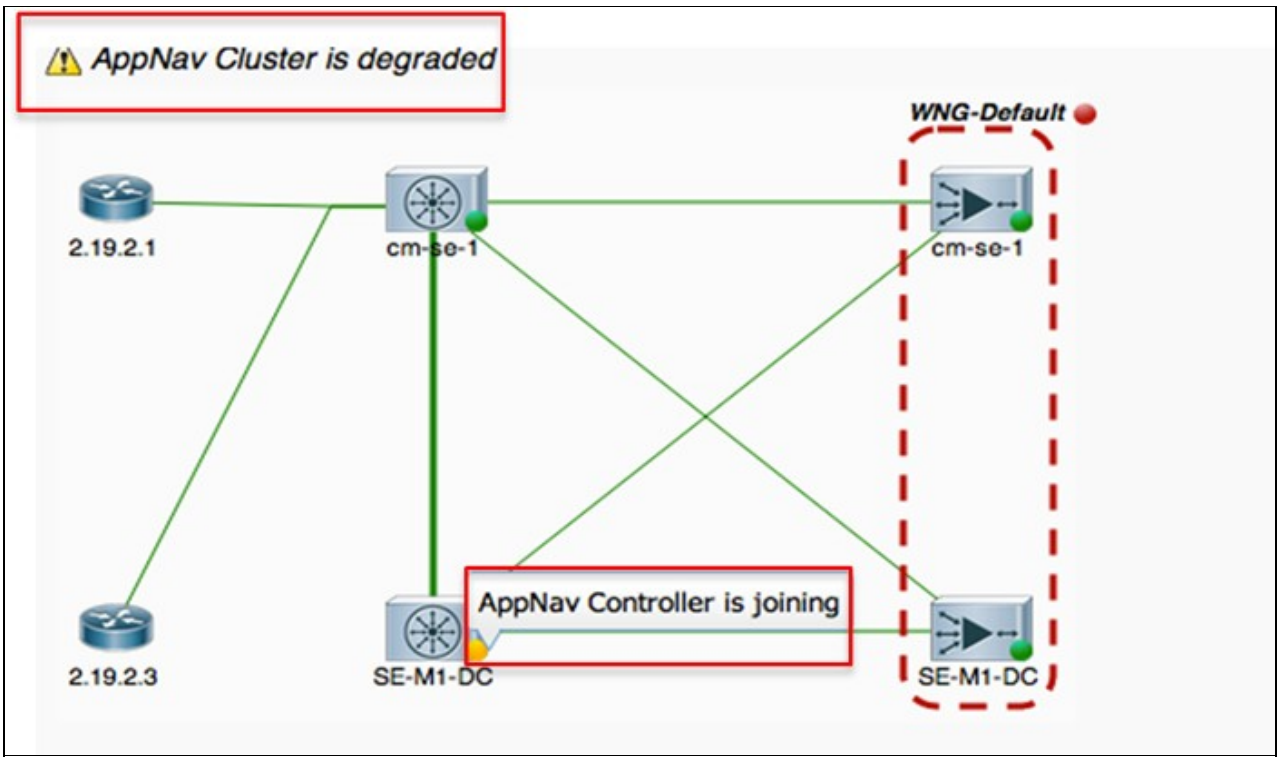


After verifying the policy assignments, you can verify the policy rules in the AppNav Policies page that remains displayed. Select any policy rule and click the **Edit** taskbar icon to change its definition.

An ANC could have a yellow or red status light if one or more policies are overloaded. Check the Overloaded Policies tab of the 360 degree device view to see a list of monitored policies that are overloaded.

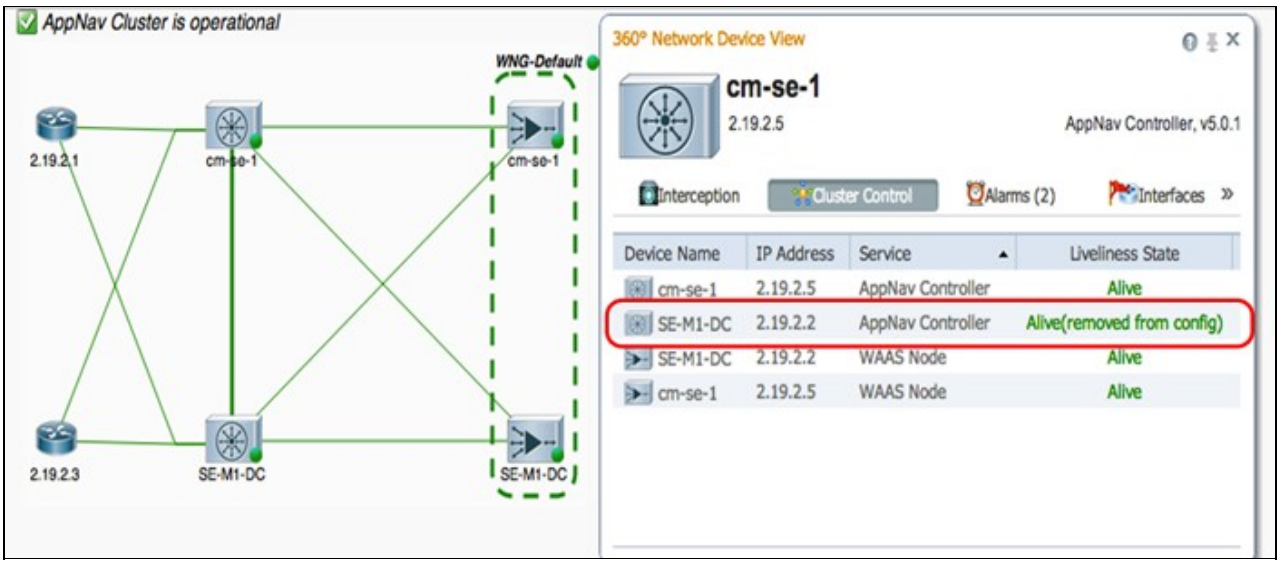


If an ANC is joining the cluster, it is shown with a yellow status light and joining status.



The Interception tab of the 360 degree device view shows that the interception path is down due to the joining state. Interception is held down until the ANC has synchronized its flow tables with the other ANCs and is ready to accept traffic. This process typically takes no more than two minutes.

If you remove an ANC from a cluster, it is still shown for a few minutes in the topology diagram and as alive in the Cluster Control tab, until all ANCs agree on the new cluster topology. It does not receive any new flows in this state.



AppNav CLI Commands for Monitoring Cluster and Device Status

Several CLI commands are useful for troubleshooting on an ANC:

- **show run service-insertion**
- **show service-insertion service-context**
- **show service-insertion appnav-controller-group**
- **show service-insertion service-node-group all**
- **show service-insertion appnav-controller** *ip-address*
- **show service-insertion service-node** [*ip-address*]
- **show service-insertion service-node-group** *group-name*

Use these commands on a WN:

- **show run service-insertion**
- **show service-insertion service-node**

You can use the **show service-insertion service-context** command on an ANC to see the service context status and stable view of the devices in the cluster:

```
ANC# show service-insertion service-context
Service Context                                : test
Service Policy                                 : appnav_default              <<< Active AppNav poli
Cluster protocol ICIMP version                 : 1.1
Cluster protocol DMP version                   : 1.1
Time Service Context was enabled               : Wed Jul 11 02:05:23 2012
Current FSM state                              : Operational                 <<< Service context st
Time FSM entered current state                 : Wed Jul 11 02:05:55 2012
Last FSM state                                 : Converging
Time FSM entered last state                    : Wed Jul 11 02:05:45 2012
Joining state                                  : Not Configured
Time joining state entered                     : Wed Jul 11 02:05:23 2012
Cluster Operational State                      : Operational                 <<< Status of this ANC
Interception Readiness State                   : Ready
Device Interception State                      : Not Shutdown                <<< Interception is no

Stable AC View:                                                              <<< Stable view of con
       10.1.1.1         10.1.1.2
Stable SN View:                                                              <<< Stable view of con
       10.1.1.1         10.1.1.2
```

```
Current AC View:
       10.1.1.1         10.1.1.2
Current SN View:
       10.1.1.1         10.1.1.2         10.1.1.3
```

If the Device Interception State field (above) shows Shutdown, it means that the CMM has shut down interception due to this ANC not being ready to receive traffic flows. For example, the ANC could still be in the joining process and the cluster has not yet synchronized flows.

The Stable View fields (above) list the IP addresses of the ANCs and WNs seen by this ANC device in its last converged view of the cluster. This is the view used for distribution operations. The Current View fields list the devices advertised by this ANC in its heartbeat messages.

You can use the **show service-insertion appnav-controller-group** command on an ANC to see the status of each ANC in the ANC group:

```
ANC# show service-insertion appnav-controller-group
All AppNav Controller Groups in Service Context
Service Context                              : test
Service Context configured state             : Enabled

AppNav Controller Group : scg
Member AppNav Controller count : 2
   Members:
       10.1.1.1         10.1.1.2

AppNav Controller                            : 10.1.1.1
AppNav Controller ID                         : 1
Current status of AppNav Controller          : Alive                     <<< Status of this ANC
Time current status was reached              : Wed Jul 11 02:05:23 2012
Joining status of AppNav Controller          : Joined                    <<< Joining means ANC is
Secondary IP address                         : 10.1.1.1                  <<< Source IP used in cl
Cluster protocol ICIMP version               : 1.1
Cluster protocol Incarnation Number          : 2
Cluster protocol Last Sent Sequence Number    : 0
Cluster protocol Last Received Sequence Number: 0

Current AC View of AppNav Controller:                                     <<< ANC and WN devices
       10.1.1.1         10.1.1.2
Current SN View of AppNav Controller:
       10.1.1.1         10.1.1.2

AppNav Controller                            : 10.1.1.2 (local)           <<< local indicates thi
AppNav Controller ID                         : 1
Current status of AppNav Controller          : Alive
Time current status was reached              : Wed Jul 11 02:05:23 2012
Joining status of AppNav Controller          : Joined
Secondary IP address                         : 10.1.1.2
Cluster protocol ICIMP version               : 1.1
Cluster protocol Incarnation Number          : 2
Cluster protocol Last Sent Sequence Number    : 0
Cluster protocol Last Received Sequence Number: 0

Current AC View of AppNav Controller:                                     <<< ANC and WN devices
       10.1.1.1         10.1.1.2
Current SN View of AppNav Controller:
       10.1.1.1         10.1.1.2         10.1.1.3
```

For a list of possible ANC statuses and joining statuses, see the **show service-insertion** command in the *Cisco Wide Area Application Services Command Reference*.

You can use the **show service-insertion service-node** command on an ANC to see the status of a particular WN in the cluster:

```
ANC# show service-insertion service-node 10.1.1.2
Service Node:                                : 20.1.1.2
Service Node belongs to SNG                  : sng2
Service Context                              : test
Service Context configured state             : Enabled

Service Node ID                              : 1
Current status of Service Node               : Alive                      <<< WN is visible
Time current status was reached              : Sun May  6 11:58:11 2011
Cluster protocol DMP version                 : 1.1
Cluster protocol incarnation number          : 1
Cluster protocol last sent sequence number   : 1692060441
Cluster protocol last received sequence number: 1441393061

AO state
--------
AO               State            For
--               -----            ---
tfo             GREEN            3d 22h 11m 17s                           <<< Overall/TFO state r
epm             GREEN            3d 22h 11m 17s                           <<< AO states reported
cifs            GREEN            3d 22h 11m 17s
mapi            GREEN            3d 22h 11m 17s
http            RED              3d 22h 14m 3s
video           RED              11d 2h 2m 54s
nfs             GREEN            3d 22h 11m 17s
ssl             YELLOW           3d 22h 11m 17s
ica             GREEN            3d 22h 11m 17s
```

You can use the **show service-insertion service-node-group** command on an ANC to see the status of a particular WNG in the cluster:

```
ANC# show service-insertion service-node-group sng2
Service Node Group name   : sng2
Service Context             : scxt1
       Member Service Node count   : 1
       Members:
       10.1.1.1       10.1.1.2

Service Node:                                : 10.1.1.1
Service Node belongs to SNG                  : sng2
Current status of Service Node               : Excluded                    <<< WN status
Time current status was reached              : Sun Nov  6 11:58:11 2011
Cluster protocol DMP version                 : 1.1
Cluster protocol incarnation number          : 1
Cluster protocol last sent sequence number   : 1692061851
Cluster protocol last received sequence number: 1441394001

AO state
--------
AO               State            For
--               -----            ---
tfo             GREEN            3d 22h 12m 52s
epm             GREEN            3d 22h 12m 52s
cifs            GREEN            3d 22h 12m 52s
mapi            GREEN            3d 22h 12m 52s
http            RED              3d 22h 15m 38s
video           RED              11d 2h 4m 29s
nfs             GREEN            3d 22h 12m 52s
ssl             YELLOW           3d 22h 12m 52s
```

```
ica              GREEN            3d 22h 12m 52s

Service Node:                                 : 10.1.1.2
Service Node belongs to WNG                   : sng2
Current status of Service Node                : Alive                      <<< WN status
Time current status was reached               : Sun Nov  6 11:58:11 2011
Cluster protocol DMP version                  : 1.1
Cluster protocol incarnation number           : 1
Cluster protocol last sent sequence number    : 1692061851
Cluster protocol last received sequence number: 1441394001

AO state
--------
AO               State            For
--               -----            ---
tfo              GREEN            3d 22h 12m 52s
epm              GREEN            3d 22h 12m 52s
cifs             GREEN            3d 22h 12m 52s
mapi             GREEN            3d 22h 12m 52s
http             RED              3d 22h 15m 38s
video            RED              11d 2h 4m 29s
nfs              GREEN            3d 22h 12m 52s
ssl              YELLOW           3d 22h 12m 52s
ica              GREEN            3d 22h 12m 52s

SNG Availability per AO                                          <<< AO status for entire WNG
-----------------------
AO               Available        Since
--               ---------        -----
tfo              Yes              3d 22h 12m 52s
epm              Yes              3d 22h 12m 52s
cifs             Yes              3d 22h 12m 52s
mapi             Yes              3d 22h 12m 52s
http             No               3d 22h 15m 38s
video            No               11d 2h 4m 29s
nfs              Yes              3d 22h 12m 52s
ssl              No               11d 2h 4m 29s
ica              Yes              3d 22h 12m 52s
```

The first WN in the example above has a status of Excluded, which means that the WN is visible to the ANC but it is excluded from the cluster because one or more other ANCs cannot see it.

The SNG Availability per AO table shows if each AO is able to service new connections. An AO is available if at least one WN in the WNG has a GREEN status for the AO.

You can use the **show service-insertion service-node** command on a WN to see the status of the WN:

```
WAE# show service-insertion service-node
Cluster protocol DMP version      : 1.1
Service started at                : Wed Jul 11 02:05:45 2012
Current FSM state                 : Operational                    <<< WN is responding to health
Time FSM entered current state    : Wed Jul 11 02:05:45 2012
Last FSM state                    : Admin Disabled
Time FSM entered last state       : Mon Jul  2 17:19:15 2012
Shutdown max wait time:
               Configured        : 120
               Operational       : 120

Last 8 AppNav Controllers
-------------------------
AC IP             My IP            DMP Version  Incarnation  Sequence        Tim
```

```
e Last Heard
-----             -----            -----------  -----------  --------        ---
------------

Reported state                                                          <<< TFO and AO reported states
--------------
Accl             State    For              Reason
----             -----    ---              ------
TFO (System)     GREEN    43d 7h 45m 8s
EPM              GREEN    43d 7h 44m 40s
CIFS             GREEN    43d 7h 44m 41s
MAPI             GREEN    43d 7h 44m 43s
HTTP             GREEN    43d 7h 44m 45s
VIDEO            GREEN    43d 7h 44m 41s
NFS              GREEN    43d 7h 44m 44s
SSL              RED      43d 7h 44m 21s
ICA              GREEN    43d 7h 44m 40s

Monitored state of Accelerators                                         <<< TFO and AO actual states
-------------------------------
TFO (System)
       Current State: GREEN
       Time in current state: 43d 7h 45m 8s
EPM
       Current State: GREEN
       Time in current state: 43d 7h 44m 40s
CIFS
       Current State: GREEN
       Time in current state: 43d 7h 44m 41s
MAPI
       Current State: GREEN
       Time in current state: 43d 7h 44m 43s
HTTP
       Current State: GREEN
       Time in current state: 43d 7h 44m 45s
VIDEO
       Current State: GREEN
       Time in current state: 43d 7h 44m 41s
NFS
       Current State: GREEN
       Time in current state: 43d 7h 44m 44s
SSL
       Current State: RED
       Time in current state: 43d 7h 44m 21s
       Reason:
       AO is not configured
ICA
       Current State: GREEN
       Time in current state: 43d 7h 44m 40s
```

The monitored state of an accelerator is its actual state but the reported state can differ because it is the lower of the system state or the accelerator state.

For more information about troubleshooting optimization on a WN, see the Troubleshooting Optimization and Troubleshooting Application Acceleration articles.

AppNav CLI Commands for Monitoring Flow Distribution Statistics

Several CLI commands are useful for troubleshooting policies and flow distribution on an ANC:

- **show policy-map type appnav** *policymap-name* -- Shows the policy rules and hit counts for each class in the policy map.
- **show class-map type appnav** *class-name* -- Shows the match criteria and hit counts for each match condition in the class map.
- **show policy-sub-class type appnav** *level1-class-name level2-class-name* -- Shows the match criteria and hit counts for each match condition in a class map in a nested AppNav policy map.
- **show statistics class-map type appnav** *class-name* -- Shows traffic interception and distribution statistics for a class map.
- **show statistics policy-sub-class type appnav** *level1-class-name level2-class-name* -- Shows traffic interception and distribution statistics for a class map in a nested AppNav policy map.
- **show statistics pass-through type appnav** -- Shows AppNav traffic statistics for each pass-through reason.
- **show appnav-controller flow-distribution** -- Shows how a specific hypothetical flow would be classified and distributed by an ANC, based on the defined policy and dynamic load conditions. This command can be useful to verify how a particular flow will be handled on an ANC and what class it belongs to.

Use these commands on a WN to troubleshoot flow distribution:

- **show statistics service-insertion service-node** *ip-address* -- Shows statistics for accelerators and traffic distributed to the WN.
- **show statistics service-insertion service-node-group name** *group-name* -- Shows statistics for accelerators and traffic distributed to the WNG.

You can use the **show statistics class-map type appnav** *class-name* command on an ANC to troubleshoot flow distribution, for example to determine why traffic might be slow for a particular class. This could be an application class map such as HTTP or, if all traffic to a branch seems slow, it could be a branch affinity class map. Here is an example for the HTTP class:

```
ANC# show statistics class-map type appnav HTTP
Class Map                        From Network to SN   From SN to Network
                                -------------------   ------------------
HTTP
 Redirected Client->Server:
   Bytes                                   3478104             11588180
   Packets                                   42861               102853
 Redirected Server->Client:
   Bytes                                1154109763              9842597
   Packets                                  790497                60070

Connections
-----------
   Intercepted by ANC                                          4        <<< Are connections being
   Passed through by ANC                                       0        <<< Passed-through connect
   Redirected by ANC                                           4        <<< Are connections being
   Accepted by SN                                              4        <<< Connections accepted b
   Passed through by SN (on-Syn)                               0        <<< Connections might be p
   Passed through by SN (post-Syn)                             0        <<< Connections might be p


Passthrough Reasons                      Packets            Bytes        <<< Why is ANC passing thr
-------------------                      -------            -----
Collected by ANC:
  PT Flow Learn Failure                        0                0       <<< Asymmetric connection;
  PT Cluster Degraded                          0                0       <<< ANCs cannot communicat
  PT SNG Overload                              0                0       <<< All WNs in the WNG are
  PT AppNav Policy                             0                0       <<< Connection policy is p
  PT Unknown                                   0                0       <<< Unknown passthrough
```

```
Indicated by SN:                                                    <<< Why are WNs passing th
  PT No Peer                                  0                0    <<< List of WN pass-throug
  ...
```

The WN pass-through reasons in the Indicated by SN section increment only if pass-through offload is configured on a WN. Otherwise, the ANC does not know that the WN is passing through a connection and does not count it.

If the Connections: Intercepted by ANC counter is not incrementing, there is an interception problem. You can use the WAAS TcpTraceroute utility to troubleshoot the placement of the ANC in the network, find asymmetric paths, and determine the policy applied to a connection. For details, see the section Connection Tracing.

AppNav CLI Commands for Debugging Connections

To debug an individual connection or set of connections on an ANC, you can use the **show statistics appnav-controller connection** command to display the active connection list.

```
anc# show statistics appnav-controller connection
Collecting Records. Please wait...
Optimized Flows:
-------------------------
Client                  Server                  SN-IP                   AC Owned

2.30.5.10:38111         2.30.1.10:5004          2.30.1.21               Yes
2.30.5.10:38068         2.30.1.10:5003          2.30.1.21               Yes
2.30.5.10:59861         2.30.1.10:445           2.30.1.21               Yes
2.30.5.10:59860         2.30.1.10:445           2.30.1.21               Yes
2.30.5.10:43992         2.30.1.10:5001          2.30.1.5                Yes
2.30.5.10:59859         2.30.1.10:445           2.30.1.21               Yes
2.30.5.10:59858         2.30.1.10:445           2.30.1.21               Yes
2.30.5.10:59857         2.30.1.10:445           2.30.1.21               Yes
2.30.5.10:59856         2.30.1.10:445           2.30.1.21               Yes

Passthrough Flows:
-------------------------
Client                  Server                  Passthrough Reason
2.30.5.10:41911         2.30.1.10:5002          PT Flowswitch Policy
```

You can filter the list by specifying the client or server IP address and/or port options and you can show detailed statistics about connections by specifying the **detail** keyword.

```
anc# show statistics appnav-controller connection server-ip 2.30.1.10 detail
Collecting Records. Please wait...

Optimized Flows
-------------------------
Client: 2.30.5.10:55330
Server: 2.30.1.10:5001
AppNav Controller Owned: Yes        <<< This ANC is seeing activity on this connection
Service Node IP:2.30.1.5            <<< Connection is distributed to this SN
Classifier Name: se_policy:p5001    <<< Name of matched class map
Flow association: 2T:No,3T:No       <<< Connection is associated with dynamic app or session (MAPI
Application-ID: 0                   <<< AO that is optimizing the connection
Peer-ID: 00:14:5e:84:41:31          <<< ID of the optimizing peer

Client: 2.30.5.10:55331
Server: 2.30.1.10:5001
AppNav Controller Owned: Yes
```

```
Service Node IP:2.30.1.5
Classifier Name: se_policy:p5001
Flow association: 2T:No,3T:No
Application-ID: 0
Peer-ID: 00:14:5e:84:41:31
...
```

You can specify the summary option to display the number of active distributed and pass-through connections.

```
anc# show statistics appnav-controller connection summary
Number of optimized flows    = 2
Number of pass-through flows = 17
```

Connection Tracing

To assist in troubleshooting AppNav flows, you can use the Connection Trace tool in the Central Manager. This tool shows the following information for a particular connection:

- If the connection was passed through or distributed to a WNG
- Pass-through reason, if applicable
- The WNG and WN to which the connection was distributed
- Accelerator monitored for the connection
- Class-map applied

To use the Connection Trace tool, follow these steps:

1. From the Central Manager menu, choose **AppNav Clusters >** *cluster-name*, then choose **Monitor > Tools > Connection Trace**.

2. Choose the ANC, the peer WAAS device, and specify connection match criteria.

3. Click **Trace** to display matching connections.

The WAAS TCP Traceroute is another tool not specific to AppNav that can help you troubleshoot network and connection issues, including asymmetric paths. You can use it to find a list of WAAS nodes between the client and server, and the configured and applied optimization policies for a connection. From the Central Manager, you can choose any device in your WAAS network from which to run the traceroute. To use the WAAS Central Manager TCP Traceroute tool, follow these steps:

1. From the WAAS Central Manager menu, choose **Monitor > Troubleshoot > WAAS Tcptraceroute**. Alternatively, you can choose a device first and then choose this menu item to run the traceroute from that device.

2. From the WAAS Node drop-down list, choose a WAAS device from which to run the traceroute. (This item does not appear if you are in the device context.)

3. In the Destination IP and Destination Port fields, enter the IP address and port of the destination to which you want to run the traceroute

4. Click **Run TCPTraceroute** to display the results.

WAAS nodes in the traced path are displayed in the table below the fields. You can also run this utility from the CLI with the **waas-tcptrace** command.

AppNav Debug Logging

The following log file is available for troubleshooting AppNav cluster manager issues:

- Debug log files: /local1/errorlog/cmm-errorlog.current (and cmm-errorlog.*)

To set up and enable debug logging of the AppNav cluster manager, use the following commands.

NOTE: Debug logging is CPU intensive and can generate a large amount of output. Use it judiciously and sparingly in a production environment.

You can enable detailed logging to the disk:

```
WAE(config)# logging disk enable
WAE(config)# logging disk priority detail
```

The options for cluster manager debugging (on 5.0.1 and later) are as follows:

```
WAE# debug cmm ?
 all      enable all CMM debugs
 cli      enable CMM cli debugs
 events   enable CMM state machine events debugs
 ipc      enable CMM ipc messages debugs
 misc     enable CMM misc debugs
 packets  enable CMM packet debugs
 shell    enable CMM infra debugs
 timers   enable CMM state machine timers debugs
```

You can enable debug logging for the cluster manager and then display the end of the debug error log as follows:

```
WAE# debug cmm all
WAE# type-tail errorlog/cmm-errorlog.current follow
```

You can also enable debug logging for the flow distribution manager (FDM) or the flow distribution agent (FDA) with these commands:

```
WAE# debug fdm all
WAE# debug fda all
```

The FDM determines where to distribute flows based on the policy and dynamic load conditions of the WNs. The FDA collects WN load information. The following log files are available for troubleshooting FDM and FDA issues:

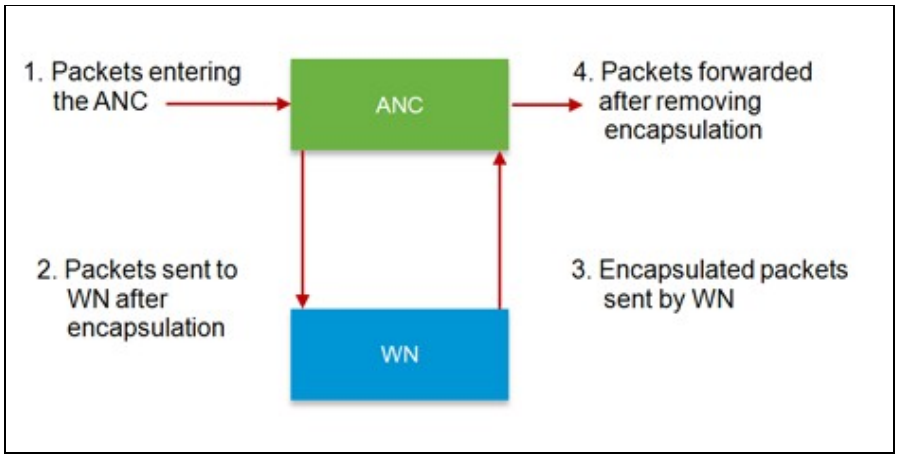
- Debug log files: /local1/errorlog/fdm-errorlog.current (and fdm-errorlog.*)
- Debug log files: /local1/errorlog/fda-errorlog.current (and fda-errorlog.*)

AppNav Packet Capture

A new **packet-capture** command is introduced to allow capturing data packets on interfaces on the Cisco AppNav Controller Interface Module. This command can also capture packets on other interfaces, and can decode packet capture files. The **packet-capture** command is preferred over the deprecated commands **tcpdump** and **tethereal**, which cannot capture packets on the Cisco AppNav Controller Interface Module. See the *Cisco Wide Area Application Services Command Reference* for details on command syntax.

Note: Either packet capture or debug capture can be active, but not both simultaneously.

Data packets sent between ANCs and WNs are encapsulated, as shown in the following diagram.



If you capture packets at points 1 or 4 in the diagram, they are unencapsulated. If you capture packets at points 2 or 3, they are encapsulated.

Here is sample output for an encapsulated packet capture:

```
anc# packet-capture appnav-controller interface GigabitEthernet 1/0 access-list all
Packet-Capture: Setting virtual memory/file size limit to 419430400
Running as user "admin" and group "root". This could be dangerous.
Capturing on eth14
 0.000000    2.58.2.11 -> 2.1.6.122    TCP https > 2869 [ACK] Seq=1 Ack=1 Win=65535 Len=0
 4.606723   2.58.2.175 -> 2.43.64.21   TELNET Telnet Data ...
...
 37.679587    2.58.2.40 -> 2.58.2.35    GRE Encapsulated 0x8921 (unknown)
 37.679786    2.58.2.35 -> 2.58.2.40    GRE Encapsulated 0x8921 (unknown)
```

Here is sample output for an unencapsulated packet capture:

```
anc# packet-capture appnav-controller access-list all non-encapsulated
Packet-Capture: Setting virtual memory/file size limit to 419430400
Running as user "admin" and group "root". This could be dangerous.
Capturing on eth14
0.751567   2.58.2.175 -> 2.43.64.21   TELNET Telnet Data ...
1.118363   2.58.2.175 -> 2.43.64.21   TELNET Telnet Data ...
1.868756   2.58.2.175 -> 2.43.64.21   TELNET Telnet Data ...
...
```

Packet capture guidelines:

- A packet-capture ACL is always applied to inner IP packet for WCCP-GRE and SIA encapsulated packets.
- Packet capture is done on all ANC interfaces if the ANC interface for the packet capture is not provided.

Here is sample output for a packet capture on a WN interface:

```
anc# packet-capture interface GigabitEthernet 0/0 access-list 10
Packet-Capture: Setting virtual memory/file size limit to 419430400
Running as user "admin" and group "root". This could be dangerous.
Capturing on eth0
```

```
  0.000000      2.1.8.4 -> 2.64.0.6     TELNET Telnet Data ...
  0.000049     2.64.0.6 -> 2.1.8.4      TELNET Telnet Data ...
  0.198908      2.1.8.4 -> 2.64.0.6     TCP 18449 > telnet [ACK] Seq=2 Ack=2 Win=3967 Len=0
  0.234129      2.1.8.4 -> 2.64.0.6     TELNET Telnet Data ...
  0.234209     2.64.0.6 -> 2.1.8.4      TELNET Telnet Data ...
```

Here is an example of decoding a packet capture file:

```
anc# packet-capture decode /local1/se_flow_add.cap
Running as user "admin" and group "root". This could be dangerous.  1   0.000000
     100.1.1.2 -> 100.1.1.1    GRE Encapsulated SWIRE  2   0.127376
     100.1.1.2 -> 100.1.1.1    GRE Encapsulated SWIRE
```

You can specify a src-ip/dst-ip/src-port/dst-port for filtering the packets:

anc# **packet-capture decode source-ip 2.64.0.33 /local1/hari_pod_se_flow.cap**

```
Running as user "admin" and group "root". This could be dangerous.
3   0.002161    2.64.0.33 -> 2.64.0.17    TCP 5001 > 33165 [SYN, ACK] Seq=0 Ack=1
Win=5792 Len=0 MSS=1460 TSV=326296092 TSER=326296080 WS=4
4   0.002360    2.64.0.33 -> 2.64.0.17    TCP 5001 > 33165 [SYN, ACK] Seq=0 Ack=1
Win=5792 Len=0 MSS=1406 TSV=326296092 TSER=326296080 WS=4
```