High Availability Overview

The goal of any high availability solution is to ensure that all network services are resilient to failure. Such a solution aims to provide continuous access to network resources by addressing the potential causes of downtime through functionality, design, and best practices. The core of the Viptela high availability solution is achieved through a combination of three factors:

- Functional hardware device redundancy. The basic strategy consists of installing and provisioning redundant hardware devices and redundant components on the hardware. These devices are connected by a secure control plane mesh of Datagram Transport Layer Security (DTLS) connections among themselves, which allows for rapid failover should a device fail or otherwise become unavailable. A key feature of the Viptela control plane is that it is established and maintained automatically, by the Viptela devices and software themselves. For more information, see Bringup Sequence of Events.

- Robust network design.

- Software mechanisms ensure rapid recovery from a failure. To provide a resilient control plane, the Viptela Overlay Management Protocol (OMP) regularly monitors the status of all Viptela devices in the network and automatically adjusts to changes in the topology as devices join and leave the network. For data plane resiliency, the Viptela software implements standard protocol mechanisms, specifically Bidirectional Forwarding Detection (BFD), which runs on the secure IPsec tunnels between vEdge routers.

Recovery from a failure is a function of the time it takes to detect the failure and then repair or recover from it. The Viptela solution provides the ability to control the amount of time to detect a failure in the network. In most cases, repair of the failure is fairly instantaneous.

Hardware Support of High Availability

A standard best practice in any network setup is to install redundant hardware at all levels, including duplicate parallel routers and other systems, redundant fans, power supplies and other hardware components within these devices, and backup network connections. Providing high availability in the Viptela solution is no different. A network design that is resilient in the face of hardware failure should include redundant vBond orchestrators, vSmart controllers, and vEdge routers and any available redundant hardware components.

Recovery from the total failure of a hardware component in the Viptela overlay network happens in basically the same way as in any other network. A backup component has been preconfigured, and it is able to perform all necessary functions by itself.

Robust Network Design

In addition to simple duplication of hardware components, the high availability of a Viptela network can be enhanced by following best practices to design a network that is robust in the face of failure. In one such network design, redundant components are spread around the network as much as possible. Design practices include situating redundant vBond orchestrators and vSmart controllers at dispersed geographical locations and connecting them to different transport...
networks. Similarly, the vEdge routes at a local site can connect to different transport networks and can reach these networks through different NATs and DMZs.

**Software Support of High Availability**

The Viptela software support for high availability and resiliency in the face of failure is provided both in the control plane, using the standard DTLS protocol and the proprietary Viptela Overlay Management Protocol (OMP), and in the data plane, using the industry-standard protocols BFD, BGP, OSPF, and VRRP.

**Control Plane Software Support of High Availability**

The Viptela control plane operates in conjunction with redundant components to ensure that the overlay network remains resilient if one of the components fails. The control plane is built on top of DTLS connections between the Viptela devices, and it is monitored by the Viptela OMP protocol, which establishes peering sessions (similar to BGP peering sessions) between pairs of vSmart controllers and vEdge routers, and between pairs of vSmart controllers. These peering sessions allow OMP to monitor the status of the Viptela devices and to share the information among them so that each device in the network has a consistent view of the overlay network. The exchange of control plane information over OMP peering sessions is a key piece in the Viptela high availability solution:

- vSmart controllers quickly and automatically learn when a vBond orchestrator or a vEdge router joins or leaves the network. They can then rapidly make the necessary modifications in the route information that they send to the vEdge routers.
- vBond orchestrators quickly and automatically learn when a device joins the network and when a vSmart controller leaves the network. They can then rapidly make the necessary changes to the list of vSmart controller IP addresses that they send to vEdge routers joining the network.
- vBond orchestrators learn when a domain has multiple vSmart controllers and can then provide multiple vSmart controller addresses to vEdge routers joining the network.
- vSmart controllers learn about the presence of other vSmart controllers, and they all automatically synchronize their route tables. If one vSmart controller fails, the remaining systems take over management of the control plane, simply and automatically, and all vEdge routers in the network continue to receive current, consistent routing and TLOC updates from the remaining vSmart controllers.

Let's look at the redundancy provided by each of the Viptela hardware devices in support of network high availability.

**vBond Orchestrator Redundancy**

The vBond orchestrator performs two key functions in the Viptela overlay network:

- Authenticates and validates all vSmart controllers and vEdge routers that attempt to join the Viptela network.
- Orchestrates the control plane connections between the vSmart controllers and the vEdge routers, thus enabling vSmart controllers and vEdge routers to connect to each other in the Viptela network.
The vBond orchestrator runs as a VM on a network server. The vBond orchestrator can also run on a vEdge router that is configured to be a vBond orchestrator, however this is not recommended, and it limits the number of vEdge router control connections to 50. If using running the vBond daemon on a vEdge router, note that only one vBond daemon can run at a time on a vEdge router, so to provide redundancy and high availability, the network must have two or more vEdge routers that function as vBond orchestrators. (Note also that it is not recommended to use a vEdge router acting as a vBond orchestrator as a regular edge router.)

Having multiple vBond orchestrators ensures that one of them will be always be available whenever a Viptela device—a vEdge router or a vSmart controller—is attempting to join the network.

**Configuration of Redundant vBond Orchestrators**

A vEdge router learns that it is acting as a vBond orchestrator from its configuration. In the `system vbond` configuration command, which defines the IP address (or addresses) of the vBond orchestrator (or orchestrators) in the Viptela overlay network, you include the `local` option. In this command, you also include the local public IP address of the vBond orchestrator. (Even though on vEdge routers and vSmart controllers you can specify a vBond orchestrator's IP address as a DNS name, on the vBond orchestrator itself you must specify it as an IP address.)

On vSmart controllers and vEdge routers, when the network has only a single vBond orchestrator, you can configure the location of the vBond system either as an IP address or as the name of a DNS server (such as vbond.viptela.com). (Again, you configure this in the `system vbond` command.) When the network has two or more vBond orchestrators and they must all be reachable, you should use the name of a DNS server. The DNS server then resolves the name to a single IP address that the vBond orchestrator returns to the vEdge router. If the DNS name resolves to multiple IP addresses, the vBond orchestrator returns them all to the vEdge router, and the router tries each address sequentially until it forms a successful connection.

Note that even if your Viptela network has only a single vBond orchestrator, it is recommended as a best practice that you specify a DNS name rather than an IP address in the `system vbond` configuration command, because this results in a scalable configuration. Then, if you add additional vBond orchestrators to your network, you do not need to change the configurations on any of the vEdge routers or vSmart controllers in your network.

**vManage NMS Redundancy**

The vManage NMSs comprise a centralized network management system that enables configuration and management of the Viptela devices in the overlay network. It also provides a real-time dashboard into the status of the network and network devices. The vManage NMSs maintain permanent communication channels with all Viptela devices in the network. Over these channels, the vManage NMSs push out files that list the serial numbers of all valid devices, they push out each device's configuration, and they push out new software images as part of a software upgrade process. From each network device, the vManage NMSs receive various status information that is displayed on the vManage Dashboard and other screens.

A highly available Viptela network contains three or more vManage NMSs in each domain. This scenario is referred to as a *cluster* of vManage NMSs, and each vManage NMS in a cluster is referred to as a *vManage instance*. A vManage cluster consists of the following architectural components:

- Application server—This provides a web server for user sessions. Through these sessions, a logged-in user can view a
high-level dashboard summary of networks events and status, and can drill down to view details of these events. A user can also manage network serial number files, certificates, software upgrades, device reboots, and configuration of the vManage cluster itself from the vManage application server.

- **Configuration database**—Stores the inventory and state and the configurations for all Viptela devices.
- **Network configuration system**—Provides the mechanism for pushing configurations from the vManage NMS to the Viptela devices and for retrieving the running configurations from these devices.
- **Statistics database**—Stores the statistics information collected from all Viptela devices in the overlay network.
- **Message bus**—Communication bus among the different vManage instances. This bus is used to share data and coordinate operations among the vManage instances in the cluster.

The following figure illustrates these architectural components. Also in this figure, you see that each vManage instance resides on a virtual machine (VM). The VM can have from one to eight cores, with a Viptela software process (vdaemon) running on each core. In addition, the VM stores the actual configuration for the vManage NMS itself.

The vManage cluster implements an active-active architecture in the following way:

- Each vManage instance in the cluster is an independent processing node.
• All vManage instances are active simultaneously.

• All user sessions to the application server are load-balanced, using either an internal vManage load balancer or an external load balancer.

• All control sessions between the vManage application servers and the vEdge routers are load-balanced. A single vManage instance can manage a maximum of about 2000 vEdge routers. However, all the controller sessions—the sessions between the vManage instances and the vSmart controllers, and the sessions between the vManage instances and the vBond orchestrators—are arranged in a full-mesh topology.

• The configuration and statistics databases can replicated across vManage instances, and these databases can be accessed and used by all the vManage instances.

• If one of the vManage instances in the cluster fails or otherwise becomes unavailable, the network management services that are provided by the vManage NMS are still fully available across the network.

The message bus among the vManage instances in the cluster allows all the instances to communicate using an out-of-band network. This design, which leverages a third vNIC on the vManage VM, avoids using WAN bandwidth for management traffic.

You configure the vManage cluster from the vManage web application server. See the Cluster Manage help article. During the configuration process, you can configure each vManage instance can run the following services:

• Application server—It is recommended that you configure each vManage instance to run an application web servers so that users are able to log in to each of the vManage systems.

• Configuration database—Within the vManage cluster, no more than three iterations of the configuration database can run.

• Load balancer—The vManage cluster requires a load balancer to distribute user login sessions among the vManage instances in the cluster. As mentioned above, a single vManage instance can manage a maximum of 2000 vEdge routers. It is recommended that you use an external load balancer. However, if you choose to use a vManage instance for this purpose, all HTTP and HTTPS traffic directed to its IP address is redirected to other instances in the cluster.

• Messaging server—It is recommended that you configure each vManage instance to run the message bus so that all the instances in the cluster can communicate with each other.

• Statistics database—Within the vManage cluster, no more than three iterations of the statistics database can run.

The following are the design considerations for a vManage cluster:

• A vManage cluster should consist of a minimum of three vManage instances.

• The application server and message bus should run on all vManage instances.

• Within a cluster, a maximum of three instances of the configuration database and three instances of the statistics database can run. Note, however, that any individual vManage instance can run both, one, or none of these two databases.
• To provide the greatest availability, it is recommended that you run the configuration and statistics databases on three vManage instances.

### vSmart Controller Redundancy

The vSmart controllers are the central orchestrators of the control plane. They have permanent communication channels with all the Viptela devices in the network. Over the DTLS connections between the vSmart controllers and vBond orchestrators and between pairs of vSmart controllers, the devices regularly exchange their views of the network, to ensure that their route tables remain synchronized. Over DTLS connections to vEdge routers, the vSmart controllers pass accurate and timely route information.

A highly available Viptela network contains two or more vSmart controllers in each domain. A Viptela domain can have up to 20 vSmart controllers, and each vEdge router, by default, connects to two of them. When the number of vSmart controllers in a domain is greater than the maximum number of controllers that a domain's vEdge routers are allowed to connect to, the Viptela software load-balances the connections among the available vSmart controllers.

While the configurations on all the vSmart controllers must be functionally similar, the control policies must be identical. This is required to ensure that, at any time, all vEdge routers receive consistent views of the network. If the control policies are not absolutely identical, different vSmart controllers might give different information to a vEdge router, and the likely result will be network connectivity issues.

To reiterate, the Viptela overlay network works properly only when the control policies on all vSmart controllers are identical. Even the slightest difference in the policies will result in issues with the functioning of the network.

To remain synchronized with each other, the vSmart controllers establish a full mesh of DTLS control connections, as well as a full mesh of OMP sessions, between themselves. Over the OMP sessions, the vSmart controllers advertise routes, TLOCs, services, policies, and encryption keys. It is this exchange of information that allows the vSmart controllers to remain synchronized.

You can place vSmart controllers anywhere in the network. For availability, it is highly recommended that the vSmart controllers be geographically dispersed.

Each vSmart controller establishes a permanent DTLS connection to each of the vBond orchestrators. These connections allow the vBond orchestrators to track which vSmart controllers are present and operational. So, if one of the vSmart controller fails, the vBond orchestrator will not provide the address of the unavailable vSmart controller to a vEdge router that is just joining the network.

### vEdge Router Redundancy

vEdge routers are commonly used in two ways in the Viptela network: to be the Viptela routers at a branch site, and to create a hub site that branch vEdge routers connect to.

A branch site can have two vEdge routers in a branch site for redundancy. Each of the router maintains:

- A secure control plane connection, via a DTLS connection, with each vSmart controller in its domain
- A secure data plane connection with the other vEdge routers at the site
Because both vEdge routers receive the same routing information from the vSmart controllers, each one is able to continue to route traffic if one should fail, even if they are connected to different transport providers.

When using vEdge routers in a hub site, you can provide redundancy by installing two vEdge routers. The branch vEdge routers need to connect to each of the hub vEdge routers, using separate DTLS connections.

You can also have vEdge routers provide redundancy by configuring up to tunnel interfaces on a single router. Each tunnel interface can go through the same of different firewalls, service providers, and network clouds, and each maintains a secure control plane connection, by means of a DTLS tunnel, with the vSmart controllers in its domain.

---

**Recovering from a Failure in the Control Plane**

The combination of hardware component redundancy with the architecture of the Viptela control plane results in a highly available network, one that continues to operate normally and without interruption when a failure occurs in one of the redundant control plane components. Recovery from the total failure of a vSmart controller, vBond orchestrator, or vEdge router in the Viptela overlay network happens in basically the same way as the recovery from the failure of a regular router or server on the network: A preconfigured backup component is able to perform all necessary functions by itself.

In the Viptela solution, when a network device fails and a redundant device is present, network operation continues without interruption. This is true for all Viptela devices—vBond orchestrators, vSmart controllers, and vEdge routers. No user configuration is required to implement this behavior; it happens automatically. The OMP peering sessions running between Viptela devices ensure that all the devices have a current and accurate view of the network topology.

Let's examine failure recovery device by device.

---

**Recovering from a vSmart Controller Failure**

The vSmart controllers are the master controllers of the network. To maintain this control, they maintain permanent DTLS connections to all the vBond orchestrators and vEdge routers. These connections allow the vSmart controllers to be constantly aware of any changes in the network topology. When a network has multiple vSmart controllers:

- There is a full mesh of OMP sessions among the vSmart controllers.
- Each vSmart controller has a permanent DTLS connection to each vBond orchestrator.
- The vSmart controllers have permanent DTLS connections to the vEdge routers. More specifically, each vEdge router has a DTLS connection to one of the vSmart controllers.

If one of the vSmart controllers fails, the other vSmart controllers seamlessly take over handling control of the network. The remaining vSmart controllers are able to work with vEdge routers joining the network and are able to continue sending route updates to the vEdge routers. As long as one vSmart controller is present and operating in the domain, the Viptela network can continue operating without interruption.

---

**Recovering from a vBond Orchestrator Failure**

In a network with multiple vBond orchestrators, if one of them fails, the other vBond orchestrators simply continue
operating and are able to handle all requests by Viptela devices to join the network. From a control plane point of view, each vBond orchestrator maintains a permanent DTLS connections to each of the vSmart controllers in the network. (Note however, that there are no connections between the vBond orchestrators themselves.) As long as one vBond orchestrator is present in the domain, the Viptela network is able to continue operating without interruption, because vSmart controllers and vEdge routers are still able to locate each other and join the network.

Because vBond orchestrators never participate in the data plane of the overlay network, the failure of any vBond orchestrator has no impact on data traffic. vBond orchestrators communicate with vEdge routers only when the routers are first joining the network. The joining vEdge router establishes a transient DTLS connection to a vBond orchestrator to learn the IP address of a vSmart controller. When the vEdge configuration lists the vBond address as a DNS name, the router tries each of the vBond orchestrators in the list, one by one, until it is able to establish a DTLS connection. This mechanism allows a vEdge router to always be able to join the network, even after one of a group of vBond orchestrators has failed.

Recovering from a vEdge Router Failure

The route tables on vEdge routers are populated by OMP routes received from the vSmart controllers. For a site or branch with redundant vEdge routers, the route tables on both remain synchronized, so if either of the routers fails, the other one continues to be able to route data traffic to its destination.

Data Plane Software Support for High Availability

For data plane resiliency, the Viptela software implements the standard BFD protocol, which runs automatically on the secure IPsec connections between vEdge routers. These IPsec connections are used for the data plane, and for data traffic, and are independent of the DTLS tunnels used by the control plane. BFD is used to detect connection failures between the routers. It measures data loss and latency on the data tunnel to determine the status of the devices at either end of the connection.

BFD is enabled, by default, on connections between vEdge routers. BFD sends Hello packets periodically (by default, every 1 second) to determine whether the session is still operational. If a certain number of the Hello packets are not received, BFD considers that the link has failed and brings the BFD session down (the default dead time is 3 seconds). When a BFD sessions goes down, any route that points to a next hop over that IPsec tunnel is removed from the forwarding table (FIB), but it is still present in the route table (RIB).

In the Viptela software, you can adjust the Hello packet and dead time intervals. If the timers on the two ends of a BFD link are different, BFD negotiates to use the lower value.

Using Affinity To Manage Network Scaling

In the Viptela overlay network, all vEdge routers establish control connections to all vSmart controllers, to ensure that the routers are always able to properly route data traffic across the network. As networks increase in size, with vEdge routers at thousands of sites and with vSmart controllers in multiple data centers managing the flow of control and data traffic among routers, network operation can be improved by limiting the number of vSmart controllers that a vEdge router can connect to. When data centers are distributed across a broad geography, network operation can also be better managed by having routers establish control connections only with the vSmart controllers collocated in the same
Establishing affinity between vSmart controllers and vEdge routers allows you to control the scaling of the overlay network, by limiting the number of vSmart controllers that a vEdge router can establish control connections (and form TLOCs) with. When you have redundant routers in a single data center, affinity allows you to distribute the vEdge control connections across the vSmart controllers. Similarly, when you have multiple data centers in the overlay network, affinity allows you to distribute the vEdge control connections across the data centers. With affinity, you can also define primary and backup control connections, to maintain overlay network operation in case the connection to a single vSmart controller or to a single data center fails.

A simple case for using affinity is when redundant vSmart controllers are collocated in a single data center. Together, these vSmart controllers service all the vEdge routers in the overlay network. The figure below illustrates this situation, showing a scenario with three vSmart controllers in the data center and, for simplicity, showing just three of the many vEdge routers in the network.

If you do not enable affinity, each vEdge router establishes a control connection—that is, a TLOC—to each of the three vSmart controllers in the data center. Thus, a total of nine TLOCs are established, and each router exchanges OMP updates with each controller. Having this many TLOCs can strain the resources of both the vSmart controllers and the vEdge routers, and the strain increases in networks with larger numbers of vEdge routers.

Enabling affinity allows each vEdge router to establish TLOC connections with only a subset of the vSmart controllers. The figure above shows each router connecting to just two of the three vSmart controllers, thus reducing the total number of TLOCs from nine to six. Both TLOC connections can be active, for a total of six control connections. It is also possible for one of the TLOC connections be the primary, or preferred, and the other to be a backup, to be used as an alternate only when the primary is unavailable, thus reducing the number of active TLOCs to three.

Affinity also enables redundancy among data centers, for a scenario in which multiple vSmart controllers are collocated in two or more data centers. Then, if the link between a vEdge router and one of the data centers goes down, the vSmart controllers in the second data center are available to continue servicing the overlay network. The figure below illustrates this scenario, showing three vSmart controllers in each of two data centers. Each of the three vEdge routers
establishes a TLOC connection to one controller in the West data center and one in the East data center.

You might think of the scenario in the figure above as one where there are redundant data centers in the same region of the world, such as in the same city, province, or country. For an overlay network that spans a larger geography, such as across a continent or across multiple continents, you can use affinity to limit the network scale either by restricting vEdge routers so that they connect only to local vSmart controllers or by having vEdge routers preferentially establish control connections with data centers that are in their geographic region. With geographic affinity, vEdge routers establish their only or their primary TLOC connection or connections with vSmart controllers in more local data centers, but they have a backup available to a more distant region to provide redundancy in case the closer data centers become unavailable. The figure below illustrates this scenario, Here, the vEdge routers in Europe have their primary TLOC connections to the two European data centers and alternate connections to the data centers in North America. Similarly, for the vEdge routers in North America, the primary connections are to the two North American data centers, and the backup connections are to the two European data centers.
As is the case with any overlay network that has multiple vSmart controllers, all policy configurations on all the vSmart controllers must be the same.

### Additional Information

- **Bringup Sequence of Events**
- **Configuring High Availability Parameters**
- **Configuring Network Interfaces**
- **Create a vManage Cluster**
- **Using Affinity to Manage Network Scale**