TriScale Clustering Tech Note
# Table of Contents

Expanding Capacity with TriScale Clustering .......................................................... 2  
How Clustering Works........................................................................................ 2  

Cluster Communication.......................................................................................... 3  
Cluster Configuration and Synchronization.............................................................. 5  
Working with Striped and Spotted IP Management Addresses.................................. 6  
Traffic Distribution within the Cluster...................................................................... 7  
Cluster and Node States......................................................................................... 8  
Cluster Management............................................................................................... 9  
Cluster Support for NetScaler Functionality .......................................................... 10  

**TriScale Clustering in Operation** ........................................................................ 11
Expanding Capacity with TriScale Clustering

Citrix TriScale clustering provides the ability to scale out NetScaler capacity to massive levels by enabling up to 32 NetScaler appliances, physical or virtual, to work in unison to deliver one or more applications. It also provides an alternative to high availability (HA) ADC appliance deployments, so that 100% of network resources are put to productive use.

- **1 Tbps+ scale** – Two to 32 NetScaler appliances can be clustered together to scale ADC capacity beyond 1 Tbps. Because capacity can be added incrementally, and only when needed, datacenter managers can scale without overprovisioning their network.

- **Unified cluster management** – NetScaler clusters provide a single policy management view for the entire cluster. All aspects of management remain constant, even as the cluster expands to meet future growth. Policy changes are made just once, and automatically propagate across all nodes.

- **Ensured transparency** – TriScale enables the NetScaler appliance cluster to appear as a single resource (i.e. single virtual IP address) to all users, preserving ADC transparency.

- **Seamless, linear scalability** – Adding or removing nodes from a NetScaler cluster incurs zero downtime and results in a predictable, linear change to aggregate performance and capacity. New nodes added to a cluster immediately advertise themselves and automatically begin to absorb their share of the load.

- **Built-in fault tolerance** – TriScale clustering provides inherent reliability. If a node fails or becomes unreachable, the traffic handled by that node is automatically redistributed to the remaining active nodes through TriScale’s normal traffic distribution mechanism. In addition, the cluster can be configured with a spare node, which remain in passive, stand-by mode, but automatically replaces a node that fails.

How Clustering Works

By grouping multiple NetScaler appliances together and configuring them to work in unison, Citrix TriScale enables IT to efficiently scale its ADC infrastructure. One or more applications can be supported by multiple NetScaler appliances simultaneously, so that they benefit from the aggregate capacity of all cluster nodes. A cluster can be treated exactly like a single node. In other words, a NetScaler cluster behaves logically one NetScaler device. This holds even from a networking perspective. For example, all ports across multiple NetScaler appliances in a cluster can be treated as they were one device. A NetScaler cluster is configured and managed as a single system.
The virtual IP (VIP) address, representing the termination point for an application, is striped across all nodes in the cluster. Application requests and responses are distributed fairly among all cluster nodes.

**Cluster Communication**

The NetScaler cluster communicates with application clients through the physical connections between the cluster nodes and the client-side connecting device – which are typically aggregated via a layer two switch. The logical grouping of these physical connections is called the client data plane. Similarly, the cluster communicates with back-end servers through the physical connections between the cluster nodes and the server-side connecting device. The logical grouping of these physical connections is called the server data plane.

In addition, cluster nodes communicate with each other using a cluster backplane. All cluster nodes reside on the same subnet and are connected via an Ethernet backplane. The backplane, which includes the physical connections between each cluster node and a backplane switch, is the backbone of the cluster system. This serves three purposes:

1. Management synchronization is accomplished via the backplane to support a single configuration across all nodes. Individual NetScaler nodes are not managed separately.
2. Synchronization of persistence and health check information ensures that individual nodes can be inserted and removed from the cluster pool.
3. Packets can be forwarded to other nodes in the cluster in the event a connection is being tracked by one node, but packets for that connection happen to arrive on a different node.

The backplane can also be wired for redundancy using typical switch redundancy topologies.
Figure 1- Cluster communication interfaces

Figure 1 shows the logical grouping of the physical connections that form the client data plane, server data plane, and cluster backplane.

Some of the benefits of clustering include:

- **Layer 2 Scalability**: Multiple ports (across multiple NetScaler boxes in a cluster) can be selected to form a LAG (Link Aggregation Group). This group behaves fully compliant with 802.1ax/LACP. Ports on the other side of the cluster can similarly be terminated on an Ethernet switch with the corresponding LAG defined.

- **Layer 3 Routing Advantages**: Ports on the NetScaler nodes can be utilized identical to a single-device configuration, and use Layer 3 techniques, such as, ECMP, to ensure incoming packet flows take advantage of all connected wires.

- **Linksets**: Linksets allow multiple cluster ports to be connected on the same broadcast domain (Ethernet segment), without the need to define a LAG on the switch side. The NetScaler devices coordinate management of broadcast traffic (ARP and ARP response), and suppress spanning tree protocol loops that may otherwise result. This removes any layer 2 switch limitation on the number of ports that can be put in a LAG. Fine grained and failure aware flow distribution code running on the NetScaler nodes determines how traffic is distributed across the links. Linksets enable even utilization of all links of a LAG.
Most importantly, the cluster does not rely on the external environment to evenly distribute traffic among cluster nodes. Ports can be selected from one node, or multiple nodes, when providing connectivity to incoming traffic. Furthermore, regardless of the way traffic arrives at the NetScaler, the cluster guarantees even workload distribution on each device or node.

**Cluster Configuration and Synchronization**

One of the cluster nodes is designated as a configuration coordinator (CCO). To synchronize the configurations on a cluster, the CCO coordinates all cluster configurations and owns the management address of the cluster, which is called the cluster IP address.

Administrative access to the CCO is achieved through the cluster IP address, as shown in Figure 2:

![Figure 2 - Configuring the cluster through the cluster IP address](image)

All configuration changes performed through the cluster IP address are propagated to the cluster nodes through the cluster backplane.

When a new appliance is added to the cluster, its configuration settings and files (SSL certificates, licenses, DNS parameters, etc.) are automatically cleared and the appliance is synchronized with the cluster configuration. (Note: Policy configurations should be saved before re-provisioning a NetScaler appliance into a cluster so that information is not lost.)
When an existing cluster node rejoins the cluster (e.g. after a failure or intentionally disabling a node), the cluster checks the node’s configuration. If there is a mismatch between the current configuration of the node and that being managed by the CCO, the re-joining node is automatically synchronized by using one of the following synchronization techniques:

**Full synchronization** – If the difference between configurations exceeds 255 commands, all of the configurations implemented on the CCO are applied to the node that is rejoining the cluster.

**Incremental Synchronization** – If the difference between configurations is less than 256 commands, only the configurations that are not the same are applied to the node that is rejoining the cluster. This results in faster synchronization when configuration deltas are small.

Only configurations performed on the CCO by accessing it through the cluster IP address are synchronized to the other cluster nodes. Configurations performed by accessing cluster nodes through their NSIP addresses are not synchronized to the other cluster nodes.

**Working with Striped and Spotted IP Management Addresses**

In a clustered deployment, the NetScaler management IP (MIP) address and subnet IP (SNIP) address can be striped or spotted.

- A striped IP address is active on all the nodes of the cluster. IP addresses configured on the cluster without specifying an owner node are active on all the cluster nodes.

- A spotted IP address is active on, and owned exclusively by a single node. IP addresses configured on the cluster by specifying an owner node are active only on the node that is specified as the owner.

Figure 3 shows striped and spotted IP addresses in a three-node cluster. The VIP address 10.102.29.66 is striped on all the cluster nodes, and SNIP address 10.102.29.99 is active on NS0, NS1 and NS2.
Traffic Distribution within the Cluster

The specific NetScaler node that receives application request or response from an external connecting device (e.g. layer two switch) is referred to as the Flow Receiver. The node that actually processes the traffic flow is called the Flow Processor. Both the Flow Receiver and Flow Processor are active nodes in the cluster, and are fully capable of handling traffic.

One of three traffic distribution mechanisms is employed to determine which node will serve as the Flow Receiver and receive traffic: equal cost multi-path (ECMP) routing, cluster aggregation (CLAG) or linkset. Each of these mechanisms uses a different algorithm to determine the Flow Receiver.

The Flow Receiver then uses internal cluster logic to select the right Flow Processor, that is, the node that will handle the traffic.
Figure 4 - Traffic distribution in a cluster

Figure 4 shows a client request flowing through the NetScaler cluster. The client sends a request to a striped virtual IP (VIP) address. A traffic distribution mechanism configured on the client data plane selects one of the cluster nodes as the Flow Receiver. The Flow Receiver receives the traffic, determines the proper Flow Processor that must process the traffic, and then steers the request to that node through the cluster backplane. If the Flow Receiver and Flow Processor reside on the same physical node the request does not traverse the cluster backplane.

Cluster and Node States

Cluster node classification includes three types of states: admin states, operational states, and health states.

**Admin State.** An admin state is configured when you add the node to the cluster. It indicates the purpose of the node, which can be in one of the following states:

**ACTIVE** – Nodes in this state serve traffic if they are operational and healthy.

**PASSIVE** – Nodes in this state do not serve traffic, but remain fully synchronized with the cluster. These nodes are useful during maintenance activity because they can be upgraded without removing the node from the cluster.

**SPARE** – Nodes in this state do not serve traffic, but remain fully synchronized with the cluster. Spare nodes act as back-up nodes for the cluster. If one of the ACTIVE nodes becomes unavailable, the operational state of one of the spare nodes becomes ACTIVE, and that node starts serving traffic.
Only nodes that have the admin state as ACTIVE, operational state as ACTIVE, and health status as UP can serve traffic. If one node in a two node cluster was to have a health Status of DOWN, the remaining node would remain functional and continue to support traffic.

**Cluster Coordination**

The following techniques are employed to elect a cluster node for coordination and ensure full reliability in the event of a failure.

An operating cluster has an elected cluster coordinator. This election process is automatic. Once elected, the coordinator is responsible for handling any action that requires the entire cluster to operate as one. Activities include adding or removing cluster nodes and handling traffic rebalance as required.

Every node in the cluster has the most up-to-date information about the cluster. Failover is built-in to the cluster such that any node is capable of taking over responsibility of another node in the event of failure. The cluster continuously monitors topology, connectivity and reachability status of all its links and services.

When a change is detected (e.g., a node joining or leaving, or a downstream router failure), a distributed algorithm runs on all nodes to find the maximcally connected set of nodes. This will ensure that the most well connected subset of nodes is selected for processing traffic through the cluster. Nodes that suffer connectivity loss are transparently disallowed from serving traffic.

**Cluster Management**

NetScaler cluster deployments are managed via a centralized dashboard, accessible via the cluster management IP (CLIP). It provides a complete view on cluster deployment across nodes. It also provides per-node details for detailed monitoring and troubleshooting.

NetScaler’s central monitoring and management solution, Command Center, also supports cluster monitoring and management by enabling the following operational tasks:

**Setting up a Cluster** – existing non-clustered NetScaler appliances can be reconfigured as nodes within a single cluster.

**Add/Remove a Node** – Incrementally add a node to an existing cluster, or remove a node from an existing cluster
Centralized monitoring – monitor the performance of cluster nodes on a real-time basis

Gathering report and log information – get aggregated stats/logs from all cluster nodes

Cluster Support for NetScaler Functionality

NetScaler clustering supports the most popular and commonly deployed NetScaler features, including:

- Load balancing
- Content switching
- SSL offload
- Compression
- Routing
- Content rewrite
- L4 denial of service (DoS) protections
- Nitro RESTful API
- HTTP denial of service (DoS) protections*
- Static and dynamic caching*
- DNS caching*

* Single node support only

In addition, each node must be of the same form factor (e.g. NetScaler MPX physical appliance or NetScaler VPX virtual appliance), matching maximum capacity rating, the same NetScaler software edition, and an identical set of licensed NetScaler features.

Failure Management and Fault Tolerance

NetScaler clustering replicates session-based information such as persistence information and SSL handshakes. A consistent view across the cluster is also maintained including health check responses and server response times. For example, a retry of an HTTP transaction that lands on another node after a failure will behave correctly and load balance to the correct server.

Unlike many chassis-based clustering solutions, where a single node failure may cause all connections across all nodes to drop, NetScaler clustering implements a more effective loss model. In every scenario, the loss of information in the cluster is proportional to the amount of loss suffered. For example, if one node in a five node cluster were to fail, the loss will be no more than 20%. During all failure scenarios, the cluster continues to guarantee that work is evenly divided among remaining nodes.
When a cluster is undergoing a loss, or recovering from joining or leaving of a node, the clustering algorithms do no cost more during recovery. This is critical, because otherwise the loss of a single node can lead to a ‘cascading’ failure, where the resultant recovery phase causes more nodes to fail.

For example, when a node fails, the work corresponding to that node is equally divided among the remaining nodes. If instead, as is common in other clusters, the entire burden of the failed node were placed on just one other node – effectively doubling its load – it could possibly cause the second node to fail as well, leading to a cascading failure.

**TriScale Clustering in Operation**

Clustering provides a dynamic ability to increase or decrease the number of NetScaler nodes serving one or more applications based on required capacity. Adding a new node is a simple operation that can be accomplished while live traffic is going through the deployment. The node is added to the cluster, and its configuration is automatically synchronized with the CCO.

Once the node is active, traffic is gracefully distributed to the new node. Existing connections are not shifted to it. Instead, the node participates in scaling up by processing its share of new traffic directed to the cluster.

NetScaler clusters also support the notion of a SPARE node, which can be part of an active cluster without participating in traffic processing. This node has its configuration synchronized with the CCO, and is ready to replace a failed or unreachable node at any time. If an active node goes down for any reason, the SPARE node assumes an ACTIVE state, and replaces the faulty node in real time.

This replacement mechanism is fully automated and does not require administrator intervention. Support for spare capacity provides redundancy in the deployment, and provides a viable alternative to traditional HA pair deployments. As best practice, it is recommended to have one SPARE node for every five to 10 active nodes in a cluster to ensure a high degree of cluster reliability and sustained capacity.