Achieving High Availability

in Linux-based Cluster Environments

High availability is an important element of enterprise server clusters, helping minimize application downtime following a server failure. This article discusses commercial packages for creating highly available applications and services on the Linux® platform, including popular options for both the Novell® SUSE® Linux Enterprise Server and Red Hat® Enterprise Linux operating systems.

BY KEVIN GUINN AND SEEMA PADGHAN

Related Categories:

High availability (HA)

Linux

Novell SUSE Linux

Red Hat Enterprise Linux

Visit www.dell.com/powersolutions for the complete category index.

Several software packages enable servers running Linux to provide highly available applications and services. This article provides a high-level overview of vendor-provided applications available for Novell SUSE Linux Enterprise Server (SLES) and Red Hat Enterprise Linux. SLES includes the Heartbeat package, which provides a framework for failover functionality. Red Hat Enterprise Linux offers the optional Red Hat Cluster Suite (RHCS), which is licensed separately and enables advanced features for highly available applications and services.

General high-availability concepts

High-availability cluster software tools typically include several common features. At a minimum, cluster software must incorporate a mechanism to define which systems are available for use as cluster nodes, which services or applications can fail over between nodes, and which interconnects can be used to convey communications between nodes. To prevent the data corruption that could be introduced when distinct subsets of the nodes have control of the same cluster resources, cluster software typically includes split-brain detection and basic node fencing or more complicated I/O fencing. Additionally, cluster software may incorporate cluster management and monitoring tools and predefined scripts to help configure common services or applications.

Different cluster storage models are also possible, and can be selected based on the nature of the applications and services for which high availability is required. A cluster can use replicated storage or shared storage. Shared storage can be configured for either exclusive access or concurrent access.

Heartbeat and SUSE Linux Enterprise Server

The Heartbeat package is available from the High-Availability Linux Project (www.linux-ha.org), and different versions are included with the SLES 9 and SLES 10 operating systems. SLES 9 includes Heartbeat 1.*x*, which allows the creation of a two-node cluster that provides basic high-availability failover services. SLES 10 includes Heartbeat 2.*x*, which provides enhanced features and functionality for multi-node clusters.

Features of Heartbeat 1.*x*

Heartbeat 1.*x* allows cluster members and resources to be configured using the following two text files in the /etc/ha.d directory:

- **ha.cf:** Defines the cluster nodes, fault detection and failover time intervals, cluster event-logging mechanism, and node-fencing method
- **haresources:** Defines groups of cluster resources, where each line defines a default node and a set of resources that will fail over together; the resources may include IP addresses, file systems, services, or applications

Features of Heartbeat 2.0

Heartbeat 2.0 supports either a Heartbeat 1.*x*–style configuration (limited to two nodes) or a configuration method that employs a modular architecture featuring Cluster Resource Manager (CRM). Clusters with up to 16 nodes have been tested during development of the CRM model, which employs the XML Cluster Information Base (CIB) for configuration. The CIB file (/var/lib/heartbeat/crm/cib.xml) is automatically replicated among nodes, and defines the following objects and actions:

- Cluster nodes
- Cluster resources, including attributes, preferences, grouping, and dependencies
- Logging, monitoring, quorum, and fencing criteria
- Actions to take when a failure or another specified criterion is encountered

Figure 1 shows the key components of the Heartbeat 2.0 architecture. The Consensus Cluster Membership service uses an election process to allow the cluster nodes to determine the Designated Coordinator (DC), which helps establish a quorum and manages cluster node membership and resource assignments. The DC maintains the authoritative copy of the cluster state and administrative policies; other nodes must forward state-change requests to the DC for processing. The Heartbeat service is responsible for checking node and link status and determining whether failure has occurred. The cluster event-logging service (ha_logd) provides a log facility for the other services and daemons in the cluster suite.

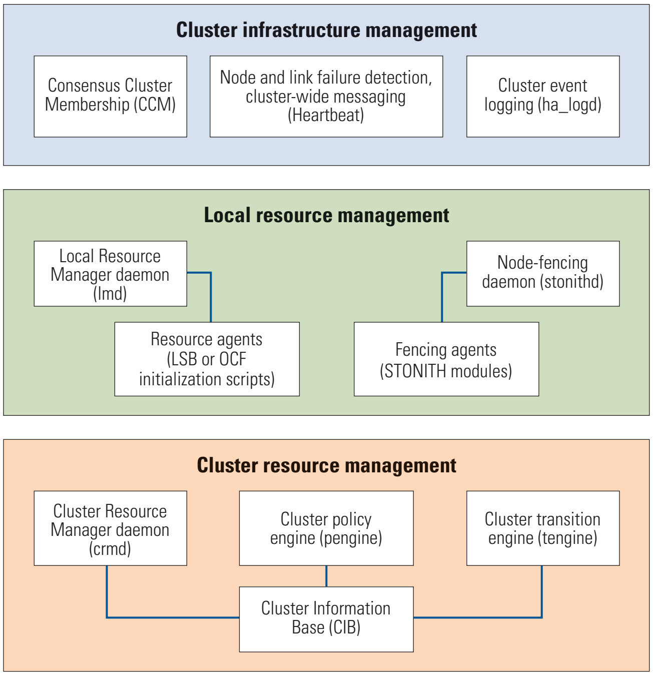
Cluster infrastructure management
Consensus Cluster Membership (CCM)
Node and link failure detection, cluster-wide messaging (Heartbeat)
Cluster event logging (ha_logd)
Local resource management
Local Resource Manager daemon (lmd)
Node-fencing daemon (stonithd)
Resource agents (LSB or OCF initialization scripts)
Fencing agents (STONITH modules)
Cluster resource management
Cluster Resource Manager daemon (crmd)
Cluster policy engine (pengine)
Cluster transition engine (tengine)
Cluster Information Base (CIB)

Figure 1. Heartbeat 2.0 architecture

To control cluster resources, Local Resource Manager (LRM) starts, stops, and monitors resource agents. The LRM daemon (lrmd) is responsible for communicating cluster events to the DC. Node-fencing agents are treated as a special type of resource, and are controlled by the node-fencing daemon, stonithd. The name *STONITH* originated as an acronym for “Shoot the Other Node in the Head,” and the program has evolved into a tool that provides several mechanisms for disabling a failed node, including serial- or network-controlled power switch devices and remote management hardware. When nodes are not able to communicate properly, fencing helps prevent different subsets of the cluster from running the same resources. This scenario is referred to as a partitioned cluster or a split-brain condition. Split-brain conditions are avoided using application design, node fencing, or resource-specific fencing.

The CRM daemon (crmd) helps manage the CIB, which allows advanced constraints and dependencies to be applied to control the behavior of nodes and resources. The cluster policy engine (pengine) interprets and applies these constraints and dependencies. The cluster transition engine (tengine) manages the state of the CRM and coordinates the process of restarting or moving resources to alternate nodes in the event of a failure.

Configuration tools

Heartbeat 2.0.5, which is included with SLES 10, introduces a graphical user interface (GUI) for cluster management and monitoring. It also includes sample scripts that aid in the configuration of common Linux services and applications, including xinetd-based services, Apache Web servers, IBM® DB2 databases, and IBM WebSphere Application Server. Many other applications, such as Network File System (NFS) and Samba (a Common Internet File System implementation) file shares, can also be configured. Heartbeat 2.0 complies with the Open Cluster Framework (OCF) resource-agent application programming interface, allowing the use of generic Linux Standard Base (LSB) initialization scripts or cluster-specific OCF resource initialization scripts.

Either version of Heartbeat can also be configured in association with Linux Virtual Server for IP load balancing. Depending on the needs of the configured resources or services, shared storage with dedicated access, replicated storage (using drbd or other means), or concurrent access to data using a cluster file system are all possible. Tight integration with Oracle® Cluster File System Release 2 is planned for future Heartbeat releases.

The GUI tools included with Heartbeat 2.0.5 in SLES 10 help simplify the configuration. Also, Novell plans to enable the use of the Heartbeat 2.0 core services as the foundation for future versions of Novell Cluster Services™ (NCS) software. NCS is licensed separately and includes predefined resource types as well as GUI configuration and monitoring tools.

Red Hat Cluster Suite and Red Hat Enterprise Linux 4

RHCS is specifically designed for Red Hat Enterprise Linux and provides two distinct types of clustering:

- **Application and service failover:** Creates *n*-node server clusters for failover of key applications and services
- **IP load balancing:** Load balances incoming IP network requests across a farm of servers

The cluster components are Cluster Manager (CMAN), Cluster Configuration System (CCS), and Resource Group Manager (rgmanager). Figure 2 shows the different services and daemons that run on a cluster node at any given time.

CCS provides access to a single cluster configuration file, /etc/cluster/cluster.conf, which is stored on each node. The configuration file includes a version number, which is updated whenever the cluster configuration changes. The CCS daemon (ccsd) running on each node manages this file and provides access to it. When ccsd is started, it finds the most recent version of the file among the cluster nodes.

CMAN is used for managing cluster membership, messaging, and notification. CMAN consists of a set of kernel patches and a user-space program (cman_tool).

The cman_tool program can be used to join or leave a cluster, disable another node, or change the value of expected votes of a cluster. CMAN depends on CCS.

The Resource Group Manager daemon (clurgmgrd) handles the management of administrator-defined cluster services (also known as resources), including administrator requests such as service start, service disable, service relocate, and service restart. It also handles service restart and service relocate following a failure.

Configuration tools

RHCS supports 16 nodes in a cluster. A GUI-based cluster configuration tool is available with Red Hat Enterprise Linux 4 RHCS (system-config-cluster). The cluster configuration—which includes the resource information, node information, fencing devices information, and failover domain information—is stored in the /etc/cluster/cluster.conf file on each node in XML format. The resources are grouped together under one service (resource group).

A failover domain is a named subset of cluster members that are eligible to run a cluster service following service failover. The failover domain can have the following characteristics:

- **Unrestricted:** Administrators can specify that a subset of members is preferred, but that a cluster service assigned to this domain can run on any available member.
- **Restricted:** Administrators can restrict the members that can run a particular cluster service. If none of the members in a restricted failover domain are available, the cluster service cannot be started (either manually or by the cluster software).
- **Unordered:** When a cluster service is assigned to an unordered failover domain, the member on which the cluster service runs is chosen from the available failover domain members with no priority ordering.
- **Ordered:** Administrators can specify a preference order among the members of a failover domain. The member

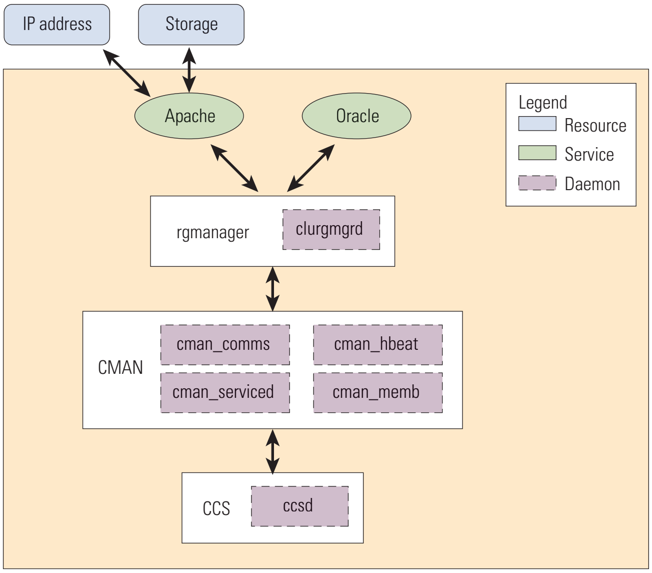

Figure 2. Red Hat Cluster Suite architecture

Feature	SUSE Linux Enterprise Server with Heartbeat	Red Hat Cluster Suite
Deployment, configuration, and management	GUI tools provided to help simplify configuration and management tasks	GUI tools provided to help simplify configuration and management tasks
Software availability	Included with SLES; also available from www.linux-ha.org	Can be purchased and downloaded from www.redhat.com
Maximum number of nodes	Heartbeat 1.*x* or 2.0 without CRM: 2 nodes Heartbeat 2.0 with CRM: 16 nodes	16 nodes
Resource types	IP address, file system, NFS exports, Samba, Apache, and LSB and OCF initialization scripts	IP address, file system (including mounted NFS volumes), NFS exports, scripts, and Red Hat Global File System
Fencing devices	Bundled support for various network and serial power switch devices, and remote management cards and devices; extensible interface allows use of custom devices, such as the Dell™ Remote Access Controller (DRAC)	Bundled support for various network and serial power switch devices, remote management cards and devices, and Fibre Channel switches; scripts can be written to allow the use of other devices, such as the DRAC
Rolling upgrades	Not supported; however, Heartbeat 2.0 supports limited 1.*x*-style configurations and includes scripts for converting basic resource types to use CRM and CIB	Not supported
Shared storage	Three possible storage models, depending on the resources and services configured in the cluster: • Shared storage with dedicated access using traditional file systems • Shared storage with concurrent access using cluster file systems • Replicated storage with dedicated access	
Hardware requirements	Industry-standard servers, such as Dell PowerEdge™ servers; optional shared storage devices, such as Dell/EMC Fibre Channel storage arrays	

Figure 3. Comparison of Novell and Red Hat high-availability software packages

at the top of the list is the most preferred, followed by the second member on the list, and so on. Administrators can choose a preferred node for a cluster service by creating an unrestricted failover domain with a single node.

CCS allows changes to the configuration when the cluster is online. The configuration information is propagated to other nodes using the send-to-other-node operation.

Failover capabilities

Similar to STONITH, fence devices enable a node to power cycle another node before restarting its services as part of the failover process. The ability to remotely disable a node helps maintain data integrity under any failure condition. Fence devices can protect against data corruption if an unresponsive (or hanging) node becomes responsive after its services have failed over, and issues I/O to a disk that is also receiving I/O from another node. In addition, if CMAN detects a node failure, the failed node is removed from the cluster. If a fence device is not used in the cluster, then a failed node may result in cluster services running on more than one node, which can cause data corruption and even system crashes.

The infrastructure in a cluster monitors the state and health of an application so that if an application-specific failure occurs, the cluster automatically restarts the application. In response to the application failure, the cluster attempts to restart the application on the node it was initially running on; if that fails, the application is restarted on another cluster node. Nodes eligible to run a cluster service can be specified by assigning a failover domain to the cluster service. In addition to automatic cluster-service failover, a cluster service can be stopped on one node and restarted on the other, allowing planned maintenance of a node system while continuing to provide application and data availability.

Two commercial options for high-availability clustering

Both Novell and Red Hat offer software packages that allow services and applications hosted in the Linux operating environment to reap the benefits of high-availability clustering; Figure 3 provides a summary of the different features of each package. Heartbeat 2.0 and the Red Hat Cluster Suite allow administrators to configure multi-node clusters with failover and monitoring capabilities for several types of resources. These packages can be configured to support most Linux-based services and applications, but typical deployments involve databases, Web-based application services, and file shares. Cost-conscious enterprise IT organizations may opt for the Heartbeat program, which is provided as part of the Novell distributions, whereas IT organizations that seek enhanced ease of use may opt to pay for the additional licenses required for an RHCS cluster and its GUI tools.

Kevin Guinn is a systems engineer in the Dell High-Availability Cluster Development Group. His current interests include storage management and business continuity. Kevin is a Microsoft® Certified Systems Engineer and has a B.S. in Mechanical Engineering from The University of Texas at Austin.

Seema Padghan is an engineering analyst in the High-Availability Cluster Solutions Engineering Group at the Dell Bangalore Development Center. Seema has a master's degree in Computer Science from Pune University in India.

FOR MORE INFORMATION

Dell and Linux:
www.dell.com/linux

Dell Linux Community:
linux.dell.com

Novell SUSE Linux Enterprise:
www.novell.com/linux/suse

High-Availability Linux Project:
www.linux-ha.org

Red Hat:
www.redhat.com

Red Hat Cluster Suite:
www.redhat.com/software/rha/cluster