Core 11.3.0.0 High Availability Management Guide

July 2021

For complete product documentation, see:
Ivanti Documentation Home Page

Copyright © 2021, Ivanti. All rights reserved.
Privacy and Legal

Contents

Copyright © 2021, Ivanti. All Rights Reserved. Privacy and Legal.

High availability overview

High Availability (HA) can be set up in a variety of ways. A basic Core HA setup contains a minimum of a Primary Core server (also referred to as original, main, or active) running in an Active state. This server is paired with a Secondary Core server (also referred to as standby, passive, or inactive) that is in a Standby state. The Secondary checks the state of its paired Primary through a process called "heartbeat" (configured during HA Standby setup).

The Secondary server periodically synchronizes with its paired Primary server, ensuring it has the latest changes as the Primary. The synchronization process frequency is configurable and automated. When the synchronization process detects any changes in the Primary, the Secondary replicates the changes. When it detects that the Primary is unresponsive, it initiates a failover process. Depending on how the system was configured for failover, the Secondary could continue to operate in a standby state, but most often it is configured to swap roles with the Primary server so that the Secondary server is in an active state while the Primary server is inactive.

HA scenario terminology

Terminology is important when discussing HA because while the server names are static (Primary/Secondary), the states can swap (Active/Standby).

The following tables describes the modes and states for Core servers in an HA environment.

TABLE 1. HA SCENARIO TERMINOLOGY

HA Scenario	Mode	State
Primary server is up.	Primary	Active
	Secondary	Standby
Primary server is down.	Primary	Not Available
	Secondary	Automatically promoted to Active and starts responding to requests.
Primary is up again. The administrator must manually sync Primary from Secondary	Primary	Swaps roles with the Secondary and returns to an Active state.
	Secondary	Returns to Standby state.

Copyright © 2021, Ivanti. All Rights Reserved. Privacy and Legal.

HA best practice recommendations

We recommend the following best practices for successfully setting up HA.

- The Primary and Secondary Core servers must have:
  - the same network access
  - identical resources (CPU, RAM, Ivanti software, and so on)
- The replication process should not be set more frequently than 60 minutes.
- Both Core servers need to have the exact same clock time.
- Never power down either server while HA is syncing.
- HA Status/Modes should be periodically monitored to ensure it matches expected configuration.
- When setting up the Secondary server, it is recommended to keep the default settings under the **Fail Over Controls** section.
- Configure all notifications under **Notification Settings** on the Secondary server. For example, email notification, sync completion and failure notification.
- Periodically review the **Sync History** log (located on the Secondary server under **Last Sync Status**) and look for sync failures. Any and all sync failures should be promptly addressed for proper HA operations.
- Periodically review the **Sync History** log (located in the Secondary server under **Last Sync Status**) and look for sync completion times. Use these completion times to review existing **Run Every XXX Minutes** interval to ensure syncs are not overlapping and there is adequate idle time between syncs.
- Ensure that any time you change any setting in the HA configuration window on the Secondary, the **StartTime** setting is reviewed to ensure it matches desired **StartTime** in relation to when it was saved. For example, if the **StartTime** is set for 1:00 pm and you make a change to any additional settings at 2:00 pm and save the configuration, the sync process will execute at 1:00 pm the following day. After the first **StartTime** executes the **Run Every** value takes over from this starting time.

Copyright © 2021, Ivanti. All Rights Reserved. Privacy and Legal.

Core Replication Details

The following Core elements are replicated when a Standby Core server replicates from a Primary Core:

- Admin Portal complete database included:
  - Users & Devices
  - Apps
  - Policies & Configs
  - Settings
  - Logs & Events
- System Manager Portal NOT included, except:
  - Device certificates and Server SSL certificates
  - System Manager local users
- All Atlas data is replicated when Atlas is enabled
  - Note that Atlas must be enabled on both the Primary and Secondary servers to work
  - Atlas-only server replication is also supported for those installations where Atlas is set up on its own server and High Availability is required
- Additional application and system files required by the Core to ensure consistency and data integrity.

When failover occurs, any number of scenarios can take place depending on how an HA environment is configured. The following scenarios are described in the following sections:

- "Two Core servers across two data centers " on page 6
- "Three Core servers across two data centers" on page 7
- "Two Core servers and two Sentry servers on one data center" on page 9
- "Three Core servers and two Sentry servers on two data centers " on page 11

Copyright © 2021, Ivanti. All Rights Reserved. Privacy and Legal.

High availability scenarios

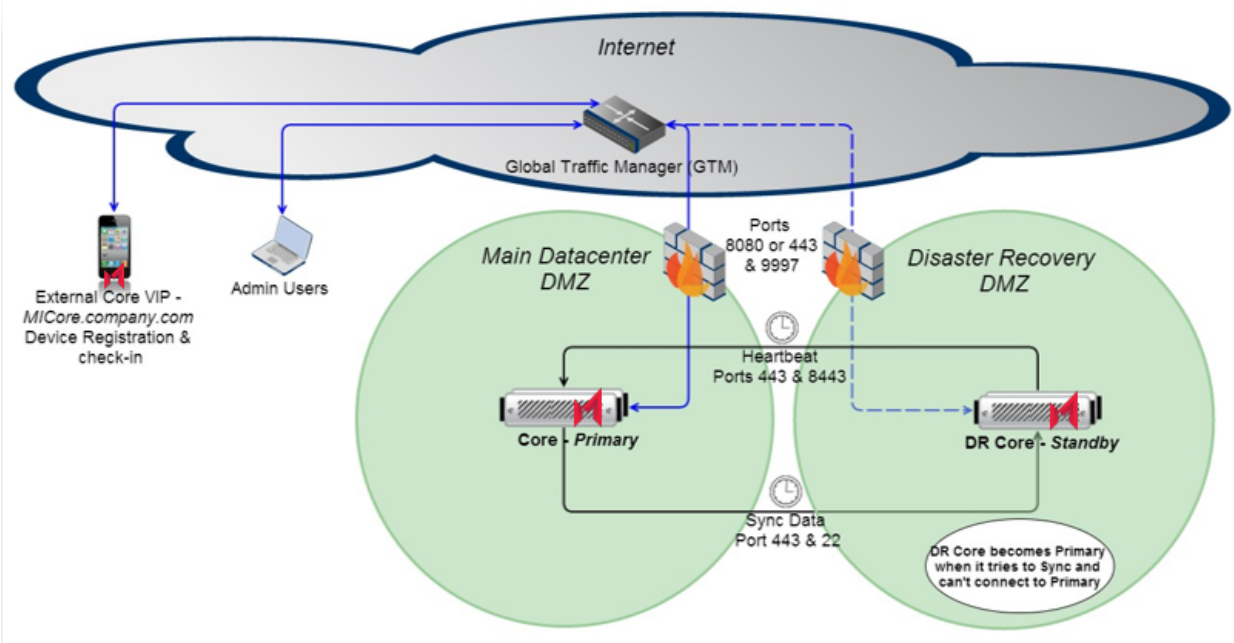
This section describes architecture scenarios for the Core High Availability (HA) solution. Each scenario focuses on key components. These scenarios are typical, but not inclusive, and illustrate examples of different network and data center configurations. There could be different redundant combination of Core servers across multiple data centers with many different routing network connections. The diagrams outline the important components and port connectivity required to ensure proper operation.

Two Core servers across two data centers

This example shows the simplest Core High Availability (HA) Architecture with its related components.

Disable SSH protection on the Firewall when connecting between two Cores

FIGURE 1. TWO CORE SERVERS ACROSS TWO DATA CENTERS



The components in the diagram are:

Copyright © 2021, Ivanti. All Rights Reserved. Privacy and Legal.

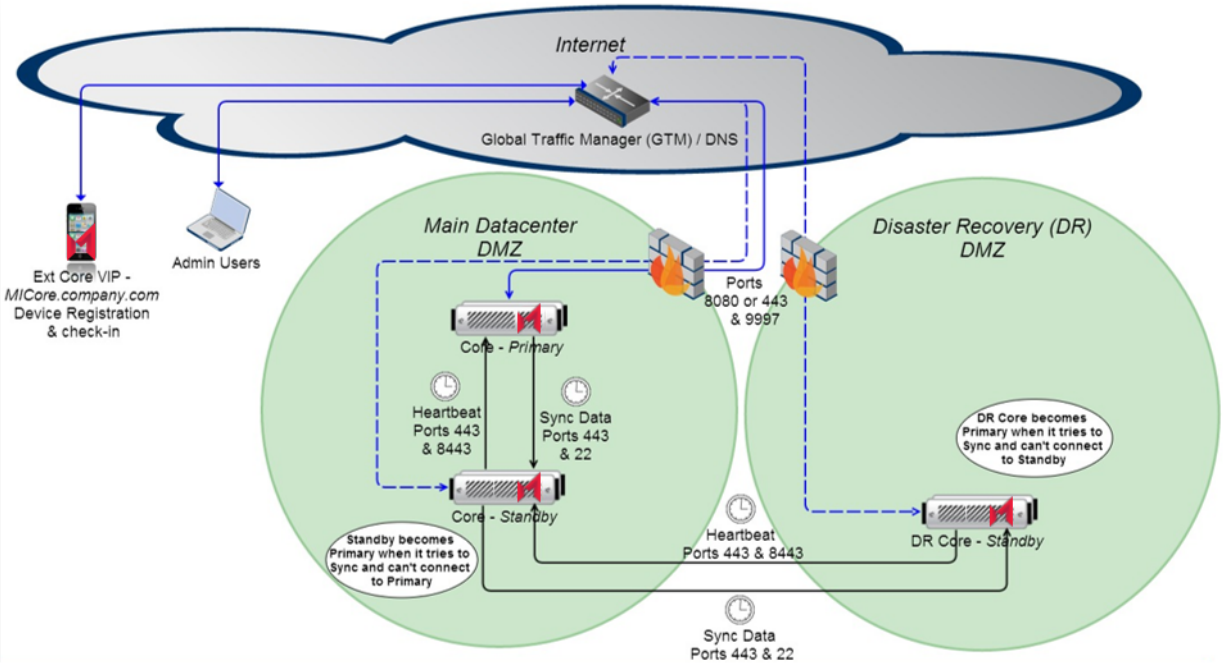
- One data center hosts the Primary Core.
- A second, Disaster Recovery (DR), data center hosts the Secondary Core.
- A Global Traffic Manager (GTM) or DNS or load balancer controls traffic to the Primary. This “traffic controller” monitors the health of the Primary and Secondary servers. When it detects the Primary has become unresponsive, it routes traffic to the Secondary.
- The Secondary Core checks the status of the Primary through a process called “heartbeat”. This process is configured during HA Standby setup. This process detects if the Primary becomes unresponsive. When this happens it initiates the failover process. When a failover occurs, the Secondary attempts to become Primary, depending on what settings have been configured; it might stay as a Secondary or become Primary.
- The Secondary periodically synchronizes with the Primary ensuring it has the latest changes as the Primary. The synchronization process frequency is configurable and is automated.
- The ports used to communicate between Cores are ports 8443, 443 and 22 as outlined in the diagram. This intra-Core communication is essential for proper Core HA operation.

Three Core servers across two data centers

The example in this section shows high availability in the Main data center using a disaster recovery approach.

Copyright © 2021, Ivanti. All Rights Reserved. Privacy and Legal.

FIGURE 2. TWO CORE SERVERS ACROSS TWO DATA CENTERS



This approach is typically used in environments where a main data center is expected to be always available and a Disaster Recovery data center is exclusively used as part of business continuity approach and typically requires manual intervention to bring online.

The key components in this architecture include:

- A main data center hosting a pair of Cores. This pair of Cores is set up as Primary and Secondary. These two Cores served as the main Core High Availability solution. The third Core serves as part of the Disaster Recovery (DR) configuration and it resides in the DR data center.
- Another data center (Disaster recovery) hosting a third Core in Secondary mode.
- A Global Traffic Manager (GTM) or DNS or load balancer that controls the traffic to the Primary Core. This “traffic controller” monitors the health of the other Cores and detects when the Primary becomes unresponsive and begins routing traffic to the Secondary in the Main data center or DR data center in case of Main data center failure. This is how external traffic is controlled and routed to the Primary Cores.

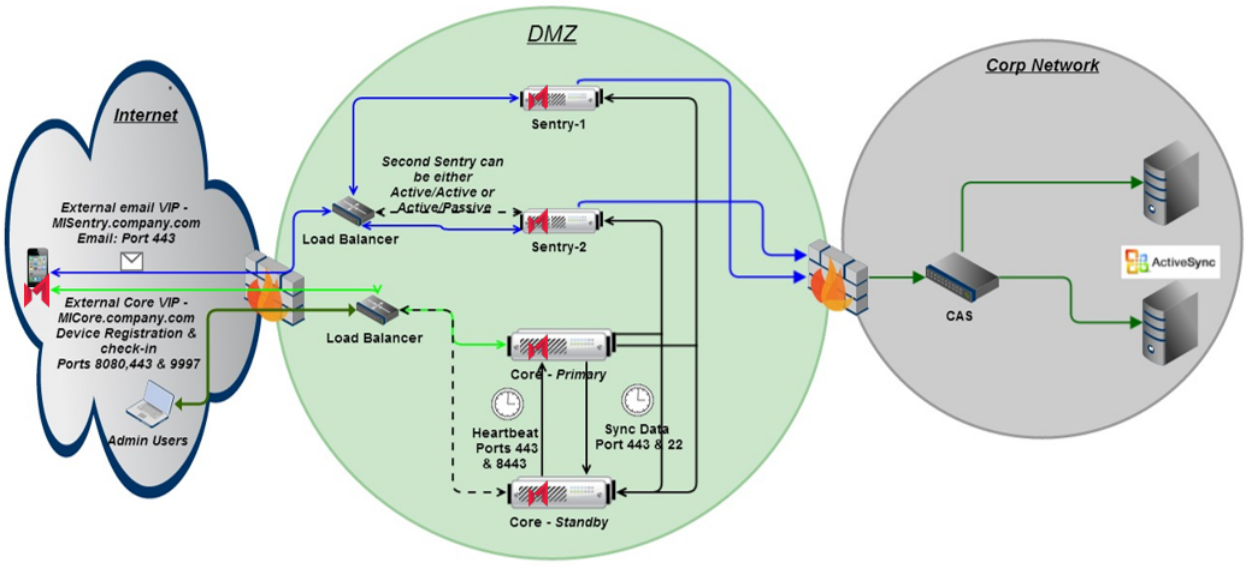
Copyright © 2021, Ivanti. All Rights Reserved. Privacy and Legal.

- The Secondary Core checks the status of the Primary through a process called “heartbeat”. This process is configured during HA Standby setup. This process detects if the Primary becomes unresponsive. When this happens it initiates the failover process. When a failover occurs, the Secondary attempts to become Primary, depending on what settings have been configured; it might stay as a Secondary or become Primary. In the case of the Core located in the DR data center, it sees the Secondary in the Main data center as its Primary Core and the failover process takes place between these two Cores.
- The Secondary Cores periodically synchronizes with its paired Primary Core, ensuring it has the latest changes as the Primary. The synchronization process frequency is configurable and is automated.
- The ports used to communicate between Cores are ports 8443, 443 and 22 as outlined in the diagram. This intra-Core communication is essential for proper Core HA operation.

Two Core servers and two Sentry servers on one data center

The example in this section describes a typical Core and Sentry High Availability Solution.

FIGURE 3. TWO CORE SERVERS AND TWO SENTRY SERVERS ON ONE DATA CENTER



While the Sentry HA details are outside the scope of this document, it is used here to show a typical Core complete HA solution architecture. For details about Sentry, please refer to the latest *Standalone Sentry Installation Guide*.

The key components in this architecture include:

Copyright © 2021, Ivanti. All Rights Reserved. Privacy and Legal.

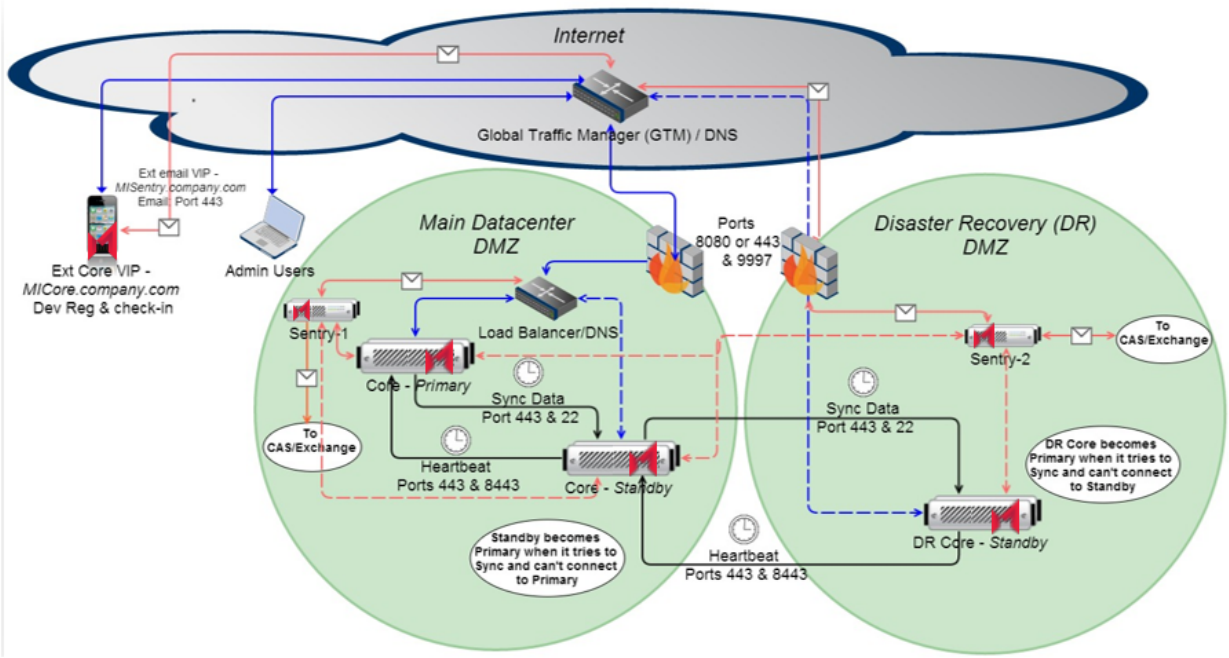
- One data center hosting a pair of Cores. This pair of Cores is set up as Primary and Secondary. These two Cores serve as the Core High Availability solution. The Sentry setup serves the same purpose, but unlike the Cores, they can be configured in Active/Active or Active/Standby configuration.
- A DNS or load balancer that controls the traffic to the Primary Core and Sentry. This “traffic controller” monitors the health of the other Core and detects when the Primary becomes unresponsive and begins routing traffic to the Secondary Core. This is how external traffic is controlled and routed to the Primary Core.
- The Secondary Core checks the status of the Primary through a process called “heartbeat”. This process is configured during HA Standby setup. This process detects if the Primary becomes unresponsive. When this happens it initiates the failover process. When a failover occurs, the Secondary attempts to become Primary, depending on what settings have been configured; it might stay as a Secondary or become Primary.
- The Secondary Cores periodically synchronizes with its Primary Core ensuring it has the latest changes as the Primary. The synchronization process frequency is configurable and it is automated.
- The ports used to communicate between Cores are ports 8443, 443 and 22 as outlined in the diagram. This intra-Core communication is essential for proper Core HA operation.

Copyright © 2021, Ivanti. All Rights Reserved. Privacy and Legal.

Three Core servers and two Sentry servers on two data centers

The example in this section describes one of the most complete and complex Core/Sentry HA Architectures with all related components.

FIGURE 4. THREE CORE SERVERS AND TWO SENTRY SERVERS ON TWO DATA CENTERS



The key components in this architecture include:

- A main data center hosting a pair of Cores. This pair of Cores is set up as Primary and Secondary. These two Cores serve as the main Core High Availability solution.
- Another data center for Disaster Recovery (DR), hosting a third Core in Secondary mode. The third Core serves as part of the DR configuration and it resides in the DR data center. This data center also hosts a Sentry to provide High Availability to the Primary Sentry.

Copyright © 2021, Ivanti. All Rights Reserved. Privacy and Legal.

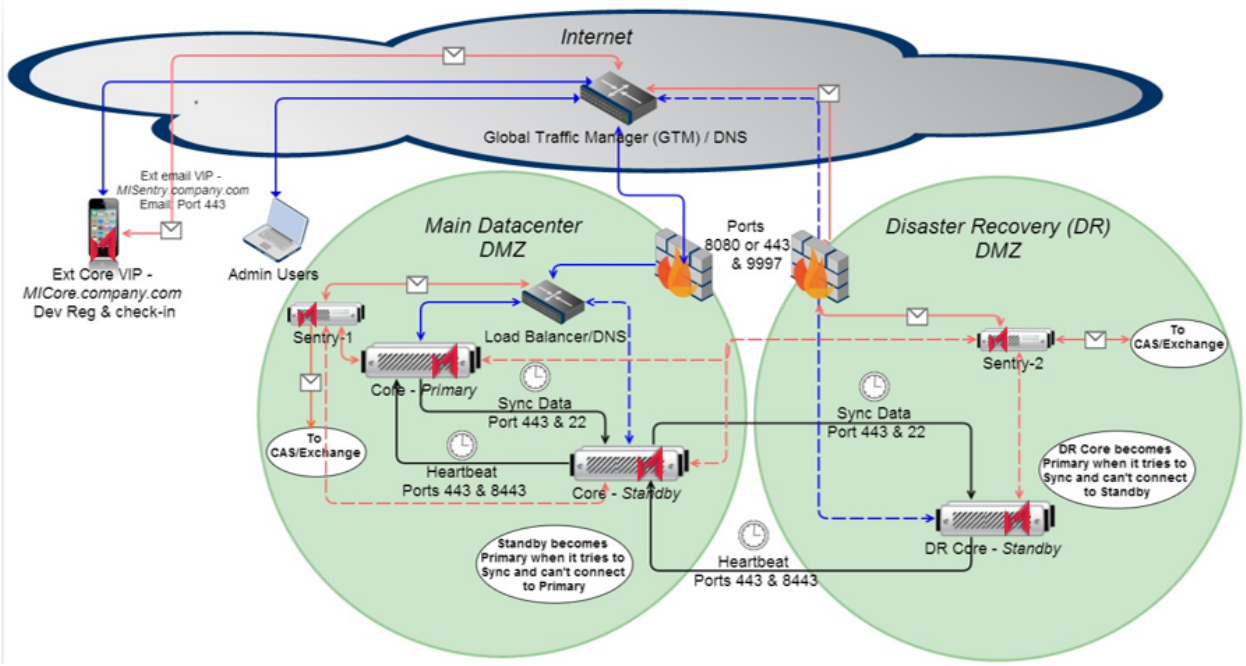
- A Global Traffic Manager (GTM) or DNS that controls the external traffic to the Main data center or the DR data center. This “traffic controller” monitors the health of the Primary Cores and detects when the Primary becomes unresponsive and begins routing traffic to the Secondary in the Main data center or DR data center in case of a Main data center failure. Within the data center, there is a load balancer, which takes care of monitoring the state of the Cores and routing the traffic accordingly. The same concept applies to the Sentry and a second Sentry can be installed in the Main data center to allow for redundancy within the same data center.
- The Secondary Core checks the status of the Primary through a process called “heartbeat”. This process is configured during HA Standby setup. This process detects if the Primary becomes unresponsive. When this happens it initiates the failover process. When a failover occurs, the Secondary attempts to become Primary, depending on what settings have been configured; it might stay as a Secondary or become Primary. In the case of the Core located in the DR data center, it sees the Secondary in the Main data center as its Primary Core and the failover process takes place between these two Cores.
- The Secondary Cores periodically synchronizes with its “Primary Core” ensuring it has the latest changes as the Primary. The synchronization process frequency is configurable and is automated.
- The ports used to communicate between Cores are ports 8443, 443 and 22 as outlined in the diagram. This intra-Core communication is essential for proper Core HA operation.

Copyright © 2021, Ivanti. All Rights Reserved. Privacy and Legal.

Load balancer setup

This section describes the recommended approach for Core to monitor, load balance, and set up automated detection of a Primary Core failure and failover to the paired Secondary Core.

FIGURE 1. LOAD BALANCER SETUP FOR CORE MONITORS



At a high level the load balancer should perform the following tasks:

1. Probe the status of the Primary server by issuing /status/status.html and receiving the “MOBILEIRON-STATUS: OK” message. This indicates a healthy Core. Any other response should be considered a failure.

Copyright © 2021, Ivanti. All Rights Reserved. Privacy and Legal.

2. Probe the status and state of the Secondary by issuing /hastatus.html and receiving the “MOBILEIRON-STATUS: OK (Mode: Standby)” response. The “OK” indicates the server is up and running. The “Mode: Standby” indicates the Core is in Standby mode and it is safe to failover traffic in case of a Primary Core failure. The other possible value for “Mode” is “Primary” which indicates the server is acting as Primary.

3. The load balancer should be set up to keep traffic routed to the Primary server and only failover traffic when the following conditions apply:
   - Primary probe fails and
   - Standby Core shows “Mode: Standby”

4. The failover should NOT resume traffic back to the Primary when the Primary becomes available (Manual Resume = Yes). Resuming traffic back to the Primary is a manual process.

Important: If the option to swap Core roles is implemented as part of the Core HA Recovery, the necessary changes must be made to the load balancer to match the Core HA architecture, that is, having the right monitors checking the correct Cores.

Copyright © 2021, Ivanti. All Rights Reserved. Privacy and Legal.

HA setup and configuration

Please consult with the Ivanti Professional Services Group or a qualified partner to enable the feature on the System Manager portal. After this feature has been enabled, you can continue with the configuration.

Procedure

1. Log into the System Manager.
2. Go to **Maintenance > HA Configuration** to open the **HA Configuration** window.
3. Modify one or more of the fields, as necessary. Refer to "HA Configuration window" below table for details.
4. Click **Save > Apply > OK.**

HA Configuration window

The following table summarizes fields and descriptions in the **HA Configuration** window:

TABLE 2. HA CONFIGURATION WINDOW FIELDS

Fields	Description
High Availability	This feature provides the ability to **Enable** or **Disable** HA. The default is **Disable**. The **Enable** radio button needs to be selected followed by **Save and Apply** in order to enable HA. Enabling and Disabling HA starts (enable) or stops the scheduling process.
Mode	This feature indicate the initial role this server will have in the HA configuration. The two options are: • **Primary** – This is the Core where all traffic and work will go to during normal operation.

Copyright © 2021, Ivanti. All Rights Reserved. Privacy and Legal.

TABLE 2. HA CONFIGURATION WINDOW FIELDS (CONT.)

Fields	Description
	• **Secondary** – This is the redundant Core server, which frequently checks the status of the Primary (heartbeat) and periodically replicates from it. The replication process is initiated and controlled by the Secondary server. This ensures the Secondary is up-to-date and ready to take over in case of a Primary failure.
Manage Trusted Hosts	This feature allows for the setup and configuration of the SSH keys between Core servers. The exchange of SSH keys is used to establish the trust relationship between servers by using public-key cryptography and challenge-response authentication. This step is fundamental for HA Cores to communicate with each other.
HA Status (Primary mode only)	This section provides details on the last recorded HA activity and only displays when **Primary** is selected for Mode. It provides the status of its pair system, its hostname/IP address, and the last time the **Secondary** has synchronized with it.
Node Status (Secondary mode only)	This option only displays when **Secondary** is selected for **Mode**. This feature indicates the current server status in relationship to the Core HA configuration. The default value of **Standby** indicates the server is a Standby server and is allowed to sync with its Primary server. The state of **Active** indicates the system was originally set up as **Standby**, but has been promoted to **Active**. This occurs when the Secondary server is not able to communicate with the Primary server and a failover event occurs OR when the **Active** button is manually selected. In either case, a Standby server in **Active** state does not sync with its Primary server. Only a Standby/passive server syncs with its Primary server.

Copyright © 2021, Ivanti. All Rights Reserved. Privacy and Legal.

Setting up the HA workflow

A proper Core HA setup configuration consists of the following steps:

Step 1 – Set up network components

Set up the network components by completing all the necessary changes to the network to allow the Standby Core server to come up and function as the Primary Core. Setup and configure network traffic components such as load balancer or DNS as described in this section.

TABLE 3. REQUIREMENTS FOR SETTING UP THE CORE SERVERS

Core Server	Collectivity	Software	Other
Primary	Bi-directional connection to Secondary Core over ports 8443, 443 & 22 in addition to standard Core FW requirements	Both Primary and Secondary servers must be running the same software release	Unique hostname for each Core
Secondary	Bi-directional connection to Primary Core over ports 8443, 443 & 22 in addition to standard Core FW requirements	Both Primary and Secondary servers must be running the same software release	Unique hostname for each Core

Copyright © 2021, Ivanti. All Rights Reserved. Privacy and Legal.

Step 2 – Enable HA on Primary server

Procedure

1. Log into the Primary Core System Manager Portal.
2. Go to **Maintenance > HA Configuration**.
3. Go to the **Service Configuration** section.
4. Go to the **High Availability** option and select **Enable**.
5. Go to the **Mode** option and select **Primary**.

Step 3 – Enable HA on Secondary server

Procedure

1. Log into the Secondary Core System Manager Portal.
2. Go to **Maintenance > HA Configuration**.
3. Go to the **Service Configuration** section.
4. Go to the **High Availability** option and select **Enable**.
5. Go to the **Mode** option and select **Secondary**.
6. Go to the **Node Status** option and select **Passive**.

Step 4 – Exchange SSH keys between servers

Procedure

1. Log into the Primary Core System Manager Portal.
    a. Select **Maintenance > HA Configuration**.
    b. Go to the **Service Configuration** section and click **Manage Trusted Hosts**.

Copyright © 2021, Ivanti. All Rights Reserved. Privacy and Legal.

2. Log into the Secondary Core System Manager Portal.
    a. Select **Maintenance > HA Configuration**.
    b. Go to the **Service Configuration** section and click **Manage Trusted Hosts**.
3. Go to the Primary server and copy the **Local Server SSH Key**.
4. Go to Secondary server.
    a. Go to **Add Trusted Host**.
    b. Paste the key from the Primary server into the **SSH Key** field.
5. Go to the Secondary server and enter the hostname or IP address of the Primary server in **Hostname/IP**.
6. Then go to the Primary server and copy the **Local server SSH key** and paste it into the **SSH key** field on the Secondary server.
7. Click **Add SSH Key** to add the key to the configuration.
8. Go to the Primary server and enter the hostname or IP address of the Secondary server in **Hostname/IP**.
9. Go to the Secondary server and copy the **Local server SSH key** and paste it into the **SSH key** field on the Primary server.
10. Click **Add SSH Key** to add the key to the configuration.
11. Click **Test Connection** on both servers to ensure both systems can communicate with each other.
12. Click **OK** to close the **HA - Test Connectivity** window.
13. Click **Save > Apply** on both the Primary and the Secondary servers.

IMPORTANT: Do not continue the setup/configuration if you do not see the **Test was successful** message on the **HA - Test Connectivity** window.

Copyright © 2021, Ivanti. All Rights Reserved. Privacy and Legal.

Step 5 – Configure and test HA sync on Secondary Core server

This step configures and tests the sync process testing the configuration of the Core HA components and if the replication is successful. This step assumes you are still logged into the System Manager Portal of the Secondary Core server. If you are not, log in now.

The initial sync will copies everything from one server to another while subsequent syncs only copy changes. Therefore, the initial sync will take more time than subsequent syncs.

Procedure

1. Go to the Secondary server.
2. Go to the **Secondary Configuration** section.
3. Select the Primary server from the **Hostname/IP** drop-down menu.
4. Click **Run Sync Now** to test all the components in the HA environment.
5. Click **Sync History** to see the sync execution details.
6. Wait for the sync process to complete then click **Refresh** to update the **Last Sync Status** fields.

Step 6 – Configure sync schedule and notifications

The final step is configuring the HA sync schedule and process notifications. You should only start this step when you are satisfied that the two servers are communicating and able to execute a sync properly. Then you can configure the schedule any remaining parameters.

Procedure

1. Go to the **Secondary Configuration** section.
2. Select a time in the **StartTime (GMT)** drop-down menu to set the starting time when the first replication is to occur. This time is in UTC time. Note that this value overwrites the scheduler every time you save this page.

Copyright © 2021, Ivanti. All Rights Reserved. Privacy and Legal.

3. Enter a value into the **Run Every** box to set how often the replication process executes. This value starts its reference point from the **StartTime** value. You can set the time in minutes, hours, or days.
4. Go to the **Fail Over Controls** section.
5. Enter the following values for:
    a. **Heartbeat Interval**: to control how often the Secondary server tries to connect to the Primary server to check it is alive.
    b. **Consecutive Heartbeat failures before failover**: to control how many consecutive connection attempts to the Primary can failed before attempting to promote the Standby server to Primary (active).
    c. **Re-check interval**: to define the amount of time between connection attempts when the previous attempt resulted in connection failure. This is the retry wait time between attempts.
6. Indicate what happens when the Primary is unavailable and a failover should occur by selecting one of the options for **If Failure Detected**:
    a. **Keep in Standby**: this option does not allow the Secondary to become Primary. The failover process gets initiated, notifications are completed, but the Secondary stays in standby mode.
    b. **Promote to Primary**: this option allows the Secondary to become Primary if the Primary becomes unavailable.
7. Go to the **Notification Settings** section and make the following selections:
    - **Email**: Enter a valid email address where the notifications should go to.
    - **Notify On: Sync Completion** (optional): When selected, the system generates a notification email every time the sync process executes and completes successfully.
    - **Notify On: Failure / This server becomes primary** (optional): When selected, the system generates an email notification every time the sync process executes and fails and/or when a Core HA failover event occurs.
8. Click **Save and Apply**.

Step 7 – Test and validate the Core HA environment

Use this test plan to help you test your Core HA systems.

Copyright © 2021, Ivanti. All Rights Reserved. Privacy and Legal.

TABLE 4. TEST AND VALIDATION PLAN FOR CORE HA

Task	Description	Steps	Expected Results	Notes
1	Check System Connectivity	On Primary server, go to **Manage Trusted Hosts > Test Connection** (against the Secondary Core) On Secondary server, go to **Manage Trusted Hosts > Test Connection** (against the Primary Core)	Successful results from both tests.	
2	Sync systems manually	On Secondary server, go to **Select primary** from the **Primary Core Hostname** menu. Click **Run Sync Now**.	In **Sync History**, all tasks show completed with no errors reported.	
3	Verify Standby Core updated correctly	On Secondary server through the CLI Enable mode, start tomcat: `>service tomcat start` Log on through the Admin portal and confirm the device count matches that in the Primary Core.	The device count is exactly the same as the Primary.	
4	Test Email Notification	On Secondary server, under **Notification Settings**, enter a valid email and click **Test**	A test email is sent by the Secondary server.	
5	Confirm Heartbeat Process	On Standby server go to **Troubleshooting > Logs select MICS > Apply**. In **View Module Logs** select **HighAvailability**.	The heartbeat activity shows in the **HighAvailability** window.	

Copyright © 2021, Ivanti. All Rights Reserved. Privacy and Legal.

Atlas configuration

To replicate Atlas data as part of a HA configuration, you must enable Atlas on both the Primary and Secondary servers.

Procedure

To enable Atlas on the Primary server:

1. Go to the Primary Atlas server (https://<Atlas-server>:8443/mics>).
2. Log into the System Manager Portal.
3. Go to **Settings > Services**.
4. Select **Enable** for **Atlas** and click **Apply**.

Procedure

To enable Atlas on the Secondary server:

1. Go to the Secondary Atlas server (https://<Atlas-server>:8443/mics>).
2. Log into the System Manager Portal.
3. Go to **Settings > Services**.
4. Select **Enable** for **Atlas** and click **Apply**.

Copyright © 2021, Ivanti. All Rights Reserved. Privacy and Legal.

HA Core software upgrade procedures

The procedure below describes the required steps to upgrade the Core servers in an HA environment. This procedure is critical because the Core HA replication process has built-in checks to enforce that replication can only occur between Cores with the same software version. Replication will not occur if they are not at the same software release level. This procedure assumes you are logging into the System Manager Portal of both the Primary and Secondary Core servers. If you are not, do so now.

Procedure

1. Disable HA on the Secondary Core.
2. Ensure that any currently running replication process is allowed to complete before upgrading the Core.

   Do not proceed unless the **Last Sync Status** has a date/time in the **Completed:** field and **Success** is displayed for **Result**.
3. Disable the load balancer from detecting any of the HA Cores from becoming unavailable and failing over.
4. Upgrade the Secondary and validate the upgrade.
   - Make this server Primary so you can download the Core upgrade software (allowing the server to connect to support portal)
   - Upgrade and validate the upgrade.
   - Return the server mode back to Secondary.
5. Upgrade the Primary Core and validate the upgrade.
6. Enable HA on the Secondary.
7. Perform test and validation on the Core HA environment. Complete the steps outlined in the "Step 7 – Test and validate the Core HA environment" on page 21 section.
8. Review and update the rest of the HA configuration parameters, especially the **StartTime** and **Run**

Copyright © 2021, Ivanti. All Rights Reserved. Privacy and Legal.

Every parameters, as these will most likely need to be updated.

9. Re-enable the load balancer.

At this point, the HA environment is back to normal operation with all Cores running on the same software release version.

Copyright © 2021, Ivanti. All Rights Reserved. Privacy and Legal.

HA command line interface (CLI) set

High Availability has the following commands available through the CLI under enable mode:

TABLE 5. HA COMMAND LINE INTERFACE SET

Command	Description
failover mode primary	This command changes the Core mode to Primary. This has the same effect as having the system automatically failover when the Primary becomes unavailable. This command is not limited by any other command or Core state and can be re-executed, achieving the same results. A system in Primary mode will not replicate with any other Core
failover mode secondary	This command changes the Core mode to Secondary. This has the same effect as making the same change through the System Manager Portal, but does not restart the replication scheduler. It is recommended to only use this command for upgrade or troubleshooting purposes. The System Manager Portal HA configuration version of this functionality should be used for normal HA operations.
failover sync <remote host address>	This command executes the Core replication process on demand. This command is identical to the **Run Sync Now** option in the System Manager Portal. This command is intended to replicate Core environments either after recovering from a failover or for troubleshooting purposes. The System Manager Portal HA configuration version of this functionality should be used for normal HA operations.
failover client-connection disable	This command prevents new device connections. This command is intended to prevent new connections while the system is going through maintenance.
failover client-connection enable	This command allows new device connections. This command is intended to reverse the disable client-connection command.

Copyright © 2021, Ivanti. All Rights Reserved. Privacy and Legal.