FORTRA

Globalscape EFT v8.2.0 User Guide

202406250836

Table of Contents

High Availability Overview

EFT High Availability (HA) solution can protect your critical business processes and ensure that crucial file transfer systems are always on, and that employees, customers, and business partners experience seamless availability of critical applications and information.

EFT with HA can:

- Maintain availability through any planned or unplanned outage
- Increase stability and flexibility by implementing multiple nodes of EFT for load balancing
- Enhance throughput and better meet important SLAs by deploying multiple nodes of EFT to allow the collective environment to use more available resources
- Improve scalability with the ability to share common configurations across nodes, eliminating the challenge of having multiple servers set up with different configurations

EFT's active-active deployment provides HA using multiple instances of EFT and a load balancer for non-stop availability of your network. And unlike active-passive failover clusters, all of the nodes in EFT's active-active deployment are put to work in production—with no standby hardware, and no clustering software.

Interoperable with Common Load Balancers

With EFT HA, you can control spikes in network traffic, minimize scalability limitations, and maximize the efficiency of large and complex environments. Globalscape's high availability solution is compatible with most major load balancers.

In addition, Globalscape is a member of the F5® Technology Alliance Program, providing a proven managed file transfer solution for interoperability of F5 BIG IP® Local Traffic Manager™ (BIG-IP LTM®) with Globalscape's managed file transfer platform, Enhanced File Transfer™ (EFT™) with High Availability (HA), to secure enterprise data at rest and in transit, without interruption.

What's the difference between active-active and active-passive load balancing?

- An active-active cluster is typically made up of at least two nodes, both actively running the same kind of service simultaneously. The main purpose of an active-active cluster is to achieve load balancing. Load balancing distributes workloads across all nodes to prevent any single node from getting overloaded. Because there are more nodes available to serve, there will also be a marked improvement in throughput and response times.
- Like the active-active configuration, active-passive also consists of at least two nodes. However, as the name "active-passive" implies, not all nodes are going to be active. In the case of two nodes, for example, if the first node is already active, the second node must be passive or on standby. The passive (a.k.a. failover) server serves as a backup that's ready to take over as soon as the active (a.k.a. primary) server gets disconnected or is unable to serve.

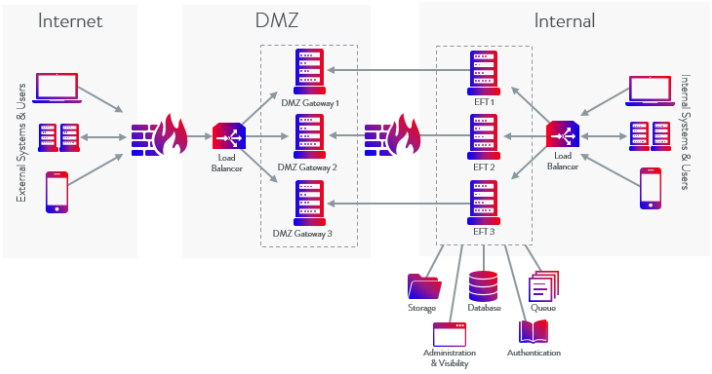
Cluster Types Supported

Cluster Type	Supported?
Traditional Active-Active cluster in same LAN	Yes
Primary and DR Site Active-Active cluster (configuration kept mostly in sync)	Yes
Active-Active cluster deployed in the cloud as a static cluster; Support with Windows tools and guidance, but no support on cloud infrastructure or services	Yes
Active-Passive cluster using Microsoft Clustering Services	Yes
Active-Passive cluster using unicast or multicast; Requires email confirmation from customer to account manager that they will be using in failover capacity only	Yes
Active-Active cluster deployed in the cloud as a dynamic, autoscaling cluster; General guidance and recommendations, but no active support	No
Distributed Active-Active cluster (GLBS or Geocluster)	No

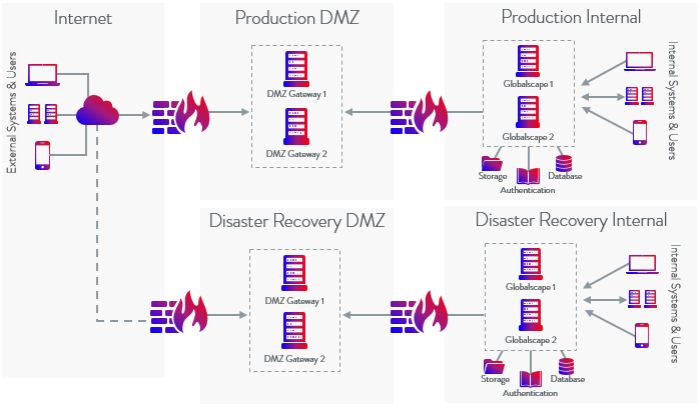
HA FAQ

EFT can be configured in an active-active cluster configuration, known as EFT High Availability (HA). In an HA deployment, two or more EFT installations can be configured in an active-active cluster with a shared configuration. EFT acts as its own cluster manager and requires a network load balancer (NLB) to distribute incoming protocol traffic. EFT HA nodes process file transfers at the network level as the NLB directs traffic to it, and can process Folder Monitor and Timer Event Rules in a round-robin fashion (that is, executing the event actions on the first node, then the second, and so on until it comes back to the first node in the list). An example 3-node deployment is shown below. Having at least 3 nodes allows you to take down one node for maintenance and still have 2 active nodes for HA.

> **NOTE:** If you attempt to access an EFT node while another node is currently being administered, you are presented with an error.



You can also install EFT HA in a DR environment:



For best results, before beginning an HA installation, please review all of the information below and refer to the references linked.

Consider the facts below before creating an HA cluster

Installation:

- To install EFT as part of a high availability (HA) cluster, refer to Active-Active HA Cluster--Installing or Upgrading the Server.
- Copy the \web\ folder from the EFT installation folder to the cluster share (%ClusterShare%)
- For true business continuity, you should have **a minimum of three nodes**, so that when one server goes down, nodes on other servers continue to process transactions in high availability, meeting those stringent uptime requirements.
- When EFT is running in HA mode and sharing a network resource, you must run the EFT server service with a Log On account that can access that shared network resource.
- Nodes should be brought online one at a time to avoid getting the nodes out of sync. A node can get out of sync if 1) on administrator login, if the configuration in memory does not match the configuration file on the shared drive; or 2) when the node fails to receive a configuration change message. In either case, the EFT server service will restart and load the latest configuration.
- You can configure additional Sites on non-HA nodes with other authentication to cooperate with the HA-clustered Sites. For example, you could use an AD-authenticated Site on a non-HA node for domain user uploads and downloads, but use the Globalscape, AD, or LDAP-authenticated Sites in the HA cluster for load balancing Event Rules that take place on the files transferred. The non-clustered AD Site would transfer files using the HA shared configuration path as the Site root folder or using virtual folders, while the Event Rule Actions take place on the files on the HA Site.
- If you are using DMZ Gateway®, each EFT must have its own DMZ Gateway; the DMZ Gateway cannot be shared among nodes.
- HA nodes must listen on all IP addresses, rather than each listening on a specific IP address. Do not change the Listening IP address for an HA node. A registry value can be configured to have a different IP address for each node. Refer to Knowledgebase article #11225 for details.
- The default EFT options for DoS/Flood Protection and Login Security settings are designed for each IP to have a single user's activity. These settings can cause EFT to ban the load balancer's IP or intermittently block its activity when all user connections are using the load balancer's IP. **These settings should be disabled when using a load balancer. It's also a good idea to** set an IP access rule to allow the load balancer IP on EFT just in case the settings are accidentally enabled later.

- To make an HA node a stand-alone server, you must uninstall EFT and then reinstall as a stand-alone server. Please contact Sales, Support, or Professional Services for assistance in migrating any existing non-HA deployments to a corresponding HA configuration.
- A stand-alone server can be converted to an HA node. Refer to Promote Stand-Alone EFT to HA node for instructions.
- On HA installations, the EFT server service is configured to restart upon failure on the **Recovery** tab of the service's properties. (Non-HA installations are configured to "take no action" upon failure.)
- You cannot backup in standby and restore in HA or vice versa. You can run restore on any node in the HA cluster. You can restore shared data, node-specific data (listening IP address, DMZ Gateway settings, registration) or both. When the restore process begins, other nodes stop with -1 error. This triggers them to be restarted by Windows Service Manager, at which point those other nodes will wait for restore operation to complete. Once the restore has completed on one of the nodes, the other nodes that had been waiting will proceed with loading configuration. After the restore completes, the node that did restore also restarts in the same way. Thus, all nodes in the cluster have restarted with restored configuration up and running.
- HA Shares should NOT reside on a Unix Samba share. EFT HA environments have a difficult time recovering from network failures that occur on the master node's system.
- The log naming convention now allows EFT to store logs on the cluster shared drive.

Configuration:

- If using SAML (Web SSO) in an HA environment, SAML needs to have the IDP's public key saved in the HA shared drive.
- After installation, the shared path (for example, \\myserver\HA_config) is shown on the Server's **High Availability** tab > **Config Path**. The shared configuration file PATH cannot be changed in the administration interface. KB #11260 describes how to change the path in the registry.
- On an HA-clustered Site, the **Usr** folder (for example, **\inetpub\EFTRoot\mySite\Usr**), which contains the users' folders, is stored in the shared configuration path.
  (for example, **\\x.x.x.x\inetpub\EFTRoot\mySite\Usr\username**)
  - When using Encrypted folders, you can only encrypt files in the directory hierarchy of the Site's root folder. Make sure that the Site root folder on the **Site > General** tab is pointing to the correct path. That is, if your HA config

drive is on **D:\HAConfig\,** you should edit the site root folder to point to **D:\HAConfig\InetPub\EFTRoot\MySite**.

- When creating an LDAP Site on an HA node, the **User list refresh interval** setting is disabled. The user list must be synchronized on all nodes at the same time.
- A local configuration path for each node must also be specified for each node for local caching. When configuration changes are made to SSH, trusted SSL certificates, OpenPGP key materials, and AML files (Advanced Workflow Module workflows), those files are cached locally, and uploads are safely synchronized to the shared config path. The other nodes then update their local cache from the central location. Thus, the central share always contains the current version of those files. When, for example, a new SSL Cert or AML file is created, the creator directly adds/modifies the files on the network share then tells the other nodes that the file was modified/created and they need to update their local cache (that is, copy the file to their local ProgramData directory).
- When using HA, **you need to specify a unique location (local) for the log files**. This is for troubleshooting purposes (to know on which node the issue occurred). Also, having two nodes write to the same file causes issues with file locking, which will cause data in the logs to be lost. For visibility into node status, enable cluster logging. Logging.cfg has new logging options specifically for HA. When configuring RSA in an HA environment be sure to have the **sdconf.rec** file stored locally for each node. Each node MUST have its own copy of sdconf.rec.

MSMQ:

- Proper communication among nodes requires:
  - Change notifications are communicated throughout the cluster using MSMQ multicast by default. If this is problematic, there is a MSMQ unicast option. Refer to the Unicast topic for details.
  - When using MSMQ Mulitcast:
    - Network adapters on all HA nodes that enable Reliable Multicast Protocol (for the adapters that provide the route between EFT HA nodes)
    - All nodes must be able to send and receive multicast messages, which requires they be on the same LAN subnet
  - L2 Switch between physical computers that host an EFT HA node (physically or virtually) must enable traffic on the LAN segment between HA nodes. Typically, this means enabling IGMP Snoop and IGMP Querier; however, complex deployments (including VPN or MPLS networks between nodes) might require packet encapsulation, such as GRE, to allow packets to operate properly between nodes.

    - Firewalls between HA nodes, and on the computers hosting EFT, must allow the traffic (both multicast and unicast) to pass in and out of the computers. (Windows Firewall will automatically enable the proper ports when enabling MSMQ.). For physical switches, be sure there are no packet filtering rules that prevent packets of type 113 to flow between nodes.
    - If you do not want to use unicast or multicast on your network, you should create a new VLAN for EFT communications. Refer to Knowledgebase articles #11221 and #11276 for more information about MSMQ, VLAN, and EFT.
- HA mode of operation for supports IPv6 addressing for inbound and outbound connections. The message queue addressing of nodes within the cluster is not supported on IPv6 addresses. Message queue addressing uses NetBIOS names, not IP addresses, and could be tied to IPv4 on the local LAN subnet that all nodes share.
- When configuration changes are made to SSH, trusted SSL certificates, OpenPGP key materials, and AML files (Advanced Workflow Engine workflows), those files are cached locally, then safely synchronized to the network share. The other nodes then update their local cache from the central location. Thus, the central share always contains the current version of those files.

Event Rules:

- Folder Sweep and archive should be enabled on load balanced Folder Monitor rules to clean up and notify on any events that occur when the primary Event Rule monitor goes down. It is possible to lose some events between when the primary goes down and the next node takes over.
- The "Run On One of" feature in Event Rules currently only supports computer (NetBIOS) names. Refer to Event Rule Load Balancing for more information about the "Run On One of" feature.
- Do not start or provision a new node immediately after making changes to the Event Rule configuration. Give the system at least 30 seconds to process and synchronize the configuration changes.
- When operating in HA mode, the Timer and Folder Monitor Event Rules will execute on ALL of the nodes of the cluster unless you specify at least one High Availability node on which to operate. **Define a default node** for load balancing, as described in High Availability Tab of a Server.
- Because all of the nodes fire **User Account Created** events, each node also runs all **User Account Created** event rules. Therefore, to avoid sending multiple email notifications (for example), you should add an "**If node name**" Condition to the rule so that only one node sends the notification (or performs other Actions).
- In every HA cluster, there will be a "Master" node that performs the Event Rule load balancing assignments.

- Any node may be master; if a master node goes offline, another node will take over as master. Whichever node declares master first becomes master. A node doesn't take over as master until at least one load-balanced Event Rule exists on the system. Prior to a load-balanced Event Rule's existence, all nodes will claim to be master. A master can go down if, for example, MSMQ is stopped, or the network can no longer communicate with the master or the EFT server service for some reason goes down.
- Every 10 seconds each node broadcasts a heartbeat to communicate that they are alive and online. This serves two purposes: 1) Notifies that the master node is up, if the master goes down, then a new node will resume master responsibilities and broadcast that they are now master; 2) Notifies the cluster that the node is online and should be included to handle load-balanced Event Rules.

Auditing and Reporting:

- The ARM reports identify nodes based on computer name. If the node's computer name changes, ARM will see it as a new node and not associate it with the old computer name. (ARM installs an additional set of reports in a "High Availability" folder. These reports are a duplicate of the existing reports, except they report based on Node name.)

COM API:

- The API was updated to include HA-specific calls.
- Only one node at a time is allowed to use the administration interface or COM connection. That is, you cannot administer more than one node at a time. (However, more than one administrator can administer the SAME node at the same time, just as in non-HA configurations.) Attempts to administer more than one HA node at a time will prompt an error on nodes other than the first.

Non-High Availability Mode vs. High Availability Mode

Function	Non-High Availability mode	High Availability mode
Startup	Searches for configuration, tries backup files to handle broken configuration, etc. Always creates "clean" configuration if none is loaded.	Loads only the configuration files from the shared location. *Creates "clean" configuration only if no configuration present. Fails to start if cannot load configuration.*
Shutdown	Updated configuration with latest settings	Does not update configuration.
Authentication managers	GS, NTAD, ODBC, LDAP	Globalscape, NTAD, and LDAP
User database refresh	Allowed	Not allowed
PCI cleanup (administrator/ user remove/ disable for inactivity and send password expiration notifications)	Nightly timer + Every time server deals with user/administrator (user/administrator connection, exposing user/administrator to GUI/COM etc.)	Nightly timer
Client Expiration	Every time when server deals with user (user connection, exposing user to GUI/COM etc.)	Nightly timer
Turning off Autosave and using "ApplyChanges" via COM	Allowed	Not allowed
GUI/COM connection	Always allowed	Only one node at a time is allowed to serve GUI/COM connection
Saving changes made by administrator to FTP.CFG	Background task accumulating changes and saving settings	Only one node can be administered at a time. As soon as you log off of the node on which you've made changes and then log on to second node, those changes would be updated on the second node.
Server restore from backup	Allowed	Allowed
Trial state	Registry	Registry

Function	Non-High Availability mode	High Availability mode
OTP passwords for clients	Allowed	Not allowed
Legacy password hashes	Allowed	Not allowed
User lock state	Continues after service restart	Breaks after service restart
Invalid login history	Continues after service restart	Resets on service restart

Active-Active HA Cluster—Installing or Upgrading the Server

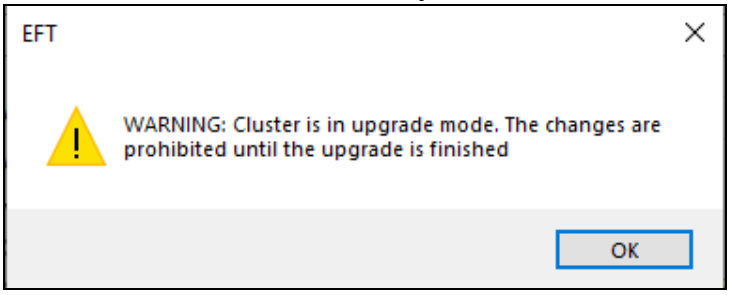
Refer to the Globalscape Knowledgebase article #11271 for information about installing or upgrading the server in an active-active, high availability (HA) configuration.

See also Upgrade HA Nodes with Zero Downtime.

Upgrade HA Nodes with Zero Downtime

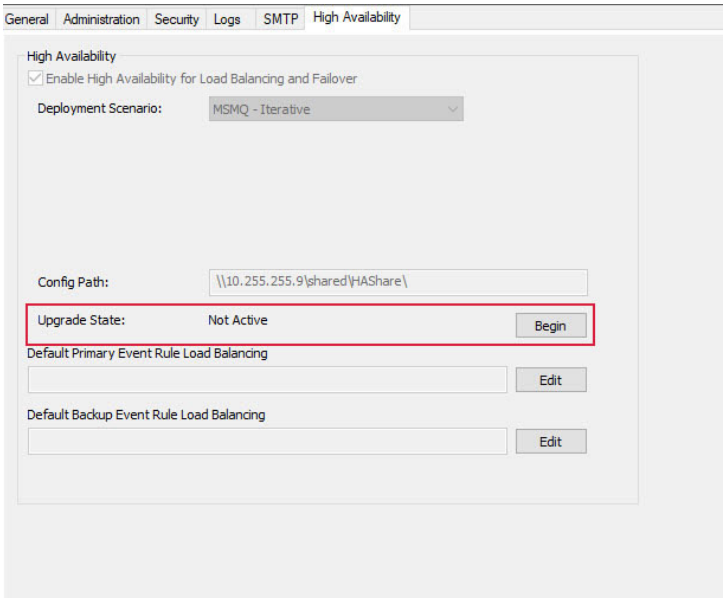
EFT administrators can upgrade their HA nodes with "zero downtime," meaning you can upgrade a cluster without stopping all nodes at the same time.

NOTE: The warning prompt "Cluster is in upgrade mode" is thrown when a user makes a configuration change and then refreshes the EFT administration interface while the server is in "read only" mode.



The Server > HA tab is used to manage the cluster upgrade state

- The button will be displayed as **Begin** when upgrades have not been started.



- The button will be displayed as **End** when upgrades have been completed. Administrators can execute the End state at any point, but nodes that have not been upgraded will fail to start until they are upgraded.
- When enabling the upgrade, EFT administrators will be prompted with a warning that the transition and change cannot be undone.
- The warning prompt should reference details to review documentation.

To upgrade all of the HA nodes at that same time—without stopping any nodes

In the EFT administration interface, on the **Server > High Availability** tab, you must manually put the cluster into the **Upgrade** state, upgrade each node one by one, then after the upgrade has completed successfully, manually transition the cluster back to **Normal** state.

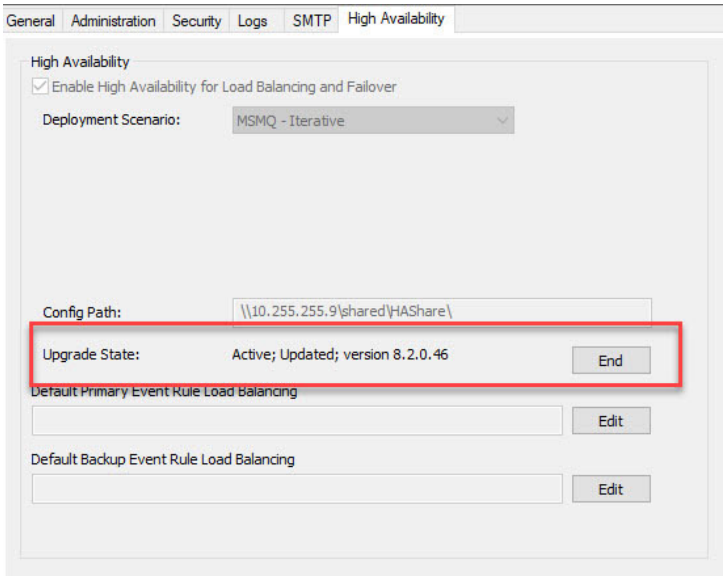
- **Upgrade State**. Allows simultaneous updates to nodes that have an **Outdated** version.

  Upgrade state limitations:

  - Backup/Restore feature is prohibited
  - No configuration changes are allowed during the Upgrade state, except moving the cluster back into the Normal state
- **Normal State**. Requires all nodes have the same version. Upon transition to Normal, the Upgrade State will display “Not Active”

  Normal state limitations:

- Loading old nodes is prohibited. EFT will not start when the nodes are not upgraded.
- You will be unable to transition from Upgrade to Normal when an upgrade has not been completed. This only affects nodes that have not been upgraded.
- EFT will prevent COM API from transitioning from Upgrade to Normal state. A proper response code/error will be presented
- EFT will prevent REST API from transitioning EFT from Upgrade to Normal state. A proper response code/error will be presented.
- When upgrade has ended (at the global level or a node that was upgraded), for any nodes that were not upgraded, the EFT Server Service will stop and fail to start until they are upgraded.
- Transition state from Upgrade to Normal are global, which affects all nodes within the cluster, so administrators should only end upgrades after all nodes have been upgraded.



Upon successful HA upgrade to a new version, the **High Availability** tab's **Upgrade State** will display “Active; Updated; x.x.x.x”, where x.x.x.x is the newer version of EFT (the version to which EFT was upgraded).

Normal state to Upgrade state

- All nodes are Outdated. Admin logins to any of the nodes and enter the cluster into the "Upgrade" state (with GUI/COM/REST).

    - EFT admins shall be able to transition from Normal to the Upgrade state via our EFT administrator interface High Availability tab.
    - EFT admins shall be able to transition from Normal to the Upgrade state via COM. Appropriate error message(s) will be provided if enabling Upgrade state on any node when the cluster is already configured for Upgrade.
    - EFT admins shall be able to transition from Normal to the Upgrade state via REST API. Appropriate error message(s) will be provided if enabling Upgrade state on any node when the cluster is already configured for Upgrade.
    - The ability to start the Upgrade state is a global setting; once set at any node, all nodes will report the same state.
    - Upgrade state should be used ONLY when upgrading to a new version of EFT and not when an EFT administrator forgets to upgrade a node in the cluster from a previous upgrade.
- Once configured for Upgrade, the cluster is in Upgrade state, and you can upgrade each node one by one.
- Outdated nodes write audit data to local *.sql files only; Each node will have their own local *.sql file.
- Updated nodes write into ARM database; If no SQL schema changes exist, there is a post process in the Normal state that will migrate the data from the *.sql file into the ARM database.
- Uses Web folder appropriate to node's version.
    - The nodes' version should match the same version.
    - Shadowfax folder will be copied from \\<HASharedPath>\Web to \\<HASharedPath>\UpgradeFromx.x.xx (where x.x.x.x is the older EFT version); This …\UpgradeFromx.x.x.x will be used for Outdated nodes.
    - Shadowfax folder \\<HASharedPath>\Web will be used for Updated nodes and will be updated with the first upgraded node.
- Outdated nodes allow admins to login in read-only mode. EFT administrators will not be able to make any configuration changes to EFT.
- Updated nodes allow admins to login in read-only mode AND move the cluster to Normal state.
- Configuration changes write in deltas only (there could be only internode mx messages).
    - Changes are not saved into the configuration files, ServerConfig.db and SiteConfigXX.db files. (Part of the advanced property HAFullConfigDumpIntervalMins.)

        - Internode mx messages are communications between EFT nodes (no QA validation).
    - Unknown internode messages are ignored; These are internal nodes for Outdated nodes that may not know about freshly added internode messages in Updated nodes.
    - Cluster Management subsystem doesn't send any messages and ignores all received messages.
    - ERLB subsystem doesn't delegate to run any event rules - all rules are processed with Master node only. Client connections to the server will be processed without any restrictions during the Upgrade state.

Upgrade state to Normal state

- All nodes are Updated. Admin will need to log into any node and end the upgrade which will switch it back to Normal state (with GUI/COM/REST)
    - EFT admins shall be able to transition from Upgrade to Normal state via our EFT Admin UI
    - EFT admins shall be able to transition from Upgrade to Normal state via COM; Appropriate error message(s) will be provided if enabling Normal state on any node when the cluster is already configured for Normal
    - EFT admins shall be able to transition from Upgrade to Normal state via REST API; Appropriate error message(s) will be provided if enabling Upgrade state on any node when the cluster is already configured for Upgrade
    - The ability to end the upgrade state is a global setting; once set at any node, all nodes will report the same state. Nodes that were not upgraded and the upgrade state has ended, will fail to start.
    - Nodes that were not upgraded after the Upgrade state ends, now Normal, can be upgraded later and will be visible and continue to work in the cluster
- Normal state limitations: Loading old nodes is prohibited. EFT will not start when the nodes are not upgraded

COM has several methods to manage cluster upgrade state

- Enum: HAUpgradeNodeState
    - NotInUpgrade = 0,
    - OldNode,
    - NewNode
- Method: ICIServer::GetHAUpgradeNodeState; Retrieves node upgrade state
- REST has two endpoints to manage cluster upgrade state:

- Get HA node upgrade state; GET /server/ha/upgrade
- Changes entire HA cluster state to the upgrade or normal state: POST /server/ha/upgrade

High Availability Message Queuing

EFT High Availability (HA) installations for active-active clustering can use Microsoft Message Queuing (MSMQ) to share configuration and other data among nodes. All MSMQ messages are two-fish encrypted. MSMQ Broadcasting is used to communicate that a change has been made to the cluster. When an administrator makes a change to the configuration (adds/disables a user, creates an event rule, etc.), the node will broadcast a message to all nodes in the cluster that the configuration file has been modified, and to read in the changes. The broadcasting system is also used to notify other nodes when AML, SSL, SSH, and OpenPGP files are created or modified.

Almost all EFT data and operations are synced between all of the nodes, except for the following node-specific data/operations:

- Trial state
- DMZ Gateway settings
- Pending certificates
- Site Start/Stop
- Temporary user lockout
- Invalid login attempts history (when the limit is reached the user is disabled cluster-wide)
- Temporary IP ban (permanent ban is cluster-wide)
- File Lock (hiding the files being uploaded from other connections)

When EFT is installed for active-active clustering, the installer determines whether MSMQ is enabled, and enables it if it is not. EFT HA relies on the MSMQ service for two important functions:

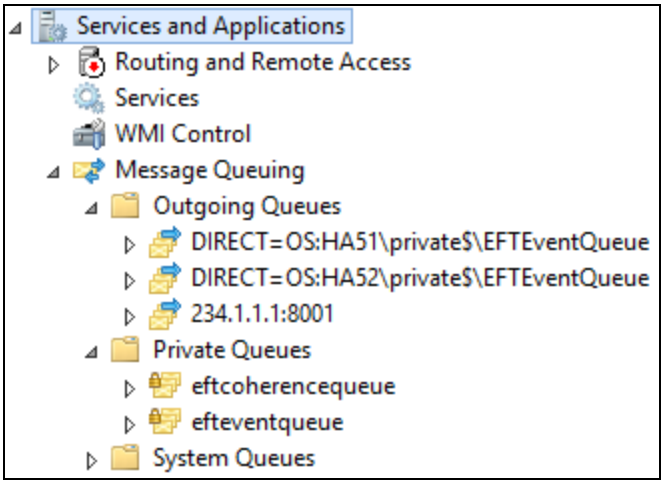
- Synchronize changes made to the cluster configuration (eftcoherencequeue)
- Load balance Event Rules (efteventqueue)

The Event queue and the Coherence queue are created at service start and destroyed when the service is stopped. They appear in the Computer Management console, under **Services and Applications > Message Queuing**.

To view Message Queuing

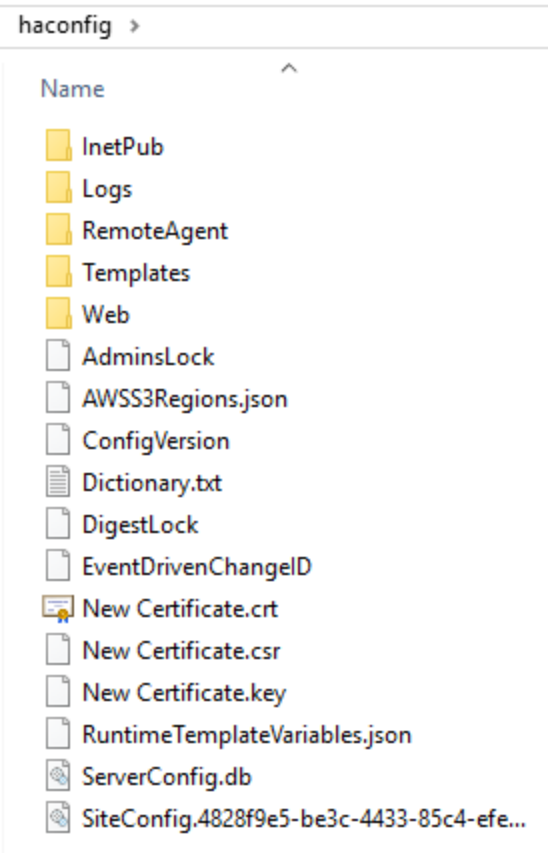
1. Right-click the **Start** icon, then click **Run**.
2. In the **Run** dialog box, type `compmgmt.msc` then press ENTER.
3. Expand the **Services and Applications** node.

4. Expand the **Message Queuing** node.



HA Nodes Configuration Files

When installing EFT in an HA configuration, you must specify a shared configuration location (which must be accessible by all of the nodes in the HA cluster). The files that all of the HA nodes will share are saved and updated in the shared configuration directory.



- An Advanced Property was added to improve performance when configuration is copied to the cluster share. When off, `HAFullConfigDumpIntervalMins = 0` or is not defined, the full configuration is copied on each change. Add the Advanced Property and set it between 1-10 minute intervals to reduce network traffic.
- A local configuration path for each node must also be specified for each node for local caching. When configuration changes are made to SSH, trusted SSL certificates, OpenPGP key materials, and AML files (Advanced Workflow tasks), those files are cached locally, and uploads are safely synchronized to the shared config path.

The other nodes then update their local cache from the central location. Thus, the central share always contains the current version of those files. When, for example, a new SSL Cert or AML file is created, the creator directly adds/modifies the files on the network share then tells the other nodes that the file was modified/created and they need to update their local cache (that is, copy the file to their local ProgramData directory).

- When using HA, **you need to specify a unique location (local) for the log files**. This is for troubleshooting purposes (to know on which node the issue occurred). Also, having two nodes write to the same file causes issues with file locking, which will cause data in the logs to be lost. For visibility into node status, enable cluster logging. Logging.cfg has new logging options specifically for HA. When configuring RSA in an HA environment be sure to have the **sdconf.rec** file stored locally for each node. Each node MUST have its own copy of sdconf.rec.
- When using Encrypted folders, you can only encrypt files in the directory hierarchy of the Site's root folder. Make sure that the Site root folder on the **Site > General** tab is pointing to the correct path. That is, if your HA config drive is on **D:\HAConfig\,** you should edit the site root folder to point to **D:\HAConfig\InetPub\EFTRoot\MySite**.

Node Discovery Using Shared Storage for HA Clusters

The EFT HA Resiliency project replaces MSMQ UDP Multicast with iterative MSMQ TCP Unicast (that is, message is sent to each the nodes individually over MSMQ TCP). In addition to the improved reliability of TCP, removing the dependence of UDP Multicast simplifies deployment on cloud infrastructures. However, a new method of discovering of nodes is required, and can be accomplished using the shared storage (with file locking semantics) that EFT already requires.

ActiveNodes.json File

For EFT HA clusters, node discovery is accomplished through an ActiveNodes.json file on the existing shared storage. All access to ActiveNodes.json is performed in exclusive access mode (so no one else changes or reads the file while accessing it). The ActiveNodes.json file is the “one source of truth” for which nodes are connected when operating in any HA mode. To determine where to send the nodes, the ActiveNodes.json must be consulted. Nodes will be responsible for adding and removing themselves from the list. The "Event Rule Master" (that is, the node that is holding the lock on Master.lck) is responsible for periodically removing nodes that failed without removing themselves.

ActiveNodes.json file example:

```
{
 "Nodes": [{
 "Name": "EFT_NODE_1",
 "IP": ["192.168.1.101"],
 "ActivationTime": "2017-03-30T15:20:30-06:00"
},{
"Name": "EFT_NODE_2",
"IP": ["192.168.1.102"],"ActivationTime": "2017-03-30T15:20:35-
06:00"
}]
}
```

Sending “Multicast” Messages

To determine which nodes to send the “Multicast” messages to, ActiveNodes.js is consulted.

1. If ActiveNodes.json has been modified (Check both last modified Date/Time and file size)

2. Open ActiveNodes.json in “OPEN_ALWAYS” and “Exclusive Access” mode (locks file from access by others)
3. Retry up to 10 times, waiting 100ms between each, if file is locked by another process
4. If cannot open the file after all retries, EFT shall write an error to EFT.log and shall indicate that cached values will be used
5. Read file contents
6. Close ActiveNodes.json
7. Parse JSON file contents (optimization, if this is done after the file is closed)
8. Store the cached values for future use
9. Send the message to each node listed in the JSON (or cached value if could not open file)

Node Adding Self

When a node starts up it updates the ActiveNodes.json file, adding itself.

- Open ActiveNodes.json in “OPEN_ALWAYS” and “Exclusive Access” mode (locks file from access by others)
    - Retry up to 300 times, waiting 100ms between each, if file is locked by another process
    - If cannot open the file after all retries, EFT shall write an error to EFT.log and Windows Event Log, and for abject failure leading to restart
    - Service should exit in FAILURE mode
- Read file contents
- Parse JSON file contents
- If this node is not already in list (normally it shouldn’t be, but if current node had failed and restarted it may)
    - Add entry for this node to the JSON
    - Write Updated JSON File Contents (truncating file as needed)
- Close ActiveNodes.json
- Store the cached values for future use

Node Removing Self

When a node shuts down it updates the ActiveNodes.json file, removing itself.

- Open ActiveNodes.json in “OPEN_ALWAYS” and “Exclusive Access” mode (locks file from access by others)

    - Read file contents
        - Retry up to 3 times, waiting 100ms between each, if file is locked by another process
        - If cannot open the file after all retries, EFT shall write an error to EFT.log and Windows Event Log
- Parse JSON file contents
- If this node is in list (it should be)
    - Remove entry for this node
    - Write Updated JSON File Contents (truncating file as needed)
- Close ActiveNodes.json

Event Rule Master Removing “Expired” Nodes

Periodically the Event Rule Master updates the ActiveNodes.js file, removing any expired nodes.

- Open ActiveNodes.json in “OPEN_ALWAYS” and “Exclusive Access” mode (locks file from access by others)
    - Retry up to 4 times, waiting 100ms between each, if file is locked by another process
    - If cannot open the file after all retries, EFT shall write a warning to EFT.log
- Read file contents
- Parse JSON file contents
- Remove expired nodes from list, (i.e. all of the following conditions are met)
- Use the “decay” mechanism in EFT (2x missed heartbeat from a node means the node is down)
- If any nodes removed, write Updated JSON File Contents (truncating file as needed)
- Close ActiveNodes.json

Server Drain, Maintenance, and Auto-Restart

Draining the server provides for graceful shutdown and transition between EFT and HA nodes. Drain mode halts any NEW incoming connections and HA Event Rule activity for all Sites on a specific node. Transfers and events in progress would continue to operate until completion. When in drain mode, the node relinquishes mastership and the master stops delegating events to the drained node. If a certain node/server is expected to restart (for example, after Windows updates), administrators can preemptively manually drain the node before the expected restart. In this way, all traffic is routed to and kicked off on the other nodes not paused. After the Windows update/maintenance of the node, traffic and event rule activity will resume normally.

Limitations

- The WTC will deny uploads for any pending transfers on nodes that are in drain mode; in-progress transfers will be permitted to complete (unless drain timeout occurs).
- When EFT enters drain mode, it denies all future connections. scClient may do some GET requests when a file completes upload during drain to obtain a refreshed directory listing. EFT will deny these requests and issue an error. These errors can be ignored; all uploads in progress will continue until completion or until drain timeout occurs.

To enable drain mode manually

1. In the EFT administration interface, click **File > Drain Server** (or use the COM API method, **ICIServer::DrainServer**). A message appears asking if you want to restart the server service after draining.
2. Click **Yes** to automatically restart; click **No** if you want to restart later.

Auto-Drain

Auto-Drain takes effect automatically when an HA node detects that it is out of sync and a restart is required. The server engages Drain mode (waits for transfers and events to complete) and then automatically restarts the node to regain sync with the cluster.

The administrator can configure the time before drain takes effect, and the time it will allow for transfer and processes to complete, and can disable drain completely, if wanted.

Refer to https://kb.globalscape.com/KnowledgebaseArticle11391.aspx for the overrides pertaining to Server Drain, Maintenance, and Auto-Restart

Unicast

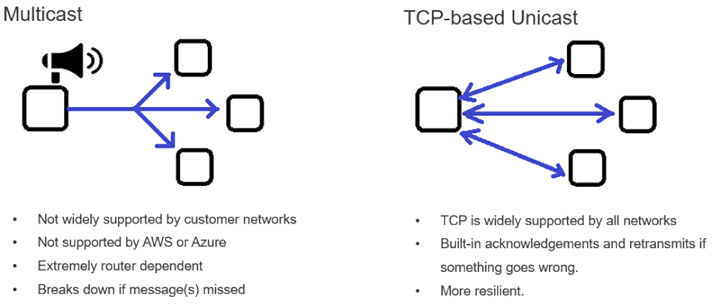
Iterative TCP Unicast (that is, message is sent to each the nodes individually over TCP) can be used instead of MSMQ UDP Multicast. In addition to the improved reliability of TCP, removing the dependence of UDP Multicast simplifies deployment on cloud infrastructures. Discovering of nodes can be accomplished using the shared storage (with file locking semantics) that EFT already requires.

The advanced property below must be enabled to use Unicast:

```
"ClusterCoherenceQueueMsmqType":"msmq-iterative"
```

Unicast provides:

- HA support for environments that do not support MSMQ Broadcasting
- Ability for HA to work in a vMotion environment
- Improved support for node-out-of-sync detection and availability.

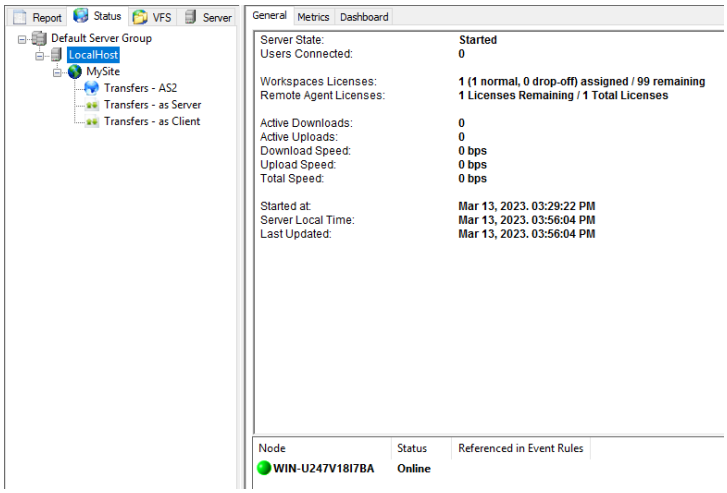


Viewing Server or Node Status

In the administration interface, you can view the status of EFT in real time, such as number of users connected, average speed, and so on. You can view Server status on the **Status** tab or on the Server node's **General** tab.

To view status on the Status tab

1. In the administration interface, connect to EFT and click the **Server** tab.
2. On the **Status** tab, click the **Server** node. EFT's statistics appear in the right pane.



3. In an HA cluster, you can see the nodes and their status at the bottom of the **Status** viewer.
    - **Online**: EFT server service is up and communicating via Heartbeat to the rest of the node in the cluster
    - **Master**: Same status as **Online**, however this node is designated as the **Master** node for Event Rules Load balancing.
      Only one node can be show as Master in the list of the nodes from the cluster. This status only is displayed if you have at least one Event Rule enabled and configured to run in more than one node from the cluster. If this status exists, the Master node will create an exclusive file lock onto **MasterNodeLock** file in the HA config shared folder.
    - **Offline**: EFT server service is down; no communication via Heartbeat is performed
    - **Unknown**: A node name is being reference in at least one Event Rule; however, this node name is not part of the cluster.

To view status on the Server tab

1. In the administration interface, connect to EFT and click the **Server** tab.
2. On the **Server** tab, click the **Server** node.
3. In the right pane, click the **General** tab. EFT's statistics appear in the right pane.
    - **Server status**: Displays "Service is started" or "Service is stopped." You can also stop and start the EFT service on this tab.
    - **Start date/time**: Displays the date and time that the EFT service was last started.
    - **Uptime**: Displays the length of time that the EFT service has been running since it was last started.
    - **Last modified time**: Displays the date and time that EFT was last modified.

- **Last modified by**: Displays the username of the user who last modified EFT.
- **Active sessions**: Displays the number of users who are currently logged in to EFT.
- **Active uploads**: Displays the number of uploads in progress.
- **Active downloads**: Displays the number downloads in progress.
- **Average speed**: Displays the average transfer speed.
- **Workspaces licenses**: Displays the number of licenses used and number licensed (allowed)
- **Web clients licenses**: Displays the number of licenses used and number licensed (allowed)

Logging for HA Nodes

The EFT cluster administrator has access to the EFT logs from the shared drive and doesn't have to navigate to the nodes to view them.

On the **Server > Logs** tab, you can specify a shared log location and prefix the log filename with a node ID. However, if you upgrade an HA environment from EFT version 7.4.x or earlier, the **Log File Settings** path is not updated; it keeps the old path where the cluster administrator had to navigate to the different nodes to view the log files. Instead, you can manually provide the shared configuration path with the server node name. (for example, \\ipaddress\share\log\%SERVER.NODE_NAME%\). Once you've changed the Log File Settings path, a folder for each node should appear in the shared path, and new log files will be saved there.

Log files are not backed up in the Backup and Cleanup Rule, because that is intended as a configuration backup. Once you've changed the Log File Settings path, you can create your own Event Rule to back up log files.

HA-Related Knowledgebase Articles

The following Knowledgebase (KB) articles can assist with configuring EFT in an HA environment:

- 11146 - Installing or Upgrading EFT in a Failover, Active-Passive Cluster
- 11175 - Load balanced Folder Monitor events fail to process files in an HA clustered environment
- 11197 - Message appears stating that the node is out of sync and that the service needs to be restarted
- 11214 - Performance Tuning EFT HA Native mode
- 11215 - NetApp NAS tuning to work with EFT in HA mode
- 11216 - How to Mirror an EFT HA (Active-Active) Cluster to another EFT HA (Active-Active) Cluster Manually
- 11221 - EFT HA can Reliable Multicast and MSMQ to communicate between nodes
- 11224 - Inbound F5 Load Balancing for an EFT HA Cluster using DMZ Gateways
- 11225 - Node-specific IP address on HA site
- 11245 - Folder Monitors and Shared Storage
- 11260 - Changing the path to the shared configuration folder for EFT with HA
- 11271 - Installing and Upgrading EFT in an Active-Active HA Cluster
- 11276 - Integrating EFT HA in a network with a Cisco switch
- 11269 - EFT banning load balancer IP address
- 11302 - Why do clients connect to EFT showing the IP of the Netscaler load balancer?
- 11314 - Load Balancers and IP addresses
- 11391 - Registry settings for HA server drain and coherence
- 11407 - Using SSL/TLS termination at F5 Load Balancer
- 11483 - Unable to set refresh interval on LDAP/AD site when EFT configured in HA
- 11486 - Event rule sending duplicate emails in HA environment for User Account Created event
- 11504 - Cluster-Related Advanced Properties
- 11542 - Promote Stand-Alone EFT to HA node