



This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

SIP Voicemail HA Deployment Guide






SIP Voicemail 8.1.1

Table of Contents

Welcome	3
HA Scenarios Overview	4
Traditional HA Architecture	7
Traditional HA Failure Scenarios	9
Traditional HA Events and Scripts	12
Solution HA Architecture	14
Solution HA Failure Scenarios	16
Solution HA Events and Scripts	20
Known Issues and Recommendations	22
Traditional HA Task Summary	25
Configuring MCP Shared Voicemail Storage (Windows)	27
Configuring MCP Shared Voicemail Storage (Linux)	28
Solution HA Task Summary	30
IP Address Takeover (Windows)	32
Creating Virtual IP control scripts (Windows)	33
Testing the Virtual IP control scripts (Windows)	37
Creating Application control scripts (Windows)	39
Creating Genesys Applications for the control scripts	42
Creating the Alarm Reaction scripts	44
Creating the SCS Alarm Conditions	46
Testing Alarm Conditions	48
IP Address Takeover (Linux)	49
Creating Virtual IP control scripts (Linux)	51
Creating Application control scripts (Linux)	53
Creating Genesys Applications for the control scripts	42
Creating the Alarm Reaction scripts	44
Creating the SCS Alarm Conditions	46
Testing Alarm Conditions	48
Document Change History	63

Welcome

SIP Voicemail HA Deployment Guide

Overview	Procedures
<p>Genesys SIP Voicemail can be deployed in two HA models: Traditional HA and Solution HA.</p> <p>For an overview of the basic HA architecture, and a description of these HA models, see:</p> <p> Overview For details about Traditional mode (all GSVM components are deployed in an HA pair), see:</p> <p> Traditional HA Architecture For details about Solution HA mode (VM SIP Server and VM Server components are deployed in HA mode), see:</p> <p> Solution HA Architecture</p> <p>Document Change History</p> <p>For details about what has changed in the SIP Voicemail HA Deployment Guide, see the Document Change History.</p>	<p>Find task summaries and procedures for both the Complete Install and the Manual Install of SIP Voicemail HA:</p> <p> Solution HA Task Summary</p> <p> Traditional HA Task Summary</p>

HA Scenarios Overview

Genesys SIP Voicemail (GSVM) can be deployed as a highly-available (HA) pair of voicemail servers, providing redundancy for the communication paths between the server components, as well as for the database information associated with the voicemail messages themselves.

Overview

GSVM supports two HA scenarios: Traditional HA and Solution HA.

Traditional HA

In this scenario, all the GSVM components are configured in a HA pair. The failure of either the VM server or VM SIP server collocated on the primary host will result in a coordinated switchover of the VM Server and VM SIP Server in the backup host. The switchover for GVP components is defined by the individual HA deployment modes of GVP and Premise SIP Server.

Key Points About Traditional HA

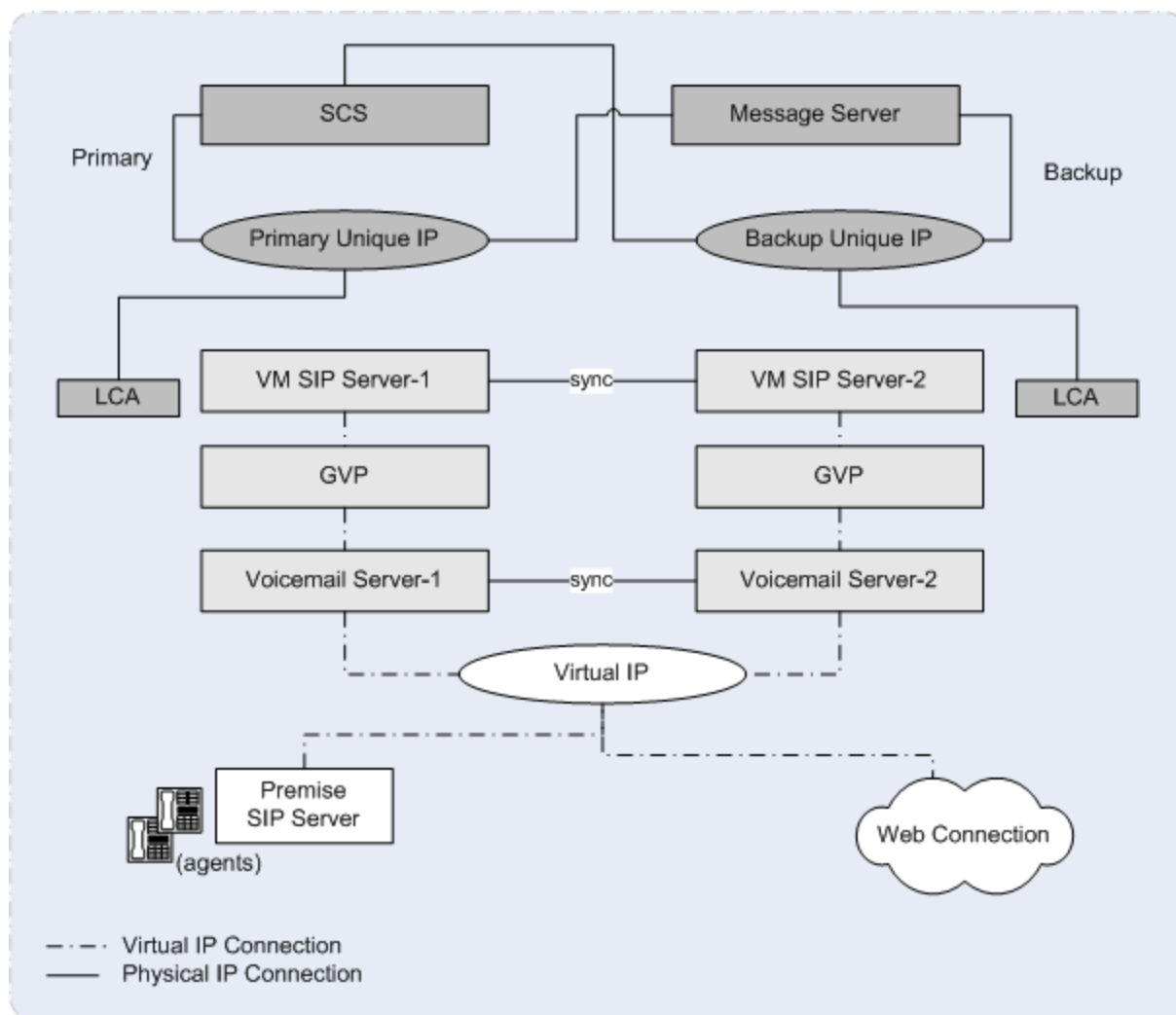
- Primary and backup VM Servers are deployed collocated with the VM SIP Server.
- RM and MCP are deployed in a standard HA configuration as supported and described in Genesys Voice Platform documentation.
- MCP instances should be configured to place their recordings in a shared directory where the VM Server can access them.

Solution HA

In this scenario, the failure of a single component results in a coordinated switchover of the entire solution from primary to backup server. This approach provides a more robust synchronization -- useful, for example, for Reporting purposes.

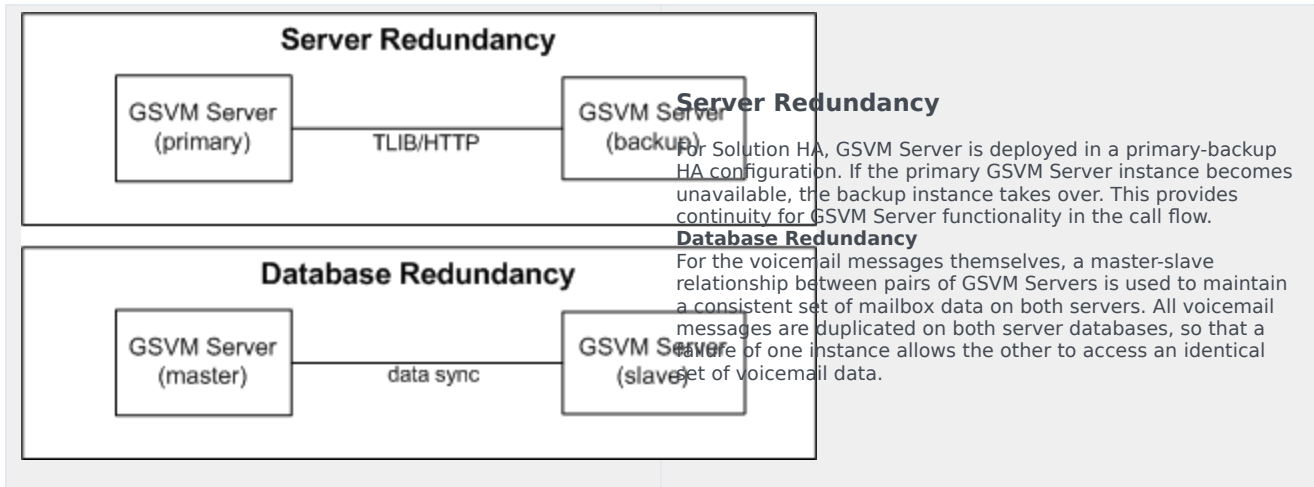
IP Interface Architecture

The following diagram shows the connections between Genesys components and the Voicemail host.



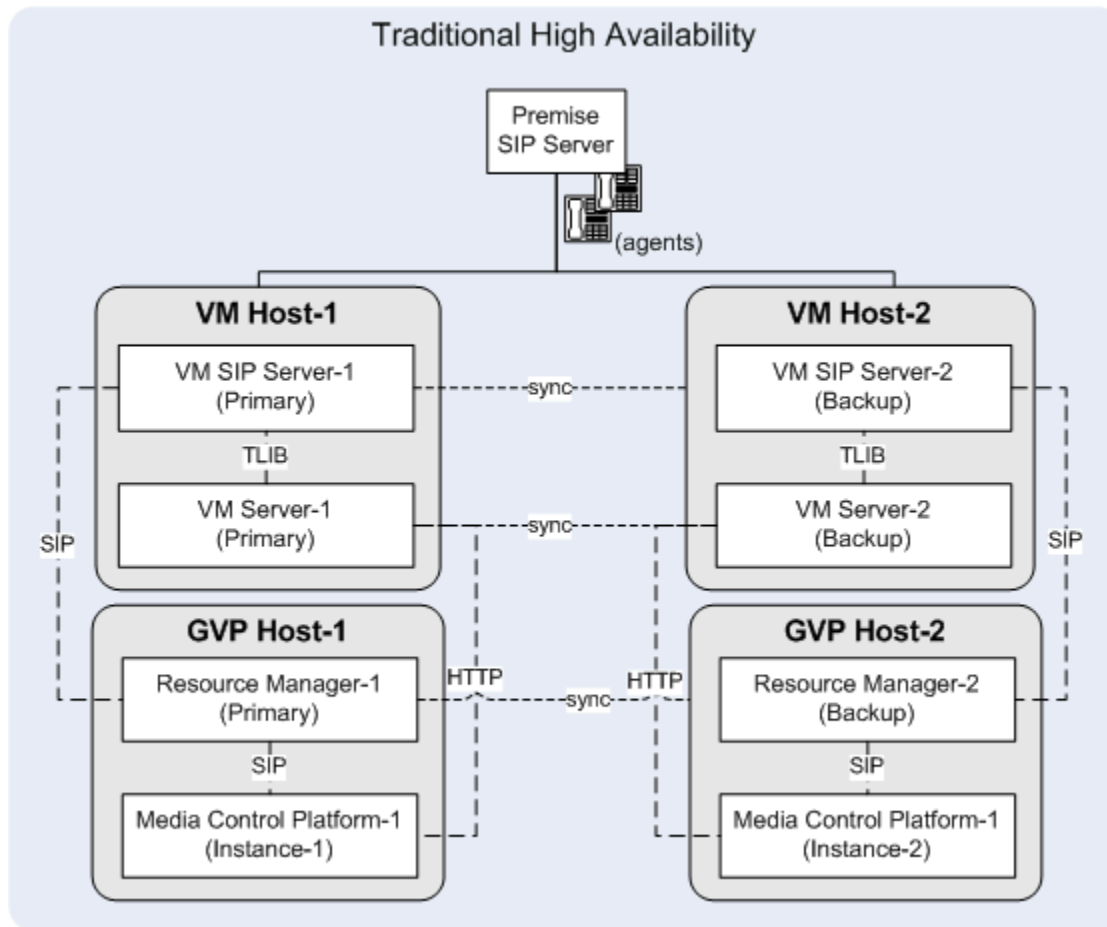
Redundancy Types

GSVM in an HA deployment supports redundancy for both server traffic and for database information related to the voicemail messages themselves.



Traditional HA Architecture

The basic HA architecture involves deploying the GVP components on one host and deploying the VM SIP server and VM server on another host. All components are deployed in HA mode. The following diagram shows the components and connections of a basic HA deployment.



Key Notes:

- GSVM Server is deployed into an existing SIP Server/GVP HA configuration.
- Resource Manager uses an active/standby HA configuration.
- The cross connections between the VM SIP Server and Resource Manager instances across the two sites are not shown in this diagram.
- The Voicemail SIP Server HA instances are configured in a primary/backup HA setup.
- The Voicemail Server is configured as both a primary-backup HA pair, and master-slave for database synchronization.

Configuration

For Traditional HA, you manually add HA GSVM servers to an existing Genesys HA deployment (SIP Server and GVP in an HA configuration).

For configuration details, see:

 [Traditional HA Task Summary](#)

Traditional HA Failure Scenarios

When an application failure happens, two types of switchover scripts are invoked: Virtual IP (VIP) control, and application control. You must add VIP control scripts on the host where the voicemail components are installed, and application control scripts on the host where SCS is running. SCS executes a script based on alarm reactions to execute the VIP and Application control scripts. For details about the events, scripts and third-party applications, see: [Traditional HA Events and Scripts](#). The following are failure scenarios for Traditional HA, where a Primary GSVM SIP Server or Primary GSVM Server failure results in a coordinated switchover of the components, from primary to backup server, using the MLCMD utility, which is installed with SCS. The switchover for GVP components is defined by the individual HA deployment modes of GVP and Premise SIP Server.

Primary Host Failure

The following items describe the switchover mechanism using the MLCMD utility when the primary VM host components fail while running in primary mode.

VM SIP Server-1 Fails

- The VM SIP Server-2 generates a 4563 event.
- Alarm reaction scripts are triggered to react to the 4563 event from the VM SIP Server-2.
- VIP takeover scripts are linked to only the primary and backup VM SIP Servers on the VM host.

MLCMD Reaction:

1. The VM Server-2 is switched to primary mode.
2. VIP is enabled on VM Host-2.
3. VIP is disabled on VM Host-1.
4. There is no impact on the GVP Host.

VM Server-1 Fails

- VM Server-2 generates a 4563 event.
- Alarm reaction scripts are created to react to the 4563 event from VM Server-2.
- VIP takeover scripts are linked to only primary and backup VM SIP servers.
- The 4563 event from VM Server-2 switches VM SIP Server-2 to primary mode in the VM Host.
- The 4563 event from VM SIP Server-2 executes the VIP control scripts in the VM Host.

MLCMD Reaction:

1. The VM SIP Server-2 is switched to primary mode.
2. The 4563 event from VM SIP Server-2 executes the VIP control scripts.
3. There is no impact on the GVP Host.

Resource Manager-1 Fails

When Resource Manager-1 fails, the actions are defined by the HA deployment for Resource Manager-1. Refer to the deployment guide of the Resource Manager.

Media Control Platform-1 Fails

When Media Control Platform-1 fails, the actions are defined by the HA deployment for Media Control Platform. Refer to the deployment guide of the Media Control Platform.

Entire VM Host Fails

- Any of the following events automatically triggers the switchover, because the alarm reaction is configured for each component individually.
- The 4563 event from VM SIPServer-2 or VM Server-2.
- VM SIP Server-2 & VM Server-2 automatically become primary.

Note: In this scenario there are chances of getting an IP conflict problem. See [Known Issues and Recommendations](#) for more information.

MLCMD Reaction:

1. VIP is enabled in VM Host-2.
2. VIP will be disabled in VM Host-1.

Backup Host Failure

If any component on the backup VM host fails while running in Primary Mode, the event 4563 is generated from the components and alarm reaction scripts are executed in the reverse direction to switch the primary VM host components to primary mode.

Switchover Cases

During manual switchover scenarios, the 4563 event is generated from the VM Server-2 and the reaction scripts are the same as when the VM Server-1 fails. For example, an administrator switches the VM Server-2 to primary mode using the SCS switchover option. When an administrator switches the VM SIP Server-2 to primary mode using the SCS switchover option, the 4563 event is generated from the VM SIP Server-2 and the reaction scripts are the same as those for the VM SIP Server-1

failure.

Key Notes:

- Failure of the components on the GVP host are handled using their native failure mechanism.
- Failure of the GVP host components do not affect the current state of the VM host components.
- When an application fails while running in backup mode, no alarm reaction scripts are triggered. The application does not have HA until the backup server is operational.
- If the Premise SIP server goes down, no alarm reaction scripts are triggered.
- MLCMD must use the application DBID. In Management Framework release 8.1.1, MLCMD can use the application name.
- VIP takeover scripts are only linked to the event for only the VM SIP Server-1 and VM SIP Server-2. This prevents duplicate execution of the takeover scripts. If any application fails on a VM host, all applications are switched. Alarm reaction scripts are triggered for the events from different applications for the single switchover scenario.
- VIP takeover scripts are connected to the SIP Server so that the VM SIP Server port is bonded to the VIP immediately when it becomes primary. If the VIP is not enabled immediately after SIP server becomes primary, the SIP server is in an unavailable state.

For example, when VM SIP Server fails, if the VIP scripts are connected to the VM Server, VM SIP server-2 is in an unavailable state until the 4563 event from the VM SIP Server-2 triggers the switchover of VM Server-2 and the 4563 event from VM Server-2 triggers the VIP enable script. VM SIP server-2 remains in an unavailable state until the Virtual IP is enabled in the host that becomes primary.

Note: This issue does not apply to the other applications, because the port and VIP bonding of the other applications do not happen at the switchover, when the applications become primary.

Traditional HA Events and Scripts

Third-Party Applications

Eight third-party applications are created for enabling and disabling VIP and switching the components.

1. TP_PRIMARY_VIP_UP: Associated with VIP_UP script on VM HOST-1
2. TP_PRIMARY_VIP_DOWN: Associated with VIP_DOWN script on VM HOST-1
3. TP_Backup_VIP_UP: Associated with VIP_UP script on VM HOST-2
4. TP_Backup_VIP_DOWN: Associated with VIP_DOWN script on VM HOST-2
5. TP_VMServer-1_SWITCHOVER: Associated with the script SC_APPLICATION_SWITCHOVER that switches VM Server-1 to primary mode
6. VMServer-2_SWITCHOVER: Associated with the script SC_APPLICATION_SWITCHOVER that switches VM Server-2 to primary mode
7. VM-SIPServer-1_SWITCHOVER: Associated with the script SC_APPLICATION_SWITCHOVER that switches VM SIP Server-1 to primary mode
8. VM-SIPServer-2_SWITCHOVER: Associated with the script SC_APPLICATION_SWITCHOVER that switches VM SIP Server-2 to primary mode

Note: Enabling and disabling the Virtual IP corresponding to the GVP hosts must be handled by their native method.

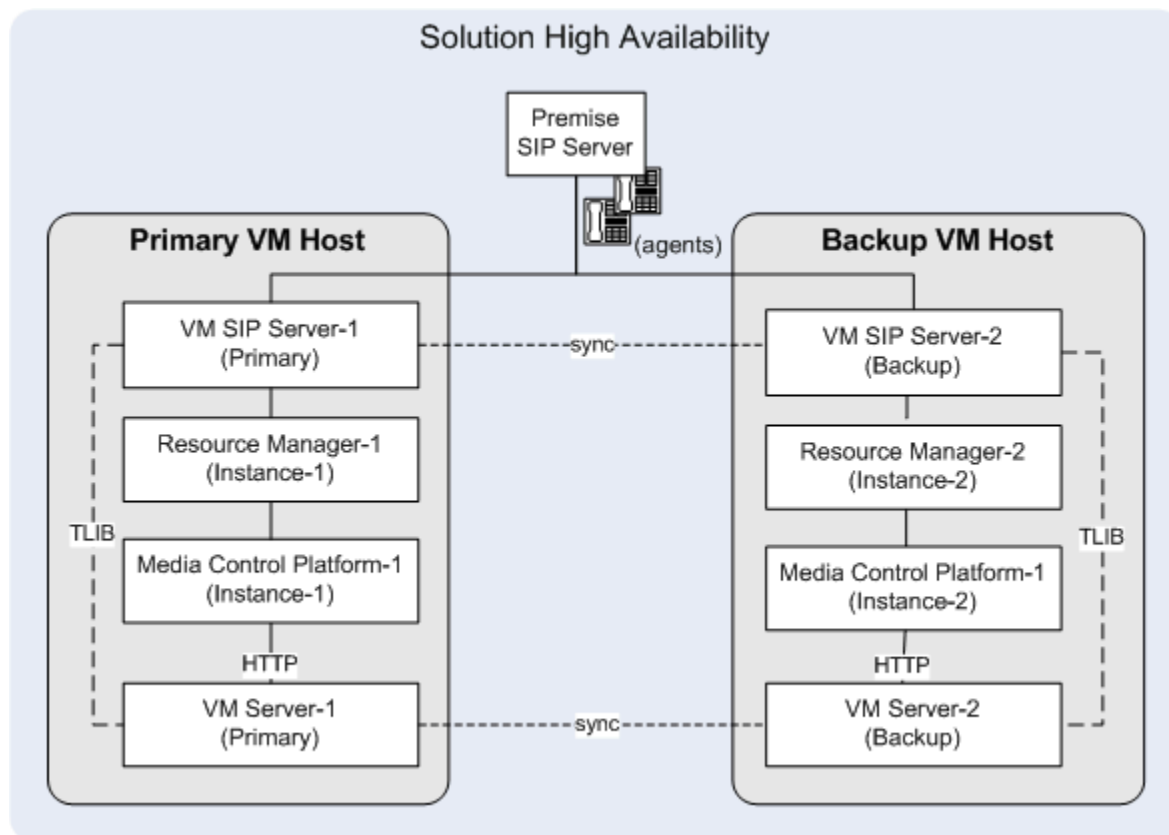
Events and Scripts Association Table

Log Event ID	Component	Script	Third-Party Application
4563	VM SIP Server-1	SC_VIP_DOWN on VM Host-2	TP_BACKUP_VIP_DOWN
		SC_VIP_UP on VM Host-1	TP_PRIMARY_VIP_UP
		SC_APPLICATION_SWITCHOVER to switch VM Server-1 to primary mode	TP_VMSERVER-1_SWITCHOVER
4563	VM SIP Server-2	SC_VIP_DOWN on VM Host-1	TP_PRIMARY_VIP_DOWN
		SC_VIP_UP on VM Host-2	TP_BACKUP_VIP_UP
		SC_APPLICATION_SWITCHOVER to switch VM Server-2 to primary mode	TP_VMSERVER-2_SWITCHOVER

Log Event ID	Component	Script	Third-Party Application
4563	GSVM Server Primary	SC_APPLICATION_SWITCHOVER to switch VM SIP Server-1 to primary mode	TP_VM- SIPSERVER-1_SWITCHOVER
4563	GSVM Server Backup	SC_APPLICATION_SWITCHOVER to switch VM SIP Server-2 to primary mode	TP_VM- SIPSERVER-2_SWITCHOVER

Solution HA Architecture

The Solution HA architecture involves deploying all GSVM components on two separate hosts. One host acts as the primary server, the other as the backup server. The following diagram shows the components and connections of a basic HA deployment.



Key Notes:

- The only applications that are deployed in an actual HA configuration are the VM SIP Servers and the Voicemail Servers. Resource Manager (RM) and Media Control Platform (MCP) are deployed in standalone mode on each of the VM hosts.
- In the case of failure of any component on one host, the whole solution is switched over to the other box. Alarm reaction scripts perform this coordinated switchover.
- HA for VM SIP Server and Voicemail server is facilitated using Virtual IP (VIP) takeover, not Microsoft Windows Network Load Balancing (NLB).
- In the event of a switchover, a service interruption is acceptable, but should be kept to a minimum.

Configuration

For Solution HA, use the Complete Install tool to deploy the overall GSVM HA solution from scratch. For configuration details, see:

 [Solution HA Task Summary](#)

Solution HA Failure Scenarios

When an application failure happens, two types of switchover scripts are invoked: Virtual IP (VIP) control, and application control. You must add VIP control scripts on the host where the voicemail components are installed, and application control scripts on the host where SCS is running. SCS executes a script based on alarm reactions to execute the VIP and Application control scripts. For details about the events, scripts and third-party applications, see: [Solution HA Events and Scripts](#). The following are failure scenarios for Solution HA, where one component failure results in a coordinated switchover of all components, from primary to backup server, using the MLCMD utility, which is installed with SCS.

Primary Host Failure

The following items describe the switchover mechanism using the MLCMD utility when the primary host components fail while running in primary mode.

VM SIP Server-1 Fails

- The VM SIPServer-2 generates a 4563 event.
- Alarm reaction scripts are triggered to react to the 4563 event from the VM SIP Server-2.
- VIP takeover scripts are linked to only the primary and backup VM SIP servers.

MLCMD Reaction:

1. The VM Server-2 is switched to primary mode.
2. VIP is enabled on the backup VM host.
3. VIP is disabled on the primary VM host.

VM Server-1 Fails

- VM Server-2 generates a 4563 event.
- Alarm reaction scripts are created to react to the 4563 event from VM Server-2.
- VIP takeover scripts are linked to only primary and backup VM SIP servers.
- The 4563 event from VM Server-2 switches VM SIP Server-2 to primary mode.
- The 4563 event from VM SIP Server-2 executes the VIP control scripts.

MLCMD Reaction:

1. VM SIPServer-2 is switched to primary mode.
 2. The 4563 event from VM SIP Server-2 executes the VIP control scripts.
-

Resource Manager-1 Fails

- The RM-1 generates 5091 and 5064 events.
- Alarm reaction scripts are created to react to the 5064 and 5091 events from the RM-1.
- The 5091 and 5064 events from the RM-1 switch the VM SIP Server-2 to primary mode.
- The 4563 event from the VM SIP Server-2 triggers the enable and disable VIP scripts and switches the VM Server-2 to primary mode.
- The event from the RM-1 triggers only the switchover of the VM SIP Server-2.
- The VM SIP Server-2 reaction scripts trigger all other reactions including the VIP enable and the VM Server-2 switchover.
- Enable and Disable VIP reaction scripts are not linked to the RM 5091 and 5064 events because they unnecessarily trigger the Enable and Disable VIP reaction script, when the RM fails while running in backup mode.

MLCMD Reaction:

1. The VM SIP Server-2 is switched to primary mode.
2. The 4563 event from the VM SIP Server-2 triggers the enable and disable VIP scripts and switches the VM Server-2 to the primary mode.

Media Control Platform-1 Fails

- The 5091 and 5064 events are generated from the Media Control Platform-1 (MCP-1).
- Alarm reaction scripts are created to react to the 5091 and 5064 events from MCP-1.
- The 5091 and 5064 events from MCP-1 switches the VM SIP Server-2 to primary mode.
- The 4563 event from VM SIP Server-2 triggers the enable and disable VIP scripts and switches the VM Server-2 to primary mode.
- Event from the MCP-1 triggers only the switchover of VM SIP Server-2.
- The VM SIP Server-2 reaction scripts trigger all other reactions including the VIP enable and the switchover of the VM Server-2.
- Enable and Disable VIP reaction scripts are not linked to the MCP 5091 and 5064 events because they unnecessarily trigger the Enable and Disable VIP reaction script, when the MCP fails while running in backup mode.

MLCMD Reaction:

1. The VM SIP Server-2 is switched to primary mode.
2. The 4563 event from the VM SIP Server-2 triggers the enable and disable VIP scripts and switches the the VM Server-2 to primary mode.

Entire Primary Host Fails

- Any one of the following events automatically perform the switchover, since alarm reactions are
-

individually configured for each component:

- 4563 from the VM SIP Server-2
- 4563 from the VM Server-2
- 5091 and 5064 from the RM-1 and MCP-1
- The VM SIPServer-2 and Voicemail Server-2 automatically switch to primary mode.

Note: In this scenario there are chances of getting an IP conflict problem. See [Known Issues and Recommendations](#) for more information. **MLCMD Reaction:**

1. VIP is enabled on the backup VM host.
2. VIP will be disabled on the primary VM host.

Backup Host Failure

If any component on the backup host fails, the primary components keep running, but with no redundancy in the solution. If the failed component is brought back up, HA is restored. If any component on the backup host fails while running in Primary Mode, the events (4563, 5091 & 5064) are generated from the components and alarm reaction scripts are executed in the reverse direction to switch the primary host components to primary mode.

Switchover Cases

During manual switchover scenarios, the 4563 event is generated from the VM Server-2 and the reaction scripts are the same as when the VM Server-1 fails. For example, an administrator switches the VM Server-2 to primary mode using the SCS switchover option. When an administrator switches the VM SIP Server-2 to primary mode using the SCS switchover option, the 4563 event is generated from the VM SIP Server-2 and the reaction scripts are the same as those for the VM SIP Server-1 failure.

Key Notes:

- If any component on the backup host fails, the primary components keep running, but with no redundancy in the solution. If the failed component is brought back up, HA is restored.
- If the Premise SIP server fails, no alarm reaction scripts are triggered.
- MLCMD must use the application DBID. In Management Framework release 8.1.1, MLCMD can use the application name.
- VIP takeover scripts are linked to the event for only the VM SIP Server-1 and VM SIP Server-2. This prevents duplicate execution of the takeover scripts. If any application fails in a host, all applications are switched. Alarm reaction scripts are triggered for the events from different applications for the single switchover scenario.
- VIP takeover scripts are connected to the SIP Server so that the VM SIP Server port is bonded to the VIP immediately when it becomes primary. If the VIP is not enabled immediately after SIP server becomes primary, the SIP server is in an unavailable state.

For example, when VM SIP Server fails, if the VIP scripts are connected to the VM Server, VM SIP server-2 is in an unavailable state until the 4563 event from the VM SIP Server-2 triggers the switchover of VM Server-2 and the 4563 event from VM Server-2 triggers the VIP enable script. VM SIP server-2 remains in an unavailable state until the Virtual IP is enabled in the host that becomes primary.

Note: This issue does not apply to the other applications, because the port and VIP bonding of the other applications do not happen at the switchover, when the applications become primary.

Solution HA Events and Scripts

Third-Party Applications

Eight third-party applications are created for enabling and disabling VIP and switching the components.

1. TP_PRIMARY_VIP_UP: Associated with VIP_UP script on primary host
2. TP_PRIMARY_VIP_DOWN: Associated with VIP_DOWN script on primary host
3. TP_Backup_VIP_UP: Associated with VIP_UP script on backup host
4. TP_Backup_VIP_DOWN: Associated with VIP_DOWN script on backup host
5. TP_VMServer-1_SWITCHOVER: Associated with the script SC_APPLICATION_SWITCHOVER that switches VM Server-1 to primary mode
6. TP_VMServer-2_SWITCHOVER: Associated with the script SC_APPLICATION_SWITCHOVER that switches VM Server-2 to primary mode
7. TP_VM-SIPServer-1_SWITCHOVER: Associated with the script SC_APPLICATION_SWITCHOVER that switches VM SIP Server-1 to primary mode
8. TP_VM-SIPServer-2_SWITCHOVER: Associated with the script SC_APPLICATION_SWITCHOVER that switches VM SIP Server-2 to primary mode

Events and Scripts Association Table

Log Event ID	Component	Script	Third-Party Application
4563	SIP Server Primary	SC_VIP_DOWN on backup host	TP_BACKUP_VIP_DOWN
		SC_VIP_UP on primary host	TP_PRIMARY_VIP_UP
		SC_APPLICATION_SWITCHOVER to switch primary GSVM Server to primary mode	TP_VMSERVER-1_SWITCHOVER
4563	SIP Server Backup	SC_VIP_DOWN on primary server	TP_PRIMARY_VIP_DOWN
		SC_VIP_UP on backup server	TP_BACKUP_VIP_UP
		SC_APPLICATION_SWITCHOVER to switch backup GSVM Server to primary mode	TP_VMSERVER-2_SWITCHOVER
4563	GSVM Server Primary	SC_APPLICATION_SWITCHOVER	TP_VM-

Log Event ID	Component	Script	Third-Party Application
5091/5064	RM Backup	to switch SIP Server Primary to primary mode	SIPSERVER-1_SWITCHOVER
	Media Control Platform Backup		
4563	GSVM Server Backup	SC_APPLICATION_SWITCHOVER to switch VM SIP Server Backup to primary mode	IP_VM-SIPSERVER-2_SWITCHOVER
5091/5064	RM Primary		
	Media Control Platform Primary		

Known Issues and Recommendations

Primary Server Disconnection

When the primary server goes down due to power or network disconnection, the script to disable the Virtual IP fails because the primary box is unavailable. When the primary server is started, there might be an IP conflict since Virtual IP is not disabled in the primary server. Genesys recommends you add the disable Virtual IP script as a startup script in the machine, so the Virtual IP is disabled by default when a system boots up and then according to the event (4563) from Genesys application in our case SIP Server the reaction scripts should be triggered. See **Creating Virtual IP control scripts (Windows, Linux)**. **Note:** There is no resolution for an IP conflict caused by unplugging an IP cable. This is a known limitation.

VIP Not Enabled Issue

The VIP not enabled issue can only occur in Windows. In this scenario, the backup server is down when the primary server starts. If the VIP down service is started after the Primary SIP server is started and the alarm condition from the server has already enabled the VIP in the machine, the VIP is disabled and VM SIP server goes into a service unavailable state.

Preventing The VIP Not Enabled Issue



Purpose: To prevent the VIP from being disabled and the VM SIP server from going into a service unavailable state.

Start

1. Add the following script as a startup script in Windows.

[+] Commands for SC_ENABLE_VIP.BAT

```
@echo off
rem Set Application, Configuration and SCS server details
rem Application Details
set application=<Name of the Application>
rem Configuration server details
set host=<IP address of the config server>
set port=<Port of the config server>
set appname=<App name the config server>
set user=<Config server user name>
set password=<Config server password>
rem Third party application details
set thirdparty=<Third party server name that is linked to the VIP_ON script>
rem for example thirdparty=primary_VIP_up
rem SCS server details
set scshost=<IP address of the SCS server>
set scsport=<Port of the SCS server>
rem set the path in which log files will be stored
```

```

Logpath= <Specify the path to log files>
rem for example Logpath= C:\Solution_HA\Logos
rem set the mlcmd utility path of the local machine in which it is copied
mlcmdpath= <Specify the mlcmd path >
rem for example mlcmdpath = C:\Solution_HA\mlcmd
echo %date% >> %Logpath%\thirdparty_status.log
echo %time% >> %Logpath%\thirdparty_status.log
:start
echo checking the status of thridparty application >> %Logpath%\thirdparty_status.log
"%mlcmdpath%\mlcmd.exe" -getappstatus-runmode %thirdparty% -cshost %host%
-csport %port% -csappname %appname% -csuser %user% -cspassword %password%
-scshost %scshost% -scsport %scsport% 1> %Logpath%\application_mode.txt 2>&1
FINDSTR "APP_STATUS_STOPPED" %Logpath%\application_mode.txt > nul
if errorlevel 0 goto next
if not errorlevel 0 goto start
:next
rem Checking the mode of application
"%mlcmdpath%\mlcmd.exe" -getappstatus-runmode %application% -cshost %host%
-csport %port% -csappname %appname% -csuser %user% -cspassword %password%
-scshost %scshost% -scsport %scsport% 1> %Logpath%\application_mode.txt 2>&1
FINDSTR "BACKUP" %Logpath%\application_mode.txt > nul
if not errorlevel 1 goto backup
rem Checking the mode of application
"%mlcmdpath%\mlcmd.exe" -getappstatus-runmode %application% -cshost %host%
-csport %port% -csappname %appname% -csuser %user% -cspassword %password%
-scshost %scshost% -scsport %scsport% 1> %Logpath%\application_mode.txt 2>&1
FINDSTR "PRIMARY" %Logpath%\application_mode.txt > nul
if errorlevel 1 goto next
timeout 3
if errorlevel 0 goto end
:backup
echo %application% is not in Primary mode, exiting the loop
>> %Logpath%\thirdparty_status.log
exit
:end
rem stopping application
"%mlcmdpath%\mlcmd.exe" -stopapp %application% -cshost %host% -csport %port%
-csappname %appname% -csuser %user% -cspassword %password% -scshost %scshost%
-scsport %scsport%
timeout 3
rem starting application
"%mlcmdpath%\mlcmd.exe" -startapp %application% -cshost %host% -csport %port%
-csappname %appname% -csuser %user% -cspassword %password% -scshost %scshost%
-scsport %scsport%
timeout 3
echo %application% is restarted since the application %thirdparty% was in unknown
state >> %Logpath%\thirdparty_status.log
echo %date% >> %Logpath%\thirdparty_status.log
echo %time% >> %Logpath%\thirdparty_status.log
echo. >> %Logpath%\thirdparty_status.log

```

2. Copy the MLCMD utility files (mlcmd.exe and mlcmd.exe.manifest files) from the SCS host onto the host running the VM components.

End

Subnet for GSVM Servers

You must connect both the primary and backup GSVM Servers within the same subnet to prevent problems refreshing the ARP cache of the gateway after the switchover of the Virtual IP. You can use the ARPING command in the Virtual IP takeover scripts to resolve this issue. ARPING is a tool included in Linux. A third-party version of ARPING can also be downloaded and installed in Windows.

Traditional HA Task Summary

This page includes the following sections:

 [Deployment Overview](#)


 [Deployment Task Summary](#)


Deployment Overview

Traditional HA is typically used when deploying Genesys SIP Voicemail into an existing SIP Server/GVP HA configuration. In this case, you use the Manual method to deploy the GSVM Server. SIP Server and GVP should already be configured for HA mode according to the procedures found in their respective Deployment Guides. To see a detailed architecture for this sample deployment layout, see the [Traditional HA Architecture](#) diagram.

Deployment Task Summary

The following table lists the tasks that are required to configure an Genesys SIP Voicemail in a Traditional HA deployment.

Objective	Related procedures and actions
<ol style="list-style-type: none"> Ensure that your system meets the deployment prerequisites. 	<ol style="list-style-type: none"> Verify the GSVM Server Prerequisites. For Traditional HA, the following Genesys components must already be installed: <ul style="list-style-type: none"> SIP Server in a primary/backup HA configuration - See the <i>Framework 8.1 SIP Server High-Availability Deployment Guide</i>. GVP Resource Manager in an active/standby HA configuration - See the "Resource Manager High Availability" chapter of the <i>Genesys Voice Platform 8.1 Deployment Guide</i>.
<ol style="list-style-type: none"> Install the components. 	<p>For Traditional HA, use the Manual Install method.</p> <ul style="list-style-type: none"> On both the primary and backup servers, complete the manual steps described in: SIP Voicemail Manual Deployment <p> Key Notes</p>

Objective	Related procedures and actions
	<ul style="list-style-type: none"> • Specify the HA role for each server: one Primary, one backup. • Enter the Virtual IP address for HTTP communication with the HA pair. • Each machine should include a Network Interface Card (NIC).
<p>3. Configure GSVM Server HA.</p>	<ol style="list-style-type: none"> 1. Configure the Virtual IP Takeover and Application Switchover scripts (Windows, Linux) 2. In both primary and backup GSVM Server Application objects, go to Start Info > Command Line Arguments, and edit the following parameter: -vms_host 0.0.0.0 . 3. Configure the MCP shared voicemail storage <ul style="list-style-type: none"> • Windows — MCP Shared Voicemail Storage (Windows) • Linux — MCP Shared Voicemail Storage (Linux) <p> Key Notes</p> <ul style="list-style-type: none"> • After installation, you must start the master GSVM Server first, before starting the backup server.

Configuring MCP Shared Voicemail Storage (Windows)

Start

1. Map a shared network folder as a local drive on the MCP hosts.
 - Share on Host1: voicemail-record
 - On MCP1: Map D: to \\Host1\voicemail-record and specify D: in the recording basepath parameter.
 - On MCP2: Map D: to \\Host1\voicemail-record and specify D: in the recording basepath parameter.
 - On the voicemail server application set voice-record-path=\\Host1\voicemail-record
2. Mount the network drive for the local system account.
 1. Download SysinternalsSuite by Mark Russianovich available at <http://technet.microsoft.com/en-us/sysinternals/bb842062.aspx>.
 2. As administrator, open an elevated cmd.exe prompt.
 3. Elevate again to root using PSEXEC.exe.
 4. Navigate to the folder containing SysinternalsSuite and execute the following command:
psexec -i -s cmd.exe
 5. Confirm you are inside of the prompt nt authority\system by typing whoami
 6. Create the persistent mapped drive as the SYSTEM account with the following command:
net use z:\servername\sharedfolder /persistent:yes.

End

Next Steps  [Back to Task Table](#)

Configuring MCP Shared Voicemail Storage (Linux)

Enable the transfer of HA voicemail server contacts from an MCP that resides in a different host if the shared record folder is available in Windows or Linux.

Depending on location of the shared record folder, complete one of these procedures:

 [Shared Record Folder in Windows](#)

 [Shared Record Folder in Linux](#)

Configuring MCP Shared Voicemail Storage (Shared Record Folder in Windows)

Start

1. Share the Windows folder with write access to users.
2. Mount the folder on the Linux machines having MCP by using the following command:

```
mount -t cifs -o username=<username>,password=<password>  
//<windows_server_ip>/<shared_folder> <local_mount_path>
```
3. Set the parameters Recording.basepath in MCP and voice-record-path in the voicemail server to the local path.

End

Next Steps

 [Back to Task Table](#)

Configuring MCP Shared Voicemail Storage (Shared Record Folder in Linux)

Start

1. Share the Linux folder with write access to users by using the following command:

```
chmod -R 777 <folderpath>
```
2. Set access control for the shared folder.
Note: The file /etc/exports serves as the access control list for file systems which may be exported to NFS clients.

- a. Add the following line to the file `/etc/exports`:
`<sharedfolderpath> *(rw, sync)`
- b. Restart the `nfs` service by using the following command:
`/etc/init.d/nfs restart`
3. Mount the folder on the MCP host using the following command:
`mount -t nfs <linux_ip>:<shared_folder_path> <local_mount_path>`

End

Next Steps

 [Back to Task Table](#)

Solution HA Task Summary

This page includes the following sections:

 [Deployment Overview](#)

 [Deployment Task Summary](#)

Deployment Overview


A typical small-scale HA deployment places all SIP Voicemail components on a single server, with backup components mirrored on a second backup server. Components should be deployed as follows:


Primary Server	Backup Server
<ul style="list-style-type: none"> • Primary Voicemail SIP Server • Primary VM Server • Resource Manager 1 (active) • MCP 1 	<ul style="list-style-type: none"> • Backup Voicemail SIP Server • Backup VM Server • Resource Manager 2 (active) • MCP 2

Note: The premise (agent) SIP Server is outside the scope of the SIP Voicemail HA configuration. If deployed in HA mode, either on the VM servers or on a separate server, a switchover from primary to backup Agent SIP Server does not impact the Genesys Voicemail components (Voicemail SIP Server, GVP Resource Manager, VM Server). To see a detailed architecture for this sample deployment layout, see the [Basic Architecture](#) diagram.

Deployment Task Summary

The following table lists the tasks that are required to configure a Genesys SIP Voicemail Server in an HA deployment.

Objective	Related procedures and actions
1. Ensure that your system meets the deployment prerequisites.	See SIP Voicemail Deployment Prerequisites .
2. Install the components.	<p>Use Complete Install Tool On both primary and backup servers, complete the procedures in the following task summary: SIP Voicemail Complete Install  Key Notes</p>

Objective	Related procedures and actions
	<ul style="list-style-type: none"> Specify the HA role for each server: one Primary, one backup. Enter the Virtual IP address for HTTP communication with the HA pair.
<p>3. Configure GSVM Server HA.</p>	<ol style="list-style-type: none"> Configure the Virtual IP Takeover scripts <ul style="list-style-type: none"> Windows — Virtual IP Address Takeover (Windows) Linux — Virtual IP Address Takeover (Linux) In both primary and backup GSVM Server Application objects, go to Start Info > Command Line Arguments, and edit the following parameter: <code>-vms_host 0.0.0.0</code>. <p> Key Notes</p> <ul style="list-style-type: none"> After installation, you must start the primary GSVM Server first, before starting the backup server.

IP Address Takeover (Windows)

Complete the following procedures to configure the Virtual IP Address Takeover scripts for Genesys SIP Voicemail Server High-Availability (HA) on the Windows 2008 64-bit operating system.

IP Address Takeover (Windows) Task Summary

Objective	Related procedures and actions
1. Create the Virtual IP address control scripts.	Creating control scripts Windows
2. Test the control scripts (Windows)	Testing control scripts
3. Create Application control scripts (Windows)	Application control scripts Windows
4. Create Genesys Applications for the control scripts.	Creating Genesys Applications for the control scripts
5. Create the Alarm Reaction scripts.	Creating Alarm Reaction scripts
6. Create the SCS Alarm Conditions.	SCS Alarm Conditions
7. Test the SCS Alarm Conditions.	Testing Alarm Conditions

Creating Virtual IP control scripts (Windows)

Start

1. On the primary VM Server, create a batch file that is named SC_VIP_UP.BAT, and enter the following commands into the file:

[+] Commands for SC_VIP_UP.BAT

```
@echo off
rem set the following parameter corresponding to your machine configuration
set VirtualIP=<Virtual IP of the Machine>
rem for example VirtualIP=172.24.133.254
set vipMask=<Subnet mask of the Machine>
rem for example vipMask=255.255.255.0
set VirtualInterface=<Name of existing network Interface in which VIP will be
created>
rem for example VirtualInterface="Local Area Connection"
set startCount= 0
set EndCount= 20
rem set the path in which log files will be stored
set Logpath= <Specify the path to log files>
rem for example Logpath= C:\Solution_HA\Logs

echo ***** SC_VIP_UP_start *****
>> %Logpath%\Takeover_On.log
echo %date% >> %Logpath%\Takeover_On.log
echo %time% >> %Logpath%\Takeover_On.log
echo. >> %Logpath%\Takeover_On.log
rem check if Virtual IP released on paired host and below loop will check up to 20
counts

:start
if %startCount% GTR %EndCount% GOTO end
    echo %startCount%
    cscript.exe ping.vbs %VirtualIP% //nologo >> %Logpath%\Takeover_On.log
    if not errorlevel 1 goto ready
    echo Looping till VIP disable in backup host >> %Logpath%\Takeover_On.log
    set /a startCount+=1
    timeout 1
    goto start

:end
echo. >> %Logpath%\Takeover_On.log
echo VIP is already enabled either in the same host or the paired host
>> %Logpath%\Takeover_On.log
goto done
exit
:ready
rem Add VirtualIP
netsh interface ip delete arpcache
netsh interface ip add address name=%VirtualInterface% addr=%VirtualIP%
mask=%vipMask% >> %Logpath%\Takeover_On.log
rem check if VirtualIP added sucesefully if not do it again
```

```

cscript.exe check_ip.vbs localhost %VirtualIP% //Nologo >> %Logpath%\Takeover_On.log
if errorlevel 1 goto done

netsh interface ip delete address name=%VirtualInterface% addr=%VirtualIP%
>> %Logpath%\Takeover_On.log
netsh interface ip add address name=%VirtualInterface% addr=%VirtualIP%
mask=%vipMask% >> %Logpath%\Takeover_On.log
if errorlevel 1 echo %VirtualIP% not added to %VirtualInterface%
>> %Logpath%\Takeover_On.log

:done
echo. >> %Logpath%\Takeover_On.log
echo ***** SC_VIP_UP_End *****
>> %Logpath%\Takeover_On.log

```

2. In the batch file, replace the variables indicated by angle brackets with appropriate values.
3. On the primary VM Server, create a batch file that is named SC_VIP_DOWN.BAT, and enter the following commands into the file:

[+] Commands for SC_VIP_DOWN.BAT

```

@echo off
rem set the following parameter corresponding to your machine configuration
set VirtualIP=<Virtual IP of the Machine>
rem for example VirtualIP=172.24.133.254
set VirtualInterface=<Name of existing network Interface in which VIP will be
created>
rem for example VirtualInterface="Local Area Connection"
rem set the path in which log files will be stored
set Logpath= <Specify the path to log files>
rem for example Logpath= C:\Solution_HA\Logs

echo ***** SC_VIP_DOWN_Start *****
>> %Logpath%\Takeover_Off.log
echo %date% >> %Logpath%\Takeover_Off.log
echo %time% >> %Logpath%\Takeover_Off.log
netsh interface ip delete address name=%VirtualInterface% addr=%VirtualIP%
>> %Logpath%\Takeover_Off.log
netsh interface ip delete arpcache
Timeout 10
cscript.exe ping.vbs %VirtualIP% //Nologo >> %Logpath%\Takeover_Off.log
echo %date% >> %Logpath%\Takeover_Off.log
echo %time% >> %Logpath%\Takeover_Off.log
echo ***** SC_VIP_DOWN_End *****
>> %Logpath%\Takeover_Off.log

```

4. In the batch file, replace the variables indicated by angle brackets with appropriate values.
5. Add the SC_VIP_DOWN script as a task in Task Scheduler and schedule it to execute when the system starts. This disables the VIP by default when the system starts and according to the mode (Primary/ Backup) of the application, the VIP will be enabled or disabled by the alarm reaction scripts.
6. Copy the MLCMD utility files (mlcmd.exe and mlcmd.exe.manifest files) that are installed with SCS onto the VM Server.
7. On the primary VM Server, create an accessory script that is named Ping.vbs, and enter the following

commands into the script:

[+] Commands for Ping.vbs

```
rem ping host and return 1 if ping successful 0 if not
On Error Resume Next

if WScript.Arguments.Count > 0 then
    strTarget = WScript.Arguments(0)
    Set objShell = CreateObject("WScript.Shell")
    Set objExec = objShell.Exec("ping -n 2 -w 1000 " & strTarget)
    strPingResults = LCase(objExec.StdOut.ReadAll)
    If InStr(strPingResults, "reply from") Then
        WScript.Echo strTarget & "responded to ping."
        wscript.Quit 1
    Else
        WScript.Echo strTarget & "did not respond to ping."
        wscript.Quit 0
    End If
Else
    WScript.Echo "target is not specified."
    wscript.Quit -1
End If
```

8. On the primary VM Server, create an accessory script that is named Check_ip.vbs, and enter the following commands into the script:

[+] Commands for Check_ip.vbs

```
rem check if IP address (arg0 ) can be found on host (arg1 )
On Error Resume Next

if WScript.Arguments.Count > 0 then
    strComputer = WScript.Arguments(0)
    targetIPAddress = WScript.Arguments(1)
    Set objWMIService = GetObject("winmgmts:" &
"{impersonationLevel=impersonate}!\\" & strComputer & "\root\cimv2")
    Set colNicConfigs = objWMIService.ExecQuery ("SELECT * FROM
Win32_NetworkAdapterConfiguration WHERE IPEnabled = True")

    WScript.Echo "Computer Name: " & strComputer & " ip " & targetIPAddress

    For Each objNicConfig In colNicConfigs

        For Each strIPAddress In objNicConfig.IPAddress
            If InStr(strIPAddress, targetIPAddress) Then
                WScript.Echo targetIPAddress & " is found on " & objNicConfig.Description
                wscript.Quit 1
            End If
        Next
    Next
    WScript.Echo targetIPAddress & "not found."
    wscript.Quit 0
Else
    WScript.Echo "target not specified."
    wscript.Quit -1
End If
```

-
9. Place the accessory scripts Ping.vbs and Check_ip.vbs in the same directory as the SC_VIP_UP.BAT and SC_VIP_DOWN.BAT files.
 10. Repeat the steps in this procedure to create the scripts on the backup VM Server.

End

Next Steps

[Testing control scripts](#) OR [Back to Task Table](#)

Testing the Virtual IP control scripts (Windows)

Start

1. Run the SC_VIP_DOWN.BAT script on the backup VM Server.
2. Run the SC_VIP_UP.BAT script on the primary VM Server.
3. Verify that the Virtual IP interface is running on the primary host by using the ipconfig command. For example:

```
@C:\GCTI\SWITCHOVER\1NIC>ipconfig
@Windows IP Configuration
@Ethernet adapter Local Area Connection:
@Connection-specific DNS Suffix . . :
@IP Address. . . . . : 10.10.11.103
@Subnet Mask . . . . . : 255.255.255.0
@IP Address. . . . . : 10.10.11.101
@Subnet Mask . . . . . : 255.255.255.0
@Default Gateway . . . . . : 10.10.11.104
```

2. Verify that the Virtual IP interface is not running on the backup VM Server. For example:

```
@C:\GCTI\SWITCHOVER\1NIC>ipconfig
@Windows IP Configuration
@Ethernet adapter Local Area Connection:
@ Connection-specific DNS Suffix . . :
@ IP Address. . . . . : 10.10.11.102
@ Subnet Mask . . . . . : 255.255.255.0
@ Default Gateway . . . . . : 10.10.11.104
```

3. Run the SC_VIP_DOWN.BAT script on the primary VM Server.
4. Run the SC_VIP_UP.BAT script on the backup VM Server.
5. Verify that the Virtual IP interface is running on the backup VM Server by using the ipconfig command. Output should appear similar to the following:

```
@Ethernet adapter Local Area Connection:
@Connection-specific DNS Suffix . . :
@IP Address. . . . . : 10.10.11.103
@Subnet Mask . . . . . : 255.255.255.0
@IP Address. . . . . : 10.10.11.102
@Subnet Mask . . . . . : 255.255.255.0
@Default Gateway . . . . . : 10.10.11.104
```

End

Next Steps

[Application control scripts Windows](#)

OR

[Back To Task Table](#)

Creating Application control scripts (Windows)

This script provides an example of switching a VM server from backup to primary mode. The script should be created for the server running SCS. The MLCMD utility that is installed with SCS can be used to switch any application. The following arguments are mandatory for the MLCMD command:

- For SCS release 8.1.0 and lower
 - DBID of the application
 - SCS host IP and SCS T-Lib port
- For SCS release 8.1.1 and higher
 - Application Name
 - Configuration Server IP, Port, Application Name, Username & Password
 - SCS host IP and SCS T-Lib port (Optional)

Creating Application control scripts (Windows)

Start

1. On the server running SCS, create a batch file that is named SC_APPLICATION_SWITCHOVER.BAT, and enter the following commands into the file, depending the SCS release version you are using:

- For SCS release 8.1.0 and lower:

[+] Commands for SC_APPLICATION_SWITCHOVER.BAT

```
@echo off
rem set Application and SCS server details
rem Application Details
set dbid=<DBID of the application>
set application=<Name of the Application>
rem SCS server details
set scshost=<IP address of the SCS server>
set scsport=<Port of the SCS server>
rem set the path in which log files will be stored
set Logpath= <Specify the path to log files>
rem for example Logpath= C:\Solution_HA\Logs
rem set the mlcmd utility path
set mlcmdpath= <Specify the mlcmd path >
rem for example mlcmdpath = C:\Program Files (x86)\GCTI\SCServer\
Solution_Control_Server
echo ***** %application% Switchover Start *****
>> %Logpath%\Switchover.log
echo %date% >> %Logpath%\Switchover.log
echo %time% >> %Logpath%\Switchover.log
rem checking the status of application
```

```

"%mlcmdpath%\mlcmd.exe" -getappstatus-runmode %dbid% -scshost %scshost%
-scsport %scsport% 1> %Logpath%\Before_Switch.txt 2>&1
FINDSTR "BACKUP" %Logpath%\Before_Switch.txt > nul
if errorlevel 1 goto end
rem switching application to primary mode
"%mlcmdpath%\mlcmd.exe" -switchapp %dbid% -scshost %scshost% -scsport %scsport%
timeout 2
rem checking the status of application after switchover
"%mlcmdpath%\mlcmd.exe" -getappstatus-runmode %dbid% -scshost %scshost%
-scsport %scsport% 1> %Logpath%\After_Switch.txt 2>&1
FINDSTR "PRIMARY" %Logpath%\After_Switch.txt > nul
if not errorlevel 1 goto done
if errorlevel 1 echo %application% switchover failed and switchover ends
>> %Logpath%\Switchover.log
exit
:end
echo %application% is not in Backup mode and switchover ends
>> %Logpath%\Switchover.log
exit
:done
echo %application% is switched to PRIMARY mode successfully
>> %Logpath%\Switchover.log
echo ***** %application% Switchover End *****
>> %Logpath%\Switchover.log
echo. >> %Logpath%\Switchover.log

```

- For SCS release 8.1.1 and higher:

[+] Commands for SC_APPLICATION_SWITCHOVER.BAT

```

@echo off
rem set application SCS and Configuration server details
rem Application Details
set application=<Name of the Application>
rem Configuration server details
set host=<IP address of the config server>
set port=<Port of the config server>
set appname=<App name the config server>
set user=<Config server user name>
set password=<Config server password>
rem SCS server details
set scshost=<IP address of the SCS server>
set scsport=<Port of the SCS server>
rem set the path in which log files will be stored
set Logpath= <Specify the path to log files>
rem for example Logpath=C:\Solution_HA\Logs
rem set the mlcmd utility path
set mlcmdpath= <Specify the mlcmd path>
rem for example mlcmdpath = C:\Program Files (x86)\GCTI\SCServer\
Solution_Control_Server
echo ***** %application% Switchover Start *****
>> %Logpath%\Switchover.log
echo %date% >> %Logpath%\Switchover.log
echo %time% >> %Logpath%\Switchover.log
rem checking the status of application
"%mlcmdpath%\mlcmd.exe" -getappstatus-runmode %application% -cshost %host%
-csport %port% -csappname %appname% -csuser %user% -cspassword %password%
-scshost %scshost% -scsport %scsport% 1> %Logpath%\Before_Switch.txt 2>&1
FINDSTR "BACKUP" %Logpath%\Before_Switch.txt > nul
if errorlevel 1 goto end
rem switching application to primary mode
"%mlcmdpath%\mlcmd.exe" -switchapp %application% -cshost %host% -csport %port%

```

```

-csapname %appname% -csuser %user% -cspassword %password% -scshost %scshost%
-scsport %scsport%
timeout 2
rem checking the status of application after switchover
"%mlcmdpath%\mlcmd.exe" -getappstatus-runmode %application% -cshost %host%
-csport %port% -csappname %appname% -csuser %user% -cspassword %password%
-scshost %scshost% -scsport %scsport% 1> %Logpath%\After_Switch.txt 2>&1
FINDSTR "PRIMARY" %Logpath%\After_Switch.txt > nul
if not errorlevel 1 goto done
if errorlevel 1 echo %application% switchover failed and switchover
ends>> %Logpath%\Switchover.log
exit
:end
echo %application% is not in Backup mode and switchover ends
>> %Logpath%\Switchover.log
exit
:done
echo %application% is switched to PRIMARY mode successfully
>> %Logpath%\Switchover.log
echo ***** %application% Switchover End *****
>> %Logpath%\Switchover.log
echo. >> %Logpath%\Switchover.log

```

2. In the batch file, replace the variables indicated by angle brackets with appropriate values.
3. Repeat these steps to create control scripts for each VM SIP Server and VM Server.

End

Next Steps

[Genesys Applications](#)

OR

[Back to Task Table](#)

Creating Genesys Applications for the control scripts

Application Object	Corresponding Script	Description
TP_PRIMARY_VIP_UP	SC_VIP_ON.BAT	Enables the Virtual IP address on the primary server.
TP_PRIMARY_VIP_DOWN	SC_VIP_OFF.BAT	Disables the Virtual IP address on the primary server.
TP_BACKUP_VIP_UP	SC_VIP_ON.BAT	Enables the Virtual IP address on the backup server.
TP_BACKUP_VIP_DOWN	SC_VIP_OFF.BAT	Disables the Virtual IP address on the backup server.
TP_VMSERVER-1_SWITCHOVER	SC_APPLICATION_SWITCHOVER.BAT	Switches the primary VM Server to primary mode.
TP_VMSERVER-2_SWITCHOVER	SC_APPLICATION_SWITCHOVER.BAT	Switches the backup VM Server to primary mode.
TP_VM-SIPSERVER-1_SWITCHOVER	SC_APPLICATION_SWITCHOVER.BAT	Switches the primary VM SIP Server to primary mode.
TP_VM-SIPSERVER-2_SWITCHOVER	SC_APPLICATION_SWITCHOVER.BAT	Switches the backup VM SIP Server to primary mode.

Creating Application objects for the shell files allows the shell files to be run as applications within the Genesys Framework.

Creating Genesys Applications for the control scripts



Start

1. In the Configuration Manager, select Environment > Applications.
2. Right-click and select New > Application.
3. Select the Third Party Server template from the Application Templates folder, and click OK.
4. On the General tab, enter a name for the Application object. For example, TP_PRIMARY_VIP_UP.
Note: You can use the suggested Application object names, or you can specify your own.
5. Select the Server Info tab.
 - a. For VIP control scripts, select the host name of the VM Server on which the corresponding control/shell script is located.
 - b. For application control scripts, select the host name of the server running SCS.
 - c. If necessary, specify a valid communication-port number by using the Edit Port option.

6. Select the Start Info tab.
7. Set the Working Directory to the location of the control/shell script, and enter the name of the script in the Command Line field.
For example, for the TP_PRIMARY_VIP_UP Application object, enter the script name that enables the Virtual IP address:
Windows: SC_VIP_UP.BAT
Linux: SC_VIP_UP.SH
For the PRIMARY_VIP_DOWN Application object, enter the script name that disables the Virtual IP address:
Windows: SC_VIP_DOWN.BAT
Linux: SC_VIP_DOWN.SH
8. Repeat the steps in this procedure to create an Application object for each of the scripts.

End

Next Steps

 [Creating Alarm Reaction scripts](#) OR  [Back to Task Table: Windows IP Address Takeover](#) OR [Linux IP Address Takeover](#)

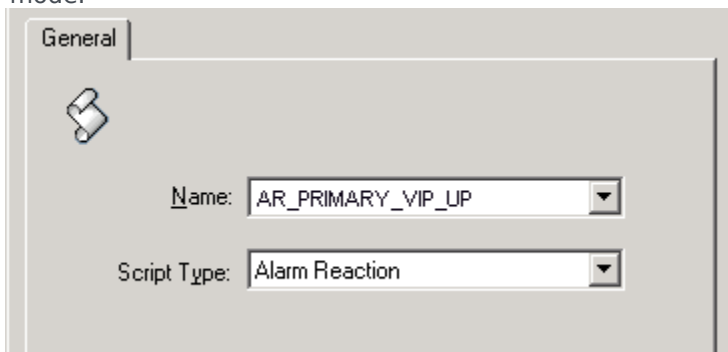
Creating the Alarm Reaction scripts

When an HA-related Alarm Condition occurs, the associated Alarm Reaction script is run. These scripts are configured to call the Application objects that you created in [Genesys Applications](#).

Creating the Alarm Reaction scripts

Start

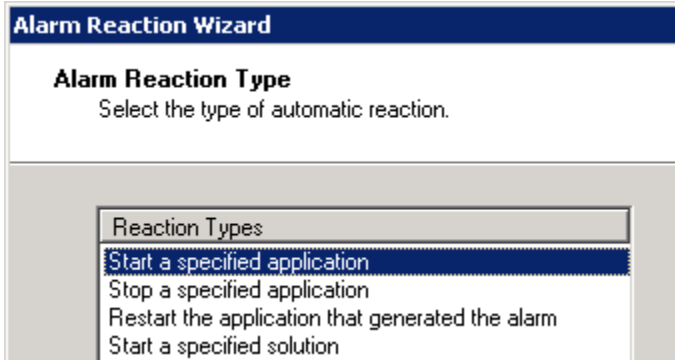
1. Open the Configuration Manager.
2. Select Resources > Scripts.
3. Right-click and select New > Script.
4. Create alarm reaction scripts: one for each of the third-party Application objects that you previously created. For example:
 - AR_PRIMARY_VIP_UP — Triggers a script that enables the Virtual IP address (to be run on the primary VM Server).
 - AR_PRIMARY_VIP_DOWN — Triggers a script that disables the Virtual IP address (to be run on the primary VM Server).
 - AR_BACKUP_VIP_UP — Triggers a script that enables the Virtual IP address (to be run on the backup VM Server host).
 - AR_BACKUP_VIP_DOWN — Triggers a script that disables the Virtual IP address (to be run on the backup VM Server).
 - AR_VMSERVER-1_SWITCHOVER — Triggers a script that switches VM Server-1 to primary mode.
 - AR_VMSERVER-2_SWITCHOVER — Triggers a script that switches VM Server-2 to primary mode.
 - AR_VM-SIPSERVER-1_SWITCHOVER — Triggers a script that switches VM SIP Server-1 to primary mode.
 - AR_VM-SIPSERVER-2_SWITCHOVER — Triggers a script that switches VM SIP Server-2 to primary mode.



5. For each of the Alarm Reaction scripts, select Alarm Reaction as the Script Type.

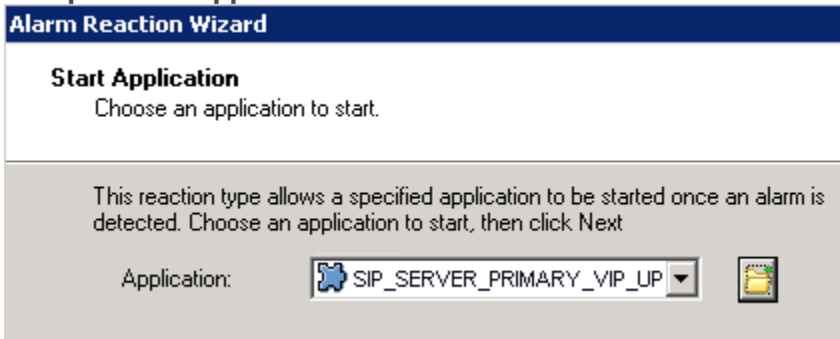
6. For each of the Alarm Reaction scripts, use the Alarm Reaction Wizard to configure the Alarm Reaction Type.
 - a. Select an Alarm Reaction script, and right-click to open the Alarm Reaction Wizard (select Wizard > Configure).
 - b. In the Alarm Reaction Wizard, click Next.

Select Start Application (used for VIP takeover scripts)



- c. In the Alarm Reaction Type dialog box, select Start a specified application or Stop a specified application, and click Next.
- d. Browse to select the corresponding Application object. For example, for the AR_PRIMARY_VIP_UP Alarm Reaction script, select the TP_PRIMARY_VIP_UP Application object of type Third Party Server.

Sample Start Application



- e. Repeat the previous steps to configure each of the Alarm Reaction scripts that you created in Step 4.

Next Steps

➡ [SCS Alarm Conditions](#) OR ➡ Back to Task Table: [Windows IP Takeover](#) OR [Linux IP Takeover](#)

Creating the SCS Alarm Conditions

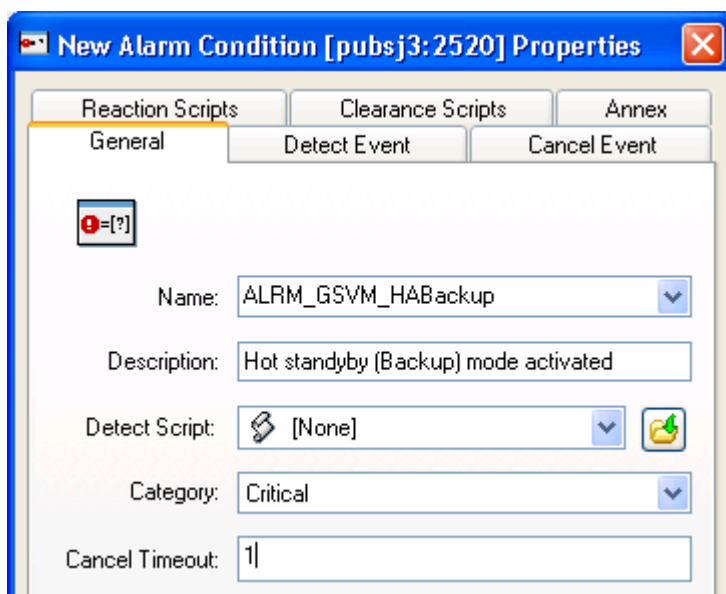
This page provides the procedure for creating SCS alarm conditions:

- [Creating the SCS Alarm Conditions](#)

Creating the SCS Alarm Conditions

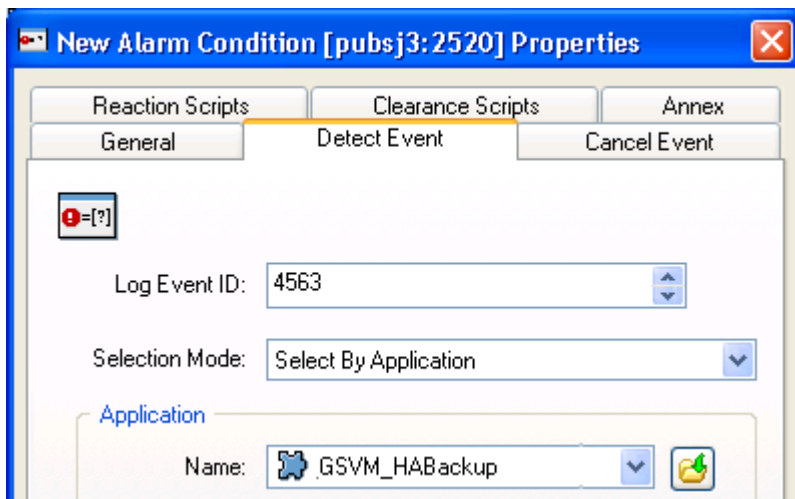
Start

1. Open the Configuration Manager.
2. Navigate to the Environment > Alarm Conditions folder.
3. Right-click and select New > Alarm Condition to open the New Alarm Condition Properties dialog box.
4. On the General tab:
 - Enter a Name for the Alarm Condition—for example, ALRM_GSVM_HABackup.
 - Optionally, enter a description.
 - For the Category value, select Critical.
 - Set Cancel Timeout to 1.



5. On the Detect Event tab:
 - Set the Log Event ID as defined in HA Events and Scripts ([Traditional](#), [Solution](#)).
 - Set the Selection Mode to Select By Application.

- For the Application Name field, click the folder icon to browse for the Application object. For example, if you are creating an Alarm Condition for the primary GSVM Server, select the primary GSVM Server Application object.



6. Click OK.
7. On the Reaction Scripts tab, add the Alarm Reaction script as defined in HA Events and Scripts ([Traditional](#), [Solution](#)).
8. Repeat the steps in this procedure to create each of the Alarm Conditions for the hot standby-related log events for VM SIP Server and VM Server, and the application failure-related log events for Resource Manager.

End

Next Steps ➡ [Testing Alarm Conditions](#) OR ➡ Back to Task Table: [Windows IP Takeover](#) OR [Linux IP Takeover](#)

Testing Alarm Conditions

Start

1. Use Telnet to access the VM Server Virtual IP interface.
2. Open the Solution Control Interface (SCI).
3. Under Alarm Conditions, select the Alarm Condition that you created in [SCS Alarm Conditions](#).
For example:
ALRM_VM_HABackup — right-click it, and then click Test.
The ALRM_VM_HABackup Alarm Condition indicates that the primary VM Server is in backup mode, which triggers the Alarm Reaction scripts that disable the Virtual IP interface at the primary VM Server and enable the Virtual IP interface at the backup VM Server.
4. Use an `ipconfig` command to verify that the Virtual IP interface is active on the backup VM Server and that the Virtual IP interface is inactive on the primary VM Server.





End



Next Steps  Back to Task Table: IP Takeover ([Windows](#), [Linux](#))

IP Address Takeover (Linux)

Complete the following procedures to configure the Virtual IP Address Takeover scripts for Genesys SIP Voicemail Server High-Availability (HA) on Linux operating systems.

IP Address Takeover (Linux) Task Summary

Objective	Related procedures and actions
1. Create a configuration file for the Virtual IP interface.	1. On each of the SIP Server host computers, locate the following file: <code>/etc/sysconfig/network-scripts/ifcfg-eth0</code> 2. Create a copy of this file, named: <code>/etc/sysconfig/network-scripts/ifcfg-eth0:1</code> 3. Define IPADDR, NETMASK, and NETWORK parameter values for the Virtual IP interface. When you are finished, the content of the file should appear similar to the following example: <pre> DEVICE=eth0:1 BOOTPROTO=static USERCTL=yes TYPE=Ethernet IPADDR=192.51.14.208 NETMASK=255.255.255.0 NETWORK=192.51.14.0 BROADCAST=192.51.14.255 ONPARENT=no </pre>
2. Create the Virtual IP address control scripts.	Complete the following procedure:  Creating Virtual IP control scripts (Linux)
3. Create the Application control scripts.	Complete the following procedure:  Creating Application control scripts (Linux)
4. Create Genesys Applications for the control scripts.	Complete the following procedure:  Creating Genesys Applications for the control scripts
5. Create the Alarm Reaction scripts.	Complete the following procedure:  Creating Alarm Reaction scripts

Objective	Related procedures and actions
6. Create the SCS Alarm Conditions.	Complete the following procedure:  SCS Alarm Conditions
7. Testing the SCS Alarm Conditions.	Complete the following procedure:  Testing Alarm Conditions

Creating Virtual IP control scripts (Linux)

Start

1. On the primary VM Server, create a shell file to enable the Virtual IP interface that is named SC_VIP_UP.SH, and enter the following commands into the file:

[+] Commands for SC_VIP_UP.SH

```
#!/bin/sh
# Set the path in which log files will be stored
Logpath=<Specify the path to log files>
# For example Logpath=/home/Logs
echo -e "\n-----Bringing VIP Up-----" >> $Logpath/Takeover_0n.log
date >> $Logpath/Takeover_0n.log
# Set the following parameter corresponding to your machine configuration
Interface=<Interface in which VIP should be added>
# For example Interface=eth0:1
Virtual_IP=<Virtual IP of the Machine>
# For example Virtual_IP=172.24.133.254
Gateway=<Gateway of the Machine>
# For example Gateway=10.1.1.1
ping_count=1
loop_count=20
# Looping up to 20 count or till VIP disabled in the HA pair
for (( i=1; i<=$loop_count;i++ ))
do
result=$(ping -c $ping_count $Virtual_IP | grep "bytes from")
if [ "$result" ]
then
echo "$i ping reply got from the HA pair" >> $Logpath/Takeover_0n.log
sleep 1
else
echo "VIP disabled in the HA pair" >> $Logpath/Takeover_0n.log
# Enabling the VIP once it is disabled in paired host
break
fi
done

/sbin/ifconfig $Interface $Virtual_IP up
# Update the ARP cache of the gateway
arping -s $Virtual_IP $Gateway -f
echo -e "\n-----VIP enabled in this machine-----" >> $Logpath/Takeover_0n.log
date >> $Logpath/Takeover_0n.log
exit
```

2. In the shell file, replace the variables indicated by angle brackets with appropriate values.
3. On both GSVM server host computers, create a shell file to disable the Virtual IP interface that is named SC_VIP_DOWN.SH, and enter the following commands into the file:

[+] Commands for SC_VIP_DOWN.SH

```
# Set the path in which log files will be stored
Logpath= <Specify the path to log files>
```

```
# For example Logpath=/home/Logs
echo -e "\n-----Bringing VIP down-----" >> $Logpath/Takeover_Off.log
date >> $Logpath/Takeover_Off.log
# Set the following parameter corresponding to your machine configuration
Interface=<Interface in which VIP should be added>
Virtual_IP=<Virtual IP of the Machine>
/sbin/ifconfig $Interface down
sleep 10
# Update arpcache of the local machine
ping -c 2 $Virtual_IP
echo -e "-----VIP disabled-----" >> $Logpath/Takeover_Off.log
```

4. In the shell file, replace the variables indicated by angle brackets with appropriate values.
5. Add the SC_VIP_DOWN script as a startup script by adding the script to the current run-level of the machine.
 - a. Copy SC_VIP_DOWN.SH to the /etc/init.d folder.
 - b. Create a symlink from your run-level directory to execute when a system starts to disable the Virtual IP. For example, if the run-level is 5, the creation of symlink is as follows:
"ln -s /etc/init.d/<scriptfile> /etc/rc.d/rc5.d/S50<scriptfile>"
The S50 tells the system to start the script when it starts.

End

Next Steps

 [Genesys Applications](#) OR  [Back to Task Table](#)

Creating Application control scripts (Linux)

This script provides an example of switching a VM server from backup to primary mode. A similar script should be created for all servers running SCS. The MLCMD utility that is installed with SCS can be used to switch any application. The following arguments are mandatory for the MLCMD command:

- For SCS release 8.1.0 and lower
 - DBID of the application
 - SCS host IP and SCS T-Lib port
- For SCS release 8.1.1 and higher
 - Application name
 - Configuration Server IP, Port, Application Name, Username & Password
 - SCS host IP and SCS T-Lib port (Optional)

Creating Application control scripts (Linux)

Start

1. For SCS release 8.1.0 and lower, on all servers running SCS, create a shell file that is named SC_APPLICATION_SWITCHOVER.SH, and enter the following commands into the file:

[+] Commands for SC_APPLICATION_SWITCHOVER.SH

```
#!/bin/sh
# Set application and SCS server details
# Application Details
dbid=<dbid of the Application>
application=<Name of the Application>
# Solution Control Server details
host=<IP address of the SCS server>
port=<Port of the SCS server>
# Set the path in which log files will be stored
Logpath= <Specify the path to log files>
# for example Logpath=/home/Logs
# Set the mlcmd utility path
mlcmdpath= <Specify the mlcmd path >
# For example mlcmdpath = /opt/genesys/scs
echo -e "Checking mode for switchover - $application \n" >> $Logpath/Switchover.log
date >> $Logpath/Switchover.log
# checking application mode - switchover if backup
init_status=$( $mlcmdpath/mlcmd_64 -getappstatus-runmode $dbid -scshost $host
-scsport $port 2>&1 | grep -w "BACKUP")
if [ "$init_status" ]
then
echo $init_status >> $Logpath/Switchover.log
$mlcmdpath/mlcmd_64 -switchapp $dbid -scshost $host -scsport $port
else
```

```

echo "$application Application is already in primary mode or not started" >>
$Logpath/Switchover.log
exit
fi
sleep 3
# checking whether Application switchover occurred correctly
switch_status=$(($mlcmdpath/mlcmd_64 -getappstatus-runmode $dbid -scshost $host
-scsport $port 2>&1 | grep -w "PRIMARY")
if [ "$switch_status" ]
then
echo "Switchover of $application Success" >> $Logpath/Switchover.log
else
echo "Switchover of $application failed" >> $Logpath/Switchover.log
fi
echo -e "\n# $application Switchover End #" >> $Logpath/Switchover.log
echo -e "-----\n\n" >> $Logpath/Switchover.log

```

2. In the shell file, replace the variables indicated by angle brackets with appropriate values.
3. For SCS release 8.1.1 and higher, on all servers running SCS, create a shell file that is named `SC_APPLICATION_SWITCHOVER.SH`, and enter the following commands into the file:

[+] Commands for `SC_APPLICATION_SWITCHOVER.SH`

```

#!/bin/sh
# Set Application SCS and Config server details
# Application Details
application=<Name of the Application>
# Config server Details
host=<IP address of the config server>
port=<Port of the config server>
appname=<App name the config server>
user=<Config server user name>
password=<Config server password>
# Solution Control Server details
host=<IP address of the SCS server>
port=<Port of the SCS server>
# Set the path in which log files will be stored
Logpath= <Specify the path to log files>
# for example Logpath=/home/Logs
# Set the mlcmd utility path
mlcmdpath= <Specify the mlcmd path >
# For example mlcmdpath = /opt/genesys/scs
echo -e "Checking mode for switchover - $application \n" >> $Logpath/Switchover.log
date >> $Logpath/Switchover.log
# checking application mode - switchover if backup
init_status=$(($mlcmdpath/mlcmd_64 -getappstatus-runmode $application -cshost $host
-csport $port -csappname $appname -csuser $user -cspassword $password -scshost
$scs_host -scsport $scs_port 2>&1 | grep -w "BACKUP")
if [ "$init_status" ]
then
echo $init_status >> $Logpath/Switchover.log
$mlcmdpath/mlcmd_64 -switchapp $application -cshost $host -csport $port -csappname
$appname -csuser $user -cspassword $password -scshost $scs_host -scsport $scs_port
else
echo "$application Application is already in primary mode or not started" >>
$Logpath/Switchover.log
exit
fi
sleep 3
# checking whether Application switchover occurred correctly

```

```
switch_status=$(mlcmdpath/mlcmd_64 -getappstatus-runmode $application -cshost $host
-csport $port -csappname $appname -csuser $user -cspassword $password -scshost
$scs_host -scsport $scs_port 2>&1 | grep -w "PRIMARY")
if [ "$switch_status" ]
then
echo "Switchover of $application Success" >> $Logpath/Switchover.log
else
echo "Switchover of $application failed" >> $Logpath/Switchover.log
fi
echo -e "\n# $application Switchover End #" >> $Logpath/Switchover.log
echo -e "-----\n\n\n" >> $Logpath/Switchover.log
```

4. In the shell file, replace the variables indicated by angle brackets with appropriate values.

End

Next Steps

 [Genesys Applications](#)

OR

 [Back to Task Table](#)

Creating Genesys Applications for the control scripts

Application Object	Corresponding Script	Description
TP_PRIMARY_VIP_UP	SC_VIP_ON.BAT	Enables the Virtual IP address on the primary server.
TP_PRIMARY_VIP_DOWN	SC_VIP_OFF.BAT	Disables the Virtual IP address on the primary server.
TP_BACKUP_VIP_UP	SC_VIP_ON.BAT	Enables the Virtual IP address on the backup server.
TP_BACKUP_VIP_DOWN	SC_VIP_OFF.BAT	Disables the Virtual IP address on the backup server.
TP_VMSERVER-1_SWITCHOVER	SC_APPLICATION_SWITCHOVER.BAT	Switches the primary VM Server to primary mode.
TP_VMSERVER-2_SWITCHOVER	SC_APPLICATION_SWITCHOVER.BAT	Switches the backup VM Server to primary mode.
TP_VM-SIPSERVER-1_SWITCHOVER	SC_APPLICATION_SWITCHOVER.BAT	Switches the primary VM SIP Server to primary mode.
TP_VM-SIPSERVER-2_SWITCHOVER	SC_APPLICATION_SWITCHOVER.BAT	Switches the backup VM SIP Server to primary mode.

Creating Application objects for the shell files allows the shell files to be run as applications within the Genesys Framework.

Creating Genesys Applications for the control scripts



Start

1. In the Configuration Manager, select Environment > Applications.
2. Right-click and select New > Application.
3. Select the Third Party Server template from the Application Templates folder, and click OK.
4. On the General tab, enter a name for the Application object. For example, TP_PRIMARY_VIP_UP.
Note: You can use the suggested Application object names, or you can specify your own.
5. Select the Server Info tab.
 - a. For VIP control scripts, select the host name of the VM Server on which the corresponding control/shell script is located.
 - b. For application control scripts, select the host name of the server running SCS.
 - c. If necessary, specify a valid communication-port number by using the Edit Port option.

6. Select the Start Info tab.
7. Set the Working Directory to the location of the control/shell script, and enter the name of the script in the Command Line field.
For example, for the TP_PRIMARY_VIP_UP Application object, enter the script name that enables the Virtual IP address:
Windows: SC_VIP_UP.BAT
Linux: SC_VIP_UP.SH
For the PRIMARY_VIP_DOWN Application object, enter the script name that disables the Virtual IP address:
Windows: SC_VIP_DOWN.BAT
Linux: SC_VIP_DOWN.SH
8. Repeat the steps in this procedure to create an Application object for each of the scripts.

End

Next Steps

 [Creating Alarm Reaction scripts](#) OR  [Back to Task Table: Windows IP Address Takeover](#) OR [Linux IP Address Takeover](#)

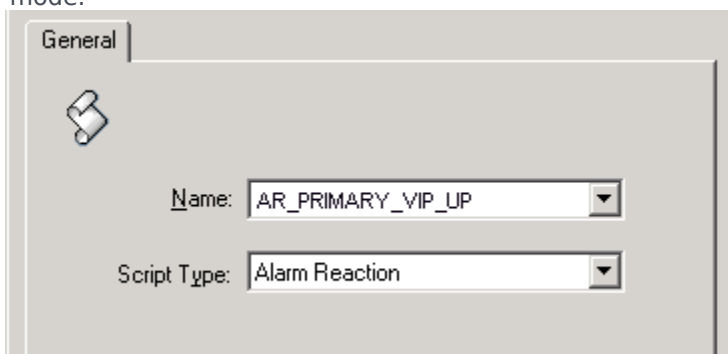
Creating the Alarm Reaction scripts

When an HA-related Alarm Condition occurs, the associated Alarm Reaction script is run. These scripts are configured to call the Application objects that you created in [Genesys Applications](#).

Creating the Alarm Reaction scripts

Start

1. Open the Configuration Manager.
2. Select Resources > Scripts.
3. Right-click and select New > Script.
4. Create alarm reaction scripts: one for each of the third-party Application objects that you previously created. For example:
 - AR_PRIMARY_VIP_UP — Triggers a script that enables the Virtual IP address (to be run on the primary VM Server).
 - AR_PRIMARY_VIP_DOWN — Triggers a script that disables the Virtual IP address (to be run on the primary VM Server).
 - AR_BACKUP_VIP_UP — Triggers a script that enables the Virtual IP address (to be run on the backup VM Server host).
 - AR_BACKUP_VIP_DOWN — Triggers a script that disables the Virtual IP address (to be run on the backup VM Server).
 - AR_VMSERVER-1_SWITCHOVER — Triggers a script that switches VM Server-1 to primary mode.
 - AR_VMSERVER-2_SWITCHOVER — Triggers a script that switches VM Server-2 to primary mode.
 - AR_VM-SIPSERVER-1_SWITCHOVER — Triggers a script that switches VM SIP Server-1 to primary mode.
 - AR_VM-SIPSERVER-2_SWITCHOVER — Triggers a script that switches VM SIP Server-2 to primary mode.

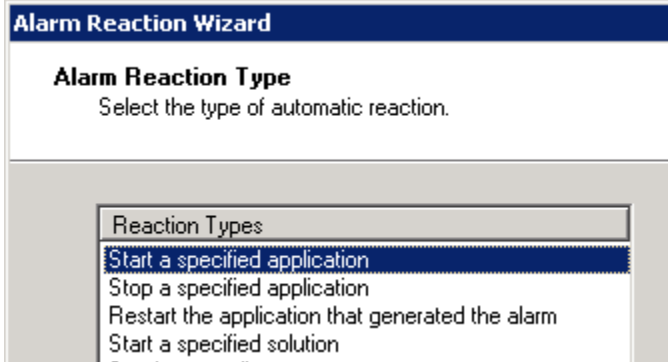


5. For each of the Alarm Reaction scripts, select Alarm Reaction as the Script Type.

6. For each of the Alarm Reaction scripts, use the Alarm Reaction Wizard to configure the Alarm Reaction Type.

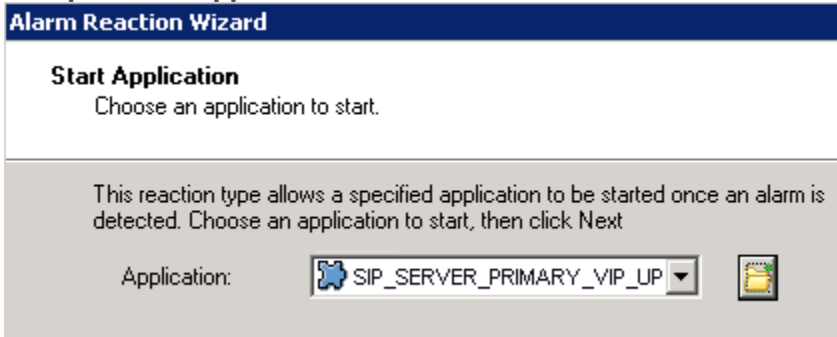
- a. Select an Alarm Reaction script, and right-click to open the Alarm Reaction Wizard (select Wizard > Configure).
- b. In the Alarm Reaction Wizard, click Next.

Select Start Application (used for VIP takeover scripts)



- c. In the Alarm Reaction Type dialog box, select Start a specified application or Stop a specified application, and click Next.
- d. Browse to select the corresponding Application object. For example, for the AR_PRIMARY_VIP_UP Alarm Reaction script, select the TP_PRIMARY_VIP_UP Application object of type Third Party Server.

Sample Start Application



- e. Repeat the previous steps to configure each of the Alarm Reaction scripts that you created in Step 4.

Next Steps

➡ [SCS Alarm Conditions](#) OR ➡ Back to Task Table: [Windows IP Takeover](#) OR [Linux IP Takeover](#)

Creating the SCS Alarm Conditions

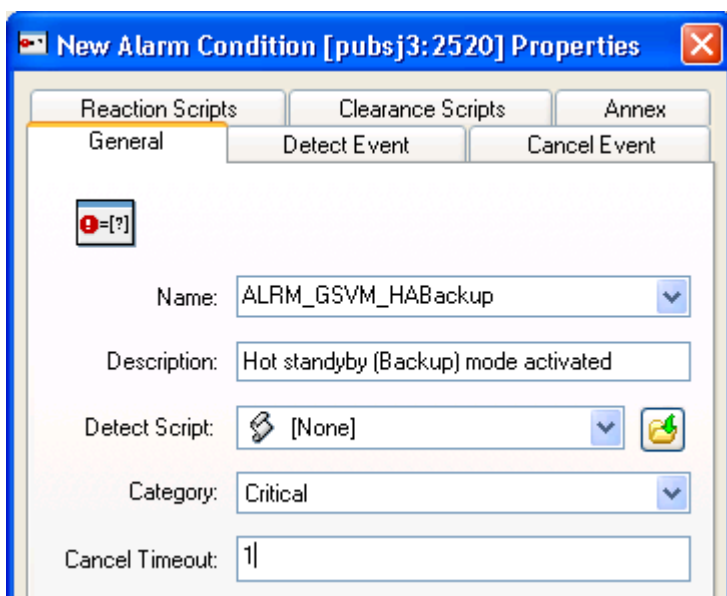
This page provides the procedure for creating SCS alarm conditions:

- [Creating the SCS Alarm Conditions](#)

Creating the SCS Alarm Conditions

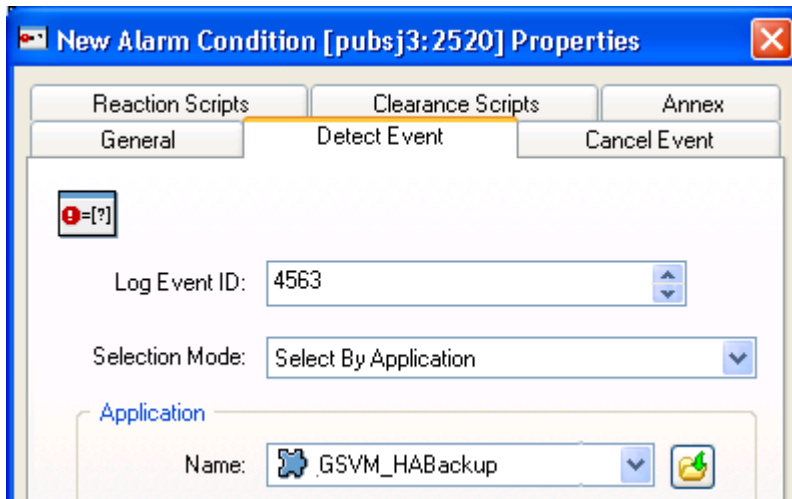
Start

1. Open the Configuration Manager.
2. Navigate to the Environment > Alarm Conditions folder.
3. Right-click and select New > Alarm Condition to open the New Alarm Condition Properties dialog box.
4. On the General tab:
 - Enter a Name for the Alarm Condition—for example, ALRM_GSVM_HABackup.
 - Optionally, enter a description.
 - For the Category value, select Critical.
 - Set Cancel Timeout to 1.



5. On the Detect Event tab:
 - Set the Log Event ID as defined in HA Events and Scripts ([Traditional](#), [Solution](#)).
 - Set the Selection Mode to Select By Application.

- For the Application Name field, click the folder icon to browse for the Application object. For example, if you are creating an Alarm Condition for the primary GSVM Server, select the primary GSVM Server Application object.



6. Click OK.
7. On the Reaction Scripts tab, add the Alarm Reaction script as defined in HA Events and Scripts ([Traditional](#), [Solution](#)).
8. Repeat the steps in this procedure to create each of the Alarm Conditions for the hot standby-related log events for VM SIP Server and VM Server, and the application failure-related log events for Resource Manager.

End

Next Steps ➡ [Testing Alarm Conditions](#) OR ➡ Back to Task Table: [Windows IP Takeover](#) OR [Linux IP Takeover](#)

Testing Alarm Conditions

Start

1. Use Telnet to access the VM Server Virtual IP interface.
2. Open the Solution Control Interface (SCI).
3. Under Alarm Conditions, select the Alarm Condition that you created in [SCS Alarm Conditions](#).
For example:
ALRM_VM_HABackup — right-click it, and then click Test.
The ALRM_VM_HABackup Alarm Condition indicates that the primary VM Server is in backup mode, which triggers the Alarm Reaction scripts that disable the Virtual IP interface at the primary VM Server and enable the Virtual IP interface at the backup VM Server.
4. Use an `ipconfig` command to verify that the Virtual IP interface is active on the backup VM Server and that the Virtual IP interface is inactive on the primary VM Server.

End

Next Steps  Back to Task Table: IP Takeover ([Windows](#), [Linux](#))

Document Change History

Check out the changes in the latest version of the SIP Voicemail HA Deployment Guide.

New in the SIP Voicemail HA Deployment Guide [07/20/12]

- [The IP Interface Architecture diagram](#) has been added to show the connections between Genesys components and the Voicemail host.
- Simplified architecture diagrams have been added for [Traditional HA Architecture](#) and [Solution HA Architecture](#).
- Details about [Traditional HA mode](#) have been added.
- Events, Control Scripts, and third-party application names have been updated. See Events and Scripts ([Traditional](#), [Solution](#)).
- VIP Control Scripts ([Linux](#), [Windows](#)) and Application Control Scripts ([Linux](#), [Windows](#)) have been updated.
- [Known Issues and Recommendations](#) have been added.
- Procedures to configure MCP shared voicemail storage for ([Linux](#) and [Windows](#)) have been added.