LifeKeeper® for Linux

Samba Recovery Kit v5.1.1
Administration Guide

February 2011

It is the policy of SIOS Technology Corp. (previously known as SteelEye Technology, Inc.) to improve products as new technology, components, software, and firmware become available. SIOS Technology Corp., therefore, reserves the right to change specifications without prior notice.

To maintain the quality of our publications, we need your comments on the accuracy, clarity, organization, and value of this book.

Address correspondence to:
ip@us.sios.com

Table of Contents

Samba Recovery Kit Administration Guide

The LifeKeeper for Linux Samba Recovery Kit provides fault resilient protection for Samba file and print shares on a Linux server existing in a heterogeneous network. This enables a failure on the primary Samba server to be recovered on a designated backup server without significant lost time or human intervention.

Document Contents

This guide contain the following topics:

- **LifeKeeper Documentation.** Provides a list of LifeKeeper for Linux documentation and where to find it.
- **Recovery Kit Requirements.** Describes the hardware and software necessary to properly setup, install, and operate the Samba Recovery Kit. Refer to the *LifeKeeper for Linux Planning and Installation Guide* for specific instructions on how to install or remove LifeKeeper for Linux software.
- **Samba Recovery Kit Overview.** Provides a brief description of the Samba Recovery Kit's features and functionality.
- **Configuring Samba with LifeKeeper.** Describes the key parts of the Samba configuration file that affect LifeKeeper, followed by configuration examples and procedures for properly configuring the Samba Recovery Kit.
- **LifeKeeper Configuration Tasks.** Describes the tasks for creating and managing your Samba resource hierarchies using the LifeKeeper GUI.
- **Samba Hierarchy Administration**. Provides information about tasks that may be required after your Samba resources are created.

LifeKeeper Documentation

The following is a list of LifeKeeper related information available from SIOS Technology Corp.:

- *LifeKeeper for Linux Release Notes*
- *LifeKeeper for Linux Online Product Manual*
- *LifeKeeper for Linux Planning and Installation Guide*

This documentation, along with documentation associated with other LifeKeeper Recovery Kits, is available online at:

http://us.sios.com/support

Recovery Kit Requirements

Your LifeKeeper configuration must meet the following requirements **prior** to the installation of the Samba Recovery Kit. Please see the *LifeKeeper for Linux Planning and Installation Guide* for specific instructions regarding the configuration of your LifeKeeper hardware and software.

- **Servers.** The Recovery Kit requires two or more servers configured in accordance with the requirements described in the *LifeKeeper for Linux Planning and Installation Guide*. See the *Release Notes* for supported Linux distributions.
- **LifeKeeper software.** You must install the same version of LifeKeeper software and any patches on *each* server.
- **LifeKeeper IP Recovery Kit.** You must have the same version of the LifeKeeper IP Recovery Kit on *each* server.
- **IP network interface.** Each server requires at least one Ethernet TCP/IP-supported network interface. In order for IP switchover to work properly, user systems connected to the local network should conform to standard TCP/IP specifications.
  Note: Even though each server requires only a single network interface, you should use multiple interfaces for a number of reasons: heterogeneous media requirements, throughput requirements, elimination of single points of failure, network segmentation, and so forth.
- **TCP/IP software.** Each server also requires the TCP/IP software.
- **Samba software**. Samba is delivered with all Linux distributions that LifeKeeper for Linux supports. The Samba Recovery Kit has been tested on Red Hat, SUSE and Miracle Linux.

Recovery Kit Installation

Please refer to the *LifeKeeper for Linux Planning and Installation Guide* for specific instructions on the installation and removal of the LifeKeeper for Linux software.

Samba Recovery Kit Overview

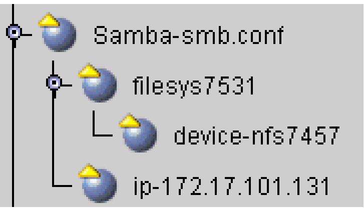
Samba is a suite of applications that speak the Server Message Block (SMB) protocol, allowing a Linux server to communicate in a heterogeneous network with servers and clients running Microsoft Windows products.

The Samba Recovery Kit enables LifeKeeper to protect Samba file and print shares on a Linux server. While Samba provides other services such as client authentication, Network Neighborhood browsing assistance and WINS name server resolution, this release of LifeKeeper does not protect these additional services. These other Samba services may coexist on a LifeKeeper server running as an unprotected instance of Samba as long as they adhere to the rules specified in the section Running Multiple Instances of Samba.

The Samba Recovery Kit provides a mechanism to recover protected Samba file and print shares from a failed primary server onto a backup server. LifeKeeper can detect failures either at the server level (via heartbeat) or resource level (by monitoring the Samba daemons) so that control of the Samba resources is transferred to a backup server.

Samba Resource Hierarchies

A typical Samba hierarchy will be comprised of a Samba resource, one or more file system resources, one or more IP resources, and possibly a print services resource. An example of a resource hierarchy protecting a Samba file share is shown below:



This *Samba-smb.conf* hierarchy protects one fileshare *filesys7531* (which is dependent upon the partition *device-nfs7457)*, and one IP address 172.17.101.131. The following sections describes how the Samba resources are configured.

Configuring Samba with LifeKeeper

There are a number of Samba configuration considerations that need to be made before attempting to create LifeKeeper for Linux Samba resource hierarchies. Samba services on a Linux server are provided by two daemon processes, **smbd** and **nmbd**. These daemon processes are controlled by the values defined in the Samba configuration file which is described below.

The Samba Configuration File

While a Samba configuration file can contain many different directives, this description focuses on those aspects of the configuration file that affect your LifeKeeper configuration. Here are some key points about the configuration file:

- The configuration file is comprised of sections which correspond to the share (or service) they provide. Each section of the configuration file contains individual configuration options (or directives) unique to that share.
- The directives that are specified are sanity checked by the Samba Recovery Kit. Failure to set the directives properly will cause Samba resource creation to fail.
- The default configuration file for Samba is typically named *smb.conf* and resides in */etc* or */etc/samba* depending on the Linux distribution.
- Configuration file names must be unique within the cluster, or must reside in a different directory on each server for Active/Active configurations. The unique naming or location is required as the Samba Recovery Kit replicates a copy of the configuration file during extension to the same location on the backup server.
- If you are running only one instance of the Samba daemons in an Active/Standby configuration, you may use the default configuration (this is recommended). However, all Samba daemons under LifeKeeper protection are managed by LifeKeeper and must not be started automatically during system startup via the init script */etc/init.d/smb* (or */etc/rc.d/init.d/smb* on some distributions).
- If more than one version of Samba will be running in an Active/Standby configuration or if you use an Active/Active configuration, unique Samba configuration file names are required. See Running Multiple Instances of Samba for more requirements and information on running multiple versions of Samba.

The following sections of this document describe the sections of the configuration file, including the options required for LifeKeeper to protect a Samba share.

[Global] Section of the Configuration File

The `[global]` section is a special section in the configuration file that must appear in every configuration file used in a LifeKeeper Samba resource hierarchy. As the name implies, any options set in this section apply to all other sections unless that directive is called out specifically in the other sections. LifeKeeper requires that certain directives be defined in the `[global]` section. Some of these directives may not exist in a default configuration file and will therefore need to be added. They are:

- **`netbios name`** – The unique name given to the set of resources that comprise a LifeKeeper Samba hierarchy. This is the name used by clients to connect to the shares via the IP addresses defined in the `interfaces` directive (e.g. NetBIOS name = server1_print1).
- **`interfaces`** – The list of network addresses for the Linux Samba server to recognize and respond. Here are the requirements for properly configuring the `interfaces` directive:
  - All subnets that are serviced by the Samba server must be listed. These must be LifeKeeper protected addresses and they must be unique within the cluster (no other Samba configuration file should use the same IP addresses).
  - The `interfaces` directive can have multiple formats, IP addresses (dot version or host name), and network interface names and can make use of wild cards. However, the Samba Recovery Kit requires the use of the IP address in dot format (100.25.104.25) without wild cards.
  - The subnet mask may be used in conjunction with the IP address but it is not used by LifeKeeper.
  - LifeKeeper IP resources for the address specified in this directive must be created prior to the creation of the Samba resource hierarchy. Additionally, if the network mask is applied to the addresses in this directive it must match the mask used on the IP resource when it was created.
  - Other non-protected instances of Samba should also use the `interfaces` directive, being sure to specify IP addresses *different* than those used by LifeKeeper.

  Note: Because of the use of the `bind interfaces only` directive discussed below, the `interfaces` directive may need to contain the localhost address of 127.0.0.1 to ensure proper operation of the utility **smbpasswd**. See Running Multiple Instances of Samba for information to help you determine whether the localhost address is needed.
- **`lock directory`** `(or lock dir)` – The name and location of a unique lock file location for the Samba instance on all servers. This directory must already exist on all servers in the cluster. **Note:** This directive is sometimes call **lock dir**. The Samba Recovery Kit will handle both directive names.
- **`bind interfaces only`** – This directive tells **smbd** and **nmbd** processes to serve SMB requests on the addresses defined in the `interfaces` directive only. It must be set to *Yes*. **Other non-protected instances of Samba running on the system must also have this directive set to *yes*.** When set to *yes*, Samba will not service requests on subnets that are not listed in the `interfaces` directive nor will it service requests for other instances of Samba that may be running on the server.

[Homes] Section of the Configuration File

The [homes] section is a special section in the configuration file to handle connection attempts to a user's home directory on a Samba server if it is not specifically defined as a share. LifeKeeper does **not** protect users' home directories via this special share; therefore **it should be removed or commented out**. In order for the LifeKeeper Samba Recovery kit to protect a Samba share it must have a `path` directive specified. The `path` directive is used to determine the file system that the LifeKeeper Samba hierarchy protects. The [homes] section does not have a path specified by default because the path is determined at the time a user makes a connection to the Samba server. It is for this reason that this special share must be removed or comment out.

[Printers] Section of the Configuration File

The `[printers]` section handles connection attempts to printers on a Samba server if it is not specifically defined as a share. LifeKeeper does not protect printer shares via this special section nor through the global directive `load printers`. Each LifeKeeper-protected printer share must be defined in its own share section in the configuration file.

Share Definition Sections of the Configuration File

All other sections in the configuration file define the file and/or print shares that clients can attempt to access for this instance. A configuration file must have one or more shares defined. The Samba configuration file can contain file shares only, print shares only or a combination of both file and print shares. LifeKeeper does not limit the number of shares that can be defined, but one must realize that a failure relating to any one share could cause the entire hierarchy to be switched over to the backup server. The following directives must be defined for each share:

- **`path`** – This directive identifies the pathname at the root of the file or print share. The value determines the File System resource to be protected as part of the Samba hierarchy. If the LifeKeeper File System resource does not already exist when the Samba resource is created, LifeKeeper will create it for you. **Note**: This directive is sometimes called **`directory`**. The recovery kit will handle both directive names.
- **`printable`** – A *Yes* value indicates that the Samba share is used as a print spool repository for printing to Linux printers. If the share is to be a regular file share then set this directive to *No* or do not specify it, as it is *No* by default unless set to *Yes* in the [global] section. If this directive is set to *Yes*, then creation of a Samba hierarchy will require the existence of LifeKeeper Print Services resource that protects the printer defined via the `printer name` directive listed below. **Note**: This directive is sometimes called **`print ok`**. The recovery kit will handle both directive names.
- **`printer name`** – This directive defines the printer name used by the share and is used to find a Print Services instance that protects the named printer. The Print Services instance will become a child resource in the Samba hierarchy. If this directive is not defined for a printer share, the Samba Recovery Kit will use the share name as the printer name. **Note**: This directive is sometimes called **`printer`**. The kit will handle both directive names.

Note: The Samba configuration file allows the use of variable substitution for a number of directives. Variable substitution should not be used for any of the directives specified above unless the variable is resolved by the Samba utility *testparm*.

Running Multiple Instances of Samba

Running multiple instances of Samba in a LifeKeeper cluster introduces additional configuration requirements and restrictions. The following Samba configuration scenarios may involve multiple instances of Samba:

- Active/Standby configuration with multiple LifeKeeper Samba instances on one server
- Active/Active configuration with multiple LifeKeeper Samba instances on more than one server

Either of these configurations could include a non-LifeKeeper protected version of Samba.

As previously noted in Configuring Samba With LifeKeeper, when running multiple instances of Samba each version must have a uniquely named configuration file, or the files must reside in different directories. Within each configuration file a number of directives are required and must be unique - in particular, `netbios name`, lock directory, pid directory, `interfaces` and `log file`. If these directives are not unique, Samba may not startup and therefore will not be available for client connections. Additionally, the `lock, log file`, and pid directories specified for each instance must exist on all servers in the cluster.

smbpasswd Utility and Multiple Instances of Samba

Although not required by LifeKeeper, some Samba utilities used by the Samba Recovery Kit expect to be able to open *smb.conf* in its default location. The Recovery Kit uses the **smbclient** and **nmblookup** utilities to connect to **smbd** and **nmbd** (respectively) in order to determine the health of the daemon processes while under LifeKeeper protection. These two utilities will not error out if they do not find *smb.conf* in its default location. However, *smb.conf* is required by the **smbpasswd** utility to be in its default location.

smbpasswd is used to maintain the *smbpasswd* file for authentication of users on client connection requests when the security level is set to *share* or *user*. If the default configuration file is missing, any attempt to change Samba passwords will fail. To avoid this problem, one of the instances of Samba must use the default configuration file if the security level is set to *share* or *user,* or if the server is acting as the **smbpasswd** server for those systems with Samba security level set to *server*. The reason for this is that **smbpasswd** uses the default configuration file to obtain the location of the *smbpasswd* file. Because of this requirement only one location for the *smbpasswd* file can exist within the LifeKeeper cluster. The configuration files for all instances of Samba in the cluster must have the directive `smb passwd file` set to the same value. Additionally, the *smbpasswd* file must be kept in sync on all servers in the cluster.

The **smbpasswd** utility is also affected by the use of the `bind interfaces only` directive, which is required by the LifeKeeper Samba Recovery Kit. With the `bind interfaces only` directive set to *Yes*, a regular user changing his Samba password will attempt to connect to a **smbd** daemon process using the localhost address of 127.0.0.1. If that address has been added to the `interfaces` directive in the configuration file used by the **smbd** daemon, and if **smbd** has connected to and is listening on that address, then the password change will be successful. If the daemon does not have that address in its configuration file `interfaces` directive, then the password change will fail. In a multiple instance environment, if the localhost is specified in more than one configuration file, only one instance will be able to start up and run. Using the `-r netbios_name` option to **smbpasswd** will work in place of adding the localhost address to the interfaces list (for example: **smbpasswd -r server1 print1...**)

Note: As previously stated, non-protected Samba instances running on a LifeKeeper server with protected Samba instances must also have the `bind interfaces only` directive set to "Yes."

Samba and User Authentication Considerations

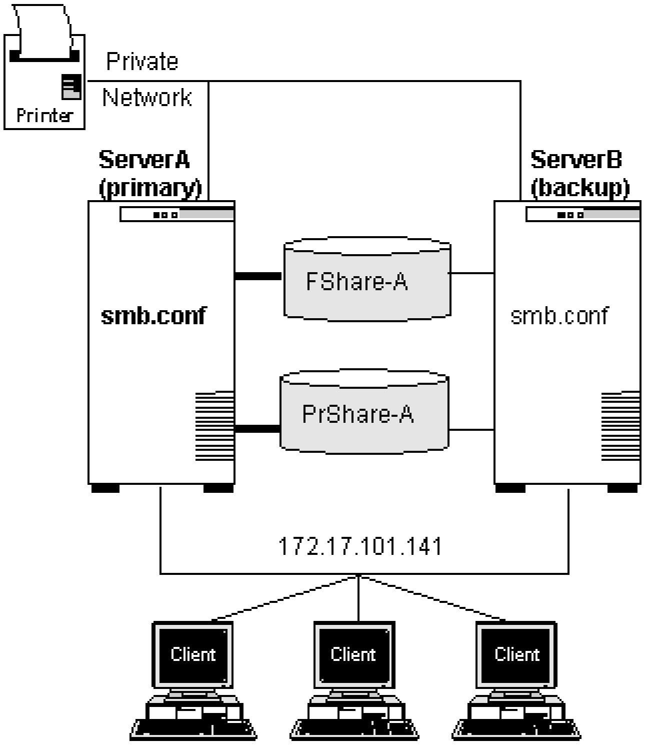
Samba supports several methods for user authentication via the security parameter (e.g. share, user, domain, ...) which must be considered when protecting Samba via LifeKeeper to ensure data files such as /etc/samba/smbpasswd or /etc/samba/secrets.tdb are kept in sync on all servers in the cluster. So when using security methods such as user, you must ensure that the smbpasswd file is kept in sync on all servers in the cluster. Additionally, security methods such as domain require synchronization of the secrets.tdb file. A LifeKeeper active/active configuration with the secrets.tdb file requires the use of the private dir parameter to specify the location of the file. The value for this parameter must be unique for each LifeKeeper Samba instance.

Samba Configuration Examples

This section contains definitions and examples of typical Samba configurations. Each example includes the configuration file entries that apply to LifeKeeper.

Active/Standby Configuration

In the Active/Standby configuration, ServerA is the primary LifeKeeper server. It exports the file and print shares that reside on a shared storage device. While ServerB may be handling other applications/services, it acts only as a backup for the Samba resources in LifeKeeper's context.

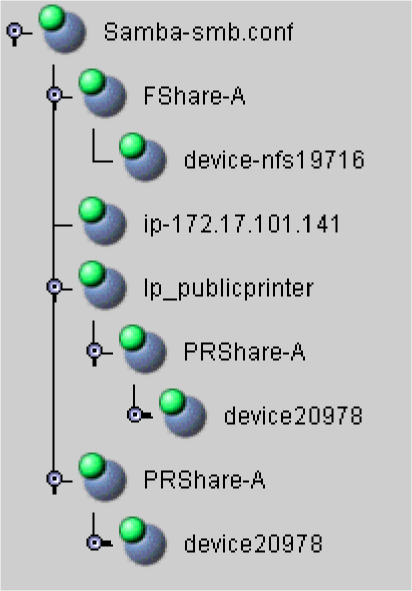


Configuration Notes:

- The clients connect to the Samba servers using the NetBIOS name LKServerA over the protected IP address (172.17.101.141), which is defined by the `interfaces` directive of the configuration file.
- The configuration file *smb.conf* has been copied to ServerB upon extension of the Samba resource hierarchy. It contains the following directives:

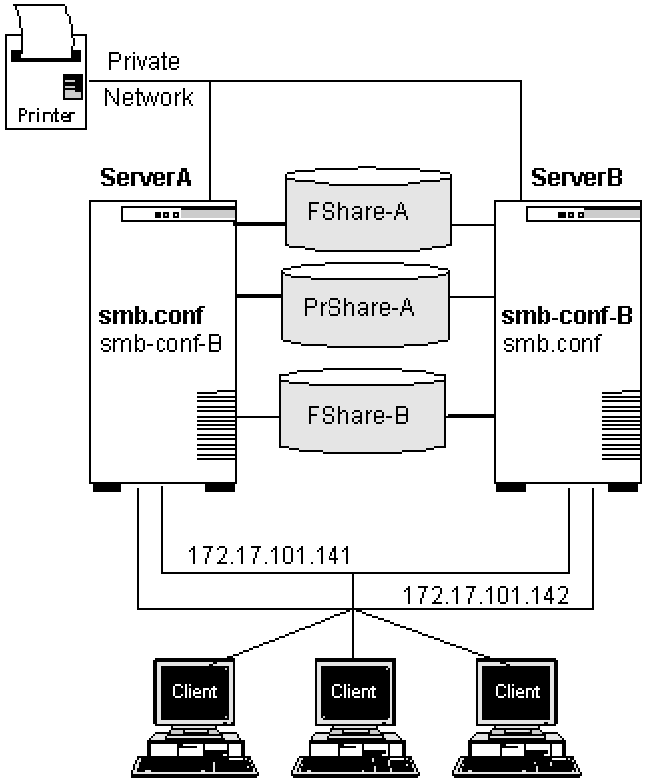
```
[global]
netbios name = LKServerA
bind interfaces only = yes
lock directory = /var/lock/samba
interfaces = 172.17.101.141 127.0.0.1
log file = /var/log/sambaServA/log

[FShare-A]
path = /FShare-A
read only = no
public = yes
valid users =
printable = no
create mode = 0664
directory mode = 0775

[PRShare-A]
path = /PRShare-A
printer = publicprinter
printable = yes
browseable = no
```

- The Samba resource hierarchy would look like the following:



Active/Active Configuration

In the Active/Active configuration below, both ServerA and ServerB are primary LifeKeeper servers for Samba resources. Each server is also the backup server for the other.



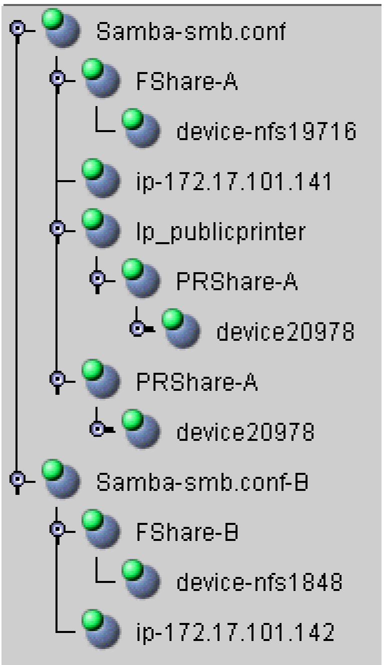
Configuration Notes:

- The clients connect to the Samba servers using the NetBIOS name LKServerA and LKServerB over the protected IP addresses (172.17.101.141 and 172.17.101.142 respectively), which are defined by the `interfaces` directive of the configuration files.
- The configuration file *smb.conf* was copied to ServerB upon extension of the Samba resource hierarchy. Likewise, the configuration file *smb.conf-B* was copied to ServerA upon extension of the Samba resource hierarchy.
- ServerA protects the file share */Fshare-A*; ServerB protects the file share */Fshare-B.*
- ServerA protects the print share */publicprinter*; ServerB does not protect a print share.

- The two configuration files contain the following directives:

smb.conf	*smb-conf-B*
[global] netbios name = LKServerA bind interfaces only = yes lock directory = /var/lock/sambaServA interfaces = 172.17.101.141 127.0.0.1 log file = /var/log/sambaServA/log [FShare-A] path = /FShare-A read only = no public = yes valid users = printable = no create mode = 0664 directory mode = 0775 [PRShare-A] path = /PRShare-A printer = publicprinter printable = yes browseable = no	[global] netbios name = LKServerB bind interfaces only = yes lock directory = /var/lock/sambaServB interfaces = 172.17.101.142 log file = /var/log/sambaServB/log [FShare-B] path = /FShare-B read only = no public = yes valid users = printable = no create mode = 0664 directory mode = 0775

- The Samba resource hierarchies would look like the following:



Samba Configuration Steps

This section provides steps that you should take to configure your Samba resources.

1. Plan your Samba configuration. This includes the following:

   – NetBIOS name(s) to be used

   – the interfaces that will be protected and allowed access to the shares

   – the file systems to be used for the Samba shares and thus protected

   – the printers if any to be accessed and protected

   – the location of the lock and log directory (or directories).

   Consideration should be given to the number of configuration files to be used and the type of configuration (Active/Standby vs. Active/Active). For example, if you have four Samba shares to protect, you could list all four shares in one configuration file, with the disadvantage that a failure of any one file system will cause the failover of the entire Samba hierarchy, including all four file shares. Alternatively, you could create four separate configuration files, each protecting one file share, which requires that four NetBIOS names be defined and managed.

2. Setup your Samba configuration file(s) based on the plan made in step 1. This includes the required directives in the `[global]` section as well as those for the file and print shares to be used. See Configuring Samba with LifeKeeper for a discussion of the global and share directives required for LifeKeeper Samba hierarchies.

3. Create protected IP addresses under LifeKeeper, which will be used for client connections to the Samba server via the NetBIOS name. The protected IP address(es) should match the value(s) placed in the `interface` directive in the configuration file. (Refer to the *LifeKeeper for Linux IP Recovery Kit Administration Guide* for details on setting up IP resources.) Test the protected IP addresses by pinging them from all clients and other cluster servers. A protected IP resource for the local host (127.0.0.1) is not required.

4. Start the Samba daemons and test client connections.

   a. The commands to start the daemons are as follows:

      s/nmbd –D –s ConfigurationFile

   b. Use the Samba utility **smbclient** to test connections to the **smbd** daemon as follows. This should be done for each address defined in the `interfaces` directive.

      smbclient –L netbios_name –U% -I Protected_IP_Address

   c. Use **nmblookup** to test connection to the **nmbd** daemon process. This should be done for each broadcast address. Use the associated broadcast address for each address defined in the `interfaces` directive. (The broadcast address can be obtained by running **ifconfig**).

      nmblookup –B broadcast_address netbios_name

5. Stop the Samba daemons started in the previous step. This is accomplished via the **kill** command. Find the running daemon processes via the **ps** command and issue a **kill pid** which will cause them to exit.

6. Create protected file system(s) under LifeKeeper that will host the Samba file and print shares as defined in the above steps. (Refer to the *LifeKeeper for Linux Online Product Manual* for information on creating a File System resource hierarchy.) This step may be skipped since File System resources will be created automatically when creating a Print Services resource or Samba resource.
7. Create directories on the protected file systems for the shares should one file system be used for multiple Samba shares.
8. Create protected Print Services hierarchies under LifeKeeper, which will be used for client printing should any printer shares be defined in the configuration file. (Refer to the *LifeKeeper for Linux Print Services Recovery Kit Administration Guide* for details on setting up Print Services resources.)
9. Create the Samba resource hierarchy in LifeKeeper and extend it to at least one backup server (see the LifeKeeper Configuration Tasks section below). The *extend* script will copy the Samba configuration file from the template server to the same location on the target server.
10. On the primary server, test client connections to the shares that are protected by the Samba hierarchy which is in service. For instance, map the shared directory from a Windows client and ensure that it can access files on the share. Repeat the test for all servers in the cluster. You should also test your Samba resource by performing a manual switchover to a backup server. (See Testing Your Resource Hierarchy.)

LifeKeeper Configuration Tasks

You can perform all LifeKeeper for Linux Samba Recovery Kit administrative tasks via the LifeKeeper Graphical User Interface (GUI). The LifeKeeper GUI provides a guided interface to configure, administer, and monitor Samba resources.

Overview

The following tasks are described in this guide, as they are unique to a Samba resource instance, and different for each Recovery Kit.

- **Create a Resource Hierarchy** - Creates a Samba resource hierarchy
- **Delete a Resource Hierarchy** - Deletes a Samba resource hierarchy
- **Extend a Resource Hierarc**hy - Extends a Samba resource hierarchy from the primary server to the backup server.
- **Unextend a Resource Hierarchy** - Unextends (removes) a Samba resource hierarchy from a single server in the LifeKeeper cluster.

The following tasks are described in the GUI Administrative Tasks section within the *LifeKeeper Online Product Manual*, because they are common tasks with steps that are identical across all Recovery Kits.

- **Create Dependency** - Creates a child dependency between an existing resource hierarchy and another resource instance and propagates the dependency changes to all applicable servers in the cluster.
- **Delete Dependency** - Deletes a resource dependency and propagates the dependency changes to all applicable servers in the cluster.
- **In Service** - Activates a resource hierarchy.
- **Out of Service** - Deactivates a resource hierarchy.
- **View/Edit Properties** - View or edit the properties of a resource hierarchy.

Note: Throughout the rest of this section, configuration tasks are performed using the **Edit** menu. You can also perform most of these tasks:

- from the toolbar
- by right clicking on a global resource in the left pane of the status display
- by right clicking on a resource instance in the right pane of the status display

Using the right-click method allows you to avoid entering information that is required when using the **Edit** menu.

Creating a Samba Resource Hierarchy

After you have completed the necessary setup tasks, use the following steps to define the Samba resource hierarchy.

1. From the LifeKeeper GUI menu, select **Edit**, then **Server**. From the menu, select **Create Resource Hierarchy**.

   The *Create Resource Wizard* dialog box will appear with a drop down list box displaying all recognized Recovery Kits installed within the cluster.
2. Select Samba Share and click **NEXT**.
3. You will be prompted to enter the following information. When the **Back** button is active in any of the dialog boxes, you can go back to the previous dialog box. This is helpful should you encounter an error requiring you to correct previously entered information. You may click **Cancel** at any time to cancel the entire creation process.

Field	**Tips**
Switchback Type	Choose either *intelligent* or *automatic* This dictates how the Samba instance will be switched back to this server when the server comes back up after a failover. The switchback type can be changed later from the General tab of the Resource Properties dialog box. **Note:** The switchback strategy should match that of the Print Server, IP or File System resource to be used by the Samba resource. If they do not match the Samba resource creation will attempt to reset them to match the setting selected for the Samba resource.
Server	Select the Server on which you want to create the hierarchy.

Location of Configuration File	Select the directory where the Samba configuration file is located.
Config File Name	Enter the name of the Samba configuration file to be used for this resource creation. The default is *smb.conf*. **Note**: LifeKeeper will read the selected configuration file, and if the file does not specify the required directives, LifeKeeper will generate an error message. It does minimal checking of the configuration file (to verify that shares exist, that they have a `path` directive, that a lock directory has been specified, and that the directory exists). Additional checking is done during the creation process.
Root Tag	Either select the default root tag offered by LifeKeeper, or enter a unique name for the resource instance on this server. The default is *Samba-configfilename,* where *configfilename* is the name of the associated configuration file. You may use letters, numbers and the following special characters: - _ . /

4. Click **Create. T**he *Create Resource Wizard* will then create your Samba resource hierarchy. LifeKeeper will validate the data entered. If LifeKeeper detects a problem, an error message will appear in the information box.

5. An information box will appear indicating that you have successfully created a Samba resource hierarchy, and you must Extend that hierarchy to another server in your cluster in order to achieve failover protection. Click **Next**.

6. Click **Continue.** LifeKeeper will then launch the *Pre-Extend Wizard*. Refer to Step 2 under Extending a Samba Resource Hierarchy (below) for details on how to extend your resource hierarchy to another server.

Note: See Failure Restoring Samba Hierarchy in the Samba Troubleshooting section for tips to follow in the case that the creation of the Samba hierarchy fails.

Extending a Samba Hierarchy

This operation can be started from the **Edit** menu, or initiated automatically upon completing the **Create Resource Hierarch**y option, in which case you should refer to Step 2 below.

1. On the **Edit** menu, select **Resource**, then Extend Resource Hierarchy. The Pre-Extend Wizard appears. If you are unfamiliar with the Extend operation, click **Next**. If you are familiar with the LifeKeeper **Extend Resource Hierarchy** defaults and want to bypass the prompts for input/confirmation, click **Accept Defaults**.

2. The *Pre-Extend Wizard* will prompt you to enter the following information.
   Note: The first two fields appear only if you initiated the Extend from the **Edit** menu.

Field	Tips
Template Server	Enter the server where your Samba resource is currently in service.
Tag to Extend	Select the Samba resource you wish to extend.
Target Server	Enter or select the server you are extending *to*.
Switchback Type	Select either intelligent or automatic. The switchback type can be changed later, if desired, from the General tab of the Resource Properties dialog box. **Note:** Remember that the switchback strategy must match that of the dependent resources to be used by the Samba resource.
Template Priority	Select or enter a priority for the template hierarchy. Any unused priority value from 1 to 999 is valid, where a lower number means a higher priority (the number 1 indicates the highest priority). The extend process will reject any priority for this hierarchy that is already in use by another system. The default value is recommended. **Note:** This selection will appear only for the initial extend of the hierarchy.
Target Priority	Either select or enter the priority of the hierarchy for the target server.
Root Tag	LifeKeeper will provide a default tag name for the new Samba resource instance on the target server. The default tag name is the same as the tag name for this resource on the template server. If you enter a new name, be sure it is unique on the target server. You may use letters, numbers and the following special characters: - _ . /

3. After receiving the message that the pre-extend checks were successful, click **Next**.
4. Depending upon the hierarchy being extended, LifeKeeper will display a series of information box showing the Resource Tags to be extended, which cannot be edited. Click **Extend.**
5. After receiving the message "Hierarchy extend operations completed" click **Next Server** to extend the hierarchy to another server, or click **Finish** if there are no other extend operations to perform.
6. After receiving the message "Hierarchy Verification Finished", click **Done**.

Unextending a Samba Hierarchy

To remove a resource hierarchy from a single server in the LifeKeeper cluster, do the following:

1. On the **Edit** menu, select **Resource**, then **Unextend Resource Hierarchy**.
2. Select the **Target Server** where you want to unextend the Samba resource. It cannot be the server where the Samba resource is currently in service. (This dialog box will not appear if you selected the Unextend task by right clicking on a resource instance in the right pane.) Click **Next**.
3. Select the Samba hierarchy to unextend and click **Next**. (This dialog will not appear if you selected the Unextend task by right clicking on a resource instance in either pane).
4. An information box appears confirming the target server and the Samba resource hierarchy you have chosen to unextend. Click **Unextend**.
5. Another information box appears confirming that the Samba resource was unextended successfully. Click **Done** to exit the Unextend Resource Hierarchy menu selection.

Deleting a Samba Hierarchy

It is important to understand what happens to dependencies and protected services when a Samba hierarchy is deleted.

- **Dependencies**: Before removing a resource hierarchy, you may wish to remove the dependencies. Dependent file systems *will* be removed unless they are used in another hierarchy. Dependent IP and Print Services resources will *not* be removed as long as the delete is done via the LifeKeeper GUI or the Samba delete script.
- **Protected Services:** If you take the Samba resource hierarchy out of service before deleting it, the Samba daemons will be stopped. If you delete a hierarchy while it is in service, the Samba daemons will continue running and offering services (without LifeKeeper protection) after the hierarchy is deleted.

To delete a resource hierarchy from all the servers in your LifeKeeper environment, complete the following steps:

1. On the **Edit** menu, select **Resource,** then **Delete Resource Hierarchy**.
2. Select the **Target Server** where you will be deleting your Samba resource hierarchy and click **Next**. (This dialog will not appear if you selected the Delete Resource task by right clicking on a resource instance in either pane.)
3. Select the **Hierarchy to Delete**. (This dialog will not appear if you selected the Delete Resource task by right clicking on a resource instance in the left or right pane.) Click **Next**.
4. An information box appears confirming your selection of the target server and the hierarchy you have selected to delete. Click **Delete**.
5. Another information box appears confirming that the Samba resource was deleted successfully.
6. Click **Done** to exit.

Testing Your Resource Hierarchy

You can test your Samba resource hierarchy by initiating a manual switchover. This will simulate a failover of a resource instance from the primary server to the backup server.

Selecting **Edit**, then **Resource**, then **In Service**. For example, an *In Service* request executed on a backup server causes the application hierarchy to be taken out of service on the primary server and placed in service on the backup server. At this point, the original backup server is now the primary server and original primary server has now become the backup server.

If you execute the *Out of Service* request, the application is taken out of service without bringing it in service on the other server.

Samba Hierarchy Administration

Once your Samba resource hierarchies are created, follow these guidelines for ongoing administration of your Samba resources.

Modifying the Samba Configuration File

When changes are required to a Samba configuration file that is used in a LifeKeeper Samba instance, perform these procedures on the server that is In Service, Protected (ISP). There are three types of configuration file changes:

- Those that do not directly impact the Samba hierarchy
- Those that directly impact the hierarchy but do not require a delete and recreation of hierarchy
- Those that directly impact the hierarchy and require a delete and recreation of the hierarchy.

See the sections below for information on how to identify the type of change you need to make, and the procedures to follow for each type.

Modifications that do not directly impact the Samba Hierarchy

Any changes to configuration file directives not used by LifeKeeper fall into this category. (See Configuring Samba with LifeKeeper for a list the directives used by the kit.) Example directives not used by LifeKeeper would include `security`, `hosts allow`, `hosts deny` and `valid users` to name a few. The procedures are as follows:

1. Take the Samba resource for the configuration file out of service. This step is required to stop the Samba daemons.
2. Make the necessary updates to the Samba configuration file.
3. Synchronize the configuration within the cluster. Use the utility **synccfg** to perform this task:

   LKROOT/lkadm/subsys/gen/samba/bin/synccfg –t TargetSys –c ConfigFile

   where *LKROOT* is the install location of LifeKeeper *(/opt/LifeKeeper* by default), *TargetSys* is the node to update and *ConfigFile* is the full path to the configuration file to copy.
4. Repeat the previous step for all servers in the hierarchy.
5. Bring the hierarchy back in service to restart the Samba daemons.

Modifications that directly impact the Samba Hierarchy

Any changes to configuration file directives used by LifeKeeper (see Configuring Samba with LifeKeeper for a list), with the exception of the `netbios name` or the physical movement of the configuration file, fall into this category. Depending on the extent of the changes, it may be quicker and easier to proceed to the third category and just recreate the hierarchy. The typical types of changes expected in this category include the addition of new file and print shares, removal of file and print shares or the addition or removal of IP interfaces.

1. Take the Samba resource for the configuration file out of service. This step is required to stop the Samba daemons.
2. Make the necessary updates to the Samba configuration file.

3. Make the necessary updates to the Samba hierarchy. This varies depending on the type of change made to the configuration file. For example:
   - If an additional IP address has been added to the `interfaces` directive, then a new IP resource needs to be created, extended and then added as a dependent child to the Samba resource hierarchy. See the *LifeKeeper Online Product Manual* for information on how to create dependencies.
   - If a new file share has been added to the configuration file, then a File System resource may need to be created, extended and added as a dependent child to the Samba resource hierarchy. If the File System resource already exists as a child in the hierarchy (e.g. the `path` directive defined for the new share has the same file system mount point as another file or print share) then it does not need to be created and added as a dependent child.
   - If a new print share is added, then File System and Print Services resources need to be created, extended and added as dependent children in the Samba hierarchy. If a print services resource does not exist that protects the printer as defined by the print share name or `printer/printer` name directive, then one must be created. See file share above to determine if a file system resource needs to be added.
   - If a file or print share is removed, or if an IP address is removed from the `interfaces` directives, delete the dependency in the Samba hierarchy and then delete the individual resource.
   - If a print share name is changed, follow the delete of print share followed by the addition of new print share.
4. Synchronize the configuration within the cluster. Use the utility **synccfg** to perform this task:

   LKROOT/lkadm/subsys/gen/samba/bin/synccfg –t TargetSys –c ConfigFile

   where *LKROOT* is the install location of LifeKeeper (*/opt/LifeKeeper* by default), *TargetSys* is the server to update and *ConfigFile* is the full path to the configuration file to copy.
5. Repeat the previous step for all servers in the hierarchy.
6. Bring the hierarchy back in service to restart the Samba daemons.

Note: If you are making a number of changes that require numerous resource creations and dependency additions or deletions, you may wish to create all the new resources before you take the Samba hierarchy out of service so that downtime is minimized.

Modifications that directly impact the Samba Hierarchy, requiring a deletion and recreation of the Hierarchy

If the `netbios name` directive is changed or the physical location of the configuration file is changed, then you must:

1. Delete the hierarchy. (See Deleting a Samba Hierarchy for details.)
2. Change the NetBIOS name or move the configuration file.
3. Create a new Samba hierarchy and extend to all backup servers.

Maintaining the *smbpasswd* File

Samba provides four different authentication methods via the `security` directive. The share and user security settings both require access to the local *smbpasswd* file to determine if access will be granted. As noted in the section Running Multiple Instances of Samba there can only be one *smbpasswd* file, and this presents a potential administration problem in a LifeKeeper cluster. If share or user level security is selected, the file must be kept in sync on all servers so that authentication will succeed after a failover.

In a cluster with only one Samba hierarchy, the use of share or user level security can be accomplished by placing the *smbpasswd* file on a file share defined in the configuration file. The access to this share should be such that only administrators have access. In a multiple instances scenario, either server level or domain level security is suggested.

Note: If firewalls are in use, ensure that the firewall will allow connections to the **smbd** daemon, and that the **nmblookup** will work.

Samba Troubleshooting

Failure restoring Samba hierarchy

Failure of a Samba hierarchy restore can leave the daemon processes **smbd** and **nmbd** running. (The restore operation is initiated via the completion of a create, failover, or manual switchover, or via a local recovery caused by a connection failure to **smbd** and **nmbd**.) If the daemons are not stopped and restarted after the problem is corrected, and the restore is attempted again, the restore could fail again.

Suggested Action

Correct the cause of the connection failure (for instance, an incorrect mask setup for the `interfaces` directive). Next, manually stop the **smbd** and/or **nmbd** daemons. Then bring the hierarchy in service, or re-create the hierarchy if the failure occurred during creation.

Stopping the daemons ensures that a re-read of the configuration file occurs during the restore.

Error Messages

Common Error Messages

Error Number	Error Message	Description
109009	Error getting netbios name from the instance information field for tag "Samba-smb.conf" on server "ServerA".	Extracting the NetBIOS name from the instance info failed. Check that the info field contains the configuration file and NetBIOS name.
109015	The Samba utility testparm failed. Unable to parse Samba configuration file.	The Samba utility **testparm** that is used to parse the configuration file failed. Run **testparm** from the command line specifying the configuration file used for the hierarchy to determine the failure.
109019	Failed to initialize for reading of the Configuration File "/tmp/smb.ini.1234".	Attempts to read the generated output of the configuration file failed. The utility **testparm** generated a bad file.
109022	Error getting configuration name from the instance information field for tag "Samba-smb.conf" on server "ServerA".	Extracting the NetBIOS name from the instance info failed. Check that the info field contains the configuration file and NetBIOS name.
109030	Failure opening "/var/lock/samba/smbd.pid" on server "ServerA": "File Not Found".	The attempted open of the daemon process ID file failed for the listed reason. Correct the problem based on the listed error code.

109050	Open of the testparm output file failed.	The open of output file created by running testparm failed because the file does not exist or does not contain any data. Run **testparm** from the command line specifying the configuration file used for the hierarchy to determine the failure.

Hierarchy Creation

Error Number	Error Message	Description
109001	Usage: “valid_cf” CfgPath CfgName TemplateSys	The **valid_cf** script requires three arguments, the directory containing the configuration file, the name of the configuration file and the name of the template system on which to validate the configuration file. You must specify all three.
109002	Must specify an absolute path to the ”smb.conf” configuration file.	You must specify the absolute path to the configuration file when running the scripts **choice_cf** and **valid_cf**.
109003	The file ”smb.conf” does not exist in ”/etc”.	You must specify the correct path to the configuration file when running **valid_cf** to validate the select configuration file.
109004	The path ”/export/fs” in ”/etc/samba/smb.conf” does not reside on a shared file system.	The Samba configuration shares must contain a path directive that can be protected by LifeKeeper via a File System resource. Edit the configuration file and change the path to a file system that LifeKeeper can protect.
109005	Usage: “choices_cf” CfgFile	The **choice_cf** script requires the full path to an existing configuration file. Please specify the correct path.
109006	Samba Configuration file not specified.	No configuration file was specified for the creation of the Samba resource hierarchy.
109007	Cannot bring hierarchy ”Samba-smb.conf” in service on server ”ServerA”. Action: After correcting the problem, try bringing the hierarchy in service manually.	The in service attempt at the end of creation failed. View the log file for possible reason for the failure.
109010	The Samba configuration file does not have the interfaces directive defined. This directive is required to create Samba File Share hierarchies.	The configuration file is missing the interfaces directive or the directive does not contain any IP addresses other than the localhost (127.0.0.1).
109011	The Samba configuration file does not have a correctly formatted interfaces directive. The interfaces directive must be in full dotted decimal IP address format with or without the mask parameter.	The interfaces directive must contain one or more IP addresses in the format of aaa.bbb.ccc.ddd or aaa.bbb.ccc.ddd/mask separated by a space.
109012	The Samba configuration file section "FileShare1" does not have a path directive defined. All Samba shares must have a path directive.	The specified share in the configuration file does not have a path directive or the directive does not contain a value.

Error Number	Error Message	Description
109014	No IP resources defined on server "ServerA". Action: Create IP resources for the IP addresses defined in the Samba configuration file interfaces directive.	The specified server does not contain any LifeKeeper protected IP resources needed for the creation of the Samba resource hierarchy.
109016	The IP(s) "100.25.104.25,100.35.104.26" defined in the interfaces directive are not under LifeKeeper protection. Action: Create IP resources for the unprotected addresses defined in the Samba configuration file interfaces directive.	LifeKeeper protected IP resources must exist for all of the IP addresses listed in the interfaces directive. Those listed are not protected by LifeKeeper.
109017	Missing configuration file name.	No configuration file exists when attempting to run **testparm** during Samba resource creation.
109018	No Samba shares found in "/etc/samba/smb.conf".	The Samba configuration specified for the resource must contain at least one file or print share.
109020	Bad configuration file. No section information found in the file "/tmp/smb.ini.1234".	The Samba configuration specified for the resource must contain at least one file or print share.
109021	Creation of Samba hierarchy with tag "Samba-smb.conf" on server "ServerA" failed.	The create of the Samba hierarchy failed. Examine the other error messages to determine the cause of the failure.
109023	The file system resource "filesys1328" is not in-service on server "ServerA".	The File System resource needed as a dependent child in the Samba hierarchy is not in service on the template server. Bring the resource in service and retry the resource creation.
109024	Selected IP resource "ip-100.25.104.26" does not exist on server "ServerA" Action: Retry the operation.	The specified IP resource tag no longer exists and is needed for the creation of the Samba hierarchy. Recreate the IP resource and retry the resource creation.
109025	LifeKeeper was unable to create a dependency between the Samba hierarchy "Samba-smb.conf" and the IP resource "ip-100.25.104.26" on server "ServerA".	The dependency creation attempt between the Samba resource and the IP resource failed. Examine the other messages to determine the cause of the failure.
109026	The Samba configuration file does not have a netbios name directive defined. Action: Add a netbios name directive in the global section of the configuration file.	All configuration files must contain a NetBIOS name. Add a NetBIOS name directive to the configuration file.
109029	The Samba configuration file "%s" directive defines a directory that does not exist. The %s can contain pid directory, lock dir, or lock directory.	All configuration files must contain an existing directory as specified by the directive. Add the missing directory.

Error Number	Error Message	Description
109035	The Samba directive "bind interfaces only" must be set to "Yes". Action: Change "bind interfaces only" to "Yes" and recreate the hierarchy.	All configuration files must have the “bind interfaces only” directive set to *Yes*. Correctly set the directive to *Yes*.
109036	Selected Printer resource "lp-admin" does not exist on server "ServerA" Action: Retry the operation.	The specified Print Services resource tag no longer exists and is needed for the creation of the Samba hierarchy. Recreate the Print Services resource and retry the resource creation.
109037	LifeKeeper was unable to create a dependency between the Samba hierarchy "Samba-smb.conf" and the Printer resource "lp-admin" on server "ServerA".	The dependency creation attempt between the Samba resource and the Print Services resource failed. Examine the other messages to determine the cause of the failure.
109038	The Printers(s) "lpadmin" defined in the configuration file are not under LifeKeeper protection. Action: Create Printer instances for the unprotected printers defined in the Samba configuration file.	LifeKeeper protected Print Services resources must exist for all of the printers defined in the configuration file. Those listed are not LifeKeeper protected.
109041	The selected configuration file "/etc/samba/smb.conf" is in use by Samba resource "Samba-smb.conf". *or* The selected netbios name “LKServer” is in use by Samba resource "Samba-smb.conf".	A Samba configuration file or netbios name can only be protected once in the cluster. Rename the configuration file or select a new NetBIOS name .

Hierarchy Extension

Error Number	Error Message	Description
109008	Replication of config file to target server ”ServerB” failed. The ”mkdir” of "/etc/samba/config_files" failed. *or* Replication of config file to target server ”ServerB” failed. The ”lcdrcp” of "/etc/samba/config_files" failed.	The attempted copy of the Samba configuration file on the extend failed. Either the **mkdir** or remote copy failed.
109042	WARNING: The configuration file "/etc/samba/smb.conf" currently exists on server "ServerB" and will be overwritten if this resource is extended.	The configuration file used for the resource hierarchy already exists on the backup server and will be overwritten. Cancel the extension to abort the overwriting of the file.

Restore

Error Number	Error Message	Description
109027	Failed start of smbd as daemon process.	The attempt to start **smbd** as a daemon failed. Check the Samba log files for additional information.
109028	Failed start of nmbd as daemon process.	The attempt to start **nmbd** as a daemon failed. Check the Samba log files for additional information.

Remove

Error Number	Error Message	Description
109033	Failed to stop smbd daemon process. Attempting to stop via SIGKILL.	The normal termination of the **smbd** daemon process failed so the daemon will be forcibly terminated.
109034	Failed to stop nmbd daemon process. Attempting to stop via SIGKILL.	The normal termination of the **nmbd** daemon process failed so the daemon will be forcibly terminated.

Resource Monitoring

Error Number	Error Message	Description
109031	Connection attempt to smbd daemon on address "100.25.104.26" failed.	The health check of the **smbd** daemon process failed when a connection attempt on the listed IP address failed. A local recovery will be attempted. If the local recovery fails, the hierarchy will be switch over to the backup server.
109032	Connection attempt to nmbd daemon failed for broadcast address "100.25.107.255" using netbios name "FILESERV1".	The health check of the **nmbd** daemon process failed when a connection attempt on the listed broadcast address failed. A local recovery will be attempted. If the local recovery fails, the hierarchy will be switched over to the backup server.
109039	No dependent IP resources were found for tag "Samba-smb.conf" on server "ServerA".	No IP children were found for the Samba hierarchy when attempting to ascertain the health of the **nmbd** daemon. Examine the logs to determine the failure.
109040	Unable to determine IP address and/or mask for IP resource "ip-100.25.104.25".	Extracting the IP address and Mask from the info for the IP resource failed. Check that the info field contains the IP address and mask.

Configuration File Synchronization Utility

Error Number	**Error Message**	**Description**
109013	The hierarchy for the specified configuration file does not have an equivalency with the target system. Select another target system or configuration file.	The -t TargetSys argument to the **synccfg** utility specified a system that does not contain an equivalency for the Samba hierarchy on the local system.
109044	Usage: synccfg -t TargetSys -c ConfigFile	The **synccfg** utility requires two arguments,-t TargetSys where the configuration file will be copied and –c ConfigFile for the name of the configuration to copy. You must specify both arguments.
109045	Specified Configuration file does not exist on this server.	The –c ConfigFile argument to the **synccfg** utility specified a configuration that does not exist on this system.
109046	The target system specified for updating is the same as the template system. Select a new target system.	The –t TargetSys argument to the **synccfg** utility specified the local server as the target of the copy. The target server must not be the local system.
109047	The specified configuration file is not used in any Samba instances. Select another configuration file.	The –c ConfigFile argument to the **synccfg** utility specified a configuration file that is not used in any LifeKeeper Samba instance. Only configuration file protected by LikeKeeper can be copied with this utility.
109048	The syncronization of the Samba configuration file failed with a status of "%s".	The attempted synchronization of the configuration files between the local system and the target system failed. The status provides the reason for the failure.
109049	The hierarchy for the specified configuration file is ISP on the target node. Select another target system or configuration file.	The -t TargetSys argument to the **synccfg** utility specified a system where the Samba hierarchy is currently ISP. The target system must not be ISP.