ExpressCluster® X File Server Agent 1.0 *for Linux*

Administrator's Guide

8/31/2007
Fourth Edition

Revision History

Edition	Revised Date	Description
First	2006/09/08	New manual
Second	2006/12/12	EXPRESSCLUSTER logo has been changed.
Third	2007/6/30	Descriptions recommending using monitor resources have been added.
Fourth	2007/08/31	Added information about setup in "Preface".

Disclaimer

Trademark Information

Table of Contents

Preface ..... **v**

Who Should Use This Guide v
Summary of the Guide v
ExpressCluster X Documentation Set vi
Conventions vii
Contacting NEC viii

Chapter 1 Overview of File Server Agent **11**

What is ExpressCluster X File Server Agent? 12
Overview of monitoring with ExpressCluster X File Server Agent 12
Mechanism of monitoring 13
Target monitoring application 14
Using the File Server Agent 15
System requirements for the File Server Agent 15
Before using the File Server Agent 15
License registration for the File Server Agent 16
Registering the license interactively 16
Registering the license by specifying a license file 17

Chapter 2 File Server Agent command reference **19**

List of file server monitoring commands 20
Monitoring chart 20
Writing scripts to exec resource 21
Suspending and resuming monitoring 21
File Server Agent command reference 22
Samba monitoring command 22
NFS monitoring command 26

Chapter 3 How to check monitor status **31**

Verifying monitor information with monitoring commands 32
Verifying alert messages on the WebManager 32
Log collection when an error has occurred 32
Alert messages 33
Messages being output by clp_sambamon 33
Messages indicating normal operation 33
Messages due to setting error 33
Messages being output when an error is detected in file server monitoring 34
Messages due to system error 35
Messages being output by clp_nfsmon 36
Messages indicating normal operation 36
Messages due to setting error 36
Messages being output when error is detected in file server monitoring 37
Messages due to system error 37

Chapter 4 Settings for ExpressCluster X File Server Agent **39**

Setting up ExpressCluster X File Server Agent 40
Step 1 Creating a failover group 41
Step 1-1 Adding a group 41
Step 1-2 Adding a group resource (floating IP address) 41
Step 1-3 Adding a group resource (disk resource) 42
Step 2 Adding the exec resource (exec 1) for starting the target monitoring application 43
Step 3 Confirmation test for target monitoring application startup 45
Saving the cluster configuration data on a floppy disk (Windows) 45
Creating a cluster using the data in a floppy disk 46
Verifying the behavior of the target monitoring application 48
(Verification 1) Starting up a group 48

(Verification 2) Stopping a group 48
(Verification 3) Moving a group 48
(Verification 4) Failing over a group 49
Step 4 Adding the exec resource (exec 2) for starting monitoring command 50
Step 5 Adding the pid monitor resource 53
Failover soon after detecting an error 55
Failover without stopping a server 55
Step 6 Verifying the settings for monitoring command 56
(Verification 1) Starting up a group 56
(Verification 2) Stopping a group 56
(Verification 3) Moving a group 57
(Verification 4) Failing over a group 57

Appendix A. Script templates 59

Setting up script templates 59
Installing the script templates on Windows machine 59
Installing script templates on Linux machine 60
Uninstalling the script templates 60
Uninstalling the script templates for Windows 60
Uninstalling the script templates for Linux 60
Details on script templates 61
For Samba service startup 61
start.sh 61
stop.sh 62
For Samba monitoring 63
start.sh 63
stop.sh 64
For NFS service startup 65
start.sh 65
stop.sh 66
For NFS monitoring 67
start.sh 67
stop.sh 68

Appendix B. Glossary 68

Appendix C. Index 71

Preface

If you newly set up Agents as monitor resource with installation of version 1.1.0-1 or later rpm, please refer to the following manuals.

Getting Started Guide
- "Supported distributions and kernel versions"
- "Applications supported by monitoring options"

Reference Guide
- Chapter 6 Monitor resource details

Please refer to this manual in case that
- you set up Agents with installation of version 1.0.X-X rpm.
- you use a shell scripts to control Agents that created with version 1.0.X-X rpm for another cluster system with version 1.1.0-1 or later rpm.

Who Should Use This Guide

The *ExpressCluster X File Server Agent Administrator's Guide* is intended for system engineers and administrators who want to build, operate, and maintain a cluster system. Instructions for setting up a cluster system with ExpressCluster X File Server Agent is covered in this guide.

Summary of the Guide

Chapter 1 **Overview of File Server Agent**
This chapter provides information on a product overview of the File Server Agent and how to set it up.

Chapter 2 **File Server Agent command reference**
This chapter provides detailed information on the script commands.

Chapter 3 **How to check monitor status**
This chapter describes the monitoring method and the messages.

Chapter 4 **Settings for ExpressCluster X File Server Agent**
This chapter describes how to create cluster configuration data using the File Server Agent and how to create a cluster.

Appendix A **Script templates**
The script template used in the ExpressCluster is introduced.

Appendix B **Glossary**

Appendix C **Index**

ExpressCluster X Documentation Set

The ExpressCluster X manuals consist of the following four guides. The title and purpose of each guide is described below:

Getting Started Guide
This guide is intended for all users. The guide covers topics such as product overview, system requirements, and known problems.

Installation and Configuration Guide
This guide is intended for system engineers and administrators who want to build, operate, and maintain a cluster system. Instructions for designing, installing, and configuring a cluster system with ExpressCluster are covered in this guide.

Reference Guide
This guide is intended for system administrators. The guide covers topics such as how to operate ExpressCluster, function of each module, maintenance-related information, and troubleshooting. The guide is supplement to the *Installation and Configuration Guide*.

Administrator's Guide (Add-on product)
This guide is intended for system administrators. The detailed information on each product package is described in this guide. There are five guides for each optional product and topics such as product overview, instruction for setting up are covered:

Alert Service Administrator's Guide
Application Server Agent Administrator's Guide
Database Agent Administrator's Guide
File Server Agent Administrator's Guide
Internet Server Agent Administrator's Guide

Conventions

In this guide, **Note**, **Important**, **Related Information** are used as follows:

Note:
Used when the information given is important, but not related to the data loss and damage to the system and machine.

Important:
Used when the information given is necessary to avoid the data loss and damage to the system and machine.

Related Information:
Used to describe the location of the information given at the reference destination.

The following conventions are used in this guide.

Convention	Usage	Example
Bold	Indicates graphical objects, such as fields, list boxes, menu selections, buttons, labels, icons, etc.	In **User Name**, type your name. On the **File** menu, click **Open Database**.
Angled bracket within the command line	Indicates that the value specified inside of the angled bracket can be omitted.	`clpstat –s[-h host_name]`
#	Prompt to indicate that a Linux user has logged on as root user.	`# clpcl -s -a`
Monospace (courier)	Indicates path names, commands, system output (message, prompt, etc), directory, file names, functions and parameters.	`/Linux/1.0/eng/server/`
Monospace bold (courier)	Indicates the value that a user actually enters from a command line.	Enter the following: **`# clpcl -s -a`**
Monospace italic (courier)	Indicates that users should replace italicized part with values that they are actually working with.	`rpm –i expressclsbuilder -<version_number>-<release_number>.i686.rpm`

Contacting NEC

For the latest product information, visit our website below:

http://www.ace.comp.nec.co.jp/CLUSTERPRO/clp/global-link.html

Chapter 1 Overview of File Server Agent

This chapter provides an overview of ExpressCluster X File Server Agent and instructions for how to register the license.

This chapter covers:

- What is ExpressCluster X File Server Agent? ····· 12
- Overview of monitoring with ExpressCluster X File Server Agent····· 12
- Using the File Server Agent ····· 15
- License registration for the File Server Agent ····· 16

What is ExpressCluster X File Server Agent?

When ExpressCluster monitors applications, failing over the application when detecting an error at application startup is possible; however, an application stalling and errors cannot be detected once application is started up.

With ExpressCluster X File Server Agent, failover can be performed when an application is stalled or an error has occurred as well as at application startup. Thus you will be able to monitor applications on the cluster system.

ExpressCluster X File Server Agent achieves the monitoring of an application as described above by using monitor resources[1] and monitoring commands.

For details on the function of this product of monitoring applications using monitor resources, see Chapter 6, “Monitor resource details” in the *Reference Guide*.

The remainder of this guide explains the monitoring function using monitoring commands.

Important:

It is recommended to perform monitoring using monitor resources since the WebManager and the – clpstat command enables you to monitor the status and to avoid describing dedicated scripts.

Overview of monitoring with ExpressCluster X File Server Agent

ExpressCluster X File Server Agent configures the following three settings to monitor the File Server:

1. the exec resource for starting the target monitoring application (exec 1)
2. the exec resource for starting the monitoring command (exec 2)
3. the pid monitor resource that monitors exec 2

[1] Supported from ExpressCluster X version 1.1.0-1.

Mechanism of monitoring

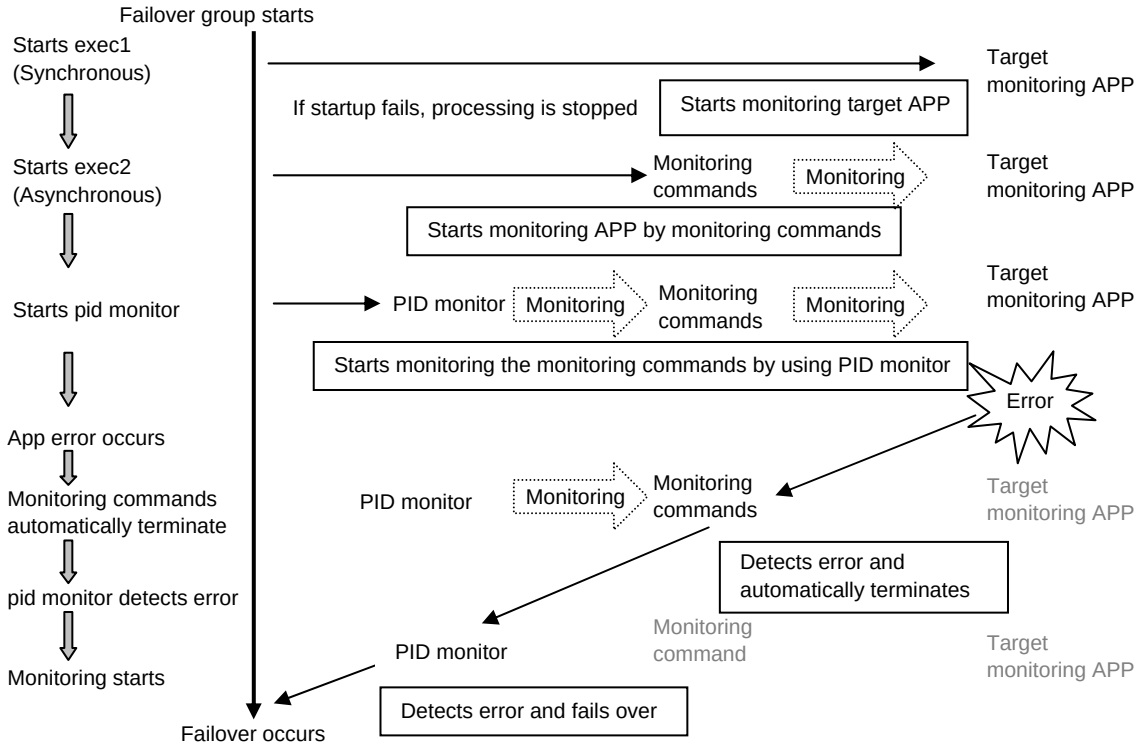
Start the target monitoring application in exec 1.Create an exec resource (exec2) that differs from this exec 1, and in this exec 2 script, write the monitoring command that monitors the application started in exec 1. Two exec resources are created to distinguish whether an error has occurred at an application startup or after it has been started.

Configure the settings for the pid monitor resource to monitor exec 2. The monitoring commands specified in exec 2 are provided by ExpressCluster X File Server Agent to monitor applications. These monitoring commands monitor the file server at the intervals specified by the parameter. If they detect any file server error, they stop their own processes.

If these commands stop when a file server error is detected, configure the settings to execute failover or server shutdown by the pid monitor that monitors them.

Using the mechanism above, you can detect a stalling problem and an error of the target monitoring application that cannot be detected by the ordinary settings and failover and shutdown become possible. The following figure explains this mechanism:

See below for overview of the application (APP) monitoring using the monitoring command:



Important:

- These commands can detect an error which does not cause the target monitoring application to end abnormally (mainly stalling problem). This is indirectly achieved, by monitoring not the process ending of target monitoring application, but the monitoring operations such as attempting to access to the application server.
- The purpose of using these commands is to monitor the operation of the target monitoring application, and not to investigate or diagnose the cause when an error occurs on the application. When an error occurs, you need to use other means, such as application logs to see the details of its cause.
- Depending on the target monitoring application, access logs are output to syslog and other places, and logs are output to the local directory of target monitoring application every time you execute the monitoring processes. Since the settings cannot be controlled by these monitoring commands, configure these settings by the target monitoring application, if necessary. However, note that it may become difficult to investigate the cause because logs are not output when an error occurs, if the target monitoring application is set not to output logs.

Run the stop command (**`clp_`*`XXXXXX XXXXXX`*` --stop`**) to stop these monitoring commands. If a process is stopped by using the kill command of Linux, the monitoring command may not be restarted because management information of the monitoring command will not be initialized.

Related Information:
For details, see Note 5 in "Samba monitoring command" on page 22, and notes for other command in this guide.

Target monitoring application

ExpressCluster X File Server Agent monitors file server working under the ExpressCluster environment. The following chart shows the version of ExpressCluster X File Server Agent and file server application that can be monitored:

File Server	ExpressCluster File Server Agent 1.0-1
Samba	Yes
NFS	Yes

Yes: Supported, No: Not supported

To monitor file servers, monitoring commands for each file server are provided.

Refer to Chapter 2, "File Server Agent command reference" for more information on commands. See ExpressCluster home page, *http://www.ace.comp.nec.co.jp/CLUSTERPRO/global-link.html* , for supported version of Samba and NFS.

Using the File Server Agent

In this section, how to use ExpressCluster X File Server Agent is introduced. The module for the File Server Agent is installed together with the ExpressCluster X. You only need to register the license to use the File Server Agent.

System requirements for the File Server Agent

Check each item on all servers where the File Server Agent will be installed. The system requirements for the File Server Agent are the following:

System requirements for the File Server Agent (monitoring module)	
Hardware	IA 32 server, x86_64 server
OS	Same system requirements for the ExpressCluster Server. File server system to be monitored needs to be able to function.
ExpressCluster	ExpressCluster X 1.0 or later
Memory space	5 MB (per command)

Obtain the latest update of the File Server Agent. Refer to the update instructions for how to update the File Server Agent.

Note:
Since these monitoring commands run as a client application of a file server system, it is necessary to make settings for client application to run on a server. Refer to the manuals for each file server system for details.

Before using the File Server Agent

The following tasks need to be completed before registering the license for the File Server Agent. If the tasks are not completed, refer to the Section II, "Installing and configuring ExpressCluster X" in the *Installation and Configuration Guide* for procedures and complete the tasks.

1. Installation of the ExpressCluster Server and ExpressCluster X Builder
2. Execution of the cluster creation command
3. License registration of the ExpressCluster Server

Upon completing 1 through 3 above, you can start operating the File Server Agent by following the procedures below:

1. License registration of the File Server Agent
   Register the license by following the procedure introduced in "License registration for the File Server Agent" on the next page.

License registration for the File Server Agent

License registration is necessary to use the File Server Agent.

To register the license, log on as root user from the server that constitutes a cluster. You have to register a unique license key for each server.

Note:
When your ExpressCluster version does not support ExpressCluster X File Server Agent, you may not be able to register the license properly. In such a case, you have to update your ExpressCluster.

Registering the license interactively

The explanation below is given using a product version as an example.

Check the following before installing a license:

- Verify that the ExpressCluster Server is installed, cluster creation command is executed, and the license for the ExpressCluster Server is registered.
- Prepare the license sheet for the File Server Agent obtained from your sales agent. You will need to enter the value indicated in the license sheet.

1. Run the following command on a server:

   # clplcnsc –i –p FSAG10

2. The following is displayed. Enter 1 for the product version:

   ```
   Software license
     1   Product version
     2   Trial version
   Select the license version [1 or 2]…1
   ```

3. You will be prompted to enter the product serial number. Specify the number indicated in the license sheet:

   ```
   Enter serial number [Ex. XXX0000000]… xxxnnnnnnn
   ```

4. You will be prompted to enter the product license key. Specify the number indicated in the license sheet.

   Note:
   Enter the license sheet information exactly as indicated because the license key is case-sensitive. To avoid any confusion with other letters and numbers, “I” and “O” in capital letter are not used in the ExpressCluster license key.

   ```
   Enter license key
   [Ex. XXXXXXXX-XXXXXXXX-XXXXXXXX-XXXXXXXX]…
   xxxxxxxx-xxxxxxxx-xxxxxxxx-xxxxxxxx
   ```

   After running the command, the message “Command succeeded” is displayed in the console to indicate that the command is successfully completed. If other completion messages are displayed, refer to “License management command” in Chapter 4, “ExpressCluster command reference” in the *Reference Guide*.

Registering the license by specifying a license file

For a trial version, you may register the license using a license file in stead of a license sheet. License registration for trial version allows you to register the license for the entire cluster by registering to only one server in a cluster.

- Run the following command on a server:

  # clplcnsc –i *filepath* –p FSAG10

Assign the file path to the license file specified by option –i.

After running the command, the message “Command succeeded” is displayed in the console to indicate that the command is successfully completed. If other completion messages are displayed, refer to “License management command” in Chapter 4, “ExpressCluster command reference” in the *Reference Guide*.

This completes the setup of ExpressCluster X File Server Agent. How to use the monitoring commands is introduced in the next chapter.

Chapter 2 File Server Agent command reference

This chapter provides information regarding how to set up and operate ExpressCluster X File Server Agent. File server monitoring commands and File Server Agent commands are also introduced.

This chapter covers:

- List of file server monitoring commands ····· 20
- File Server Agent command reference ····· 22

List of file server monitoring commands

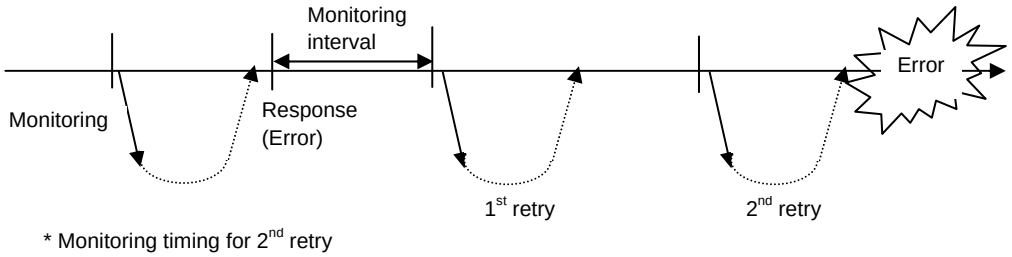
The File Server Agent provides file server monitoring commands to be written in the script.

Command	Used for	Refer to
clp_sambamon	Monitors Samba	Page 22
clp_nfsmon	Monitors NFS	Page 26

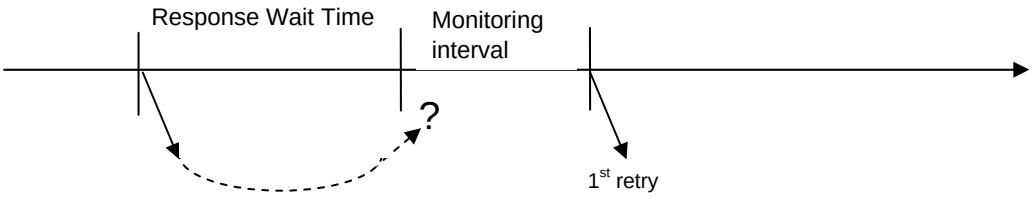
Note:
These commands should be run as root user. If a file server monitoring command is run by a user other than root user, you may not be able to acquire the license information and cannot run the command. When running these commands, /usr/sbin must be added to a path although it is normally added.

Monitoring chart

File Server Agent detects an error in the following timing:



If the Agent does not get a response within the specified wait time, it is considered as an error and will perform a retry.



Writing scripts to exec resource

Note the following when you write a script to an exec resource:

- Before creating an exec resource in which you will write start/end of monitoring commands, complete the exec resource in which start/end of a target monitoring application is written. Then check if a failover group successfully starts, stops, moves, and fails over. If you create an exec resource where the start/end of the monitoring commands are written without checking these issues, it would be difficult to determine the cause of an error is detected by the monitoring commands at failover group startup. For example, if an error has actually occurred, various environments of the target monitoring application were set properly, or a parameter value of the monitoring commands is appropriate, is hard to distinguish.

- Configure the dependency of the resources so that the exec resource for start/end of the monitoring commands would start up after the exec resource in which start/end of a target monitoring application to be written starts up. If you make incorrect settings, the monitoring command may consider that an error has occurred in target monitoring application.

These monitoring commands can also be written to monitor multiple file servers in one failover group.

Example:

Failover Group A

See Appendix A “Script templates” for example of s writing a script.

Suspending and resuming monitoring

The File Server Agent can suspend and resume monitoring. While monitoring is suspended, file server maintenance can be performed. Follow the procedures below:

1. Start up the monitoring command to start monitoring.
2. To suspend monitoring, run the following command from a server console whenever you wish to suspend:
   Monitoring Command watchid `--pause`
3. The following message is displayed on the ExpressCluster WebManager to indicate that monitoring has been suspended:
   `clp_xxxxmon will stop monitoring..` [ID`:`***`watchid`***]
4. To resume monitoring, run the following command from the server console whenever you wish to resume:
   Monitoring Command watchid `--resume`
5. The following message is displayed on the ExpressCluster WebManager to indicate that monitoring is suspended:
   clp_xxxxmon will start monitoring.. [ID***:watchid***]

File Server Agent command reference

This section provides information on command functions. Note the following to understand the command functions:

- **Command line**

  Gives an idea what the user actually enters:

  "[ ]" indicates that the enclosed parameter may be skipped.

  "|" indicates one of the parameters separated by this symbol should be chosen.

- **Description**

  Provides information on function.

- **Option**

  Provides information on parameters used in the command line.

- **Remarks**

  Gives supplementary information such as details on parameter settings.

- **Monitoring method**

  Gives explanation on how monitoring is done.

- **Note**

  Gives information on what should be noted when using the command.

- **Example of command entry**

  Provides example of writing script when actually using the command.

Samba monitoring command

clp_sambamon: the clp_sambamon monitors Samba.

Command line:

Start Monitoring clp_sambamon	*Identifier* -s *Shared_name* [-n *IP_address*] [-u *User_name*] [-p *Password*] [-i *Monitoring_interval*] [-c *Retry_count*] [-r *Response_wait_time*]
Stop Monitoring clp_sambamon	*Identifier* --stop [*Stop_wait_time*]
Interrupt Monitoring clp_sambamon	*Identifier* --pause
Resume Monitoring clp_sambamon	*Identifier* --resume
Display Information clp_sambamon	*Identifier* --disp
Delete Information clp_sambamon	*Identifier* --del

Description	This monitoring command monitors Samba by specifying a shared name. The command stops when any Samba error is detected. The command also stops, interrupts, and resumes monitoring. To specify --pause/--resume/--disp/--del, run from the root privileged console.

Option	*Identifier*	Specifies an identifier to uniquely identify the monitoring command. You must set an identifier.
	-s *Shared_name*	Specifies a shared name of Samba server to be monitored. You must set an identifier.
	-a *IP_address*	Specifies an IP address of Samba server. Default value: 127.0.0.1
	-u *User_name*	Specifies a user name when log on to Samba service. Default value: None
	-p *Password*	Specifies the password when log on to Samba service. Default value: None
	-i *Monitoring_interval*	Specifies Samba monitoring interval in seconds (between 1 and 10000). Default value: 30
	-c *Retry_count*	Specifies how many retries will be made after an error is detected when monitoring Samba (between 1 and 10000). Default value: 5
	-r *Response_wait_time*	Specifies a response wait time (between 1 and 10000) in Samba monitoring in seconds. Default value: 60
	--stop	Stops the monitoring command.
	Stop_wait_time	Specifies time to wait for the monitoring command to stop normally (between 1 and 10000) in seconds. Default value: 60
	--pause	Temporarily interrupts monitoring.
	--resume	Resumes monitoring.
	--disp	Displays a process ID (pid) of the monitoring command identifier managed by the monitoring command. Not for ordinary use. (See Note 5)
	--del	Deletes information about the monitoring command identifier managed by the monitoring command. Not for ordinary use. (See Note 5)

Remarks	About identifier: You need to specify a unique identifier on the system to control the monitoring command. You cannot use an identifier that is already used by a monitoring command for another monitoring command. Specify an identifier using alphanumeric characters. An identifier should be up to 255 bytes and is case sensitive. Specify an identifier as a first argument of the monitoring command. About -s parameter: Specifies a shared name configured in smb.conf file. About -a parameter: When Samba is used with a floating IP address, an IP address can be specified explicitly by using –a parameter. About –u, –p parameters: Make sure to specify these parameters when guest connection is not enabled by Samba settings. About s,-a,-u,-p parameters: A text can be specified up to 255 bytes with these parameters. The actual effective length of text varies by each parameter. No verification for the effective length is performed with this monitoring command. The effective length complies with the specification of Samba.

Monitoring Method	This monitoring command monitors the following: By connecting to Samba server and verify establishment of tree connection to resources of the Samba server. The command determines the following results as an error: (1) No response to the issued Samba service request within a response wait time. (2) A response to Samba service request is invalid. When an error is repeated the number of times set to retry count, it is considered as a Samba error.

Note 1	If the message “clp_sambamon will start monitoring the share resource [xxx]. [ID:***watchid***]” is not displayed after “clp_sambamon started monitoring Samba daemon. [ID:*watchid*” is displayed on ExpressCluster WebManager alert view when starting up a failover group that operates this monitoring command, the parameter value and Samba environment may not match. Check the Samba environment.
Note 2	Specify the smb.conf file for the shared name to be monitored to enable a connection from a local server. Allow guest connection when the security parameter of the smb.conf file is “share.”
Note 3	Samba functions except file sharing and print sharing.
Note 4	If the smbmount command is run on the monitoring server when the Samba authentication mode is “Domain” or “Server,” it may be mounted as a user name specified by -u parameter of the clp_sambamon command.
Note 5	Management information is not initialized properly when the monitoring command process is stopped by a kill command during monitoring and the monitoring command with the same identifier may fail to start. If this happens, run the “clp_sambamon identifier --disp” command and you will see the process ID corresponding to the specified identifier. Use the ps command to check the process ID of the executable file and if the process is other than the monitoring command, remove management information by using the “clp_sambamon identifier --del” command. A monitoring command malfunctions if you run the “clp_sambamon identifier --del” command by specifying a normally operating identifier.

Example of command entry	[start.sh] **`clp_sambamon sambawatch -s Shared_name`** [stop.sh] **`clp_sambamon sambawatch --stop`** This command starts up from an exec resource. For monitoring startup command, start.sh, using only -s parameter should be sufficient unless user authentication is required.

NFS monitoring command

clp_nfsmon: the clp_nfsmon monitors NFS.

Command line:

Start Monitoring clp_nfsmon	*Identifier* -d *Directory* [-a *IP_address*] [-n *Port_number*] [-i *Monitoring_interval*] [-c *Retry_count*] [-r *Response_wait_time*]
Stop Monitoring clp_nfsmon	*Identifier* --stop [*Stop_wait_time*]
Interrupt Monitoring clp_nfsmon	*Identifier* --pause
Resume Monitoring clp_nfsmon	*Identifier* --resume
Display Information clp_nfsmon	*Identifier* --disp
Delete Information clp_nfsmon	*Identifier* --del

Description	NFS is monitored by specifying a directory. This monitoring command stops when an NFS error is detected.

Option	*Identifier*	Specifies an identifier to uniquely identify the monitoring command. You must set an identifier.
	-d *Directory*	Specifies a directory for sharing files. You must set an identifier.
	-a *IP_address*	Specifies an IP address of the server that monitors NFS. Default value: One of the IP addresses set to a server.
	-n *Port_number*	Specifies a port number of the NFS server. Default value: None (automatically achieved from portmapper)
	-i *Monitoring_interval*	Specifies NFS monitoring interval in seconds (between 1 and 10000). Default value: 30
	-c *Retry_count*	Specifies how many retries will be made after an error is detected in NFS monitoring (between 1 and 10000). Default value: 5
	-r *Response_wait_time*	Specifies a response wait time (between 1 and 10000) in NFS monitoring in seconds. Default value: 60
	--stop	Stops the monitoring command
	Stop_wait_time	Specifies time to wait for the monitoring command to stop normally in seconds (between 1 and 10000). Default value: 60
	--pause	Temporarily interrupts monitoring.
	--resume	Resumes monitoring.
	--disp	Displays a process ID (pid) of the monitoring command identifier managed by the monitoring command. Not for ordinary use. (See Note 3)
	--del	Deletes information about the monitoring command identifier managed by the monitoring command. Not for ordinary use. (See Note 3)

Remarks	About identifier: You need to specify a unique identifier on the system to control the monitoring command. You cannot use an identifier that is already used by a monitoring command for another monitoring command. Specify an identifier using alphanumeric characters. An identifier should be up to 255 bytes and is case sensitive. Specify an identifier as a first argument of a monitoring command. About -d parameter: Specify a directory specified in exports file. If 2 bytes character is included, a directory name displayed in ExpressCluster WebManager alert view gets garbled. About -a parameter: When NFS is used with a floating IP address, an IP address can be specified explicitly by using -a parameter.

	About –n parameter: Specify this if you want to use a port number other than 2049 that NFS always uses. About -d, -a, -k parameters: A text can be specified up to 255 bytes with these parameters. The actual effective length of text varies by each parameter. No verification for the effective length is performed with this monitoring command. The effective length complies with the specification of NFS.

Monitoring Method	This monitoring command monitors the following: Connect to the NFS server and run NFS test command. The command determines the following results as an error: (1) No response to the NFS service within a response wait time. (2) Response to the NFS service request is invalid. When an error is repeated the number of times set to retry count, it is considered as NFS error.

Note 1	If the message “clp_nfsmon wick start monitoring the directory 'xxx'. [ID:***watchid***]” is not displayed after “clp_nfsmon started monitoring NFS. [ID:***watchid***]” is displayed on ExpressCluster WebManager alert view when starting up a failover group that operates this monitoring command, the parameter value and NFS environment may not match. Check the NFS environment.
Note 2	Specify the exports file for the shared directory to be monitored to enable the connection from a local server.
Note 3	Management information is not initialized properly when the monitoring command process is stopped by a kill command during monitoring and the monitoring command with the same identifier may fail to start. If this happens, run the “clp_nfsmon identifier --disp” command and you will see the process ID corresponding to the specified identifier. Use the ps command to check the process ID of the executable file and if the process is other than the monitoring command, delete the management information by using the “clp_nfsmon identifier --del” command. A monitoring command malfunctions if you run the “clp_nfsmon identifier --del” command by specifying a normally operating identifier.

Example of command entry	[start.sh] **`clp_nfsmon nfswatch -d directory`** [stop.sh] **`clp_nfsmon nfswatch --stop`** This command starts up from an exec resource. For monitoring startup command, start.sh, using only -d parameter should be sufficient.

There are cases when you need to specify a user name and password for each monitoring command. They need to be written explicitly as parameters of the monitoring command in the startup script of the failover group. However, user name and password should not be explicitly specified for security reasons.

Password maintenance function reports a user name and password to the monitoring command by specifying only a user name as long as a combination of user name and password is saved in the password maintenance file in advance. User name and password are maintained per monitoring command.

You need to create a new password maintenance file to use the password maintenance function.

Create a password maintenance file below as necessary:

Monitoring Command Name	Password Maintenance File
clp_sambamon	/opt/nec/clusterpro/work/clp_sambamon

Grant access permission only to root user for the password maintenance file. Delimit user name and password with a comma (,) in the password maintenance file. When registering multiple users, add one user per line.

```
user1,password1
user2,password2
user3,password3
        :
        :
userN,passwordN
```

Note:
Up to 255 bytes can be assigned for the length of a user name and password.
Do not include unnecessary space and tab code.
Do not specify same user name multiple times.

When using the password maintenance function, it is required to write user name specified by -u parameter and not to write password specified by -p parameter. See below for parameter specification and behavior of the monitoring command:

	With -u Parameter	Without –u parameter
With –p parameter	The value specified by each parameter is valid.	The default user name and password specified by parameter are valid.
Without –p parameter	If the password that corresponds to the user name specified by –u parameter is written in the password maintenance file, the password in the password maintenance file is effective. If it is not written, a user name and default password specified by –u parameter is valid.	The default values for each command are valid.

Chapter 3 How to check monitor status

This chapter provides instructions for how to check the monitor status by using ExpressCluster X File Server Agent. This chapter also includes information about results of commands and messages on the screen or in the log.

This chapter covers:

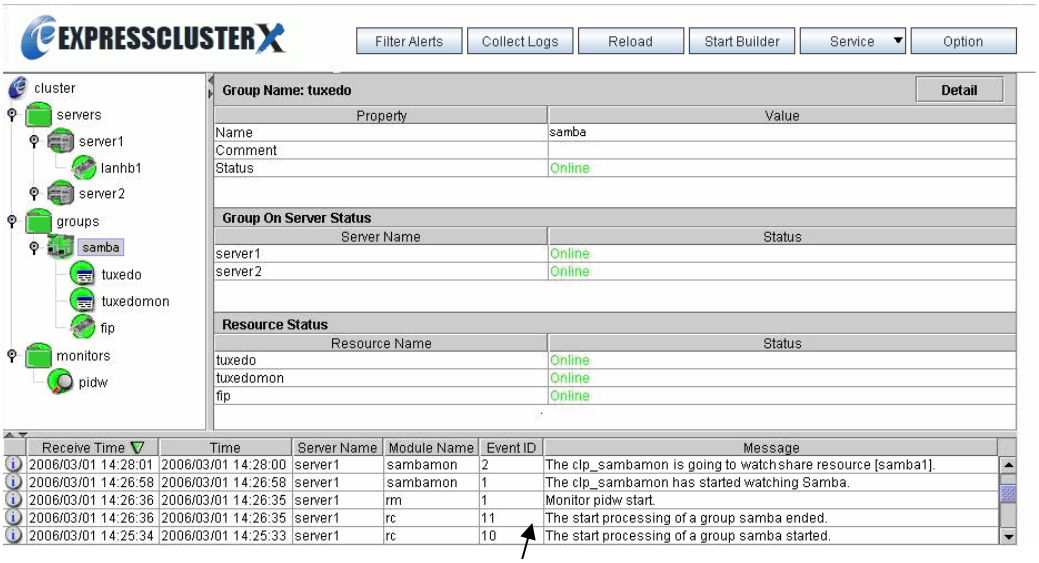
- Verifying monitor information with monitoring commands ····· 32
- Alert messages ····· 33
- Messages being output by clp_sambamon ····· 33
- Messages being output by clp_nfsmon ····· 36

Verifying monitor information with monitoring commands

Verify the monitor status by using the monitoring commands on the ExpressCluster WebManager alert view.

Verifying alert messages on the WebManager

You can view alert messages that ExpressCluster X displays on the WebManager.



Displayed on the ExpressCluster WebManager alert view.

As shown above, messages are displayed on the alert view in the lower part of the WebManager window. If a message is long, it is displayed in multiple lines. In such a case, other message may interrupt a message displayed in multiple lines depending on the timing.

The same message displayed in the alert message is logged to the syslog.

Refer to “Alert messages” on page 33 for details on alert messages.

Log collection when an error has occurred

Error logs of monitoring commands will be produced in the same directory where error logs of the ExpressCluster Server are produced. Logs are collected in the same way as how ExpressCluster logs are collected. For more information, see “Collecting logs by using the WebManager” in Chapter 1, “Functions of the WebManager” and “Collecting logs” in Chapter 4, “ExpressCluster command reference” in the *Reference Guide*.

Alert messages

Messages being output by clp_sambamon

Messages indicating normal operation

#	Message	Explanation	Remarks
1	clp_sambamon started monitoring the Samba daemon. [ID:xx]	clp_sambamon has started up.	-
2	clp_sambamon will start monitoring the share resource [xxx]. [ID:xx]	clp_sambamon has started monitoring the directory xxx.	An error may have occurred if this message is not displayed right after the message above. In such a case, troubleshoot the error according to the error message.
3	clp_sambamon will stop monitoring the Samba [xxx]. [ID:xx]	clp_sambamon stops monitoring.	-
7	clp_sambamon will stop monitoring. [ID:xx]	clp_sambamon suspended monitoring.	Monitoring is suspended by using the --pause command.
8	clp_sambamon will start monitoring. [ID:xx]	clp_sambamon restarted monitoring.	Monitoring is restarted by using the --resume command.
9	clp_sambamon got the password. [ID:xx]	clp_sambamon acquired password info from the password maintenance file.	-
52	clp_sambamon trial version is effective till nn/nn/nn (mm/dd/yyyy).	Running with a trial version license.	-

Messages due to setting error

#	Message	Explanation	Remarks
11	The parameter value of the clp_sambamon is invalid. [ID:xx]	The format for parameter value of clp_sambamon is invalid.	Check the parameter value of the monitoring command.
12	'-s' parameter is not specified at the clp_sambamon command. [ID:xx]	-s parameter is not specified to clp_sambamon.	Check the parameter value of the monitoring command.
13	An identifier is not specified in the clp_sambamon command.	The identifier is not specified in clp_sambamon.	Check the parameter value of the monitoring command.
14	The specified identifier is already used. [ID:xx]	clp_sambamon cannot be restarted because the same identifier is already used.	Check the parameter value of the monitoring command.
15	clp_sambamon could not perform the end processing. [ID:xx]	End processing could not be performed with --stop parameter.	Check the parameter value of the monitoring command.
51	The license of clp_sambamon is not registered.	The license is not registered	Register the license.
53	The trial version license has expired in nn/nn/nn (mm/dd/yyyy).	The trial version license has expired.	-

#	Message	Explanation	Remarks
55	The trial version license is effective from nn/nn/nn (mm/dd/yyyy).	The trial version license cannot be used yet.	-
56	The registration license overlaps.	The license key you have registered overlaps with another key.	Register a different license key for each server.
62	clp_sambamon has detected an error in Samba [xxx] (tree connect error[nn]). [ID:xx]	clp_sambamon could not connect to the Samba shared resource. This error code is for SMB_COM_NEGOTIATE of CIFS. Normally, the error code is 67.	Shared name and user name may not be correct. Check the parameter and Samba environment.
64	clp_sambamon has detected an error in Samba [xxx] (connect error[nn]). [ID:xx]	clp_sambamon could not connect to Samba.	Connecting to Samba from local server may not be allowed. Check the Samba environment.
72	The clp_sambamon has detected network error(connect error[nn]). [ID:xx]	clp_sambamon could not connect to the Samba server. The error code is the error number of the socket system call.	The IP address may be invalid or the Samba service may have stopped. Check the parameter and Samba environment.

Messages being output when an error is detected in file server monitoring

#	Message	Explanation	Remarks
6	clp_sambamon detected an error and will be terminated. [ID:xx]	Terminating because clp_sambamon detected an error.	Troubleshoot the problem by following the message displayed immediately before.
62	clp_sambamon has detected an error in Samba [xxx] (tree connect error[nn]). [ID:xx]	clp_sambamon could not connect to the Samba shared resource. The error code is for SMB_COM_SESSION_SETUP_ANDX of CIFS.	Troubleshoot the error according to the error code.
63	clp_sambamon has detected an error in Samba [xxx] (share type error). [ID:xx]	The type of shared resource of the Samba is “error.”	Check the shared name specified by parameter and restart Samba.
64	clp_sambamon has detected an error in Samba [xxx] (connect error[nn]). [ID:xx]	clp_sambamon could not connect to Samba.	Troubleshoot the error according to the error code.
65	clp_sambamon has detected an error in Samba [xxx] (negotiate error[nn]). [ID:xx]	clp_sambamon failed to negotiate SMB protocol. The error code is for SMB_COM_NEGOTIATE of CIFS.	Troubleshoot the error according to the error code.
72	clp_sambamon has detected network error(connect error[nn]). [ID:xx]	An error was detected during the connection process to the Samba server. The error code is the error number of the socket system call.	Troubleshoot the error according to the error code.

#	Message	Explanation	Remarks
73	clp_sambamon has detected network error(send error[nn]). [ID:xx]	An error was detected during the data sending process to the Samba server. The error code is the error number of the socket system call.	Troubleshoot the error according to the error code.
74	clp_sambamon has detected network error(select error[nn]). [ID:xx]	An error was detected during the response wait process from the Samba server. The error code is the error number of the socket system call.	Troubleshoot the error according to the error code.
75	clp_sambamon has detected network error(recv error[nn]). [ID:xx]	An error was detected during the data receiving process from the Samba server. The error code is the error number of the socket system call.	Troubleshoot the error according to the error code.

Messages due to system error

#	Message	Explanation	Remarks
42	clp_sambamon has detected system error (xxx nn). [ID:xx]	A Linux system error has occurred. xxx indicates a function name and nn indicates error code.	Check the system status based on error codes
54	Failed to check the license information of clp_sambamon.	Failed to verify the license information.	The license management module of ExpressCluster may be old. Check the module update.
61	clp_sambamon could not get the server name(socket error[nn]). [ID:xx]	Failed to acquire the server name. The error code is the error number of the socket system call.	Troubleshoot the error according to the error code.
71	clp_sambamon has detected network error(socket error[nn]). [ID:xx]	Failed to create socket. The error code is the error number of the socket system call.	Troubleshoot the error according to the error code.

Messages being output by clp_nfsmon

Messages indicating normal operation

#	Message	Explanation	Remarks
1	clp_nfsmon started monitoring the NFS daemon. [ID:xx]	clp_nfsmon has started.	-
2	clp_nfsmon will start monitoring the directory 'xxx'. [ID:xx]	clp_nfsmon has started monitoring the directory xxx.	An error may have occurred if this message is not displayed right after the message above. In such a case, an error message may be displayed after a while. Then troubleshoot the error according to the error message.
3	clp_nfsmon will stop monitoring the NFS shared directory 'xxx'. [ID:xx]	clp_nfsmon stops monitoring.	-
7	clp_nfsmon will stop monitoring. [ID:xx]	clp_nfsmon suspended monitoring.	Monitoring is suspended by using the --pause command.
8	clp_nfsmon will start monitoring. [ID:xx]	clp_nfsmon restarted monitoring.	Monitoring is restarted by using the --resume command.
52	clp_nfsmon trial version is effective till nn/nn/nn (mm/dd/yyyy).	Running with a trial version license.	-

Messages due to setting error

#	Message	Explanation	Remarks
11	The parameter value of clp_nfsmon is invalid. [ID:xx]	The parameter value of clp_nfsmon is invalid in its format.	Check the parameter value of the monitoring command.
12	'-d' parameter is not specified at the clp_nfsmon command. [ID:xx]	-d parameter is not specified to clp_nfsmon.	Check the parameter value of the monitoring command.
13	The identifier is not specified in the clp_nfsmon command.	The identifier is not specified to clp_nfsmon.	Check the parameter value of the monitoring command.
14	The specified identifier is already used. [ID:xx]	clp_nfsmon cannot be restarted because the same identifier is already used. End processing could not be performed with --stop parameter.	Check the parameter value of the monitoring command
15	clp_nfsmon could not perform the end processing. [ID:xx]	End processing could not be performed with --stop parameter.	Check the parameter value of the monitoring command.
51	The license of clp_nfsmon is not registered.	The license is not registered	Register the license.
53	The trial version license has expired on nn/nn/nn (mm/dd/yyyy).	The trial version license has expired.	-
55	The trial version license is effective from nn/nn/nn (mm/dd/yyyy).	The trial version license cannot be used yet.	-
56	The registration license overlaps.	The license key you have registered overlaps with another key.	Register a different license key for each server.

#	Message	Explanation	Remarks
0	clp_nfsmon: mount error(xxxx)	clp_nfsmon could not mount the shared directory of NFS.	Shared directory name may not be correct. Check the parameter and NFS environment.
0	clp_nfsmon: RPC: Port mapper failure – xxxx	clp_nfsmon could not connect to NFS.	IP address may not be correct or the portmap service may have stopped. Check the parameter and NFS environment.
0	clp_nfsmon: RPC: Program not registered.	clp_nfsmon could not connect to NFS.	The NFS service may have stopped. Check NFS environment.
0	clp_nfsmon: RPC: Remote system error - Connection refused.	The portmap service could not function properly.	The port number may not be correct. Check the parameter and NFS environment.

Messages being output when error is detected in file server monitoring

#	Message	Explanation	Remarks
6	clp_nfsmon detected an error and will be terminated. [ID:xx]	Terminating because clp_nfsmon detected an error.	Troubleshoot the problem by following the message displayed immediately before.
66	clp_nfsmon has detected an error in NFS. [ID:xx]	Terminating because clp_nfsmon detected an error.	Troubleshoot the problem by following the message displayed immediately before.
0	clp_nfsmon: xxxxxxxx	Message being output by NFS.	Troubleshoot the problem by following the message.

Messages due to system error

#	Message	Explanation	Remarks
42	clp_nfsmon has detected system error (xxx nn). [ID:xx]	A Linux system error has occurred. xxx indicates a function name and nn indicates error code.	Check the system status based on error codes
54	Failed to check the license information of clp_nfsmon.	Failed to verify the license information.	The license management module of ExpressCluster may be old. Check the module update.
61	clp_nfsmon could not get the server name(socket error[nn]). [ID:xx]	Failed to acquire the sever name. The error code is the error number of the socket system call.	Troubleshoot the problem based on error codes.

Chapter 4 Settings for ExpressCluster X File Server Agent

This chapter provides an explanation on how to set ExpressCluster X File Server Agent.

This chapter covers:

- Setting up ExpressCluster X File Server Agent ····· 40
- Step 1 Creating a failover group ····· 41
- Step 2 Adding the exec resource (exec 1) for starting the target monitoring application ····· 43
- Step 3 Confirmation test for target monitoring application startup ····· 45
- Step 4 Adding the exec resource (exec 2) for starting monitoring command ····· 50
- Step 5 Adding the pid monitor resource ····· 53
- Step 6 Verifying the settings for monitoring command ····· 56

Setting up ExpressCluster X File Server Agent

To monitor the applications by using ExpressCluster X File Server Agent, follow the steps below.

Step 1 Creating a failover group (for the target monitoring application)

Create a failover group for monitoring the target monitoring application and performing failover when an error occurs. Then add a group resource excluding exec resource, such as disk resource and IP resource.

*If you have already created a failover group before setting up ExpressCluster X File Server Agent and it is used for monitoring, use that group. In this case, you do not need to perform Step 1.

Step 2 Adding exec 1 (for starting the target monitoring application)

Add the exec resource for starting the target monitoring application to the failover group you have created in Step 1. In this guide, this exec resource is called “exec 1.”

Step 3 Confirmation test

If you have finished Step 2, reflect the settings information to the server, and check that the target monitoring application is started normally by exec 1.

Step 4 Adding exec 2 (for starting the monitoring command)

Add the exec resource for starting the monitoring command to the failover group. In this guide, this exec resource is called “exec 2.”

Step 5 Adding a pid monitor resource

Add a pid monitor resource for monitoring the exec 2 that has been added in Step 4.

Step 6 Verifying the settings for monitoring command and pid monitor resource

Reflect the settings information of the exec 2 and the pid monitor resource that you have added in Steps 4 and 5 to the server, and check that they can operate normally.

Step 1 Creating a failover group

Create a failover group for the target monitoring application. Add the group resource excluding exec resource, such as this disk resource or IP resource.

Note:
If you have already created a failover group before setting up ExpressCluster X File Server Agent, you can use this group. In this case, you do not have to perform Steps 1-1 to 1-3.

Step 1-1 Adding a group

Add a failover group to an existing cluster system. The explanation is given with an example of creating a failover group that includes FIP resource and disk resource to add to a cluster system that consists of two servers (server1, 2). Replace the value when you actually operate.

Note:
When creating a new cluster, refer to Chapter 3, "Creating the cluster configuration data using the Builder" in the *Installation and Configuration Guide*.

1. Start up the Builder.
   (Default path: C:\Program Files\CLUSTERPRO\clpbuilder-l\clptrek.html)
2. Open the cluster configuration data where a group will be added.
3. Click **Groups** in the tree view, and select **Add** on the **Edit** menu.
4. **Group Definition** dialog box is displayed. Enter the group name **failover1** in the **Name** box, and click **Next**.
5. Click **server1** in the **Available Servers** and click **Add**. **server1** is added to the **Servers that can run the Group**. Likewise, add server2.
6. Check that **server1** and **server2** are set in this order to the **Servers that can run the Group,** and then click **Finish**.

Proceed to the following Step 1-2.

Step 1-2 Adding a group resource (floating IP address)

Add a floating IP address to the group you created in Step 1-1.

1. Click **failover1** in the tree view, and click **Add** on the **Edit** menu.
2. The **Resource Definition** dialog box is displayed. Select the group resource type "**floating ip resource**" in the **Type** box, and enter the group name (floating ip address) in the **Name** box. Click **Next**.
3. Enter the IP address (ex. **10.0.0.12**) in the **IP Address** box. Click **Next**.
4. **Recovery Operation at Activation Failure Detection** and **Recovery Operation at Deactivation Failure Detection** are shown. Ignore them and click **Next**.
5. Click **Finish** in the screen shown.

Proceed to the following Step 1-3.

Step 1-3 Adding a group resource (disk resource)

Add a disk resource to the group you added a FIP resource in Step 1-2.

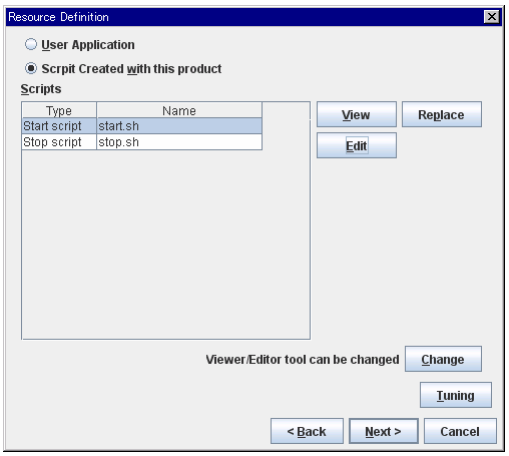
1. Click **failover1** in the tree view, and click **Add** on the **Edit** menu.
2. Enter the 2nd group resource information in the **Resource Definition** dialog box. Select the group resource type (**disk resource**) in the **Type** box, and enter the group name **disk1** in the **Name** box. Click **Next**.
3. Enter the device name (ex. **/dev/sdb2**), mount point (ex. **/mnt/sdb2**) to their corresponding boxes. Select the file system (ex. **ext3**) on the **File System** box and the disk type **Disk** from the **Disk Type** box. Click **Next**.
4. **Recovery Operation at Activation Failure Detection** and **Recovery Operation at Deactivation Failure Detection** are shown. Ignore them and click **Next**.
5. Click **Finish** in the screen shown.

Proceed to the following Step 2.

Step 2 Adding the exec resource (exec 1) for starting the target monitoring application

Add the exec resource for starting the target monitoring application to the failover group that you have created in Step 1. In this guide, this exec resource is called exec 1. You can use the script templates for starting the target monitoring application. Here, the settings for starting the SAMBA daemon are given as an example.

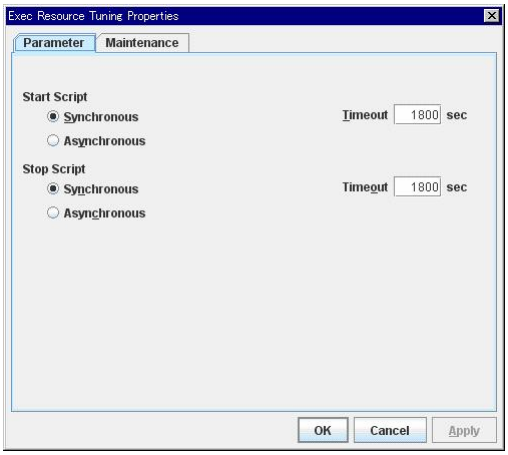
1. If the Builder has not been started yet, start it.
   (Default path: C:\Program Files\CLUSTERPRO\clpbuilder-l\clptrek.html)
2. Right-click the target failover group, and click **Add Resource**.
3. The **Resource Definition** dialog box is displayed. Select **execute resource** in **Type**. Enter any name you like (enter **SAMBA** in this example), and click **Next**.
4. The following window is displayed. Confirm that **Script Created with this product** is selected. Select **Start script** in **Type** in the **Scripts** table, and click **Replace**.



5. Select the script to start the application. Move to the folder of the script templates, and specify the script, “start.sh” in the “`samba`” folder. Click **Open**.

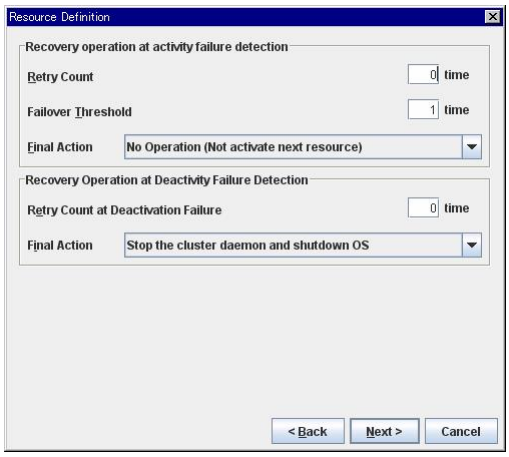
Note:
The following is the default installation folder of the script templates.
C:\Program Files\CLUSTERPRO\clpbuilder-l\scripts\linux

6. The replacement confirmation dialog box is displayed. Specify the original file to be replaced and the path, and select **Yes**.
7. In the same way, replace the end script. Select **Stop Script** in **Type** from the **Scripts** table and click Replace. Specify the script, “stop.sh” in the “`Samba`” folder, and click **Open**.
8. Modify the script. If you click **Edit**, the editor will open. You can modify the script according to the environment. See Appendix A “Script templates”, for what to be modified.
9. Click **Tuning**, and check that **Synchronous** is set for **Start Script** and **Stop Script** on the **Parameter** tab. (**Synchronous** is set by default.)

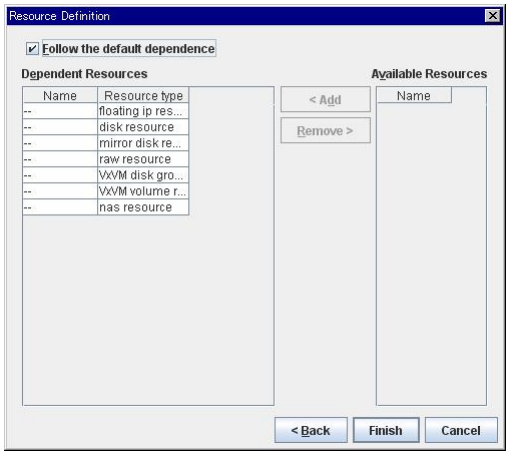


Note:
If you select **Synchronous**, the script is executed with the return value. Here, you need to set **Synchronous** to **Start Scrip** and **Stop Script** because it is required to confirm that the application has been started successfully.

10. In the following screen, you do not need to change the settings. If necessary, configure the settings according to your environment. Click **Next**.



11. In the following screen, you do not need to change the settings. Make sure that the disk resource and IP resource exist on **Dependent Resources**, and click **Finish**.



12. Exec resource for the target monitoring application (exec 1) has been created. Proceed to the following Step 3.

Step 3 Confirmation test for target monitoring application startup

After completing the Steps 1 and 2, convert the configuration information to the server, and check that the monitored application is started normally by exec 1 after that.

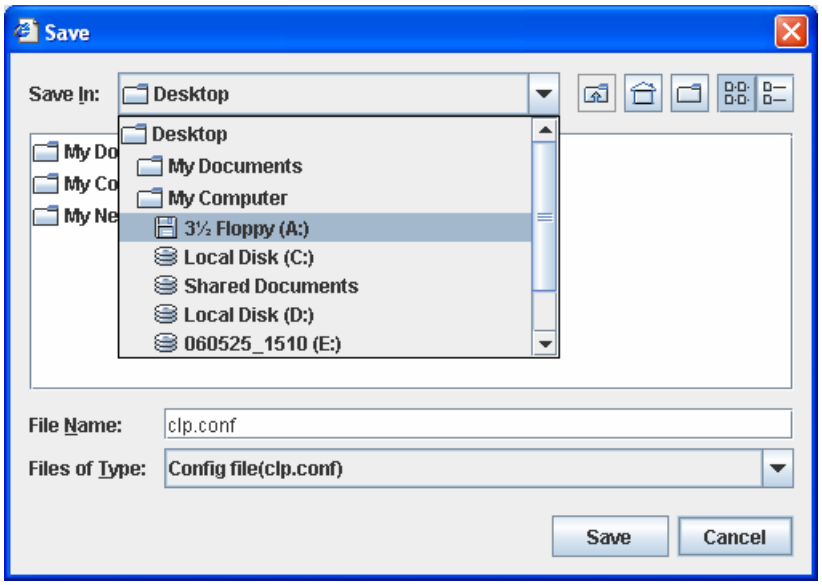
The following describes how to modify the settings to the server using an example of editing the cluster configuration data with the Builder installed on the Windows machine and saving it to the floppy disk.

Related Information:
For using the Builder installed on the Linux machine or saving the cluster configuration data on the file system, refer to "Saving the cluster configuration data" in Chapter 3 "Creating the cluster configuration data using the Builder" in the *Installation and Configuration Guide*.

Saving the cluster configuration data on a floppy disk (Windows)

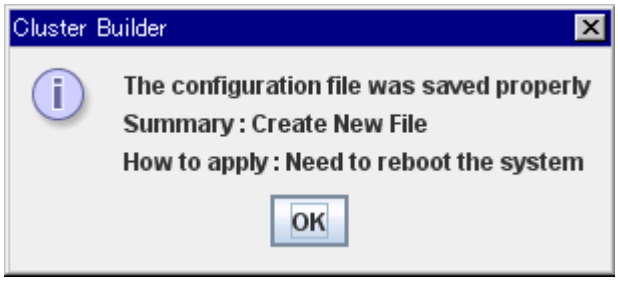
To save the cluster configuration data created with the Builder on Windows machine, follow the procedures below:

1. Insert the floppy disk into the floppy disk drive. Click **Save the configuration file** on the **File** menu. You can directly save it in the floppy disk without creating a directory in the floppy disk.
2. The following dialog box is displayed. Select the floppy disk drive in the **Save** box and click **Save**.



Note:
- Two files (clp.conf and clp.conf.rep) and the directory (scripts) are saved. If any of these are missing, the command to create a cluster does not run successfully. Make sure to treat these three as a set. If you have changed a newly created configuration data, clp.conf.bak is also created as well as those three files.

When the cluster configuration data is saved, the following message is displayed:



3. View what is contained in the floppy disk to check the two files (clp.conf and clp.conf.rep) and the directory (scripts) are located in the floppy disk. If they are in a directory in the floppy disk, you need to specify the directory when you run the command to create a cluster.

Creating a cluster using the data in a floppy disk

To create a cluster using the cluster configuration data saved on a floppy disk, follow the procedures below. If the data is saved in a file system, refer to Chapter 4, "Installing ExpressCluster" in the *Installation and Configuration Guide*.

Note:
- You have to restart all servers after installing the server RPM.
- The server used to create a cluster is the one specified as a master server at the time of creating the cluster configuration data.

1. Insert the floppy disk that contains the cluster configuration data created with the Builder into the floppy disk drive of the server specified as a master server.

Note:
In the clpcfctrl command, /dev/fd0 is used as the floppy disk device, and /mnt/floppy as the mount point by default. If your environment is different from these default values, specify them using the option. For details on option, refer to the *Reference Guide*.

2. Distribute the configuration data saved in the floppy disk to the servers.
   - To use the floppy disk that contains the data saved on Linux, run the following command:
     `clpcfctrl --push -l`
   - To use the floppy disk (1.44-MB formatted) that contains the data saved on Windows, or to use the Windows-formatted floppy disk that contains the data on Linux, run the following command:
     `clpcfctrl --push -w`

Note:
If the two files (clp.conf and clp.conf.rep) and one directory (scripts) that are requisite to create a cluster are saved directly in the floppy disk, you do not need to specify any directory.

The message below is displayed:

```
Need to shutdown system and reboot
please shutdown system after push. (hit return) :
```

3. Press **ENTER**.
   When the cluster is successfully created, the message below is displayed:

```
success.(code:0)
```

Related Information:
For troubleshooting of the clpcfctrl command, refer to the *Reference Guide*.

Verifying the behavior of the target monitoring application

Execute the operations below from the WebManager or the command line to verify that the target monitoring application is running normally.

Note:
Be sure to execute Steps 1 to 4 below before you move to the next step (adding exec 2). If you execute these steps after adding exec 2, it becomes difficult to distinguish whether an error is caused when the application is being started or after it has been started.

Execute the following steps to confirm the monitoring target operation with the WebManager. Start the WebManager (http://*FIP_address_for_the WebManager_group*: *port_number* (default value is 29003)), and execute the following four verification steps.

(Verification 1) Starting up a group

1. Right-click the icon of the failover group that you want to start in the WebManager tree view, and click **Start**.
2. Check that the group icon has turned green. Also check that the application is running normally.

(Verification 2) Stopping a group

1. Right-click the icon of the failover group that you want to start in the WebManager tree view, and click **Stop**.
2. Check that the group icon has turned gray. Also, check that the application has stopped.

Note:
Check that the group has been started and/or stopped among all the servers that start the failover group.

(Verification 3) Moving a group

Move a failover group among servers. Check that the started group properly moves by moving the group to other servers in order.

1. Right-click the icon of the failover group that you want to move in the WebManager tree view.
2. Click the group icon, and check that the group has been started on the destination server in the table view. Also, check that the application is running normally.

Note:
Depending on the application startup/stop process described in the script, it may take a few minutes to complete moving a group. See the tree view on the main screen to check that the group has been moved successfully.

(Verification 4) Failing over a group

Shut down the server on which the failover group is running, and check that the group fails over on the destination server.

1. In the WebManager tree view, right-click the server on which the group you want to fail over is running, and click **Reboot OS**.
2. Restart the server, and check that the failover group fails over to the destination server. Also, check that the application is running normally on the destination server.

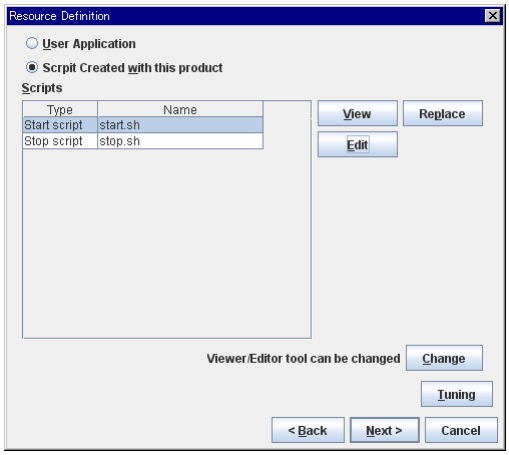
You have completed verifying that the application is running normally by exec 1. Proceed to the following Step 4.

Note:
To verify the operation of the application from the command line, run the following commands.
(Verification 1) clpgrp -s group_name
(Verification 2) clpgrp -t group_name
(Verification 3) clpgrp -m group_name
Run the clpstat command to confirm the cluster status.

Step 4 Adding the exec resource (exec 2) for starting monitoring command

Add the exec resource for starting the monitoring command to the failover group. In this guide, this exec resource is called exec 2.

1. If the Builder has not been started, start it.
   (Default path: C:\Program Files\CLUSTERPRO\clpbuilder-l\clptrek.html)
2. Right-click the target failover group, and click **Add Resource**.
3. The **Resource Definition** dialog box is displayed. Select **execute resource** in **Type**. Enter any name you like (enter "SAMBAMON" in this example), and click **Next**.
4. The following window is displayed. Confirm that **Script Created with this product** is selected. Select **Start script** in **Type** in the **Scripts** table, and click **Replace**.

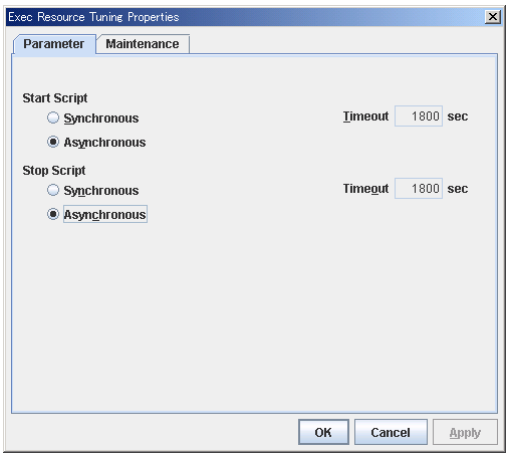


5. Specify the script to start the monitoring command. Move to the folder of the script templates, and specify the script "start.sh" in the "`samba-mon`" folder. Then click **Open**.

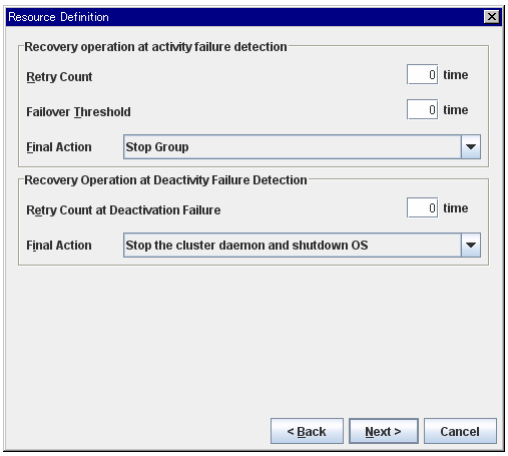
Note:
- The following is the default installation folder of script template:
  C:\Program Files\CLUSTERPRO\clpbuilder-l\scripts\linux
- The target monitoring application name is given to the folder of the script templates for starting up the target monitoring application. "-mon" is added after the target monitoring application name in the folder of the script templates for starting up the target monitoring application.

6. The confirmation dialog box asking for replacement is displayed. Specify the original file to be replaced and the path, and select **Yes**.
7. In the same way, replace the end script. Select **Stop Script** in **Type** and click **Replace**. Specify the script "stop.sh" in the "`samba-mon`" folder, and click **Open**.
8. Modify the script. Open the editor by clicking **Edit** to modify the script according to your environment. See Appendix A "Script templates" for what to be modified.
9. Click **Tuning**, and check that **Asynchronous** is selected for **Start Script** and **Synchronous** for **Stop Script** on the **Parameter** tab.

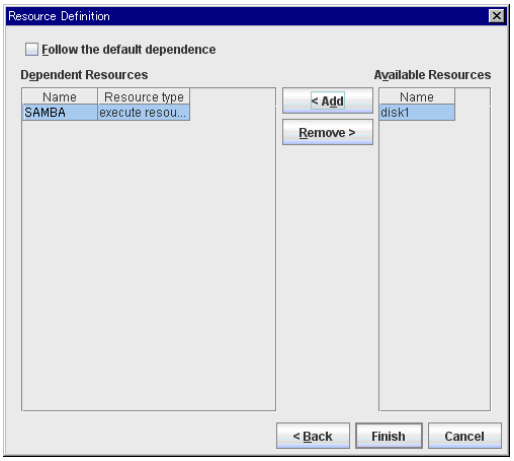
Important:
If you select **Asynchronous**, the script is not started synchronously by the return value, and it is run by itself. The monitoring command enters the monitoring mode and will not take the return value. Be sure to set **Start Script** of the exec resource for running monitoring commands to **Asynchronous**.



10. The following window is displayed. Set **0** to **Failover Threshold** of **Recovery Operation at Activation Failure Detection**, and set **Stop Group** to **Final Action**. Click **Next**.



11. In the following window, uncheck **Follow the default dependence** and set the exec resource (exec 1) created in Step 2 to **Dependent Resources**.

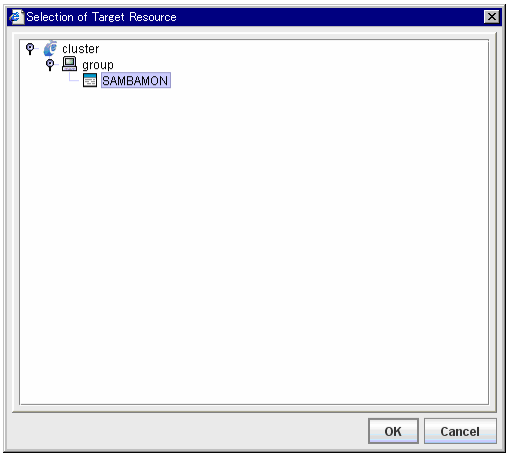


Click the **Finish** button. You have completed adding the exec resource (exec 2) for starting the monitoring command. Proceed to the following Step 5.

Step 5 Adding the pid monitor resource

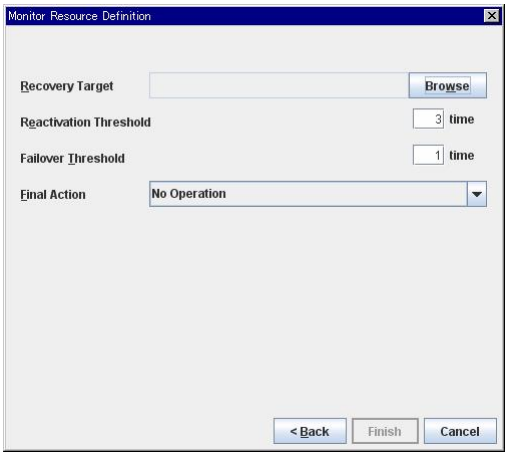
Add the pid monitor resource for monitoring the exec 2 that has been added in Step 4.

1. If the Builder has not been started yet, start it.
   (default path: C:\Program Files\CLUSTERPRO\clpbuilder-l\clptrek.html)
2. Right-click **Monitors** on the tree view, and click **Add Monitor Resource**.
3. The **Monitor Resource Definition** dialog box is displayed. Select the monitor resource type (pid monitor) and enter a monitor resource name in the **Name** box. Click **Next**.
4. Click **Browse** in the window displayed. The **Selection of Target Resource** window is displayed.

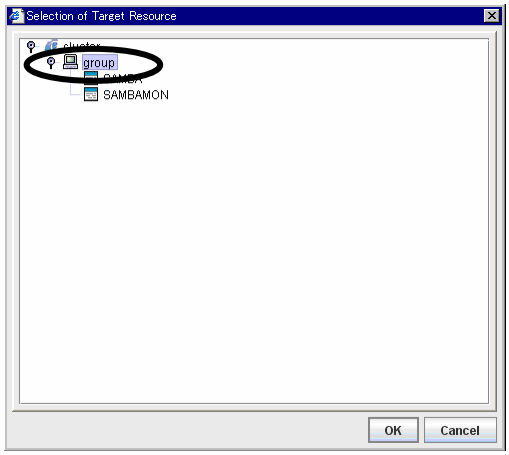


Note:
In the **Selection of Target Resource** window, only the exec resource set to **Asynchronous** in **Exec Resource Tuning Properties** is displayed as selectable target. If nothing can be selected, open the setting window for target exec resource and set the exec resource to **Asynchronous**.

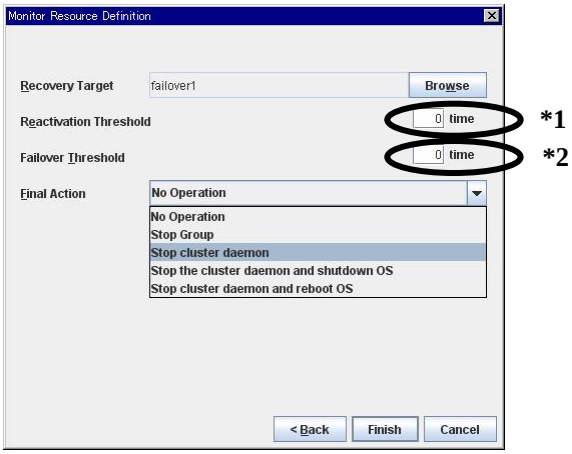
5. Specify the exec resource (exec 2) that starts the monitoring command and click **OK.**
6. Click **Next** to move to the next window and then click **Browse** of **Recovery Target**.



7. In the **Selection of Recovery Target** dialog box, select the failover group that the target monitoring application exists.



8. Select an action to be taken for **Final Action**. This determines the operation of the ExpressCluster X when the monitoring command detects an error in the monitored application.



9. When you finish configuring the settings, click **Finish**. You have finished creating the monitor resource. Move to Step 6.

Failover soon after detecting an error

If you want to execute failover when an error is detected, set **0** to **Reactivation Threshold**. If you want to reactivate the group when an error is detected, specify a value other than **0** to **Reactivation Threshold**.

When you set a value other than 0 to **Reactivation Threshold**, you need to set a value longer than the time below to **Wait Time to Start Monitoring**.

From when a monitoring command is started to when it is terminated due to an error:

The time to terminate the monitoring command is determined by the values of -i parameter (monitoring interval), -c parameter (retry count) and -r parameter (response wait time).

Example: When the monitoring command is terminated because an error is detected in the file server:

- -i (monitoring interval)
- : 60 (sec.)
- -c (retry count)
- : 2 (times)
- -r (response wait time)
- : 120 (sec.)

Approximate time to finish the monitoring command:

60(sec.) × 2(times) + monitoring period = 120 + α (sec.)

* Monitoring interval × retry count + monitoring period

Depending on the type of error, the time to end the monitoring command slightly differs. For the monitoring command operation depending on the error type, see “Alert messages” on page 33.

Failover without stopping a server

If you want to perform failover without stopping a server as a cluster when an error is detected, set **1** to **Failover Threshold**. In this case, other failover groups are operated continuously on the server that detected the error. When an error is also detected in the failover destination server, a failover occurs again and the failover group returns. If you want to perform failover by stopping a server as a cluster when an error is detected, set **0** to **Failover Threshold**.

In addition, specify either **Stop cluster daemon**, **Stop the cluster daemon and shut down OS,** or **Stop cluster daemon and reboot OS** to **Final Action**.

In this case, other failover group is terminated or a failover occurs because the server that detected an error will not be operated as a cluster.

Considering that the recovery will be done when an error occurs, it is recommended to specify as follows:

Failover Threshold 0

Final Action Stop cluster daemon

Step 6 Verifying the settings for monitoring command

In Step 2, you have added the exec resource for starting target monitoring application to the failover group created in Step 1. In Step 3, you have checked that the target monitoring application runs normally.

Then in Step 4, you have added the exec resource for starting monitoring commands. Finally, in Step 5, you have added the monitor resource for monitoring the exec resource for starting monitoring commands.

In Step 6, after updating a failover group, check that monitoring commands run normally by performing the following steps in the same way as you checked the operation of the target monitoring application.

Note:
You need to reflect the cluster configuration data edited in Steps 4 and 5 to the cluster system. For how to reflect the data, see "Creating a cluster using the data in a floppy disk" in Step 3.

Execute the following operation from the WebManager, and check that the target monitoring application runs normally.

Start the WebManager (http://*IP_address_for_the WebManager_group*: *port_number* (default value: 29003)), and execute the following four verification steps.

(Verification 1) Starting up a group

1. Right-click the icon of the failover group that you want to start in the WebManager tree view, and click **Start**.
2. Check that the group icon has turned green.
3. Check that the messages of the Event ID1 and ID2 of the monitoring command are displayed in the WebManager alert view.

Note:
If a monitoring command displays an error message in the ExpressCluster WebManager when the group is started, a parameter value of the monitoring command may not be appropriate. If an error message is displayed on a particular server, there may be an error in the environment settings of the target monitoring application.

(Verification 2) Stopping a group

1. In the WebManager tree view, right-click the icon of the failover group you want to stop, and click **Stop**.
2. Check that the group icon has turned gray.
3. Check that the message of the Event ID 3 of the monitoring command is displayed in the WebManager alert view.

Note:
Check that the group has been started and/or stopped among all the servers that start the failover group.

(Verification 3) Moving a group

Move a failover group among servers. Check that the started group properly moves by moving the group to other servers in order.

1. In the WebManager tree view, right-click the icon of the failover group that you want to move, and click **Move**.
2. Click the group icon, and check that the group has been started on the destination server in the table view.
3. Check that the monitoring command message is displayed in the WebManager alert view.

   Server name: *Name_of_the_server_from_which_the_group_is_moved*

   Event ID: 3

   Server name: *Name_of_the_server_to_which_the_group_is_moved*

   Event ID: 1, 2

Note:
Depending on the application startup/stop process described in the script, it may take a few minutes to complete moving a group. See the tree view on the main screen to check that the group has been moved.

(Verification 4) Failing over a group

Shut down the server on which the failover group is running, and verify that the group fails over to the destination server.

1. In the WebManager tree view, right-click the server in which the group that you want to fail over is running, and click **Reboot OS**.
2. Check that the server is restarted and the failover group fails over to the destination server.
3. Verify that the message of the monitoring command is displayed in the WebManager alert view.

   Server name: *Name_of_the_server_from_which_the_failover_group_fails_over*

   Event ID: 3

   Server Name: *Name_of_the_server_to_which_the_failover_group_fails_over*

   Event ID: 1, 2

Note:
If you set an extremely small value such as 1 (one) to the monitoring interval and response time parameters, you may not be able to perform monitoring normally. Check the operation carefully beforehand.

Related Information:
For more information on messages, see “Alert messages” on page 33.

You have successfully completed the settings for ExpressCluster X File Server Agent.

Note:
Run the following commands to confirm the group failover from the command line.
(Verification 1) clpgrp -s *group_name*
(Verification 2) clpgrp -t *group_name*
(Verification 3) clpgrp -m *group_name*
Run the clpstat command to check the cluster status.

Appendix A. Script templates

Setting up script templates

ExpressCluster provides various script templates to monitor applications. Write the necessary information into templates to monitor applications effectively.

The script templates can be used from the ExpressCluster X Builder. There are Windows version and Linux version for the Builder and the templates are prepared for each version. Choose and set up the script templates according to your environment.

Obtain the latest version of the script templates and follow the instruction that comes with the product to update.

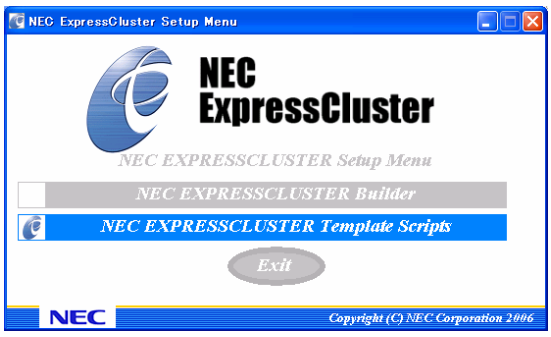
Installing the script templates on Windows machine

Log on as administrator when installing the script templates on Windows machine.

Follow the procedures below:

1. Insert the ExpressCluster X CD into a CD drive.
   The Setup Menu **window is displayed automatically. If the window is not displayed, execute the menu.exe on the CD drive directly.**
2. Click **ExpressCluster for Linux**.

3. Click **ExpressCluster Template Scripts**.



4. A dialog box to verify where to install the template scripts is displayed. Check the path and click **Extract**.

Installing script templates on Linux machine

Set up the script templates after the Builder for the Linux version has been set up. The script templates are installed in a folder having the scripts for the Builder. Refer to Chapter 3, "Creating the cluster configuration data using the Builder" in the *Installation and Configuration Guide* for how to install the Builder.

Log on as root user when installing the script templates on Linux machine.

Follow the procedures below:

1. Insert the ExpressCluster X CD into a CD drive.
2. Mount the CD.

   # mount /dev/cdrom
3. Move to the following directory:

   # cd /mnt/cdrom/Linux/1.0/en/script/Linux
4. Install the script templates using the rpm command.

   # rpm –i expresscls-script-1.0.0-1.linux.i686.rpm

Note:
The rpm file name may vary depending on the version of the Agent. The command description may also vary depending on the type of Linux.

Uninstalling the script templates

Uninstalling the script templates for Windows

To uninstall the script templates, log on as an administrator and delete all folders except defaultl and defaultw under C:\Program Files\CLUSTERPRO\clpbuilder-l\scripts\linux

Note:
Check the directory because the underlined part may vary depending on the terminal environment.

Uninstalling the script templates for Linux

To uninstall the script templates, run the following command as root user:

 # rpm -e expresscls-script

If you uninstall the Builder for Linux, the script templates (the Builder for Linux) will be uninstalled as well, but the rpm module information will be remained. Therefore, it is necessary to forcefully delete the package using the rpm command:

(Example): **rpm –e --force expresscls-script**

Details on script templates

Two templates, start.sh and stop.sh, are provided for the scripts. Modify and use them according to your system requirements. Modifications to the scripts need to be made are underlined boldface italics.

For Samba service startup

start.sh

```
#! /bin/sh
#****************************************
#*            start.sh                  *
#****************************************

if [ "$CLP_EVENT" = "START" ]
then
if [ "$CLP_DISK" = "SUCCESS" ]
then
   echo "NORMAL1"
   if [ "$CLP_SERVER" = "HOME" ]
   then
      echo "NORMAL2"
   else
      echo "ON_OTHER1"
   fi
   echo "SAMBA start"
#
# Start up SAMBA
#
   /sbin/service smb start

else
   echo "ERROR_DISK from START"
fi
elif [ "$CLP_EVENT" = "FAILOVER" ]
then
if [ "$CLP_DISK" = "SUCCESS" ]
then
   echo "FAILOVER1"
   if [ "$CLP_SERVER" = "HOME" ]
   then
      echo "FAILOVER2"
   else
      echo "ON_OTHER2"
   fi
   echo "SAMBA start"
#
# Start up SAMBA
#
   /sbin/service smb start

else
   echo "ERROR_DISK from FAILOVER"
fi
else
echo "NO_CLP"
fi
echo "EXIT"
exit 0
```

stop.sh

```
#! /bin/sh
#***************************************
#*             stop.sh                 *
#***************************************

if [ "$CLP_EVENT" = "START" ]
then
if [ "$CLP_DISK" = "SUCCESS" ]
then
   echo "NORMAL1"
   if [ "$CLP_SERVER" = "HOME" ]
   then
      echo "NORMAL2"
   else
      echo "ON_OTHER1"
   fi
   echo "SAMBA stop"
#
# Stop SAMBA
#
   /sbin/service smb stop

else
   echo "ERROR_DISK from START"
fi
elif [ "$CLP_EVENT" = "FAILOVER" ]
then
if [ "$CLP_DISK" = "SUCCESS" ]
then
   echo "FAILOVER1"
   if [ "$CLP_SERVER" = "HOME" ]
   then
      echo "FAILOVER2"
   else
      echo "ON_OTHER2"
   fi
   echo "SAMBA stop"
#
# Stop SAMBA
#
   /sbin/service smb stop

else
   echo "ERROR_DISK from FAILOVER"
fi
else
echo "NO_CLP"
fi
echo "EXIT"
exit 0
```

For Samba monitoring

start.sh

```
#! /bin/sh
#************************************
#*            start.sh              *
#************************************

if [ "$CLP_EVENT" = "START" ]
then
if [ "$CLP_DISK" = "SUCCESS" ]
then
   echo "NORMAL1"
   if [ "$CLP_SERVER" = "HOME" ]
   then
      echo "NORMAL2"
   else
      echo "ON_OTHER1"
   fi
   echo "clp_sambamon start"
#
# Change shared file name and shared printer name to appropriate value.
#
   clp_sambamon sambawatch -s XXXX

else
   echo "ERROR_DISK from START"
fi
elif [ "$CLP_EVENT" = "FAILOVER" ]
then
if [ "$CLP_DISK" = "SUCCESS" ]
then
   echo "FAILOVER1"
   if [ "$CLP_SERVER" = "HOME" ]
   then
      echo "FAILOVER2"
   else
      echo "ON_OTHER2"
   fi
   echo "clp_sambamon start"
#
# Change shared file name and shared printer name to appropriate value.
#
   clp_sambamon sambawatch -s XXXX

else
   echo "ERROR_DISK from FAILOVER"
fi
else
echo "NO_CLP"
fi
echo "EXIT"
exit 0
```

stop.sh

```
#! /bin/sh
#***************************************
#*             stop.sh                 *
#***************************************

if [ "$CLP_EVENT" = "START" ]
then
if [ "$CLP_DISK" = "SUCCESS" ]
then
   echo "NORMAL1"
   if [ "$CLP_SERVER" = "HOME" ]
   then
      echo "NORMAL2"
   else
      echo "ON_OTHER1"
   fi
   echo "clp_sambamon stop"
#
# Stop clp_sambamon
#
   clp_sambamon sambawatch -stop

else
   echo "ERROR_DISK from START"
fi
elif [ "$CLP_EVENT" = "FAILOVER" ]
then
if [ "$CLP_DISK" = "SUCCESS" ]
then
   echo "FAILOVER1"
   if [ "$CLP_SERVER" = "HOME" ]
   then
      echo "FAILOVER2"
   else
      echo "ON_OTHER2"
   fi
   echo "clp_sambamon stop"
#
# Stop clp_sambamon
   clp_sambamon sambawatch -stop

else
   echo "ERROR_DISK from FAILOVER"
fi
else
echo "NO_CLP”
fi
echo "EXIT"
exit 0
```

For NFS service startup

start.sh

```
#! /bin/sh
#************************************
#*            start.sh              *
#************************************

if [ "$CLP_EVENT" = "START" ]
then
if [ "$CLP_DISK" = "SUCCESS" ]
then
   echo "NORMAL1"
   if [ "$CLP_SERVER" = "HOME" ]
   then
      echo "NORMAL2"
   else
      echo "ON_OTHER1"
   fi
   echo "NFS start"
#
# Start up NFS service
#
   /sbin/service nfs start

else
   echo "ERROR_DISK from START"
fi
elif [ "$CLP_EVENT" = "FAILOVER" ]
then
if [ "$CLP_DISK" = "SUCCESS" ]
then
   echo "FAILOVER1"
   if [ "$CLP_SERVER" = "HOME" ]
   then
      echo "FAILOVER2"
   else
      echo "ON_OTHER2"
   fi
   echo "NFS start”
#
# Start up NFS service
#
   /sbin/service nfs start

else
   echo "ERROR_DISK from FAILOVER"
fi
else
echo "NO_CLP"
fi
echo "EXIT"
exit 0
```

stop.sh

```
#! /bin/sh
#***************************************
#*             stop.sh                 *
#***************************************

if [ "$CLP_EVENT" = "START" ]
then
if [ "$CLP_DISK" = "SUCCESS" ]
then
   echo "NORMAL1"
   if [ "$CLP_SERVER" = "HOME" ]
   then
      echo "NORMAL2"
   else
      echo "ON_OTHER1"
   fi
   echo "NFS_stop"
#
# Stop NFS service
#
   /sbin/service nfs stop

else
   echo "ERROR_DISK from START"
fi
elif [ "$CLP_EVENT" = "FAILOVER" ]
then
if [ "$CLP_DISK" = "SUCCESS" ]
then
   echo "FAILOVER1"

   if [ "$CLP_SERVER" = "HOME" ]
   then
      echo "FAILOVER2"
   else
      echo "ON_OTHER2"
   fi
   echo "NFS stop"
#
# Stop NFS service
#
   /sbin/service nfs stop

else
   echo "ERROR_DISK from FAILOVER"
fi
else
echo "NO_CLP"
fi
echo "EXIT"
exit 0
```

For NFS monitoring

start.sh

```
#! /bin/sh
#************************************
#*            start.sh              *
#************************************

if [ "$CLP_EVENT" = "START" ]
then
if [ "$CLP_DISK" = "SUCCESS" ]
then
   echo "NORMAL1"
   if [ "$CLP_SERVER" = "HOME" ]
   then
      echo "NORMAL2"
   else
      echo "ON_OTHER1"
   fi
   echo "clp_nfsmon start"
#
# Change shared directory name to appropriate value.
#
   clp_nfsmon nfswatch -d XXXX

else
   echo "ERROR_DISK from START"
fi
elif [ "$CLP_EVENT" = "FAILOVER" ]
then
if [ "$CLP_DISK" = "SUCCESS" ]
then
   echo "FAILOVER1"
   if [ "$CLP_SERVER" = "HOME" ]
   then
      echo "FAILOVER2"
   else
      echo "ON_OTHER2"
   fi
   echo "clp_nfsmon start"
#
# Change shared directory name to appropriate value.
#
   clp_nfsmon nfswatch -d XXXX

else
   echo "ERROR_DISK from FAILOVER"
fi
else
echo "NO_CLP"
fi
echo "EXIT"
exit 0
```

stop.sh

```
#! /bin/sh
#***************************************
#*             stop.sh                 *
#***************************************

if [ "$CLP_EVENT" = "START" ]
then
if [ "$CLP_DISK" = "SUCCESS" ]
then
   echo "NORMAL1"
   if [ "$CLP_SERVER" = "HOME" ]
   then
      echo "NORMAL2"
   else
      echo "ON_OTHER1"
   fi
   echo "clp_nfsmon stop"
#
# Stop clp_nfsmon
#
clp_nfsmon nfswatch -stop

else
   echo "ERROR_DISK from START"
fi
elif [ "$CLP_EVENT" = "FAILOVER" ]
then
if [ "$CLP_DISK" = "SUCCESS" ]
then
   echo "FAILOVER1"
   if [ "$CLP_SERVER" = "HOME" ]
   then
      echo "FAILOVER2"
   else
      echo "ON_OTHER2"
   fi
   echo "clp_nfsmon stop"
#
# Stop clp_nfsmon
#
clp_nfsmon nfswatch -stop

else
   echo "ERROR_DISK from FAILOVER"
fi
else
echo "NO_CLP"
fi
echo "EXIT"
exit 0
```

Appendix B. Glossary

Cluster partition	A partition on a mirror disk. Used for managing mirror disks. (Related term: Disk heartbeat partition)
Interconnect	A dedicated communication path for server-to-server communication in a cluster. (Related terms: Private LAN, Public LAN)

Virtual IP address	IP address used to configure a remote cluster.
Management client	Any machine that uses the WebManager to access and manage a cluster system.
Startup attribute	A failover group attribute that determines whether a failover group should be started up automatically or manually when a cluster is started.
Shared disk	A disk that multiple servers can access.
Shared disk type cluster	A cluster system that uses one or more shared disks.
Switchable partition	A disk partition connected to multiple computers and is switchable among computers. (Related terms: Disk heartbeat partition)
Cluster system	Multiple computers are connected via a LAN (or other network) and behave as if it were a single system.
Cluster shutdown	To shut down an entire cluster system (all servers that configure a cluster system).
Active server	A server that is running for an application set. (Related term: Standby server)
Secondary server	A destination server where a failover group fails over to during normal operations. (Related term: Primary server)
Standby server	A server that is not an active server. (Related term: Active server)
Disk heartbeat partition	A partition used for heartbeat communication in a shared disk type cluster.
Data partition	A local disk that can be used as a shared disk for switchable partition. Data partition for mirror disks. (Related term: Cluster partition)
Network partition	All heartbeat is lost and the network between servers is partitioned. (Related terms: Interconnect, Heartbeat)
Node	A server that is part of a cluster in a cluster system. In networking terminology, it refers to devices, including computers and routers, that can transmit, receive, or process signals.
Heartbeat	Signals that servers in a cluster send to each other to detect a failure in a cluster. (Related terms: Interconnect, Network partition)
Public LAN	A communication channel between clients and servers. (Related terms: Interconnect, Private LAN)

Failover	The process of a standby server taking over the group of resources that the active server previously was handling due to error detection.
Failback	A process of returning an application back to an active server after an application fails over to another server.
Failover group	A group of cluster resources and attributes required to execute an application.
Moving failover group	Moving an application from an active server to a standby server by a user.
Failover policy	A priority list of servers that a group can fail over to.
Private LAN	LAN in which only servers configured in a clustered system are connected. (Related terms: Interconnect, Public LAN)
Primary (server)	A server that is the main server for a failover group. (Related term: Secondary server)
Floating IP address	Clients can transparently switch one server from another when a failover occurs. Any unassigned IP address that has the same network address that a cluster server belongs to can be used as a floating address.
Master server	The server displayed on top of the **Master Server** in **Cluster Properties** in the Builder.
Mirror connect	LAN used for data mirroring in a data mirror type cluster. Mirror connect can be used with primary interconnect.
Mirror disk type cluster	A cluster system that does not use a shared disk. Local disks of the servers are mirrored.

Appendix C. Index

A

adding a group, 41
adding a group resource (disk resource), 42
adding a group resource (floating IP address), 41
adding the exec resource (exec 1), 39, 43
adding the exec resource (exec 2), 39, 50
alert messages, 32, 33

B

before using the File Server Agent, 15

C

clp_nfsmon, 20, 26
clp_sambamon, 20, 22
command reference, 19
Confirmation test, 39, 45
creating a failover group, 41

D

details on script templates, 61

E

ExpressCluster, 15

F

failing over a group, 49, 57
File Server, 14
File Server Agent, 11, 12
For NFS monitoring, 67
For NFS service startup, 65
For Samba monitoring, 63
For Samba service startup, 61

H

hardware, 15

I

installing script templates on Linux machine, 60
installing the script templates on Windows machine, 59

L

license registration, 11, 15, 16
list of monitoring commands, 20
log collection when an error has occurred, 32

M

mechanism of monitoring, 13
memory space, 15
Messages being output by clp_nfsmon, 36
Messages being output by clp_sambamon, 33
Messages being output when an error is detected in file server monitoring, 34
Messages being output when error is detected in file server monitoring, 37
messages due to setting error, 33, 36
messages due to system error, 35, 37
messages indicating normal operation, 33, 36
monitor information with monitoring commands, 32
monitoring chart, 20
monitoring command, 39, 56
moving a group, 48, 57

N

NFS, 26

O

OS, 15

P

pid monitor resource, 53

R

registering the license, 17
registering the license interactively, 16

S

Samba, 14, 22
setting up, 39, 40
setting up script templates, 59
specifying a license file, 17
start.sh, 61, 63, 65, 67
starting up a group, 48, 56
stop.sh, 62, 64, 66, 68
stopping a group, 48, 56
suspending and resuming monitoring, 21
system requirements, 15

T

target monitoring application, 14, 39, 45

U

uninstalling the script templates, 60
uninstalling the script templates for Linux, 60

uninstalling the script templates for Windows, 60
Using the File Server Agent, 15

V

verifying the behavior of the target monitoring application, 48
verifying the settings, 39, 56

W

WebManager, 32
writing scripts, 21