

Sterling Control Center



How-To Guide

Version 54.1

Sterling Control Center



How-To Guide

Version 54.1

Note

Before using this information and the product it supports, read the information in "Notices" on page 79.

This edition applies to version 5.4.1 of IBM Sterling Control Center and to all subsequent releases and modifications until otherwise indicated in new editions.

© **Copyright IBM Corporation 2000, 2013.**

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Chapter 1. Setting up data visibility groups 1

- How can I allow a user to see only inventory processes across multiple servers? 3
- How Can I Allow a User to see only data from a specified Remote Node? 3
- How can I allow a user to see only when a specific file transfer occurs? 4
- How can I create a rule that applies only to data that a specific user can see? 4

Chapter 2. Setting object permissions.. 5

- Can I control how users work with Sterling Control Center objects? 5
- How can I let everyone use a calendar but allow only an administrator to update the calendar? 5
- How can I restrict the use and update of a calendar to limited users? 6
- How can I allow all users to use a calendar but only one user to update the calendar? 6

Chapter 3. Setting up actions 7

- How can I create an action to release a Sterling Connect:Direct process? 7
- How can I create an action that sends an email that includes an SLC name? 7
- How can I create an action to send an email and generate an alert? 7
- How do I generate a high-severity alert and an SNMP trap? 8
- How do I pass an operating system command and variables to open a help desk ticket? 8

Chapter 4. Setting up rules 11

- Rules involving general topics 11
 - How can I know when a Process completes? .. 11
 - How can I know when a Process has a problem? 11
 - How can I know when a copy completes? . . . 12
 - How can I know when a copy completes unsuccessfully? 13
 - How can I know when a monitored server goes down? 13
 - How can I know when a monitored server starts? 14
 - How can I send an email in the event of a critical database problem? 14
 - How can I generate a server down alert only after the server is down more than 5 minutes?.. 14
 - How can I generate an alert when a managed server stops and handle the alert automatically when the server restarts? 15
 - How can I know when a Sterling Connect:Direct Process goes in the Held in Error Queue? . . . 16
 - How can I avoid receiving alerts from both servers involved in a process? 17

- How can I specify more than one message ID in the rule criteria? 17
- How can I know if a new version of objects was created? 18
- How can I know when Sterling Control Center is not monitoring a server regularly?. 18
- How can I know if the number of items being executed by a server exceeds a specified threshold? 18
- How can I know when my Sterling Connect:Direct Secure Plus certificates are about to expire? 20
- Rules involving Sterling B2B Integrator 20
 - How can I know when a Sterling B2B Integrator Adapter stops? 20
 - How can I know when a Sterling B2B Integrator adapter is enabled?. 21
 - How can I know when a Sterling B2B Integrator adapter goes down and stays down for more than 5 minutes? 21
- Rules involving Sterling File Gateway 21
 - How to get notified when a Sterling File Gateway route fails? 21
 - How can a producer know when a file is delivered to a Sterling B2B Integrator mailbox? . 22
 - How can a producer know whether a file was delivered to a consumer's mailbox? 22
 - How can a producer get notified when a file does not arrive in their mailbox within 5 minutes after the connection is initiated? 23
- Rules involving Sterling Connect:Enterprise . . . 23
 - How can I know whether a Sterling Connect:Enterprisedaemon goes down and stays down for more than 5 minutes? 23
 - How can I know when a Sterling Connect:Enterprise for UNIX daemon stops? .. 24
 - How can I know when a Sterling Connect:Enterprise for UNIX daemon starts? .. 24
 - How can I know when a Sterling Connect:Enterprise for UNIX offline add was performed? 25
 - How can I know when a Sterling Connect:Enterprise for UNIX auto connect fails? . 25
 - How can I know when a Sterling Connect:Enterprise remote connect connection is performed? 25
 - How can I know when a Sterling Connect:Enterprise for UNIX auto connect collect operation is performed? 25
 - How can I know when a Sterling Connect:Enterprise for UNIX auto connect transmit operation is performed? 25
 - How can I know when a Sterling Connect:Enterprise for UNIX offline extraction is performed? 26

How can I know when a Sterling Connect:Enterprise remote connection request operation was performed?	26	How do I know which type of SLC to use?.	44
How can I know when a Sterling Connect:Enterprise remote connection add operation was performed?	26	How can I be alerted if one process runs too long?	44
How can I know when a Sterling Connect:Enterprise remote connect delete is performed?	26	How can I be alerted if a critical process fails to complete on time?	45
How can I know when a Sterling Connect:Enterprise remote connect directory list is performed?.	26	How can I be alerted when related Sterling Connect:Enterprise batches arrive?.	45
How can I know when a Sterling Connect:Enterprise remote disconnection is performed?.	26	How can I be alerted if a related group of processes runs too long?	47
How can I know when a Sterling Connect:Enterprise for z/OS auto connect CONNECT operation is performed?	26	How can I be alerted if a related group of critical processes do not all complete on time?	49
How can I know when a Sterling Connect:Enterprise for z/OS auto connect DISCON operation is performed?	27	How can I be alerted if a critical process does not start on time?.	49
How can I know when a Sterling Connect:Enterprise for z/OS auto connect send operation is performed?	27	How can I be alerted if one or more related processes do not start or complete on time?	49
How can I know when a Sterling Connect:Enterprise for z/OS auto connect receive operation is performed?	27	How can I be alerted if one or more related processes might not complete on time?	50
QuickFile rules	27	How can I reduce the number of events with the same message generated by workflow SLCs?	51
How can I know if a request for a file has been satisfied in time?	27	How can I be alerted if a Sterling Connect:Direct file transfer may not complete on time?	51
How can I know if an uploaded file is downloaded by a recipient within a certain amount of time?.	29	How can I be alerted if an hourly process is not on time?	52
Chapter 5. Troubleshooting rules	33	How can a producer know when a file was not delivered on time?	54
Troubleshooting by using the Database Events report	33	How can a producer get notified if the consumer does not pick up a file within 15 minutes?	54
Chapter 6. Setting up schedules.	35	How can I use RUNBATCH to create schedules for an hourly process?	55
How can I build a schedule to prevent rules from triggering during my maintenance windows?	35	Chapter 9. Troubleshooting SLCs	59
Chapter 7. Reports example	37	SLC tips	59
How do I set up an automated report of alerts for the previous day?	37	How do I fix my SLC?.	60
Chapter 8. Setting up SLCs	39	SLC not enabled.	60
What is the correlation between SLCs and rules? ..	39	Match criteria.	60
When should an SLC be used versus a rule? ..	39	A.M. versus P.M.	61
When do I need SLCs?	39	Monitor windows	61
What events may be generated by SLCs and when?	40	Monitored servers	62
SLC notifications for calendar schedule-based SLCs	40	Rules and actions	62
SLC notifications for duration schedule-based SLCs	42	Milestone times	62
		SLC order	63
		Time zones	63
		Everything is correct except the alerts	63
		What else can I do?.	64
		Chapter 10. Updating information for multiple servers.	77
		About changing passwords in mass	77
		How can I do a bulk update of the passwords used by Sterling Control Center for monitored servers?	77
		Notices	79
		Index	83

Chapter 1. Setting up data visibility groups

Use data visibility groups (DVGs) when there is a need to limit what data (events) a specific user can monitor.

For example, assume that a company has one IBM Sterling Connect:Direct server on which both Payroll and Order Processing work is performed. The IBM Sterling Control Center administrator wants to give both Payroll and Order Processing personnel access to Sterling Control Center functions. However, the administrator does not want Order Processing personnel to have any visibility to Payroll file activity; nor Payroll personnel to have any access to Order Processing activity.

Data visibility groups in Sterling Control Center provide this capability.

Data visibility groups are basically a set of rules that are applied to events that are created in Sterling Control Center. When the rule matches the criteria specified, the name of the data visibility group is added to the attributes of the event. So, the Sterling Control Center administrator can set up a data visibility group called Payroll. This DVG classifies events as “Payroll” events (by tagging them with the “Payroll” Data Visibility Group attribute) when any one of these criteria is met:

- Any Process submitted by the user PAY01 is a Payroll process.
- Any Process that runs between this Sterling Connect:Direct server and the remote Payroll server is a Payroll process.
- Any Process Step that deals with a file that contains “corp.payroll” in the name is a Payroll Process Step.

And the Sterling Control Center administrator can set up a data visibility group called OrderProcessing. This data visibility group classifies events as “OrderProcessing” events when any one of these criteria is met:

- All Processes submitted by user ORD01 are Order Processing processes.
 - All Processes submitted by user ORD02 are Order Processing processes.
 - All Process Steps that deal with files that begin with “ORDER” are Order Processing processes.
1. For the Payroll data visibility group, the administrator would set up the following criteria:

Criterion #	Name	Key	Operator	Value
1	Payroll Criteria - User PAY01	Submitter	Matches	PAY01
		Servers and Server Groups	Matches	(CDServer)()
2	Payroll Criteria - Remote Node	Remote Node	Matches	ABCPayrollNode
		Servers and Server Groups	Matches	(CDServer)()
3	Payroll Criteria - Payroll File	Destination File	Wildcard	corp.payroll*
		Source File	Wildcard	corp.payroll*

2. For the OrderProcessing data visibility group, the administrator would set up the following criteria:

Criterion #	Name	Key	Operator	Value
1	OrderProcessing Criteria 1 - User ORD01	Submitter	Matches	ORD01*
		Servers and Server Groups	Matches	(CDServer)()
2	OrderProcessing Criteria 2 - User ORD02	Submitter	Matches	ORD02
		Servers and Server Groups	Matches	(CDServer)()
3	OrderProcessing Criteria 3 - File ORDER	Destination File	Matches	ORDER*
		Source File	Matches	ORDER*
		Server and Server Groups	Matches	(CDServer)()

3. A Sterling Control Center role can now be defined that is restricted to data visibility group Payroll. The administrator creates a role name Payroll-Role. On the Restrictions tab of the Role Properties sheet, the administrator moves the Payroll DVG to the Selected Data Visibility Groups.

Name	Value
Selected Data Visibility Groups	Payroll

4. Another role can be defined that is restricted to data visibility group OrderProcessing. The administrator creates a role named OrderProcessing - Role. On the Restrictions tab of the Role Properties sheet, the administrator moves the OrderProcessing DVG to the Selected Data Visibility Groups.

Name	Value
Selected Data Visibility Groups	OrderProcessing

5. Sterling Control Center users can now be defined and assigned the appropriate Payroll or Order Processing roles. The administrator sets up the following criteria for the Payroll - Role on the User Properties screen:

Name	Value
User ID	PayrollUser
Password	*****
Retype Password	*****
Role	Payroll - Role

6. The administrator would set up the following criteria for the Order Processing role on the User Properties tab:

Name	Value
User ID	OrderProcessingUser

Name	Value
Password	*****
Retype Password	*****
Role	OrderProcessing - Role

When the PayrollUser user signs on to the Sterling Control Center console only Payroll activities are displayed due to that user's DVG restriction. It is as if those activities were the only activities in the Sterling Connect:Direct server. And User OrderProcessingUser would see Order Processing activities only. If the Payroll role is permitted to define rules or SLCs, those rules or SLCs would apply to Payroll activity only. The rules would have no effect on Order Processing activity.

How can I allow a user to see only inventory processes across multiple servers?

Create a data visibility group with a criterion that includes selections of Servers and a Process Name that represents inventory processes. Create a role that is restricted to the Inventory data visibility group. Assign that role to the user who is only allowed to see inventory processes.

About this task

The administrator would create a data visibility group with the following parameters criteria:

Name	Key	Operator	Value
Inventory Data Visibility Group Criteria 1	Servers and Server Group	Matches	(CDServer CEzOS)()
	Process Name	Matches	inventory*

How Can I Allow a User to see only data from a specified Remote Node?

You can restrict a user to see only data from a specific remote node by specifying that node in a data visibility group.

Procedure

1. Create a data visibility group with a criterion that specifies the Remote Node. The following table illustrates sample criterion of a data visibility group for Node A:

Name	Key	Operator	Value
NodeA Criteria 1	Remote Node	Matches	NodeA

2. Create a role with the data visibility group selected and assign that role to your user.

How can I allow a user to see only when a specific file transfer occurs?

You can restrict a user to see only when a specific file transfer occurs by specifying the specific file transfer in a DVG.

Procedure

1. Create a data visibility group with a criterion that defines the specific file transfer that the user can see. The following table illustrates sample criterion of a data visibility group for FileABC.

Name	Key	Operator	Value
FileABC Criteria	Destination File	Matches	FileABC
	From Server	Matches	FTPServerXYZ

2. Create a role with the data visibility group selected and assign that role to the user.

How can I create a rule that applies only to data that a specific user can see?

Create a data visibility group that defines the data you want the user to see. Create a role that is restricted to that data visibility group. Assign that role to the user.

About this task

Do one of the following procedures:

Procedure

- Let the Sterling Control Center administrator create the rule and set the data visibility group of the user's role in the parameters of the rule.
- Have the users create their own rules; the users are forced to supply their data visibility group as part of the parameter of the rule since they are restricted to that data visibility group.

Example

Name	Key	Operator	Value
PayrollRule	Data Visibility Group	Matches	Payroll
	Event Type	Matches	Process Step Ended

Chapter 2. Setting object permissions

Can I control how users work with Sterling Control Center objects?

You can use object permissions to limit the users who can view/use an object and the users who can update objects.

You can control these types of permissions for the following objects in Sterling Control Center:

Note: Unrestricted users can always view/use the objects.

- Rule actions
- Rule schedules
- SLC schedules
- Message lists
- Metadata schedules
- Calendars
- Email lists
- Report schedules
- Metadata actions

Setting object permissions

You set object permissions on the Permissions tab for the object type you are working with. The following table illustrates the Permissions tab settings for a calendar object:

Name	Selected Roles	Visibility
PayCalendar	Payroll	This Calendar is visible to restricted users in these Selected Restricted Roles

On the Permissions tab:

- “Selected Restricted Roles:” Identify which Restricted roles can update the object.
- The radio button identifies the users who can view/use the object.

How can I let everyone use a calendar but allow only an administrator to update the calendar?

You can let everyone use a calendar while restricting calendar updates to administrators by using the Calendar Properties.

Procedure

Select “This Calendar is visible to all users” on the **Permissions** tab in the Calendar Properties window.

How can I restrict the use and update of a calendar to limited users?

You can restrict the use and update of a calendar to limited users on the Permissions tab of the Calendar Properties.

Before you begin

Select the roles you want to restrict, select **This Calendar is visible to restricted users in these Selected Restricted Roles**, and update the calendar.

Procedure

1. Select **This Calendar is visible to restricted users in these Selected Restricted Roles** on the **Permissions** tab of the Calendar Properties window.
2. Specify the roles of the users you want to be able to use.
3. Update the calendar in the Selected Restricted Roles.

Example

The following table illustrates sample settings of the **Permissions** tab of the PrivateOPCalendar Properties:

Name	Selected Restricted Roles	Visibility
PrivateOPCalendar	OrderProcessing - Role	This Calendar is visible to restricted users in these Selected Restricted Roles
	OrderTraineeRole	

How can I allow all users to use a calendar but only one user to update the calendar?

You can allow all users to use a calendar and restrict updates to one user with the Calendar Properties Permissions tab.

Procedure

1. Select "This Calendar is visible to all users" on the Permissions tab of the calendar.
2. Specify the role of the user you want to be able to update the calendar in the Selected Restricted Roles.

Note: Do not assign the role to any other user.

3. Update the Calendar Properties.

Example

The following table illustrates sample settings of the Calendar Settings Permissions tab:

Name	Selected Restricted Roles	Calendar Visibility
BiWeekly	Payroll - Role	This Calendar is visible to all users

Chapter 3. Setting up actions

How can I create an action to release a Sterling Connect:Direct process?

You can create an action with a Server Command to release a Sterling Connect:Direct® process.

About this task

Create an action with Server Command equal to the following command:
"change process hold=no pname=&processName; pnumber=&processid;"

You can do this easily using the Release Process Command template provided in the Create Action wizard. The following table illustrates sample settings of a Create Action wizard:

Action	Server	Command Template	Server Command
Server Command	<Server that generated the event>	Release Process Command	change process hold=no pname= &processName; pnumber= &processid;

To make the action release processes to attempt to run again, you could specify this action in a rule. The rule could, for example, watch for Sterling Connect:Direct Processes that make the transition to the Held in Error queue. (See *How can I know when a Sterling Connect:Direct Process goes in the Held in Error Queue?*)

How can I create an action that sends an email that includes an SLC name?

To plug the SLC name in the subject line in emails generated as a result of SLC events, you can use the event element variable called `groupId`.

Procedure

1. First, you create an action and add the email address along with the descriptive text you want in the Subject line on the **E-mail Properties** tab.
2. Click **Insert Var** and select `groupId` in the Select Variable window. Emails generated as a result of this action contain the SLC name in their subject.

How can I create an action to send an email and generate an alert?

You can create an action that sends an email notification to a system administrator and generates an in-compliance alert.

About this task

Specify this action in a rule setup to trigger when a certain critical Process completes successfully. The following example illustrates this setup.

Procedure

1. Click **Manage > Rules and Actions > Actions** and click the Create button to add an action.
2. Complete **Name** and **Description**.
3. Click **Next** and create email entries similar to the following sample. Use **Insert Var** to add variables to Subject and Message. The following table illustrates sample settings for an email Create Action wizard:

Name	To:	From:	Subject	Message
Notify SysAdmin Success	Admin@xyz.com	SCCCConsole@xyz.com	&processId; &processName; completed	Process &processId; &processName; completed normally.

4. Click **Next** and choose **0 - In compliance** for Alert Severity.
5. Leave the remaining wizard panels unchanged.

How do I generate a high-severity alert and an SNMP trap?

You can set up an action that generates a high-severity alert and an SNMP trap to an Enterprise System Management (ESM) host. The ESM host must be defined in the Sterling Control Center ESM Host settings.

About this task

You could specify this action in a rule that triggers, for example, whenever any file transfer on a managed Sterling Connect:Enterprise server fails (that is, completes with a return code greater than 4).

For more information, see *Sterling Control Center Settings* in the Help or *System Administration*.

Procedure

1. Create a General action with a **Name** and **Description** similar to the following values:
 - Name: Generate Alert and Trap
 - Description: Generate high-severity alert and trap to the console
2. Skip the email wizard panel and go to the Alert panel. Select **1 - High** for severity and click **Next**.
3. On the SNMP Trap panel, select **SNMP Trap**.
4. Then click through subsequent panels and confirm your choices.

How do I pass an operating system command and variables to open a help desk ticket?

You can create an action that calls a user-written program (named GenHelpTicket.exe in this example) that opens a Help Desk ticket when the rule it is attached to is triggered.

About this task

The action also passes an XML string of event data to the user-written program. When calling an operating system command, remember that the called program

must reside on the same computer on which the Sterling Control Center engine is located.

Procedure

1. Create an action with a **Name** and **Description** similar to the following.
 - Name: Open Ticket
 - Description: Call GenHelpTicket to open ticket
2. Skip to the **OS Command** wizard panel and make the following entry.
`"c:\Program Files\GenHelpTicket.exe" &processName; &processId;`

Note: The entry must be enclosed in quotation marks because of the space in the file path name. Here is an example that involves UNIX where no quotation marks are required: `cmd /c copy C:\Users\sflan\apkeyf1.txt D:\Users\sflan\apkeyf2.txt`

3. Use **Insert Var** to insert the Process Name and Process ID variable names.
4. Click **Next** through the remaining panels and confirm your choices.

Chapter 4. Setting up rules

Rules involving general topics

The general rules examples discuss set up of rules to monitor general situations frequently seen by Sterling Control Center users.

These rule examples do not describe the detailed steps necessary to perform a procedure. Some examples describe only the idea behind a solution to a problem. For a detailed explanation of rules and how to set them up, see *System Administration*.

How can I know when a Process completes?

You can trigger a rule when a Process completes by setting the event type of that rule to Process Ended.

About this task

For a rule to trigger when a Process completes on a managed server, set the event type of the rule to Process Ended. Processes include an IBM Sterling B2B Integrator business process or a Sterling Connect:Direct Process. Processes can also include, for example, a batch added by a remote user to a Sterling Connect:Enterprise mailbox. No other rule attributes must set. However, to limit the times when this rule would fire, you can set additional values such as an individual server or server group, a Process Name, or a submitter.

The following example illustrates this type of rule setup.

To know when the Process ISITDONE completes on any server in the Server Group known as My Group, specify the following criteria as parameters when you create the rule:

Key	Operator	Value
Event Type	Matches	Process Ended
Servers and Server Groups	Matches	()(MyGroup)
Process Name	Matches	ISITDONE

This rule will now fire only when Processes named ISITDONE completes on servers in MyGroup.

How can I know when a Process has a problem?

Create a rule that triggers when a Process completes in error on a managed server.

About this task

No other rule attributes must be set. However, you can limit the times this rule would fire by setting additional values. For example, to know when the Process ISITDONE completes in error on any server in the Server Group known as MyGroup, specify the following parameters:

Key	Operator	Value
Event Type	Matches	Process Ended
Process Name	Matches	ISITDONE
Servers and Server Groups	Matches	()(MyGroup)
Return Code	Not equal to	0

This rule fires only when Processes named ISITDONE complete with a non-zero return code on servers in the server group named MyGroup.

How can I know when a copy completes?

For a rule to trigger when a file transfer completes on a managed server, specify an event type of Process Step Ended.

About this task

File transfers are accomplished via what Sterling Control Center interprets as Process steps. For Sterling Control Center purposes, this category even includes transfers that involve Sterling B2B Integrator servers that do not include a business process. For a rule to trigger when a Process step completes on a managed server (Sterling B2B Integrator Business Process Activity, a transfer handled by a Sterling B2B Integrator adapter such as the CDServerAdapter, a Sterling Connect:Direct Process Step, or a batch added by a remote user to a Sterling Connect:Enterprise mailbox), set the event type to Process Step Ended. You do not need to set other rule attributes. However, to limit the times when this rule fires, set attributes such as server groups, servers, and schedules. You can also set parameters like Destination File, Source File, Process Name, or Step Name.

Note: For managed Sterling Connect:Enterprise servers, the Key parameter Submitter equates to the senders Mailbox ID. The Process Name equates to Batch ID, and Remote Node equates to the recipients Mailbox ID.

For example, to limit the triggering of your rule to when a file is added to the mailbox named Fred, specify a Key of Remote Node, with an Operator of Matches, and a Value of Fred.

For a Sterling B2B Integrator example: To know when the activity named WaitService in the Sterling B2B Integrator business process ISITDONE completes on the managed server named phoenix-gis 43, specify the following parameters when you create the rule:

Key	Operator	Value
Process Name	Matches	ISITDONE
Step Name	Matches	WaitService
Event Type	Matches	Process Step Ended
Server ID	Matches	phoenix-gis 43

This rule fires only when the Activity named WaitService completes in a business process named ISITDONE on the Sterling B2B Integrator-managed server named phoenix-gis 43.

How can I know when a copy completes unsuccessfully?

To trigger a rule when a file transfer completes unsuccessfully, you must specify the Event Type and Return Code parameter keys.

About this task

File transfers are accomplished via what Sterling Control Center interprets as Process Steps. For Sterling Control Center purposes, this category even includes transfers with Sterling B2B Integrator servers that do not include a business process. For a rule to trigger when a Process Step completes in error on a managed server (whether a Sterling B2B Integrator business process activity, a Sterling Connect:Direct Process step, or, for example, a batch added by a remote user to a Sterling Connect:Enterprise mailbox), you must specify two parameter keys: an event type of Process Step Ended and a Return Code with an Operator of Not Equal To and a Value of 0. No other rule attributes must be set. However, you can further limit the times this rule would fire by setting additional values such as a particular server or a schedule.

For example, to know when the activity called WaitService in business process ISITDONE completes in error on Sterling B2B Integrator-managed server phoenix-gis 43, specify the following parameters when you create the rule:

Key	Operator	Value
Process Name	Matches	ISITDONE
Step Name	Matches	WaitService
Event Type	Matches	Process Step Ended
Server Id	Matches	phoenix-gis 43
Return Code	Not equal to	0

How can I know when a monitored server goes down?

You can be notified when a monitored server goes down.

About this task

When Sterling Control Center is no longer in communication with a monitored server, it assumes the server has stopped and generates an event of type Server Status with a message ID of CCTR034E.

Procedure

1. To be alerted of this condition, create a rule that specifies the following parameter:

Key	Operator	Value
Message ID	Matches	CCTR034E

2. Specify an action with a severity level that generates an alert (such as severity level 2).

Tip: The Sterling Control Center consoles also relate real-time information about server up or down status. For more information about how server status is shown, see *Sterling Control Center Console* in *User*.

How can I know when a monitored server starts?

About this task

When Sterling Control Center establishes communication with a monitored server, it assumes that the server is started and generates an event of type Server Status with a message ID of CCTR033E.

Procedure

To be alerted of this condition, create a rule that specifies the following parameter:

Key	Operator	Value
Message ID	Matches	CCTR033E

Note: Of course, the Sterling Control Center GUI also relates real-time information about server up or down status. A server in up status displays in normal text, with an activity graph illustrating any current activity.

How can I send an email in the event of a critical database problem?

About this task

This case involves monitoring for a CJDB026E message, which indicates that an SQL exception has occurred within the Sterling Control Center engine that is not a truncation or duplication exception. When such a problem occurs, Sterling Control Center shuts down. In this event, you would probably want to email a responsible party that Sterling Control Center has initiated shutdown due to a critical database problem.

Procedure

1. To do so, first set up a new action that will send an email to the appropriate party and generate a high-severity alert.
2. Next, create a rule that monitors for a message ID of CJDB026E by specifying the following parameter:

Key	Operator	Value
Message ID	Matches	CJDB026E

3. Within this rule, associate the email action you just created.

How can I generate a server down alert only after the server is down more than 5 minutes?

To generate an alert when a server is down more than 5 minutes, you must create a linked rule. The linked rule monitors when the server stops and takes no action unless the server does not start in more than 5 minutes.

About this task

When Sterling Control Center can no longer communicate with a monitored server, it assumes that the server stopped. It then generates an event of type Server Status with a message ID of CCTR034E. Also, when Sterling Control Center establishes

communication with a monitored server, it assumes that the server is started. It generates an event of type Server Status with a message ID of CCTR033E.

This information can be used in conjunction with a linked rule to generate an alert for a server down condition only if the server is down for more than, say, 5 minutes. This alert gives a system administrator a chance to restart the server before any alert is generated. This rule is useful when brief server outages are of minor consequence and you want an alert generated only when a longer-term outage occurs.

A linked rule is a normal rule with additional attributes.

To construct a rule that alerts you when a server is down more than 5 minutes:

Procedure

1. Build a rule that monitors for the message that Sterling Control Center generates when it thinks a server stopped.
2. Specify a Key of Message ID, an Operator of Matches, and a Value of CCTR034E. You can specify additional parameters to limit when this rule is triggered.
3. Select the **No Operation** action, since you do not want to generate an alert when the server is first discovered to be down. You want to generate an alert only after it is down for more than 5 minutes.
4. Select **Enabled** on the Linked Rules wizard that appears next.
5. Specify a following parameter **Key of Message ID**, an **Operator of Matches**, and a **Value of CCTR033E**, as an event that is generated by the Sterling Control Center engine when it finds that a managed server was started. You can also specify other parameters to refine the criteria under which CCTR033E events are matched by the linked part of the rule.
6. Since you selected **No Operation** for the initial action, choose the same for the **Resolution** action. Nothing must be done if the server comes back up within 5 minutes. Choose an appropriate action, such as alert1, for the Non-Resolution action.

Important: If you selected an alert action for the initial action, you must choose alert0 for the resolution action, which tells Sterling Control Center to clear the first alert, and again an appropriate action for the Non-Resolution action, such as alert1. When the server is first discovered to be down, a lower-severity alert is generated. If the server remains down for more than the allowed amount of time, a higher-severity alert is generated.

7. Set the timeout value to the time the Sterling Control Center engine must wait before taking either the Resolution or the Non-Resolution action, in this case 5 minutes.

How can I generate an alert when a managed server stops and handle the alert automatically when the server restarts?

You can use a linked rule to generate an alert when a managed server stops and automatically handle the alert when the server restarts.

About this task

When Sterling Control Center can no longer communicate with a monitored server, it assumes that the server stopped. It then generates an event of type Server Status

with a message ID of CCTR034E. Also, when Sterling Control Center establishes communication with a monitored server, it assumes that the server is started and generates an event of type Server Status with a message ID of CCTR033E. This information can be used in conjunction with a linked rule to generate an alert for a server down condition and to automatically clear any alert generated if and when the server restarts.

A linked rule is nothing more than a normal rule with additional attributes. To construct a rule to alert you when a server is down:

Procedure

1. Build a rule with a parameter by using a **Key** of **Message ID**, an **Operator** of **Matches**, and a **Value** of **CCTR034E**. CCTR024E is the event generated by the Sterling Control Center engine when it thinks a server stopped. You can specify additional parameters to limit when this rule is triggered.
2. Select **alert1** as the **Action** to generate an alert of severity 1 when the Server stops.
3. Select **Enabled** on the Linked Rules wizard that appears next.
4. For linked rule parameters, specify a parameter with a **Key** of **Message ID**, an **Operator** of **Matches**, and a **Value** of **CCTR033E**, as an event that is generated by the Sterling Control Center engine when it finds that a managed server starts. Other parameters might also be specified as necessary to ensure that only the appropriate CCTR033E events are matched.
5. Since you want to clear the alert generated when the server down occurred when the system sees that the server restarted, choose **alert0** for the **Resolution** action, and the action **No Operation** for the **Non-Resolution** action.
6. Select **alert0** for the **Non-Resolution** action as well if you always want the alert cleared. The **Non-Resolution** action does not occur until the number of minutes specified for timeout transpires.
7. Set the timeout value to the time the Sterling Control Center engine waits before taking either the **Resolution** or the **Non-Resolution** action. That is, set the timeout value to the maximum time you feel it takes for a server outage to be handled.
- 8.

How can I know when a Sterling Connect:Direct Process goes in the Held in Error Queue?

You can configure a rule to trigger an alert when a Sterling Connect:Direct Process is placed in the Hold Queue after it received an error.

About this task

For a rule to trigger when a Sterling Connect:Direct Process is placed in the Hold Queue after it received an error (status = HE, Held in Error), perform the following tasks:

Procedure

1. Specify a parameter with the following settings:

Key	Operator	Value
Message Id	Matches	CNCD061I

2. Values for Servers and Server Groups and additional parameters can be specified to make this rule more specific if wanted.

When this rule is associated with an action that issues the Sterling Connect:Direct Release Process command, whenever Sterling Control Center finds a Sterling Connect:Direct Process is moved to the Hold Queue with an HE status, it tells the managed server to release it to run again.

How can I avoid receiving alerts from both servers involved in a process?

If both servers involved in a failed process are managed by Sterling Control Center, two alerts may be generated.

About this task

The redundant alert can be eliminated by setting up a rule that causes Sterling Control Center to ignore one of the two alert conditions.

Procedure

1. Create a rule that specifies the following parameters:

Key	Operator	Value
Event Type	Matches	Process Step Ended
Return Code	Not equal to	0
Local Node (P or S)	Matches	S

2. Specify an action of “No operation” and finish by confirming your choices.

Note: Except for the additional parameter of “Local Node (P or S) Matches S,” this rule is essentially a duplicate of the Bad Return Code rule that ships with Sterling Control Center. To work, it must be placed above the Bad Return Code rule in the Rules listing. In other words, it must have a higher priority in the listing.

CAUTION:

This technique works except in the following situations:

- If a PNODE = SNODE Process step fails, two alerts are still generated.
- No alerts are generated if the PNODE server involved in the Process is unmanaged. To solve this problem, move a rule that specifies a particular server as the SNODE for a Process before this rule in the Rules listing.

How can I specify more than one message ID in the rule criteria?

Use regular expressions to specify more than one message ID in a rule.

About this task

The following table illustrates sample criterion of a rule for SI Adapter down more than 5 minutes:

Key	Operator	Value
Message Id	Reg Ex	CACL004E CDIS028E

For details, see *Regular expressions* in *System Administration*.

How can I know if a new version of objects was created?

About this task

When a new version of objects is created for a server in the Sterling Control Center object repository a Server command event is generated with one of the following message IDs:

- CCFG191I – New version of Netmap Nodes created
- CCFG192I – New version of Netmap Modes created
- CCFG193I – New version of Netmap Communication Paths created
- CCFG194I – New version of Init ParmS created
- CCFG195I – New version of Translation tables created
- CCFG196I – New version of User Proxies created
- CCFG197I – New version of User Auth entries created
- CCFG198I – New version of Sterling Connect:Direct Secure Plus Nodes created
- CCFG199I – New version of Sterling Connect:Direct Secure Plus Key Certificates created
- CCFG200I – New version of Sterling Connect:Direct Secure Plus Trusted Certificates created

You may write multiple rules, one for each message ID, or write one rule that specifies one, or more, of these message IDs using RegEx as the Operator. See *How can I specify more than one Message ID in the rule criteria?*.

How can I know when Sterling Control Center is not monitoring a server regularly?

To know when monitoring of a server is not taking place on a regular basis, use the built-in rule, *Monitoring rate out of compliance*.

Before you begin

Note: To be on the safe side, make a copy of the built-in rule, and then edit and enable your copy, instead of modifying and using the built-in rule.

About this task

The monitor rest time is the amount of time (usually in seconds) Sterling Control Center waits before polling the server to check status and collect statistics. The monitor rest time is used to determine whether monitoring is in or out of compliance.

How can I know if the number of items being executed by a server exceeds a specified threshold?

Sterling Control Center can generate events when queues on monitored Sterling B2B Integrator or Sterling Connect:Direct servers exceed, or return to, thresholds you set.

About this task

For monitored Sterling B2B Integrator servers, Sterling Control Center can generate events for thresholds on the business process execution queues (Q0-Q9). For monitored Sterling Connect:Direct servers, Sterling Control Center can generate events for thresholds on the transmission control queues (EXEC, HOLD, TIMER, and WAIT). The Queue Limits tab of the Server Properties dialog is used to set the thresholds for these queues.

There are two events that may be generated when queues are monitored by Sterling Control Center:

```
CCNS020E Monitored Queue Limit has been exceeded. Server ID: {0} Queue: {1}
Limit: {2} Depth: {3}
CCNS030I Monitored Queue Limit back in compliance. Server ID: {0} Queue: {1}
Limit: {2} Depth: {3}
```

You can create a rule to watch for either of these events or a single linked rule that watches for both. The following example illustrates how to construct a rule that alerts you when a queue threshold exceeds your specified limit for more than 5 minutes.

Procedure

1. Build a rule that monitors for the message that Sterling Control Center generates when a queue limit has been exceeded.
2. Specify the following parameters.

Key	Operator	Value
Message ID	Matches	CCNS020E

You can specify additional parameters to limit when this rule is triggered.

3. Select the **No Operation** action. This action will not generate an alert when the queue threshold is first discovered to have been exceeded. You want to generate an alert only after it has been exceeded for more than 5 minutes.
4. On the Linked Rules wizard, select **Enabled**.
5. Specify the following parameters for the linked rule:

Key	Operator	Value
Message ID	Matches	CCNS030I

This event is generated by the Sterling Control Center engine when it finds that a monitored queue's threshold has returned to the threshold specified. You can also specify other parameters to refine the criteria under which CCNS030I events are matched by the linked part of the rule.

6. Because you selected **No Operation** for the initial action, select **No Operation** for the Resolution action. As a result, no action will be taken if the monitored queue returns to, or below, the specified threshold.
7. Set the timeout value to 5 minutes. This is the amount of time the Sterling Control Center engine must wait before taking either the Resolution or Non-Resolution action.

How can I know when my Sterling Connect:Direct Secure Plus certificates are about to expire?

You can use the built-in Certificate Expiry Warning rule that is included in Sterling Control Center to generate an alert when Sterling Connect:Direct Secure Plus key and trusted certificates are about to expire.

For managed Sterling Connect:Direct servers, for which Sterling Control Center is able to perform configuration management of both Sterling Connect:Direct Secure Plus key and trusted certificates, you can use the Certificate Expiry Warning rule to be alerted when either type of certificate is approaching a user-defined number of days before expiration.

This rule watches for events with one of two message identifiers:

```
CCFG229I Trusted Certificate Expiry notification. Server ID: {0} Certificate: {1}
Expires: {2} Days left: {3} and
CCFG230I Key Certificate Expiry notification. Server ID: {0} Certificate: {1}
Expires: {2} Days left: {3}
```

By default, this rule generates a Sev 1 alert when the number of days left before a certificate expires is less than or equal to 60. You can edit this rule or create your own version of it to watch for only the CCFG229I message or CCFG230I message. In addition, you can edit it to trigger sooner or later, or to perform a different action than the Sev 1 alert, or in addition to the Sev 1 alert.

Attention: This rule depends on either the user, or the system, to initiate daily configuration object synchronizations.

Rules involving Sterling B2B Integrator

Sample rules discuss how to set up rules to monitor Sterling B2B Integrator components. However, they may not describe the detailed steps necessary to set up a rule. Some examples describe only the idea behind a solution to a problem. For a detailed explanation of rules and how to set them up, see *System Administration*.

How can I know when a Sterling B2B Integrator Adapter stops?

You can create a rule so that Sterling Control Center detects when Sterling B2B Integrator stops.

About this task

When Sterling Control Center finds that a Sterling B2B Integrator Adapter on a Sterling B2B Integrator-monitored server transitions from being up to being down, an event of type Server Status with a message ID of CGIS011I is generated. You can write a rule that specifies a parameter with a Key of Message ID, an Operator of Matches, and a Value of CGIS011I. To limit the circumstances under which the rule is triggered, specify values for Server Groups, Servers, as well as other parameters.

How can I know when a Sterling B2B Integrator adapter is enabled?

When Sterling Control Center finds that a Sterling B2B Integrator Adapter on a Sterling B2B Integrator-monitored server has transitioned from being down to up, that is, enabled, an event of type Server Status with a message ID of CGIS010I is generated.

How can I know when a Sterling B2B Integrator adapter goes down and stays down for more than 5 minutes?

When a Sterling B2B Integrator adapter goes from an enabled to a disabled state on a Sterling B2B Integrator-managed server, a Server Status event with a message ID of CGIS011I is generated. When a Sterling B2B Integrator adapter starts up/is enabled, a Server Status event with a message ID of CGIS010I is generated.

About this task

Using this knowledge, you can build a linked rule to alert you only if the adapter goes down and stays down for more than 5 minutes.

Procedure

1. The first part of the rule is to watch for the message ID of CGIS011I. The action taken could be **No Operation**, or a low severity alert action.
2. Select **Enabled** on the Linked Rules window that appears next.
3. As a Linked Rules parameter, specify a **Message ID** value of CGIS010I to match on the event when a Sterling B2B Integrator adapter starts up.
4. Do one of the following:
 - If you selected **No Operation** for the initial action, choose **No Operation for the Resolution** action also, and an appropriate action, such as alert1, for the **Non-Resolution** action.
 - If you selected an alert action, such as alert2, for the initial action, choose **alert0** for the resolution action. This action tells Sterling Control Center to clear the first alert generated, and an appropriate action for the **Non-Resolution** action.
5. Set the timeout value to the time the Sterling Control Center engine waits before taking either the **Resolution** or the **Non-Resolution** action.

Rules involving Sterling File Gateway

These Sterling File Gateway examples discuss how to set up rules to monitor Sterling File Gateway components. However, they do not all describe the detailed steps necessary to set up a rule. Some examples describe only the idea behind a solution to a problem. For a detailed explanation of rules and how to set them up, see *System Administration*.

How to get notified when a Sterling File Gateway route fails?

You can create a rule that generates an alert when a Sterling File Gateway route fails.

About this task

Procedure

Set up a rule to generate an alert, and in the rule specify the following parameter:

Key	Operator	Value
Message Id	Matches	FG_0456

Sterling File Gateway “event codes” are mapped to Sterling Control Center event message IDs. To look up a particular event code, go to the Sterling File Gateway documentation. Event Code FG_0456 is named RouteFailed and is the appropriate value to use for this scenario.

How can a producer know when a file is delivered to a Sterling B2B Integrator mailbox?

You can create a rule to alert a producer when a file is delivered to a Sterling B2B Integrator mailbox.

Procedure

Assuming files are delivered with one of the monitored protocols, set up a rule with the following parameters. Substitute an appropriate value for your mailbox location.

Key	Operator	Value
SI.Mailbox Path	Matches	/Producer/Broadcast
Event Type	Matches	Process Step Ended
SI.Is Put (true or false)	Matches	true

How can a producer know whether a file was delivered to a consumer's mailbox?

You can create a rule to alert a producer if a file is delivered to a consumer’s mailbox.

Procedure

Set up a rule with the following parameters. Substitute an appropriate value for your mailbox location.

Key	Operator	Value
SI.Mailbox Path	Matches	/RQ103PGPConsumerE/ Inbox
SI.Is Put (true or false)	Matches	true
Event Type	Matches	Process Step Ended

How can a producer get notified when a file does not arrive in their mailbox within 5 minutes after the connection is initiated?

You can create a linked rule so that a producer is notified when a file does not arrive in their mailbox within 5 minutes after the connection is initiated.

About this task

To implement this scenario, set up a linked rule.

Procedure

1. Specify the following parameters. Substitute an appropriate value for the submitter in your situation.

Key	Operator	Value
Event Type	Matches	Connection Started
Submitter	Matches	partnerid

2. On the Actions window, select no operation.
3. Next, define the following parameters on the Linked Rules window:
 - a. Check **Enabled** to enable the linked rule
 - b. Select **Event Type matches Process Step Ended** for **Parameters**
 - c. Select **no operation** for **Resolution Action**
 - d. Select **alert1** for **Non-Resolution Action**
 - e. Select **5 minutes** for **Timeout**. This parameter defines the time period to wait before taking the non-resolution action (alert1 which indicates a high alert)

Rules involving Sterling Connect:Enterprise

The Sterling Connect:Enterprise® examples discuss how to set up rules to monitor Sterling Connect:Enterprise components. These examples do not describe the detailed steps necessary to set up a rule. Some examples describe only the idea behind a solution to a problem. For a detailed explanation of rules and how to set them up, see *System Administration*.

How can I know whether a Sterling Connect:Enterprisedaemon goes down and stays down for more than 5 minutes?

When a daemon goes down, a Server Status event with a message ID of CNCE115I is generated. When a daemon starts, a Server Status event with a message ID of CNCE114I is generated. Using this knowledge a linked rule can be built to accomplish the objective of alerting you only if the daemon goes down and stays down for more than 5 minutes.

About this task

Use the following procedure to create a linked rule:

Procedure

1. The first part of the rule is simply to watch for the message ID of CNCE115I. The action taken could be **No Operation**, or a low-severity alert action.
2. Select **Enabled** on the Linked Rules dialog.
3. Specify a **Message ID** value of CNCE114I as a Linked Rules parameter.
4. Take one of the following actions:
 - If you selected **No Operation** for the initial action, choose the same for the **Resolution Action**, and an appropriate action, such as alert1, for the **Non-Resolution Action**.
 - If you selected an alert action, such as alert2, for the initial action, choose alert0 for the **Resolution Action**. This setting tells Sterling Control Center to clear the first alert generated, and an appropriate action for the **Non-Resolution Action**.
5. Set the **Timeout** value to the time the Sterling Control Center engine must wait before taking either the Resolution or the Non-Resolution action.

How can I know when a Sterling Connect:Enterprise for UNIX daemon stops?

Sterling Control Center can find that a master type Daemon on a monitored Sterling Connect:Enterprise for UNIX server changed from running to not running, that is, the daemon stopped. If the daemon stops, Sterling Control Center generates an event of type Server Status with a message ID of CNCE115I.

Procedure

You can simply write a rule that specifies a parameter with a Key of Message ID along with a Value of CNCE115I. To limit when the rule is triggered, you can also specify additional parameters:

Key	Operator	Value
Message Id	Matches	CNCE115I
Daemon Name	Matches	
Daemon Type	Matches	
Daemon SID	Matches	
Daemon Host	Matches	
Daemon Originator	Matches	
Daemon Resource	Matches	
Daemon State	Matches	

How can I know when a Sterling Connect:Enterprise for UNIX daemon starts?

When Sterling Control Center finds that a protocol or other type of daemon on a Sterling Connect:Enterprise for UNIX-monitored server has transitioned from being down to running, it generates an event of type Server Status with a message ID of CNCE114I.

How can I know when a Sterling Connect:Enterprise for UNIX offline add was performed?

About this task

This works only for Sterling Connect:Enterprise for UNIX nodes.

Attention: If you specify only a message ID as a parameter, your rule will be triggered twice per Remote Connection Request operation.

Procedure

Look for a Process End or Process Step End event, with a message ID of CNCE012I.

How can I know when a Sterling Connect:Enterprise for UNIX auto connect fails?

Look for a message ID of CNCE113I.

How can I know when a Sterling Connect:Enterprise remote connect connection is performed?

Look for a message ID of CNCE009I.

How can I know when a Sterling Connect:Enterprise for UNIX auto connect collect operation is performed?

Look for a Process Ended or Process Step Ended event, with a message ID of CNCE013I.

About this task

Note: This currently works only for Sterling Connect:Enterprise for UNIX nodes.

CAUTION:

If you specify only a message ID as a parameter, your rule will be triggered twice per Remote Connection Request operation.

How can I know when a Sterling Connect:Enterprise for UNIX auto connect transmit operation is performed?

Before you begin

Look for a Process Ended or Process Step Ended event, with a message ID of CNCE014I.

Note: This currently works only for Sterling Connect:Enterprise for UNIX nodes.

CAUTION:

If you specify only a message ID as a parameter, your rule will be triggered twice per Remote Connection Request operation.

How can I know when a Sterling Connect:Enterprise for UNIX offline extraction is performed?

Look for a Process Ended or Process Step Ended event, with a message ID of CNCE011I.

About this task

Note: This currently works only for Sterling Connect:Enterprise for UNIX nodes.

CAUTION:

If you specify only a message ID as a parameter, your rule will be triggered twice per Remote Connection Request operation.

How can I know when a Sterling Connect:Enterprise remote connection request operation was performed?

Look for a Process End or Process Step End event, with a message ID of CNCE005I.

About this task

CAUTION:

If you specify only a message ID as a parameter, your rule will be triggered twice per Remote Connection Request operation.

How can I know when a Sterling Connect:Enterprise remote connection add operation was performed?

Look for a Process End or Process Step End event, with a message ID of CNCE003I.

About this task

CAUTION:

If you specify only a message ID as a parameter, your rule will be triggered twice per Remote Connection Request operation.

How can I know when a Sterling Connect:Enterprise remote connect delete is performed?

Look for a message ID of CNCE007I.

How can I know when a Sterling Connect:Enterprise remote connect directory list is performed?

Look for a message ID of CNCE006I.

How can I know when a Sterling Connect:Enterprise remote connect disconnection is performed?

Look for a message ID of CNCE010I.

How can I know when a Sterling Connect:Enterprise for z/OS auto connect CONNECT operation is performed?

Look for a message ID of CNCE209I.

About this task

Note: This currently works only for Sterling Connect:Enterprise for z/OS nodes.

How can I know when a Sterling Connect:Enterprise for z/OS auto connect DISCON operation is performed?

Look for a message ID of CNCE210L.

About this task

Note: This currently works only for Sterling Connect:Enterprise for z/OS® nodes.

How can I know when a Sterling Connect:Enterprise for z/OS auto connect send operation is performed?

Look for a Process Ended or Process Step Ended event, with a message ID of CNCE001I.

About this task

Note: This currently works only for Sterling Connect:Enterprise for z/OS nodes.

CAUTION:

If you specify only a message ID as a parameter, your rule will be triggered twice per Remote Connection Request operation.

How can I know when a Sterling Connect:Enterprise for z/OS auto connect receive operation is performed?

Look for a Process Ended or Process Step Ended event, with a message ID of CNCE002I.

About this task

Note: This currently works only for Sterling Connect:Enterprise for z/OS nodes.

CAUTION:

If you specify only a message ID as a parameter, your rule will be triggered twice per Remote Connection Request operation.

QuickFile rules

The QuickFile examples discuss how to set up rules to monitor QuickFile transfers. These examples do not describe the detailed steps necessary to set up a rule. Some examples describe only the idea behind a solution to a problem. For a detailed explanation of rules and how to set them up, see *System Administration*.

How can I know if a request for a file has been satisfied in time?

To be alerted when a request for a file from a user is not satisfied within a specified amount of time, you should use a linked rule, as opposed to an SLC.

When QuickFile is used to request files from a user, a single message is generated by QuickFile. Sterling Control Center creates a Server Status event from that message, which can be used to trigger the start of a linked rule. When the user

responds to the file request, the events associated with a file upload are generated. The Sterling Control Center Process Step Ended event from those events can be used to satisfy the linked rule.

For example, you can use a linked rule to generate an alert if a user's QuickFile file request is not satisfied in 60 minutes. To define a linked rule for this QuickFile scenario:

1. Create the regular portion of the linked rule with the following parameters:

Key	Operator	Value
Event Type	Matches	Server Status
QuickFile Event Type	Matches	fileEvent
QuickFile Operation	Matches	Requested
QuickFile User Email Addr	Matches	requestor@email.com (requesting user's email address)
QuickFile Recipient Email Addr	Matches	requestee@otheremail.com (requested user's email address)

2. On the Actions window, select **no operation** so that when a request event occurs, the linked rule timer starts.
3. On the Linked Rules window, specify the following parameters for the linked rule:

Field	Value
Enabled	Yes
Resolution Action	no operation
Non-Resolution Action	alert2
Timeout	60 minutes

Because no alert is required, if the requestee sends the file within 60 minutes, the resolution action is set to no operation. The non-resolution action is set to alert2 to generate a severity 2 alert if the requestee does not respond within 60 seconds.

4. Specify the following parameters for the linked rule:

Key	Operator	Value
Event Type	Matches	Process Step Ended
QuickFile Event Type	Matches	fileEvent
QuickFile Operation	Matches	Sent
QuickFile User Email Addr	Matches	requestee@otheremail.com (requested user's email address)
QuickFile Recipient Email Addr	Matches	requestor@email.com (requesting user's email address)

How can I know if an uploaded file is downloaded by a recipient within a certain amount of time?

To be alerted when files uploaded using QuickFile are not downloaded by a recipient within a certain amount of time, use a workflow SLC.

About this task

To address this scenario, associate a duration SLC schedule with the workflow SLC with a minimum duration of 00:00:00 and an appropriate maximum duration, which should be the expected time between the upload of a file and the download by the recipient. Within the workflow SLC, you define one milestone for the upload and one milestone for the download, and specify stepName as the correlator value.

Tip: For workflow SLCs that contain a correlator variable, you can include the runtime value the correlator was set to in any resulting alerts, traps, or emails. For traps to include the correlator value, you need to update the contents of the SnmpAdaptorWrapper.xml file by setting the slcCorrelatorValue element to a nonzero value. For email actions to include the correlator value, simply include the slcCorrelatorValue variable in the email subject or message. In addition, the slcCorrelatorValue variable can be used in both OS Command and Server Command actions.

For example, you can use a workflow SLC to make sure that whenever a file is uploaded by QuickFile for a recipient, it is downloaded by that recipient in 5 minutes. If it is not downloaded within 5 minutes, you can be alerted.

Note: This example depends on the monitor rest time for the QuickFile server being set to no more than one minute. Longer monitor rest times could result in false alerts being generated because of the choices made for the duration values in this example. Normally, you would not put such tight time restrictions on your SLC. It was done here simply for demonstration purposes.

Procedure

1. Create a duration SLC schedule that will be referenced by the workflow SLC with the following values:

Parameter	Value
Name	Max 5 minutes
Minimum Duration	000:00:00
Maximum Duration	000:05:00

2. Create the workflow SLC with the following values:

Parameter	Value
Name	Ensure downloads done in 5 minutes
Enabled	Selected
Generate notification if event has not occurred	Selected

Parameter	Value
Suppress Milestone Messages	Selected When this parameter is enabled, events are generated only for the workflow, not for its milestones.
Monitor Window Tolerances: Start	1 hour
Monitor Window Tolerances: End	1 hour An event is generated if downloads are not completed within 1 hour and 5 minutes of the associated upload. The event will state that the workflow did not complete within the monitoring window.

3. Select the Max 5 minutes SLC schedule you created.
4. Specify the following parameters for the workflow SLC:

Parameter	Value
Milestone time Relative to: Actual start of Workflow SLC	Selected
Concurrency Count	1 Set to the maximum number of uploads and downloads expected per hour.
Correlator Source	stepName For this example, stepName is used as the correlator because for uploads, the stepName consists of the QuickFile package identifier value, recipient user email address, and QuickFile file name. For downloads, the stepName consists of the QuickFile package identifier, user Email address, and QuickFile file name. As a result, the Sterling Control Center event stepName element has the same value for a file uploaded for a specific user as it does when that same file is downloaded by that user.

5. Add a milestone with the following parameters that will monitor for an upload:

Key	Operator	Value
QuickFile File Name	Wild Card	*
QuickFile Event Type	Wild Card	fileEvent
QuickFile Operation	Wild Card	Sent

Key	Operator	Value
Server	Wild Card	QuickFile* Provide a value such as the name of your QuickFile server, or something that matches the QuickFile servers that you want the workflow SLC to apply to. In this example, any QuickFile servers that contain "QuickFile" in their server ID will be monitored for uploads.

6. Set the schedule for the milestone with the following parameters:

Parameter	Value
Name	Upload
Minimum Duration(Dmin)	000:00:00
Maximum Duration(Dmax)	000:01:00

7. Create the download milestone.

Tip: To create the download milestone, you can duplicate the Upload milestone, name it Download, and change the QuickFile Operation to Downloaded.

What to do next

Sterling Control Center comes with the following built-in rules that watch for corresponding SLC events that potentially could be generated as a result of this workflow SLC:

- Proc duration short – CSLC041E – alert2
- Proc duration okay – CSLC042I – alert0
- Proc duration not ended – CSLC034E – alert2
- Proc duration exceeded – CSLC044E – alert1
- Proc duration way over – CSLC045E – alert1

You may want to add your own rules and actions that are similar to the existing, built-in rules, which reference these message IDs. You can use a built-in rule to create your own rule by duplicating the rule, giving it a new name, specifying additional parameters, such as the name of the workflow SLC, and specifying your own actions.

Chapter 5. Troubleshooting rules

Rules can fail to work as expected for many reasons. Working is defined here as being triggered by the conditions under which a rule is expected to be triggered.

Here are some examples of why a rule might not be triggered as expected:

- A rule will not be triggered if the wrong Event Type is selected even though all of the other parameters are correct.
- A rule will not be triggered if no events matching its parameters have been generated.
- A rule will not be triggered even if an event matches its parameters if the event occurred outside of the schedule associated with the rule.
- A rule will not be triggered if the matching event has already triggered a rule that appears higher in the list.
- A rule must be enabled in order to be triggered.
- Back-level versions of managed servers, that is, managed servers not on current maintenance, may not generate statistics records, or statistics records with the needed information, for the events constructed by Sterling Control Center to have the information needed to trigger the rule.
- Process events (Process Started, Process Step Started, Process Step Ended, and Process Ended) are all generated at the same time by Sterling Control Center for Sterling Connect:Enterprise for z/OS, Sterling Connect:Enterprise for UNIX, QuickFile, and most native FTP servers because Sterling Control Center only “sees” the file transfer ending event. Be aware of this when writing linked rules for process events for these types of servers.

Troubleshooting by using the Database Events report

One method you can use to determine the parameter that can be defined in a rule is to cause the condition that you want to trigger the rule on a managed server. After Sterling Control Center captures the data generated by the condition, and written an event to its events database table, use the Sterling Control Center Database Events report to capture the exact contents of the events in order to examine them.

Procedure

1. First, define a new report. In the create report wizard, after selecting a Report Type of **DB Events Report**, click **Next**.
2. For the DB Events Report Filter parameters, specify as much information as you can to limit the amount of data returned.
3. At a minimum, specify a starting and an ending Date/Time for the report as illustrated in the following table:

Key	Operator	Value
Date Time	Greater than	Today 0:00 America/Chicago

The following criteria causes all events generated between 15:00 and 15:15 to be retrieved. You can specify additional filter keys as you see fit to limit the size of the report.

Key	Operator	Value
Date Time	Less than	Today 15:15:00 America/Chicago
Date Time	Greater than	Today 15:00:00 America/Chicago

4. Click **Next** when you finish specifying filter criteria. The Available Columns wizard is displayed.
5. Select **Date Time**, **Event Type**, and **XML_STRING** columns and click **Next**. The Sort wizard is displayed.
6. Select **Date Time** and click **Next**, and then run the report.

The report generated contains the events generated during the specified time frame that also match any additional filter criteria specified. The XML_STRING value shows the contents of the actual Sterling Control Center event. From these values, you are able to determine the parameters necessary for your rule to be triggered as expected.

For example, say that you wanted to write a rule to be triggered when a Sterling Connect:Direct server stops.
7. First, while Sterling Control Center is monitoring a server, stop the server.
8. Second, create a Database Events Report over the time frame the server was stopped.
9. Locate the event generated by the server down condition in the report.

From this event, you can see that when a server goes down, the message ID is CCTR034E.
10. You can use this information to write a server down rule. In creating the server down rule, specify a parameter with a Key of Message ID, an Operator of Matches, and Value of CCTR034E.

Chapter 6. Setting up schedules

How can I build a schedule to prevent rules from triggering during my maintenance windows?

To prevent rules from triggering during your maintenance window, you must create a server down rule with no associated schedule.

About this task

Next, create a rule schedule for the maintenance window. Duplicate the server down rule, specify the schedule created for the maintenance window, and change the action to No Operation. Then, give the second server down rule a higher priority than the first server down rule.

You can set up a schedule to deal with a problem common to all rules: triggering during maintenance periods when servers are expected to be down. For example, if you have weekly maintenance windows on Sundays from 1 p.m. to 4 p.m., you may want to know when servers go down, except during these maintenance windows.

Procedure

1. Create a Server Down rule, as described in *How can I know when a monitored server goes down?*, with no associated schedule.
2. Create a rule schedule that specifies all Sundays from 1 p.m. to 4 p.m. This task is easy to do, because a calendar named Sunday ships with Sterling Control Center which specifies all Sundays in a year. The following table illustrates sample criterion of a rule schedule. Be sure to enable **Check Schedule against when event occurred**

Calendar Name	Time Zone	Start time	End Day	End Time
Sunday	(UTC-06:00) Central Time (US & Canada)	13:00	Calendar Start Day	16:00

3. Duplicate the server down rule you created before and name it Server Down 1 p.m. to 4 p.m.
4. When prompted for a schedule for this new rule, specify the "Sundays 1 p.m. to 4 p.m." rule schedule you previously created.
5. Change the action to **No Operation**.
6. Make this second server down rule a higher priority rule than the first server down rule that has no calendar associated with it.

Results

Now, when a server goes down, the Sterling Control Center engine first sees your new server down rule. If it happens to be Sunday between 1 p.m. to 4 p.m., this rule is triggered and performs no operation. If it is not Sunday between 1 p.m. to 4 p.m., this rule is not triggered. Instead, the old server down rule, the one with no associated schedule, is triggered. Whatever action you specified for this second server down rule is performed.

By taking advantage of the fact that only one Sterling Control Center rule is triggered per event, and by putting first the server down rule that takes no action, you accomplish your goal of knowing when a server down event occurs outside of the maintenance window.

Chapter 7. Reports example

How do I set up an automated report of alerts for the previous day?

You can use the Automated Reports facility of Sterling Control Center to report things such as the alerts from a previous day.

About this task

The basic steps to setting up an automated report that lists all the alerts for the previous day are:

- Define an Alerts report and save it.
- Define a report schedule and save it.
- Automate the Alerts report.

Here are the specific steps:

Procedure

1. Click **Tools > Reports > Define/Run**.
2. Add a report, select **System** as the **Report Category** and **Alerts Report** as the **Report Type**.
3. Select the filter criteria as illustrated in the following table. This criteria narrows the alerts to yesterday's only. This step is the key step in this process.

Key	Operator	Value
Alert Date/Time	Less than	Today 00:00:00 America/Chicago
Alert Date/Time	Greater than	Yesterday 00:00:00 America/Chicago

4. Run the report and save it.
5. Define a report schedule by which to run the report (**Tools > Reports > Define Schedule**); for example: Daily at 6:00 a.m. Specify the schedule name, choose a calendar (for this example, Daily with a time of 6:00 a.m.), and save the schedule.
6. Add a new automated report (**Tools > Reports > Automate**).
7. Give the new automated report a name and select the report you defined in step 1 through step 4.
8. Give the automated report the schedule you defined in step 5.
9. Select an email list of recipients to send the mail to.
10. Select other parameters, including the report file format, the file extension, a From address and an email Subject line, and then save the report. The report is generated daily at the time you specified and sent to the list of addressees you specified.

Chapter 8. Setting up SLCs

This information offers information to help you understand service-level criteria (SLCs) and how they are implemented. It describes the differences between rules and SLCs, gives tips for deciding when to use one versus the other, and gives examples of common scenarios in which you might want to set up an SLC.

Not all of the examples describe the detailed steps necessary to set up an SLC. Some describe only the idea behind a solution to a problem. For a detailed explanation of SLCs and how to set them up, see *System Administration*.

What is the correlation between SLCs and rules?

SLCs generate SLC type events when certain conditions occur or fail to occur because Processes either execute or fail to execute.

Processes can include the following items:

- Sterling Connect:Direct Processes
- Sterling B2B Integrator business process activities
- FTP gets and puts
- The movement of data into and out of Sterling Connect:Enterprise mailboxes
- Sterling File Gateway Arrived Files, Routes, and Deliveries

The conditions under which SLCs generate events are determined by the SLCs you have defined and enabled.

However, actions (such as generating an alert or sending an email) are performed when rules are triggered. Rules are triggered by events that Sterling Control Center generates either for external reasons (data is collected from managed servers) or for internal reasons (an SLC triggers).

When should an SLC be used versus a rule?

You do not use SLCs to the exclusion of rules or vice versa. In fact, without rules SLCs cannot alert you to anything.

SLCs cannot not generate alerts. An SLC may cause an SLC type event to be generated. That event may trigger a rule, which in turn may have an associated action that generates an alert. The question is: "Which do I need to solve my problem?"

Note: By default, Sterling Control Center comes with a number of predefined, enabled rules. Alerts may be generated simply by defining and enabling SLCs, but only because of these predefined rules. Once you understand what the predefined rules do, you may decide to disable some. Disable rather than delete them as they are not easily restored.

When do I need SLCs?

Always use an SLC when you want to know that something has not happened.

Rules are triggered only by events. SLCs are able to generate events when a specified condition, such as that a process must start by 10:00 a.m. does or does not occur.

An SLC can tell you when certain things failed to occur, such as:

- Processes or process steps that are supposed to start by a specific time
- Processes or process steps that are supposed to end by a specific time
- Processes that are supposed to run for a specified duration

What events may be generated by SLCs and when?

All SLC-related events are of Event Type SLC. They all contain a message ID that indicates why they were generated and the SLC name (groupID), as well as process information such as process name and number when possible.

SLC notifications for calendar schedule-based SLCs

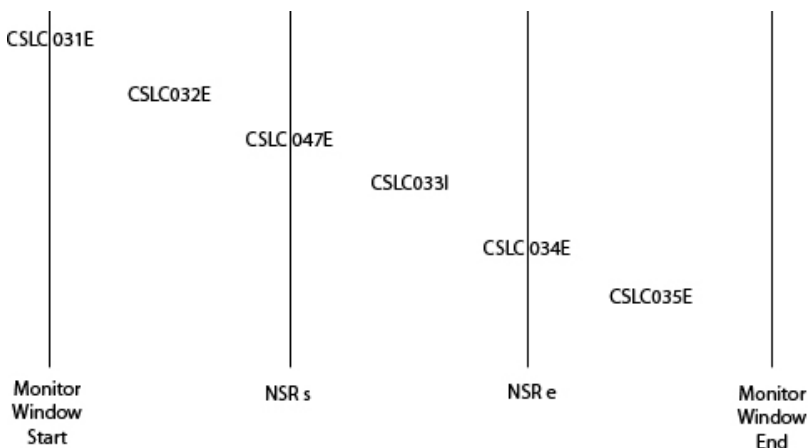
You use a calendar schedule in an SLC when the exact time an occurrence is expected to start or finish is important. Calendar schedules let you specify the normal start range and normal end range for an occurrence.

In the following charts:

- NSRs means Normal Start Range start
- NSRe means Normal Start Range end
- NERs means Normal End Range start
- NERe means Normal End Range end

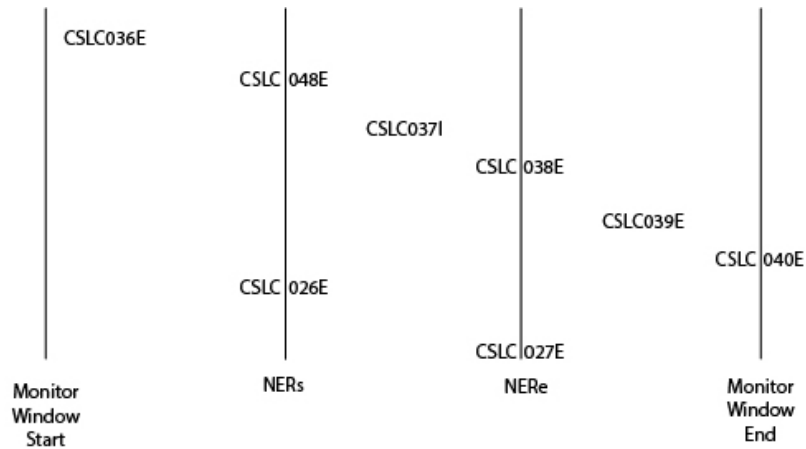
SLC message IDs associated with start range

The following message IDs are relevant to the start range of an SLC associated with a calendar schedule.



SLC message IDs associated with end range

The following message IDs are relevant to the end range of an SLC associated with a calendar schedule.



Message IDs for calendar schedule-based SLCs

The following message IDs are relevant to an SLC based on a calendar schedule:

Message ID
Message

CSLC026E

Did not start by NERs.

CSLC027E

Did not start by NERe.

CSLC031E

Running prior to start of monitoring

Note: Not relevant for workflow SLCs.

CSLC032E

Started prior to NSRs

CSLC033I

Started.

CSLC034E

Did not start by NSRe.

CSLC035E

Started after NSRe.

CSLC036E

Ended before NERs.

CSLC037I

Ended.

CSLC038E

Did not end by NERe.

CSLC039E

Ended after NERe.

CSLC040E

Did not complete before the end of the monitoring window.

CSLC047E

Did not start by NSRs.

CSLC048E

Running after NERs.

Note: Message IDs CSLC026E and CLSC027E are not generated for SLCs with calendar schedules that include an NSR specification.

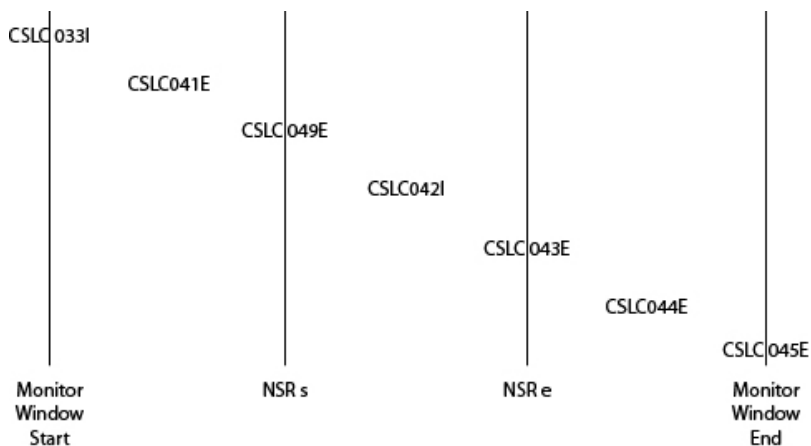
SLC notifications for duration schedule-based SLCs

Certain message IDs can come into play for SLCs based on a duration schedule.

In the following chart:

- dMin means minimum duration
- dMax means maximum duration

The chart shows when a message ID can come into play for an SLC based on a duration schedule.



Message IDs for duration schedule-based SLCs

These message IDs pertain to schedule-based SLCs.

Message ID Message

CSLC033I

Started.

CSLC041E

Ended before dMin.

CSLC042I

Ended when expected - between dMin and dMax.

CSLC043E

Did not end by dMax.

CSLC044E

Ended after dMax.

CSLC045E

Did not complete before the end of the monitoring window.

CSLC049E

Did not end by dMin.

CSLC050E

Duration could not be determined - was running before monitoring started.

Other Message IDs for both calendar and duration schedule-based SLCs

These message IDs pertain to both duration schedule-based SLCs and calendar schedule-based SLCs.

Message ID

Message

CSLC228E

Did not start before the end of the monitoring window.

CSLC229E

On-time completion might be in jeopardy.

CSLC171I

Step ahead of schedule (25% dMin).

CSLC172E

Step behind schedule (25% dMin).

CSLC173I

Step ahead of schedule (50% dMin).

CSLC174E

Step behind schedule (50% dMin).

CSLC175I

Step ahead of schedule (75% dMin).

CSLC176E

Step behind schedule (75% dMin).

CSLC177I

Step ahead of schedule (25% dMax).

CSLC178E

Step behind schedule (25% dMax).

CSLC179I

Step ahead of schedule (50% dMax).

CSLC180E

Step behind schedule (50% dMax).

CSLC181I

Step ahead of schedule (75% dMax).

CSLC182E

Step behind schedule (75% dMax).

How do I know which type of SLC to use?

There are three types of SLCs in Sterling Control Center:

- Standard SLCs
- Wildcard SLCs
- Workflow SLCs

If you are interested in knowing when a single Process or Process step runs or fails to run, use either a standard SLC or wildcard SLC. With wildcard SLCs you specify a generic name (using a wildcard character) rather than a specific, individual one. So, if you know the specific Process name, server name, etc., use a standard SLC.

Note: You can use a wildcard SLC in place of a standard SLC, but the Sterling Control Center engine processes standard SLCs more efficiently. Use standard SLCs when you have the choice.

If you want to ensure that multiple Processes and/or Process steps run, use a workflow SLC.

How can I be alerted if one process runs too long?

You can define an SLC duration schedule to be alerted if one process runs too long.

About this task

Since in this situation you are interested only in a single process, either a standard or wildcard SLC would be appropriate to use. If you know the specific process and server name, use a standard SLC because it is more efficient than a wildcard SLC.

You are interested only in knowing whether the process runs for the correct amount of time, not whether it runs during a specific time period. Therefore, you can define an SLC duration schedule. In this example:

Procedure

1. On the General pane, set **Schedule Type** to **Duration Schedule**, set **Name** to **Duration**, and select **Enabled**.
2. On the Parameters pane, specify a **Maximum Duration** value that corresponds to the maximum process duration.
3. Specify a **Minimum Duration** value that corresponds to the minimum Process duration.

In this example, Minimum Duration is set to 000:00:00 and Maximum Duration is set to 000:10:00. In other words, the duration schedule specifies that 0 seconds to 10 minutes elapsed time is acceptable for the SLC you associate the schedule with.

Note: Choose the minimum duration wisely for your SLC. Remember that the minimum granularity for measuring the elapsed time of processes and process steps in Sterling Control Center is seconds. So, if a process or process step runs in less than a second, it can appear to Sterling Control Center to run in 0 seconds.

Sterling Control Center generates events if a process or process step takes less than the specified minimum duration. Specifically, an SLC event with message ID CSLC041E is generated. Remember, processes or process steps that take less time to run than expected can signal a problem as serious as processes or process steps that take more than the maximum expected duration. For example, if you are expecting to receive a 100 GB file and the file takes less than a second to be received, there is a good chance you did not receive the expected data.

4. Reference the duration schedule you defined when constructing your SLC. In this example, on the Schedules pane, set Selected Schedules to Duration.
5. Beyond specifying the duration schedule for your standard or wildcard SLC, specify appropriate values for server groups, servers, and other SLC match criteria to define the process, processes, process step, or process steps you are interested in knowing about.

How can I be alerted if a critical process fails to complete on time?

Use a standard or wildcard SLC and specify appropriate normal end range start (NERs) and normal end range end (NERe) values in the calendar SLC schedule associated with the SLC.

How can I be alerted when related Sterling Connect:Enterprise batches arrive?

To be alerted when related batches arrive in a Sterling Connect:Enterprise mailbox, use a workflow SLC.

Procedure

1. Associate a duration SLC schedule with the workflow SLC with a minimum duration of 00:00:00 and an appropriate maximum duration. This duration might be the expected time between the arrival of the first and last of the related batches.
2. Within the workflow SLC, define one milestone per batch to be received. The milestones you define must specify parameters that uniquely identify each batch.
3. Each milestone must specify values for duration. Specify minimum and maximum values that match the values in the duration schedule.

Example

In the following example, four batches, named File1, File2, File3, and File4 are expected to be received by a managed Sterling Connect:Enterprise server named

kukla35 within 5 minutes of one another. This data is to be sent by a user named Raj. You want to know when they all arrived, or that they did not arrive within the allotted time.

1. Create a duration SLC schedule to be referenced by the workflow SLC.
Because we expect all four batches to arrive within 5 minutes of one another, this duration schedule specifies a dMin of 0 hours, 0 minutes, and 0 seconds, and a dMax of 0 hours, 5 minutes, and 0 seconds.
2. Create the Workflow SLC.
On the General pane, the **Enabled** and **Generate Notification if event has not occurred** check boxes are selected.
The checkbox for **Suppress Milestone Messages** is selected. This checkbox is selected because we want events generated only for the workflow, not for its milestones.
The **Start Monitor Window Tolerance** is set to 1 hour. Also, the **End Monitor Window Tolerance** is set to 1 hour. This causes an event to be generated if all batches are not received within 1 hour and 5 minutes after the first batch arrives. The event states that the workflow did not complete within the monitoring window.
3. Select the duration schedule created in a previous step.
4. The concurrence count must be set to the number of related groups of batches expected to arrive per hour. We expect at most one per hour, so set concurrence count to 1.
5. Add the first milestone to the workflow SLC, the one corresponding to the first batch.
6. Here are the parameters for the milestone, named File1:

Key	Operator	Value
Server	Wild Card	kukla35
Process name	Wild Card	File1
Submitter	Wild Card	Raj

The server the batches are to arrive on is specified, as is the name of the batch and the sender of the data. Remember for Sterling Connect:Enterprise, Process name equates to batch ID, and submitter equates to the origin of the data.

7. Set the schedule for the milestone.
The minimum and maximum durations correspond to the durations for the workflow SLC schedule. In this example, the **Minimum Duration (Dmin)** is set to 000:00:00 and the **Maximum Duration (Dmax)** is set to 000:05:00.
8. Duplicate this milestone three times by using the **Duplicate button**. Duplicate the milestone once for each additional related batch, setting the process name to the new related batch name. The following table illustrates the Milestones pane settings for this example:

#	Name	Dmin	Dmax
1	File1	000:00:00	000:05:00
2	File2	000:00:00	000:05:00
3	File3	000:00:00	000:05:00
4	File4	000:00:00	000:05:00

Sterling Control Center comes with built-in rules that watch for corresponding SLC events that potentially could be generated as a result of this workflow SLC. They are:

- Proc duration short - CSLC041E - alert2
- Proc duration okay - CSLC042I - alert0
- Proc duration not ended - CSLC043E - alert2
- Proc duration exceeded - CSLC044E - alert1
- Proc duration way over - CSLC045E - alert1

You might want to add your own rules and actions that are similar to the existing, built-in rules, which reference these message IDs. Do so by duplicating them, giving them new names, and specifying additional parameters, such as the name of the workflow SLC, and specifying your own actions.

For example, the Parameters pane could contain the following criteria:

Key	Operator	Value
Message Id	Matches	CSLC041E
SLC Name	Matches	Watch for 4 batches

Be sure, after constructing the new rules, to assign them a higher priority than existing rules that specify the same SLC event message ID. More specific rules should always be higher in priority than their more general counterparts.

That is all that needs be done. This workflow SLC is triggered whenever a batch arrives with an ID of File1, File2, File3, or File4 that was sent by Raj to kukla35. An SLC event will be generated when the last of the four batches arrives, or after 5 minutes, and after an hour and 5 minutes if they do not all arrive.

How can I be alerted if a related group of processes runs too long?

To monitor whether a group of Processes runs too long, construct an SLC duration schedule and reference that schedule in a workflow SLC group.

About this task

The workflow SLC group contains appropriate values for server groups, servers, and other SLC match criteria. The match criteria specifies the Process, Processes, Process step, or Process steps you are interested in. For example, say that PROCESSA runs on SERVERA, PROCESSB runs on SERVERB, and PROCESSC runs on SERVERC, and that all must run within 10 minutes.

Procedure

1. Since you are interested in knowing that multiple related Processes run within a specified amount of time, use a workflow SLC. Associate a duration SLC schedule with it. For this example, the duration SLC schedule has the following attributes:

Name	Schedule Type	Enabled	Minimum Duration	Maximum Duration
Duration	Duration Schedule	Yes	000:00:00	000:10:00

- The workflow SLC that references this duration SLC schedule is defined as follows:

Name	Enabled	Generate notification if event has not occurred	Monitor Window Tolerances Start	Monitor Window Tolerances End
Watch for Processes A, B, and C	Yes	Yes	6	6

Monitor Window Tolerances: Start and End specify when you want an SLC to begin and end monitoring. In this instance, because a duration SLC schedule is being associated with the workflow SLC, Start Monitor Window Tolerance is immaterial. The reason is that we do not care what time the Processes begin, only that they all run within a specified duration.

The End Monitor Window Tolerance tells Sterling Control Center how long to continue watching for the end of the workflow beyond the schedule associated with the workflow SLC. You might make the value for both the Start and End Monitor Window Tolerances as short as zero hours.

- Specify the duration SLC schedule previously defined. The schedule specifies a minimum duration of 0 seconds and a maximum duration of 10 minutes.
- On the Parameters pane, the value you select for Milestone Times Relative To, Actual Start or Scheduled Start, only matters if NSR and NER or both values are specified for the SLCs milestones.

Concurrency Count is set to 1 because we expect only one occurrence of this workflow to be executing within the monitoring window specified.

For SLCs with SLC calendar schedules, Concurrency Count tells Sterling Control Center how many occurrences of the workflow are expected to run during the scheduled time. If "Generate notification if event has not occurred" is selected for an SLC with a calendar schedule, then SLC events are generated for each instance that does not run as expected. For SLCs with SLC duration schedules, Concurrency Count tells Sterling Control Center how many occurrences of the workflow might run simultaneously.

- On the Milestones window, specify three milestones, one for each Process (PROCESSA, PROCESSB, and PROCESSC).
- Define parameters that specify PROCESSA on SERVERA for the first milestone, which we call PROCA, as illustrated in the following table:

Key	Operator	Value
Server	Wild card	SERVERA
Process Name	Wild Card	PROCESSA

- On the Schedule window, specify a minimum and maximum duration of 000:00:00 to 000:10:00 (0 minutes to 10 minutes) for milestone PROCA.
- Additional milestones, similar to the first, must be defined for PROCESSB on SERVERB, and PROCESSC on SERVERC.

Click the **Duplicate button** to copy the PROCA milestone makes it easier to create milestones PROCB and PROCC.

The following table illustrates the milestones criteria:

#	Name	Dmin	Dmax
1	PROCA	000:00:00	000:10:00

#	Name	Dmin	Dmax
2	PROCB	000:00:00	000:10:00
3	PROCC	000:00:00	000:10:00

How can I be alerted if a related group of critical processes do not all complete on time?

You can be alerted if a related group of critical processes do not all complete on time.

About this task

Use a workflow SLC. Associate a calendar SLC schedule with the workflow with appropriate times for normal end range start (NERs) and normal end range end (NERe).

Procedure

1. Within the workflow SLC, define one milestone per critical process.
The milestones you define must specify parameters to uniquely identify each process.
2. Each milestone must specify values for duration, and or values for NSRs and NSRe, and or values for NERs and NERe. The choices you make here affect only the additional milestone SLC events that might be generated.
3. If you know approximately how long each Process in the group must last, specify dMin and dMax values for each milestone. Milestone SLC events are generated if the Processes run shorter than the dMin value specified, or if they run longer than the dMax value specified.
4. The NSR and NER values are all relative to when the workflow begins. If you specify **Milestone Times Relative to Actual Start of Workflow SLC**, they are determined by the start of any one of the Processes corresponding to the milestones defined. If you specify **Milestone Times Relative to Scheduled Start of Workflow SLC**, NSR and NER values are relative to the scheduled start of the workflow. You might want to specify NSR values for one or more of the milestones if you know that other Processes within the workflow must begin within a specified amount of time of the workflow itself starting.

How can I be alerted if a critical process does not start on time?

Use a standard or wildcard SLC. Specify NSRs and NSRe values in the calendar SLC schedule associated with the SLC.

How can I be alerted if one or more related processes do not start or complete on time?

You can be alerted if one or more related processes do not start or complete on time.

About this task

Use a workflow SLC with one milestone for each critical process. Associate a calendar SLC schedule.

Procedure

1. Use a workflow SLC.
2. Associate a calendar SLC schedule with the workflow with appropriate times for NSRs, NSRe, NERs, and NERe.
3. Within the workflow SLC, define one milestone per critical Process. The milestones you define must specify parameters to uniquely identify each Process.
4. Each milestone must specify values for duration, and or values for NSRs and NSRe, and or values for NERs and NERe. These choices affect only the additional milestone SLC events that might be generated.
5. If you know approximately how long each Process in the group lasts, specify dMin and dMax values for each milestone. Milestone SLC events are generated if the Processes run shorter than the dMin value specified, or if they run longer than the dMax value specified.
6. If you select **Milestone Times Relative to Actual Start of Workflow SLC** when setting up the workflow, the NSR and NER values for milestones are relative to when the workflow begins. The workflow begins when any of the Processes corresponding to the milestones start of the workflow.
7. If you select **Milestone Times Relative to Scheduled Start of Workflow SLC** when setting up the workflow, the NSR and NER values specified for milestones are relative to the scheduled start time of the workflow (its NSRs value in the schedule assigned to it).
8. If you know that other Processes within the workflow begin within a specified amount of time of the workflow itself starting, you might want to specify NSR values for one or more of the milestones.

How can I be alerted if one or more related processes might not complete on time?

You can use a Jeopardy List to get an alert if one or more related processes do not complete on time.

About this task

Workflow SLCs have a parameter called Jeopardy List.

A jeopardy list is simply a list of message IDs that might be associated with SLC events.

For the lists of message identifiers that might be associated with SLC events, see *SLC notifications for calendar schedule-based SLCs* and *SLC notifications for duration schedule-based SLCs*.

Procedure

1. To create a jeopardy list, click the Create button next to the Jeopardy List drop-down list:
2. Specify a **Name** and **Description**, then press **Next**:
3. On the subsequent wizard screen, check the message IDs you want to use as an indication that the workflow might not complete on time. If any SLC events are generated for the workflow that contain a message ID specified in the jeopardy list, then an additional SLC event with a message ID of CSLC229I (on time completion in jeopardy) is generated for every workflow milestone that has yet to start.

Note: Right-click the message list and choose Suggest Jeopardy List to get a suggested list of messages.

How can I reduce the number of events with the same message generated by workflow SLCs?

If one milestone in a workflow SLC does not start on time, you might not care to be notified that every subsequent milestone is also going to be late starting.

About this task

To suppress these subsequent SLC events, use the Fire-Once List parameter when setting up your workflow SLC. To create a fire-once list:

Procedure

1. Click the Create button by the Fire-Once List.
2. Specify a **Name** and **Description**, then press **Next**. The wizard panel is displayed.
3. Select the message IDs for which you want SLC events to be generated only once per instance of the workflow.

Note: To get a suggested list of fire-once messages, right-click the messages displayed and choose Suggest Fire-Once list.

How can I be alerted if a Sterling Connect:Direct file transfer may not complete on time?

Before you begin

When Sterling Control Center can ascertain the size of a file being transmitted, and you have specified a duration SLC schedule for the transfer, SLC events with the following message IDs may be generated.

Message ID

Message Text

CSLC171I

Step ahead of schedule (25% dMin)

CSLC172E

Step behind schedule (25% dMin)

CSLC173I

Step ahead of schedule (50% dMin)

CSLC174E

Step behind schedule (50% dMin)

CSLC175I

Step ahead of schedule (75% dMin)

CSLC176E

Step behind schedule (75% dMin)

CSLC177I

Step ahead of schedule (25% dMax)

CSLC178E

Step behind schedule (25% dMax)

CSLC179I

Step ahead of schedule (50% dMax)

CSLC180E

Step behind schedule (50% dMax)

CSLC181I

Step ahead of schedule (75% dMax)

CSLC182E

Step behind schedule (75% dMax)

Sterling Control Center comes with predefined rules configured to watch for these events.

Examine the rules in the Rules Listing to determine whether these rules fit your requirements.

How can I be alerted if an hourly process is not on time?

To be alerted if an hourly Process is not on time, create 24 calendar SLC schedules. Assign the 24 schedules to one standard SLC.

About this task

Say that you have a Process named HOURLY that is supposed to run every hour on Server REGULAR. How can you be alerted when it fails to run?

The easiest way to handle this requirement is to create 24 different calendar SLC schedules, one for each hour of the day, and then assign all 24 of them to one standard SLC that specifies appropriate matching criteria (a Server name of REGULAR and a Process name of HOURLY) for its parameters.

Procedure

1. Select **Manage > Service Level Criteria (SLCs) > SLC Schedules** and click the **Create** button to create a schedule.
2. Create a schedule with NERs between midnight and 1:00 a.m. The following table illustrates sample settings of an SLC schedule:

Schedule Type	Name	Description	Enabled
Calendar Schedule	1AM	NERs between midnight and 1AM	Yes

This schedule is for the first hour of the day.

3. As our process is supposed to run 24 hours a day, 7 days a week, set **Calendar Name** to the predefined calendar called Daily.

4. It is not necessary to set a start range. Simply setting an end range that encompasses the hour within which the process is supposed to run is sufficient. The following table illustrates sample settings for the SCL Schedule Parameters:

Calendar Name	Time Zone	End Day	Start Time	End Time
Daily	(UTC - 06:00) Central Time (US & Canada)	Calendar Start Day	00:00	01:00

5. Duplicate this SLC calendar schedule 23 times, modifying the name and NER values appropriately.

SLC Calendar Schedule Name	NERs	NERe
Midnight to 1AM	00:00	01:00
1AM to 2AM	01:00	02:00
2AM to 3AM	02:00	03:00
3AM to 4AM	03:00	04:00
4AM to 5AM	04:00	05:00
5AM to 6AM	05:00	06:00
6AM to 7AM	06:00	07:00
7AM to 8AM	07:00	08:00
8AM to 9AM	08:00	09:00
9AM to 10AM	09:00	10:00
10AM to 11AM	10:00	11:00
11AM to 12PM	11:00	12:00
12PM to 1PM	12:00	13:00
1PM to 2PM	13:00	14:00
2PM to 3PM	14:00	15:00
3PM to 4PM	15:00	16:00
4PM to 5PM	16:00	17:00
5PM to 6PM	17:00	18:00
6PM to 7PM	18:00	19:00
7PM to 8PM	19:00	20:00
8PM to 9PM	20:00	21:00
9PM to 10PM	21:00	22:00
10PM to 11PM	22:00	23:00
11PM to Midnight	23:00	24:00

6. Since you know the exact process and server names (HOURLY and REGULAR, in that order), use a standard SLC to generate events when your process, HOURLY, does not run.

Set the Start and End Window Tolerance values to 1 hour. These values allow the hourly Process to start or end as much as one hour early or late and still be counted as running. If you do not want any tolerance at all, you might set the Start and End Window Tolerance values to zero hours.

Check **Generate Notification if Event Has Not Occurred** to cause SLC events to be generated when HOURLY does not run.

7. Select the Server REGULAR.
8. Select all 24 calendar SLC schedules you created.
9. On the Processess/Batches pane, specify the process name HOURLY.

How can a producer know when a file was not delivered on time?

To notify a producer when a file was not delivered on time, configure a workflow SLC that uses a calendar schedule.

Procedure

1. Set up a workflow SLC that uses a calendar schedule to define a typical time range with a normal end range start (NERs) and normal end range end (NERe).
2. In addition to the workflow schedule and other required information, enter the following parameters for the Workflow SLCs milestone. Substituting appropriate values for the keys, Server, SI.Mailbox Path, and Dest File Name, that meet your needs.

Key	Operator	Value
Server	Wild Card	*
SI.Is Put(true/false)	Wild Card	true
SI.Mailbox Path	Wild Card	/Producer/Mailbox
Dest File Name	Wild Card	Producerfile.txt

3. To define the milestone schedule, specify the maximum amount of time for the system to wait before a milestone SLC event is generated.
For example, set **Maximum Duration(Dmax)** to 15 minutes.

How can a producer get notified if the consumer does not pick up a file within 15 minutes?

To notify a producer if the consumer does not pick up a file within 15 minutes of its arrival, use a Workflow SLC with a duration schedule and milestones.

About this task

Create Milestone 1 to watch for the delivery (called File delivered to consumer MBX in this example).

Create Milestone 2 to monitor the pickup of the data delivered (called Consumer Picks Up the File)

Since multiple files can be picked up and delivered simultaneously, you must also specify a correlator value for the Workflow, which ties the name of the delivered file to the name of the picked up file.

Procedure

1. Create a workflow SLC with a duration schedule of 15 minutes. Set the **Minimum Duration** to 000:00:00 and **Maximum Duration** to 000:15:00.
2. To ensure that the milestones within the workflow are associated with the same file, specify destFile as the Correlator Source at the workflow level. The following table illustrates sample settings for the Parameters pane:

Milestone times Relative to Actual start of Workflow SLC	Concurrency Count	Correlator Source
Yes	1	destFile

3. To create the first milestone to watch for deliveries, specify the following parameters. Substitute appropriate values for the keys, SI.Mailbox Path, Dest File Name, and Server, that meet your needs:

Key	Operator	Value
SI.Mailbox Path	Wild Card	/RQ103PGPConsumerE/Inbox*
Dest File Name	Wild Card	*
Server	Wild Card	*
SI.Is Put(true/false)	Wild Card	true

4. To define the milestone schedule, again specify 15 minutes for the **Maximum Duration(Dmax)**.
5. To create the second milestone to watch for pickups that do not occur within 15 minutes of file delivery (SI.Is Put is false), specify the following criteria as parameter keys:

Key	Operator	Value
SI.Mailbox Path	Wild Card	The pickup location, for example, /RQ103PGPConsumerE/Inbox*
Dest File Name	Wild Card	*
SI.Is Put(true/false)	Wild Card	false
Server	Wild Card	*

6. As the last step, define the schedule for the second milestone the same way you did for the first by specifying 15 minutes for the **Maximum Duration(Dmax)**.

How can I use RUNBATCH to create schedules for an hourly process?

RUNBATCH takes a template and a script.

Template

For more information about the RUNBATCH utility and how to use it, see *Create multiple objects in System Administration*.

The template name is *recurringschedule.tmp*. It is located in the *Installation Directory/conf/templates* directory:

```
<calendarSchedule>
  <id>&name;</id>
  <desc>&name;</desc>
  <enabled>true</enabled>
  <ner>
    <start>&start;</start>
    <end>&end;</end>
    <day>0</day>
  </ner>
```

```
<calendarId>Daily</calendarId>
<timeZone>America/Chicago</timeZone>
<roles></roles>
</calendarSchedule>
```

Note: Substitute an appropriate value for `timeZone` in your template.

Script

The script name is `recurringschedule.txt`. It should be put in the *Installation Directory/conf/f* directory.

```
copy recurringschedule
&name; =0000 to 0100
&start; =00:00
&end; =01:00
SCHEDULE
copy recurringschedule
&name; =0100 to 0200
&start; =01:00
&end; =02:00
SCHEDULE
copy recurringschedule
&name; =0200 to 0300
&start; =02:00
&end; =03:00
SCHEDULE
copy recurringschedule
&name; =0300 to 0400
&start; =03:00
&end; =04:00
SCHEDULE
copy recurringschedule
&name; =0400 to 0500
&start; =04:00
&end; =05:00
SCHEDULE
copy recurringschedule
&name; =0500 to 0600
&start; =05:00
&end; =06:00
SCHEDULE
copy recurringschedule
&name; =0600 to 0700
&start; =06:00
&end; =07:00
SCHEDULE
copy recurringschedule
&name; =0700 to 0800
&start; =07:00
&end; =08:00
SCHEDULE
copy recurringschedule
&name; =0800 to 0900
&start; =08:00
&end; =09:00
SCHEDULE
copy recurringschedule
&name; =0900 to 1000
&start; =09:00
&end; =10:00
SCHEDULE
copy recurringschedule
&name; =1000 to 1100
&start; =10:00
&end; =11:00
```

```

SCHEDULE
copy recurringschedule
&name; =1100 to 1200
&start; =11:00
&end; =12:00
SCHEDULE
copy recurringschedule
&name; =1200 to 1300
&start; =12:00
&end; =13:00
SCHEDULE
copy recurringschedule
&name; =1300 to 1400
&start; =13:00
&end; =14:00
SCHEDULE
copy recurringschedule
&name; =1400 to 1500
&start; =14:00
&end; =15:00
SCHEDULE
copy recurringschedule
&name; =1500 to 1600
&start; =15:00
&end; =16:00
SCHEDULE
copy recurringschedule
&name; =1600 to 1700
&start; =16:00
&end; =17:00
SCHEDULE
copy recurringschedule
&name; =1700 to 1800
&start; =17:00
&end; =18:00
SCHEDULE
copy recurringschedule
&name; =1800 to 1900
&start; =18:00
&end; =19:00
SCHEDULE
copy recurringschedule
&name; =1900 to 2000
&start; =19:00
&end; =20:00
SCHEDULE
copy recurringschedule
&name; =2000 to 2100
&start; =20:00
&end; =21:00
SCHEDULE
copy recurringschedule
&name; =2100 to 2200
&start; =21:00
&end; =22:00
SCHEDULE
copy recurringschedule
&name; =2200 to 2300
&start; =22:00
&end; =23:00
SCHEDULE
copy recurringschedule
&name; =2300 to 2400
&start; =23:00
&end; =24:00
SCHEDULE

```

Chapter 9. Troubleshooting SLCs

SLCs can fail to work as expected for many reasons.

In this instance, “working” SLCs are defined as generating appropriate SLC events at the times determined by the SLC criteria, the schedule or schedules associated with the SLC.

These troubleshooting examples discuss SLC events that were not generated as expected and why. An SLC Debug Report is also available to assist you in troubleshooting SLC issues. Some reasons that SLC events fail to be generated include the following:

- SLC events are not generated by an SLC if it is not enabled.
- SLC events are not generated if Process type events (Process Started, Process Ended, Process Step Started, and Process Step Ended) do not match the SLC parameters specified.
- Back-level versions of managed servers, that is, managed servers not on current maintenance, may not generate statistics records, or may generate statistics records without the needed information, for the events constructed by Sterling Control Center to have the information needed to match the SLC parameters specified.

SLC tips

These tips can help you troubleshoot SLCs.

- To prevent extraneous events being generated by the SLC service, define only two of the following items in the schedules associated with an SLC:
 - Normal Start Range (NSR)
 - Normal End Range (NER)
 - Duration

That is, if an SLC has a calendar schedule that specifies both an NSR and NER, do not associate a duration schedule with the SLC, too. If an SLC has a duration schedule associated with it, and you also want to associate a calendar schedule with it, the calendar schedule should specify either an NSR or an NER, but not both. Do this to avoid conflicts that can occur due to the inherent duration derived by taking the difference between the calendar schedule NSR and NER values and the explicit duration specified in the duration schedule.

- Review the predefined rules Sterling Control Center comes with for SLC event types, and the actions they specify. Ensure that only the rules you wish to be triggered are enabled and that the actions they specify are appropriate. Disable rules that watch for events you do not care about.

For example, Sterling Control Center ships with a rule named Process Not Started Warn, which is triggered when an SLC event is generated at the start of the start range, NSRs, specified if an appropriate Process type event (Process started, or Process step started) does not transpire by that time. If knowing that a Process or Process step has not started by NSRs is immaterial to you, either disable the Process Not Started Warn rule, which is the best choice. The next best choice is to change the action that it refers to no operation. You can also change the action Alert Severity to <no alert generated>.

How do I fix my SLC?

If SLCs would do what you meant, as opposed to what you told them to do, they would work as expected the first time. Sometimes that is not the case.

There are many ways to set up SLCs incorrectly. There are things you can check to troubleshoot the issues you encounter.

SLC not enabled

It might be obvious, but sometimes SLCs do not work properly simply because they are not enabled. Always check SLCs that do not function correctly and make sure that they are enabled.

Match criteria

The primary reason SLCs fail to work as planned (assuming they are enabled) is that the data in the events that are supposed to trigger them does not exactly match the criteria specified.

For example, you might specify a process name in the SLC of *Xyzzy*, but if the process name is actually *XYZZY*, it does not trigger the SLC. Why? The *yzzy* in the criteria is in lowercase while the actual name shows *YZZY* in uppercase, which does not match.

Or, you might specify a destination file name of `c:\myfile.dat`. But if the destination file name is actually `C:\myfile.dat`, it does not trigger the SLC. Why? The `c` in the criteria is in lowercase, while the actual name shows `C` in uppercase, which does not match.

There are times when you must either use a wildcard SLC or a workflow SLC, as opposed to a standard SLC, to accomplish your monitoring goals. One of those times is you when you are monitoring transfers to generation data groups (GDGs). When transferring files to GDGs, the destination file name is not known until the file is created.

For example, if your process is transferring a file to the data set `XYZZY.GDG(+1)`, the SLC destination file name must be specified as `XYZZY.GDG.*` (assuming the Regular Expression check box is cleared).

Note: When specifying match criteria for wildcard SLCs or milestones in workflow SLCs, use only the Regex (Regular Expression) option if you specified a regular expression for the match criteria.

To find out whether specifying faulty match criteria is the reason your SLC is not working properly, run a Database Events report. View the details of the events you expect to trigger your SLC. Specify event parameters that match the Process you feel were supposed to trigger your SLC but did not. Include the `XML_STRING` value at a minimum in the report output; then closely compare the match criteria specified in your SLC with the output generated.

The Test button on many SLC wizard panels is also helpful. Use it to try out your wildcard or regular expressions to see whether they match the values you expect the events that trigger your SLC to contain.

A.M. versus P.M.

When constructing calendar schedules for use with SLCs, or specifying Normal Start Range (NSR) or Normal End Range (NER) values for workflow SLC milestones, remember that you must specify times using the 24-hour clock.

For example, if a Process is to start between 2:00 and 2:15 p.m., the time specified for NSRs and NSRe, respectively, must be 14:00 and 14:15. A typical mistake is to specify 02:00 and 02:15, but that means 2:00 to 2:15 in the morning.

Monitor windows

Another reason an SLC might not work as planned is that the events expected to trigger it occur outside its monitoring window.

In addition to the schedules associated with SLCs, SLC criteria include Start and End Window Tolerance values, which might range from zero to 24 hours.

Start and end events for the monitored Process or file transfer can occur outside the monitoring window of the SLC. When these events happen, it is as if the process or transfer never occurred. SLC events are generated as such.

Another problem related to monitoring windows can occur when SLCs have multiple calendar schedules. Those schedules can overlap due to the size of the start or end window tolerance value specified.

Note: For SLCs and milestones with duration schedules, only the end window tolerance value is used.

Sterling Control Center before version 5.0 have a minimum window tolerance of one hour.

For example, you have an SLC with two schedules. One schedule specifies an NERs of 02:00 and an NERe of 02:15, and a second specifies an NERs of 03:00 and an NERe of 03:15. The SLC has a start and end window tolerance of one hour. The monitoring windows for the two SLC instances overlap and might cause unexpected events to be generated, as illustrated in the following figure:



The monitor window for the first SLC instance ranges from 1:00 to 3:15. The monitor window for the second SLC instance ranges from 2:00 to 4:15. If the first process that ought to trigger the SLC does not run at all, but the second process starts anytime between 2:00 and 3:15, rather than generating SLC events that indicate the first Process did not run but the second process completed on time, the SLC events generated indicate that the first process completed late and the second process did not run.

In cases such as this one, consider setting a monitor tolerance value of zero hours. Adjust the SLC schedules as needed to avoid getting erroneous SLC events.

Monitored servers

Back-level versions of monitored servers that are not on current maintenance may not generate statistics records correctly if they generate them at all.

This causes issues with Sterling Control Center events, and as a result, Sterling Control Center may not have the information needed to match specified SLC parameters. For example, when monitoring transfers of files to GDGs, the dataset name in the process step start event must match the dataset name in the process step end event, or SLCs do not work correctly. If the server being monitored is not on current maintenance, a known (and resolved) issue causes the dataset name in the Process step start event not to be fully resolved. That is, instead of the destination dataset name in the process step start event being, for example, XYZZY.GDG.G0001V00, it is XYZZY.GDG(+1), while the dataset name in the Process step end event is fully resolved. If this occurs the SLC will not “work.” Even if the match criteria for the SLC is triggered by the dataset name in the step start event, because the dataset name in the step end event is different (the dataset name in the step end event is always the fully resolved GDG dataset name), the SLC logic does not recognize it as the terminating event.

You can tell if this issue is causing problems by closely examining the statistics records for the Process and seeing whether or not the dataset name in the step start and step end events match.

Note: For every schedule associated with an SLC, a separate SLC instance exists.

The *IBM Sterling Control Center Release Notes* is one resource you can use to judge whether your monitored servers are at an appropriate maintenance level for monitoring by Sterling Control Center.

Rules and actions

SLCs by themselves cause SLC events to be generated rather than alerts.

You must have the appropriate rules and actions in place to watch for the SLC events and perform the actions you want performed. These actions include generating alerts and emails, when those SLCs are not met.

When shipped, Sterling Control Center comes with built-in rules that specify one of the built-in actions for all the various SLC events that might be generated. Ensure that the appropriate rules and actions are in place, enabled or disabled, and in the appropriate priority order to get the results you want.

Note: Refer to *What events may be generated by SLCs and when?* and *Message IDs for rules* in *System Administration*. They can provide the list of SLC type events that could occur.

Milestone times

Values specified for Workflow SLC Milestone Normal Start Range and Normal End Range are offsets to the scheduled or actual start of the monitored workflow. These range values are durations, as opposed to clock times.

For example, if a workflow milestone is supposed to end no later than 1:05 a.m., you would not enter 01:05 as the Normal End Range end value. Instead, enter the number of minutes relative to the schedule associated with the workflow which combined yield 1:05.

SLC order

Unlike standard SLCs, wildcard and workflow SLCs must be ordered the way Sterling Control Center rules are.

In the situation in which a single event matches the criteria of more than one wildcard SLC, or more than one workflow SLC milestone, SLC order determines which SLC the event will trigger.

Time zones

SLCs will not work properly during the change from daylight saving time to standard time, and vice versa.

The time zone parameter specified for Calendar Schedules associated with SLCs must be the same value as the time zone setting specified for the servers that you are monitoring.

Everything is correct except the alerts

Sometimes you may think an SLC is not working because you are not seeing expected alerts in the Active Alerts Monitor.

Remember, alerts are nothing more than Sterling Control Center events with an additional element named Alert, along with an alert value (0, 1, 2, or 3). An alert element is added to an event by the rule service only if an enabled Sterling Control Center rule matches the event and specifies an action with an alert severity of 0, 1, 2, or 3. The Active Alerts Monitor displays all unhandled events that contain an alert element with a value greater than 0.

If you are not seeing expected alerts in the Active Alerts monitor, check to see if your situation resembles one of the following scenarios:

- Make sure that the rule that should be triggering the expected alerts has been properly defined by:
 - Specifying the appropriate SLC message ID
 - Associating an appropriate schedule and action with it
 - Enabling the rule
- The Active Alerts Monitor has the ability to overlay existing alerts with new correlated alerts. All SLC events contain an SLC instance identifier unique to each active SLC. The Active Alerts Monitor uses the SLC instance identifier to know when alerts it receives are correlated, and instead of displaying multiple alerts for the same SLC instance, displays only the latest.

For example, an SLC with a Normal Start Range (NSR) specified will generate an SLC notification with a message ID of CSLC047E if the NSRs time transpires without an appropriate starting event being seen, and another SLC notification with a message ID of CSLC034E if the NSRe time also transpires without an appropriate starting event being seen. In this case, the Active Alerts Monitor may first display the SLC event with the CSLC047E message, and then overlay that event with the SLC event containing the CSLC034E message, because their SLC_INSTANCE_ID values will be the same, which means they're correlated.

In fact, if the latest correlated event is an alert of severity 0, instead of displaying the newest alert in the view, the existing correlated alert (if one exists) is removed from the Active Alerts Monitor. For example, an SLC with a duration schedule generates an SLC notification with a message ID of CSLC049E because the dMin time passes without seeing an appropriate ending Process type event, and another SLC notification with a message ID of CSLC042I because an appropriate ending Process type event is seen before the dMax time occurs. If the rule triggered when an SLC notification with message ID CSLC042I occurs specifies an action with an alert severity of 0, then the alert with the message ID of CSLC049E is removed from the Active Alerts Monitor when the CSLC042I event occurs, instead of being replaced.

- The engine may not be collecting and processing events from monitored servers in a timely fashion. As a result, Sterling Control Center may issue contradictory messages. If you see out-of-compliance events for a process or process step, followed by an in-compliance event, this is typically caused by a lag between the time statistics are generated by monitored servers and the time statistics are collected and processed by Sterling Control Center.

For example, if an SLC-type event with a CSLC027E message (Process did not start by NERe) is followed by an SLC-type event with a CSLC037I message (Process ended within NERe) for the same SLC instance, it is almost certainly caused by the engine not receiving and processing the Process end or Process step end statistic prior to the NERe even though the process ended within the NERe.

- The concurrence count for the SLC may not be set properly. For example, you may see more alerts than expected if the concurrence count is set too high for SLCs with calendar schedules, or you may see fewer alerts than expected if the concurrence count is set too low for SLCs with duration schedules.
- Make sure the alerts are associated with the server, servers, or server group for which you opened the Active Alerts monitor. Some SLC events, in particular the ones for events that did not occur, may not be associated with any servers and, in that case, the only way to see them is by opening an Active Alerts monitor for all of Sterling Control Center.

What else can I do?

If you feel an SLC is not working but should be, check to see whether the expected SLC events were generated. One way to do this is to use the SLC Debug Report tool.

Procedure

1. Select **Tools > Reports > Run SLC Debug Report**.
2. Fill in the dialog that appears with as much information as you can, and then click **Run Report**.
3. At a minimum, you need to provide the specific SLC ID (the name of the SLC Group) you want the report to focus on and a time frame for events related to the SLC. You have the option to specify process information. If you choose to do so, you must specify all of the following pieces of information: Sterling Control Center Server ID, the Process Name, and Process number for each process specified.

Note: Note the format shown for the Start Time and End Time (YYYY/MM/DD HH:MM) must be used exactly if the default values are altered.

The report is saved to the Sterling Control Center engine log folder under the name, SLCDebugReport.html. Retrieve this file from the log folder of the Sterling Control Center engine install location to view the report results.

You may subsequently print this information, or copy and paste it into an email to send to someone to get help with your SLC.

The report will include:

- The definition of the SLC
- The definition of all schedules referenced by the SLC
- All related SLC events produced for the SLC specified in the time frame given, including any events that triggered the SLC events, such as a Process Start event
- The definition of any rules that were triggered by the SLC events generated
- The definition of any actions that were taken by the rules triggered
- The engine email system settings

If any process information is specified, the resulting report will also include all the process-related events for those processes.

The following shows a sample SLC Debug Report:

```
SLC Debug Utility Version 1.0
=====
Date/Time: Wed Aug 04 15:13:09 CDT 2010
SLC ID: bothStandardSLC1
Start Range: Wed Aug 04 00:00:00 CDT 2010
End Range: Wed Aug 04 17:00:00 CDT 2010
Process ID: [Name=sample2 #28754 Server= aix534000sp]
```

```
bothStandardSLC1:
<?xml version="1.0" encoding="UTF-8"?>
<slaGroup>
  <id>bothStandardSLC1</id>
  <name>bothStandardSLC1</name>
  <desc></desc>
  <enabled>false</enabled>
  <nodes>
    <node>aix534000sp</node>
  </nodes>
  <processes>
    <process>sample2</process>
  </processes>
  <fileNames></fileNames>
  <submitters></submitters>
  <remoteNodes></remoteNodes>
  <schedules>
    <schedule>bothDuration1</schedule>
  </schedules>
  <serverGroups></serverGroups>
  <visibilityGroups>
    <visibilityGroup>Both</visibilityGroup>
  </visibilityGroups>
  <missingEvent>true</missingEvent>
  <monitorToleranceStart>1</monitorToleranceStart>
  <monitorToleranceEnd>1</monitorToleranceEnd>
</slaGroup>
```

Schedules:

```
bothDuration1:
<?xml version="1.0" encoding="UTF-8"?>
<durationSchedule>
  <id>bothDuration1</id>
  <desc></desc>
```

```
<enabled>true</enabled>
  <simpleSlc>>false</simpleSlc>
  <dMin>000:00:00</dMin>
  <dMax>000:02:00</dMax>
  <vistype>private</vistype>
  <roles>
    <role>bothRole</role>
  </roles>
</durationSchedule>
```

SLC Event Report:

Event # 1

Event ID: 83947193024450792

Event Time: 2010/08/04 19:27:46.870 UTC

Processed Time: 2010/08/04 19:27:46.886|2010/08/04 19:27:46.883 UTC

Event Type: SLC

Message: CSLC033I Process started. SLC:

bothStandardSLC1.bothDuration1.aix534000sp.sample2.true.1.83947193024450767

SLC Instance ID: 83947193024450767

Rule Triggered: Process started on-time|Process started on-time-both

Alert: 0|3

Triggering event:

```
<event><alert>|</alert><ruleInstanceId>|</ruleInstanceId><actionId>|
</actionId><ruleId>|</ruleId><localNode>P</localNode><nodeName>
aix534000sp</nodeName><dvgt>Both</dvg><actionsFlag>2010/08/04
19:27:46.945|2010/08/0419:27:46.851</actionsFlag><dateTime>
2010/08/0419:27:49.000</dateTime><eMailFlag>0|0</eMailFlag><eventId>
83947193024450786</eventId><eventType>3</eventType><fileSize>
-1</fileSize><messageId>XSMG200I</messageId><nodeId>aix534000sp
</nodeId><nodeType>1</nodeType><origNode>aix534000sp</origNode>
<percentComplete>-1</percentComplete><processId>28754</processId>
<processName>sample2</processName><remoteNode>hpig4000sp</remoteNode>
<returnCode>0</returnCode><seqNum>1</seqNum><shortText>
Process started, process:28754 name:sample2SNODE:hpig4000sp
</shortText><slaFlag>0</slaFlag><submitterId>qatest</submitterId>
<trapFlag>0|0</trapFlag><userOpFlag>0|0</userOpFlag><SUBI>
qatest@aix534000sp</SUBI><SBND>aix534000sp</SBND><STPT>2010/08/04
19:27:49.000</STPT><STRT>2010/08/0419:27:49.000</STRT><PRTY>10
</PRTY><CLAS>1</CLAS><recCat>CAPR</recCat><recordId>PSTR<
/recordId></event>Event XML:<event><ruleInstanceId>|
</ruleInstanceId><activityType>P</activityType><sourceEventTime>
1280950069000</sourceEventTime><groupId>bothStandardSLC1</groupId>
<dvgt>Both</dvg><actionsFlag>2010/08/04 19:27:46.886|2010/08/04
19:27:46.883</actionsFlag><actionId>[alert0]|[alert3]</actionId>
<alert>0|3</alert><dateTime>2010/08/0419:27:46.870</dateTime>
<eMailFlag>0|0</eMailFlag><eventId>83947193024450792</eventId>
<eventType>6</eventType><fileSize>-1</fileSize><messageId>
CSLC033I</messageId><nodeId>aix534000sp</nodeId><nodeType>1
</nodeType><percentComplete>-1</percentComplete><processId>
28754</processId><processName>sample2</processName><remoteNode>
hpig4000sp</remoteNode><returnCode>0</returnCode><ruleId>Process
started on-time|Process startedon-time-both</ruleId><seqNum>
83947193024450792</seqNum><shortText>CSLC033I Process
started.SLC:bothStandardSLC1.bothDuration1.aix534000sp.sample2.
true.1.83947193024450767</shortText><slaFlag>0</slaFlag><slcId>
bothStandardSLC1.bothDuration1.aix534000sp.sample2.true.1.
83947193024450767</slcId><slcInstanceId>83947193024450767
</slcInstanceId><slcSource1>bothStandardSLC1.bothDuration1.
aix534000sp.sample2.true.1</slcSource1><slcSource2>aix534000sp.
sample2.28754.aix534000sp</slcSource2><slcSourceEventId>
83947193024450786</slcSourceEventId><trapFlag>0|0</trapFlag>
<userOpFlag>0|0</userOpFlag></event>
```

Event # 2

Event ID: 83947193024450794

Event Time: 2010/08/04 19:27:47.438 UTC

Processed Time: 2010/08/04 19:27:47.448|2010/08/04 19:27:47.445 UTC

Event Type: SLC
 Message: CSLC042I Process ended when expected - between dMin and dMax. SLC:
 bothStandardSLC1.bothDuration1.aix534000sp.sample2.true.1.83947193024450767
 SLC Instance ID: 83947193024450767
 Rule Triggered: Proc duration okay|Proc duration okay - both
 Alert: 0|3
 Triggering event:
 <event><alert>|</alert><ruleInstanceId>|</ruleInstanceId><actionId>|</actionId>
 <ruleId>|</ruleId><localNode>P</localNode><nodeName>aix534000sp</nodeName>
 <dvgt>Both</dvgt><actionsFlag>2010/08/04 19:27:47.465|2010/08/04
 19:27:47.432</actionsFlag><dateTime>2010/08/04
 19:27:50.000</dateTime><emailFlag>0|0</emailFlag><eventId>83947193024450791
 </eventId><eventType>4</eventType><fileSize>-1</fileSize><messageId>
 XSMG252I</messageId><nodeId>aix534000sp</nodeId><nodeType>1</nodeType>
 <origNode>aix534000sp</origNode><percentComplete>-1</percentComplete>
 <processId>28754</processId><processName>sample2</processName><remoteNode>
 hpig4000sp</remoteNode><returnCode>0</returnCode><seqNum>1</seqNum><sh
 ortText>Process completed,SNODE:hpig4000sp</shortText><slaFlag>0</slaFlag>
 <submitterId>qatest</submitterId><trapFlag>0|0</trapFlag><userOpFlag>0|0
 </userOpFlag><SUBI>qatest@aix534000sp</SUBI><SBND>aix534000sp</SBND>
 <STPT>2010/08/04 19:27:50.000</STPT><STRT>2010/08/0419:27:49.000</STRT>
 <recCat>CAPR</recCat><recordId>PRED</recordId></event>Event XML:
 <event><ruleInstanceId>|</ruleInstanceId><activityType>P</activityType>
 <sourceEventTime>1280950070000</sourceEventTime><groupId>bothStandardSLC1
 </groupId><dvgt>Both</dvgt><actionsFlag>2010/08/04 19:27:47.448|2010/08/04
 19:27:47.445</actionsFlag><actionId>[alert0]|[alert3]</actionId><alert>
 0|3</alert><dateTime>2010/08/0419:27:47.438</dateTime><emailFlag>0|0
 </emailFlag><eventId>83947193024450794</eventId><eventType>6</eventType>
 <fileSize>-1</fileSize><messageId>CSLC042I</messageId><nodeId>aix534000sp
 </nodeId><nodeType>1</nodeType><percentComplete>-1</percentComplete><proc
 essId>28754</processId><processName>sample2</processName><remoteNode>
 hpig4000sp</remoteNode><returnCode>0</returnCode><ruleId>Proc duration
 okay|Proc duration okay - both</ruleId><seqNum>83947193024450794</seqNum>
 <shortText>CSLC042I Process ended when expected - between dMin and dMax.
 SLC:bothStandardSLC1.bothDuration1.aix534000sp.sample2.true.1.
 83947193024450767</shortText><slaFlag>0</slaFlag><slcId>bothStandardSLC1.
 bothDuration1.aix534000sp.sample2.true.1.83947193024450767</slcId>
 <slcInstanceId>83947193024450767</slcInstanceId><slcSource1>bothStandardSLC1.
 bothDuration1.aix534000sp.sample2.true.1</slcSource1><slcSource2>aix534000sp.
 sample2.28754.aix534000sp</slcSource2><slcSourceEventId>83947193024450791<
 /slcSourceEventId><trapFlag>0|0</trapFlag><userOpFlag>0|0</userOpFlag>
 </event>

Rules:
 Process started on-time-both:
 <?xml version="1.0" encoding="UTF-8"?>
 <rule>
 <actionId>alert3</actionId>
 <desc>CSLC033I - alert3</desc>
 <enabled>true</enabled>
 <eventTypes>
 <eventType>6</eventType>
 </eventTypes>
 <id>Process started on-time-both</id>
 <match>"event[contains(messageId,'CSLC033I')]"</match>
 <priority>1</priority>
 <relative_priority>1</relative_priority>
 <visibilityGroups>
 <visibilityGroup>Both</visibilityGroup>
 </visibilityGroups>
 <linkedRule>
 <enabled>>false</enabled>
 <nonResolutionActionId>alert0</nonResolutionActionId>
 <resolutionActionId>alert0</resolutionActionId>
 <resolutionTimeout>5</resolutionTimeout>
 </linkedRule>
 <effectiveMatch>"event[contains(messageId,'CSLC033I')] and (

```
eventType = &apos;6&apos; ) ]&quot;;</effectiveMatch></rule>
```

Proc duration okay:

```
<?xml version="1.0" encoding="UTF-8"?>
<rule>
  <actionId>alert0</actionId>
  <desc>CSLC042I - alert0</desc>
  <enabled>true</enabled>
  <id>Proc duration okay</id>
  <match>&quot;;/event[eventType=&apos;6&apos; and
contains(messageId,&apos;CSLC042I&apos;)]&quot;;</match>

  <priority>21</priority><eventType>6</eventType></rule>
```

Process started on-time:

```
<?xml version="1.0" encoding="UTF-8"?>
<rule>
  <actionId>alert0</actionId>
  <desc>CSLC033I - alert0</desc>
  <enabled>true</enabled>
  <id>Process started on-time</id>
  <match>&quot;;/event[eventType=&apos;6&apos; and
contains(messageId,&apos;CSLC033I&apos;)]&quot;;</match>

  <priority>7</priority><eventType>6</eventType></rule>
```

Proc duration okay - both:

```
<?xml version="1.0" encoding="UTF-8"?>
<rule>
  <actionId>alert3</actionId>
  <desc>CSLC042I - alert3</desc>
  <enabled>true</enabled>
  <eventTypes>
    <eventType>6</eventType>
  </eventTypes>
  <id>Proc duration okay - both</id>
  <match>&quot;;/event[contains(messageId,&apos;CSLC042I&apos;)]&quot;;</match>
  <priority>8</priority>
  <relative_priority>8</relative_priority>
  <visibilityGroups>
    <visibilityGroup>Both</visibilityGroup>
  </visibilityGroups>
</linkedRule>
<enabled>>false</enabled>
<nonResolutionActionId>alert0</nonResolutionActionId>
<resolutionActionId>alert0</resolutionActionId>
<resolutionTimeout>5</resolutionTimeout>
</linkedRule>
<effectiveMatch>&quot;;/event[contains(messageId,&apos;CSLC042I&apos;) and
( eventType = &apos;6&apos; ) ]&quot;;</effectiveMatch></rule>
```

Actions:

alert0:

```
<?xml version="1.0" encoding="UTF-8"?>
<action>
  <alert>0</alert>
  <desc></desc>
  <id>alert0</id>
</action>
```

alert3:

```
<?xml version="1.0" encoding="UTF-8"?>
<action>
  <alert>3</alert>
  <desc></desc>
  <id>alert3</id>
```

</action>

Engine Email System Settings:

SMTP Host Name: shuttlerun
SMTP Host Port: 25

Events Related to Process ID: [Name=sample2 #28754 Server= aix534000sp]

Event # 1

Event ID: 83947193024450783
Event Time: 2010/08/04 19:27:48.000 UTC
Processed Time: 2010/08/04 19:27:46.804|2010/08/04 19:27:46.775 UTC
Event Type: Process status

Record ID: QCEX
Message: CNCD053I Process moved to Execution queue.

Rule Triggered: |
Alert: |

Event XML:

```
<event><alert>|</alert><ruleInstanceId>|</ruleInstanceId><actionId>|</actionId><ruleId>|</ruleId><processQueue>EXEC</processQueue><nodeName>aix534000sp</nodeName><localNode>P</localNode><dvgt>Both</dvgt><actionsFlag>2010/08/04 19:27:46.804|2010/08/04 19:27:46.775</actionsFlag><dateTime>2010/08/04 19:27:48.000</dateTime><emailFlag>0|0</emailFlag><eventId>83947193024450783</eventId><eventType>9</eventType><fileSize>-1</fileSize><messageId>CNCD053I</messageId><nodeId>aix534000sp</nodeId><nodeType>1</nodeType><origNode>aix534000sp</origNode><percentComplete>-1</percentComplete><processId>28754</processId><processName>sample2</processName><remoteNode>hpig4000sp</remoteNode><seqNum>0</seqNum><shortText>CNCD053I Process moved to Execution queue.</shortText><slaFlag>0</slaFlag><submitterId>qatest</submitterId><trapFlag>0|0</trapFlag><userOpFlag>0|0</userOpFlag><SUBI>qatest@aix534000sp</SUBI><SBND>aix534000sp</SBND><status>PE</status><recCat>CAEV</recCat><recordId>QCEX</recordId></event>
```

Event # 2

Event ID: 83947193024450786
Event Time: 2010/08/04 19:27:49.000 UTC
Processed Time: 2010/08/04 19:27:46.945|2010/08/04 19:27:46.851 UTC
Event Type: Process started

Record ID: PSTR
Message: Process started, process:28754 name:sample2 SNODE:hpig4000sp

Rule Triggered: |
Alert: |

Event XML:

```
<event><alert>|</alert><ruleInstanceId>|</ruleInstanceId><actionId>|</actionId><ruleId>|</ruleId><localNode>P</localNode><nodeName>aix534000sp</nodeName><dvgt>Both</dvgt><actionsFlag>2010/08/04 19:27:46.945|2010/08/04 19:27:46.851</actionsFlag><dateTime>2010/08/04 19:27:49.000</dateTime><emailFlag>0|0</emailFlag><eventId>83947193024450786</eventId><eventType>3</eventType><fileSize>-1</fileSize><messageId>XSMG200I</messageId><nodeId>aix534000sp</nodeId><nodeType>1</nodeType><origNode>aix534000sp</origNode><percentComplete>-1</percentComplete><processId>28754</processId><processName>sample2</processName><remoteNode>hpig4000sp</remoteNode><returnCode>0</returnCode><seqNum>1</seqNum><shortText>Process started, process:28754 name:sample2 SNODE:hpig4000sp</shortText><slaFlag>0</slaFlag><submitterId>qatest</submitterId><trapFlag>0|0</trapFlag><userOpFlag>0|0</userOpFlag><SUBI>qatest@aix534000sp</SUBI><SBND>aix534000sp</SBND><STPT>2010/08/04 19:27:49.000</STPT><STRT>2010/08/04 19:27:49.000</STRT><PRTY>10</PRTY><CLAS>1</CLAS><recCat>CAPR</recCat><recordId>PSTR</recordId></event>
```

Event # 3

Event ID: 83947193024450787
Event Time: 2010/08/04 19:27:49.000 UTC
Processed Time: 2010/08/04 19:27:46.962|2010/08/04 19:27:46.948|2010/08/04 19:27:46.958 UTC
Event Type: Process step started
Record ID: LSST
Message: Local Step started.
Rule Triggered: ||
Alert: ||
Event XML:
<event><alert>||</alert><ruleInstanceId>||</ruleInstanceId><actionId>|</actionId><ruleId>|</ruleId><localNode>P</localNode><nodeName>aix534000sp</nodeName><direction>outBound</direction><dvgt>Both|Up</dvgt><actionsFlag>2010/08/0419:27:46.962|2010/08/04 19:27:46.948|2010/08/04 19:27:46.958</actionsFlag><dateTime>2010/08/0419:27:49.000</dateTime><destFile>/data/cd4000sp/ndm/cfg/upfile.from_aix</destFile><eMailFlag>0|0|0</eMailFlag><eventId>83947193024450787</eventId><eventType>1</eventType><fileSize>-1</fileSize><fromNode>P</fromNode><messageId>XSMG201I</messageId><nodeId>aix534000sp</nodeId><nodeType>1</nodeType><origNode>aix534000sp</origNode><percentComplete>-1</percentComplete><processId>28754</processId><processName>sample2</processName><remoteNode>hpig4000sp</remoteNode><seqNum>2</seqNum><shortText>Local Step
cfg/msgfile.cfg</sourceFile><stepName>Up</stepName><submitterId>qatest</submitterId><trapFlag>0|0|0</trapFlag><userOpFlag>0|0|0</userOpFlag><SUBI>qatest@aix534000sp</SUBI><SBND>aix534000sp</SBND><STRT>2010/08/04 19:27:49.000</STRT><RSTR>N</RSTR><recCat>CAPR</recCat><recordId>LSST</recordId></event>

Event # 4
Event ID: 83947193024450788
Event Time: 2010/08/04 19:27:49.000 UTC
Processed Time: 2010/08/04 19:27:47.331|2010/08/04 19:27:47.279|2010/08/0419:27:47.326 UTC
Event Type: Process step ended
Record ID: CTRC
Message: Copy step successful.
Rule Triggered: |bothRule|upRule
Alert: |1|1
Event XML:
<event><alert>|1|1</alert><ruleInstanceId>||</ruleInstanceId><actionId>|[bothAlert1]| [upAlert1]</actionId><ruleId>|bothRule|upRule</ruleId><localNode>P</localNode><nodeName>aix534000sp</nodeName><direction>outBound</direction><dvgt>Both|Up</dvgt><actionsFlag>2010/08/04 19:27:47.331|2010/08/04 19:27:47.279|2010/08/0419:27:47.326</actionsFlag><dateTime>2010/08/04 19:27:49.000</dateTime><destFile>/data/cd4000sp/ndm/cfg/upfile.from_aix</destFile><eMailFlag>0|1280950067279|1280950067326</eMailFlag><eventId>83947193024450788</eventId><eventType>2</eventType><fileSize>4313802</fileSize><fromNode>P</fromNode><messageId>SCPA000I</messageId><nodeId>aix534000sp</nodeId><nodeType>1</nodeType><origNode>aix534000sp</origNode><percentComplete>-1</percentComplete><processId>28754</processId><processName>sample2</processName><remoteNode>hpig4000sp</remoteNode><returnCode>0</returnCode><seqNum>3</seqNum><shortText>Copy stepsuccessful.</shortText><slaFlag>0</slaFlag><sourceFile>/data/cd4000sp/ndm/cfg/msgfile.cfg</sourceFile><stepName>Up</stepName><submitterId>qatest</submitterId><trapFlag>0|0|0</trapFlag><userOpFlag>0|0|0</userOpFlag><SUBI>qatest@aix534000sp</SUBI><SBND>aix534000sp</SBND><STPT>2010/08/04 19:27:49.000</STPT><STRT>2010/08/04 19:27:49.000</STRT><certIssuer>(O=SCI/L=Irving/ST=Texas/C=US/SHA1=fed0aa6547066dccfc7476c19428b3a000b9c1e3/SN=00)</certIssuer><certSubject>(C=US/ST=Texas/O=SCI/OU=SV/CN=cdwopsxp01.csg.stercomm.com/emailAddress=jason_lege1@stercomm.com/SHA1=e40edf02b0bb93463fb213f364600f7ae5551ad7/SN=7B)</certSubject><secureEnabled>Y</secureEnabled><secureProtocol>TLSv1</secureProtocol><cipherSuite>TLS_DHE_RSA_WITH_AES_256_CBC_SHA</cipherSuite><OMSG>SCPA000I</OMSG><OCCD>0</OCCD><LMSG>SCPA000I</LMSG><LCCD>0</LCCD><RUSZ>16000</RUSZ><RSTR>N</RSTR><XLAT>N</XLAT><LKFL>N</LKFL><recCat>CAPR</recCat><CKPT>Y</CKPT><ECMP>Y</ECMP><SCMP>N</SCMP><DDS3>R</DDS3><bytesSent>4313802</bytesSent><bytesRead>4313802</bytesRead><SRCR>188</SRCR><SRUX>58</SRUX><DBYW>4313802</DBYW><DRCW>188</DRCW>

><DBYX>885370</DBYX><<DRUX>58</DRUX><recordId>CTRC</recordId></event>

Event # 5

Event ID: 83947193024450789
Event Time: 2010/08/04 19:27:49.000 UTC
Processed Time: 2010/08/04 19:27:47.351|2010/08/04
19:27:47.335|2010/08/04 19:27:47.347 UTC
Event Type: Process step started
Record ID: LSST
Message: Local Step started.
Rule Triggered: ||
Alert: ||

Event XML:

```
<event><alert>||</alert><ruleInstanceId>||</ruleInstanceId><actionId>|  
|</actionId><ruleId>||</ruleId><localNode>P</localNode><nodeName>  
aix53400sp</nodeName><direction>inBound</direction><dvg>Both|Down  
</dvg><actionsFlag>2010/08/04 19:27:47.351|2010/08/0419:27:47.335|2010/08/04  
19:27:47.347</actionsFlag><dateTime>2010/08/0419:27:49.000</dateTime>  
<destFile>/data/cd400sp/ndm/cfg/downfile.from_hp</destFile><emailFlag>0|0|0  
</emailFlag><eventId>83947193024450789</eventId><eventType>1</eventType  
><fileSize>-1</fileSize><fromNode>S</fromNode><messageId>XSMG201I</messageId>  
<nodeId>aix53400sp</nodeId><nodeType>1</nodeType><origNode>aix53400sp</origNode>  
<percentComplete>-1</percentComplete><processId>28754</processId><processName>  
sample2</processName><remoteNode>hpig400sp</remoteNode><seqNum>4</seqNum>  
<shortText>Local Step started.</shortText><slaFlag>0</slaFlag><sourceFile>/data/  
cd400sp/ndm/cfg/upfile.from_aix</sourceFile><stepName>Down</stepName><submitterId>  
qatest</submitterId><trapFlag>0|0|0</trapFlag><userOpFlag>0|0|0</userOpFlag>  
<SUBI>qatest@aix53400sp</SUBI><SBND>aix53400sp</SBND><STRT>2010/08/04  
19:27:49.000</STRT><RSTR>N</RSTR><recCat>CAPR</recCat><recordId>LSST  
</recordId></event>
```

Event # 6

Event ID: 83947193024450790
Event Time: 2010/08/04 19:27:50.000 UTC
Processed Time: 2010/08/04 19:27:47.429|2010/08/04
19:27:47.383|2010/08/04 19:27:47.424 UTC
Event Type: Process step ended
Record ID: CTRC
Message: Copy step successful.
Rule Triggered: |bothRule|downRule
Alert: |1|1

Event XML:

```
<event><alert>|1|1</alert><ruleInstanceId>||</ruleInstanceId><actionId>|[bothAlert1]|  
[downAlert1]</actionId><ruleId>|bothRule|downRule</ruleId><localNode>P</localNode><no  
deName>aix53400sp</nodeName><direction>inB  
ound</direction><dvg>Both|Down</dvg><actionsFlag>2010/08/04 19:27:47.429|2010/08/04  
19:27:47.383|2010/08/04 19:27:47.424</actionsFlag><dateTime>2010/08/04  
19:27:50.000</dateTime><destFile>/data/cd400sp/ndm/cfg/downfile.from_hp</destFile><e  
MailFlag>0|1280950067383|1280950067424</emailFlag><eventId>83947193024450790</eventId  
><eventType>2</eventType><fileSize>4313802</fileSize><fromNode>S</fromNode><messageId  
>SCPA000I</messageId><nodeId>aix53400sp</nodeId><nodeType>1</nodeType><origNode>aix5  
3400sp</origNode><percentComplete>-1</percentComplete><processId>28754</processId><p  
rocessName>sample2</processName><remoteNode>hpig400sp</remoteNode><returnCode>0</ret  
urnCode><seqNum>0</seqNum><shortText>Copy step  
successful.</shortText><slaFlag>0</slaFlag><sourceFile>/data/cd400sp/ndm/cfg/upfile.  
from_aix</sourceFile><stepName>Down</stepName><submitterId>qatest</submitterId><trapF  
lag>0|0|0</trapFlag><userOpFlag>0|0|0</userOpFlag><SUBI>qatest@aix53400sp</SUBI><SBN  
D>aix53400sp</SBND><STPT>2010/08/04 19:27:50.000</STPT><STRT>2010/08/04  
19:27:49.000</STRT><certIssuer>/O=SCI/L=Irving/ST=Texas/C=US/SHA1=fed0aa6547066dccfc  
7476c19428b3a000b9c1e3/SN=00</certIssuer><certSubject>/C=US/ST=Texas/O=SCI/OU=SV/CN  
=cdwopsxp01.csg.stercomm.com/emailAddress=jason_lege1@stercomm.com/SHA1=e40edf02b0bb9  
3463fb213f364600f7ae5551ad7/SN=7B</certSubject><secureEnabled>Y</secureEnabled><secu  
reProtocol>TLSv1</secureProtocol><cipherSuite>TLS_DHE_RSA_WITH_AES_256_CBC_SHA</ciphe  
rSuite><OMSG>SCPA000I</OMSG><OCCD>0</OCCD><LMSG>SCPA000I</LMSG><LCCD>0</LCCD><RUSZ>16  
000</RUSZ><RSTR>N</RSTR><XLAT>N</XLAT><LKFL>N</LKFL><recCat>CAPR</recCat><CKPT>Y</CKP
```

T><ECMP>Y</ECMP><<SCMP>N</SCMP><<DDS3>R</DDS3><<bytesSent>4313802</bytesSent><<bytesRead>4313802</bytesRead><<SRCR>188</SRCR><<SRUX>58</SRUX><<DBYW>4313802</DBYW><<DRCW>188</DRCW><<DBYX>885370</DBYX><<DRUX>58</DRUX><<recordId>CTRC</recordId></event>

Event # 7

Event ID: 83947193024450791

Event Time: 2010/08/04 19:27:50.000 UTC

Processed Time: 2010/08/04 19:27:47.465|2010/08/04 19:27:47.432 UTC

Event Type: Process ended

Record ID: PRED

Message: Process completed, SNODE:hpig4000sp

Rule Triggered: |

Alert: |

Event XML:

```
<event><alert>|</alert><ruleInstanceId>|</ruleInstanceId><actionId>|</actionId><ruleId>|</ruleId><localNode>P</localNode><nodeName>aix534000sp</nodeName><dvg>Both</dvg><actionsFlag>2010/08/04 19:27:47.465|2010/08/04 19:27:47.432</actionsFlag><dateTime>2010/08/04 19:27:50.000</dateTime><emailFlag>0|0</emailFlag><eventId>83947193024450791</eventId><eventType>4</eventType><fileSize>-1</fileSize><messageId>XSMG252I</messageId><nodeId>aix534000sp</nodeId><nodeType>1</nodeType><origNode>aix534000sp</origNode><percentComplete>-1</percentComplete><processId>28754</processId><processName>sample2</processName><remoteNode>hpig4000sp</remoteNode><returnCode>0</returnCode><seqNum>1</seqNum><shortText>Process completed, SNODE:hpig4000sp</shortText><slaFlag>0</slaFlag><submitterId>qatest</submitterId><trapFlag>0|0</trapFlag><userOpFlag>0|0</userOpFlag><SUBI>qatest@aix534000sp</SUBI><SBND>aix534000sp</SBND><STPT>2010/08/04 19:27:50.000</STPT><STRT>2010/08/04 19:27:49.000</STRT><recCat>CAPR</recCat><recordId>PRED</recordId></event>
```

Event # 8

Event ID: 83947193024450792

Event Time: 2010/08/04 19:27:46.870 UTC

Processed Time: 2010/08/04 19:27:46.886|2010/08/04 19:27:46.883 UTC

Event Type: SLC

Message: CSLC033I Process started. SLC:

bothStandardSLC1.bothDuration1.aix534000sp.sample2.true.1.83947193024450767

SLC Instance ID: 83947193024450767

Rule Triggered: Process started on-time|Process started on-time-both

Alert: 0|3

Triggering event:

```
<event><alert>|</alert><ruleInstanceId>|</ruleInstanceId><actionId>|</actionId><ruleId>|</ruleId><localNode>P</localNode><nodeName>aix534000sp</nodeName><dvg>Both</dvg><actionsFlag>2010/08/04 19:27:46.945|2010/08/04 19:27:46.851</actionsFlag><dateTime>2010/08/04 19:27:49.000</dateTime><emailFlag>0|0</emailFlag><eventId>83947193024450786</eventId><eventType>3</eventType><fileSize>-1</fileSize><messageId>XSMG200I</messageId><nodeId>aix534000sp</nodeId><nodeType>1</nodeType><origNode>aix534000sp</origNode><percentComplete>-1</percentComplete><processId>28754</processId><processName>sample2</processName><remoteNode>hpig4000sp</remoteNode><returnCode>0</returnCode><seqNum>1</seqNum><shortText>Process started, process:28754 name:sample2 SNODE:hpig4000sp</shortText><slaFlag>0</slaFlag><submitterId>qatest</submitterId><trapFlag>0|0</trapFlag><userOpFlag>0|0</userOpFlag><SUBI>qatest@aix534000sp</SUBI><SBND>aix534000sp</SBND><STPT>2010/08/04 19:27:49.000</STPT><STRT>2010/08/04 19:27:49.000</STRT><PRTY>10</PRTY><CLAS>1</CLAS><recCat>CAPR</recCat><recordId>PSTR</recordId></event>
```

Event XML:

```
<event><ruleInstanceId>|</ruleInstanceId><activityType>P</activityType><sourceEventTime>1280950069000</sourceEventTime><groupId>bothStandardSLC1</groupId><dvg>Both</dvg><actionsFlag>2010/08/04 19:27:46.886|2010/08/04 19:27:46.883</actionsFlag><actionId>[alert0] |[alert3]</actionId><alert>0|3</alert><dateTime>2010/08/04 19:27:46.870</dateTime><emailFlag>0|0</emailFlag><eventId>83947193024450792</eventId><eventType>6</eventType><fileSize>-1</fileSize><messageId>CSLC033I</messageId><nodeId>aix534000sp</nodeId><nodeType>1</nodeType><percentComplete>-1</percentComplete><processId>28754</processId><processName>sample2</processName><remoteNode>hpig4000sp</remoteNode><returnCode>0</returnCode><ruleId>Process started on-time|Process started
```

on-time-both</ruleId><seqNum>83947193024450792</seqNum><shortText>CSLC033I Process started. SLC:
bothStandardSLC1.bothDuration1.aix534000sp.sample2.true.1.83947193024450767</shortText><slaFlag>0</slaFlag><slcId>bothStandardSLC1.bothDuration1.aix534000sp.sample2.true.1.83947193024450767</slcId><slcInstanceId>83947193024450767</slcInstanceId><slcSource1>bothStandardSLC1.bothDuration1.aix534000sp.sample2.true.1</slcSource1><slcSource2>aix534000sp.sample2.28754.aix534000sp</slcSource2><slcSourceEventId>83947193024450786</slcSourceEventId><trapFlag>0|0</trapFlag><userOpFlag>0|0</userOpFlag></event>

Event # 9

Event ID: 83947193024450794

Event Time: 2010/08/04 19:27:47.438 UTC

Processed Time: 2010/08/04 19:27:47.448|2010/08/04 19:27:47.445 UTC

Event Type: SLC

Message: CSLC042I Process ended when expected - between dMin and dMax. SLC:
bothStandardSLC1.bothDuration1.aix534000sp.sample2.true.1.83947193024450767

SLC Instance ID: 83947193024450767

Rule Triggered: Proc duration okay|Proc duration okay - both

Alert: 0|3

Triggering event:

<event><alert>|</alert><ruleInstanceId>|</ruleInstanceId><actionId>|</actionId><ruleId>|</ruleId><localNode>P</localNode><nodeName>aix534000sp</nodeName><dvg>Both</dvg><actionsFlag>2010/08/04 19:27:47.465|2010/08/04 19:27:47.432</actionsFlag><dateTime>2010/08/04 19:27:50.000</dateTime><emailFlag>0|0</emailFlag><eventId>83947193024450791</eventId><eventType>4</eventType><fileSize>-1</fileSize><messageId>XSMG252I</messageId><nodeId>aix534000sp</nodeId><nodeType>1</nodeType><origNode>aix534000sp</origNode><percentComplete>-

1</percentComplete><processId>28754</processId><processName>sample2</processName><remoteNode>hpig4000sp</remoteNode><returnCode>0</returnCode><seqNum>1</seqNum><shortText>Process completed,

SNODE:hpig4000sp</shortText><slaFlag>0</slaFlag><submitterId>qatest</submitterId><trapFlag>0|0</trapFlag><userOpFlag>0|0</userOpFlag><SUBI>qatest@aix534000sp</SUBI><SBND>aix534000sp</SBND><STPT>2010/08/04 19:27:50.000</STPT><STRT>2010/08/04 19:27:49.000</STRT><recCat>CAPR</recCat><recordId>PRED</recordId></event>

Event XML:

<event><ruleInstanceId>|</ruleInstanceId><activityType>P</activityType><sourceEventTime>1280950070000</sourceEventTime><groupId>bothStandardSLC1</groupId><dvg>Both</dvg><actionsFlag>2010/08/04 19:27:47.448|2010/08/04 19:27:47.445</actionsFlag><actionId>[alert0] |[alert3]</actionId><alert>0|3</alert><dateTime>2010/08/04 19:27:47.438</dateTime><emailFlag>0|0</emailFlag><eventId>83947193024450794</eventId><eventType>6</eventType><fileSize>-1</fileSize><messageId>CSLC042I</messageId><nodeId>aix534000sp</nodeId><nodeType>1</nodeType><percentComplete>-

1</percentComplete><processId>28754</processId><processName>sample2</processName><remoteNode>hpig4000sp</remoteNode><returnCode>0</returnCode><ruleId>Proc duration okay|Proc duration okay -

both</ruleId><seqNum>83947193024450794</seqNum><shortText>CSLC042I Process ended when expected - between dMin and dMax. SLC:

bothStandardSLC1.bothDuration1.aix534000sp.sample2.true.1.83947193024450767</shortText><slaFlag>0</slaFlag><slcId>bothStandardSLC1.bothDuration1.aix534000sp.sample2.true.1.83947193024450767</slcId><slcInstanceId>83947193024450767</slcInstanceId><slcSource1>bothStandardSLC1.bothDuration1.aix534000sp.sample2.true.1</slcSource1><slcSource2>aix534000sp.sample2.28754.aix534000sp</slcSource2><slcSourceEventId>83947193024450791</slcSourceEventId><trapFlag>0|0</trapFlag><userOpFlag>0|0</userOpFlag></event>

SLC: bothStandardSLC1.bothDuration1.aix534000sp.sample2.true.1.83947193024450767

SLC Instance ID: 83947193024450767

Rule Triggered: Proc duration okay|Proc duration okay - both

Alert: 0|3

Triggering event:

<event><alert>|</alert><ruleInstanceId>|</ruleInstanceId><actionId>|</actionId><ruleId>|</ruleId><localNode>P</localNode><nodeName>aix534000sp</nodeName><dvg>Both</dvg><actionsFlag>2010/08/04 19:27:47.465|2010/08/04 19:27:47.432</actionsFlag><dateTime>2010/08/04 19:27:50.000</dateTime><emailFlag>0|0</emailFlag><eventId>83947193024450791</eventId><eventType>4</eventType><fileSize>-1</fileSize><messageId>XSMG252I</messageId><nodeId>aix534000sp</nodeId><nodeType>1</nodeType><origNode>aix534000sp</origNode><percentCo

```

mplete>-1</percentComplete><processId>28754</processId><processName>sample2</processName>
<remoteNode>hpig4000sp</remoteNode><returnCode>0</returnCode><seqNum>1</seqNum><shortText>Process completed,
SNODE:hpig4000sp</shortText><slaFlag>0</slaFlag><submitterId>qatest</submitterId><trapFlag>0|0</trapFlag><userOpFlag>0|0</userOpFlag><SUBI>qatest@aix534000sp</SUBI><SBND>aix534000sp</SBND><STPT>2010/08/04 19:27:50.000</STPT><STRT>2010/08/04
19:27:49.000</STRT><recCat>CAPR</recCat><recordId>PRED</recordId></event>
Event XML:
<event><ruleInstanceId>|</ruleInstanceId><activityType>P</activityType><sourceEventTime>1280950070000</sourceEventTime><groupId>bothStandardSLC1</groupId><dvgt>Both</dvgt><actionsFlag>2010/08/04 19:27:47.448|2010/08/04
19:27:47.445</actionsFlag><actionId>[alert0]|[alert3]</actionId><alert>0|3</alert><dateTime>2010/08/04
19:27:47.438</dateTime><emailFlag>0|0</emailFlag><eventId>83947193024450794</eventId><eventType>6</eventType><fileSize>-1</fileSize><messageId>CSLC042I</messageId><nodeId>aix534000sp</nodeId><nodeType>1</nodeType><percentComplete>-1</percentComplete><processId>28754</processId><processName>sample2</processName><remoteNode>hpig4000sp</remoteNode><returnCode>0</returnCode><ruleId>Proc duration okay|Proc duration okay -
both</ruleId><seqNum>83947193024450794</seqNum><shortText>CSLC042I Process ended when
expected - between dMin and dMax. SLC:
bothStandardSLC1.bothDuration1.aix534000sp.sample2.true.1.83947193024450767</shortText><slaFlag>0</slaFlag><slcId>bothStandardSLC1.bothDuration1.aix534000sp.sample2.true.1.83947193024450767</slcId><slcInstanceId>83947193024450767</slcInstanceId><slcSource1>bothStandardSLC1.bothDuration1.aix534000sp.sample2.true.1</slcSource1><slcSource2>aix534000sp.sample2.28754.aix534000sp</slcSource2><slcSourceEventId>83947193024450791</slcSourceEventId><trapFlag>0|0</trapFlag><userOpFlag>0|0</userOpFlag></event>

```

The following event elements might be seen in events included in an SLC Debug Report:

actionId

Action performed when the event was processed

actionsFlag

Tells when the event was processed by the Rule service

activityType

The type of SLC component that generated the event:

- WF = Workflow
- M = Milestone
- S = Process Step
- P =Process

dateTime

Time the event was generated (values are in UTC)

destFile

Destination file name watched for by this SLC instance

groupId

SLC name

nodeId

Server name watched for by this SKC instance

ruleId

Rule triggered by the event

slcId

For Wildcard SLCs, formed by concatenating the SLC Name, the SLC schedule name, the server name specified, the destination file name specified, and other values

slcSourceEventId

Event ID of the event that caused this SLC event to be generated (not present in the sample debug report provided because it was the lack of process events that caused these SLC events to be generated)

Chapter 10. Updating information for multiple servers

About changing passwords in mass

If the same user ID and password are used to attach multiple servers to Sterling Control Center, you may want to perform a mass change of all passwords to avoid having to individually update each server.

You can create a template to use with a script and then run RUNBATCH to do a mass password change. For a detailed explanation of the sample script (script.txt) and templates, see *Create multiple objects in System Administration*.

How can I do a bulk update of the passwords used by Sterling Control Center for monitored servers?

When you add a server to monitor, you specify a user ID and password for Sterling Control Center to use to log onto the server.

About this task

Instead of individually changing the password property for each server, you can follow this basic procedure:

Procedure

1. Copy the xml definitions of all servers whose passwords you want to update from the *Installation Directory/conf/services/nodes* folder to the template folder, and change all .xml extensions to .tmp.
2. Edit these new template files and insert a password variable by replacing the text between `<password>` and `</password>` with `&pw;`.
3. Create a script that includes the following three lines for each template supplying your data in the italicized items:

```
copy <template filename (without the .tmp extension)>  
&pw;=<new password value>  
SERVERUPDATE
```
4. Type **runBatch** hostname port userid password scriptname (Microsoft Windows) or **runBatch.sh** hostname port userid password scriptname (UNIX).

Results

Here is a sample template created from the xml definition for a server named SFG75039 (the server ID is italicized below). The sample template is named SFG75039.tmp. The encoded version of the encrypted password was replaced with `&pw;`.

```
<service>  
  <apiPort>1463</apiPort>  
  <applyMetadataRules>true</applyMetadataRules>  
  <connectionTimeout>30</connectionTimeout>  
  <contact/>  
  <contactEMail/>  
  <contactName/>  
  <contactPhone/>  
  <desc/>  
  <hostName>SFG</hostName>
```

```

<id>SFG75039</id>
<ignoreServerProvidedTZ>>false</ignoreServerProvidedTZ>
<licenseExpiryWarning>30</licenseExpiryWarning>
<maxCompletedProcs>100</maxCompletedProcs>
<notManaged>>false</notManaged>
<notMonitored>>false</notMonitored>
<password>&pw;</password>
<pollingRate>5</pollingRate>
<protocol>TCP/IP</protocol>
<serverMetaData1/>
<serverMetaData10/>
<serverMetaData2/>
<serverMetaData3/>
<serverMetaData4/>
<serverMetaData5/>
<serverMetaData6/>
<serverMetaData7/>
<serverMetaData8/>
<serverMetaData9/>
<serverTimeZoneId>America/Chicago</serverTimeZoneId>
<serviceClass>com.sterlingcommerce.scc.agent.services.node.cd.CDNodeService</serviceClass>
<serviceId>SFG75039</serviceId>
<serviceType>1</serviceType>
<skipProcessStepInfo>>false</skipProcessStepInfo>
<sourcePort/>
<synchObjectTime>SYSTEM</synchObjectTime>
<tracingEnabled>>false</tracingEnabled>
<userId>cduser</userId>
</service>

```

Here is a sample script which tells RUNBATCH to copy the SFG75039 template, substitute the new value (newpassword) for the specified password variable, and update the xml definition for the SFG75039 server. Note that the SERVERUPDATE command tells RUNBATCH to only update the information for the server versus create the xml definition from scratch.

```

copy SFG75039
&pw; = newpassword
SERVERUPDATE

```

The last step is to run runBatch.bat file from the installation directory/bin folder to run the sample script.

Notices

This information was developed for products and services offered in the U.S.A.

IBM® may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing

IBM Corporation

North Castle Drive

Armonk, NY 10504-1785

U.S.A.

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing

Legal and Intellectual Property Law

IBM Japan Ltd.

19-21, Nihonbashi-Hakozakicho, Chuo-ku

Tokyo 103-8510, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be

incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation

J46A/G4

555 Bailey Avenue

San Jose, CA 95141-1003

U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© IBM 2013. Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. 2013.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

IBM, the IBM logo, and [ibm.com](http://www.ibm.com)[®] are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at <http://www.ibm.com/legal/copytrade.shtml>.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency which is now part of the Office of Government Commerce.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java™ and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.

Linear Tape-Open, LTO, the LTO Logo, Ultrium and the Ultrium Logo are trademarks of HP, IBM Corp. and Quantum in the U.S. and other countries.

Connect Control Center®, Connect:Direct®, Connect:Enterprise, Gentran®, Gentran®:Basic®, Gentran:Control®, Gentran:Director®, Gentran:Plus®, Gentran:Realtime®, Gentran:Server®, Gentran:Viewpoint®, Sterling Commerce™, Sterling Information Broker®, and Sterling Integrator® are trademarks or registered trademarks of Sterling Commerce®, Inc., an IBM Company.

Other company, product, and service names may be trademarks or service marks of others.

Index

A

- actions
 - creating
 - to send an email 7
- alerts
 - none showing in Active Alerts Monitor 63
- Allow user
 - see only data from specified remote node 3
 - see only inventory processes across multiple servers 3
 - see only when specific file transfer occurs 4

B

- bulk password update 77

C

- calendar
 - permissions 6
 - restricting use 6
- calendar permissions 5
- changing passwords in mass 77
- Copy completes
 - rules 12
- Create action to generate alert
 - how to 7
- Create action to send an email and generate an alert
 - how to 7
- Create action to send email
 - how to 7
- Create an action to release a Sterling Connect:Direct Process
 - How to 7
- Create rule
 - applies only to data a specific user can see 4

D

- data visibility group
 - when to use 1
- download by recipient satisfied in time 29
- DVG
 - when to use 1

E

- email
 - creating an action to send 7

G

- Generate high-severity alert
 - how to 8
- Generate SNMP trap
 - how to 8

H

- How to
 - create action to send an email and generate an alert 7
 - generate a high-severity alert 8
 - generate SNMP trap 8
- How to know
 - when a Process completes 11
 - when a Process has a problem 11

O

- objects
 - controlling user permissions 5
- opening a help desk ticket
 - passing an operating system command and variables 8
- out-of-compliance events
 - troubleshooting 63

P

- password
 - bulk update 77

Q

- queue limit
 - notify if items exceed threshold 19
- QuickFile linked rule 27
- QuickFile workflow SLC 29

R

- request for file satisfied in time 27
- rule
 - server down alert more than five minutes 14
- rules
 - avoid receiving alerts from both servers in a process 17
 - build a schedule to prevent rules from triggering during maintenance window 35
 - copy completes 12
 - email critical database problem 14
 - file is delivered to a Sterling B2B Integrator mailbox 22
 - general topics 11
 - how to specify more than one message ID in rule criteria 17

rules (continued)

- know when a server is not being monitored regularly 18
- notify producer when a file does not arrive in mailbox within 5 minutes after connection is initiated 23
- notify producer whether a file is delivered to a consumer's mailbox 22
- QuickFile 27
- server goes down 13
- server starts 14
- Sterling B2B Integrator 20
- Sterling B2B Integrator adapter
 - enabled 21
- Sterling B2B Integrator adapter has stopped 20
- Sterling Connect:Enterprise 23
- Sterling Connect:Enterprise for UNIX
 - autoconnect 25
 - daemon stops 23, 24
- Sterling Connect:Enterprise for UNIX offline add 24
- Sterling Connect:Enterprise for UNIX offline extraction 25
- Sterling Connect:Enterprise for UNIX remote connect 25
- Sterling Connect:Enterprise for z/OS
 - autoconnect 27
 - remote connect disconnection 26
- Sterling Connect:Enterprise offline extraction 26
- Sterling Connect:Enterprise remote connect delete 26
- Sterling Connect:Enterprise remote connect disconnection 26
- Sterling Connect:Enterprise remote connection add 26
- Sterling File Gateway 21
 - notify route fails 22
- when a Sterling Connect:Direct Process goes in Held in error Queue 16
- when server stops handle alert automatically when server restarts 15
- when Sterling B2B Integrator adapter goes down for more than 5 minutes 21

S

- See only data from specified remote node
 - allow user 3
- See only inventory processes across multiple servers
 - allow user 3
- See only when specific file transfer occurs 4

- Server goes down
 - rules 13
- set up
 - automated report of alerts for previous day 37
- SLC
 - monitor windows 61
- SLC notification
 - for calendar schedule-based SLCs 40
- SLC notifications
 - for calendar duration-based SLCs 42
- SLC order 63
- SLC tips 59
- SLCnot enabled 60
- SLCs
 - A.M. versus P.M. 61
 - alert when related Sterling Connect:Enterprise batches arrive 45
 - does not complete on time 51
 - events generated by 40
 - Hourly Process is not on time 52
 - how to fix 60
 - maintenance level of monitored servers 62
 - match criteria 60
 - milestone times 63
 - one or more related processes
 - not complete on time 50
 - process fails to complete on time 45
 - process runs too long 44
 - reduce number of events with same message generated by workflow SLCs 51
 - related group of critical processes
 - do not all complete on time 49
 - related processes
 - do not start or complete on time 49
 - fail to start or complete on time 49
 - related Processes
 - run too long 47
 - rules and actions 62
 - Sterling B2B Integrator
 - know when a file was not delivered on time 54
 - notify producer if consumer does not pick up file within 15 minutes 54
 - time zones 63
 - troubleshooting 64
 - troubleshooting overview 59
 - use RUNBATCH to create schedules
 - for hourly process 55
 - versus a rule 39
 - when to use 40
- SLCs and rules
 - when use 39
- Sterling Connect:Enterprise for UNIX
 - daemon stops
 - rules 23
- Sterling File Gateway
 - notify route fails
 - rules 22

T

- troubleshooting
 - out-of-compliance 63
 - using the Database Events report 33
- troubleshooting rules 33
- troubleshooting SLCs 64

U

- using SLCs 39

W

- When a Process completes
 - how to know 11
- When a Process has a problem
 - how to know 11



Product Number: 5725-D02

Printed in USA