

# Configuring Logging



This appendix describes how to configure logging for SDX components and applications.

Topic	Page
Overview	G-1
References	G-1
Configuration Tasks	G-2

## Overview

---

SDX components and applications generate event messages that you can save in logs—either by writing the messages to text files or by using the system log (syslog) facilities. You can use these logs to monitor the SDX components and troubleshoot problems.

## References

---

For more information about system logging, see:

- The syslog Protocol – draft-ietf-syslog-protocol-00.txt (June 2004 expiration)

## Configuration Tasks

---

To configure logging with SDX Configuration Editor:

- 1 Access the logging configuration for the SDX component or application.

Refer to the documentation for the SDX component or application for information about how to access the logging configuration. Figure G-1 shows a typical logging configuration.

- 2 Identify the fields for the method that you want to use to save the logs.
  - To save the information in text files, complete the File Logger fields (see *Configuring the Software to Save Event Messages in Text Files*, later in this appendix).
  - To save the information through system logging, complete the Syslog Logger fields (see *Configuring System Logging*, later in this appendix).
- 3 Add additional instances of logs that you require. For information about defining the types of event messages that the software saves, see *Specifying Filters for Logs*, later in this appendix.
- 4 Save the configuration.

**Log destinations**

Create a New Instance of   Delete an Instance

---

**File Logger (file-1)**

Filter

Filename

Rollover Filename

Maximum File Size

---

**File Logger (file-2)**

Filter

Filename

Rollover Filename

Maximum File Size

---

**File Logger (file-3)**

Filter

Filename

Rollover Filename

Maximum File Size

---

**▼ Syslog Logger (syslog-1)**

Filter

Syslog Host

Syslog facility

Format

---

**▼ Syslog Logger (syslog-2)**

Filter

Syslog Host

Syslog facility

Format

**Figure G-1** Example of logging configuration

### *Configuring the Software to Save Event Messages in Text Files*

You can configure the software to save event messages in log files. You can also specify the maximum size for a log file and a rollover file for the log file. In this case, the software saves event messages in the current log file until it reaches the maximum size.

When the log file reaches the maximum size, the software closes the log file and renames it with the name you specified for the rollover file. If a previous rollover file exists, the software overwrites it. The software then reopens the log file and continues to save event messages in it.

To configure the software to save event messages in text files:

- 1 Enable the types of event messages that you want to record.
- 2 Specify the absolute paths to the files in which you want to record the logs.

By default, SDX components and applications write log files in the folder where the application is started. However, the user under which the J2EE application server or Web application server runs may not have write access to this folder. For logging to work properly, configure the component or application to write logs in folders to which this user has write access. If you are using the version of JBoss packaged with the SDX software, add the absolute path `/opt/UMC/jboss/server/default/log/` to the filenames and rollover filename for each log. For example, for the debug log, use the filename `/opt/UMC/jboss/server/default/log/vta_debug.log`.

- 3 (Optional) Specify the absolute paths to the rollover files.
- 4 (Optional) Enable checking of the size of the file for the current logs, and specify the maximum file size.

#### **Filter**

- Specifies the type of messages that this log file contains.
- Value – see *Defining Priorities*, later in this appendix
- Default for file-1 – `/debug-` – All event messages of severity level debug and higher
- Default for file-2 – `/info-` – All event messages of severity level info and higher
- Default for file-3 – `/error-` – All event messages of severity level error and higher

#### **Filename**

- Specifies the absolute path of the filename that contains the current logs.
- Value – text string
- Example – `/opt/UMC/jboss/server/default/log/`

### **Rollover Filename**

- Specifies the absolute path of the filename that contains the log history.
- Value – text string
- Example – `/opt/UMC/jboss/server/default/log/`

### **Maximum File Size**

- Specifies the maximum size in KB of the log file and the rollover file.



**Note:** Do not set the maximum file size to a value greater than the available disk space.

- Value – integer in the range 0–4294967295
- Default – 1000000

### *Configuring System Logging*

You can configure the software to save event messages on a host that you have configured as a system logging server. You can also specify the facility for system logging and the format in which the messages will be saved on the host.

To configure system logging:

- 1 Enable the types of event messages that you want to record.
- 2 Specify the hosts that will record the logs.
- 3 (Optional) Enable and specify the facilities that generate the event messages.
- 4 (Optional) Enable and specify the format in which the messages will be recorded.

### **Filter**

- Specifies the type of messages that the software records.
- Value – see *Defining Priorities*, later in this appendix
- Default for syslog-1 – `/debug` – all event messages of severity level debug
- Default for syslog-2 – `/info-warning` – event messages with a minimum severity level of info and a maximum severity level of warning

### **Syslog Host**

- Specifies the IP address or name of a host that collects event messages via a standard system logging daemon.
- Value – IP address or text string
- Default – localhost

### **Syslog facility**

- Specifies the type of system log in accordance with the system logging protocol (see *References*, earlier in this appendix).
- Value – integer in the range 0–23; each integer corresponds to the standard number for a system logging client
- Default – 3

### **Format**

- Specifies how the information in an event message is printed.
- Value – MessageFormat string as specified in <http://java.sun.com/j2se/1.4.2/docs/api/java/text/MessageFormat.html>. The fields available for events are:
  - › 0 – time and date of the event
  - › 1 – name of the thread generating the event
  - › 2 – text message of the event
  - › 3 – category of the event
  - › 4 – priority of the event.
- Default for text files – {0,time,HH:mm:ss.SSS z} {0,date,dd.MM.yyyy} [{1}] [{3}] [{4}] {2}

An example message for the default setting is:

```
14:13:24.366 EST 19.01.2004 [main] [Start-up module] [20]
SAE STARTUP DONE
```

- Default for syslog – SSP[{1}] [{3}] [{4}] {2}

Since the syslog system usually timestamps all log messages, no time information is included in the default format. An example message for the default setting is:

```
SSP[main] [Start-up module] [20] SAE STARTUP DONE
```

### *Specifying Filters for Logs*

In the filter field of each type of log, you can specify an expression that defines the *categories* and *priorities* of event messages that the software saves.

### Defining Categories

The category of an event message defines the SDX component that generated the event message. Default filters for SDX logs do not define categories; they include event messages for all SDX components.

If you want to view only event logs in a specific category you can define a variable `<category>`, which is a text string that matches the name of a category. This variable is not case sensitive. To view the names of categories for event messages, you should view a log file for one of the default filters (see *Configuring the Software to Save Event Messages in Text Files*, earlier in this appendix).

For example, the category `Cops` defines event messages generated by the COPS server. Similarly, the category `CopsMsg` defines a particular sort of event message that the COPS server generates.

Juniper Networks Customer Service can also provide names of categories, especially for troubleshooting purposes.

### Defining Priorities

The priority of an event message defines its level of severity. The event filter provides 128 levels of severity numbered 1–127. A higher number indicates a higher level of severity. Common levels of severity also have a specific name, as shown in Table G-1.

**Table G-1** Named severity levels

Name	Severity Level
logmin	1
debug	10
info	20
notice	30
warning	40
error	50
crit	60
alert	70
emerg	80
panic	90
logmax	127

You can define a priority as follows:

- Specify an explicit severity. For example:
  - > debug – defines only debug messages
- Specify a minimum severity and a maximum severity. For example:
  - > info-warning – defines messages of minimum severity level of info and a maximum severity level of warning
- Accept the default minimum (logmin) or maximum (logmax) severity by omitting the minimum or maximum severity. For example:
  - > info- – defines messages of minimum severity level info and maximum severity level logmax
  - > -warning – defines messages of minimum severity level logmin and maximum severity level warning
- Specify no severities to log all event messages.

The syntax for the priority takes the format:

[<severity>] | [<minimumSeverity>]-[<maximumSeverity>]

Use either the name or the number of a severity level shown in Table G-1 for the variables in this syntax.

## Defining Filters

You specify a filter by defining an expression with the following format:

singlematch [,singlematch]\*

- singlematch – [!] ( <category> | ([<category>]/[<severity>] | [<minimumSeverity>]-[<maximumSeverity>] ) )
- ! – do not log matching events
- <category> – see the earlier section, *Defining Categories*
- [<severity>] | [<minimumSeverity>]-[<maximumSeverity>] – see the previous section, *Defining Priorities*.

The software filters events by evaluating each subexpression in order from left to right. When the software determines that an event message matches a subexpression, the software logs or ignores the message accordingly. You can specify an unlimited number of subexpressions; however, the order in which you specify the subexpressions will affect the result.

Table G-2 shows some examples of filters.

**Table G-2** Examples of filters for event messages

Syntax	Event Messages Saved
/	All event messages
/info-	Event messages of level info and above from all categories
Cops/debug	Debug events from COPS category only
!Cops,/debug	All debug events except those from COPS category
CopsMsg/info-,!CopsMsg,Cops	All messages from COPS category, except those from CopsMsg category with level less than info

