Novell®

Sentinel™

Rev: 01

www.novell.com

June 29, 2007

Using Sentinel 5.x Collectors in Sentinel 6

Product Version(s): Requires Sentinel 6.0 or higher

Novell®

Legal Notices

Novell Inc. makes no representations or warranties with respect to the contents or use of this documentation and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc. reserves the right to revise this publication and to make changes to its content, at any time, without obligation to notify any person or entity of such revisions or changes.

Further, Novell, Inc. makes no representations or warranties with respect to any software, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Novell, Inc. reserves the right to make changes to any and all parts of Novell software, at any time, without any obligation to notify any person or entity of such changes.

Any products or technical information provided under this Agreement may be subject to U.S. export controls and the trade laws of other countries. You agree to comply with all export control regulations and to obtain any required licenses or classification to export, re-export, or import deliverables. You agree not to export or re-export to entities on the current U.S. export exclusion lists or to any embargoed or terrorist countries as specified in the U.S. export laws. You agree to not use deliverables for prohibited nuclear, missile, or chemical biological weaponry end uses. Please refer to www.novell.com/info/exports/ for more information on exporting Novell software. Novell assumes no responsibility for your failure to obtain any necessary export approvals.

Novell, Inc. has intellectual property rights relating to technology embodied in the product that is described in this document. In particular, and without limitation, these intellectual property rights may include one or more of the U.S. patents listed at http://www.novell.com/company/legal/patents/ and one or more additional patents or pending patent applications in the U.S. and other countries.

Novell, Inc.
404 Wyman Street, Suite 500
Waltham, MA 02451
U.S.A.
www.novell.com

Online Documentation: To access the online documentation for this and other Novell products, and to get updates, see www.novell.com/documentation.

Novell Trademarks

For Novell trademarks, see the Novell Trademark and Service Mark list (http://www.novell.com/company/legal/trademarks/tmlist.html).

Third-Party Materials

All third-party trademarks are the property of their respective owners.

Third-Party Legal Notices

Sentinel 6 may contain the following third-party technologies:

- Apache Axis and Apache Tomcat, Copyright © 1999 to 2005, Apache Software Foundation. For more information, disclaimers and restrictions, see http://www.apache.org/licenses/
- Apache Lucene, Copyright © 1999 to 2005, Apache Software Foundation. For more information, disclaimers and restrictions, see http://www.apache.org/licenses/
- ANTLR. For more information, disclaimers and restrictions, see http://www.antlr.org
- Boost, Copyright © 1999, http://Boost.org
- BSF, licensed by the Apache Software Foundation Copyright © 1999-2004. For more information, disclaimers and restrictions see http://xml.apache.org/dist/LICENSE.txt.
- Bouncy Castle, Copyright © 2000-2004, the Legion of Bouncy Castle. For more information, disclaimers and restrictions see http://www.bouncycastle.org
- Checkpoint. Copyright © Check Point Software Technologies Ltd.
- Concurrent, utility package. Copyright © Doug Lea. Used without CopyOnWriteArrayList and ConcurrentReaderHashMap classes.
- Crypto++ Compilation. Copyright © 1995-2003, Wei Dai, incorporating the following copyrighted work: mars.cpp by Brian Gladman and Sean Woods. For more information, disclaimers and restrictions see http://www.eskimo.com/
- Crystal Reports Developer and Crystal Reports Server. Copyright © 2004 Business Objects Software Limited
- DataDirect Technologies Corp. Copyright © 1991-2003
- edpFTPj, licensed under the Lesser GNU Public License. For more information, disclaimers and restrictions see http://www.enterprisedt.com/products/edtftpj/purchase.html
- Enhydra Shark, licensed under the Lesser General Public License available at: http://shark.objectweb.org/license.html
- Esper. Copyright 2005-2006, Codehaus.
- ICEsoft ICEbrowser. ICEsoft Technologies, Inc. Copyright © 2003-2004
- ILOG, Inc. Copyright © 1999-2004
- Installshield Universal. Copyright © 1996–2005, Macrovision Corporation and/or Macrovision Europe Ltd
- Java 2 Platform, Standard Edition. Copyright © Sun Microsystems, Inc. For more information, disclaimers and restrictions see http://java.sun.com/j2se/1.4.2/j2re-1_4_2_10-license.txt

  The Java 2 Platform may also contain the following third-party products:

  - CoolServlets © 1999
  - DES and 3xDES © 2000 by Jef Poskanzer
  - Crimson © 1999-2000 The Apache Software Foundation
  - Xalan J2 © 1999-2000 The Apache Software Foundation
  - NSIS 1.0j © 1999-2000 Nullsoft, Inc

- Eastman Kodak Company © 1992
- Lucinda, a registered trademark or trademark of Bigelow and Holmes
- Taligent, Inc
- IBM, some portions available at: http://oss.software.ibm.com/icu4j/

For more information regarding these third-party technologies and their associated disclaimers and restrictions, see: http://java.sun.com/j2se/1.4.2/j2se-1_4_2-thirdpartylicensereadme.txt

- JavaBeans Activation Framework (JAF). Copyright © Sun Microsystems, Inc. For more information, disclaimers and restrictions see http://www.java.sun.com/products/javabeans/glasgow/jaf.html
- JavaMail. Copyright © Sun Microsystems, Inc. For more information, disclaimers and restrictions see http://www.java.sun.com/products/javamail/downloads/index.html
- Java Ace, by Douglas C. Schmidt and his research group at Washington University and Tao (with ACE wrappers) by Douglas C. Schmidt and his research group at Washington University, University of California, Irvine and Vanderbilt University. Copyright © 1993-2005. For more information, disclaimers and restrictions see http://www.cs.wustl.edu/~schmidt/ACE-copying.html
- Java Authentication and Authorization Service Modules, licensed under the Lesser General Public License. For more information, disclaimers and restrictions see http://free.tagish.net/jaas/index.jsp
- Java Network Launching Protocol (JNLP). Copyright © Sun Microsystems, Inc. For more information, disclaimers and restrictions, please see http://www.java.sun.com/products/javawebstart/download-jnlp.html
- Java Service Wrapper. Portions copyrighted as follows: Copyright © 1999, 2004 Tanuki Software and Copyright © 2001 Silver Egg Technology. For more information, disclaimers and restrictions, see http://wrapper.tanukisoftware.org/doc/english/license.html
- JIDE. Copyright © 2002 to 2005, JIDE Software, Inc.
- JLDAP. Copyright 1998-2005 The OpenLDAP Foundation. All rights reserved. Portions Copyright (C) 1999 - 2003 Novell, Inc. All Rights Reserved.
- jTDS is licensed under the Lesser GNU Public License. For more information, disclaimers and restrictions see http://jtds.sourceforge.net/
- MDateSelector. Copyright © 2005, Martin Newstead, licensed under the Lesser General Public License. For more information, disclaimers and restrictions see http://web.ukonline.co.uk/mseries
- Monarch Charts. Copyright © 2005, Singleton Labs
- Net-SNMP. Portions of the code are copyrighted by various entities, which reserve all rights. Copyright © 1989, 1991, 1992 by Carnegie Mellon University; Copyright © 1996, 1998 to 2000, the Regents of the University of California; Copyright © 2001 to 2003 Networks Associates Technology, Inc.; Copyright © 2001 to 2003, Cambridge Broadband, Ltd.; Copyright © 2003 Sun Microsystems, Inc. and Copyright © 2003 to 2004, Sparta, Inc. For more information, disclaimers and restrictions, see http://net-SNMP.sourceforge.net
- The OpenSSL Project. Copyright © 1998-2004. The Open SSL Project. For more information, disclaimers and restrictions, see http://www.openssl.org
- Oracle Help for Java. Copyright © 1994-2006, Oracle Corporation
- RoboHELP Office. Copyright © Adobe Systems Incorporated, formerly Macromedia.
- SecurityNexus. Copyright © 2003 to 2006. SecurityNexus, LLC. All rights reserved.
- Skin Look and Feel (SkinLF). Copyright © 2000-2006 L2FProd.com. Licensed under the Apache Software License. For more information, disclaimers and restrictions see https://skinlf.dev.java.net/
- Sonic Software Corporation. Copyright © 2003-2004. The SSC software contains security software licensed from RSA Security, Inc
- Tinyxml. For more information, disclaimers and restrictions see http://grinninglizard.com/tinyxmldocs/index.html
- SecurityNexus. Copyright © 2003 to 2006. SecurityNexus, LLC. All rights reserved.
- Xalan and Xerces, both of which are licensed by the Apache Software Foundation Copyright © 1999-2004. For more information, disclaimers and restrictions see http://xml.apache.org/dist/LICENSE.txt

- yWorks. Copyright © 2003 to 2006, yWorks.

NOTE: As of publication of this documentation, the above links were active. In case you find any of these links broken/inactive, please contact: Novell, Inc., 404 Wyman Street, Suite 500, Waltham, MA 02451 U.S.A.

Contents

About this Guide

This manual gives you a general understanding of how the changes in Sentinel 6 will affect Sentinel 5 collectors, whether the collectors are provided by Novell or are custom developed. It is intended mainly for the system administrators migrating Sentinel 5 collectors to Sentinel 6.

Additional Documentation

The other manuals on this product are available at the following URLs:

- http://www.novell.com/documentation/sentinel5
- http://www.novell.com/documentation/sentinel6
- http://support.novell.com/products/sentinel/collectors.html

The additional documentation includes:

- Sentinel User's Guide for Sentinel 6
- Connector documentation for Sentinel 5
- Connector documentation for Sentinel 6
- Collector documentation for Sentinel 5
- Collector documentation for Sentinel 6
- Documentation on differences between Sentinel 5 and Sentinel 6 for the following connection methods:
  - Audit
  - File
  - Database
  - Syslog
  - WMI

Documentation Conventions

The following are the conventions used in this manual:

- `ls, --help`: commands, options
- Go to *Start* > *Program Files* > *Control Panel* to perform this action: Multiple actions in a step
- Any references to Sentinel 5.x also apply to Sentinel 4.x. Sentinel 5.x is used for simplicity.
- For more information, refer to *Chapter Name* in *Guide Name*: This is a reference to a chapter/section in another book.

NOTE: Any important notes for the user are mentioned as a Note.

Caution: A Caution indicates information that the user should read to avoid a potentially undesirable result.

Introduction

Sentinel 6 has introduced several key changes to the collector infrastructure with the introduction of the Event Source Management (ESM) framework. The following table shows some of the major functions in Sentinel and what application is used to perform those functions.

Function	Sentinel 5.1.3 and previous versions	Sentinel 6
Build Collector	Collector Builder	Collector Builder
Modify Collector	Collector Builder	Collector Builder
Debug Collector	Collector Builder	Sentinel Control Center
Deploy Collector	Collector Builder	Sentinel Control Center
Set Parameter Values	Collector Builder	Sentinel Control Center*

Some collectors written for Sentinel 5.x and previous versions of Sentinel (referred to as "Sentinel 5.x" in this document for simplicity) may require modification in order to work with Sentinel 6. This document's purpose is to enable the collector developer with Sentinel 5 experience and familiarity with the Sentinel 6 ESM framework to successfully address potential problem areas and port custom collectors to Sentinel 6. Some issues to consider include:

- Degree of customization to the Collector and availability of a comparable Sentinel 6 collector from Novell
- Legacy connection method vs. Connector plugin
- JDBC vs. ODBC connection
- GETCONFIG command changes
- FILE command changes
- SHELL commands and associated paths
- Legacy heartbeat messages
- Multiple instances of a collector, implemented using multiple .chn files
- Non-default record delimiter in Collectors using the FILE connection method

This document should be used in conjunction with several the other documentation sources listed in Additional Documentation.

Connection Methods

In Sentinel 5.x, the connectivity code for file, serial, socket, ODBC, SNMP and process connections was embedded in the Collector Engine and Collector Manager processes. Other connectivity methods (for example, WMI, JDBC, SDEE, Syslog and LEA) were handled by creating an external process and connecting to that process using the internal process connector. Collectors were built, modified, and deployed using the Collector Builder application.

---

* In Sentinel 6, parameter value defaults are set in Collector Builder, but the actual values for a deployed Collector are set in the Sentinel Control Center.

In Sentinel 6, Connector plugins have replaced the embedded connection methods. This standardizes the way that Sentinel acquires data from devices that it monitors and provides the ability to easily perform updates when a newer version of the Connector is released. It is possible to use the Sentinel 6 Process Connector plugin to continue using a custom connection method from Sentinel 5.x, but some features of Event Source Management will be unavailable. To take advantage of all of the new features of Event Source Management, the old Sentinel 5.x connection methods should be replaced by Sentinel 6 Connector plugins.

The following table contains the available Sentinel 5.x connectors and the equivalent Connector plugins for Sentinel 6.

Sentinel 5	**Sentinel 6**
File connector internal to Collector Manager	File Connector plugin
Serial connector internal to Collector Manager	No current support planned
Socket connector internal to Collector Manager	Socket Client Connector plugin (not yet released)
ODBC script commands internal to Collector Manager	Supported, but does not use either a Connector or Event Source. Only the Collector is needed. However, it is highly recommend that the code be modified to support the DB Connector instead.
Process connector internal to Collector Manager	Process Connector plugin
SNMP connector available through port configuration, internal to collector manager	SNMP Connector plugin
WMI using EventLog.exe and the process connector	WMI Connector plugin
JDBC using dbconnector.bat or dbconnector.sh and the process connector	Database (DB) Connector plugin
SDEE using sdee-app.bat or sdee-app.sh and the process connector	SDEE Connector plugin
Syslog connector using SyslogConnectorAgent.bat or SyslogConnectorAgent.sh and the process connector	Syslog Connector plugin
Audit connector using SyslogConnectorAgent.bat or SyslogConnectorAgent.sh, modified for Audit, and the process connector	Audit Connector plugin
LEA connector using lea_client.exe and the process connector	LEA Connector plugin
Custom application using the process connector	Process Connector plugin and custom application

For more information about any of these connectors, refer to the Additional Documentation section of this document.

RX State

In Sentinel 5.x, the RX State had several pieces of functionality. It allowed the person creating the collector to specify the delimiter (to indicate the end of a record) and set the timeout limit for receiving data. During operation, the RX State internally set the values of two variables when it returned control to the parsing script.

In Sentinel 6, some of this functionality has changed.

s_RXBufferString and i_RXBufferLength

In Sentinel 5.x, the RX State internally set the values of two variables, s_RXBufferString and i_RXBufferLength, when it returned control to the parsing script.

In Sentinel 6, the same variables are set, using the same format, for backward compatibility. For example, the DB Connector plugin still returns name-value pair data in s_RXBufferString. In addition, the Connector plugins may also set additional variables, which are described in the documentation for each Connector.

The addition of new variables to the Connector plugins should not have a negative impact on Sentinel 5.x collectors running in Sentinel 6. However, it is a good practice to check the connector documentation for the names of these new variables and ensure that your collector script does not use the same variable names.

Delimiter

In Sentinel 5.x, the RX state specified a record delimiter, which was embedded in the parsing script. For deployed collectors, this information can be viewed in the Sentinel 5.x Collector Builder application.

In Sentinel 6, the original delimiter is not available to the Connector plugin and must be specified in the Connector plugin's configuration. This may be done at deployment time or by setting connection mode properties in the collector's package.xml file. The supported set of properties can be found in the respective Connector plugin documents.

Timeout

In Sentinel 5.x, the RX state can be set to timeout if it is of type TIMEOUT or DELIMITED TIMEOUT. This information is embedded in the parsing script and can be viewed in the Sentinel 5.x Collector Builder application. The occurrence of a timeout would cause a returning RX state with i_RXBufferLength set to 0 and a blank s_RXBufferString. Many Sentinel 5.x collectors performed an action when the timeout was detected; they could

- send an internal event called a heartbeat
- stop processing and exit (for example, when a vuln scan read is complete)
- start reading from a new input file (file rotation)

In Sentinel 6, the timeout period can be set for older scripts utilizing the timeout capability of the RX state when deploying the Collector by setting the *RX_TIMEOUT_DELAY* parameter.

Heartbeat

In Sentinel 5.x, many collectors sent a heartbeat when a timeout happened. This indicated that although no data was received, the collector was still up and running. In Sentinel 6, the Event Source Management framework monitors Collectors, Connectors, and Event Sources and displays information about whether they are still running. Therefore, the heartbeat messages in Sentinel 5.x Collectors may be considered redundant. In most Sentinel 5.x Collectors, the heartbeat messages may be disabled in Sentinel 6 when the Collector is deployed by setting the parameter *Send_Heartbeat_Messages* to *off*.

Collectors that Exit

In Sentinel 5.x, some Collectors are designed to run and then automatically exit using a Stop state in the Collector template. For example, a vulnerability collector may be designed to run until it reaches the end of the file (indicated by the timeout), and then stop.

In Sentinel 6, the Event Source Management framework does not recognize the Stop state. It may show an error for a Collector that intentionally ran and then exited. The workaround is to simply manually stop the Collector in the ESM *Live View*.

Collectors with File Rotation

In Sentinel 5.x, some Collectors are designed to use the timeout to indicate that the end of the log file had been reached and it was necessary to rotate to a new log file. Because the internal file connector is no longer used in Sentinel 6, using the GETCONFIG command in Sentinel 6 for file rotation will not work. For more information about how to use the log file rotation functionality built into the Connector Plugin, refer to the Connector documentation or *File Connector Differences in Sentinel 6.*

Paths and Directories

In Sentinel 5.x, a running instance of a collector shared the same file space as the collector code:

```
%ESEC_HOME%\wizard\Elements
$ESEC_HOME/wizard/Elements
```

The environment variable WORKBENCH_HOME referred to %ESEC_HOME%\wizard.

In Sentinel 6, the running instance and the development instance have been separated into two different locations

```
%ESEC_HOME%\data\collector_workspace (for development)
%ESEC_HOME%\data\collector_mgr.cache\collector_instances (for
   running instances)
```

The value for the environment WORKBENCH_HOME has been changed from %ESEC_HOME%\wizard (in Sentinel 5.x) to simply %ESEC_HOME% (in Sentinel 6).

In Sentinel 6, a new directory has been created called %ESEC_HOME%\data. This directory will typically have several subdirectories with the same names as the Collectors. The data directory is used to store data that needs to be preserved after the collector is modified, such as an offset file. (If a collector is modified and redeployed, the running directory in %ESEC_HOME%\data\collector_mgr.cache \collector_instances will also be modified.)

These path and directory changes may affect Collector script implementations.

Collector Script Commands

Several commands have been modified or deprecated in Sentinel 6. All Sentinel 5.x Collectors should be examined to determine whether they use these commands.

SHELL

Collectors that use the SHELL command should be examined to ensure the directory path is valid. Any collector that uses an absolute directory path or a relative path that does not strictly traverse down a path (for example, one that uses ..\..\ ), should be examined for correctness in the Sentinel 6 environment.

FILE Commands

In Sentinel 6, the FILEA, FILER, FILEW, and FILEL commands have all been modified to work from the %ESEC_HOME%\data directory.

GETCONFIG

The GETCONFIG command has been updated to use the Sentinel 6 paths. The following properties have been updated:

- System.PortScript returns the running instance directory name (for example, WMI_6_0_Collector_68714633-A987-1029-A520-000C29F2D765, which is a combination of the script collector name and the instance UUID)
- System.Data_Dir returns the path to the data directory (%ESEC_HOME%\data)
- System.Agent_Dir returns the path to the parent directory of all the running collector instances (for example, %ESEC_HOME%\data\collector_mgr.cache\collector_instances)
- System.Local.Dir returns the path to the collector's running instance directory (equivalent to System.Agent.Dir concatenated with System.PortScript)

Deprecated Commands

In Sentinel 6 several parsing commands have been deprecated.

- DISPLAY – replaced by debugging functionality in Sentinel 6
- INDICATOR – replaced by the EVENT command
- POPUP – replaced by debugging functionality in Sentinel 6

Although the DB commands (DBOPEN, DBGETROW, and others) used in ODBC collectors are still functional in Sentinel 6, Novell recommends that ODBC collectors be updated to connect through JDBC instead. The advantages of using JDBC instead of ODBC are explained in the document *DBConnector Differences in Sentinel 6.*

Building Collectors and Parameter Usage

In Sentinel 5.x, after modifying a Collector in the Collector Builder, it was necessary to manually "build" the Collector and then upload it to the Collector Manager. During the build, several things happened:

1. The parameter values were saved to a file with a .par extension.
2. A script file was written to a file with an .asd extension.

In Sentinel 6, the manual build step is no longer necessary. Collectors are automatically built before execution by the Event Source Management framework. The parameter values entered when the Collector is deployed are stored in the database and are also placed in the parameters.csv file in the Collector's running instance directory before execution. This set of values is built into the .asd file automatically when the collector is executed.

When a Sentinel 5.x collector is imported into Sentinel 6, the names of the parameters will be standardized. These name changes will be visible in Collector Builder.

Caution: The Collector may stop functioning if you change the new parameter names.

Importing Collectors

After completing any necessary collector modifications, you can import the 5.x collector into Sentinel 6.

To import a collector from an existing 5.x implementation in Sentinel 6:

1. Either install the Sentinel Control Center (version 6) on the machine with the Collector files or copy the Collector files to a machine on which the Sentinel Control Center (version 6) is installed.
2. Open Sentinel Control Center as a user with Event Source Management permissions.
3. Go to *Event Source Management > Live View*.
4. Follow the instructions in the Sentinel User's Guide for importing a connector.

5. Follow the instructions in the Sentinel User's Guide to start importing the collector from a directory.
6. Complete the fields for any missing metadata.

Multiple Instances of the Same Collector

In Sentinel 5, there were two ways to support multiple instances of the same collector code. The most common method was to copy that collector and modify the parameters of the second copy. This method is still supported in Sentinel 6.

In Sentinel 5.x, the other way to support multiple instances of the same collector was to add a new column to the parameter configuration in Collector Builder. This information was saved to the parameters (.par) file. A new .asd script file was created automatically during the build process, and the collector developer had to manually create a new .chn file that referred to the new .asd file. The new .chn file was then selected during the port configuration when deploying the collector.

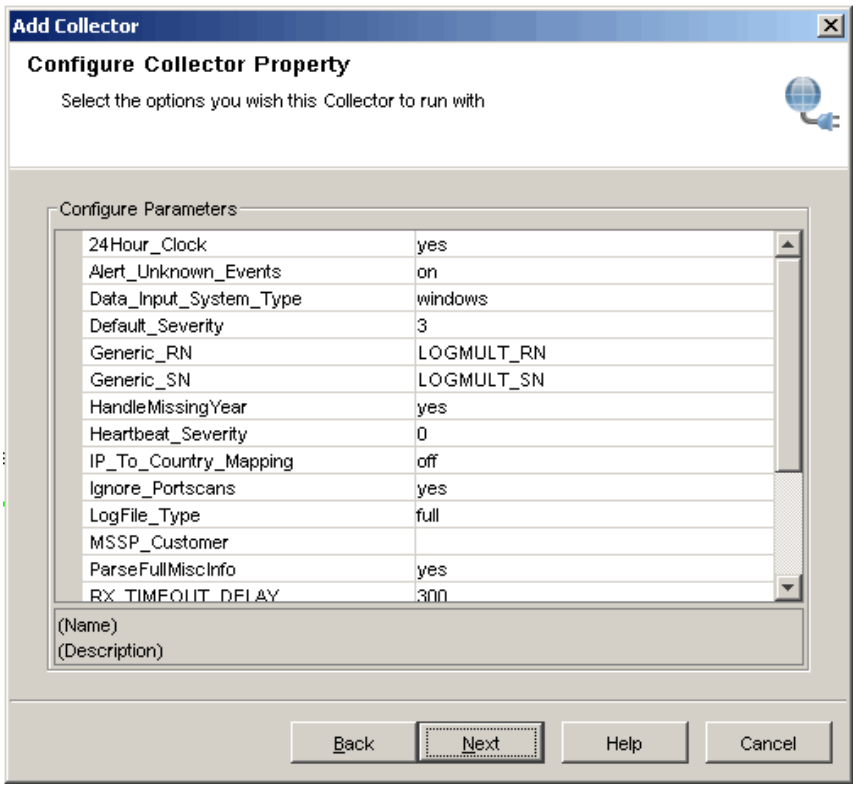
In Sentinel 6, this method of deploying two instances of a Collector is not supported. Only the first set of parameter values from the .par file will be recognized when the collector is imported, and that set will be used to set the default values for the collector. To achieve the same functionality in Sentinel 6, you must deploy the Collector twice, once with each set of parameter values.

To determine whether you have any Sentinel 5.x collectors that were using this configuration, you can search the Collector directories on the Collector Manager(s) for directories with more than one .chn file.

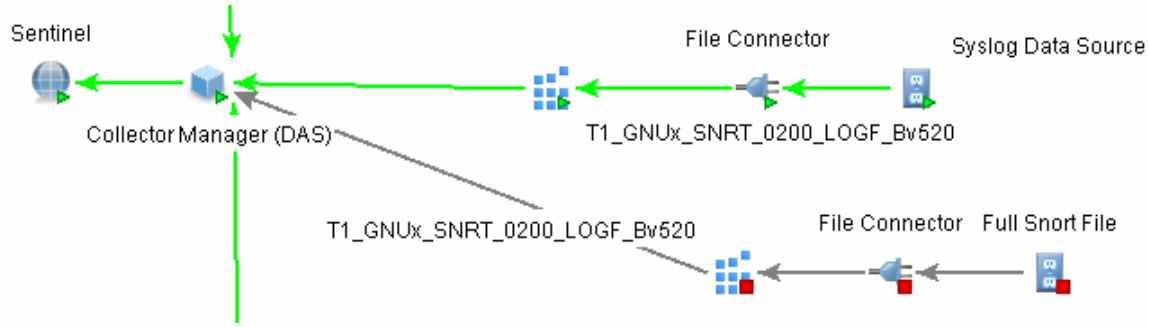
The instructions below assume general knowledge about how to import and deploy Collectors and Connectors in Sentinel 6. They also assume general knowledge about how to use Collector Builder. For more information about any of these topics, refer to the Additional Documentation section of this document.

To deploy two instances of a Collector:

1. Log into the Sentinel Control Center as a user with permission to deploy Collectors.
2. Go to Event Source Management > Live View.
3. Import the Collector using the steps in the *Sentinel User's Guide*.
4. Import the appropriate Connector, if it is not already in the Sentinel system.
5. Deploy two instances of the Collector, using the same imported Collector but setting the parameter values appropriately for each instance. The following screen shows one set of parameter values.



6. Configure one or more event sources, as appropriate for each of the two Collector instances. The final configuration for two instances of the Sentinel 5.x Snort collector, one created using the “full” configuration and one using the “syslog” configurations might look like this.



NOTE: The values used in Sentinel 5.x for the second instance of a deployed Collector may be viewed in Collector Builder on the Parameters screen.

Revision History

Revision 01

Initial Document

July 2007