NetWitness® Platform

Fluentd Integration and Configuration Guide

Fluentd

Event Source Product Information:

Vendor: CNCF

Event Source: Event sources supported by Fluentd

Versions: V1.18

NetWitness Product Information:

Supported On: NetWitness Platform 12.3 and later

Event Source Log Parser: Parser Specific to Event Sources to be Used

Collection Method:Using NetWitness Output Plugin for Fluentd

Event Source Class.Subclass: Configuration Management

Contact Information

NetWitness Community at https://community.netwitness.com contains a knowledge base that answers common questions and provides solutions to known problems, product documentation, community discussions, and case management.

Trademarks

RSA and other trademarks are trademarks of RSA Security LLC or its affiliates ("RSA"). For a list of RSA trademarks, go to https://www.rsa.com/en-us/company/rsa-trademarks. Other trademarks are trademarks of their respective owners.

License Agreement

This software and the associated documentation are proprietary and confidential to RSA Security LLC or its affiliates are furnished under license, and may be used and copied only in accordance with the terms of such license and with the inclusion of the copyright notice below. This software and the documentation, and any copies thereof, may not be provided or otherwise made available to any other person.

No title to or ownership of the software or documentation or any intellectual property rights thereto is hereby transferred. Any unauthorized use or reproduction of this software and the documentation may be subject to civil and/or criminal liability.

This software is subject to change without notice and should not be construed as a commitment by RSA.

Third-Party Licenses

This product may include software developed by parties other than RSA. The text of the license agreements applicable to third-party software in this product may be viewed on the product documentation page on NetWitness Community. By using this product, a user of this product agrees to be fully bound by terms of the license agreements.

Note on Encryption Technologies

This product may contain encryption technology. Many countries prohibit or restrict the use, import, or export of encryption technologies, and current use, import, and export regulations should be followed when using, importing or exporting this product.

Distribution

Use, copying, and distribution of any RSA Security LLC or its affiliates ("RSA") software described in this publication requires an applicable software license.

RSA believes the information in this publication is accurate as of its publication date. The information is subject to change without notice.

THE INFORMATION IN THIS PUBLICATION IS PROVIDED "AS IS." RSA MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WITH RESPECT TO THE INFORMATION IN THIS PUBLICATION, AND SPECIFICALLY DISCLAIMS IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Miscellaneous

This product, this software, the associated documentations as well as the contents are subject to NetWitness' standard Terms and Conditions in effect as of the issuance date of this documentation and which can be found at https://www.netwitness.com/standard-form-agreements/.

September 2024

Contents

Introduction

Fluentd is an open source data collector, which lets you unify the data collection and consumption for a better use and understanding of data. To collect events through fluentd and ship it to NewWitness via NW managed LogStash, we need to install output plugin “netwitnes” for fluentd. The output plugin “netwitness” for fluentd would collect events from fluentd eco system and forward them to NW managed LogStash.

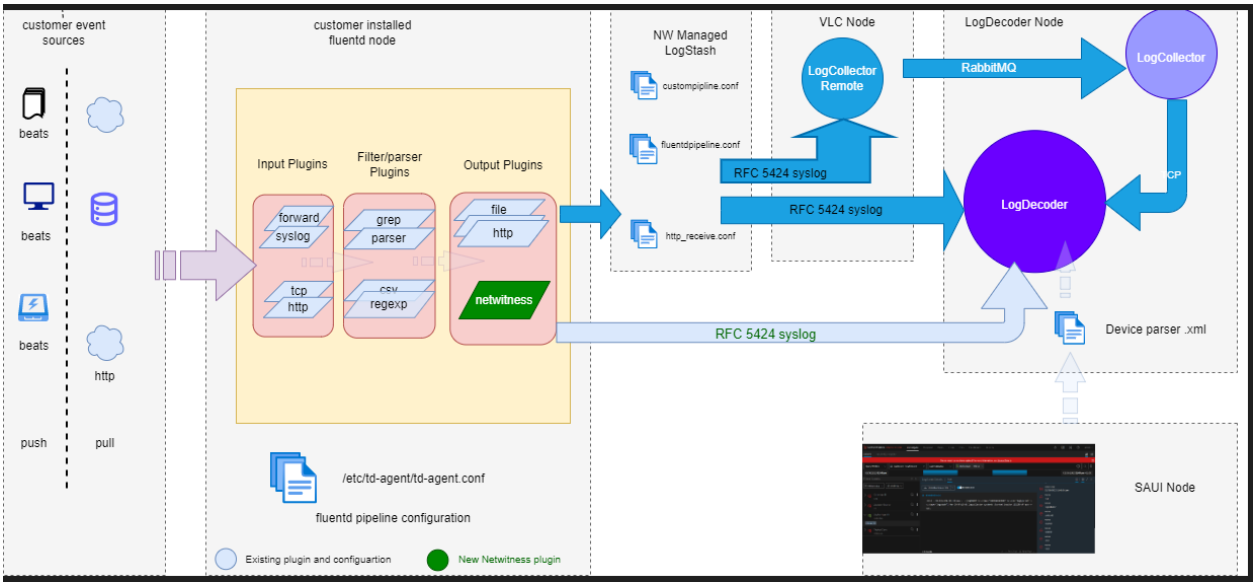
IMPORTANT: Fluentd Netwitness output plugin supports only TCP. UDP protocol is not supported.

NetWitness output plugin supports the following event flow:

- Fluentd → Netwitness output plugin → LogStash custom pipeline -> NW LogDecoder

Note: The following fluentd input plugin integrations are tested and supported:
- File
- Kafka
Other fluentd input plugins are supported, but additional proof of concepts are required to identify configurations for integration.

Fluentd NetWitness integration Architecture:



To forward the events from Fluentd to NetWitness, you must complete these tasks:

I. Install Fluentd

II. Install Netwitness Output Plugin for Fluentd Event Collection

III. Configure Fluentd

Install Fluentd

To install Fluentd:

Follow the steps mentioned in official fluentd portal to install fluentd https://docs.fluentd.org/installation. This step can be skipped if Fleuntd is already installed.

Install Netwitness Output Plugin for Fluentd Event Collection

To install NetWitness output plugin for Fluentd event collection:

Download the output plugin “netwitness” from Netwitness FluentD Output Plugin Download and run the following command on the host where fluent is installed.

```
fluent-gem install <<path>>/fluent-plugin-netwitness-1.0.2.gem
```

Configure Fluentd

To configure Fluentd:

Make configuration changes in the Fleuntd configuration file **/etc/fluent/fluentd.conf** in the host where Fluentd is installed. This enables Fluentd to collect the events and forward them to either NW LogStash.

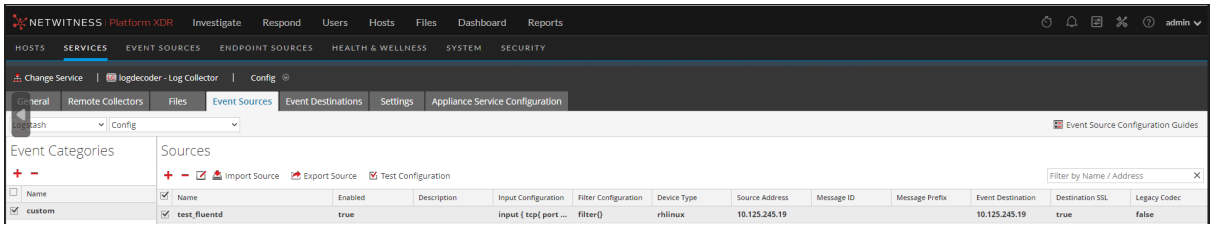
For illustration purposes, we have explained the mandatory input and output plugin configurations. Other plugins such as filter, parser and formatter can be referred from https://docs.fluentd.org/ and used based on need basis.

Ship Events to Log Decoder through LogStash Custom Pipeline

Create LogStash Custom Pipeline

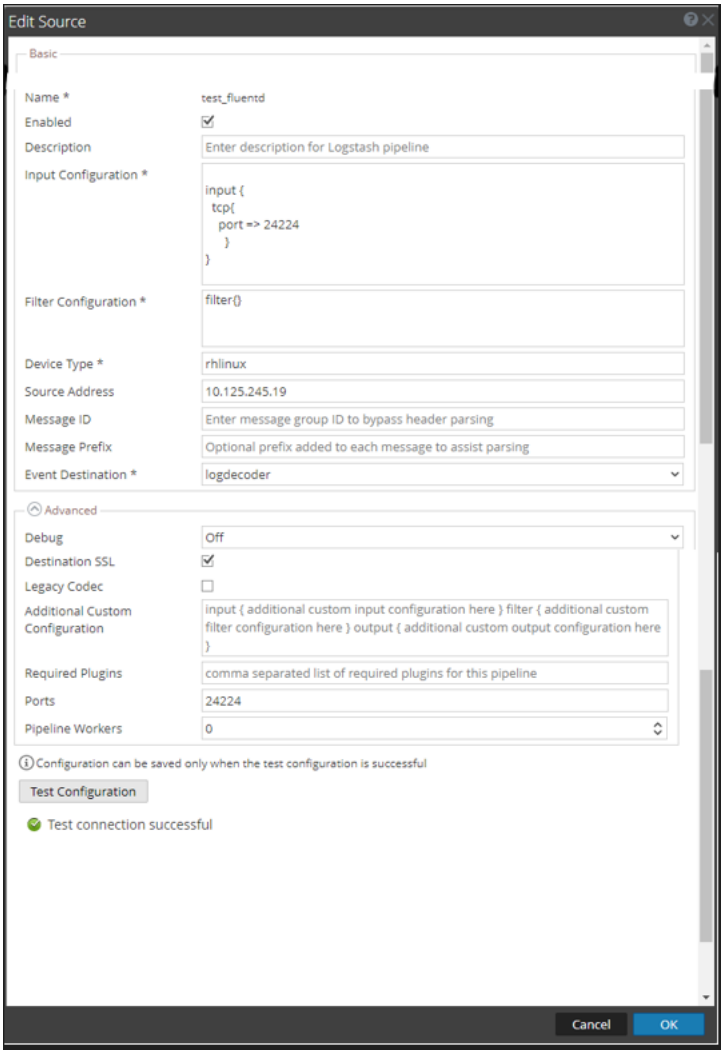
To create LogStash custom pipeline:

1. NetWitness Platform menu, select **Admin** > **Services**. The **Services** view is displayed.
2. In the **Services** grid, select a Log Collector service, and from the **Actions (** **)** menu, choose **View** > **Config**.
3. In the **Event Sources** view, select **Logstash** / **Config** from the drop-down menu.



4. In the **Event Categories** panel toolbar, click  . Select custom event category and create a pipeline by providing required details.

Sample custom logstash pipeline to collect event from Fluentd:



Input Plugin Configuration

Only for illustration purposes, we have used fluentd provided “tail” input plugin. Customer can use any other input plugin supported by fluentd as per their need. More details on supported input plugins and required configuration, refer https://docs.fluentd.org/input.

```
<source>
@type tail
path /var/log/messages
pos_file /var/log/fluent/httpd-access.log.pos
tag NwLogs_lc
<parse>
@type none
</parse>
</source>
```

Output Plugin Configuration

In order to transfer the events to NW managed LogStash and eventually to NW LD, we must use output plugin “netwitness” installed in Install Netwitness Output Plugin for Fluentd Event Collection. This plugin forwards events over TCP and supports both SSL and non SSL communications.

Non SSL communication

```
<match NwLogs_lc>
@type netwitness
host <<logstash host>>
port <<logstash port>>
use_ssl false
</match>
```

SSL Enabled Communication

Client Side Certificate Details for SSL:

```
<match NwLogs_lc>
@type netwitness
host <<Logstash host>>
ssl_port <<Logstash port>>
use_ssl true
ssl_cacert /ssl_cacert /var/log/fluent/truststore/org_ca.crt
ssl_key /ssl_key /var/log/fluent/truststore/client.key
ssl_cert /ssl_cert /var/log/fluent/truststore/client_combined.crt
</match>
```

Server Side Certificate Details for SSL:

```
input {
tcp{
port => 24224
ssl_enable=> true
ssl_cert=> "/etc/logstash/pki/logstash_combined.crt"
ssl_key => "/etc/logstash/pki/logstash.key"
ssl_certificate_authorities => "/etc/pki/tls/private/org_ca.crt"
}
}
```

Once config changes are made, restart the fluentd service using the following command:

```
systemctl restart fluentd.service
```

Note:
- The server side SSL certificate configuration should be done in the LogStash custom pipeline input configuration.
- By default, NW managed LogStash custom pipeline uses NetWitness codec and forwards the events in RFC – 5424 formats.
- Based on the event sources, select the Log Parsers to be deployed. If NW provided Log Parsers do not exist for any specific event type, then custom Log Parser can be developed and used.

Fluentd Output Configuration Parameters

Name	Description
source_ host	This configurable field is the identifier of the machine that originated the message. This will take either hostname or ip address as input. By default, this field is set to '-'.
use_ssl	This configurable field is used to enable SSL or TLS encryption for the connection between fluentd and a remote server. By default, the field is set to 'false'. If this field is not configured explicitly, the default value is considered and non-ssl communication is established. If this field is set to 'true', the ssl communication is established between fluentd and Logstash.
device_ type	This configurable filed is to identify the device or application that originated the message. This field will take either single device name (for example, rhlinux) or multiple comma separated device names (for example, rhlinux,oracle,kubernetes). If single name is provided, then the name will appear in the header. If multiple names with comma separated are provided, then the comma separated devices will be assigned to **lc.pn** in the structured data section. By default, the field is set to '-' .
message_id	This configurable field is to identify the type of message. This will take message ID as input. By default, the field is set to '-'.
include_ header_ time	This is a configurable Boolean field. By default, the value is set to ‘false’. If this field is set to ‘true’, then the UTC timestamp will appear in the header section of the RFC5424.
lc_cid	This configurable field is used to populate the log collection ID in the structured data section of RFC5424.
use5424	This is a configurable Boolean field. By default, the value is set to ‘false’. If the value is set to ‘true’, then the backend logic will generate the RFC5424 header and prepend it to the respective logs. If the value is set to ‘false’, then the plain logs will be forwarded to the desired destination without the RFC5424 header.

Getting Help with NetWitness Platform

Self-Help Resources

There are several options that provide you with help as you need it for installing and using NetWitness:

- See the documentation for all aspects of NetWitness here: https://community.netwitness.com/t5/netwitness-platform/ct-p/netwitness-documentation.
- Use the **Search** and **Create a Post** fields in NetWitness Community portal to find specific information here: https://community.netwitness.com/t5/netwitness-discussions/bd-p/netwitness-discussions.
- See the NetWitness Knowledge Base: https://community.netwitness.com/t5/netwitness-knowledge-base/tkb-p/netwitness-knowledge-base.
- See the documentation for Logstash JDBC input plugin here: https://www.elastic.co/guide/en/logstash/current/plugins-inputs-jdbc.html.
- See Troubleshooting section in the guides.
- See also NetWitness® Platform Blog Posts.
- If you need further assistance, Contact NetWitness Support.

Contact NetWitness Support

When you contact NetWitness Support, please provide the following information:

- The version number of the NetWitness Platform or application you are using.
- Logs information, even source version, and collection method.
- If you have problem with an event source, enable **Debug** parameter (set this parameter to **On** or **Verbose**) and collect the debug logs to share with the NetWitness Support team.

Use the following contact information if you have any questions or need assistance.

NetWitness Community Portal	https://community.netwitness.com In the main menu, click **Support > Case Portal > View My Cases**.
International Contacts (How to Contact NetWitness Support)	https://community.netwitness.com/t5/support/ct-p/support
Community	https://community.netwitness.com/t5/netwitness-discussions/bd-p/netwitness-discussions

Feedback on Product Documentation

You can send an email to feedbacknwdocs@netwitness.com to provide feedback on NetWitness Platform documentation.