# RSA NetWitness Platform

Event Source Log Configuration Guide

# Amazon GuardDuty

Last Modified: Tuesday, August 4, 2020

**Event Source Product Information:**

**Vendor**: Amazon
**Event Source**: GuardDuty
**Versions**: all

**RSA Product Information:**

**Supported On**: Security Analytics 10.6.2 and later
**Event Source Log Parser**: cef

**Note:** The CEF parser parses this event source as **device.type=amazonguardduty**.

**Collection Method**: Plugin Framework
**Event Source Class.Subclass**: Host.Cloud

To configure Amazon GuardDuty, you must complete these tasks:

I. Configure the Amazon GuardDuty event source

II. Set Up Amazon GuardDuty Event Source in RSA NetWitness

# Configure the Amazon GuardDuty Event Source

Amazon GuardDuty is a managed threat detection service that continuously monitors for malicious or unauthorized behavior to help you protect your AWS accounts and workloads. The service analyzes Amazon CloudTrail and AWS VPC Flow Log data to look for issues such as inbound port scans, possible backdoor access to your systems, unauthorized use of your account, and many other potential problems.

GuardDuty can be used to monitor a group of AWS accounts and have their findings routed to another AWS account—the master account—that is owned by a security team. Amazon GuardDuty starts to generate customized threat intelligence for you.

GuardDuty is a regional service. This means that when you enable GuardDuty in an AWS Region, all findings are generated and delivered in that region. Using the RSA NetWitness plugin framework, we can establish a connection with Amazon GuardDuty to provide visibility into the AWS Network.

**Note:** The GuardDuty plugin is meant for collecting the security alerts provided by AWS. The AWS GuardDuty alerts are sent in JSON format, as detailed in the AWS documentation here: https://docs.aws.amazon.com/guardduty/latest/ug/get-findings.html

## Enable the GuardDuty Service

To use GuardDuty, you must first enable it.

### To enable GuardDuty:

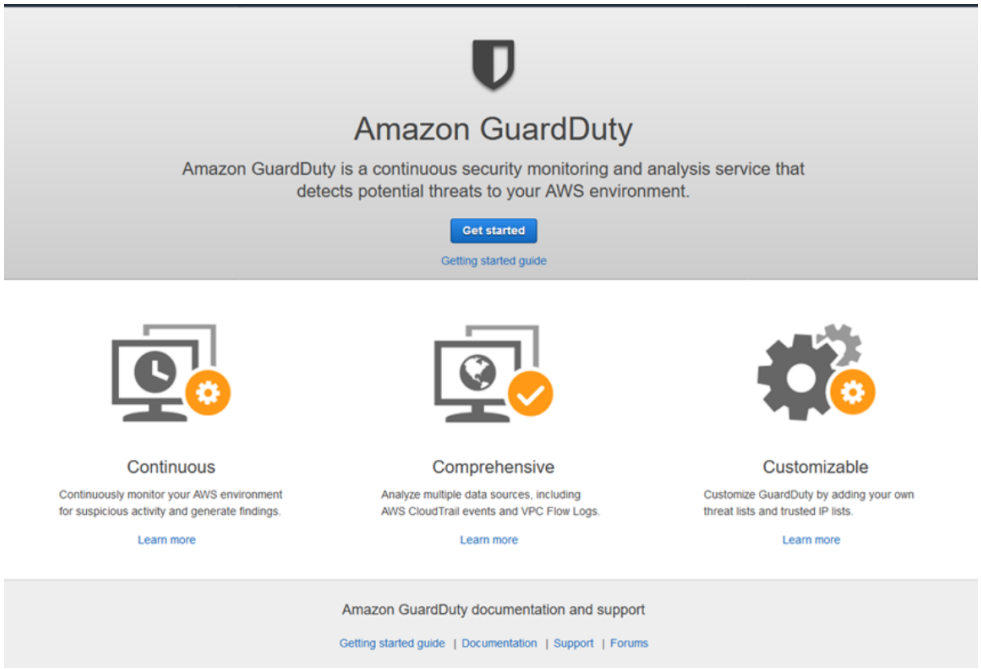
1. The IAM identity (user, role, group) that you use to enable GuardDuty must have the required permissions. To grant the permissions required to enable GuardDuty, attach the following policy to an IAM user, group, or role:

```
{
"Version": "2012-10-17",
"Statement": [
{
   "Effect": "Allow",
   "Action": ["guardduty:*"],
   "Resource": "*"
},
{
   "Effect": "Allow",
   "Action": ["iam:CreateServiceLinkedRole"],
   "Resource": "arn:aws:iam::123456789012:role/aws-service-
role/guardduty.amazonaws.com/AWSServiceRoleForAmazonGuardDuty",
   "Condition": {
      "StringLike": {"iam:AWSServiceName": "guardduty.amazonaws.com"}
   }
},
{
   "Effect": "Allow",
   "Action": ["iam:PutRolePolicy","iam:DeleteRolePolicy"],
   "Resource": "arn:aws:iam::1234567890123:role/aws-service-
role/guardduty.amazonaws.com/AWSServiceRoleForAmazonGuardDuty"
}
]
}
```

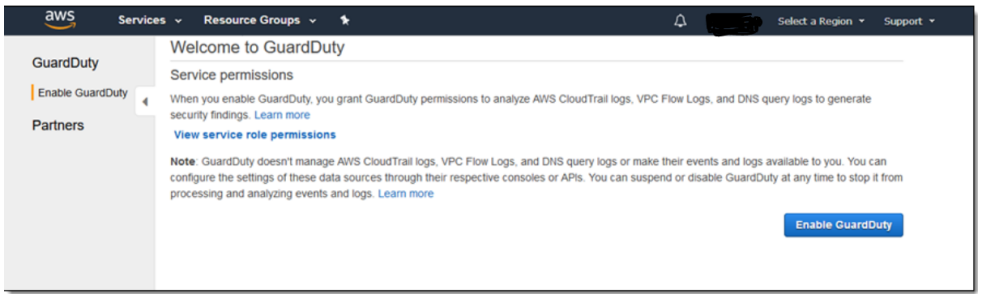
**Note:** Replace the sample account ID in the example above with your actual AWS account ID.

2. Use the credentials of the IAM identity from step 1 to sign in to the GuardDuty console.

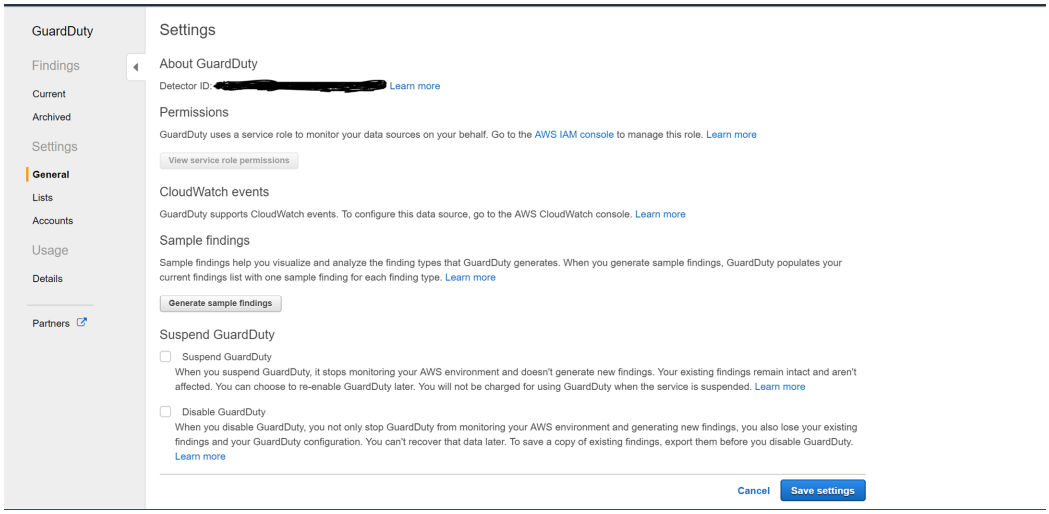
a. When you open the GuardDuty console for the first time, choose **Get Started**.




b. Then choose **Enable GuardDuty**.




3. In the GuardDuty console, note the value for the **Detector ID** parameter, which you will need later.


GuardDuty
Findings
Current
Archived
Settings
General
Lists
Accounts
Usage
Details
Partners
Settings
About GuardDuty
Detector ID: [redacted] Learn more
Permissions
GuardDuty uses a service role to monitor your data sources on your behalf. Go to the AWS IAM console to manage this role. Learn more
View service role permissions
CloudWatch events
GuardDuty supports CloudWatch events. To configure this data source, go to the AWS CloudWatch console. Learn more
Sample findings
Sample findings help you visualize and analyze the finding types that GuardDuty generates. When you generate sample findings, GuardDuty populates your current findings list with one sample finding for each finding type. Learn more
Generate sample findings
Suspend GuardDuty
Suspend GuardDuty
When you suspend GuardDuty, it stops monitoring your AWS environment and doesn't generate new findings. Your existing findings remain intact and aren't affected. You can choose to re-enable GuardDuty later. You will not be charged for using GuardDuty when the service is suspended. Learn more
Disable GuardDuty
When you disable GuardDuty, you not only stop GuardDuty from monitoring your AWS environment and generating new findings, you also lose your existing findings and your GuardDuty configuration. You can't recover that data later. To save a copy of existing findings, export them before you disable GuardDuty. Learn more
Cancel
Save settings

# Set Up the Amazon GuardDuty Event Source in NetWitness Platform

In RSA NetWitness Platform, perform the following tasks:

I. Deploy the **amazonguardduty** package and CEF parser from Live

II. Configure SELinux mode to Enforcing mode in the Log Decoder

III. Configure the event source.

## Deploy Amazon GuardDuty Files from Live

Amazon GuardDuty uses the cef parser.

### To deploy the cef parser from Live:

1. In the RSA NetWitness Platform menu, select **CONFIGURE**.

   The **Live Content** tab is displayed.

2. Browse Live Content for the **Common Event Format (cef)** parser, using **Log Device** as the **Resource Type**.

3. Select the **cef** parser from the list and click **Deploy** to deploy it to the appropriate Log Decoders, using the Deployment Wizard.

4. You also need to deploy the Amazon GuardDuty package. Browse Live for Amazon GuardDuty content, typing "Amazon GuardDuty" into the Keywords text box, then click **Search**.

5. Select the package and click **Deploy** to deploy it to the appropriate Log Collectors.

   **Note:** On a hybrid installation, you need to deploy the package on both the VLC and the LC.

6. Restart the **nwlogcollector** service.

For more details, see the Add or Update Supported Event Source Log Parsers topic, or the *Live Services Management Guide*.

## Configure SELinux mode to Enforcing mode in the Log Decoder

Run the script `update_selinux_policy.sh`, which is provided in the package, as a root user, after you deploy the package to the Log Decoder.

To enable the Enforcing mode for the SELinux, run the script on the Log Decoder:

```
sh
/etc/netwitness/ng/logcollection/content/collection/cmdscript/amaz
onvpc/update_selinux_policy.sh
```

**Note:** You only need to run this script once, during the initial configuration. Also, you do not need to run the script in NetWitness version 11.2 and later.

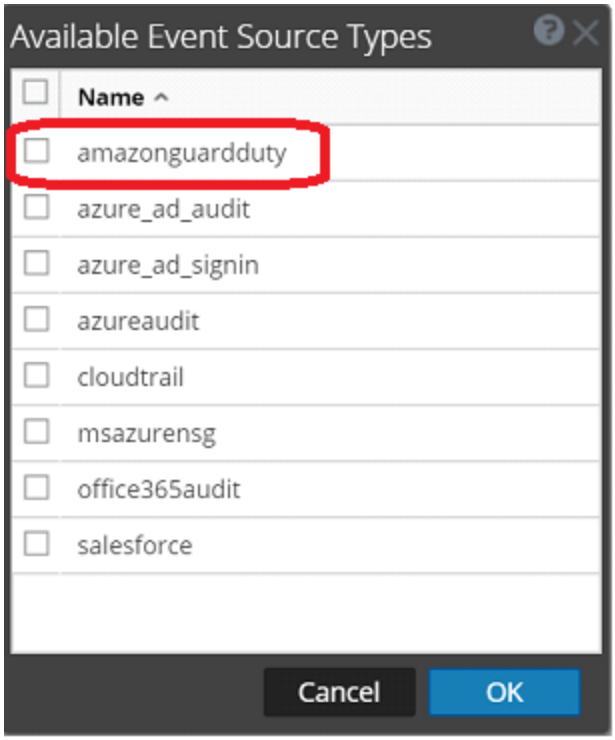
## Configure the Event Source

### To configure the Amazon GuardDuty Event Source:

1. In the RSA NetWitness Platform menu, select **ADMIN > Services**.
2. In the Services grid, select a Log Collector service, and from the Actions menu, choose **View** > **Config**.
3. In the **Event Sources** tab, select **Plugins/Config** from the drop-down menu.

   The Event Categories panel displays the File event sources that are configured, if any.
4. In the Event Categories panel toolbar, click +.

   The Available Event Source Types dialog is displayed.

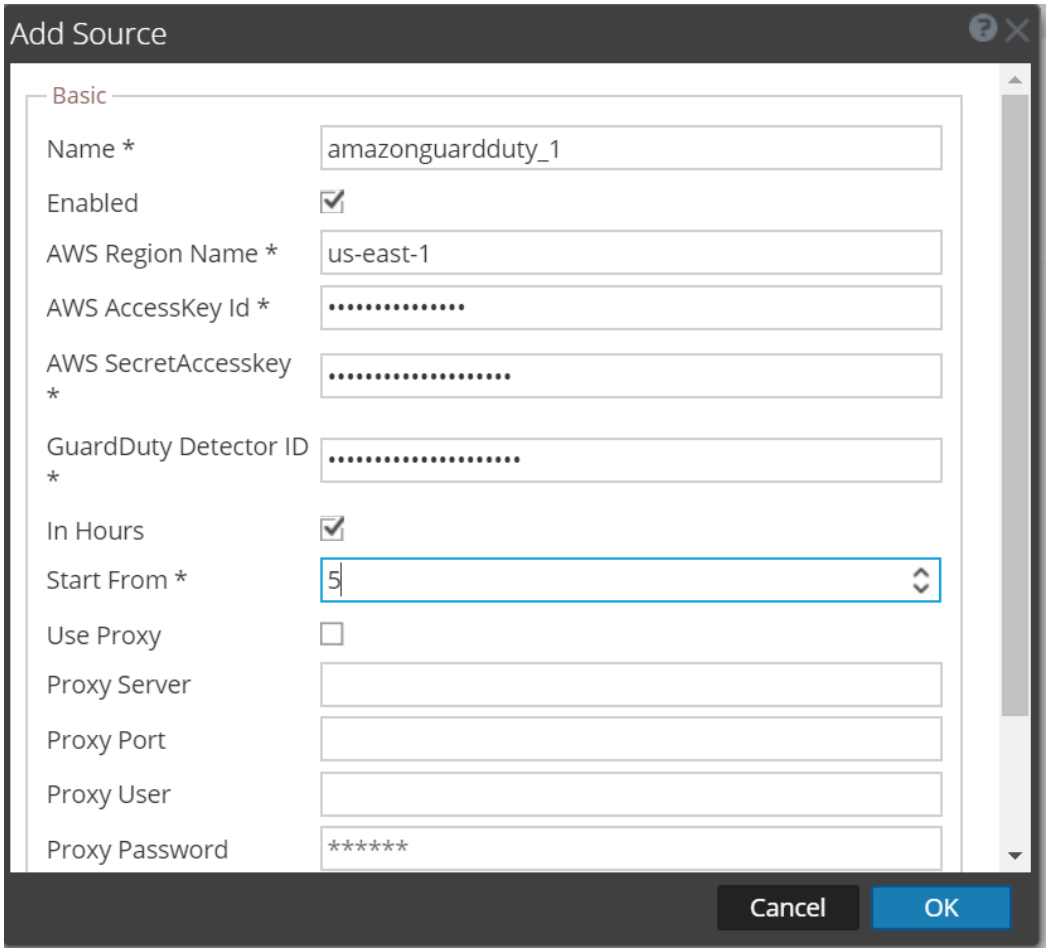



5. Select **amazonguardduty** from the list, and click **OK**.

   The newly added event source type is displayed in the Event Categories panel.

6. Select the new type in the Event Categories panel and click + in the Sources panel toolbar.

   The Add Source dialog is displayed.




7. Define parameter values, as described in Amazon GuardDuty Collection Configuration Parameters.

8. Click **Test Connection**.

   The result of the test is displayed in the dialog box. If the test is unsuccessful, edit the device or service information and retry.

   **Note:** The Log Collector takes approximately 60 seconds to return the test results. If it exceeds the time limit, the test times out and RSA NetWitness Platform displays an error message.

9. If the test is successful, click **OK**.

   The new event source is displayed in the Sources panel.

# Amazon GuardDuty Collection Configuration Parameters

The following table describes the configuration parameters for the Amazon GuardDuty integration with RSA NetWitness Platform. Fields marked with an asterisk (*) are required.

**Note:** When run from behind an SSL proxy, if certificate verification needs to be disabled, uncheck the **SSL Enable** checkbox in the Advanced section.

| Name | Description |
|---|---|
| Name * | Enter an alpha-numeric, descriptive name for the source. This value is only used for displaying the name on this screen. |
| Enabled | Select the box to enable the event source configuration to start collection. The box is selected by default. |
| AWS Region Name * | Name of the region where GuardDuty is enabled. |
| AWS Access Key Id * | Access key for the AWS account. |
| AWS Secret Access Key * | Secret access key for the AWS account. |
| GuardDuty detector ID | Detector ID for the enabled GuardDuty account. This ID was shown in step 3 of Enable the GuardDuty Service section. |
| In Hours | Specifies whether **Start From** represents number of hours or days.<br>• **Selected** (default): if selected, **Start From** represents number of hours.<br>• **Cleared**: if not checked, indicates **Start From** represents number of days. |
| Start From * | Specifies the number of hours or days (see the **In Hours** parameter above) prior to the current time, from which log collection should start. |

| Name | Description |
|---|---|
| Use Proxy | Check to enable proxy. |
| Proxy Server | If you are using a proxy, enter the proxy server address. |
| Proxy Port | Enter the proxy port. |
| Proxy User | Username for the proxy (leave empty if using anonymous proxy). |
| Proxy Password | Password for the proxy (leave empty if using anonymous proxy). |
| Source Address | Input the IP address that needs to appear as the **device.ip**. |

**Note:** Please avoid using special characters in the **Proxy User** and **Proxy Password** sections.



## Trademarks

For a list of RSA trademarks, go to www.emc.com/legal/emc-corporation-trademarks.htm#rsa.