How to Configure Log Streaming to AWS CloudWatch

https://campus.barracuda.com/doc/79463440/

To stream log data from your firewall to AWS CloudWatch, you must configure AWS Cloud Integration and configure syslog streaming on the firewall. The IAM role assigned to the firewall instance must include an IAM policy allowing the firewall instance access to AWS CloudWatch. Configure syslog streaming with AWS CloudWatch as the destination. The configured log group is automatically created, and the logs are placed into a folder using either the instance ID or the hostname as the name. No additional configuration is required for AWS CloudWatch to collect the following metrics:

Custom VPN Metrics

- Client-to-site VPN tunnels
- SSL VPN clients
- Site-to-site VPN tunnels up
- Site-to-site VPN tunnels down

Custom System Metrics

- Load
- Used memory
- Protected IPs

Custom Firewall Metrics

- Bytes in
- Bytes out
- Bytes total
- Packets in
- Packets out
- Packets total
- Connections dropped
- IPS Hits
- Forwarding Connections new
- Forwarding Connections total
- Connections new
- Connections total
- Connections blocked
- Connections failed

Before You Begin

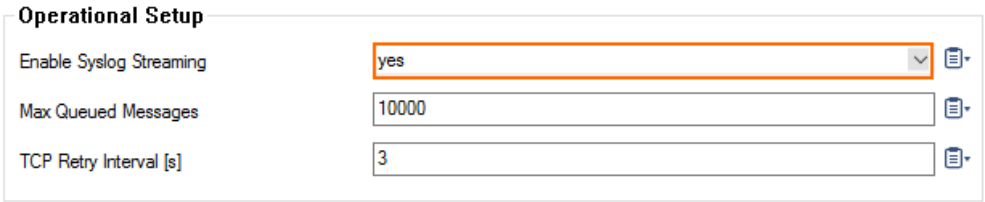
The firewall must be deployed with an IAM role that allows access to AWS CloudWatch. For more information, see How to Create an IAM Role for a CloudGen Firewall in AWS.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogGroup",
                "logs:CreateLogStream",
                "logs:PutLogEvents",
                "logs:DescribeLogStreams",
                "logs:DescribeLogGroups"
            ],
            "Resource": [
                "arn:aws:logs:*:*:*"
            ]
        }
    ]
}
```

Step 1. Enable Syslog Streaming

Enable syslog streaming and, optionally, configure the AWS region if it is different from the region of the firewall instance.

1. Go to **CONFIGURATION > Full Configuration > Box > Infrastructure Services > Syslog Streaming**.
2. Click **Lock**.
3. Set **Enable Syslog Streaming** to **yes**.



4. In the left menu, expand the **Configuration Mode** section and click **Switch to Advanced View**.
5. Enter the AWS CloudWatch region. E.g., `eu-west-1`
6. Click **Send Changes** and **Activate**.

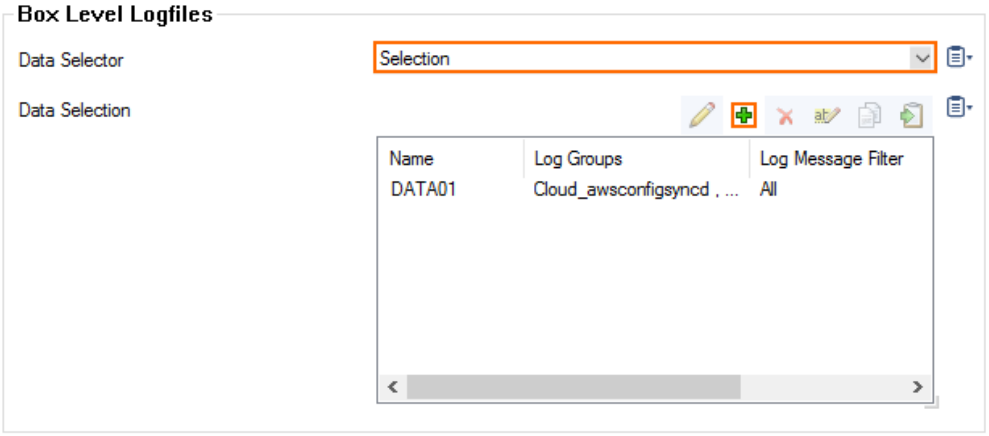
Step 2. Configure Logdata Filters

Define profiles specifying the log file types to be transferred / streamed. Log file are classified into top level, box level, and service level log data sources.

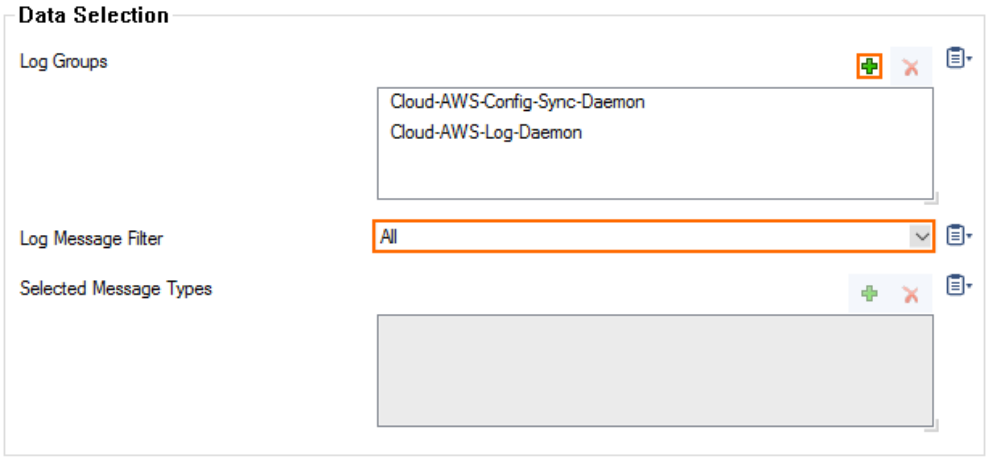
1. Go to **CONFIGURATION > Full Configuration > Box > Infrastructure Services > Syslog Streaming**.
2. In the left menu, select **Logdata Filters**.
3. Click **Lock**.
4. In the **Filters** table, click **+** to add a new filter. The **Filters** window opens.
5. Enter a **Name**.
6. Click **OK**.
7. In the **Data Selection** table, add the **Top Level Log Files** log files to be streamed. You can select:
   - **Fatal_log**
   - **Firewall_Audit_Log** – The firewall audit log must be enabled and configured, and **Audit Delivery** must be set to **Syslog Proxy**. For more information, see How to Enable the Firewall Audit Log Service. Alternatively, the firewall audit log can also be streamed as a part of the firewall service logs.
   - **Panic_log**



8. Configure the **Box Level Logfile** filters:
   1. From the **Data Selector** list, select which files for this category are streamed:
      - **All** – All box level logs are streamed.
      - **None** – Box level logs are not streamed.
      - **Selection** – Only box level log files defined in the **Data Selection** list are streamed.

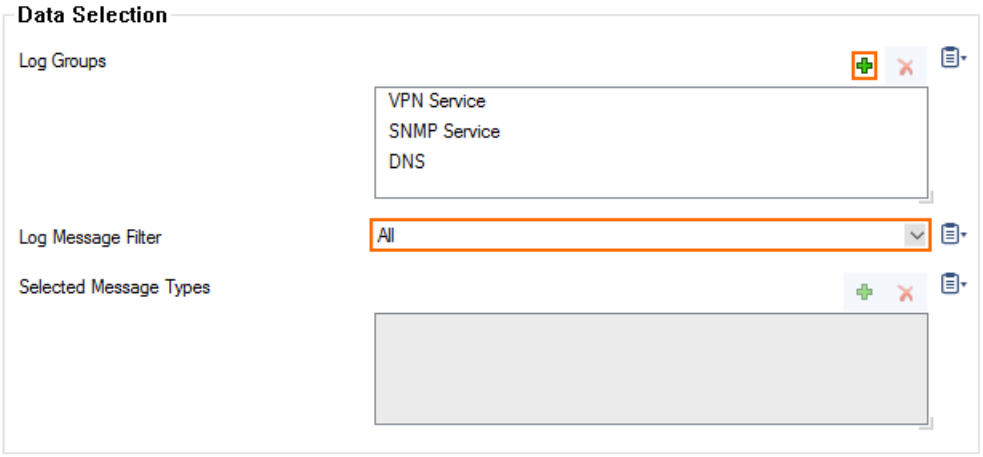


2. (**Selection** only) Click **+** to add custom filters to the **Data Selection** table.
    1. In the **Log Groups** table, click **+**.
    2. Select the box level log files, or select **Other** to enter a **user defined log group pattern** to stream log files matching this pattern.
    3. (optional) From the **Log Level Filter** list, select the message types from the log group that are streamed.
    4. (**Selection** only) In the **Selected Messages Types** table, click **+** to add message types.



9. Configure the **Service Level Logfile** filters:
    1. From the **Data Selector** list, select which files for this category are streamed:
        - **All** – All service logs are streamed.
        - **None** – Service level logs are not streamed.
        - **Selection** – Only service level log files defined in the **Data Selection** list are streamed.
    2. (**Selection** only) Click **+** to add custom filters to the **Data Selection** table.
        1. In the **Log Groups** table, click **+**.
        2. Select the box level log files, or select **Other** to enter a **user defined log group pattern** to stream log files matching this pattern.
        3. (optional) From the **Log Level Filter** list, select the message types from the log group that are streamed.
        4. (**Selection** only) In the **Selected Messages Types** table, click **+** to add message types.

5. Click **OK**.

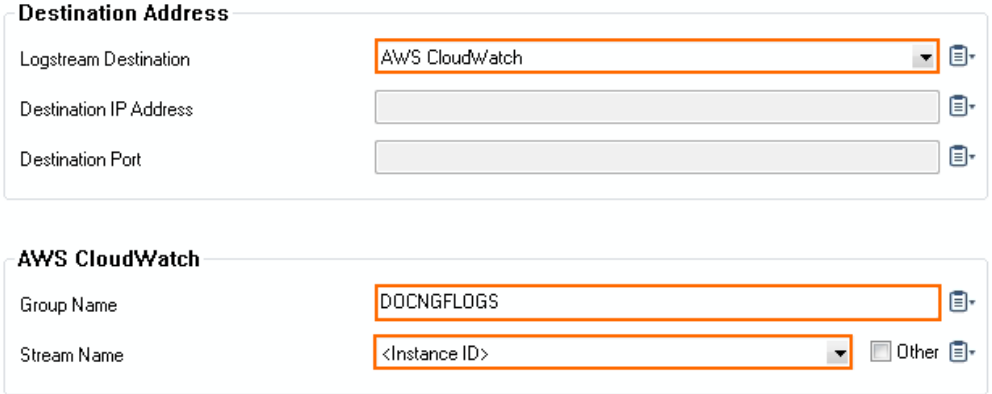


10. Click **Send Changes** and **Activate**.

Step 3. Configure AWS CloudWatch as the Logstream Destination

Configure the firewall to send the syslog stream to AWS CloudWatch. The AWS CloudWatch log group name is created automatically, with one stream per firewall.

1. Go to **CONFIGURATION > Full Configuration > Box > Infrastructure Services > Syslog Streaming**.
2. In the left menu, select **Logstream Destinations**.
3. Click **Lock**.
4. In the **Destinations** table, click **+** to add a new filter. The **Destinations** window opens.
5. Enter a **Name**.
6. Click **OK**.
7. From the **Logstream Destination** list, select **AWS CloudWatch**.
8. In the **AWS CloudWatch** section, enter the name of the AWS CloudWatch log **Group Name**.
9. (optional) Select the **Stream Name** from the drop-down list, or select **Other** and enter the stream name. The stream name must be unique in the AWS CloudWatch group.



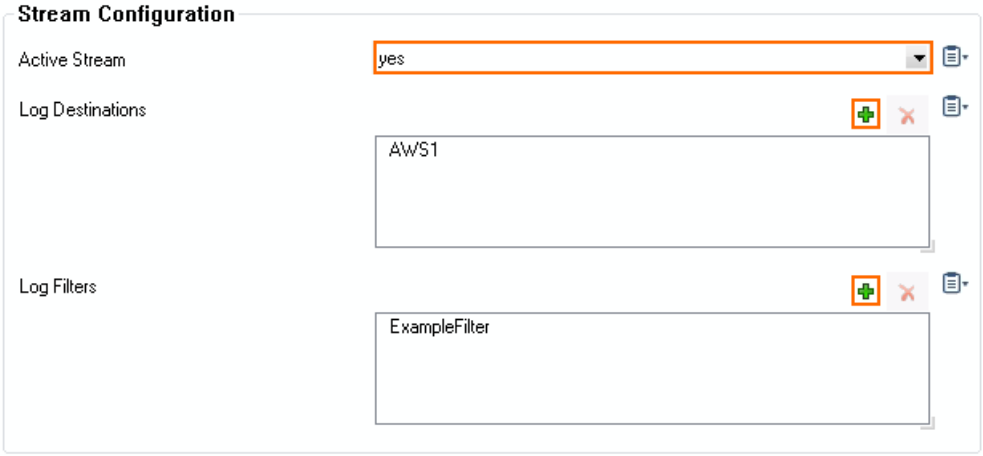
10. Click **OK**.

11. Click **Send Changes** and **Activate**.

Step 4. Configure the Logdata Streams to AWS CloudWatch

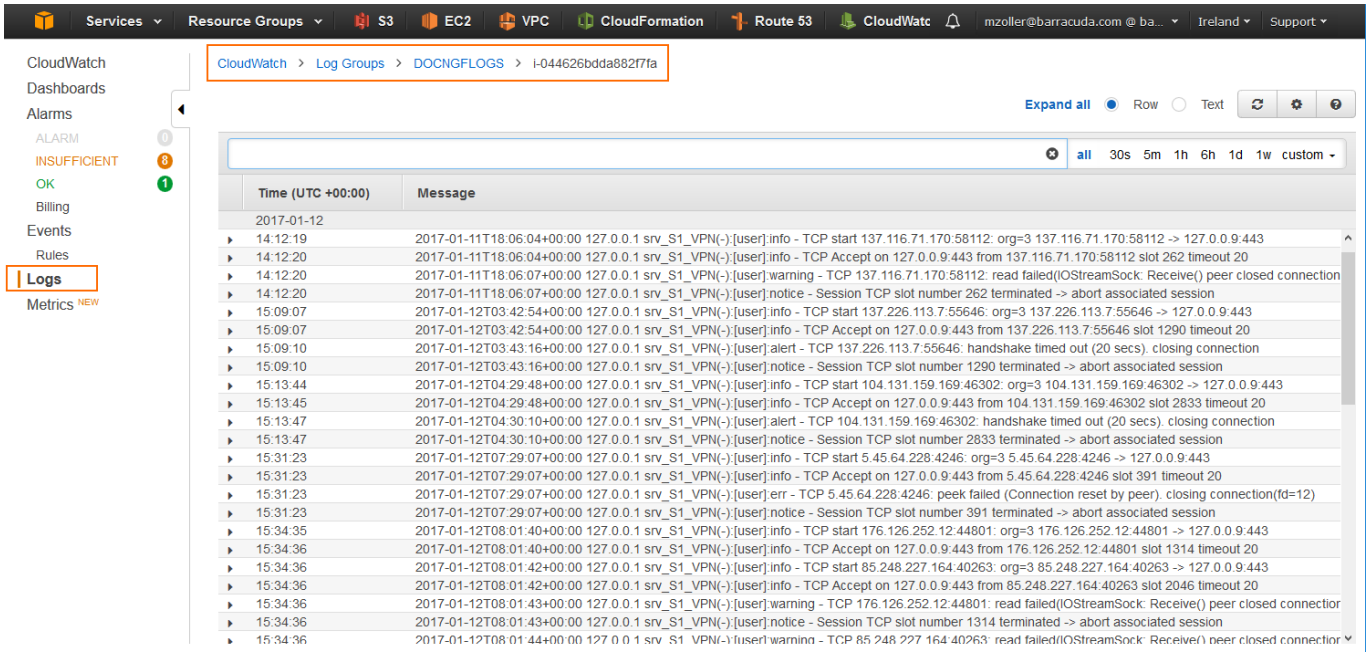
Combine the logdata filters and logstream destination to a logdata stream.

1. Go to **CONFIGURATION > Full Configuration > Box > Infrastructure Services > Syslog Streaming**.
2. In the left menu, select **Logdata Streams**.
3. Click **Lock**.
4. In the **Streams** table, click **+** to add a new syslog stream. The **Streams** window opens.
5. Enter a **Name**.
6. Click **OK**.
7. Set **Active Stream** to **yes**.
8. In the **Log Destinations** table, click **+** and select the logstream destination configured in step 3.
9. In the **Log Filters** table, click **+** and select the logdata filter configured in step 2.



10. Click **OK**.
11. Click **Send Changes** and **Activate**.

All logs covered by the logdata filter are now streamed to AWS CloudWatch. It might take up to 30 minutes for logs to be displayed in the console.


Services
Resource Groups
S3
EC2
VPC
CloudFormation
Route 53
CloudWatc
mzoller@barracuda.com @ ba...
Ireland
Support
CloudWatch
Dashboards
Alarms
ALARM 0
INSUFFICIENT 8
OK 1
Billing
Events
Rules
Logs
Metrics NEW
CloudWatch > Log Groups > DOCNGFLOGS > i-044626bdda882f7fa
Expand all
Row
Text
all 30s 5m 1h 6h 1d 1w custom
Time (UTC +00:00)
Message
2017-01-12
14:12:19 2017-01-11T18:06:04+00:00 127.0.0.1 srv_S1_VPN(-):[user]:info - TCP start 137.116.71.170:58112: org=3 137.116.71.170:58112 -> 127.0.0.9:443
14:12:20 2017-01-11T18:06:04+00:00 127.0.0.1 srv_S1_VPN(-):[user]:info - TCP Accept on 127.0.0.9:443 from 137.116.71.170:58112 slot 262 timeout 20
14:12:20 2017-01-11T18:06:07+00:00 127.0.0.1 srv_S1_VPN(-):[user]:warning - TCP 137.116.71.170:58112: read failed(IOStreamSock: Receive() peer closed connection
14:12:20 2017-01-11T18:06:07+00:00 127.0.0.1 srv_S1_VPN(-):[user]:notice - Session TCP slot number 262 terminated -> abort associated session
15:09:07 2017-01-12T03:42:54+00:00 127.0.0.1 srv_S1_VPN(-):[user]:info - TCP start 137.226.113.7:55646: org=3 137.226.113.7:55646 -> 127.0.0.9:443
15:09:07 2017-01-12T03:42:54+00:00 127.0.0.1 srv_S1_VPN(-):[user]:info - TCP Accept on 127.0.0.9:443 from 137.226.113.7:55646 slot 1290 timeout 20
15:09:10 2017-01-12T03:43:16+00:00 127.0.0.1 srv_S1_VPN(-):[user]:alert - TCP 137.226.113.7:55646: handshake timed out (20 secs). closing connection
15:09:10 2017-01-12T03:43:16+00:00 127.0.0.1 srv_S1_VPN(-):[user]:notice - Session TCP slot number 1290 terminated -> abort associated session
15:13:44 2017-01-12T04:29:48+00:00 127.0.0.1 srv_S1_VPN(-):[user]:info - TCP start 104.131.159.169:46302: org=3 104.131.159.169:46302 -> 127.0.0.9:443
15:13:45 2017-01-12T04:29:48+00:00 127.0.0.1 srv_S1_VPN(-):[user]:info - TCP Accept on 127.0.0.9:443 from 104.131.159.169:46302 slot 2833 timeout 20
15:13:47 2017-01-12T04:30:10+00:00 127.0.0.1 srv_S1_VPN(-):[user]:alert - TCP 104.131.159.169:46302: handshake timed out (20 secs). closing connection
15:13:47 2017-01-12T04:30:10+00:00 127.0.0.1 srv_S1_VPN(-):[user]:notice - Session TCP slot number 2833 terminated -> abort associated session
15:31:23 2017-01-12T07:29:07+00:00 127.0.0.1 srv_S1_VPN(-):[user]:info - TCP start 5.45.64.228:4246: org=3 5.45.64.228:4246 -> 127.0.0.9:443
15:31:23 2017-01-12T07:29:07+00:00 127.0.0.1 srv_S1_VPN(-):[user]:info - TCP Accept on 127.0.0.9:443 from 5.45.64.228:4246 slot 391 timeout 20
15:31:23 2017-01-12T07:29:07+00:00 127.0.0.1 srv_S1_VPN(-):[user]:err - TCP 5.45.64.228:4246: peek failed (Connection reset by peer). closing connection(fd=12)
15:31:23 2017-01-12T07:29:07+00:00 127.0.0.1 srv_S1_VPN(-):[user]:notice - Session TCP slot number 391 terminated -> abort associated session
15:34:35 2017-01-12T08:01:40+00:00 127.0.0.1 srv_S1_VPN(-):[user]:info - TCP start 176.126.252.12:44801: org=3 176.126.252.12:44801 -> 127.0.0.9:443
15:34:36 2017-01-12T08:01:40+00:00 127.0.0.1 srv_S1_VPN(-):[user]:info - TCP Accept on 127.0.0.9:443 from 176.126.252.12:44801 slot 1314 timeout 20
15:34:36 2017-01-12T08:01:42+00:00 127.0.0.1 srv_S1_VPN(-):[user]:info - TCP start 85.248.227.164:40263: org=3 85.248.227.164:40263 -> 127.0.0.9:443
15:34:36 2017-01-12T08:01:42+00:00 127.0.0.1 srv_S1_VPN(-):[user]:info - TCP Accept on 127.0.0.9:443 from 85.248.227.164:40263 slot 2046 timeout 20
15:34:36 2017-01-12T08:01:43+00:00 127.0.0.1 srv_S1_VPN(-):[user]:warning - TCP 176.126.252.12:44801: read failed(IOStreamSock: Receive() peer closed connectior
15:34:36 2017-01-12T08:01:43+00:00 127.0.0.1 srv_S1_VPN(-):[user]:notice - Session TCP slot number 1314 terminated -> abort associated session

Figures

1. cloudwatch_01.png
2. cloudwatch_02.png
3. cloudwatch_03.png
4. cloudwatch_03a.png
5. cloudwatch_04.png
6. cloudwatch_05.png
7. cloudwatch_06.png
8. cloudwatch_07.png

© Barracuda Networks Inc., 2024 The information contained within this document is confidential and proprietary to Barracuda Networks Inc. No portion of this document may be copied, distributed, publicized or used for other than internal documentary purposes without the written consent of an official representative of Barracuda Networks Inc. All specifications are subject to change without notice. Barracuda Networks Inc. assumes no responsibility for any inaccuracies in this document. Barracuda Networks Inc. reserves the right to change, modify, transfer, or otherwise revise this publication without notice.