Collecting and Analyzing Logs with Amazon CloudWatch Logs Insights

**SPL-232 - Version 1.0.12**

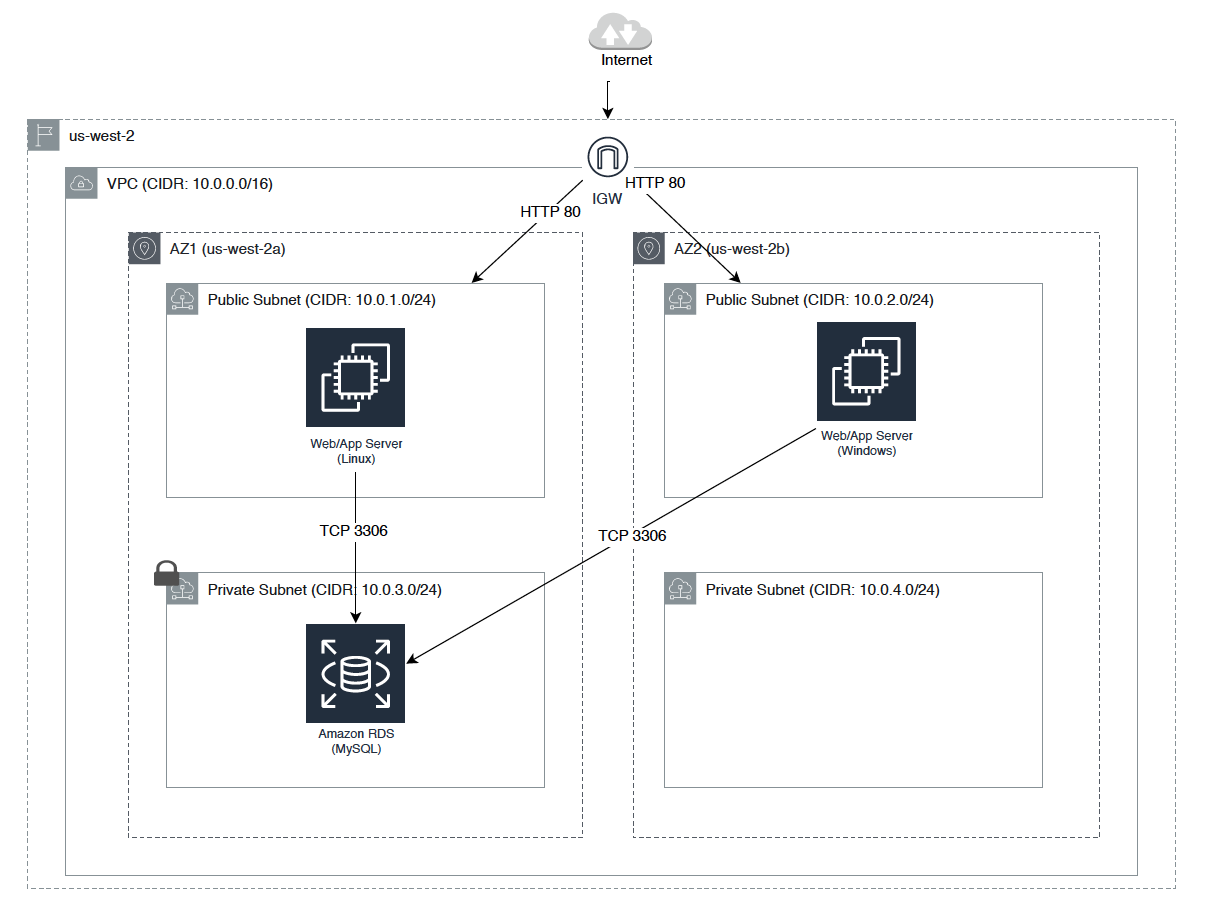
© 2024 Amazon Web Services, Inc. or its affiliates. All rights reserved. This work may not be reproduced or redistributed, in whole or in part, without prior written permission from Amazon Web Services, Inc. Commercial copying, lending, or selling is prohibited. All trademarks are the property of their owners.

Note: Do not include any personal, identifying, or confidential information into the lab environment. Information entered may be visible to others.

Corrections, feedback, or other questions? Contact us at [*AWS Training and Certification*](https://support.aws.amazon.com/#/contacts/aws-training).

**Lab overview**

This lab mimics a typical web application built around n-tier architecture, which is a proven and suitable architecture in ecommerce applications. N-tier architecture helps developers to create flexible and reusable applications. This lab covers the security topic by looking for various log streams to inspect any potential security vulnerabilities. The main assets in this lab are two Amazon Elastic Compute Cloud (Amazon EC2) instances (Linux and Windows) to host a website, an Amazon Relational Database Service (Amazon RDS) (with MySQL engine) database instance, two public subnets, a private subnet, and an Amazon Virtual Private Cloud (Amazon VPC). The below diagram explains the architecture of the lab.



**Note:** Once you choose **Start Lab** provisioning of the RDS and EC2 instances takes between 15 and 20 minutes. While booting up this lab, grab a cup of coffee, relax, and browse through the AWS Systems Manager and Amazon CloudWatch documentation.

OBJECTIVES

By the end of this lab, you will be able to do the following:

* Install and configure CloudWatch Agents on the EC2 instances using Systems Manager
* Collect web server and system logs from an EC2 instance and publish them over to CloudWatch Logs
* Enable Amazon VPC Flow Logs across VPCs and subnets
* Enable CloudWatch Logs for an Amazon RDS database instance
* Explore logs using CloudWatch Logs Insights
* Inspect logs using CloudWatch Logs Insights and CloudWatch Metrics

TECHNICAL KNOWLEDGE PREREQUISITES

To successfully complete this lab, you should be familiar with services like Amazon EC2, CloudWatch, AWS CloudTrail, Amazon VPC, and Amazon RDS. You should also be familiar with using the AWS Management Console.

**Start lab**

1. To launch the lab, at the top of the page, choose **Start lab**.

**Caution:** You must wait for the provisioned AWS services to be ready before you can continue.

1. To open the lab, choose **Open Console**.

You are automatically signed in to the AWS Management Console in a new web browser tab.

**WARNING:** **Do not change the Region unless instructed.**

COMMON SIGN-IN ERRORS

**Error: You must first sign out**



If you see the message, **You must first log out before logging into a different AWS account:**

* Choose the **click here** link.
* Close your **Amazon Web Services Sign In** web browser tab and return to your initial lab page.
* Choose **Open Console** again.

**Error: Choosing Start Lab has no effect**

In some cases, certain pop-up or script blocker web browser extensions might prevent the **Start Lab** button from working as intended. If you experience an issue starting the lab:

* Add the lab domain name to your pop-up or script blocker’s allow list or turn it off.
* Refresh the page and try again.

**Task 1: Setting up instance logging agents using AWS Systems Manager**

This task involves setting up CloudWatch Agents on Amazon EC2 instances using the Systems Manager.

**Additional information:** What is AWS Systems Manager? Systems Manager is a collection of capabilities for configuring and managing your Amazon EC2 instances, on-premises servers and virtual machines (VMs), and certain other AWS resources. The service includes a unified interface that lets you easily centralize operational data and automate tasks across your AWS resources. Systems Manager shortens the time to detect and resolve operational problems in your infrastructure. The service gives you a complete view of your infrastructure performance and configuration, simplifies resource and application management, and makes it easy to operate and manage your infrastructure at scale.

You will use the Systems Manager Parameter Store to store the agent configuration and install the CloudWatch Agents. These agents collect metrics and logs from the EC2 instances and send them to CloudWatch Logs for further analysis.

ASSIGN THE SYSTEMS MANAGER IAM POLICIES TO EC2 INSTANCES

To use the Systems Manager service, first ensure the EC2 instances are attached with the correct Simple Systems Manager (SSM) policies through an AWS Identity and Access Management (IAM) role. An IAM role defines a set of permissions for making AWS service requests. IAM roles are not associated with a specific user or group. Instead, trusted entities, such as IAM users, applications, or AWS services (for example, EC2), assume roles. In this case, the two EC2 instances assume a role that was created when you launched this tab. Ensure the role is attached to the both EC2 instances.

1. At the top of the AWS Management Console, in the search bar, search for and choose

EC2

.

1. In the left navigation pane, choose **Instances**, and select **LinuxEC2Host**.
2. Choose **Actions** > **Security** > **Modify IAM Role**.
3. You will notice the role is already attached to the instance.
4. Choose **Cancel**.
5. Repeat these steps for the **WindowsEC2Host** instance.

TEST THE WEBSITE ON BOTH LINUX AND WINDOWS SERVERS

Before proceeding to the next step, check out the webpages deployed on both EC2 instances.

1. In the left navigation pane of this lab page, copy the URL from **LinuxWebsiteURL**.
2. Open a new browser tab, and paste the URL in the address bar. A simple web package displays.
3. In the left navigation pane of this lab page, copy the URL from **WindowsWebsiteURL**.
4. Open a new browser tab, and paste the URL in the address bar. A default Internet Information Services (IIS) webpage displays.

Later, you will explore the web server logs from these instances and look for different browsing patterns.

INSTALL CLOUDWATCH AGENTS ON EC2 INSTANCES

1. At the top of the AWS Management Console, in the search bar, search for and choose

Systems Manager

.

1. In the left-hand navigation pane, choose **Run Command**.
2. Choose **Run a Command**.
3. For **Command document**, select **AWS-ConfigureAWSPackage**.

**Note:** If the command is not on the first page, search for the command OR use the arrows at the top-right to move to the next page.

1. For **Action**: Select **Install**.
2. For **Name**: Enter

AmazonCloudWatchAgent

1. For **Targets**: Select **Specify instance tags**.
2. For **Tag key**: Enter

CloudwatchAgentInstalled

1. For **Tag value**: Enter

true

1. Choose **Add**.

**Note:** In complex production environments with thousands of instances, the recommended approach is to select **Specifying a tag**. In this approach, you give the same tag name for each resource, and SSM systematically applies the configuration for every resource identified by that tag.

1. For **Output options**, configure:

* Select **Enable an S3 bucket**.
* Select **Choose an S3 bucket name from the list**.
* From the dropdown, select the bucket name that starts with **-xxxxx-logbucket-xxxxx** and matches **LogBucket** in the left-hand navigation. These logs are useful for troubleshooting any unexpected errors.

1. Choose **Run**.

**Note:** Within 30 seconds, the status of the instances should change to *Success*. If the status does not change, choose .

START CLOUDWATCH AGENTS ON EC2 INSTANCES

Log configuration details are stored in the Systems Manager Parameter store at the start of the lab. See the configuration details by choosing **Parameter Store** in the Systems Manager. The configuration data is in JavaScript Object Notation (JSON).

1. In the left-hand navigation pane, choose **Run Command**.
2. Choose **Run command**.
3. For **Command document**, select **AmazonCloudWatch-ManageAgent**.

**Note:** If the command is not on the first page, search for the command OR use the arrows at the top-right to move to the next page.

1. For **Optional Configuration Location**: Enter

ConfigParamForEC2s

**Note:** This field is required.

1. For **Targets**: Select **Specify instance tags**.
2. For **Tag key**: Enter

MonitoringConfig

1. For **Tag value**: Enter

BasicMixOSMonitoring

1. Choose **Add**.
2. For **Output options**, configure:

* Select **Enable an S3 bucket**.
* Select **Choose an S3 bucket name from the list**.
* From the dropdown, select the bucket name that starts with **-xxxxx-logbucket-xxxx** and matches **LogBucket** in the left-hand navigation.

1. Choose **Run**.

**Note:** Within 30 seconds, the status of the instances should change to *Success*. If the status does not change, choose .

**Task 2: Network logging**

To capture different types of unauthorized network activities, enable Amazon VPC Flow Logs. This feature enables you to capture information about the IP traffic going to and from network interfaces in your VPC. Flow log data can be published to CloudWatch Logs and the Amazon Simple Storage Service (Amazon S3). After you’ve created a flow log, you can retrieve and view its data in the chosen destination. Flow logs can help you with a number of tasks; for example, to troubleshoot why specific traffic is not reaching an instance or a database, which in turn helps you diagnose overly restrictive security group rules. You can also use flow logs as a security tool to monitor the traffic that is reaching your instance.

ENABLE FLOW LOGS AT SUBNET LEVEL

 You can create a flow log for a VPC, subnet, or network interface. If you create a flow log for a subnet or VPC, each network interface in the VPC or subnet is monitored automatically. To create a flow log, specify the resource for which to create the flow log, type of traffic to capture (accepted traffic, rejected traffic, or all traffic), and destination to which you want to publish the flow log data. After you’ve created a flow log, it can take a few minutes to begin collecting and publishing data to the chosen destination.

1. At the top of the AWS Management Console, in the search bar, search for and choose

CloudWatch

.

1. Choose **Log groups**.
2. Choose **Create log group**.
3. For **Log group name**: Enter

VPCFlowLogGroup

1. Choose **Create**.
2. At the top of the AWS Management Console, in the search bar, search for and choose

VPC

.

1. In the left-hand navigation pane, choose **Subnets**.
2. Select all subnets that start with **Log Collection**.
3. For **Actions**: Select **Create flow log**.
4. For **Filter**: Select **All**.
5. For **Destination**: Select **Send to CloudWatch Logs**.
6. For **Destination log group**: Enter

VPCFlowLogGroup

1. For **IAM role**: Select **VPCFlowLogsRole**.

**Note:** The IAM role that is associated with the flow log must have sufficient permissions to publish flow logs to the specified log group in CloudWatch Logs. In addition, the role must have the trust relationship that allows the flow logs service to assume the role.

1. Choose **Create flow log**.

**Task 3: Enable database logging**

Publishing database logs to CloudWatch Logs lets you perform real-time analysis of the log data, store the data in highly durable storage, and manage the data with the CloudWatch Logs Agent. Monitoring is an important part of maintaining the reliability, availability, and performance of Amazon RDS and your AWS solutions. You should collect monitoring data from all parts of your AWS solution so that you can more easily debug a multi-point failure if one occurs. For example, with Amazon RDS, you can monitor network throughput; I/O for read, write, and/or metadata operations; client connections; and burst credit balances for your database instances.

1. At the top of the AWS Management Console, in the search bar, search for and choose

RDS

.

1. In the left-hand navigation pane, choose **Databases**.
2. Select the **DB identifier**.
3. Choose **Modify**.
4. Scroll down to **Log exports**, and select Error log.
5. Choose **Continue**.
6. From the **Scheduling of modifications** section, choose  **Apply immediately**.
7. Choose **Modify DB instance**.
8. Select the **DB identifier**.
9. Choose **Actions** menu and select **Reboot**.
10. Choose **Confirm**.

**Note:** Wait for the database to be available.

**Task 4: Validating and exploring log streams**

A log stream is a sequence of log events that share the same source. More specifically, a log stream is generally intended to represent the sequence of events coming from the application instance or resource being monitored. For example, a log stream may be associated with an Apache access log on a specific host.

1. At the top of the AWS Management Console, in the search bar, search for and choose

CloudWatch

.

1. In the left-hand navigation pane, choose **Log groups**.
2. You should see the following log groups: AccessLogGroup, IISLogGroup, VPCFlowLogGroup, CloudTrailLogGroup, and /aws/rds/instance/… (Also, you may notice two more log groups and SSHLogGroup. These appear if anyone tries to reach the EC2 servers through SSH or RDP).

 Choosing the name of each log group displays the log stream files for that group. Choose a log stream filename to see the logging events. If you choose the name of a log group and a log stream file is not displayed, go back through the previous tasks and ensure you completed all of the steps.

EXPLORE THE LOGS USING CLOUDWATCH LOGS INSIGHTS

CloudWatch Logs Insights enables you to interactively search and analyze your log data in Amazon CloudWatch Logs. You can perform queries to help you quickly and effectively respond to operational issues. If an issue occurs, you can use CloudWatch Logs Insights to identify potential causes and validate deployed fixes. The service includes a purpose-built query language with a few simple but powerful commands. CloudWatch Logs Insights provides sample queries, command descriptions, query autocompletion, and log field discovery to help you get started quickly.

Sample queries are included for several types of AWS service logs. For example query information, see [*https://docs.aws.amazon.com/AmazonCloudWatch/latest/logs/CWL\_QuerySyntax-examples.html*](https://docs.aws.amazon.com/AmazonCloudWatch/latest/logs/CWL_QuerySyntax-examples.html).

EXAMPLE 1: LOOK INTO WEB SERVER TRAFFIC AND ACCESS PATTERNS

As part of lab provisioning, each EC2 instance is set up with a web server: an Apache web server on the Linux instance and an IIS web server on the Windows instance. Test them by opening the URLs listed in the left-hand navigation (copy and paste the URL in a new browser tab or window).

To generate more log data on each web server, one of the EC2 instances is constantly calling these servers. You can explore these logs with CloudWatch Insights.

1. At the top of the AWS Management Console, in the search bar, search for and choose

CloudWatch

.

1. In the left-hand navigation pane, under **Logs**, choose **Log Insights**.
2. Choose on the *Select log group(s)* , and choose **AccessLogGroup**.
3. Keep the default query, which will show the 20 most recent log entries from the Apache web server. Choose **Run query**.
4. Review the log entries, which include information such as where the calls are originating from and what the response codes are (200 or 404).
5. Now, find out how many hits the Apache web server received in a 5-minute interval. Copy and paste the text below into the query window:

fields @timestamp, @message

| sort @timestamp desc

| stats count() by bin(5m)

1. Choose **Run query**.

**Note:** You can run the same queries for **IISLogGroup** and observe how the load is balanced between these two web servers.

EXAMPLE 2: LOOK INTO VPC FLOW LOGS TO IDENTIFY DATABASE CALLS THAT ARE NOT LEGITIMATE

Amazon VPC Flow Logs capture information about the IP traffic going to and from network interfaces in a VPC. Use the flow logs to investigate network traffic patterns and identify threats and risks across your VPC network. In this example, you will look for all the database calls attempted on the wrong database port. This lab is provisioned with a MySQL Amazon RDS database instance. The default MySQL port number is 3306, and any calls made on 3306 are legitimate calls. Any other calls (such as those to port number 4306) are wrong.

1. At the top of the AWS Management Console, in the search bar, search for and choose

CloudWatch

.

1. In the left-hand navigation pane, under **Logs**, choose **Log Insights**.
2. Choose on the *Select log group(s)* , and choose **VPCFlowLogGroup**.
3. Copy and paste the text below into the query window:

fields @srcaddr, @dstport

|parse '\* \* \* \* \* \* \* \* \* \* \* \* \* \*' as version, account\_id, interface\_id, srcaddr, dstaddr, srcport, dstport, protocol, packets, bytes, start, end, action, log\_status

| stats count(\*) as callsNotOnPort3306 by srcaddr, dstport

|filter dstport != 3306 and dstport != 80

| sort by callsNotOnPort3306 desc

| limit 20

1. Choose **Run query**.
2. Observe the query results for the different sources that are trying to attempt calls to the database on port other than 3306. Do you recognize these IP addresses?

EXAMPLE 3: LOOK FOR DATABASE ERROR LOGS

In **Task 3** you enabled RDS database logs to publish over to CloudWatch Logs, let’s inspect those logs.

1. At the top of the AWS Management Console, in the search bar, search for and choose

CloudWatch

.

1. In the left-hand navigation pane, under **Logs**, choose **Log Insights**.
2. Choose on the *Select log group(s)* , and choose **/aws/rds/instance/xxxxxxx/error**.
3. Copy and paste the text below into the query window:

fields @timestamp, @message

| sort @timestamp desc

1. Choose **Run query**.
2. Observe the query results, look for the host IP, where those calls are coming and why those calls failing?

EXAMPLE 4: LOOK FOR REJECTED CALLS BY SOURCE IP

In this query, you will peek into all the flow logs and look for IP addresses that are trying to hack the network.

1. At the top of the AWS Management Console, in the search bar, search for and choose

CloudWatch

.

1. In the left-hand navigation pane, under **Logs**, choose **Log Insights**.
2. Choose on the *Select log group(s)* , and choose **VPCFlowLogGroup**.
3. In the right-hand navigation pane, choose **Queries**.
4. Under **Sample queries** expand **VPC Flow Logs** > **Top 20 source IP addresses with highest number of rejected requests**.
5. Choose **Apply**.
6. Choose **Run query**.
7. Observe the IP address with the most rejects. What other IP addresses are appearing, and why?

EXAMPLE 5: CREATE A CLOUDWATCH METRIC FILTER FOR ALL 404 ERROR RESPONSE CODES

1. At the top of the AWS Management Console, in the search bar, search for and choose

CloudWatch

.

1. In the left-hand navigation pane, choose **Log groups**.
2. Choose **AccessLogGroup**.
3. Choose **Actions** menu and select **Create metric filter**.
4. For **Filter pattern**: Enter

404

1. For **Select log data to test**: Select a file that ends with **http.log**.
2. Choose **Test pattern**.
3. Choose **Next**.
4. For **Filter name**: Enter

accesslogs-404-errors

1. For **Metric namespace**: Enter

WebsiteLogMetrics

1. For **Metric name**: Enter

accesslogs-404-errors

1. For **Metric value**: Enter

1

1. For **Default value**: Enter

0

1. Choose **Next**.
2. Choose **Create metric filter**.

Now, repeat these steps for the **IISLogGroup**:

1. In the left-hand navigation pane, choose **Log groups**.
2. Choose **IISLogGroup**.
3. Choose **Actions** menu and select **Create metric filter**.
4. For **Filter Pattern**: Enter

404

1. For **Select log data to test**: Select a file that ends with **http.log**.
2. Choose **Test Pattern**.
3. Choose **Next**.
4. For **Filter name**: Enter

iislogs-404-errors

1. For **Metric namespace**: Enter

WebsiteLogMetrics

1. For **Metric name**: Enter

iislogs-404-errors

1. For **Metric value**: Enter

1

1. For **Default value**: Enter

0

1. Choose **Next**.
2. Choose **Create metric filter**.
3. In the left-hand navigation pane, under **Metrics**, choose **All metrics**.
4. You should see **WebsiteLogMetrics**. Choose **WebsiteLogMetrics**, and then choose **Metrics with no dimensions**.

**Note:** If you do not see it, wait a few minutes and refresh the page.

1. Select **accesslogs-404-errors** and **iislogs-404-errors**.
2. Choose the **Graphed metrics** tab.
3. For both metrics, under **Statistic**, select **Sum**.
4. For both metrics, under **Period**, select **30 Seconds**.
5. Above the graph, choose **1h**.
6. Hover over points in the graph to inspect the number of 404 errors on the Linux and Windows servers.

**Task 5: Log discovery**

CloudWatch Logs enables you to centralize the logs from all of your systems, applications, and AWS services in a single, highly scalable service. You can then easily view the logs, search them for specific error codes or patterns, filter them based on specific fields, or archive them securely for future analysis.

 CloudWatch Logs enables you to see all of your logs, regardless of their source, as a single and consistent flow of events ordered by time. You can query them and sort them based on other dimensions, group them by specific fields, create custom computations with a powerful query language, and visualize log data in dashboards.

As shown in the previous examples, you can extract a lot of information out of your logs. Challenge yourself to answer the following questions using what you’ve learned from this lab. When you think you have the answers, scroll down to the **Solutions** section.

Q1. How many successful HTTP calls were made from each web server?

Q2. What are all the IP addresses that are making calls on port 3306 (call to database)? Are those servers making any other calls? Is this a cause for concern?

Q3. What are the top 20 IP addresses in the Access logs and IIS logs?

Q4. What is the IP address of the last IAM user logged in, and what action did the user take?

Q5. Is anyone attempting to log in to either of these web servers through ports 22? If so, what IP address?

SOLUTIONS

Q1. How many successful HTTP calls were made from each web server?

To answer this question, use CloudWatch Insights. Select **AccessLogGroup** or **IISLogGroup**, and use the following query text:

fields @message

| filter @message like /200/

| stats count() by bin(2h)

Q2. What are all the IP addresses that are making calls on port 3306? Are those servers making any other calls? Is this a cause for concern?

To answer this question, use CloudWatch Insights. Select **VPCFlowLogGroup**, and use the following query text:

fields @srcAddr

| stats count(\*) as callsOnPort3306 by srcAddr

|filter dstPort = 3306

| sort by callsOnPort3306 desc

| limit 20

For the second part of the question, look at Examples 2 and 3. If an intruder is trying to reach the database illegitimately, it is a concern, and you want to protect from those calls.

Q3. What are the top 20 IP addresses in the Access logs and IIS logs?

To answer this question, use CloudWatch Insights. Select **AccessLogGroup** or **IISLogGroup**, and use the following query text:

*Query for Access logs*

fields @srcIp

|parse '\* - \* [\*] "\* \* \*" \* \*' as srcIp, identity, dateTimeString, httpVerb, url, protocol, statusCode, bytes

| stats count(\*) as numHitsBySrcIp by srcIp

| sort by numHitsBySrcIp desc

| limit 20

*Query for IIS logs*

fields @srcIp

|parse '\* \* \* \* \* \* \* \* \* \* \* \* \*' as date, time, destIp, httpMethod, url, filler1, port, filler2, srcIp, msg, filler3, httpCode

| stats count(\*) as numHitsBySrcIp by srcIp

| sort by numHitsBySrcIp desc

| limit 20

Q4. What is the IP address of the last IAM user logged in, and what action did the user take?

To answer this question, from CloudWatch **Insights**, select **CloudTrailLogGroup** copy this query, trying to find out the ssh calls on Linux server.

fields @timestamp, sourceIPAddress, userIdentity.userName, eventType, eventName, eventSource

|filter userIdentity.type = "IAMUser"

|sort @timestamp desc

|limit 20

**WARNING:** Incase if you receive blank results, replace the **IAMUser** with **AssumedRole** in above query and run it again.

Q5. Is anyone attempting to log in to either of these web servers through ports 22? If so, what IP address?

*Query for port 22 (SSH) calls*

To find the SSH calls on the Linux server, use CloudWatch Insights. Select **SSHLogGroup**, and use the following query text:

a. calls using wrong access key

parse '\* \* \* \* \* \* \* \* \* \* \* \*' as Mon, day, timestamp, destip, id, msg1, msg2, msg3, srcIp, msg5, destPort, msg7

| filter msg1 = 'Connection'

| sort by @timestamp desc

b. Calls using wrong user credentials

parse '\* \* \* \* \* \* \* \* \* \* \* \*' as Mon, day, timestamp, destip, id, msg1, msg2, msg3, msg4, srcIp, msg6, destPort

| filter msg2 = 'user'

| sort by @timestamp desc

c. General query

fields @timestamp, @message

| sort @timestamp desc

| limit 25

**Conclusion**

 Congratulations! You now have successfully:

* Installed and configured CloudWatch Agents on the EC2 instances using Systems Manager
* Collected web server and system logs from an EC2 instance and publish them over to CloudWatch Logs
* Enabled Amazon VPC Flow Logs across VPCs, subnets, and security groups
* Enabled CloudWatch Logs for an Amazon RDS database instance
* Explored logs using CloudWatch Logs Insights
* Inspected logs using CloudWatch Logs Insights and CloudWatch Metrics

**End lab**

Follow these steps to close the console and end your lab.

1. Return to the **AWS Management Console**.
2. At the upper-right corner of the page, choose **AWSLabsUser**, and then choose **Sign out**.
3. Choose **End lab** and then confirm that you want to end your lab.

**Additional resources**

* For more information about AWS Systems Manager, see [*https://docs.aws.amazon.com/systems-manager/latest/userguide/what-is-systems-manager.html*](https://docs.aws.amazon.com/systems-manager/latest/userguide/what-is-systems-manager.html)
* For more information about Amazon CloudWatch, see [*https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/WhatIsCloudWatch.html*](https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/WhatIsCloudWatch.html)
* For more information about VPC Flow Logs, see [*https://docs.aws.amazon.com/vpc/latest/userguide/flow-logs.html/*](https://docs.aws.amazon.com/vpc/latest/userguide/flow-logs.html)
* For more information about AWS IAM, see [*https://docs.aws.amazon.com/IAM/latest/UserGuide/introduction.html*](https://docs.aws.amazon.com/IAM/latest/UserGuide/introduction.html)

For more information about AWS Training and Certification, see [*https://aws.amazon.com/training/*](https://aws.amazon.com/training/).

*Your feedback is welcome and appreciated.*  
If you would like to share any feedback, suggestions, or corrections, please provide the details in our [*AWS Training and Certification Contact Form*](https://support.aws.amazon.com/#/contacts/aws-training).