Monitoring AWS for suspicious traffic

Applicability

- Product: Splunk Cloud Platform or Splunk Enterprise
- Feature: Splunk App for AWS and Splunk Add-on for AWS
- Data: Amazon Web Services: CloudTrail and CloudWatch
- Data: Amazon Web Services: VPC Flow Logs
- Function: Monitoring

Scenario

You are an Amazon Web Services (AWS) admin who manages access to AWS resources and services across your organization. You use Amazon's virtual private cloud (VPC) - an on-demand managed cloud-computing service that isolates computing resources for each client.

Amazon's VPC service enables you to launch Elastic Compute Cloud (EC2) instances and leverage other Amazon resources. The traffic that flows in and out of this VPC can be controlled via network access-control rules and security groups. Amazon also has a feature called VPC Flow Logs that enables you to log IP traffic going to and from the network interfaces in your VPC. This data is stored using Amazon CloudWatch Logs. Inside the VPC container, the environment resembles a physical network.

Attackers may abuse the AWS infrastructure with insecure VPCs so they can co-opt AWS resources for command-and-control nodes, data exfiltration, and more. After an EC2 instance is compromised, an attacker may initiate outbound network connections for malicious reasons. Monitoring these network traffic behaviors is important for understanding the type of traffic flowing in and out of your network and to alert you to suspicious activities.

You can use these searches to monitor your AWS network traffic for evidence of anomalous activity and suspicious behaviors, such as a spike in blocked outbound traffic in your VPC.

To run these searches, install the AWS App for Splunk (version 5.1.0 or later) and Splunk Add-on for AWS (version 4.4.0 or later), and configure your VPC flow logs. Some searches also require configuration of AWS configuration inputs and CloudTrail inputs.
Support searches

► Baseline of blocked outbound traffic from AWS

This search establishes, on a per-hour basis, the average and the standard deviation of the number of outbound connections blocked in your VPC flow logs by each source IP address (IP address of your EC2 instances). Also recorded is the number of data points for each source IP. This table outputs to a lookup file to allow the detection search to operate quickly.

```
sourcetype=aws:cloudwatchlogs:vpcflow action=blocked (src_ip=10.0.0.0/8 OR src_ip=172.16.0.0/12 OR src_ip=192.168.0.0/16) ( dest_ip!=10.0.0.0/8 AND dest_ip!=172.16.0.0/12 AND dest_ip!=192.168.0.0/16) | bucket _time span=1h | stats count AS numberOfBlockedConnections BY _time, src_ip | stats count(numberOfBlockedConnections) AS numDataPoints, latest(numberOfBlockedConnections) AS latestCount, avg(numberOfBlockedConnections) AS avgBlockedConnections, stdev(numberOfBlockedConnections) AS stdevBlockedConnections BY src_ip | table src_ip, latestCount, numDataPoints, avgBlockedConnections, stdevBlockedConnections | outputlookup baseline_blocked_outbound_connections | stats count
```

Detection searches

► Detect spikes in blocked outbound traffic from AWS

This search works best when you run the Baseline of blocked outbound traffic from AWS support search once to create a history of previously seen blocked outbound connections.

This search detects spikes in blocked outbound network connections originating from within your AWS environment. It also updates the cache file that factors in the latest data.

You can modify dataPointThreshold and deviationThreshold to better fit your environment. The dataPointThreshold variable is the number of data points required to meet the definition of "spike." The deviationThreshold variable is the number of standard deviations away from the mean that the value must be to be considered a spike.

The false-positive rate may vary based on the values of dataPointThreshold and deviationThreshold. False positives may occur when AWS administrators roll out policies enforcing network blocks, causing sudden increases in the number of blocked outbound connections.

Click here to read a detailed breakdown of how this search works.

```
sourcetype=aws:cloudwatchlogs:vpcflow action=blocked (src_ip=10.0.0.0/8 OR src_ip=172.16.0.0/12 OR src_ip=192.168.0.0/16) ( dest_ip!=10.0.0.0/8 AND dest_ip!=172.16.0.0/12 AND dest_ip!=192.168.0.0/16) [search sourcetype=aws:cloudwatchlogs:vpcflow action=blocked (src_ip=10.0.0.0/8 OR src_ip=172.16.0.0/12 OR src_ip=192.168.0.0/16) ( dest_ip!=10.0.0.0/8 AND dest_ip!=172.16.0.0/12 AND dest_ip!=192.168.0.0/16)]
```
| stats count AS numberOfBlockedConnections BY src_ip |
| inputlookup baseline_blocked_outbound_connections append=t |
| fields - latestCount |
| stats values(*) AS * BY src_ip |
| rename numberOfBlockedConnections AS latestCount |
| eval newAvgBlockedConnections=avgBlockedConnections + (latestCount-avgBlockedConnections)/720 |
| eval newStdevBlockedConnections=sqrt((pow(stdevBlockedConnections, 2)*719 + (latestCount-
| newAvgBlockedConnections)*(latestCount-avgBlockedConnections))/720)) |
| eval avgBlockedConnections=coalesce(newAvgBlockedConnections, avgBlockedConnections), stdevBlockedConnections=coalesce(newStdevBlockedConnections, stdevBlockedConnections), numDataPoints=if(isnull(latestCount), numDataPoints, numDataPoints+1) |
| table src_ip, latestCount, numDataPoints, avgBlockedConnections, stdevBlockedConnections |
| outputlookup baseline_blocked_outbound_connections |
| eval dataPointThreshold = 5, deviationThreshold = 3 |
| eval isSpike=if((latestCount > avgBlockedConnections+deviationThreshold*stdevBlockedConnections) AND numDataPoints > dataPointThreshold, 1, 0) |
| where isSpike=1 |
| table src_ip |
| stats values(dest_ip) AS "Blocked Destination IPs", values(interface_id) AS "resourceId" count AS numberOfBlockedConnections, dc(dest_ip) AS uniqueDestConnections BY src_ip |

---

**Contextual searches**

► **AWS network interface details via resourceId**

In order to run this search, you need to configure your [AWS configuration inputs](#).

This search retrieves all the activity from a specific IP address and creates a table containing the time, ARN, username, the type of user, the IP address, the AWS region the activity was in, the API called, and whether or not the API call was successful.

```splunk
| search sourcetype=aws:cloudtrail |
| iplocation sourcelIPAddress |
| search sourcelIPAddress={src_ip} |
| spath output=user path=userIdentity.arn |
| spath output=awsUserName path=userIdentity.userName |
| spath output=usertype path=userIdentity.type |
| rename sourcelIPAddress AS src_ip |
| table _time, user, userName, userType, src_ip, awsRegion, eventName, errorCode |
```

---

**Investigative searches**

► **Get all AWS activity from IP address**

In order to run this search, you need to configure your [CloudTrail inputs](#).

This search retrieves all the activity from a specific IP address and creates a table containing the time, ARN,
username, the type of user, the IP address, the AWS region the activity was in, the API called, and whether or not the API call was successful.

| search sourcetype=aws:cloudtrail
| iplocation sourceIPAddress
| search sourceIPAddress={src_ip}
| spath output=user path=userIdentity.arn
| spath output=awsUserName path=userIdentity.userName
| spath output=userType path=userIdentity.type
| rename sourceIPAddress AS src_ip
| table _time, user, userName, userType, src_ip, awsRegion, eventName, errorCode

---

**Additional resources**

This use case is included within [Splunk Enterprise Security](https://www.splunk.com), a Splunk app that provides prebuilt content and searches to help answer root-cause questions in real-time about malicious and anomalous events in your IT infrastructure. In addition, Splunk Enterprise Security provides a number of other searches to help reinforce your Cloud Security posture, including:

- Detecting AWS cross-account activity
- Detect suspicious new instances in your Elastic Compute Cloud (EC2) environment
- Detecting AWS network ACL activity
- Detecting AWS suspicious provisioning activities
- Monitoring users in AWS
- Monitoring AWS Elastic Compute Cloud (EC2) for suspicious login activities
- Monitoring AWS S3 for suspicious activities
- Monitoring AWS Elastic Compute Cloud (EC2) for unusual modifications

---

The information provided in Splunk Lantern is intended for informational and educational purposes only. All information is provided in good faith, however, Splunk disclaims any and all representations and warranties, express and implied, regarding the information provided, including without limitation any warranties and representations regarding the completeness, adequacy or accuracy of the information. You agree to take full responsibility for the results arising from the use of the information provided.