



# Closed Loop Hybrid Cloud Security & Compliance

TECHNICAL WHITEPAPER

MAY 2019

Bashyam Anant, Vice President Product Management  
Suresh Kasinathan, Cloud Security Architect  
Naveen Ramachandrappa, Senior Machine Learning Engineer

Cavirin Systems  
5201 Great America Parkway Suite 419  
Santa Clara, CA 95050

## Table of Contents

|                                                |          |
|------------------------------------------------|----------|
| <b>Executive Summary .....</b>                 | <b>2</b> |
| <b>An Ounce of Prevention .....</b>            | <b>2</b> |
| <b>Closed Loop Security .....</b>              | <b>2</b> |
| <b>Closed Loop Security Architecture .....</b> | <b>5</b> |
| Closed Loop Monitoring Framework .....         | 5        |
| Closed Loop Remediation Framework.....         | 5        |
| <b>Closed Loop Security in Action .....</b>    | <b>6</b> |
| Scenario 1: Security Group ingress rules ..... | 6        |
| Scenario 2: AWS GuardDuty findings.....        | 8        |
| <b>Closed Loop Compliance.....</b>             | <b>9</b> |
| <b>Next Steps.....</b>                         | <b>9</b> |



## Executive Summary

Cybersecurity skills shortage combined with the growing variety and complexity of public cloud services will require organizations to reinvent their approaches to security and compliance posture management. In this blog, we underscore the unification of proactive and reactive security as a key strategy for protecting and ensuring security and compliance of hybrid/multi-cloud infrastructure. We refer to this unification as “Closed Loop Security”.

Closed Loop Security begins with comprehensive visibility of organizational resources where ever they exist, assessing their security and compliance posture through assessment of technical controls and monitoring for configuration changes and threats via monitoring solutions like AWS CloudTrail/CloudWatch, GCP StackDriver, Azure Monitor and AWS GuardDuty to create prioritized, risk-driven change management plans that can be enacted through AWS Lambda, Azure/Google Functions and Ansible. We describe the risk scoring methodology that underpins Closed Loop Security and the machine learning approach embedded in it that ensures its efficacy.

## An Ounce of Prevention

In Cybersecurity Insider’s March 2019 survey (<https://www.cavirin.com/images/resources/pdf/Cloud-Security-Report-Cavirin-0519B.pdf>), 28% of organizations experienced a data or systems breach in their cloud infrastructure in the prior 12 months. Top cloud workloads, as a percent of surveyed organizations in the cloud, included: storage (43%), databases (38%), servers/containers (36%) and software-defined networking (29%). Unauthorized access through misuse of employee credentials and improper access controls (42%) takes the number one spot in this year’s survey as the single biggest perceived vulnerability in cloud security tied with insecure interfaces and APIs (42%). Rounding out the Top 3 are misconfigurations of the cloud platform (40%), such as misconfigured, globally-readable s3 buckets that make the news on a regular basis including the recent leak of 540 million Facebook consumer records by two application partners (<https://www.securityweek.com/aws-s3-buckets-exposed-millions-facebook-records>).

It is easy to point out that security lapses are preventable. What confounds SecOps users is the growing volume, variety and velocity of cloud services consumption. Even start ups like Cavirin with a relatively small cloud footprint have several thousand AWS, GCP and Azure resources including compute instances, block store volumes, object stores, VPCs, Subnets, Security Groups and so on and are bombarded with 1000’s of infrastructure alerts per hour. Continuous Integration and Deployment pipelines guarantees that Cavirin cloud infrastructure is changing every hour. That ounce of prevention is great to preach but much harder to practice.

## Closed Loop Security

Closed Loop Security is a refinement of NIST Cybersecurity Framework (NIST CSF - <https://www.nist.gov/cyberframework>). It unifies proactive assessment of configuration and vulnerability checks with reactive risk signals from monitoring systems like AWS Cloud Trail / Cloud Watch and threat detection systems like AWS GuardDuty. Put together, we believe Closed Loop Security helps organizations protect their AWS resources despite the volume, variety and velocity of AWS, GCP and Azure IaaS and PaaS services adoption.

Like NIST CSF, Closed Loop Security has five steps, visualized in Figure 1 and described in detail below:





Figure 1: Closed Loop Security

**STEP 1: IDENTIFY.** Discover and identify cloud resources where ever they exist. Cavin uses a combination of cloud APIs and remote sessions to build its inventory of 60+ AWS, GCP, Azure, container and OS resource types at the compute, storage, networking, containers and PaaS layers. Quite simply you cannot protect what you don't see. Each resource has a criticality (0.8 = LOW, 5 = HIGH) based on an assessment of confidential data, system integrity and availability requirements.

| RESOURCES                                                                                                                                                            |             |                |             |                    |                                |  |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------|----------------|-------------|--------------------|--------------------------------|--|
| <div>1 filters applied</div> <div>           Total Resources: 735   Non-Compute: 594   Compute Accessible: 11   Inaccessible: 130   Rows Fetched: 150         </div> |             |                |             |                    |                                |  |
| RESOURCE ID                                                                                                                                                          | ENVIRONMENT | SERVICE/OS     | CRITICALITY | CYBERPOSTURE SCORE | GROUP NAME                     |  |
| sg-0588dec42a3d2f0cf                                                                                                                                                 | AWS         | Security Group | 0.8         | 100                | AWS CavDemo AllRegions         |  |
| sg-3cfc1454                                                                                                                                                          | AWS         | Security Group | 0.8         | 75                 | AWS CavDemo AllRegions         |  |
| arn:aws:sns:us-east-1:206...                                                                                                                                         | AWS         | PubSub         | 0.8         | 0                  | AWS CavDemo AllRegions, 6 more |  |
| sg-081cdd76ab70a66ed                                                                                                                                                 | AWS         | Security Group | 0.8         | 100                | AWS CavDemo AllRegions, 6 more |  |
| sg-0bd130f62513c6b34                                                                                                                                                 | AWS         | Security Group | 0.8         | 100                | AWS CavDemo AllRegions, 6 more |  |
| vol-0936e472e7d429c45                                                                                                                                                | AWS         | Block Store    | 0.8         | 0                  | AWS CavDemo AllRegions, 6 more |  |
| vol-0691449b1d84e8d4b                                                                                                                                                | AWS         | Block Store    | 0.8         | 0                  | AWS CavDemo AllRegions, 6 more |  |

Figure 2: Single pane inventory of 30+ AWS resource types

**STEP 2: PROTECT.** Protect cloud and resources by evaluating 100,000+ configuration and vulnerability controls sourced from AWS best practices, Cavin thought leadership, vulnerability feeds, threat



detection feeds (such as AWS GuardDuty), Center for Internet Security (CIS), Defense Information Systems Agency (DISA) and more.

**STEP 3: PREDICT.** A proprietary scoring algorithm scores each resource (0 = HIGH RISK, 100 = LOW RISK) in a cloud account and creates aggregates at the organization, resource group, resource type and resource level. The scoring is based on the following tradeoffs:

- A weight (0-10) associated with the technical control assigned by a proprietary machine-learning classifier
- Resource criticality
- Number of resources failing a given control

In simple terms, high-weighted technical controls that are failing on lots of critical resources will depict a low score (implying high risk) in our scoring algorithm.

Scoring serves several purposes:

- A CISO can gauge their organization's security posture, assess trends and figure out where to direct response plans
- A SecOps user can slice and dice scores from the company level to a resource group to a resource type (e.g. S3 buckets) to an individual resource (an S3 bucket) and diagnose problems
- A DevOps users can create change management plans by prioritizing policies that offer the greatest security posture improvement

**STEP 4: RESPOND.** Remediate failing policies, prioritized using the scoring approach, using one of the following options:

- Manual change management using reports
- Publish security findings to ticketing systems (e.g. JIRA, ServiceNow), notification systems (e.g. Slack, PagerDuty) and SIEMs (e.g. Splunk, GCP Cloud Security Command Center)
- Serverless remediation of cloud resources using AWS Lambda, Azure or Google functions
- Ansible playbooks based remediation for Operating System resources

At the end of this step, an organization should be able to achieve a security baseline.

**STEP 5: MONITOR.** Achieving a security baseline is a great start. However, infrastructure changes can happen continuously. Incorporating signals from monitoring systems like AWS CloudTrail and CloudWatch (or Azure Monitor, GCP StackDriver) and threat detection platforms like AWS Guard Duty is an essential aspect of Closed Loop Security. Cavarin closes the loop by enacting the following (the flow is based on AWS but generalizes to Azure and GCP):

- As AWS resources log events to CloudTrail, Cavarin evaluates them against its technical controls. Cavarin also tracks new and deleted resources through CloudTrail.
- Cavarin accumulates a change log in the form of resources failing technical controls, new resources and deleted resources.
- If the extent of these changes exceeds a customer-configurable threshold, Cavarin reassesses the customer's AWS account using the PROTECT workflow of STEP 2 and the cycle continues through the PREDICT and RESPOND steps, as in Figure 1.



This approach of connecting reactive risk signals to proactive technical controls assessment is at heart of Closed Loop Security. It is a breakthrough because: (a) it reduces the time and cost to react to alerts by linking them to technical controls, risk scoring and remediation (b) minimizes operator alert fatigue by dispositioning alerts automatically (c) applies risk-based scoring to alerts

## Closed Loop Security Architecture

Cavirin's Closed Loop Security has two components that can be used together or separately:

- Monitoring framework
- Remediation framework

## Closed Loop Monitoring Framework

As Figure 3 demonstrates, Cavirin's monitoring framework automates the following steps, though these are largely hidden from the Cavirin user (the flow is based on AWS but generalizes to Azure and GCP):

1. An AWS service (e.g. Security Group) logs events (e.g. port opened on Security Group) via AWS CloudTrail
2. Cloudwatch alarms filter events based on what Cavirin can analyze.
3. Cloudwatch acts on alerts by triggering a Cavirin-authored AWS Lambda Function
4. The Cavirin Lambda Function processes the alerts and attempts to match them to policies available in Cavirin. Either way, it publishes alerts in a Cavirin format to SQS Queue
5. The Cavirin system periodically polls the SQS queue for new alerts
6. The SecOps users views alerts within the Cavirin dashboard

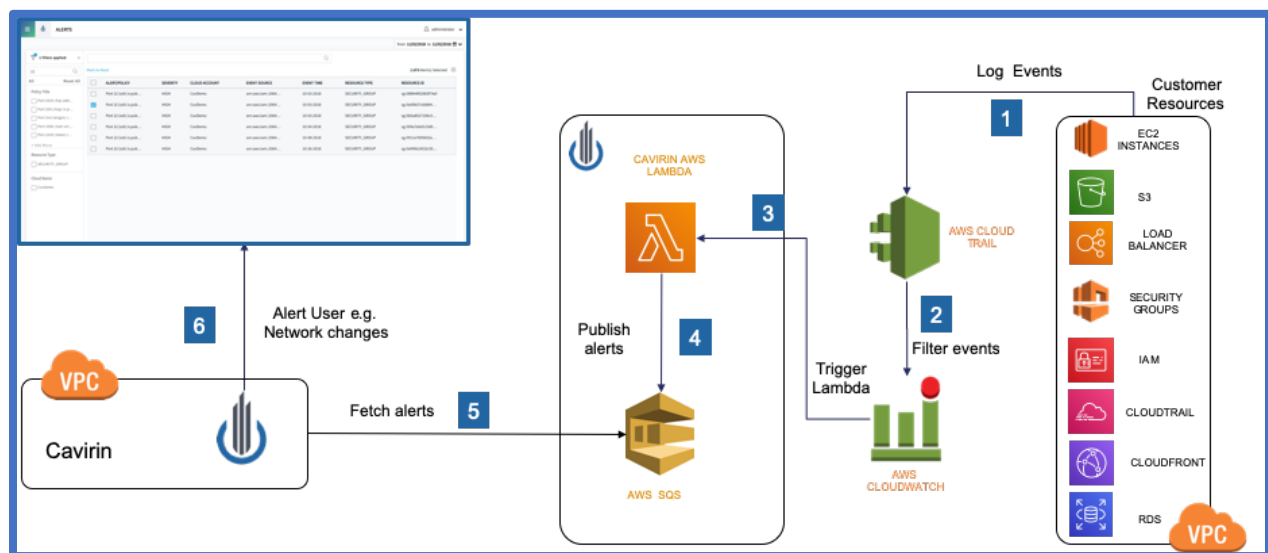


Figure 3: Closed Loop Monitoring Framework

## Closed Loop Remediation Framework

As Figure 4 demonstrates, Cavirin's remediation framework automates the following steps, though these are largely hidden from the Cavirin user (the flow is based on AWS but generalizes to Azure and GCP):



1. A SecOps picks one or more policies to remediate from the Prioritized Issues list within the Cavarin dashboard. The Prioritized Issues is a result of the PREDICT step of Closed Loop Security
2. Cavarin publishes a remediation request to an AWS SNS Topic
3. A Cavarin-authored AWS Lambda Function subscribes to remediation requests
4. AWS Lambda Function performs remediation
5. AWS Lambda posts remediation confirmations to an AWS SQS Queue
6. Cavarin system periodically polls the message queue for remediation confirmations
7. SecOps users view alerts related to completed remediations. Completed remediations contribute to Cavarin's change log for the customer's AWS account. If the extent of remediation confirmations, policy failures, deleted resources and new resources exceeds a customer-configurable threshold, Cavarin reassesses the customer's AWS account using the protect workflow of STEP 2 and the cycle continues through the PREDICT and RESPOND steps, as in Figure 1.

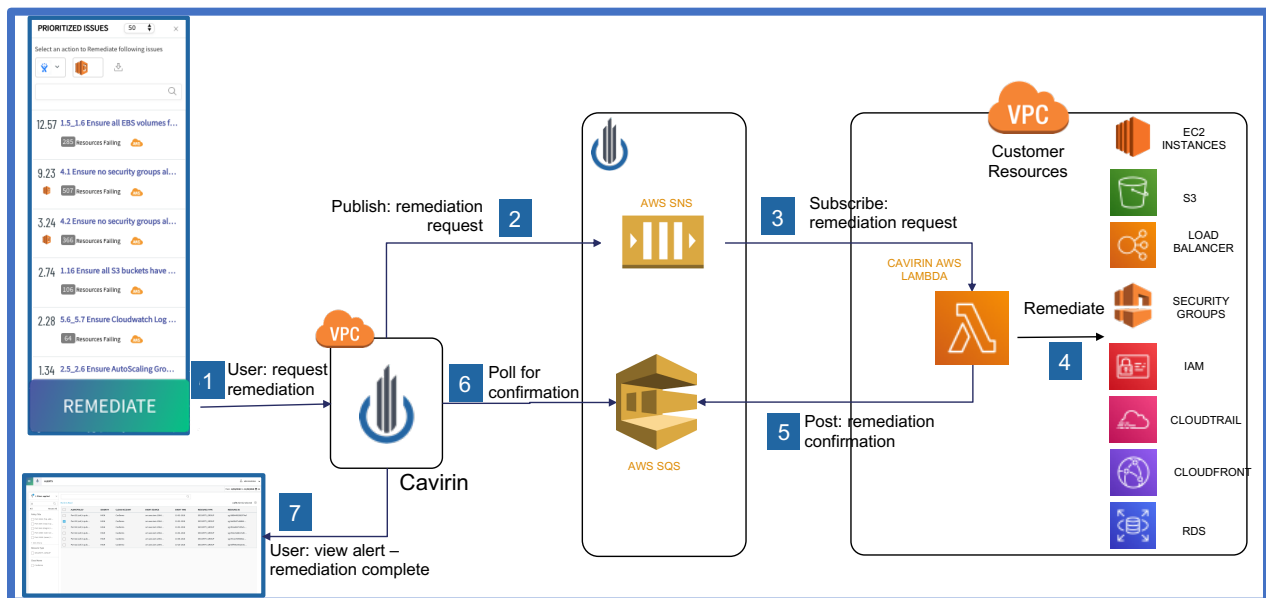


Figure 4: Closed Loop Remediation Framework

## Closed Loop Security in Action

We demonstrate Closed Loop Security through two scenarios. The examples are based on AWS but generalize to Azure and GCP resources.

### Scenario 1: Security Group ingress rules

The scenario starts with an AWS account that has a security group with several ports open for inbound traffic, which is a risky configuration. Figure 5 shows the current CyberPosture score for AWS security groups based on Steps 1 (IDENTIFY), 2 (PROTECT) and 3 (PREDICT). The prioritized issues shown on the right side of Figure 5 lists security group inbound ports with their improvement potential. Remediating the issue (closing an open port) is one-click away as shown in Figure 6. Post remediation, the



remediation is acknowledged via a CloudTrail alert shown to the user in Figure 7. The next assessment shows the improved score, as show in Figure 8.

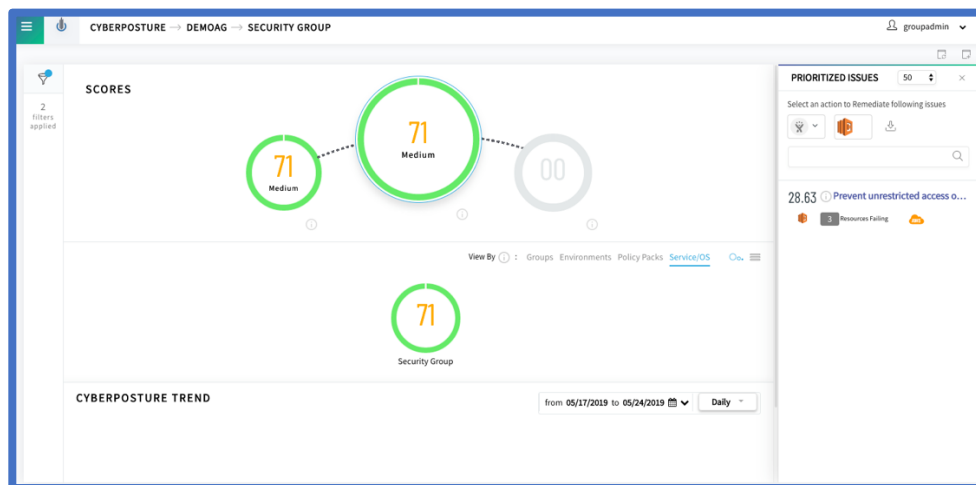


Figure 5: CyberPosture Score with Open Security Groups

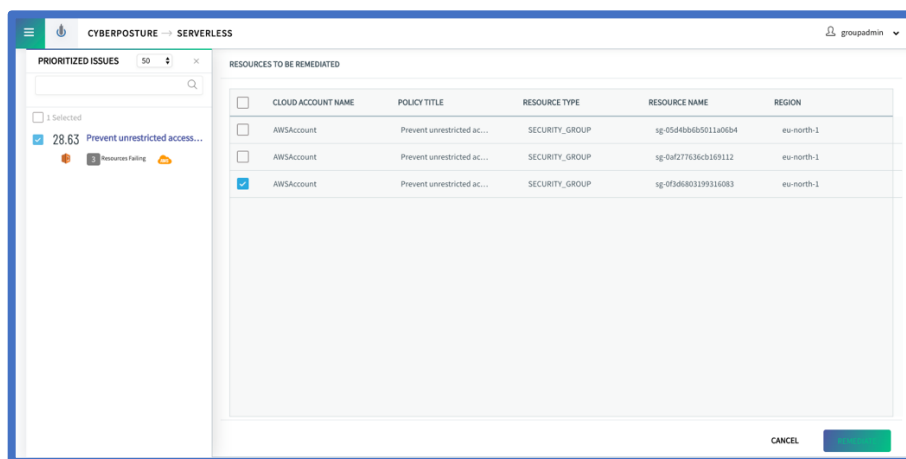


Figure 6: One-click remediation

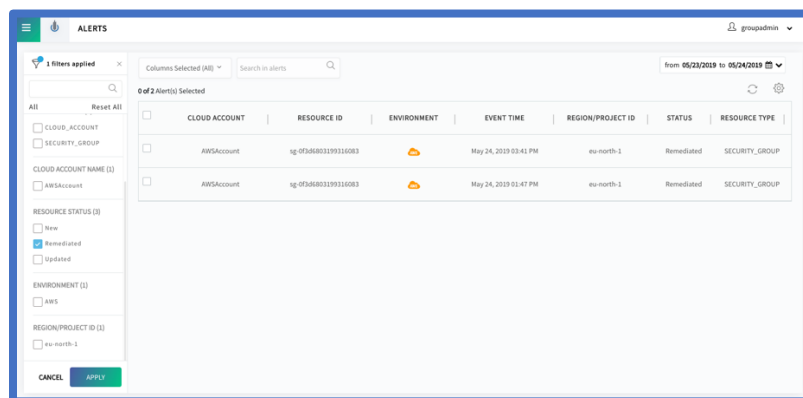


Figure 7: Remediation confirmation viewed through AWS Cloudtrail alerts



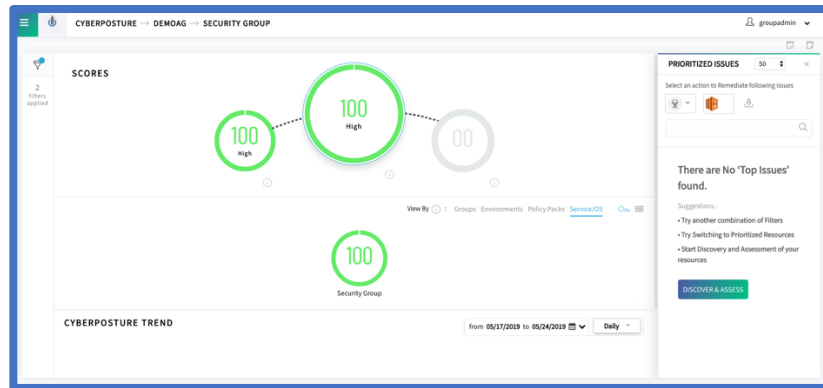


Figure 8: CyberPosture score after remediation

## Scenario 2: AWS GuardDuty findings

With the configuration of a permissioned AWS service account, Cavarin can ingest AWS GuardDuty findings that are accessible via one or more GuardDuty detectors. Cavarin ingests all documented AWS GuardDuty findings ([https://docs.aws.amazon.com/guardduty/latest/ug/guardduty\\_finding-types-active.html](https://docs.aws.amazon.com/guardduty/latest/ug/guardduty_finding-types-active.html)) including cryptominers and suspicious API, user and machine activity. These findings are mapped to EC2 instances or AWS accounts discovered by Cavarin and reduces their scores based on our algorithm. The scenario reflects the following steps:

- An AWS EC2 instance or an IAM entity (user, group, role) or an API exhibits suspicious activity – this is reported as finding in AWS GuardDuty and viewed as an alert in Cavarin as shown in Figure 9.
- Cavarin scores the affected EC2 instances or AWS accounts and reflects it on the dashboard and prioritized issues as shown in Figure 10
- The user can act on these alerts in AWS Console (e.g. by quarantining EC2 instances or disabling IAM users) and indicate “Mark as Read” against the findings within Cavarin– see Figure 9.
- In a subsequent assessments, the CyberPosture score increases as shown in Figure 10.

The screenshot shows the Alerts dashboard with 16 of 16 alerts selected. A 'Mark As Read' button is highlighted. The table below lists the details of the alerts.

|                                     | CLOUD ACCOUNT | RESOURCE ID  | ENVIRONMENT | EVENT TIME            | ALERT/POLICY                       | REGION/PROJECT ID | STATUS  | RESOURCE TYPE |
|-------------------------------------|---------------|--------------|-------------|-----------------------|------------------------------------|-------------------|---------|---------------|
| <input checked="" type="checkbox"/> | AWSAccount    | 416987053547 | Prod        | May 24, 2019 03:41 PM | Recon:IAMUser/MaliciousIPCaller... | eu-north-1        | Updated | CLOUD_ACCOUNT |
| <input checked="" type="checkbox"/> | AWSAccount    | 416987053547 | Prod        | May 24, 2019 02:12 PM | UnauthorizedAccess:IAMUser/Ma...   | eu-north-1        | New     | CLOUD_ACCOUNT |
| <input checked="" type="checkbox"/> | AWSAccount    | 416987053547 | Prod        | May 24, 2019 02:12 PM | Recon:IAMUser/MaliciousIPCaller... | eu-north-1        | New     | CLOUD_ACCOUNT |
| <input checked="" type="checkbox"/> | AWSAccount    | 416987053547 | Prod        | May 24, 2019 02:00 PM | UnauthorizedAccess:IAMUser/Ma...   | eu-north-1        | Updated | CLOUD_ACCOUNT |
| <input checked="" type="checkbox"/> | AWSAccount    | 416987053547 | Prod        | May 24, 2019 01:54 PM | UnauthorizedAccess:IAMUser/Ma...   | eu-north-1        | Updated | CLOUD_ACCOUNT |
| <input checked="" type="checkbox"/> | AWSAccount    | 416987053547 | Prod        | May 24, 2019 01:54 PM | Recon:IAMUser/MaliciousIPCaller... | eu-north-1        | Updated | CLOUD_ACCOUNT |

Figure 9: AWS GuardDuty findings list



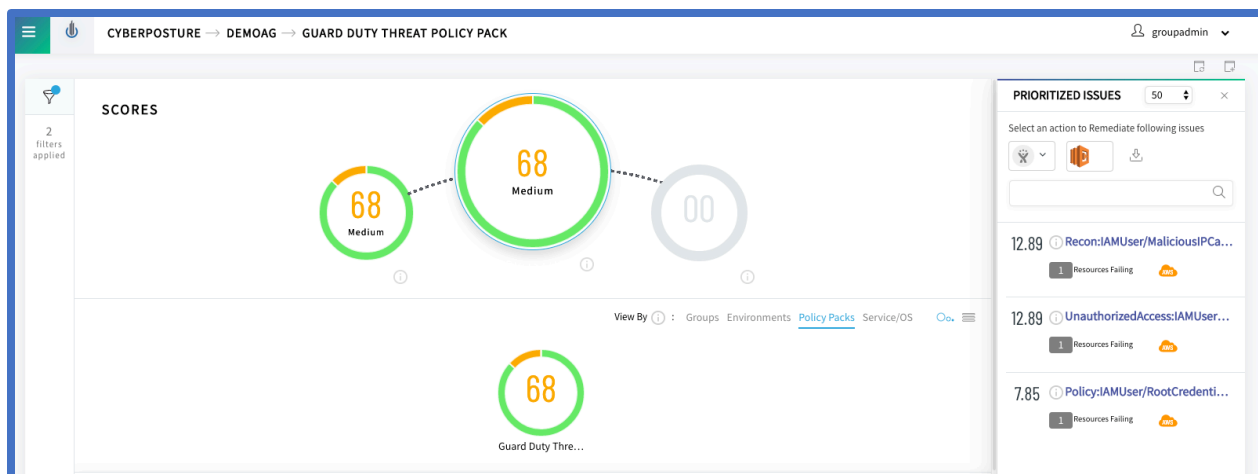


Figure 10: CyberPosture score due to AWS GuardDuty findings

## Closed Loop Compliance

Cavirin's approach to compliance mirrors Closed Loop Security with a few significant differences. For compliance frameworks like HIPAA, PCI, GDPR and AICPA SOC2, Cavirin has developed a machine learning Recommender System that accomplishes the following:

- For each compliance control, the Recommender System identifies the best-fit technical control from Cavirin's technical controls repository.
- For each compliance control, the Recommender System can assign a similarity score for each technical control that can be used to create differential weights for each technical control that serves to evaluate a compliance control

Compliance policy packs contribute to a compliance posture score which can be sliced and diced by environment (AWS, GCP, Azure, Docker, On-Premises), resource group, policy pack and resource type (e.g. object stores) to create prioritized action plans.

## Next Steps

Organizations report that compliance (34%) and lack of visibility into cloud security (33%) are their top cloud adoption challenges in the Cybersecurity Insiders March 2019 survey. We believe that Closed Loop Security and Compliance as described here can help organizations safely migrate to and expand their AWS usage. Try Cavirin <https://www.cavirin.com/why-cavirin/get-cyberposture-score>.

## About Cavirin

Cavirin is a privately-held, global provider of risk, cybersecurity, and compliance posture intelligence for the hybrid cloud. Cavirin's continuous protection, monitoring, and remediation, across hybrid infrastructure, automates compliance and secures organizations from possible cyber attacks. Powered by Cavirin's patent-pending machine learning technology, our CyberPosture Intelligence, provides users the most accurate credit-like scoring view of their security posture, so they can make informative decisions when protecting their organization. For more information, visit [www.cavirin.com](http://www.cavirin.com) or follow us at [www.twitter.com/cavirin](https://twitter.com/cavirin).

