AXONIUS

Cloud Asset Compliance for AWS

How to Ensure and Enforce Compliance with the CIS Amazon Web Services Foundations Benchmark 1.2

White Paper • February 2020

Table of Contents.

Introduction. 4

Overview 4
Defining Cloud Asset Management 4
Cloud Asset Compliance 4
What is CIS? 4

The CIS Amazon Web Services Foundations Benchmark 1.2. 5

1. Identity and Access Management. 6

1.1 Avoid the use of the "root" account 6
1.3 Ensure credentials unused for 90 days or greater are disabled 6
1.4 Ensure access keys are rotated every 90 days or less 7
1.5 Ensure IAM password policy requires at least one uppercase letter 7
1.6 Ensure IAM password policy require at least one lowercase letter 7
1.7 Ensure IAM password policy require at least one symbol 7
1.8 Ensure IAM password policy require at least one number 8
1.9 Ensure IAM password policy requires minimum length of 14 or greater 8
1.10 Ensure IAM password policy prevents password reuse 8
1.11 Ensure IAM password policy expires passwords within 90 days or less 8
1.12 Ensure no root account access key exists 9
1.13 Ensure MFA is enabled for the "root" account 9
1.14 Ensure hardware MFA is enabled for the "root" account 9
1.15 Ensure security questions are registered in the AWS account 10
1.16 Ensure IAM policies are attached only to groups or roles 10
1.17 Maintain current contact details 10
1.18 Ensure security contact information is registered 10
1.20 Ensure a support role has been created to manage incidents with AWS Support 11
1.21 Do not setup access keys during initial user setup for all IAM users that have a console password 12
1.22 Ensure IAM policies that allow full "*:*" administrative privileges are not created 12

2. Logging. 13

2.1 Ensure CloudTrail is enabled in all regions 13
2.2 Ensure CloudTrail log file validation is enabled 13
2.3 Ensure the S3 bucket used to store CloudTrail logs is not publicly accessible 13
2.4 Ensure CloudTrail trails are integrated with CloudWatch Logs 14
2.5 Ensure AWS Config is enabled in all regions 14
2.6 Ensure S3 bucket access logging is enabled on the CloudTrail S3 bucket 14
2.7 Ensure CloudTrail logs are encrypted at rest using KMS CMKs 15
2.8 Ensure rotation for customer created CMKs is enabled 15
2.9 Ensure VPC flow logging is enabled in all VPCs 15

3. Monitoring 16

3.1 Ensure a log metric filter and alarm exist for unauthorized API calls 16
3.2 Ensure a log metric filter and alarm exist for Management Console sign-in without MFA 16
3.3 Ensure a log metric filter and alarm exist for usage of "root" account 16
3.4 Ensure a log metric filter and alarm exist for IAM policy changes 16
3.5 Ensure a log metric filter and alarm exist for CloudTrail configuration changes 17
3.6 Ensure a log metric filter and alarm exist for AWS Management Console authentication failures 17
3.7 Ensure a log metric filter and alarm exist for disabling or scheduled deletion of customer created CMKs 17
3.9 Ensure a log metric filter and alarm exist for AWS Config configuration changes 18

3.11 Ensure a log metric filter and alarm exist for changes to Network Access Control Lists (NACL) 18
3.12 Ensure a log metric filter and alarm exist for changes to network gateways 18
3.13 Ensure a log metric filter and alarm exist for route table changes 19
3.14 Ensure a log metric filter and alarm exist for VPC changes 19

4. Networking. 20

4.1 Ensure no security groups allow ingress from 0.0.0.0/0 to port 22 20
4.2 Ensure no security groups allow ingress from 0.0.0.0/0 to port 3389 20
4.3 Ensure the default security group of every VPC restricts all traffic 20
4.4 Ensure routing tables for VPC peering are "least access" 21

CIS Amazon Web Services (AWS) Foundations Benchmark Checklist. 22

IAM 22
Logging 23
Monitoring 23
Networking 23

Cloud Asset Compliance for AWS 24

About Axonius. 24

Introduction.

Overview

Twenty years ago, the term "asset management" in IT had a very specific and defined meaning. The term ITAM referred to a simple process that included:

1. Inventory – Getting a detailed inventory of all hardware, software, and network assets
2. License Management – Making sure that all assets are running properly licensed software
3. Lifecycle Management – Deciding which assets should be decommissioned and managing the software licenses on these assets and updating the inventory

Using the traditional definition, IT Asset Management would fall squarely in the hands of the IT and Desktop Support teams. However, the process of gathering data about every asset and understanding what software is running is critical and foundational to cybersecurity.

Defining Cloud Asset Management

In 2020, the traditional definition isn't enough. Megatrends like cloud and IoT adoption have forced us to reexamine our most basic definitions of the word "asset." No longer a term just referring to a piece of hardware on a physical network, the push to move workloads to the cloud has strained some on fundamental assumptions around asset management:

- Cloud Asset Inventory – Who is responsible for maintaining an inventory of all cloud instances?
- Cloud Asset Security – Based on the inventory, who is responsible for ensuring that all cloud environments are covered by the organization's security solutions?
- Cloud Asset Compliance – How do all cloud environments adhere to or deviate from the overall security policy?

Cloud Asset Compliance

In this paper, we'll focus on the third piece of the puzzle: Cloud Asset Compliance, the process of understanding which security and management requirements must be met for an organization's cloud workloads. More specifically, we'll look at the CIS Controls® and CIS Benchmarks™ to understand widely accepted best practices for cloud compliance.

What is CIS?

The Center for Internet Security (CIS) is a nonprofit with a mission to safeguard public and private organizations against cyber threats with a charter to:

1. Identify, develop, validate, promote, and sustain best practice solutions for cyber defense
2. Build and lead communities to enable an environment of trust in cyberspace

The CIS Controls® and CIS Benchmarks™ are the global standard and recognized best practices for securing IT systems and data against the most pervasive attacks. The CIS Controls® are a prioritized set of 20 actions designed to protect an organization and data from known cyber attack vectors.

The CIS Amazon Web Services Foundations Benchmark 1.2.

CIS released the CIS Amazon Web Services (AWS) Foundations Benchmark, intended for system and application administrators, security specialists, auditors, help desk, platform deployment, and/or DevOps personnel who plan to develop, deploy, assess, or secure solutions in Amazon Web Services. This benchmark was created using a consensus review process comprised of subject matter experts. Consensus participants provide perspective from a diverse set of backgrounds including consulting, software development, audit and compliance, security research, operations, government, and legal.

The CIS AWS Foundations Benchmark consists of rules in 4 distinct categories:

1. Identity and Access Management
2. Logging
3. Monitoring
4. Networking

We'll now examine the individual rules in these categories, explain why they matter, and then show how Axonius customers are able to ensure they meet the benchmark set forth by CIS.

1. Identity and Access Management.

1.1 Avoid the use of the "root" account

What It Means: The "root" account has unrestricted access to all resources in the AWS account. It is highly recommended that the use of this account be avoided.

Why It Matters: The "root" account is the most privileged AWS account. Minimizing the use of this account and adopting the principle of least privilege for access management will reduce the risk of accidental changes and unintended disclosure of highly privileged credentials.

1.2 Ensure multi-factor authentication (MFA) is enabled for all IAM users that have a console password

What It Means: Multi-Factor Authentication (MFA) adds an extra layer of protection on top of a username and password. With MFA enabled, when a user signs into an AWS website, they will be prompted for their username and password as well as for an authentication code from their AWS MFA device. It is recommended that MFA be enabled for all accounts that have a console password.

Why It Matters: Enabling MFA provides increased security for console access as it requires the authenticating principal to possess a device that emits a time-sensitive key and have knowledge of a credential.

1.3 Ensure credentials unused for 90 days or greater are disabled

What It Means: AWS IAM users can access AWS resources using different types of credentials, such as passwords or access keys. It is recommended that all credentials that have been unused in 90 or greater days be removed or deactivated.

Why It Matters: Disabling or removing unnecessary credentials will reduce the window of opportunity for credentials associated with a compromised or abandoned account to be used.

1.4 Ensure access keys are rotated every 90 days or less

What It Means: Access keys consist of an access key ID and secret access key, which are used to sign programmatic requests that you make to AWS. AWS users need their own access keys to make programmatic calls to AWS from the AWS Command Line Interface (AWS CLI), Tools for Windows PowerShell, the AWS SDKs, or direct HTTP calls using the APIs for individual AWS services. It is recommended that all access keys be regularly rotated.

Why It Matters: Rotating access keys will reduce the window of opportunity for an access key that is associated with a compromised or terminated account to be used. Access keys should be rotated to ensure that data cannot be accessed with an old key which might have been lost, cracked, or stolen.

1.5 Ensure IAM password policy requires at least one uppercase letter

What It Means: Password policies are, in part, used to enforce password complexity requirements. IAM password policies can be used to ensure password are comprised of different character sets. It is recommended that the password policy require at least one uppercase letter.

Why It Matters: Setting a password complexity policy increases account resiliency against brute force login attempts

1.6 Ensure IAM password policy require at least one lowercase letter

What It Means: Password policies are, in part, used to enforce password complexity requirements. IAM password policies can be used to ensure password are comprised of different character sets. It is recommended that the password policy require at least one lowercase letter.

Why It Matters: Setting a password complexity policy increases account resiliency against brute force login attempts.

1.7 Ensure IAM password policy require at least one symbol

What It Means: Password policies are, in part, used to enforce password complexity requirements. IAM password policies can be used to ensure password are comprised of different character sets. It is recommended that the password policy require at least one symbol.

Why It Matters: Setting a password complexity policy increases account resiliency against brute force login attempts.

1.8 Ensure IAM password policy require at least one number

What It Means: Password policies are, in part, used to enforce password complexity requirements. IAM password policies can be used to ensure password are comprised of different character sets. It is recommended that the password policy require at least one number.

Why It Matters: Setting a password complexity policy increases account resiliency against brute force login attempts.

1.9 Ensure IAM password policy requires minimum length of 14 or greater

What It Means: Password policies are, in part, used to enforce password complexity requirements. IAM password policies can be used to ensure password are at least a given length. It is recommended that the password policy require a minimum password length 14.

Why It Matters: Setting a password complexity policy increases account resiliency against brute force login attempts.

1.10 Ensure IAM password policy prevents password reuse

What It Means: IAM password policies can prevent the reuse of a given password by the same user. It is recommended that the password policy prevent the reuse of passwords.

Why It Matters: Preventing password reuse increases account resiliency against brute force login attempts.

1.11 Ensure IAM password policy expires passwords within 90 days or less

What It Means: IAM password policies can require passwords to be rotated or expired after a given number of days. It is recommended that the password policy expire passwords after 90 days or less.

Why It Matters: Reducing the password lifetime increases account resiliency against brute force login attempts. Additionally, requiring regular password changes help in the following scenarios:

- Passwords can be stolen or compromised sometimes without your knowledge. This can happen via a system compromise, software vulnerability, or internal threat.
- Certain corporate and government web filters or proxy servers have the ability to intercept and record traffic even if it's encrypted.
- Many people use the same password for many systems such as work, email, and personal.
- Compromised end user workstations might have a keystroke logger.

1.12 Ensure no root account access key exists

What It Means: The root account is the most privileged user in an AWS account. AWS Access Keys provide programmatic access to a given AWS account. It is recommended that all access keys associated with the root account be removed.

Why It Matters: Removing access keys associated with the root account limits vectors by which the account can be compromised. Additionally, removing the root access keys encourages the creation and use of role-based accounts that are least privileged.

1.13 Ensure MFA is enabled for the "root" account

What It Means: The root account is the most privileged user in an AWS account. MFA adds an extra layer of protection on top of a username and password. With MFA enabled, when a user signs into an AWS website, they will be prompted for their username and password as well as for an authentication code from their AWS MFA device.

Note: When virtual MFA is used for root accounts, it is recommended that the device used is NOT a personal device, but rather a dedicated mobile device (tablet or phone) that is managed to be kept charged and secured independent of any individual personal devices. ("non-personal virtual MFA") This lessens the risks of losing access to the MFA due to device loss, device trade-in or if the individual owning the device is no longer employed at the company.

Why It Matters: Enabling MFA provides increased security for console access as it requires the authenticating principal to possess a device that emits a time-sensitive key and have knowledge of a credential.

1.14 Ensure hardware MFA is enabled for the "root" account

What It Means: The root account is the most privileged user in an AWS account. MFA adds an extra layer of protection on top of a username and password. With MFA enabled, when a user signs into an AWS website, they will be prompted for their username and password as well as for an authentication code from their AWS MFA device. For Level 2, it is recommended that the root account be protected with a hardware MFA.

Why It Matters: A hardware MFA has a smaller attack surface than a virtual MFA. For example, a hardware MFA does not suffer the attack surface introduced by the mobile smartphone on which a virtual MFA resides.

1.15 Ensure security questions are registered in the AWS account

What It Means: The AWS support portal allows account owners to establish security questions that can be used to authenticate individuals calling AWS customer service for support. It is recommended that security questions be established.

Why It Matters: When creating a new AWS account, a default super user is automatically created. This account is referred to as the "root" account. It is recommended that the use of this account be limited and highly controlled. During events in which the Root password is no longer accessible or the MFA token associated with root is lost/destroyed it is possible, through authentication using secret questions and associated answers, to recover root login access.

1.16 Ensure IAM policies are attached only to groups or roles

What It Means: By default, IAM users, groups, and roles have no access to AWS resources. IAM policies are the means by which privileges are granted to users, groups, or roles. It is recommended that IAM policies be applied directly to groups and roles but not users.

Why It Matters: Assigning privileges at the group or role level reduces the complexity of access management as the number of users grow. Reducing access management complexity may in-turn reduce opportunity for a principal to inadvertently receive or retain excessive privileges.

1.17 Maintain current contact details

What It Means: Ensure contact email and telephone details for AWS accounts are current and map to more than one individual in your organization. An AWS account supports a number of contact details, and AWS will use these to contact the account owner if activity judged to be in breach of Acceptable Use Policy or indicative of likely security compromise is observed by the AWS Abuse team. Contact details should not be for a single individual, as circumstances may arise where that individual is unavailable. Email contact details should point to a mail alias which forwards email to multiple individuals within the organization; where feasible, phone contact details should point to a PABX hunt group or other call-forwarding system.

Why It Matters: If an AWS account is observed to be behaving in a prohibited or suspicious manner, AWS will attempt to contact the account owner by email and phone using the contact details listed. If this is unsuccessful and the account behavior needs urgent mitigation, proactive measures may be taken, including throttling of traffic between the account exhibiting suspicious behavior and the AWS API endpoints and the Internet. This will result in impaired service to and from the account in question, so it is in both the customers' and AWS' best interests that prompt contact can be established. This is best achieved by setting AWS account contact details to point to resources which have multiple individuals as recipients, such as email aliases and PABX hunt groups.

1.18 Ensure security contact information is registered

What It Means: AWS provides customers with the option of specifying the contact information for account's security team. It is recommended that this information be provided.

Why It Matters: Specifying security-specific contact information will help ensure that security advisories sent by AWS reach the team in your organization that is best equipped to respond to them.

1.19 Ensure IAM instance roles are used for AWS resource access from instances

What It Means: AWS access from within AWS instances can be done by either encoding AWS keys into AWS API calls or by assigning the instance to a role which has an appropriate permissions policy for the required access. "AWS Access" means accessing the APIs of AWS in order to access AWS resources or manage AWS account resources.

Why It Matters: AWS IAM roles reduce the risks associated with sharing and rotating credentials that can be used outside of AWS itself. If credentials are compromised, they can be used from outside of the AWS account they give access to. In contrast, in order to leverage role permissions an attacker would need to gain and maintain access to a specific instance to use the privileges associated with it.

Additionally, if credentials are encoded into compiled applications or other hard to change mechanisms, then they are even more unlikely to be properly rotated due to service disruption risks. As time goes on, credentials that cannot be rotated are more likely to be known by an increasing number of individuals who no longer work for the organization owning the credentials.

1.20 Ensure a support role has been created to manage incidents with AWS Support

What It Means: AWS provides a support center that can be used for incident notification and response, as well as technical support and customer services. Create an IAM Role to allow authorized users to manage incidents with AWS Support.

Why It Matters: By implementing least privilege for access control, an IAM Role will require an appropriate IAM Policy to allow Support Center Access in order to manage Incidents with AWS Support.

1.21 Do not setup access keys during initial user setup for all IAM users that have a console password

What It Means: AWS console defaults the checkbox for creating access keys to enabled. This results in many access keys being generated unnecessarily. In addition to unnecessary credentials, it also generates unnecessary management work in auditing and rotating these keys.

Why It Matters: Requiring that additional steps be taken by the user after their profile has been created will give a stronger indication of intent that access keys are [a] necessary for their work and [b] once the access key is established on an account that the keys may be in use somewhere in the organization.

Note: Even if it is known the user will need access keys, require them to create the keys themselves or put in a support ticket to have the created as a separate step from user creation.

1.22 Ensure IAM policies that allow full "*:*" administrative privileges are not created

What It Means: IAM policies are the means by which privileges are granted to users, groups, or roles. It is recommended and considered a standard security advice to grant least privilege—that is, granting only the permissions required to perform a task. Determine what users need to do and then craft policies for them that let the users perform only those tasks, instead of allowing full administrative privileges.

Why It Matters: It's more secure to start with a minimum set of permissions and grant additional permissions as necessary, rather than starting with permissions that are too lenient and then trying to tighten them later.

Providing full administrative privileges instead of restricting to the minimum set of permissions that the user is required to do exposes the resources to potentially unwanted actions. IAM policies that have a statement with "Effect": "Allow" with "Action": "*" over "Resource": "*" should be removed.

2. Logging.

2.1 Ensure CloudTrail is enabled in all regions

What It Means: AWS CloudTrail is a web service that records AWS API calls for your account and delivers log files to you. The recorded information includes the identity of the API caller, the time of the API call, the source IP address of the API caller, the request parameters, and the response elements returned by the AWS service. CloudTrail provides a history of AWS API calls for an account, including API calls made via the Management Console, SDKs, command line tools, and higher-level AWS services (such as CloudFormation).

Why It Matters: The AWS API call history produced by CloudTrail enables security analysis, resource change tracking, and compliance auditing. Additionally,

- Ensuring that a multi-regions trail exists will ensure that unexpected activity occurring in otherwise unused regions is detected
- Ensuring that a multi-regions trail exists will ensure that Global Service Logging is enabled for a trail by default to capture recording of events generated on AWS global services
- For a multi-regions trail, ensuring that management events configured for all type of Read/Writes ensures recording of management operations that are performed on all resources in an AWS account

2.2 Ensure CloudTrail log file validation is enabled

What It Means: CloudTrail log file validation creates a digitally signed digest file containing a hash of each log that CloudTrail writes to S3. These digest files can be used to determine whether a log file was changed, deleted, or unchanged after CloudTrail delivered the log. It is recommended that file validation be enabled on all CloudTrails.

Why It Matters: Enabling log file validation will provide additional integrity checking of CloudTrail logs.

2.3 Ensure the S3 bucket used to store CloudTrail logs is not publicly accessible

What It Means: CloudTrail logs a record of every API call made in your AWS account. These log files are stored in an S3 bucket. It is recommended that the bucket policy, or access control list (ACL), applied to the S3 bucket that CloudTrail logs to prevents public access to the CloudTrail logs.

Why It Matters: Allowing public access to CloudTrail log content may aid an adversary in identifying weaknesses in the affected account's use or configuration.

2.4 Ensure CloudTrail trails are integrated with CloudWatch Logs

What It Means: AWS CloudTrail is a web service that records AWS API calls made in a given AWS account. The recorded information includes the identity of the API caller, the time of the API call, the source IP address of the API caller, the request parameters, and the response elements returned by the AWS service. CloudTrail uses Amazon S3 for log file storage and delivery, so log files are stored durably. In addition to capturing CloudTrail logs within a specified S3 bucket for long term analysis, real-time analysis can be performed by configuring CloudTrail to send logs to CloudWatch Logs. For a trail that is enabled in all regions in an account, CloudTrail sends log files from all those regions to a CloudWatch Logs log group. It is recommended that CloudTrail logs be sent to CloudWatch Logs.

Note: The intent of this recommendation is to ensure AWS account activity is being captured, monitored, and appropriately alarmed on. CloudWatch Logs is a native way to accomplish this using AWS services but does not preclude the use of an alternate solution.

Why It Matters: Sending CloudTrail logs to CloudWatch Logs will facilitate real-time and historic activity logging based on user, API, resource, and IP address, and provides opportunity to establish alarms and notifications for anomalous or sensitivity account activity.

2.5 Ensure AWS Config is enabled in all regions

What It Means: AWS Config is a web service that performs configuration management of supported AWS resources within your account and delivers log files to you. The recorded information includes the configuration item (AWS resource), relationships between configuration items (AWS resources), any configuration changes between resources. It is recommended to enable AWS Config in all regions.

Why It Matters: The AWS configuration item history captured by AWS Config enables security analysis, resource change tracking, and compliance auditing.

2.6 Ensure S3 bucket access logging is enabled on the CloudTrail S3 bucket

What It Means: S3 Bucket Access Logging generates a log that contains access records for each request made to your S3 bucket. An access log record contains details about the request, such as the request type, the resources specified in the request worked, and the time and date the request was processed. It is recommended that bucket access logging be enabled on the CloudTrail S3 bucket.

Why It Matters: By enabling S3 bucket logging on target S3 buckets, it is possible to capture all events which may affect objects within target buckets. Configuring logs to be placed in a separate bucket allows access to log information which can be useful in security and incident response workflows.

2.7 Ensure CloudTrail logs are encrypted at rest using KMS CMKs

What It Means: AWS CloudTrail is a web service that records AWS API calls for an account and makes those logs available to users and resources in accordance with IAM policies. AWS Key Management Service (KMS) is a managed service that helps create and control the encryption keys used to encrypt account data and uses Hardware Security Modules (HSMs) to protect the security of encryption keys. CloudTrail logs can be configured to leverage server-side encryption (SSE) and KMS customer created master keys (CMK) to further protect CloudTrail logs. It is recommended that CloudTrail be configured to use SSE-KMS.

Why It Matters: Configuring CloudTrail to use SSE-KMS provides additional confidentiality controls on log data as a given user must have S3 read permission on the corresponding log bucket and must be granted decrypt permission by the CMK policy.

2.8 Ensure rotation for customer created CMKs is enabled

What It Means: AWS Key Management Service (KMS) allows customers to rotate the backing key which is key material stored within the KMS which is tied to the key ID of the Customer Created customer master key (CMK). It is the backing key that is used to perform cryptographic operations such as encryption and decryption. Automated key rotation currently retains all prior backing keys so that decryption of encrypted data can take place transparently. It is recommended that CMK key rotation be enabled.

Why It Matters: Rotating encryption keys helps reduce the potential impact of a compromised key as data encrypted with a new key cannot be accessed with a previous key that may have been exposed.

2.9 Ensure VPC flow logging is enabled in all VPCs

What It Means: VPC Flow Logs is a feature that enables you to capture information about the IP traffic going to and from network interfaces in your VPC. After you've created a flow log, you can view and retrieve its data in Amazon CloudWatch Logs. It is recommended that VPC Flow Logs be enabled for packet "Rejects" for VPCs.

Why It Matters: VPC Flow Logs provide visibility into network traffic that traverses the VPC and can be used to detect anomalous traffic or insight during security workflows.

3. Monitoring.

3.1 Ensure a log metric filter and alarm exist for unauthorized API calls

What It Means: Real-time monitoring of API calls can be achieved by directing CloudTrail Logs to CloudWatch Logs and establishing corresponding metric filters and alarms. It is recommended that a metric filter and alarm be established for unauthorized API calls.

Why It Matters: Monitoring unauthorized API calls will help reveal application errors and may reduce time to detect malicious activity.

3.2 Ensure a log metric filter and alarm exist for Management Console sign-in without MFA

What It Means: Real-time monitoring of API calls can be achieved by directing CloudTrail Logs to CloudWatch Logs and establishing corresponding metric filters and alarms. It is recommended that a metric filter and alarm be established for console logins that are not protected by multi-factor authentication (MFA).

Why It Matters: Monitoring for single-factor console logins will increase visibility into accounts that are not protected by MFA.

3.3 Ensure a log metric filter and alarm exist for usage of "root" account

What It Means: Real-time monitoring of API calls can be achieved by directing CloudTrail Logs to CloudWatch Logs and establishing corresponding metric filters and alarms. It is recommended that a metric filter and alarm be established for root login attempts.

Why It Matters: Monitoring for root account logins will provide visibility into the use of a fully privileged account and an opportunity to reduce the use of it.

3.4 Ensure a log metric filter and alarm exist for IAM policy changes

What It Means: Real-time monitoring of API calls can be achieved by directing CloudTrail Logs to CloudWatch Logs and establishing corresponding metric filters and alarms. It is recommended that a metric filter and alarm be established changes made to Identity and Access Management (IAM) policies.

Why It Matters: Monitoring changes to IAM policies will help ensure authentication and authorization controls remain intact.

3.5 Ensure a log metric filter and alarm exist for CloudTrail configuration changes

What It Means: Real-time monitoring of API calls can be achieved by directing CloudTrail Logs to CloudWatch Logs and establishing corresponding metric filters and alarms. It is recommended that a metric filter and alarm be established for detecting changes to CloudTrail's configurations.

Why It Matters: Monitoring changes to CloudTrail's configuration will help ensure sustained visibility to activities performed in the AWS account.

3.6 Ensure a log metric filter and alarm exist for AWS Management Console authentication failures

What It Means: Real-time monitoring of API calls can be achieved by directing CloudTrail Logs to CloudWatch Logs and establishing corresponding metric filters and alarms. It is recommended that a metric filter and alarm be established for failed console authentication attempts.

Why It Matters: Monitoring failed console logins may decrease lead time to detect an attempt to brute force a credential, which may provide an indicator, such as source IP, that can be used in other event correlations.

3.7 Ensure a log metric filter and alarm exist for disabling or scheduled deletion of customer created CMKs

What It Means: Real-time monitoring of API calls can be achieved by directing CloudTrail Logs to CloudWatch Logs and establishing corresponding metric filters and alarms. It is recommended that a metric filter and alarm be established for customer created CMKs which have changed state to disabled or scheduled deletion.

Why It Matters: Data encrypted with disabled or deleted keys will no longer be accessible.

3.8 Ensure a log metric filter and alarm exist for S3 bucket policy changes

What It Means: Real-time monitoring of API calls can be achieved by directing CloudTrail Logs to CloudWatch Logs and establishing corresponding metric filters and alarms. It is recommended that a metric filter and alarm be established for changes to S3 bucket policies.

Why It Matters: Monitoring changes to S3 bucket policies may reduce time to detect and correct permissive policies on sensitive S3 buckets.

3.9 Ensure a log metric filter and alarm exist for AWS Config configuration changes

What It Means: Real-time monitoring of API calls can be achieved by directing CloudTrail Logs to CloudWatch Logs and establishing corresponding metric filters and alarms. It is recommended that a metric filter and alarm be established for detecting changes to CloudTrail's configurations.

Why It Matters: Monitoring changes to AWS Config configuration will help ensure sustained visibility of configuration items within the AWS account.

3.10 Ensure a log metric filter and alarm exist for security group changes

What It Means: Real-time monitoring of API calls can be achieved by directing CloudTrail Logs to CloudWatch Logs and establishing corresponding metric filters and alarms. Security Groups are a stateful packet filter that controls ingress and egress traffic within a VPC. It is recommended that a metric filter and alarm be established changes to Security Groups.

Why It Matters: Monitoring changes to security group will help ensure that resources and services are not unintentionally exposed.

3.11 Ensure a log metric filter and alarm exist for changes to Network Access Control Lists (NACL)

What It Means: Real-time monitoring of API calls can be achieved by directing CloudTrail Logs to CloudWatch Logs and establishing corresponding metric filters and alarms. NACLs are used as a stateless packet filter to control ingress and egress traffic for subnets within a VPC. It is recommended that a metric filter and alarm be established for changes made to NACLs.

Why It Matters: Monitoring changes to NACLs will help ensure that AWS resources and services are not unintentionally exposed.

3.12 Ensure a log metric filter and alarm exist for changes to network gateways

What It Means: Real-time monitoring of API calls can be achieved by directing CloudTrail Logs to CloudWatch Logs and establishing corresponding metric filters and alarms. Network gateways are required to send/receive traffic to a destination outside of a VPC. It is recommended that a metric filter and alarm be established for changes to network gateways.

Why It Matters: Monitoring changes to network gateways will help ensure that all ingress/egress traffic traverses the VPC border via a controlled path.

3.13 Ensure a log metric filter and alarm exist for route table changes

What It Means: Real-time monitoring of API calls can be achieved by directing CloudTrail Logs to CloudWatch Logs and establishing corresponding metric filters and alarms. Routing tables are used to route network traffic between subnets and to network gateways. It is recommended that a metric filter and alarm be established for changes to route tables.

Why It Matters: Monitoring changes to route tables will help ensure that all VPC traffic flows through an expected path.

3.14 Ensure a log metric filter and alarm exist for VPC changes

What It Means: Real-time monitoring of API calls can be achieved by directing CloudTrail Logs to CloudWatch Logs and establishing corresponding metric filters and alarms. It is possible to have more than 1 VPC within an account, in addition it is also possible to create a peer connection between 2 VPCs enabling network traffic to route between VPCs. It is recommended that a metric filter and alarm be established for changes made to VPCs.

Why It Matters: Monitoring changes to IAM policies will help ensure authentication and authorization controls remain intact.

4. Networking.

4.1 Ensure no security groups allow ingress from 0.0.0.0/0 to port 22

What It Means: Security groups provide stateful filtering of ingress/egress network traffic to AWS resources. It is recommended that no security group allows unrestricted ingress access to port 22.

Why It Matters: Removing unfettered connectivity to remote console services, such as SSH, reduces a server's exposure to risk.

4.2 Ensure no security groups allow ingress from 0.0.0.0/0 to port 3389

What It Means: Security groups provide stateful filtering of ingress/egress network traffic to AWS resources. It is recommended that no security group allows unrestricted ingress access to port 3389.

Why It Matters: Removing unfettered connectivity to remote console services, such as RDP, reduces a server's exposure to risk.

4.3 Ensure the default security group of every VPC restricts all traffic

What It Means: A VPC comes with a default security group whose initial settings deny all inbound traffic, allow all outbound traffic, and allow all traffic between instances assigned to the security group. If you don't specify a security group when you launch an instance, the instance is automatically assigned to this default security group. Security groups provide stateful filtering of ingress/egress network traffic to AWS resources. It is recommended that the default security group restrict all traffic.

The default VPC in every region should have its default security group updated to comply. Any newly created VPCs will automatically contain a default security group that will need remediation to comply with this recommendation.

Note: When implementing this recommendation, VPC flow logging is invaluable in determining the least privilege port access required by systems to work properly because it can log all packet acceptances and rejections occurring under the current security groups. This dramatically reduces the primary barrier to least privilege engineering - discovering the minimum ports required by systems in the environment. Even if the VPC flow logging recommendation in this benchmark is not adopted as a permanent security measure, it should be used during any period of discovery and engineering for least privileged security groups.

Why It Matters: Configuring all VPC default security groups to restrict all traffic will encourage least privilege security group development and mindful placement of AWS resources into security groups which will in-turn reduce the exposure of those resources.

4.4 Ensure routing tables for VPC peering are "least access"

What It Means: Once a VPC peering connection is established, routing tables must be updated to establish any connections between the peered VPCs. These routes can be as specific as desired - even peering a VPC to only a single host on the other side of the connection

Why It Matters: Being highly selective in peering routing tables is a very effective way of minimizing the impact of breach as resources outside of these routes are inaccessible to the peered VPC.

CIS Amazon Web Services (AWS) Foundations Benchmark Checklist.

To ensure compliance with all rules of the CIS Amazon Web Services Foundations Benchmark, a security team would need to check all AWS instances against the following set of rules to determine which instances are not compliant.

IAM

Let's start with logging. For each of the rules below, you'll need to check that every AWS instance and account passes.

Section	Rule	Category	# Pass/Fail
1.1	Avoid the use of the "root" account	IAM	
1.2	Ensure multi-factor authentication (MFA) is enabled for all IAM users that have a console password	IAM	
1.3	Ensure credentials unused for 90 days or greater are disabled	IAM	
1.4	Ensure access keys are rotated every 90 days or less	IAM	
1.5	Ensure IAM password policy requires at least one uppercase letter	IAM	
1.6	Ensure IAM password policy require at least one lowercase letter	IAM	
1.7	Ensure IAM password policy require at least one symbol	IAM	
1.8	Ensure IAM password policy require at least one number	IAM	
1.9	Ensure IAM password policy requires minimum length of 14 or greater	IAM	
1.10	Ensure IAM password policy prevents password reuse	IAM	
1.11	Ensure IAM password policy expires passwords within 90 days or less	IAM	
1.12	Ensure no root account access key exists	IAM	
1.13	Ensure MFA is enabled for the "root" account	IAM	
1.14	Ensure hardware MFA is enabled for the "root" account	IAM	
1.15	Ensure security questions are registered in the AWS account	IAM	
1.16	Ensure IAM policies are attached only to groups or roles	IAM	
1.17	Maintain current contact details	IAM	
1.18	Ensure security contact information is registered	IAM	
1.19	Ensure IAM instance roles are used for AWS resource access from instances	IAM	
1.20	Ensure a support role has been created to manage incidents with AWS Support	IAM	
1.21	Do not setup access keys during initial user setup for all IAM users that have a console password	IAM	
1.22	Ensure IAM policies that allow full "*:*" administrative privileges are not created	IAM	

Logging

Next, you'll want to go through the logging-related requirements to identify CloudTrail issues, CMK rotation, and VPC flow logging.

Section	Rule	Category	# Pass/Fail
2.1	Ensure CloudTrail is enabled in all regions	Logging	
2.2	Ensure CloudTrail log file validation is enabled	Logging	
2.3	Ensure the S3 bucket used to store CloudTrail logs is not publicly accessible	Logging	
2.4	Ensure CloudTrail trails are integrated with CloudWatch Logs	Logging	
2.5	Ensure AWS Config is enabled in all regions	Logging	
2.6	Ensure S3 bucket access logging is enabled on the CloudTrail S3 bucket	Logging	
2.7	Ensure CloudTrail logs are encrypted at rest using KMS CMKs	Logging	
2.8	Ensure rotation for customer created CMKs is enabled	Logging	
2.9	Ensure VPC flow logging is enabled in all VPCs	Logging	

Monitoring

These Monitoring rules center around log filtering and alerting.

Section	Rule	Category	# Pass/Fail
3.1	Ensure a log metric filter and alarm exist for unauthorized API calls	Monitoring	
3.2	Ensure a log metric filter and alarm exist for Management Console sign-in without MFA	Monitoring	
3.3	Ensure a log metric filter and alarm exist for usage of "root" account	Monitoring	
3.4	Ensure a log metric filter and alarm exist for IAM policy changes	Monitoring	
3.5	Ensure a log metric filter and alarm exist for CloudTrail configuration changes	Monitoring	
3.6	Ensure a log metric filter and alarm exist for AWS Management Console authentication failures	Monitoring	
3.7	Ensure a log metric filter and alarm exist for disabling or scheduled deletion of customer created CMKs	Monitoring	
3.8	Ensure a log metric filter and alarm exist for S3 bucket policy changes	Monitoring	
3.9	Ensure a log metric filter and alarm exist for AWS Config configuration changes	Monitoring	
3.10	Ensure a log metric filter and alarm exist for security group changes	Monitoring	
3.11	Ensure a log metric filter and alarm exist for changes to Network Access Control Lists (NACL)	Monitoring	
3.12	Ensure a log metric filter and alarm exist for changes to network gateways	Monitoring	
3.13	Ensure a log metric filter and alarm exist for route table changes	Monitoring	
3.14	Ensure a log metric filter and alarm exist for VPC changes	Monitoring	

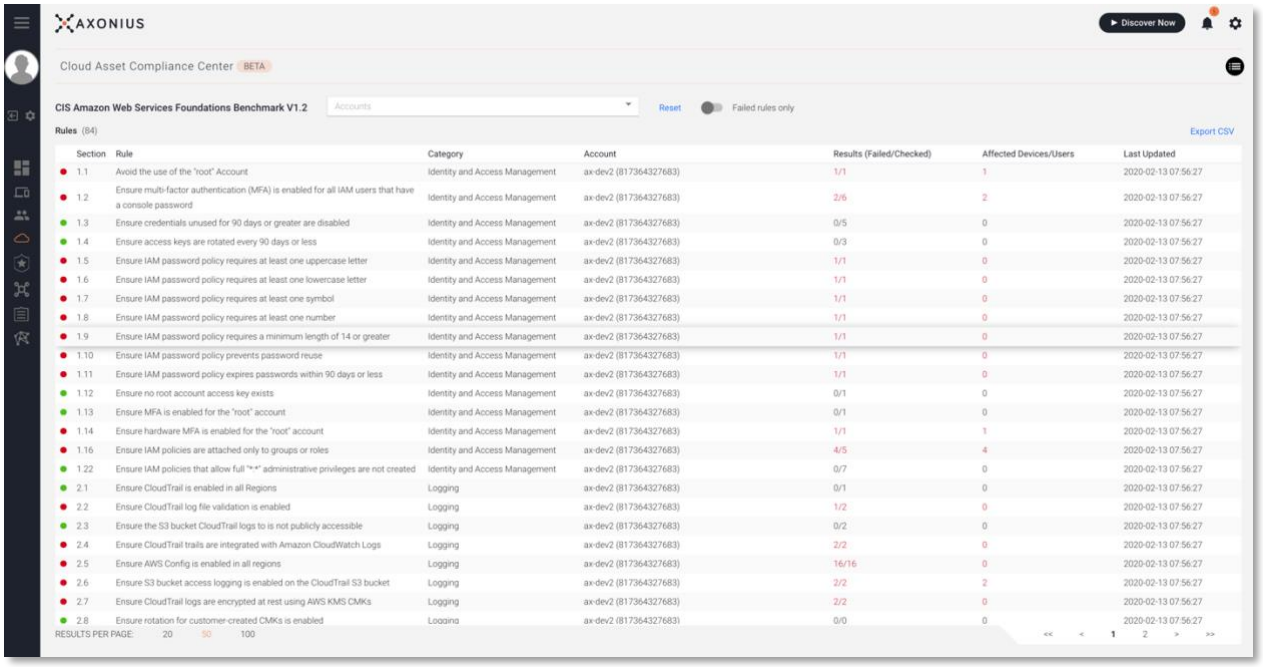
Networking

The final rules relate to port ingress, traffic restriction, and least access routing tables for VPC peering.

Section	Rule	Category	# Pass/Fail
4.1	Ensure no security groups allow ingress from 0.0.0.0/0 to port 22	Networking	
4.2	Ensure no security groups allow ingress from 0.0.0.0/0 to port 3389	Networking	
4.3	Ensure the default security group of every VPC restricts all traffic	Networking	
4.4	Ensure routing tables for VPC peering are "least access"	Networking	

All of these rules are valid and will improve any organization's AWS security

Cloud Asset Compliance for AWS



Launched in February 2020, the Axonius Cloud Asset Compliance for AWS add-on aggregates and correlates data from customers' AWS environments to show how each instance and account adheres to the scored rules defined in the CIS Amazon Web Services Foundations Benchmark 1.2.

About Axonius.

Axonius is the cybersecurity asset management platform that gives organizations a comprehensive asset inventory, uncovers security solution coverage gaps, and automatically validates and enforces security policies. By seamlessly integrating with over 200 security and management solutions, Axonius is deployed in minutes, improving cyber hygiene immediately.

Covering millions of devices at customers like the New York Times, Schneider Electric, Landmark Health, AppsFlyer, and many more, Axonius was named the Most Innovative Startup of 2019 at the prestigious RSAC Innovation Sandbox and was named to the CNBC Upstart 100 list and Forbes 20 Rising Stars.

For more information, please visit Axonius.com.