Any prior version of the Shadow Health CIS Compliance for AWS document that was released before March 2019 is no longer valid. This document is the most updated version and can be shared freely. It is recommended that you verify the valid status of this document by visiting:

https://support.shadowhealth.com/hc/en-us/sections/360000985473-IT-Legal
1. Identity and Access Management
   1.1 Avoid the use of the "root" account
   1.2 Ensure MFA is enabled for all IAM users that have a password
   1.3 Ensure credentials unused for 90 days or greater are disabled
   1.4 Ensure access keys are rotated every 90 days or less
   1.5 Ensure IAM password policy requires at least one uppercase letter
   1.6 Ensure IAM password policy requires at least one lowercase letter
   1.7 Ensure IAM password policy requires at least one symbol
   1.8 Ensure IAM password policy requires at least one number
   1.9 Ensure IAM password policy requires a minimum length of 14 or greater
   1.10 Ensure IAM password policy prevents password reuse
   1.11 Ensure IAM password policy expires passwords within 90 days or less
   1.12 Ensure no root account access key exists
   1.13 Ensure MFA is enabled for the "root" account
   1.14 Ensure hardware MFA is enabled for the "root" account
   1.15 Ensure security questions are registered in the AWS account
   1.16 Ensure IAM policies are attached only to groups or roles
   1.17 Maintain Current contact details
   1.18 Ensure Security contact information is registered
   1.19 Ensure IAM instance roles are used for AWS resource access from instances
   1.20 Ensure a support role has been created 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
   1.22 Ensure IAM policies that allow full "*:*" administrative privileges are not created

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

3. Monitoring
   3.1 Ensure a log metric filter and alarm exist for unauthorized API calls
   3.2 Ensure a log metric filter and alarm exist for Management Console sign-in without
      MFA
3.3 Ensure a log metric filter and alarm exist for usage of "root" account 8
3.4 Ensure a log metric filter and alarm exist for IAM policy changes 8
3.5 Ensure a log metric filter and alarm exist for CloudTrail configuration changes 8
3.6 Ensure a log metric filter and alarm exist for AWS Management Console authentication failures 9
3.7 Ensure a log metric filter and alarm exist for disabling or scheduled deletion of customer-created CMKs 9
3.8 Ensure a log metric filter and alarm exist for S3 bucket policy changes 9
3.9 Ensure a log metric filter and alarm exist for AWS Config configuration changes 9
3.10 Ensure a log metric filter and alarm exist for security group changes 9
3.11 Ensure a log metric filter and alarm exist for changes to Network Access Control Lists (NACL) 9
3.12 Ensure a log metric filter and alarm exist for changes to network gateways 9
3.13 Ensure a log metric filter and alarm exist for route table changes 10
3.14 Ensure a log metric filter and alarm exist for VPC changes 10

4. Networking 10
4.1 Ensure no security groups allow ingress from 0.0.0.0/0 to port 22 10
4.2 Ensure no security groups allow ingress from 0.0.0.0/0 to port 3389 10
4.3 Ensure the default security group of every VPC restricts all traffic 10
4.4 Ensure routing tables for VPC peering are "least access" 10
Shadow Health CIS Compliance

1. Identity and Access Management

1.1 Avoid the use of the "root" account
Root accounts are not used and credentials are not shared with anyone else. As is the industry best practice, Shadow Health leverages IAM Groups, Roles and Users to grant access to specific AWS resources. This is enforced by automated account logging and alerts.

1.2 Ensure MFA is enabled for all IAM users that have a password
For extra security, Shadow Health enables multi-factor authentication (MFA) for all IAM users. This is enforced by automated compliance reporting.

1.3 Ensure credentials unused for 90 days or greater are disabled
Shadow Health removes unused credentials, accounts, and access keys after 90 days via automated jobs.

1.4 Ensure access keys are rotated every 90 days or less
Shadow Health removes unused credentials, accounts, and access keys after 90 days via automated jobs.

1.5 Ensure IAM password policy requires at least one uppercase letter
Shadow Health adheres to a strong password policy as defined by CIS standards and industry best practices. These policies are enforced by configuration rules. This policy includes the requirement of a minimum of one uppercase letter.

1.6 Ensure IAM password policy requires at least one lowercase letter
Shadow Health adheres to a strong password policy as defined by CIS standards and industry best practices. These policies are enforced by configuration rules. This policy includes the requirement of a minimum of one lowercase letter.

1.7 Ensure IAM password policy requires at least one symbol
Shadow Health adheres to a strong password policy as defined by CIS standards and industry best practices. These policies are enforced by configuration rules. This policy includes the requirement of a minimum of one symbol.
1.8 Ensure IAM password policy requires at least one number
Shadow Health adheres to a strong password policy as defined by CIS standards and industry best practices. These policies are enforced by configuration rules. This policy includes the requirement of a minimum of one number.

1.9 Ensure IAM password policy requires a minimum length of 14 or greater
Shadow Health adheres to a strong password policy as defined by CIS standards and industry best practices. These policies are enforced by configuration rules. This policy includes the requirement of a minimum of 14 characters.

1.10 Ensure IAM password policy prevents password reuse
Shadow Health adheres to a strong password policy as defined by CIS standards and industry best practices. These policies are enforced by configuration rules. This policy includes the restriction of using a password that has already been used.

1.11 Ensure IAM password policy expires passwords within 90 days or less
Shadow Health adheres to a strong password policy as defined by CIS standards and industry best practices. These policies are enforced by configuration rules. This policy includes the expiration of passwords older than 90 days.

1.12 Ensure no root account access key exists
To avoid generating Access Keys for the root account, Shadow Health creates IAM users that have defined administrative privileges. The root account is monitored for access key creation with logging and alerts in place to ensure compliance.

1.13 Ensure MFA is enabled for the "root" account
Multi-Factor Authentication is enabled for Shadow Health’s root account. This is enforced by config rules.

1.14 Ensure hardware MFA is enabled for the "root" account
The Multi-Factor Authentication that is enabled for Shadow Health’s root account is hardware based. This is enforced by config rules.
1.15 Ensure security questions are registered in the AWS account
At Shadow Health, Security Questions are setup in each account to help recover root login access if it were ever lost.

1.16 Ensure IAM policies are attached only to groups or roles
At Shadow Health, we assign IAM Policies to IAM Groups or IAM Roles to reduce the complexity of access management as the number of users grow. This is enforced by a config rule.

1.17 Maintain Current contact details
AWS Uses contact details to alert the account owner and security contact when prohibitive or suspicious activities are observed within an account. At Shadow Health, we ensure that current and relevant contact details are maintained at all times.

1.18 Ensure Security contact information is registered
AWS Uses contact details to alert the account owner and security contact when prohibitive or suspicious activities are observed within an account. At Shadow Health, we ensure that current and relevant contact details are registered at all times.

1.19 Ensure IAM instance roles are used for AWS resource access from instances
This benchmark is not required for CIS compliance. At Shadow Health, we use RBAC security implementations to provide a greater level of protection and flexibility to our Kubernetes servers in lieu of the suggested IAM roles.

1.20 Ensure a support role has been created to manage incidents with AWS Support
Shadow Health maintains an IAM Role authorized 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
At Shadow Health, we take additional steps upon profile creation to understand the intent of usage and storage of keys.
1.22 Ensure IAM policies that allow full "*:*" administrative privileges are not created

Shadow Health ensures that IAM policies do not allow full administrative privileges and that the policies follow the principle of least privilege. This is enforced by config rules.

2. Logging

2.1 Ensure CloudTrail is enabled in all regions

Shadow Health ensures that CloudTrail is always enabled in all regions via custom functions that will ensure CloudTrail is enabled and immediately remedy any non-compliant accounts or regions.

2.2 Ensure CloudTrail log file validation is enabled

Shadow Health ensures that CloudTrail is always enabled in all regions via custom functions that will validate CloudTrail compliance and immediately remedy any non-compliant accounts or regions.

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

Config rules are used to ensure that Shadow Health CloudTrail buckets are never publicly accessible.

2.4 Ensure CloudTrail trails are integrated with CloudWatch Logs

Shadow Health ensures that CloudTrail trails are always sent to CloudWatch logs for processing. This is enforced by Config rules.

2.5 Ensure AWS Config is enabled in all regions

Shadow Health uses automation to ensure that Config is enabled on all regions of all accounts. This is enforced by Cloudformation.

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

Configuring logs to be placed in a separate bucket allows access to log information which can be useful in security and incident response workflows. Shadow Health ensures that Logging is enabled for all CloudTrail S3 Buckets. This is enforced by Config rules.
2.7 Ensure CloudTrail logs are encrypted at rest using KMS CMKs
Shadow Health configures CloudTrail to use SSE-KMS to provide additional confidentiality controls on log data. This is enforced by Config rules.

2.8 Ensure rotation for customer-created CMKs is enabled
Shadow Health rotates encryption keys to 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. This is enforced by Config rules.

2.9 Ensure VPC flow logging is enabled in all VPCs
Shadow Health has VPC Flow Logs enabled to provide visibility into network traffic that traverses the VPC and to detect anomalous traffic or insight during security workflows. This is enforced by Config rules.

3. Monitoring

3.1 Ensure a log metric filter and alarm exist for unauthorized API calls
Shadow Health monitors unauthorized API calls which will help reveal application errors and may reduce time to detect malicious activity. This is automated via CloudWatch alarms.

3.2 Ensure a log metric filter and alarm exist for Management Console sign-in without MFA
Shadow Health monitors for single-factor console logins to increase visibility into accounts that are not protected by MFA. This is automated via CloudWatch alarms.

3.3 Ensure a log metric filter and alarm exist for usage of "root" account
Shadow Health monitors for root account logins which will provide visibility into the use of a fully privileged account and an opportunity to reduce the use of it. This is automated via CloudWatch alarms.

3.4 Ensure a log metric filter and alarm exist for IAM policy changes
Shadow Health monitors changes to IAM policies which will help ensure authentication and authorization controls remain intact. This is automated via CloudWatch rules.
3.5 Ensure a log metric filter and alarm exist for CloudTrail configuration changes
Shadow Health monitors changes to CloudTrail’s configuration which will help ensure sustained visibility to activities performed in the AWS account. This is automated via CloudWatch rules.

3.6 Ensure a log metric filter and alarm exist for AWS Management Console authentication failures
Shadow Health monitors failed console logins to detect brute force attempts, which may provide an indicator, such as source IP, that can be used in other event correlation. This is automated via CloudWatch alarms.

3.7 Ensure a log metric filter and alarm exist for disabling or scheduled deletion of customer-created CMKs
Shadow Health monitors deletion or disabling of CMKs. Data encrypted with disabled or deleted keys will no longer be accessible. This is automated via CloudWatch alarms.

3.8 Ensure a log metric filter and alarm exist for S3 bucket policy changes
Shadow Health monitors changes to S3 bucket policies to reduce time to detect and correct permissive policies on sensitive S3 buckets. This is automated via CloudWatch rules.

3.9 Ensure a log metric filter and alarm exist for AWS Config configuration changes
Shadow Health monitors changes to AWS Config configurations which will help ensure sustained visibility of configuration items within the AWS account. This is automated via CloudWatch rules.

3.10 Ensure a log metric filter and alarm exist for security group changes
Shadow Health monitors changes to security groups which will help ensure that resources and services are not unintentionally exposed. This is automated via CloudWatch rules.

3.11 Ensure a log metric filter and alarm exist for changes to Network Access Control Lists (NACL)
Shadow Health monitors changes to NACLs to help ensure that AWS resources and services are not unintentionally exposed. This is automated via CloudWatch rules.
3.12 Ensure a log metric filter and alarm exist for changes to network gateways
Shadow Health monitors changes to network gateways which will help ensure that all ingress/egress traffic traverses the VPC border via a controlled path. This is automated via CloudWatch rules.

3.13 Ensure a log metric filter and alarm exist for route table changes
Shadow Health monitors changes to route tables to help ensure that all VPC traffic flows through an expected path. This is automated via CloudWatch rules.

3.14 Ensure a log metric filter and alarm exist for VPC changes
Shadow Health monitors changes to VPC configuration will help ensure that all VPCs remain intact. This is automated via CloudWatch rules.

4. Networking

4.1 Ensure no security groups allow ingress from 0.0.0.0/0 to port 22
Shadow Health removes unfettered connectivity to remote console services, such as SSH, reduces a server’s exposure to risk. This is enforced by Config rules.

4.2 Ensure no security groups allow ingress from 0.0.0.0/0 to port 3389
Shadow Health removes unfettered connectivity to remote console services, such as RDP, reduces a server's exposure to risk. This is enforced by Config rules.

4.3 Ensure the default security group of every VPC restricts all traffic
Shadow Health configures all VPC default security groups to restrict all traffic. This 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. This is enforced by Config rules.

4.4 Ensure routing tables for VPC peering are "least access"
This benchmark is not required for CIS compliance. Shadow Health is highly selective in peering routing tables and always ensures that VPC peering is kept to a “least access” methodology. This is a very effective way of minimizing the impact of a breach as resources outside of these routes are inaccessible to the peered VPC.