# 1 Control Plane Components

This section consists of security recommendations for the direct configuration of Kubernetes control plane processes. These recommendations assume that the OpenShift cluster has 3 master nodes, as that is the default configuration on installation. These recommendations may not be directly applicable for cluster operators in environments where these components are managed by a 3rd party such as OpenShift Dedicated, Azure Red Hat OpenShift or Red Hat OpenShift Service on AWS.  
All Audit and Remediation commands assume that you are logged into the OpenShift cluster with the cluster admin role, cluster bound.  
Internal Only - General

## 1.1 Master Node Configuration Files

Internal Only - General

|  |  |  |  |
| --- | --- | --- | --- |
| Control Reference ID | Control Name & Description | Description | Remediation |
| 1.1.2 | Ensure that the API server pod specification file ownership is set to root:root (Manual) | Ensure that the API server pod specification file ownership is set to root:root. | No remediation required; file permissions are managed by the operator. Internal Only - General |
| 1.1.6 | Ensure that the scheduler pod specification file ownership is set to root:root (Manual) | Ensure that the scheduler pod specification file ownership is set to root:root. | No remediation required; file permissions are managed by the operator. |
| 1.1.8 | Ensure that the etcd pod specification file ownership is set to root:root (Manual) | Ensure that the /etc/kubernetes/manifests/etcd.yaml file ownership is set to root:root. | No remediation required; file permissions are managed by the operator. |
| 1.1.10 | Ensure that the Container Network Interface file ownership is set to root:root (Manual) |  |  |
| 1.1.12 | Ensure that the etcd data directory ownership is set to etcd:etcd (Manual) | Ensure that the etcd data directory ownership is set to etcd:etcd. | No remediation required; file ownership is managed by the operator. Internal Only - General |
| 1.1.13 | Ensure that the kubeconfig file permissions are set to 600 or more restrictive (Manual) |  |  |
| 1.1.14 | Ensure that the kubeconfig file ownership is set to root:root (Manual) | Ensure that the kubeconfig file ownership is set to root:root. | No remediation required; file permissions are managed by the operator. |
| 1.1.16 | Ensure that the Scheduler kubeconfig file ownership is set to root:root (Manual) | Ensure that the kubeconfig file ownership is set to root:root. | No remediation required; file permissions are managed by the operator. |
| 1.1.19 | Ensure that the OpenShift PKI directory and file ownership is set to root:root (Manual) |  |  |
| 1.1.21 | Ensure that the OpenShift PKI key file permissions are set to 600 (Manual) | Ensure that the OpenShift PKI key files have permissions of 600. | No remediation required; file permissions are managed by the operator. |

### 1.1.1 Ensure that the API server pod specification file permissions are set to 600 or more

Internal Only - General

### 1.1.23 with revised verbiage

to reflect latest version options. Ticket # 18972  
Revised path for Kubelet CA on recommendation 1.2.6 on Ticket #18929  
Revised path for certificates for recommendation 1.2.5 on Ticket # 18928  
Updated default section of recommendation 1.2.9 on Ticket # 18880  
Modified verification settings to reflect OCP 4.6+ for recommendation 1.2.4 on Ticket #18873  
Removed references to end- of-life versions of OCP.  
Date  
6/01/2023  
Version  
V1.4.0  
6/01/2023  
V1.4.0  
6/01/2023  
V1.4.0  
6/01/2023  
V1.4.0  
6/01/2023  
V1.4.0  
6/01/2023  
V1.4.0  
6/01/2023  
V1.4.0  
6/01/2023  
V1.4.0  
6/01/2023  
V1.4.0  
Internal Only - General  
Changes for this version  
Revised the insecure port option to specifically reference OCP v4.6+ on Ticket #18892  
Removed the recommendation that references Kubelet Server Certificates from the Master Node section on Ticket # 18891  
Adjustment to the audit process made for recommendation 1.2.16 on Ticket # 18888  
Added aescgm encryption reference for recommendation

## 1.2 API Server

This section contains recommendations relating to API server configuration flags.  
OpenShift includes two API servers, the OpenShift API server and the Kubernetes API server. All API calls are directed to the Open Shift API server and then Kubernetes objects are delegated to kube-apiserver.  
The cluster configuration resource (CR) APIServer holds configuration settings (like serving certificates, client CA and CORS domains) shared by all API servers in the system, among them especially kube-apiserver and openshift-apiserver. The canonical name of an instance is 'cluster'. Changes to the API server configurations should be done in the APIServer custom resource definition (CRD): apiservers.config.openshift.io  
The OpenShift API Server is managed by the openshift-apiserver-operator  
The Kubernetes API Server is managed by the openshift-kube-apiserveroperator  
Both are managed by the Cluster Version Operator  
Internal Only - General

|  |  |  |  |
| --- | --- | --- | --- |
| Control Reference ID | Control Name & Description | Description | Remediation |
| 1.2.1 | Ensure that anonymous requests are authorized (Manual) | When anonymous requests to the API server are allowed, they must be authorized. | None. The default configuration should not be modified. Internal Only - General |
| 1.2.2 | Use https for kubelet connections (Manual) | Use https for kubelet connections. | No remediation is required. OpenShift platform components use X.509 certificates for authentication. OpenShift manages the CAs and certificates for platform components. This is not configurable. |
| 1.2.3 | Ensure that the kubelet uses certificates to authenticate (Manual) | Enable certificate based kubelet authentication. | No remediation is required. OpenShift platform components use X.509 certificates for authentication. OpenShift manages the CAs and certificates for platform components. This is not configurable. |
| 1.2.4 | Verify that the kubelet certificate authority is set as appropriate (Manual) | Verify kubelet's certificate before establishing connection. | No remediation is required. OpenShift platform components use X.509 certificates for authentication. OpenShift manages the CAs and certificates for platform components. This is not configurable. Internal Only - General |
| 1.2.5 | Ensure that the --authorization-mode argument is not set to AlwaysAllow (Manual) | Do not always authorize all requests. | None. RBAC is always on and the OpenShift API server does not use the values assigned to the flag authorization-mode. |
| 1.2.6 | Verify that RBAC is enabled (Manual) | Turn on Role Based Access Control. | None. |
| 1.2.7 | Ensure that the APIPriorityAndFairness feature gate is enabled (Manual) | Limit the rate at which the API server accepts requests. | No remediation is required. |
| 1.2.8 | Ensure that the admission control plugin AlwaysAdmit is not set (Manual) | Do not allow all requests. | None. |
| 1.2.9 | Ensure that the admission control plugin AlwaysPullImages is not set (Manual) | Always pull images. | None. |
| 1.2.10 | Ensure that the admission control plugin ServiceAccount is set (Manual) | Automate service accounts management. | None. |
| 1.2.11 | Ensure that the admission control plugin NamespaceLifecycle is set (Manual) | Reject creating objects in a namespace that is undergoing termination. | None. |
| 1.2.12 | Ensure that the admission control plugin SecurityContextConstraint is set (Manual) | Reject creating pods that do not match Pod Security Policies. | None. Internal Only - General |
| 1.2.13 | Ensure that the admission control plugin NodeRestriction is set (Manual) | Limit the Node and Pod objects that a kubelet could modify. | None. |
| 1.2.14 | Ensure that the --insecure-bind-address argument is not set (Manual) | Do not bind the insecure API service. | None. |
| 1.2.15 | Ensure that the --insecure-port argument is set to 0 (Manual) | Do not bind to insecure port. | None. Internal Only - General |
| 1.2.16 | Ensure that the --secure-port argument is not set to 0 (Manual) | Do not disable the secure port. | None. |
| 1.2.17 | Ensure that the healthz endpoint is protected by RBAC (Manual) | Disable profiling, if not needed. | None. |
| 1.2.18 | Ensure that the --audit-log-path argument is set (Manual) | Enable auditing on the Kubernetes API Server and set the desired audit log path. | None required. This is managed by the cluster apiserver operator. |
| 1.2.19 | Ensure that the audit logs are forwarded off the cluster for retention (Manual) | Retain the logs for at least 30 days or as appropriate. | Follow the documentation for log forwarding. Forwarding logs to third party systems. Internal Only - General |
| 1.2.21 | Configure Kubernetes API Server Maximum Audit Log Size (Manual) | Audit logs are rotated upon reaching a maximum size, which is set to 100 MB or greater by default. | The audit-log-maxsize parameter is set by default by Red Hat and not supported to change. |
| 1.2.22 | Ensure that the --request-timeout argument is set (Manual) | The API server minimum request timeout defines the minimum number of seconds a handler must keep a request open before timing it out. | None |
| 1.2.23 | Ensure that the --service-account-lookup argument is set to true (Manual) | Validate service account before validating token. | None. |
| 1.2.24 | Ensure that the --service-account-key-file argument is set as appropriate (Manual) | Explicitly set a service account public key file for service accounts on the apiserver. | The OpenShift API server does not use the service-account-key-file argument. The ServiceAccount token authenticator is configured with serviceAccountConfig.publicKeyFiles. OpenShift does not reuse the apiserver TLS key. This is not configurable. Internal Only - General |
| 1.2.27 | Ensure that the --client-ca-file argument is set as appropriate (Manual) | Setup TLS connection on the API server. | None. |
| 1.2.28 | Ensure that the --etcd-cafile argument is set as appropriate (Manual) | etcd should be configured to make use of TLS encryption for client connections. | None. Internal Only - General |
| 1.2.29 | Ensure that encryption providers are appropriately configured (Manual) | Where etcd encryption is used, appropriate providers should be configured. | Follow the OpenShift documentation for encrypting etcd data. |
| 1.2.30 | Ensure that the API Server only makes use of Strong Cryptographic Ciphers (Manual) |  |  |
| 1.2.31 | Ensure unsupported configuration overrides are not used (Manual) | OpenShift supported an option called unsupportedConfigOverrides that allowed users to opt into unsupported behavior. This option is no longer supported by OpenShift and should not be used. | None. |

### 1.2.20 Ensure that the maximumRetainedFiles argument is set to 10 or as appropriate

Internal Only - General

## 1.3 Controller Manager

This section contains recommendations relating to Controller Manager configuration flags. In OpenShift 4, the Controller Manager is managed with the cluster Controller Manager Operator. There are two operators: the OpenShift Controller operator and the Kube Controller operator. The OpenShift Controller Manager operator manages the OpenShift Controller Manager. The Kubernetes Controller Manager operator manages and updates the Kubernetes Controller Manager deployed on top of OpenShift. All calls are directed to the OpenShift Controller Manager and then Kubernetes objects are delegated to the Kubernetes Controller Manager.  
Internal Only - General

|  |  |  |  |
| --- | --- | --- | --- |
| Control Reference ID | Control Name & Description | Description | Remediation |
| 1.3.1 | Ensure that controller manager healthz endpoints are protected by RBAC (Manual) | Disable profiling, if not needed. | None. |
| 1.3.2 | Ensure that the --use-service-account-credentials argument is set to true (Manual) | Use individual service account credentials for each controller. | None. |
| 1.3.4 | Ensure that the --root-ca-file argument is set as appropriate (Manual) | Allow pods to verify the API server's serving certificate before establishing connections. | None. |

## 1.4 Scheduler

This section contains recommendations relating to Scheduler configuration flags.  
In OpenShift 4, the Scheduler is managed with the Kubernetes Scheduler Operator. The Kubernetes Scheduler Operator manages and updates the Kubernetes Scheduler deployed on top of OpenShift Container Platform. The operator is installed with the Cluster Version Operator (CVO). The Kubernetes Scheduler Operator contains the following components:  
• Operator • Bootstrap manifest renderer • • Configuration observer  
Installer based on static pods  
By default, the Operator exposes Prometheus metrics through the metrics service  
Internal Only - General

|  |  |  |  |
| --- | --- | --- | --- |
| Control Reference ID | Control Name & Description | Description | Remediation |
| 1.4.1 | Ensure that the healthz endpoints for the scheduler are protected by RBAC (Manual) | Disable profiling, if not needed. | None. |
| 1.4.2 | Verify that the scheduler API service is protected by RBAC (Manual) | Do not bind the scheduler service to non-loopback insecure addresses. | None. |

## 1.8 Utilize Client Certificates to Authenticate Hardware

Assets Use client certificates to authenticate hardware assets connecting to the organization's trusted network.  
v8  
v7  
● ●  
●  
Internal Only - General  
MITRE ATT&CK Mappings:  
Techniques / Subtechniques  
Tactics  
Mitigations  
T1048, T1189  
TA0001, TA0010  
M1041  
Internal Only - General

# 2 etcd

This section covers recommendations for etcd configuration. The OpenShift clusteretcd-operator (CEO) is an operator that handles the scaling of etcd during cluster bootstrap and regular operation. The operator also manages provisioning etcd dependencies such as TLS certificates.  
OpenShift uses X.509 certificates to provide secure communication to etcd. OpenShift generates these files and sets the arguments appropriately. etcd certificates are used for encrypted communication between etcd member peers, as well as encrypted client traffic. The following certificates are generated and used by etcd and other processes that communicate with etcd:  
• Peer certificates: Used for communication between etcd members. • Client certificates: Used for encrypted server-client communication. Client  
certificates are currently used by the API server only, and no other service should connect to etcd directly except for the proxy. Client secrets (etcd-client, etcdmetric-client, etcd-metric-signer, and etcd-signer) are added to the openshiftconfig, openshift-monitoring, and openshift-kube-apiserver namespaces. • Server certificates: Used by the etcd server for authenticating client requests. • Metric certificates: All metric consumers connect to proxy with metric-client  
certificates.  
Internal Only - General

|  |  |  |  |
| --- | --- | --- | --- |
| Control Reference ID | Control Name & Description | Description | Remediation |
| 2.1 | Ensure that the --cert-file and --key-file arguments are set as appropriate (Manual) | Configure TLS encryption for the etcd service. | OpenShift does not use the etcd-certfile or etcd-keyfile flags. Certificates for etcd are managed by the etcd cluster operator. |
| 2.3 | Ensure that the --auto-tls argument is not set to true (Manual) | Do not use self-signed certificates for TLS. | This setting is managed by the cluster etcd operator. No remediation required. |
| 2.5 | Ensure that the --peer-client-cert-auth argument is set to true (Manual) | etcd should be configured for peer authentication. | This setting is managed by the cluster etcd operator. No remediation required. |
| 2.6 | Ensure that the --peer-auto-tls argument is not set to true (Manual) | Do not use automatically generated self-signed certificates for TLS connections between peers. | This setting is managed by the cluster etcd operator. No remediation required. |
| 2.7 | Ensure that a unique Certificate Authority is used for etcd (Manual) | Use a different certificate authority for etcd from the one used for Kubernetes. | None required. Certificates for etcd are managed by the OpenShift cluster etcd operator. |

## 2.2 Ensure Authorized Software is Currently Supported

Ensure that only currently supported software is designated as authorized in the software inventory for enterprise assets. If software is unsupported, yet necessary for the fulfillment of the enterprise’s mission, document an exception detailing mitigating controls and residual risk acceptance. For any unsupported software without an exception documentation, designate as unauthorized. Review the software list to verify software support at least monthly, or more frequently.  
● ● ●

## 2.8 as it is same as 1.2.33 on

Ticket # 18855  
Improved remediation of active garage collection process for recommendation  
Updated the audit procedure to get the streaming connection timeout runtime config for recommendation

# 3 Control Plane Configuration

This section contains recommendations for cluster-wide areas, such as authentication and logging. Unlike section 1 these recommendations should apply to all deployments.  
Internal Only - General

## 3.1 Authentication and Authorization

Internal Only - General

|  |  |  |  |
| --- | --- | --- | --- |
| Control Reference ID | Control Name & Description | Description | Remediation |
| 3.1.1 | Client certificate authentication should not be used for users (Manual) | Kubernetes provides the option to use client certificates for user authentication. However as there is no way to revoke these certificates when a user leaves an organization or loses their credential, they are not suitable for this purpose. It is not possible to fully disable client certificate use within a cluster as it is used for component to component authentication. | Configure an identity provider for the OpenShift cluster following the OpenShift documentation. Once an identity provider has been defined, you can use RBAC to define and apply permissions. After you define an identity provider and create a new cluster-admin user you can reduce the attack surface by removing the default kubeadmin user. |

## 3.2 Logging

Internal Only - General

|  |  |  |  |
| --- | --- | --- | --- |
| Control Reference ID | Control Name & Description | Description | Remediation |
| 3.2.1 | Ensure that a minimal audit policy is created (Manual) | Kubernetes can audit the details of requests made to the API server. | None. |
| 3.2.2 | Ensure that the audit policy covers key security concerns (Manual) | Ensure that the audit policy created for the cluster covers key security concerns. | Update the audit log policy profile to use WriteRequestBodies. Internal Only - General |

## 3.3 Configure Data Access Control Lists

Configure data access control lists based on a user’s need to know. Apply data access control lists, also known as access permissions, to local and remote file systems, databases, and applications.  
● ● ●

## 3.10 Encrypt Sensitive Data in Transit

Encrypt sensitive data in transit. Example implementations can include: Transport Layer Security (TLS) and Open Secure Shell (OpenSSH).

## 3.11 Encrypt Sensitive Data at Rest

Encrypt sensitive data at rest on servers, applications, and databases containing sensitive data. Storage-layer encryption, also known as server-side encryption, meets the minimum requirement of this Safeguard. Additional encryption methods may include application-layer encryption, also known as client-side encryption, where access to the data storage device(s) does not permit access to the plain-text data.

## 3.12 Segment Data Processing and Storage Based on

Sensitivity Segment data processing and storage based on the sensitivity of the data. Do not process sensitive data on enterprise assets intended for lower sensitivity data.

# 4 Worker Nodes

This section consists of security recommendations for the components that run on Kubernetes worker nodes.  
Note that these components may also run on Kubernetes master nodes, so the recommendations in this section should be applied to master nodes as well as worker nodes where the master nodes make use of these components.  
Internal Only - General

## 4.1 Establish and Maintain a Secure Configuration Process

Establish and maintain a secure configuration process for enterprise assets (end-user devices, including portable and mobile, non-computing/IoT devices, and servers) and software (operating systems and applications). Review and update documentation annually, or when significant enterprise changes occur that could impact this Safeguard.  
● ● ●

|  |  |  |  |
| --- | --- | --- | --- |
| Control Reference ID | Control Name & Description | Description | Remediation |
| 4.1.2 | Ensure that the kubelet service file ownership is set to root:root (Automated) | Ensure that the kubelet service file ownership is set to root:root. | None. |
| 4.1.4 | If proxy kubeconfig file exists ensure ownership is set to root:root (Manual) | If kube-proxy is running, ensure that the file ownership of its kubeconfig file is set to root:root. | None required. The configuration is managed by OpenShift operators. |
| 4.1.6 | Ensure that the --kubeconfig kubelet.conf file ownership is set to root:root (Automated) |  |  |
| 4.1.8 | Ensure that the client certificate authorities file ownership is set to root:root (Automated) |  |  |
| 4.1.10 | Ensure that the kubelet configuration file ownership is set to root:root (Automated) | Ensure that if the kubelet refers to a configuration file with the --config argument, that file is owned by root:root. | None. Internal Only - General |

### 4.1.9 Ensure that the kubelet --config configuration file has permissions set to 600 or more

Internal Only - General

## 4.2 Kubelet

This section contains recommendations for kubelet configuration. In OpenShift 4, the kubelet is managed by the Machine Config Operator.  
Internal Only - General

|  |  |  |  |
| --- | --- | --- | --- |
| Control Reference ID | Control Name & Description | Description | Remediation |
| 4.2.1 | Activate Garbage collection in OpenShift Container Platform 4, as appropriate (Manual) |  |  |
| 4.2.2 | Ensure that the --anonymous-auth argument is set to false (Automated) | Disable anonymous requests to the Kubelet server. | Create a kubeletconfig to explicitly disable anonymous authentication. Examples of how to do this can be found in the OpenShift documentation. |
| 4.2.3 | Ensure that the --authorization-mode argument is not set to AlwaysAllow (Automated) |  |  |
| 4.2.4 | Ensure that the --client-ca-file argument is set as appropriate (Automated) | Enable Kubelet authentication using certificates. | None. Changing the clientCAFile value is unsupported. |
| 4.2.5 | Verify that the read only port is not used or is set to 0 (Automated) | Disable the read-only port. | In earlier versions of OpenShift 4, the read-only-port argument is not used. Follow the instructions in the documentation to create a kubeletconfig CRD and set the kubelet-read-only-port is set to 0. |
| 4.2.6 | Ensure that the --streaming-connection-idle-timeout argument is not set to 0 (Automated) | Do not disable timeouts on streaming connections. | Follow the instructions in the documentation to create a kubeletconfig CRD and set the streamingConnectionIdleTimeout to the desired value. Do not set the value to 0. |
| 4.2.7 | Ensure that the --make-iptables-util-chains argument is set to true (Manual) | Allow Kubelet to manage iptables. | None. |
| 4.2.10 | Ensure that the --rotate-certificates argument is not set to false (Manual) | Enable kubelet client certificate rotation. | None. |
| 4.2.11 | Verify that the RotateKubeletServerCertificate argument is set to true (Manual) | Enable kubelet server certificate rotation. | None. Internal Only - General |
| 4.2.12 | Ensure that the Kubelet only makes use of Strong Cryptographic Ciphers (Manual) | Ensure that the Kubelet is configured to only use strong cryptographic ciphers. | Follow the directions above and in the OpenShift documentation to configure the tlsSecurityProfile. Configuring Ingress. Please reference the OpenShift TLS security profile documentation for more detail on each profile. |

## 4.3 Ensure the Use of Dedicated Administrative Accounts

Ensure that all users with administrative account access use a dedicated or secondary account for elevated activities. This account should only be used for administrative activities and not internet browsing, email, or similar activities.  
● ● ●  
MITRE ATT&CK Mappings:  
Techniques / Subtechniques  
Tactics  
Mitigations  
T1083, T1222  
TA0005, TA0007  
M1026  
Internal Only - General

## 4.4 Use Unique Passwords

Where multi-factor authentication is not supported (such as local administrator, root, or service accounts), accounts will use passwords that are unique to that system.  
● ●  
MITRE ATT&CK Mappings:  
Techniques / Subtechniques  
T1078  
Tactics  
TA0001  
Mitigations  
M1026  
Internal Only - General

## 4.7 Limit Access to Script Tools

Limit access to scripting tools (such as Microsoft PowerShell and Python) to only administrative or development users with the need to access those capabilities.  
● ●  
v8  
v7  
Internal Only - General

## 4.8 Uninstall or Disable Unnecessary Services on

Enterprise Assets and Software Uninstall or disable unnecessary services on enterprise assets and software, such as an unused file sharing service, web application module, or service function.  
● ●  
Internal Only - General  
Controls Version  
Control  
IG 1 IG 2 IG 3  
v7

## 4.9 Configure Trusted DNS Servers on Enterprise Assets

Configure trusted DNS servers on enterprise assets. Example implementations include: configuring assets to use enterprise-controlled DNS servers and/or reputable externally accessible DNS servers.  
● ●

# 5 Policies

This section contains recommendations for various Kubernetes policies which are important to the security of the environment.  
Internal Only - General

## 5.1 Establish Secure Configurations

Maintain documented, standard security configuration standards for all authorized operating systems and software.  
● ● ●  
v8  
v7  
MITRE ATT&CK Mappings:  
Techniques / Subtechniques  
T1499  
Tactics  
TA0040  
Mitigations  
M1037  
Internal Only - General

|  |  |  |  |
| --- | --- | --- | --- |
| Control Reference ID | Control Name & Description | Description | Remediation |
| 5.1.1 | Ensure that the cluster-admin role is only used where required (Manual) | The RBAC role cluster-admin provides wide-ranging powers over the environment and should be used only where and when needed. | Identify all clusterrolebindings to the cluster-admin role. Check if they are used and if they need this role or if they could use a role with fewer privileges. Where possible, first bind users to a lower privileged role and then remove the clusterrolebinding to the cluster-admin role : oc delete clusterrolebinding [name] |
| 5.1.2 | Minimize access to secrets (Manual) | The Kubernetes API stores secrets, which may be service account tokens for the Kubernetes API or credentials used by workloads in the cluster. Access to these secrets should be restricted to the smallest possible group of users to reduce the risk of privilege escalation. | Where possible, remove get, list and watch access to secret objects in the cluster. |
| 5.1.3 | Minimize wildcard use in Roles and ClusterRoles (Manual) | Kubernetes Roles and ClusterRoles provide access to resources based on sets of objects and actions that can be taken on those objects. It is possible to set either of these to be the wildcard "\*" which matches all items. Use of wildcards is not optimal from a security perspective as it may allow for inadvertent access to be granted when new resources are added to the Kubernetes API either as CRDs or in later versions of the product. | Where possible replace any use of wildcards in clusterroles and roles with specific objects or actions. |
| 5.1.4 | Minimize access to create pods (Manual) | The ability to create pods in a namespace can provide a number of opportunities for privilege escalation, such as assigning privileged service accounts to these pods or mounting hostPaths with access to sensitive data (unless Pod Security Policies are implemented to restrict this access) As such, access to create new pods should be restricted to the smallest possible group of users. | Where possible, remove create access to pod objects in the cluster. |
| 5.1.5 | Ensure that default service accounts are not actively used. (Manual) | The default service account should not be used to ensure that rights granted to applications can be more easily audited and reviewed. | None required. |
| 5.1.6 | Ensure that Service Account Tokens are only mounted where necessary (Manual) | Service accounts tokens should not be mounted in pods except where the workload running in the pod explicitly needs to communicate with the API server | Modify the definition of pods and service accounts which do not need to mount service account tokens to disable it. |

## 5.2 Use Unique Passwords

Use unique passwords for all enterprise assets. Best practice implementation includes, at a minimum, an 8-character password for accounts using MFA and a 14-character password for accounts not using MFA.  
● ● ●

|  |  |  |  |
| --- | --- | --- | --- |
| Control Reference ID | Control Name & Description | Description | Remediation |
| 5.2.1 | Minimize the admission of privileged containers (Manual) | Do not generally permit containers to be run with the securityContext.privileged flag set to true. | Create an SCC that sets allowPrivilegedContainer to false and take it into use by assigning it to applicable users and groups. |
| 5.2.5 | Minimize the admission of containers with allowPrivilegeEscalation (Manual) | Do not generally permit containers to be run with the allowPrivilegeEscalation flag set to true. | Create an SCC that sets allowPrivilegeEscalation to false and take it into use by assigning it to applicable users and groups. |
| 5.2.6 | Minimize the admission of root containers (Manual) | Do not generally permit containers to be run as the root user. | None required. By default, OpenShift includes the nonroot and nonroot-v2 SCCs that restrict the ability to run as nonroot. If additional SCCs are appropriate, follow the OpenShift documentation to create custom SCCs. Internal Only - General |
| 5.2.7 | Minimize the admission of containers with the NET\_RAW capability (Manual) | Do not generally permit containers with the potentially dangerous NET\_RAW capability. | Create an SCC that sets requiredDropCapabilities to include ALL or at least NET\_RAW and take it into use by assigning it to applicable users and groups. |
| 5.2.8 | Minimize the admission of containers with added capabilities (Manual) | Do not generally permit containers with capabilities assigned beyond the default set. | Utilize the restricted-v2 SCC or create an SCC that sets allowedCapabilities and defaultAddCapabilities to an empty list and take it into use by assigning it to applicable users and groups. Internal Only - General |
| 5.2.9 | Minimize the admission of containers with capabilities assigned (Manual) | Do not generally permit containers with capabilities. | Review the use of capabilities in applications running on your cluster. Where a namespace contains applications which do not require any Linux capabilities to operate, consider adding a SCC which forbids the admission of containers which do not drop all capabilities. |
| 5.2.10 | Minimize access to privileged Security Context Constraints (Manual) | OpenShift has the concept of Security Context Constraints (SCCs) that supplement the Pod Security Admission controller. SCCs allow you to group elevated container capabilities and assign those capabilities to users and groups. For example, you can have an SCC that restricts the ability to launch privileged containers and assign that SCC to all authenticated users. As a result, users requesting a pod that contains a privileged container will be rejected. You can find more information on SCCs in the OpenShift documentation. | Remove any users and groups who do not need access to an SCC, following the principle of least privilege. You can remove users and groups from an SCC using the oc edit scc $NAME command. Additionally, you can create your own SCCs that contain the container functionality you need for a particular use case and assign that SCC to users and groups if the default SCCs are not appropriate for your use case. |

## 5.3 Disable Dormant Accounts

Delete or disable any dormant accounts after a period of 45 days of inactivity, where supported.  
● ● ●

|  |  |  |  |
| --- | --- | --- | --- |
| Control Reference ID | Control Name & Description | Description | Remediation |
| 5.3.1 | Ensure that the CNI in use supports Network Policies (Manual) | There are a variety of CNI plugins available for Kubernetes. If the CNI in use does not support Network Policies it may not be possible to effectively restrict traffic in the cluster. | None required. |
| 5.3.2 | Ensure that all Namespaces have Network Policies defined (Manual) | Use network policies to isolate traffic in your cluster network. | Follow the documentation and create NetworkPolicy objects as you need them. Internal Only - General |

## 5.4 Restrict Administrator Privileges to Dedicated

Administrator Accounts Restrict administrator privileges to dedicated administrator accounts on enterprise assets. Conduct general computing activities, such as internet browsing, email, and productivity suite use, from the user’s primary, non-privileged account.  
● ● ●

|  |  |  |  |
| --- | --- | --- | --- |
| Control Reference ID | Control Name & Description | Description | Remediation |
| 5.4.1 | Prefer using secrets as files over secrets as environment variables (Manual) | Kubernetes supports mounting secrets as data volumes or as environment variables. Minimize the use of environment variable secrets. | If possible, rewrite application code to read secrets from mounted secret files, rather than from environment variables. |
| 5.4.2 | Consider external secret storage (Manual) | Consider the use of an external secrets storage and management system, instead of using Kubernetes Secrets directly, if you have more complex secret management needs. Ensure the solution requires authentication to access secrets, has auditing of access to and use of secrets, and encrypts secrets. Some solutions also make it easier to rotate secrets. | Refer to the secrets management options offered by your cloud provider or a third-party secrets management solution. |

## 5.5 Extensible Admission Control

Internal Only - General

|  |  |  |  |
| --- | --- | --- | --- |
| Control Reference ID | Control Name & Description | Description | Remediation |
| 5.5.1 | Configure Image Provenance using image controller configuration parameters (Manual) |  |  |

## 5.7 General Policies

These policies relate to general cluster management topics, like namespace best practices and policies applied to pod objects in the cluster.  
Internal Only - General

|  |  |  |  |
| --- | --- | --- | --- |
| Control Reference ID | Control Name & Description | Description | Remediation |
| 5.7.1 | Create administrative boundaries between resources using namespaces (Manual) | Use namespaces to isolate your Kubernetes objects. | Follow the documentation and create namespaces for objects in your deployment as you need them. |
| 5.7.3 | Apply Security Context to Your Pods and Containers (Manual) | Apply Security Context to Your Pods and Containers | Follow the Kubernetes documentation and apply security contexts to your pods. For a suggested list of security contexts, you may refer to the CIS Security Benchmark for Docker Containers. |
| 5.7.4 | The default namespace should not be used (Manual) | Kubernetes provides a default namespace, where objects are placed if no namespace is specified for them. Placing objects in this namespace makes application of RBAC and other controls more difficult. | Ensure that namespaces are created to allow for appropriate segregation of Kubernetes resources and that all new resources are created in a specific namespace. |

### 5.7.2 Ensure that the seccomp profile is set to docker/default in your pod definitions (Manual)

Internal Only - General  
Appendix: Change History ................................................ Error! Bookmark not defined.  
Internal Only - General  
Overview  
All CIS Benchmarks™ (Benchmarks) focus on technical configuration settings used to maintain and/or increase the security of the addressed technology, and they should be used in conjunction with other essential cyber hygiene tasks like:  
• Monitoring the base operating system and applications for vulnerabilities and  
quickly updating with the latest security patches.  
• End-point protection (Antivirus software, Endpoint Detection and Response  
(EDR), etc.).  
• Logging and monitoring user and system activity.  
In the end, the Benchmarks are designed to be a key component of a comprehensive cybersecurity program.  
Important Usage Information  
All Benchmarks are available free for non-commercial use from the CIS Website. They can be used to manually assess and remediate systems and applications. In lieu of manual assessment and remediation, there are several tools available to assist with assessment:  
• CIS Configuration Assessment Tool (CIS-CAT® Pro Assessor) • CIS Benchmarks™ Certified 3rd Party Tooling  
These tools make the hardening process much more scalable for large numbers of systems and applications.  
NOTE: Some tooling focuses only on the Benchmark Recommendations that can  
be fully automated (skipping ones marked Manual). It is important that ALL Recommendations (Automated and Manual) be addressed since all are important for properly securing systems and are typically in scope for audits.  
Key Stakeholders  
Cybersecurity is a collaborative effort, and cross functional cooperation is imperative within an organization to discuss, test, and deploy Benchmarks in an effective and efficient way. The Benchmarks are developed to be best practice configuration guidelines applicable to a wide range of use cases. In some organizations, exceptions to specific Recommendations will be needed, and this team should work to prioritize the problematic Recommendations based on several factors like risk, time, cost, and labor. These exceptions should be properly categorized and documented for auditing purposes.  
Internal Only - General  
Apply the Correct Version of a Benchmark  
Benchmarks are developed and tested for a specific set of products and versions and applying an incorrect Benchmark to a system can cause the resulting pass/fail score to be incorrect. This is due to the assessment of settings that do not apply to the target systems. To assure the correct Benchmark is being assessed:  
• Deploy the Benchmark applicable to the way settings are managed in the  
environment: An example of this is the Microsoft Windows family of Benchmarks, which have separate Benchmarks for Group Policy, Intune, and Stand-alone systems based upon how system management is deployed. Applying the wrong Benchmark in this case will give invalid results.  
• Use the most recent version of a Benchmark: This is true for all Benchmarks, but especially true for cloud technologies. Cloud technologies change frequently and using an older version of a Benchmark may have invalid methods for auditing and remediation.  
Exceptions  
The guidance items in the Benchmarks are called recommendations and not requirements, and exceptions to some of them are expected and acceptable. The Benchmarks strive to be a secure baseline, or starting point, for a specific technology, with known issues identified during Benchmark development are documented in the Impact section of each Recommendation. In addition, organizational, system specific requirements, or local site policy may require changes as well, or an exception to a Recommendation or group of Recommendations (e.g. A Benchmark could Recommend that a Web server not be installed on the system, but if a system's primary purpose is to function as a Webserver, there should be a documented exception to this Recommendation for that specific server).  
In the end, exceptions to some Benchmark Recommendations are common and acceptable, and should be handled as follows:  
• The reasons for the exception should be reviewed cross-functionally and be well  
documented for audit purposes.  
• A plan should be developed for mitigating, or eliminating, the exception in the  
•  
future, if applicable. If the organization decides to accept the risk of this exception (not work toward mitigation or elimination), this should be documented for audit purposes.  
It is the responsibility of the organization to determine their overall security policy, and which settings are applicable to their unique needs based on the overall risk profile for the organization.  
Internal Only - General  
CIS has developed Build Kits for many technologies to assist in the automation of hardening systems. Build Kits are designed to correspond to Benchmark's “Remediation” section, which provides the manual remediation steps necessary to make that Recommendation compliant to the Benchmark.  
When remediating systems (changing configuration settings on deployed systems as per the Benchmark's Recommendations), please approach this with caution and test thoroughly.  
The following is a reasonable remediation approach to follow:  
• CIS Build Kits, or internally developed remediation methods should never be  
applied to production systems without proper testing.  
• Proper testing consists of the following:  
o Understand the configuration (including installed applications) of the targeted systems. Various parts of the organization may need different configurations (e.g., software developers vs standard office workers).  
o Read the Impact section of the given Recommendation to help determine if  
there might be an issue with the targeted systems.  
o Test the configuration changes with representative lab system(s). If issues  
arise during testing, they can be resolved prior to deploying to any production systems.  
o When testing is complete, initially deploy to a small sub-set of production systems and monitor closely for issues. If there are issues, they can be resolved prior to deploying more broadly.  
o When the initial deployment above is completes successfully, iteratively deploy to additional systems and monitor closely for issues. Repeat this process until the full deployment is complete.  
Summary  
Using the Benchmarks Certified tools, working as a team with key stakeholders, being selective with exceptions, and being careful with remediation deployment, it is possible to harden large numbers of deployed systems in a cost effective, efficient, and safe manner.  
NOTE: As previously stated, the PDF versions of the CIS Benchmarks™ are  
available for free, non-commercial use on the CIS Website. All other formats of the CIS Benchmarks™ (MS Word, Excel, and Build Kits) are available for CIS SecureSuite® members.  
CIS-CAT® Pro is also available to CIS SecureSuite® members.  
Internal Only - General  
Target Technology Details  
This document provides prescriptive guidance for establishing a secure configuration posture for OpenShift 4. The set of configuration files mentioned throughout this benchmark are specific to Red Hat’s CNCF certified Kubernetes distribution, Red Hat OpenShift Container Platform. Each section includes information about the default configuration of an OpenShift cluster and a set of recommendations for hardening the configuration, inspired by the CIS Kubernetes benchmark. For each hardening recommendation, information on how to implement the control and/or how to verify or audit the control is provided. In some cases, remediation information is also provided. The majority of the settings in the hardening guide are in place by default. The audit information for these settings is provided so that you can verify that the cluster admin has not made changes that would be less secure than the OpenShift defaults. A small number of items require configuration. Finally, there are some recommendations that require decisions by the customer, such as audit log size, retention and related settings. The recommendations that require decisions based on your needs are:  
• Configure encryption of data at rest in etcd datastore • Manage Image Provenance • Set the --event-qps argument as appropriate • Configure the API server audit log retention • Configure cluster logging to forward audit logs off the cluster • Configure the audit log file size • Adjust the garbage collection settings as needed • Create custom Security Context Constraints as needed • Configure Network Policies as appropriate •  
In OCP 4.6 and above, configure audit policies as appropriate  
Intended Audience  
This document is intended for system and application administrators, security specialists, auditors, help desk, and platform deployment personnel who plan to develop, deploy, assess, or secure solutions that incorporate OpenShift 4.  
Internal Only - General  
Consensus Guidance  
This CIS Benchmark™ was created using a consensus review process comprised of a global community of subject matter experts. The process combines real world experience with data-based information to create technology specific guidance to assist users to secure their environments. Consensus participants provide perspective from a diverse set of backgrounds including consulting, software development, audit and compliance, security research, operations, government, and legal.  
Each CIS Benchmark undergoes two phases of consensus review. The first phase occurs during initial Benchmark development. During this phase, subject matter experts convene to discuss, create, and test working drafts of the Benchmark. This discussion occurs until consensus has been reached on Benchmark recommendations. The second phase begins after the Benchmark has been published. During this phase, all feedback provided by the Internet community is reviewed by the consensus team for incorporation in the Benchmark. If you are interested in participating in the consensus process, please visit https://workbench.cisecurity.org/.  
Internal Only - General  
Typographical Conventions  
The following typographical conventions are used throughout this guide:  
Convention  
Meaning  
Stylized Monospace font  
Monospace font  
Used for blocks of code, command, and script examples. Text should be interpreted exactly as presented.  
Used for inline code, commands, UI/Menu selections or examples. Text should be interpreted exactly as presented.  
<Monospace font in brackets>  
Text set in angle brackets denote a variable requiring substitution for a real value.  
Italic font  
Bold font  
Used to reference other relevant settings, CIS Benchmarks and/or Benchmark Communities. Also, used to denote the title of a book, article, or other publication.  
Additional information or caveats things like Notes, Warnings, or Cautions (usually just the word itself and the rest of the text normal).  
Internal Only - General  
Recommendation Definitions  
The following defines the various components included in a CIS recommendation as applicable. If any of the components are not applicable it will be noted, or the component will not be included in the recommendation.  
Title  
Concise description for the recommendation's intended configuration.  
Assessment Status  
An assessment status is included for every recommendation. The assessment status indicates whether the given recommendation can be automated or requires manual steps to implement. Both statuses are equally important and are determined and supported as defined below:  
Automated  
Represents recommendations for which assessment of a technical control can be fully automated and validated to a pass/fail state. Recommendations will include the necessary information to implement automation.  
Manual  
Represents recommendations for which assessment of a technical control cannot be fully automated and requires all or some manual steps to validate that the configured state is set as expected. The expected state can vary depending on the environment.  
Profile  
A collection of recommendations for securing a technology or a supporting platform. Most benchmarks include at least a Level 1 and Level 2 Profile. Level 2 extends Level 1 recommendations and is not a standalone profile. The Profile Definitions section in the benchmark provides the definitions as they pertain to the recommendations included for the technology.  
Detailed information pertaining to the setting with which the recommendation is concerned. In some cases, the description will include the recommended value.  
Rationale Statement  
Detailed reasoning for the recommendation to provide the user a clear and concise understanding on the importance of the recommendation.  
Internal Only - General  
Impact Statement  
Any security, functionality, or operational consequences that can result from following the recommendation.  
Audit Procedure  
Systematic instructions for determining if the target system complies with the recommendation.  
Remediation Procedure  
Systematic instructions for applying recommendations to the target system to bring it into compliance according to the recommendation.

## 6.2 Activate audit logging

Ensure that local logging has been enabled on all systems and networking devices.  
● ● ●

## 6.3 Enable Detailed Logging

Enable system logging to include detailed information such as an event source, date, user, timestamp, source addresses, destination addresses, and other useful elements.  
● ●  
Internal Only - General  
MITRE ATT&CK Mappings:  
Techniques / Subtechniques  
Tactics  
Mitigations  
T1543  
TA0003, TA0004  
M1047  
Internal Only - General

## 6.4 Ensure adequate storage for logs

Ensure that all systems that store logs have adequate storage space for the logs generated.  
● ●  
v8  
v7  
MITRE ATT&CK Mappings:  
Techniques / Subtechniques  
Tactics  
Mitigations  
T1543  
TA0003, TA0004  
M1047  
Internal Only - General

## 6.8 Define and Maintain Role-Based Access Control

Define and maintain role-based access control, through determining and documenting the access rights necessary for each role within the enterprise to successfully carry out its assigned duties. Perform access control reviews of enterprise assets to validate that all privileges are authorized, on a recurring schedule at a minimum annually, or more frequently.  
●

## 7.3 Perform Automated Operating System Patch

Management Perform operating system updates on enterprise assets through automated patch management on a monthly, or more frequent, basis.  
● ● ●

## 8.1 Establish and Maintain an Audit Log Management

Process Establish and maintain an audit log management process that defines the enterprise’s logging requirements. At a minimum, address the collection, review, and retention of audit logs for enterprise assets. Review and update documentation annually, or when significant enterprise changes occur that could impact this Safeguard.  
● ● ●

## 8.2 Collect Audit Logs

Collect audit logs. Ensure that logging, per the enterprise’s audit log management process, has been enabled across enterprise assets.  
● ● ●

## 8.3 Enable Operating System Anti-Exploitation Features/

Deploy Anti-Exploit Technologies Enable anti-exploitation features such as Data Execution Prevention (DEP) or Address Space Layout Randomization (ASLR) that are available in an operating system or deploy appropriate toolkits that can be configured to apply protection to a broader set of applications and executables.  
● ●  
● ●  
MITRE ATT&CK Mappings:  
Techniques / Subtechniques  
T1609  
Tactics  
TA0002  
Mitigations  
M1028  
Internal Only - General

## 8.5 Collect Detailed Audit Logs

Configure detailed audit logging for enterprise assets containing sensitive data. Include event source, date, username, timestamp, source addresses, destination addresses, and other useful elements that could assist in a forensic investigation.  
● ●

## 8.8 Enable Command-line Audit Logging

Enable command-line audit logging for command shells, such as Microsoft Powershell and Bash.  
● ●  
● ●  
Internal Only - General

## 9.2 Ensure Only Approved Ports, Protocols and Services

Are Running Ensure that only network ports, protocols, and services listening on a system with validated business needs, are running on each system.  
● ●  
MITRE ATT&CK Mappings:  
Techniques / Subtechniques  
T1133  
Tactics  
TA0001  
Mitigations  
M1026  
Internal Only - General

## 9.3 Maintain and Enforce Network-Based URL Filters

Enforce and update network-based URL filters to limit an enterprise asset from connecting to potentially malicious or unapproved websites. Example implementations include category-based filtering, reputation-based filtering, or through the use of block lists. Enforce filters for all enterprise assets.

## 9.4 Apply Host-based Firewalls or Port Filtering

Apply host-based firewalls or port filtering tools on end systems, with a defaultdeny rule that drops all traffic except those services and ports that are explicitly allowed.  
● ● ●  
v8  
v7  
MITRE ATT&CK Mappings:  
Techniques / Subtechniques  
T1610  
Tactics  
TA0002  
Mitigations  
M1038, M1050  
Internal Only - General

## 9.5 Implement Application Firewalls

Place application firewalls in front of any critical servers to verify and validate the traffic going to the server. Any unauthorized traffic should be blocked and logged.  
●  
MITRE ATT&CK Mappings:  
Techniques / Subtechniques  
T1046  
Tactics  
TA0007  
Mitigations  
M1030, M1042  
Internal Only - General

## 10.5 Enable Anti-Exploitation Features

Enable anti-exploitation features on enterprise assets and software, where possible, such as Microsoft® Data Execution Prevention (DEP), Windows® Defender Exploit Guard (WDEG), or Apple® System Integrity Protection (SIP) and Gatekeeper™.  
● ●

## 12.6 Use of Secure Network Management and

Communication Protocols Use secure network management and communication protocols (e.g., 802.1X, Wi-Fi Protected Access 2 (WPA2) Enterprise or greater).

## 12.8 Establish and Maintain Dedicated Computing

Resources for All Administrative Work Establish and maintain dedicated computing resources, either physically or logically separated, for all administrative tasks or tasks requiring administrative access. The computing resources should be segmented from the enterprise's primary network and not be allowed internet access.  
●  
v8  
v8  
Internal Only - General  
Controls Version  
Control  
IG 1 IG 2 IG 3  
v7

## 12.9 Deploy Application Layer Filtering Proxy Server

Ensure that all network traffic to or from the Internet passes through an authenticated application layer proxy that is configured to filter unauthorized connections.  
●  
●  
MITRE ATT&CK Mappings:  
Techniques / Subtechniques  
Tactics  
Mitigations  
T1078, T1078.002  
TA0001, TA0004  
M1026  
Internal Only - General

## 13.4 Only Allow Access to Authorized Cloud Storage or

Email Providers Only allow access to authorized cloud storage or email providers.  
● ●  
● ●  
Internal Only - General  
MITRE ATT&CK Mappings:  
Techniques / Subtechniques  
T1552  
Tactics  
TA0006  
Mitigations  
M1026  
Internal Only - General

## 13.10 Perform Application Layer Filtering

Perform application layer filtering. Example implementations include a filtering proxy, application layer firewall, or gateway.

## 14.1 Segment the Network Based on Sensitivity

Segment the network based on the label or classification level of the information stored on the servers, locate all sensitive information on separated Virtual Local Area Networks (VLANs).  
● ●  
● ●  
MITRE ATT&CK Mappings:  
Techniques / Subtechniques  
T1098  
Tactics  
TA0003  
Mitigations  
M1030  
Internal Only - General

## 14.2 Enable Firewall Filtering Between VLANs

Enable firewall filtering between VLANs to ensure that only authorized systems are able to communicate with other systems necessary to fulfill their specific responsibilities.  
● ●  
MITRE ATT&CK Mappings:  
Techniques / Subtechniques  
T1046  
Tactics  
TA0007  
Mitigations  
M1030, M1042  
Internal Only - General

## 14.4 Encrypt All Sensitive Information in Transit

Encrypt all sensitive information in transit.  
● ●  
● ●  
MITRE ATT&CK Mappings:  
Techniques / Subtechniques  
Tactics  
Mitigations  
T1048, T1189  
TA0001, TA0010  
M1041  
Internal Only - General

## 14.6 Protect Information through Access Control Lists

Protect all information stored on systems with file system, network share, claims, application, or database specific access control lists. These controls will enforce the principle that only authorized individuals should have access to the information based on their need to access the information as a part of their responsibilities.  
● ● ●  
MITRE ATT&CK Mappings:  
Techniques / Subtechniques  
Tactics  
Mitigations  
T1083, T1222  
TA0005, TA0007  
M1022  
Internal Only - General

## 14.7 Enforce Access Control to Data through Automated

Tools Use an automated tool, such as host-based Data Loss Prevention, to enforce access controls to data even when data is copied off a system.  
●  
MITRE ATT&CK Mappings:  
Techniques / Subtechniques  
T1078  
Tactics  
TA0001  
Mitigations  
M1026  
Internal Only - General

## 14.8 Encrypt Sensitive Information at Rest

Encrypt all sensitive information at rest using a tool that requires a secondary authentication mechanism not integrated into the operating system, in order to access the information.  
v8  
v7  
● ●  
●  
Internal Only - General  
MITRE ATT&CK Mappings:  
Techniques / Subtechniques  
T1552  
Tactics  
TA0006  
Mitigations  
M1022  
Internal Only - General

## 14.9 Enforce Detail Logging for Access or Changes to

Sensitive Data Enforce detailed audit logging for access to sensitive data or changes to sensitive data (utilizing tools such as File Integrity Monitoring or Security Information and Event Monitoring).  
● ●  
●  
MITRE ATT&CK Mappings:  
Techniques / Subtechniques  
Tactics  
Mitigations  
T1543  
TA0003, TA0004  
M1026  
Internal Only - General

## 16.2 Configure Centralized Point of Authentication

Configure access for all accounts through as few centralized points of authentication as possible, including network, security, and cloud systems.  
● ●  
● ●  
MITRE ATT&CK Mappings:  
Techniques / Subtechniques  
Tactics  
Mitigations  
T1078, T1098  
TA0003, TA0006  
M1027, M1032  
Internal Only - General

## 16.9 Disable Dormant Accounts

Automatically disable dormant accounts after a set period of inactivity.  
● ● ●  
MITRE ATT&CK Mappings:  
Techniques / Subtechniques  
T1552  
Tactics  
TA0006  
Mitigations  
M1028  
Internal Only - General

## 16.11 Leverage Vetted Modules or Services for Application

Security Components Leverage vetted modules or services for application security components, such as identity management, encryption, and auditing and logging. Using platform features in critical security functions will reduce developers’ workload and minimize the likelihood of design or implementation errors. Modern operating systems provide effective mechanisms for identification, authentication, and authorization and make those mechanisms available to applications. Use only standardized, currently accepted, and extensively reviewed encryption algorithms. Operating systems also provide mechanisms to create and maintain secure audit logs.

# 600 or more restrictive (Manual)

Ensure that the kubeconfig file ownership is set to root:root (Manual)  
Ensure that the Scheduler kubeconfig file permissions are set to 600 or more restrictive (Manual)  
Set Correctly  
Yes No  
o  
o  
o  
o  
o  
o  
o  
o  
Ensure that the Scheduler kubeconfig file ownership is set to root:root (Manual)  
o  
o  
Ensure that the Controller Manager kubeconfig file permissions are set to 600 or more restrictive (Manual)  
o  
o  
Ensure that the Controller Manager kubeconfig file ownership is set to root:root (Manual)  
Ensure that the OpenShift PKI directory and file ownership is set to root:root (Manual)  
o  
o  
o  
o  
Ensure that the OpenShift PKI certificate file permissions are set to 600 or more restrictive (Manual)  
o  
o  
Ensure that the OpenShift PKI key file permissions are set to 600 (Manual)  
o  
o  
API Server  
Ensure that anonymous requests are authorized (Manual)  
Use https for kubelet connections (Manual)  
Ensure that the kubelet uses certificates to authenticate (Manual)  
Verify that the kubelet certificate authority is set as appropriate (Manual)  
o  
o  
o  
o  
o  
o  
o  
o  
Internal Only - General  
1.2.5  
1.2.6  
1.2.7  
1.2.8  
1.2.9  
1.2.10  
1.2.11  
1.2.12  
1.2.13  
1.2.14  
1.2.15  
1.2.16  
1.2.17  
1.2.18  
1.2.19  
CIS Benchmark Recommendation  
Set Correctly  
Yes No  
Ensure that the --authorization-mode argument is not set to AlwaysAllow (Manual)  
o  
o  
Verify that RBAC is enabled (Manual)  
Ensure that the APIPriorityAndFairness feature gate is enabled (Manual)  
o  
o  
o  
o  
Ensure that the admission control plugin AlwaysAdmit is not set (Manual)  
o  
o  
Ensure that the admission control plugin AlwaysPullImages is not set (Manual)  
o  
o  
Ensure that the admission control plugin ServiceAccount is set (Manual)  
o  
o  
Ensure that the admission control plugin NamespaceLifecycle is set (Manual)  
Ensure that the admission control plugin SecurityContextConstraint is set (Manual)  
o  
o  
o  
o  
Ensure that the admission control plugin NodeRestriction is set (Manual)  
o  
o  
Ensure that the --insecure-bind-address argument is not set (Manual)  
o  
o  
Ensure that the --insecure-port argument is set to 0 (Manual)  
Ensure that the --secure-port argument is not set to 0 (Manual)  
o  
o  
o  
o  
Ensure that the healthz endpoint is protected by RBAC (Manual)  
o  
o  
Ensure that the --audit-log-path argument is set (Manual) o  
Ensure that the audit logs are forwarded off the cluster for retention (Manual)  
o  
o  
o  
Internal Only - General  
CIS Benchmark Recommendation  
Ensure that the maximumRetainedFiles argument is set to 10 or as appropriate (Manual)  
Set Correctly  
Yes No  
o  
o  
Configure Kubernetes API Server Maximum Audit Log Size (Manual)  
o  
o  
Ensure that the --request-timeout argument is set (Manual)  
o  
o  
Ensure that the --service-account-lookup argument is set to true (Manual)  
o  
o  
Ensure that the --service-account-key-file argument is set as appropriate (Manual)  
Ensure that the --etcd-certfile and --etcd-keyfile arguments are set as appropriate (Manual)  
Ensure that the --tls-cert-file and --tls-private-key-file arguments are set as appropriate (Manual)  
Ensure that the --client-ca-file argument is set as appropriate (Manual)  
Ensure that the --etcd-cafile argument is set as appropriate (Manual)  
Ensure that encryption providers are appropriately configured (Manual)  
Ensure that the API Server only makes use of Strong Cryptographic Ciphers (Manual)  
o  
o  
o  
o  
o  
o  
o  
o  
o  
o  
o  
o  
o  
o  
Ensure unsupported configuration overrides are not used (Manual)  
o  
o  
Controller Manager  
Ensure that controller manager healthz endpoints are protected by RBAC (Manual)  
o  
o  
1.2.20  
1.2.21  
1.2.22  
1.2.23  
1.2.24  
1.2.25  
1.2.26  
1.2.27  
1.2.28  
1.2.29  
1.2.30  
1.2.31  
1.3  
1.3.1  
Internal Only - General  
CIS Benchmark Recommendation  
Ensure that the --use-service-account-credentials argument is set to true (Manual)  
Ensure that the --service-account-private-key-file argument is set as appropriate (Manual)  
Ensure that the --root-ca-file argument is set as appropriate (Manual)  
Scheduler  
Set Correctly  
Yes No  
o  
o  
o  
o  
o  
o  
Ensure that the healthz endpoints for the scheduler are protected by RBAC (Manual)  
o  
o  
Verify that the scheduler API service is protected by RBAC (Manual)  
o  
o  
etcd  
Ensure that the --cert-file and --key-file arguments are set as appropriate (Manual)  
o  
o  
Ensure that the --client-cert-auth argument is set to true (Manual)  
o  
o  
Ensure that the --auto-tls argument is not set to true (Manual)  
Ensure that the --peer-cert-file and --peer-key-file arguments are set as appropriate (Manual)  
o  
o  
o  
o  
Ensure that the --peer-client-cert-auth argument is set to true (Manual)  
o  
o  
Ensure that the --peer-auto-tls argument is not set to true (Manual)  
o  
o  
Ensure that a unique Certificate Authority is used for etcd (Manual)  
o  
o  
Control Plane Configuration  
1.3.2  
1.3.3  
1.3.4  
1.4  
1.4.1  
1.4.2  
2  
2.1  
2.2  
2.3  
2.4  
2.5  
2.6  
2.7  
3  
Internal Only - General  
CIS Benchmark Recommendation  
Set Correctly  
Yes No  
Authentication and Authorization  
Client certificate authentication should not be used for users (Manual)  
o  
o  
Logging  
Ensure that a minimal audit policy is created (Manual)  
Ensure that the audit policy covers key security concerns (Manual)  
o  
o  
o  
o  
Worker Nodes  
Worker Node Configuration Files  
Ensure that the kubelet service file permissions are set to 644 or more restrictive (Automated)  
o  
o  
Ensure that the kubelet service file ownership is set to root:root (Automated)  
o  
o  
If proxy kube proxy configuration file exists ensure permissions are set to 644 or more restrictive (Manual)  
o  
o  
If proxy kubeconfig file exists ensure ownership is set to root:root (Manual)  
o  
o  
Ensure that the --kubeconfig kubelet.conf file permissions are set to 644 or more restrictive (Automated)  
o  
o  
Ensure that the --kubeconfig kubelet.conf file ownership is set to root:root (Automated)  
o  
o  
Ensure that the certificate authorities file permissions are set to 644 or more restrictive (Automated)  
o  
o  
Ensure that the client certificate authorities file ownership is set to root:root (Automated)  
o  
o  
3.1  
3.1.1  
3.2  
3.2.1  
3.2.2  
4  
4.1  
4.1.1  
4.1.2  
4.1.3  
4.1.4  
4.1.5  
4.1.6  
4.1.7  
4.1.8  
Internal Only - General  
CIS Benchmark Recommendation  
Ensure that the kubelet --config configuration file has permissions set to 600 or more restrictive (Automated)  
Set Correctly  
Yes No  
o  
o  
Ensure that the kubelet configuration file ownership is set to root:root (Automated)  
o  
o  
Kubelet  
Activate Garbage collection in OpenShift Container Platform 4, as appropriate (Manual)  
Ensure that the --anonymous-auth argument is set to false (Automated)  
o  
o  
o  
o  
Ensure that the --authorization-mode argument is not set to AlwaysAllow (Automated)  
o  
o  
Ensure that the --client-ca-file argument is set as appropriate (Automated)  
Verify that the read only port is not used or is set to 0 (Automated)  
Ensure that the --streaming-connection-idle-timeout argument is not set to 0 (Automated)  
o  
o  
o  
o  
o  
o  
Ensure that the --make-iptables-util-chains argument is set to true (Manual)  
o  
o  
Ensure that the kubeAPIQPS [--event-qps] argument is set to 0 or a level which ensures appropriate event capture (Manual)  
o  
o  
Ensure that the --tls-cert-file and --tls-private-key-file arguments are set as appropriate (Manual)  
o  
o  
Ensure that the --rotate-certificates argument is not set to false (Manual)  
o  
o  
Verify that the RotateKubeletServerCertificate argument is set to true (Manual)  
o  
o  
4.1.9  
4.1.10  
4.2  
4.2.1  
4.2.2  
4.2.3  
4.2.4  
4.2.5  
4.2.6  
4.2.7  
4.2.8  
4.2.9  
4.2.10  
4.2.11  
Internal Only - General  
CIS Benchmark Recommendation  
4.2.12  
Ensure that the Kubelet only makes use of Strong Cryptographic Ciphers (Manual)  
Set Correctly  
Yes No  
o  
o  
5  
5.1  
5.1.1  
5.1.2  
5.1.3  
5.1.4  
5.1.5  
5.1.6  
5.2  
5.2.1  
5.2.2  
5.2.3  
5.2.4  
5.2.5  
Policies  
RBAC and Service Accounts  
Ensure that the cluster-admin role is only used where required (Manual)  
o  
o  
Minimize access to secrets (Manual)  
Minimize wildcard use in Roles and ClusterRoles (Manual)  
Minimize access to create pods (Manual)  
Ensure that default service accounts are not actively used. (Manual)  
o  
o  
o  
o  
o  
o  
o  
o  
Ensure that Service Account Tokens are only mounted where necessary (Manual)  
o  
o  
Security Context Constraints  
Minimize the admission of privileged containers (Manual) o  
Minimize the admission of containers wishing to share the host process ID namespace (Manual)  
o  
o  
o  
Minimize the admission of containers wishing to share the host IPC namespace (Manual)  
Minimize the admission of containers wishing to share the host network namespace (Manual)  
Minimize the admission of containers with allowPrivilegeEscalation (Manual)  
o  
o  
o  
o  
o  
o  
5.2.6  
Minimize the admission of root containers (Manual)  
o  
o  
Internal Only - General  
CIS Benchmark Recommendation  
Minimize the admission of containers with the NET\_RAW capability (Manual)  
Minimize the admission of containers with added capabilities (Manual)  
Minimize the admission of containers with capabilities assigned (Manual)  
Minimize access to privileged Security Context Constraints (Manual)  
Network Policies and CNI  
Ensure that the CNI in use supports Network Policies (Manual)  
Ensure that all Namespaces have Network Policies defined (Manual)  
Secrets Management  
Set Correctly  
Yes No  
o  
o  
o  
o  
o  
o  
o  
o  
o  
o  
o  
o  
Prefer using secrets as files over secrets as environment variables (Manual)  
o  
o  
Consider external secret storage (Manual)  
o  
o  
Extensible Admission Control  
Configure Image Provenance using image controller configuration parameters (Manual)  
o  
o  
General Policies  
Create administrative boundaries between resources using namespaces (Manual)  
o  
o  
Ensure that the seccomp profile is set to docker/default in your pod definitions (Manual)  
o  
o  
Apply Security Context to Your Pods and Containers (Manual)  
o  
o  
5.2.7  
5.2.8  
5.2.9  
5.2.10  
5.3  
5.3.1  
5.3.2  
5.4  
5.4.1  
5.4.2  
5.5  
5.5.1  
5.7  
5.7.1  
5.7.2  
5.7.3  
Internal Only - General  
CIS Benchmark Recommendation  
Set Correctly  
Yes No  
5.7.4  
The default namespace should not be used (Manual)  
o  
o  
Internal Only - General  
Appendix: CIS Controls v7 IG 1 Mapped Recommendations  
1.1.1  
1.1.2  
1.1.3  
1.1.4  
1.1.5  
1.1.6  
1.1.7  
1.1.8  
1.1.9  
1.1.10  
1.1.11  
1.1.12  
1.1.13  
1.1.14  
1.1.15  
Recommendation  
Ensure that the API server pod specification file permissions are set to 600 or more restrictive  
Ensure that the API server pod specification file ownership is set to root:root  
Ensure that the controller manager pod specification file permissions are set to 600 or more restrictive  
Ensure that the controller manager pod specification file ownership is set to root:root  
Ensure that the scheduler pod specification file permissions are set to 600 or more restrictive  
Ensure that the scheduler pod specification file ownership is set to root:root  
Ensure that the etcd pod specification file permissions are set to 600 or more restrictive  
Ensure that the etcd pod specification file ownership is set to root:root  
Ensure that the Container Network Interface file permissions are set to 600 or more restrictive  
Ensure that the Container Network Interface file ownership is set to root:root  
Ensure that the etcd data directory permissions are set to

# Unplaced Controls

|  |  |  |  |
| --- | --- | --- | --- |
| Control Reference ID | Control Name & Description | Description | Remediation |
| 10 | or as appropriate (Manual) | Retain 10 or an appropriate number of old log files. | None. |
| 700 | or more restrictive (Manual) | Ensure that the etcd data directory has permissions of 700 or more restrictive. | No remediation required. File permissions are managed by the etcd operator. |