CIS Talos Linux Benchmark

v1.0.0 - 07-22-2024

Terms of Use

Please see the below link for our current terms of use:

https://www.cisecurity.org/cis-securesuite/cis-securesuite-membership-terms-of-use/

Table of Contents

Overview

All CIS 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 for vulnerabilities and quickly updating with the latest security patches.
- Monitoring applications and libraries for vulnerabilities and quickly updating with the latest security patches.

In the end, the CIS Benchmarks are designed as a key **component** of a comprehensive cybersecurity program.

This document provides prescriptive guidance for establishing a secure configuration posture for Linux systems based on Talos Linux.
Talos Linux does not include any facilities for interactive use. This means that shells, user accounts, a console for local access, and an SSH daemon for remote access are not available.
To obtain the latest version of this guide, please visit http://workbench.cisecurity.org. If you have questions, comments, or have identified ways to improve this guide, please write us at feedback@cisecurity.org.

Intended Audience

This benchmark 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 Talos Linux.

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/.

Typographical Conventions

The following typographical conventions are used throughout this guide:

Convention	Meaning
`Stylized Monospace font`	Used for blocks of code, command, and script examples. Text should be interpreted exactly as presented.
`Monospace font`	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	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.
Bold font	Additional information or caveats things like **Notes**, **Warnings**, or **Cautions** (usually just the word itself and the rest of the text normal).

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.

Description

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.

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.

Default Value

Default value for the given setting in this recommendation, if known. If not known, either not configured or not defined will be applied.

References

Additional documentation relative to the recommendation.

CIS Critical Security Controls® (CIS Controls®)

The mapping between a recommendation and the CIS Controls is organized by CIS Controls version, Safeguard, and Implementation Group (IG). The Benchmark in its entirety addresses the CIS Controls safeguards of (v7) “5.1 - Establish Secure Configurations” and (v8) '4.1 - Establish and Maintain a Secure Configuration Process” so individual recommendations will not be mapped to these safeguards.

Additional Information

Supplementary information that does not correspond to any other field but may be useful to the user.

Profile Definitions

The following configuration profiles are defined by this Benchmark:

- **Level 1 - Server**

  Items in this profile intend to:

  - be practical and prudent;
  - provide a clear security benefit; and
  - not inhibit the utility of the technology beyond acceptable means.

  This profile is intended for servers.

- **Level 2 - Server**

  This profile extends the "Level 1 - Server" profile. Items in this profile exhibit one or more of the following characteristics:

  - are intended for environments or use cases where security is paramount.
  - acts as defense in depth measure.
  - may negatively inhibit the utility or performance of the technology.

  This profile is intended for servers.

Acknowledgements

This Benchmark exemplifies the great things a community of users, vendors, and subject matter experts can accomplish through consensus collaboration. The CIS community thanks the entire consensus team with special recognition to the following individuals who contributed greatly to the creation of this guide:

This benchmark is based upon previous Linux benchmarks published and would not be possible without the contributions provided over the history of all of these benchmarks. The CIS community thanks everyone who has contributed to the Linux benchmarks.

Author
Andrew Rynhard, Sidero Labs
Steve Francis, Sidero Labs
Andrey Smirnov, Sidero Labs

Contributor
Phil White, Center for Internet Security
Justin Brown, Center for Internet Security

Recommendations

1 Initial Setup

Items in this section are advised for all systems. If the defaults of Talos Linux are used, there is very little that a user needs to do to meet the recommendations of this benchmark, although there are additional settings that can be deployed to marginally increase the robustness of security.

1.1 Boot Settings

The recommendations in this section focus on securing the bootloader and settings involved in the boot process directly. Talos supports booting UEFI systems in SecureBoot mode. When combined with TPM-based disk encryption, this provides Trusted Boot, which extends protection to *initrd*, and a fully signed execution path from firmware to userspace.

1.1.1 Install Talos with SecureBoot enabled (Automated)

Profile Applicability:

- Level 2 - Server

Description:

Secure Boot is a standard that ensures systems boot only to a trusted operating system.

Talos supports booting UEFI systems in SecureBoot mode. When combined with TPM-based disk encryption, this provides Trusted Boot experience.

Note: SecureBoot is not supported on x86 platforms in BIOS mode.

The implementation uses systemd-boot as a boot menu implementation, while the Talos kernel, initramfs and cmdline arguments are combined into the Unified Kernel Image (UKI) format. UEFI firmware loads the systemd-boot bootloader, which then loads the UKI image. Both systemd-boot and Talos UKI image are signed with the key, which is enrolled into the UEFI firmware.

As Talos Linux is fully contained in the UKI image, the full operating system is verified and booted by the UEFI firmware.

Rationale:

Enabling SecureBoot ensures that only signed and validated Talos images are able to be loaded and, in conjunction with a TPM2 module, decrypt the disk.

Audit:

```
$ talosctl -n <IP> get securitystate

NODE   NAMESPACE   TYPE            ID              VERSION   SECUREBOOT
       runtime     SecurityState   securitystate   1         true
```

Confirm that SECUREBOOT shows as *true*.
Note that the *--insecure* flag will be necessary if the system is not yet part of a Kubernetes cluster.

Remediation:

Ensure Talos Linux is installed and booted from images that support SecureBoot, as per the documentation:
https://www.talos.dev/latest/talos-guides/install/bare-metal-platforms/secureboot
Note that secureboot images signed by SideroLabs are available for all platforms, or enterprises may sign their own images. Additional steps are usually required with cloud provider platforms to register the Sidero signing keys.

Default Value:

The default depends on whether a secureboot image was used for initial installation.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	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™.		●	●
v7	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.		●	●

1.2 Encrypt System disks

It is possible to enable encryption for system disks at the OS level. Such encryption can add defense-in-depth, and help protect data from actors with physical access to the machine or drives.

1.2.1 Encrypt System Disks with KMS or TPM methods (Automated)

Profile Applicability:

- Level 1 - Server

Description:

It is possible to enable encryption for the STATE and EPHEMERAL partitions of the system disks at the OS level. The STATE partition contains the most sensitive node data: secrets and cluster certificates. The EPHEMERAL partition may contain sensitive workload data. Data is encrypted using LUKS2, which is provided by the Linux kernel modules and cryptsetup utility.

Talos Linux supports four encryption methods, which can be combined together for a single partition:

- static - encrypt with the static passphrase (weakest protection, for STATE partition encryption it means that the passphrase will be stored in the META partition, which is not encrypted, and thus can be trivially bypassed).
- nodeID - encrypt with the key derived from the node UUID (this protects against data being leaked or recovered from a drive that has been physically removed from a Talos Linux node, but does not protect the data from access in the same system.).
- kms - encrypt using a key sealed with a network KMS (strong, but requires network access to decrypt the data.)
- tpm - encrypt with the key derived from the hardware TPM (strong, when used with SecureBoot).

It is recommended to use KMS or TPM methods if available.

Rationale:

Encrypting the partitions can prevent an attacker with physical access to the machine, or the ability to boot an alternate operating system remotely, from being able to retrieve data such as cluster certificates and other information that can be used to compromise a cluster.

Impact:

If using a network KMS, then a node will be unable to boot if the KMS system is unreachable at the time of boot.

If a system is configured to use TPM encryption without suitable TPM2.0 supporting hardware, it will fail during the install process with the message "failed to call key handler at slot 0: stat /dev/tpm0: no such file or directory", and then revert back to maintenance mode.

Audit:

```
talosctl read /system/state/config.yaml | yq '.machine.systemDiskEncryption'
```

The *keys* entries should have values of *tpm* or *kms* in order to pass.
For example:

```
systemDiskEncryption:
    ephemeral:
      provider: luks2
      keys:
        - slot: 0
          tpm: {}
    state:
      provider: luks2
      keys:
        - slot: 0
          tpm: {}
```

Remediation:

Because there is no in-place encryption of the STATE partition, it is recommended to set up encryption during the initial install of a Talos Linux machine.

Using Omni

The simplest way to enable disk encryption is if Omni is in use to manage the Talos Linux machines. In this case, simply specify the check-box "Encrypt Disks", under Cluster Features, when creating the cluster in the Omni UI. All nodes that join the cluster will have encryption enabled, using Omni as the KMS. This can also be achieved using cluster templates, by setting:

```
features:
  diskencryption: true
```

via Machine Configurations

If using SecureBoot, apply the following patch to enable TPM based disk encryption:

```
# tpm-disk-encryption.yaml
machine:
  systemDiskEncryption:
    ephemeral:
      provider: luks2
      keys:
        - slot: 0
          tpm: {}
    state:
      provider: luks2
      keys:
        - slot: 0
          tpm: {}
```

Apply the following to enable a network KMS:

```
systemDiskEncryption:
        state:
            provider: luks2
            keys:
                - kms:
                    endpoint: https://192.168.88.21:4443
                  slot: 0
        ephemeral:
            provider: luks2
            keys:
                - kms:
                    endpoint: https://192.168.88.21:4443
                  slot: 0
```

Default Value:

Talos Linux will by default enable encryption of confidential data at rest stored in etcd, but does not enable encryption of data on the disk partitions by default.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	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.		●	●
v7	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.			●

1.3 Ensure updated software is installed (Manual)

Profile Applicability:

- Level 1 - Server

Description:

Periodically, software updates are released for included software either due to security flaws or to include additional functionality or bug fixes.

Rationale:

Talos Linux is regularly updated to incorporate the latest Linux Kernel, Kubernetes versions, Talos bug fixes and other improvements. It is recommended to always run and upgrade to the latest stable release of Talos Linux. Users can be notified of the availability of new Talos Linux releases by subscribing to release-notifications on the Talos Linux GitHub repository. Similarly, it is recommended to run and upgrade to the latest stable Kubernetes release on all clusters.

As with any software installation, organizations need to determine if a given update meets their requirements and verify the compatibility and supportability of any additional software against the update revision that is selected.

Audit:

Verify there are no updates to install by comparing the output of:

```
talosctl version
```

With the most recent available stable release in the Talos Linux github repository.

Remediation:

Update Talos Linux as per https://www.talos.dev/latest/talos-guides/upgrading-talos/

Additional Information:

Site policy may mandate a testing period before install onto production systems for available updates.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	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.	●	●	●
v8	7.4 Perform Automated Application Patch Management Perform application updates on enterprise assets through automated patch management on a monthly, or more frequent, basis.	●	●	●
v7	3.4 Deploy Automated Operating System Patch Management Tools Deploy automated software update tools in order to ensure that the operating systems are running the most recent security updates provided by the software vendor.	●	●	●
v7	3.5 Deploy Automated Software Patch Management Tools Deploy automated software update tools in order to ensure that third-party software on all systems is running the most recent security updates provided by the software vendor.	●	●	●

2 Time Synchronization

It is recommended that physical systems and virtual guests lacking direct access to the physical host's clock be configured to synchronize their time using NTP. Talos Linux has integrated support for NTP in the machined daemon, that is part of every Talos Linux install. Thus the recommendation is to ensure NTP is configured and operating correctly.

2.1 Ensure NTP is configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

The Talos Linux daemon `machined` implements the Network Time Protocol (NTP). NTP is designed to synchronize system clocks across a variety of systems and use a source that is highly accurate.

Rationale:

Proper configuration of NTP is vital to ensuring time synchronization is working properly. By default, Talos Linux installs with a single time server referenced, which is currently `time.cloudflare.com`.

Audit:

Run the following command and verify that at least two time servers are configured:

```
# talosctl get TimeServer -o yaml

node: 192.168.64.118
metadata:
    namespace: network
    type: TimeServerStatuses.net.talos.dev
    id: timeservers
    version: 3
    owner: network.TimeServerSpecController
    phase: running
    created: 2024-06-04T04:05:47Z
    updated: 2024-06-05T17:50:14Z
spec:
    timeServers:
        - 2.pool.ntp.org
        - time.cloudflare.com
```

The synchronization status of the system can be checked with:

```
#talosctl get timestatus -o yaml
node: 192.168.64.118
metadata:
    namespace: runtime
    type: TimeStatuses.v1alpha1.talos.dev
    id: node
    version: 20
    owner: time.SyncController
    phase: running
    created: 2024-06-04T04:05:47Z
    updated: 2024-06-05T14:45:14Z
spec:
    synced: true
    epoch: 18
    syncDisabled: false
```

Remediation:

Configure additional time servers as needed.

The following commands would add "2.pool.ntp.org" to the list of time servers.
Note that if the system is using the default time server, then defining a time server in the machine configuration will replace the default. (i.e. the operation below would replace the default time server, not append it to the list of time servers, if there is currently no time server specified in the machine configuration.)

Execute the following command to create a patch to update the time servers:

```
# cat <<EOF > time-servers.yaml
machine:
  time:
    servers:
      - 2.pool.ntp.org
EOF
```

Then apply the patch to the machine to activate the new configuration:

```
talosctl patch mc --patch @time-servers.yaml
```

Default Value:

If not configured with a time server, Talos Linux will use a single default time server. The default is `pool.ntp.org` in Talos versions lower than 1.7, and `time.cloudflare.com` in version 1.7 and later.

Note that the default time server is not visible in the machine configuration file, but can be inspected with the command:

```
talosctl  get TimeServer -o yaml
node: 192.168.64.118
metadata:
    namespace: network
    type: TimeServerStatuses.net.talos.dev
    id: timeservers
    version: 3
    owner: network.TimeServerSpecController
    phase: running
    created: 2024-06-04T04:05:47Z
    updated: 2024-06-05T17:50:14Z
spec:
    timeServers:
        - time.cloudflare.com
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	8.4 Standardize Time Synchronization Standardize time synchronization. Configure at least two synchronized time sources across enterprise assets, where supported.		●	●
v7	6.1 Utilize Three Synchronized Time Sources Use at least three synchronized time sources from which all servers and network devices retrieve time information on a regular basis so that timestamps in logs are consistent.		●	●

3 Kubernetes

Talos Linux will install default vanilla Kubernetes, with many of the CIS Kubernetes benchmark recommendations applied by default. It is recommended that the CIS Kubernetes benchmark application be validated for any sensitive installation. Specific further controls implemented or recommended by Talos Linux are below:

- Control plane scheduling: restrict workloads on control plane nodes.
- Pod Security Standards: set the Restricted policy, which is aimed at enforcing current Pod hardening best practices.

3.1 Ensure control plane scheduling is disabled (Automated)

Profile Applicability:

- Level 1 - Server

Description:

Control plane scheduling allows workloads to run on control plane nodes.

Rationale:

Restricting access to control plane nodes is vital to ensuring that sensitive data (e.g. encryption secrets, PKI, etc.) is not exposed. Allowing workload scheduling on control plane nodes may enable workloads to access sensitive data present on control plane nodes, and may also adversely impact control plane node stability (if workloads consume excess CPU or Memory resources.)

Impact:

Preventing workloads from running on control plane nodes excludes control plane compute resources from the set available for workloads.

Audit:

Run the following command against a control plane node and verify that control plane scheduling is disabled:

```
# talosctl read /system/state/config.yaml | yq
'.cluster.allowSchedulingOnControlPlanes | select(. != null)'

false
```

Remediation:

The following commands would disable control plane scheduling.
Create a patch to disallow scheduling on control planes:

```
cat <<EOF > disallowSchedulingPatch.yaml
cluster:
    allowSchedulingOnControlPlanes: false
EOF
```

Then apply the patch to the machine:

```
talosctl patch mc --patch @disallowSchedulingPatch.yaml
```

Default Value:

The default value for `allowSchedulingOnControlPlanes` is `false`. The default is not shown in the machine config.

References:

1. null

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	0.0 Explicitly Not Mapped Explicitly Not Mapped			
v7	0.0 Explicitly Not Mapped Explicitly Not Mapped			

3.2 Ensure Pod Security Standards policy is restricted (Automated)

Profile Applicability:

- Level 1 - Server

Description:

Kubernetes Pod Security Standards define different isolation levels for Pods. These standards let you define how you want to restrict the behavior of pods in a clear, consistent fashion.

Rationale:

Restricting workloads is vital to ensure that workloads adhere to the principle of least privilege. The Restricted policy is aimed at enforcing current Pod hardening best practices, at the expense of some compatibility. It is targeted at operators and developers of security-critical applications, as well as lower-trust users.

Audit:

Run the following command and verify that workloads are restricted:

```
# talosctl get admissioncontrolconfigs.kubernetes.talos.dev admission-control
-o yaml | yq '.spec.config[0].configuration.defaults.enforce'
restricted
```

Remediation:

The following commands would enable the `restricted` policy for workloads:
Create a patch that defines the policy:

```
cat <<EOF > admissionPatch.yaml
cluster:
  apiServer:
    admissionControl:
      - name: PodSecurity
        configuration:
          defaults:
            enforce: restricted
EOF
```

Then apply the patch to the machines:

```
talosctl patch mc --patch @admissionPatch.yaml
```

Default Value:

baseline

The Baseline policy is aimed at ease of adoption for common containerized workloads while preventing known privilege escalations. This policy is targeted at application operators and developers of non-critical applications.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	0.0 Explicitly Not Mapped Explicitly Not Mapped			
v7	0.0 Explicitly Not Mapped Explicitly Not Mapped			

4 Access Control

Role Based Access in Talos

Talos uses certificates to authorize users. The certificate subject’s organization field is used to encode user roles. There is a set of predefined roles that allow access to different API methods:

- *os:admin* grants access to all methods;
- *os:operator* grants everything os:reader role does, plus additional methods: rebooting, shutting down, etcd backup, etcd alarm management, and so on;
- *os:reader* grants access to “safe” methods (for example, that include the ability to list files, but do not include the ability to read file contents);
- *os:etcd:backup* grants access to /machine.MachineService/EtcdSnapshot method. Role Based Access Control is enabled for the Talos Linux API by default.

It is recommended that the least privileged access role that supports an administrator's required use be used.

4.1 Ensure RBAC is enabled (Automated)

Profile Applicability:

- Level 1 - Server

Description:

RBAC allows granular control over permissions for users.

Rationale:

RBAC is vital to ensure that the principle of least privilege is adhered to.

Audit:

Run the following command and verify that RBAC is enabled:

```
# talosctl read /system/state/config.yaml | yq '.machine.features.rbac |
select(. != null) '
true
```

Remediation:

Generate a new client configuration access file with the role of os:admin.
(Additional configurations and certificates for different roles can be generated by passing --roles flag.)

```
talosctl config new --roles=os:admin admin
```

The above command will create a new client configuration file *admin* with a new certificate with os:admin role.

The following commands would then enable RBAC:
Create a patch:

```
cat <<EOF > rbacPatch.yaml
machine:
  features:
    rbac: true
EOF
```

Then apply the patch to the machines:

```
talosctl patch mc --patch @rbacPatch.yaml
```

Enabling RBAC will reboot the node

Default Value:

RBAC is enabled by default for all clusters created with Talos Linux of version 0.11 or later, but was disabled in clusters created with prior versions.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	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.			●

5 Network Configuration

This section provides guidance on for securing the network configuration of the system through kernel parameters, access list control, and firewall settings.

5.1 Network Parameters (Host Only)

The following network parameters are intended for use if the system is to act as a host only. A system is considered host only if the system has a single interface, or has multiple interfaces but will not be configured as a router.

Because Kubernetes nodes generally have to act as routers in order to forward packets to the Pod subnets, it is unlikely that a Talos linux node will be configured as a Host Only.

5.1.1 Ensure packet redirect sending is disabled (Automated)

Profile Applicability:

- Level 2 - Server

Description:

ICMP Redirects are used to send routing information to other hosts. As a host itself does not act as a router (in a host only configuration), there is no need to send redirects.

Rationale:

An attacker could use a compromised host to send invalid ICMP redirects to other router devices in an attempt to corrupt routing and have users access a system set up by the attacker as opposed to a valid system.

Audit:

Run the following commands and verify output matches:

```
# talosctl read /proc/sys/net/ipv4/conf/all/send_redirects
0
# talosctl read /proc/sys/net/ipv4/conf/default/send_redirects
0
```

Remediation:

Run the following commands to set the active kernel parameters and persist the settings:
Create the patch to disable sending redirects:

```
# cat <<EOF > send-redirects.yaml
machine:
  sysctls:
    net.ipv4.conf.all.send_redirects: "0"
    net.ipv4.conf.default.send_redirects: "0"
EOF
```

Apply the patch to the machines:

```
talosctl patch mc --patch @send-redirects.yaml
```

Default Value:

Sending redirects is enabled by default.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	0.0 Explicitly Not Mapped Explicitly Not Mapped			
v7	0.0 Explicitly Not Mapped Explicitly Not Mapped			

5.2 Network Parameters (Host and Router)

The following network parameters are intended for use on both host only and router systems. A system acts as a router if it has at least two interfaces and is configured to perform routing functions.

5.2.1 Ensure source routed packets are not accepted (Automated)

Profile Applicability:

- Level 1 - Server

Description:

In networking, source routing allows a sender to partially or fully specify the route packets take through a network. In contrast, non-source routed packets travel a path determined by routers in the network. In some cases, systems may not be routable or reachable from some locations (e.g. private addresses vs. Internet routable), and so source routed packets would need to be used.

Rationale:

Setting `net.ipv4.conf.all.accept_source_route`, `net.ipv4.conf.default.accept_source_route`, `net.ipv6.conf.all.accept_source_route` and `net.ipv6.conf.default.accept_source_route` to 0 disables the system from accepting source routed packets. Assume this system was capable of routing packets to Internet routable addresses on one interface and private addresses on another interface. Assume that the private addresses were not routable to the Internet routable addresses and vice versa. Under normal routing circumstances, an attacker from the Internet routable addresses could not use the system as a way to reach the private address systems. If, however, source routed packets were allowed, they could be used to gain access to the private address systems as the route could be specified, rather than rely on routing protocols that did not allow this routing.

Audit:

Run the following commands and verify output matches:

```
talosctl read /proc/sys/net/ipv4/conf/all/accept_source_route
0
talosctl read  /proc/sys/net/ipv4/conf/default/accept_source_route
0
talosctl read /proc/sys/net/ipv6/conf/default/accept_source_route
0
talosctl read  /proc/sys/net/ipv6/conf/all/accept_source_route
0
```

Remediation:

Run the following commands to set the active kernel parameters and persist the settings:
Create the patch file:

```
# cat <<EOF > reject-sourceroutes.yaml
machine:
  sysctls:
    net.ipv4.conf.all.accept_source_route: "0"
    net.ipv4.conf.default.accept_source_route: "0"
    net.ipv6.conf.all.accept_source_route: "0"
    net.ipv6.conf.default.accept_source_route: "0"
EOF
```

Apply the patch file to the machines:

```
talosctl patch mc --patch @reject-sourceroutes.yaml
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	0.0 Explicitly Not Mapped Explicitly Not Mapped			
v7	0.0 Explicitly Not Mapped Explicitly Not Mapped			

5.2.2 Ensure ICMP redirects are not accepted (Automated)

Profile Applicability:

- Level 2 - Server

Description:

ICMP redirect messages are packets that convey routing information and tell your host (acting as a router) to send packets via an alternate path. It is a way of allowing an outside routing device to update your system routing tables. By setting `accept_redirects` to 0, the system will not accept any ICMP redirect messages, and therefore, won't allow outsiders to update the system's routing tables.

Rationale:

Attackers could use bogus ICMP redirect messages to maliciously alter the system routing tables and get them to send packets to incorrect networks and allow your system packets to be captured.

Audit:

Run the following commands and verify output matches:

```
# talosctl read /proc/sys/net/ipv4/conf/all/accept_redirects
0
# talosctl read /proc/sys/net/ipv4/conf/default/accept_redirects
0
# talosctl read /proc/sys/net/ipv6/conf/all/accept_redirects
0
# talosctl read /proc/sys/net/ipv6/conf/default/accept_redirects
0
```

Remediation:

Run the following commands to set the active kernel parameters and persist the settings:
Create the patch with the sysctls to disable accepting redirects on both ipv4 and ipv6, for new and existing interfaces:

```
# cat <<EOF > accept-redirects.yaml
machine:
  sysctls:
    net.ipv4.conf.all.accept_redirects: "0"
    net.ipv4.conf.default.accept_redirects: "0"
    net.ipv6.conf.all.accept_redirects: "0"
    net.ipv6.conf.default.accept_redirects: "0"
EOF
```

Then apply the patch to the machines:

```
talosctl patch mc --patch @accept-redirects.yaml
```

Default Value:

```
/proc/sys/net/ipv4/conf/all/accept_redirects
0
/proc/sys/net/ipv4/conf/default/accept_redirects
1
/proc/sys/net/ipv6/conf/all/accept_redirects
1
/proc/sys/net/ipv6/conf/default/accept_redirects
1
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	0.0 Explicitly Not Mapped Explicitly Not Mapped			
v7	0.0 Explicitly Not Mapped Explicitly Not Mapped			

5.2.3 Ensure secure ICMP redirects are not accepted (Automated)

Profile Applicability:

- Level 2 - Server

Description:

Secure ICMP redirects are the same as ICMP redirects, except they come from gateways listed on the default gateway list. It is assumed that these gateways are known to your system, and that they are likely to be secure.

Rationale:

It is still possible for even known gateways to be compromised. Setting `net.ipv4.conf.all.secure_redirects` to 0 protects the system from routing table updates by possibly compromised known gateways.

Audit:

Run the following commands and verify output matches:

```
# talosctl read /proc/sys/net/ipv4/conf/all/secure_redirects
0
# talosctl read /proc/sys/net/ipv4/conf/default/secure_redirects
0
```

Remediation:

Run the following commands to set the active kernel parameters and persist the settings:
Create the patch file to disable redirects, for both new and existing interfaces:

```
# cat <<EOF > secure-redirects.yaml
machine:
  sysctls:
    net.ipv4.conf.all.secure_redirects: "0"
    net.ipv4.conf.default.secure_redirects: "0"
EOF
```

Then apply the patch to the machine, to effect both changes:

```
talosctl patch mc --patch @secure-redirects.yaml
```

Default Value:

```
/proc/sys/net/ipv4/conf/all/secure_redirects
1
/proc/sys/net/ipv4/conf/default/secure_redirects
1
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	0.0 Explicitly Not Mapped Explicitly Not Mapped			
v7	0.0 Explicitly Not Mapped Explicitly Not Mapped			

5.2.4 Ensure suspicious (martian) packets are logged (Automated)

Profile Applicability:

- Level 2 - Server

Description:

When enabled, this feature logs packets with un-routable source addresses (so called "martian packets") to the kernel log.

Rationale:

Enabling this feature and logging these packets allows an administrator to investigate the possibility that an attacker is sending spoofed packets to their system.

Impact:

Note that extra message logging can cause log buffers to fill and rotate more quickly, possibly making it harder to track events locally.

Audit:

Run the following commands and verify output matches:

```
# talosctl read /proc/sys/net/ipv4/conf/all/log_martians
1
# talosctl read /proc/sys/net/ipv4/conf/default/log_martians
1
```

Remediation:

Run the following commands to set the active kernel parameters and persist the settings:
Create the patch to define the changes in the machine config to log martian packets:

```
cat <<EOF > log-martians.yaml
machine:
  sysctls:
    net.ipv4.conf.all.log_martians: "1"
    net.ipv4.conf.default.log_martians: "1"
EOF
```

Apply the patch to the machines:

```
talosctl patch mc --patch @log-martians.yaml
```

Default Value:

```
/proc/sys/net/ipv4/conf/all/log_martians
0
/proc/sys/net/ipv4/conf/default/log_martians
0
```

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	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.	●	●	●
v8	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.		●	●
v7	6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices.	●	●	●
v7	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.		●	●

5.2.5 Ensure broadcast ICMP requests are ignored (Automated)

Profile Applicability:

- Level 1 - Server

Description:

Setting `net.ipv4.icmp_echo_ignore_broadcasts` to 1 will cause the system to ignore all ICMP echo and timestamp requests to broadcast and multicast addresses.

Rationale:

Accepting ICMP echo and timestamp requests with broadcast or multicast destinations for your network could be used to trick your host into starting (or participating) in a Smurf attack. A Smurf attack relies on an attacker sending large amounts of ICMP broadcast messages with a spoofed source address. All hosts receiving this message and responding would send echo-reply messages back to the spoofed address, which is probably not routable. If many hosts respond to the packets, the amount of traffic on the network could be significantly multiplied.

Impact:

Disabling icmp responses to the broadcast address will make network mapping harder, and may impact some network management tools.

Audit:

Run the following command and verify output matches:

```
# talosctl read /proc/sys/net/ipv4/icmp_echo_ignore_broadcasts
1
```

Remediation:

Run the following commands to set the active kernel parameter and persist the setting: Create the patch file that will amend the machine configuration to ignore icmp packets to broadcast addresses:

```
# cat <<EOF > icmp-echo-ignore-broadcasts.yaml
machine:
  sysctls:
    net.ipv4.icmp_echo_ignore_broadcasts: "1"
EOF
```

Apply the patch to machines:

```
talosctl patch mc --patch @icmp-echo-ignore-broadcasts.yaml
```

Default Value:

ICMP responses to echo requests to the broadcast address are disabled by default.
/proc/sys/net/ipv4/icmp_echo_ignore_broadcasts 1

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	0.0 Explicitly Not Mapped Explicitly Not Mapped			
v7	0.0 Explicitly Not Mapped Explicitly Not Mapped			

5.2.6 Ensure bogus ICMP responses are ignored (Automated)

Profile Applicability:

- Level 1 - Server

Description:

Setting `icmp_ignore_bogus_error_responses` to 1 prevents the kernel from logging bogus responses (RFC-1122 non-compliant) from broadcast reframes, keeping file systems from filling up with useless log messages.

Rationale:

Some routers (and some attackers) will send responses that violate RFC-1122 and attempt to fill up a log file system with many useless error messages.

Audit:

Run the following command and verify output matches:

```
# talosctl read /proc/sys/net/ipv4/icmp_ignore_bogus_error_responses
1
```

Remediation:

Run the following commands to set the active kernel parameter and persist the setting:
Create the patch file with the machine configuration to ignore non-compliant responses:

```
# cat <<EOF > icmp-ignore-bogus-error-responses.yaml
machine:
  sysctls:
    net.ipv4.icmp_ignore_bogus_error_responses: "1"
EOF
```

Apply the patch file to machines:

```
talosctl patch mc --patch @icmp-ignore-bogus-error-responses.yaml
```

Default Value:

`icmp_ignore_bogus_error_responses` defaults to 1 (i.e. ignoring bogus errors.)

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	0.0 Explicitly Not Mapped Explicitly Not Mapped			
v7	0.0 Explicitly Not Mapped Explicitly Not Mapped			

5.2.7 Ensure TCP SYN Cookies is enabled (Automated)

Profile Applicability:

- Level 1 - Server

Description:

When `tcp_syncookies` is set, the kernel will handle TCP SYN packets normally until the half-open connection queue is full, at which time, the SYN cookie functionality kicks in. SYN cookies work by not using the SYN queue at all. Instead, the kernel simply replies to the SYN with a SYN|ACK, but will include a specially crafted TCP sequence number that encodes the source and destination IP address and port number and the time the packet was sent. A legitimate connection would send the ACK packet of the three way handshake with the specially crafted sequence number. This allows the system to verify that it has received a valid response to a SYN cookie and allow the connection, even though there is no corresponding SYN in the queue.

Rationale:

Attackers use SYN flood attacks to perform a denial of service attacked on a system by sending many SYN packets without completing the three way handshake. This will quickly use up slots in the kernel's half-open connection queue and prevent legitimate connections from succeeding. SYN cookies allow the system to keep accepting valid connections, even if under a denial of service attack.

Audit:

Run the following command and verify output matches:

```
# talosctl read /proc/sys/net/ipv4/tcp_syncookies
1
```

Remediation:

Run the following commands to set the active kernel parameter and persist the setting:
Create the patch to enable syncookies:

```
# cat <<EOF > enable_syncookies.yaml
machine:
  sysctls:
    net.ipv4.tcp_syncookies: "1"
EOF
```

Then apply the patch to the machines:

```
talosctl patch mc --patch @enable_syncookies.yaml
```

Default Value:

Syn cookies are enabled by default. /proc/sys/net/ipv4/tcp_syncookies
1

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	0.0 Explicitly Not Mapped Explicitly Not Mapped			
v7	0.0 Explicitly Not Mapped Explicitly Not Mapped			

5.2.8 Enable Reverse Path Filtering if there are only symmetrical routes (Automated)

Profile Applicability:

- Level 2 - Server

Description:

Setting `net.ipv4.conf.all.rp_filter` and `net.ipv4.conf.default.rp_filter` to 1 forces the Linux kernel to utilize reverse path filtering on a received packet to determine if the packet was valid. Essentially, with reverse path filtering, if the return packet does not go out the same interface that the corresponding source packet came from, the packet is dropped (and logged if `log_martians` is set).

Rationale:

Setting these flags is a good way to deter attackers from sending your system bogus packets that cannot be responded to.

Impact:

One instance where this feature breaks down is if asymmetrical routing is employed. This would occur when using dynamic routing protocols (bgp, ospf, etc) on your system. If you are using asymmetrical routing on your system, you will not be able to enable this feature without breaking the routing.

Audit:

Run the following commands and verify output matches:

```
# talosctl  read /proc/sys/net/ipv4/conf/all/rp_filter
1
# talosctl  read /proc/sys/net/ipv4/conf/default/rp_filter
1
```

Remediation:

Use the following commands to enable reverse-path filtering:
Create the patch file that sets the appropriate sysctls:

```
# cat <<EOF > reversepath-filter.yaml
machine:
  sysctls:
    net.ipv4.conf.all.rp_filter: "1"
    net.ipv4.conf.default.rp_filter: "1"
EOF
```

Then apply the patch to machines:

```
talosctl patch mc --patch @reversepath-filter.yaml
```

Default Value:

Reverse path filtering is not enabled by default, due to the likelihood of it breaking functionality in asymmetrical network topologies.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	0.0 Explicitly Not Mapped Explicitly Not Mapped			
v7	0.0 Explicitly Not Mapped Explicitly Not Mapped			

5.3 Firewall Configuration

The Talos Linux Ingress Firewall is a simple and effective way to limit network access to the services running on the host, which includes both Talos Linux and Kubernetes standard services (e.g. apid and kubelet), as well as any additional workload services that may be running on the host. The Talos Linux Ingress Firewall doesn't affect the traffic between Kubernetes pods/services: please use CNI Network Policies for that.

In block mode, the ingress firewall will by default block encapsulated traffic (e.g. VXLAN) between cluster nodes. This traffic needs to be explicitly allowed for the Kubernetes networking to function properly. Please refer to the documentation for CNI in use for the required ports.

Some default CNI configurations are listed below:

- Flannel, Calico: vxlan UDP port 4789
- Cilium: vxlan UDP port 8472

Note that traffic is always allowed on the following network interfaces:

- lo
- siderolink
- kubespan

5.3.1 Configure Control Plane Firewall

The Kubernetes control plane nodes run essential services for both Kubernetes and Talos Linux cluster management, and as such should be protected from unauthorized network traffic. The following rules improve the security of the cluster and cover only standard Kubernetes and Talos Linux services. If additional services are running with host networking in the cluster, they should be covered by additional rules.

5.3.1.1 Apply an appropriate control plane firewall policy (Manual)

Profile Applicability:

- Level 2 - Server

Description:

A default deny all policy on connections to control plane nodes ensures that any unconfigured network usage will be rejected.

Rationale:

Only trusted networks should be allowed to access Kubernetes and other APIs.

Impact:

Restricting network access to services may make troubleshooting and remediation of issues more complex. For example, if remote access to `etcd` is not enabled, certain diagnosis and management commands may be not be available.

Incorrect configuration of the Ingress Policy can result in a non-accessible cluster or Talos Linux API. The configuration should be applied in `--mode=try` to ensure it is reverted in the case of a mistake.

Audit:

This will display the currently active ingress firewall policy and rules, if any.

```
talosctl read /system/state/config.yaml | yq 'select(.kind ==
"NetworkDefaultActionConfig"),select(.kind == "NetworkRuleConfig" )'
```

Remediation:

Note that this is an example policy only, which assumes there is no existing Ingress firewall configured, and provides the following restrictions:

- Talos Linux apid and Kubernetes API are accessible from anywhere
- kubelet and trustd APIs are only accessible within the cluster
- etcd API is limited to controlplane nodes

In the example we assume these template variables to describe the cluster:

- $CLUSTER_SUBNET, e.g. 172.20.0.0/24 - the subnet which covers all machines in the cluster
- $CP1, $CP2, $CP3 - the IP addresses of the controlplane nodes
- $VXLAN_PORT - the UDP port used by the CNI for encapsulated traffic

```
talosctl read /system/state/config.yaml  > controlplane.yaml
cat << EOF >> controlplane.yaml
---
apiVersion: v1alpha1
kind: NetworkDefaultActionConfig
ingress: block
---
apiVersion: v1alpha1
kind: NetworkRuleConfig
name: kubelet-ingress
portSelector:
  ports:
    - 10250
  protocol: tcp
ingress:
  - subnet: $CLUSTER_SUBNET
---
apiVersion: v1alpha1
kind: NetworkRuleConfig
name: apid-ingress
portSelector:
  ports:
    - 50000
  protocol: tcp
ingress:
  - subnet: 0.0.0.0/0
  - subnet: ::/0
---
apiVersion: v1alpha1
kind: NetworkRuleConfig
name: trustd-ingress
portSelector:
  ports:
    - 50001
  protocol: tcp
ingress:
  - subnet: $CLUSTER_SUBNET
---
apiVersion: v1alpha1
kind: NetworkRuleConfig
name: kubernetes-api-ingress
portSelector:
  ports:
    - 6443
  protocol: tcp
ingress:
  - subnet: 0.0.0.0/0
  - subnet: ::/0
---
apiVersion: v1alpha1
kind: NetworkRuleConfig
name: etcd-ingress
portSelector:
  ports:
    - 2379-2380
  protocol: tcp
ingress:
```

```
  - subnet: $CP1/32
  - subnet: $CP2/32
  - subnet: $CP3/32
---
apiVersion: v1alpha1
kind: NetworkRuleConfig
name: cni-vxlan
portSelector:
  ports:
    - $VXLAN_PORT
  protocol: udp
ingress:
  - subnet: $CLUSTER_SUBNET
EOF
talosctl apply -f controlplane.yaml
```

Default Value:

No ingress firewall is applied.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.4 Implement and Manage a Firewall on Servers Implement and manage a firewall on servers, where supported. Example implementations include a virtual firewall, operating system firewall, or a third-party firewall agent.	●	●	●
v7	9.4 Apply Host-based Firewalls or Port Filtering Apply host-based firewalls or port filtering tools on end systems, with a default-deny rule that drops all traffic except those services and ports that are explicitly allowed.	●	●	●

5.3.2 Configure Worker node Ingress Firewall

The Kubernetes worker nodes should be protected from unauthorized network traffic in order to protect the services running on them, and the cluster itself. The following rules improve the security of the cluster and cover only standard Kubernetes and Talos Linux services. If additional services are running with host networking in the cluster, they should be covered by additional rules.

5.3.2.1 Apply an appropriate worker node firewall policy (Manual)

Profile Applicability:

- Level 2 - Server

Description:

A default "deny all" policy on connections to worker nodes ensures that any unconfigured network usage will be rejected.

Rationale:

Only trusted networks should be allowed to access Kubernetes and other APIs.

Impact:

Incorrect configuration of the Ingress Policy can result in a non-accessible cluster or Talos Linux API. The configuration should be applied in `--mode=try` to ensure it is reverted in the case of a mistake.

Incorrect configuration can render application deployments on the worker nodes unreachable.

Audit:

This will display the currently active ingress firewall policy and rules, if any:

```
talosctl read /system/state/config.yaml | yq 'select(.kind ==
"NetworkDefaultActionConfig"),select(.kind == "NetworkRuleConfig" )'
```

Remediation:

Note that this is an example policy only, which assumes there is no existing Ingress firewall configured, and provides the following restrictions:

- Talos Linux apid and kubelet API are accessible only from within the cluster. (Control plane nodes are used as endpoints to terminate `talosctl` commands, and will relay API requests to the workers.)

The example assumes these template variables to describe the cluster:

- $CLUSTER_SUBNET, e.g. 172.20.0.0/24 - the subnet which covers all machines in the cluster
- $CP1, $CP2, $CP3 - the IP addresses of the controlplane nodes
- $VXLAN_PORT - the UDP port used by the CNI for encapsulated traffic

```
talosctl read /system/state/config.yaml > worker.yaml
cat << EOF >> worker.yaml
---
apiVersion: v1alpha1
kind: NetworkDefaultActionConfig
ingress: block
---
apiVersion: v1alpha1
kind: NetworkRuleConfig
name: kubelet-ingress
portSelector:
  ports:
    - 10250
  protocol: tcp
ingress:
  - subnet: $CLUSTER_SUBNET
---
apiVersion: v1alpha1
kind: NetworkRuleConfig
name: apid-ingress
portSelector:
  ports:
    - 50000
  protocol: tcp
ingress:
  - subnet: $CLUSTER_SUBNET
---
apiVersion: v1alpha1
kind: NetworkRuleConfig
name: cni-vxlan
portSelector:
  ports:
    - $VXLAN_PORT
  protocol: udp
ingress:
  - subnet: $CLUSTER_SUBNET
EOF
talosctl apply -f worker.yaml
```

Default Value:

No ingress firewall is applied.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	4.4 Implement and Manage a Firewall on Servers Implement and manage a firewall on servers, where supported. Example implementations include a virtual firewall, operating system firewall, or a third-party firewall agent.	●	●	●
v7	9.4 Apply Host-based Firewalls or Port Filtering Apply host-based firewalls or port filtering tools on end systems, with a default-deny rule that drops all traffic except those services and ports that are explicitly allowed.	●	●	●

5.4 Network Encryption

Network encryption is required for compliance with standards such as PCI that require encryption of data in transit. While Kubernetes does not natively support encryption, Talos Linux enables full node to node network encryption with automated secure key exchange simply by enabling KubeSpan.

5.4.1 Enable KubeSpan where network level encryption is required (Manual)

Profile Applicability:

- Level 1 - Server

Description:

KubeSpan securely and transparently establishes full encryption between all members of a cluster – even if they are running on completely different networks and behind firewalls. All cluster members will find each other, and automatically update reachability information.

KubeSpan operates outside of Kubernetes. It works with any CNI, and even if Kubernetes is not ready or broken.

KubeSpan provides full Wireguard based encryption for all traffic with the Kubernetes cluster, automatically.

KubeSpan uses UDP port 51820 to carry all KubeSpan encrypted traffic. Thus best practice is to ensure that one end of all possible node-node communication paths allows UDP port 51820, inbound.

KubeSpan adds no value in a single node cluster.

Rationale:

Encryption of network traffic provides defense against untrusted networks or bad actors with access to the physical network infrastructure. Ensuring traffic is encrypted prevents such actors obtaining confidential information such as security tokens, certificates, or passwords.

Impact:

Enabling Wireguard encryption adds overhead and reduces network throughput.

Audit:

Ensure each node reports a KubeSpan identity:

```
talosctl get kubespanidentities
NODE             NAMESPACE   TYPE                ID      VERSION
ADDRESS                                        PUBLICKEY
192.168.64.118   kubespan    KubeSpanIdentity    local   1
fdbc:4650:47f9:5902:c6c:8ff:fe6d:ca1c/128
8/0Xz4C8twGaEUIf3CZPj9/wXpPsTeNI5NArhurRpWY=
```

You can also examine the status of the nodes KubeSpan peers:

```
talosctl get kubespanpeerstatuses
ID                                              VERSION   LABEL
ENDPOINT           STATE   RX         TX
06D9QQOydzKrOL7oeLiqHy9OWE8KtmJzZII2A5/FLFI=    63        talos-default-
controlplane-2
172.20.0.3:51820   up      15043220   17869488

THtfKtfNnzJs1nMQKs5IXqK0DFXmM//0WMY+NnaZrhU=    62        talos-default-
controlplane-3
172.20.0.4:51820   up      14573208   18157680

nVHu7l13uZyk0AaI1WuzL2/48iG8af4WRv+LWmAax1M=    60        talos-default-
worker-2
172.20.0.6:51820   up      130072     46888
```

Remediation:

To enable KubeSpan for a new cluster, we can use the `--with-kubespan` flag in `talosctl gen config` to generate the appropriate configurations.
To enable KubeSpan on each node in an existing cluster:
Create a patch file with the machine configuration to enable KubeSpan:

```
cat <<EOF > kubespanPatch.yaml
machine:
  network:
    kubespan:
      enabled: true
cluster:
  discovery:
    enabled: true
EOF
```

Apply the patch to the nodes:

```
talosctl patch mc --patch @kubespanPatch.yaml
```

Default Value:

KubeSpan is not enabled by default.

References:

1. https://www.talos.dev/latest/talos-guides/network/kubespan/

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	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).		●	●
v7	14.4 Encrypt All Sensitive Information in Transit Encrypt all sensitive information in transit.		●	●

6 Logging and Auditing

The items in this section describe how to configure logging using tools included in the distribution.

In addition to the local log files, it is also recommended that sites collect copies of their system logs on a secure, centralized log server via an encrypted connection. Not only does centralized logging help sites correlate events that may be occurring on multiple systems, but having a second, remote copy of the system log information may be critical after a system compromise where the attacker has modified the local log files on the affected system(s). If a log correlation system is deployed, configure it to process the logs described in this section.

Because it is often necessary to correlate log information from many different systems (particularly after a security incident) it is recommended that the time be synchronized among systems and devices connected to the local network. The standard Internet protocol for time synchronization is the Network Time Protocol (NTP), which is supported by Talos Linux.

It is important that all logs described in this section be monitored on a regular basis and correlated to determine trends. A seemingly innocuous entry in one log could be more significant when compared to an entry in another log.

6.1 Configure Kernel Logging

Logging services should be configured to prevent information leaks and to aggregate logs on a remote server so that they can be reviewed in the event of a system compromise and ease log analysis.

Talos Linux keeps recent logs in a ring buffer. The use of a ring buffer prevents the possibility of logs consuming all storage space and impacting system operation, but means that historical log events can be overwritten from the local system. It is strongly recommended that all logs are sent to a remote server. Talos Linux provides direct support for shipping these logs to a remote server.

6.1.1 Ensure kernel logging is configured to send logs to a remote service. (Automated)

Profile Applicability:

- Level 1 - Server

Description:

Ensures that kernel logs are pushed to a remote endpoint, for better forensic correlation and to protect against local loss.

Rationale:

Kernel logs can be used for auditing and detecting security events.

Audit:

Run the following command and verify that kernel logging is configured:

```
# talosctl read /system/state/config.yaml  | yq 'select(.kind == 
"KmsgLogConfig")'; talosctl read /proc/cmdline | grep 'talos.logging.kernel'
```

If logging is configured, representative output will look like:

```
talos.logging.kernel=tcp://LOGGINGHOST:5044/
```

or

```
apiVersion: v1alpha1
kind: KmsgLogConfig
name: remote-log
url: udp://LOGGINGHOST:5044/
```

depending on the method used to configure logging.
LOGGINGHOST will be the configured IP of the log receiver.
Note that it is possible to configure kernel logging in two ways, hence the use of two commands in the above test.

Remediation:

The following commands would configure kernel logging:
Create a patch to enable logging:

```
cat <<EOF > kernel-logging.yaml
machine:
  install:
    extraKernelArgs: [talos.logging.kernel=tcp://host:5044/]
EOF
```

Then apply the patch to the machines:

```
talosctl patch mc --patch @kernel-logging.yaml
```

Default Value:

Remote shipping of kernel logs is not enabled.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	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.	●	●	●
v8	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.		●	●
v7	6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices.	●	●	●
v7	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.		●	●

6.2 Configure Service Logging

Logging services should be configured to prevent information leaks and to aggregate logs on a remote server so that they can be reviewed in the event of a system compromise and ease log analysis.

Talos Linux keeps recent logs in a ring buffer. The use of a ring buffer prevents the possibility of logs consuming all storage space and impacting system operation, but means that historical log events can be overwritten from the local system. It is strongly recommended that all logs are sent to a remote server. Talos Linux provides direct support for shipping these logs to a remote server.

6.2.1 Ensure service logging is configured (Automated)

Profile Applicability:

- Level 1 - Server

Description:

Service logging ensures that service logs are pushed to a remote endpoint.

Rationale:

Service logs can be used for auditing and detecting security events. Sending logs to a remote log server provides the ability to forensically reconstruct events which may have impacted the operations or security of a system even after a system crash or reboot.

Audit:

Run the following command and verify that kernel logging is configured:

```
# talosctl read /system/state/config.yaml | yq
'.machine.logging.destinations'
```

Remediation:

The following commands would configure service logging:
Create a patch file with the configuration to define logging:

```
cat <<EOF > service-logging.yaml
machine:
    logging:
        destinations:
            - endpoint: tcp://##LOGDESTINATION_IP##:12345
              format: "json_lines"
EOF
```

Replace ##LOGDESTINATION_IP## with the IP address of the log server.
Then apply the patch to the machines:

```
talosctl patch mc --patch @service-logging.yaml
```

Default Value:

Remote shipping of service logs is not enabled.

CIS Controls:

Controls Version	Control	IG 1	IG 2	IG 3
v8	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.	●	●	●
v8	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.		●	●
v7	6.2 Activate audit logging Ensure that local logging has been enabled on all systems and networking devices.	●	●	●
v7	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.		●	●

Appendix: Summary Table

CIS Benchmark Recommendation		Set Correctly	
		Yes	No
1	**Initial Setup**		
1.1	**Boot Settings**		
1.1.1	Install Talos with SecureBoot enabled (Automated)	☐	☐
1.2	**Encrypt System disks**		
1.2.1	Encrypt System Disks with KMS or TPM methods (Automated)	☐	☐
1.3	Ensure updated software is installed (Manual)	☐	☐
2	**Time Synchronization**		
2.1	Ensure NTP is configured (Automated)	☐	☐
3	**Kubernetes**		
3.1	Ensure control plane scheduling is disabled (Automated)	☐	☐
3.2	Ensure Pod Security Standards policy is restricted (Automated)	☐	☐
4	**Access Control**		
4.1	Ensure RBAC is enabled (Automated)	☐	☐
5	**Network Configuration**		
5.1	**Network Parameters (Host Only)**		
5.1.1	Ensure packet redirect sending is disabled (Automated)	☐	☐
5.2	**Network Parameters (Host and Router)**		
5.2.1	Ensure source routed packets are not accepted (Automated)	☐	☐

CIS Benchmark Recommendation		Set Correctly	
		Yes	No
5.2.2	Ensure ICMP redirects are not accepted (Automated)	☐	☐
5.2.3	Ensure secure ICMP redirects are not accepted (Automated)	☐	☐
5.2.4	Ensure suspicious (martian) packets are logged (Automated)	☐	☐
5.2.5	Ensure broadcast ICMP requests are ignored (Automated)	☐	☐
5.2.6	Ensure bogus ICMP responses are ignored (Automated)	☐	☐
5.2.7	Ensure TCP SYN Cookies is enabled (Automated)	☐	☐
5.2.8	Enable Reverse Path Filtering if there are only symmetrical routes (Automated)	☐	☐
5.3	**Firewall Configuration**		
5.3.1	**Configure Control Plane Firewall**		
5.3.1.1	Apply an appropriate control plane firewall policy (Manual)	☐	☐
5.3.2	**Configure Worker node Ingress Firewall**		
5.3.2.1	Apply an appropriate worker node firewall policy (Manual)	☐	☐
5.4	**Network Encryption**		
5.4.1	Enable KubeSpan where network level encryption is required (Manual)	☐	☐
6	**Logging and Auditing**		
6.1	**Configure Kernel Logging**		
6.1.1	Ensure kernel logging is configured to send logs to a remote service. (Automated)	☐	☐

CIS Benchmark Recommendation		Set Correctly	
		Yes	No
6.2	**Configure Service Logging**		
6.2.1	Ensure service logging is configured (Automated)	☐	☐

Appendix: CIS Controls v7 IG 1 Mapped Recommendations

Recommendation		Set Correctly	
		Yes	No
1.3	Ensure updated software is installed	☐	☐
5.2.4	Ensure suspicious (martian) packets are logged	☐	☐
5.3.1.1	Apply an appropriate control plane firewall policy	☐	☐
5.3.2.1	Apply an appropriate worker node firewall policy	☐	☐
6.1.1	Ensure kernel logging is configured to send logs to a remote service.	☐	☐
6.2.1	Ensure service logging is configured	☐	☐

Appendix: CIS Controls v7 IG 2 Mapped Recommendations

Recommendation		Set Correctly	
		Yes	No
1.1.1	Install Talos with SecureBoot enabled	☐	☐
1.3	Ensure updated software is installed	☐	☐
2.1	Ensure NTP is configured	☐	☐
5.2.4	Ensure suspicious (martian) packets are logged	☐	☐
5.3.1.1	Apply an appropriate control plane firewall policy	☐	☐
5.3.2.1	Apply an appropriate worker node firewall policy	☐	☐
5.4.1	Enable KubeSpan where network level encryption is required	☐	☐
6.1.1	Ensure kernel logging is configured to send logs to a remote service.	☐	☐
6.2.1	Ensure service logging is configured	☐	☐

Appendix: CIS Controls v7 IG 3 Mapped Recommendations

Recommendation		Set Correctly	
		Yes	No
1.1.1	Install Talos with SecureBoot enabled	☐	☐
1.2.1	Encrypt System Disks with KMS or TPM methods	☐	☐
1.3	Ensure updated software is installed	☐	☐
2.1	Ensure NTP is configured	☐	☐
5.2.4	Ensure suspicious (martian) packets are logged	☐	☐
5.3.1.1	Apply an appropriate control plane firewall policy	☐	☐
5.3.2.1	Apply an appropriate worker node firewall policy	☐	☐
5.4.1	Enable KubeSpan where network level encryption is required	☐	☐
6.1.1	Ensure kernel logging is configured to send logs to a remote service.	☐	☐
6.2.1	Ensure service logging is configured	☐	☐

Appendix: CIS Controls v8 IG 1 Mapped Recommendations

Recommendation		Set Correctly	
		Yes	No
1.3	Ensure updated software is installed	☐	☐
5.2.4	Ensure suspicious (martian) packets are logged	☐	☐
5.3.1.1	Apply an appropriate control plane firewall policy	☐	☐
5.3.2.1	Apply an appropriate worker node firewall policy	☐	☐
6.1.1	Ensure kernel logging is configured to send logs to a remote service.	☐	☐
6.2.1	Ensure service logging is configured	☐	☐

Appendix: CIS Controls v8 IG 2 Mapped Recommendations

Recommendation		Set Correctly	
		Yes	No
1.1.1	Install Talos with SecureBoot enabled	☐	☐
1.2.1	Encrypt System Disks with KMS or TPM methods	☐	☐
1.3	Ensure updated software is installed	☐	☐
2.1	Ensure NTP is configured	☐	☐
5.2.4	Ensure suspicious (martian) packets are logged	☐	☐
5.3.1.1	Apply an appropriate control plane firewall policy	☐	☐
5.3.2.1	Apply an appropriate worker node firewall policy	☐	☐
5.4.1	Enable KubeSpan where network level encryption is required	☐	☐
6.1.1	Ensure kernel logging is configured to send logs to a remote service.	☐	☐
6.2.1	Ensure service logging is configured	☐	☐

Appendix: CIS Controls v8 IG 3 Mapped Recommendations

Recommendation		Set Correctly	
		Yes	No
1.1.1	Install Talos with SecureBoot enabled	☐	☐
1.2.1	Encrypt System Disks with KMS or TPM methods	☐	☐
1.3	Ensure updated software is installed	☐	☐
2.1	Ensure NTP is configured	☐	☐
4.1	Ensure RBAC is enabled	☐	☐
5.2.4	Ensure suspicious (martian) packets are logged	☐	☐
5.3.1.1	Apply an appropriate control plane firewall policy	☐	☐
5.3.2.1	Apply an appropriate worker node firewall policy	☐	☐
5.4.1	Enable KubeSpan where network level encryption is required	☐	☐
6.1.1	Ensure kernel logging is configured to send logs to a remote service.	☐	☐
6.2.1	Ensure service logging is configured	☐	☐

Appendix: Change History

Date	Version	Changes for this version
7/22/2024	1.0.0	Initial Release