Guide
Cisco public

Cisco Compute UCS Manager Hardening Guide

Version 2.0

July 2025

Contents

Document information

Document summary	Prepared for	Prepared by
V2.0	Cisco Field	Aaron Kapacinskas
Changes		
Updated SED section with controller and drive security flag information		
Updated SED section with table describing tri-mode controller behavior for various settings		
Updated SED section with encrypted VD creation mechanics		
Updated key management section for SEDs describing KMIP client certificate behavior and support		
Removed Appendix B for policy configuration and linked to new external document on same		
Relabeled Appendices since Appendix B was removed		
Added section on vendor key updates for secure boot		

Intended use and audience

This document contains confidential material that is proprietary to Cisco Systems, Inc. The materials, ideas, and concepts contained herein are to be used exclusively to assist in the configuration of Cisco® hardware and software solutions.

Bias statement

The documentation set for this product strives to use bias-free language. For purposes of this documentation set, bias-free is defined as language that does not imply discrimination based on age, disability, gender, racial identity, ethnic identity, sexual orientation, socioeconomic status, or intersectionality. Exceptions may be present in the documentation due to language that is hardcoded in the user interfaces of the product software, language used based on standards documentation, or language that is used by a referenced third-party product.

Legal notices

All information in this document is provided in confidence and shall not be published or disclosed, wholly or in part, to any other party without written permission from Cisco.

Prerequisites

We recommend reviewing the Cisco UCS® release notes, installation guide, and user guide before proceeding with any configuration. Please contact Cisco support or your Cisco representative if you need assistance.

Introduction

This guide focuses on implementing a Cisco Unified Computing System™ (Cisco UCS) with emphasis on best practices regarding security. The guide focuses on deploying Cisco UCS in UCSM Managed Mode (UMM). This deployment scenario sees the compute system utilizing Cisco UCS Manager with an integrated fabric presented by a pair of fabric interconnects defining the Cisco UCS domain. This hardening guide will explore the Cisco UCS ecosystem, hardware capabilities, software settings, policies, and service profiles that are critical to a secure deployment.

Development philosophy

At the core of the Cisco UCS platform lies a development philosophy centered on proactive security measures. Using an approach designed for preemptive threat mitigation and continuous enhancement, Cisco leverages in-house technologies and research to fortify its UCS architecture against emerging threats. Incorporating robust industry practices and adhering to stringent security protocols, the Cisco UCS platform is built to meet the highest standards of security certifications, ensuring compliance with regulatory frameworks and assuring customers of a resilient and safeguarded infrastructure. Moreover, the management features embedded within the Cisco UCS solution provide administrators with comprehensive tools for monitoring, auditing, and controlling access, enabling proactive threat identification and rapid response to potential security breaches.

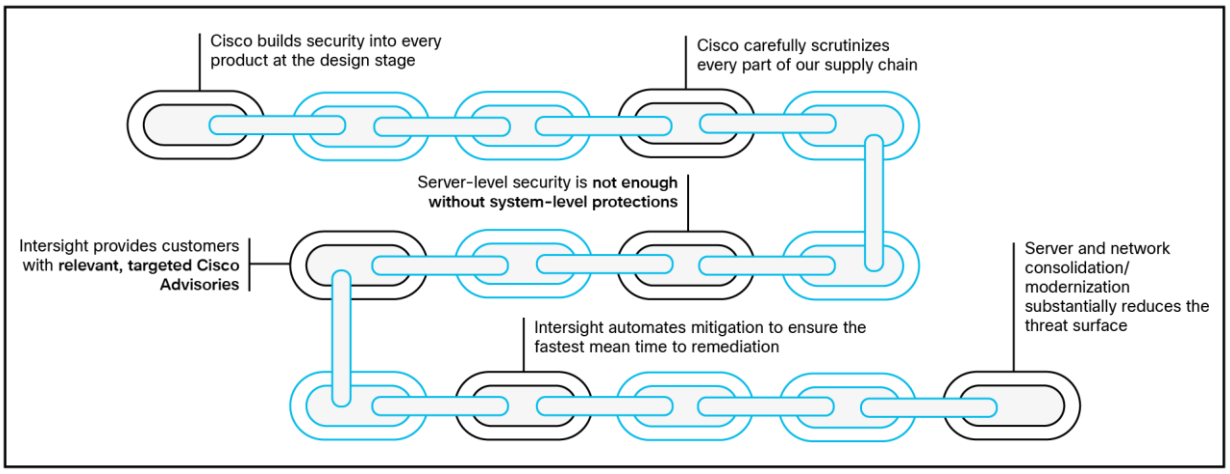

Figure 1.
The Cisco security value chain.

In addition to its development and certification framework, Cisco UCS utilizes advancements in confidential computing and secure storage to keep user applications and data protected. Implementing NIST-approved encryption techniques, secure boot processes, and hardware-based isolation mechanisms, Cisco UCS ensures data confidentiality, integrity, and availability throughout its lifecycle. Through secure storage solutions, and federally certified secure interfaces, users can leverage the Cisco UCS platform confidently, knowing their data remains protected against unauthorized access. This white paper discusses the implementation of these features, demonstrating how Cisco UCS meets and exceeds the security and accountability requirements in today's enterprise environments.

Cisco Unified Computing System solution components

Cisco UCS has a unique architecture that integrates compute, data network access, and storage network access into a common set of components under a single-pane-of-glass management interface. Cisco UCS fuses access layer networking and servers. This high-performance, next-generation server system provides a data center with a high degree of workload agility and scalability. The hardware and software components support Cisco's unified fabric, which runs multiple types of data center traffic over a single converged network adapter.

A Cisco UCS compute system is available in many blade- or rack-mount configurations. Systems that come with Fabric Interconnects (FIs) can run with Cisco Unified Computing System Manager (UCSM or UCSM Managed Mode, UMM) or with Cisco Intersight cloud-management services (Intersight Managed Mode, IMM). Systems without Fis will run in standalone mode and can be managed through a Baseboard Management Console (BMC; also called the Cisco Integrated Management Console [Cisco IMC]) or with Intersight.

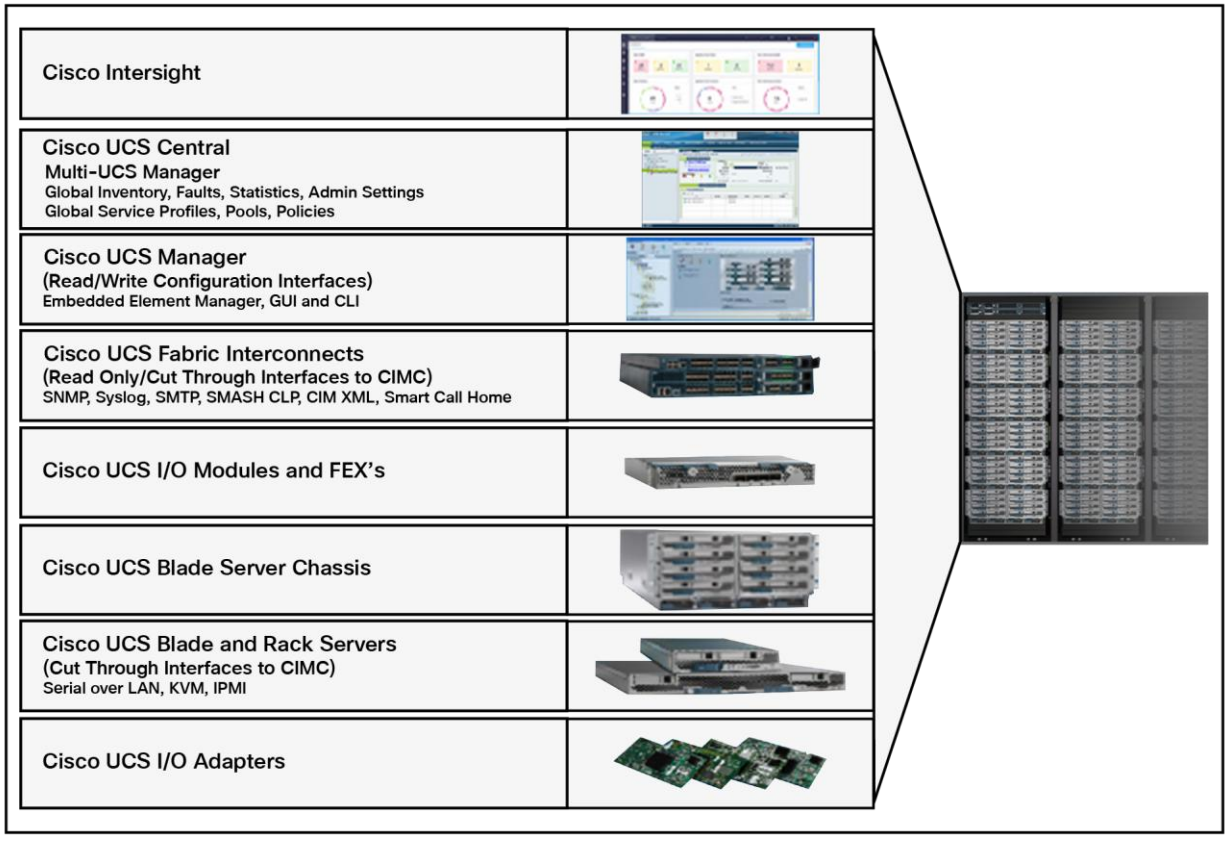

Figure 2.
The Cisco UCS ecosystem with various server platforms and I/O adapters complemented with robust management and monitoring tools.

Cisco UCS ecosystem

A Cisco UCS fabric interconnect is a networking switch or head unit to which the Cisco UCS chassis or rack server connects. The fabric interconnect is a core part of Cisco UCS. Cisco UCS is designed to improve scalability and reduce the Total Cost of Ownership (TCO) of data centers by integrating all components into a single platform that acts as a single unit. Access to networks and storage is provided through the Cisco UCS fabric interconnect. Each server is dual connected, with one Small Form-Factor Pluggable (SFP) port for each fabric interconnect for high availability. This design helps ensure that all virtual NICs (vNICs) within Cisco UCS are dual connected as well, essentially guaranteeing node availability.

Unified fabric

With a unified fabric, multiple types of data-center traffic can run over a single Data-Center Ethernet (DCE) network. Instead of having a series of different Host Bus Adapters (HBAs) and Network Interface Cards (NICs) present in a server, a unified fabric uses a single converged network adapter. This type of adapter can carry LAN and SAN traffic on the same cable.

Cisco UCS uses Fibre Channel over Ethernet (FCoE) to carry Fibre Channel and Ethernet traffic on the same physical Ethernet connection between the fabric interconnect and the server. This connection terminates at a converged network adapter on the server, and the unified fabric terminates on the uplink ports of the fabric interconnect. On the core network, the LAN and SAN traffic remains separated. Cisco UCS does not require that you implement a unified fabric across the data center. The converged network adapter presents an Ethernet interface and a Fibre Channel interface to the operating system. At the server, the operating system is not aware of the FCoE encapsulation because it sees a standard Fibre Channel HBA.

At the fabric interconnect, the server-facing Ethernet port receives the Ethernet and Fibre Channel traffic. The fabric interconnect (using Ethertype to differentiate the frames) separates the two traffic types. Ethernet frames and Fibre Channel frames are switched to their respective uplink interfaces.

The following is a summary of the interconnected components in a typical deployment.

- **Cisco UCS Manager**—Cisco UCS Manager is the centralized management interface for Cisco UCS. Cisco UCS Manager is the interface used to set up the fabric interconnects for Cisco UCS service profiles and for general hardware management. For more information on Cisco UCS Manager, see “Introduction to Cisco UCS Manager” in the Cisco UCS Manager Getting Started Guide.
- **Cisco UCS fabric interconnects**—The Cisco UCS fabric interconnect is the core component of Cisco UCS deployments, providing both network connectivity and management capabilities for the Cisco UCS system. Cisco UCS fabric interconnects run the Cisco UCS Manager control software and consist of the following components:
  - Different generation of Cisco UCS fabric interconnects, for example, Cisco UCS 6536/6400/6300 fabric interconnect
  - Transceivers for network and storage connectivity
  - Expansion modules for various fabric interconnects
  - Cisco UCS Manager software
  - For more information on Cisco UCS fabric interconnects, see Cisco UCS Fabric Infrastructure Portfolio.

- **Cisco UCS I/O modules and Cisco UCS Fabric Extenders (FEXs)**—I/O modules are also known as Cisco FEXs or simply FEX modules. These modules serve as line cards to the FIs in the same way that Cisco Nexus® series switches can have remote line cards. I/O modules also provide interface connections to blade servers. They multiplex data from blade servers and provide this data to FIs and do the same in the reverse direction. In production environments, I/O modules are always used in pairs to provide redundancy and failover.
- **Cisco UCS blade server chassis**—The Cisco UCS 5100 Series Blade Server Chassis is a crucial building block of Cisco UCS, delivering a scalable and flexible architecture for current and future data center needs, while helping reduce total cost of ownership.
- **Cisco UCS blade and rack servers**—Cisco UCS B-Series blade servers are at the heart of the UCS solution. They come in various system resource configurations in terms of CPU, memory, and hard disk capacity. Cisco UCS rackmount servers are standalone servers that can be installed and controlled individually. Cisco provides Fabric Extenders (FEXs) for rackmount servers. FEXs can be used to connect and manage rackmount servers from FIs; rackmount servers can also be directly attached to the fabric interconnect.
- **Cisco UCS I/O adapters**—Cisco UCS B-Series blade servers are designed to support up to two network adapters. This design can reduce the number of adapters, cables, and access-layer switches by as much as half because it eliminates the need for multiple parallel infrastructure for both LAN and SAN at the server, chassis, and rack levels.

Upstream switches

Upstream or Top-of-Rack (ToR) switches are required to manage north-south traffic, that is, traffic to resources outside the fabric interconnect connectivity. You should configure the upstream switches to accommodate nonnative VLANs for Ethernet traffic.

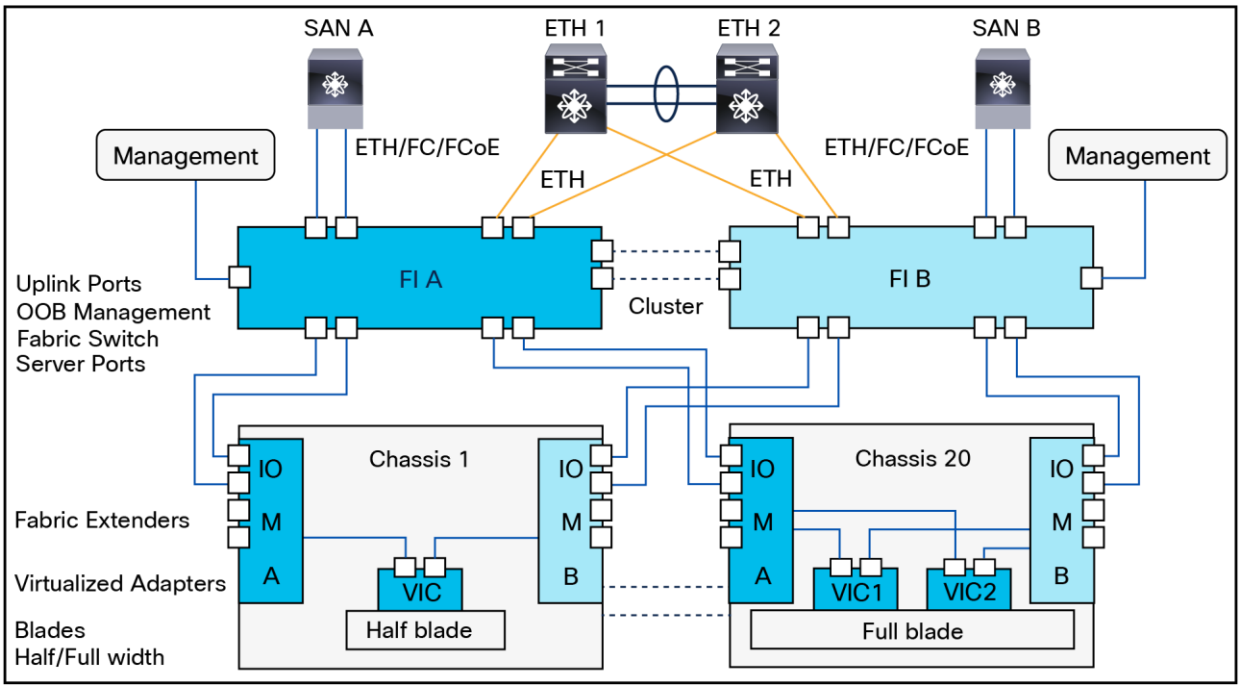

Figure 3.
Typical Cisco UCS VIC to FI to upstream switch connectivity.

Server types

Cisco UCS servers come in a variety of types. These include rack servers and chassis-based blade (half- and full-size). Some servers are specialized for specific applications (for example, AI) or for specific deployment types (for example, edge deployments). Such systems will have specific hardware designed to facilitate the workload, such as a bank of GPUs or a dedicated DPU-based smart NIC with an embedded firewall. For specific model details and the full line of available server form factors and capabilities, see the Cisco documentation. Regardless of the server type, the systems utilizing UMM will need the same or closely similar security considerations.

Cisco Integrated Management Console

The Cisco UCS C-Series Rack-Mount Server ships with Cisco Integrated Management Console (Cisco IMC) firmware. Cisco IMC is a separate management module built into the motherboard. A dedicated ARM-based processor, separate from the main server CPU, runs the Cisco IMC firmware. The system ships with a running version of the firmware. You can update the firmware, but no initial installation is needed.

Cisco UCS C-Series rack servers support operating systems such as Windows, Linux, VMware ESXi, Oracle, and so on. For more information on supported operating systems, see Standalone C-Series UCS Server Compatibility at http://www.cisco.com/en/US/products/ps10477/prod_technical_reference_list.html. You can use the Cisco IMC to install an OS on the server using the KVM console and vMedia (Virtual Media).

The Cisco IMC management service is used only when the server is operating in standalone mode. If your Cisco UCS C-Series server is integrated into a UCS system with fabric interconnects, you must manage it using Cisco UCS Manager or you must use the Cisco Intersight SaaS service to claim and manage the system.

Cisco UCS sizing

Sizing a Cisco UCS system, regardless of management mechanism (that is. UCSM, IMM, or standalone), is an important aspect of securing the system because the nature of the deployment can determine or guide security decisions. For example, if you deploy your system using a virtualization solution, then using Intel® Trust Domain Extensions (Intel TDX) could be an important consideration for fencing virtual environments when designing for confidential computing. Bare-metal or containerized deployments might suggest use of Intel SGX or AMD SEV. There are also sizing ramifications to keep in mind regarding storage at rest for applications you may be targeting or for air-gapped and non-networked environments.

Sizing a Cisco Unified Computing System (Cisco UCS) for applications involves determining the appropriate hardware resources and configurations to meet the performance and capacity requirements of applications whether they run on bare metal or in virtualized environments. The following are some guidelines on how to size Cisco UCS systems for applications in each scenario.

Sizing for bare-metal applications:

- Define application requirements:
  - Identify specific resource requirements of the application, including CPU, memory, storage, and network bandwidth
  - Consider peak workloads, expected growth, and any specific performance characteristics of the application
- Select the appropriate Cisco UCS server model:
  - Choose a Cisco UCS server model that aligns with the performance and scalability requirements of the application
  - Consider factors such as the number of sockets and cores, memory capacity, available PCIe slots, and storage options
  - Choose a CPU that meets your preferred confidential computing needs
- Configure CPU and memory:
  - Determine the optimal CPU configuration based on the application's CPU utilization patterns
  - Allocate sufficient memory to meet the application's requirements, considering such factors as caching, data processing, and scalability
- Storage configuration:
  - Select the appropriate storage configuration, including the type of storage (for example, HDD, SSD) and RAID levels
  - Consider the required storage capacity, I/O performance, and redundancy needs
  - Determine if you need to use Data-at-Rest Encryption (DARE) and select SEDs if needed
- Network considerations:
  - Size the network infrastructure based on the application's network bandwidth requirements
  - Determine the number and type of network interfaces needed for the application
- Power and cooling requirements:
  - Assess the power and cooling requirements of the chosen Cisco UCS server to ensure they align with the data center's capabilities
- Consider future growth:
  - Plan for future growth by selecting a Cisco UCS server model that provides scalability to accommodate increased workloads over time

Sizing for virtualized environments:

- Hypervisor selection:
  - Choose a hypervisor (for example, VMware vSphere, Microsoft Hyper-V) based on the application's compatibility and feature requirements
  - Choose an appropriate CPU to meet your confidential computing needs specific to virtualized deployments
- Calculate virtual resource requirements:
  - Estimate the resource requirements for each Virtual Machine (VM), including vCPUs, memory, and storage
  - Consider factors such as resource overcommitment, VM density, and peak usage patterns
- Determine host-to-VM ratio:
  - Decide on the host-to-VM ratio based on the application's characteristics and the capabilities of the selected Cisco UCS server model
  - Consider factors such as CPU and memory oversubscription, workload variability, and HA (High Availability) requirements
- Consider network and storage virtualization:
  - Plan for network virtualization (for example, VLANs, VXLANs) and storage virtualization (for example, SAN or NAS integration) to meet the needs of virtualized workloads

In both scenarios, working closely with application owners, understanding the application's characteristics, and regularly monitoring and adjusting the infrastructure are critical for ensuring optimal performance and resource utilization. It is also at this time that you should consider how this system will fit into any key-management solutions you have for encrypted services and how you will implement any firewall or other perimeter access and authentication solutions. Cisco UCS Manager provides a centralized platform for managing and configuring servers, providing a unified approach to hardware management in both bare-metal and virtualized environments.

Component failure and redundancy

Cisco Unified Computing System (Cisco UCS) is designed with a strong emphasis on hardware redundancy to enhance system reliability and availability. The hardware redundancy in Cisco UCS involves redundant components and mechanisms to handle failures effectively. In encrypted environments, it is critical that access to the PKI infrastructure be maintained both for reliable access to systems and for any potential secure decommissioning that may need to occur.

Here are some key aspects of Cisco UCS hardware redundancy and how failures are handled:

- Unified fabric and I/O modules:
  - Cisco UCS fabric interconnects are deployed in redundant failover pairs.
  - I/O modules are designed with redundancy. If one module fails, traffic is automatically redirected through the redundant module.

- Blade server chassis:
  - In the case of Cisco UCS X-Series blade servers, the chassis itself is designed for redundancy.
  - Power supplies and fans within the chassis are redundant, ensuring that the failure of one component does not disrupt the operation of the entire system.
- Power-supply redundancy:
  - Cisco UCS rack servers come with redundant power supplies to ensure continuous power availability.
- Service profiles and stateless computing:
  - Cisco UCS uses service profiles to abstract the server's identity and configuration from its physical hardware.
  - If a blade server fails, its service profile and associated security settings can be quickly associated with a spare blade, reducing downtime.
- Predictive failure:
  - Predictive failure alerts are generated to notify administrators of components that are likely to fail soon, allowing for proactive replacement.
- Management redundancy with Cisco UCS Manager and Intersight:
  - Cisco UCS Manager, the central management software, is often deployed in a redundant configuration.
  - Cloud-based Intersight-managed servers benefit from the inherent availability of cloud services.
- Automated failure handling:
  - Cisco UCS is designed to automatically handle failures without manual intervention.
  - When a failure occurs, the system can automatically reroute traffic, shift workloads, or initiate other recovery measures to minimize downtime.

By integrating these redundancy features, Cisco UCS aims to deliver a highly reliable and available computing infrastructure. The emphasis on automated failure handling and proactive monitoring helps reduce the impact of hardware failures and ensures the continuous operation of critical workloads in data-center environments.

The Cisco Secure Development Lifecycle – CSDL

Cisco products and components are developed, integrated, and tested using the Cisco Secure Development Lifecycle (CSDL). Secure product development and deployment have several components, ranging from following specified design and development practices, to testing their implementation, to providing customers with a set of recommendations for deployments that maximize the security of their system.

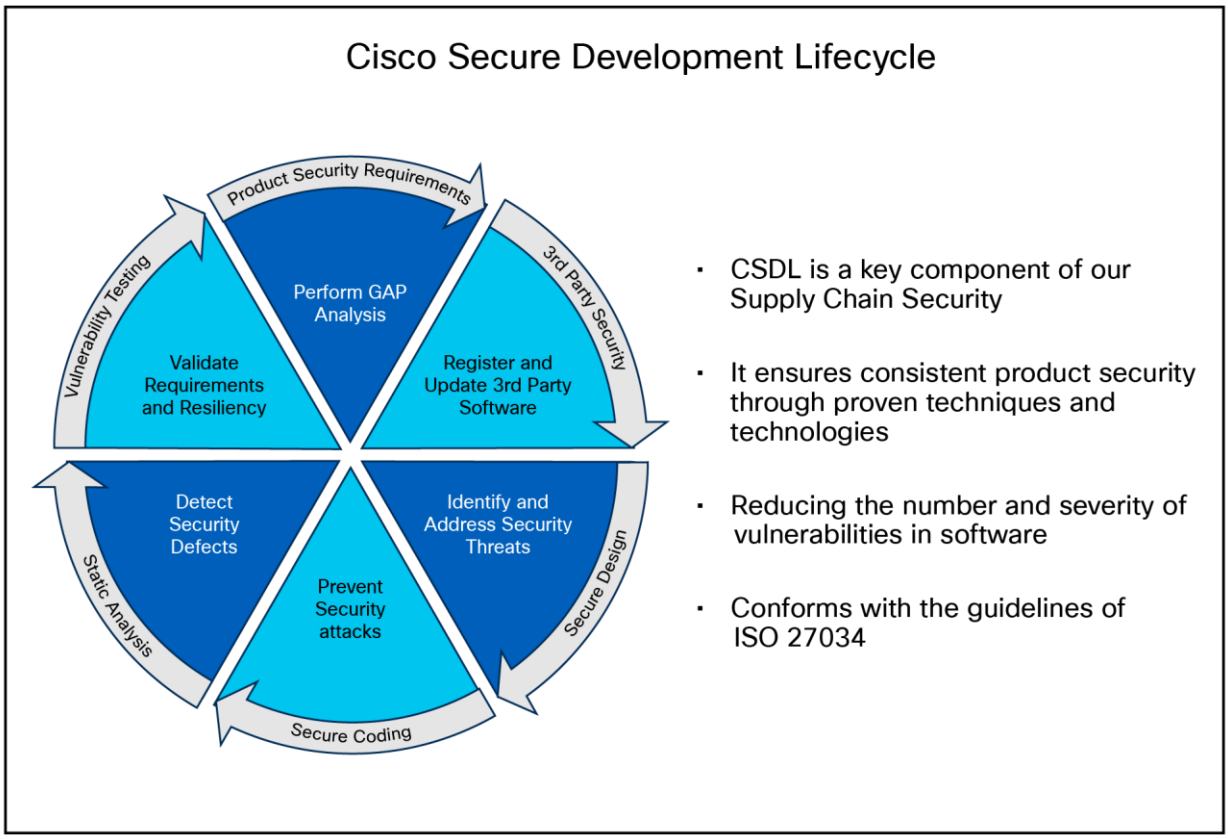

Figure 4.
The Cisco Secure Product Development Lifecycle.

CSDL philosophy

A poor product design can open the way to vulnerabilities. The CSDL is designed to mitigate these potential issues. At Cisco, our secure-design approach requires two types of considerations:

- Design with security in mind
- Use threat modeling to validate the design's security

Designing with security in mind is an ongoing commitment to personal and professional improvement through:

- Training
- Applying Product Security Baseline (PSB) design principles
- Considering other industry-standard secure-design principles
- Being aware of common attack methods and designing safeguards against them
- Taking full advantage of designs and libraries that are known to be highly secure
- Protecting all potential entry points

Cisco also reduces design-based vulnerabilities by considering known threats and attacks:

- Follow the flow of data through the system
- Identify trust boundaries where data may be compromised
- Based on the data-flow diagram, generate a list of threats and mitigations from a database of known threats, tailored by product type
- Prioritize and implement mitigations to the identified threats

The goal of this effort is to enforce a set of security processes and ensure a security mindset at every stage of development:

- Secure design
- Secure coding
- Secure analysis
- Vulnerability testing
- Secure deployments

Development milestones

Each iteration of the product’s development addresses needs for ongoing security fixes and general feature enhancements that include security components (new deployment models, changes in management, partner onboarding, etc.). At every stage of development, the product(s) undergo potential enhancements relative to findings and new features.

- The system is configured in the Quality-Assurance (QA) testing stage to accommodate the relevant settings identified above and run through a typical deployment test.
- The result is a validated set of best practices for security and is communicated through the CSDL process and exposed in the documentation.

CSDL product adherence methodologies

Cisco CSDL adheres to Cisco Product-Development Methodology (PDM), ISO/IEC 27034, and ISO 9000 compliance requirements. The ISO/IEC 27034 standard provides an internationally recognized standard for application security. Details for ISO/IEC 27034 can be found here. The ISO 9000 family of quality management systems standards is designed to help organizations ensure that they meet the needs of customers and other stakeholders while meeting statutory and regulatory requirements related to a product or service. ISO 9000 details are here.

The CSDL process is not a one-time approach to product development. It is recursive, with vulnerability testing, penetration testing, and threat modelling added to subsequent development of CSDL. This process follows ISO 9000 and ISO 27034 standards as part of an internationally recognized set of guidelines. The approaches involved often use a solution-wide methodology; for example, utilizing our continually updated Cisco SSL crypto module to guarantee that Cisco UCS (along with other elements in the Cisco offering) is always secure and meets FIPS certification requirements.

Cisco Security and Trust Organization (S&TO)

Cisco Security and Trust Organization has the core responsibility to implement CSDL. In the effort to accomplish this, S&TO encompasses various groups with core responsibilities around delivering a secure product or responding to security concerns as they arise.

Figure 5.
The various groups within Cisco S&TO.

Supply-chain security

A critical aspect of secure product development and deployment is ensuring that the components that go into the system are legitimate and uncompromised. To this end, Cisco takes exceptional measures to ensure supply-chain integrity.

Counterfeit prevention

The Cisco Value Chain describes the development model used for all Cisco products, including HyperFlex®. Cisco is a leader in industry and international standards on counterfeit reduction and has been engaged in decades-long efforts to prevent and detect the distribution of counterfeit products. Cisco incorporates tools and processes to prevent counterfeiting—beginning with product development, through the manufacturing process, and in the marketplace.

In collaboration with Cisco's Brand Protection, Legal, and other teams at Cisco, an end-user portal has been developed to aid customers in these efforts and can be accessed at: anticounterfeit.cisco.com.

Cisco's Brand Protection Team has conducted numerous investigations into counterfeiting operations and worked with local law enforcement to disrupt those operations. The portal includes examples of the Brand Protection Team's work over the years, and the numerous resources that are available for Cisco customers and partners.

The Cisco Value Chain has the following characteristics:

- Comprehensiveness across all stages of a solution's lifecycle
- Multilayer approach, focused protection against:
  - Source code corruption
  - Hardware counterfeit
  - Misuse of intellectual property

This multilayered approach is shown below.

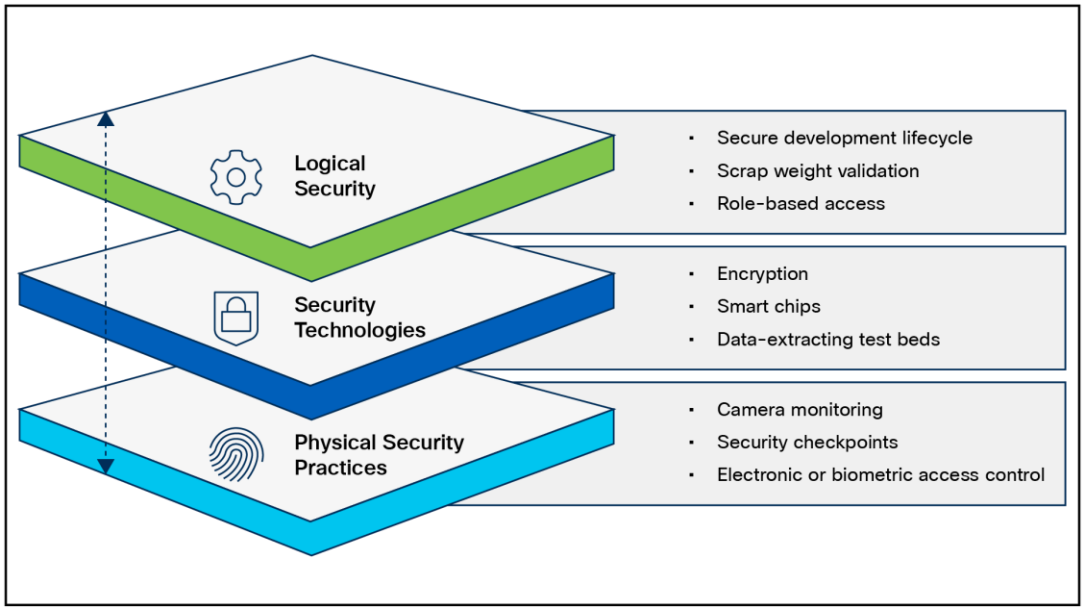

Figure 6.
Layers of the Cisco Value Chain

Consortiums for secure vendors

Table 1. Secure vendor-consortium memberships

Name	Component(s)	Description	Status
TAPA	Supply chain	The Transported Asset Protection Association's (TAPA) Security Standards act as a worldwide benchmark for supply-chain security and resilience, providing guidance, processes, and tools that reduce loss exposure, protect assets, and the costs of cargo theft.	Member
CTPAT	Supply chain	Customs Trade Partnership Against Terrorism (CTPAT) Trade Compliance Program is a voluntary program that provides the opportunity for importers who have made a commitment of resources to assume responsibility for monitoring their own compliance in exchange for benefits.	Member

Advisories, vulnerabilities, and incident responses

CERT advisory

Computer Emergency Response Team (CERT) advisories come up as new vulnerabilities are identified. Cisco's internal CERT team monitors and alerts product groups to potential issues that might affect their respective components. When these items are identified by CERT or are otherwise indicated by vendor partners (VMware, etc.), patches are either developed or acquired from the respective vendors.

Incident response

The Cisco Product Security Incident Response Team (PSIRT) is responsible for responding to Cisco product security incidents. The Cisco PSIRT is a dedicated, global team that manages the receipt, investigation, and public reporting of information about security vulnerabilities and issues related to Cisco products and services. Cisco defines a security vulnerability as a weakness in the computational logic (that is, the code) found in software and hardware components that, when exploited, results in a negative impact to confidentiality, integrity, or availability. Cisco reserves the right to deviate from this definition based on specific circumstances. The Cisco PSIRT adheres to ISO/IEC 29147:2018, which are guidelines for disclosure of potential vulnerabilities established by the International Organization for Standardization.

The on-call Cisco PSIRT works 24 hours a day with Cisco customers, independent security researchers, consultants, industry organizations, and other vendors to identify possible security vulnerabilities and issues with Cisco products and networks.

All vulnerabilities disclosed in Cisco Security Advisories are assigned a Common Vulnerability and Exposures identifier (CVE-ID) and a Common Vulnerability Scoring System (CVSS) score to aid in identification. Additionally, all vulnerabilities are classified based on a Security Impact Rating (SIR).

Cisco uses version 3.1 of the Common Vulnerability Scoring System (CVSS) as part of its standard process of evaluating reported potential vulnerabilities in Cisco products. The CVSS model uses three distinct measurements or scores that include Base, Temporal, and Environmental calculations. Cisco provides an evaluation of the Base vulnerability score and, in some instances, a Temporal vulnerability score. End users are encouraged to compute an Environmental score based on their network parameters.

In addition, Cisco uses the Security Impact Rating (SIR) as a way to categorize vulnerability severity in a simpler manner. The SIR is based on the CVSS Base score, adjusted by PSIRT to account for variables specific to Cisco, and is included in every Cisco Security Advisory.

Cisco PSIRT assigns a Common Vulnerabilities and Exposures Identifier (CVE-ID) to any vulnerability that is found in a Cisco product and that qualifies to receive this identifier. Usually, all vulnerabilities with Medium, High, or Severe SIRs — that is, a CVSS score of 4.0 or greater — will qualify for a CVE-ID.

CVE and vulnerability remediation

CVE reporting is a function of the previously mentioned PSIRT alert mechanism and is the first step in vulnerability remediation. Once a CVE is known to affect a system, the patched release should be identified.

Additional vulnerability testing measures

Cisco also utilizes an internal tool for threat modeling called Threat-builder. This tool is used to explicitly map out application components and services and to identify potential attack surfaces and develop line items for direct evaluation. This information along with industry tools is used for vulnerability and exploit testing by Cisco's ASIG (Advanced Security Initiatives Group). ASIG also uses fuzzing and manual testing as part of their suite of tools.

Certifications and compliance

Certification process

Federal compliance and audit-based certifications are critical components of a standardized and predictable security posture. They are critical in most federal deployments, especially those dealing with financial and defense arenas. The Cisco Global Certification Team (GCT) works to complete various certifications.

Common Criteria (CC)

Common Criteria for Information Technology Security Evaluation (Common Criteria or CC) is an international standard (ISO/IEC 15408) for computer security certification, currently in v3.1 rev 5.

- System users specify their security functional and assurance requirements through the use of protection profiles. Vendors can then make claims about the security attributes of their products, and testing laboratories can evaluate the products to determine if they actually meet the claims.
    - Customers have security needs defined in a set of CC guidelines and the evaluation is conducted as follows:
        - This is my system; this is what I say it can do to meet those.
        - Let us (vendor and lab) agree on a test, here is the procedure.
        - Here are my results.
        - You (lab) run it on your own and verify.
        - Deliver certification

A key part of Evaluation Assurance Level (EAL) is the Security Target document. This comprises a rigorous definition of functions, features, and intended use, tailored for the specific hardware or software component under test (the Target of Evaluation [TOE]). The EAL rating determines the extent of the testing, and the confidence that security is as claimed. Simply stated, EAL indicates the degree to which something does what it says

Cisco's Unified Computing System (Cisco UCS) has achieved Common Criteria (CC) certification, which is a globally recognized standard for evaluating the security of IT products. This certification ensures that Cisco UCS meets stringent security requirements, making it a reliable choice for organizations with high-security needs. Cisco UCS products, such as Cisco UCS Manager and various server models, have undergone rigorous evaluation to achieve this certification. This includes assessments of their secure installation, configuration, and operational use.

EAL evaluation for Cisco UCS are continuously ongoing. The current EAL2 UCS evaluations are as follows:

```
#20-0038305 EAL2 non-NDPP
CIMC 4.0 EAL2
C-Series, S-Series

#19-228723 EAL2 non-NDPP
UCSM 4.0
B-Series, C-Series, S-Series, 2200, 2300, 2400, 6300
```

FIPS

The Federal Information Processing Standard (FIPS) Publication 140-2 and 140-3 is a U.S. government computer security standard used to approve cryptographic modules.

Cisco UCS is compliant with FIPS140-2 level 1 through direct implementation of the FIPS-compliant CiscoSSL crypto module. The module, once implemented, is vetted by a third-party that is federally certified to ascertain compliance status. Cisco UCS is compliant with FIPS 140-3 level 1 for new releases of UCSM and CIMC firmware moving forward beginning in fall of 2024 with systems using CiscoSSL and CIscoSSH with the FIPS Object Model (FOM) v7.3a or later.

Cisco UCS systems have the following FIPS compliant features:

- Utilizes CiscoSSL module
  - Is already FIPS compliant
  - Has an SSH-approved cipher list
  - Provides SSL/TLS implementation
  - Eliminates weak or compromised components
- Is regularly updated

- Has a module lab-validated to have been incorporated correctly
  - Builds logs
  - Provides source access identifying calls to the module
  - Ensures that all admin access points to the cluster are covered here
- Uses SSH for CLI
- Uses HTTPS for UI

A comprehensive list of Cisco FIPS-compliant products is listed here along with the corresponding reference with NIST:

- Cisco FIPS-certified products: http://www.cisco.com/c/en/us/solutions/industries/government/global-government-certifications/fips-140.html
- Cryptographic Module Validation Program (CMVP) vendor list: http://csrc.nist.gov/groups/STM/cmvp/documents/140-1/1401vend.html
- Cryptographic Module Validation Program (CMVP) Certificate #4747: https://csrc.nist.gov/projects/cryptographic-module-validation-program/certificate/4747

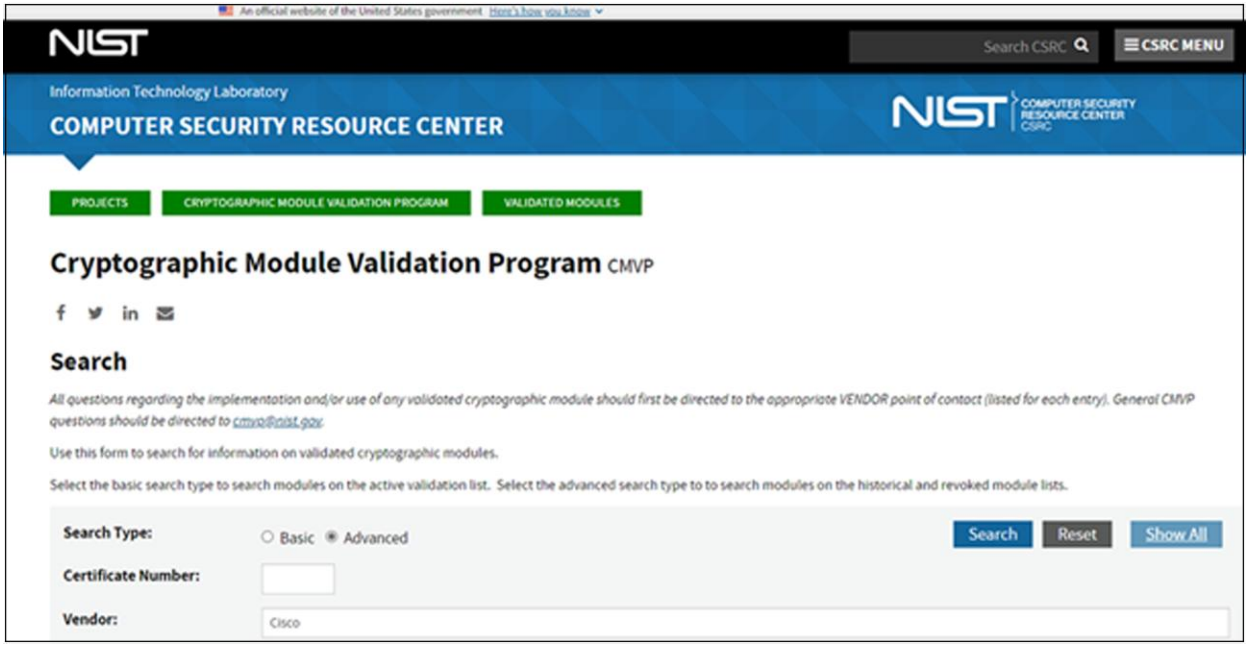

Figure 7.
FIPS vendor listings

CNSA (Commercial National Security Algorithm)

This is a schema that is detailed in RFC 9151: RFC 9151: Commercial National Security Algorithm (CNSA) Suite Profile for TLS and DTLS 1.2 and 1.3 (rfc-editor.org).

The Commercial National Security Algorithm (CNSA) describes which algorithms should be in use and what their profiles should look like. It is intended to give guidance for secure and interoperable communications, including guidelines for certificates, for national security reasons.

Cisco supports both Elliptic Cryptographic Certificates (ECC) and RSA certificates, so these requirements are met:

- Elliptic Curve Digital Signature Algorithm (ECDSA) and Elliptic Curve Diffie-Hellman (ECDH) key pairs are on the curve P-384. FIPS 186-4, Appendix B.4, provides useful guidance for elliptic curve key pair generation that should be followed by systems that conform to the RFC.
- RSA key pairs (public or private) are identified by the modulus size expressed in bits; RSA-3072 and RSA-4096 are computed using moduli of 3072 bits and 4096 bits, respectively. Cisco's FIPS certification through CiscoSSL implements federally approved crypto modules to satisfy the complexity requirements as well.

CNSA compliance is just a matter of making sure to implement a cryptographic ecosystem according to the CNSA requirements since Cisco UCS supports all the documented methods.

Other certifications and procedural guidelines

ISO/IEC 27001 is not a certification for specific pieces of hardware as much as it is a dozen or so "best practices" in the form of checklists and guidelines for how organizations manage their security controls internally. It observes such things as building access, password management, badging into a copier to make copies, etc. Training on a frequent basis is a part of the standard.

Cisco is ISO/IEC 27001-certified. This is a link to our ISO/IEC 27001 certificate: Cisco Secure Cloud Analytics (StealthWatch®) ISO/IEC 27001:2013, 27017:2015, 27018:2019.

IPV6

The Office of Management and Budget (OMB) has directed [OMB-2020, OMB-2010, OMB-2005] the National Institute of Standards and Technology (NIST) to develop the technical infrastructure (standards and testing) necessary to support wide-scale adoption of IPv6 in the U.S. Government (USG). In response, NIST developed a technical standards profile for U.S. Government acquisition of IPv6-enabled networked information technology. The USGv6 Profile includes a forward-looking set of protocol specifications published by the Internet Engineering Task Force (IETF), encompassing basic IPv6 functionality, and specific requirements and key optional capabilities for routing, security, multicast, network management, and quality of service.

The profile also contains NIST-defined requirements for IPv6-aware firewalls and intrusion detection systems. The program also established a robust testing infrastructure to enable IPv6 products to be tested for compliance to profile requirements and for interoperability by accredited laboratories using standardized test methods. Cisco UCS platforms are in the process of completing this qualification.

DISA APL

The Defense Information Security Agency Approved Product List is a multifaceted federal certification that gives approval for products to operate in secure environments. It is currently under way with the Cisco Global Certification Team and the Cisco Compute Business Unit.

Other certifications and procedural guidelines

ISO 27001 is not a certification for specific pieces of hardware as much as a dozen or so “best practices” in the form of checklists and guidelines for how organizations manage their security controls internally. It observes things such as building access, password management, badging into a copier to make copies, etc. Training on a frequent basis is a part of the standard.

Cisco is ISO 27001 certified. This is a link to our various ISO certificates: Trust Portal - Cisco

The Cisco Intersight Platform has completed its ISO 27001:2013 First Surveillance Audit from the external certification body and auditor Coalfire, and the certificate issued has been uploaded to Trust Portal site. The First Surveillance Audit included a review of the establishment and overall operating effectiveness of control areas that form Cisco Intersight’ s Information Security Management System.

Other NIST compliance

Platform FW resiliency, BIOS protection guidelines, BIOS integrity measurement

The following NIST guidelines describe how to properly implement firmware and BIOS software in a product. Cisco UCS firmware and BIOS implementations are guided by and compliant with these specifications.

NIST 800-193: https://csrc.nist.gov/pubs/sp/800/193/final

NIST 800-147B: https://csrc.nist.gov/News/2014/SP-800-147B,-BIOS-Protection-Guidelines-for-Server

NIST 800-155: https://csrc.nist.gov/pubs/sp/800/155/ipd

Cybersecurity Maturity Model Certification (CMMC)

Cybersecurity Maturity Model Certification (CMMC) is an assessment framework and assessor certification program designed to increase the trust in measures of compliance to a variety of standards published by NIST.

NIST 800-171

Conducting a NIST 800-171 self-assessment — also known as a CMMC self-assessment or SPRS assessment — is a critical component of DFARS 252.204-7019 compliance. This is dependent on your deployment scenario, and you need to evaluate your organization against all 320 objectives and upload your score to the Supplier Performance Risk System (SPRS).

Data sanitization

Cisco UCS is compliant with NIST-based data sanitization standards. See the section “Securely decommissioning a system,” below.

NIST 800-88: SP 800-88 Rev. 1, Guidelines for Media Sanitization | CSRC

Post Quantum Cryptography and UCS

See “Appendix B – PQC definitions” for definitions of various PQC terminology.

NSA defines the cryptography requirements for National Security Systems (NSS) used in Commercial National Security Algorithm (CNSA) Suite documents. CNSA is the NSA’s mandated suite of conventional algorithms, and CNSA 2.0 is the post-quantum suite. A list of the CNSA 1.0 and CNSA 2.0 algorithms is shown below.

CNSA requirements are enforced by inclusion in Common Criteria (CC) and Commercial Solution for Classified (CSfC) certifications. New versions of Common Criteria (CC) Protection Profiles (PPs) are being created that include the use of CNSA 1.0 or CNSA 2.0 requirements. The new PPs are expected to be published starting in October 2024 and completed in 4Q CY 2025. CSfC currently requires CNSA 1.0. CSfC updates allowing CNSA 2.0 are expected to be available in 4Q CY 2025.

Of particular interest is a new NDcPP (network device collaborative protection profile), expected to be published in 2025. By 2026, network devices will be required to comply with either CNSA 1.0 or 2.0. The transition is dependent on use cases, such as FW/SW signatures and verification, when it is not feasible to support both CNSA 1.0 and 2.0. Many use cases, such as transport protocols, allow support for both CNSA 1.0 and 2.0.

CNSA 2.0 instructs government buyers to prefer compliance in 2026, and it requires compliance by 2030. CNSA 2.0-required compliance will likely be accelerated to 2027 for CSfC.

Table 2. PQC algorithms

Function/use case	Algorithms	
	CNSA 1.0	**CNSA 2.0**
General system-wide, secret-based encryption and decryption	AES-256	
	FIPS PUB 197	
General system-wide secure key exchange protocol	ECDH-384	ML-KEM-1024 (CRYSTAL-Kyber 1024)
	DH-3072	
	RSA-3072	FIPS-203
SUDI and AIK certificates’ signature signing and verification	ECC P-384	ML-DSA-87 (CRYSTALS-Dilithium)
	FIPS PUB 186-4 (superseded by 186-5 in Feb 2024)	FIPS-204
	RSA-3072	
	FIPS PUB 186-4 (superseded by 186-5 in Feb 2024)	

Function/use case	Algorithms	
	CNSA 1.0	**CNSA 2.0**
General system-wide hashing usage	SHA	SHA
	FIPS 180-4	FIPS 180-4
	Use SHA-384 for all classification levels	Use SHA-384 or SHA-512 for all classification levels
Image signing	RSA-3072	LMS*
		FIPS SP 800-208
	FIPS PUB 186-4 (superseded by 186-5 in Feb 2024)	RFC 8554
		*Currently supported by SWIMS
		XMSS
		FIPS SP 800-208
	ECC P-384	RFC 8391
	FIPS PUB-186-4 (superseded by 186-5 in Feb 2024)	ML-DSA-87 (CRYSTALS-Dilithium)
		FIPS-204

For **general encryption**, used when we access secure websites, NIST has selected the CRYSTALS-Kyber algorithm. Among its advantages are comparatively small encryption keys that two parties can exchange easily, as well as its speed of operation.

For digital **signatures**, used when we need to verify identities during a digital transaction or to sign a document remotely, NIST has selected the three algorithms CRYSTALS-Dilithium, FALCON, and SPHINCS+ (read as “Sphincs plus”). Reviewers noted the high efficiency of the first two, and NIST recommends CRYSTALS-Dilithium as the primary algorithm, with FALCON for applications that need smaller signatures than Dilithium can provide. The third, SPHINCS+, is somewhat larger and slower than the other two, but it is valuable as a backup for one chief reason: it is based on a different math approach than all three of NIST’s other selections.

Software priorities

The top priority for Software (SW) is PQC for transport protocols to protect against “Harvest Now, Decrypt Later” (HNDL) attacks. In these scenarios, users are at risk of having their information exposed in the future. This is mitigated through the use of PQC algorithms. CiscoSSL and CiscoSSH, the crypto modules used in UCSM and CIMC, are currently in early testing before general availability.

The second priority is image signing and verification. While initially used to support quantum-safe hardware requirements, support will travel up the software stack as verification capabilities become available with various vendors (for example, Microsoft) providing PQC keys for use.

The third priority is identities and certificates. Viable support depends on numerous external entities, such as standards (NIST, IETF, etc.), PKI vendors, and the Certification Authority Browser (CAB) Forum. The migration to PQC certificates will occur once all the industry vendor pieces are in place.

Hardware priorities

The top priorities for New Product Introduction (NPI) Hardware (HW) are PQC algorithms for software/firmware verification and device identities. CNSA 2.0 requests vendors to upgrade their existing products to versions that have these PQC capabilities. Users have asked Cisco about this upgrade capability. However, many Cisco devices support LDWM for secure-boot bootloader validation, a quantum-safe algorithm; therefore, it is not recommended to update a device's identity for security concerns. In-field upgrades of Cisco hardware to incorporate PQC capabilities are not warranted in most cases.

System-level security

System boot

Secure system boot relies on a set of trusted Cisco technologies. Here are the fundamental concepts of Cisco Trustworthy Technologies:

Chain of trust

A chain of trust exists when the integrity of each element of code on a system is validated before that piece of code is allowed to run. A chain of trust starts with a root of trust element. The root of trust validates the next element in the chain (usually firmware) before it is allowed to start, and so on. Through the use of signing and trusted elements, a chain of trust can be created which boots the system securely and validates the integrity of Cisco software.

Cisco Secure Boot

Cisco Secure Boot helps to ensure that the code that executes on Cisco hardware platforms is authentic and unmodified. Cisco hardware-anchored secure boot protects the microloader (the first piece of code that boots) in tamper-resistant hardware, establishing a root of trust that helps prevent Cisco network devices from executing tainted network software. Subsequent boot of the installed operating system is verified and attested with the Trusted Platform Module (TPM).

Cisco Secure Boot helps ensure that the code that executes on Cisco hardware platforms is genuine and untampered. A typical Unified Extensible Firmware Interface (UEFI)-based boot process without secure boot starts at the UEFI firmware and works up to the boot loader and the operating system. A tampered UEFI firmware can result in the entire boot process being compromised.

Using a hardware-anchored root of trust, digitally signed software images, and a unique device identity, Cisco hardware-anchored secure boot establishes a chain of trust that boots the system securely and validates the integrity of the software. The root of trust (also known as a microloader), which is protected by tamper-resistant hardware, first performs a self-check and then verifies the UEFI firmware, and thus kicks off the chain of trust leading up to the integrity verification of the entire Cisco IOS® XR operating system.

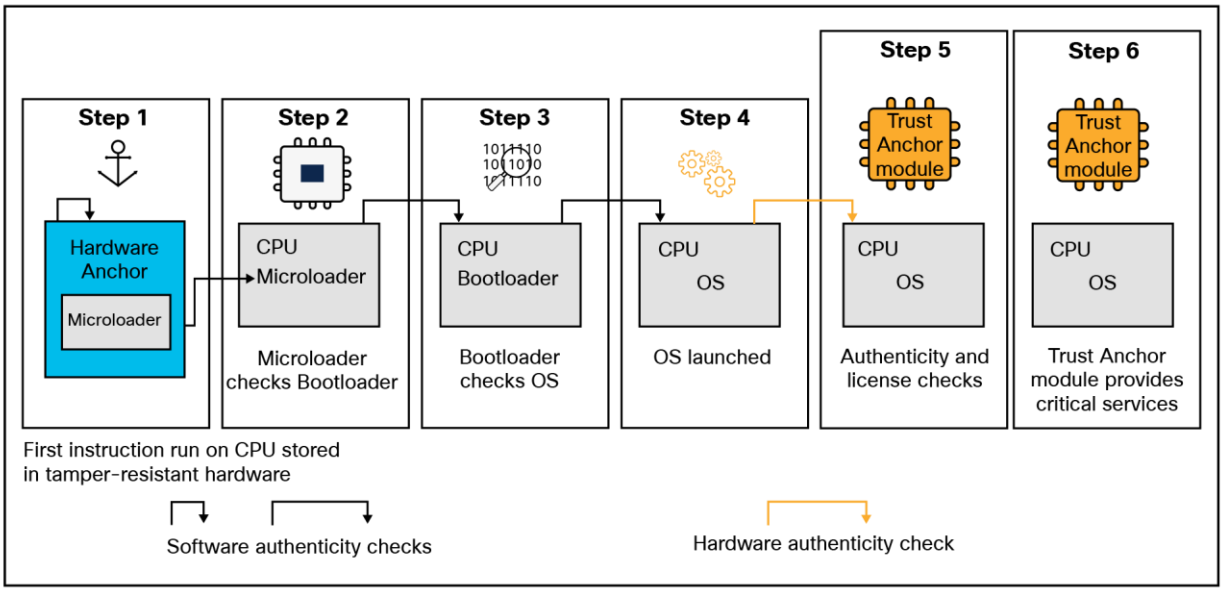

Figure 8.
Cisco Secure Boot process.

Image signing

Image signing is a two-step process for creating a unique digital signature for a given block of code. First, a hashing algorithm, similar to a checksum, is used to compute a hash value of the block of code. The hash is then encrypted with a Cisco private key, resulting in a digital signature that is attached to and delivered with the image. Signed images may be checked at runtime to verify that the software has not been modified.

Hardware root of trust – Trust Anchor module and Trusted Platform Module (2.0)

A trusted element in the scope of system software is a piece of code that is known to be authentic. A trusted element must either be immutable (stored in such a way as to prevent modification) or authenticated through validation mechanisms. Cisco anchors the root of trust, which initiates the boot process, in tamper-resistant hardware. The hardware-anchored root of trust protects the first code running on a system from compromise and becomes the root of trust for the system.

The Trust Anchor module (TAm) is a proprietary, tamper-resistant chip found in many Cisco products and features nonvolatile secure storage, a Secure Unique Device Identifier, and crypto services, including random number generation (RNG), secure storage, key management, and crypto services to the running OS and applications.

The hardware root of trust is a Cisco ACT2 Trust Anchor module (TAm). This module has the following characteristics:

- Immutable identity with IEEE 802.1AR (Secure UDI- X.509 cert)
- Anti-theft and anti-counterfeiting
- Built-in cryptographic functions
- Secure storage for certificates and objects
- Certifiable NIST SP800-92 random number generation

Once a system is securely booted, it is often important to get external verification that this is indeed the case. This is done through attestation. “Attestation” is evidence of a result; for example, “The host was booted with secure boot enabled and signed code.” This is accomplished through the Trusted Platform Module (TPM).

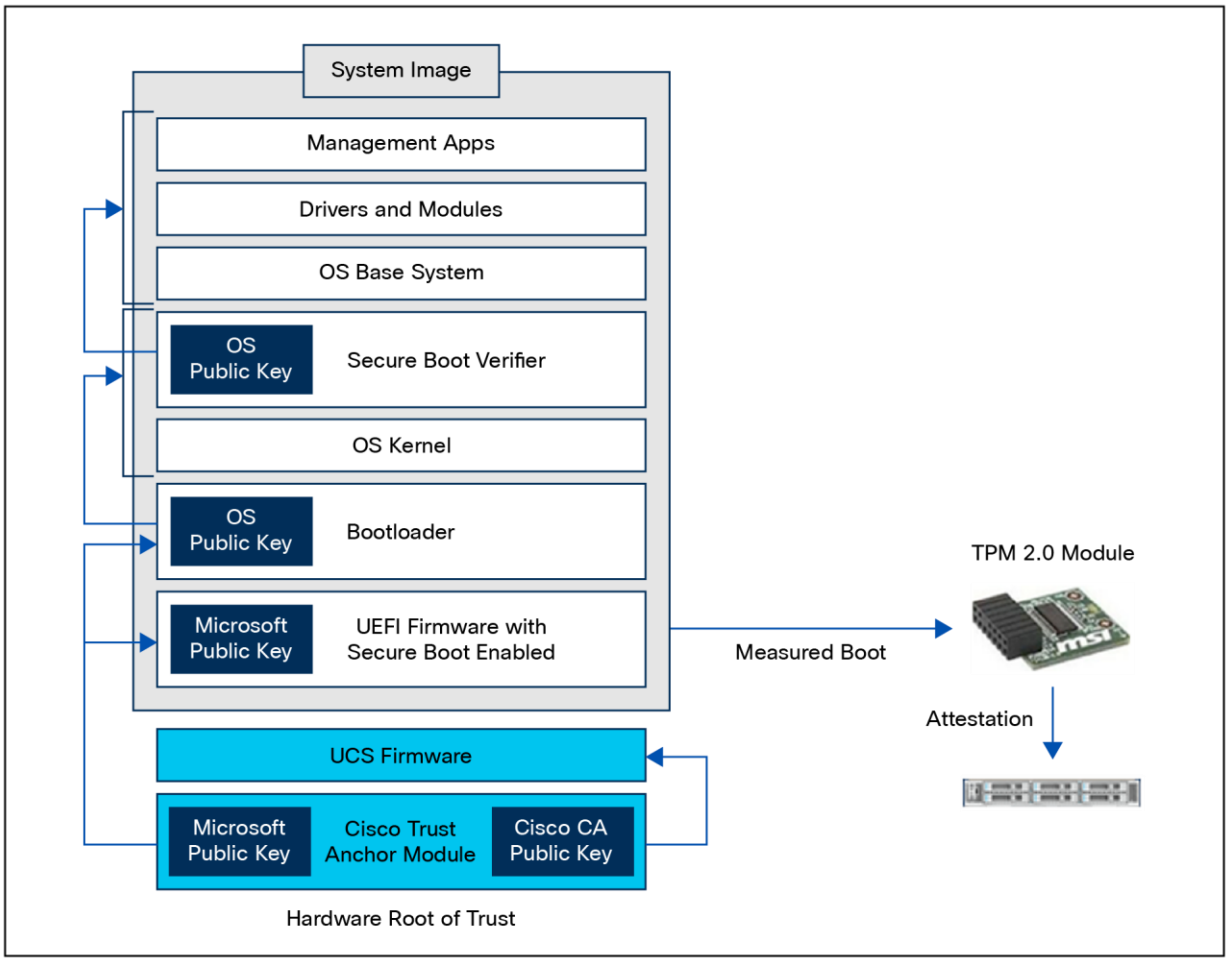

Figure 9.
Use of TAm and TPM in the entire process.

Cisco sources its TPMs from Infineon. The current Cisco UCS TPMs are SLB9670 and SLB9672.

Here’s a listing of the EAL certifications for Infineon TPMs: https://csrc.nist.gov/projects/cryptographic-module-validation-program/validated-modules/search?SearchMode=Basic&Vendor=infineon&CertificateStatus=Active&ValidationYear=0

Immutable identity

The Secure Unique Device Identifier, or SUDI, is an X.509v3 certificate, which maintains the product identifier and serial number. The identity is implemented at manufacturing and is chained to a publicly identifiable root certificate authority. The SUDI can be used as an unchangeable identity for configuration, security, auditing, and management.

The SUDI credential in the Trust Anchor module can be based on either an RSA or an Elliptic Curve Digital Signature Algorithm (ECDSA). The SUDI certificate, the associated key pair, and its entire certificate chain are stored in the tamper-resistant Trust Anchor module chip. Furthermore, the key pair is cryptographically bound to a specific Trust Anchor chip, and the private key is never exported. This feature makes cloning or spoofing the identity information virtually impossible.

The SUDI can be used for asymmetric key operations such as encryption, decryption, signing, and verifying that allow passage of the data to be operated upon. This capability makes remote authentication of a device possible. It enables accurate, consistent, and electronic identification of Cisco products for asset management, provisioning, version visibility, service entitlement, quality feedback, and inventory management.

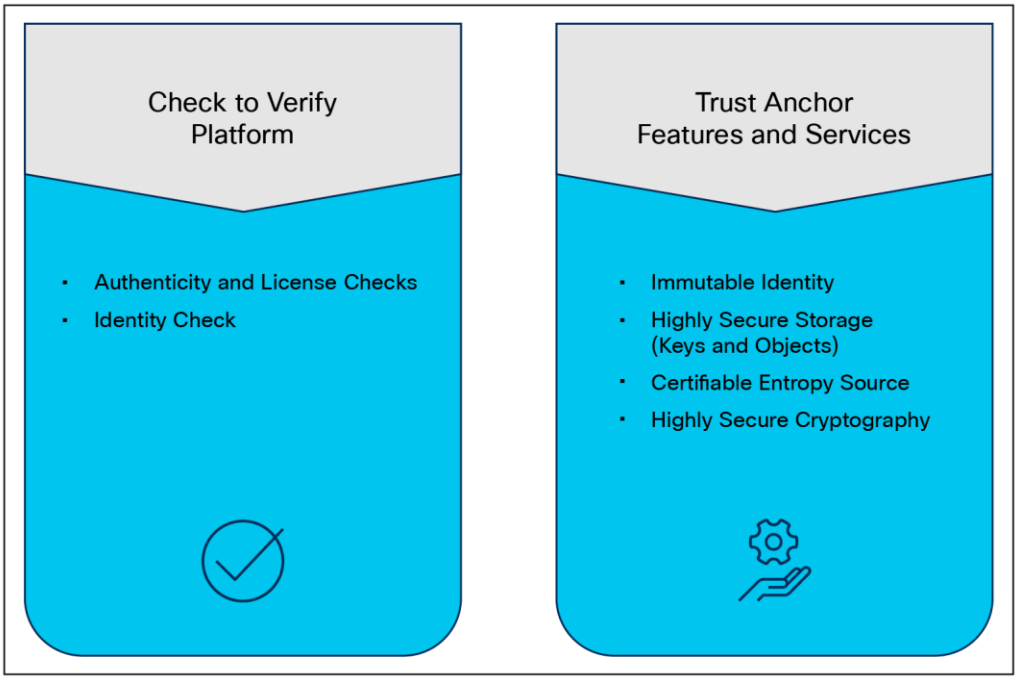

Figure 10.
TAm functions

Currently the secure boot process, when enabled, is in effect during boot including both the system firmware and the installed operating system. The end-to-end security model that this enables, when combined with the secure UI and CLI, encompasses the hardware Trust Anchor module (TAm), to secure the system boot, to secure the OS boot with externally verifiable attestation using the Trusted Platform Module (TPM).

This implementation covers the following:

- Secure boot, secured by public keys stored in the write-protected hardware root of trust
- Ensuring that only a trusted OS image, including drivers, is booted by verifying signatures
- Support of attestation of secure boot through TPM 2.0

The detailed process flow for secure boot of the system and OS with attestation capability is shown below. Note that the certificate-based hardware root of trust validates the Cisco UCS firmware, which ensures a clean BIOS that is set for key validation of the hypervisor, bootloader, and so on. This guarantees that the hardware and hypervisor in the Cisco HyperFlex system have not been tampered with. External validation of this can be made through attestation using the TPM 2.0 module in Cisco UCS.

Card boot – TAm

Cisco UCS systems have a variety of add-in cards that serve many different functions. These range from additional VICs, to NICs, DPU offload, GPUs, and various HBAs. As part of a secure deployment posture, it is not only important to be able to securely boot the main system, including the UEFI BIOS, bootloader, and operating system. The system must also securely boot the firmware that runs on the add-in cards themselves. To this end, most cards in use with Cisco UCS have a trust anchor built in that validates the card BIOS at card boot. These systems serve the same function as the TAm on the Cisco UCS motherboard in securing server firmware and ensuring legitimate code execution at system startup.

Secure boot vendor key updates

When a vendor, such as Microsoft, updates or otherwise changes their secure signatures, these keys need to be updated on the UCS system to maintain secure boot operations. These updated certificates are embedded in the signed and secured Cisco UCS Firmware. When the UCS system firmware is updated, these new keys are added, and secure boot continues to function. This is an ongoing process from various vendors and from Cisco and happens automatically.

Runtime defenses

Runtime defenses (RTDs) target injection attacks of malicious code into running software. Cisco runtime defenses include Address Space Layout Randomization (ASLR), Built-in Object Size Checking (BOSC), and X-Space Security Enhancement. Runtime defenses are complementary.

Cisco Runtime defenses enable the following protections:

- Make it harder or impossible for attackers to exploit vulnerabilities in running software
- Are complementary; can be implemented individually or deployed together

CPU hardware protections

Cisco UCS supports both Intel and AMD processors. The latest generations of these CPUs and their accompanying chipsets have extensions and programmatic capabilities addressing memory encryption and secure code execution and isolation.

Intel Boot Guard

Intel Boot Guard (4th Gen CPU and greater) is a security technology designed to enhance the integrity of the boot process and protect against unauthorized firmware and bootloader modifications on systems using Intel processors. It is part of Intel's broader security initiatives to safeguard the boot process from potential threats and ensure the system starts up securely. Here are the key aspects of Intel Boot Guard:

- Boot process integrity:
  - Intel Boot Guard focuses on protecting the boot process, ensuring that the system starts up using only authorized and unaltered firmware and bootloader components.
- Hardware-based protection:
  - Intel Boot Guard operates at the hardware level, utilizing a combination of hardware-based mechanisms within the Intel chipset and processor.
- Verified boot:
  - During the boot process, Intel Boot Guard verifies the digital signature of the firmware and bootloader components before allowing them to execute. Digital signatures are used to verify the authenticity and integrity of the firmware and bootloader code.
- Measures against unauthorized modifications:
  - Intel Boot Guard helps prevent unauthorized modifications to the firmware and bootloader, protecting against various attacks that attempt to inject malicious code or compromise the boot process.
- Key rollback protection:
  - To prevent attacks that involve rolling back to a previously signed firmware version with known vulnerabilities, Intel Boot Guard includes protections against key rollback.
- Configurability:
  - System manufacturers have some flexibility in configuring Intel Boot Guard based on their specific security requirements. They can, for example, decide which firmware and bootloader components are subject to verification.
- Integration with secure boot:
  - Intel Boot Guard works in conjunction with other security technologies, such as Unified Extensible Firmware Interface (UEFI) Secure Boot. Secure boot ensures that only signed and authenticated code is allowed to run during the boot process.
- OEM customization:
  - Original Equipment Manufacturers (OEMs) can customize Intel Boot Guard policies to align with their specific security needs, allowing them to adapt the technology to their hardware implementations.

It's important to note that while Intel Boot Guard enhances system security, it is just one component of a comprehensive security strategy. Secure firmware, secure boot, and other security features collectively contribute to creating a more resilient and secure computing environment. Additionally, the specifics of Intel Boot Guard may vary among different Intel processor generations, so it's advisable to refer to Intel's official documentation for the most accurate and up-to-date information.

AMD Platform Secure Boot (PSB)

AMD Platform Secure Boot (PSB) is a security feature designed to enhance the security of AMD processors and platforms by focusing on the boot process. PSB is part of AMD's security initiatives to protect against unauthorized code execution during the system boot-up process.

Key features of AMD Platform Secure Boot include:

- Secure boot mechanism:
  - AMD PSB is a secure-boot mechanism that ensures the integrity of the boot process by verifying the authenticity of firmware and bootloader components before allowing them to execute.
- Protection against unauthorized code execution:
  - AMD PSB helps protect the system from threats related to unauthorized or malicious code attempting to run during the boot sequence.
- Integration with industry standards:
  - AMD PSB is designed to work in conjunction with industry-standard secure boot protocols, such as Unified Extensible Firmware Interface (UEFI) Secure Boot.
- Chain of trust:
  - AMD PSB establishes a chain of trust from the initial firmware load through the bootloader and into the operating system, ensuring that each step in the boot process is verified and secure.
- Cryptographic verification:
  - Cryptographic methods, such as digital signatures, are used to verify the authenticity and integrity of firmware and bootloader code. Only code with valid signatures is allowed to run.
- Protection against rootkits and bootkits:
  - By securing the boot process, AMD PSB helps defend against certain types of attacks, including rootkits and bootkits, which aim to compromise the system at an early stage of boot-up.
- OEM customization:
  - Original Equipment Manufacturers (OEMs) can configure and customize AMD PSB settings based on their specific security requirements. This flexibility allows OEMs to adapt the security features to their hardware implementations.
- Secure deployment of virtualization:
  - In virtualized environments, AMD PSB can contribute to the security of the hypervisor and virtual machines by ensuring a secure boot process for the entire virtualization stack.

It's important to note that the specifics of AMD PSB and its integration with different processor generations may vary. For the most accurate and up-to-date information about AMD Platform Secure Boot, refer to AMD's official documentation. Security features are continually evolving, and AMD may introduce enhancements or updates to its security technologies over time.

Security Protocol and Data Model (SPDM)

To defend against attacks targeting mutable components in Cisco UCS systems, the Security Protocol and Data Model (SPDM) specification enables a secure transport implementation that challenges a device to prove its identity and the correctness of its mutable component configuration. This feature is supported on Cisco UCS servers, starting with Cisco UCS Manager Release 4.2(1d).

SPDM defines messages, data objects, and sequences for performing message exchanges between devices over a variety of transport and physical media. It orchestrates message exchanges between Baseboard Management Controllers (BMCs) and end-point devices over a Management Component Transport Protocol (MCTP). Message exchanges include authentication of hardware identities accessing the BMC. The SPDM enables access to low-level security capabilities and operations by specifying a managed level for device authentication, firmware measurement, and certificate management. Endpoint devices are challenged to provide authentication, and BMC authenticates the endpoints and only allows access for trusted entities.

Cisco UCS Manager optionally allows uploads of external security certificates to the BMC. A maximum of 40 SPDM certificates is allowed, including native internal certificates. Once the limit is reached, no more certificates can be uploaded. User-uploaded certificates can be deleted but internal, or default, certificates cannot.

An SPDM security policy allows you to specify one of three security-level settings. Security can be set at one of the three levels listed below:

- Full security
    - This is the highest MCTP security setting. When you select this setting, a fault is generated when any endpoint authentication failure or firmware measurement failure is detected. A fault will also be generated if any of the endpoints do not support either endpoint authentication or firmware measurements.
- Partial security (default)
    - When you select this setting, a fault is generated when any endpoint authentication failure or firmware measurement failure is detected. There will NOT be a fault generated when the endpoint doesn’t support endpoint authentication or firmware measurements.
- No security
    - When you select this setting, there will NOT be a fault generated for any failure (either endpoint measurement or firmware measurement failures).

You can also upload the content of one or more external, or device, certificates into the BMC. Using an SPDM policy allows you to change or delete security certificates or settings as desired. Certificates can be deleted or replaced when no longer needed.

Default passwords

A password-strength option is enabled by default on all management modes. Strong passwords must meet the following requirements:

- Must contain a minimum of eight characters and a maximum of 64 characters
- Must contain at least three of the following:
  - Lowercase letters
  - Uppercase letters
  - Numbers
  - Special characters
- Must not contain a character that is repeated more than three times consecutively (for example, aaabb)
- Must not be identical to the username or the reverse of the username
- Must pass a password dictionary check. For example, the password must not be based on a standard dictionary word.
- Must not contain the following symbols: $ (dollar sign), ? (question mark), and = (equals sign)
- Should not be blank for local user and admin accounts

Additional password profile options:

- Change count: maximum times a password can be changed within the change interval
- Change interval: time frame used by the change count
- No change interval: minimum hours a local user must wait before changing newly created password
- Change during interval: capability to change the password during the change interval

After deployment and initial configuration are complete, make sure that any default passwords are changed or updated.

Multifactor Authentication (MFA)

Cisco UCS Manager supports use of two-factor authentication for remote user logins. Two-factor authentication login requires a username, a token, and a password combination in the password field.

Two-factor authentication is supported when you use Remote Authentication Dial-In User Service (RADIUS) or Terminal Access Controller Access Control System + (TACACS+) provider groups with designated authentication domains with two-factor authentication for those domains.

With the Duo implementation, Multifactor Authentication is performed through the Duo Authentication Proxy, which is an on-premises software service that receives authentication requests from your local devices and applications using RADIUS or LDAP, optionally performs primary authentication against your LDAP directory or RADIUS authentication server, and then contacts Duo to perform secondary authentication. Once the user approves the two-factor request, which is received as a push notification from Duo Mobile, or as a phone call or by other means, the Duo proxy returns access approval to the device or application that requested authentication.

Access methods to management and configuration interfaces

The management plane consists of functions that achieve the management goals of the system. Any management function undertaken by the user must rely on interaction through secure protocols, whether they are managing through command line or UI. This is handled through HTTPS for any UI access, whether UCSM, CIMC, or SaaS-based Cisco Intersight. Authenticated, tokenized access is used for in-house development through an API. SSH for encrypted command line access is also supported. Management security also entails role-based access control as well as auditing and logging of system activities and user input, all of which are incorporated into every management mechanism.

Interactive management sessions using the command line take advantage of SSH or SCP. This is available for UCSM or standalone CIMC deployments. These sessions take place in the embedded and abstracted management shell. This shell is hardened, does not allow root access, and cannot run user-space applications.

Role-Based Access Control

Local authentication is enabled by default. Use HTTPS and SSH for maximum security when accessing the Cisco UCS device. Numerous authentication methods provide enhanced security. There is a maximum of 48 local user accounts. Remote authentication uses LDAP, RADIUS and TACACS+ with a maximum of 16 TACACS+ servers, 16 RADIUS servers, and 16 LDAP providers for a total of 48 providers. Roles defined in these domains are used to restrict and define access for different users. Refer to the deployment and configuration guides for your specific management Role-Based Access Control (RBAC) configurations (UCSM, Intersight, local CIMC).

Authentication domains

The default (local) authentication and the console authentication can utilize different providers. Furthermore, authentication grouping uses a maximum of 16 groups and a maximum of eight providers per group. The provider authentication ordering method provides flexibility on what providers to use and what backups will be in place. The default authentication ports are configurable.

Default roles include AAA, admin, facility-manager, network, operations, read-only, server-equipment, server-profile, server-security, and storage. Additionally, roles can be customized by creating new roles and assigning privileges. The locales are used to define one or more organizations a user is allowed to access.

SSL key management – UI certificates and self-encrypting drives

Cisco UCS ships with a self-signed certificate using a default 1024-length key pair. To employ a more secure method, use trusted third-party certificates from a trusted source that affirms the identity of the Cisco UCS device.

Key management is also a core function for self-encrypting drives (SEDs). SED keys can be managed either locally or remotely with a third-party key management server such as CipherTrust. Local key management requires a security key (passphrase) to be entered into the system. Remote key management requires configuration of the Key Management Server (KMS) and the proper distribution of certificates and public and private keys.

Key and hash handling on the system

Symmetric keys

For Cisco UCS M5 and M6 platforms, the only root symmetric key is stored in the ACT2 Trust Anchor module. It cannot be read out from the ACT2. It may be used to encrypt other random symmetric encryption keys before being stored on the filesystem. No symmetric keys appear in the clear in any nonvolatile filesystem. These keys are only used internally and never sent anywhere.

For Cisco UCS M7 and M8 platforms, the CIMC has two AES vault keys randomly generated at manufacture and fused in the One-Time Programmable (OTP) memory. Software is not able to read these keys; only the hardware crypto module can access and utilize them. Cisco UCS also stores and uses a symmetric key in the TPM2; it cannot be read out. These keys are used to encrypt other symmetric keys before being stored to nonvolatile memory. Most of the nonvolatile filesystem is encrypted; only the CIMC images' partition is not, and that partition only contains signature-verified firmware images. On the Cisco UCS M8 platform, that partition is also encrypted. The keys are only used internally.

Asymmetric private keys

For Cisco UCS M5 and M7 platforms, whether ACT2 (TAm) or TPM2, the Cisco SUDI certificate is stored in the Trust Anchor module and cannot be read out.

For Cisco UCS M5 and M6 blade servers, the private key for a "KVM certificate" (external TLS communications) is not encrypted on the filesystem but is only transferred internally, to the CIMC. An attacker would need physical access to the system, to open the chassis, and to use hardware probes to read the traffic from flash memory. In Cisco Intersight mode, this key is transferred encrypted through the Device Connector (DC) on the system over TLS 1.3.

For Cisco UCS M5 rack servers, the private key for TLS is stored unencrypted (for internal transfer) and encrypted for session use.

For Cisco UCS M6 rack servers, the private key for TLS file is stored encrypted while the filesystem is unencrypted.

For Cisco UCS M7 blade servers, the private key for TLS communications is stored on an encrypted filesystem. UCSM sends the private key internally to CIMC in clear text.

For Cisco UCS M7 rack servers, the TLS private key is stored on an encrypted filesystem, and the file is also encrypted.

Password hash storage

For Cisco UCS M5 and M6 platforms, both AES and SHA are supported, based on password mode.

- Non-IPMI mode: SHA512 hash of password ((Salt + password) hashing occurs here.)
- IPMI mode: AES 128-bit CBC encryption of password

For Cisco UCS M7 and higher platforms, passwords are hashed using SHA512, and AES-256 is then encrypted in stored files with keys in the hardware trust anchor. The filesystem is also encrypted.

Secure configuration of UCSM-based systems

Deployment and management at scale

Cisco UCS can be deployed in three different ways, depending on your needs, infrastructure, and preferences. Cisco Unified Fabric deployments can take place with UCSM or with Intersight. Deployments without a fabric – that is, standalone – are handled using the baseboard management console.

UCSM

Cisco UCS Manager enables you to manage general and complex server deployments. Systems deployed and managed with UCSM are in UCSM Managed Mode (UMM). For example, you can manage a general deployment with a pair of Fabric Interconnects (FIs), which are the redundant server access layer that you get with the first chassis or rack mount. These can scale dramatically (see your hardware specifications sheet for the deployment limits for your server model. This can be a combination of blades and rack-mount servers to support the workload in your environment. As you add more servers, you can continue to perform server provisioning, device discovery, inventory, configuration, diagnostics, monitoring, fault detection, and auditing.

Service profiles and policies in Cisco UCS

A service profile is a software definition of a server and its LAN and SAN connectivity. A service profile defines a single server and its storage and networking characteristics. Service profiles are stored in Cisco UCS fabric interconnects and are managed through specific versions of Cisco UCS Manager (the web interface for the fabric interconnect) or through purpose-written software using the API. When a service profile is deployed to a server, Cisco UCS Manager automatically configures the server, adapters, fabric extenders, and fabric interconnects to match the configuration specified in the service profile. This automation of device configuration reduces the number of manual steps required to configure servers, Network Interface Cards (NICs), Host Bus Adapters (HBAs), and LAN and SAN switches.

Security enforcement through policy

In UCSM, the service profile is a central component that defines the compute, network, and storage characteristics for a server within the Cisco UCS infrastructure. In essence, it abstracts the physical hardware configuration from the logical configuration, enabling rapid provisioning, mobility, and scalability within the data-center environment.

The service profile contains:

- Hardware identity: this includes details such as WWPN (Worldwide Port Name) and WWNN (Worldwide Node Name) for Fibre Channel, MAC addresses for Ethernet, BIOS settings, and firmware versions.
- Boot policies: these define how the server boots, which operating system it uses, and where it boots from (local disk, SAN, LAN, etc.).
- Host firmware package: this specifies the firmware versions to be applied to the server's components, ensuring consistency and compliance.

UCSM provides various policies that help enforce a secure deployment posture within the service profiles:

- Unified port policies: define the configurations for Ethernet and Fibre Channel ports, enabling administrators to set specific security-related parameters such as VLAN settings, port channel settings, QoS (Quality of Service) policies, etc.
- Server BIOS policies: allow administrators to configure security-related settings in the server's BIOS, such as enabling or disabling specific hardware features, setting passwords, enabling secure boot, or configuring Trusted Platform Module (TPM) settings
- Local disk configuration policies: govern how local disks are configured, encrypted, or formatted, providing security measures for data stored on local disks
- Boot security policies: control boot-related security settings, including secure-boot configurations and control over boot devices
- Maintenance policies: define maintenance windows and other policies that can help enforce security-related updates or configurations during specified maintenance periods.
- Role-Based Access Control (RBAC): allows administrators to define roles and privileges, ensuring that only authorized personnel have access to critical UCSM functions and configurations, enhancing security by limiting access based on job responsibilities
- See IMM Security Policy Checklist and Intersight Help for recommended policy configuration settings.

For example, in Figure 11 below, in the UCSM UI navigation panel on the left, select the Server icon and then expand the policies menu item in the middle window. This will give you access to the wizards that allow you to complete configuration settings for each server related policies. There are also polices present in the other navigation items on the left. Navigate to the other icons and set policies for LAN, SAN, Storage, and Chassis as needed. See the IMM Security Policy Checklist and Intersight Help links in the reference section at the end of this paper for recommended policy configuration settings has a comprehensive list of policies and their default and recommended settings that are relevant to the security posture of the system. Only policies relating to the security of the system are covered in that policy reference.

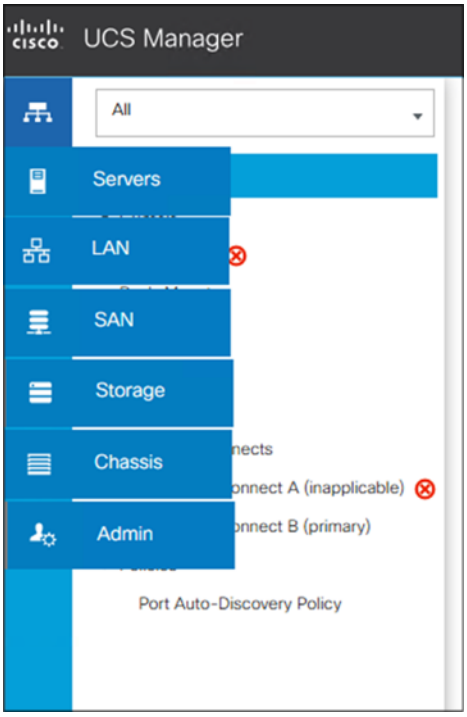

Figure 11.
UCSM Navigation icons. Policies can be set under Servers, LAN, SAN, Storage, and Chassis.

Figure 12 demonstrates the top-level policy menu for Server settings.

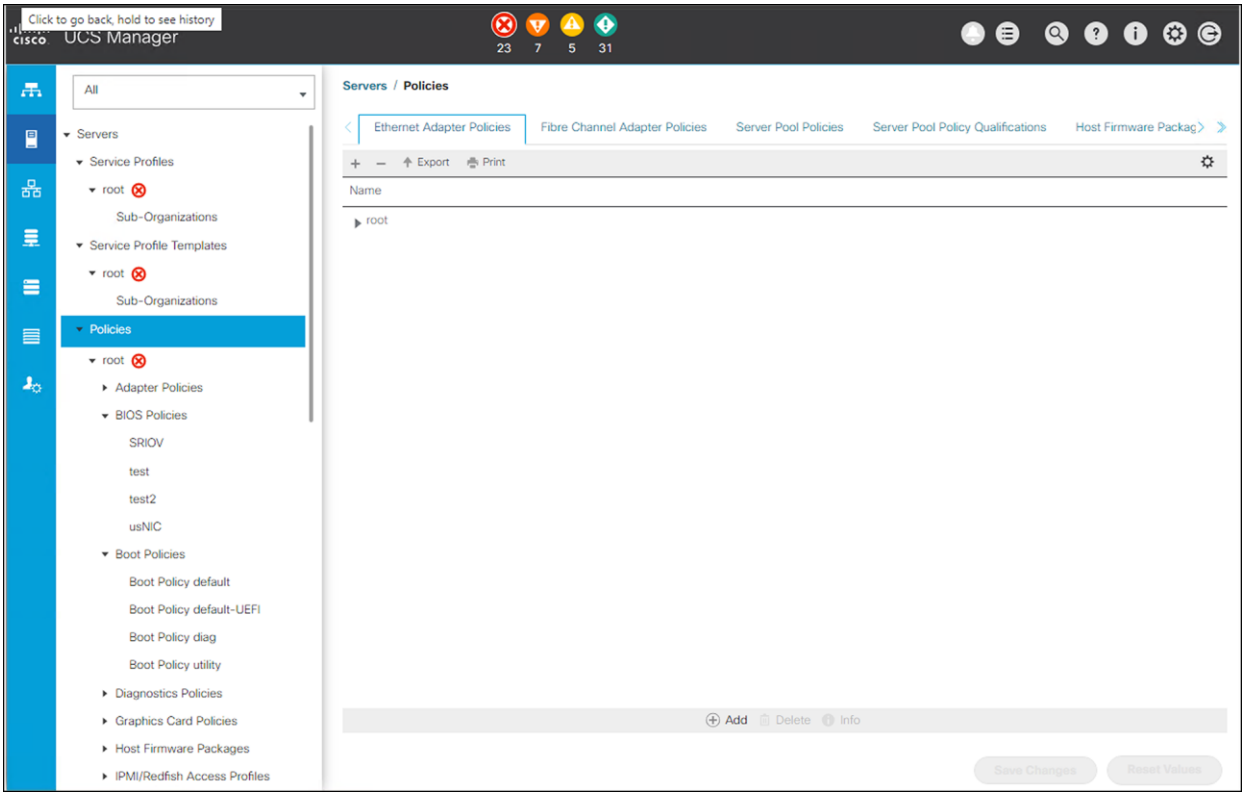

Figure 12.
Server policies top-level menu

Drilling down into the first BIOS policy under the Advanced tab, we can see some of the processor directives.

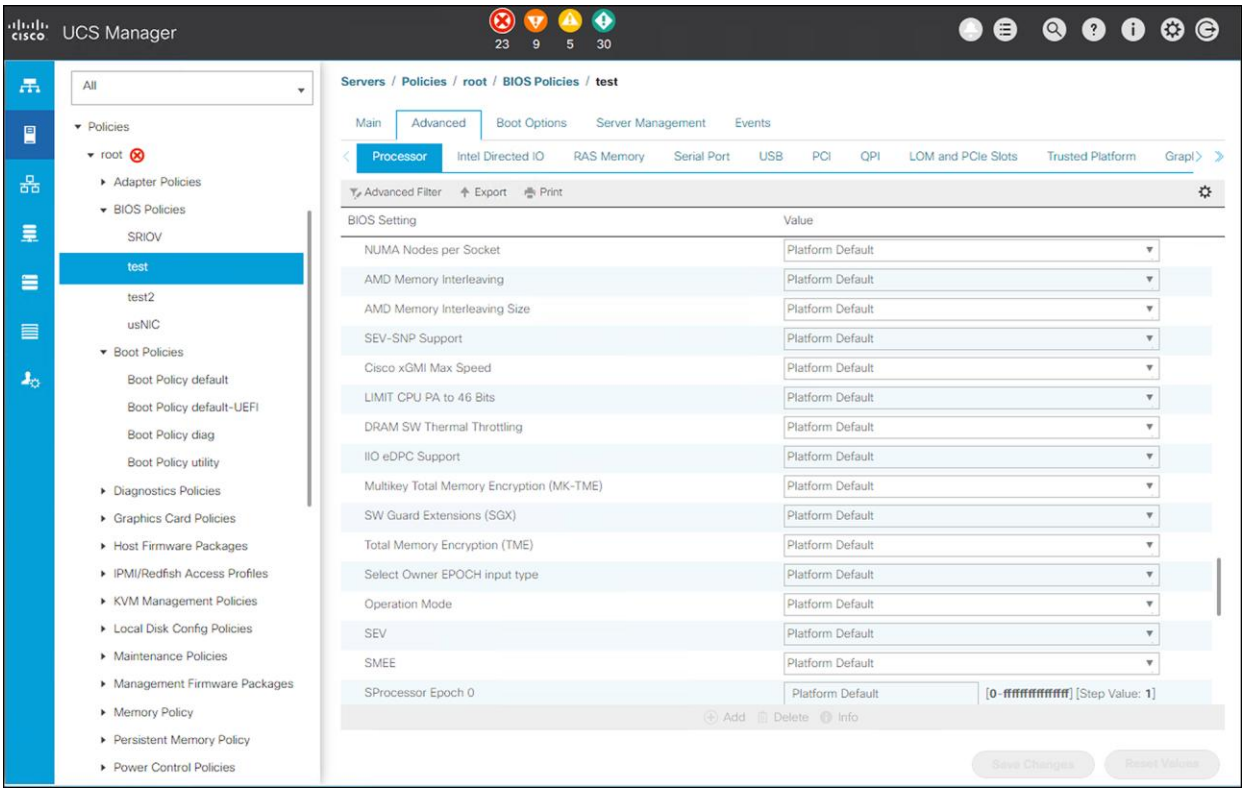

Figure 13.
Setting processor directives in the “test” BIOS policy

By utilizing these policies within UCSM when creating and managing service profiles, organizations can establish a more secure deployment posture, ensuring that servers are provisioned and configured according to predefined security best practices and policies. This helps in reducing potential vulnerabilities and maintaining a standardized and secure computing environment within the Cisco UCS infrastructure.

See IMM Security Policy Checklist and Intersight Help for recommended policy configuration settings.

UCSM XML-based API

The Cisco UCS Manager XML API is a programmatic interface to Cisco UCS. The API accepts XML documents through HTTP or HTTPS. Developers can use any programming language to generate XML documents that contain the API methods. Configuration and state information for Cisco UCS is stored in a hierarchical tree structure known as the management information tree, which is completely accessible through the XML API.

The Cisco UCS Manager XML API supports operations on a single object or an object hierarchy. An API call can initiate changes to attributes of one or more objects such as chassis, blades, adapters, policies, and other configurable components.

Authentication methods authenticate and maintain the session. For example:

- aaaLogin: initial method for logging in
- aaaRefresh: refreshes the current authentication cookie
- aaaLogout: exits the current session and deactivates the corresponding authentication cookie

Use the aaaLogin method to get a valid cookie. Use aaaRefresh to maintain the session and keep the cookie active. Use the aaaLogout method to terminate the session (this also invalidates the cookie). A maximum of 256 sessions to the Cisco UCS can be opened at any one time.

Administrative operations

In Figure 14 below, under the Admin icon in the navigation menu on the left, we have access to a host of settings related to managing logging, users, authentication, keys, and communication services.

The top-level menu for the Admin settings is shown below.

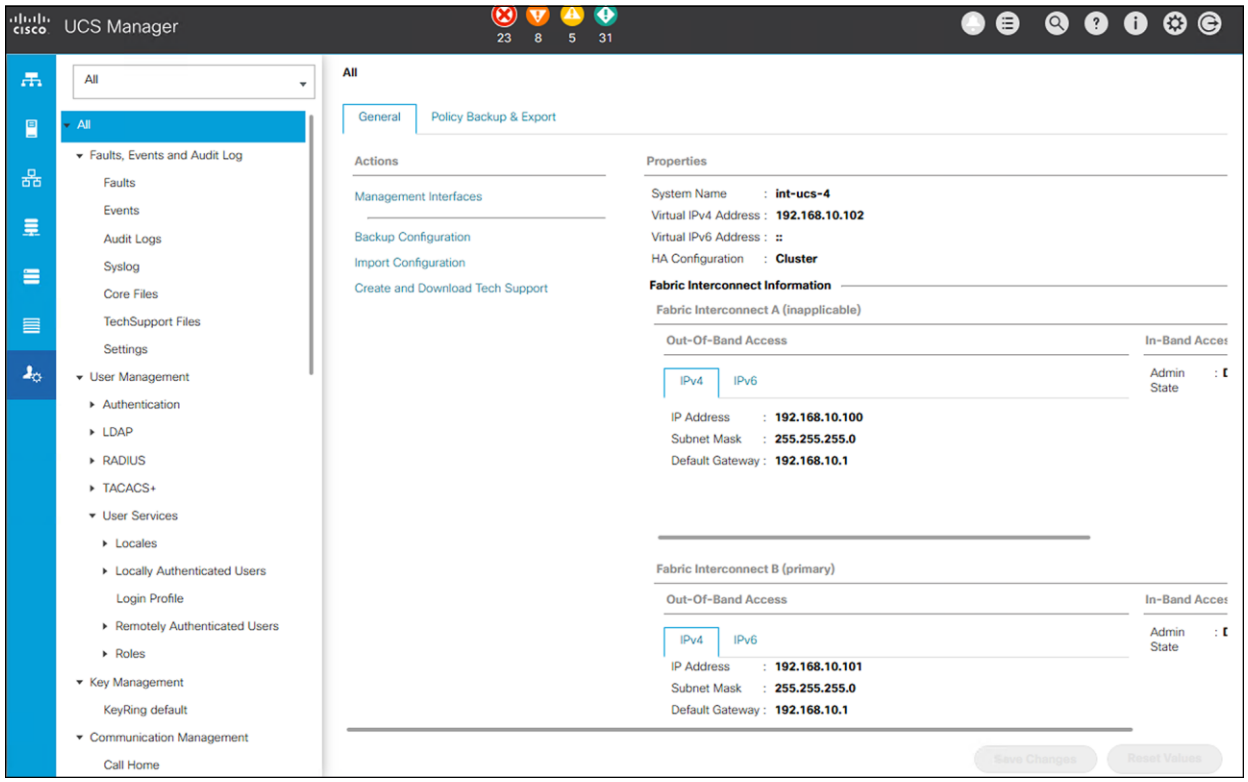

Figure 14.
Top-level administrative settings in UCSM

User management and AAA

User management in the Admin settings provides facilities to manage local users as well as externally authenticated users. Configure these settings as appropriate for your environment.

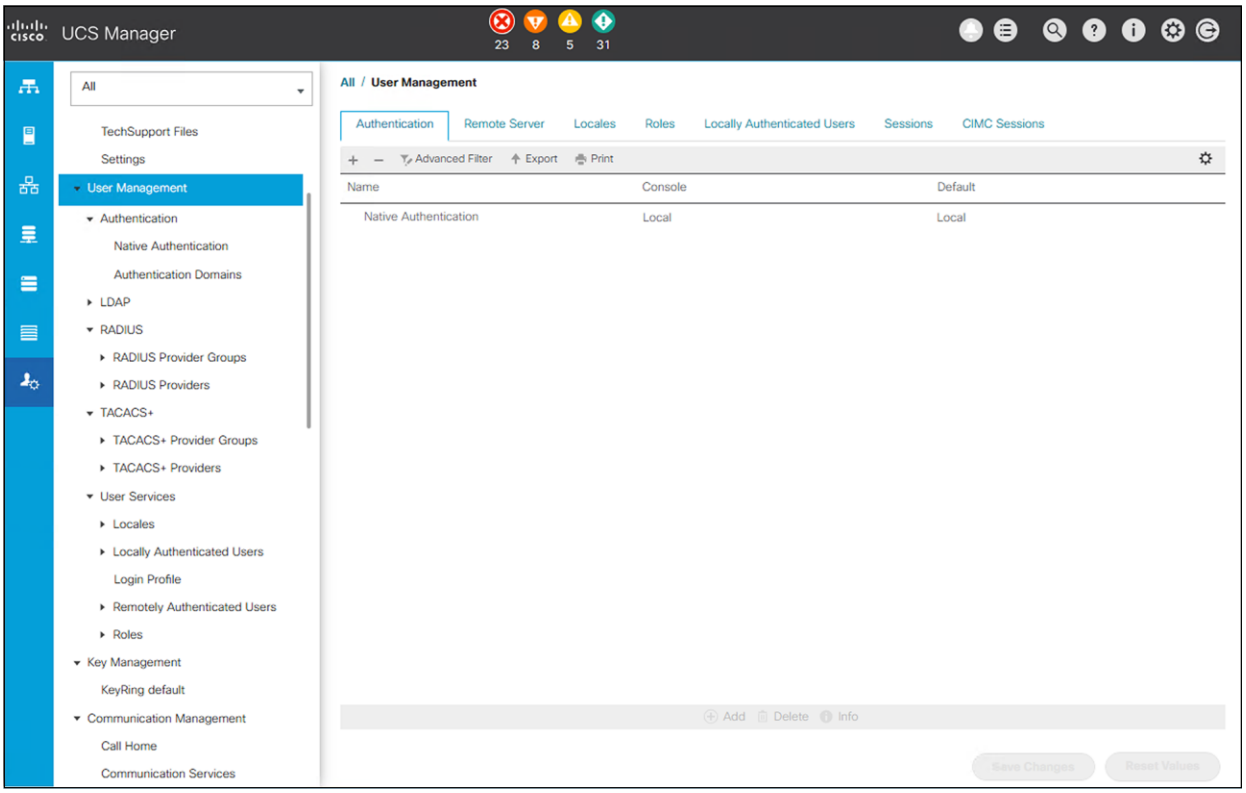

Figure 15.
User management in the administrative section of UCSM

Secure communication services

Traffic-type distinction

In-band management traffic means that the management traffic travels through a VLAN out through the uplink data ports on the FI. Ethernet-uplink ports generally have a much larger bandwidth than the dedicated management port. Out-of-band management traffic leverages the same IP space as the fabric-interconnect-management IP addresses and travels through the 1 Gbit copper ports on the fabric interconnects.

Communication settings

The secure communications services section in the Admin settings allows you to configure HTTP/HTTPS, set call-home addresses, manage UI and SSH sessions, set CIMC UI access, choose cipher suites, and set SNMP parameters.

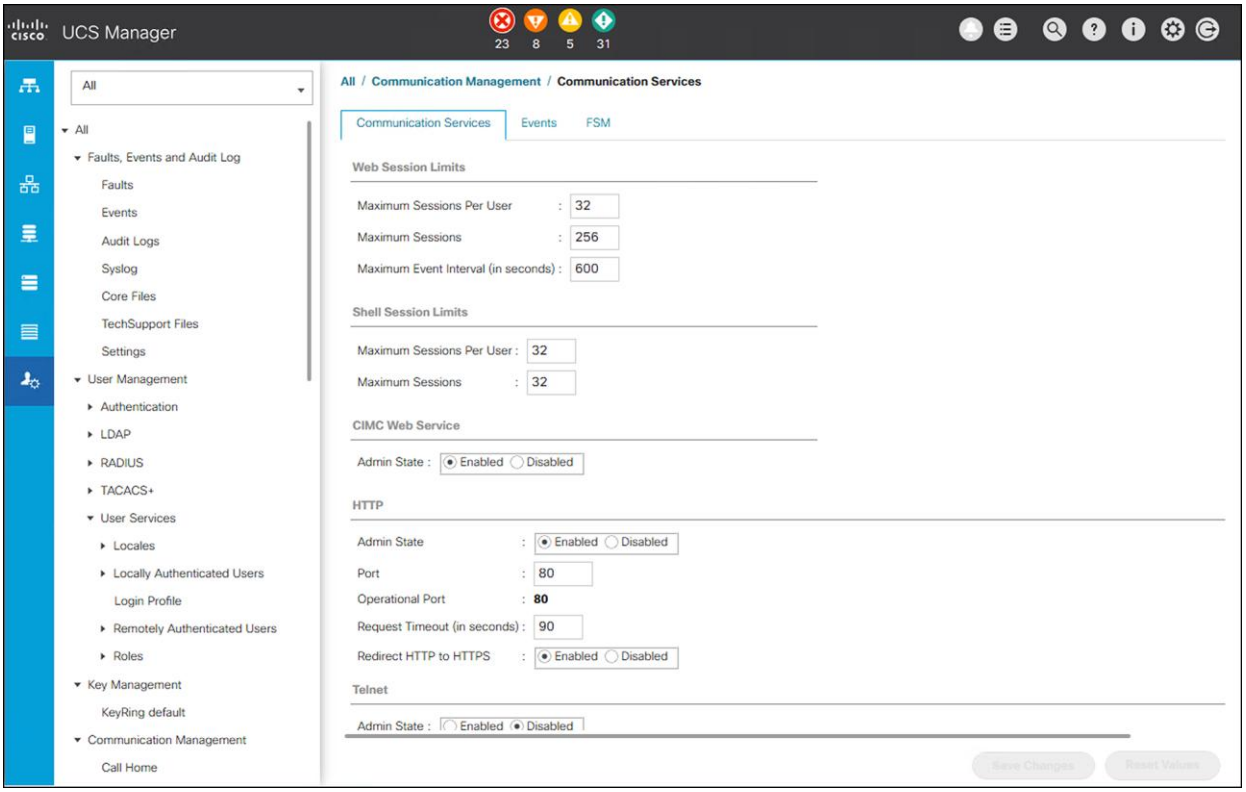

Figure 16.
Settings and adjustments that can be made for communication services in UCSM

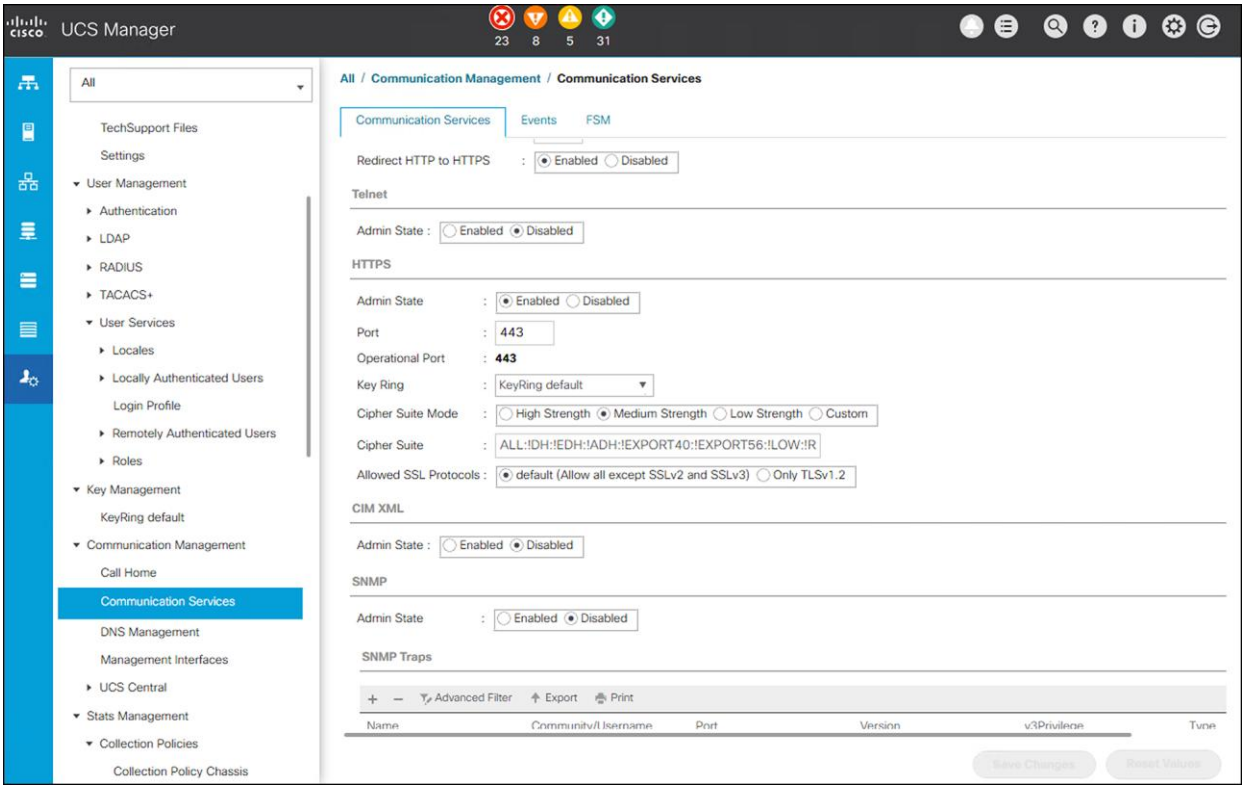

Figure 17.
Secure communication settings for HTTPS including cipher suite selection

SSH

SSH in UCSM cannot be disabled. See the following for additional details on communication services: https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/ucs-manager/GUI-User-Guides/Admin-Management/4-3/b_cisco_ucs_admin_mgmt_guide_4-3/m_ucs_manager_communication_services.html

SSH in CIMC can be disabled: https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/c/sw/cli/config/guide/b_Cisco_CIMC_CLI_Configuration_Guide/Cisco_CIMC_CLI_Configuration_Guide_chapter8.pdf

Note that this procedure only works for standalone Cisco UCS systems. FI attached servers that utilize UCSM cannot have SSH disabled on FI or CIMC.

About CiscoSSH

CiscoSSH is a common module, based on Linux and enhanced by Cisco, that is derived from OpenSSH (and pkix-ssh). It enables Cisco products to achieve FIPS compliance when used with the CiscoSSL FIPS Object Module (FOM) or a FIPS capable OpenSSL.

Most scanners that users run to scan for security vulnerabilities (CVEs) are not sophisticated enough to actually look at the backing code. Typically, security scanners use version information to alert users to possible security issues. Since CiscoSSH is a fork of multiple open-source upstream code bases, these can be inaccurate. OpenSSH, for example, does not branch. OpenSSH has a single mainline branch, and new releases are derived from there. This means security patches are applied to the main branch and released as a new version. OpenSSH does not backport CVE fixes or security patches. However, the CiscoSSH team does backport CVE patches when possible.

What does this mean for security scanners? It means the scanner's reports are often incorrect when based solely on a version of OpenSSH. Please validate the CVE information with Cisco to see when and if a particular security patch has been implemented and released. Also, please note the “fixed” version information, because it is likely to be different from what the security scanner reports.

Middlebox compatibility mode

During development of the TLSv1.3 standard, it became apparent that, in some cases, even if a client and server both support TLSv1.3, connections could sometimes still fail. This is because middleboxes on the network between the two peers do not understand the new protocol and prevent the connection from taking place. In order to work around this problem, the TLSv1.3 specification introduced “middlebox compatibility mode.” This made a few optional changes to the protocol to make it appear more like TLSv1.2 so that middleboxes would let it through.

Middlebox compatibility mode makes the TLSv1.3 handshake flow look more like a TLSv1.2 handshake. This is accomplished by filling in legacy fields in handshake messages and by sending a TLSv1.2 handshake message eliminated from the pure TLSv1.3 implementation. See the information on Middlebox compatibility mode here: Middlebox compatibility mode.

These changes are largely superficial in nature but do include sending some small but unnecessary messages. OpenSSL has middlebox compatibility mode on by default, so most users should not need to worry about this. However, applications may choose to switch it off by calling the function SSL_CTX_clear_options() and passing SSL_OP_ENABLE_MIDDLEBOX_COMPAT as an argument.

If the remote peer is not using middlebox compatibility mode and there are problematic middleboxes on the network path, then this could cause spurious connection failures.

OpenSSL/CiscoSSL supports middlebox compatibility, but UCSM and CIMC provide no means to disable it. Keeping this option enabled prevents communication problems with applications that do not use the TLS 1.3 protocol per se and rely on some of these TLS 1.2 fields to exist in order to function. This does not break TLS 1.3.

It is important to note that UCSM and CIMC do not use OpenSSL; they use CiscoSSL. Similar to CiscoSSH, the CiscoSSL module is developed by Cisco in-house, based on OpenSSL. See the “About CiscoSSH” section, above, for details.

UCSM UI session information

Here are some specifics regarding UI session information for UCSM:

- The session token/ID is removed on logout or when it reaches the end of validity.
- UCSM/HTTPD only recognizes and accepts the tokens maintained in its HTTPD process memory cache.
- UCSM uses Apache APR utils for session-token generation. APR_util generates the UUID by reading the system clock and adding the fudge factor.

Logging

The Faults, Events, and Audit Log section of the Admin settings allows configuration of logging and fault-notification parameters. This is also where you can set syslog information for remote consolidated logging.

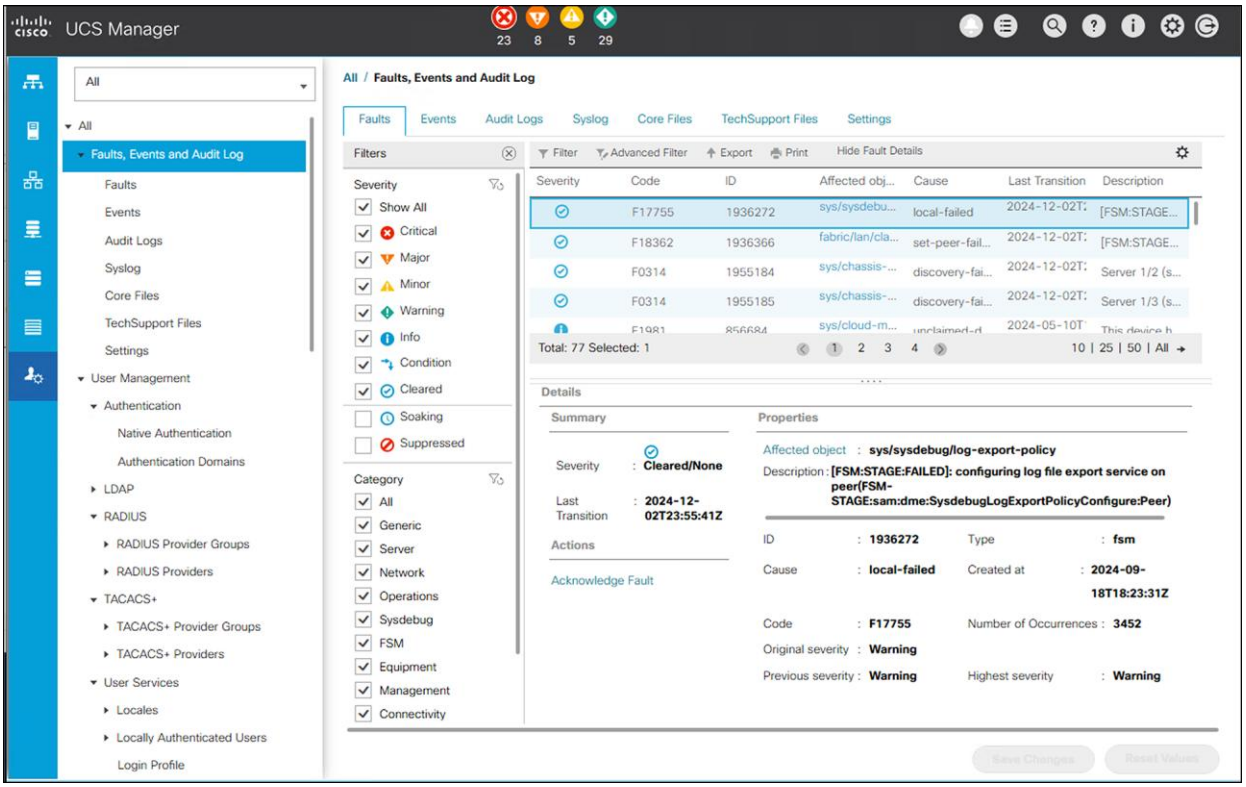

Figure 18.
Setting up logging parameters in UCSM from the administrative configuration section

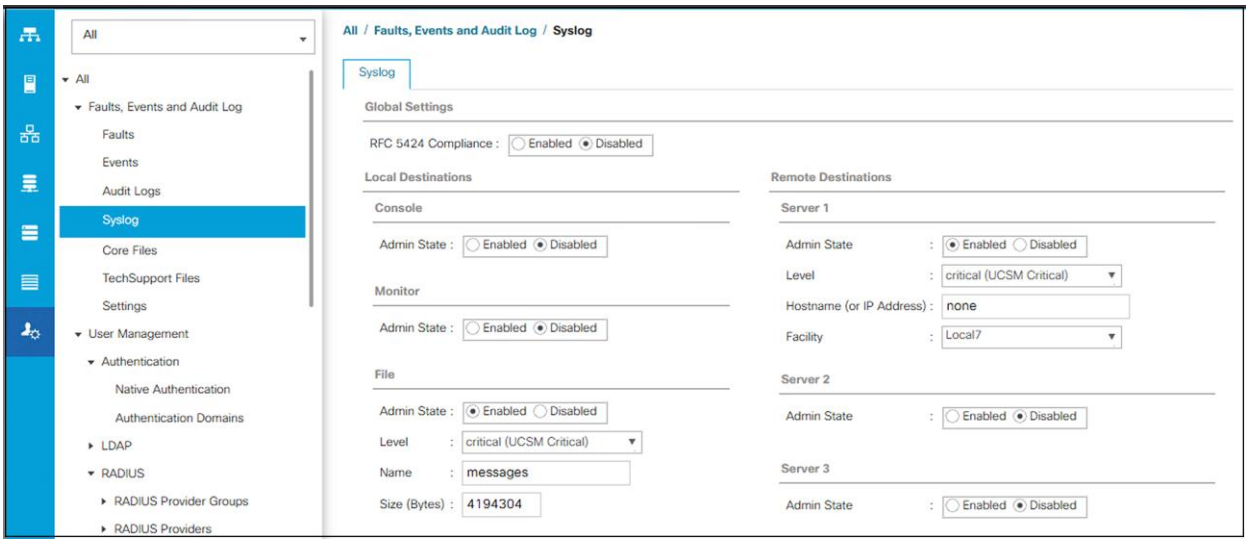

Figure 19.
Syslog settings, including remote server entries

Table 3 describes the types of faults UCSM will report against fault types in UCSM alerting and logging.

Table 3. Fault types in UCSM alerting and logging

Type	Description
FSM	An FSM task has failed to complete successfully, or the Cisco UCS Manager is retrying one of the stages of the FSM.
Equipment	The Cisco UCS Manager has detected that a physical component is inoperable or has another functional issue.
Server	The Cisco UCS Manager is unable to complete a server task, such as associating a service profile with a server.
Configuration	The Cisco UCS Manager is unable to successfully configure a component.
Environment	The Cisco UCS Manager has detected a power problem, thermal problem, voltage problem, or a loss of CMOS settings.
Management	The Cisco UCS Manager has detected a serious management issue, such as one of the following: • Critical services could not be started. • The primary switch could not be identified. • Components in the instance include incompatible firmware versions.
Connectivity	The Cisco UCS Manager has detected a connectivity problem, such as an unreachable adapter.
Network	The Cisco UCS Manager has detected a network issue, such as a link down.
Operational	Cisco UCS Manager has detected an operational problem, such as a log capacity issue or a failed server discovery.

Table 4 describes the fault severities for the various types of faults UCSM will report on.

Table 4. Severity types in UCSM alerting and logging

Severity	Description
Cleared	A notification that the condition that caused the fault has been resolved, and the fault has been cleared.
Condition	An informational message about a condition, possibly independently insignificant.
Critical	A service-affecting condition that requires immediate corrective action. For example, this severity could indicate that the managed object is out of service and its capability must be restored.
Info	A basic notification or informational message, possibly independently insignificant.
Major	A service-affecting condition that requires urgent corrective action. For example, this severity could indicate a severedegradation in the capability of the managed object and that its full capability must be restored.
Minor	A non-service-affecting fauh condition that requires corrective action to prevent a more serious fault from occurring. For example, this severity could indicate that the detected alarm condition is not currently degrading the capacity of the managed object.
Warning	A potential or impending service-affecting fault that currently has no significant effects in the system. Action should be taken to further diagnose, if necessary, and correct the problem to prevent it from becoming a more serious service-affecting fault.

Table 5 lists the fields present in the faults. These fields will be reflected in the audit and system event logs.

Table 5. Attributes in UCSM for alerting and logging.

Attribute	Description
Fault Instance ID (Table Index)	A unique integer that identifies the fault
Affected Object DN	The distinguished name of the mutable object that has the fault.
Affected Object O1D	The Object identifier (OID) of the mutable object that has the fault.
Creation Time	The time that the fault was created.
Last Modification	The time when any of the attributes were modified.
Code	A code that provide.s infonuationspecific to the nature of the fault.
Type	The fault type.

Attribute	Description
Cause	The probable cause of the fault.
Severity	The severity of the fault.
Occurrence	The number of times that a fault ha'i occurred since it was created.
Description	A human readable stringthatcontains all information related to the fault.

Audit records

To gain an understanding of existing, emerging, and historic events that are related to security incidents, an organization should have a unified strategy for event logging and correlation. This strategy must leverage logging from all network devices and use prepackaged and customizable correlation capabilities. Cisco UCS has a syslog capability that allows aggregation of logs at a centralized log server.

After centralized logging is implemented, a structured approach must be developed to analyze logs and track incidents. Based on the needs of the organization, this approach can range from a simple diligent review of log data to an advanced rule-based analysis.

The Cisco UCS audit log has a maximum of 10,000 entries. It utilizes a circular, FIFO logging design, so the oldest entries will drop off as new entries are created. If you are at the 10,000-entry limit, you can rotate the logs on your centralized server, or you can export the log file as .csv from the GUI; another alternative is to issue commands through PowerShell to pull logs.

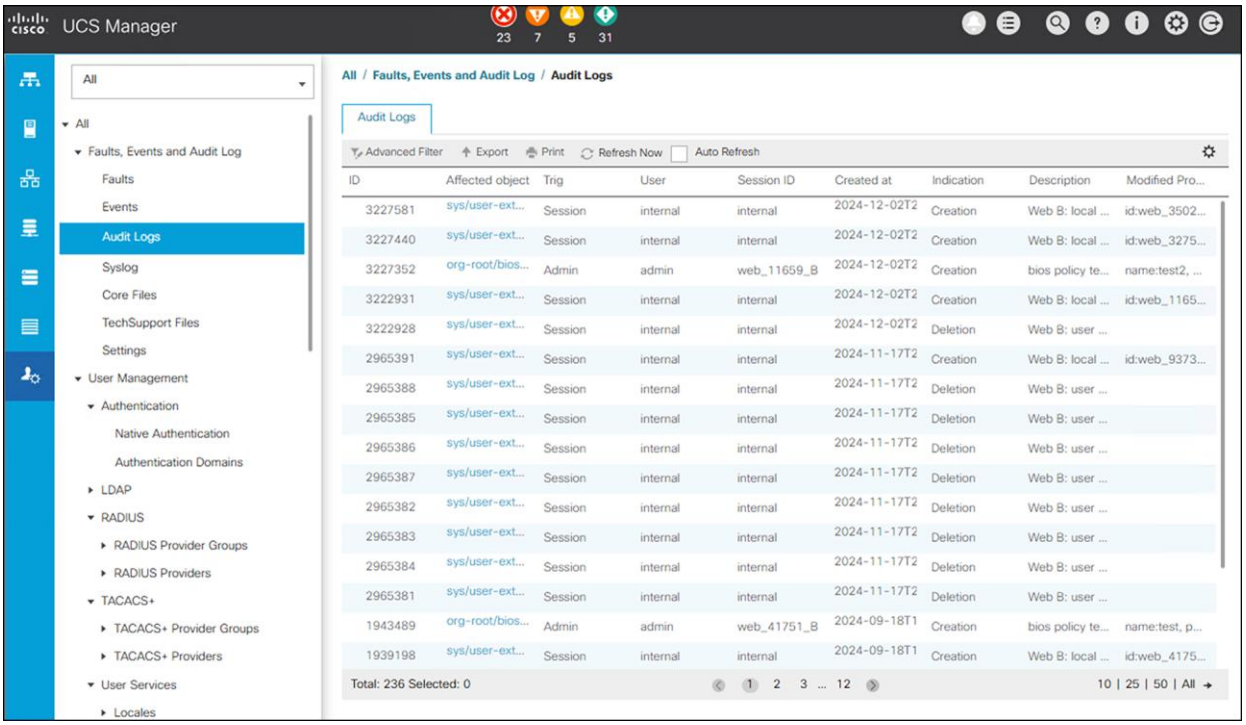

Figure 20.
Audit records in UCSM

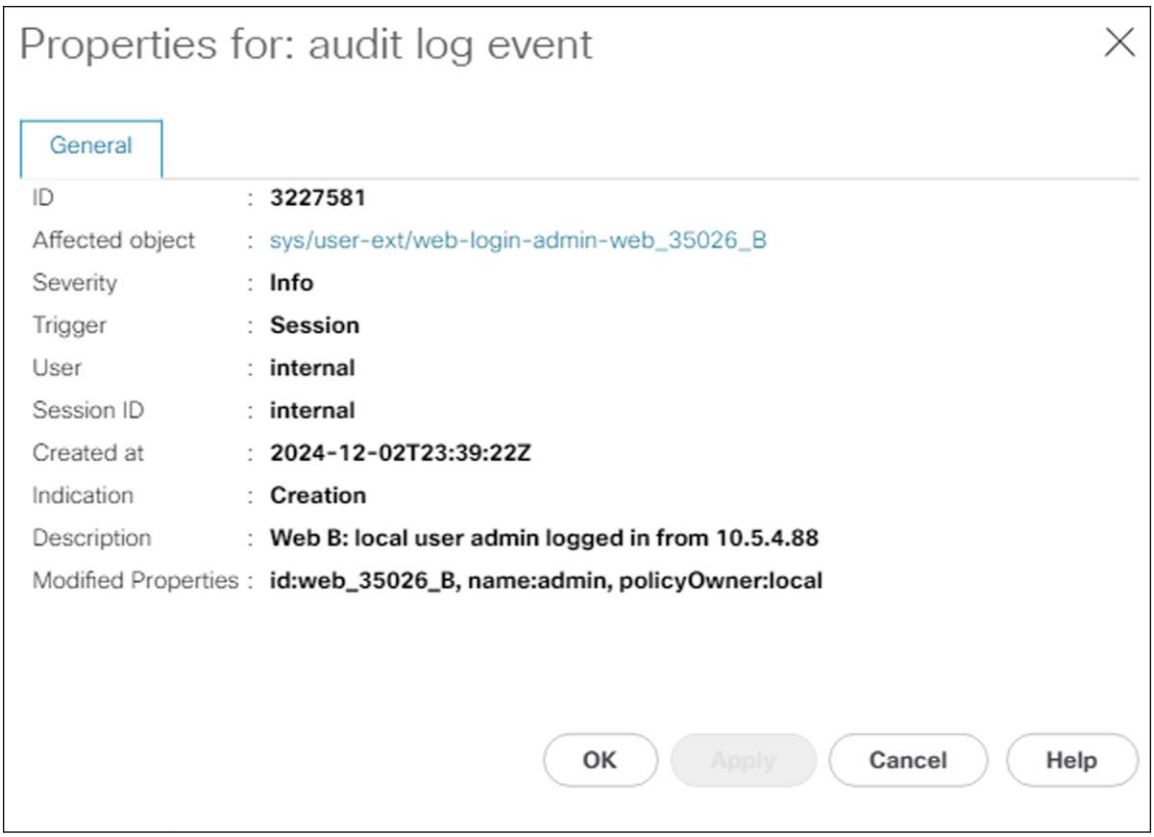

Figure 21.
A typical audit log event entry

Tech Support File

Generating a Tech Support File is important for Cisco® TAC troubleshooting and for general log mining. A Tech Support File will contain all the system logs as well as all the diagnostic and event information for the system. It is critical for identifying and solving any system issues but is also invaluable as a forensic security tool. These logs can be used in conjunction with Splunk® analysis software to further an investigation or to offer additional insights into the system.

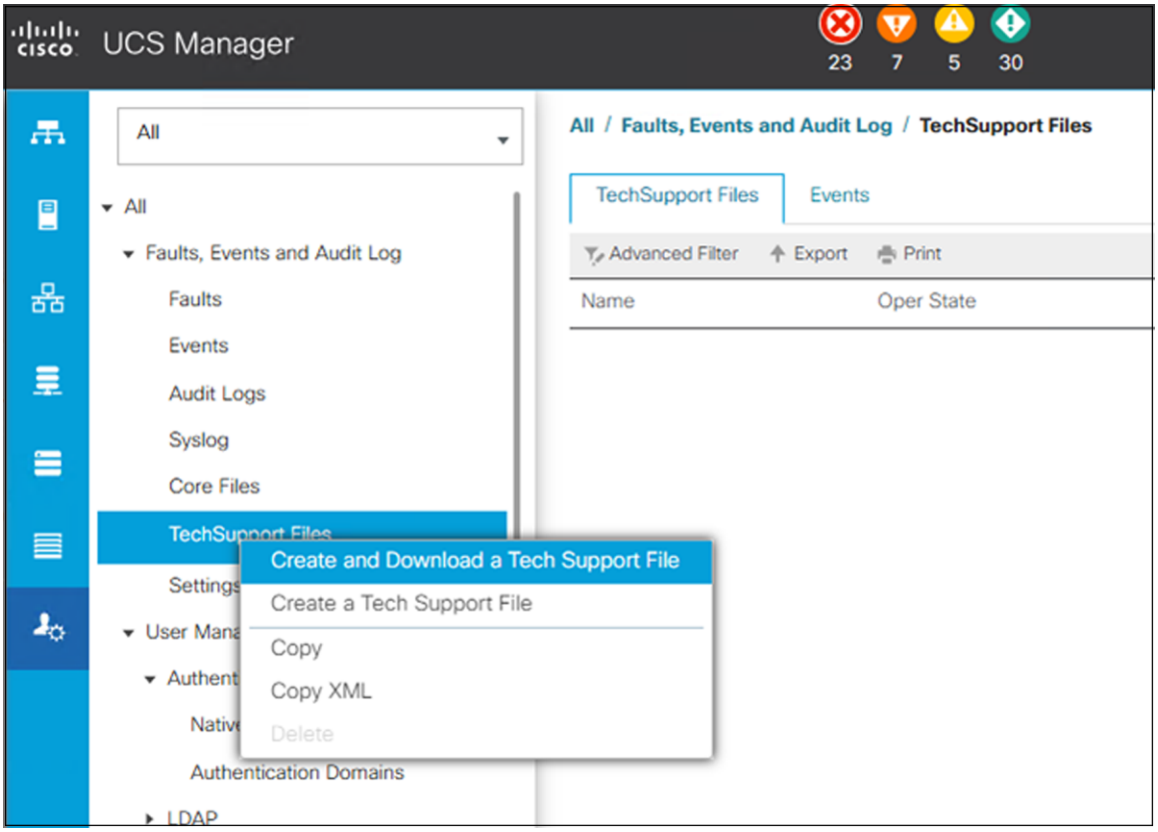

Figure 22.
Creating a Technical Support File in the admin section of UCSM

Monitoring

Server system monitoring is crucial for security because it provides real-time visibility into the performance, health, and activities of servers within an IT infrastructure. Monitoring helps identify and respond to potential security threats, vulnerabilities, and irregularities, contributing to a proactive and effective security posture. Effective monitoring provides the following:

- Early detection of anomalies
  - Detect abnormal patterns or behaviors on servers, which may indicate a security incident. Early detection allows for a quicker response to potential threats
- Identification of security incidents
  - Identify security incidents such as unauthorized access, malware infections, or suspicious network activities

- Visibility into system health
  - Insights into the overall health and performance of servers. A sudden drop in performance or unexpected system behavior may indicate a security compromise or the presence of malicious activities.
- Alerts and notifications
  - Generate alerts and notifications when predefined thresholds or security policies are breached
- Resource utilization monitoring
  - Unusual spikes in CPU, memory, or network usage may indicate a security incident, such as a denial-of-service attack or a compromised system engaging in malicious activities
- Log analysis for security events
  - Analyze server logs for security-related events. This includes authentication attempts, access logs, and error messages that may reveal signs of unauthorized access or other security incidents.
- User-activity monitoring
  - Monitoring user activities on servers helps in detecting suspicious behavior, such as unauthorized access or privileged users performing unexpected actions.
- Compliance and auditing
  - Server monitoring helps provide the necessary data for audits. It verifies that security policies are enforced and that systems are in compliance with industry or organizational standards.
- Incident response and forensics:
  - In the event of a security incident, monitoring data serves as valuable forensic evidence. It helps security teams understand the nature of the incident, trace its origins, and implement corrective measures to prevent future occurrences.
- Patch and update management
  - Monitoring systems can track the status of server patching and updates. Ensuring that servers are up to date on security patches is essential for protecting against known vulnerabilities.
- Capacity planning for security resilience
  - Monitoring assists in capacity planning, allowing organizations to anticipate resource demands and ensuring that servers are equipped to handle security-related loads, such as increased traffic during a DDoS attack.

By actively monitoring server systems, organizations can enhance their capability to prevent, detect, and respond to security threats effectively. It is a foundational element of a comprehensive security strategy, providing the insights needed to maintain a secure and resilient IT environment.

Cisco UCS Manager monitoring background

The core of Cisco UCS Manager is made up of three core elements: the Data Management Engine (DME), Application Gateway (AG), and user-accessible northbound interfaces (SNMP, Syslog, XMLAPI, and UCS CLI). With Cisco UCS Manager, there are three main ways of monitoring UCS servers: XML API, SNMP, and Syslog. Both SNMP and syslog are interfaces only used for monitoring because they are “read-only,” not allowing an end user to change the configuration. Alternatively, UCS XML API monitoring is “read-write,” which allows an end user to both monitor UCS and change the configuration if needed.

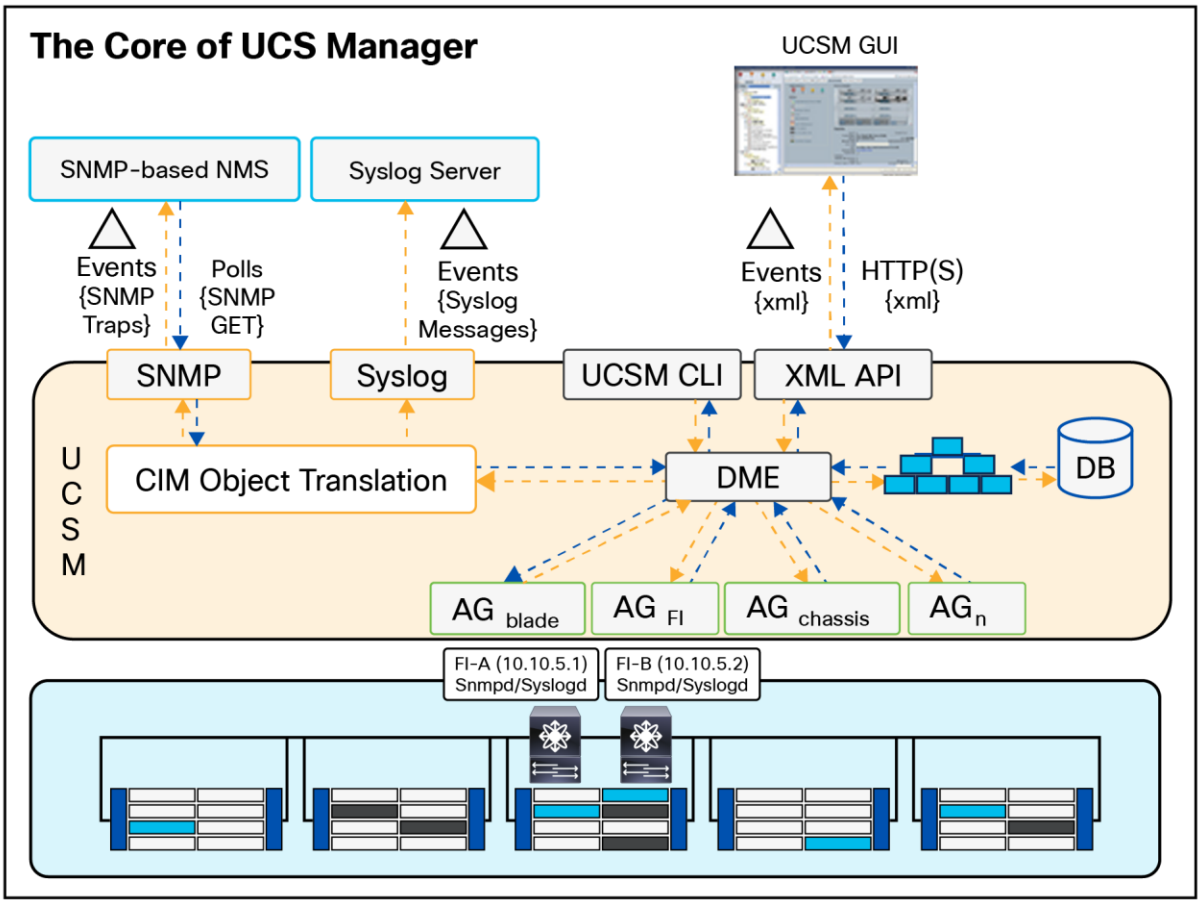

Figure 23.
UCS Manager monitoring architecture

- **Data Management Engine (DME):** the DME is the center of the UCS Manager universe, or the "queen bee" of the entire system. It is the maintainer of the UCS XML database which houses the inventory database of all physical elements (blade / rackmount servers, chassis, I/O modules, fabric interconnects, etc.), the logical configuration data for profiles, policies, pools, vNIC and/or vHBA templates, and various networking-related configuration details (VLANs, VSANs, port channels, network uplinks, server downlinks, etc.). It maintains the current health and state of all components of all physical and logical elements in a UCS domain and maintains the transition information of all Finite State Machine (FSM) tasks occurring.
- The inventory, health, and configuration data of managed endpoints stored in the UCS XML database always show current data, delivered in near real time. As fault conditions are raised and cleared on endpoints, the DME will create, clear, and remove faults in the UCS XML database as those fault conditions are raised or mitigated. The faults stored in the UCS XML database are only the ones actively occurring, because the DME, by default, does not store a historical log of all faults that have occurred in a UCS domain.
- **Application Gateway (AG):** AGs are the software agents, or "worker bees," that communicate directly with the endpoints to provide the health and state of the endpoints to the DME. AGs manage configuration changes from the current state to the desired state during FSM transitions when changes are made to the UCS XML database. AG-managed endpoints include servers, chassis, I/O modules, fabric extenders, fabric interconnects, and Cisco NX-OS. A server's AGs actively monitor the server through the IPMI and SEL logs through the Cisco Integrated Management Controller (CIMC), to provide the DME with the health, state, configuration, and potential fault conditions of a device. The I/O module's AG and chassis AG communicate with the Chassis Management Controller (CMC) to get information about the health, state, configuration, and fault conditions visible to the CMC. The fabric interconnect and/or NX-OS AG communicates directly with NX-OS to get information about the health, state, configuration, statistics, and fault conditions visible by NX-OS on the fabric interconnects. All AGs provide the inventory details to DME about endpoints during the various discovery processes. AGs perform the state changes necessary to configure an endpoint during FSM-triggered transitions, monitor the health and state of the endpoints, and notify the DME of any faults or conditions.
- **Northbound interfaces:** the northbound interfaces include SNMP, Syslog, CLI, and XML API. The XML API present in the Apache webserver layer are used to send login, logout, query, and configuration requests through HTTP or HTTPS. SNMP and Syslog are both consumers of data from the DME. SNMP inform requests and traps are translated directly from the fault information stored in the UCS XML database. Inversely, SNMP GET requests are sent through the same object translation engine in reverse, where the DME receives a request from the object translation engine and the data is translated from XML data from the DME to a SNMP response. Syslog messages use the same object translation engine as SNMP, where the source of the data (faults, events, audit logs) is translated from XML into a UCS Manager-formatted syslog message.

Monitoring with UCSM

Cisco UCS Manager can detect system faults: critical, major, minor, and warnings. Cisco recommends that you monitor all faults of either critical or major severity status, because immediate action is not required for minor faults and warnings.

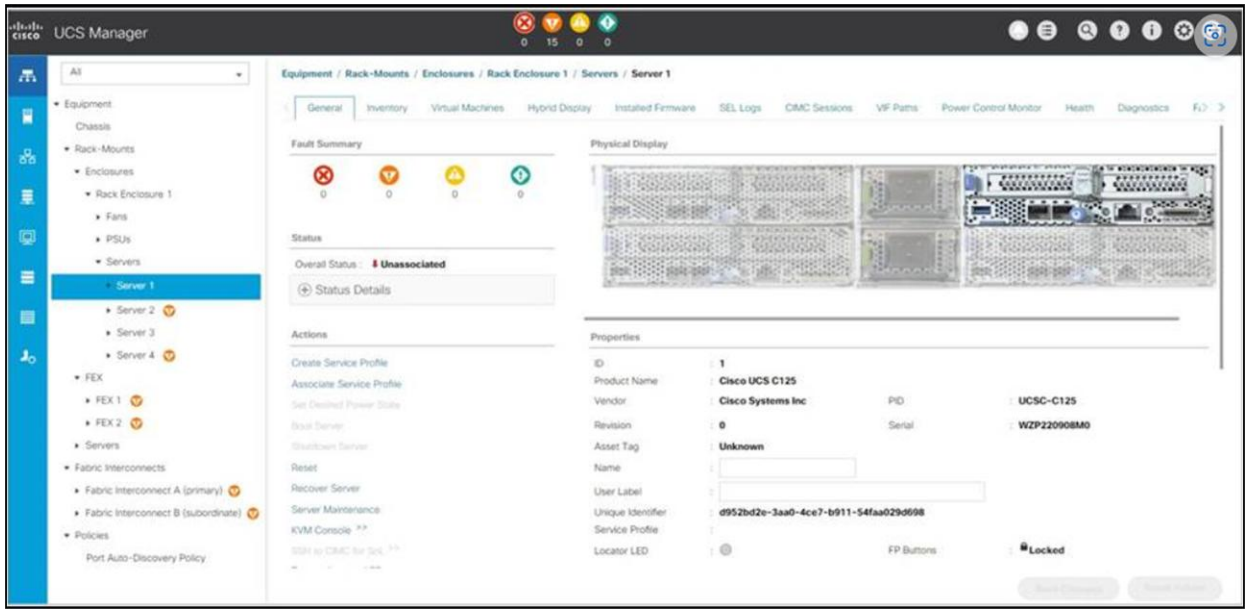

Figure 24.
UCSM fault and system summary

Logging is critical for investigation, audit, forensic analysis, and insight into overall system behavior and performance. The following logs are maintained by UCSM and can be examined at any time.

- System log
  - System logs including faults, failures, and alarm thresholds (Syslog)
  - The three types of Syslogs: Fault, Event, and Audit logs
  - The Global Fault Policy and settings that control Syslogs
- System event log
  - System hardware events for servers and chassis components and their internal components (System Event Log [SEL] logs)
  - The SEL policy that controls SEL logs
- Simple Network Management Protocol
  - SNMP for monitoring devices from a central network management station and the host and user settings
  - Fault suppression policies for SNMP traps, Call Home notifications, and specific devices

Monitoring with UCSM also includes:

- Statistics collection and threshold policies for adapters, chassis, host, ports, and servers
- Cisco Call Home and Smart Call Home–embedded device support
- Hardware monitoring using the UCS Manager user interface
- Cisco NetFlow Monitor for IP network traffic accounting, usage-based network billing, network planning, security, denial-of-service monitoring capabilities, and network monitoring

UCS Manager monitoring best practices

The recommendation for monitoring a UCS Manager environment would be to monitor all faults of either critical or major severity and that are not of type “FSM.” FSM-related faults are transient in nature, because they are triggered when an FSM transition occurs in UCS Manager. Generally speaking, FSM-related faults will resolve themselves automatically because most are triggered after a task fails the first time but will be successful on a subsequent attempt. An example of an FSM task failure would be when an FSM task waiting for a server to finish BIOS POST fails during a service profile association. This condition can happen when a server with many memory DIMMs takes longer to successfully finish POST than the default timeout of the FSM task. This timeout would raise an FSM fault on this task, but by default would keep retrying up to the defined FSM task retry limit. If a subsequent retry is successful, the FSM task fault raised will be cleared and removed. However, if subsequent retries are unsuccessful and the retry limit is reached, the FSM task will be faulted and another fault will be raised against the affected object. In this example, a configuration failure would be raised against the service profile, by reason that the association process failed because the server did not perform a successful BIOS POST.

If you are looking for a list of the most critical fault codes to monitor, refer to the “Syslog Messages to Monitor” section in Chapter 3 of the “Monitoring UCS Manager with Syslog” (see the link to this document under “Additional Cisco Monitoring Resources” at the end of this guide). The fault codes listed are the same codes for all interfaces (SNMP, Syslog, or XML API).

Securely decommissioning a system

Securely decommissioning a server is a critical process to ensure that sensitive data is properly handled and the server is retired in a way that minimizes the risk of data breaches or unauthorized access. Failing to decommission a server securely can lead to data exposure, legal and regulatory issues, and potential harm to an organization's reputation. Below are the levels of importance and the methods for securely decommissioning a server and its data.

Decommissioning a system or components of a system–specifically, the drives–requires special considerations in many circumstances. It is not sufficient to simply remove a drive or rotate a system out of production without sanitization. For older versions of Cisco UCS firmware, there are third-party applications that will run NIST-approved sanitization routines on plain-text drives or encrypted drives. The Commission Regulation (EU) 2019/424 requires that data be securely disposed of. Secure data disposal is accomplished by using commonly available tools that erase the data from the various drives, memory, and storage in Cisco UCS servers and reset them to factory settings. You must be familiar with what devices are present in your UCS server and run the appropriate tools for secure data deletion. In some cases, you may need to run multiple tools.

Beginning in early 2024, Cisco UCS firmware supports disk and system sanitization. This can more appropriately be termed data sanitization. Cisco IMC supports this NIST 800-88-compliant data-sanitization feature. Using the data-sanitization process, Cisco IMC erases all sensitive data, thus making extraction or recovery of user data impossible. As Cisco IMC progresses through the erase process, the status report is updated. You can check the status and progress of the data-sanitization process for each individual device erased from the report. Cisco IMC reboots when the data-sanitization process is completed and generates a report.

The erase process for data sanitization is performed in the following order on the server components:

1. Storage
2. VIC
3. BIOS
4. Cisco IMC

You can choose to either perform data sanitization on all the server components or select only storage and VIC components for data sanitization.

Proper decommissioning reduces the risk of data breaches. If data is not securely wiped, deleted, or destroyed, it may be accessible to unauthorized individuals who can exploit it for malicious purposes.

Secure decommissioning is essential for compliance with data-protection and privacy regulations. Many regulations, such as GDPR or HIPAA, require organizations to safeguard data even during disposal. It also serves to protect an organization's intellectual property.

Additional procedural steps can be taken as well. These include physical destruction and environmentally responsible recycling of components where appropriate. If this is not possible because the systems will be reused, other things can be done such as disabling and removing all user accounts and resetting server configurations.

Secure decommissioning is a comprehensive process that involves technical, procedural, and organizational measures. By following these methods, organizations can minimize the risks associated with retiring servers and ensure that sensitive data is handled responsibly and securely.

Server Secure Erase

Server Secure Erase (accessed via Redfish API) allows you to initiate a workflow to delete all data from the server. This includes data on the BIOS, Baseboard Management Controller (BMC), nonvolatile RAM (NVRAM), Dual In-line Memory Modules (DIMMs), embedded Multi-Media Cards (eMMCs), Virtual Interface Cards (VIC), and storage-disk components (except remote logical unit numbers [LUNs]). During this time, disruptive server actions, such as power actions, OS installation, firmware upgrades, decommissioning, and server-profile deployment are disabled.

Secure Erase is supported on Cisco UCS Series M5 and later servers. This action adheres to EU Lot 9 regulations for data storage devices and meets NIST SP 800-88 standards for data sanitization.

Depending on the disk capacity, Secure Erase may take several hours to more than a day to complete. See the table in the Intersight Help Center to get the estimated time required to complete a Secure Erase operation.

The following limitations are associated with Secure Erase:

- Secure Erase on a VIC cannot be performed from Cisco Integrated Management Controller (CIMC) on B-Series servers.
- The M6 eMMC does not comply with the data sanitization requirements of EU Lot 9 Regulation.
- Secure Erase is unable to completely erase FlexUtil in C-Series servers and Cisco FlexFlash partitions, including the hypervisor, on B-Series servers.
- Secure Erase may fail due to faulty components such as NVDIMMs and disks. You can remove the faulty component and retry.
- Standalone servers lose their network configuration and Intersight connectivity after a Secure Erase operation.

Scrub

Scrub is typically used in server decommissions that will result in reuse of the system. It is distinct from the data sanitization in Secure Erase for servers discussed above. This policy determines what happens to local data and to the BIOS settings on a server during the discovery process, when the server is reacknowledged, or when the server is disassociated from a service profile.

Local disk-scrub policies apply only to hard drives that are managed by Cisco Intersight and do not apply to other devices such as USB drives.

Depending upon how you configure a scrub policy, the following can occur at those times:

Disk scrub

One of the following occurs to the data on any local drives on disassociation:

- If disk scrub is enabled, disassociation deletes the initial 200 MB of data from the master boot record or boot sectors, thus preventing the system from booting from an already installed OS, if any. For secure deletion of data on drives, refer to the UCS Secure Data Deletion for Commission Regulation (EU) 2019/424 Users Guide.
- If disk scrub is disabled (default), disassociation preserves all data on any local drives, including local storage configuration.

For a server associated with a service profile, disk scrub occurs during disassociation, based on the scrub policy used in the service profile. For an unassociated server, disk scrub occurs during the server discovery process, based on the default scrub policy.

BIOS scrub

One of the following occurs to the BIOS settings when a service profile containing a scrub policy is disassociated from a server:

- If BIOS scrub is enabled, disassociation erases all BIOS settings for the server and resets them to the BIOS defaults for that server type and vendor.
- If BIOS scrub is disabled (default), disassociation preserves the existing BIOS settings on the server.

Secure operation of applications

Next, we will examine the secure Cisco UCS posture environment for operating applications. It is crucial that the system be able to ensure that applications can run in a fenced and protected compute regime. To this end, confidential computing is used, in hardware, to ensure isolated, encrypted, and protected execution.

Confidential computing

Confidential computing is a cloud-computing technology that isolates sensitive data in a protected CPU enclave during processing. The contents of the enclave—the data being processed, and the techniques that are used to process it—are accessible only to authorized programming code and are invisible and unknowable to anything or anyone else, including the cloud provider.

A confidential-computing secure enclave refers to a protected and isolated environment within a computing system where sensitive data or operations can be securely processed or stored. This secure enclave ensures that the data within it is protected from unauthorized access, even from other parts of the system or privileged software layers.

The concept of secure enclaves is primarily focused on maintaining the confidentiality, integrity, and privacy of sensitive information, especially when dealing with critical data or executing sensitive operations. These enclaves use hardware-based security mechanisms to create isolated and trusted spaces within the system's memory or processing units, offering a high level of protection against various types of attacks, including those attempting to access or manipulate the enclave's contents.

As company leaders rely more and more on public and hybrid-cloud services, data privacy in the cloud is imperative. The primary goal of confidential computing is to provide greater assurance to leaders that their data in the cloud is protected and confidential, and to encourage them to move more of their sensitive data and computing workloads to public cloud services.

For years, cloud providers have offered encryption services to help protect data at rest (in storage and databases) and data in transit (moving over a network connection). Confidential computing eliminates the remaining data security vulnerability by protecting data in use—that is, during processing or runtime.

How confidential computing works

Applications process data, and to do this, they interface with a computer’s memory. Before an application can process (encrypted) data, it must go through decryption in memory. Because the data is, for a moment, unencrypted, it is left exposed. It can be accessed, encryption-free, right before, during, and right after it has been processed. This leaves it exposed to threats such as memory dump attacks, which involve capturing and using Random Access Memory (RAM) that has been placed on a storage drive in the event of an unrecoverable error.

The attacker triggers this error as part of the attack, forcing the data to be exposed. Data is also exposed to root user compromises, which occur when the wrong person gains access to admin privileges and can therefore access data before, during, and after it has been processed.

Confidential computing fixes this issue by using a hardware-based architecture referred to as a Trusted Execution Environment (TEE). This is a secure coprocessor inside a CPU. Embedded encryption keys are used to secure the TEE. To make sure the TEEs are only accessible to the application code authorized for it, the coprocessor uses attestation mechanisms that are embedded within. If the system comes under attack by malware or unauthorized code as it tries to access the encryption keys, the TEE will deny the attempt at access and cancel the computation.

This allows sensitive data to stay protected while in memory. When the application tells the TEE to decrypt it, the data is released for processing. While the data is decrypted and being processed by the computer, it is invisible to everything and everyone else. This includes the cloud provider, other computer resources, hypervisors, virtual machines, and even the operating system.

Intel and AMD offer different technologies and approaches to achieve confidential-computing secure enclaves within their respective processor architectures. Below is a comparison between Intel's technologies (SGX, TDX, and TME) and AMD's features (SEV and SME) in terms of their approaches, functionalities, and key characteristics.

Intel's technologies:

- **Intel SGX (Software Guard Extensions):** these create isolated secure enclaves within the CPU's memory, allowing applications to protect sensitive code and data. This enables developers to create isolated execution environments for applications, protecting data and code even from higher-privileged software layers. It provides memory encryption, secure execution, remote attestation, and isolation.
- **Intel TDX (Total Memory Encryption and Intel Trusted Execution Technology):** this focuses on enhancing security in virtualized environments by providing memory encryption and secure execution environments for virtual machines. It provides total memory encryption, secure boot processes, and hardware-based isolation to protect against attacks in virtualized environments. Intel TDX protects VMs from unauthorized access and tampering, ensures secure migrations, and provides a trusted execution environment.
- **Intel TME (Total Memory Encryption):** this encrypts system memory to safeguard against unauthorized access, ensuring data confidentiality even if an attacker gains physical access to the memory. It protects system memory contents through encryption, ensuring data confidentiality and integrity. Intel TME aims to prevent data breaches and unauthorized access to memory contents.

AMD's technologies:

- **AMD SEV (Secure Encrypted Virtualization):** this focuses on enhancing security in virtualized environments by providing hardware-based memory encryption for VMs. It offers memory encryption for each VM, isolating them from each other and the hypervisor, protecting against attacks in cloud environments. AMD SEV provides memory encryption, isolation, and facilitates secure VM migrations between physical hosts.
- **AMD SME (Secure Memory Encryption):** this encrypts the system's memory, protecting against unauthorized access and physical attacks by encrypting memory contents. It encrypts system memory contents transparently without requiring specific software modifications, protecting against memory snooping attacks. AMD SME protects memory contents through encryption, enhancing security against physical attacks

Comparing the two, both Intel and AMD technologies aim to provide hardware-based security mechanisms to protect sensitive data and create secure enclaves. Intel SGX and AMD SEV focus on creating isolated execution environments for applications or virtual machines, whereas Intel TME and AMD SME concentrate on encrypting system memory to protect against unauthorized access.

Intel's TDX is more tailored for virtualized environments, offering features for VM security, while AMD's SEV is similarly focused on enhancing security in virtualized environments. Each technology has its unique characteristics, such as Intel SGX's focus on secure execution or AMD SEV's capabilities for secure VM migrations.

Overall, both Intel and AMD technologies contribute significantly to confidential computing by offering hardware-based security features, encryption mechanisms, and isolation to protect against various threats and attacks targeting sensitive data and applications. The choice between these technologies often depends on specific use cases, system requirements, and compatibility with the existing infrastructure.

Why use confidential computing?

In summary, confidential computing is critically important in server operations for the following reasons:

- To protect sensitive data, even while in use—and to extend cloud computing benefits to sensitive workloads. When used together with data encryption at rest and in transit with exclusive control of keys, confidential computing eliminates the single largest barrier to moving sensitive or highly regulated data sets and application workloads from an inflexible, expensive on-premises IT infrastructure to a more flexible and modern public-cloud platform.
- To protect intellectual property. Confidential computing isn't just for data protection. The TEE can also be used to protect proprietary business logic, analytics functions, machine-learning algorithms, or entire applications.
- To collaborate securely with partners on new cloud solutions. For example, one company's team can combine its sensitive data with another company's proprietary calculations to create new solutions—without either company sharing any data or intellectual property that it doesn't want to share.
- To eliminate concerns when choosing cloud providers. Confidential computing lets a company leader choose the cloud-computing services that best meet the organization's technical and business requirements, without worrying about storing and processing customer data, proprietary technology and other sensitive assets. This approach also helps alleviate any additional competitive concerns if the cloud provider also provides competing business services.
- To protect data processed at the edge. Edge computing is a distributed computing framework that brings enterprise applications closer to data sources such as IoT devices or local edge servers. When this framework is used as part of distributed cloud patterns, the data and application at edge nodes can be protected with confidential computing.

The Confidential Computing Consortium

In 2019, a group of CPU manufacturers, cloud providers, and software companies—Alibaba, AMD, Baidu, Fortanix, Google, IBM/Red Hat, Intel, Microsoft, Oracle, Swisscom, Tencent, and VMware—formed the Confidential Computing Consortium (CCC) under the auspices of The Linux Foundation. Cisco Systems, Inc., is a member of the consortium.

The CCC's goals are to define industry-wide standards for confidential computing and to promote the development of open-source confidential computing tools. Two of the Consortium's first open-source projects, Open Enclave SDK and Red Hat Enarx, help developers build applications that run without modification across TEE platforms.

However, some of today's most widely used confidential computing technologies were introduced by member companies before the formation of the Consortium. For example, Intel SGX (Software Guard Extensions) technology, which enables TEEs on the Intel Xeon® CPU platform, has been available since 2016; in 2018 IBM made confidential computing capabilities generally available with its IBM Cloud Hyper Protect Virtual Servers and IBM Cloud Data Shield products.

Secure data delivery and storage

The third and final pillar we will examine is the secure storage and delivery of data. This deals with encryption, key management, and data ingress/egress isolation. Traditional ciphers used in data encryption are nearing their functional end-of-life due to the encroaching capabilities of quantum computing. It is becoming increasingly important to consider and utilize post-quantum cryptography as part of a holistic approach securing data. To this end, Cisco announced membership in the Post-Quantum Cryptography Alliance in February of 2024. The goal is to guide and implement quantum-resistant ciphers in the industry and across Cisco products.

SED controller and drive states

Cisco UCSM reports on the security status of Self-Encrypting Drives (SEDs) and the disk controller itself using security flags. These flags indicate the item’s current state regarding encryption and access.

The storage controller and disks have the following security flags:

- **DriveSecurityCapable**—Indicates that the controller or disk is capable of supporting SED management.
- **DriveSecurityEnabled**—Indicates that the controller is Security enabled and disks in this controller can be further secured using the storage policy.

Note:

Before you configure drive security, the controller flag will be set to DriveSecurityCapable. After you configure drive security, this status will change to DriveSecurityEnabled.

The following security flags are exclusive to storage disks:

- **Locked**—The drive, initially locked in the primary server, is transferred to the current server. To access the data, you must unlock the drive by either entering the manual security key or reconnecting to the original KMIP key management server.
- **Foreign**—The drive, previously configured with virtual drives in the primary server, is relocated to the current server. To preserve and access the original virtual drive data, you must import this configuration. If these Virtual Drives must be secured, you must unlock the Physical Drives before importing the foreign configuration.

- **Unencrypted**—The drive can be encrypted but is currently not encrypted.
- **Unlocked**—The drive is currently encrypted, but the data is accessible to the user unencrypted.

Important Notes:

- Initially, a secure-enabled SED drive shows a "Drive State" of JBOD and a Security Flag of Unlocked when it is encrypted and not part of a Virtual Disk.
- Moving a secured drive to a new server will change the Security Flag to Foreign Locked.
- To resolve a locked SED drive, you must unlock it or perform a secure erase.

Tri-mode disk controller behavior

The Microchip Tri-mode disk controller is a drive controller capable of handling SAS, SATA, and NVMe drives. It is configurable to manage keys for SEDs and is required to enable and disable encryption. The following two conditions apply:

1. Disabling security on the controller requires a reboot.
2. Changing key mode from local to remote and vice versa is a destructive process. Any VDs must be removed and security must be disabled. A reboot is required.

The tri-mode controllers have a BIOS setting that enables or disables a controller password requirement. Note that the passphrase mentioned here is NOT the passphrase used to generate the local key encryption KEK. This is the controller passphrase entered in the BIOS to secure the controller itself.

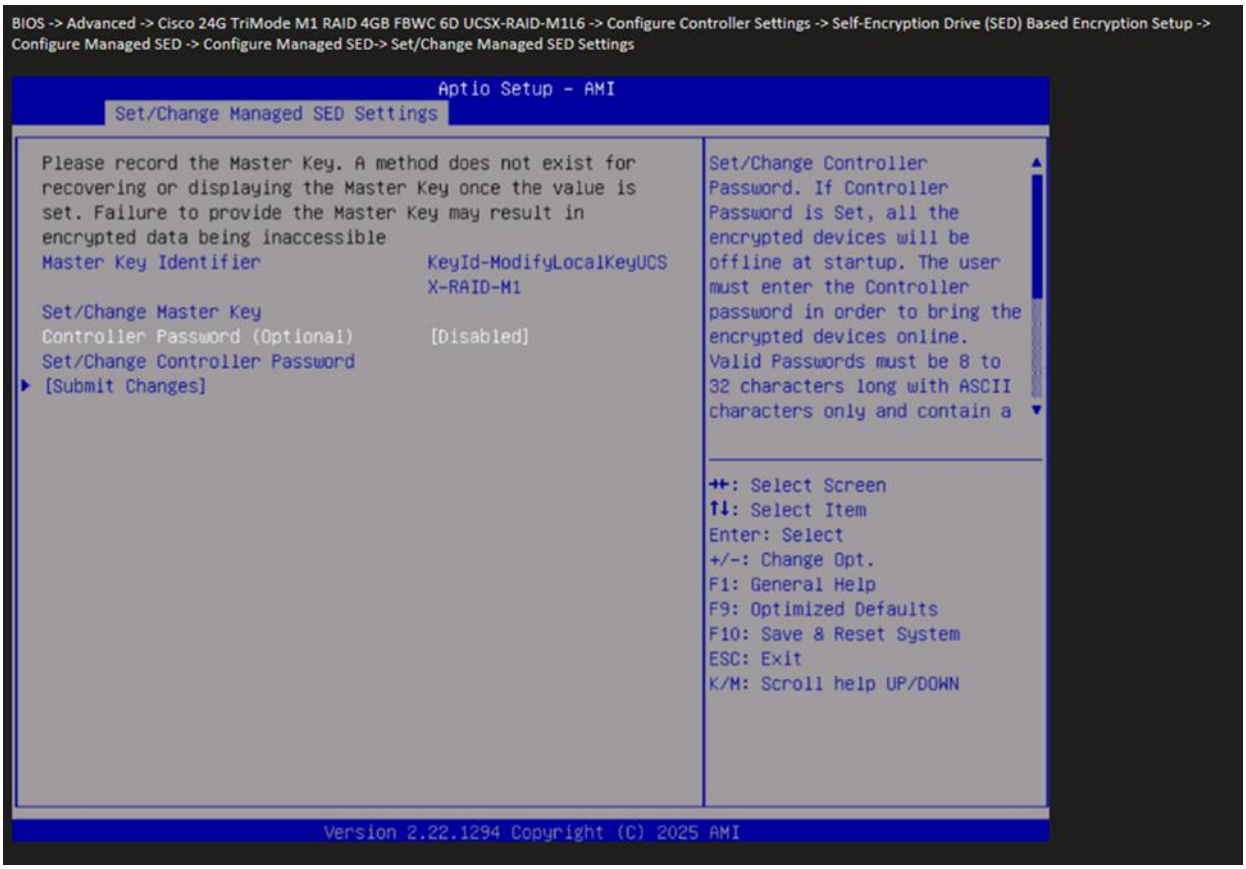

Figure 25.
The BIOS screen for entering the tri-mode controller password.

The following table lists the tri-mode controller boot behavior given the various possible controller states.

Table 6. Drive controller boot and passphrase behavior for various conditions.

Key Type	Controller BIOS passphrase	KMS Available	Boot behavior	Notes
Local	Disabled	N/A	Intervention not required. Legacy cached key behavior.	Cached key boot.
Local	Enabled	N/A	Intervention required for SED access. Manual entry of controller passphrase required.	Interrupted boot.
Remote	Disabled	No	Intervention not required. Legacy cached key behavior.	Cached key boot.
Remote	Disabled	Yes	No intervention required.	Normal boot.
Remote	Enabled	No	Intervention required for SED access. Manual entry of controller passphrase required.	Interrupted boot.
Remote	Enabled	Yes	No intervention required.	Normal boot.

SED drives with encrypted Virtual Disks (VDs)

If you build a secure RAID VD via storage policy and the SEDs are in an unencrypted state, the secure VD will fail validation saying that the drives are unencrypted. During validation the following error will be displayed “Existing drive group SED at end point is not secure, disable encryption for this drive group”. The SEDs must be in an encrypted state with security flag of “unlocked” to create a secure VD.

Encryption and key management

Encryption and remote key management play critical roles in ensuring secure data delivery, particularly in scenarios where sensitive information is transmitted or stored. These security measures contribute to protecting data confidentiality, integrity, and authenticity.

Encryption is primarily employed to ensure the confidentiality of data during transmission or while stored on a system. Cisco UCS has support for hardware-based encrypted drives (SEDs) and can maintain a local key or be configured to securely use a remote Key Management Server (KMS). This is encryption for data at rest (DARE). Data in transit can be encrypted in many ways, and Cisco UCS, with its robust ecosystem, can take advantage of all on-wire encryption solutions that are contained in Cisco products. This can be accomplished in hardware—for example, on point-to-point or perimeter network devices, or by using “Cisco on Cisco” (e.g., running a Cisco virtual firewall on a Cisco UCS server) with the myriad virtual solutions that can run directly on a containerized UCS or hypervisor-based deployment. By encrypting data at both ends—during transmission and storage—organizations ensure that sensitive information remains secure throughout its lifecycle.

Key management is an important aspect of an encryption deployment. Remote key management involves securely storing encryption keys separate from the encrypted data. Keys are often considered as sensitive as the data they encrypt. By managing keys remotely and securely, organizations prevent a single point of failure and reduce the risk of unauthorized access to both data and keys.

Regularly rotating and updating encryption keys is a security best practice. Remote key management systems facilitate the secure rotation and distribution of new keys. This helps ensure that even if a key is compromised, the window of vulnerability is limited, and older keys are no longer in use.

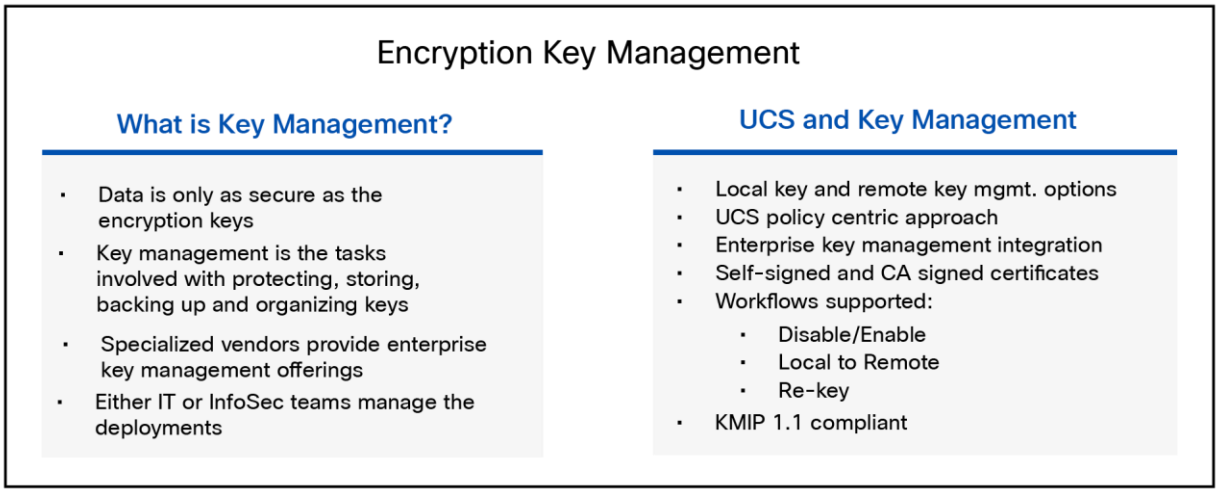

Figure 26.
UCS key management features.

Manual key

A manual key policy uses a passphrase, supplied by the user, which is used to generate the Key Encryption Key (KEK). It is vitally important that you back up or otherwise store your passphrase in a safe location; for example, in a secured password manager. If the passphrase is lost, it is unretrievable, and you will not be able to unlock or rekey your drives. The passphrase should contain at least eight characters and at least one each of the following: upper case letter, lower case letter, number, and special character.

Remote key

Once you are in the policy wizard for SED drive management, click Start once you have named the policy. Select the Remote Key Management radio button.

Here you will enter your KMS information for your primary and secondary KMS server, if one is available. You also have the option to change the secure communication port if you have set up your KMS with a custom value. Upload the root CA certificate from your KMS server and proceed to create the policy.

Self-Encrypting Drives (SEDs)

Data-at-rest encryption on a Cisco UCS server can happen in software for VMs (for example, Vormetric Transparent Client of VMcrypt) or by the operating system (for example, Microsoft's BitLocker). This can also be accomplished using hardware. To that end, Self-Encrypting Drives (SEDs) were developed and have many advantages. SEDs have a negligible impact on performance speed and latency. The encryption process is completely integrated into the drive, so there is no need for other system components to step in and perform any heavy lifting. SEDs are independent of the operating system, so even if a hacker attacks a computer, it is nearly impossible to access the SED (and the encryption keys stored within) when the computer is turned off.

In a Cisco UCS server, SEDs can utilize a local key (security key) or a remote key management solution. Remote key management is the recommended method since it doesn't rely on stored or "remembered" passphrases. The key management software optimizes the SED's decryption and encryption functions and key management, relieving the user of any active SED administration. SEDs are also inexpensive to deploy and maintain. SEDs encrypt the moment they come off the assembly line. Management software does the rest, ensuring that SEDs do their job without the need for human intervention.

Self-Encrypting Drives (SEDs) have special hardware that encrypts incoming data and decrypts outgoing data in real time. The data on the disk is always encrypted on the disk and stored in the encrypted form. The encrypted data is always decrypted when read from disk. A media-encryption key controls this encryption and decryption. This key is never stored in the processor or memory. Cisco UCS Manager supports SED security policies on Cisco UCS C-Series servers, B-Series M5 servers, and S-Series servers.

SEDs must be locked by providing a security key. The security key, which is also known as a key-encryption key or an authentication passphrase, is used to encrypt the media-encryption key. If the disk is not locked, no key is required to fetch the data.

Cisco UCS Manager enables you to configure security keys locally or remotely. When you configure the key locally, you must remember the key. If you forget the key, it cannot be retrieved, and the data is lost. You can configure the key remotely by using a key management server (also known as a KMIP server). This method addresses the issues related to safekeeping and retrieval of the keys in local key management.

Encryption and decryption for SEDs are done through the hardware and thus do not affect overall system performance. SEDs reduce disk retirement and redeployment costs through instantaneous cryptographic erasure. Cryptographic erasure is done by changing the media-encryption key. When the media-encryption key of a disk is changed, the data on the disk cannot be decrypted and is immediately rendered unusable. With Cisco UCS Manager Release 3.1(3), SEDs offer disk theft protection for C-Series and S-Series servers.

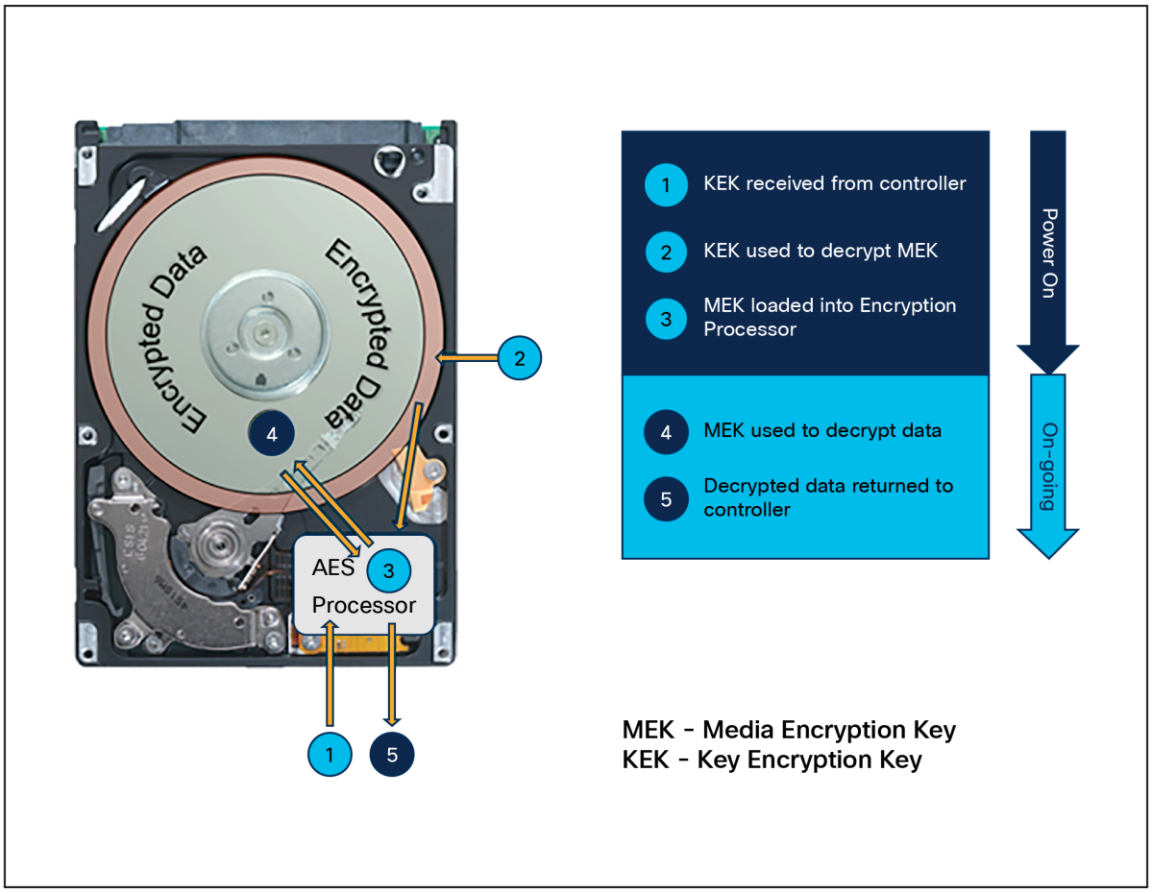

Figure 27.
Anatomy of a self-encrypting drive.

Instant Secure Erase (ISE) drives

Instant Secure Erase is a superset of non-crypto-based Secure Erase; it utilizes encryption to make data unreadable. Instant Secure Erase contains the commands of Secure Erase but also adds a "crypto" erase command. This command can be utilized by both hard disks and solid-state drives if available. With ISE, each disk creates a cipher key that is used to decrypt/encrypt data as it is being read or written. When the crypto command is accessed, the cipher key is destroyed and all data on the disk is unable to be read. Because there is no need to overwrite the data, ISE only takes a few milliseconds to make the disk unreadable compared to other sanitization methods, which can take several hours, depending on the number of passes and the size of the disk. Instant Secure Erase is also supported by the NIST (under cryptographic erase), and usually coupled with the FIPS (Federal Information Processing Standard) certification.

While Instant Secure Erase uses cryptographic techniques to securely erase data, it does not offer data encryption to protect data at rest. SEDs are required for this (see the previous section).

Virtual Interface Card (VIC)

A Cisco UCS VIC (Virtual Interface Card) plays a crucial role in enhancing secure data delivery within the Cisco Unified Computing System (Cisco UCS) infrastructure. The VIC is a key component that provides network connectivity for servers within the Cisco UCS platform.

The primary function of a Cisco UCS VIC is to provide network connectivity for servers. It supports Ethernet and Fibre Channel over Ethernet (FCoE) protocols, allowing servers to communicate with the network infrastructure securely. Cisco UCS utilizes a unified fabric approach, where both data and storage traffic share the same high-speed fabric. The VIC enables this convergence, simplifying cabling and providing a more efficient and flexible network architecture. Cisco UCS VICs support adaptive networking features, allowing them to dynamically adjust to changing network conditions. This adaptability helps optimize data delivery performance and responsiveness. The VIC also supports Quality-of-Service (QoS) features, allowing administrators to prioritize and manage network traffic based on application requirements. This helps ensure that critical data is delivered with the appropriate level of service.

Cisco UCS VICs are designed to work seamlessly with virtualized environments. They provide support for VMware, Microsoft Hyper-V, and other hypervisors, enabling secure data delivery in virtualized infrastructures.

Cisco UCS servers that include VICs incorporate security features such as secure boot processes. This helps establish a chain of trust during server bootup, ensuring the integrity of the system and reducing the risk of compromise.

A VIC, when used in conjunction with a fabric interconnect, is the cornerstone of a Cisco UCS server's traffic isolation and segmentation. This can be essential for enhancing security by keeping different types of traffic separate, preventing unauthorized access to sensitive data.

By combining these features, a Cisco UCS VIC contributes to the secure and efficient delivery of data within the UCS infrastructure. It plays a central role in enabling unified fabric, supporting virtualized environments, and providing the necessary connectivity for secure and reliable data communication.

Conclusion

When we combine the inherent security features of the Cisco UCS Manager platform with common sense security practices such as:

- Maintaining physical security
- Keeping server OS and firmware patched and updated to mitigate new threats
- Disabling functions that are not required
- Maintaining application security with RBAC, patching, and firewalls
- Storing and delivering data securely with encryption in hardware on the server and during data transmission through ecosystem design we can ensure that our server environments are as secure as possible.

For more information

For additional information, see the following resources:

- Cisco Trust Portal
- Cisco Security
- Cisco Security Advisories
- DMTF and Redfish
- Cisco UCS Manager XML API Programmer's Guide – Cisco UCS Manager XML API Method Descriptions [Cisco UCS Manager]
- UCSM Multi-Factor Authentication
- Cisco UCS Manager Network Management Guide (TCP and UDP ports, etc.)
- File server security (concerns (blog))
- Cisco Trustworthy Technologies
- Audit Log entries and retention (Cisco Community discussion)
- Cisco UCS Secure Data Deletion
- Cisco joins Post-Quantum Cryptography Alliance
- Post Quantum Cryptography Alliance

Additional Cisco network-monitoring resources (cited in this document):

- Cisco UCS Manager MIB Reference Guide
- Cisco UCS Manager Fault Reference Guide
- Cisco UCS C-Series Standalone Servers MIB Reference Guide
- Cisco C-Series Servers Fault Reference Guide
- Monitoring Cisco UCS Manager Using Syslog
- Cisco UCS Manager and C-Series Standalone Servers Tech Talk available here
- See IMM Security Policy Checklist and Intersight Help for recommended policy configuration settings.

Appendix A – UCS networking ports

UCSM network ports – TCP and UDP

Default open ports

Table 7. Default open ports

Port	Interface	Protocol	Traffic type	Fabric interconnect	Usage
22	CLI	SSH	TCP	Cisco UCS 6200 Series Cisco UCS 6300 Series Cisco UCS 6400 Series Cisco UCS 6500 Series	Cisco UCS Manager CLI access
80	XML	HTTP	TCP	Cisco UCS 6200 Series Cisco UCS 6300 Series Cisco UCS 6400 Series Cisco UCS 6500 Series	Cisco UCS Manager GUI and third-party management stations Client download
443	XML	HTTP	TCP	Cisco UCS 6200 Series Cisco UCS 6300 Series Cisco UCS 6400 Series Cisco UCS 6500 Series	Cisco UCS Manager login page access Cisco UCS Manager XML API access
743	KVM	HTTP	TCP	Cisco UCS 6200 Series Cisco UCS 6300 Series Cisco UCS 6400 Series	CIMC Web Service / Direct KVM
7546	CFS	CFSD	TCP	Cisco UCS 6400 Series Cisco UCS 6500 Series	Cisco Fabric Service

TCP and UDP ports

The tables below list the incoming and outgoing TCP and UDP ports used in Cisco UCS for management access.

Table 8. Incoming ports

Port	Interface	Protocol	Traffic type	Usage
23	CLI	Telnet	TCP	Cisco UCS Manager CLI access
22	CLI	SSH	TCP	Cisco UCS Manager CLI access
443	Static HTML	HTTPS	TCP	Cisco UCS Manager login page access
80	Static HTML	HTTP	TCP	Client download
443	XML	HTTPS	TCP	Cisco UCS Manager XML API access
80	XML	HTTP	TCP	Ports used by Cisco UCS Manager GUI and third-party management stations
23	Serial-over-LAN	Telnet	TCP	COM1 port access on a specified server
22	Serial-over-LAN	SSH	TCP	COM1 port access on a specified server
161	SNMP	SNMP	UDP	SNMP MIBs exposed for monitoring
623	IPMI-over-LAN	RMCP	UDP	IPMI access to BMCs
2068	KVM	HTTPS	TCP	Data path for the BMCs
5988	CIMC XML	HTTP	TCP	Send CIMC messages over HTTP
743	KVM	HTTP	TCP	CIMC Web Service / Direct KVM
5661		HTTPD	TCP	Internal communication This port is applicable only to Cisco UCS 6400 Series Fabric Interconnects. It is disabled in Cisco UCS Manager Release 4.0(4f) and later releases.
7162		HTTPD	TCP	Internal communication This port is applicable only to Cisco UCS 6400 Series Fabric Interconnects. It is disabled in Cisco UCS Manager Release 4.0(4g) and later releases.
7546	CFS	CFSD	TCP	Cisco Fabric Service This port is applicable only to UCS 6400 Series Fabric Interconnects.

Table 9. Outgoing ports

Port	Service	Protocol	Traffic type	Usage
1812	AAA	RADIUS	UDP	AAA server authentication requests
1813	AAA	RADIUS	UDP	AAA server authentication requests
49	AAA	TACACS	TCP	AAA server authentication requests
389	AAA	LDAP	UDP	AAA server authentication requests
123	Time sync	NTP	UDP	Synchronize the time with global time servers
162	SNMP traps	SNMP	UDP	Send traps to a remote network-management system
25	Call Home	SMTP	TCP	Email-based and web-based notifications for critical system events
514	Syslog	SYSLOG	UDP	Cisco UCS Manager–generated syslog messages
53	Name resolution	DNS	UDP	DNS queries
69	TFTP	TFTP	UDP	File transfers
115	SFTP	SFTP	TCP	File transfers
20-21	FTP	FTP	TCP	File transfers
21	SCP	SCP	TCP	File transfers

Appendix B – PQC definitions

AIK – Attestation Identity Key. The Trusted Platform Module (TPM) can be used to create cryptographic public/private key pairs in such a way that the private key can never be revealed or used outside the TPM (that is, the key is non-migratable). An AIK can be used to guarantee that a certain cryptographic operation occurred in the TPM of a particular computer because any operation that uses the private key of such a key pair must have occurred inside that specific TPM.

An AIK can also be useful to prove cryptographically that a private key has this property and that any use of it must have occurred inside that TPM.

An attestation identity key is used to provide such cryptographic proof by signing the properties of the non-migratable key and providing the properties and signature to the CA for verification. Since the signature is created using the AIK private key, which can only be used in the TPM that created it, the CA can trust that the attested key is truly non-migratable and cannot be used outside that TPM.

CA – Certificate Authority, a trusted certificate signature provider

CC – Common Criteria is an international standard for computer security certification. It has various evaluation levels called EALs. Most organizations typically certify to EAL 2.

Cisco SKS – Cisco Session Key Services, basically proprietary SKIP

CNSA – Commercial National Security Algorithm (or Suite) is a set of cryptographic algorithms promoted by the National Security Agency as a replacement for NSA Suite B Cryptography algorithms.

CSfC - Commercial Solution for Classified certifications

DH – Diffie-Hellman key-exchange algorithm

EAP-TLS - Extensible Authentication Protocol–Transport Layer Security (EAP-TLS) is an IETF open standard defined in RFC 5216. More colloquially, EAP-TLS is the authentication protocol most commonly deployed on WPA2-Enterprise networks to enable the use of X.509 digital certificates for authentication.

EAP-TLS is considered the gold standard for network authentication security, but despite being universally recognized as ultra-secure, it's still not widely implemented.

ECC - Elliptic-curve cryptography is an approach to public-key cryptography based on the algebraic structure of elliptic curves.

ECDH - Elliptic-Curve Diffie–Hellman is a key-agreement protocol that allows two parties, each having an elliptic-curve public–private key pair, to establish a shared secret over an insecure channel.

HSS – Hierarchical signature system

IETF - Internet Engineering Task Force, founded in 1986, is the premier standards development organization (SDO) for the internet.

IKE - IKE (Internet Key Exchange) is a protocol used in IPsec (internet protocol security) for ensuring secure, authenticated key exchange and establishing Security Associations (SAs). IKE plays a crucial role in setting up the cryptographic parameters for securing IP communications.

IKE automates the process of generating, exchanging, and managing cryptographic keys required for IPsec, and also negotiates the IPsec Security Associations (SAs) parameters.

Kyber – Kyber is a Key-Encapsulation Mechanism (KEM) designed to be resistant to quantum decryption attacks. It is used to establish a shared secret between two communicating parties without an attacker in the transmission system being able to decrypt it. This is an asymmetric cryptosystem.

LDWM (Lamport, Diffie, Winternitz, and Merkle) – a special hashing scheme developed for signatures that is considered to be quantum resistant

LMS – Leighton-Micali signature, a stateful hash-based algorithm and its multi-tree variants used for HSS

MACsec – MAC address security. MACsec typically relies on PPK

NDcPP – Network device collaborative protection profile

OTN SEC – Optical transport network security

QKD – Quantum key distribution is a secure communication method that implements a cryptographic protocol involving components of quantum mechanics. It enables two parties to produce a shared random secret key known only to them, which then can be used to encrypt and decrypt messages. The process of quantum key distribution is not to be confused with quantum cryptography, but it is the best-known example of a quantum-cryptographic task.

PPK – Pre-placed key (symmetric encryption key, prepositioned in a cryptographic unit)

PQC – Post-quantum cryptography

SHA – Secure hash algorithm

Shim – A shim is a pre-bootloader that runs on UEFI systems and is meant to be a bit of code, signed by Microsoft, that embeds Cisco's certificate (which signs Cisco's GRUB binaries) so that the system can load the "real" bootloader: GRUB.

SKIP – SKIP (Simple Key-Management for Internet Protocol) is a protocol for sharing encryption keys1. It generates platform-independent encryption keys for specific sender-receiver pairs2. The SKIP cipher is a transposition cipher that reorders letters in a message3. In SKIP, the master key is hashed to produce the key used for IP packet-based encryption and authentication.

SUDI – Secure Unique Device Identifier is an IEEE 802.1AR-compliant secure device identity in an X.509v3 certificate that maintains the product identifier and serial number. The SUDI can be used for asymmetric key operations such as encryption, decryption, signing, and verifying that allow passage of the data to be operated upon.

TPM – Trusted Platform Module. An immutable hardware key store.

XMSS – eXtended Merkle Signature Scheme, a stateful hash-based algorithm and its multi-tree variants used for HSS.

Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV Amsterdam,
The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at **https://www.cisco.com/go/offices**.

Printed in USA C07-4855701-01 07/25