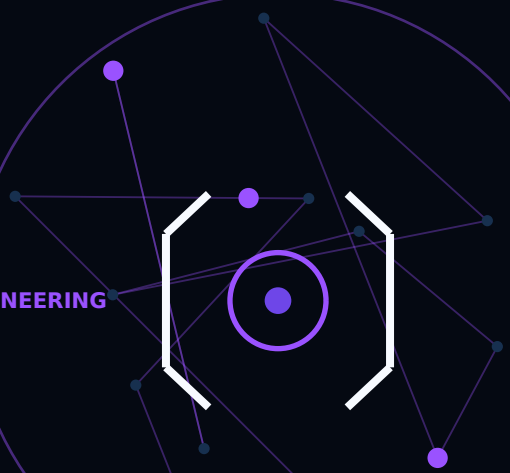


ENGINEERING STANDARDS LIBRARY / PLATFORM ENGINEERING

Hardware and Software Lifecycle and Refresh Standard

Lifecycle intelligence, supportability, refresh planning, compatibility management, technical debt, and controlled retirement.

PERMANENT DOCUMENT ID

MYTHOS-PLT-0005

PUBLICATION

Candidate Standard
Version 1.0.0-candidate

ASSURANCE PROFILE

Standard / Advanced
High Assurance

PERMANENT PUBLICATION IDENTITY

Hardware and Software Lifecycle and Refresh Standard

DOCUMENT ID
MYTHOS-PLT-0005

VERSION
1.0.0-candidate

STATUS
Candidate Standard

PUBLISHER
Mythos Systems

PUBLICATION DATE
2026-07-24

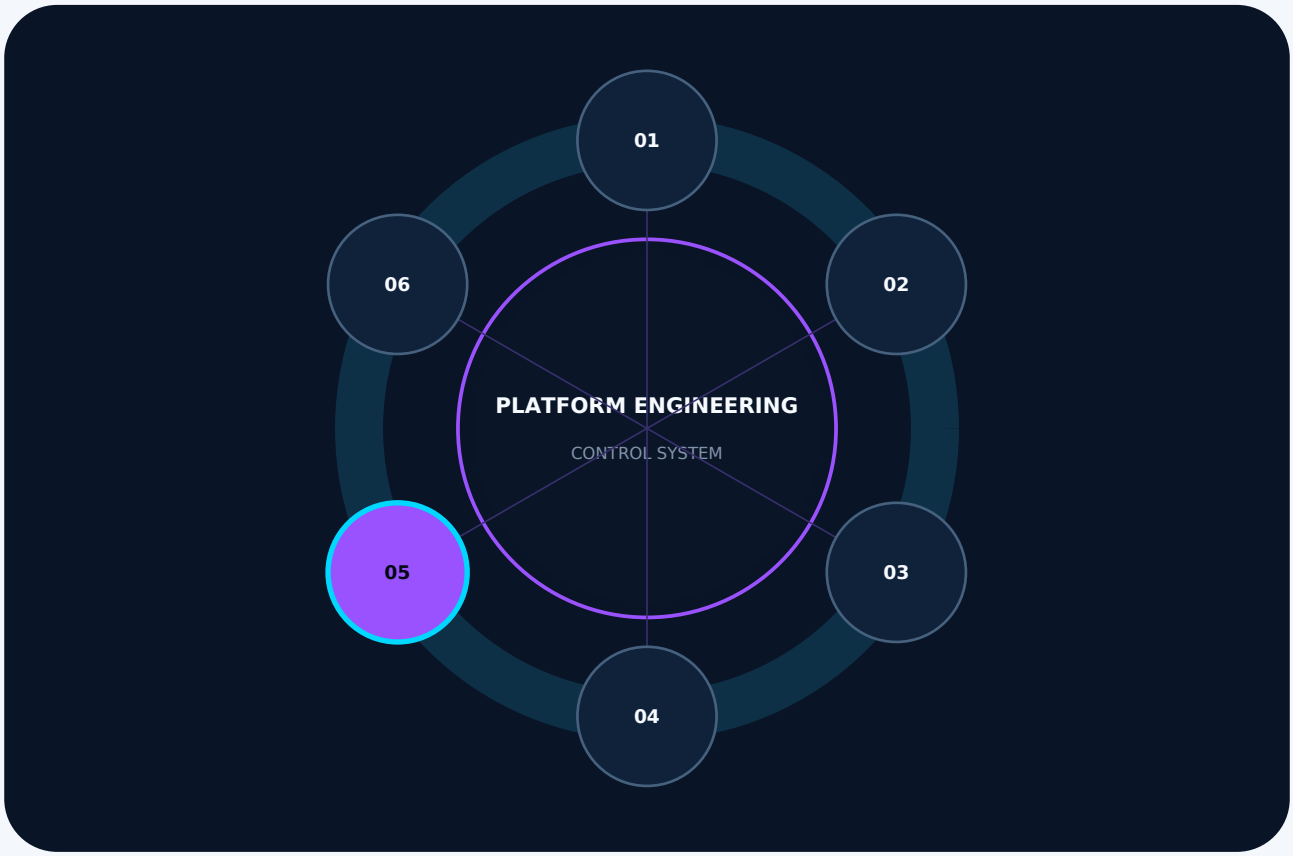
SCHEDULED REVIEW
2027-01-24

12	6	4	1
ATOMIC CRITERIA	ASSURANCE VIEWS	IMPLEMENTATION PROFILES	PERMANENT IDENTITY

Publication doctrine. This publication establishes durable guidance or requirements for its defined scope. Interpretation must preserve the permanent document identity, recorded evidence, controlled exceptions, and formal revision history.

DOMAIN CONTROL SYSTEM

MYTHOS-PLT-0005 inside the platform engineering standard constellation



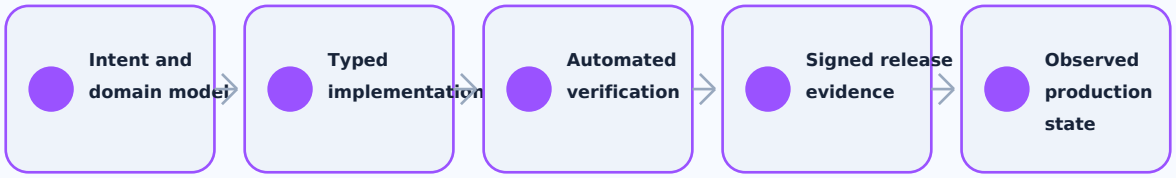
Connected assurance	Architecture, implementation, operations, evidence, and lifecycle are evaluated as one controlled system.
Specialization rule	Companion standards may add precision, but they cannot silently weaken this publication's requirements.

EXECUTIVE POSITION

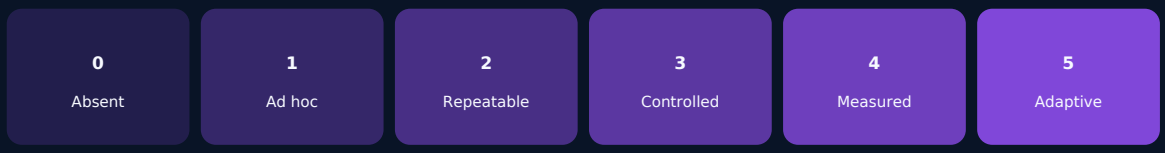
Hardware and Software Lifecycle and Refresh Standard

The architecture of asset and lifecycle management has failed.

Evidence-bearing delivery path



Assurance maturity spectrum



CONTENTS

Publication navigation

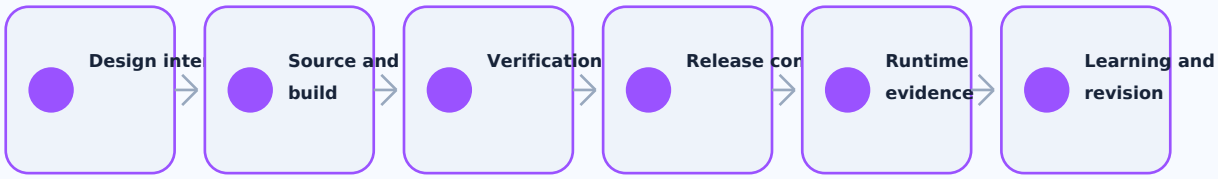
CONTROL TOPOLOGY

Assurance model and evaluation surface

The standard is assessed as an evidence-bearing system. Capability scores do not replace mandatory gates, implementation proof, or operational validation.

Capability family	Evaluation nodes
EOL/EOS Tracking and Automation (Weight: 20%)	1
Hardware Refresh Cadences (Weight: 15%)	1
Software Dependency Pruning (Weight: 15%)	1
Cabling Structure and Management (Weight: 20%)	1
Secure Disposal and Sanitization (Weight: 15%)	1
Asset Inventory (DCIM) as Code (Weight: 15%)	1
Suite Composition	2
Additional Evaluation Dimension	4

Lifecycle control loop



EVALUATION INDEX

Criteria and evidence map

#	Capability family	Criterion
1	EOL/EOS Tracking and Automation (Weight: 20%)	Lifecycle Visibility
2	Hardware Refresh Cadences (Weight: 15%)	Proactive Replacement
3	Software Dependency Pruning (Weight: 15%)	Legacy Software Eradication
4	Cabling Structure and Management (Weight: 20%)	Physical Plant Rigor
5	Secure Disposal and Sanitization (Weight: 15%)	Data Destruction
6	Asset Inventory (DCIM) as Code (Weight: 15%)	Infrastructure as Data
7	Suite Composition	MLCS-Lifecycle
8	Suite Composition	MLCS-Physical
9	Additional Evaluation Dimension	Lifecycle Dimensions
10	Additional Evaluation Dimension	The Lifecycle Doctrine
11	Additional Evaluation Dimension	Scoring Model
12	Additional Evaluation Dimension	Suite Composition

NORMATIVE PUBLICATION

Requirements, evaluation methodology, and implementation guidance

The following body preserves the substantive source standard while normalizing publication status, identity, navigation, and visual presentation.

Executive Mandate

The architecture of asset and lifecycle management has failed.

Organizations run production-critical infrastructure on 7-year-old routers and operating systems that haven't received a security patch in years. They track thousands of physical servers and software licenses using static Excel spreadsheets that are obsolete the moment they are saved. In the datacenter, they bind cables with plastic zip ties, creating physically rigid bundles that guarantee a single cable change will disrupt the entire link, all while having no idea what a randomly colored fiber cable actually connects.

This standard exists because:

1. EOL/EOS is a Critical Security Vulnerability: Running hardware or software past its End-of-Life (EOL) or End-of-Support (EOS) date is an unconditional security failure. Vendors stop releasing patches, meaning zero-day vulnerabilities will remain permanently unpatched. Systems MUST be retired or isolated before they hit EOL.
2. Spreadsheets are Not Asset Management: If an organization cannot programmatically query the exact firmware version, warranty status, and location of a piece of hardware via an API, they do not own their infrastructure; they are hostage to it. Asset management MUST be an automated, immutable, API-driven Source of Truth (e.g., DCIM/IPAM).
3. Zip Ties are Architectural Sabotage: Plastic zip ties damage fiber optics, restrict copper airflow, and create rigid bundles that make maintenance impossible without causing widespread physical disruption. Cabling MUST be bound exclusively with hook-and-loop (Velcro) and managed via structured, color-coded pathways.
4. Refresh Cycles Must Be Proactive, Not Reactive: Waiting for a server to catch fire or a switch to panic before replacing it is break-fix theater. Organizations MUST implement scheduled, rolling refresh cadences based on MTBF (Mean Time Between Failures), thermal degradation curves, and technological obsolescence.
5. Data Sanitization Must Be Cryptographic: Throwing a hard drive in the trash or running a single `dd` pass is a data breach waiting to happen. Hardware retirement MUST include cryptographic erasure (destroying the encryption keys) or physically verified degaussing/shredding.

This document establishes the Mythos Lifecycle Standard (MLCS), a comprehensive, evidence-based methodology for evaluating hardware and software lifecycle processes against automation rigor, physical hygiene, and zero-legacy compliance.

Scope and Applicability

In Scope

This standard covers the evaluation of:

1. Hardware End-of-Life (EOL) and End-of-Support (EOS) tracking
1. IT Asset Management (ITAM) and DCIM automation
1. Software dependency lifecycle and SBOM pruning
1. Technology refresh cadences and capital expenditure (CapEx) planning
1. Structured cabling standards (TIA-606-C, color-coding, hook-and-loop)
1. Secure data sanitization and hardware disposal

Out of Scope

1. General datacenter cooling and power topology (covered in MYTHOS-NET-0009)
1. General CI/CD supply chain security (covered in MYTHOS-PLT-0001)
1. General vulnerability scoring (CVEs) (covered in MYTHOS-SEC-0008)

Audience

1. Datacenter facility managers and mechanical engineers
1. IT asset managers and procurement teams
1. Network engineers and cable plant technicians
1. Platform engineers managing software dependency lifecycles
1. Security teams evaluating end-of-life risk and data sanitization
1. Standards bodies seeking a reference lifecycle methodology

Definitions and Taxonomy

Lifecycle Dimensions

Dimension	Definition
EOL (End-of-Life)	The date after which a vendor will no longer manufacture or sell a piece of hardware or software version.
EOS (End-of-Support)	The date after which a vendor will no longer provide security patches, bug fixes, or technical support for a product. Running EOS software is a critical security failure.
DCIM (Datacenter Infrastructure Management)	Software tools that monitor, measure, and manage datacenter resources (power, cooling, space, connectivity) and track hardware lifecycles in real-time.
Cryptographic Erasure	The process of rendering data permanently unreadable by securely destroying the encryption keys used to encrypt the data, rather than attempting to overwrite the physical disk.
Structured Cabling	A standardized architecture for telecommunications cabling, using dedicated pathways, patch panels, strict color-coding, and hook-and-loop fasteners to ensure modularity.

The Lifecycle Doctrine

> A spreadsheet is not an asset database. A zip tie is a physical chokehold. An EOS switch is a compromised switch. If the hardware cannot be cryptographically erased before disposal, the data is already breached.

Evaluation Methodology

Scoring Model

Each capability is scored from 0 to 5.

Score	Meaning
0	Fails basic task; spreadsheets, zip ties, running EOS hardware, no disposal policy
1	Basic asset tracking, but manual EOL checks and disorganized cabling
2	Functional DCIM, but reactive refresh cycles and inconsistent cable colors
3	Solid production performance; automated EOL alerts, scheduled refreshes, TIA-606-C cabling, crypto-erasure
4	Strong performance; zero-landfill disposal, predictive refresh modeling, zero zip-ties
5	Best-in-class; sets the standard for mathematically verified, zero-legacy lifecycle management

Weighting

Category	Weight	Rationale
EOL/EOS Tracking & Automation	20%	Running unsupported hardware/software is an unconditional security failure
Hardware Refresh Cadences	15%	Reactive break-fix causes catastrophic outages and unplanned spending
Software Dependency Pruning	15%	Bloated software portfolios accumulate massive technical debt and vulnerabilities
Cabling Structure & Management	20%	Zip ties and unlabeled cables cause massive MTTR during physical outages
Secure Disposal & Sanitization	15%	Improperly disposed drives guarantee data exfiltration
Asset Inventory (DCIM) as Code	15%	If assets aren't API-driven, they cannot be automated or reconciled

Total: 100%

A system MUST score at least 3.0 in EOL/EOS Tracking and Cabling Structure to be considered for production use.

Evaluation Categories

EOL/EOS Tracking and Automation (Weight: 20%)

Lifecycle Visibility

Does the organization have real-time, automated visibility into the support status of all assets?

Score	Criteria
0	Manual tracking via spreadsheets; no one knows what is EOS until it breaks
1	Basic ITAM tool used, but EOL dates must be checked manually on vendor websites
2	EOL dates imported into ITAM, but alerts are not automated
3	Automated EOL/EOS alerts triggered 6-12 months before expiration; integrated with procurement
4	Continuous API sync with vendor support portals; live firmware compliance tracking
5	Perfect tracking: formally verified asset registries, zero EOS assets in production, mathematical proof of continuous support coverage

Hardware Refresh Cadences (Weight: 15%)

Proactive Replacement

Is hardware replaced proactively based on obsolescence and reliability curves?

Score	Criteria
0	Break-fix only; hardware runs until catastrophic failure
1	Ad-hoc replacements when performance becomes a bottleneck
2	Standard 5-7 year refresh cycle, but execution is delayed and unplanned
3	Scheduled rolling refresh cycles based on MTBF and technological obsolescence (e.g., 3-4 years for compute, 5 for network)
4	Predictive refresh modeling using thermal and power degradation data from DCIM
5	Perfect cadence: formally verified lifecycle bounds, zero unplanned hardware failures, seamless CapEx integration

Software Dependency Pruning (Weight: 15%)

Legacy Software Eradication

Is the software portfolio continuously pruned of deprecated and unused components?

Score	Criteria
0	Legacy frameworks and OS versions kept running indefinitely ("we're afraid to touch it")
1	Periodic audits of installed software, but no enforcement of removal
2	SBOM (Software Bill of Materials) generated for all apps, but EOL dependencies remain
3	Automated CI/CD checks block the deployment of software with EOL dependencies

Score	Criteria
4	Continuous dependency scanning auto-quarantines apps running deprecated libraries
5	Perfect pruning: formally verified software registries, zero EOL dependencies in production, mathematical proof of minimal attack surface

Cabling Structure and Management (Weight: 20%)

Physical Plant Rigor

Is the cabling plant strictly organized, labeled, and safely bound?

Score	Criteria
0	"Spaghetti" cabling; plastic zip ties used; unlabeled fibers
1	Basic patch panels used, but inconsistent labeling and zip ties remain
2	TIA-606-C compliant labels at both ends; hook-and-loop (Velcro) straps used exclusively
3	Strict color-coding enforced (e.g., Red = Security, Blue = Copper, Orange = Internal Fiber, Yellow = OOB); zero zip ties on the premises
4	Cable paths documented in DCIM; fiber optic OTDR traces mapped; locking clips on critical fiber
5	Perfect structure: formally verified cable topologies, zero undocumented strands, automated patch-port validation via smart patch panels

Secure Disposal and Sanitization (Weight: 15%)

Data Destruction

How is data irrecoverably destroyed before hardware leaves the facility?

Score	Criteria
0	Drives thrown in the trash or sold on eBay with a quick `format`
1	Single-pass overwrite (e.g., `dd if=/dev/zero`), but no verification
2	Multi-pass wipe (DoD 5220.22-M) with cryptographic verification
3	Cryptographic Erasure: drives are encrypted at rest (AES-256); disposal = destroying the KEK (Key Encryption Key)
4	Physical degaussing and shredding of SSDs/HDDs with video evidence and certificate of destruction
5	Perfect sanitization: formally verified zero-recoverability, zero-landfill hardware recycling policy, cryptographic attestation of destruction

Asset Inventory (DCIM) as Code (Weight: 15%)

Infrastructure as Data

Is the physical asset inventory treated as an API-driven, declarative source of truth?

Score	Criteria
0	Asset data lives in a spreadsheet on a file share

Score	Criteria
1	Dedicated DCIM/ITAM platform used, but data entry is manual
2	DCIM auto-discovers network gear via SNMP, but compute/space/power tracked manually
3	API-first DCIM: all provisioning scripts automatically update asset location, rack elevation, and power draw
4	Real-time reconciliation: DCIM detects physical changes (e.g., server moved) via RFID/sensors and alerts on drift
5	Perfect inventory: formally verified asset parity, zero manual data entry, cryptographic attestation of physical topology

The Mythos Lifecycle Suite (MLCS)

Traditional facility audits measure "did we buy new stuff." The MLCS evaluates EOL risk exposure, physical cable traceability, and data sanitization guarantees.

Suite Composition

MLCS-Lifecycle

1. EOL Audit: 100+ tests querying the asset registry for any hardware or software past its EOS date, verifying zero critical assets are unsupported.
1. Software Prune Test: 50+ tests attempting to deploy a container with an EOL base image (e.g., Ubuntu 18.04), verifying the CI/CD pipeline blocks it.

MLCS-Physical

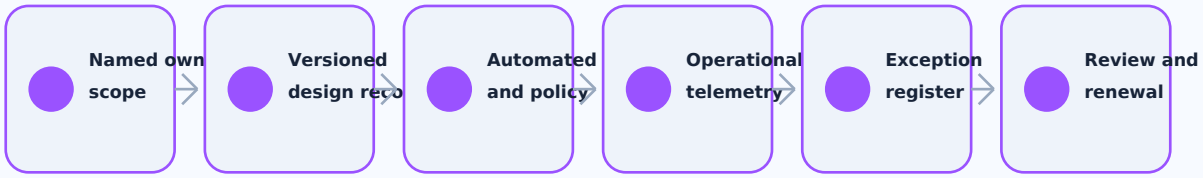
1. Cable Audit: 100+ visual and DCIM inspections verifying zero zip ties are present and all cables match the strict color-code matrix.
1. Blind Trace Test: 50+ tests requiring a technician to trace a specific port to its origin, verifying the DCIM and labeling system allows instant identification.
1. Sanitization Verification: 50+ tests attempting to recover data from retired drives, verifying cryptographic erasure or physical shredding was successful.

Execution Protocol

ASSURANCE AND ADOPTION

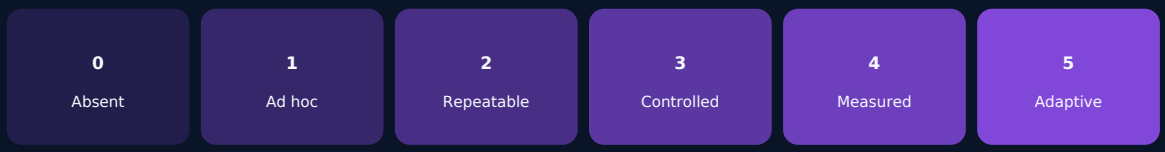
Implementation evidence and review gates

Required assurance chain



A passing score is not sufficient without current evidence. High-assurance adoption requires traceable design decisions, reproducible builds, independently reviewable tests, controlled secrets, runtime observability, failure exercises, and a dated exception register.

Assurance maturity spectrum



Current authority register

Authority	Version / snapshot	Applicability	Review by
Kubernetes Project: Kubernetes Documentation	v1.36 current documentation	High	2026-10-22
SLSA Project: Supply-chain Levels for Software Artifacts Specification	1.2 current specification snapshot	Moderate	2026-10-22
SPDX Project: SPDX Specification	3.0.1 stable; 3.1-RC1 under review	High	2026-10-22
OWASP CycloneDX: CycloneDX Bill of Materials Standard	1.7	High	2027-01-24
OpenTelemetry Project: OpenTelemetry Documentation and Specification	Rolling specification snapshot; profiles public alpha and declarative configuration stable in 2026	High	2026-10-22
Cloud Native Computing Foundation: Cloud Native Landscape and Project Documentation	Rolling catalog snapshot	Moderate	2026-08-23

Research limitation. Official documentation establishes published interfaces and declared behavior. It does not prove the performance, security, interoperability, cost, or operational suitability of a specific implementation. Those claims require controlled testing and retained evidence.