Defensible10

ISAUnited
INSTITUTE OF SECURITY ARCHITECTURE UNITED

Defensible 10

Annex H (Normative): D08-Monitoring, Detection & Incident Response Architecture

Technical Standards

Standards Committee
1-5-2026

About ISAUnited

The Institute of Security Architecture United is the first dedicated Standards Development Organization (SDO) focused exclusively on cybersecurity architecture and engineering through security-by-design. As an international support institute, ISAUnited helps individuals and enterprises unlock the full potential of technology by promoting best practices and fostering innovation in security.

Technology drives progress; security enables it. ISAUnited equips practitioners and organizations across cybersecurity, IT operations, cloud/platform engineering, software development, data/AI, and product/operations with vendor-agnostic standards, education, credentials, and a peer community—turning good practice into engineered, testable outcomes in real environments.

Headquartered in the United States, ISAUnited is committed to promoting a global presence and delivering programs that emphasize collaboration, clarity, and actionable solutions to today's and tomorrow's security challenges. With a focus on security by design, the institute champions the integration of security into every stage of architectural and engineering practice, ensuring robust, resilient, and defensible systems for organizations worldwide.

Disclaimer

ISAUnited publishes the ISAUnited Defensible 10 Standards Technical Guide to provide information and education on security architecture and engineering practices. While efforts have been made to ensure accuracy and reliability, the content is provided “as is,” without any express or implied warranties. This guide is for informational purposes only and does not constitute legal, regulatory, compliance, or professional advice. Consult qualified professionals before making decisions.

Limitation of Liability

ISAUnited - and its authors, contributors, and affiliates - shall not be liable for any direct, indirect, incidental, consequential, special, exemplary, or punitive damages arising from the use of, inability to use, or reliance on this guide, including any errors or omissions.

Operational Safety Notice

Implementing security controls can affect system behavior and availability. First, validate changes in non-production, use change control, and ensure rollback plans are in place.

Third-Party References

This guide may reference third-party frameworks, websites, or resources. ISAUnited does not endorse and is not responsible for the content, products, or services of third parties. Access is at the reader’s own risk.

Use of Normative Terms (“Must”, “Should”)

- Must: A mandatory requirement for conformance to the standard.
- Must Not: A prohibition; implementations claiming conformance shall not perform the stated action.
- Should: A strong recommendation; valid reasons may exist to deviate in particular circumstances, but the full implications must be understood and documented.

Acceptance of Terms

By using this guide, readers acknowledge and agree to the terms in this disclaimer. If you disagree, refrain from using the information provided.

For more information, please visit our Terms and Conditions page.

License & Use Permissions

The Defensible 10 Standards (D10S) are owned, governed, and maintained by the Institute of Security Architecture United (ISAUnited.org).

Standards Development & Governance Notice

This standard is one of the ten Parent Standards in the Defensible 10 Standards (D10S) series. Each Parent Standard is governed by ISAUnited's Standards Committee, peer-reviewed by the ISAUnited Technical Fellow Society, and maintained in the Defensible 10 Standards GitHub repository for transparency and version control.

Contributions & Collaboration

ISAUnited maintains a public GitHub repository for standards development. Practitioners may view and clone materials, but contributions require:

- ISAUnited registration and vetting
- Approved Contributor ID
- Valid GitHub username

All Sub-Standard contributions must follow the Defensible Standards Submission Schema (D-SSF) and are peer-reviewed by the Technical Fellow Society during the annual Open Season.

Obsolete and withdrawn documents should not be used; please use replacements.

Abstract

The ISAUnited Defensible 10 Standards provide a structured, engineering-grade framework for implementing robust and measurable cybersecurity architecture and engineering practices. The guide outlines the frameworks, principles, methods, and technical specifications required to design, build, verify, and operate reliable systems.

Developed under the ISAUnited methodology, the standards align with modern enterprise realities and integrate Security by Design, continuous technical validation, and resilience-based engineering to address emerging threats. The guide is written for security architects and engineers, IT and platform practitioners, software and product teams, governance and risk professionals, and technical decision-makers seeking a defensible approach that is testable, auditable, and scalable.

This document includes a series of Practitioner Guidance, Cybersecurity Students & Early-Career Guidance, and Quick Win Playbook callouts.

Practitioner Guidance- Actionable steps and patterns to apply the technical standards in real environments.

Cybersecurity Student & Early-Career Guidance- Compact, hands-on activities that turn each section's ideas into a small, verifiable artifact.

Quick Win Playbook- Immediate, evidence-driven actions that improve posture now while reinforcing good engineering discipline.

Together, these elements help organizations translate intent into engineered outcomes and sustain long-term protection and operational integrity.

Foreword

Message from ISAUnited Leadership

Cybersecurity is at a turning point. As digital systems scale, reactive and checklist-driven practices do not keep pace with adversaries. The ISAUnited position is clear: security must be practiced as engineered design, grounded in scientific principles, structured methods, and defensible evidence. Our mission is to professionalize cybersecurity architecture and engineering with standards that are actionable, testable, and auditable.

ISAUnited Defensible 10 Standards: First Edition is a practical framework for that shift. The standards in this book are not theoretical. They translate intent into measurable specifications, controls, and verification, and enable teams to design and operate resilient systems at enterprise scale.

About This First Edition

This edition publishes 10 Parent Standards, one for each core domain of security architecture and engineering. Sub-standards will follow in subsequent editions, contributed by ISAUnited members and reviewed by our Technical Fellow Society, to provide focused, technology-aligned detail. Adopting the Parent Standards now positions organizations for seamless integration of Sub Standards as they are released on the ISAUnited annual update cycle.

Why “Defensible Standards”

Defensible means the work can withstand technical, operational, and adversarial scrutiny. These standards are designed to be demonstrated with evidence, featuring clear architecture, measurable specifications, and verification, so that practitioners can confidently stand behind their designs.

Contents

Annex H (Normative): D08-Monitoring, Detection & Incident Response Architecture

ISAUnited's Defensible 10 Standards
Parent Standard: D08-Monitoring, Detection, and Incident Response Architecture
Document: ISAU-DS-MDIR-1000
Last Revision Date: January 2026
Peer-Reviewed By: ISAUnited Technical Fellow Society
Approved By: ISAUnited Standards Committee

Section 1. Standard Introduction

Monitoring, Detection, and Incident Response (MDIR) capabilities form the operational nervous system of a secure enterprise architecture, providing the continuous visibility, analytical depth, and automated responsiveness required to defend against modern cyber threats. As organizations extend across on-premises, cloud, and hybrid ecosystems, the complexity of correlating security telemetry, identifying advanced threats, and executing timely responses has grown exponentially. This evolution demands architectures that can unify Security Information and Event Management (SIEM), Security Orchestration, Automation, and Response (SOAR), Extended Detection and Response (XDR), and behavioral analytics into a cohesive, defensible system.

Traditional MDIR programs often emerge from ad hoc deployments of security tools, lacking the architectural integration, engineering rigor, and validation processes necessary for sustained effectiveness. This fragmented approach leads to inefficiencies, inconsistent threat detection, and delayed responses that adversaries can exploit. A defensible MDIR architecture requires deliberate engineering design, integration with enterprise-wide security controls, and continuous validation against adversarial tactics. This standard serves as the authoritative foundation for designing, implementing, and maintaining a measurable and resilient MDIR architecture. It is intended for cybersecurity engineers, architects, SOC managers, and technical leaders who seek to integrate advanced monitoring, precision detection, and orchestrated response capabilities into the enterprise security fabric. It provides guidance for unifying telemetry sources, applying intelligence-driven detection engineering, automating containment workflows, and ensuring operational sustainability through verifiable, repeatable engineering practices.

Objective

The objective of this standard is to establish foundational principles for Monitoring, Detection & Incident Response Architecture, guiding security professionals toward a structured, engineering-based approach for achieving continuous situational awareness, rapid threat detection, and effective incident response.

Define a rigorous Monitoring, Detection, and Incident Response architecture that:

1. Establishes unified, enterprise-wide telemetry collection across information technology, cloud, software as a service, and operational technology

environments, with defined coverage expectations and source onboarding governance.
2. Standardizes event normalization and enrichment so that correlation, analytics, and investigations operate on consistent fields and timestamps, with measurable parser health and ingestion quality objectives.
3. Engineers detection capability using adversary technique mapped correlations, behavioral analytics, and tuned anomaly methods, with measurable targets for detection fidelity and false positive control validated in Verification and Validation.
4. Orchestrates repeatable response through tested automation workflows and analyst-guided playbooks, with containment safety controls, rollback capability, and measurable mean time to respond performance.
5. Operationalizes threat intelligence by ingesting, correlating, and maintaining indicator and behavior updates that improve detection relevance, accelerate response, and prevent stale intelligence from degrading fidelity.
6. Protects the monitoring, detection, and incident response platform itself as a critical system by enforcing least privilege administration, hardened management boundaries, high availability and disaster recovery, and continuous health and drift monitoring with fail-closed behavior.
7. Produces audit-ready evidence by generating immutable logs, response traces, validation artifacts, and change records that support independent verification, incident forensics, and Evidence Pack traceability.

It addresses the full MDIR lifecycle, from telemetry collection and normalization through correlation and analytics to automated or guided response, ensuring that every function is measurable, integrated with the enterprise architecture, and capable of withstanding operational stress and adversarial testing. The focus is on achieving operational excellence by embedding efficiency-enhancing technologies such as SOAR, AI-driven anomaly detection, threat intelligence integration, and real-time orchestration into the MDIR design.

Justification

The modern threat landscape requires organizations to detect and contain threats within minutes, not days. Attackers are increasingly adept at exploiting gaps in monitoring coverage, bypassing static detection rules, and overwhelming manual response workflows. While frameworks such as NIST SP 800-61 and ISO/IEC 27035 provide high-level incident response guidance, they do not define the architectural integration, automation strategy, or measurable technical specifications needed to engineer MDIR systems for complex, distributed enterprise environments.

Obsolete and withdrawn documents should not be used; please use replacements.

Without a disciplined engineering approach, MDIR implementations suffer from alert fatigue, slow containment, and limited cross-domain visibility—conditions that lead to prolonged dwell times, higher-impact breaches, and operational instability. Critical challenges not fully addressed by compliance frameworks include cross-platform telemetry normalization, detection engineering aligned with MITRE ATT&CK, adaptive response orchestration across IT/OT/cloud environments, and continuous tuning to reduce false positives without compromising detection fidelity.

This standard closes that gap by defining a technically rigorous, security-by-design methodology for MDIR. It prescribes integrating SIEM, SOAR, XDR, User and Entity Behavior Analytics (UEBA), and threat intelligence platforms into a unified architecture that prioritizes automation, reduces operational friction, and enforces traceable engineering decisions.

By adopting this standard, organizations and academic programs can equip SOC teams, architects, and engineers with the structure, clarity, and measurable practices necessary to ensure that monitoring, detection, and response capabilities are not only practical but defensible, adaptable, and sustainable in the face of evolving threats.

Evidence

Evidence Packs (EPs) provide the proof layer for adopting this Parent Standard. For Domain 08, the Evidence Pack repository is EP-08 (D08) and is organized to mirror the sections that drive traceability and adoption:

- EP-08.1 Requirements (Inputs)
- EP-08.2 Technical Specifications (Outputs)
- EP-08.3 Foundational Standards
- EP-08.4 Control Mappings
- EP-08.5 Verification and Validation activities.

This structure links architectural intent in Section 5 to measurable implementation in Section 6, and then to Verification and Validation in Section 12, enabling organizations to demonstrate conformance through repeatable, time-bound artifacts rather than declarations.

Section 2. Definitions

These definitions ensure a consistent understanding and interpretation across ISAUnited members, implementers, and peer reviewers, supporting defensible engineering and implementation practices. Where possible, definitions align with industry-recognized terminology from NIST, ISO, and ISAUnited's internal frameworks and methodologies.

Adversary Simulation – Controlled exercises (for example, red team, purple team, or breach and attack simulation) that emulate real-world attack behaviors to validate detection fidelity and response effectiveness.

Alert Enrichment – The process of adding context to alerts or incidents, such as asset criticality, identity attributes, threat intelligence, and historical activity, to improve triage and response decisions.

Anomaly Detection – The use of statistical methods or machine learning to identify deviations from established baselines that may indicate malicious activity or abnormal system behavior.

Architecture Decision Record (ADR) – A structured engineering record that documents a security architecture decision, including the problem, options considered, constraints, decision rationale, and the planned tests and evidence.

Attack Surface Monitoring (ASM) – The continuous process of identifying, mapping, and tracking accessible assets and services that could be exploited by an adversary.

Breach and Attack Simulation (BAS) – Automated testing that simulates adversary techniques to validate detection and control effectiveness, often used for repeatable regression testing.

Chain of Custody – Documentation that proves the integrity and handling history of evidence artifacts from collection through storage and use in an investigation or audit.

CI/CD Pipeline – Continuous integration and continuous delivery practices that automate building, testing, and deploying code and configuration artifacts, including detections and playbooks managed as code.

Clock Synchronization – The enforcement of consistent system time across components, typically using NTP, to ensure accurate event ordering and defensible forensic timelines.

Correlation Rule – A defined logic set that links multiple related events or indicators to generate a higher-confidence alert or incident.

Obsolete and withdrawn documents should not be used; please use replacements.

Detection as Code – The practice of managing detection content, such as rules, correlations, and logic in version-controlled repositories with peer review and automated validation.

Detection Engineering – The process of designing, tuning, validating, and maintaining detection rules and analytics to improve fidelity, reduce false positives and false negatives, and align detections to adversary tactics and techniques.

Disaster Recovery (DR) – The ability to restore MDIR services and data after outage or failure, including planned failover, recovery procedures, and validated continuity targets.

Drift – Unauthorized or unplanned changes to configurations, parsers, rules, playbooks, or platform settings that can degrade detection reliability or response safety.

Elastic Common Schema (ECS) – A structured event and field naming convention that standardizes telemetry from multiple sources to support consistent parsing, enrichment, and correlation.

Evidence Pack (EP) – A structured collection of artifacts produced by the practitioner to prove conformance to the standard, linking Requirements, Technical Specifications, and Verification and Validation evidence.

Extended Detection and Response (XDR) – A detection and response capability that correlates telemetry across multiple security layers, such as endpoint, network, identity, email, and cloud, to support unified detection and response actions.

Fail Closed – A design behavior where a control or system defaults to denying action or raising an alert when a dependency fails, rather than allowing bypass or silent failure.

False Negative – A failure condition where malicious activity occurs, but the detection system does not generate an alert or incident for it.

False Positive Rate (FPR) – The percentage of alerts determined not to represent actual security incidents, affecting analyst workload and detection efficiency.

High Availability (HA) – The ability of MDIR platforms and services to remain operational through redundancy, fault tolerance, and tested failover designs.

Incident Response Playbook – A predefined set of technical and procedural steps for responding to specific incident types, designed to support consistent containment, eradication, recovery, and evidence capture.

Indicator of Compromise (IOC) – An observable artifact or signal that may indicate malicious activity, such as a hash, domain, IP address, process behavior, or account anomaly.

Ingestion Latency – The time between event generation at a source and its availability in the centralized telemetry platform for correlation and alerting.

Interface Control Document (ICD) – A structured specification that defines how systems exchange data, including required fields, formats, authentication, privileges, and fail-closed behavior.

Mean Time to Detect (MTTD) – The measured time between the start of an incident or malicious activity and the time the incident is detected.

Mean Time to Respond (MTTR) – The measured time between detection and containment, including response actions required to limit impact and restore normal operations.

Open Cybersecurity Schema Framework (OCSF) – An open event schema framework designed to standardize security telemetry fields across sources to support correlation, portability, and analytics.

Parser Health – A measurable indicator of log parsing correctness and completeness, including parse error rate, field completeness, and schema compliance.

Security Information and Event Management (SIEM) – A centralized platform that collects, normalizes, correlates, and analyzes telemetry to support detection, investigation, and audit requirements.

Security Orchestration, Automation, and Response (SOAR) – A capability that integrates tools and workflows to automate response tasks, orchestrate cross-platform actions, and enforce consistent containment procedures.

Security Telemetry – Data generated by security and IT systems, such as logs, alerts, events, and metrics that support monitoring, detection, investigation, and response.

Service Level Objective (SLO) – A measurable target for a system function such as ingestion latency, parser failure rate, or alert response timing, used to assess operational performance.

Tamper Evident Storage – Storage controls that make unauthorized modification detectable through immutability or integrity validation, such as append-only settings, hashing, or write-once controls.

Threat Hunting – A proactive security practice where analysts search for hidden threats and suspicious behaviors using hypothesis-driven investigations and intelligence.

Threat Intelligence Platform (TIP) – A capability that aggregates, enriches, manages, and distributes threat intelligence to improve detection relevance and response speed.

Time Synchronization (NTP) – The use of Network Time Protocol or equivalent mechanisms to ensure consistent time across systems for defensible sequencing of events.

User and Entity Behavior Analytics (UEBA) – Analytics that model baseline behavior for users and entities and identify anomalies that may indicate compromised accounts or insider threats.

Section 3. Scope

The Monitoring, Detection, and Incident Response (MDIR) architecture encompasses the frameworks, processes, and technologies that enable organizations to continuously monitor their environments, identify potential security incidents, and execute rapid, coordinated responses. As enterprises operate across increasingly complex, interconnected ecosystems that span on-premises infrastructure, multi-cloud platforms, SaaS services, and operational technology (OT) networks, the challenge of achieving unified visibility and timely threat mitigation has intensified.

This standard defines the architectural expectations and technical guardrails necessary to design, integrate, and sustain a defensible MDIR posture across the enterprise. It is designed to help practitioners establish real-time situational awareness, detect known and unknown threats, orchestrate automated and analyst-guided responses, and continuously validate detection efficacy while maintaining operational efficiency. Telemetry sources must adhere to structured schemas (Elastic Common Schema (ECS) / Open Cybersecurity Schema Framework (OCSF)) and documented ingestion SLOs or SLAs to maintain analytic fidelity.

Applicability

- **All Operational Domains**: Applies to IT, cloud, OT, and hybrid environments where security monitoring, threat detection, and incident response are critical to protecting business operations.

- **Enterprise, Government, and Academic Environments**: Intended for use by SOC teams, security architects, detection engineers, incident responders, and academic institutions advancing MDIR practices.
- **Cross-Domain Integration**: Addresses the architectural integration of SIEM, SOAR, XDR, UEBA, and threat intelligence platforms into a unified detection and response framework.

Key Focus Areas

- **Centralized Security Telemetry Management**: Aggregates and normalizes logs, alerts, and events from disparate systems into a unified architecture.
- **Advanced Threat Detection Engineering**: Utilizes behavioral analytics, correlation rules, machine learning, and MITRE ATT&CK-aligned detections to improve fidelity and reduce false positives.
- **Security Orchestration, Automation, and Response (SOAR)**: Automates containment, remediation, and workflow execution to reduce Mean Time to Detect (MTTD) and Mean Time to Respond (MTTR).
- **Threat Intelligence Integration**: Ingests, enriches, and operationalizes threat intelligence for proactive detection and faster response to emerging threats.
- **Proactive Threat Hunting**: Empowers analysts to identify stealthy threats and indicators of compromise before they escalate into full incidents.
- **Incident Response Coordination**: Aligns automated and manual playbooks to ensure consistent, repeatable, and auditable response actions.
- **Continuous Validation**: Validate detection accuracy through adversarial simulation and automated test generation.

Outcomes

MDIR operates as both a consumer of upstream identity, network, and data security telemetry and a producer of actionable intelligence and orchestrated responses for risk and governance processes. Its outcomes should be understood through the Defensible Loop (D-Loop):

- Define: Establish the monitoring scope, telemetry requirements, detection priorities, incident thresholds, and operational objectives needed to support enterprise visibility and response.
- Design: Architect normalized telemetry pipelines, detection logic, enrichment models, response playbooks, and supporting integrations so that MDIR functions as a unified and defensible system.
- Deploy: Implement and operationalize SIEM, SOAR, XDR, UEBA, threat intelligence integration, and supporting workflows across enterprise, cloud, SaaS, and operational technology environments.

Obsolete and withdrawn documents should not be used; please use replacements.

- Detect: Identify known and unknown threats through correlation, analytics, behavioral monitoring, threat hunting, and continuous observation of adversary activity across the environment.
- Defend: Contain, remediate, and recover through coordinated analyst guided and automated response actions that reduce dwell time, limit operational impact, and preserve critical services.
- Demonstrate: Produce measurable, time bound evidence of MDIR effectiveness through logs, metrics, validation results, incident records, and other traceable artifacts that support verification, auditability, and continuous improvement.

This scope establishes the foundation for building an MDIR capability that is not merely a collection of security tools, but an engineered architectural system aligned to Define, Design, Deploy, Detect, Defend, and Demonstrate. In this way, MDIR supports continuous visibility, rapid threat containment, defensible response, and measurable proof of operational effectiveness against evolving cyber threats.

Section 4. Use Case

Achieving resilient monitoring, detection, and response requires more than deploying tools—it demands engineered practice in real-world, hybrid enterprise environments. The following consolidated use case reflects a complex scenario typical of organizations operating across on-premises, multi-cloud, SaaS, and OT estates. It exposes common visibility and response gaps, ties detections to adversary behaviors (e.g., MITRE ATT&CK), and maps each weakness to targeted technical defenses grounded in SIEM, SOAR, XDR, and detection engineering. The outcome is an operational playbook that links day-to-day MDIR actions, collection, correlation, hunting, and orchestration, to measurable, defensible reductions in dwell time and impact.

Table H-1:

Use Case Name	Unified Threat Detection & Automated Response in Hybrid Enterprises
Objective	Achieve unified visibility, high-fidelity threat detection, forensic-grade telemetry retention, and automated incident response across a hybrid enterprise environment by integrating SIEM, SOAR, XDR, and UEBA into a cohesive MDIR architecture.
Scenario	

Use Case Name	Unified Threat Detection & Automated Response in Hybrid Enterprises
	A global financial services provider operating across on-premises data centers, multiple public cloud platforms, and SaaS applications experienced prolonged dwell time from a targeted ransomware campaign. The attack leveraged fileless malware, lateral movement through cloud workloads, and identity-based privilege escalation to evade detection. Security audits revealed fragmented monitoring capabilities, isolated detection rules tied to specific tools, and slow, manual incident response workflows that failed to contain the threat promptly.
Actors	SOC Manager, Threat Hunting Lead, Detection Engineer, Incident Response Lead, Threat Intelligence Analyst, Cloud Security Engineer, OT Security Specialist, Automation Engineer.
Challenges Identified	• Fragmented visibility with no centralized telemetry aggregation. • High false positive rate overwhelmed analysts with non-actionable alerts. • Manual triage and containment extended MTTR beyond acceptable thresholds. • No dynamic integration of threat intelligence into detection logic. • Absence of proactive threat hunting capabilities. • Inconsistent log retention and schema parity across platforms, reducing correlation accuracy.
Technical Solution	1) Unified SIEM Architecture: Centralized log and telemetry ingestion from IT, cloud, and OT environments with a normalized event field using ECS or OCSF for parsing and enrichment. 2) SOAR-Driven Automation: Integrated SOAR to trigger automated containment actions (host isolation, account lockdown) with MITRE ATT&CK-aligned playbooks and sandbox validation of playbook actions before production release. 3) XDR and UEBA Integration: Correlated endpoint, network, identity, and cloud workload telemetry; applied UEBA for insider threat detection. 4) Detection Engineering & Threat Intelligence: Tuned correlation rules to reduce false positives; integrated TIP feeds for real-time IOC updates. 5) Proactive Threat Hunting: Scheduled hunts for high-value assets and critical workloads; incorporated findings into new detection logic.
Expected Outcome	• Reduced MTTD by 80%, cutting detection time from hours to minutes. • Reduced MTTR to under 20 minutes via automated containment. • False positives reduced by 60%, improving analyst efficiency. • Dynamic threat coverage maintained through continuous intelligence integration. • Consistent monitoring and response coverage across IT, OT, cloud, and SaaS environments.

Key Takeaways

- A unified MDIR architecture must converge SIEM, SOAR, XDR, and UEBA into a single telemetry and automation ecosystem to achieve real-time situational awareness.
- Schema consistency (ECS / OCSF) and normalized log ingestion are prerequisites for high-fidelity correlation and analytics accuracy.
- SOAR playbooks and automation workflows must be validated in sandbox environments to prevent false or destructive actions during live incidents.
- Continuous adversarial simulation and breach-and-attack testing validate detection accuracy and ensure defined MTTD ≤ 10 minutes / MTTR ≤ 20 minutes targets are defensible.
- Threat intelligence integration should be bi-directional—enrich detections and feed confirmed incident data back into intelligence sources.
- Metrics such as dwell-time reduction, false-positive rate, and automation success rate must be measured and trended to prove MDIR maturity.
- MDIR engineering success depends on collaboration between security operations, cloud, DevOps, and identity teams to maintain end-to-end telemetry and response coverage.

Practitioner Guidance:

- Use this use-case model to benchmark your current detection and response capability.
- Begin by mapping your telemetry sources to confirm coverage across IT, OT, cloud, and SaaS.
- Validate that detection rules and automation playbooks align with MITRE ATT&CK tactics and have documented owners.
- Establish performance baselines (MTTD, MTTR, FPR) before any architectural change, then re-measure after SIEM/SOAR/XDR integration to quantify improvement.
- Maintain a living “evidence pack” of playbook test runs, detection validation results, and KPI trend reports to support verification & validation requirements in § 12.

Section 5. Requirements (Inputs)

To implement the Monitoring, Detection & Incident Response (MDIR) Architecture, the following baseline architectural, operational, and environmental conditions Must be in place. These inputs enable the defensibility and enforceability of the Technical Specifications (§ 6) and the subsequent sub-standards.

To implement the Monitoring, Detection & Incident Response (MDIR) Architecture, the following baseline architectural, operational, and environmental conditions Must be in place. These inputs enable the defensibility and enforceability of the Technical Specifications (§ 6) and subsequent sub-standards.

5.1 Centralized Telemetry Aggregation and Normalization
A SIEM, or equivalent centralized log platform, Must ingest, normalize, and correlate events from IT, OT, cloud, and SaaS sources. Parsers, field mappings, and enrichment Must follow structured schemas such as ECS or OCSF to preserve analytic fidelity. Parser health, ingestion latency, and schema compliance Must be monitored and reported.
An authoritative asset inventory and criticality context Must be available to enrich events and alerts (asset owner, environment, business criticality, internet exposure, and identity context where applicable).

5.2 Security Orchestration, Automation and Response Platform
A SOAR capability Must automate containment, remediation, and notification workflows integrated with SIEM, EDR or XDR, ticketing, and identity systems. Playbooks Must be version-controlled, peer-reviewed, and validated in sandbox environments before production release. Execution metrics (MTTR, success rate, rollback records) Must be captured for Verification and Validation (§ 12).

5.3 Extended Detection and Response Integration
XDR telemetry ingestion and correlation Must be implemented for endpoints, networks, identities, and cloud workloads, producing unified incidents where cross-domain correlation is required. Event exchange between XDR and SIEM or SOAR Must support end-to-end visibility and response workflows. Critical-severity detections should be verified using automated adversary simulation to confirm detection fidelity before production promotion.

5.4 User and Entity Behavior Analytics
Behavioral baselines Must exist for human and machine entities where identity misuse, privilege escalation, or insider threat scenarios are in scope. UEBA models should be retrained on a defined cadence and after major telemetry, rule, or environment changes. UEBA outputs Must be linked to response playbooks for credential misuse and anomalous access patterns.

5.5 Threat Intelligence Platform Integration
A threat intelligence capability Must ingest, enrich, and operationalize structured and unstructured intelligence. Indicators and adversary behaviors Must be correlated with internal telemetry and mapped to MITRE ATT and CK where applicable. Indicator lifecycle management Must include deduplication, expiry, and suppression of stale indicators to avoid fidelity loss. Bidirectional feedback, including confirmation signals from investigations back into intelligence stores, should be implemented when feasible.

5.6 Detection Engineering Framework
A documented detection engineering process Must govern creation, tuning, and validation of correlation rules, analytic models, and detection signatures. Detection content Must reside in version-controlled repositories as detection as code and Must be supported by automated testing pipelines. Ownership, review cadence, and performance objectives (including MTTD ≤ 10 min for critical alerts) Must be defined and tracked.

5.7 Forensic Grade Logging and Retention
All security events and alerts Must be centrally logged in tamper-evident, hash-verified storage for a minimum of 12 months, or as required by regulation. Clock synchronization across all components Must be enforced to ≤ 1 second. Log integrity Must be tested on a defined cadence, at least quarterly, using checksum and replay validation.

5.8 Incident Response Playbook Library
A documented and tested library of playbooks Must cover common and high-impact threats. Each playbook Must map to MITRE ATT and CK tactics and techniques where applicable and Must define escalation paths, containment actions, and required evidence artifacts. Playbooks Must be tested on a defined cadence, at least quarterly, and integrated with SOAR for automated or semi-automated execution where appropriate.

5.9 MDIR Platform Resilience and Self Protection
SIEM, SOAR, and XDR consoles Must operate on segmented management networks with MFA-protected administration and least privilege service accounts. High availability and disaster recovery configurations Must be implemented and tested on a defined cadence, at least quarterly. Health monitoring Must alert on parser failures, ingestion gaps, storage capacity, HA quorum loss, and log integrity violations.

5.10 Continuous Validation and Adversary Simulation
A continuous validation capability Must exist to test detection and response paths under realistic conditions. Breach-and-attack simulation (BAS), red team, or purple team exercises should run on a defined cadence and after material

changes to telemetry, rules, playbooks, or integrations. Failed validations Must trigger the rule returning or workflow correction before production deployment.

5.11 Metrics Ownership and Readiness Gates
Owners and targets Must exist for MTTD, MTTR, FPR, and playbook validation rates. A readiness gate checklist Must track each requirement (§ 5.1–5.10) with status and an evidence link. Baseline metrics Must be recorded before § 6 implementation and compared post-deployment to quantify improvement.

Evidence Pack

Evidence for Section 5 prerequisites Must be collected and maintained in EP-8.1 (Requirements). Each requirement in Sections 5.1 through 5.11 Must be supported by at least one dated artifact that identifies the owner, current status, and monitoring, integration, or enforcement boundary. Evidence Must be retained in a version controlled or otherwise traceable repository according to organizational audit and retention requirements.

Minimum evidence expectations for EP-8.1 include:

- Telemetry source inventory, onboarding status, and coverage records.
- SOAR playbook inventory, workflow ownership, and rollback readiness records.
- XDR and cross-domain integration records for endpoint, identity, network, and cloud telemetry.
- UEBA baseline scope, model ownership, and review cadence records.
- Threat intelligence feed governance, update cadence, and indicator lifecycle records.
- Detection engineering governance, testing path, tuning process, and ownership records.
- Forensic logging, retention, time synchronization, and integrity configuration records.
- Incident response playbook library, escalation path, and evidence capture records.
- MDIR platform protection, administrative access, health monitoring, and resilience records.
- Continuous validation planning, exercise ownership, and correction workflow records.
- Metrics ownership, readiness gate status, and baseline performance records.

EP-8.1 entries Must link forward to implementation evidence in EP-8.2 (Technical Specifications), supporting alignment evidence in EP-8.3 (Foundational Standards) where applicable, mapping support in EP-8.4 (Control Mappings) where applicable, and test results in EP-8.5 (Verification and Validation).

Practitioner Guidance:

- Validate unification before expansion: Confirm telemetry sources ingest to a single schema and time source before enabling advanced analytics or automation.
- Automate with safeguards: Sandbox SOAR playbooks and maintain rollback procedures to prevent disruption during containment.
- Version control detections: Treat correlation rules and playbooks as code with peer review and automated testing prior to deployment.
- Measure and trend: Track MTTD, MTTR, and FPR continuously; validate with BAS and adversary simulation to prove detection fidelity.
- Protect the platform: Harden MDIR systems as critical assets, enforce MFA, least privilege, HA, and DR testing, and immutable logging.
- Maintain traceable evidence: Archive configs, parser updates, playbook versions, and test results in tamper-evident repositories for audit and Verification and Validation (§ 12).

Section 6. Technical Specifications (Outputs)

Technical specifications define the concrete, defensible outputs that must be implemented to satisfy this standard. Each output is a required engineering area that transforms policy into measurable, actionable security outcomes. Together, these specifications establish a robust, resilient foundation for enterprise monitoring, detection, and response capabilities across on-premises, cloud, SaaS, and OT environments.

Outputs must be:

- **Measurable:** validated by scans, logs, audits, or tests
- **Actionable:** implementation-ready, not policy slogans
- **Aligned:** traceable to §5 Requirements and sub-standards

6.1 Centralized Telemetry and Log Management

Objective. Provide complete, normalized, and time-synchronized visibility across IT, OT, cloud, and SaaS environments.

- **Enterprise SIEM Deployment**
  Teams Must deploy a centralized SIEM, or equivalent platform, that ingests and correlates telemetry from all in-scope systems and security layers.
  Depends on: §5.1, §5.7

Obsolete and withdrawn documents should not be used; please use replacements.

Acceptance: ≥ 99 % of defined log sources onboarded; ingestion latency ≤ 5 minutes for critical sources.
Evidence: Ingestion dashboards, source coverage map, and latency metrics.

- **Data Normalization Schema**
  Teams Must enforce structured schemas (ECS or OCSF) for parsed events and enrichment fields. Parser health Must be monitored with alerting.
  Depends on: §5.1
  Acceptance: Parser failure rate < 0.5 % of daily events; schema compliance ≥ 98 %.
  Evidence: Parser logs, schema validation reports.
- **Telemetry Completeness and Critical Source Objectives**
  Teams Must define required log classes and required fields for crown jewel coverage, including identity, administrative actions, endpoint telemetry, DNS, proxy or egress signals, and cloud control plane events. Missing critical sources or missing required fields Must be detected and alerted.
  Depends on: §5.1
  Acceptance: Critical source availability ≥ 99.5 %; missing critical fields < 0.1 % of relevant events; ingestion gaps detected ≤ 15 minutes.
  Evidence: Source heartbeat reports, missing field dashboards, gap alerts.
- **Forensic Grade Retention and Integrity**
  Teams Must store logs in tamper-evident repositories using write once or append only controls for ≥ 12 months. Clock skew Must be enforced to ≤ 1 second across MDIR components.
  Depends on: §5.7
  Acceptance: Retention ≥ 12 months; integrity checks pass on defined cadence; time sync variance ≤ 1 second.
  Evidence: Integrity hash reports, NTP sync audits, retention configuration exports.

6.2 Threat Detection Engineering and Analytics
Objective. Engineer, validate, and continuously improve detection fidelity and coverage.

- **Correlation Rule Framework**
  Teams Must maintain MITRE ATT and CK aligned detections as code with peer review and automated testing pipelines.
  Depends on: §5.6
  Acceptance: Critical alerts MTTD ≤ 10 minutes; false positive rate < 10 % for critical alerts.
  Evidence: Rule repository logs, CI validation results, alert outcome samples.
- **Coverage Mapping and Gap Management**
  Teams Must maintain technique coverage mapping for priority threats and crown jewel use cases and Must track gaps in a managed backlog with owners and target dates.
  Depends on: §5.6, §5.11
  Acceptance: 100 % of priority techniques mapped; quarterly coverage review

Obsolete and withdrawn documents should not be used; please use replacements.

completed; open gaps have owner and due date.
Evidence: Coverage heat map, gap backlog export, review records.

- **Behavioral and Machine Learning Analytics**
  Teams Should deploy UEBA and anomaly analytics tuned to operational baselines, with retraining after major telemetry, rule, or environment changes.
  Depends on: §5.4
  Acceptance: Model performance metrics tracked and reviewed; drift detection in place; false positive impact measured.
  Evidence: Model training artifacts, drift reports, performance metrics.
- **Threat Hunting Playbooks**
  Teams Must publish hypothesis-driven hunt procedures targeting crown jewel assets and advanced adversary behaviors and Must convert validated findings into detections or response improvements.
  Depends on: §5.8, §5.10
  Acceptance: Hunt cadence met; validated findings produce detection or playbook updates on schedule.
  Evidence: Hunt reports, detection improvement diffs, updated playbook references.

6.3 Security Orchestration, Automation, and Response (SOAR)
Objective. Automate and validate incident response to reduce MTTR and human error.

- **Automated Containment Playbooks**
  Teams Must enable pre-approved workflows for endpoint isolation, account lockdown, and malicious IP or domain blocking for defined critical scenarios.
  Depends on: §5.2, §5.8
  Acceptance: MTTR ≤ 20 minutes for critical alerts where automation is authorized; success rate tracked.
  Evidence: SOAR execution logs, containment success metrics, and incident samples.
- **Human in the Loop Validation**
  Teams Must require analyst confirmation for high-impact actions and Must test playbooks in sandbox environments before production promotion.
  Depends on: §5.2
  Acceptance: 100 % high impact actions gated by approval; sandbox test evidence recorded before release.
  Evidence: Sandbox test artifacts, approval records, promotion logs.
- **Automation Safety and Blast Radius Controls**
  Teams Must implement blast radius limits, time-bounded containment, kill switches, and rollback procedures for automation failures.
  Depends on: §5.2, §5.9
  Acceptance: 100 % containment actions reversible or time-bound; kill switch tested quarterly; rollback success ≥ 99 % in drills.
  Evidence: Safety policy configurations, drill results, rollback records.

- **Case Management and Timeline Integrity**
  Teams Must maintain incident case records with defensible timelines, decision logs, and evidence pointers, including Evidence Pack linkage for critical incidents.
  Depends on: §5.7, §5.8
  Acceptance: 100% of critical incidents have timeline integrity and evidence linkage; chain-of-custody fields are present.
  Evidence: Case exports, timeline samples, evidence linkage logs.
- **Metrics Driven Optimization**
  Teams Must track execution times, failure modes, and rollback success rates, and Must feed findings into §12 Verification and Validation trend analysis.
  Depends on: §5.11
  Evidence: Metrics dashboards, post-change reviews, V and V cross-references.

6.4 Extended Detection and Response (XDR) Integration
Objective. Unify endpoint, network, identity, and cloud telemetry for cross-domain detection and automated containment.

- **Unified Correlation and Alert Enrichment**
  Teams Must integrate XDR feeds with SIEM and SOAR and enrich incidents with asset criticality, threat intelligence context, and historical incident data.
  Depends on: §5.3, §5.5
  Acceptance: Cross-domain incidents deduplicated and correlated; enrichment coverage tracked.
  Evidence: Alert enrichment samples, integration health logs, and incident correlation examples.
- **Identity Signal Enrichment**
  Teams Should enrich identity-driven incidents with identity and access context, such as privilege tier, recent elevation, token anomalies, device trust posture, and suspicious sign-in signals, where applicable.
  Depends on: §5.3, §5.4
  Acceptance: Identity enrichment present for ≥ 95 % of identity-driven incidents; missing enrichment auto-ticketed.
  Evidence: Enriched incident samples, enrichment completeness reports, ticket records.
- **Automated Remediation Hooks**
  Teams Must enable predefined remediation hooks for host quarantine, token revocation, or account disablement as appropriate, and Must record evidence for each action.
  Depends on: §5.3
  Acceptance: Action execution logged; failure paths documented; rollback available where applicable.
  Evidence: Remediation execution logs, Evidence Pack ID cross references, and rollback records.

6.5 Threat Intelligence Operationalization

Objective. Convert threat intelligence into immediate detection and response value.

- **TIP Integration and Correlation**
  Teams Must aggregate multiple feeds and Must deduplicate, normalize, and map indicators and adversary behaviors to internal telemetry.
  Depends on: §5.5
  Acceptance: Feed health ≥ 99 %; ingest to enrich latency ≤ 30 minutes.
  Evidence: Feed health dashboard, enrichment latency logs, mapping records.
- **Real-Time Detection Updates and Indicator Lifecycle**
  Teams Must push intelligence updates into SIEM, SOAR, and XDR without manual delay and Must expire stale indicators per policy.
  Depends on: §5.5, §5.6
  Acceptance: Indicator expiry enforced; duplicate indicators suppressed; update jobs meet latency objectives.
  Evidence: Update job logs, indicator expiry records, and deduplication reports.
- **Threat Actor Profiling and Sharing**
  Teams Should maintain profiles for priority adversaries and Should participate in ISAC or ISAO sharing where applicable to sector risk.
  Depends on: §5.5
  Evidence: Profile repository, sharing records, curation notes.

6.6 MDIR Platform Resilience and Self-Protection

Objective. Ensure the MDIR platform remains available, secure, and tamper-evident under stress or attack.

- **Administrative Access Hardening**
  Teams Must enforce MFA for all consoles and APIs, use least privilege service accounts with short-lived credentials, and log administrative actions to immutable storage.
  Depends on: §5.9
  Acceptance: MFA enforced; privileged actions logged; credential rotation evidence available.
  Evidence: Access policy exports, admin activity logs, and rotation records.
- **High Availability and Disaster Recovery (HA and DR)**
  Teams Must design for multi-zone redundancy and Must test failover quarterly without loss of detections or evidence.
  Depends on: §5.9
  Acceptance: Quarterly failover tests pass; detection and evidence continuity maintained.
  Evidence: Failover test reports, HA quorum alerts, continuity checks.
- **Health and Drift Monitoring**
  Teams Must alert on parser failures, ingestion gaps, and rule or playbook drift, and Must execute corrective automation where safe.
  Depends on: §5.9, §5.10
  Acceptance: Drift detected within defined SLO; corrections tracked;

Obsolete and withdrawn documents should not be used; please use replacements.

unauthorized changes escalated.
Evidence: Health dashboards, drift resolution logs, and change records.

- **Telemetry and Evidence Tamper Detection**
  Teams Must detect and alert on log deletion, retention policy changes, ingestion suppression attempts, and other tampering behaviors across critical sources and MDIR components.
  Depends on: §5.7, §5.9
  Acceptance: Tamper events alert ≤ 5 minutes; unauthorized pipeline or retention changes not undetected.
  Evidence: Tamper detection rules, alert samples, configuration diff logs.

Evidence Pack

Evidence for Section 6 technical specifications Must be collected and maintained in EP-8.2 (Technical Specifications). Each output in Sections 6.1 through 6.6 Must be supported by at least one dated artifact that demonstrates implementation, enforcement, and the applicable measurement point. Evidence Must be retained in a version controlled or otherwise traceable repository according to organizational audit and retention requirements.

Minimum evidence expectations for EP-8.2 include:

- Centralized telemetry, source onboarding, schema normalization, and log integrity implementation records.
- Detection engineering, correlation logic, coverage mapping, and tuning implementation records.
- SOAR workflow, containment automation, approval gate, and rollback implementation records.
- XDR integration, alert enrichment, and remediation hook implementation records.
- Threat intelligence ingestion, indicator lifecycle, and update propagation implementation records.
- MDIR platform protection, administrative hardening, resilience, and drift monitoring implementation records.
- Incident case management, evidence linkage, and timeline integrity records.
- Metrics, dashboards, and operational measurement records for detection and response performance.

EP-8.2 entries Must link back to EP-8.1 (Requirements) to demonstrate prerequisite readiness and Must link forward to EP-8.5 (Verification and Validation) for testing, validation, and formal acceptance results. Where applicable, entries may also support EP-8.3 (Foundational Standards) and EP-8.4 (Control Mappings).

Practitioner Guidance:

To ensure the successful implementation of these technical specifications:

- Establish integration sequence: Achieve telemetry unification (§5.1) before enabling automation (§5.2–§5.3).
- Validate automations safely: Sandbox all SOAR and XDR remediation playbooks before production to prevent service disruption.
- Treat detections as code: Maintain correlation rules in version-controlled repositories with peer review and automated testing.
- Measure continuously: Track MTTD, MTTR, and FPR per §12 V&V; feed results into rule tuning and playbook updates.
- Preserve evidence: Link each control change to an Evidence Pack ID and update the §12 matrix the same day the change ships.
- Harden the platform: Apply least privilege, MFA, HA and DR testing, and immutable logging per §5.9 and §6.6 to keep the MDIR system defensible.

Quick Win Playbook:

Title: SOAR Playbook Auto-Test Harness for Containment Workflows

Objective: Prevent production automation failures by enforcing repeatable, pre-deployment regression testing for containment playbooks, with measured pass-fail outcomes and auditable evidence tied to an Evidence Pack ID.

Target: Implement an automated pre-deployment test harness that validates top-priority SOAR playbooks (for example, ransomware containment, phishing response, and privileged account lockdown) before promotion to production (§6.3, §6.6).

Component/System: SOAR platform and associated CI/CD pipeline.

Protects: Response automation reliability and operational continuity.
Stops/Detects: Logic errors, mis-scoped isolation, broken integrations, and unintended service disruption caused by unvalidated playbooks.

Action:
• Integrate SOAR playbook repositories with the CI/CD system.
• Create automated test cases that replay synthetic alerts and verify expected outputs (ticket creation, endpoint isolation, notification workflow).
• Require all tests to pass before promotion to production; failed tests auto-block deployment.
• Log each test run and link outputs to §12 V&V evidence requirements.

	Proof (Evidence Pack EP-08.2): CI/CD job logs, test summary report, approval diff, and rollback validation records. **Metric**: 100 % of playbooks pass pre-deployment tests; rollback success ≥ 99 % in quarterly drills; zero production incidents attributable to automation failure. **Rollback**: Re-deploy the last validated playbook version; retain superseded test artifacts and approval records in the Evidence Pack.

Section 7. Cybersecurity Core Principles

The following ISAUnited Cybersecurity Core Principles are foundational to the design, implementation, and ongoing management of a secure Monitoring, Detection & Incident Response (MDIR) Architecture. Each principle guides architectural decisions, technical controls, and operational practices to ensure MDIR systems are resilient, measurable, and engineered to withstand real-world adversarial techniques.

Purpose and Function:

Security principles provide more than technical direction; they embed discipline, clarity, and foresight into every recommendation. By grounding technical specifications and implementation strategies in well-defined principles, ISAUnited ensures that sub-standards do not merely react tactically to incidents but are designed to sustain detection accuracy, operational efficiency, and response effectiveness over time.

Table H-2: Principles and MDIR Applicability:

Principle Name	Code	Applicability to Monitoring, Detection & Incident Response Architecture
Least Privilege	ISAU-RP-01	MDIR components, such as SIEM, SOAR, XDR, and UEBA, operate with the minimum access required to ingest telemetry, perform analytics, and execute response actions, reducing the risk of tool or credential compromise.
Zero Trust	ISAU-RP-02	All telemetry sources, automated actions, and analyst interventions are continuously authenticated, authorized, and verified before data ingestion or response execution—there is no implicit trust between integrated systems.

Principle Name	Code	Applicability to Monitoring, Detection & Incident Response Architecture
Complete Mediation	ISAU-RP-03	Every detection alert, correlation event, and automated response action must be validated against current policy, threat intelligence, and context before execution.
Defense in Depth	ISAU-RP-04	Multiple layers of detection and response—spanning endpoints, networks, cloud workloads, and identity systems—ensure no single point of detection or automation failure can blind the MDIR capability.
Secure by Design	ISAU-RP-05	Detection rules, correlation logic, and automation playbooks are built with security requirements embedded from inception, ensuring operational readiness and avoiding post-deployment retrofitting.
Minimize Attack Surface	ISAU-RP-06	MDIR systems limit externally exposed management interfaces, API endpoints, and integration channels to reduce the risk of direct compromise.
Secure Defaults	ISAU-RP-10	All detection rules, response playbooks, and integrations default to the most restrictive, secure settings, requiring explicit approval to relax them.
Evidence Production	ISAU-RP-15	MDIR systems generate immutable logs, forensic-quality evidence, and comprehensive audit trails of all detections, analyst actions, and automated responses, supporting investigations and compliance.
Make Compromise Detection Easier	ISAU-RP-16	System designs prioritize visibility, correlation, and analyst observability to speed the detection of compromise indicators and reduce attacker dwell time.
Protect Availability	ISAU-RP-20	MDIR systems, including SIEM and SOAR platforms, are designed for high availability, fault tolerance, and disaster recovery to ensure continuous threat monitoring and response capabilities.

Note: *Each principle is instantiated through Technical Specifications (§ 6), validated in Verification & Validation (§ 12), and supported by Security Controls (§ 9).*

Practitioner Guidance:

These principles must be embedded into all MDIR architectural decisions and technical implementations. They form the engineering foundation for all sub-standards developed under this Parent Standard, ensuring that every MDIR capability is not only operationally functional but also defensible by design. Implementers should consistently validate that each new detection rule, correlation model, or automated workflow aligns with these principles to maintain long-term resilience, adaptability, and auditability.

Section 8. Foundational Standards Alignment

Internationally recognized frameworks from NIST and ISO establish baseline expectations for logging, monitoring, detection, and incident-response management. Monitoring, Detection & Incident Response (MDIR) builds on these foundations, extending them into a defensible, engineering-based model that unifies telemetry, detection engineering, automation, and resilience across hybrid environments.

Purpose and Function

- Demonstrate alignment with globally accepted NIST/ISO practices for continuous monitoring, detection, and response.
- Bridge compliance-level guidance to ISAUnited's engineering methodology (§ 6 Telemetry, Detection Engineering, SOAR, TI Ops, Platform Resilience).
- Reinforce audit credibility and architectural consistency for sub-standard development and evidence mapping.
- Provide a stable baseline for clause-level traceability and version control in § 12 Verification & Validation.

Table H-3. Applicable Foundational Standards

Framework	Standard ID	Reference Focus
NIST	SP 800-53 Rev. 5	Security and privacy controls supporting MDIR, including Audit and Accountability, Incident Response, System and Information Integrity, and Security Assessment.

Obsolete and withdrawn documents should not be used; please use replacements.

Framework	Standard ID	Reference Focus
NIST	SP 800-61 Rev. 3	Incident response recommendations and life cycle guidance for detection, analysis, containment, eradication, and recovery.
NIST	SP 800-92	Log management foundations for designing log infrastructures, operational processes, and protection of log data.
NIST	SP 800-137	Information Security Continuous Monitoring strategy and program guidance for ongoing awareness and control effectiveness.
NIST	SP 800-137A	Continuous monitoring program assessment guidance to evaluate ISCM implementation and maturity.
NIST	SP 800-207	Zero Trust Architecture guidance for continuous verification, identity-anchored trust decisions, and transaction-level evaluation.
NIST	SP 800-160 Vol. 1 Rev. 1	Systems Security Engineering methods for engineering trustworthy, secure systems and verifiable security outcomes.
ISO/IEC	27001:2022	ISMS requirements establish governance expectations for monitoring, incident management, and continual improvement.
ISO/IEC	27002:2022	Control guidance for logging and monitoring and time synchronization, including controls 8.15 through 8.18.
ISO/IEC	27035-1:2023	Information security incident management principles and process model foundation for the ISO/IEC 27035 series.
ISO/IEC	27035-2:2023	Guidelines to plan and prepare for incident response and to learn lessons from incident response.
ISO/IEC	27035-3:2020	Guidelines for ICT incident response operations, including detection, triage, analysis, response, containment, and recovery.
ISO	22301:2019	Business continuity management system requirements supporting availability, resilience, and continuity expectations for MDIR operations.

Framework	Standard ID	Reference Focus

***NOTE**: ISAUnited Charter Adoption of Foundational Standards.*

Per the ISAUnited Charter, the institute formally adopts the International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) and the National Institute of Standards and Technology (NIST) as its foundational standards bodies, consistent with their public encouragement of organizational adoption. Parent Standards align with ISO/IEC and NIST for architectural grounding and auditability, and this alignment cascades down to Sub-Standards as invariant, minimum requirements that may be tightened but not weakened. ISAUnited does not restate or speak on behalf of ISO/IEC or NIST; practitioners shall consult the official publications and terminology of these organizations, verify scope and version currency against the latest materials, and implement controls in a manner consistent with ISAUnited security invariants and the requirements of this standard.

Sub-Standard Expectations

Sub-standards developed under ISAU-DS-MDIR-1000 must:

- Cite relevant clauses from Table H-3 (e.g., NIST 800-61 IR-1–4; ISO/IEC 27035-2 § 8 Incident Analysis) for every normative output they extend.
- Translate those clauses into testable engineering behaviors — policy-as-code or control-as-code with defined verification / validation steps in § 12.
- Document any intentional divergence with compensating controls and executive risk acceptance records; archive passing evidence under its Evidence Pack ID.
- Maintain a concise mapping table: § 6 Output to Framework / Clause to Test ID to Evidence Pack ID to ensure clause-level traceability.

Evidence Pack

Evidence for Section 8 foundational standards alignment Must be collected and maintained in EP-8.3 (Foundational Standards). Each NIST or ISO reference in Table H-3 Must be supported by at least one dated mapping artifact that identifies the applicable clause or practice, the affected Section 6 output, the implementation or enforcement reference, and the related Evidence Pack cross reference.

Minimum evidence expectations for EP-8.3 include:

- Clause level mapping records linking Section 6 outputs to applicable NIST and ISO clauses or practices in Table H-3.

Obsolete and withdrawn documents should not be used; please use replacements.

- Standards mapping records showing the affected MDIR capability, such as telemetry, detection engineering, SOAR, threat intelligence operations, or platform resilience.
- Version history records showing when mappings were created, updated, reviewed, or replaced.
- Change rationale records for any update to a mapped clause, citation, or alignment decision.
- Divergence records for any intentional deviation, including compensating controls, approval reference, review date, and validation method.
- Cross references to supporting implementation evidence in EP-8.2 and validation evidence in EP-8.5 where applicable.

EP-8.3 entries Must remain current. Any change to a detection rule, telemetry policy, automation workflow, retention setting, or resilience control that affects a mapped NIST or ISO reference Must update the alignment record in the same change cycle and include an updated evidence reference.

Practitioner Guidance:

- Map at clause level only: For each § 6 output (e.g., 6.1 Telemetry, 6.2 Detection Engineering, 6.3 SOAR, 6.5 TI Ops, 6.6 Resilience), record the applicable NIST/ISO clause and the control-as-code enforcement method, then link to its Evidence Pack ID.
- Maintain currency: When a detection rule, SOAR playbook, or policy changes, update its NIST/ISO citation concurrently and store the change diff with the Evidence Pack.
- Apply the strictest regime: If multiple clauses overlap (e.g., NIST AU-6 and ISO 27002 8.16), adopt the most stringent requirement and document the rationale once in the mapping sheet.
- Scope discipline: Keep foundational frameworks here; map MITRE ATT&CK, CSA, and CIS only within § 9 (Security Controls).
- Traceability to V&V: Ensure each clause mapping feeds directly into § 12 V&V to demonstrate alignment between governance standards, technical outputs, and defensible evidence.

Section 9. Security Controls

This section identifies the technical control families and control references directly supported or enforced by the Monitoring, Detection & Incident Response (MDIR) Architecture Parent Standard. These controls explicitly link architectural and

engineering guidance to recognized cybersecurity frameworks, ensuring traceability, auditability, and consistency of implementation across diverse enterprise environments.

Purpose and Function

Security controls translate the architectural intent of this standard into actionable, measurable safeguards. These controls provide the tactical foundation to enforce confidentiality, integrity, availability, authenticity, and auditability within MDIR systems — while also ensuring that the MDIR platform itself is hardened against compromise.

By mapping to CSA CCM, CIS Controls v8, and OWASP, ISAUnited ensures:

- Clear alignment with recognized industry best practices and regulatory compliance frameworks.
- Interoperability across diverse organizational contexts and technology stacks.
- Consistency and reusability of controls in sub-standards aligned to this Parent Standard, facilitating structured implementation and validation.

These mappings also enable engineers and auditors to explicitly measure and validate the defensibility of MDIR implementations guided by this standard.

Implementation Guidance

Sub-standard authors and practitioners must adhere to the following guidelines:

- Explicitly reference at least three technical controls from one or more authoritative cybersecurity frameworks.
- Provide the framework acronym, control ID, and concise description.
- Align selected controls explicitly with the technical specifications, outputs, and core security principles in this Parent Standard.
- Select concrete, implementation-level controls rather than high-level policy statements, ensuring actionable guidance.

Table H-4. Control Mappings for Monitoring, Detection & Incident Response Architecture:

Framework	Control ID	Control name and description	Aligns to §6
CSA CCM v4	LOG-02	Audit Log Protection: ensure audit logs are protected against unauthorized access, modification, or deletion, with defined retention and integrity controls.	6.1, 6.6

Framework	Control ID	Control name and description	Aligns to §6
CSA CCM v4	LOG-03	Security Monitoring and Alerting: monitor security-relevant events and trigger alerts based on those events and metrics.	6.1, 6.2, 6.4
CSA CCM v4	LOG-06	Clock Synchronization: Use a reliable time source across relevant systems to support accurate event correlation and forensics.	6.1, 6.6
CSA CCM v4	SEF-03	Incident Response Plans: Establish and maintain an incident response plan and supporting relationships, roles, and escalation paths.	6.3, 6.6
CSA CCM v4	SEF-04	Incident Response Testing: test and update incident response plans at defined intervals and after significant changes.	6.3, 6.6
CSA CCM v4	SEF-05	Incident Response Metrics: define and monitor them to improve performance and consistency.	6.3
CIS Controls v8.1	8.2	Collect Audit Logs: enable and collect audit logs from enterprise assets consistent with defined log requirements.	6.1
CIS Controls v8.1	8.9	Centralize Audit Logs: centralize audit log collection for correlation, analysis, and retention.	6.1
CIS Controls v8.1	8.10	Retain Audit Logs: retain audit logs for a defined period aligned to operational and compliance needs.	6.1
CIS Controls v8.1	8.11	Conduct Audit Log Reviews: Review audit logs to detect anomalies and indicators of compromise on a defined cadence.	6.2
CIS Controls v8.1	17.4	Establish and Maintain an Incident Response Process: Define roles, communication channels, and reporting requirements.	6.3, 6.6
CIS Controls v8.1	17.7	Conduct Routine Incident Response Exercises: conduct exercises for key personnel on a defined cadence to validate readiness.	6.3, 6.6

Framework	Control ID	Control name and description	Aligns to §6
OWASP Top 10 (2021)	A09	Security Logging and Monitoring Failures: Insufficient logging, alerting, and monitoring reduce the effectiveness of detection and response.	6.1, 6.2, 6.3
OWASP ASVS v4.x	V7	Error Handling and Logging: application logging requirements supporting monitoring, correlation, and incident investigation.	6.1, 6.2
OWASP Cheat Sheet	Logging	Logging guidance for event attributes needed for investigation and monitoring, including actor, action, target, timestamp, and outcome.	6.1

***NOTE**: Use of External Control Frameworks.*

ISAUnited maps to external control frameworks to provide alignment and traceability, but does not speak on behalf of those organizations. Practitioners shall consult and follow the official practices, recommendations, and implementation guidance of the Center for Internet Security (CIS), the Cloud Security Alliance (CSA), and the Open Worldwide Application Security Project (OWASP) when applying controls. Always verify control identifiers, scope, and version currency against the publishers' latest materials. Where wording differs, use the framework's official documentation while maintaining consistency with ISAUnited security invariants and this standard's requirements.

Additional References

As the MDIR domain matures or as additional authoritative frameworks become relevant, authors and contributors may include supplementary controls from:

- NIST SP 800-137 (Information Security Continuous Monitoring)
- ISO/IEC 27035 (Information Security Incident Management)

Sub-Standard Expectations

Sub-standards developed under the Monitoring, Detection & Incident Response Architecture Parent Standard are required to:

- Select and enforce explicit technical controls relevant to their targeted MDIR focus (e.g., SIEM configuration, SOAR automation, detection engineering, threat hunting, platform hardening).

Obsolete and withdrawn documents should not be used; please use replacements.

- Provide detailed mappings of these controls to defined validation, implementation, and operational criteria.
- Justify and document any deviation from control families referenced at this Parent Standard level, ensuring transparency and defensibility of any modifications or exceptions.

Evidence for Section 9 control mappings Must be collected and maintained in EP-8.4 (Control Mappings). Each control reference in Table H-4 Must be supported by at least one dated mapping record that identifies the framework, control identifier, related Section 6 output, implementation reference, and associated Section 12 verification and validation activity.

Minimum evidence expectations for EP-8.4 include:

- Controls to outputs mapping records linking each CSA CCM, CIS Controls, and OWASP control to one or more Section 6 outputs and the responsible owner.
- Control implementation reference records identifying where the mapped control is implemented, such as telemetry policy, detection logic, automation workflow, retention setting, alert enrichment, administrative hardening, or resilience control.
- Control review and update records showing when mappings were created, revised, reviewed, or replaced.
- Exception records for any deviation from a mapped control, including compensating measures, approval reference, review date, and evidence reference.
- Cross references to implementation evidence in EP-8.2 and validation evidence in EP-8.5 that demonstrate the mapped control in operation.

EP-8.4 entries Must remain current. Any change to a telemetry control, detection method, automation workflow, retention policy, or resilience safeguard that affects a mapped control Must update the control mapping record in the same change cycle and include an updated evidence reference.

Section 10. Engineering Discipline

This section defines the architectural thinking, rigorous engineering processes, and disciplined operational behaviors required to implement the Monitoring, Detection & Incident Response (ISAU-DS-MDIR-1000) standard.

ISAUnited’s Defensible Standards treat monitoring and response as engineered systems—grounded in systems thinking, critical reasoning, and Verification & Validation (V&V)—that produce measurable, auditable, and defensible outcomes across telemetry, detection, automation, and incident containment.

Obsolete and withdrawn documents should not be used; please use replacements.

10.1 Purpose & Function

Purpose. Establish a repeatable, auditable engineering system that integrates systems thinking, lifecycle control, adversary-aware design, and measurable outcomes for monitoring, detection, and response.

Function in D10S. Parent Standards define the invariants and expectations. Sub-Standards translate them into policies-as-code/controls-as-code, test specifications, and evidence artifacts that live within delivery and operations pipelines.

10.2 Systems Thinking

Goal: Make the MDIR system end-to-end legible—boundaries, data flows, trust relationships, interfaces, and dependencies—so that controls bind to where telemetry and automation risk actually occur.

10.2.1 System Definition & Boundaries

- Declare system scope, stakeholders, and in-/out-of-scope assets (SIEM, SOAR, XDR/EDR, UEBA, TIP, threat-hunting, telemetry collectors, automation engines, ticketing, evidence store; OT/ICS as applicable).
- Model trust zones and crossings (log source to collector, collector to SIEM, SIEM to SOAR/XDR, SOAR to target systems, analyst to console, automation to identity store).
- Define boundary invariants—e.g., *no unsigned telemetry ingestion*, *MFA + short-lived tokens for admin planes*, *no fail-open integrations.*

10.2.2 Interfaces & MDIR Contracts

- Maintain Interface Control Documents (ICDs) for telemetry ingestion, detection logic updates, playbook triggers, and evidence exchange.
- For each interface, specify: identity type (human vs service), privileges, data schema (ECS/OCSF), latency SLOs, retention, time-sync, fail-closed behaviors, and audit fields (event_id, source_id, rule_id, playbook_id, evidence_pack_id).

10.2.3 Dependencies & Emergent Behavior

- Map shared services (NTP, vault/keys, directory auth, CI/CD, evidence repo, network orchestration).
- Identify emergent risk from composition (e.g., parser failure + alert suppression to blind spot; automation loop + excessive privilege to self-inflicted outage; shared service failure to loss of detection fidelity).

10.2.4 Failure Modes & Safeguards

- For each critical path, document likely failures (parser error, telemetry gap, SOAR timeout, automation mis-scope, HA failover).
- Design safeguards (negative tests, alert on parser fail, transaction signing, quorum alerts, no fail-open on auth).

Required Artifacts (min): MDIR context diagram with trust boundaries; data-flow map; ICD set; invariants register.

10.3 Critical Thinking

Goal: Replace assumption-based configurations with explicit, reviewable reasoning that withstands adversarial pressure and audit scrutiny.

10.3.1 Decision Discipline

- Use Architecture Decision Records (ADRs): problem to options to constraints/assumptions to trade-offs to decision to invariants to test/evidence plan (who/when/how measured).

10.3.2 Engineering Prompts

- Boundaries – What telemetry boundaries exist and why? Which zones have explicit trust contracts?
- Interfaces – What invariants must always hold (auth, integrity, schema)? How are they tested?
- Adversary Pressure – Which ATT&CK techniques are credible here, and how are they detected or contained?
- Evidence – What objective signals prove the control works today and after change (MTTD, FPR, alert volume, parser pass rate)?
- Failure – When this fails, does it fail safe (alert vs silence)? What is the operator response path?

Required Artifacts (min): ADRs; assumptions/constraints log; evidence plan per decision.

10.4 Domain-Wide Engineering Expectations

Secure System Design

- Define MDIR boundaries (SIEM, SOAR, XDR, TIP, telemetry pipeline, evidence store).
- Validate boundaries and trust relationships via structured architecture reviews using § 10.2 artifacts.
- Ensure protections enforce the principle of least privilege, segmentation, and availability tiers aligned to MTTD/MTTR objectives.

Implementation Philosophy — "Built-in, not Bolted-on."

- Integrate telemetry onboarding, detection engineering, and response automation at design time.

- Express controls as policy-as-code or control-as-code bound to invariants (e.g., “no unsigned log feeds,” “automation must include rollback,” “alert pipeline must fail closed”).

Lifecycle Integration

- Embed MDIR controls and tests throughout design review, build, deploy, and operations.
- Use version-controlled repositories with required ADRs and Evidence Pack updates on every change.

Verification Rigor (V&V)

- Combine automated checks (parser health, alert latency, coverage %, HA failover) with targeted probes (red/purple tests, automation replay, noise injection).
- Require continuous validation in pipelines and runtime schedules tied to performance objectives (MTTD ≤ 10 min, MTTR ≤ 20 min, FPR < 10 %).

Operational Discipline

- Monitor for telemetry drift, rule staleness, feed latency, and unauthorized config changes; auto-remediate where safe with time-bounded exceptions.
- Maintain runbooks/SOPs for detection failures, automation rollback, HA failover, and incident timeline recording; log results to Evidence Pack.

10.5 Engineering Implementation Expectations

- Detections / Responses as Code. Store correlation rules, alert enrichments, and SOAR playbooks as signed artifacts in version control (Sigma, OpenC2, custom YAML).
- Structured Deployment Pipelines. Automate validation and promotion with CI/CD gates, rollback plans, and peer review records.
- Explicit Coverage Mapping. Maintain dashboards for telemetry coverage (IT/OT/cloud/SaaS), ATT&CK technique coverage, and automation trigger paths.
- Automated Testing & Negative Validation. Run simulated detections and automation replays before production; validate fail-closed behaviors and rollback success.
- Traceable Architecture Decisions. Link each change (ADR ID, Test ID, Evidence Pack ID) for audit continuity.

Required Artifacts (min): policy/control-as-code repos; enforcement/test gates; boundary ICDs; coverage metrics; automated test logs; evidence ledger (see § 12).

10.6 Sub-Standard Alignment (inheritance rules)
Sub-Standards must operationalize this discipline with MDIR-specific detail:

- **ISAU-DS-MDIR-1010 (SIEM Correlation Engineering)**
  Maintain correlation rules as code. Validate with synthetic events and coverage dashboards. Peer review all commits. Tie each release to an Evidence Pack ID.
- **ISAU-DS-MDIR-1020 (SOAR Automation and Playbooks)**
  Automate playbook testing through CI/CD. Simulate containment actions in sandbox environments. Validate escalation paths. Maintain rollback procedures and record results as evidence.
- **ISAU-DS-MDIR-1030 (Threat Intelligence Operations and Detection Fusion)**
  Validate TIP to SIEM integration and indicator lifecycle management. Automate expiry and deduplication. Capture evidence for feed health, propagation latency, and coverage impact.
- **ISAU-DS-MDIR-1040 (Detection Validation and Threat Hunting)**
  Maintain BAS and purple team tests on a defined cadence. Record coverage improvements and false-negative reductions in §12 metrics. Convert validated hunt findings into detections and playbook updates.
- **ISAU-DS-MDIR-1050 (Platform Resilience and Self Protection)**
  Conduct HA and DR tests. Verify tamper-evident logging and administrative MFA. Archive failover results and integrity checks to the Evidence Pack.

10.7 Evidence & V&V (what proves it works)

Establish an MDIR Evidence Pack for each environment containing:

- Design Evidence: Architecture diagrams, ICDs, invariant register, ADRs.
- Build Evidence: Detection/automation code history, schema tests, coverage maps, CI/CD results.
- Operate Evidence: Parser health reports, alert latency metrics, automation logs, HA tests, SOAR execution stats, and incident timelines.
- Challenge Evidence: BAS/purple team results, automation failure replays, rollback drills, resilience tests.

Each control defines objective pass/fail criteria, test frequency, responsible owner, and retention period. Map Evidence Pack IDs into § 12 traceability.

Practitioner Guidance:

- Maintain a living sheet mapping Controls to Outputs to Tests to Evidence; update with each code or policy change and attach proof (log coverage, BAS runs, parser diffs).
- Favor controls expressed and tested as code; time-bound exceptions must have compensating controls and explicit Evidence Pack IDs.
- Ensure V&V evidence is reviewed quarterly to validate continuous effectiveness and audit readiness.

Section 11. Associate Sub-Standards Mapping

Purpose of Sub-Standards

ISAUnited Defensible Sub-Standards are detailed, domain-specific extensions of the Monitoring, Detection & Response Architecture Parent Standard (ISAU-DS-MDIR-1000).

Each Sub-Standard delivers:

- Granular technical guidance tailored to specialized MDIR capabilities and operational functions.
- Actionable implementation strategies translating architectural intent into measurable detection and response controls.
- Precise validation methodologies ensuring outputs are defensible, auditable, and resistant to evasion.
- Alignment with foundational architectural principles, § 6 Technical Specifications, and § 10 Engineering Discipline.

Sub-Standards bridge the gap between the Parent's architectural vision and the detailed, testable technical requirements needed for robust engineering, validation, and auditing across detection, monitoring, automation, and incident-response workflows.

Scope and Focus of MDIR Sub-Standards

SIEM Architecture & Correlation Engineering
Example – *ISAU-DS-MDIR-1010: SIEM Architecture, Correlation Logic & Log Management*

- Defines onboarding, normalization, and enrichment standards for telemetry ingestion.
- Specifies MITRE ATT&CK-aligned correlation-rule engineering and tuning.
- Requires forensic-grade log retention and immutability ≥ 12 months.
- Implements automated rule-validation and coverage-testing pipelines.

SOAR Automation & Playbook Engineering
Example – *ISAU-DS-MDIR-1020: SOAR Workflow, Automation & Playbook Development*

- Requires modular, version-controlled playbooks for high-impact incidents.
- Mandates human-in-the-loop approval for high-risk containment actions.
- Enforces rollback and fail-safe mechanisms for automation failures.
- Integrates SOAR with SIEM, XDR/EDR, IAM, and ticketing systems for end-to-end orchestration.

Obsolete and withdrawn documents should not be used; please use replacements.

XDR & Cross-Domain Correlation
Example – *ISAU-DS-MDIR-1030: Extended Detection & Response (XDR) Integration*
- Unifies endpoint, network, identity, and cloud telemetry.
- Enriches alerts with asset criticality, threat-intelligence, and historical data.
- Implements automated remediation hooks and containment triggers.
- Deduplicates cross-domain alerts to minimize noise and analyst fatigue.

Threat Intelligence Operationalization
Example – *ISAU-DS-MDIR-1040: Threat Intelligence Integration, Enrichment & Automation*
- Ingests and enriches structured/unstructured feeds from multiple sources.
- Maps IoCs/TTPs to internal detection logic for proactive coverage.
- Automates intelligence push to SIEM/SOAR/XDR rulesets.
- Defines intelligence-sharing protocols with ISAC/ISAO partners.

Detection-as-Code & Continuous Validation
Example – *ISAU-DS-MDIR-1050: Detection-as-Code & Automated Detection Validation*
- Maintains version-controlled repositories for detection rules and analytics models.
- Automates validation using BAS/adversary-simulation tools.
- Maps detections to MITRE ATT&CK tactics and techniques.
- Performs continuous detection-health checks and false-positive trending.

Proactive Threat Hunting
Example – *ISAU-DS-MDIR-1060: Threat Hunting Methodologies & Integration*
- Defines hypothesis-driven hunting methodologies leveraging active TI.
- Integrates hunting toolsets with SIEM, EDR, and UEBA platforms.
- Requires hunts to generate new detections and automation logic.
- Implements coverage reporting and ATT&CK heat-map tracking.

Table H-5. Example Future Sub-Standards

Sub-Standard ID	Sub-Standard Name	Focus Area
ISAU-DS-MDIR-1010	SIEM Architecture, Correlation Logic & Log Management	SIEM & Correlation
ISAU-DS-MDIR-1020	SOAR Workflow, Automation & Playbook Development	SOAR Automation
ISAU-DS-MDIR-1030		XDR & Cross-Domain

Sub-Standard ID	Sub-Standard Name	Focus Area
	Extended Detection & Response (XDR) Integration	
ISAU-DS-MDIR-1040	Threat intelligence integration, Enrichment & Automation	Threat Intel Ops
ISAU-DS-MDIR-1050	Detection-as-Code & Automated Detection Validation	Detection-as-Code
ISAU-DS-MDIR-1060	Threat Hunting Methodologies & Integration	Threat Hunting

Note: Future identifiers under MDIR continue the 1xxx series to maintain consistency with ISAUnited numbering.

Development and Approval Process

ISAUnited uses an open, peer-driven annual process to propose, review, and publish sub-standards:

- Open Season Submission — Proposals must cite the §6 outputs and §7 principles they extend, plus clause-level NIST/ISO anchors from §8.
- Technical Peer Review — Evaluate engineering rigor, testability, scope clarity, and cross-domain consistency.
- Approval & Publication — Assign identifier/version and publish as an actionable extension of ISAU-DS-MDIR-1000.

Each Sub-Standard Will Specify

- **Inputs (Requirements)**: Preconditions and dependencies required for implementation (§ 5).
- **Outputs (Technical Specifications)**: Measurable engineering deliverables (§ 6).
- **Validation Methodologies**: Testing, V&V, and Evidence Pack linkage (§ 12).
- **Implementation Guidelines**: Scalable and secure deployment patterns aligned with § 10 Engineering Discipline.
- **Control Mappings**: Relevant § 9 Security Controls (CSA CCM, CIS v8, and OWASP).

Practitioner Guidance:

- Treat each Sub-Standard as a measurable extension of this Parent Standard; it inherits all § 10 Engineering Discipline requirements and § 12 V&V processes.
- Use Table H-4 to plan adoption, establish telemetry and automation foundations (1010–1020) before expanding into threat intelligence, validation, and hunting.
- Maintain a Mapping Sheet showing Parent Output to Sub-Standard Clause to Test Case to Evidence Pack ID.
- Synchronize evidence and control mappings whenever a Sub-Standard is updated to prevent drift across the Defensible Standards library.
- Engage the Technical Fellow Society annually to peer-review Sub-Standard implementations for interoperability and continued defensibility.

Section 12. Verification and Validation (Tests)

Purpose of This Section

This section outlines the structured evaluation methods necessary to ensure that the implemented MDIR controls, architecture, and engineering decisions align with the intent of this standard. It mandates measurable, repeatable testing procedures to confirm that the solution is technically defensible, resilient against compromise, and aligned with ISAUnited's engineering discipline.

> **Verification** confirms that the MDIR system has been implemented in accordance with the defined Requirements (Inputs) and Technical Specifications (Outputs) of this standard.
>
> **Validation** ensures that the MDIR system performs effectively under real-world operational conditions, produces reliable detections, executes responses accurately, and withstands adversarial testing, including attempts to compromise the MDIR platform itself.

Core Verification Activities

- Confirm that all MDIR technical controls defined in the Technical Specifications have been implemented in the production or target environment.
- Review and validate MDIR configuration baselines against engineering and security benchmarks (e.g., CIS Benchmarks for SIEM, SOAR, XDR platforms).

Obsolete and withdrawn documents should not be used; please use replacements.

- Verify interoperability between MDIR components (e.g., SIEM, SOAR, XDR, UEBA, TIP) to ensure no new vulnerabilities are introduced through integration.
- Conduct peer review of MDIR architectural artifacts, data flow diagrams, playbook logic, and detection rule mappings.
- Audit role-based access and privilege boundaries for MDIR administrative interfaces to ensure least-privilege operation.

Core Validation Activities

- Perform adversarial testing targeting MDIR components, including alert suppression attempts, detection evasion techniques, and SOAR playbook manipulation.
- Validate resilience against threat models such as log tampering, false-positive flooding, and credential compromise of MDIR service accounts.
- Test operational resilience, including MDIR system failover, redundancy failback, and response continuity during outages.
- Execute simulated incident scenarios (e.g., ransomware outbreak, insider data exfiltration, APT lateral movement) to assess detection coverage and response accuracy.
- Measure performance of MDIR controls against defined metrics (e.g., Mean Time to Detect (MTTD), Mean Time to Respond (MTTR), false positive/false negative rates, automation success rate).

Required Deliverables

All Verification & Validation efforts must produce documented outputs that include:

1. Test Plans & Procedures – Scope, objectives, tools, data, and simulation paths.
2. Validation Reports – Results, pass/fail, residual-risk ranking.
3. Evidence Artifacts – Logs, screenshots, incident timelines, audit outputs, hash proofs.
4. Corrective Action Plans – Remediation tasks with owner, timeline, and re-test criteria.
5. Evidence Pack Registry – Unique ID per test; cross-linked to § 5/§ 6 requirements and § 10.7 ledger.

Retention: ≥ 12 months post-test or until superseded by new validation.

Common Pitfalls to Avoid

- Configuration-only validation without end-to-end firing tests – Detections are “enabled” but never proven to fire using synthetic events or adversary simulation across SIEM, XDR, and SOAR paths.
- Telemetry blind spots hidden by dashboards - Coverage appears healthy, but critical sources are missing, parsers are broken, schemas drifted, or ingestion latency exceeds objectives, undermining correlation and timelines.
- Alert quality not measured – Teams tune to reduce noise but do not track false positives and false negatives separately, so detection fidelity degrades silently over time.
- Automation not tested under realistic conditions – SOAR playbooks are deployed without sandbox replay, rollback drills, or blast-radius controls, causing outages or evidence loss during containment.
- Evidence artifacts not defensible – Screenshots exist without immutable logs, time synchronization proof, hash verification, or chain-of-custody fields, making investigation and audit reconstruction unreliable.
- Fail-open behavior during component loss – Failover, degraded modes, or integration failures allow gaps such as missing logs, bypassed correlation, or unlogged admin actions instead of failing closed with alerts.
- No regression testing after change - Parser updates, rule changes, new integrations, or TI feed updates ship without re-running V&V tests, allowing drift and regressions to persist.
- MDIR platform not treated as a protected asset – Administrative access hardening, segmented management networks, and service-account privilege boundaries are not tested, leaving SIEM and SOAR planes exposed.

Table H-6. Traceability Matrix — Requirements (§5) to Verification/Validation (§12) to Related Technical Specs (§6):

Requirement ID	Requirement (summary)	Verification (build-correct)	Validation (works-right)	Related Technical Specs
5.1	Centralized telemetry aggregation	SIEM deployed; source onboarding list complete; parsers/normalization verified	Synthetic events from each source are ingested, normalized, and correlated end-to-end.	6.1
5.2	SOAR platform	SOAR integrated with SIEM/XDR/IAM/ticketing; playbooks version-controlled	Tabletop + sandbox fire playbooks; high-risk steps require approval; rollback succeeds.	6.3
5.3	XDR capability	XDR collecting endpoint/network/identity/cloud		6.4

Requirement ID	Requirement (summary)	Verification (build-correct)	Validation (works-right)	Related Technical Specs
		telemetry; alert enrichment enabled	Cross-domain detections are correlated into single incidents; automated hooks include hosts/accounts.	
5.4	UEBA	Baselines trained; anomalous behavior models deployed	Known insider/compromise scenarios trigger anomalies with acceptable FPR.	6.2
5.5	Threat intelligence integration	TIP feeds active; IOC/TTP mappings push to SIEM/SOAR/XDR	New TI updates raise detections within target latency; deprecated indicators expire.	6.5; 6.2
5.6	Detection engineering framework	ATT&CK-aligned rule catalog; testing pipeline in CI/CD; ownership defined	Red/purple/BAS validate priority detections; false positives reduced to target level.	6.2
5.7	Forensic-grade logging & retention	Tamper-evident storage; retention set; time sync/NTP verified	Incident reconstructions succeed; audit samples pass integrity checks.	6.1
5.8	IR playbook library	Playbooks documented, reviewed, and mapped to ATT&CK; escalation paths defined.	Live drills meet MTTD/MTTR targets; playbook steps execute without data loss.	6.3; 6.2
5.9	MDIR platform resilience & self-protection	Admin MFA enforced; segmented mgmt networks; HA topology documented; immutable log settings verified	Planned failover maintains detections and evidence; attempts to access admin paths without MFA are blocked and alerted; integrity checks succeed.	6.6; 6.1

How to use the matrix

- **Plan:** Map every §5 requirement to ≥1 verification and ≥1 validation tied to a §6 spec.
- **Execute:** Run activities and record an Evidence Pack ID per row.
- **Maintain:** When requirements/controls/specs change, update tests/evidence and re-validate priority detections and playbooks.

Evidence Pack

Evidence for Section 12 verification and validation activities Must be collected and maintained in EP-8.5 (Verification and Validation). Each verification and validation activity Must produce at least one dated artifact that demonstrates execution, result, pass or fail status, and the applicable metric or acceptance point. Evidence Must be retained in a version controlled or otherwise traceable repository according to organizational audit and retention requirements.

Minimum evidence expectations for EP-8.5 include:

- Test plans and procedure records covering scope, objectives, tools, simulation path, frequency, owners, and pass or fail criteria.
- Verification result records confirming implementation of telemetry, detection, orchestration, logging, resilience, and administrative protection controls.
- Validation reports from adversarial testing, simulated incident scenarios, automation exercises, and resilience testing.
- Metrics records showing Mean Time to Detect, Mean Time to Respond, false positive and false negative trends, and automation success rates.
- Evidence artifacts such as logs, screenshots, incident timelines, audit outputs, parser results, and hash or integrity proofs.
- Corrective action and retest records showing remediation owner, completion target, and revalidation outcome.
- Evidence Pack registry and traceability records linking each test activity to the applicable Section 5 requirement, Section 6 output, and Table H-6 reference.

EP-8.5 entries Must link back to EP-8.2 (Technical Specifications) for implementation evidence and to the Section 12 traceability matrix for requirement, test, and evidence alignment.

Practitioner Guidance:

- Simulate realistic attack conditions—not configuration-only checks—to test end-to-end detection and automation paths.
- Integrate MDIR V&V into CI/CD pipelines: each pull request that touches detections or playbooks must include updated matrix rows, test evidence, and pass/fail results.
- Reject any merge lacking a valid Evidence Pack ID and measurable success criteria.
- Re-run validation after every material change to ensure no regression in MTTD/MTTR or automation success.
- Use continuous validation dashboards to trend detection health, alert fidelity, and automation reliability over time.

Quick Win Playbook:

Title: Daily Detection Validation Run for Priority Techniques

Objective: Establish repeatable daily proof that priority detections still fire as intended after rule changes, parser updates, schema modifications, or telemetry source drift, and that detection regressions are surfaced within 24 hours.

Target: Execute a daily automated validation run against the top 10 critical detections aligned to priority ATT&CK techniques (§6.2, §6.3; validated in §12).

Component/System: CI or scheduled validation runner integrated with the SIEM and SOAR test environment.

Protects: Detection fidelity and correlation reliability by catching regressions before they become operational blind spots.

Stops or Detects: Inactive rules, mis-parsing, schema drift, and false negative conditions.

Action:
• Schedule ATT&CK-aligned adversary simulation tests daily for the priority detection set.
• Compare expected alerts to observed alerts and correlation output.
• Auto-file tickets for any missed detections and block promotion of related changes until corrected.

Proof (Evidence Pack EP-08.1): Test run outputs, alert and correlation logs, detection coverage report, and remediation pull request or change record.

	Metric: 100 % of priority detections tested daily; broken rule discovery time ≤ 24 hours; false negative rate trends down quarter over quarter. **Rollback:** Revert to the last validated rule set and retain superseded artifacts and diffs in the Evidence Pack.

Section 13. Implementation Guidelines

This section does not prescribe vendor-specific tactics. Parent Standards are stable, long-lived architectural foundations. Here we define how sub-standards and delivery teams Must translate the Parent's intent (ISAU-DS-MDIR-1000) into operational behaviors that are testable, automatable, and auditable for Monitoring, Detection, and Incident Response (MDIR).

Delivery mechanics (pipeline orchestration, artifact signing/attestation, promotion/rollback) are governed by Annex J.

Purpose of This Section in Sub-Standards

Sub-standards Must use Implementation Guidelines to:

- Translate Parent expectations into enforceable MDIR behaviors (parser health gates, detection coverage SLOs, ATT and CK mapping completeness, automation rollback proofs, immutable evidence policies).
- Provide stack-agnostic practices that improve adoption, reduce failure, and align with ISAUnited's defensible by design philosophy.
- Highlight common failure modes and prevent them through measurable gates and automated tests.
- Offer repeatable patterns, expressed as code, that enforce telemetry fidelity, detection accuracy, orchestration safety, and engineering discipline across SIEM, SOAR, XDR, UEBA, TIP, and evidence repositories.

Open Season Guidance for Contributors

Contributors developing sub-standards Must:

- Align all guidance with the Parent's strategic posture and §6 outputs (MTTD ≤ 10 min; MTTR ≤ 20 min; FPR < 10 % for critical alerts; no fail open ingestion; sandbox validated playbooks).
- Avoid vendor and product names and express controls as requirements, tests, and evidence linked to an Evidence Pack ID.

Obsolete and withdrawn documents should not be used; please use replacements.

- Include lessons learned (what fails, why it fails, and how the test proves correction).
- Focus on repeatable engineering patterns (policies as code and controls as code).
- Provide a minimal Standards Mapping: Spec or Control to NIST or ISO clause (from §8) to Evidence Pack ID (keep CSA, CIS, and ATT and CK mapping in §9).

Technical Guidance

A. Organizing Principles (normative)

1. Everything as Code - Detection rules, correlation logic, playbooks, parser configurations, SOAR workflows, and XDR and TI enrichment policies Should be version-controlled, peer reviewed, and released on protected branches.
2. Non-Bypassable Security Gates - Every merge or release Should pass gates bound to §6 and §12 objectives, including:
   - Parser failure < 0.5 % and ingestion latency ≤ 5 min for critical sources.
   - Critical detections MTTD ≤ 10 min and FPR < 10 % for critical alerts.
   - SOAR rollback success ≥ 99 % for playbooks that can impact availability or evidence.
   - Evidence Pack ID present for every code change.
3. Immutable, Reproducible Releases - Manual configuration changes post-build Should Not be the operating model. Rules, playbooks, and schemas Should be signed and pinned. Deployment Should verify integrity at runtime.
4. Least Privilege and Separation of Duties (MDIR context) - Distinct identities for detection engineering, automation, and validation pipelines Should be enforced. Secrets Should be vaulted and rotated. Any SoD violation Should generate a release blocking alert.
5. Environment Parity - Staging environments Should mirror production for telemetry schemas, detection logic, automation flows, and TI feeds. Drift Should be monitored and reconciled before promotion.

B. Guardrails by Pipeline Stage (normative)

1. **Pre-Commit and Local**
   • Signed commits and secrets scanning Should run by default.
   • Lint correlation rules and playbooks. Unmapped ATT and CK identifiers or undocumented detections Should be rejected.
   • Synthetic test stubs Should be created for new rules and playbooks.
2. **Pull Request and Code Review**
   • CODEOWNERS approval Should be required for rule and playbook changes.
   • Coverage gates Should verify that changed detections fire in sandbox or test conditions.
   • The pull request Should include planned §12 Test IDs and an Evidence Pack ID stub.

Obsolete and withdrawn documents should not be used; please use replacements.

3. **Build and Package**
   • Deterministic, signed rule and playbook bundles Should be produced.
   • Corresponding BAS and validation suites Should be packaged for modified detections.
4. **Pre-Deploy and Release**
   • Drift checks Should compare deployed schemas and policies to approved baselines.
   • Canary rollout Should be used for detection bundles and automation workflows with health SLOs and auto rollback.
   • Positive and negative tests Should include parser health, alert latency, automation rollback, and schema validation.
5. **Deploy and Runtime**
   • MTTD and MTTR SLOs and SoD rules Should be enforced.
   • Unapproved log sources or automations lacking Evidence Pack linkage Should be blocked.
   • SOAR safety monitors and XDR alert integrity checks Should run continuously.
6. **Post-Deploy Validation and Operations**
   • Continuous validation (BAS and purple testing per §12) Should be integrated into operations.
   • Security SLOs Should be tracked: MTTD ≤ 10 min; MTTR ≤ 20 min; automation success ≥ 95 %.
   • Evidence Packs Should be generated per release, including policy diffs, validation results, and rollback records.

C. Identity, Access, and Secrets (normative alignment to §6.1–§6.6)

- Dedicated service identities for SIEM, SOAR, XDR, and TI integrations Should be used. mTLS and signed tokens Should be implemented for API exchanges where feasible.
- Secrets Should be managed by approved vault services with audit logging and rotation ≤ 90 days.
- Telemetry records Should include trace_id, rule_id, policy_version, and timestamp for end to end forensic traceability.

D. MDIR Supply-Chain Integrity (normative; mechanics in Annex J)

- Only signed rule and playbook packages that passed §12 tests Should be deployed. Artifact sources and namespaces Should be restricted.
- Unverified plugins or feeds Should be quarantined until validated. Integrity and license checks Should be enforced where applicable.
- Build and deploy identities Should be separated. Production writes from build jobs Should Not be permitted. Tamper events Should be treated as release blocking.

E. Measurement and Acceptance (aligned to §6 and §12)

Implementers Should define acceptance criteria that can be tested and evidenced. At minimum:

- **Telemetry Integrity**: Parser failure < 0.5 %; ingestion latency ≤ 5 min for critical sources; time sync ≤ 1 second.
- **Telemetry Completeness**: Critical source availability ≥ 99.5 %; missing critical fields < 0.1 % of relevant events; ingestion gaps detected ≤ 15 min.
- **Detection Fidelity**: MTTD ≤ 10 min for critical detections; FPR < 10 % for critical alerts; coverage mapping for priority ATT and CK techniques maintained.
- **Automation Safety**: MTTR ≤ 20 min where automation is authorized; rollback success ≥ 99 % in quarterly drills; blast radius limits and kill switches validated.
- **Resilience and Availability**: HA failover tests quarterly with no evidence loss; administrative MFA enforced; tamper detection alerts ≤ 5 min.
- **Evidence Completeness**: Every change links §5 to §6 to §12 via Evidence Pack ID; Test IDs and outcomes recorded for each release.

Common Pitfalls (and the Engineered Countermeasure)

1. Unowned or stale detections are prevented by an ownership dashboard with expiry dates and Evidence Pack linkage (see §13 Quick Win).
2. Schema drift or broken parsers are prevented by a parser health monitor that blocks release when failure exceeds 0.5 %.
3. Untested automation is prevented by mandatory sandbox replay and rollback verification before promotion.
4. Evidence gaps are prevented by blocking merges that lack an Evidence Pack ID and a §12 Test ID reference.
5. Separation of duties collapse is prevented by distinct pipelines and an alertable SoD monitor that blocks release when identity overlap is detected.

Practitioner Guidance:

- Embed these practices in CI/CD so configuration, validation, and evidence capture run by default, not by exception.
- Maintain a living traceability sheet: Controls to Outputs to Tests to Evidence (§ 12), updated in the same change that modifies rules or playbooks.
- Run operational cadence checks: weekly detection health review and quarterly automation safety drills, with documented outcomes.
- Capture lessons learned and feed them into Open Season submissions so sub-standards mature from field evidence, not opinion.

Obsolete and withdrawn documents should not be used; please use replacements.

Quick Win Playbook:

Title: Detection and Playbook Ownership Dashboard

Objective: Eliminate unowned, stale, or undocumented detections and playbooks by enforcing ownership, review cadence, expiry dates, and evidence linkage that support § 12 Verification and Validation.

Target: Deploy an ownership and lifecycle dashboard that ties every active detection rule, correlation, and playbook to an accountable owner, review cadence, and expiry date (§ 6.2, § 10.4, § 13.3).

Component/System: SIEM or detection management repository (dashboard, spreadsheet, or Git-based metadata file).

Protects: Detection fidelity and analyst efficiency by preventing rule rot and undocumented response logic.
Stops or Detects: Unowned detections, expired detections, missing ATT&CK mapping, and playbooks lacking validation evidence.

Action:

1. Export the current detection and playbook inventory from the system of record.
2. Add required metadata fields: owner, created_date, last_review, expiry_date, ATT&CK_ID, Evidence_Pack_ID.
3. Publish status views (green current, yellow review due, red expired or unowned).
4. Send a weekly report to SOC leadership and auto-create tickets for every red item until resolved.

Proof (Evidence Pack EP-08.1): Dashboard screenshot, inventory export, remediation ticket log, and commit history showing metadata enforcement.

Metric:
• 100 % of detections and playbooks have assigned owners and Evidence Pack IDs.
• 0 expired detections older than 90 days.
• ≥ 95 % review completion rate per quarter.

Rollback: Restore the previous inventory snapshot (read-only) and retain superseded artifacts in the Evidence Pack.

Appendices

Appendix A: EP-08 Engineering Traceability Matrix (ETM)

This Engineering Traceability Matrix (ETM) connects the Monitoring, Detection & Incident Response Architecture Parent Standard requirements to measurable technical specifications, cybersecurity core principles, control mappings, and Verification and Validation activities. It gives practitioners a single, structured view of what must be established, implemented, how conformance is evaluated, and how supporting evidence is maintained. The ETM also reinforces flow down discipline by showing how Parent Standard requirements translate into enforceable outputs, traceable alignment, and testable acceptance evidence.

Evidence Pack alignment: Evidence supporting this ETM is organized across the five D08 Evidence Pack locations. For each matrix row, primary acceptance evidence is maintained in EP-8.5 (Verification and Validation), with supporting artifacts referenced from EP-8.1 (Requirements), EP-8.2 (Technical Specifications), EP-8.3 (Foundational Standards), and EP-8.4 (Control Mappings).

Req ID	Requirement (Inputs) (§5)	Technical Specifications (Outputs) (§6)	Core Principles (§7)	Control Mappings (§9)	Verification – Build Correct (§12)	Validation – Works Right (§12)	Evidence Pack ID
5.1	Centralized telemetry aggregation and normalization	§6.1 Centralized Telemetry and Log Management	RP-16 Make Compromise Detection Easier; RP-15 Evidence Production; RP-05 Secure by Design	CSA CCM LOG-03; CIS 8.2, 8.9; OWASP A09	SIEM deployed; onboarding list complete; parsers and schema checks in place; ingestion latency measured	Synthetic events ingested end-to-end; correlation confirms visibility across critical sources; gap alerts fire on missing sources	EP-08.0 / EP-08.5
5.2	SOAR platform and workflow integration	§6.3 SOAR	RP-03 Complete Mediation; RP-10 Secure Defaults; RP-20 Protect Availability	CSA CCM SEF-03, SEF-04, SEF-05; CIS 17.4, 17.7	SOAR integrated with SIEM, XDR, IAM, and ticketing; playbooks version controlled; sandbox tests pass	Tabletop and sandbox runs execute safely; approvals enforced; rollback succeeds; MTTR performance validated	EP-08.2
5.3	XDR integration and cross-	§6.4 XDR Integration	RP-04 Defense in	CSA CCM LOG-03;	XDR feeds active;		EP-08.3

Req ID	Requirement (Inputs) (§5)	Technical Specifications (Outputs) (§6)	Core Principles (§7)	Control Mappings (§9)	Verification – Build Correct (§12)	Validation – Works Right (§12)	Evidence Pack ID
	domain correlation		Depth; RP-16 Make Compromise Detection Easier	CIS 8.9; OWASP A09	deduplication enabled; enrichment fields configured; evidence logging enabled	Cross-domain incidents correlate into single cases; remediation hooks execute and record evidence; identity enrichment completeness is measured	
5.4	UEBA baselines and anomaly analytics	§6.2 Detection Engineering and Analytics	RP-02 Zero Trust; RP-04 Defense in Depth; RP-16 Make Compromise Detection Easier	OWASP A09; CIS 8.11	UEBA baselines trained; detection outputs integrated into case flow; model drift monitoring configured	Known insider and credential misuse scenarios trigger with acceptable FPR; anomalies link to response playbooks and case timelines	EP-08.1
5.5	Threat intelligence integration	§6.5 Threat Intelligence Operationalization	RP-03 Complete Mediation; RP-05 Secure by Design; RP-16 Make Compromise Detection Easier	CSA CCM LOG-03; CIS 8.11; OWASP Logging guidance	Feed health validated; dedupe and expiry rules active; mapping to telemetry fields confirmed	Indicators and TTP updates propagate within target latency; stale indicators expire; enrichment improves detection or triage outcomes	EP-08.4
5.6	Detection engineering framework	§6.2 Detection Engineering and Analytics	RP-05 Secure by Design; RP-15 Evidence Production; RP-10 Secure Defaults	CSA CCM LOG-03; CIS 8.11; OWASP A09	Rules as code repository present; peer review gates enforced; CI tests pass; ownership and review cadence documented	BAS and purple exercises validate priority detections; tuning reduces false positives without degrading coverage	EP-08.1

Req ID	Requirement (Inputs) (§5)	Technical Specifications (Outputs) (§6)	Core Principles (§7)	Control Mappings (§9)	Verification – Build Correct (§12)	Validation – Works Right (§12)	Evidence Pack ID
5.7	Forensic-grade logging and retention	§6.1 Telemetry and Log Management; §6.6 Platform Resilience	RP-15 Evidence Production; RP-20 Protect Availability	CSA CCM LOG-02, LOG-06; CIS 8.10; OWASP Logging guidance	Tamper-evident storage enabled; retention set; time sync verified; integrity hash tests pass	Incident reconstruction succeeds from immutable logs; audit samples validate integrity and timeline accuracy	EP-08.5
5.8	Incident response playbook library	§6.3 SOAR; §6.2 Detection Engineering	RP-03 Complete Mediation; RP-04 Defense in Depth	CSA CCM SEF-03, SEF-04; CIS 17.4, 17.7	Playbooks documented, reviewed, mapped to ATT&CK; escalation paths defined; evidence fields defined	Live drills meet MTTD and MTTR targets; playbook steps execute without evidence loss; case timelines complete	EP-08.2
5.9	Platform resilience and self-protection	§6.6 Platform Resilience and Self-Protection	RP-01 Least Privilege; RP-06 Minimize Attack Surface; RP-20 Protect Availability	CSA CCM LOG-02, LOG-06; CIS 17.4	MFA enforced; management segmentation documented; HA topology validated; drift monitors configured	Planned failover preserves detections and evidence; unauthorized admin access attempts are blocked and alerted; tamper alerts fire	EP-08.5
5.10	Continuous validation and adversary simulation	§6.2 Detection Engineering; §6.3 SOAR	RP-15 Evidence Production; RP-16 Make Compromise Detection Easier	CIS 17.7; CSA CCM SEF-04, SEF-05	Validation cadence defined; test runners configured; evidence capture paths defined	Daily or scheduled adversary simulation validates top detections; regressions auto-ticketed; false negative trend tracked	EP-08.1
5.11	Metrics ownership and readiness gates	§6.1–§6.6 (all outputs)	RP-05 Secure by Design; RP-	CIS 8.11; CSA CCM SEF-05	Owners, thresholds, and dashboards	Post-change KPI	EP-08.0

Req ID	Requirement (Inputs) (§5)	Technical Specifications (Outputs) (§6)	Core Principles (§7)	Control Mappings (§9)	Verification – Build Correct (§12)	Validation – Works Right (§12)	Evidence Pack ID
			15 Evidence Production		defined; readiness checklist exists; baseline captured	comparisons show improvement or regression; corrective actions tracked and closed	

Notes

- Sub-EP entries represent future IAM sub-standards to be developed; each will inherit this EP structure and include §6/§12 mappings and Quick Win artifacts.
- For every row, practitioners should record the Test-ID(s) executed and the exact EP-06.xx link in the project's register to keep traceability current.

Appendix B: EP-08 Evidence Pack Matrix

This summary matrix provides practitioners with a clear, readable view of how the Monitoring, Detection & Incident Response Architecture Evidence Pack repository is organized for adoption of the Parent Standard. Each Evidence Pack location corresponds to a core section of the annex standard, enabling consistent evidence collection, review, and traceability without creating future sub-standard evidence structures.

Evidence Pack alignment: EP-8 is the Evidence Pack repository for D08. Evidence is organized into five sections aligned with locations. EP-8.1 captures readiness artifacts for Section 5, EP-8.2 captures implementation artifacts for Section 6, EP-8.3 preserves foundational standards alignment evidence for Section 8, EP-8.4 maintains external control mapping evidence for Section 9, and EP-8.5 contains verification and validation evidence for Section 12. Together, these five locations provide traceability from prerequisites to implementation to proof.

Layer	EP Identifier	Purpose	Evidence Categories Included
Parent EP	EP-08.0	Annex wide index for MDIR evidence. Acts as the readme and pointer set for all EP-08.x sub packs and their latest pass or fail status for Quick Wins and §12 tests.	• MDIR architecture index: diagrams list, repository locations, owners, and revision dates • Evidence Pack registry: EP-08.x inventory with scope and links • Invariants register pointers (no fail open ingestion, immutable logging, rollback required, time sync) • Quick Win index and latest outcomes with references to EP-08.1, EP-08.2, and EP-08.5 • Traceability pointers: mapping sheet locations linking §5 to §6 to §12
Sub EP	EP-08.1	Detection engineering and validation evidence supporting §6.2 and §12 detection fidelity, including daily or scheduled validation runs.	• Detection rules as code repository references and change history • MITRE ATT and CK mapping and coverage views for priority techniques • Automated test outputs for detections (synthetic events and adversary simulation) • False positive and false negative tracking evidence, including tuning records • Quick Win evidence for daily detection validation runs, including pass fail results and remediation records

Layer	EP Identifier	Purpose	Evidence Categories Included
Sub EP	EP-08.2	SOAR playbooks and automation safety evidence supporting §6.3, including regression testing, approvals, and rollback proof.	• Playbook library as code references, version history, and ownership • Sandbox validation outputs for high-impact playbooks • Automation safety controls: approval gates, blast radius limits, kill switch proof, rollback records • Playbook execution logs and success metrics used for §12 validation • Quick Win evidence for the SOAR playbook auto test harness, including CI results and promotion blocks
Sub EP	EP-08.3	XDR integration and cross-domain correlation evidence supporting §6.4 and §12 cross-domain validation.	• Integration of health logs and connectivity checks for XDR to SIEM and SOAR • Enrichment completeness reports (asset criticality, identity context where applicable) • Cross-domain correlation examples showing deduped incidents • Automated remediation hook logs, with outcomes and rollback records where applicable
Sub EP	EP-08.4	Threat intelligence operationalization evidence supporting §6.5 and §12 intelligence propagation and hygiene validation.	• Feed health and ingestion latency reports • Deduplication and expiry records for indicators • Evidence of intelligence propagation into SIEM, SOAR, and XDR logic • Priority adversary profiles and sharing records where applicable • Change records showing indicator lifecycle policy updates and validation outcomes
Sub EP	EP-08.5	MDIR platform resilience and self-protection evidence supporting §6.6 and §12 resilience validation.	• Administrative access hardening proof: MFA enforcement, privileged action logs, service account scope evidence • Management network segmentation artifacts and approved integration paths

Layer	EP Identifier	Purpose	Evidence Categories Included
			• HA and DR designs and quarterly failover results showing no evidence of loss • Drift and tamper detection alerts and resolution records • Log integrity and time synchronization proofs for MDIR components
Future Sub EPs	EP-08.6+	Reserved for future MDIR sub-standards and additional Evidence Pack bundles as the standard expands.	• Reserved for future areas such as OT specific response constraints, advanced detection validation suites, or expanded evidence traceability exports

Notes for editors

- Each EP-08.x entry should reference the exact §6 outputs and the §12 test identifiers exercised by its artifacts and should indicate which invariant is proven (for example, no fail open ingestion, rollback required, time sync enforced, immutable evidence).
- EP-08.0 should include a human-readable index pointing to every sub EP location, owner, last update date, and the latest pass or fail status for associated Quick Wins and priority V&V tests.
- Evidence packs should remain few and meaningful. If an artifact does not support a §5 requirement, a §6 output, or a §12 test, it should not be required evidence.

Adoption References

NOTE: ISAUnited Charter Adoption of External Organizations.

ISAUnited formally adopts the work of the International Organization for Standardization / International Electrotechnical Commission (ISO/IEC) and the National Institute of Standards and Technology (NIST) as foundational standards bodies, and the Center for Internet Security (CIS), the Cloud Security Alliance (CSA), and the Open Worldwide Application Security Project (OWASP) as security control–framework organizations. This adoption aligns with each organization's public mission and encourages practitioners and institutions to use it. ISAUnited incorporates these organizations into its charter so that every Parent Standard and Sub-Standard is grounded in a common, defensible foundation.

a) **Foundational Standards (Parent level).**
ISAUnited adopts *ISO/IEC* and *NIST* as foundational standards organizations. Parent Standards align with these bodies for architectural grounding and auditability, and extend that foundation through ISAUnited's normative, testable specifications. This alignment does not supersede *ISO/IEC* or *NIST*.
b) **Security Control Frameworks (Control level).**
ISAUnited adopts *CIS, CSA*, and *OWASP* as control framework organizations. Control mappings translate architectural intent into enforceable technical controls within Parent Standards and Sub-Standards. These frameworks provide alignment at the implementation level rather than at the foundational level.
c) **Precedence and scope.**
Foundational alignment (*ISO/IEC, NIST*) establishes the architectural baseline. Control frameworks (*CIS, CSA, OWASP*) provide enforceable mappings. ISAUnited's security invariants and normative requirements govern implementation details while remaining consistent with the adopted organizations.
d) **Mapping.**
Each cited control mapping is tied to a defined output, an associated verification and validation activity, and an Evidence Pack ID to maintain end-to-end traceability from requirement to control, test, and evidence.
e) **Attribution.**
ISAUnited cites organizations by name, respects attribution requirements, and conducts periodic alignment reviews. Updates are recorded in the Change Log with corresponding evidence.
f) **Flow-downs.**

Obsolete and withdrawn documents should not be used; please use replacements.

(Parent to Sub-Standard). Parent alignment to the International *ISO/IEC* and *NIST* flows down as architectural invariants and minimum requirements that Sub-Standards must uphold or tighten. Parent-level mappings to *CIS, CSA*, and *OWASP* flow down as implementation control intents that Sub-Standards must operationalize as controls-as-code, tests, and evidence. Each flow-down shall reference the Parent clause, the adopted organization name, the Sub-Standard clause that implements it, the associated verification/validation test, and an Evidence Pack ID for traceability. Any variance requires a written rationale, compensating controls, and a time-bounded expiry recorded with an Evidence Pack ID.

Change Log and Revision History

Review Date	Changes	Committee	Action	Status
March 2026	Standards v 1.0 Submitted for Peer Review	Technical Fellow Society	Peer review	Pending
January 2026	Standards v 1.0 Published Draft	Task Group ISAU-TG39-2024	Published	Complete
December 2025	Standards Revision	Standards Committee	Submitted	Complete
October 2025	Standards Revision	Task Group ISAU-TG39-2024	Submitted	Complete
December 2024	Standards Development (Parent D01)	Task Group ISAU-TG39-2024	Draft Complete	Complete

End of Document
IO.