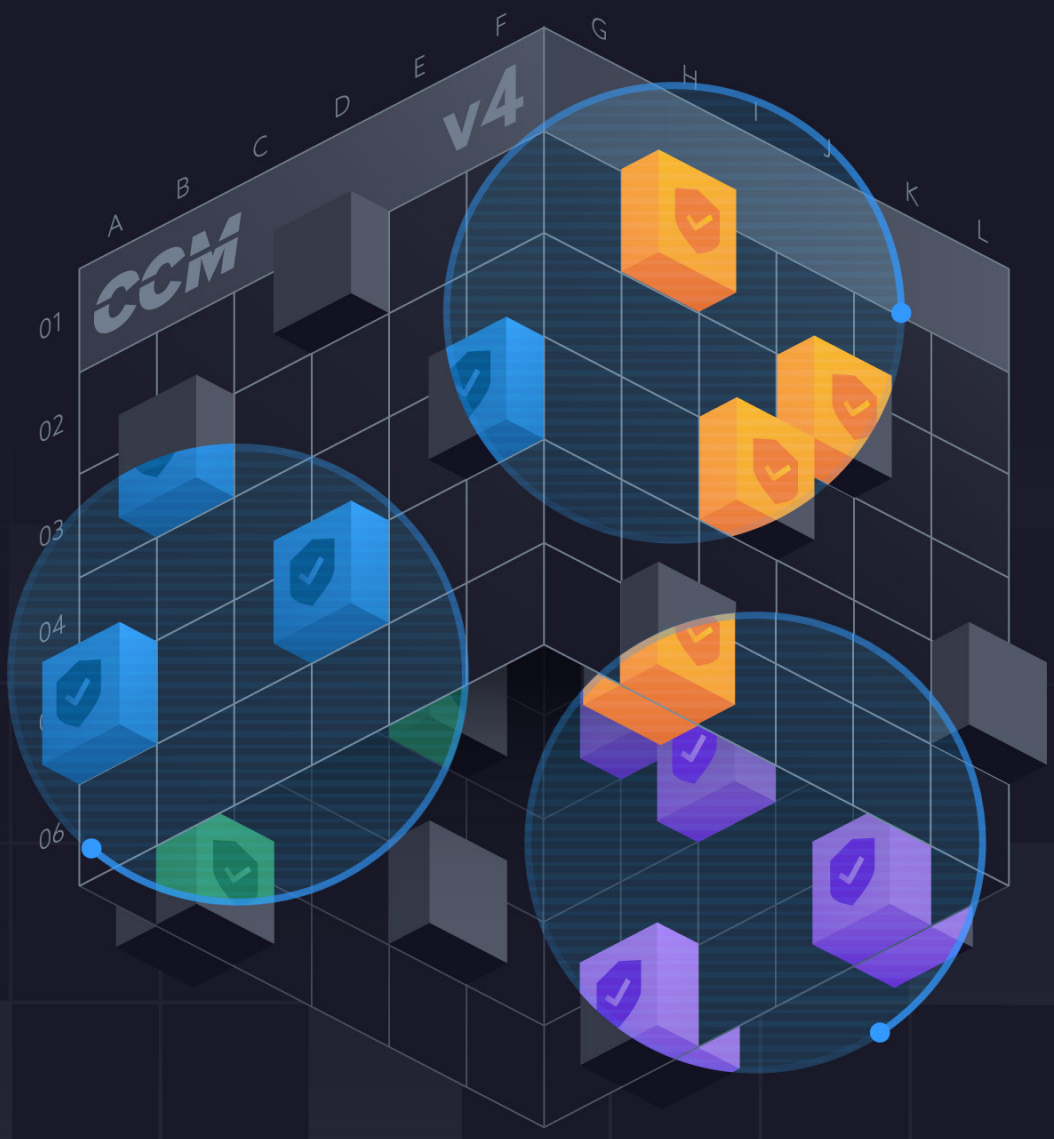
The Continuous Audit Metrics Catalog

Version 1.0



cloud security alliance®

The permanent and official location for The Continuous Audit Metrics Working Group is https://cloudsecurityalliance.org/research/working-groups/continuous-audit-metrics/.

Acknowledgments

Authors:

Jonathan Lewis Christopherson
Willy Fabritius
Raj Krishnamurthy
Daniele Catteddu
Kevin Murphy
Alain Pannetrat
Chris Pedigo
Mosi Platt
Max Pritikin (co-chair)
Anthony Scarfe
Carlos Victoria

Contributors:

Christian Banse
Michael Bently
James Condon
John DiMaria
Tinsae Erkailo
Alexandre Higuchi
Michaela Iorga
Amanda King
Julien Mauvieux
Brian Milbier
Dili Origbo
Judy Owen
Massimiliano Rak
Louis Seefried
Jonathan Villa

Special Thanks:

Bowen Close

*Many thanks to the **reviewers** who submitted a lot of valuable feedback on the early version of this document, which was released as a Request for Comment in June 2021.*

Table of Contents

1. Introduction

Are traditional IT security assurance tools outdated?

With DevOps and fast-paced technological evolutions, many cloud customers think that a third-party audit conducted once a year is no longer sufficient; they want their cloud service providers (CSPs) to offer continuous assurance of ongoing effectiveness regarding security processes and practices.

The blog post Continuous Auditing and Continuous Certification describes STAR (Security Trust Assurance and Risk) Continuous: "an innovative framework designed to provide compliance assurance to cloud customers on a monthly, daily, or even hourly basis."[1] STAR Continuous is based on the idea of "continuous auditing," achieved by continuously measuring specific attributes of an information system and comparing these results with pre-established security objectives. The results of this continuous auditing process are then shared in real-time with customers in a way that protects the cloud provider's confidential operations. This process must be automated in order to scale in cloud environments.

Selecting and measuring meaningful security attributes of an information system presents a significant challenge. While traditional security auditing processes can rely on a large body of knowledge and well-established references such as ISO/IEC 27001, ISO/IEC 27017, or the CSA CCM, there is no such foundation available for continuous auditing of cloud services. The closest existing references to address this topic are ISO/IEC 27004:2016 and NIST SP 800-55-rev1, but they focus mainly on traditional information systems and describe processes that often require human intervention. The work presented here is a first attempt to provide a foundation for continuous auditing of cloud services by defining a catalog of security metrics relevant to cloud computing with measurement processes that can be largely automated.

This catalog is the product of the work conducted by industry experts in the CSA Continuous Audit Metrics Working Group, which was established in early 2020. Given the novelty of our approach, this catalog does not aim to be exhaustive and complete; instead, this release aims to gather feedback from the community and guide our ongoing work while broadening awareness of continuous assurance within the cloud community.

Proposed metrics were designed to be consistent with the newly released CSA Cloud Control Matrix v4 controls (CCMv4).[2] These metrics aim to support internal CSP governance, risk, and compliance (GRC) activities and provide a helpful baseline for service-level agreement transparency. Additionally, depending on the success of this work and the STAR Continuous program's evolution, these metrics might be integrated within the STAR Program in the future, providing a foundation for continuous certification.

---

[1] Pannetrat, A. (2020, March 20). *Continuous Auditing and Continuous Certification*. Cloud Security Alliance. https://cloudsecurityalliance.org/blog/2020/03/20/continuous-auditing-and-continuous-certification/

[2] Cloud Security Alliance Cloud Controls Matrix Working Group. (2021, June 7). *Cloud Controls Matrix and CAIQ v4*. Cloud Security Alliance. https://cloudsecurityalliance.org/artifacts/cloud-controls-matrix-v4/

The remainder of this document is divided into the following sections:

- **Section 2:** An overview of security metrics and their origin, purpose, and use for continuous auditing.
- **Section 3:** The structure of the metrics catalog.
- **Section 4:** A list of 34 cloud security metrics.

We welcome feedback from the community on the continuous audit metrics catalog presented here, including:

- Ideas for new metrics covering new controls in the CCMv4;
- Suggestions for improvements or requests for clarification on existing metrics; and
- Experience reports on the implementation of these metrics in IT systems.

Members of the community interested in further contributing to this work are invited to create an account on https://circle.cloudsecurityalliance.org/ and join the "Continuous Audit Metrics" Community there. Alternatively, you can also send an email to research@cloudsecurityalliance.org.

2. Security Metrics and Continuous Auditing

2.1 What Are Metrics?

A security metric is a description of a process that measures a particular characteristic of an information system, in order to obtain information about the effectiveness of the information security management system.

More precisely, ISO/IEC 19086-1:2016 describes a metric as a standard for measurement that defines the rules for performing the measurement and for understanding the results of the measurement. In this context, a measurement is defined as a process to quantify or qualify an attribute. According to ISO/IEC 27000:2014, an attribute is a property or characteristic of an object that can be distinguished quantitatively or qualitatively by human or automated means.

As a process, a measurement involves the gathering of data such as system logs, test results, configuration files, security events, and sometimes the results of other measurements. These elements are often collectively referred to as *evidence*. ISO/IEC 27000 and many other sources refer to the result of a measurement as a *measure*. More recent initiatives, such as ISO 27004, NIST SP 500-307, ISO/IEC 19086, and CSA's STAR, prefer the term *measurement result*, as the word *measure* has multiple meanings in information security and is a source of confusion when it comes to metrics. We also use the term *measurement result* in this work.

Note that security professionals sometimes use the word metric colloquially to describe measurement results, which can create confusion. To avoid such confusion, this document uses the terminology established in relevant international standards where applicable.

2.1.1 Terminology

The terminology used in this work is largely based on ISO/IEC 19086-1:2016, the standard framework for cloud service-level agreements (SLA). This notably includes the following terms:

- **Attribute:** Property or characteristic of an object that can be distinguished quantitatively or qualitatively by human or automated means.
- **Measurement:** The logical sequence of operations used in quantifying or qualifying an attribute.
- **Measurement Result:** The qualitative or quantitative value obtained as the output of a measurement.
- **Metric:** A standard of measurement that defines the conditions and the rules for performing the measurement and for understanding the results of a measurement.
- **Cloud Service-Level Objective (SLO):** A commitment made by a CSP for a specific, quantitative characteristic of a cloud service, where the value follows the interval scale or ratio scale.

- **Cloud Service-Qualitative Objective (SQO):** A commitment made by a CSP for a specific, qualitative characteristic of a cloud service, where the value follows the nominal scale or ordinal scale.

Note: The term service level indicator (SLI) is sometimes used in the literature[3] to describe the equivalent of a *measurement result* in the context of performance measurement rather than security.

2.2 Benefits of metrics

Adopting metrics offers several benefits; we list the key ones below:

2.2.1 Measuring the Effectiveness of an Information System

Using metrics allows organizations to assign qualitative or quantitative values to various attributes of an information system. By carefully selecting attributes that reflect the implementation of security control, metrics can be used to measure the effectiveness of these controls.

By implementing metrics, organizations get better visibility of their security posture and can potentially identify blind spots. Changes or deviations from controls result in changes in measurement results, indicating a progression or a regression of effectiveness and enabling a data-driven approach to risk management.

2.2.2 Increasing the Maturity of an Organization's Governance and Risk Management Approach

Directly linked to the previous benefit, metrics also support the improvement and evolution of an organization's governance and risk management approach. In other words, metrics are a key tool for fostering the maturity of the organization's risk management program. Measuring the effectiveness of a set of controls provides a better understanding of how resources can be best allocated, allows for benchmarking with internal and external standards and best practices, and guides organizations towards a more mature security governance and risk management approach.

Moreover, the process of implementing metrics will in itself help organizations gain maturity. Organizations that select and implement security metrics are required to adopt the necessary tools to categorize their assets and measure associated security attributes. This work is not trivial, so the ability to conduct it illustrates that the organization has reached a certain level of maturity in information security management. Implementing even a few key metrics successfully can drive an organization towards a stronger security posture.

Metrics facilitate these goals when they are specific, measurable, achievable, relevant and time-bound.

---

[3] See for example, Google's Site Reliability Engineering (SRE): Jones, C., Wilkes, J., Murphy, N., & Smith, C. (2017). *Site Reliability Engineering: Service Level Objectives*. Google Site Reliability Engineering. https://sre.google/sre-book/service-level-objectives/

2.2.3 Increasing Transparency, Fostering accountability, and Enabling Continuous Auditing and Compliance

Organizations that adopt metrics can provide relevant stakeholders with a view into their security and privacy practices and better explain and justify their SLA. Such an increased level of transparency fosters trust and accountability within the overall supply chain—each member of the chain can build more reliable SLAs as information asymmetry between the parties is reduced. The benefits also extend to the relationship between the CSP, its customer, and relevant regulatory authorities. An increased level of transparency allows cloud customers to improve their due diligence approach and accountability programs, which will be based on clearer data.

In the same context, metrics support the ability to measure control performance at required time intervals, enabling continuous auditing and continuous compliance. Continuous auditing is beneficial to both internal and external stakeholders:

- Internally, organizations can use metrics and objectives to continuously measure the performance of their information security. This can help maintain a proper security baseline between formal audits and drives continuous improvement.[4]
- Externally, organizations can also use metrics to monitor security and share results with their external stakeholders and in particular their customers, who seek assurance that the organization's information security continuously meets expected levels. Automation and application programming interfaces (APIs) can make this process extremely efficient.

We will further examine continuous auditing in the next paragraphs.

Metrics balance transparency with enabling organizations to maintain operational confidentiality of their internal policies. For example, the metric TVM-03-M1 reflects the "percentage of high and critical vulnerabilities that are remediated *within the organization's policy timeframes*" rather than against an absolute value such as "ten days." This is a balance, and other metrics may include specific, non-subjective, values now or in the future. As a general goal, metric expressions were selected to represent percentages of objectives being met rather than defined baseline measures. Additionally, the SLOs are recommendations and the actual objective is always determined by the organization's risk profile.

In the future, once metrics are well established in the industry, they could also be used for benchmarking purposes, allowing organizations to compare cloud providers in real-time. However, this assumes a high level of standardization in the definition of metrics and in their implementation. This would also require striking a balance between the protection of CSP's confidential operations and the need to disclose actionable information to relevant stakeholders.

[4] In some cases, the data obtained as a byproduct of using metrics for security can also help enhance broader aspects of internal IT, such as capacity planning.

2.3 Continuous Auditing

One of the main benefits of developing metrics is that it enables the creation of a continuous auditing process, which allows an organization to show control compliance at all times. To understand why this is important, it's useful to first examine some of the shortcomings of traditional assurance mechanisms.

Traditional security assurance is based on verifying that controls are correctly selected, designed, implemented, enforced, and monitored. This process is largely a manual task performed by humans through evidence and documentation review and repeated every 6 or 12 months. This approach has solidified over the years through standardization and best practices, but in today's cloud-centric environment it suffers from several important shortcomings.

First, if we seek to obtain more continuous assurance regarding the security of an information system, this traditional approach does not scale in terms of cost and feasibility. Manual or semi-automated assessment processes designed to be conducted every six months are unlikely applicable for verifications that are expected to be performed on a daily or hourly basis.

Second, while traditional security assessments may seem appropriate for policy and procedural controls, they will fall short when dealing with the evaluation of technical security measures. This is especially the case if they are applied to environments that are continuously evolving while being exposed to changing threats and vulnerabilities. It makes sense to implement automated and continuous assessments of technical measures, as many organizations already do, and we can even partially extend that idea to policy and procedural controls—while evaluation of policy and procedural controls cannot be directly automated, we can implement automated techniques for the collection of evidence to prove their effectiveness.

Third, humans make mistakes and may overlook small but important details when doing reviews repeatedly. In contrast, an automated assessment can be repeated indefinitely, without mistake, provided that the underlying tools are trustworthy.

As a consequence of the shortcomings of traditional assurance tools, organizations that want continuous assurance must rethink their approach to security assessments. For continuous assurance, manual assessments must be traded for automated measurements, which largely leave humans out of the loop. Instead of assessing controls directly, tools are used to measure the security attributes of an information system and infer indirectly whether controls are effectively in place.

For example, consider the Supply Chain Management, Transparency, and Accountability (STA) domain of CCMv4, which contains 14 control objectives. Taken together, the goal of these control objectives is to ensure that adequate tools, policies, and procedures are in place to establish, document, approve, communicate, apply, evaluate, and maintain aspects of the supply chain used in delivering CSP products and services. Notably, evaluating compliance to these control objectives means reviewing documentation, tools, processes, and governance. This kind of work is largely manual and will be done every few months, at best. Despite providing periodic assurance on supply chain management, this approach fails to keep up with the supply chain evolutions and risks associated with fast-paced product development. Many organizations mitigate the risks by having

specific technical processes in place, some of which can be automatically and regularly measured once the right tools are in place. For example:

- Maintaining an adequate inventory of supply chain relationships and automatically scanning for production packages and reconciling them with the inventory every two weeks (see STA-07-M3).
- Observing ingress and egress connections daily and evaluating if such connections are on the approved allowlist of supply chain providers in the inventory (see STA-07-M5).

These supply chain measurements provide quantitative or qualitative values that can be contrasted with predefined objectives set by the organization in relationship with its risk appetite. An organization that is able to set such objectives and then provide its stakeholders with measurement results that continuously support whether these objectives are met is an organization with significant maturity and awareness. Further, these metrics also surface the interdependencies across CCMv4 control domains. For example, the effective measurement of automated STA metrics is dependent on the implementation of appropriate Logging and Monitoring (LOG) and Datacenter Security (DCS) controls as well.

2.4 Linking Metrics to the CCM

The metrics presented in the metric catalog are linked to CCMv4, which was released in January 2021. The Cloud Controls Matrix (CCM) is CSA's flagship cybersecurity framework for cloud computing, featuring 197 control objectives categorized in 17 security domains.

Each metric is linked to a primary CCMv4 control objective, and using that metric should provide organizations with visibility regarding the effectiveness of the implementation of that primary CCM control objective. In practice, there is no one-to-one correspondence between metrics and security controls. In fact, many metrics provide insights into the implementation of more than one control objective and, conversely, several metrics might be needed to effectively measure the implementation of one control. The catalog provided in this document recognizes this fact by supplementing the "primary control objective" of each metric with a list of additional related CCMv4 Control IDs. This link between metrics and controls is important because it helps support organizations' compliance efforts by anchoring security measurements into a well-known control framework that auditors recognize.

Notes:

- The metrics catalog we publish in this first release contains metrics related to a subset of the CCM control. The metrics catalog is meant to be a "living document" and additional metrics and extended coverage of the CCM controls will be added over time.
- The metrics provided in the CSA catalog are not to be considered the "only" way to measure a CCM control implementation effectiveness, but rather "a possible way" to achieve such a goal. Some organizations might use different metrics to achieve the same goals.

For the Cloud Security Alliance, the explicit link between metrics and the CCM opens up the possibility of creating a continuous certification framework, which would supplement the existing

certifications and attestations currently offered in the STAR program.[5] To obtain a "continuous certification," organizations would need to demonstrate that they continuously meet a certain number of SLOs/SQOs.[6]

2.5 Selecting and Using Metrics for Continuous Auditing

The metrics presented in this document are not designed as a one-size-fits-all. Each organization is different and is likely to use the proposed metrics in a different way, and some metrics might be rightfully ignored. The three main considerations that will drive an organization to select and use a metric are:

- Risk management priorities;
- Maturity; and
- Transparency.

When seeking continuous assurance, it makes sense to first focus on the most critical risks that need to be addressed. As such, an organization may choose to start with only a handful of metrics that target those risks. Consider, for example, an organization that uses numerous cloud services from different vendors: it may make sense for them to focus on supply chain metrics. An organization that offers health data storage might focus instead on metrics related to cryptography and key management. The difference will not only appear in the selection of a metric but also in the frequency of measurement they select in the implementation of that metric—a critical security attribute will likely be measured more frequently.

Some metrics in this catalog may be simple to implement, while others may rely on the assumption that the organization has certain complex processes or tools in place. For example, any metric that relies on the categorization of assets implicitly assumes that the organization has tools that can identify and categorize all relevant assets. Obviously, not all organizations have the level of maturity that is reflected by the existence of such tools. Maturity is therefore also a limiting factor in the selection of metrics. Organizations can review these metrics as guidance for the development of their security monitoring strategy, with a goal of increasing their capabilities over time.

The metrics in the catalog offer different levels of flexibility—some metrics are policy-dependent, involving percentages of events that fall within the organization's policy, where other metrics target more absolute measurements. Policy-dependent metrics are more flexible, but are also easier to manipulate—organizations with informal or less mature policies can still achieve good results.

---

[5] Cloud Security Alliance. (n.d.). *Security, Trust, Assurance, and Risk (STAR)*. Retrieved October 7, 2021, from https://cloudsecurityalliance.org/star/

[6] Pannetrat, A. (2020, March 20). *Continuous Auditing and Continuous Certification*. Cloud Security Alliance. https://cloudsecurityalliance.org/blog/2020/03/20/continuous-auditing-and-continuous-certification/

An organization implementing a continuous auditing program can use the catalog presented in this document to support numerous tasks, such as:

- Identifying security attributes of the information system that can be measured in an automated way according to selected metrics, with measurement results providing a valid indication that certain security controls are in place.
- Defining a frequency of measurement for each attribute based on feasibility, cost, and risk levels.
- Associating measurement results with objectives that should be attained (SLOs and SQOs).
- Informing relevant stakeholders whether the objectives are met.

3. Catalog Structure

3.1 Metric Description

Because metrics describe the measurement of security attributes in an information system, it is tempting to describe them with a detailed technical representation relying on complex XML or JSON schemas. For example, ISO/IEC 19086 proposes a machine-readable model for metrics that attempts to describe every nuance of a metric (another example is OpenMetrics[7]). This approach favors interoperability and reproducibility. The downside of this approach is that it narrows the scope of applicability of the metrics to organizations that have the precise technical capability or tooling to implement the requirements of the metric. Today, it is unclear if the industry is ready to take this road. In this work, we take a simpler approach and focus on the definition of metrics independent of their technical representation. We want to garner the feedback of the community on the value of metrics rather than their format, which could be the focus of later attention if necessary.

Each entry in the metric catalog contains the following fields:

Primary CCMv4 Control ID	A primary security control in the CSA CCMv4 that can be related to the defined metric. Implementing the corresponding metric should provide measurements that can be used to partially or fully support the corresponding security control. The reference to a CSA CCM control is somewhat arbitrary, because in some cases a metric is applicable to more than one security control. Nevertheless, a reference to a CCM control is useful to show that the metrics are anchored in existing security practices and it provides a way to broadly identify what coverage is achieved in terms of security.
Primary Control Description	The description of the primary control ID from CSA CCMv4, to help the reader.
Related CCMv4 Control IDs	A list of all other CCMv4 controls that are related to the metric in addition to the primary control already described. A metric may be related to a control in at least two ways: • The metric may provide assurance regarding the effectiveness of more than one CCMv4 control. • The metric may rely on the assumption that other CCMv4 controls are in place because these other controls appear as necessary conditions for the proper implementation of the metric.

[7] OpenMetrics. (n.d.). *The OpenMetrics project — Creating a standard for exposing metrics data.* Retrieved October 7, 2021, from https://openmetrics.io/

Metric ID	Note: Each metric is provisionally named after the primary control ID to reflect the primary mapping.
Metric Description	A brief description of the metric.
Expression	A definition of the security attribute and its measurement method, which forms the core description of the metric. The expression is either: 1. A mathematical formula describing the measurement; or, 2. A description of the conditions and rules for performing the measurement of a security attribute.
Rules	A list of rules that MUST be followed to perform a measurement and obtain measurement results with this metric. When the expression is a mathematical formula, the rules can be used to detail how different fields in the formula are calculated.
SLO Recommendations	Industry best-practice recommended objectives that organizations should meet, in terms of the measurement results obtained through the metrics (e.g., minimum expected level). This information represents a general recommendation—not a requirement—and it should be adapted to the organization's risk profile.

Implementation Guidelines

(Sometimes provided, presented after the table for readability.) A set of guidelines and clarifications that may assist the reader in the interpretation and implementation of the proposed metric.

3.2 Sampling Period and Measurement Frequency

There are two distinct temporal characteristics that can affect how a metric is measured:

> 1. **Measurement Frequency:** How often a measurement is produced (e.g., every 24 hours).
> 2. **Sampling Period:** The timespan of events included in the measurement (e.g., the last 30 days).

Measurement frequency is usually selected by taking into account the risk appetite of the organization and the technical capabilities of the corresponding measurement tools. If a metric is very important to the risk management of an organization, it will likely be applied at a higher frequency as

is possible (e.g., every day). On the other hand, some measurements are costly in terms of resources and it may not be feasible to apply them with high frequency.

A sampling period is used to limit the scope of measurement to events that cover a specific period of time. For example, cloud SLAs typically calculate availability, taking into account disruptions that have happened over a period of 30 days. Measurement frequency does not necessarily need to match the sampling period. For example, it's possible to provide a new measurement every day for data that covers the past 30 days (i.e., a moving average). This can sometimes lead to confusion when trying to discuss metrics.

Note that many metrics do not apply to events and, as a consequence, not all metrics have a sampling period.

4. Metrics Catalog

This section provides a list of 34 security metrics. This list is not meant to be limitative—organizations are encouraged to expand upon this list to suit their needs.

4.1 Metric AIS-06-M1

Primary CCMv4 Control ID	AIS-06
Primary Control Description	Establish and implement strategies and capabilities for secure, standardized, and compliant application deployment. Automate where possible.
Related CCMv4 Control IDs	DCS-06, GRC-05
Metric ID	**AIS-06-M1**
Metric Description	This metric measures the percentage of running production code that can be directly traced back to automated security and quality tests that verify the compliance of each build.
Expression	Percentage of compliant code: 100 * A/B A = Total number of pieces of production code that have an associated verification step B = Total number of pieces of production code
Rules	"Production code" is code deployed to the production runtime environment(s) within the scope of the Information Security Program defined in the GRC-05 control objective.
SLO Recommendations	95%

Implementation Guidelines

There must be a software inventory of deployed production code (see DCS-06 for more information). Production code must be quantified based on the organization's definition of deployed code running in production (e.g., microservices, builds, releases, packages, libraries, serverless functions, etc.).

This should be the same number used to measure AIS-07.

The definition of "deployed production code" used for the software inventory should be aligned with application security scanning, testing, and/or reporting methods where possible to simplify measurement.

The likelihood of standardized deployments can decrease as the number of different deployment systems increases. If the software deployment pipeline has multiple stages where change could be introduced and end-to-end validation cannot be performed, then this metric may be more suitable for an organization:

0%<=Percentage of steps in the software deployment pipeline that have an associated verification step<=100%

There should be a mechanism to identity deviations and, if deviations from the standard are approved, then the system should account for (and manage) the exception as approved.

This metric should at least be aligned with an organization's development or release cycle to provide timely input for correction in the next deployment or release. For example, if an organization uses an Agile development methodology with two-week sprints, then the metric should be measured at least every two weeks to provide data for review at sprint retros.

4.2 Metric AIS-07-M3

Primary CCMv4 Control ID	AIS-07
Primary Control Description	Define and implement a process to remediate application security vulnerabilities, automating remediation when possible.
Related CCMv4 Control IDs	DCS-06, GRC-05
Metric ID	**AIS-07-M3**
Metric Description	This metric measures the coverage for application vulnerability remediation across the production code.
Expression	Percentage: 100 * A/B A = Number of deployed production applications with acceptable level of risk from application security vulnerabilities B = Total number of deployed production applications

Rules	Production Application = Applications tracked within the software inventory established in DCS-06 Acceptable level of risk from application security vulnerabilities: Vulnerabilities categorized as medium or low risk as well as critical or high vulnerabilities marked or identified as "Accepted" (i.e., remediation not required). Examples of accepted vulnerabilities can be false positives or vulnerabilities with compensating controls that make the residual risk of exploitation acceptable.
SLO Recommendations	80% Rationale: The 2020 Application Security Observability Report from Contrast Labs found 26% of applications had at least one serious vulnerability, with 79% of those vulnerabilities remediated within 30 days. That leaves 20% of applications with serious vulnerabilities after 30 days, so the SLO to have 80% of production code with acceptable level of risk from application security vulnerabilities should be achievable for the average organization.

Implementation Guidelines

There must be a software inventory of deployed production code (see DCS-06 for more information). Production code must be quantified based on the organization's definition of deployed code running in production (e.g., microservices, builds, releases, packages, libraries, serverless functions, etc.). This should be the same number used to measure AIS-06.

The definition of "deployed production application" used for the software inventory should be aligned with application security scanning, testing, and/or reporting methods where possible to simplify measurement.

"Acceptable Level of Risk" should be defined by the organizations vulnerability management guidelines (e.g., only "critical" and "high" vulnerabilities, or "medium vulnerabilities and higher," etc.). Classification of vulnerabilities as "high" or "critical" risk, etc., should be defined in the vulnerability management tool based on an industry-accepted scoring system, such as the Common Vulnerability Scoring System (CVSS).[8] For instance, vulnerabilities with a CVSS score of nine or higher are "critical," and vulnerabilities with CVSS scores between seven and nine could be defined as "high" risk.

[8] National Institute of Standards and Technology. (n.d.). *National Vulnerability Database: Vulnerability Metrics*. NIST. Retrieved October 6, 2021, from https://nvd.nist.gov/vuln-metrics/cvss

4.3 Metric AIS-07-M6

Primary CCMv4 Control ID	AIS-07
Primary Control Description	Define and implement a process to remediate application security vulnerabilities, automating remediation when possible.
Related CCMv4 Control IDs	AIS-03, TVM-10, GRC-02
Metric ID	**AIS-07-M6**
Metric Description	This metric measures the percentage of critical vulnerabilities that are not fixed or marked as accepted within the time specified by policy.
Expression	Percentage: 100 * A/B A = Number of unaccepted critical or high vulnerabilities with an age greater than the policy defined maximum age B = Total number of critical or high vulnerabilities within this period **Example:** Percentage: 100 * 1-(A/B) A = Number of deployed production appliances with unaccepted critical or high vulnerabilities with an age greater than the policy defined maximum age B = Total number of deployed production applications
Rules	Production Application = Applications tracked within the software inventory established in DCS-06 Acceptable level of risk from application security vulnerabilities: Vulnerabilities categorized as medium or low risk as well as critical or high vulnerabilities marked or identified as "Accepted" (i.e., remediation not required). Examples of accepted vulnerabilities can be false positives or vulnerabilities with compensating controls that make the residual risk of exploitation acceptable.
SLO Recommendations	N/A

Implementation Guidelines

1. Classification of vulnerabilities as "high" or "critical" risk should be defined in the vulnerability management tool based on an industry-accepted scoring system, such as the Common Vulnerability Scoring System (CVSS).[9] For instance, vulnerabilities with a CVSS score of nine or higher are "critical," and vulnerabilities with CVSS scores between seven and nine could be defined as "high" risk.
2. Date and time of vulnerability discovery could be obtained from the vulnerability management tool as it scans and detects vulnerabilities.
3. Date and time of vulnerability remediation or acceptance could be obtained in the following ways:
    a. From the vulnerability management tool as it scans and finds that a previously detected vulnerability is no longer present/detected.
    b. From the patch deployment tool (e.g., SCCM) as it successfully deploys and installs a patch that fixes an identified vulnerability.
    c. From the application/code release tool as it moves into production the new version of the application that no longer contains the code vulnerability.

Frequency of evaluation should be aligned with the frequency of vulnerability scans. (Scans should happen at LEAST monthly, but more frequently is recommended.)

Vulnerability scans can be done at a predefined frequency or whenever new code is built or deployed.

4.4 Metric BCR-06-M1

Primary CCMv4 Control ID	BCR-06
Primary Control Description	Exercise and test business continuity and operational resilience plans at least annually or upon significant changes.
Related CCMv4 Control IDs	BCR-01, BCR-02
Metric ID	**BCR-06-M1**
Metric Description	This metric reports the percentage of critical systems that passed Business Continuity Management and Operational Resilience (CCMv4 domain BCR) tests.

[9] National Institute of Standards and Technology. (n.d.). *National Vulnerability Database: Vulnerability Metrics*. NIST. Retrieved October 6, 2021, from https://nvd.nist.gov/vuln-metrics/cvss

Expression	Percentage: 100*A/B A = Number of critical systems that passed BCR tests during the sampling period B = Total number of critical systems operating during the sampling period
Rules	Criteria for system criticality must be defined and there must be a list of critical systems identified. Recovery point objective(s) and recovery time objective(s) must be defined for critical systems. This metric does not attempt to measure the appropriateness of the RPOs or RTOs. This metric is dependent on control BCR-02 providing reasonable assurance of sufficient RPOs and RTOs for critical systems. BCR testing intervals must be defined.
SLO Recommendations	80% BCR/chaos testing is intended to be a learning activity, and it should test both the core of the system and the edges of the system. A perfect score indicates that edge cases and previously undefined scenarios are not being tested. Too low of a score indicates that an organization hasn't learned from their tests. New tests should be continually added and old tests may be retired. This metric should show regular variability.

Implementation Guidelines

Critical systems should be identified in accordance with the CCMv4 implementation guidelines for BCR-02.

For this metric, "passed" means achieving the RPO(s) within the RTO(s) defined for each critical system in the scope of the assessment/audit, according to the CCMv4 implementation guidelines for BCR-02.

The sampling period for this metric should align with the testing intervals defined by the business continuity plan, in accordance with the CCMv4 implementation guidelines for BCR-04.

BCR tests should include chaos testing where possible. "Chaos engineering is the discipline of experimenting on a software system in production in order to build confidence in the system's capability to withstand turbulent and unexpected conditions."[10]

[10] Wikipedia contributors. (2021, October 6). *Chaos engineering*. Wikipedia. https://en.wikipedia.org/wiki/Chaos_engineering

4.5 Metric CCC-03-M1

Primary CCMv4 Control ID	CCC-03
Primary Control Description	Manage the risks associated with applying changes to organization assets, including application, systems, infrastructure, configuration, etc., regardless of whether the assets are managed internally or externally (i.e., outsourced).
Related CCMv4 Control IDs	DCS-06
Metric ID	**CCC-03-M1**
Metric Description	Percentage of all assets that have change management technology integrated.
Expression	Percentage: 100 * A/B A = Number of assets that have change management technology integrated B = Total number of assets
Rules	Change management technology covers release management tools that enable automated deployment and rollback of software builds in production.
SLO Recommendations	80% This provides flexibility for organizations to move quickly. The signal is if this measure is going down or going up. The exact level is a measure of the organization's risk tolerance.

Implementation Guidelines

This metric requires the implementation of CCMv4 DCS-06, "Assets Cataloguing and Tracking," and the capability to determine which assets or asset groups are deployed using change management technology that can rollback changes and/or stop deployment of risky changes based on automated test results.

Given the dynamic nature of cloud environments, the metric can provide more value if the variations in the release management system's coverage over the population of assets is reported over time. The percentage of assets that fall within an accepted number of deviations provides stakeholders assurance of whether change control is getting better, worse, or being maintained. Larger populations of more than 1,000 assets can use six standard deviations as an acceptable level of change over time (i.e., Six Sigma). Smaller populations of assets will need to use fewer standard deviations as an acceptable level of change, perhaps even just one deviation. For more information on the use of standard deviation in security metrics, see the related excerpt of Andrew Jaquith's *Security Metrics: Replacing Fear, Uncertainty, and Doubt*.[11]

4.6 Metric CCC-07-M1

Primary CCMv4 Control ID	CCC-07
Primary Control Description	Implement detection measures with proactive notification in case of changes deviating from the established baseline.
Related CCMv4 Control IDs	DCS-06, CCC-03
Metric ID	**CCC-07-M1**
Metric Description	This metric measures the percent of positive test results from all configuration tests performed.
Expression	Percentage: 100 * A/B A = Number of configuration items that were tested and passed successfully B = Total number of configuration items that were tested
Rules	This metric captures the number of tests passed out of the total number of tests defined. Each test is assumed to verify a "configuration item," which is an arbitrarily defined as any component for which a test can be defined.
SLO Recommendations	95%

[11] Jaquith, A. (2007). *Security Metrics: Replacing Fear, Uncertainty, and Doubt* (1st ed.). Addison-Wesley Professional.

Implementation Guidelines

This metric assumes that CCC-03 has been successfully implemented and thus assumes that enough configuration items, at least in terms of number of DCS-06 assets, have change management technology to make this metric meaningful.

This metric does not take into account a measure of risk for the configuration tests that have failed. The resulting flat percentage may not tell the full story of risk incurred from a control failure. Future work may incorporate risk measures such as "high and critical" configuration tests.

The frequency of reporting this metric should tie in to the frequency of deployments/expected changes, minimally once a week. This metric should be measured on an automated, continuous basis.

Since the scope is under the control of the organization, metric results should be relatively high. The signal from this metric is that the existing system for change management is working or failing. A low percentage may not indicate a significant cybersecurity risk, but it may be a leading indicator of future security risk if the practice doesn't improve.

This is different than IVS-04, which measures the number of hardening tests against all assets.

4.7 Metric CEK-03-M2

Primary CCMv4 Control ID	CEK-03
Primary Control Description	Provide cryptographic protection to data at-rest and in-transit, using cryptographic libraries certified to approved standards.
Related CCMv4 Control IDs	CEK-04, DCS-06, CEK-0-1
Metric ID	**CEK-03-M2**
Metric Description	This metric measures if the cryptographic module continues to be up to "approved standards."
Expression	Percentage: 100 * A/B A = Number of assets responsible for data at-rest or in-transit where the cryptographic library has passed Automated Cryptographic Validation Protocol tests or equivalent tests B = Total number of assets responsible for data at-rest or in-transit

Rules	N/A
SLO Recommendations	85% SQO is the expression output (percent remediated within policy-specified time constraints). As this is an important aspect of functionality, targets should be around 85%

Implementation Guidelines

This leverages asset management and off-the-shelf automated functionalities while allowing for flexibility against policy (which has previously passed a CEK-01 audit).

4.8 Metric CEK-04-M1

Primary CCMv4 Control ID	CEK-04
Primary Control Description	Use encryption algorithms that are appropriate for data protection, considering the classification of data, associated risks, and usability of the encryption technology.
Related CCMv4 Control IDs	CEKM-05
Metric ID	**CEK-04-M1**
Metric Description	This metric measures the percentage of assets with cryptographic functions that meet an organization's defined cryptographic requirements.
Expression	Percentage: 100 * A/B A = Number of assets with a cryptographic function that meets cryptographic requirements B = Total number of assets with a cryptographic function
Rules	The specification should be reported for all the adopted cryptographic suites.
SLO Recommendations	90%

Implementation Guidelines

For a minimum viable product, the scope of evaluation may be limited to public-facing services, in which case a scan of all externally facing assets should be made and the scanned values compared against the requirements of the policy.

The SLO used for this metric may need to be increased or decreased based on the scope of assets covered by the metric.

This metric depends on the data classification tool in DSP-03 and requires that an organization determine the appropriate level of encryption for each classification, then requires comparison of the expected encryption applied versus the actual encryption applied and reports on the difference.

IPY-03 covers a subset of this measurement.

4.9 Metric DCS-06-M1

Primary CCMv4 Control ID	DCS-06
Primary Control Description	Catalogue and track all relevant physical and logical assets located at all of the CSP's sites within a secured system.
Related CCMv4 Control IDs	LOG-05
Metric ID	**DCS-06-M1**
Metric Description	This metric measures the ratio of managed assets (i.e., cataloged and tracked) to detected assets. The goal is to provide a signal if the asset cataloging and tracking system stops working.
Expression	Percentage: 100 * A/B A = Number of distinct assets seen in security audit logs during the sampling period that are in an asset catalogue B = Number of distinct assets seen in security audit logs during the sampling period
Rules	The assumption is that the design of the DCS-06 control process(es) was found to be effective by internal or external audits.
SLO Recommendations	95%

Implementation Guidelines

This relies on the security audit logs as defined in LOG-05 and the asset catalog defined in DCS-06.

This assumes LOG-05 is inclusive of logs of a number of events such as network traffic, network scanning, and physical asset inventory. It assumes that the logs include network traffic logging and logs from other assets and that they are sufficient to detect unexpected assets. We assume "everything that is worthy is logged." It depends on the auditor to ensure the logging is "complete enough."

This is consistent with the metric for UEM-04 and implementors may benefit from the similarities.

The following is likely dependent on the STA-01 through STA-06 and the SSRM. As those mature, perhaps any third-party CSPs used by the organization where shared responsibility of controls resides in the organization should be included as logical assets for this catalog. For example, if a CSP provides a micro-service inherent in the operations of an offering, that micro-service is a logical asset. This ensures that metrics where DCS numbers are used in the denominator include those micro-services. This is intended to ensure the "coverage" is accurate and inclusive of third-party CSPs where the organization is responsible for the controls.

4.10 Metric DSP-04-M2

Primary CCMv4 Control ID	DSP-04
Primary Control Description	Classify data according to its type and sensitivity level.
Related CCMv4 Control IDs	DSP-05, DSP-01, DSP-03, DSP-04
Metric ID	**AIS-07-M3**
Metric Description	This metric measures the ratio of data assets that have been classified according to the data classification policies specific to each organization. An organization may have a predefined list of data types (e.g., health care record, payment card record, identification number, etc.) and/or data sensitivity levels (e.g., Confidential, Internal Use Only, Public).
Expression	Percentage: 100 * A/B A = Total number of data records classified by type and/or sensitivity B = Total number of data records stored

Rules	The total number of records classified by type and/or sensitivity is a count of all data assets that have a defined classification by type or sensitivity level ("undefined" classifications are not counted for this variable). The total number of records stored is a count of all data assets that have been collected and are stored in the system, such as DSP-03. This metric measures data in terms of distinct data records, not distinct data types.
SLO Recommendations	99%

Implementation Guidelines

All data records must have corresponding metadata related to its data type and/or sensitivity. A list of data types and sensitivity levels must be defined. Records that do not meet any of the data classification types or sensitivity levels will have an "undefined" classification and are not considered as "classified" for this metric.

4.11 Metric DSP-04-M3

Primary CCMv4 Control ID	DSP-04
Primary Control Description	Classify data according to its type and sensitivity level.
Related CCMv4 Control IDs	DSP-05, DSP-01, DSP-03, DSP-04, DCS-06
Metric ID	**AIS-07-M3**
Metric Description	This metric measures the ratio of assets in the asset catalog that have been classified according to the data classification policies specific to each organization. An organization may have a predefined list of data types (e.g., health care record, payment card record, identification number, etc.) and/or data sensitivity levels (e.g., Confidential, Internal Use Only, Public).

Expression	Percentage: 100 * A/B A = Total number of assets in the asset catalog that are classified by type and/or sensitivity of the data on that asset B = Total number of assets in the organization's asset catalog
Rules	The total number of assets classified by type and/or sensitivity of the data contained on the asset is a count of all assets that have a defined classification by type or sensitivity level ("undefined" classifications are not counted for this variable). The total number of assets is a count of all assets that have been collected and are stored in the system, such as DSC-06.
SLO Recommendations	99%

Implementation Guidelines

All asset records must have corresponding metadata related to the type and/or sensitivity of data stored on the asset. A defined list of data types and sensitivity levels must be defined. Assets that do not contain data of the data classification types or sensitivity levels will have an "undefined" data classification and are not considered as "classified" for this metric.

4.12 Metric DSP-05-M1

Primary CCMv4 Control ID	DSP-05
Primary Control Description	Create data flow documentation to identify what data is processed, stored, or transmitted where. Review data flow documentation at defined intervals, at least annually, and after any change.
Related CCMv4 Control IDs	DSP-03
Metric ID	**DSP-05-M1**

Metric Description	This metric measures the percentage of records from the data inventory required by control DSP-03 that are included in data flow documentation. CSPs and their stakeholders can use this metric to determine whether the volume of data covered by the data flow documentation is sufficient or needs to be updated to satisfy defined business requirements.
Expression	Percentage: 100 * A/B A = Number of data records or data stores from the DSP-03 inventory correctly included in the data flow documentation B = Total number of data records or data stores in the DSP-03 inventory
Rules	This metric can be measured by counting the number of records in a data store or by simply counting the data stores themselves. "[C]orrectly included" means that the data record or data store is represented in the data flow documentation in accordance with the organization's defined requirements for representing inventories in the documentation. Generally, this means it exists in the documentation and is properly labeled with appropriate DSP-04 classifications if appropriate. Note: • The DSP-03 control objective is to "Create and maintain a data inventory, at least for any sensitive data and personal data." • The DSP-04 control objective is to "Classify data according to its type and sensitivity level."
SLO Recommendations	80%

Implementation Guidelines

This metric supports an incomplete DSP-03 inventory so long as it is a statistically significant random sampling of "at least [...] any sensitive data and personal data" (e.g., meets the DSP-03 control language objective).

This metric makes the assumption that the data flow diagram(s) is available in a machine-readable format but does not measure automated creation of the data inventory or the data flow documentation. The generation of the data flow document MAY be manual, although the result MUST be digitized in order to perform automated comparisons against discovered data repositories.

This metric assumes the data flow documentation is in the form of a graph with nodes and edges, where data stores are nodes in that graphs. In order to count the number of records, there needs to be metadata with the number of records for each datastore. It measures the percentage of data stores (and their records) that are correctly captured as nodes in the graph.

For reference, a "data flow inventory" similar to DSP-03 is required by CSA's *Code of Conduct for GDPR Compliance*,[12] Control #5: Data Transfer.

This should be evaluated every two weeks or in accordance with the organization's development release cycles.

4.13 Metric DSP-05-M2

Primary CCMv4 Control ID	DSP-05
Primary Control Description	Create data flow documentation to identify what data is processed, stored, or transmitted where. Review data flow documentation at defined intervals, at least annually, and after any change.
Related CCMv4 Control IDs	N/A
Metric ID	**DSP-05-M2**
Metric Description	This metric measures the percentage of data streams from the data inventory required by control DSP-03 that are included in the data flow documentation. CSPs and their stakeholders can use a metric like this to determine whether the different uses of data covered by the data flow documentation is sufficient or needs to be updated to satisfy defined business requirements.
Expression	Percentage: 100 * A/B A = Number of data streams from the DSP-03 inventory correctly included in the data flow documentation B = Total number of data streams in the DSP-03 inventory
Rules	"Data streams" are the connections from data sources to data consumers illustrated in data flow diagrams. These connections should be included in the data inventory required by control DSP-03. This may be a complete inventory of all data streams or a reasonable sample of data streams.
SLO Recommendations	80%

[12] Cloud Security Alliance. (n.d.). *Code of Conduct for GDPR Compliance*. Retrieved October 6, 2021, from https://cloudsecurityalliance.org/privacy/gdpr/code-of-conduct/

Implementation Guidelines

This metric supports an incomplete inventory of data streams so long as it is a reasonable sampling of streams for "at least [...] any sensitive data and personal data" (e.g., is intended to measure flows of data of the types that meet the DSP-03 control language objective regarding data inventories). Sampled data streams should be captured from live data streams of user and system activities.

This metric assumes the data flow documentation is available in a machine-readable format. The generation of the data flow document MAY be manual, although the result MUST be digitized in order to perform automated comparisons against discovered data flows.

For reference, a "data flow inventory" similar to DSP-03 is required by CSA's *Code of Conduct for GDPR Compliance*,[13] Control #5: Data Transfer.

This should be evaluated every two weeks, or in accordance with the organization's development release cycles

4.14 Metric GRC-04-M1

Primary CCMv4 Control ID	GRC-04
Primary Control Description	Establish and follow an approved exception process as mandated by the governance program whenever a deviation from an established policy occurs.
Related CCMv4 Control IDs	AIS-07
Metric ID	**GRC-04-M1**
Metric Description	This metric measures the effectiveness of the governance program's exception handling process.
Expression	Percentage: 100 * A/B A = Number of active policy exceptions where the time to resolution is within the documented timeline for resolution, during the sampling period B = Total number of active policy exceptions, during the sampling period

[13] Cloud Security Alliance. (n.d.). *Code of Conduct for GDPR Compliance*. Retrieved October 6, 2021, from https://cloudsecurityalliance.org/privacy/gdpr/code-of-conduct/

Rules	An exception policy must be defined and must cover the entire lifecycle of an exception. Active policy exceptions that happen during the sampling period but which are not resolved yet are counted in B, not A.
SLO Recommendations	90%

Implementation Guidelines

This metric requires organizations to maintain records of policy exceptions that include the approval date and resolution date for calculation of mean time to resolution. The records could be as simple as entries in a spreadsheet or as complex as records for exception tracking in a GRC or vulnerability management system.

This metric also requires organizations to define the threshold(s) for acceptable resolution time(s). The definition could be as simple as a statement in a policy document that applies to all exceptions, or individually-defined target dates for resolution of each exception, based on risk. In the case of the latter, the requirements for setting the target resolution date(s) should be established in a policy and the target date(s) will need to be tracked in the policy exception records.

For example, if there is a ticketing system for remediation this tracks if the close date for the ticket was met.

If an organization has very few exceptions, then slipping on even one will dramatically affect their percentage. This is inherent in statistics and is not seen as a problem for now.

4.15 Metric IAM-07-M1

Primary CCMv4 Control ID	IAM-07
Primary Control Description	De-provision or respectively modify access of movers/leavers or system identity changes in a timely manner in order to effectively adopt and communicate identity and access management policies.
Related CCMv4 Control IDs	IAM-03, IAM-06, IAM-10
Metric ID	**IAM-07-M1**

Metric Description	This metric measures the percentage of users leaving the organization that were deprovisioned from the identity management system in compliance with identity and access management policies.
Expression	Percentage of leavers de-provisioned in compliance with IAM policies: 100 * A/B A = Number of terminated users deprovisioned within policy guidelines during the sampling period B = Total number of terminated users during the sampling period
Rules	The time lapse between a user's termination and account deactivation must be measured in seconds. The time lapse between user's termination and account deactivation = time stamp of account deactivation event – time stamp of employee termination or role change event recorded in the HR system
SLO Recommendations	99%

Implementation Guidelines

Steps to compute this may look like:

1. Account deactivation timestamps can be obtained from the identity management system
2. Employee termination or change event timestamps can be obtained from the Human Capital Information System (e.g., Workday).

This metric only evaluates termination/deprovisioning events as an indicator of efficacy. It does not measure job role change, which can be captured in IAM-08.

The recommended sampling period for this metric is monthly, but CSPs should ensure the sampling period and frequency of evaluation align with their rate of change and risk tolerance.

4.16 Metric IAM-08-M2

Primary CCMv4 Control ID	IAM-08
Primary Control Description	Review and revalidate user access for least privilege and separation of duties with a frequency that is commensurate with organizational risk tolerance.

Related CCMv4 Control IDs	IAM-03, IAM-05, IAM-06, IAM-10
Metric ID	**IAM-08-M2**
Metric Description	This metric measures the time elapsed since the last recertification for all types of privileges (including user roles, group memberships, read/write/execute permissions to files/databases/scripts/jobs, etc.). The metric returns the longest time identified. For example, if the longest time elapsed for a recertification of a privilege is 95 days, the metric will return this number. The value returned should not be greater than the frequency of privilege recertification or review defined in the organization's policies.
Expression	Percentage: 100 * A/B A = Number of accounts reviewed with correct access in the last 90 days B = Total number of accounts
Rules	Date of last recertification is the date and time that a privilege was reviewed and recertified in the most recent recertification. If a date of last recertification does not exist, this should be replaced with the date a privilege was granted or an account was created.
SLO Recommendations	95%

Implementation Guidelines

The identity management system or system used to automate the account privilege recertification process (an example of this type of systems is Identity IQ by SailPoint) should maintain timestamps of account creations and privilege-granting events (e.g., addition to user groups, granting of security roles, etc.). These timestamps can be used to calculate this metric.

Orphaned accounts (i.e., accounts that have not been terminated at the time of measurement in IAM-07) should be captured by the User Access Review described in IAM-08.

This captures any problem in the process such as reviewing a small number of accounts resulting in a poor score or reviewing a large number of accounts and discovering them to be in error, also resulting in a bad score.

The measurement should be taken monthly to align with IAM-07, even if recertifications occur on a different periodicity.

4.17 Metric IAM-09-M1

Primary CCMv4 Control ID	IAM-09
Primary Control Description	Define, implement, and evaluate processes, procedures, and technical measures for the segregation of privileged access roles such that administrative access to data, encryption, and key management capabilities and logging capabilities are distinct and separated.
Related CCMv4 Control IDs	IAM-03, IAM-05, IAM-10
Metric ID	**IAM-09-M1**
Metric Description	This metric measures the segregation of duties of non-production staff having access to production roles and vice-versa.
Expression	Percentage of users with segregation of privileged access roles: 100*(1-(A/B)) A = Number of users with admin access to more than one of the following capabilities: production data management, encryption and key management, and logging B = Number of users with access to production data management, encryption and key management, or logging capabilities
Rules	Capabilities are privileged roles or functions. Examples of production data management capabilities are the AmazonRDSFullAccess policy in AWS, the Cloud SQL Admin & Cloud SQL Editor roles in the Google Cloud Platform (GCP), and db_owner role for a Microsoft Azure SQL Database. Examples of encryption and key management capabilities are the AWSKeyManagementServicePowerUser policy in AWS, Cloud KMS Admin with Cloud KMS CryptoKey Encrypter/Decrypter roles in GCP, or Microsoft Azure Key Vault Admin. Examples of logging capabilities are the AWSCloudTrail_FullAccess policy in AWS, Monitoring Admin & Editor roles in GCP, or Monitoring Contributor in Microsoft Azure.
SLO Recommendations	99%

Implementation Guidelines

1. Identify privileged roles in an organization and map to the roles identified in this metric.
2. Run the metric across all users with privilege.

In just-in-time (JIT) access capabilities, the audit should evaluate the ability for a user to be provisioned the privilege, even if the individual did not request the privilege during the measurement.

4.18 Metric IPY-03-M2

Primary CCMv4 Control ID	IPY-03
Primary Control Description	Implement cryptographically secure and standardized network protocols for the management, import, and export of data.
Related CCMv4 Control IDs	CEK, IVS-02, IPY-02, DSP-05
Metric ID	**IPY-03-M2**
Metric Description	This metric measures the percentage of data flows that use an approved, standardized cryptographic security function for interoperable transmissions of data.
Expression	Percentage of data flows that use cryptographically secure and standardized network protocols: 100 * A/B A = Count of data flows that use an approved, standardized cryptographic security function B = Count of all data flows
Rules	This metric depends on a known inventory of data flows such as is required by DSP-05. This inventory may be built from IPY-02 and/or DSP-05 (see DSP-05-M2), or other options could exist (e.g., a data flow might be counted as an asset type in a DCS-06 asset inventory). The count of all data flows is the count of items in the inventory used to satisfy DSP-05. Approved cryptographic security functions should be established by an organization policy or standard, as required by CEK-01.

Rules, cont.	Determining which data flows use an approved cryptographic security function can occur using analytics on the encrypted traffic flows or can occur by examining the associated configurations using the tooling from AIS-06 or from the CCC domain.
SLO Recommendations	99.99%

Implementation Guidelines

NIST 140-2 Annex A is a plausible set of interoperability-specific policy choices for standard cryptographic security functions. Other regions might drive different choices.

This metric should be a continuous measure over the previous hour. For example: over the previous hour, 86% of protocol flows were detected to be TLS 1.3 with selected cipher suites, gRPC, remote access VPN, or other types within the current policy set and listed in an interoperability specific policy to ensure interoperability.

4.19 Metric IVS-04-M1

Primary CCMv4 Control ID	IVS-04
Primary Control Description	Harden host and guest OS, hypervisor, or infrastructure control plane according to their respective best practices, and supported by technical controls, as part of a security baseline.
Related CCMv4 Control IDs	DCS-06, CCC-03, CCC-07
Metric ID	**IVS-04-M1**
Metric Description	This metric measures the percentage of assets in compliance with the provider's configuration security policy and hardening baselines derived from accepted industry sources (e.g., NIST, vendor recommendations, Center for Internet Security Benchmarks, etc.).
Expression	Percentage: 100 * A/B A = Number of production assets that are in compliance with hardening baseline B = Total number of production assets

Rules	Examples of hardening baselines include Center for Internet Security Benchmarks, DISA STIGs, vendor-recommended best practices, NIST security guidance, etc.
SLO Recommendations	99.99%

Implementation Guidelines

This metric of "assets that are in compliance" is inclusive of assets that have failed an initial test and where remediation is still within the SLA timeframe. If an asset is not fixed within the timeframe, it impacts the metric.

Hardening baselines derived from accepted industry sources (e.g., NIST, vendor recommendations, Center for Internet Security Benchmarks, etc.) and in compliance with the provider's configuration security policy are expressed in test code, which is run against the targeted asset on a regular basis.

If an asset fails these tests, an alert is generated and the team is expected to fix the problem within a policy-defined SLA timeframe (likely inclusive of risk thresholds for various timeframes).

4.20 Metric LOG-03-M1

Primary CCMv4 Control ID	LOG-03
Primary Control Description	Identify and monitor security-related events within applications and the underlying infrastructure. Define and implement a system to generate alerts to responsible stakeholders based on such events and corresponding metrics.
Related CCMv4 Control IDs	N/A
Metric ID	**LOG-03-M1**
Metric Description	This metric measures the percentage of logs configured to generate security alerts for anomalous activity across control domains such as: Application & Interface Security; Business Continuity Management, Change Control & Configuration Management; Identity & Access Management; Infrastructure & Virtualization Security; Threat & Vulnerability Management; and Universal Endpoint Management.

Expression	Percentage of logs configured with security alerts: 100 * A/B A = Number of log sources with security alerts configured B = Total number of log sources
Rules	Log sources can be the system log(s) or input(s) to the logging pipeline(s). Security alerts include traditional alerts triggered when a log records events in a control domain above a specified threshold, as well as alerts generated by anomaly detection using machine learning.
SLO Recommendations	95%

Implementation Guidelines

This metric measures alerts based on items of interest occurring within a log.

This metric requires CSPs to have an inventory of log sources or inputs for their logging pipeline(s) and the ability to determine a unique count of those log sources or inputs to the logging pipeline with anomaly detection or security alerts configured for them.

An example implementation may look like this:

4.21 Metric LOG-05-M1

Primary CCMv4 Control ID	LOG-05
Primary Control Description	Monitor security audit logs to detect activity outside of typical or expected patterns. Establish and follow a defined process to review and take appropriate and timely actions on detected anomalies.
Related CCMv4 Control IDs	LOG-03, LOG-01
Metric ID	**LOG-05-M1**

Metric Description	This metric reports the effectiveness of the log monitoring and response process by measuring the percentage of discovered anomalies resolved within required timelines.
Expression	Percentage of anomalies resolved in compliance with policy: 100 * A/B if B is not 0, or 100 when B is 0 A = Number of anomalies detected during the sampling period that were reviewed and resolved within a timeframe that is in compliance with policy B = Total number of anomalies detected during the sampling period.
Rules	Anomalies that have been detected during the sampling period but have not been reviewed and resolved during the sampling period are not counted in A. An anomaly is any event happening outside of typical or expected patterns.
SLO Recommendations	95%

Implementation Guidelines

Activity "outside of typical or expected patterns" is something for the CSP to define. A common mechanism is to use indicators of compromise to detect anomalies. For example, see OASIS STIX Version 2.1, 4.6 Indicator.[14]

If no anomalous events are detected during the sample period, the resulting metric (a divide by zero error) is not included in the metrics reported.

4.22 Metric LOG-10-M1

Primary CCMv4 Control ID	LOG-10
Primary Control Description	Establish and maintain a monitoring and internal reporting capability over the operations of cryptographic, encryption, and key management policies, processes, procedures, and controls.

[14] OASIS. (2020, March 20). *STIX Version 2.1: 4.6 Indicator*. https://docs.oasis-open.org/cti/stix/v2.1/cs01/stix-v2.1-cs01.html#_muftrcpnf89v

Related CCMv4 Control IDs	CEK-03, CEK-04, CEK-05, CEK-06, CEK-07, CEK-08, CEK-09, CEK-10, CEK-11, CEK-12, CEK-13, CEK-14, CEK-15, CEK-16, CEK-17, CEK-18, CEK-19, CEK-20, CEK-21
Metric ID	**LOG-10-M1**
Metric Description	This metric measures the percentage of cryptography, encryption, and key management controls with defined metrics.
Expression	Percentage of encryption controls with defined metrics: 100 * A/B A = Number of metrics reported in the CEK domain B = Total number of controls in the CEK domain
Rules	N/A
SLO Recommendations	80%

Implementation Guidelines

This requires defining metrics beyond the minimal set currently defined in order to meet the recommended SLO.

Metrics for all CEK controls may not be easily automated, for example CEK-01, CEK-02, CEK-06, CEK-07, and CEK-08.

This is measured against the total number of controls in the CEK domain, rather than the number of controls asserted as met in the last audit. This simplifies the metric, as the implementers do not need programmatic access to the previous audit results.

Generally, the recommended frequency should be the maximum frequency recommended for CEK metrics.

4.23 Metric LOG-13-M2

Primary CCMv4 Control ID	LOG-13
Primary Control Description	Define, implement and evaluate processes, procedures and technical measures for the reporting of anomalies and failures of the monitoring system and provide immediate notification to the accountable party.
Related CCMv4 Control IDs	LOG-03, LOG-08, SEF-06
Metric ID	**LOG-13-M2**
Metric Description	This metric measures "failures [e.g., uptime] of the monitoring system." The other aspects of this control such as "reporting of anomalies" and "immediate notification to the accountable party" are to be measured using other metrics.
Expression	Percentage of uptime: 100 * A/B A = Number of minutes of uptime during the sampling period B = Duration of the sampling period in minutes
Rules	Uptime = (total number of minutes in the sampling period - downtime in minutes during the sampling period) Downtime = any minute where health checks for any component of the monitoring system failed
SLO Recommendations	99%

Implementation Guidelines

"Minutes" provides sufficient granularity to measure uptime up to a target of five nines. It should be noted, though, that the recommended frequency of evaluation is daily rather than yearly and therefore a five nines score during any particular day cannot be extrapolated as a yearly uptime. This reflects the objective of measuring and reporting on potential failures of the monitoring system for "immediate" notification at least daily.

To determine if a system is up," a health check is expected. This metric does not mandate a specific health check. Many uptime monitoring solutions exist that can be used as implementation examples

or as services. A recommended level of health check is one that tests the functionality of the monitoring system during the minute of the test. For example, a simple TCP/IP ping measures "uptime" but is insufficient to measure the availability of the functionality of a monitoring system. Testing that log entries are being persistently recorded during that minute is a more accurate measure of uptime availability.

The LOG-03 monitoring and alerting objective can reasonably be met by deploying multiple monitoring and alerting systems that are responsible for different areas of a complex environment. If multiple independent monitoring systems are deployed and only one fails a health check during any given minute, is the system as a whole "up" or "down" during that minute? For the purpose of this metric, if any monitoring and alerting system fails a health check during a minute then the system as a whole is considered to be "down" during that minute. This is simplistic, easy, and accurately captures the increased complexity of running multiple monitoring systems.

This simplification does not, however, support considerations like "this subset monitoring and alerting system only covers a small number of low risk elements of the infrastructure." Future versions of this metric may include "coverage" or "risk" elements to the metric expression.

4.24 Metric SEF-05-M1

Primary CCMv4 Control ID	SEF-05
Primary Control Description	Establish and monitor information security incident metrics.
Related CCMv4 Control IDs	N/A
Metric ID	**SEF-05-M1**
Metric Description	This metric measures the percentage of security events sourced from automated systems.
Expression	Percentage of security events from automated systems: 100 * A/B A = Number of security events sourced from automated systems during the sampling period B = Total number of security events that were recorded during the sampling period

Rules	The log sources configured with security alerts for the LOG-03-M1 metric are examples of automated systems.
SLO Recommendations	90%

Implementation Guidelines

Automated systems include logging and monitoring systems as well as systems that generate alerts for review, including threat intelligence systems.

Security events manually entered by individuals or organizations for triage are from "non-automated systems," (e.g., vulnerability disclosure emails, security event tickets created by staff or customers, etc.).

4.25 Metric SEF-06-M1

Primary CCMv4 Control ID	SEF-06
Primary Control Description	Define, implement, and evaluate processes, procedures, and technical measures supporting business processes to triage security-related events.
Related CCMv4 Control IDs	LOG-03, SEF-01, SEF-05
Metric ID	**SEF-06-M1**
Metric Description	This metric measures the percentage of security events triaged within policy timeframe targets.
Expression	Percentage of security events triaged in compliance with policy: 100 * A/B A = Number of security events triaged within policy-defined time limit during the sampling period B = Total number of security events logged during the sampling period
Rules	Policy targets as established in SEF-01 are used here as a proxy for "within a reasonable time." This metric is manipulatable by selecting an easy-to-achieve policy target, but doing so should create friction during the initial audit.

SLO Recommendations	99%

Implementation Guidelines

Events occur and are classified as part of triage process. This can occur automatically and/or there can be manual triage steps. Once the event reaches its final categorization, it is "triaged." As long as this completes within the organization's target time period, it is "within the SLO."

It may be aggressive for a small organization that does not have a lot of events to report this metric frequently.

4.26 Metric SEF-06-M2

Primary CCMv4 Control ID	SEF-06
Primary Control Description	Define, implement, and evaluate processes, procedures, and technical measures supporting business processes to triage security-related events.
Related CCMv4 Control IDs	LOG-03, SEF-01, SEF-05
Metric ID	**SEF-06-M2**
Metric Description	This metric indicates if security event triage process times are stable, improving, or worsening.
Expression	Slope represented as a percentage: A = SLOPE(triage times for security events, dates for security events) * 100
Rules	The SLOPE is the of the linear regression of the triage times as graphed against the dates (or sequence numbers) for security events within the time period.
SLO Recommendations	< 0 A slope of 0 means the triage process is stable A slope of <0 means the triage process is improving A slope of >0 means the triage process is worsening

Implementation Guidelines

Events occur and are classified as part of triage process. This can occur automatically and/or there can be manual triage steps. Once the event reaches its final categorization it is "triaged." As long as this completes within the organization's target time period, it is "within the SLO."

The slope of time to triage indicates if the event triage process has improved, stayed the same, or increased (worsened).

This metric does not capture if the triage time is within a specific policy target. It only captures that the organization has in fact defined, implemented, and has a process for evaluating their triage process. This meets the objective of the control.

This can be implemented in spreadsheets as the "SLOPE" function within formulas and charts (see Excel or Sheets[15]).

4.27 Metric STA-07-M3

Primary CCMv4 Control ID	STA-07
Primary Control Description	Develop and maintain an inventory of all supply chain relationships.
Related CCMv4 Control IDs	DCS-06, LOG-03
Metric ID	**STA-07-M3**
Metric Description	The percentage of third-party software components seen in [production assets] that are sourced from an approved supplier in the software inventory.
Expression	Percentage: 100 * A/B A = Total number of third-party software components "seen" during the sampling period that are from authorized providers B = Total number of third-party software components "seen" during the sampling period

[15] Google. (n.d.). *SLOPE - Docs Editors Help*. Google Docs Editors Help. Retrieved October 6, 2021, from https://support.google.com/docs/answer/3094048?hl=en

Rules	N/A
SLO Recommendations	99.9%

Implementation Guidelines

A software component is a discrete unit of software, such as a library or package, with uniquely identifiable attributes.

A simplistic approach is to track all software libraries and ensure they are in the inventory of approved libraries. A more advanced approach is to use context to determine if the software should be running on this particular asset.

For example: Bastion (jumphost) software may be approved for use on a hardened bastion asset but may not be appropriate for a non-hardened asset.

The implementor SHOULD have sufficient context in the STA-07 inventory to make this distinction. It is not mandated.

The use of "seen" allows for sampling. There is nothing currently in the metric to expose how statistically significant the sampling was. It is assumed that an initial audit confirmed significant sampling was used.

4.28 Metric STA-07-M5

Primary CCMv4 Control ID	STA-07
Primary Control Description	Develop and maintain an inventory of all supply chain relationships.
Related CCMv4 Control IDs	LOG-05, LOG-03
Metric ID	**STA-07-M5**
Metric Description	The percentage of approved supply chain upstream cloud services relationships that are not recorded in logged data connections.

Expression	Percentage: 100 * A/B A = The total number of unique providers with observed connections B = Total number of unique providers in the inventory Both A and B are measured over the same sampling period.
Rules	N/A
SLO Recommendations	99%

Implementation Guidelines

This measurement requires a list of CSP Connections that are approved and expected and an ability to log all connections to expected endpoints of those providers.

4.29 Metric TVM-03-M1

Primary CCMv4 Control ID	TVM-03
Primary Control Description	Define, implement, and evaluate processes, procedures, and technical measures to enable both scheduled and emergency responses to vulnerability identifications, based on the identified risk.
Related CCMv4 Control IDs	TVM-08
Metric ID	**TVM-03-M1**
Metric Description	This metric measures the percentage of high and critical vulnerabilities that are remediated within the organization's policy timeframes. This reflects the time between when a vulnerability is identified on an organization's assets and when remediation is complete.
Expression	Percentage: 100 * A/B A = Number of high and critical vulnerabilities identified during the sampling period and remediated within policy timeframes B = Total number of high and critical vulnerabilities identified during the sampling period

Rules	High and critical vulnerabilities are defined consistent with the implementation of TVM-08. If a vulnerability is identified but not yet remediated when the measurement is made, the measurement date is used as the remediation date in order to evaluate if the vulnerability has been mitigated within the defined policy timeframe, as expected for the calculation of A.
SLO Recommendations	99%

Implementation Guidelines

To compute the denominator: The "total number of high and critical vulnerabilities" are any such vulnerabilities that are still open from previous periods plus all such newly identified during the current sample period. A minimal example framework for vulnerability prioritization is CVSS v3.0, where "high" and "critical" are defined as 7.0 and above.

To compute the numerator:

1. Fetch all critical or high vulnerabilities newly identified during the current period
2. Fetch all critical or high vulnerabilities that are still open (not closed) from the previous period

For example, assume the following data sets for three example weekly periods:

Example period 1: 04/25–05/01

(a) Number of critical + high vulnerabilities created during the period = 10
(b) Number of critical + high vulnerabilities still open from previous periods = 3
As of the beginning of the current period, 04/25
(c) Number of critical + high vulnerabilities closed during the period within policy = 8
(d) Total number of critical + high vulnerabilities closed = 8

Example period 2: 05/02–05/08

(e) Number of critical + high vulnerabilities created during the period = 15
(f) Number of critical + high vulnerabilities still open from previous period = 5
as of the beginning of the current period, 05/02
e.g., a+b-d, but in an actual implementation this is possibly determined with a database query
(g) Number of critical + high vulnerabilities closed during the period within policy = 14
(h) Total number of critical + high vulnerabilities closed = 20

Metric for the period 05/02–05/08: Numerator = (g)/[(e) + (f)] = 14/(15+5) = 70%

Example period 3: 05/09–05/15

(i) Number of critical + high vulnerabilities created during the period = 5
(j) Number of critical + high vulnerabilities still open from previous periods = 0 (e+f-h) *as of the beginning of the current period, 05/02*
(k) Number of critical + high vulnerabilities closed during the period within policy = 4
(l) Total number of critical + high vulnerabilities closed = 4

Metric for the period 05/09–05/15: Numerator = (k)/[(i) + (j)] = 4/(5+0) = 80%

4.30 Metric TVM-07-M1

Primary CCMv4 Control ID	TVM-07
Primary Control Description	Define, implement, and evaluate processes, procedures, and technical measures for the detection of vulnerabilities on organizationally managed assets at least monthly.
Related CCMv4 Control IDs	TVM-07, UEM-14, DCS-06
Metric ID	**TVM-07-M1**
Metric Description	This metric measures the percentage of managed assets scanned monthly.
Expression	Percentage: 100 * A/B A = Number of assets from the organization's asset catalog that have been scanned during the sampling period B = Total number of assets in the organization's asset catalog
Rules	The "asset catalog" refers to the cataloging requirements of DCS-06, which requires "catalogue[ing] and track[ing] all relevant physical and logical assets located at all of the CSP's sites within a secured system."
SLO Recommendations	99%

Implementation Guidelines

This metric requires organization-managed assets to be maintained in the catalog required by DCS-06. The asset catalog must be integrated with the vulnerability management process to track when assets in the catalog are scanned.

4.31 Metric TVM-10-M1

Primary CCMv4 Control ID	TVM-10
Primary Control Description	Establish, monitor, and report metrics for vulnerability identification and remediation at defined intervals.
Related CCMv4 Control IDs	AIS-07, TVM-08
Metric ID	**TVM-10-M1**
Metric Description	This metric measures the percentage of publicly known vulnerabilities that are identified for an organization's assets within the organization's required timeframes. The purpose of this metric is to determine how long it takes an organization to start tracking vulnerabilities for triage. This measure is important because Palo Alto Networks reported that Internet assets are scanned once every 15 minutes or less after CVEs are published. This metric does not include the time to remediation (which is measured by TVM-03-M1).
Expression	Percentage: 100 * A/B A = Number of high and critical vulnerabilities identified for remediation within policy timeframes B = Total number of high and critical vulnerabilities identified or carried-over into the sampling period

Rules	**To compute the numerator:** Determine the total number of high and critical vulnerabilities that have been identified for remediation per TVM-01 policies and procedures. In order to compute the metric, use the following logic: For each high or critical vulnerability that were identified during the period or carried forward from the previous period: 1. Check the vulnerability publish date. Is this date < the date on which the asset was commissioned? If so, use the asset commission date as the "from date"; if not, use the vulnerability publish date as the "from date." 2. Subtract the vulnerability identification date from the "from date." The identification date is the date that your organization has acknowledged the vulnerability to be acted upon (this may be the ticket create date on Jira for the given vulnerability). 3. Evaluate if #b > the policy duration (in days). If so, add +1 to the count. **To compute the denominator:** The "total number of high and critical vulnerabilities": 1. Opened during the current period and/or 2. Carried over from the previous period because the policy timeframe spans period. For example, let us say that we measure the control on a weekly basis: Sunday–Saturday, and the policy timeframe is 3 days. Any issue identified on or after Thursday is not due, per policy, until the following period. These vulnerabilities are "carried over" into the following period.
SLO Recommendations	N/A

Implementation Guidelines

1. Classification of vulnerabilities as "high" or "critical" risk should be defined in the vulnerability management tool based on an industry-accepted scoring system such as the Common Vulnerability Scoring System (CVSS)[16] or risk-based vulnerability management system (RBVM).[17] For instance, vulnerabilities with a CVSS score of nine or higher are "critical," and vulnerabilities with CVSS scores between seven and nine could be defined as "high" risk.
2. Date and time of vulnerability discovery could be obtained from the vulnerability management tool as it scans and detects high and critical vulnerabilities for remediation

[16] National Institute of Standards and Technology. (n.d.). *National Vulnerability Database: Vulnerability Metrics*. NIST. Retrieved October 6, 2021, from https://nvd.nist.gov/vuln-metrics/cvss

[17] Rolleston, J. (2020, July 9). *What is Risk-Based Vulnerability Management?* Kenna Security. https://www.kennasecurity.com/blog/what-is-risk-based-vulnerability-management/

3. Date and time of vulnerability remediation could be obtained in the following ways:
    a. From the vulnerability management tool as it scans and finds that a previously detected vulnerability is no longer present/detected.
    b. From the patch deployment tool (e.g., SCCM) as it successfully deploys and installs a patch that fixes an identified vulnerability.
    c. From the application/code release tool as moves into production the new version of the application that no longer contains the code vulnerability.

This metric depends on a policy timeline target for the identification (completion of the triage process) for known vulnerabilities.

4.32 Metric UEM-04-M1

Primary CCMv4 Control ID	UEM-04
Primary Control Description	Maintain an inventory of all endpoints used to store and access company data.
Related CCMv4 Control IDs	LOG-05
Metric ID	**UEM-04-M1**
Metric Description	This metric provides an indication of endpoints that are actively maintained in the asset inventory.
Expression	Percentage: 100 * A/B A = Number of endpoints that meet the following two conditions simultaneously: A1: The endpoint is seen in security audit logs as a result of activities outside of typical or expected patterns (cf. LOG-05). A2: The endpoint is in the company's inventory of all endpoints used to store and access company data (cf. UEM-04). B = The total number of endpoints in the company's inventory all endpoints used to store and access company data (cf. UEM-04). The datasets referred in A2 and B are the same. Condition A1 is limited to security log events that happened during the sampling period.

Rules	The data used in the expression is all data from the control period. This metric assumes that the following two CCMv4 controls are in place (or an equivalent): • LOG-05: Monitor security audit logs to detect activity outside of typical or expected patterns. • UEM-04: Maintain an inventory of all endpoints used to store and access company data.
SLO Recommendations	95%

Implementation Guidelines

The data used in the expression is all data from the control period.

This assumes the LOG-05 logs ("security audit logs to detect activity outside of typical or expected patterns") are inclusive of endpoint identities.

The frequency of evaluation for UEM-04 must match the length of time log data is maintained.

Examples of endpoints that store or access company data include end-user devices, point of sale systems, databases, IoT systems, and data integration systems.

4.33 Metric UEM-05-M1

Primary CCMv4 Control ID	UEM-05
Primary Control Description	Define, implement, and evaluate processes, procedures, and technical measures to enforce policies and controls for all endpoints permitted to access systems and/or store, transmit, or process organizational data.
Related CCMv4 Control IDs	UEM-04
Metric ID	**UEM-05-M1**
Metric Description	This metric describes the ability of an organization to control the configuration and behavior of assets which directly create, read, write, or delete organizational data.

Expression	Percentage: 100 * A/B A = Number of unique endpoints with suitable policy enforcement tools that have reported compliance state within the sampling period B = The total number of endpoints in the company's inventory of all endpoints used to store and access company data (cf. UEM-04)
Rules	This metric assumes that the following CCMv4 control are in place (or an equivalent): UEM-04: Maintain an inventory of all endpoints used to store and access company data. The capability to measure and enforce desired configuration and/or policy must be a discrete, measurable entity, such as an agent, an implementation constraint (e.g., containers or read-only filesystems), or an external control (e.g., network authentication and access policies or software-defined networking). In order to provide this measurement, the discrete capability must be an approved mechanism or mechanisms whose implementation is mandated by policy. In order for this measurement to be meaningful, UEM-04 must have a measurement greater than 95%.
SLO Recommendations	99%

Implementation Guidelines

If there are devices that are in an exception group, they still count as a policy control being applied for the purposes of this metric.

This metric does not differentiate between partial reporting and full reporting of all of the policies from a given system; it only concerns the capability of that system to report.

See UEM-04 for examples of systems which access or store organizational data.

Technical measures to enforce policies and controls for endpoints include API tools such as OSQuery, DCM tools, MDM tools, VPN Access Policy Controls, etc.

4.34 Metric UEM-09-M1

Primary CCMv4 Control ID	UEM-09
Primary Control Description	Configure managed endpoints with anti-malware detection and prevention technology and services.
Related CCMv4 Control IDs	TVM-02, DCS-05, DCS-06, DSP-01
Metric ID	**UEM-09-M1**
Metric Description	This metric measures the percentage of instances which are an running anti-malware/virus service.
Expression	Percentage: 100 * A/B A = The total number of managed endpoints from employee devices initiating observed connections where a device check inclusive (of verifying malware protection) has been passed B = The total number of managed endpoints from employee devices initiating observed connections
Rules	"Device check" is some form of posture assessment during connection or path establishment. Some examples: • Trusted platform module posture assessment • VPN posture assessment • ZTNA/MF posture assessment • Performing "out of band" checks by correlating connection log information with independent posture assessment monitoring
SLO Recommendations	99%

Implementation Guidelines

This depends on an asset database such as from DCS-06. The targeted classifications of assets in scope must be identified in DCS-05 (e.g., "employee devices").

References

Cloud Security Alliance. (n.d.-a). *Code of Conduct for GDPR Compliance*. Retrieved October 6, 2021, from https://cloudsecurityalliance.org/privacy/gdpr/code-of-conduct/

Cloud Security Alliance. (n.d.-b). *Security, Trust, Assurance, and Risk (STAR)*. Retrieved October 7, 2021, from https://cloudsecurityalliance.org/star/

Contrast Security. (n.d.). *2020 Application Security Observability Report*. Retrieved October 6, 2021, from https://www.contrastsecurity.com/hubfs/2020-Contrast-Labs-Application-Security-Observability_Annual_Report_07152020.pdf

CSA Cloud Controls Matrix Working Group. (2021, June 7). *Cloud Controls Matrix and CAIQ v4*. Cloud Security Alliance. https://cloudsecurityalliance.org/artifacts/cloud-controls-matrix-v4/

Google. (n.d.). *SLOPE - Docs Editors Help*. Google Docs Editors Help. Retrieved October 6, 2021, from https://support.google.com/docs/answer/3094048?hl=en

Jaquith, A. (2007). *Security Metrics: Replacing Fear, Uncertainty, and Doubt* (1st ed.). Addison-Wesley Professional.

Jones, C., Wilkes, J., Murphy, N., & Smith, C. (2017). *Site Reliability Engineering: Service Level Objectives*. Google Site Reliability Engineering. https://sre.google/sre-book/service-level-objectives/

National Institute of Standards and Technology. (n.d.). *National Vulnerability Database: Vulnerability Metrics*. NIST. Retrieved October 6, 2021, from https://nvd.nist.gov/vuln-metrics/cvss

National Institute of Standards and Technology. (2020, June 22). *Computer Security Resource Center: Automated Cryptographic Validation Testing*. NIST. https://csrc.nist.gov/Projects/Automated-Cryptographic-Validation-Testing

OASIS. (2020, March 20). *STIX Version 2.1: 4.6 Indicator*. https://docs.oasis-open.org/cti/stix/v2.1/cs01/stix-v2.1-cs01.html#_muftrcpnf89v

OpenMetrics. (n.d.). *The OpenMetrics project — Creating a standard for exposing metrics data*. Retrieved October 7, 2021, from https://openmetrics.io/

Pannetrat, A. (2020, March 20). *Continuous Auditing and Continuous Certification*. Cloud Security Alliance. https://cloudsecurityalliance.org/blog/2020/03/20/continuous-auditing-and-continuous-certification/

Rolleston, J. (2020, July 9). *What is Risk-Based Vulnerability Management?* Kenna Security. https://www.kennasecurity.com/blog/what-is-risk-based-vulnerability-management/

Wikipedia contributors. (2021, October 6). *Chaos engineering*. Wikipedia. https://en.wikipedia.org/wiki/Chaos_engineering