Feature

Implementing an Information Security Continuous Monitoring Solution—A Case Study

Tieu Luu is director of research and product development for SuprTEK, where he leads the development of innovative products and services for the company, including the PanOptes Continuous Monitoring Platform.

The threats to government computer systems and networks continue to evolve and grow due to steady advances in the sophistication of attack technology, the ease of obtaining such technology, and the increasing use of these techniques by state and nonstate actors to gain intelligence and/or disrupt operations. The US Government Accountability Office (GAO) cites that from 2006 to 2012, the number of cyberincidents reported by federal agencies to the US Computer Emergency Readiness Team (US-CERT) grew from 5,503 to 48,562, an increase of 782 percent.[1]

As one of the responses to this growing threat, the executive branch of the US government has established as one of its cross agency priority (CAP) goals[2] the continuous monitoring of federal information systems to enable departments and agencies to maintain an ongoing near-real-time awareness and assessment of information security risk and rapidly respond to support organizational risk management decisions. In November 2013, the US Office of Management and Budget (OMB) issued memorandum M-14-03 requiring all federal departments and agencies to establish an information security continuous monitoring (ISCM) program.[3] The US Department of Homeland Security (DHS) has been tasked to work with all of the departments and agencies to help them implement continuous monitoring through the Continuous Diagnostics and Mitigation (CDM) program.

To help it comply with the OMB mandate, one large US government agency has contracted with SuprTEK, an IT engineering and professional services firm, to develop a continuous monitoring system that is responsible for monitoring millions of devices across a globally distributed network. The system has enabled the client to improve its processes for risk and vulnerability management, certification and accreditation (C&A), compliance and reporting, and secure configuration management, greatly improving the security posture of its systems and saving countless work hours by automating many of the previously manual processes.

DEFINING ISCM

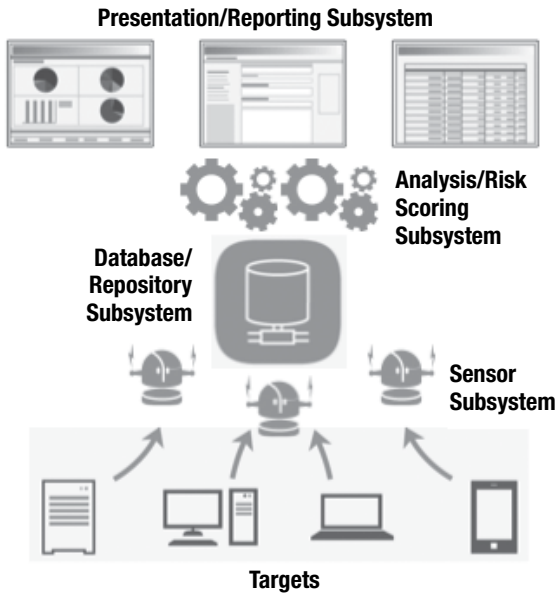
So what exactly is ISCM? "Information security continuous monitoring is defined as maintaining ongoing awareness of information security, vulnerabilities and threats to support organizational risk management decisions."[4] This means continuously collecting information to provide a comprehensive understanding of everything that is deployed on an enterprise's networks and using this information to assess compliance against security policies and exposure to threats and vulnerabilities. This information provides IT managers with a comprehensive and up-to-date inventory of assets and how they are configured so that they understand what is on their networks and where the networks may be vulnerable. It helps system administrators properly prioritize vulnerabilities based on how pervasive they may be across the enterprise and their potential impact to the mission or business, rather than trying to patch everything and continuously play catch-up with newly discovered vulnerabilities. The information provides auditors with up-to-the-minute information on each system's security posture so that they can properly decide whether or not a system should be approved to go live on the production network or be taken offline if a critical finding is not properly remediated or mitigated. The collected information is also entered into a set of risk-scoring algorithms to quantify the security posture across the entire enterprise and identify and prioritize the worst problems to fix first so that executives can focus their scarce IT resources.

IMPLEMENTATION ARCHITECTURE

A continuous monitoring system is essentially a data analytics application, so at a high level, the architecture for a continuous monitoring system, depicted in **figure 1**, resembles that of most typical data analytics/business intelligence (BI) applications. DHS has defined a technical

reference architecture for continuous monitoring called the Continuous Asset Evaluation, Situational Awareness, and Risk Scoring (CAESARS) reference architecture[5] based on the work of three leading US federal agencies that have successfully implemented continuous monitoring solutions: the US Department of State (DOS), the US Internal Revenue Service (IRS) and the US Department of Justice (DOJ).

Figure 1–Continuous Monitoring System Architecture



Source: Tieu Luu. Reprinted with permission.

The CAESARS reference architecture represents the essential functional components of an ISCM and risk-scoring system, as depicted in **figure 1**. The four functional subsystems defined by CAESARS are:

- **Sensor subsystem**—Responsible for collecting data such as hardware and software inventory, configurations, compliance and vulnerabilities from the targets (i.e., assets or devices such as the computing, network and mobile devices on an enterprise's networks). The sensor subsystem may be composed of agent-based and agentless software, as well as hardware devices that scan the devices and networks and send data back to the database/repository subsystem.
- **Database/repository subsystem**—Responsible for storing the findings collected by the sensor subsystem. The database/repository subsystem is also responsible for storing and managing the technical security policies and implementation guidance that define how the targets should be configured. Targets are assessed against these baseline configurations to determine compliance and how well they are secured.
- **Analysis/risk-scoring subsystem**—Responsible for correlating, fusing, deconflicting and deduplicating the findings collected by the sensor subsystem in addition to assessing compliance of the findings against the baselines. Once the collected data have been processed by the analysis capabilities, the risk-scoring capabilities are responsible for using this information to quantify security posture and risk of the enterprise using algorithms that take into account the severity of the findings, the probability of exploit and the impact of successful exploit.
- **Presentation and reporting subsystem**—Responsible for presenting the results of the analysis and risk-scoring subsystem through various dashboards and reports to "motivate administrators to reduce risk; motivate management to support risk reduction; inspire competition; and measure and recognize improvement."[6] The subsystem has to be able to present information at an aggregate level across the enterprise as well as to be able to drill down into specific devices and findings to support remediation.

DATA INTEGRATION CHALLENGES

As with most data analytics/BI applications, data integration presents many challenges for a continuous monitoring system. Most large enterprises have multiple tools that make up the sensor subsystem, e.g., they may use a network access control (NAC) solution to detect devices, vulnerability scanners to detect vulnerabilities on devices, code analyzers and scanners to detect software flaws, and configuration scanners to assess compliance against security policies. Thus, it becomes the classic master data management (MDM) problem where the complete picture of an IT asset (e.g., hardware, operating

system, software applications, patches, configuration, vulnerabilities) has to be pieced together from disparate systems. Some of the key challenges with trying to piece together all of the required data from these types of tools are described in **figure 2**.

A data ingest capability was implemented as an asynchronous layer around the database/repository subsystem with a Secure Content Automation Protocol (SCAP)-based[7] interface to consume data from the sensor subsystem. As mentioned, the use of SCAP alleviated some integration challenges by enabling a common format, but also created other challenges due to variations in implementation by the different sensors. Ultimately, those variations were accounted for via the use of different interpreters based on version information in the data that are received by the ingester. Techniques from MDM were applied to address some of the other data integration challenges. For example, cross-referencing is a common technique in MDM where a master table is defined for an entity that contains all of the potential identifiers for that entity across the disparate systems. In this case, the cross-reference capability defined a master identifier for devices and also contained all of the other identifiers for devices used by the various sensor tools (e.g., MAC address, Internet Protocol [IP] address, host name) that were used to match the findings from the sensors to the correct device. There was no panacea to address the challenges with data completeness and quality. It required a great deal of close monitoring and validation when integrating sensor data from a new site and working with the site's administrators to correct the issues that were identified. Various system reports were used to check for completeness and quality (e.g., what sites were publishing data and what data they were publishing). To deal with issues around overlapping and conflicting findings from different sensors, a trust model that defined which sensors to trust for which types of findings (i.e., for findings of this type, trust the results from sensor A over the results from sensor B) was implemented. For example, for vulnerability assessments, the results from authenticated, agent-based scanners were considered more credible than the results from agentless, network-based scanners.

Figure 2—Examples of Key Data Integration Challenges

Challenge	Examples
Asset identification	Different tools use different ways to identify devices (e.g., MAC address, IP address, hostname, internal identifier); there needs to be a reliable way to correlate all of these identifiers to be able to aggregate and fuse together the data from all of these sources.
Incomplete and/or inaccurate data	The completeness and the quality of the data from sensors are not always reliable. For example, during the early stages of rollout of an ISCM system, many departments start by just detecting and reporting hardware inventory without running any scans for vulnerability detection, configuration and compliance assessment, so there is an incomplete picture of the asset. Inexperienced administrators may also incorrectly run scans on devices so the reported findings may be questionable (e.g., results for Microsoft Windows Domain Controller Security Technical Implementation Guidance [STIG] reported for machines that are just regular Windows boxes causing a number of false positive findings on those boxes).
Conflicting findings	There can be overlapping and/or conflicting data from multiple sensors detecting and reporting findings on the same device. For example, in a large enterprise, there are often multiple tools that perform vulnerability scanning and it is not uncommon to find that these tools report different levels of vulnerability exposure and patch compliance on the same device.
Integrating with multiple data access mechanisms and formats	Multiple tools mean multiple mechanisms for data access and multiple data formats. Some tools provide good application programming interfaces (APIs) for data access, others provide access directly to their database and others support only manual exports. Some systems send their data in batches while others send them in an event-driven model. Formats can vary greatly from log files, to comma-separated values (CSV) files, to Extensive Markup Language (XML), to only human-readable reports.
Different interpretations of standards	The NIST's SCAP is increasingly being adopted by the tools to automate assessment procedures as well as to standardize data content and formats. SCAP standards help to alleviate some of these issues, but also present their own challenges. As most developers know, the use of standards does not necessarily guarantee interoperability as a result of different interpretation of standards, support for different versions, and so forth. For example, an issue was discovered with the use of Common Platform Enumeration (CPE), a SCAP standard that is used to standardize how operating systems and application software are represented as strings. Subtle variations in how wildcard characters were used in CPE syntax caused significant differences in vulnerability and patch compliance assessment results.

Source: Tieu Luu. Reprinted with permission.

DATA ARCHITECTURE CHALLENGES

The database/repository subsystem needs a robust architecture that can support multiple interaction models—a lot of writes to ingest data from the sensor subsystem, batch and real-time processing to support the analytics, and *ad hoc* queries from users. Additionally, it needs to be able to accommodate a rich and evolving set of information that is collected about an enterprise's IT assets. For example, the initial phase of the DHS's CDM program is focused on hardware and software asset management, configuration settings, known vulnerabilities and malware. The dataset required to support these use cases includes devices, software applications, patches, configurations, vulnerabilities and operational metadata (e.g., owning/administering organizations, locations, supported systems). Subsequent phases of the program add other use cases, such as auditing, event and incident detection, privilege management, and ports/protocols/services, which greatly expand the dataset that the database/repository subsystem will have to support. Key data architecture challenges presented by these requirements are described in **figure 3**.

This system started with a single database architecture, but evolved into a three-stage data architecture to support the diverse and sometimes conflicting requirements described herein. The purpose of the first stage was to provide a warehouse or collection area to quickly write the data coming in from the sensors, assemble all the messages and reconcile them with existing records in the repository. A great deal of data transformation at the point of data ingestion could create a bottleneck, so the schema for this first stage was designed to closely resemble the data models used by Asset Reporting Format (ARF)[8] and Asset Summary Reporting (ASR).[9] Once the data were ingested, a separate set of jobs would perform the consolidation, correlation and fusion to create the complete, up-to-date profile of the asset. Next the data were extracted, transformed and loaded (ETL) into the second stage, which was a dimensional (e.g., star and snowflake schema) database that was optimized for the analytics and to support the presentation and reporting subsystem. The third stage was a set of Online Analytical Processing (OLAP) cubes that were built from the dimensional database to support the hierarchical dashboards with high-speed roll-up and drill-down analysis of the data.

ANALYTICS CHALLENGES

The main types of analytics required in a continuous monitoring solution include correlation, fusion and deconfliction of sensor findings; compliance assessment; risk scoring; historical trending; and *ad hoc* queries. In addition to helping identify the vulnerabilities that an enterprise is exposed to, along with the scope of exposure and potential

Figure 3—Examples of Key Data Architecture Challenges

Challenge	Examples
Consolidating data from multiple sources	In a large enterprise, the database/repository subsystem may be ingesting data from hundreds of sensors. In this system, the ingest capabilities were implemented to be asynchronous, idempotent and sequencing independent for efficiency and fault tolerance. As a result, the complete set of information for an asset may be distributed across multiple messages, possibly out of order and from multiple sources at different times. The database/repository subsystem needs to consolidate all of this information into a cohesive model that can be applied to analysis and risk scoring.
Conflicting data models	The database/repository subsystem needs a data model that allows the system to quickly write the rich set of information received from the sensors. In this system, the database/repository subsystem received data from the sensors in the ARF and the ASR standards. ARF is used primarily for transmitting information on hardware inventory and operational metadata. ASR is used for transmitting the actual findings discovered about those assets by the sensors. ARF is a very relational model while ASR is more denormalized. Thus, the datasets have conflicting schema design requirements.
Efficiently supporting a diverse set of analytics, dashboards and reports	The schemas for efficient ingest of the ARF and ASR messages do not necessarily make for efficient processing of the analytics nor efficiently supporting the dashboard and reporting requirements from the presentation and reporting subsystem. In addition, different portions of the analytics may require different models. For example, precomputed OLAP cubes are great for the risk-scoring dashboards that present an aggregated enterprise view of risk, as well as to provide the ability to drill down into specific departments along the organizational hierarchy or along other dimensions. However, OLAP cubes are not going to be as effective in supporting *ad hoc* queries and exploration of the data because they require *a priori* definition of the specific intersections of facts and dimensions that are desired so that they can be precomputed. This may not always be known ahead of time for exploratory use cases.

Source: Tieu Luu. Reprinted with permission.

impact, these analytics capabilities also help an enterprise assess how well it has implemented the security controls defined in its policies, e.g., the SANS Top 20 Critical Security Controls.[10] Risk scoring is applied to these assessments to quantify how well the organization is doing and prioritizes the worst problems to fix first. The risk-scoring algorithms can get quite complex when taking into consideration the different types of defects/findings, the severities of the findings, the threats and the impact on the affected assets. Additionally, the organization has to consider whether or not the findings can be remediated, mitigated and accepted, or whether the risk can be transferred to another organization. The analytics and risk scoring have to be applied at multiple levels, from the individual asset or device level, to the network enclave level, to the department level and, finally, up to the enterprise level. This enables the comparative analyses required to identify the worst areas to fix first and enables administrators to drill down into specific assets that have to be remediated. Some of the challenges that may be encountered when implementing these analytics capabilities are described in **figure 4**.

Rigorous engineering discipline combined with agile development methodologies were key to overcoming the challenges associated with the complexity of the analytics' algorithms, as well as to continuously correct and/or evolve the analytics to keep up with changes in the operational environment. Accounting for the quality and consistency issues in the sensor data published from the various sites required a combination of technical and nontechnical solutions. For example, the algorithms were implemented to be robust enough to account for missing data, but then were assigned default values that would penalize the sites for missing data and this was used to drive behavior to ensure that the organization would publish their sensor data correctly in the future. Ensuring that the data could be properly aggregated from multiple sites across the enterprise ultimately required the centralization of the definition of the taxonomies that were used to organize the assets for reporting. So while

Figure 4—Examples of Key Analytics Challenges

Challenge	Examples
Inconsistent data sets across departments	Just as data quality and completeness present a challenge to data integration, they present perhaps an even bigger challenge to implementing the analytics capabilities. Different departments may not consistently provide all of the data necessary to calculate the analytics so that equivalent comparisons can be performed across departments. For example, one of the components in the risk scoring measures was whether or not antivirus signature databases are kept up to date, but there were some departments with sensors that lacked the capability to check that on certain platforms. As a result, the scoring algorithm had to be adjusted to deal with cases of a missing date on the antivirus signature check. This had to be fixed after this particular capability was already deployed into production. In many cases, it is difficult to discover such issues until after the capability has already been deployed.
Aggregation of analytics results across multiple dimensions	The capability to apply and aggregate the analytics at multiple levels can be challenging to implement correctly. There are often multiple hierarchies that the results have to be aggregated against (e.g., active directory structure, organizational structure, IT system/program structure, locations, chain of command). In a very large enterprise with a federated deployment, these challenges can be further exacerbated with different departments and sites, independently organizing their assets using their own taxonomies. With these independent taxonomies, it becomes difficult to reliably aggregate the results together across the enterprise, thereby skewing the results of the analytics.
Accounting for timeliness of sensor findings	Different sensors may report findings for devices at different intervals that can make it challenging when trying to pull together the complete set of findings for a device. For a large enterprise with multiple sites reporting at different times, this can be exacerbated. In addition, for certain findings (e.g., software inventory), some sensors report only a snapshot in time of the current inventory without any differential information (e.g., this software was added or this software was deleted). As a result, there needs to be intelligence in the analytics to know what time window to look across to determine the most recent set of findings for a device and what findings to exclude because they have been superseded.
Evolving requirements and algorithms for analytics	Government, industry and academia are constantly defining new metrics and risk-scoring algorithms to keep pace with the emergence of new cyberthreats. For example, the DOS defined a good baseline model with iPost[11] and the DHS is expanding on that with its CDM scoring model. The risk scoring built for this client was also based on the iPost model, but has been customized for the client and has been updated and enhanced numerous times since it was first implemented. Different sectors also have their own set of metrics and models such as the US Department of Energy's (DOE) Cybersecurity Capability Maturity Model (C2M2)[12] for organizations in the energy industry. As a result, the analytics capabilities in the system have to be able to keep pace with these evolving metrics, models and algorithms.

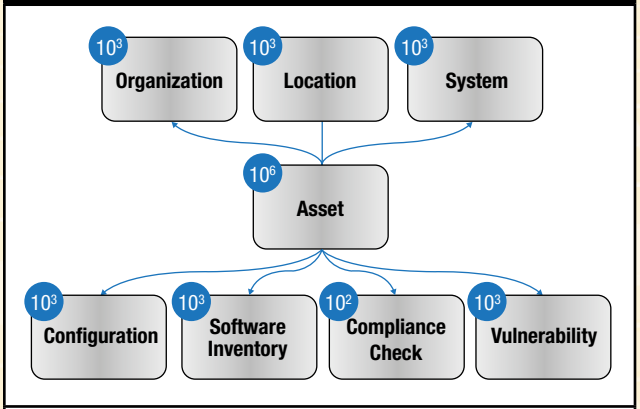
Source: Tieu Luu. Reprinted with permission.

this took away some flexibility for the sites to dynamically define their own taxonomies, the ability to correctly and reliably aggregate the data outweighed this drawback.

PERFORMANCE AND SCALABILITY CHALLENGES

While not on the same scale that large Internet companies face in their applications, in general, a continuous monitoring solution still stores and processes large amounts of data so there are performance and scalability challenges. For example, the client agency described here has somewhere between 5 million and 10 million assets with thousands of software applications and patches, thousands of compliance and configuration settings, and thousands of vulnerabilities to assess against these assets on a daily basis. **Figure 5** depicts these key datasets and the order of magnitude in the number of records that were collected.

Figure 5—Number of Records for Key Datasets Collected



Source: Tieu Luu. Reprinted with permission.

SCAP standards such as ARF, ASR and the Extensible Configuration Checklist Description Format (XCCDF) are rather verbose XML formats and can be very central processing unit (CPU)- and memory-intensive to process. This system has a fixed-time window each night for running the batch jobs that process all of the data collected from the sensors and there have been occasions when the processing duration exceeded the allotted time. These problems are not unique to continuous monitoring and there are many available solutions to address them (e.g., the use of fast-streaming XML parsers to quickly write the ARF, ASR and XCCDF data to the database and have separate jobs to do the consolidation and correlation so that no bottleneck is created at ingestion). Data are stored in multiple formats that are specifically optimized for the analytics they are supporting. Wherever possible, preprocessing is used to speed up response times (e.g., precomputed results in OLAP cubes to drive the dashboards). And then, of course, portions of the architecture have been migrated to Hadoop (e.g., HBase for the data warehouse and Map/Reduce and Pig for some of the analytics) to increase the scalability.

CONCLUSION

An ISCM solution applies many of the technologies from data analytics, business intelligence and MDM applications to the complex domain of cybersecurity. Thus, one may encounter many of the same challenges faced by these types of applications around data integration, data architecture, analytics, and performance and scalability, with additional complexities introduced by the use cases, datasets and standards that are specific to cybersecurity.

Implementing an ISCM solution across a large enterprise is a complex undertaking and there are many other challenges from the deployment, operations and governance perspectives that need to be considered. For example, the deployment approach needs to ensure that sensors are deployed in such a way that provides complete coverage of an enterprise's IT landscape. From an operations perspective, an ISCM solution has a broad set of stakeholders (e.g., chief information officers [CIOs], chief information security officers [CISOs], program managers, system administrators) and they all need to be trained to properly operate and use the capabilities provided. Executives such as CIOs and CISOs need to know how to interpret the results that are displayed in the dashboards, while the system administrators need to know how to properly scan their assets and publish findings. And perhaps most important, governance is needed to make all of this work: First, to require that all of the departments use the tool to inventory and scan their assets in accordance with enterprise security policies and, finally, to enforce the necessary mitigating or remediating actions to address the findings.

ENDNOTES

[1] Government Accountability Office, Report to Congressional Committees, "High-Risk Series: An Update," USA, February 2013, *www.gao.gov/assets/660/652133.pdf*

[2] Performance.gov, "Cross-Agency Priority Goal—Cybersecurity," *www.performance.gov/content/cybersecurity#overview*

[3] Office of Budget Management, "M-14-03. Enhancing the Security of Federal Information and Information Systems," USA, *www.whitehouse.gov/sites/default/files/omb/memoranda/2014/m-14-03.pdf*

[4] National Institute of Standards and Technology, Special Publication 800-137, "Information Security Continuous Monitoring (ISCM) for Federal Information Systems and Organizations," USA, *http://csrc.nist.gov/publications/nistpubs/800-137/SP800-137-Final.pdf*

[5] Department of Homeland Security, "Continuous Asset Evaluation, Situational Awareness, and Risk Scoring (CAESARS) Reference Architecture Report," USA, *www.dhs.gov/xlibrary/assets/fns-caesars.pdf*

[6] *Ibid.*

[7] National Institute of Standards and Technology, "The Security Content Automation Protocol (SCAP)," USA, *http://scap.nist.gov/*

[8] National Institute of Standards and Technology, "ARF—The Asset Reporting Format," USA, *http://scap.nist.gov/specifications/arf/*

[9] National Institute of Standards and Technology, "ASR—The Asset Summary Reporting," USA, *http://scap.nist.gov/specifications/asr/*

[10] SANS Institute, "Top 20 Critical Security Controls," USA, *www.sans.org/critical-security-controls*

[11] Department of State, "iPost," USA, *www.state.gov/documents/organization/156865.pdf*

[12] Department of Energy, "Cybersecurity Capability Maturity Model (C2M2)," USA, *http://energy.gov/oe/services/cybersecurity/cybersecurity-capability-maturity-model-c2m2-program/cybersecurity*