Continuous Security Monitoring

Version 1.5
Released: September 25, 2013

Securosis, L.L.C. 515 E. Carefree Highway Suite #766 Phoenix, AZ 85085 **T** 602-412-3051
info@securosis.com www.securosis.com

Author's Note

The content in this report was developed independently of any sponsors. It is based on material originally posted on the Securosis blog, but has been enhanced, reviewed, and professionally edited.

Special thanks to Chris Pepper for editing and content support.

Thanks to our licensees for this project:

Qualys

Qualys, Inc. (NASDAQ: QLYS), is a pioneer and leading provider of cloud security and compliance solutions with over 6,000 customers in more than 100 countries, including a majority of each of the Forbes Global 100 and Fortune 100. The QualysGuard Cloud Platform and integrated suite of solutions help organizations simplify security operations and lower the cost of compliance by delivering critical security intelligence on demand and automating the full spectrum of auditing, compliance and protection for IT systems and web applications. Founded in 1999, Qualys has established strategic partnerships with leading managed service providers and consulting organizations, including Accuvant, BT, Dell SecureWorks, Fujitsu, NTT, Symantec, Verizon, and Wipro. The company is also a founding member of the Cloud Security Alliance (CSA). For more information, please visit www.qualys.com.

Tenable Network Security

Tenable Network Security is relied upon by more than 17,000 organizations, including the entire U.S. Department of Defense and many of the world's largest companies and governments, to stay ahead of emerging vulnerabilities, threats and compliance-related risks. Its Nessus and SecurityCenter solutions continue to set the standard for identifying vulnerabilities, preventing attacks and complying with a multitude of regulatory requirements. For more information, please visit www.tenable.com.

Tripwire

Tripwire is a leading global provider of risk-based security and compliance management solutions, enabling enterprises, government agencies and service providers to effectively connect security to their business. Tripwire provides the broadest set of foundational security controls including security configuration management, vulnerability management, file integrity monitoring, log and event management. Tripwire solutions deliver unprecedented visibility, business context and security business intelligence allowing extended enterprises to protect sensitive data from breaches, vulnerabilities, and threats. Learn more at www.tripwire.com or follow us @TripwireInc on Twitter.

Copyright

Continuous Security Monitoring Table of Contents

Executive Summary

Given that you can't prevent all attacks, you need to ensure you detect attacks as quickly as possible. The concept of *continuous monitoring* has been gaining momentum, driven by both compliance mandates (notably PCI-DSS) and the US Federal Government's guidance on Continuous Diagnostics and Mitigation, as a means to move beyond periodic assessment. This makes sense given the speed that attacks can proliferate within your environment. In this paper, Securosis will help you assemble a toolkit (including both technology and process) to implement our definition of Continuous Security Monitoring (CSM) to monitor your information assets to meet a variety of needs in your organization.

Defining CSM

Given the different definitions of security monitoring, we advocate a *risk-based* approach to monitoring and assessing critical devices. That means ensuring the most critical assets are truly monitored continuously, and by "continuous" we mean uninterrupted. We've heard all the excuses about why it's not practical to monitor everything continuously, and for a majority of devices in your environment, you probably don't need continuous monitoring. Yet for those devices that are very critical, intermittent assessment leaves a window of exposure for the attackers. A window that you can't afford.

Classifying Assets

Now that you understand there are some devices that you need to monitor continuously, and others where periodic assessment is sufficient, you've got to do the work to determine which devices fall into which category. This involves having a means for ongoing discovery of new assets in your environment, because you can't monitor (or protect) devices you don't know about. You can achieve this discovery via either active scanning of your network address space or passively monitoring network traffic looking for new devices. Or more likely both. Either way, awareness of your network topology is a critical success factor for CSM.

Next you need a consistent and objective way to classify those assets based on criticality. There are many ways to classify assets, and we tend to favor one based on *business criticality*. Basically devices that have access to information that could result in significant losses to the organization are necessarily more critical and warrant more frequent monitoring/assessment. Another key aspect of classification is gaining internal consensus, especially when most senior managers have the opinion that systems supporting their business are the most important systems. Not everything can be an absolutely critical device, so tough choices need to be made, and everyone must agree with those choices.

Use Cases

As we dig into the specific use cases driving CSM, we see a bulk of the projects aiming to meet either generating compliance documentation, tracking changes on the monitored devices, and/or detecting attacks. To understand each use case a little better, here is a short description:

- **Attacks:** This is using security monitoring to identify potential attacks and/or compromise of systems. This is the general concept we have described in our monitoring-centric research for years. It also involves both an outside-in (attacker's view) and an inside-out (insider's view) of the IT environment to ensure all attack surface is sufficiently monitored.

- **Change control:** An operations-centric use case is to monitor for changes, both to detect unplanned (possibly malicious or dangerous) changes, and to verify that planned changes complete successfully.

- **Compliance:** Finally, there is the checkbox use case, where a mandate or guidance requires monitoring and/or scanning technology; less sophisticated organizations have no choice but to do *something*. But keep in mind that the mandated product of this initiative is documentation that you are doing something — not necessarily an improved security posture, identification of security issues, or confirmation of activity.

The attack use case is bigger, broader, and more difficult than change management; compliance is the least sophisticated. Obviously you can define more granular use cases, but these three cover most of what people expect from security monitoring.

This is a reversal of the order in which most organizations adopt security technologies. Many start with a demand to achieve compliance, then move to an internal control process to deal with changes — typically internal — and finally are ready to address potential attacks by analyzing aggregated data. Of course there are many paths to security and many organizations jump right to the attack use case, especially those under immediate or perpetual attack.

Selecting the CSM Platform

To implement CSM you'll need to decide on the technology platform to aggregate your data sources and perform the CSM analysis. You have a bunch of candidates, and probably a few already operational in your environment — though likely underutilized. These include your SIEM and also your Vulnerability Management platform. Not to spoil the ending, but shockingly enough, the platform you choose will depend on your use case.

Be wary of any platform without a scalable data model that can evolve to handle additional data sources over time. Again, depending on your use case, you may not need those capabilities immediately, but don't let a short-sighted technology choice sacrifice your ability to grow into the attack use case someday.

Evolving to CSM

Depending on which platform you choose on which to build your CSM capability, you may be simply adding capabilities to an existing in-house product, or you could be facing a rip and replace of existing technology. Either way, you'll need to go through a structured planning effort involving identifying the new data sources to provide the raw materials for the analysis needed for the use case. Then you'll need to document the visualizations, alerts and reports need to achieve the desired results. Finally you'll then need to apply solid project management discipline to make sure the evolution happens on time and within budget.

Once you get to the implementation phase, then you make your plan into reality by importing the new data and installing the policies and dashboards. Testing and verification of the accuracy of the new capabilities comes next, and then you are ready to take the new use cases into production.

At this point your new use case is operational and you are benefitting from continuous security monitoring. But attaining CSM is only the first part of your journey. New technology deployments and capabilities such as cloud computing, as well as emerging attacks, will require you to continuously evolve your security monitoring environment to keep pace.

Why. Continuous. Security. Monitoring?

Remember the old marketing tagline, "Get Ahead of the Threat?" It seems pretty funny now, doesn't it? Given the kinds of attacks you face and attackers' increasing sophistication — you never see the threats coming. Thus your only option is to *React Faster and Better*. The bad news is that it won't get easier any time soon because the attackers are getting better and the defenses are not improving at nearly the pace. Don't shoot the messenger, but understand this is the reality of today's information security landscape.

The behavior of most organizations over the past decade hasn't helped, either. Most companies spend the bulk of security budgets on preventative controls proven ineffective over and over again. Part of this is due to compliance mandates for ancient technologies, but only very forward-thinking organizations invest sufficiently in the detection and response aspects of their security programs. Unfortunately organizations become enlightened only *after* cleaning up major data breaches. For the unenlightened, detection and response remain horribly under-resourced and underfunded.

Unfortunately organizations become enlightened only *after* cleaning up major data breaches. For the unenlightened, detection and response remain horribly under-resourced and underfunded.

At the same time the US government has been pushing a "continuous monitoring" (CM) agenda on both military and civilian agencies to provide "situational awareness," which is really just a fancy term for understanding what the hell is happening in your environment at any given time.

In fact, the Department of Homeland Security (DHS) is now referring to this capability as "Continuous Diagnostics and Mitigation," which sets an objective that "officials at each agency will be able to quickly identify which problems to fix first, and empower technical managers to prioritize and mitigate risks." Uh, OK.

Ultimately regardless of what you call it (CM or CDM), the problem is that this monitoring requirement applies to a variety of operational disciplines in the public sector, and it doesn't necessarily mean

'continuous'. CM (or CDM) is a good first step, but as with most first steps, too many organizations take it as the destination rather than the start of a long journey.

We have always strongly advocated security monitoring, and have published a ton of research on these topics, from our philosophical foundation: Monitor Everything, to our SIEM research: (Understanding and Selecting, and SIEM Replacement). And don't forget our process modeling of Network Security Operations, which is all about security monitoring. We don't need to be sold on the importance of security monitoring, but evidently the industry still needs convincing, given the continued failure of even large organizations to realize they must combine a strong set of controls with equally strong capabilities for detection, monitoring, and incident response.

To complicate matters technology continues to evolve with the advent of BYOD, increasing mobility, cloud computing, virtualization and a host of other innovations focused on making technology more accessible, scalable and effective. This also means the tools and processes for security monitoring today are different than even 18 months ago, and the tools will look different 18 months from now. This Continuous Security Monitoring (CSM) paper will evaluate these advancements, flesh out our definition of CSM, break down the use cases for CSM, and consider the technology platforms that can provide this cornerstone of your security program.

React Faster and Better

We have gotten a lot of mileage out of our React Faster and Better concept, which really just means you need to accept and plan for the fact that you cannot stop all attacks. Even more to the point (and potentially impacting your wallet), success is heavily determined by how quickly you detect attacks and how effectively you respond to them. We suggest you read that paper for a detailed perspective on what is involved in incident response — along with ideas on the organization, processes, and tools required to do it well.

This paper will not rehash that territory — instead it will help you assemble a toolkit (including both technology and process) to monitor your information assets to meet a variety of needs in your organization. Clearly you want to find when you're being attacked. If you don't understand the importance of this aspect of security, just consider that a majority of breaches (at least according to the latest Verizon Business Data Breach Incident Report) continue to be identified by third parties, such as payment processors and law enforcement. That means organizations typically have no idea when they are compromised, and that is a big problem.

But you also can use CSM for both change control and compliance purposes, so we'll cover those use cases as well.

Why CSM?

We can groan all day and night about how behind-the-times the PCI-DSS remains, or how the US government has defined Continuous Monitoring. **But attackers innovate and move much more quickly than regulation, and that is not going to change.** So you need to understand these mandates for what they are: a low bar to get you moving toward a broader goal of continuous security monitoring.

> Without PCI and the US government mandating security data aggregation and analysis, we would still be spending most of our time evangelizing the need for even simplistic monitoring in the first place.

But before we take the typical cynical approach and gripe about what's wrong, let's recognize the yeoman's work already done to highlight the importance of monitoring to protecting information (data). Without PCI and the US government mandating security data aggregation and analysis, we would still be spending most of our time evangelizing the need for even simplistic monitoring in the first place. The fact that we don't is a testament to the industry's ability to parlay a mandate into something productive.

That said, if you are looking to solve security problems and identify advanced attackers, you need to go well beyond mandates. This paper will define what we call "Continuous Security Monitoring" and dig into the different sources of data you need to figure out how big your problem is. See what we did there? You have a problem, and we won't argue that — your success hinges on determining what has been compromised and for how long.

Defining CSM

Since you can never "get ahead of the threat" you need to react faster to what's happening, which requires shortening the window of exposure with extensive security monitoring. We tipped our hats to both the PCI Council and the US government for requiring monitoring as a key aspect of their mandates. The US government pushed it a step further by including 'continuous' in its definition. We love the term 'continuous', but it has caused a lot of confusion among folks responsible for monitoring.

As we are prone to do, it is time to wade through the hyperbole to define what we mean by Continuous Security Monitoring, and then identify some of the challenges you will face in moving towards this ideal.

Defining CSM

We need not spend any time defining security monitoring — we have been writing about it for years. But we need to consider how *continuous* any monitoring really needs to be, given the current state-of-the-art in attack tactics. Many solutions claim to offer "continuous monitoring", but all too many simply scan or otherwise assess devices every couple days — if that often. Which would seem to be acceptable, given NIST's official definition of Continuous Security Monitoring:

> *Information security continuous* monitoring (ISCM) is maintaining ongoing* awareness of information security, vulnerabilities, and threats to support organizational risk management decisions.*
>
> **The terms "continuous" and "ongoing" in this context mean that security controls and organizational risks are assessed, analyzed and reported at a frequency sufficient to support risk-based security decisions as needed to adequately protect organization information. Data collection, no matter how frequent, is performed at discrete intervals.*
> *NIST 800-137 (PDF)*

Wait, what? So to NIST 'continuous' doesn't actually mean continuous, but instead *"a frequency … needed to adequately protect organization information."*

Sorry, but no. We have heard all the excuses for why it is not practical to monitor everything continuously, including concerns about device resources consumption, excessive bandwidth usage, and inability to deal with an avalanche of alerts. All those issues ring hollow because intermittent assessment leaves a window of exposure for attackers, and for critical devices you don't have that luxury. Our definition of continuous is more in line with the dictionary definition:

> ***con.tin.u.ous:*** *adjective \kən-ˈtin-yue-əs\ — marked by uninterrupted extension in space, time, or sequence.*

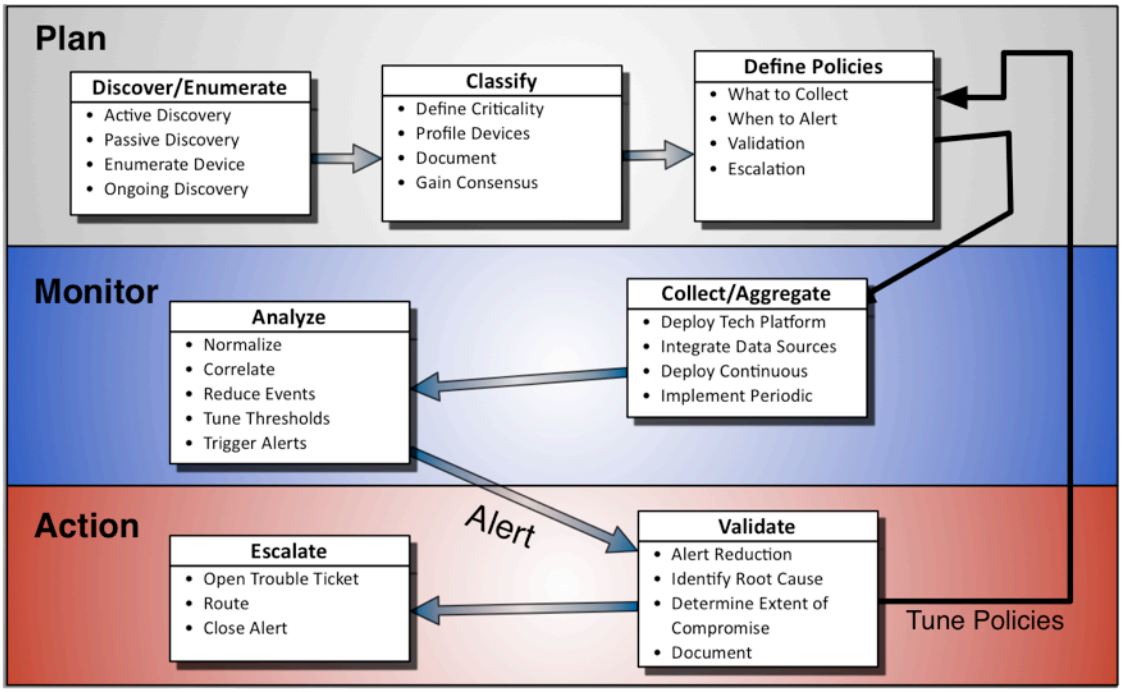
The key word there is *uninterrupted*: always active. The constructionist definition of continuous security monitoring should be that the devices in question are monitored at all times — there is no window where attackers (or internal operations people) can make a change adversely impacting security posture without it being immediately detected. But we are neither constructionist nor religious — we take a realistic and pragmatic approach, which means accepting that not every organization can or should monitor all devices at all times.

We incorporate asset criticality into our concept of CSM. Some devices have access to very important stuff. Those devices need to be monitored *continuously*.

We incorporate asset criticality into our concept of CSM. Some devices have access to very important stuff. You know, the stuff that if leaked will result in blood (likely yours and your team's) flowing through the halls. The stuff that just *cannot* be compromised. Those devices need to be monitored continuously. And then there is everything else. In that "everything else" bucket land all the other devices you need to monitor and assess, but not as urgently or frequently. For the sake of efficiency, you will monitor these devices periodically, so long as you have other methods to detect and identify compromised devices, such as network analytics/anomaly detection and aggressive egress filtering.

The secret to success at CSM is in choosing your high-criticality assets well, so we will get into that later. Another critical success factor is discovering when new devices appear, classifying them quickly, and getting them into the monitoring system quickly. This requires strong process and technology to ensure you have visibility into all your networks, can aggregate the data you need, and have sufficient computational analysis horsepower.

Adapting the Network Security Operations process map we published a few years back, here is our Continuous Security Monitoring Process.



The process is broken down into three phases. In the Plan phase you define policies, classify assets, and continuously discover new assets in your environment. In the Monitor phase you pull data from devices and other sources, to aggregate and eventually analyze, in order to alert if a potential attack or other situation of concern becomes apparent. You will monitor not only to detect attacks, but also to confirm changes and identify unauthorized changes, and to substantiate compliance with organizational and regulatory standards (mandates). It's critical here to make sure you manage the signal to noise ratio by effectively determining what will be monitored and when alerts will fire, as the system can stream alerts nonstop without careful tuning.

In the final phase you take action (determine what action to take, if any) by validating the alert and escalating as needed. As with all our process models, not all these activities will work or fit in your environment. We publish these maps to provide ideas about what you'll need to do — they always require customization to your needs.

The Challenge of Full Visibility

As we mentioned above, the key challenge in CSM is classifying assets, but your ability to do so is directly related to the visibility of your environment. *You cannot monitor or protect devices you don't know about.* So the key enabler for this entire CSM concept is an understanding of your network topology and the devices that connect to your networks. By continuously analyzing your attack surface, the goal is to avoid an "oh crap" moment, when a bunch of unknown devices and/or

applications show up — and you have no idea what they are, what they have access to, or whether they are steaming piles of malware.

There are a number of discovery techniques, including actively scanning your entire address space for devices and profiling what you find. That works well enough and is how most vulnerability management offerings handle discovery, so active discovery is one requirement. But scanning a full address space scan can have a substantial network impact, and isn't appropriate during peak traffic times. Be sure to search both your IPv4 and IPv6 address spaces. You don't have IPv6, you say? You will need to confirm that — many devices have IPv6 turned on by default, broadcasting those addresses to potential attackers.

> The key enabler for the CSM concept is an understanding of your network topology. The goal is to avoid an "oh crap" moment, when a bunch of unknown devices and/or applications appear on your network.

You should supplement active discovery with a passive discovery capability that monitors network traffic and identifies new devices, traffic to malicious sites, and unauthorized communications from their network communications. Sophisticated passive analysis can also profile devices and identify vulnerabilities, but passive monitoring's primary goal is to find new *unmanaged* devices faster, which then triggers a full active scan on identification. Passive discovery is also helpful for identifying devices hidden behind firewalls and on protected segments, which block active discovery and vulnerability scanning.

It is also important to visualize your network topology — a drill-down map is worth a million words. Being able to isolate a device, understand where it fits in your topology, and drill down into previous assessments, dramatically accelerates the process of discovering the root cause of issues during the validation and escalation phases.

Complicating factors for discovery include cloud computing and mobility. With the lack of control and visibility over devices outside the cozy confines of your network perimeter, figuring out which devices have access to critical data stores is increasingly difficult. Cloud computing provides the ability to spin up and take down instances at will without human involvement — perhaps outside your data center. This clearly impacts visibility, so your discovery processes need to be integrated with your cloud consoles to ensure you know about and can assess newly-minted instances, applications and services.

Similarly, intelligent mobile devices with access to critical enterprise data create easy targets for attackers probing your network. So mobile devices need to be assessed on connection using network security controls to ensure they have an adequate security posture and access only to authorized data.

Classifying Assets

Identifying your critical assets and monitoring them continuously is a key success factor for your security program — at least if you are interested in figuring out what has been compromised. But reality says you can't watch everything all the time, even with these new security big data analytical thingies.

The success of your security program hinges on your ability to *prioritize* what to do. That was the focus of our Vulnerability Management Evolution research last year. Prioritizing requires you to determine how different asset classes will be monitored. You need a consistent process to classify assets. To define this process let's borrow liberally from Mike Rothman's Pragmatic CSO methodology — identifying what's important is the critical first step.

> *So a critical step is to make sure you've got a clear idea about priorities and to get the senior management buy-in on what those priorities are. You don't want to spend a lot of money protecting a system that has low perceived value to the business. That would be silly.*
> *— The Pragmatic CSO, p 25.*

One of the hallmarks of a mature security program is having this elusive buy-in from all levels and areas of the organization. And that doesn't happen by itself.

Business System Focus

When you talk to folks about their data leak prevention (DLP) efforts, a big impediment to sustainable success is the ongoing complexity of classification. It's just overwhelming to try putting all your organization's data into buckets and then maintaining those buckets over time. The same issues apply to classifying computing assets for monitoring purposes.

Does this server fit into that bucket? What about that network security device? And that smartphone? Multiply that by a couple hundred thousand servers, endpoints, and users, and you start to understand the challenges of classification. One way to avoid being overwhelmed is to think about your computing devices in terms of business systems.

To understand what that means, let's return to The Pragmatic CSO:

> *The key to any security program is to make sure that the most critical business systems are protected. You are not concerned about specific desktops, servers or switches. You need only be focused on fully functioning business systems. Obviously every fully functioning system consists of many servers, switches, databases, storage, and applications. All of these components need to be protected to ensure the safety of the system. — The Pragmatic CSO, p 23*

This requires aligning specific devices with the business systems they serve. Each device inherits the criticality of its most critical business system. Simple, right? Components such as SANs and perimeter security gateways are used by multiple business systems, so they need to be classified with the most critical business system they serve.

By the way, you are doing this already if you have any regulatory oversight. You know those in-scope assets for your PCI assessment? Those PCI-relevant systems have access to protected data, which is a form of classification. Those require protection in accordance with the PCI-DSS guidance. Those efforts have been based on what you need to do to understand your PCI (or other mandate) scope, and we are extending that mentality across your entire environment.

Limited Buckets

To understand the difficulty of managing all these combinations, consider the inability of many organizations to implement Role-Based Access Control on their key enterprise applications. That is largely because something like a general ledger application has hundreds of roles (R), with each role involving multiple access rules (A). Each employee (E) may have multiple roles, so RBAC requires managing *A * R * E* entitlements. Good luck with that.

We suggest limiting the number of buckets used to classify business systems. Maybe it's 2. You know, the stuff that gets you fired if breached, and the stuff that doesn't. Maybe it's 3 or 5 — no more. We are talking about monitoring devices in this paper, but you also need to implement and manage different security controls for each level. It's the concept we called Vaulting a couple years ago, also commonly known as "security enclaves".

After identifying and classifying your business systems into a manageable number of buckets you can start to think about how to monitor each class of devices according to its criticality. Be sure to build in triggers and catalysts to revisit your classifications. For example you should revisit the classifications if a business system is opened to trading partners or you authorize a new device to access critical data. *As long as you understand these classifications are based upon your needs at particular points in time, and need to be updated periodically, this process works well.*

Internet-accessibility

As much as we try to focus on business systems in our classification efforts, you should still factor in the additional risk presented by Internet-facing devices. These devices are sitting ducks, open to reconnaissance and attacks by anyone with a scanner or Metasploit. If there is a hole in these devices, it will be found and exploited quickly. Therefore these devices carry more risk and likely warrant more frequent assessments.

Employees Count Too

We have been discussing business systems and associated computing devices used to support them, but we cannot forget the weakest link in pretty much every organization: employees. You need to classify employees just like business systems. Do they have access to stuff that would be bad if breached, and how do they access it — mobile vs. desktop, remote vs. on-network, etc.?

You should place very limited trust in endpoint devices and the employees that use them. We see new stories of this 0-day or that breach daily, compounded by an idiotic action taken by some employee. This should force you to apply more discipline and tighter controls to all of your computing devices. To further illuminate this concept, there is definitely a different risk profile for a low-level employee operating on a device sitting on the corporate network, than your CFO accessing unannounced financials on an Android tablet from a cafe in China. Part of your CSM process must be classifying, protecting, and monitoring employee devices. As legally appropriate, of course.

Gaining Consensus

Now that you have bought into this classification discipline you need to make it reality. This is where the fun begins — it requires buy-in within the organization, which is never easy. First you need to enumerate the influencers who can derail this process before it even begins. You need to get them lined up before you pitch the idea formally to anyone. There are really two efforts here.

Now that you have bought into this classification discipline you need to make it reality. This is where the fun begins — it requires buy-in within the organization, which is never easy.

1. **The Informal:** Akin to back-room politics, you need to get to the influencers and get them on board with the idea. These are informal discussions about getting support.
2. **The Formal:** Here you present to some kind of task force or senior management group for approval for an enterprise-wide initiative, with the funding, resources, etc. required to make it happen.

It is unwise to make a formal pitch before making sure you won't get your hat handed to you (see, we're trying to be polite) when you ask for approval. Unless you have recently had a breach — then nobody says much of anything about whatever you want to do. In terms of who to approach informally, it's the folks who run ops (since this will impact them), the IR team (who will need to investigate things the monitors find), and likely the network folks because you will consume some network resources and need to tap some areas of the network.

Once those folks are on board with the concept (and we know that is a non-trivial requirement), you can present the different classifications and monitoring scenarios for each level of criticality and get approval to move forward. Of course you can monitor plenty of stuff without organizational buy-in. *But not as a core function of your security program.* You would be stuck doing one-off monitoring of certain assets or systems rather than taking an enterprise-wide approach to monitoring, to detect issues and shorten the window of organizational exposure.

One of the key aspects of this horse trading is defining exceptions. You have special folks in your environment who play by different rules.

Once you get the green light you need to start horse trading. This is one of those aspects of getting things done in the real world they don't teach in the CISSP course. Once you are through this process you will consider passing healthcare reform anywhere in the world a cakewalk. All kidding aside, success is constrained by the types of compromises you are willing to make. You need to keep the big picture in mind: without a way to classify your assets, you cannot really succeed with any monitoring initiative. Stay focused on the long term and get the support of the folks you need.

One of the key aspects of this horse trading is defining exceptions. You have special folks in your environment who play by different rules. You may not know who they are, but they exist, and you need special policies for them. Yes, it's irritating that they don't have to play by the same rules as everybody else, but that's life. Again, success isn't a matter of getting rid of all exceptions — it is about minimizing them and making sure that even the special folks adhere to a certain level of security and monitoring control.

If you have too many exceptions your monitoring program becomes too complex and unwieldy. Too few exceptions and you will never get the consensus you need to succeed. So go into this process understanding that a lot of work is required before you ever put your hands on a keyboard, start aggregating data, or otherwise doing anything technical.

The Use Cases

At the highest level we generally see three discrete use cases for CSM:

- **Attacks:** This is using security monitoring to identify potential attacks and/or compromise of systems. This is the general concept we have described in our monitoring-centric research for years.
- **Change control:** An operations-centric use case is to monitor for changes, both to detect unplanned (possibly malicious or dangerous) changes, and to verify that planned changes complete successfully.
- **Compliance:** Finally, there is the checkbox use case, where a mandate or guidance requires monitoring and/or scanning technology; less sophisticated organizations have no choice but to do *something*. But keep in mind that the mandated product of this initiative is documentation that you are doing something — not necessarily an improved security posture, identification of security issues, or confirmation of activity.

Notice these use cases are listed from broadest and most challenging, to narrowest and most limited. The attack use case is bigger, broader, and more difficult than change management; compliance is the least sophisticated. Obviously you can define more granular use cases, but these three cover most of what people expect from security monitoring.

This is a reversal of the order in which most organizations adopt security technologies. Many start with a demand to achieve compliance, then move to an internal control process to deal with changes — typically internal — and finally are ready to address potential attacks by analyzing aggregated data. Of course there are many paths to security and many organizations jump right to the attack use case, especially those under immediate or perpetual attack.

We made a specific decision to address the broadest use case first — largely because even if you are not yet looking for attacks, you will need to, soon enough. So we will start by laying out the entire process to monitor for attacks, and then show how you can streamline your implementation for other use cases.

The Attack Use Case

The first use case we'll address tends to be the highest profile, given it's focused on stopping attacks. We mentioned NIST's "official" definition of Continuous Monitoring above ("*assessed, analyzed and reported at a frequency sufficient to support risk-based security decisions"*). As such, **Your monitoring strategy should as 'continuous' as it needs to be.** Just like advanced attackers are only as advanced as they need to be. We appreciate this clarification, which reflects the fact that some assets need to be monitored at all times, others not so much.

That's why we delved into the importance of classifying your assets and applying a risk-based filter to your monitoring efforts. As we dig into this specific use case, it makes sense to be a bit more specific about what you are trying to identify in this use case:

- Determining your attack surface by understanding your vulnerabilities and exploitable devices
- Prioritize remediating those devices based on which have the most risk of compromise
- Identify malware in your environment
- Detect intrusion attempts at all levels of your environment
- Gain awareness and track adversaries in your midst
- Detect exfiltration of sensitive data
- Identify the extent of any active compromise and provide useful information for clean-up
- Verify clean-up and elimination of the threat

Data Sources

To address this laundry list of goals you need the following data sources:

- **Assets:** As we discussed under classification, you cannot monitor what you don't know about; likewise you need to know how critical an asset is to choose the most appropriate way to monitor it. As we described in our Vulnerability Management Evolution research, this requires an ongoing (dare we say, 'continuous') discovery capability to detect new devices appearing on your network, and then a consistent mechanism for profiling and classifying them.

- **Network topology & telemetry:** Next you need to understand the network layout, specifically where critical assets reside. Assets accessible to attackers are higher priority than inaccessible assets, so it is quite possible to have a device which is technically vulnerable and contains critical data, but is prioritized lower than a less-valuable asset in harm's way.

- **Events & logs:** Any technological device generates log and event data. This includes security gear, network infrastructure, identity sources, data center servers, and applications, among others. Patterns in the logs may indicate attacks if you know how to look; logs also offer substantiation and forensic evidence after an attack.

- **Configurations:** Configuration details and unauthorized configuration changes may also indicate attacks. Malware generally needs to change device configuration to cause its desired behavior.

- **Vulnerabilities:** Known vulnerabilities provide another perspective on device vulnerability, and can be attacked by exploits in the wild.

- **Device forensics:** An advanced data source providing a very detailed image (including memory, disk, etc.) of the device at any point in time. Typically associated with an incident investigation, this capability is evolving into a broader endpoint activity monitoring function, granularly tracking the activity on the device over a long period of time. This enables much more detailed analysis of the activity leading to a compromise. Investigators can then establish a timeline of the attack to isolate indicators of compromise and then understand how the device was used by the attackers.

- **Network forensics:** Capturing full packet streams enables replay of traffic into and out of devices. This is very useful for identifying attack patterns, and also for forensics after an attack.

That is a broad list of data, but — depending on the sophistication of your CSM process — you may not need all these sources. More data is better than less, but everyone needs to strike a balance between capturing everything and only aggregating what's immediately useful. You do not get a second chance to capture data but resource constraints have a strong influence on the scope of collection efforts.

Getting the Data

So how can you collect all this data on an ongoing basis, and with what frequency? It's time to get back to those asset classifications you decided on earlier. For critical assets gather as much data as possible. That likely means some kind of agentry on the devices to gather data at all times and send it back to the CSM aggregation point for pseudo-real-time analysis. We say 'pseudo-real-time' because due to the nature of monitoring and the laws of physics, there is always lag between when something happens on the device and when it can be analyzed by the CSM system.

We say 'pseudo-real-time' because due to the nature of monitoring and the laws of physics, there is always lag between when something happens on the device and when it can be analyzed by the CSM system.

As mentioned above, Internet-accessible devices bring additional risk given they are subject to any and every recon attempt from anyone with an Internet connection. Thus, these devices should be monitored and/or assessed more frequently and likely via an external scan (from outside your perimeter). You want to be able to assess these devices from the perspective of an attacker, and that attacker is very likely coming from outside your perimeter. So when gathering CSM data, both the inside-out and outside-in perspectives are important.

For devices which do not quite require always-on monitoring or forensics, you need to determine the frequency of vulnerability scanning, file integrity monitoring, and/or configuration change monitoring. Depending on criticality you might want to scan daily or weekly. You also need to determine whether you need a credentialed scan for a far more granular assessment, or if an uncredentialed scan will suffice. Of course today's malware spreads almost instantaneously, so if you don't catch a configuration change or another indicator of attack for a week, who knows how much damage will happen before you notice? This is why classification is so important — an attacker may start (and gain presence) by compromising an infrequently scanned device, but at some point they will need to go after a critical device you should be monitoring continuously — and therefore you should be in a position to detect the attack.

Another important aspect of data collection is automation. The only way to scale any kind of monitoring initiative is to have the data sent to the aggregation platform without human intervention, for both efficiency and accuracy. One aspect of the 'continuous' monitoring approach espoused by the US government is moving away from accreditation every couple years, instead monitoring devices more frequently. It seems obvious, and it is. Today security success requires shortening the window between compromise and detection. To be clear, there is a place for third-party and/or manual assessment to confirm controls, but *operational* automation of data collection is essential.

We should also point out the blurry line between monitoring and defense, particularly for critical devices which are monitored continuously. Monitoring is a passive alerting function compared to prevention — actively blocking attacks. The nuances of what to block and what to only monitor, as well as how to avoid false positives and negatives, are both beyond the scope of this paper. But all things being equal, if you can identify a clear attack on a critical device, you should position yourself to prevent it.

Quantifying CSM Risk

A key aspect of prioritizing which devices require remediation is understanding the risk a compromised device poses to your organization, based on how you classified the asset (as described above). We have never been fans of risk scoring because it can be far too subjective, and the algorithms tend to capture risk of compromise rather than true organizational or economic risk.

Once again, NIST offers useful perspective:

> *True risk scoring can be difficult to achieve using the NIST SP 800-37 revision 1 definition, and many "risk scoring" methodologies do not demonstrate a correlation to true risk measurement. Instead, they usually measure the state of a collection of security capabilities or controls.*
> *NIST IR 7756, p 26 (PDF)*

Of course many folks (including some of our friends) have spent a significant portion of their careers trying to quantify risk, and we applaud those efforts. We are sure they will love our risk quantification shortcuts. *But for our definition of CSM it is not clear we need to truly quantify risk — mostly we merely need to assess* ***relative*** *risk to support decision-making.* Issues with critical assets need higher priority. You can prioritize further based on whether the device can be accessed via external or only internal attackers. Inaccessible device are lower priority. Then you need to define a coarse hierarchy of potential exposures. It might look like this:

1. Devices exhibiting anomalous behavior and/or indicators of compromise
2. Vulnerable devices with exploit code in the wild
3. Configuration issues which could result in full compromise
4. Devices vulnerable to non-weaponized attack
5. Everything else

Obviously compromised devices need to be addressed ASAP. Next on the list would be a device vulnerable to an exploit which has been seen in the wild. You may not have seen an active attack yet, but you will. Attackers are reliable that way, once weaponized code is available. Next you address configuration errors which could result in a compromised device. Finally you deal with standard vulnerabilities as part of the normal patch/maintenance cycle. This list is not intended to be comprehensive — simply to illustrate you don't need a very complicated algorithm to determine security risk in your environment and drive remediation decisions.

Clearly it's difficult to make decisions based on disparate information sources using inconsistent metrics and reporting environments. The compelling value of CSM is in integrating all these disparate data sources into a common platform for more effective decision support.

We aren't trying to minimize the work required to aggregate all this data and define the rules to determine what is an attack, *versus* an exploitable vulnerability, *versus* a problematic configuration issue. It is decidedly non-trivial, but keep your eye on the prize. Today these data sources are aggregated and analyzed within separate management systems. Clearly it's difficult to make decisions based on disparate information sources using inconsistent metrics and reporting environments. The compelling value of CSM is in integrating all these disparate data sources into a common platform for more effective decision support.

The Change Control Use Case

Beyond detecting an attack, another key capability of using CSM for Change Control is to isolate any unplanned non-malicious changes to figure out why the change occurred outside normal change processes. You can also verify planned and authorized changes to close the operational process loop.

Before we discuss the data sources you need, we should mention monitoring frequency. As with the attack use case, the NIST definition — *monitor as frequently as you need to* — fits here as well. For highly critical devices you want to look for changes continuously, because if the device is attacked or suffers a bad change the result could be data loss. As we mentioned under the attack use case, automation is critical to maintaining a consistent and accurate monitoring process. Ensure you minimize human effort, increase efficiency, and minimize human error.

> If it's not an attack, you need to isolate any unplanned non-malicious changes to figure out why the change occurred outside normal change processes.

Data Sources

To evaluate a specific change you will want to collect the following data sources:

- **Assets:** As we discussed above, you cannot monitor what you don't know about; without knowing how critical an asset is, you cannot choose the most appropriate way to monitor it. This requires an ongoing — dare we say, 'continuous'— discovery capability to detect new devices appearing on your network, as well as a mechanism for profiling and classifying them.
- **Work orders:** A key aspect of change control is handling unauthorized and authorized changes. To do that you need an idea of which changes are part of a patch, update, or maintenance request. That requires a link to your work management system to learn whether a device was scheduled for work.

- **Patching process:** Sometimes installing security patches is outside the purview of the operations group, rather something the security function takes care of. Not that we think that's the right way to run things, but not all operational processes are managed in the same system. If different systems are used to manage the work involved in changes and patches you need visibility into both.

- **Configurations:** This use case is all about determining differentials in configurations and software loaded on devices, and using that information to figure out whether you're under attack or it's an operational process problem. This requires the ability to assess the configuration of devices, and to store a change history so you can review deltas to pinpoint exactly what any specific change did and when.

- **File integrity:** Another indication of a change is when a system or other sensitive file changes. You should be able to pinpoint when a file is changed, by whom, and whether it's authorized.

We have always been fans of more data rather than less, so if you can collect device forensics, more detailed event logs, and/or network full packet captures — do that.

Decision Flow

Unlike the attack use case, which shows more variation in how you evaluate alerts generated by the monitoring process, the decision flow for change control is straightforward:

1. **Detect change:** Through your security monitoring initiative you will be notified that a change happened on a device you are watching.

2. **Is this change authorized?** Next you will want to cross-reference any changes against the work management system(s) managing all the operational changes in your environment. It is important you link your operational tracking systems with the CSM environment — otherwise you will spend a lot of time investigating *authorized* changes. We understand these systems tend to be run by different operational groups, but to have a fully functional process those walls need to be broken down.

3. **If authorized, was the change completed successfully?** If the change was completed then move on. Nothing else to see here. The hope is this verification can be done in an automated fashion to ensure you aren't spending time validating stuff that already completed successfully, so your valuable (and expensive) humans can spend their time dealing with exceptions. If the change failed for some reason, you need to send that information back into the work management system (perhaps some fancy DevOps thing, or your trouble ticket system) to have the work done again.

4. **If not authorized, is it an attack?** At this point you need to do a quick triage to figure out whether this is an attack warranting further investigation or escalation, or merely an operational failure. Context is important for determining whether it's an ongoing attack.

5. **If it's an attack, investigate:** If you determine it's an attack you need to investigate. We dealt with this process in both Incident Response Fundamentals and React Faster and Better.

6. **If it's not an attack, figure out who screwed up:** If you made it to this point the good news is that your unauthorized change is an operational mishap rather than an attack. So you need to figure out why the mistake happened and take corrective measures within the change process to ensure it doesn't happen again.

One further clarification on the distinction between the attack and change control use cases. If you have only implemented the change use case and collected the data appropriate for it, then your visibility into what malware is doing and how broadly it has spread up to this point will be limited. But that doesn't mean starting with change control provides no value for detecting attacks. An alert of an unauthorized change can give you a heads-up for an imminent issue — you just may not have the data to fully investigate it.

Taking Action

The entire point of any monitoring initiative is to make better decisions on what needs to be done and how to allocate resources. First let's take in-process attacks off the table — they were covered in the attack use case, and obviously take priority over pretty much everything else. So how do you determine whether it's an attack? Look at this in terms of attack surface. Does the change make the device easier to attack or control? If so it is *effectively* an attack. Some operational failures result in increased attack surface and so should be handled as attacks, even if the actor wasn't malicious.

> An innocent operational failure that increases attack surface isn't any less of a problem than a malicious action.

This focus on attack surface takes intent out to enable simpler and more objective analysis. An innocent operational failure that increases attack surface isn't any less of a problem than a malicious action. The device is more exposed than it was before the change and needs to be remediated. That's why we favor the attack use case as the basis for security monitoring, with a simplification to deal with change control and compliance.

In case of an operational mishap you have a further decision to make: when to roll it back. That depends on the nature of the change, the criticality of the device, and whether the rollback can be automated. For changes that don't increase attack surface there is less urgency to roll back, unless the change broke an application or otherwise impacted availability. So operational mishaps can be put back into the stack of work and processed according to the other operational processes managing workflow in the organization.

The Compliance Use Case

Compliance is typically the first use case implemented, mostly because PCI-DSS (and some other regulations) mandated the aggregation and parsing of event logs many years ago. Regardless of how you adopt CSM, make sure whatever monitoring infrastructure you put in place will be extensible and relevant to all your current and future use cases.

The goal of compliance is to document and substantiate the controls you have in place to pacify an auditor. It is not to solve actual security problems. Yes, that is a nuance, and if you adequately protect information assets you are likely to be able to prove compliance. But the converse is clearly not true. Just being compliant does *not* mean you are secure.

The goal of compliance is to document and substantiate the controls you have in place to pacify an auditor. It is not to solve actual security problems.

In terms of frequency of monitoring, you have a lot more leeway in this less stringent use case. In the attack and change control use cases, you need to continuously monitor critical assets to identify dangerous situations. But to be compliant you typically need to assess devices quarterly. As long as you are collecting and parsing event logs on protected devices in a secure fashion (PCI Requirement 10), you're good. Well, assuming that compliant equals good, but you are not necessarily secure. To be clear, logging is good, so do more of that. It helps when you have this information during incident response or investigation, so we are happy that PCI and other compliance hierarchies mandate it.

PCI also specifically requires assessment after 'significant' change (Requirement 11.2.3), but what does that mean? That kind of nebulous verbiage gives you the leeway to assess and monitor devices when you want to, and the PCI Council (and card brands) the leeway to string you up after a breach. Compliance mandates like PCI may also specify a more 'continuous' monitoring approach such as an IDS (Requirement 11.4), which is also a good practice. But remember: this use case isn't about being secure — it's about just meeting the base requirements expressly mandated by regulation and/or guidance. So putting up an IDS to monitor your perimeter and fire one alert meets the requirement. Compliance is great, right?

We will climb down off the soapbox now. Smart security professionals realize that compliance is a means to an end. They can use a compliance mandate to free up budget for equipment and processes that also assist with the attack and change control use cases.

Data Sources

The data sources you use for compliance tend to be pretty consistent with the attack use case, though without the more sophisticated telemetry and forensic data you would need to really figure out what happened:

- **Assets:** Your asset base is the fundamental data source for *all* use cases. You need an ongoing discovery capability to detect new devices on your network, and then a mechanism for profiling and classifying them.
- **Events & logs:** Pretty much everything can and should be logged as part of the compliance use case — including security gear, network infrastructure, identity sources, data center servers, and applications. This is helpful for demonstrating that the controls in place work, which is the goal of this use case.
- **Patches:** Keeping a device up to date is typically mandated by compliance regulations, so you need to generate reports showing which devices were updated when.
- **Configurations:** Another aspect of compliance is implementing and maintaining secure configurations. You will need to document the posture of protected devices periodically. Differentials and history are less important because compliance is based on a point-in-time view of your infrastructure.
- **Vulnerabilities:** Mandates also require periodic vulnerability scans of protected devices. So you need to document what was scanned, what was found, and eventually what was fixed, if the scan showed clear deficiencies.
- **Other documentation:** Some mandates also require periodic penetration tests and other less automated functions. So you need the ability to store this unstructured data in the CSM repository if it will be used for compliance automation.

Preparing for the Audit

Unlike more action-oriented decision flows such as the attack and change control use cases, compliance is all about using data to prepare for assessment. You know when you need to be ready — auditors don't show up unannounced like mystery shoppers. You also know the nature of the documentation you need to provide. Shame on you if you aren't prepared for an audit, when you know exactly what's expected and when to deliver it.

To help you prepare for an audit and make it as painless as possible, here is a streamlined process adapted from Mike Rothman's Pragmatic CSO methodology.

1. **Describe your security program:** What about blasting the auditor with all sorts of reports to convince them you know what you're doing? There is plenty of time for that, but only after you provide context for your security program — specifically how your CSM capabilities provide an accurate and timely view of information needed to understand your compliance posture.

2. **Address past deficiencies:** This is probably not your first audit, so you need to update the auditor on how you addressed previous findings. Here you can leverage your CSM platform to search for specifics and substantiate fixes for issues pointed out in the last assessment. One of the best ways to build your credibility with the assessor is to own your past mistakes and prove with data that you have fixed things.

3. **Substantiate your controls:** Now you can work through the data, showing the assessor that you have implemented the necessary controls effectively. This documentation should be straightforward to generate, given that CSM platforms have tons of pre-built reporting templates to relate collected data to the requirements of the leading mandates and regulations.

4. **Streamline preparation:** Compliance is an overhead function so you should focus on reducing the cost to prepare for the audit. Learn from each audit how to more effectively package and present your data. Get a feel for where you need additional information, or perhaps where your existing collection efforts provide more data than you need. Use these lessons to optimize your monitoring environment for this use case, which will make it easier to prepare for the next audit.

5. **Get back to work:** The problem with compliance is that it doesn't directly help you protect your data or meet the other objectives of your security program. The less time you spend preparing for audits, and the quicker you can get them finished, the more time you can spend on other more productive aspects of your job. Compliance is table stakes so you cannot afford to neglect it, but the more time you spend on it, the less you have to take care of more strategic security activities.

Selecting the CSM Platform

> Not to spoil the ending, but shockingly enough, the platform you choose will depend on your use case.

Now you need to decide on the technology platform to aggregate your data sources and perform the CSM analysis. You have a bunch of candidates, and probably a few already operational in your environment — though likely underutilized. Not to spoil the ending, but shockingly enough, the platform you choose will depend on your use case.

Many folks feel their eyes glaze over when someone uses the word 'platform'. Security folks have a long and tattered history with all sorts of 'platforms', none of which have really done what they were supposed (promised) to do. Now we have the opportunity to reset expectations, which is why looking at the CSM platform in terms of use cases is critical. Let's start with general platform requirements and what you need:

- **Secure and scalable:** Depending on your primary use case and the data sources you choose to aggregate, you may have significant scalability requirements. For lighter use cases such as compliance data storage demands are less intense. But we like planning for the future, which means picking a solution that can scale up even if you don't need it yet. That comes back to architecture and deployment models, as described in our Security Management 2.0 paper. Keep in mind that the CSM environment includes sensitive data. So you will want to make sure your platform provides adequate security (strong authentication, data protection at rest, data integrity, etc.) to protect your protected information.

- **Analytics:** Monitoring is all about being able to find patterns in disparate data sources, which requires the ability to analyze lots of data. Does that mean you need "big data" analytics? Again it depends on the use case, but make sure you can both look for patterns you already know about (standard attack scenarios) and also unknown situations that are clearly not normal. Also keep in mind that analytics (and the content to drive the analytics) changes over time, so selecting a platform driven by ongoing security research is critical to keeping the monitoring system current.

- **Agentry:** For the attack and change control use cases you need to get information directly from monitored endpoints, which requires some kind of agent running on devices. Does it need to be a persistent agent? Not necessarily. You can get much of the data you need via

credentialed scans or dissolving agents. But for continuous monitoring you will need something on devices looking for indicators of malicious activity.

- **Flexible alerting:** Collecting data is good, but alerts make that data useful. You will want to ensure each alert provides enough information to actually do something about it. Whether that's a poor man's capability to manage an incident or integration with a broad investigative platform, you will need some way to operationally use the information from the platform. With the increasing availability of third-party threat intelligence you should also look for the ability to pull in external research feeds to search for specific indicators in the monitored environment.

- **Visualization:** A good dashboard environment offers user-selectable elements, and defaults for both technical and non-technical users. The dashboard should focus on the highest-level information (which devices are at risk, aggregate reports, system health, etc.), and provide the ability to drill down as appropriate. Given the current state of technology, a web-based interface with significant customization is now table stakes.

- **Reporting:** If compliance is your primary use case, your requirements are all about reporting. You need to produce artifacts to document how the security monitoring environment substantiates the effectiveness of controls on devices in scope. Even if another use case is your driver you will need some measure of ongoing reporting to satisfy compliance requirements.

Now that we know what the CSM platform is, let's take a minute to mention what it doesn't need to be — at least today:

- **Real time:** One of the biggest confusions in security monitoring is over "real time". You are aggregating data from an event that already happened, so it cannot actually be in real time. That said, the sooner you get the data, analyze it, and determine whether you have an issue, the better. Compliance doesn't require any kind of real-time response. Change control requires more timeliness for critical devices, and the attack use case urgently requires fast reaction, so the shorter the window between event and alert, the better. *But keep in mind that 'real-time' alerts aren't useful if you cannot respond immediately.* If you have a limited triage/ investigations staff (and who doesn't?), that limits the value of 'real-time' response.

- **Big data centric:** Big data is all the rage in all sorts of security discussions. But for compliance and change control it is generally overkill. And depending on the capabilities of your adversaries, advanced analytics may not add value to your efforts.

Eventually you may need a true security analytics platform with pseudo-real-time data collection to drive your CSM process. If you are facing truly advanced attackers you might need much more robust searching and forensics capabilities (perhaps including big data analytics). But if you are starting with compliance or change control, advanced analytics are likely to be more capability than you need.

Doesn't the SIEM Do This?

You could certainly make a case that your current SIEM or Log Management product is already well-positioned to become your CSM platform. SIEM does a good job with most of the requirements above. And it already consumes most of the data sources for our use cases, with the exception of endpoint forensics and network packet capture… and a number of SIEMs are gaining the ability to capture network traffic. So clearly a SIEM is a reasonable choice for a CSM platform.

But there are reasons to be cautious, starting with the fact that a SIEM may be more horsepower than you need for your specific use case. If you just want to generate compliance reports, there are likely several easier options. For change control most SIEMs don't handle configuration assessment and reporting differentials without customization and clever policy building. None of these are show-stoppers but SIEM is likely a good fit for pseudo-real-time detection of attacks.

As with every technology, don't just look at cost from the standpoint of purchase price. SIEM remains a complex technology that will require initial deployment assistance and ongoing tuning to keep the system current and effective. Those costs can get lost in the cost analysis, so keep that in mind.

What about the Vulnerability Management Platform?

You could make a similar case for a vulnerability management platform, as we described in Vulnerability Management Evolution, which provides the ability to see what's vulnerable and what has changed, in order to determine areas of potential exposure. For the compliance and change control use cases a VM platform makes a lot of sense. Most VM products and services offer configuration assessment as a core feature, with capabilities such as integrated log/event aggregation and alerting.

But be wary of VM platforms without a scalable data model that can evolve to handle additional data sources over time. Again, depending on your use case, you may not need those capabilities immediately, but don't let a short-sighted technology choice sacrifice your ability to grow into the attack use case someday. Investing in CSM is a strategic decision to execute on your security program, so make sure your vulnerability management vendor is constantly adding new capabilities and services to their platform. That might include web application scanning, passive discovery, file integrity monitoring or log aggregation. That kind of expansive scope shows a platform mentality, which is a good sign for your ability to handle future requirements.

But when you take a step back to consider, the lines between SIEMs and vulnerability management platforms continue to blur. Over time we expect a consolidation of these product categories into broader security monitoring platforms with advanced analytics. There is no good reason to maintain two separate products or services to detect attacks *over time*.

Yet, with many organizations already having deployed both SIEM and VM there will be a period of coexistence on the path to a common platform. The length of this period will be enterprise-specific and dependent on the degree of organizational separation (if there are two separate groups responsible for SIEM and VM), the ability for a CSM platform to support both the SIEM and VM features, and cost. Clearly consolidating onto a common platform is preferable, but not at the exclusion of functionality or causing organizational disruption.

In the cases where consolidation is not practical, you'll want to define which system is the primary alerting and visualization tool and then ensure the other supporting platforms can send data to the primary for analysis in a timely and accurate fashion.

> When you take a step back to consider, the lines between SIEMs and vulnerability management platforms continue to blur. We expect a consolidation of these product categories into broader security monitoring platforms with advanced analytics.

To Cloud or Not to Cloud

Many providers offer managed security monitoring platforms (typically from the vulnerability management side) to take the operational and 24/7 responsibility of managing a SOC (Security Operations Center) off the enterprise's plate. The question of whether a cloud-based service is the best choice ultimately comes back to use cases (again!). Under the right circumstances a managed CSM offering makes very good sense.

Service offerings are often marketed on the following capabilities:

- **24/7 device monitoring:** You have a ton of computing devices but inadequate resources to properly monitor them. That's a key situation where managed security monitoring can help. These services are generally architected to aggregate data on-site (via an appliance) and ship the collected data to the service provider for analysis and alerting. The provider should have a correlation system to identify issues and a bunch of analysts who can verify issues quickly and notify you of potential problems. But few of these services support device-centric agents or passive discovery/monitoring for the pseudo-real-time detection that is so valuable in the attack use case.

- **Vulnerability management & configuration assessment:** Vulnerability management and configuration assessment were two of the first capabilities to migrate to the cloud. These services are mature and scalable, and offer comprehensive reporting. Areas of ongoing evolution include support for device agents and more sophisticated alerting, but rolling out new services tends to be faster and less disruptive with cloud infrastructure.

- **Compliance reporting:** Another no-brainer for a services option is basic log aggregation and reporting — typically driven by compliance. This isn't very complicated, and is a good fit for a service offering. It also gets you out of the business of managing storage and updating reports when a requirement or mandate changes — providers should handle all that.

As much as it makes security purists wince, every buying decision comes down to economics. Depending on your funding model and your organization's attitude toward capital expenses, leasing a service for security monitoring may be a better option than buying outright — even if that means compromising a bit on functionality. Of course there are other ways to turn a capital purchase into an operational expense, and we're sure your CFO will have plenty of ideas on that front, but buying a service is a simple way to avoid capital expenditure.

Make sure any managed service meets your needs *before* you consider economics — especially if there is a risk that Accounting might drive you into a long-term commitment to an unsuitable product. No OpEx vs. CapEx tradeoff can make a service meet security requirements.

Evolving to CSM

Depending on which platform you choose to build your CSM capability on, you may be simply adding capabilities to an existing in-house product, or you could be facing a rip and replace of existing technology. In the latter case we suggest you consult the very detailed process to rip and replace your SIEM in Security Management 2.0. The technologies may vary, but the process required to move to a new platform is the same.

If you are looking at basically adding a new use case to an existing platform you can take a more streamlined approach. You still need to both plan and implement, but much less than for a platform migration. Your plan needs to be very clear and specific about when new capabilities are added if new software or modules are required, and where any additional data will come from; then smoke-test the new functionality to ensure it doesn't break anything and that it produces correct information; finally determine who will perform the work.

Plan

- **Identify new data sources:** First you need to figure out what new data you need in your CSM platform for your new use case. For example if you are expanding from compliance to change control you will need a history of vulnerabilities, patches, and configurations so you can see the differentials. Where will you get that data? How can you automate importing the data into the system?

- **Define visualizations, alerts, and reports:** Once the data is in the system, what do you do next? You need to figure out how it can help solve your problem. That means designing visualizations and/or dashboards to help make decisions. You also need to figure out which alerts and reports are relevant for your particular use case. In the planning phase you define a place to start, but that may not be where you finish. Considerable tuning will be required to make each function useful.

- **Allocate resources:** Who does the work? When will they do it? How long will it take to add new capabilities? Depending on the sophistication of your use case and your internal resources, this may be a good time to engage professional services and enlist third-party assistance.

- **Define the timeline:** Estimate the time it will take to roll out your new use case, including time for testing and verification. There is likely to be some 'guesstimation' but you just need rough numbers. Plan the project commencement date and publish to the team. Solicit

feedback and adjust before commencing, because you need to share accountability with the operations team(s) to make sure everyone is invested in the project's success.

- **Preparation:** Do as much work as possible before you go into production. Depending on your platform, you may be able to set up a different instance or implementation to identify and fix issues with before they become problems in production.

Implement

- **Import new data:** This varies based on the deployment model you choose (on-site or cloud), but rolling out a new use case will require you to add new data to the system. You may do a bulk import if you are pulling from an existing system such as patch or configuration management, or you might just start collecting new data as needed. This might involve installing agents on critical devices. Verify the system is collecting all the new data correctly, and that there are no issues with data accuracy or integrity. You will be making operational decisions based on this data so you need it to be accurate.

- **Install policies, dashboards, and reports:** Next deploy the rules that comb through your data and fire alerts for your use case. Hopefully you created most of what you need earlier, during the planning stages. For pseudo-real-time analysis you will need to tune the rules. Each additional rule incurs a significant processing cost. It's math — analyzing multiple data sources against many rules causes the system to do exponentially more work, impacting performance and throughput.

- **Test and verify:** Are your dashboards and reports being generated properly? Are the correct alerts firing in a timely fashion? Generate copies of reports and send them to the team for review. Verify that you get what you need — now is the time to find any problems, while you can find and fix problems *before* they hit production.

At this point your new use case is operational and you are benefitting from continuous security monitoring. But attaining CSM is only the first part of your journey. New technology deployments and capabilities such as cloud computing, as well as emerging attacks, will require you to continuously evolve your security monitoring environment to keep pace.

New technology deployments and capabilities such as cloud computing, as well as emerging attacks, will require you to continuously evolve your security monitoring environment to keep pace.

CSM — Step by Step

When looking at the amount of work to embrace a Continuous Security Monitoring approach, it can be a little overwhelming at first glance. We thought it would be helpful to break the work into a set of logical phases and list the tasks involved in each phase. We'll start with the functions needed for all the use cases and then go through the specifics of each use case.

As with our Quant research, these tasks lists represent a very detailed and granular set of activities. Not all of these activities may be appropriate for your environment or makes sense with your interpretation of the use case. It's really just a representative list to give you a place to start planning your activities.

Requirements for All Use Cases

These tasks are required for each use case. First, you need to select the technology foundation for your CSM initiative. After that you implement the technology, integrate with other systems, and the discover the assets. Regardless of what problem you are solving, you can't monitor it unless you know it exists.

Task	Description
Select technology	As described in the paper, determine which technology with serve as the CSM platform.
Define feeder systems	Are there other complimentary enterprise security systems that can feed data to the CSM platform? If so, then determine how to integrate.
Implement platform	Install and/or update the platform to implement the use cases needed for CSM.
Discover assets	Identify the assets in your environment either via an active or passive scan of all network address space. You could also import information from a CMDB or other asset repository.
Classify assets	As described in the paper, you need to classify assets to determine criticality.

Phase 1: Compliance

Task	Description
Confirm regulations/mandates	Ensure you have full coverage for all regulations and mandates you need to report against.
Determine data sources	Given the regulations/mandates, assemble the list of data sources needed to substantiate controls.
Define frequency of data collection	The compliance use case is not about identifying change or attacks, so frequency of data collection is dependent on the regulation.
Identify in-scope devices	Which devices need to be reported on? For example, PCI-DSS is only concerned with devices with access to protected data. So you don't need to collect from devices without that access.
Customize reports	Many monitoring tools come with pre-packaged reports for most regulations/mandates. Using those templates as a starting point, factor in feedback from past audits and build the reports you need for audits/assessments.
Verify accuracy of data	As part of the QA process for the reports, ensure the data in the reports is consistent with the data from your other security/ monitoring tools.
Collect data	Once QA is passed, move the data collection efforts into production and collect at the frequency defined above.
Run reports	In preparing for an upcoming assessment, generate the reports you'll need for the specific regulation/mandate. Ensure data accuracy.
Refine and tune reports	At the end of the audit, revisit the reports and tune based on feedback from the assessor and the periodic changes to the regulations/mandates.

Phase 2: Change Control

Task	Description
Implement Agentry	After classifying the devices (above), install agentry on those critical devices requiring truly continuous monitoring.
Define scanning frequency	For those devices not requiring an agent, define the frequency of scanning. Remember this requires a balance between consuming resources, the ability to work through alerts (found via the scans), and the need to minimize the window of exploitation.
Define alerts	Define the alerts based on the criticality of device and type of configuration change.
Configure dashboards and alerts in CSM platform	Look at how you want to visualize the change reporting and implement those dashboards in the CSM platform.
Collect data from assets	Once the alerts and dashboards are configured, start collection data from the assets in a burn-in period.
Verify accuracy of data	Verify the accuracy of the data to ensure nothing was lost in translation, normalization, etc.
Tune alerts and thresholds	Once the system has been monitoring for a period of time, go back and refine the alerts, thresholds, and dashboards to ensure you meet the needs of your change control monitoring use case.

Phase 3: Attacks

Task	Description
Implement Agentry	For critical devices requiring continuous monitoring, you'll need to deploy an agent on the device.
Define scanning frequency	For those devices not as critical, you can scan periodically. So again, based on the asset classification, scan as often as needed to ensure if the device is attacked, you'll know in sufficient time.
Build threat models for attacks and build alerts	You need to know what you are looking for in order to set alerts to find it. This requires you to understand the interactions between the different data sources you are analyzing (firewall logs vs. identity stores vs. database transaction records). How do you know that? Basically you need to do a threat modeling exercise based on the kinds of attack vectors you want to find.
Define visualizations	Determine which dashboards and other visualizations you want to see everyday and other means of drilling down into the data.
Define alerts	Based on the threat models, define the alerts. This involves defining alert priorities and thresholds for the conditions causing the alerts.
Establish escalation	Once an alert happens, you'll need to validate the alert and then marshal resources to deal with it. Define the escalation process and specific accountabilities.
Configure dashboards and alerts in CSM platform	Implement the alerts and visualizations in the CSM Platform.
Collect data from assets	Start collecting data from the assets to be monitored.
Verify accuracy of data	Ensure the data is accurate.
Tune alerts and thresholds	As alerts come into the system and you validate/investigate the alerts, tune the thresholds and the alert conditions to improve accuracy and minimize false positives.
Define actions on alerts	Identify root cause, determine extent of compromise, escalate to operational teams or incident response team, as necessary.

Summary

Given that you can't get ahead of the threat, your success at protecting critical corporate data is to react faster to imminent attacks. Yet, a bulk of security spending continues to funnel to outdated controls that neither deter the attackers nor provide enough information to clean up a compromised device. The good news is the US Federal Government and a variety of industry-specific security mandates have gotten religion about the importance of security monitoring. In face, the industry has come around to the mentality that a point in time assessment no longer provides sufficient information to detect attacks or understand data loss. But security monitoring means a lot of things to a lot of people, and it can be confusing to know how much monitoring is enough.

Securosis advocates a risk-based monitoring approach, involving classifying assets based on the perceived risk to the organization if compromised. Based on the asset classification, devices are monitored *as frequently as they need to be*. Critical devices should be **continuously** monitored to alert as soon as anomalous activity or unauthorized change is detected — as those actions tend to be first indicators of a successful attack. Less critical devices can be assessed periodically, providing the ability to match the scrutiny on the device to its importance to your organization.

Additional use cases beyond detecting attacks for continuous security monitoring include monitoring for change control and monitoring for compliance. Regardless of the use case deployed initially, it's wise to invest in a monitoring technology platform applicable to all the use cases, across your entire enterprise.

The good news is that this continuous security monitoring platform may already be installed in your environment, you just may not be not leveraging all of the capabilities in the platform. Whether you are extending the use of an existing technology, or deploying something new — planning the implementation remains a key requirement for a successful evolution to these new capabilities.

And one final note, the first set of alerts that come streaming out of your CSM platform is not the end of your journey. It's merely the beginning. With the rapid evolution of both attack tactics and your own technology infrastructure, you'll need to be continually adapting and evolving your monitoring capabilities to keep pace with the attackers. But given the reality that you can't stop them, your best path to success is to detect them as early as possible.

If you have any questions on this topic, or want to discuss your situation specifically, feel free to send us a note at info@securosis.com or ask via the Securosis Nexus (http://nexus.securosis.com/).

About the Analyst

Mike Rothman, Analyst/President

Mike's bold perspectives and irreverent style are invaluable as companies determine effective strategies to grapple with the dynamic security threatscape. Mike specializes in the sexy aspects of security — such as protecting networks and endpoints, security management, and compliance. Mike is one of the most sought-after speakers and commentators in the security business, and brings a deep background in information security. After 20 years in and around security, he's one of the guys who "knows where the bodies are buried" in the space.

Starting his career as a programmer and networking consultant, Mike joined META Group in 1993 and spearheaded META's initial foray into information security research. Mike left META in 1998 to found SHYM Technology, a pioneer in the PKI software market, and then held executive roles at CipherTrust and TruSecure. After getting fed up with vendor life, Mike started Security Incite in 2006 to provide a voice of reason in an over-hyped yet underwhelming security industry. After taking a short detour as Senior VP, Strategy at eIQnetworks to chase shiny objects in security and compliance management, Mike joined Securosis with a rejuvenated cynicism about the state of security and what it takes to survive as a security professional.

Mike published The Pragmatic CSO <http://www.pragmaticcso.com/> in 2007 to introduce technically oriented security professionals to the nuances of what is required to be a senior security professional. He also possesses a very expensive engineering degree in Operations Research and Industrial Engineering from Cornell University. His folks are overjoyed that he uses literally zero percent of his education on a daily basis. He can be reached at mrothman (at) securosis (dot) com.

About Securosis

Securosis, LLC is an independent research and analysis firm dedicated to thought leadership, objectivity, and transparency. Our analysts have all held executive level positions and are dedicated to providing high-value, pragmatic advisory services. Our services include:

- **The Securosis Nexus**: The Securosis Nexus is an online environment to help you get your job done better and faster. It provides pragmatic research on security topics that tells you exactly what you need to know, backed with industry-leading expert advice to answer your questions. The Nexus was designed to be fast and easy to use, and to get you the information you need as quickly as possible. Access it at <https://nexus.securosis.com/>.

- **Primary research publishing**: We currently release the vast majority of our research for free through our blog, and archive it in our Research Library. Most of these research documents can be sponsored for distribution on an annual basis. All published materials and presentations meet our strict objectivity requirements and conform to our Totally Transparent Research policy.

- **Research products and strategic advisory services for end users**: Securosis will be introducing a line of research products and inquiry-based subscription services designed to assist end user organizations in accelerating project and program success. Additional advisory projects are also available, including product selection assistance, technology and architecture strategy, education, security management evaluations, and risk assessment.

- **Retainer services for vendors**: Although we will accept briefings from anyone, some vendors opt for a tighter, ongoing relationship. We offer a number of flexible retainer packages. Services available as part of a retainer package include market and product analysis and strategy, technology guidance, product evaluation, and merger and acquisition assessment. Even with paid clients, we maintain our strict objectivity and confidentiality requirements. More information on our retainer services (PDF) is available.

- **External speaking and editorial**: Securosis analysts frequently speak at industry events, give online presentations, and write and/or speak for a variety of publications and media.

- **Other expert services**: Securosis analysts are available for other services as well, including Strategic Advisory Days, Strategy Consulting engagements, and Investor Services. These tend to be customized to meet a client's particular requirements.

Our clients range from stealth startups to some of the best known technology vendors and end users. Clients include large financial institutions, institutional investors, mid-sized enterprises, and major security vendors.

Additionally, Securosis partners with security testing labs to provide unique product evaluations that combine in-depth technical analysis with high-level product, architecture, and market analysis. For more information about Securosis, visit our website: <http://securosis.com/>.