Interested in learning more about security?

SANS Institute InfoSec Reading Room

Security Visibility in the Enterprise

What is Security Visibility? Don't our tools already do that? We pass compliance, aren't we secure? In this description of a security visibility program we address many common questions related to security project management from the perspective of a large enterprise and its multi-year journey.

Security Visibility in the Enterprise

GIAC (GCPM) Gold Certification

Author: Jim Hendrick, jrhendri@roadrunner.com
Advisor: Barbara Filkins

Accepted: September 2, 2014

Abstract

What is Security Visibility? Don't our tools already do that? We pass compliance, aren't we secure? In this description of a security visibility program we address many common questions related to security project management from the perspective of a large enterprise and its multi-year journey. Many security tools or services promise to provide protection, deliver insight, achieve compliance, and many other things. And many projects run using vendor "best practices" leave the customer feeling either unsatisfied or overwhelmed. This program took a large corporation from basic logging for compliance using an external managed monitoring service on a journey to bringing monitoring and analysis into the core of the organization. More than a "how to" on deploying a SIEM, this paper describes common pitfalls and at least one way to avoid them. While by no means the only way to improved security visibility, the results show that success is within reach.

1. Introduction

A large (Fortune 100) company decided to improve its corporate "security visibility." Through this effort they intended to move from simply meeting regulatory and compliance requirements toward a more mature model capable of focusing on specific areas of risk. The objectives of the program were simple but (as you will see) not easy. High level goals included:

- Align security efforts with an ongoing process to analyze risk.
- Improve overall security visibility with the ability to produce meaningful metrics on how well things are working.
- Maintain compliance with multiple standards including Payment Card Industry (PCI)[1], Sarbanes-Oxley (SOX)[2] attestation ability, and internal audit.
- Migrate from an external Managed Security Services Provider (MSSP) to an internal Security Incident and Event Management system (SIEM).
- Establish and staff an internal Security Operations Center (SOC).

This analysis follows the company's journey at a very high level, identifying the key elements from both a Project Management perspective as well as from the point of view of developing a framework for building better visibility into security issues. The overall effort has been ongoing for more than two years, included two main projects and intersected with several other large projects. Ultimately the approach shows how positive results were achieved and the high-level goals met.

2. From Ideas to Concept

The first challenge in developing any project is taking a group of people with separate ideas and defining the initial concept. Gaining approval for a project is obviously critical, and the initial team including technical and management staff needs to create a clear concise vision of the goal. One of the major challenges in security is clearly

[1] Payment Card Industry Data Security Standard
[2] Sarbanes-Oxley Act of 2002

communicating the right "view" of security to the right audience at the right time. In other words, answering basic questions *appropriate to the audience*.

What is security visibility? It depends on who you ask! One way to frame it is to consider the following often divergent goals:

- Senior management needs a concise view of your threats and ability to respond. Top management wants to know simple things like: "How do I compare with my peers?", "Am I spending the right amount of money?" "Am I better off than I was this time last year?"
- Operational security teams need a high level (near real time) view of events and threats plus the ability to see all the details quickly (Babbin et al., 2006). Operational management has a different need more related to: "Are we going to pass PCI compliance?" "Are there signs of malware in our systems?" "Are our insiders misusing their access?"
- Organizations are trying to build "baselines" of what is considered normal behavior using techniques from business analytics. Front line analysts need even greater detail to answer things like: "Which devices are trying to communicate with known malicious sites on the Internet?" "What systems are probing our networks?", "Are we seeing any indications of <insert latest threat>?"

All of these perspectives are valid pieces of the puzzle. If you cannot explain to top managers how well (or poorly) your security efforts are working, they will be far less likely to agree to your requests for more funding, staff, tools or whatever it is you need. And unless you are using the information that your tools provide, all you have are racks full of blinking lights that generate huge amounts of unused data.

The definition of visibility is the "capability of being readily noticed" or "affording an unobstructed view" (Webster, n.d.). Applying this definition to security within an enterprise, we defined *security visibility as the capability to provide a clear view into the operation of security controls and making the relevant information easy to see.* That is, the program should provide an "unobstructed view" into how our systems

and networks are running and making it "readily noticed" when something unexpected occurs.

There were additional key concerns of our stakeholders we had to address.

Don't our tools already do that? Certainly security tools can provide a lot of very detailed data, often too much unless it can be put into some meaningful context. In fact the fields of Statistics and Data Visualization provide a lot of guidance that actually *reducing the amount of data is important in communicating the message*. But tools can no more provide security than a workshop full of equipment can build a house. Tools simply provide capabilities that make it possible for humans to do something. In our context, security tools generate messages about events but it is up to us to determine which events are meaningful and under what circumstances a given event is meaningful. Turning these events into "visibility" means defining what you want to see as well as figuring out what you can ignore. Once you know this, you can determine which specific logs you need, which events you can exclude (at least initially) and how to build views into the collected data that will let you know when something abnormal happens.

We pass compliance, aren't we secure? Perhaps one of the worst misconceptions is that achieving compliance (PCI or otherwise) is all you need to be secure (Chuvakin, 2010). Regulations like Sarbanes-Oxley (SOX) and standards like those created by the Payment Card Industry (PCI) are created to address specific threats like cardholder data breach for PCI or misleading or fraudulent financial activity with SOX. The goal of these is to direct an organization's actions to meet someone's view of "best practices". Compliance with these is intended to improve overall protection against the identified threats, but cannot "make you secure" by itself.

How can we objectively measure security visibility "best practices"? There is no shortage of security professionals that will be happy to expound on how well you are doing currently or how vulnerable you are depending on whether they think you are hoping to be reassured or need to be scared (often into buying their product or service).

If we try to bring a bit of rigor to security, we can apply lessons from statistics and other analysis disciplines to what has often been more art than science. Hopefully, by replacing some of the fear and myth around security with actual supportable data, the

problems of identifying and quantifying risk, determining how and what to monitor and where to apply resources will be easier to solve by the inclusion of such disciplines as (Jacobs & Rudis, 2014):

- Domain Expertise - Experience in multiple areas within security is necessary to provide focus on the problem.
- Data Management - The increasing volumes of events is driving the need for storing, correlating and reporting in different ways.
- Programming – The ability to develop tools specific to the purposes of security is necessary to analyze data from multiple sources.
- Statistics – These tools and techniques can identify important information within this increasing volume of data. However a bit of care is needed to avoid finding meaning where there is none.
- Visualization to communicate the results to the right audience in the right way.

3. Building the Security Visibility Program

Most large organizations have adopted some aspects of Project Management methodology. A key skill for the project manager is to recognize whether the in-place methodology is appropriate for a specific effort. Test the waters a bit here. If the organization functions well with its methodology be cautious about adding (or removing) more layers of documentation or process. Work within established processes will generally go easier than a new or non-standard request. Even if the change is positive, you may find yourself "fighting" the system. However, if you strongly believe there is too much (or too little) for your project to be successful, you should be able to support your proposed changes to management and modify an existing process to suit.

3.1. Selling the Concept

Once your concept is established, you must gain support from the major groups that will be necessary throughout the project and often beyond. Identifying the key people and determining whether or not they support the concept can help you determine the best way to proceed. If there is resistance, it can be frustrating but often a successful strategy

is to work "behind the scenes" to try and understand the reason for it and see if it can be resolved. Learning to be a good listener will help win people over but remember that *they* need to believe you understand things from their perspective. Simply saying "I understand" does not make it true. Conversely if there is immediate enthusiasm there will be a sense of urgency and an expectation that you are ready to start immediately. Once you sell the concept, be ready to begin. If you know it will take some research before a detailed proposal can be created, you may benefit from doing some of this before you raise the idea. Essentially you need to know your audience and time your proposal accordingly.

In our case, the company had a strong initial sponsor leading their operational security team. He was able to convince senior management of the need for this program. There was already interest in improving overall security even at the Board of Directors level, so there was little effort needed to sell the concept. The knowledge that we did not need to convince upper management that improving visibility was necessary gave the team some latitude as we moved forward to produce a formal project proposal. Essentially it allowed us to focus on putting together a clear vision of how the project would meet this goal.

Knowing early whether your project will have support or whether you need to continually convince people of its merit is critical to success or failure. Starting this effort early will save you time and trouble throughout the project. Even if you find out there is no support for the idea at this time, it is better to find out sooner rather than when you are well into the project.

3.2. From Concept to Project – Building Support

Once you have sold people on the concept the actual "management" part of project management starts. That is, you need to identify the specific processes you will be following, begin producing documentation, and plan how to fit your effort into whatever methodology exists.

In our case, the company uses a phase-gate model consisting of a two-phase structure for initiating projects. At inception, the initial sponsor and a small team prepare a "Stage 1" proposal, gathering data sufficient to produce high-level requirements. This

proposal must be approved by senior management in order to begin spending money and assigning resources. It describes what functionality the proposed project will provide, such as an existing problem statement and how the project will solve it**.** It also needs to provide estimates of how much this will cost and how long it will take. At this first phase, it is expected that estimates should be +/- 50%. If this Stage 1 proposal is successful, it authorizes spending roughly 10 to 20% of the initial estimates in order to refine the overall budget to within +/- 10%, and prepare a "Stage 2 Commit" document. This Stage 2 Commit is used to obtain the final go / no-go for the project.

The format for a Stage 1 proposal will vary somewhat. In our case the actual proposal was submitted for approval after a presentation with senior management that described the current and future state of security monitoring.[3] Given that there was already agreement that we should improve security visibility, the focus was to describe how the proposed solution would work. This was done using several examples, including analogies relating real-world events to security monitoring, to show how important it can be to correlate different events.

We also felt it was important to show that we already had considered several of the critical components, so key individuals from different teams shared the presentation. This not only allowed us to have technical and business experts available to answer questions, but demonstrated that we already had broad support. Finally, the last part of the presentation gave a summary of the requested financials for the initial spend along with a rough timeline including when we would have "Stage 2" ready for final approval.

Note: A word about the dynamics of the Project Management process in the "real world". When a Stage 1 proposal is presented to senior management, the project sponsor tends to become somewhat tied to those numbers. While the intent of Stage 1 is to authorize a limited expenditure to identify unknowns that could impact the project and prepare a more detailed and accurate picture, when returning for Stage 2 approval, the initial sponsor may be reluctant to deliver "bad news" about their initial Cost or Time estimates (see how subtly Functionality got dropped here?) and Stage 2 moves forward essentially with the numbers from Stage 1. This is a very easy trap to fall into (after all,

[3] See high level diagrams in Appendix A "Previous alerting…" and Appendix B "End-state…"

no one likes to admit they were wrong to their management); however it misses the point of the two stage process. The expectation should be set at all levels that projects do not pass into Stage 2 just because of a date on a calendar, nor should the estimates presented in Stage 1 be held up as something not to be questioned, like a quest for the holy grail (ARTHUR: "Good Idea, O Lord!" GOD: "Of course it's a good idea.")[4]*. It is precisely this skill of setting expectations both down and up the management chain that can make or break a project management program. It is far better that some projects either get dropped or sent "back to the drawing board" than if they move forward simply to avoid uncomfortable discussions with senior management.*

While far from having gathered complete requirements, part of the "Stage 1" proposal does include building the framework around which those requirements will be developed. At the point when the Security Visibility Program began, two key components were already well-established at the company. These constraints drove many of the activities needed to start putting together a formal project:

- External MSSP – All security monitoring and alerting had been performed by an external provider for over five years. During this period two major compliance efforts took place. The company went through SOX attestation efforts and PCI testing. Both efforts were developed using the MSSP as critical controls. In addition, a separate internal audit department had modified their "Security Monitoring and Oversight" review around these MSSP controls.
- Internal Log Management – As part of developing the above MSSP project it was necessary to build a large log collection and retention infrastructure. This took place over roughly three years, with several Request For Proposals (RFPs) going out to the major product vendors and eventually a solution was chosen for this implementation.

3.2.1. Let Requirements Drive Scope

Our basic goal was to improve security visibility. It was assumed that this meant "Replace the MSSP" and this quickly led to "We need a SIEM" so these became requirements. As a result, picking a product was one piece of the overall program.

[4] Monty Python and the Holy Grail: http://www.montypython.net/grailwork1.php

A fundamental understanding was that having an external managed service (MSSP) provide alerting was very limiting. Essentially we were only able to use the standard offering and the process of requesting new functionality from the vendor was difficult at best. While meeting the basic regulatory and compliance needs, it was not practical to make many modifications to produce alerts tailored to the needs within the company. Implementing a SIEM and SOC was seen as a way to gain more control and ultimately better visibility.

Rather than simply turning on all default SIEM content (and trying to figure out how to respond), we used the requirements gathering process to identify what was really needed. In business analysis / software development terminology, this would identify "use cases", and that nomenclature was adopted here. As a simple example: "Monitor login events and alert on more than X failures by the same account in Y minutes" could define a very basic use case.

One of the key elements in any project is to involve the right people as soon as possible. This does not mean you need to run every idea through a committee, but once you need to take an idea from concept to something more formal getting the right people early is a real benefit. Not only does it help with requirements gathering, it begins to build a sense of becoming a team or of "ownership" and this can take time so the sooner you start the better. "With people, slow is fast and fast is slow."[5]

The existence of established MSSP and log management programs allowed us to identify the key people we would need to develop requirements. We identified members of the internal audit and compliance teams who were using the MSSP as part of their ongoing efforts. And the people who were already supporting log management were brought on as experts in that area. The one missing component was an existing internal SOC team. There were some resources that were responsible for handling incoming alerts from the MSSP, but it was quickly recognized that a much larger and more formal team would be needed. Discussions on how to address this were started early in parallel with the more detailed requirements gathering around specific alerts and reports would be needed from the SIEM in order to maintain compliance.

[5] Unclear if it originated here, but it is at least re-quoted in Covey (1989).

Working with the internal teams, the regulatory and compliance dependencies were identified and documented as requirements. Note: It was accepted that this would result in replicating some controls that might be inefficient. Given the number and complexity of these controls and how tightly they were built around the MSSP processes, it was agreed that transitioning the source of the compliance artifacts from MSSP to an internal SOC/SIEM was going to be hard enough, and changing their content at the same time was not worth the risk. Once successful replacement of the MSSP was accomplished, the program planned to expand this basic set of features over time.

This concept of starting with basic functionality is key to convincing stakeholders that the goal is achievable. It allows time for related processes to adapt at their own pace instead of creating multiple external dependencies in an already complex project.

Starting with a modest set of use case requirements also led the team to recognize the need to create processes for identifying, defining and building the set of use cases desired. (Actually one can would argue that need would exist in any case, but documenting it as a formal part of the program was a big win.)

When you are building support and beginning to gather requirements, start with some very modest goals that include maintaining key functionality (like compliance) and make use of existing technology (like centralized log management) wherever possible. This will gain you supporters in critical areas and make running the project much easier. Generally people are very willing to help as long as they believe you are considering their needs as well as your own. This is not simply good advice overall; it can either make or break a project.

3.3. Defining the Program Structure - Getting to Stage 2

Once you have defined the concept, established high-level requirements, and identified the right set of people (stakeholders), you can move to a more formal project structure that will carry you through the project.

In our case, after approval of Stage 1 there was a large amount of work necessary to prepare the more accurate set of requirements and estimates needed for the Stage 2 proposal. With large efforts it sometimes helps to create a set of related sub-projects, each

with its own formal project structure including project manager, documentation and reporting processes. As there was a natural split between implementing a SIEM and establishing a new SOC team, the program used two projects, with a single program manager and "steering committee" to coordinate differences in planning.

3.3.1. Building the SIEM

Beginning with the basic set of use cases obtained from looking at the dependencies on the MSSP, several vendors' SIEM offerings were identified. Each vendor was given basic requirements including the necessary use cases along with the design of the existing log management system. After technical and business analysis of the different options was completed, a product selection was made. This vendor then took more detailed data (performance and volume from the existing logging and details on the mandatory use cases) and developed a proposal initially including purchasing, installation *and* customization of the production SIEM.

In our case, while we required development, testing and staging environments in addition to production, these were not in the initial vendor proposal. Whether overlooked in requirements gathering or communications with the vendor, it speaks to the need for the Project Manager to have senior resources on your team to ensure that what is designed and quoted by a vendor actually meets the requirements. Had this gone unnoticed it could have resulted in one of those uncomfortable conversations (umm – we need to buy about four times as much hardware and software as we originally said…)

Along with hardware and software, the SIEM also needed to replace the same alerts and reports that the MSSP was providing in order to support compliance efforts. But how would that work? Many vendors sell compliance packages but one of the issues with the MSSP was lack of flexibility, so we opted for building custom content

Since we had no internal expertise in SIEM development, we sought out vendors that specialized in building and implementing SIEM solutions. A survey of the market led to several vendors who claimed to be able to build our SIEM environment. Reviews of their offerings finally identified a small specialty shop with a convincing proposal. They were vendor-neutral, and presented a convincing process for identifying use case requirements. They had their own library of existing use cases that could be used to kick-

start the effort. Along with building the foundational set needed to replace the MSSP they would be able to deliver a larger set of proposed use cases that could be built as the visibility program matured over time. They had development and testing staff available (both on-site and remote) and were selected to build and deploy the first set necessary for "go live" and also help us develop the life-cycle management processes for use cases in the future.

3.3.2. Building a SOC

This was a new effort requiring creation of a new team and so was more process and organizationally driven than the (arguably easier) task of building a SIEM. While the SIEM project team was selecting vendors and defining technical requirements, the SOC project team was looking at options for how to form this new team. From informal high level conversations, it was determined that the internal SOC could begin using resources from the existing Incident Management team, the ones who monitor and address outages in applications, systems and networks. They had a rich set of skills and processes already in place to manage incidents including identifying the owners of affected systems, contacting the right technical and business staff, and running conference calls at any hour of the day or night 24x7. They also had a well-established way to track and report on incidents (e.g., service level agreements related to time to respond, time to close, and so forth) and these were recognized as being needed for a SOC to succeed.

Once the Incident Management team's senior managers agreed, several models to create a SOC were considered. Again, several outside vendors were contacted and a series of presentations and workshops took place to show different proposed solutions for building this team.

As things progressed, we decided that a multi-tier SOC team that would be created and staffed in a hybrid model comprised of an external vendor and internal resources. The vendor would provide "first tier" staff and be able to increase or decrease the coverage according to demand. Their team would provide round-the-clock monitoring of SIEM alerts, reviewing them and closing the ones that could be done through a scripted procedure. Incidents that required more scrutiny would be escalated to internal

staff, including some members of the security team (the ones who handled the MSSP alerts) as well as newly identified folks from the Incident Management team.

Once this model was chosen, the vendors were asked to provide proposals and bids for this service, customer references were contacted and a SOC vendor was selected.

3.3.3. Achieving "Stage 2 Commit"

At this point, the job was reasonably well defined with the right people, vendors, and tools involved. We needed to now prepare fairly accurate time and cost estimates before we could get final approval to move forward. This involved coordinating two separate projects and three external vendors (one providing SIEM hardware and software, one performing SIEM development, and one to help build and staff the SOC) in order to create an overall estimate accurate to within +/- 10%. Not an easy task at all.

Alongside your internal project challenges, you will often run into external complications to your plans. Our effort was no different. One of the underlying assumptions of both SIEM and SOC projects was that they would tie into the existing internal (incident management) ticketing system. Well we quickly discovered that there was a separate project to migrate that ticketing system to a different product at (almost exactly) the same time! This created the additional requirement to integrate the SIEM with the new ticketing system and make sure the security incident tracking needed by the SOC could be satisfied. Oh -- the SIEM development vendor was acquired by a larger company during this period….

The lesson here is that you need to accommodate unpredicted change. Your project documentation should be written to so you can adapt more easily when unexpected things happen. This is especially critical as requirements get written into contracts. There should be very clear requirements documented as part of the final requirements review that state the basic assumptions, what specific things will be used to measure successful delivery of a milestone and what change control processes will be followed when the need arises, regardless if change occurs.

Needing an overall timeline to be coordinated between the projects -- think critical path but across multiple projects -- we created a simple high-level representation for the regular reporting to the Security Visibility Program Steering Committee.

Quarterly high-level timeline (Program level)

	Q1			Q2			Q3			Q4
SIEM POC										
Use case Development & Deployment			Requirements Gathering			Test Build 1	Deploy Build 1	Test Build 2	Deploy Build 2	
SIEM Prod										
SOC Mobilization			Roadmap / Framework			Final Vendor Negotiation	Vendor on-boarding	SOC Pilot		SOC Go-live
SOC Staffing Tier1 & Tier2			1st Tier1 response procedure			1st use case response moved to SOC team		Tier-2 analysts on-board		
Hire Security & Business Analysts					#1 on-board	#2 on-board	#3 on-board			
Operational Artifacts & Training			Incident Plan update			Use case Run books		Per use case response procedures		
Program Management	Kick-off			CIO Deep Dive						

Within both projects a series of regular meetings was established with representatives from each key team. In order to meet an aggressive schedule (is there any other kind of schedule anymore?) the agreemnet was made that representatives be empowered to the greatest extent possible to make on the spot decisions for their teams including resource commitments and dates for delivering critical pieces of the project.

The first task of these meetings was to prepare detailed project plans for the SIEM and SOC projects and the overall Stage 2 Commit proposal for the Visibility Program. Over a series of weeks, the "final" initial set of requirements was documented including the following requirement categories:

Jim Hendrick <jrhendri@roadrunner.com>

Initial SOC Requirements

- Organizational and Process design
- Staffing and training needs for the analysts at each tier
- An approved network access model for vendor SOC staff at Tier 0
- Creation of SOC Workflows and Reporting requirements
- Validation of SOC Workflow for each SIEM use case as it was developed
- Testing the SIEM interfaces to the SOC (e.g. with the Ticketing system)
- Development of a SOC Governance model

Initial SIEM Requirements

- SIEM architecture with the selected product
- Establishing a Proof of Concept (POC) SIEM environment
- SIEM use case software (system) development lifecycle (SDLC) plan
- SIEM development testing and deployment environments
- SIEM use cases to be part of the initial production "go live"
- Verification of all required data for each SIEM use case
- Creation of a Referential Database and all support processes

Once this plan was agreed on as viable by all the stakeholders, a meeting was scheduled with senior management to present the request for formal "Stage 2 Commit" for the project to go forward. The presentation was framed again as a story, briefly touching on the initial proposed goals. Since a key part of the presentation was to convince the senior leaders we had a cohesive team, the presentation was shared with the project sponsor, the internal architect, and representatives from each vendor speaking to their piece of the proposal. Essentially it told a story of how the existing MSSP process would function given three basic use cases, comparing that to how the proposed SIEM and SOC would provide better results. At the end, the overall timeline looking forward approximately one year was presented, including budgetary numbers.

Clearly how each organization handles the formal request for allocation of project resources will differ. In this case the "Stage-2 Commit" proposal was done at a "CIO Deep Dive" meeting with IT executives representing different business, each with different goals but all reporting to a single CIO. The questions ranged from "How will this impact my PCI compliance?" to "What volume of database transactions do you

expect in the SIEM?" As always, "your mileage may vary" but in this case, our presentation was very well received and having the right project team members to respond readily to each question played a critical part.

3.4. The Project was Accepted! (Now what?)

Great news! Your proposal is now an official project and the most senior management of your organization are supporting it! Oh – and they expect regular reports of its progress describing glowing success after glowing success…

Well – to be fair not really. Most executives do not rise to that level without understanding that nothing is perfect. However there is the temptation to only report good news. Fight this temptation! Not that you should always tell tales of woe since you could be identified as the voice of doom and gloom and the project will be almost automatically seen in a negative light. But don't be afraid to deliver an honest message. If you are encountering problems that your local level of management cannot resolve, you and your local management may need to take it up with more senior people. But hopefully you will have an overall good report showing regular progress, calling out milestones, touching only briefly on some hurdles that were overcome and identifying clearly anywhere that you need more support (obviously including any proposed changes to requirements that change the overall timeline, cost or functionality).

After the initial excitement of Stage 2 approval, our project teams were left to get down to the business of delivering the project. At this point a POC SIEM had been installed and the technical team worked on the development and testing side and the infrastructure components.

3.4.1. SIEM Production Environment

Obviously the full development, testing and production environments needed to be established, including final technical specifications for hardware and software. All the system installation and support details had to be identified and minor issues worked out before anything could be installed.

Our team discovered that the internal server support team was not currently providing the base operating system needed by the SIEM, that the recommended file

system type was also not something the server staff was familiar with. Specific service level agreements needed to be worked out for monitoring and addressing hardware or software outages. A new network VLAN with specific rules governing access had to be designed and created, switches and the appropriate NICs needed to be verified, and all physical data center requirements for rack space, power, cabling and cooling. The support processes for the new application needed to be created including defining how user accounts would be created and use two-factor authentication for the application, and on in great detail. The easiest part was actually doing the work.

While this was resolved quickly, it is a point where more detailed planning might have helped identify these issues earlier.

3.4.2. Detailed Use Case Development and Deployment

Parallel with this, the set of use cases for initial "go live" was finalized to provide basic coverage in several areas:

- Operational (monitoring the SIEM and logging systems themselves)
- Compliance (those needed to maintain SOX or PCI)
- Perimeter (dashboards and alerting for critical network events)
- Critical Access (privileged user groups or access to critical systems)

A total of thirty-five use cases were identified and specifications created for each including necessary details; which specific events from what distinct sources are needed? What referential data will be imported and what will be done with it? What output will the use case create? Will it alert directly? Will it create a regular report for review? Will it populate a dynamic "watch list" for use by other use cases? Will it require such lists *from* other use cases? Will an external "threat feed" of suspicious addresses, sites, domains be used? Will *that* provide addresses only or other data like threat category, degree of "maliciousness", "confidence score" or "last seen date"?

This was essentially a software development effort so all the normal aspects of developing code with a geographically dispersed team needed to be in place. Remote access mechanisms were needed. The team needed "on-boarding" tasks including

defining access permissions, and establishing secure access to a source code control and bug tracking system.

3.4.3. Referential Data

One of our most critical decisions was to create a separate data store where the SIEM could get access to the environment specific data that actually adds the most value in correlating raw events. This data included: a system inventory and which applications they support; definition of specific network ranges; lists of in-scope systems for PCI; user data including lists of privileged users (executive accounts), administrative groups from directory services (domain admins or users authorized to access administrative passwords from the central password vault); and other data sets that were defined as needed by the agreed on use cases.

There are certainly ways to manually define a lot of this in most SIEM tools. And one-off import tasks can be developed. This project decided there would be advantages in the long run if the SIEM could interface with a single database rather than implementing each import method independently. The data in other systems (e.g. asset inventory, LDAP, Active Directory) had to be obtained regardless; however a "referential" database can define a data model specific to what the SIEM needs. Multiple sources were combined in the database and provided to the SIEM as a unified object. Additionally, changes to the source data (either technology or process) are somewhat isolated from the SIEM. A user interface to the referential database itself can be provided to give the security team access to more easily research this data during analysis.

As a key point, you should be wary of adding a new critical system that requires its own ongoing maintenance. If the project does not include agreement and verifiable processes around how each data source ***will be kept up to date*** in the future, the value of the data will rapidly degrade. This does not just apply to the referential data, but that seems most appropriate to include here. In our case, the resources necessary for building this database were not initially fully accounted for in the project estimates.

3.4.4. SOC Mobilization & Staffing

The SOC project was also moving along at this point, including finalizing negotiations with the external vendor, getting their resources on-board, providing

network and system access to their team, and establishing processes for doing all this securely. Along with providing flexible staffing to monitor the SIEM, the selected vendor also provided excellent support in the planning and oversight areas, one reason they were chosen. They were very valuable in helping define initial metrics and service level agreements and worked well with our internal Incident Management team to make sure the "hybrid model" could succeed.

While the external SOC vendor was one critical component, a second equally important effort was underway creating the internal organization. While the Incident Management team did identify existing staff who would be assigned to the new SOC roles, additional positions needed to be created and filled. One of the most critical was a new position for SOC Manager. This person would have overall responsibility for the hybrid organization. This role was filled by various internal managers, with the full time person coming on roughly two months after the initial go-live. This placed even greater reliance on proper planning and documentation so the new manager would be able to get "up to speed" as easily as possible.

A major part of establishing the SOC was ensuring training at the proper levels. External training in security specific areas was provided as well as training with the SIEM itself. Perhaps the most critical was making sure SOC personnel at every tier knew how to respond. As the parallel SIEM use case development proceeded, each released set of code was deployed to the test system and the SOC team was directly involved in all aspects of testing. Workflows were developed and tested to support each use case, and many "dry runs" were performed throughout this phase of the project.

One key milestone in this SOC creation was to cut over a single type of incident to be handled by the SOC to allow the team to begin to gain confidence in themselves and their team members. This was a "user malware" use case and included early SIEM alerts but also relied heavily on an existing incident type that was being handled by the Incident Management team when a user called the help desk to report suspected malware. This gave the new SOC team the benefit of experience from the existing help desk ticket process, while beginning to add in the SIEM alerting and the new workflow.

All through the process of creating this new SOC team, the focus was on building confidence in the team. Whether this was from training, including them in the process development or early exposure to one use case the result was building a feeling that this is *their* project and the new SOC was *their* team. You need to realize that building loyalty in any team is more critical than any product or vendor selection. The right team can achieve success with nearly any tool.

3.4.5. Operational Planning, Documentation and "Go-live"

In support of the SIEM and SOC efforts was a parallel set of tasks to make sure the proper processes were in place, supported by clear documentation that was readily available. Additional technical writing resources were added to make sure everything was in a central repository and had a common format and degree of detail.

As continuity in regulatory and compliance was identified early on as a critical requirement, appropriate individuals were involved in many phases of the project, especially as the go-live date approached. During this phase they were directly involved to make sure what was produced would support their continued efforts.

Testing continued as the initial set of use cases were built and tested, bugs identified and addressed, and workflow documentation updated. As most use cases reached production readiness, a trial period of two weeks took place during which a SOC Pilot was conducted, running as they would in normal operations. This included dry-runs each morning, with the rest of the day spent in review meetings to identify what worked well and what needed to be fixed. It also included the off-site SOC staff working outside of normal business hours and presenting what they would have escalated to the on-site teams the next morning.

Throughout this pre-go-live phase, technical tuning was identified and several use cases were sent back for rework. It was also decided to have the SIEM produce reports at first versus automatically creating tickets for each alert. Those reports provided the data to tune use cases without the added overhead of a large volume of (false positive) alert tickets in the system. The team also begin developing overall metrics and reporting at the program level. A full-time SOC manager had not yet been hired, so this effort did not get the attention it would once the new manager came on-board.

Jim Hendrick <jrhendri@roadrunner.com>

By the go-live date, the new SOC team was nervous but ready. The SIEM itself had been running with live data for several months. The development team was accustomed to the environment and had a well-established process for tuning changes. The audit and compliance teams had approved the new artifacts. And the Security Visibility Program Steering Committee gave the go ahead to cut-off the feed to the old MSSP (their contract end date was chosen as the go-live date although it could have been extended if the SOC and SIEM had not been ready). While the first few weeks had occasional bumps, all in all the go-live went very well.

Both projects under the Security Visibility Program had many components that transitioned to ongoing operations. Functions like tuning use cases, refining workflows, creating and tracking metrics continue. In fact, the management of use cases itself is worthy of dedicated study. (In this instance, it is guided by the periodic assessment of overall threat to the organization.). However the closure of the projects in this program was met roughly two months after the go-live date (when the hiring of a full-time SOC manager took over responsibility for daily operations).

Since the projects formally ended, the Security Visibility Program is showing successes including:

- New use cases are reviewed against ongoing measurement of risk.
- The number of tickets per month is more predictable than from the MSSP.
- Compliance efforts successfully transitioned to the new system.
- SIEM tuning is ongoing as an operational process to reduce false positives.
- SOC Incidents have more consistent data when beginning an investigation.

4. Conclusions

The lessons from this program apply not only to security but to all projects. There are lots of resources that will happily expound on the current "management techniques" or provide reams of processes and procedures that should be applied, but hopefully the items below from our experience will provide you some guidance.

- **Think Long Term**. Start with an overall direction in mind and apply steady pressure in that direction. Even outside formal Project Management, consistent leverage can move even large organizations if maintained over time.
- **Be Flexible**. Realize no technology or formal organizational structure is permanent. Look for points where things connect and try to design in flexibility.
- **Adopt and Adapt**. Find aspects of formal Project Management that seem to apply within your organization, and adapt as you go. But if a model exists, you will find more support if you can use terminology and processes already accepted.
- **Remember the basics** - Define initial requirements, get the right people involved, make initial estimates, define how to measure success, and control changes.
- **Tune this to Your Organization** – There is a principle in seamanship that essentially says "obey all the rules of the road, but break any rule needed to avoid collision". Applying this here boils down to recognizing when to push for change (e.g. a PM technique) and when not to. Remember the goal is to get your projects done successfully – and if that means using a light application of PM doctrine in one place, and more formal planning in another, so be it.
- **Manage your vendors carefully** – Many vendors provide good products and services, but remember that they are primarily driven by their profits (as is your organization, most likely). This simply stated means their loyalty is to their own company. And on projects where multiple vendors are involved you need to be multiply vigilant in your managing the relationships. It is far too easy for vendors to engage in finger pointing, leaving you stuck with the problem.

Jim Hendrick <jrhendri@roadrunner.com>

5. References

Babbin, J. et al. (2006). *Security log management identifying patterns in the chaos*. Rockland, MA: Syngress.

Chuvakin, A. (2010). *PCI compliance understand and implement effective PCI data security standard compliance.* (2nd ed.). Burlington, MA: Syngress.

Covey, S. (1989). *The seven habits of highly effective people: Restoring the character ethic*. New York: Simon and Schuster.

Jacobs, J., & Rudis, B. (2014) *Data driven security: Analysis, visualization and dashboards*.; Wiley.

Jaquith, A. (2007). *Security metrics: replacing fear, uncertainty, and doubt*. Upper Saddle River, NJ: Addison-Wesley.

Payment Card Industry Data Security Standard v3. (2013, November). Retrieved September, 2014, from https://www.pcisecuritystandards.org/documents/PCI_DSS_v3.pdf

PCI DSS Quick Reference Guide. (2010, October). Retrieved September, 2014, from https://www.pcisecuritystandards.org/documents/PCI SSC Quick Reference Guide.pdf

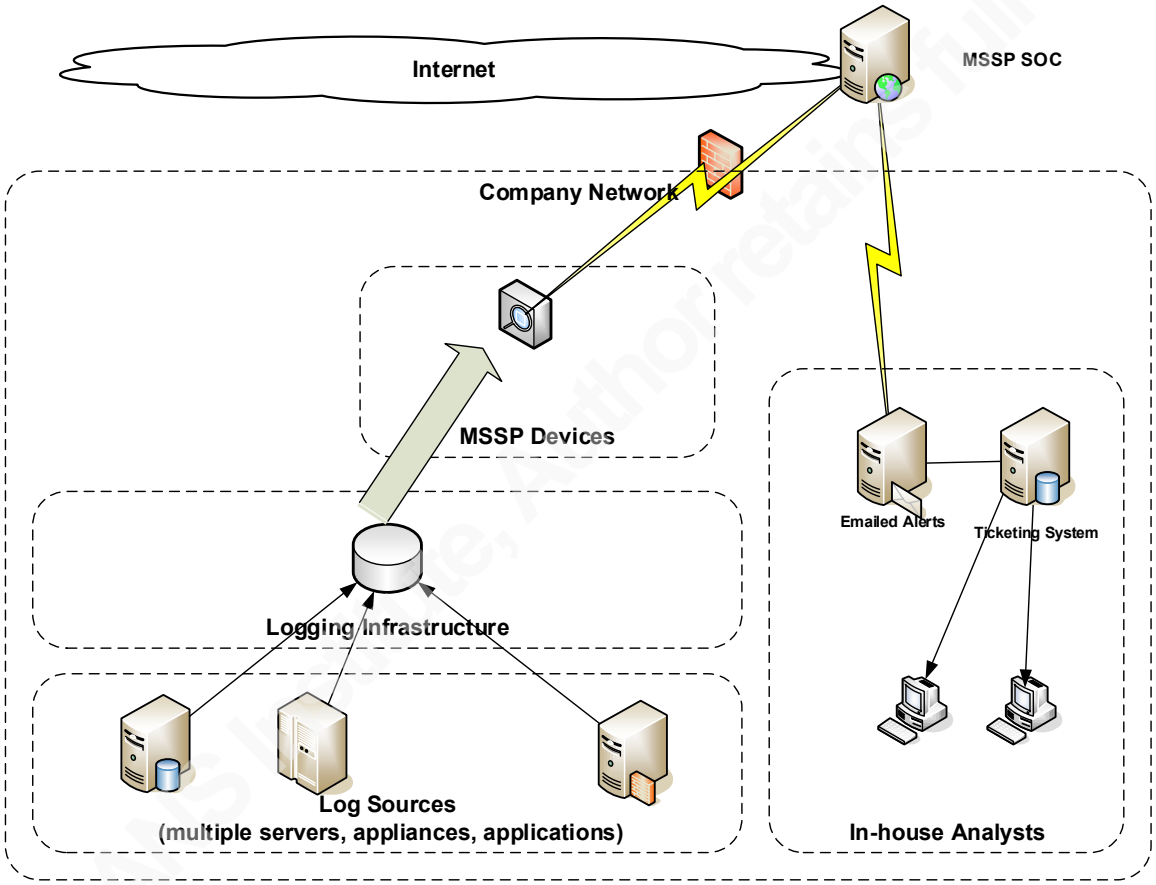
Sarbanes, P., & Oxley, M. (2002, July 1). Sarbanes-Oxley Act of 2002. Retrieved September 8, 2014, from http://www.gpo.gov/fdsys/pkg/STATUTE-116/pdf/STATUTE-116-Pg745.pdf#page=1

Sarbanes-Oxley Basics. (2011). Retrieved September 8, 2014, from http://www.sarbanes-oxley-101.com/sarbanes-oxley-faq.htm

Visibility [Def 2]. (n.d.). In Merriam Webster Online, Retrieved August 29, 2014 from http://www.merriam-webster.com/dictionary/visibility.

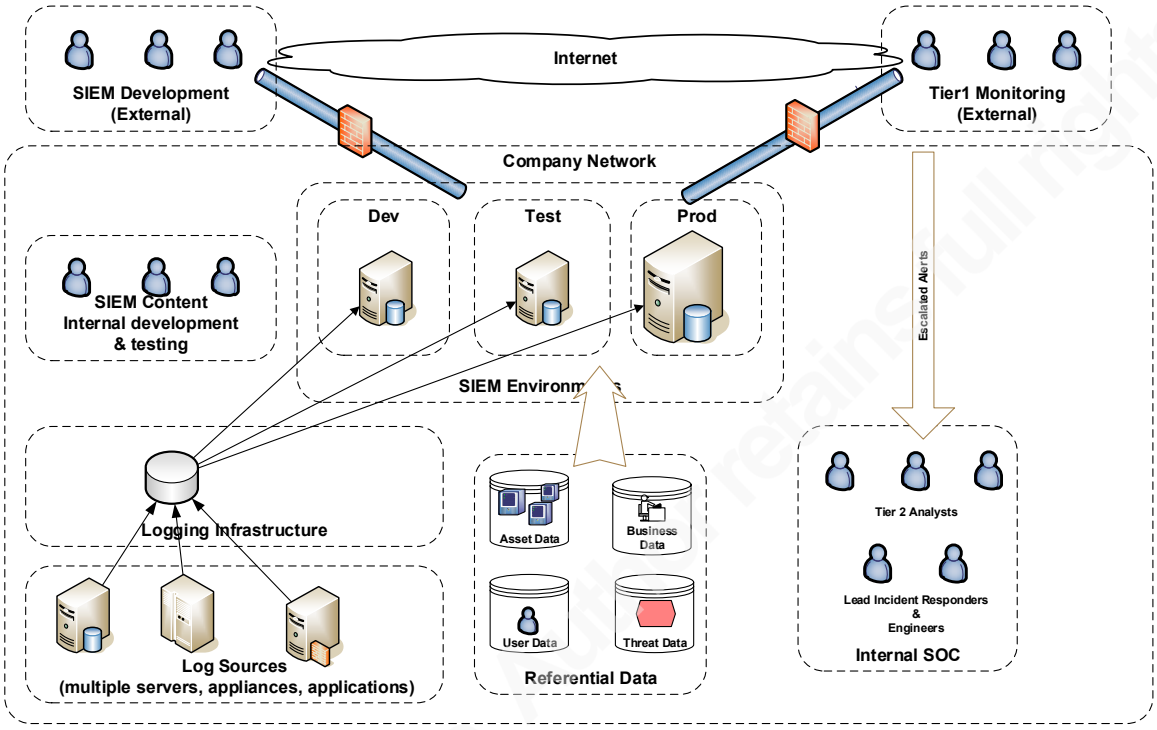
Jim Hendrick <jrhendri@roadrunner.com>

6. Appendix A: Previous alerting high level flow, using MSSP



Jim Hendrick <jrhendri@roadrunner.com>

7. Appendix B: End-state alerting using SOC & SIEM



Jim Hendrick <jrhendri@roadrunner.com>

Last Updated: June 22nd, 2018

Upcoming SANS Training

Click Here for a full list of all Upcoming SANS Events by Location

SANS Paris June 2018	Paris, FR	Jun 25, 2018 - Jun 30, 2018	Live Event
SANS Minneapolis 2018	Minneapolis, MNUS	Jun 25, 2018 - Jun 30, 2018	Live Event
SANS Vancouver 2018	Vancouver, BCCA	Jun 25, 2018 - Jun 30, 2018	Live Event
SANS London July 2018	London, GB	Jul 02, 2018 - Jul 07, 2018	Live Event
SANS Cyber Defence Singapore 2018	Singapore, SG	Jul 09, 2018 - Jul 14, 2018	Live Event
SANS Charlotte 2018	Charlotte, NCUS	Jul 09, 2018 - Jul 14, 2018	Live Event
SANSFIRE 2018	Washington, DCUS	Jul 14, 2018 - Jul 21, 2018	Live Event
SANS Cyber Defence Bangalore 2018	Bangalore, IN	Jul 16, 2018 - Jul 28, 2018	Live Event
SANS Pen Test Berlin 2018	Berlin, DE	Jul 23, 2018 - Jul 28, 2018	Live Event
SANS Riyadh July 2018	Riyadh, SA	Jul 28, 2018 - Aug 02, 2018	Live Event
Security Operations Summit & Training 2018	New Orleans, LAUS	Jul 30, 2018 - Aug 06, 2018	Live Event
SANS Pittsburgh 2018	Pittsburgh, PAUS	Jul 30, 2018 - Aug 04, 2018	Live Event
SANS San Antonio 2018	San Antonio, TXUS	Aug 06, 2018 - Aug 11, 2018	Live Event
SANS August Sydney 2018	Sydney, AU	Aug 06, 2018 - Aug 25, 2018	Live Event
SANS Boston Summer 2018	Boston, MAUS	Aug 06, 2018 - Aug 11, 2018	Live Event
Security Awareness Summit & Training 2018	Charleston, SCUS	Aug 06, 2018 - Aug 15, 2018	Live Event
SANS Hyderabad 2018	Hyderabad, IN	Aug 06, 2018 - Aug 11, 2018	Live Event
SANS New York City Summer 2018	New York City, NYUS	Aug 13, 2018 - Aug 18, 2018	Live Event
SANS Northern Virginia- Alexandria 2018	Alexandria, VAUS	Aug 13, 2018 - Aug 18, 2018	Live Event
SANS Krakow 2018	Krakow, PL	Aug 20, 2018 - Aug 25, 2018	Live Event
SANS Chicago 2018	Chicago, ILUS	Aug 20, 2018 - Aug 25, 2018	Live Event
Data Breach Summit & Training 2018	New York City, NYUS	Aug 20, 2018 - Aug 27, 2018	Live Event
SANS Prague 2018	Prague, CZ	Aug 20, 2018 - Aug 25, 2018	Live Event
SANS Virginia Beach 2018	Virginia Beach, VAUS	Aug 20, 2018 - Aug 31, 2018	Live Event
SANS San Francisco Summer 2018	San Francisco, CAUS	Aug 26, 2018 - Aug 31, 2018	Live Event
SANS Copenhagen August 2018	Copenhagen, DK	Aug 27, 2018 - Sep 01, 2018	Live Event
SANS SEC504 @ Bangalore 2018	Bangalore, IN	Aug 27, 2018 - Sep 01, 2018	Live Event
SANS Wellington 2018	Wellington, NZ	Sep 03, 2018 - Sep 08, 2018	Live Event
SANS Amsterdam September 2018	Amsterdam, NL	Sep 03, 2018 - Sep 08, 2018	Live Event
SANS Tokyo Autumn 2018	Tokyo, JP	Sep 03, 2018 - Sep 15, 2018	Live Event
SANS Tampa-Clearwater 2018	Tampa, FLUS	Sep 04, 2018 - Sep 09, 2018	Live Event
SANS MGT516 Beta One 2018	Arlington, VAUS	Sep 04, 2018 - Sep 08, 2018	Live Event
SANS Cyber Defence Canberra 2018	OnlineAU	Jun 25, 2018 - Jul 07, 2018	Live Event
SANS OnDemand	Books & MP3s OnlyUS	Anytime	Self Paced