Global Information Assurance Certification Paper

Critical Security Controls: Software Designed Inventory, Configuration, and Governance

GIAC (GCCC) Gold Certification

Author: Lenny Rollison, lennyrollison@hotmail.com
Advisor: Adam Kliarsky

Accepted: May 19, 2016

Abstract

How can the lack of inventory be the root cause of audit, compliance, and security failure? Security at the "speed of change" can no longer be a sociological disorder (Conner, 1993). Manage and automate the CIA triad from the rack to the internet with software designed solutions that take chaos back to the point of origin and drive "risk based visibility" (Williams, 2015).

1. Introduction

1.1. Revolution of Intelligence

The events of September 11, 2001, show us how isolated communication and the inability to share intelligence could paralyze decision making (Johnston, 2003). Failure, and the impact, is known to drive response, or reaction, as does fear (Williams, 2015). As an example, intelligence agencies, first responders, and countries around the world seemed to have woken up to communication breakdown and warnings after gathering the evidence, and collaborating across the globe. Without a believable threat, risk was accepted, and the likelihood ignored. In the movie *Tora! Tora! Tora!* a dramatization of the Japanese attack on Pearl Harbor, in 1941, Japanese Admiral Isoroku Yamamoto made the following statement, “I fear all we have done is to awaken a sleeping giant, and fill him with terrible resolve” (Williams & Fleischer, 1970). Compare the previous thoughts with software designed technologies that consolidate ecosystems into usable independence. Standalone intelligence agencies now feel the need to share information, communicate, and report (Roberts, 2004). Even though police, fire, and medical exist as separate functions, the efforts are unified when the information provided is the same, and is communicated simultaneously. With the ability to access automated and centralized intelligence, time gives way to thoughts that never had space so develop. Security, audit, and compliance give way to assurance. Development, Security, and Operations as independent thought factories give way to DevSecOps.

Security, audit, and compliance failure has given rise to increased data breaches and a call for revolution. Revolution implies thought. Thought implies action. Action implies results. For revolution to exist, agreement must be founded on some core strategy based on a desire that requires fulfillment (Gompert, Kugler, & Libicki, 1999). The results of security, audit, and compliance cannot be silo initiatives written up as an independent third party review acted upon separately, but are required to be a holistic approach meeting the needs or requirements of the business and regulation. Failure of security, audit, and compliance have given way to the revolution of risk based assurance (Ashley, 2014).

Lenny Rollison, lennyrollison@hotmail.com

How is risk based assurance determined or measured? “Audit is quality control” (Cole & Tarala, 2015). If audit is to validate the quality or state, there must be an underlying framework, standard, or best practice that is believed to accomplish the level of quality to be validated and measured. The historical approach to audit does not have its foundations in security, but “accounting, fraud, and compliance” (Cole & Tarala, 2015). Assumptions do not create reality any more than false, or unclear expectations create results. If there is no understanding of how data breaches occur or why, there can be no revolution. Attackers as the enemy, or offense, are rising in expertise, funding, and capabilities (Taylor, 2016). Corporations and small business, or defense, must rise to the occasion as do security, audit, and compliance. Solutions of thought must be delivered to reign in the anarchy. The solutions must deliver repeatable results based on known success. The offense must understand the defense, and overcome the risk based exceptions.

Frameworks, controls, other standards, and affiliated international projects are becoming a computerized consolidation of diverse data delivering continuous automated risk based scoring (NIST, 2015). Risk based scoring delivers an agreed upon consensus of a threat status, and if you are winning or losing. Using colors such as red, green, and yellow are generally understood as bad, good, and cautionary.

Deconstructing silos, consolidating frameworks or standards, and unifying tools, is not easily delivered by the masses of the information technology community today, either because of cost or complexity. In order to create this ecosystem of balance, unfortunately, one tool does not exist to deliver a response to the revolution for developers, infrastructure, or security. In fact, the plethora of applications seem no different than the online Google Play or Apple Stores in regards to choice. No one tool solves efficiency requirements of security initiatives from start to finish (“CIS Critical Security Controls: Solutions Directory,” n.d.). Because of this risk, base metrics are not elevated due to language barriers similar to patients and doctors, security professionals and executives (Cole & Hoelzer, 2012). In response to the language barrier and tool deficiency, tools or solutions are paired to meet a need or regulatory framework, free or at a cost. Pairing of tools, products, and solutions and their use, or process, create a methodology. These processes, or methodologies, often become a paradigm that changes the face of industry such as the Ford assembly

line during the industrial revolution. Influenced and "inspired by" canneries and meat-packing plants that utilized *continuous-flow* production methods, the time to create an automobile was reduced from "more than twelve hours to two hours and thirty minutes" (Ford's assembly, 2009). Ford knew that his purpose of delivering an affordable automobile to every American was in utilizing continuous-flow production mechanisms, automation. Create an information security assembly line by combining multiple continuous-flow tools, or small ecosystems, and achieve economies of scale, and the beginning of an industry paradigm and information technology revolution.

1.2. Response to the Call for Action

The Council on Cybersecurity houses the most current Critical Security Controls whose efforts are, "thwarting government and private sector daily compromise, tearing down failing traditional security, and prioritizing effective defense measures" (Cole & Tarala, 2015). The controls are built to defend against "initial compromise, long term access, and the ability to cause damage" (Cole & Tarala). The thought revolution, or guiding principles, in designing the controls are core components of philosophy abbreviated as: focus on damage control, consistent controls across the enterprise, automation with periodic measurement, consistent defense, effective and timely root cause analysis, and "the ability to measure the effectiveness while providing a common language to communicate about risk" (Cole & Tarala).

Currently, the CSC, Critical Security Controls consist of twenty prioritized controls "based on the NSA attack mitigation scores" and grouped into three categories: system, network, and application (Cole & Tarala, 2015). Each control must be based on an actual attack or it cannot be considered a control. The recommendation as a starting point is to focus on the top five of the twenty parent controls:

1. "Inventory of Authorized and Unauthorized Devices
2. Inventory of Authorized and Unauthorized Software
3. Secure Configurations for Hardware and Software
4. Continuous Vulnerability Assessment and Remediation
5. Controlled Use of Administrative Privileges" (Cole & Tarala).

Affiliated projects display the amount of consolidated encyclopedia of efforts around collaboration and the generation of controls: “SCAP, Security Content Automation Protocol, the US Department of State iPost, and the Australian Department of Defense Top 35 Mitigation Strategies, or Australian DSD Top 35.” (Cole & Tarala, 2015). “SCAP compromises specifications for organizing and expressing security-related information in standardized ways, as well as related referenced data such as unique identifiers for vulnerabilities” (The Security Content, 2016). The US DoS iPost web site defines iPost as, “iPost is the custom application that continuously monitors and reports risk on the IT infrastructure at the Department of State (DoS)” (U.S. Department of State, 2010). The Australian DSD Top 35 are also prioritized based on analysis of reported incidents and boast that if companies were to implement all of them 85% of attacks would be mitigated. (Australian Signals Directorate, 2013). Four controls are identified as priority controls:

1. “Application whitelisting of permitted/trusted programs, to prevent execution of malicious or unapproved programs including .DLL files e.g. using Microsoft AppLocker.
2. Patch application e.g. PDF Viewer, Flash Player, Microsoft Office and Java. Patch or mitigate “extreme risks” vulnerabilities within two days.
3. Patch operating system vulnerabilities. Patch or mitigate “extreme vulnerabilities” within two days. Avoid continuing to use Microsoft Windows XP or earlier versions.
4. Minimise the number of users with domain or local administrator privileges. Such users should use a separate unprivileged account for e-mail and web browsing” (Australian Signals Directorate).

The CSC and Australian DSD Top 35 provide compelling evidence of where a security program should be focusing based on evidence of reduction in attack vectors and long term compromise. Based on the first control from the CSC, Inventory of Authorized and Unauthorized Devices, the following software based solutions enable companies to facilitate a paradigm, culture change, and revolution. The solutions will deliver a single point of reference for physical and virtual inventory, configuration and change management, as well as reporting and visibility for enterprise governance. A system of reference is key to any security, audit and compliance program that will be automated and measured

against regulatory compliance. In the whitepaper, Comparing IT Security Standards, (Kuligowski, 2009), inventory as a base component is required of them all. Failure does not have to apply to security, audit, and compliance; risk based assurance is possible.

2. World Class Software Designed Solutions

2.1. DCIM, Data Center Infrastructure Management

Available from resellers or direct, DCIM software designed data centers start with: asset inventory, capacity management, and energy management. This is a great start to advance the intelligence of a dumb rack. Why monitor only the contents of the rack, when you can model and monitor an entire data center based on use and implementation. Now that more companies collocate verses owning the data center, knowledge of the data center collocation is no longer restricted to the proficiency and expertise of the collocation provider. DCIM facilitates documentation and process from the power source to the rack in single panes of glass providing enterprise level dashboards that easily showcase personalized data center intelligence (Nlyte, 2016).

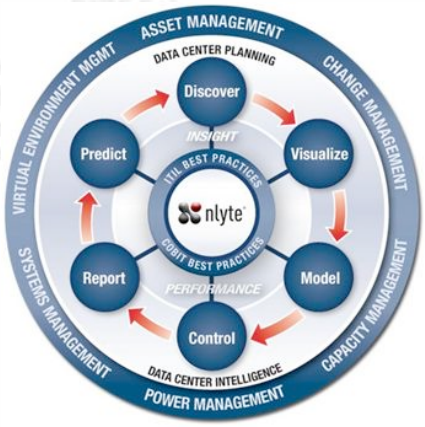

Figure 1: Data Center Planning and Intelligence Lifecycle

Companies utilize collocation facilities, or 3rd party data centers, to achieve high availability; however, high availability can be short circuited as recorded in a Cloud Service research paper, “the worst, most sustained downtime has always been caused by power issues” (Li, Z., Liang, M., O’Brien, L., & Zhang, H, 2013). Poor documentation and understanding of single points of failure are not limited to servers, but also heating, cooling, and network. Power outages are the most

visible availability failures (Li, Z., Liang, M., O'Brien, L., & Zhang, H). MRC, monthly recurring charges, provide clarity or error for what is being purchased and should be reviewed on a periodic basis.

Facilities management solutions once reserved only for large data centers have become cost effective enough to become available to collocation customers who self-manage their own infrastructure. Lack of planning, not knowing implications to equipment purchase, and human errors cause repeatable failures (Bigelow, 2011). As servers are delivered for racking and stacking, the asset management and change control process is already underway (Nlyte, 2016).

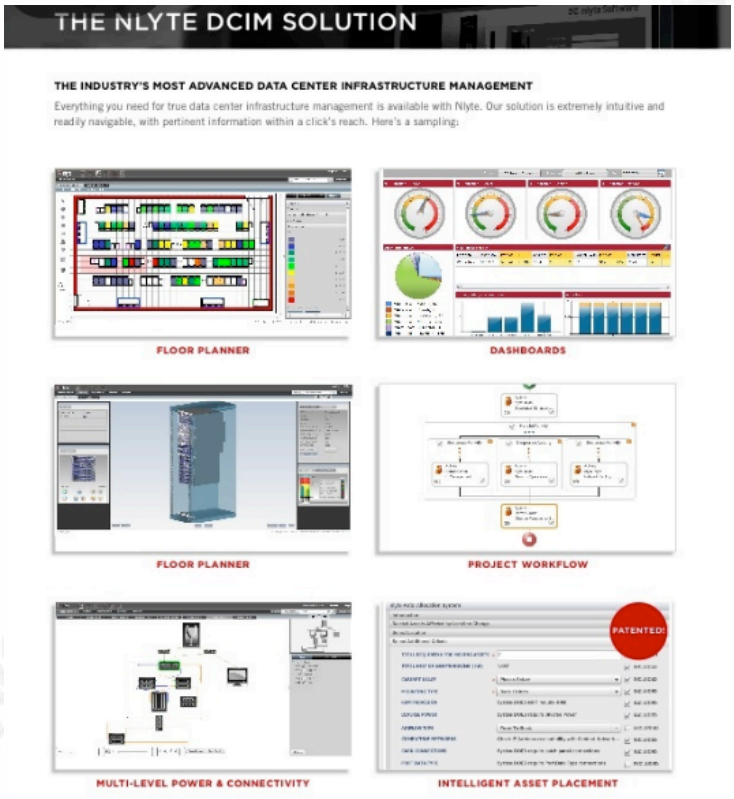

Figure 2: A DCIM Solutions High Level Overview

Some collocation data centers do have office space that is primarily reserved for temporary use, on-site implementations, or maintenance. Because collocation data centers are not corporate offices, space, power, and capacity planning may be something of an afterthought, and a process that occurs at the time of racking the server. After returning to the office a technician may take a packing slip or hand drawn rendering and transfer the information to a spreadsheet. Spreadsheets are often used as a manual inventory system, and

to generate renderings for space, power, and capacity. The spreadsheet drawings often contain data specific for inventory and operational use: location, server height, name of the server, model number. This often happens after the installation process in order to facilitate faster service delivery or emergencies. Unfortunately, the static spreadsheets are often overwritten, outdated, and noted as the root cause of misinformation and outages. Loss of inventory could easily occur if one employee is the single authority for updating and maintaining the inventory. DCIM removes single point of failure risks, provides the ability to trend hardware lifecycle, and can be made readily available from a web site. The CDMB, configuration database management system, of DCIM can also be synchronized with other products to complete the physical and virtual inventory.

Virtual machines hop from one physical host to another for performance, fault tolerance, or higher availability. Many DCIM solutions provide connectors that synchronizes the virtualization database with the DCIM database, and displays which virtual hosts are on what physical host at any given time. Though virtual server inventory can be scripted or manually exported, this is a point-in-time inventory that cannot be viewed live and in real-time.

Adding company, line of business descriptions, or organizational data as attributes, is also a point-in-time manual process that must be re-created on each export. Even with many flow diagram software programs equipment can be depicted but the data may not be synchronized with a database or viewed as living infrastructure. The equipment blocks may have attributes, but they only exist inside the drawing. With DCIM, organization of assets becomes a one-time data entry with the ability to track and add attributes as necessary. Multi-tenant shared hosting environments can rely on virtual cluster groupings, and set limits on movement of virtual machines; however, visibility to entire farms of servers landing what is on one virtual host is more difficult to predict and is much more clear when impact analysis can be reviewed in living software. Intelligence at the rack layer through the logical layer provides a software designed solution and expands the ability to adequately plan inventory as it relates to space, power, and capacity availability, for the full lifecycle of the asset. Nlyte DCIM offers this technology for virtualization synchronization (Nlyte, 2016).

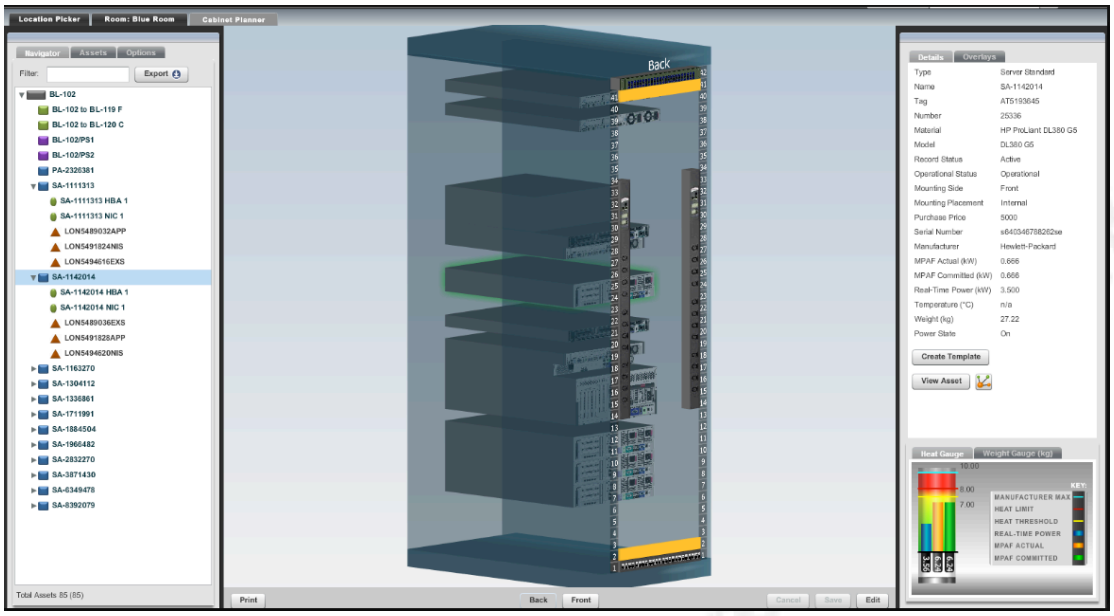

Figure 3: Automated Rack Elevation and Asset Management

If inventory is an afterthought, buy the server, rack the server, and deliver service to meet the project deadline, this implies no sales forecast, commonly referred to as sales verses engineering, and often a customer promised deadline. Operations is usually the last to hear of the deadline, but must communicate there is no longer any rack space or power. Another fault of spreadsheet inventory, is maintaining the hardware lifecycle that tracks only a serial number. Support and warranty if only exported based on serial number will waste thousands of dollars annually if not measured against a current state of use. Four-hour replacement parts are much more expensive than next business day. As ludicrous as what was just described sounds, it happens every day, and is common in small to medium size companies. Some companies will have good process and no tools, and others will have great tools and bad process.

Spreadsheet addiction is a “technology accepted model” (Spreadsheet Addiction, n.d.). Burns Statistics codifies spreadsheet addiction, “Addiction is the persistent use of a substance where that use is detrimental to the user” (Spreadsheet Addiction, n.d.). Burns continues to say, “The perception of the ease-of-use of spreadsheets is to some extent an illusion. It is dead easy to get an answer from a spreadsheet, however, it is not necessarily easy to get the right answer.” Automated rack drawings are a reality. Rack drawings in spreadsheets is a reality. Notifications from a server being removed from a rack is a reality. Rack intelligence living inside a software designed solution is far superior to a

dumb rack that only lives in a spreadsheet. Migrating business process from spreadsheets to DCIM, is a mark of maturity (Nlyte 2016).

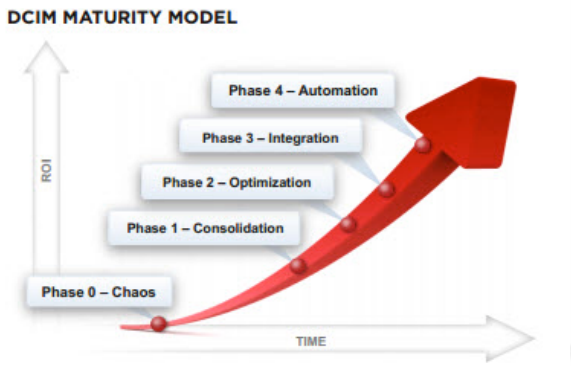

Figure 4: DCIM Maturity Model

Neither DCIM, nor spreadsheet, offer solution to where is my server, if there is no dedicated resource to fulfill the role or responsibility. Process based automation reduces the guesswork, and provides documentation of workflow management and change control as the alternative. If drawings to represent infrastructure environments are going to be created, the one-time entry should be facilitated in software designed tools that delivers and maintains a configuration database management system, synchronizes with other systems, and automates a holistic picture of the documented infrastructure.

2.2. Configuration Management

2.2.1. Reference Architecture

A small ripple in the wave of the future, coining the phrase “infrastructure-as-code”, configuration management is changing the pace of service delivery by removing manual tasks, configuration drift, and non-compliance. Configuration management fits into the overall SDLC, software development lifecycle, as a small tool across the visible landscape and is called a pipeline. Consider the entire application change control pipeline to circumvent any audit and compliance failures, and be prepared to review more than one tool in order to deliver seamless automation. The following ecosystem graphic from Profitbricks that depicts solutions for a full reference architecture (Toll, 2015).

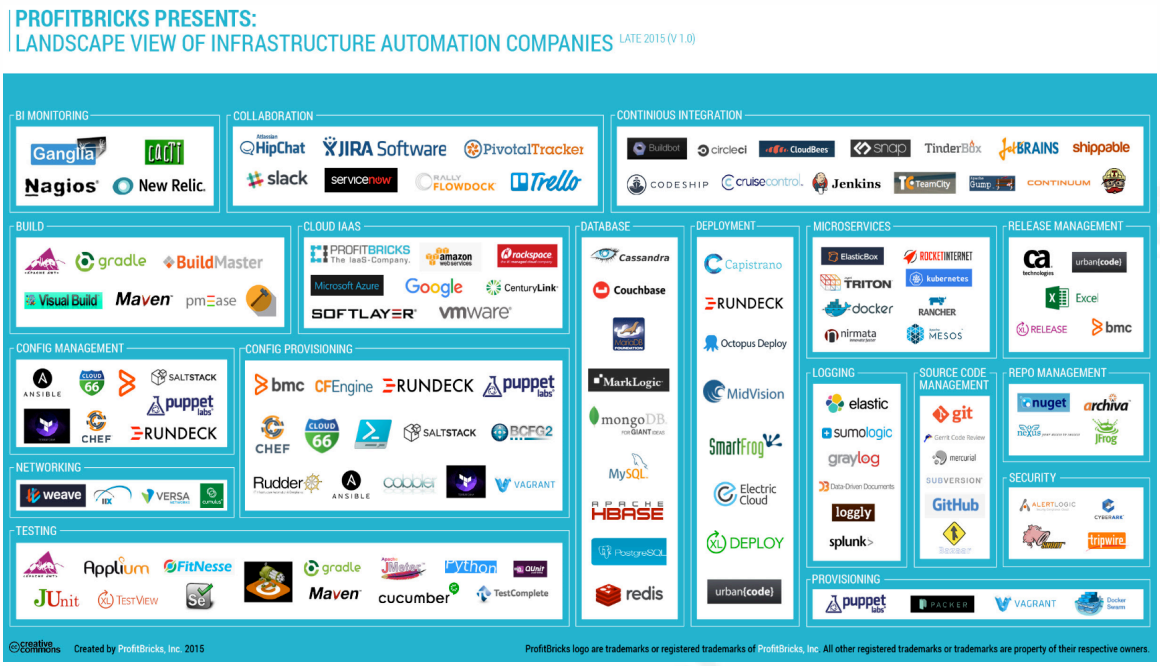

Figure 5: Landscape of Infrastructure Automation Companies

Delivery and configuration management can be manual or automated; however, automation as a guiding principle produces reduced time to market, cost benefit, and faster customer feedback (Chickowski, 2014). Configuration Management, CM, is precluded by an existing inventory as is Continuous Delivery, and Continuous Deployment. Continuous Delivery is the ability to deliver infrastructure nodes holistically from one environment, say Amazon Web Services, to another, Azure. This ability makes Continuous Delivery, CD, compelling; think of disaster recovery, or simply small environments for testing. Continuous Deployment is the ability to deploy change, few or thousands, continuously in all stages of the application change control pipeline up through production by automation. CM and CD are considered subcomponents of the information technology paradigm known as DevOps, a continuous improvement concept (Claps, 2015).

Lenny Rollison, lennyrollison@hotmail.com

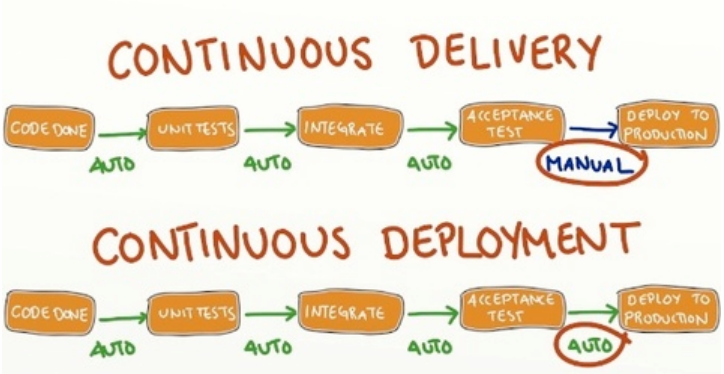

Figure 6: Comparing Continuous Delivery and Continuous Deployment

"The term "DevOps" typically refers to the emerging professional movement that advocates a collaborative working relationship between Development and IT Operations, resulting in the fast flow of planned work (i.e. high deploy rates), while simultaneously increasing the reliability, stability, resilience and security of the production environment. – Gene Kim" (Mas, 2014). DevOps is also the divorce of repetitive waste driven by codifying solutions that like relay teams pass the baton smoothly from one team member to the next, but automated, and with proven tools that energize economies of scale. DevSecOps was coined as a phrase soon after DevOps, and emphasizes the importance of security as an enabler, not a solution blocker (Leitz & Kennedy, 2015). The need for security to exist in the sprint cycle of agile development also become apparent. If the scrum cycle is two weeks or sixty days, automated security tools must exist inside the cycles. Security teams will be asked to evaluate the latest DevOps technologies. Transforming an audit or security team outside of the standard SDLC process will be a daunting task. Audit, security, or compliance should not be a bolt-on to development or operation lifecycles (Leitz & Kennedy).

Figure 7: What DevSecOps is not

As an example, functional application testing is required; however, security testing may not be. As companies rush to implement integration models,

the early phases may not contain a security approach in the SDLC process (Continuous Delivery and DevOPs, 2016).

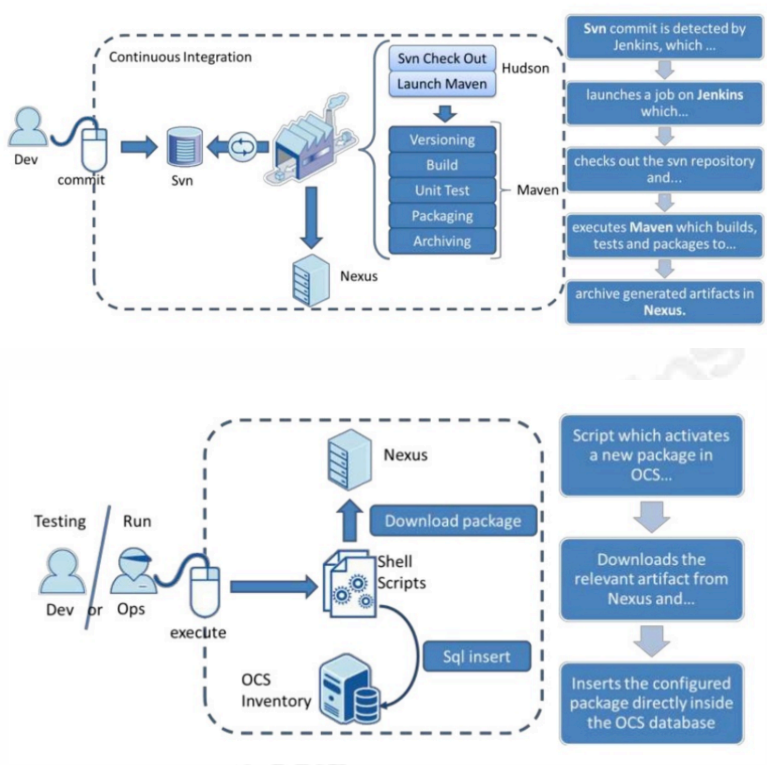

Figure 8: No Security Testing.

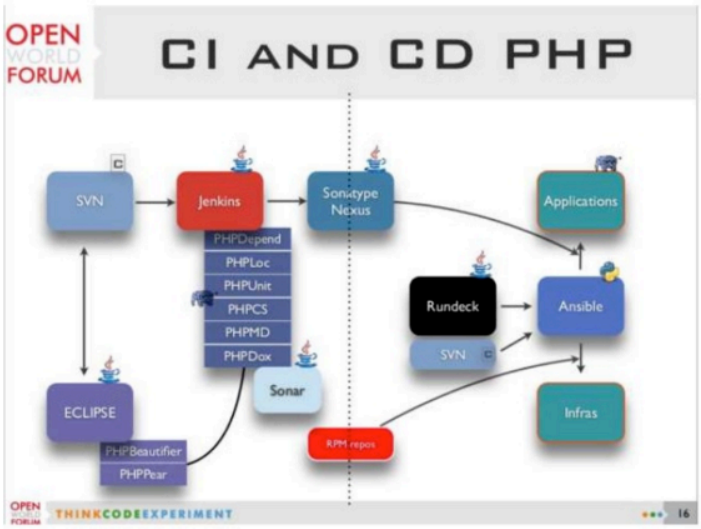

Figure 9: Security Testing.

From the plethora of applications available for DevSecOps, beware of security evaluation fatigue as teams either search for better tools, or bounce from one to another. Clogeny provides an example of where the tools can fit together

to complete an application change control cycle (Devops and Infrastructure, 2015).

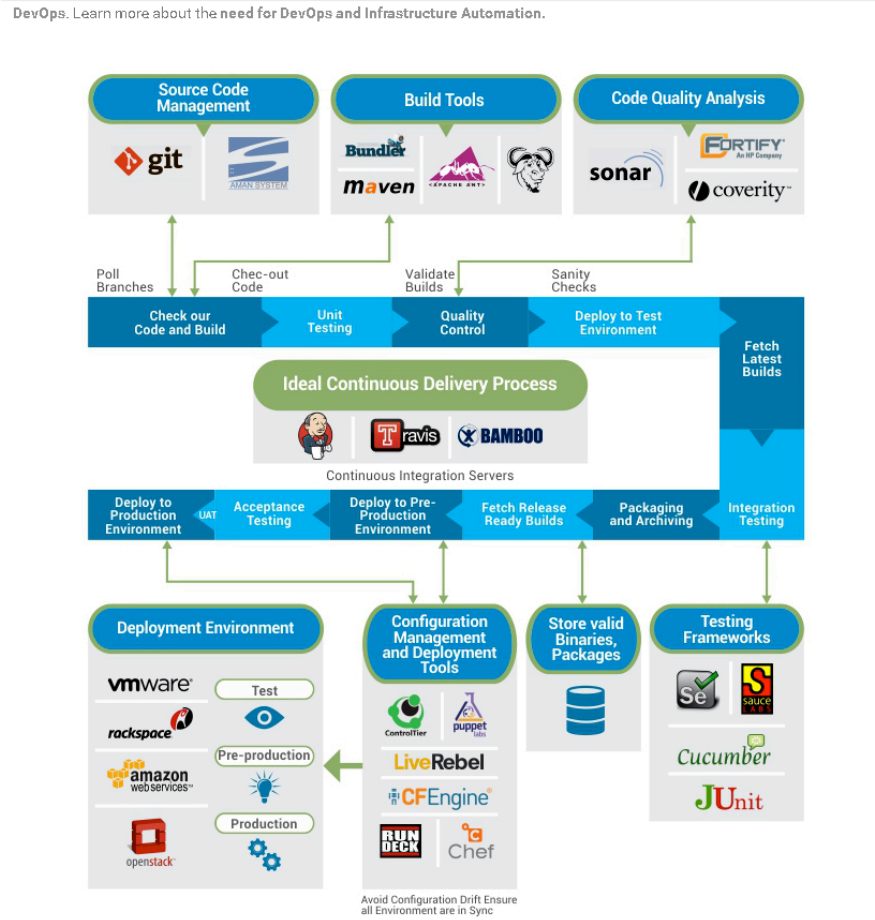

Figure 10: Devops and Infrastructure Automation

It is clear that after software deployment, traditional security challenges still exist, as no other security tools are referenced in the architectures. Many production facilities are commonly expected to brunt traditional minimum defense in depth posture: having only deployed firewalls, operating system patches, anti-virus, and vulnerability scans. Too much reliance on production security is assumed, and risk is elevated. Use configuration management tools to measure your hardening standard holistically across all production environments, and then map out a remediation story supported by senior management.

2.2.2. Securing Configuration Management

Configuration management is a tool that can be used to automate infrastructure, and like any tool, must be secured. For defining an audit posture,

the compliance outline consists of “internal corporate policies and procedures, government regulations, and industry best practices” (Cole & Tarala, 2015). This conglomerate of policy, procedures, and standards aids in defining required security standards at a company, and can be a great line of demarcation for defining the security requirements if none exist.

In order to communicate compliance with corporate policy or government regulation, documentation will be required as well as some visual diagrams. Product or service overview can be provided in an at-a-glance conceptual design delivered in a simplified technical specification. Forcing a product team, owner, or developer, to use technical writing skills is the beginning of the collaborative and technical inventory process.

Barring any snowflakes, operating systems do not have to be limited to Windows or Open Source, however, the reality is a high percentage are. Because of this, the security measures, or defense in depth, can be applied to any process or solution regardless of familiarity, or lack thereof. One example is PCI-DSS, the Payment Card Industry Data Security Standard, and the twelve requirements that are mandated on any system that stores, transmits, or processes credit card data. Tier I systems require all twelve requirements. Tier II systems, or systems that answer outbound calls only from Tier I segmented networks, limit the scope of the twelve requirements to as little as four. Tier III systems that have no inbound or outbound calls to Tier II or Tier I systems are considered out of scope. Classifying products and services based on a business model, service level agreements, and regulatory compliance, is an easy way to determine what security is already in place, and where the bar can be raised for all products or solutions. Defining this threshold may even deliver a risk metric based on what audit, security, and compliance controls are or are not required to be in place, and potentially where applications should be placed in applicable environments. These environments set the stage for communicating requirements and setting expectations.

After consulting policy, procedure, regulation, and service delivery agreements, defense in depth is the next logical scoping mechanism. Separation of environments, or segmentation, either logical or physical, and any interconnectivity between sensitive environments is often a driver for security (Official PCI Security Standards, 2016).

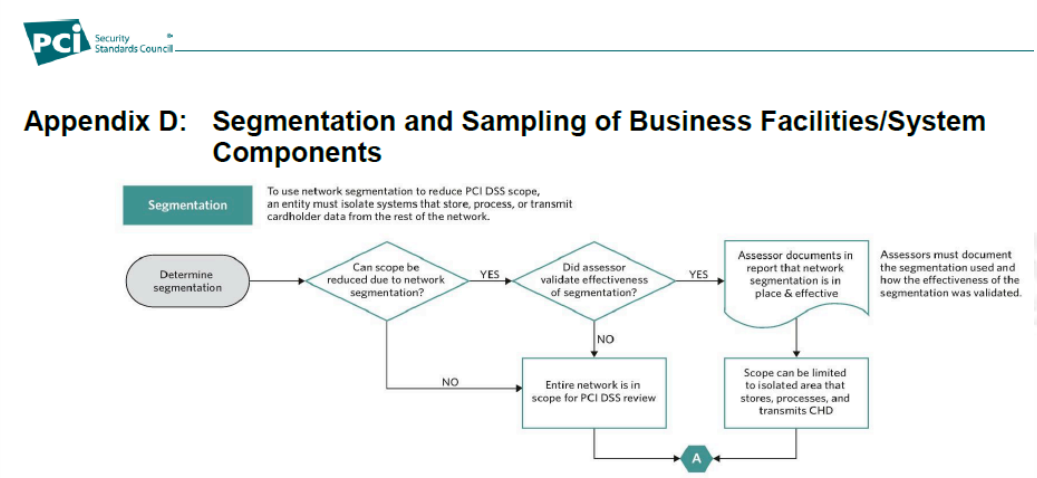

Figure 11: PCI-DSS Scoping and Segmentation

After defining logical and physical segmentation, access control and authentication management are required next steps. A security conundrum for many configuration management systems begins in this phase for network segmentation, the application change control pipeline, and separation of duties. Without due diligence and security process, it may appear that code will simply be delivered straight from the developer's workstation from a corporate environment to production (Robbins, 2012).

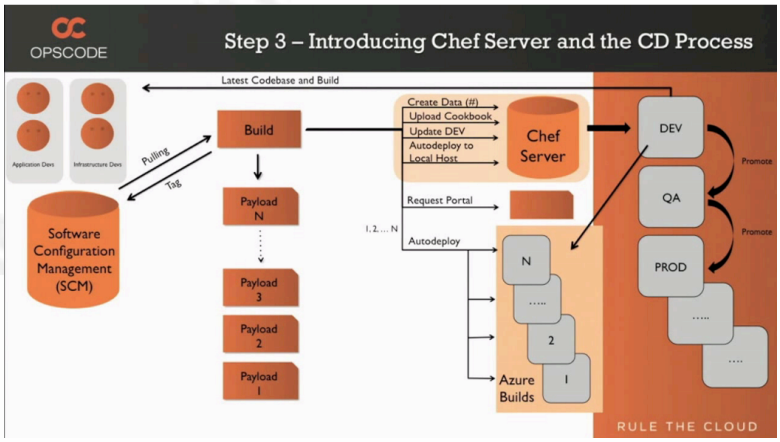

Figure 12: Flow Diagram between DEV, QA, Production

In reality code does get delivered from development straight to production; however, the physical files themselves have a number of checkpoints along the way, or milestones, before becoming a production artifact. Segmentation of environments may not prevent all security incidents, but governance and compliance still require as a best practice separation of environments. This

regulated, mandatory separation of environments can be compared to being schizophrenic when having to communicate this paradox. Supporting the DevOps movement, or DevSecOps movement, still requires a segregated production environment, that is by standard not operated by developers (Mas, 2014).

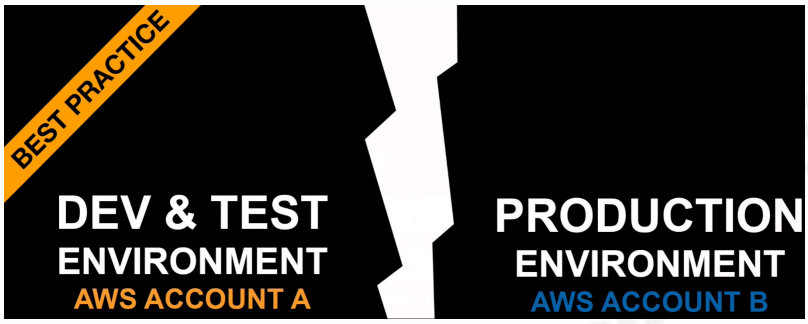

Figure 13: Separation of Duties & Environments

Two things must now be considered before jumping the chasm between development and the service level agreement production environment: the package integrity, and getting the file into the environment. Securing the application change control process in application development now becomes a separate SDLC initiative than securing the production deployment process, which usually requires different tools altogether, either for deployment automation, or security automation. Depending on the size of companies, their teams, their expertise, and the regulation of the products involved, the number of environments required to maintain either a policy driven security posture, or regulatory compliance may differ dramatically. As an example, developers will not be maintaining the automation environment, only the product being automated. There may be one team managing the application deployment pipeline, and another team managing the production pipeline, as well as different automation tools used altogether. Planning for the number of environments to be managed, and by whom, drives a culture shift by adding just one tool, configuration management.

Transferring artifacts, or software deployment packages, is not the responsibility or the function of configuration management tools. Many development teams or integrators may not understand this deployment requirement. Some CM tools will not transfer large files very well; there must be a mechanism for doing so. Nexus is a common artifact repository that is commonly used for transferring or replicating artifacts. Think of this as

transferring any package securely, whether it is a credit card or a deployment file, and hopefully not malware from a corporate environment. When the files land, there should be security check, such as an anti-virus scan and hash validation. These transfers should be made over secure protocols utilizing SSL certificates from well-known certificate authorities, and at each handoff between tools or environments. The private keys should be secured with security teams, secured on the server and within the operating system, and not shared between development or production environments.

Managing SSH keys, or secrets, becomes another challenge of managing users of the configuration management tool as well as the CM tool. Because the majority of these tools are open source, SSH is the chosen authentication mechanism. Allowing direct SSH connections from a corporate environment to production is not recommended.

Chef Server, a configuration management tool, utilizes cookbooks and recipes as infrastructure-as-code. Cookbooks and recipes are available on a public supermarket, but a private supermarket can be configured to maintain a secure, approved cookbook repository as a standard set of deployment configurations. For accessing the private supermarket, there are a few pre-requisites that are security related: Chef Server using Chef Oath Provider, an administrative password, and a key for the private account. The private supermarket encryption mechanisms should be maintained by the operations security team, and not developers or server operators. Chef utilizes data bags for storing data, including passwords. The data bags can be encrypted but the private keys are not secured and will be made available everywhere the cookbook is published. To lock this down, Chef Vault, developed by Nordstrom's, can be utilized to "encrypt Chef Data Bag Items using the public keys of a list of chef nodes. This allows only chef nodes to decrypt the encrypted values" (Webber, 2016). Use Chef Vault to reduce the exposure of secrets. HashiCorp's Vault can operationalize secrets separation between development and production. Vault is primarily used for managing secrets, tokens, passwords, certificates, and others adding security functionality that is lacking in many configuration management tools (Vault by HashiCorp, n.d.).

Chef can be used as a client-server, or in local mode. A workstation or management server is used to communicate with the Chef Server, the master server. The chef-clients are nodes to be configured. For full configuration

capabilities, chef-client is recommended to be run as local administrator on Windows, or with root privileges on open source platforms. Looking at this in a positive manner, some users or groups that have required administrative privileges can now be removed once Chef has been fully operationalized. Also, consider utilizing Group Managed Service accounts, gMSA for the chef-client service. Domains utilizing gMSA service accounts will have their passwords managed by the Windows 2012 domain controllers, and reduces administrative overhead; Windows 2008 and above domain members required (Symalla, 2012). If a host is compromised, consider the gMSA password compromised too (Symalla). For open sources servers consider integrating them into active directory, and utilize security groups instead of single user access.

Understanding the security impact of software being chosen and installed into an organization's environment should not be underestimated. A few due diligence steps may include, but may not be limited to: research the company's web site for a security page, sign up for any notifications related to security or product updates, frequent the blogs, check for CVE vulnerabilities by version, and look for any Metaspolit modules released specific to the proposed tool. While testing, vulnerability scan or pen you're your test environment and require the vendor to respond accordingly. Though DevSecOps is a revolution, utilize security best practices to configure and measure the success of any tool deployment, not just configuration management.

2.2.3. Continuous Monitoring

After automating the configuration of your infrastructure and securing Chef Server, Chef Compliance can be used to audit the environment, and can be tailored to measure against security best practices. Reports can be generated against CIS, Center for Internet Security, Benchmarks shipped as profiles, or a "collection of rules for a particular security framework" (Webber, 2016). Validate compliance; then remediate. Not every CIS benchmark has been included. Due to the newness of the Chef Compliance product this may be a small challenge, but a great learning opportunity for not only familiarizing oneself with the CIS Benchmarks, but also a new tool. Credentialed scans can be run utilizing SSH or WinRM directly against the environment, or an audit cookbook can be used to report to Chef Server, and then to Chef Compliance for report generation. This task can be coordinated between

operations, security and development, and from some blogs the difficulty level appears to be minimal (Hedgpeth, 2016).

As hardening standards provide security assurance, attackers never rest, and threats continue to arise as systems become older. If CIS benchmarks are not already utilized, this one feature offers a compliance check against any environment which can then be remediated utilizing Chef Server. Chef Compliance requires implementation, another server, and a resource to manage the tool.

3. Conclusion

Just having an automated inventory system does not make the inventory visible or collaboratively available to other tools that depend on the inventory. Just having an automated delivery pipeline or collaborative DevSecOps environment, doesn't provide enterprise governance visibility for security, audit, or compliance controls. "Business as usual" posture is as a new mandatory requirement of Payment Card Industry PCI-DSS v3.2 (Official PCI Security Standards, 2016). In order to maintain a continuous-flow inventory, service delivery model, and portal for governance visibility, standalone tools are provided as solutions in the cloud to administer collaboration.

Service Now, an ITSM, information technology service management, SaaS offerings provides plugins for DCIM and configuration management solutions, and have built in pre-configured modules for audit and compliance. These combined service offerings provide a unique blend of visibility from purchase and deployment, change control of application delivery, and measuring in one portal for security, audit, and compliance.

Vulnerability scans mandated for compliance, or simply for board of director visibility, can be uploaded on a monthly or quarterly basis from a SCAP compliant scanner, and delivered in line graphs. CSV vulnerability scan files already having been uploaded, can be tied to audit workflow for PCI-DSS. Once the audit cycle has arrived, the audit module can populate a current percentage of audit completion as easily as the security incident pie chart below (Service Now, 2016).

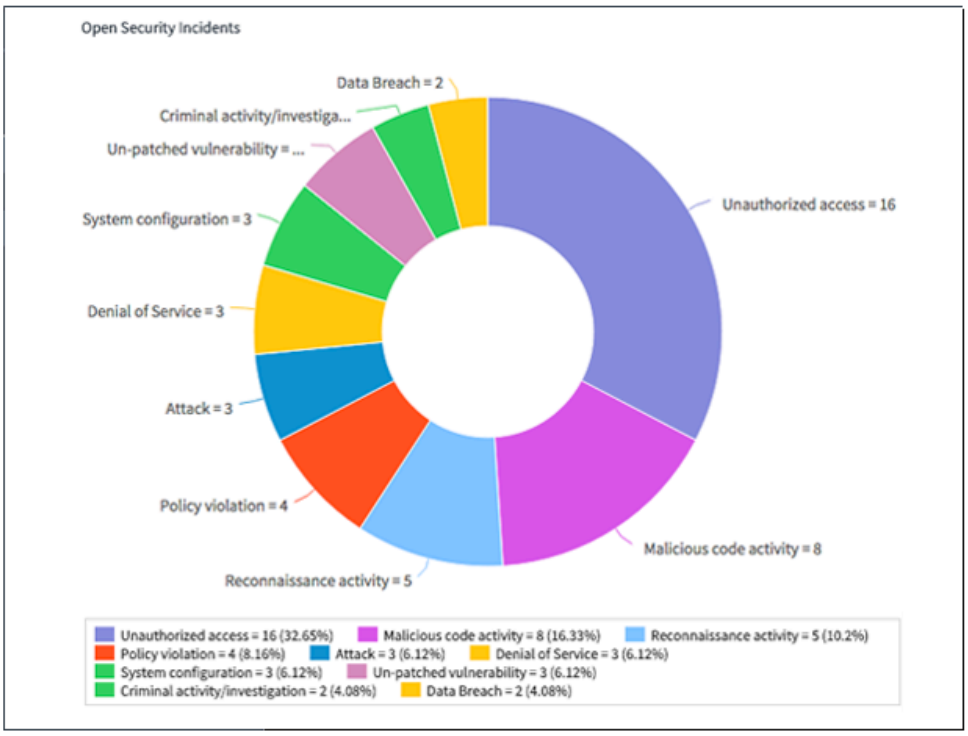

Figure 14: Security Incident Pie Chart from Service Now

SaaS, software as a service, is delivering subscription costs savings and solutions to businesses like collocation does to companies owning their own data center; the cost benefit is in outsourcing the data center. Buying and hosting individual tools does not provide economies of scale across a global organization; however, the ecosystem of tools can be provided at subscription costs to both small and large companies at tiered pricing levels, hosted in the cloud, and visible as a governance mechanism. Cloud based software designed solutions consolidate audit, security and compliance data in one solution as integrated ecosystems, creating a new paradigm for information security.

References

Ashley. (2014, November 21). The Future of IT Security and Compliance Program Management? It's in the Cloud… Retrieved April 10, 2016 from https://www.tracesecurity.com/blog/the-future-of-it-security-and-compliance-program-management-its-in-the-clou#.VzasPvkrJga

Australian Signals Directorate. (2013, April). Retrieved April 30, 2016, from http://www.asd.gov.au/infosec/mitigationstrategies.htm

Bigelow, S. (2011, June). The causes and costs of data center system downtime: Advisory Board Q&A. Retrieved May 7m 2016, from http://searchdatacenter.techtarget.com/feature/The-causes-and-costs-of-data-center-system-downtime-Advisory-Board-QA

Chickowski, E. (2014, May 6). DevOps and faster feedback: Fewer problems, better features (part1) – DevOps.com. Retrieved May 16, 2016, from http://devops.com/2014/05/06/faster-feedback/

CIS Critical Security Controls: Solutions Directory. (n.d.). Retrieved April 23, 2016, from https://sans.org/critical-security-controls/vendors-solutions

Claps, G. (2015, August 12). The "Pragmatic" Agile Lean Practitioner. Retrieved April 30, 2016, from https://www.quora.com/What-is-Continuous-Delivery

Cole, E. & Hoelzer, D. (2012) *Technical Communication and Presentation Skills for Security Professionals.* The SANS Institute.

Cole, E. & Tarala, J. (2015) *Implementing & Auditing the Critical Security Controls – In Depth.* The SANS Institute.

Conner, D. (1993). *Managing at the speed of change: How resilient managers succeed and prosper where others fail.* New York: Villard Books.

Continuous Delivery and DevOps: SlideShow Gallery. (2016, April 12). Retrieved April 30, 2016, from

https://www.sonatype.org/nexus/continuous-delivery-and-devops-slideshow-gallery/

Devops and Infrastructure Automation, Devops Automation. (2015). Retrieved April 30, 2016, from http://clogeny.com/devops-and-infrastructure-automation/

Ford's assembly line starts rolling. (2009). Retrieved April 23, 2016, from https://www.history.com/this-day-in-history/fords-assebly-line-starts-rolling

Gompert, D.C., Kugler, R.L., & Libicki, M.C. (1999). *Mind the gap promoting a transatlantic revolution in military affairs.* Washington, D.C.: National Defense University Press.

Hedgpeth, A. (2016, May 04). Tutorial for Setting Up Chef Compliance Server on Azure. Retrieved May 7, 2016, from http://www.anniehedgie.com/setting-up-compliance

Johnston, D. (2003, July 24). 9/11 Congressional Report Faults F.B.I.-C.I.A. Lapses. Retrieved April 10, 2016, from http://www.nytimes.com/2003/07/24/us/9-11-congressional-report-faults-fbi-cia-lapses.html

Kuligowski, C. (2009). *Comparison of IT Security Standards* [White Paper]. Retrieved April 30, 2016, from http://federalcybersecurity.org/CourseFiles/Whitepapers/ISOvNIST.pdf

Leitz, S., & Kennedy, S. (2015, April 20). Enterprise Cloud Security via DevSecOps | USA 2015 | RSA Conference. Retrieved April 30, 2016, from https://www.rsaconference.com/events/us15/agenda/sessions/1686/enterprise-cloud-security-via-devsecops#sthash.1MZmiBmb.dpdf

Li, Z., Liang, M., O'Brien, L., & Zhang, H. (2013, December 23). ArXiv.org. Retrieved May 7, 2016, from http://arxiv.org/pdf/1312.6485.pdf

Mas, I. (2014, August 7). DevOps for the Enterprise. Retrieved May 7, 2016, from https://aws.amazon.com/campaigns/emea-devops/

National Institute of Standards and Technology. (2015, September 10). Retrieved April 23, 2016, from http://csrc.nist.gov/news_events/cif_2015/security-automation/day2_security-automation_330-420.pdf

Nlyte Software. (2016). Retrieved April 30, 2016, from http://computer-room-design.com/tag/nlyte-software/

Official PCI Security Standards Council Site – Verify PCI Compliance, Download Data Security and Credit Card Security Standards. (2016, April). Retrieved May 7, 2016, from https://www.pcisecuritystandards.org/documents/PCI_DSS_v3-2.pdf

Official PCI Security Standards Council Site – Verify PCI Compliance, Download Data Security and Credit Card Security Standards. (2016, April). Retrieved May 7, 2016, from https://www.pcisecuritystandards.org/pdfs/PCIDSS.pdf

Robbins, J. (2012). Continuous Deployment & Delivery Culture Hacks @ QCON 2012. Retrieved January 14, 2015, from http://www.slideshare.net/jesserobbins/continuous-deployment-delivery/44

Roberts, J. (2004, May 18). Communication Breakdown on 9/11. Retrieved April 10, 2016, from http://www.cbsnews.com/news/communication-breakdown-on-9-11/

Security Operations | Enterprise SecOps | ServiceNow. (n.d.). Retrieved May 7, 2016, from http://www.servicenow.com/products/security-operations.html

Spreadsheet Addiction - Burns Statistics. (n.d.). Retrieved May 16, 2016, from http://www.burns-stat.com/documents/tutorial/spreadsheet-addiction/

Symalla, D. (2012, December 16). Windows Server 2012: Group Managed Service Accounts. Retrieved May 7, 2016, from https://blogs.technet.microsoft.com/askpfeplat/2012/12/16/windows-server-2012-group-managed-service-accounts/

Taylor, H. (2016, February 05). A $445B economic threat you aren't prepared for. Retrieved April 10, 2016, from http://www.cnbc.com/2016/02/05/an-insider-look-at-whats-driving-the-hacking-economy.html

The Security Content Automation Protocol (SCAP). (2009, May 12). Retrieved April 23, 2016, from https://scap.nist.gov/

Toll, W. (2015, December 11). Infrastructure Automation Ecosystem Landscape [infographic] | ProfitBricks Blog. Retrieved April 30,

2016, from https://blog/profitbricks.com/infrastructure-automation-ecosystem-landscape/

U.S. Department of State. (2010, May). Retrieved April 30, 2016, from https://findit.state.gove/search?query=ipost

Vault by HashiCorp. (n.d.). Retrieved May 7, 2016, from https://www.vaultproject.io/

Webber, F. (2016, April 21). Managing Secrets with Chef. Retrieved April 22, 2016, from https://chef.io/webinars/

Williams, E. (Producer), & Fleischer, R., Fukasaku, K., Masuda, T., (Directors). (1970). *Tora! Tora! Tora!* [Motion Picture]. Japan: Twentieth Century Fox Film Corporation

Williams, J. (2015, March). The Case for Visibility: SANS 2nd Annual Survey on the State of Endpoint Risk and Security. Retrieved February 13, 2016, from https://www.sans.org/reading-room/whitepapers/analyst/case-visibility-2nd-annual-survey-state-endpoint-risk-security-35927