Professional Expertise Distilled

Agile IT Security Implementation Methodology

Plan, develop, and execute your organization's robust agile security with IBM's Senior IT Specialist

Jeff Laskowski

[PACKT] PUBLISHING enterprise professional expertise distilled

Agile IT Security Implementation Methodology

Plan, develop, and execute your organization's robust agile security with IBM's Senior IT Specialist

Jeff Laskowski

[PACKT] PUBLISHING enterprise professional expertise distilled

BIRMINGHAM - MUMBAI

Agile IT Security Implementation Methodology

Copyright © 2011 Packt Publishing

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, without the prior written permission of the publisher, except in the case of brief quotations embedded in critical articles or reviews.

Every effort has been made in the preparation of this book to ensure the accuracy of the information presented. However, the information contained in this book is sold without warranty, either express or implied. Neither the author, nor Packt Publishing, and its dealers and distributors will be held liable for any damages caused or alleged to be caused directly or indirectly by this book.

Packt Publishing has endeavored to provide trademark information about all of the companies and products mentioned in this book by the appropriate use of capitals. However, Packt Publishing cannot guarantee the accuracy of this information.

First published: November 2011

Production Reference: 1151111

Published by Packt Publishing Ltd.
Livery Place
35 Livery Street
Birmingham, B3 2PB, UK.

ISBN 978-1-84968-570-2

www.packtpub.com

Cover Image by David Gimenez (bilbaorocker@yahoo.co.uk)

Credits

Author
Jeff Laskowski

Acquisition Editor
Rashmi Phadnis

Technical Editor
Ajay Shanker

Project Coordinator
Joel Goveya

Proofreader
Aaron Nash

Indexer
Tejal Daruwale

Graphics
Manu Joseph
Conidon Miranda

Production Coordinator
Prachali Bhiwandkar

Cover Work
Prachali Bhiwandkar

About the Author

Jeff Laskowski C|EH is a senior IT Specialist with IBM's Software group, the author of *Agile IT Security Implementation Methodologies*, and a freelance author for IBM Developer Works. His expertise in the area of software delivery and security spans more than a decade. During this tenure, Jeff was a principal consultant for application quality at Compuware. As such, Jeff enabled businesses around the globe to proactively integrate effective security practices into their organizations. Jeff joined the IBM team in 2006 and is now a Lead Engineer for the Great Lakes software business unit for Security.

Without all of you, this book would not have been possible.

First and foremost I would like to thank my family, especially my wife and daughters, Nicole, Chloe, and Chelsea Laskowski. Thank you so much for giving me the time to write this book. My brother, Andrew Lahser, for the initial inspiration and helping set the vision. To my father and mother, Michael Laskowski, and Andrea Soultanian, for the motivation to work on this book. My sister, Michele Laskowski, for creativeness.

I would like to thank a number of IBMers for the design of this book. To my bosses, Bradley Lewis and Joseph Noonen for your dedication, guidance and support in the publishing of this book. Sussan Vissar for your hard work connecting this book with Packt publishing. I would like to thank Dean Phillips, Susan Brule-Haefele, Kristin Lovejoy, Jeff Crume, Doug Lahsmit, Mary Ellen Zurko, Steven Bade, Calvin Powers, and everyone on the IBM SAB steering committee in the technical guidance, review, and feedback on this book.

Thank you to Packt publishing for making this book a reality. Thank you Rashmi Phadnis and Ajay Shanker for your time in publishing and editing of this book. Thank you Joel Goveya and James Lumsden for your help in organizing and publishing *Agile IT Security Implementation Methodology*.

---

www.PacktPub.com

Support files, eBooks, discount offers and more

You might want to visit `www.PacktPub.com` for support files and downloads related to your book.

Did you know that Packt offers eBook versions of every book published, with PDF and ePub files available? You can upgrade to the eBook version at `www.PacktPub.com` and as a print book customer, you are entitled to a discount on the eBook copy. Get in touch with us at `service@packtpub.com` for more details.

At `www.PacktPub.com`, you can also read a collection of free technical articles, sign up for a range of free newsletters and receive exclusive discounts and offers on Packt books and eBooks.

`http://PacktLib.PacktPub.com`

Do you need instant solutions to your IT questions? PacktLib is Packt's online digital book library. Here, you can access, read and search across Packt's entire library of books.

Why Subscribe?

- Fully searchable across every book published by Packt
- Copy and paste, print and bookmark content
- On demand and accessible via web browser

Free Access for Packt account holders

If you have an account with Packt at `www.PacktPub.com`, you can use this to access PacktLib today and view nine entirely free books. Simply use your login credentials for immediate access.

Instant Updates on New Packt Books

Get notified! Find out when new books are published by following `@PacktEnterprise` on Twitter, or the *Packt Enterprise* Facebook page.

Table of Contents

Preface

Agile methodologies attack risk at its core. To identify risk early and often is the premise of agile security. Risk comes in many forms. Organizations face security threats every day. Risks are in the delivery and maintenance of the countermeasures. Agile IT Security focuses on the details of the steps in delivering and maintaining IT security measures while keeping the big picture in mind. This allows us to see the forest for the trees and deliver more value in our day-to-day activities. Agile IT Security Implementation Methodology allows IT security to evolve from the practices that were established many years ago and teaches the fundamentals of evolving traditional approach to IT security into an Agile approach.

What this book covers

Chapter 1, *Why Agile IT Security?*, will introduce the need for Agile Security and the common problems related to IT security today.

Chapter 2, *New Security Threats*, will introduce the evolving threats that organizations encounter.

Chapter 3, Agile Security Team, will introduce the concept of an agile IT security team and the getting started approaches and the concepts related to initiating a new agile effort.

Chapter 4, *Agile Principles*, will discuss the approaches available to the agile IT security team. It's important to realize that not all approaches may be used by a team, but a combination of the approaches that best fit the project, organization, and team.

Chapter 5, *Agile Risk-Driven Security*, will look at the various ways we can use risk to understand an organization's security landscape.

Chapter 6, *Agile Blueprint*, will look at extending our bullpen with the use of agile blueprinting.

Chapter 7, *Lean Implementation Principles*, will cover principles that come directly from the lean manufacturing process that revolutionized the automotive industry, among other manufacturing industries. Lean manufacturing looks for ways to constantly improve the manufacturing process. Lean manufacturing principles align closely with IT security implementation.

Chapter 8, *Agile IT Security Governance and Policy*, will help to round out Agile IT security beyond security countermeasures implementation.

Chapter 9, *Security Policy and Agile Awareness Programs*, is intended to help security professionals educate the employees of an enterprise.

Chapter 10, *Impact on IT Security*, will cover the importance of agile structure as it relates to risk in an organization.

Chapter 11, *Barriers to Agile*, will discuss some key aspects to consider that will improve early success with Agile.

Chapter 12, *Agile Planning Techniques*, discusses additional planning techniques using Mind Mapping.

Chapter 13, *Compliance and Agile*, discusses the Agile impact of compliance.

Chapter 14, *Effective Agile IT Security*, reviews the process of Agile IT security implementation methodology.

What you need for this book

To best understand this book, you should have a basic understanding of basic information technology infrastructure, concepts, and design, and a basic understanding of how IT security fits into the general framework of IT.

Who this book is for

This book is designed for anyone who is curious about efficiencies in delivering IT security policies and countermeasures. This book focuses on teaching the fundamental methodologies of an agile approach to IT security. Its intent is to compare traditional IT security implementation approaches to new Agile methodologies. The intention of this book is to teach IT Security professionals the concepts and principles that IT development has been using for years to help minimize risk and work more efficiently. The book is targeted at the IT security management, director and architects, but is useful for anyone responsible for the deployment of IT security countermeasures. Security people with a strong knowledge of Agile software development will find this book a good review of agile concepts.

IT security, many years ago, was simply an extension of building security. We placed security guards at the entrances and exits and it was assumed that anyone inside the building had access to almost everything, including IT systems. Some IT systems included simple logons but, for the most part, additional security was unnecessary. The advent of the modem in the 1980s enabled black hats to start hacking. Some occasional hacking was previously committed against the phone company to try to make free phone calls. The '80s hackers were just geeky and mostly harmless, despite the fact that they did cause much harm. Back then, hackers would simply try to connect to an open modem and gain access to the remote system. Once in, a hacker wasn't burdened with modern-day firewalls, intruder detection systems, or honeypots. The hacker, who was often simply someone attempting to gain bragging rights on the local bulletin board system, had a fairly free reign over the system. However, the average hacker of today has become criminal. Through the early 1990s, it was difficult to find any legal stories in which the black hats experienced any serious consequences. In today's security landscape, we see litigation damages on a daily basis among the multiple organizations in which IT systems are involved.

My first dealings with Agile IT Security occurred when I was first assigned to a security implementation project in which I was asked to manage. This implementation centered on identity and access management. With a strong background in Agile software development, I knew how I wanted to run my team but I knew I wouldn't be able to find seasoned Agile security professionals. I didn't have any time to teach the members of my team the principles they would need; they would have to learn through attrition. I had three weeks to design, build, and deploy the solution.

Initially, my teammates were taken back with my loose style and openness. My style was called into question initially once or twice by the stakeholders, but I reassured them that my style is sound and most importantly it will produce results. Worst of all was the chaos; the environment was unstable and unmanaged. Network, system and security administration were difficult to find and schedule time with. Building this security solution was going to be tricky and difficult. Most stakeholders thought it couldn't be done and that I would be unsuccessful, but surprisingly I completed the project in three weeks. Three weeks ahead of schedule. Better than that was the respect I received from the people I managed and the stakeholders I supported. My visibility and stature sprung high after this and my credibility has never diminished. Agile software is ingrained inside my brain, second nature, and I apply the principles and practices without second thought. I have taken the principles and practices that I have learned and incorporated them into this book.

I think success can be measured in a number of ways. Success can be measured by your success in your organization and you can also measure your success from a personal standpoint. One of my coworkers I worked with long ago had extremely meager means. His family had little and he worked hard to put himself through school and get a good job. Simply being on the team was a bigger personal accomplishment than others on the team. Consider the success you have had in your professional career, consider your technical success, personal success, and your leadership success. Consider the areas you want to excel in and consider Agile IT Security as a way to help you succeed in all three areas of success.

Reader feedback

Feedback from our readers is always welcome. Let us know what you think about this book—what you liked or may have disliked. Reader feedback is important for us to develop titles that you really get the most out of.

To send us general feedback, simply send an e-mail to `feedback@packtpub.com`, and mention the book title via the subject of your message.

If there is a book that you need and would like to see us publish, please send us a note in the **SUGGEST A TITLE** form on `www.packtpub.com` or e-mail `suggest@packtpub.com`.

If there is a topic that you have expertise in and you are interested in either writing or contributing to a book, see our author guide on `www.packtpub.com/authors`.

Customer support

Now that you are the proud owner of a Packt book, we have a number of things to help you to get the most from your purchase.

Errata

Although we have taken every care to ensure the accuracy of our content, mistakes do happen. If you find a mistake in one of our books—maybe a mistake in the text or the code—we would be grateful if you would report this to us. By doing so, you can save other readers from frustration and help us improve subsequent versions of this book. If you find any errata, please report them by visiting `http://www.packtpub.com/support`, selecting your book, clicking on the **errata submission form** link, and entering the details of your errata. Once your errata are verified, your submission will be accepted and the errata will be uploaded on our website, or added to any list of existing errata, under the Errata section of that title. Any existing errata can be viewed by selecting your title from `http://www.packtpub.com/support`.

Piracy

Piracy of copyright material on the Internet is an ongoing problem across all media. At Packt, we take the protection of our copyright and licenses very seriously. If you come across any illegal copies of our works, in any form, on the Internet, please provide us with the location address or website name immediately so that we can pursue a remedy.

Please contact us at `copyright@packtpub.com` with a link to the suspected pirated material.

We appreciate your help in protecting our authors, and our ability to bring you valuable content.

Questions

You can contact us at `questions@packtpub.com` if you are having a problem with any aspect of the book, and we will do our best to address it.

1
Why Agile IT Security?

Sir Tim Berners-Lee, often referred to as the father of the modern-day Internet, regrets placing the duplicate slashes between "HTTP" and the Web address. Sir Tim admits that every time he hears a radio ad say "slash slash," he feels bad about the immeasurable amount of wasted airtime and he wishes that he had written the HTTP protocol better. The truth is that, when the HTTP protocol was created, no one ever imagined that it would become the vehicle for 21st century commerce or that it would become the technology standard for remote workers, social networking, banking, and more. When Sir Tim and his team created the HTTP protocol, they did not design it to be the e-commerce platform that it is today, in which all computers are interconnected in a simply insecure way. The Internet was created without taking security into account. It requires security to be layered over insecurity, which is rather difficult. The bottom line is that most modern companies do not do a very good job of securing their companies from the hackers and black hat threats of the world.

This chapter will introduce the need for Agile Security and the common problems related to IT security today. As a preview, here are the key security concepts that will be discussed in this chapter:

- What does security look like today?
- The history of security
- Understand the security challenges of today
- How and why Agile security was developed
- What do security breaches cost an organization?
- Current trends in security
- Understanding the risk that we face in security

Security built on insecurity

An organization's internal and external audiences can be divided into two categories: white hats and black hats. The white hats are good employees or customers who wouldn't do something wrong, even if they could get away with such actions. Black hats are individuals who are extremely inclined to do wrong. The term is denoted from the western movies in which the villains, or "bad guys", would wear black hats and the heroes would wear white hats. The reference to black hat that occurs throughout this book is a synonym for any person who could potentially harm an organization. The term "black hat" has a larger meaning that includes hackers, crackers, Trojan propagators, misinformed employees of an organization, and so on. Another point about black hats is that, for the purposes of this book we do not care about the questionable intentions and the motives behind the security breach are unimportant - we are only concerned with preventing breaches.

Perimeter security model

Today, most IT security organizations have evolved from the practices that were established many years ago. Companies often have not considered or changed their approach to security in many years. I term this mindset "the castle approach": put a big, fat wall around our perimeter, which we call a firewall. Then, we place tall, slender towers on the corners to watch out for intruders, which we call intruder detection systems. Next, we put a deep trench around the castle and fill it with water, which, in modern times, is similar to our intruder prevention system. Sometimes, we even put another wall inside the castle to protect the royal family, similar to present-day application firewalls. This defensive style worked well for many kingdoms in the Dark Ages, and then the invention of gunpowder brought about the demise of this defense strategy. Aggressors could then simply fire cannon and destroy the side of the castle. The next style of defense was a lower-walled, star-shaped pattern that allowed the military to move about freely and defend its stronghold in a more agile fashion.

Security landscape

Today's black hats are pushing through our limited defenses. Organizations that think they are protected seldom are. We need to rethink our approach to security and better adapt to the ever-changing IT landscape. Neither IT nor the hackers will wait for security to catch up and it is a race that professionals are losing. As IT security professionals, we need to focus on adapting to the needs of the organization versus following a plan. Rather than document issues, we need to implement improved measures and emphasize the importance of security professionals, tools, and policies as vehicles with which to aid the organizations and their employees.

The concept of agile security directly evolved from agile software development practices. Extreme Programming, Scrum, OpenUp, and RAD are some examples of agile software development lifecycle methodologies that have exploded onto the software development scene. My decision to combine some of the agile software development best practices with IT security practices produced incredible results. IT security became more agile and better equipped to deal with the demands of the organizations it supported. Security professionals were happier and had a better sense of their self-worth within the organization. Finally, policy was more focused and easily understood by members of the organization. To change is to grow and agile is one strong change for any IT security department.

Most organizations are established to achieve three main goals: to make a profit, to reduce cost structure and, finally, to remediate risk. Time must be spent in all areas to maintain proper balance in an organization. It is important to note that the primary reason that IT security is in place is to reduce an organization's risk. As fundamental as this sounds, agile security understands that there is a fine line between risk and security. It may seem like those two terms are synonymous, but the truth is that they are slightly different. We need to separate the need for security and boil it down to the risk associated with security threats. We need to understand the threats from a risk perspective and work under that premise.

For most organizations, security is not viewed as a profit center, but we consider some ways that can reduce costs for the company. Additionally, it is possible to establish security practices that will increase the availability and accessibility of the services that run our company. Most organizations would not dream of operating without certain types of insurance, so we must translate our IT security risk into plain, simple English in a manner that resembles how insurance companies present their arguments about threats. In fact, most companies spend a great deal of money on risk mitigation through insurance, business continuity teams, and disaster recovery teams. It is the job of an agile security professional to begin a dialogue in terms that executives can comprehend.

Security damages

So what happens when an organization gets hacked? The Web Incident Hacking Database's 2008 annual report estimates that the average organization loses approximately $6.6 million for every IT security breach. It also indicates that the cost of notifying customers that their personal data may have been compromised is $202 dollars per customer.

When systems are compromised, stock prices decline sharply, and shareholders tend to be disappointed. Some organized hackers have been known to purchase a corporation's stocks short before posting that the target organization has been hacked. Selling short allows an individual to profit when an organization's stock price drops. Hackers know that if they can compromise a system, the stock price will drop. When such breaches occur, organizations may have to defend themselves from independent or class action lawsuits that arise from the resulting damages. Regulatory compliance fines and audit levels can add up rapidly after a breach, not to mention the fact that the company will lose current and potential customers directly from the damages and indirectly from the media and injuries to its brand image. Some organizations will hire additional resources, such as monitoring services and consultants, which become costly very quickly. It is important to use a risk-based security approach to help mitigate the aforementioned threats.

Security trends

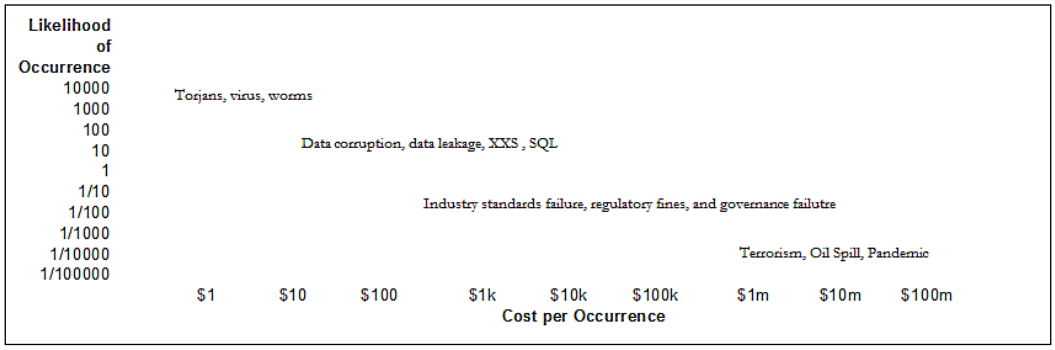
According to its 2008 annual report, the IBM X-Force team witnessed one million SQL injection attacks that year. It also determined that 90% of all vulnerabilities found in Web applications were exploitable. This finding indicates that 90% of the time when a vulnerability is identified in a Web application, that Web application is revealing sensitive data about the server from which it is operating or the data connected to the server. The X-Force team also discovered that the number of automated attacks on Web servers has increased almost 30 times in late 2008. Such a staggering number reveals that security is only going to become more difficult for the IT security professional. Furthermore, as attacks increase, it also means that the number of vulnerabilities will rise, equaling more risk for any organization.

The X-Force team has also determined that the number of organized hackers is increasing around the globe. Countries with high rates of education as well as elevated unemployment provide the perfect environment in which to spawn hackers, and hacking circles are actively recruiting new members in such places. Such a trend further complicates IT security professionals' jobs.

Security risk

Not all risk is created equally. Most people would agree that auto accidents and being hit by a meteor are both risks we face every day. Obviously, the risk of being hit by a meteor is far less than being in an auto accident. So obviously, some risks have a lower frequency of occurrence. Furthermore, not only do we need to consider the frequency, but the damageability as well.

For most people, being attacked by a bee is far less damaging than being attacked by a lion. Simply put, some risks are more damaging while other risks are less damaging. To bring this concept to IT terms, some events like Viruses, Trojans, and employees stealing data will happen frequently with the damage ability being relatively low. On the other side of the equation, oil spills, pandemics, and terrorism will happen seldom but are extremely damaging when they do occur. It is important to determine both the frequency and the cost per occurrence for every risk associated to an organization:



Summary

This chapter introduced the concepts related to modern day security risks and concerns, how the modern day architecture has evolved from the security of the past and created the workplace we have today. Agile Security is a style with the concept of efficiency at its heart. The style was developed out of the need to address the modern day challenges in the reality of the security landscape today. By now, you should have a general understanding of what is influencing security at the organization level today. In the next chapter, we will look at the security risks and the modern day threats, what they are, and the damages associated with them.

2
New Security Threats

Evolving risks

How do we defend ourselves and our organizations against constantly evolving modern threats? Some organizations have adopted software development agile principles to better address the ever-changing security landscape. It is vital that security adjusts quickly in order to challenge potential threats. Security professionals need to be able to work swiftly and accurately in a high trust environment among other professionals. Agile IT security offers just such a method. As you read through the list of modern threats, consider how you will combat them. It should become increasingly evident that agile practices are needed for us to mitigate risk and to establish security initiatives in a timely fashion.

This chapter will introduce the evolving threats that organizations encounter. We will cover the areas related to:

- The security challenges of cloud computing
- The new threat that Web 2.0 denotes to the organization
- Increases in the early adoption rates and the risks
- How regulatory compliance threatens the organization
- What advanced persistence threats are and how it effects the organization
- What Cyberware is and its effect on the IT security landscape
- How money mules work and how they could impact an organization

Cloud computing risks

Clouds, which I define as shared infrastructure, offer organizations many exciting new ways to deliver IT. There is no limit to the number of ways that clouds can be applied. The reason why I define clouds as shared infrastructure is because many people define them differently, and my broad definition allows me to discuss multiple scenarios under one general topic. And, truth be told, most security issues that relate to cloud employment tend to be the same, despite the minor differences in cloud formations. One of the first types of clouds that companies typically utilize is known as software as a service (SAAS). SAAS is extremely advantageous for a number of companies because they do not have to sell the hardware related to the software. In other words, once the company purchases the software, it takes on the day-to-day operations of managing the software. Salesforce.com is a good SAAS example of a company that tracks and maintains all of its sales information in one centralized location. All that its salespeople need is Internet access in order to connect with corporate sales information. Furthermore, any hardware that is associated with running the software is not realized, which essentially means that a company has less hardware to maintain, less energy costs, and less staffing. Typically, the monthly maintenance fees are much higher for SAAS applications. From a security perspective, this type of implementation contains a number of vulnerabilities. Many companies have complained about individuals not being removed from these target applications after their employment has been terminated. This is just one of many such problems that relate to SAAS cloud computing.

Web 2.0 risks

Other cloud implementations share servers with other companies, a situation that offers organizations the advantage of paying for the server or servers only when they are used. In essence, when a server is not being used by an organization, that organization is not paying for hosting at all and, when an application is being heavily used, extra servers will be provisioned for that organization at an additional cost. This cost-effective model is both powerful and useful. It also makes sense from a green perspective since energy is only used when it is needed. One of the main issues with a shared server cloud is that the corporate perimeter is no longer within its specific operation's walls, which means the company is at the mercy of the hosting company to protect its applications. Thus, if another application is running on the same shared infrastructure, any vulnerabilities that it has may impact the other applications and data present on the host.

Towards the end of the first decade in 21st century, we saw the emergence of new Web applications called Web 2.0. The new Web 2.0 technology, which is based on Flash, AJAX, Struts, and other technologies to give Web users a rich experience, was greeted with great fanfare as consumers quickly took to it. With Web 2.0, users can design their own Web experiences. Prior to its launch, all Web content was created by Web developers, which meant that much of the material was very dry. By allowing users to define Web content, Web 2.0 applications have exploded in recent years. However, Web 2.0 has vulnerabilities that can affect application usability and security. Some of this new technology has actually brought back old weaknesses, because the security implications of the new technology are not fully understood. Cross-site scripting, which is a method of executing evil JavaScript codes on a vulnerable application, has begun to reemerge in a number of Flash-based applications. A good example of a Flash-based application is YouTube. Flash language is extremely graphical, which makes it useful for customizing video. Most video-based Web applications use Flash as a video player. Inside Flash, developers can use variables, or containers, to help reuse code. If these containers were not sanitized, they could be used to execute evil code. Recently, a number of Flash applications have been discovered to contain undefined variables or containers, which could easily allow evil code to be executed by the susceptible Web application. Once that evil code is executed, all kinds of malicious things can happen to the user, such as the theft of session tokens, hijacked browsers, monitored user activity, and so on.

Bandwidth risks

As we move into the second decade of the 21st century, another area of concern relates to bandwidth and throughput. It is estimated that most companies and entities double bandwidth consumption every 18 months. If this trend continues, which it has during the last couple of years, traditional IT security may not hold up because most endpoint security systems, such as firewalls, intruder detection and prevention systems, routers, and so on., are based on reading traffic. If the rate of traffic continues to increase at this clip, data capturing may not be the answer for all security (chart 1). Graph 1 clearly reveals that society is accepting new technologies even faster than it had before. It took radio 38 years to be accepted as a medium of communication by 50 million consumers. In comparison, it took one billion people less than nine months to adopt the iPhone. Facebook is another great example of the extremely fast adoption of technology, and is one of the reasons why our bandwidth doubles every 18 months. This bandwidth explosion means that security professionals need to find other ways to secure their organizations. We must do a better job with identity management, data protection, and application development, in addition to reinforcing our security perimeter defenses.

Adoption rates:

Technology	Time taken to reach 50 million users
Radio	38 Years
TV	13 years
Internet	4 years
iPod	3 years
Facebook (100 million)	9 months
iPhone (1billion)	9 months

Chart 1. Adoption rate statistics

Regulatory compliance

Compliance is simply another security risk. It is very similar to a hacker in the sense that it is an external force that is causing the organization to adopt practices and procedures to protect against monetary losses. So, monetary losses are the typical result when a black hat successfully breaches a company's defenses. Similarly, if compliance regulations are not met, the company will suffer monetary losses. Simply put, an organization must defend against compliance threats in the same manner in which they would defend against threats from the external community. Today, most organizations face a number of compliance regulations that depend on their respective industries. It is typical for most organizations to comply with three to five different types of compliance, such as HIPAA, SOX, PCI, and so on. The nice thing about compliance is that if we do a good job in the core areas of security, most complaints will be satisfied. In some cases, organizations already own the controls to meet the compliance regulations even though they do not have them implemented. Occasionally, organizations have all the controls in place and cannot pull the information out of the existing endpoint security log and events systems. In some cases, all we need to do is have a better mechanism for pooling information into a centralized place.

Advance persistent threats

Advance persistent threats are constant attacks against an organization or government agency. Some people today believe Advanced Persistent Threats only refer to attacks on western governments; however, I believe this is untrue and unfair.

Many other companies have been the targets of advance persistent threats and they should be considered as part of any organization's security practices. Advance Persistent Threats are not anything in terms of vehicles used to hack, like Trojans, viruses, and phishing scams. What's new about Advance Persistent Threats is using these attacks coming together. Advanced refers to the fact that the viruses, worms, and Trojans are typically new and hard to detect. The Persistent refers to a constant wave of attacks. A lot of people think that a company is bombarded by attacks, but this is untrue. Most of the time, it is a slight trickling of attacks over long periods of time. The Threat part of advance persistent threat is interesting because this refers to a coordinated attack between attackers.

An Advance Persistent Threat may look something like this: a black hat group is targeting an organization and develops a new Trojan to infect the desktops of that company's employees. Once the Trojan is completed, the Advance Persistent Threat black hats will begin to search for information about the company. The black hats may find that an employee just published a new article or report; that employee will then become the victim of the attack. The black hats will then look for a high ranking officer in the organization to pose as. The black hats will pose as that high-ranking officer and send an email with something like, "This is Mr. Executive, CIO, and I notice you published an article today and I saw some grammatical errors in publication. I made some notes below and you can correct and resubmit it from the link below". What is difficult about this attack is that it is based on real events and real people in power, which makes infections all too real.

Social engineering risks

Social engineering, which is simply convincing an individual that you are someone other than who you are, is the newest trend in hacking. A YouTube vignette provides a perfect example of social engineering. In it, two gentlemen worked together, one to call a Wal-Mart store and the other to videotape the prank inside the store. The caller convinced the Wal-Mart store manager that he was from the corporate IT department. Requesting her assistance on a project, the caller asked her to grab her scissors, go to the cash register, and cut the cord between the computer and the keyboard. He said that he was attempting to convert all the store's cash register keyboards to wireless. The store manager proceeded to follow his instructions. Although the hackers in the semi-harmless ruse did not attempt to steal any money, the prank cost Wal-Mart time and resources to repair the registers and a loss of sales that day.

This example is simple and fairly harmless, but do not be fooled; social engineering is not limited to the simple and harmless. Another example combines social engineering with an email phishing scam. My wife recently received a phone call from a man who claimed to be a University of Michigan researcher. He said that he was conducting a survey for an environmental project. He requested my wife's email address and name and proceeded to send her an email. My wife went to the site and tried to find a survey, but the screen was simply blank. Evidently, a file had tried to execute on her PC. Thankfully, my wife uses a Macintosh and the ineffective .exe file was simply saved to her desktop. If my wife had been using a Windows-based PC, there is no way to determine how badly her system and the medical patient information it contained for her work would have been compromised. If that email had been sent without the phone call from someone who claimed to be from the University of Michigan, I doubt my wife would have ever taken a second look at it. I believe that social engineering will become a significant challenge in the years to come, and we must do a better job securing our organization from such attacks.

Mobile risks

Organizations today have evolved from having almost every employee and staff member located within the walls of the building to having people working from home, satellite offices, and mobile locations. Organizations have also started to outsource key organizational processes, such as human resources, payroll, and IT, all of which can occur outside the bricks-and-mortar facility. People are working from across the city, across the country, and across the world. Remote workers means more productive employees, but it also means more potential security risks. IT security professionals' concerns must focus on who can access the internal systems, which of them are accessible, and when they can be accessed. Do the remote workers need copies of our data to do their jobs? For example, IT development outsources may need a copy of production data in order to participate in testing and product development on their systems. Remote workers and remote access pose risks from both internal and external black hats. Disgruntled employees have access to our internal systems with less supervision. Remote connectivity offers hackers new vehicles with which to gain unauthorized access to our systems. Remote access requires both tooling and policies to properly secure an organization.

Now more than ever, people are using mobile devices to connect to the world. BlackBerries, iPhones, and smart phones are some of the tools that people are using to connect to the organization. Unfortunately, these devices may not be ready for primetime use. It greatly depends on the sensitivity of the data that the organization is trying to protect, the device itself, and the technology that is running.

Some devices lack encryption, which means that data can be extracted if the device becomes lost or stolen. Furthermore, an increasing number of devices are allowing third-party code to be run on them. Anyone can create applications for these devices and none of them contains virus protection. It is best to consider all mobile devices and do some threat- and risk-modeling around the acceptable use of such devices, the impact that they can have on organizational data leakage, and define security politics with respect to their usage.

Espionage risks

Most companies will see some form of corporate espionage at some point or another. Corporate espionage is done for a number of reasons, the leading cause being disgruntled workers. Other times employees are simply looking to make money from corporate secrets. Either way, whether employees are stealing information because they're mad or because they want to make money, the end result is the same. One of the most difficult problems with corporate espionage is that it tends to focus on unstructured data. Unstructured data includes files in Word, Excel, PowerPoint, or PDFs. These files are very difficult to secure and manage, because they do not contain good security instrumentation. Nevertheless, we need to focus on identifying this information and categorizing the risk factors that pertain to it. The section on tooling in this book addresses ways of securing unstructured data. For now, think about the sensitive unstructured data in your organization.

Social networking risks

Basic computer communication information, such as email, instant messaging, Facebook, and blogging, is becoming increasingly difficult to track. Communication applications can be used to leak sensitive data. Some mediums, such as email, are easier to protect than the other aforementioned examples. A number of services are available to monitor and track email that flows in and out of the company. Organizations need to consider information leakage that could result when employees voluntarily offer or are scammed into posting sensitive data within an instant message chat room or on a Facebook page.

People-profiling is a new type of threat that we are seeing in the world of hacking. With the advent of social networking sites and blogging, it is becoming easier for non-technical black hats to profile someone else online. An IT specialist at a bank asked a friend to find one of his friends whom the IT specialist did not know and agree to let this bank IT specialist attempt to gain access to the victim's major accounts. He promised not to take anything from the victim and said that he would not use any technical hacking techniques. Rather, his intention was to gain as much public information about the individual and make some intelligent guesses at some of her passwords.

After a few months of profiling and watching his victim on social networking and blogging sites, the bank IT specialist was able to profile the victim and determine where she grew up, the school system she attended, the birthdays, names, and ages of her family members and pets, and a slew of other information. Once the bank IT specialist was able to gain access to the victim's email address, it was game over because all the IT specialist needed to do was click on the "I forgot my password" button and reset that information for her checking and other accounts. This all relates back to the organization and its employees. It is human nature for employees to use personal information for passwords and pass phrases. If an employee of an organization can be easily profiled, then a black hat may gain access to the organization if proper countermeasures are not in place.

Zero-day exploits

Zero-day or zero-hour exploits are a couple of security world misunderstandings. A zero-day exploit is simply a vulnerability that is both discovered and breached during the same day. Although such exploits are extremely rare, they strike fear into the hearts of many IT security professionals because they are extremely challenging to defend against. The truth is that every application, server, protocol, and line of code contains thousands of undiscovered vulnerabilities, each of which have the potential to be found and exploited in a zero-day manner. The combination of threat- and risk-modeling is the key to understanding which systems are potentially threatened and how much time needs to be spent mitigating the risks. Risk-based security looks at the threat level for each system and determines the priority of defending against zero-day scenarios while analyzing whether the frequency of defense is low and the cost to defend is high. A number of security professionals refer to all unfixed vendor exploits as zero-day exploits, which causes much misunderstanding.

Cyberwarfare, Cyberterrorism, and Hactavism

Cyberwarfare and Cyberterrorism refer to the belief that black hats are funded by government organizations. The belief is that some countries like China are believed to have massive armies of state funded black hats. These black hats can target other governments or companies for reasons such as espionage, sabotage, and disruption. The idea behind Cyberterrorism is similar to Cyberwarfare, but the objective is to cause fear and panic. Advanced Persistent Threats seem to be the weapon of choice for Cyberwarfare. It was believed at one point that Google was the victim of Advance Security Threats. Advanced Persistent Threats coordinated by the Chinese government led groups to believe this. Cyberwarfare and Cyberterrorism play an important role in security; they help draw attention to the industry and help raise awareness about IT Security, which is good. As long as programs like 60 Minutes and other national media want to spend air time talking about IT security, we need to let them.

Hactivism is also closely related to Cyberterrorism and Cyberwarfare. Hactivism is hacking for a cause. Hactivists are typically politically motivated and target "bad" or "wrong" companies or political organizations. Most Hactivists are more interested in disrupting with Denial of Service or DOS, vandalizing, or creating worms for these organizations. Unfortunately, too many people hide behind Hactivism when caught hacking. Either way, hacking is hacking, no matter what the reason behind the black hats.

Money mules

A money mule is technically a person who transfers stolen money from a victim to a black hat. The money mule scam is commonly referred to as the act of recruiting, obtaining, and transferring the money. In the past, money mules have been associated with low-tech email phishing scams, asking people for money in some sort of way. Black hats typically targeted users with the intent of gaining access to their online checking account. In recent years, the attacks have become more sophisticated.

Black hats have started to use advanced malware that can actually fool a user into thinking an online account has the correct balance, while the black hat is pulling money out of the account. This type of attack is referred to as a "man in the browser" attack. The computers are infected by means of Driveby downloads. A Driveby download is the downloading of a file or files without the knowledge of the victim. This can happen in a bunch of different ways; for example, a website may ask if you would like to install the most recent version of a plugin or ActiveX control. Or the website takes advantage of a known vulnerability like the Windows metafile vulnerability. Metafile vulnerability was an early problem with Internet Explorer 6 that allowed files to be remotely installed. Sometimes well-disguised downloads can look like something else.

The newest trend with money mules is that black hats are starting to target corporations. The main target of these money mules are small businesses with large bank accounts. One of the problems is that most business accounts are not protected by the banks like personal banking accounts. Many businesses are finding out that a large amount of cash is missing and there's no way to recoup it.

Summary

In this chapter, we looked into some of the real world risks and complexities surrounding IT security today. We looked at how the risk can impact the organizations and the unique challenges each risk represents.

In previous chapters, we discussed the security landscape. In this chapter, we saw the more detailed risks that make up the landscape. The greatest challenge any IT security team faces is the combination of these threats and risks; the IT security landscape is moving and changing at a rate faster than anyone expected, yet the average IT security department is lucky to see any growth in the IT security budget at all. In the next chapter, we will begin to understand an approach to IT security that will allow us to do more with the limited resources we have.

3
Agile Security Team

Getting started with Agile

Getting started with Agile may seem overwhelming at first, but I will offer a few suggestions on where to begin and how to overcome the initial false starts. First you should determine what level of support you have in the organization or this effort. Does your organization support the adoption of Agile? Is the organization resistant to the adoption of Agile? If you don't know, it may be helpful to read the section in *Chapter 4* on Culture, to better assess the overall acceptances of Agile in your culture. If you feel the organization is going to be resistant, the next area to consider is whether your direct manager will be accepting of Agile and its approach. Based on the support of the organization and your manager, you can determine which principles of Agile you would like to support. If you feel support for Agile will be great, then multiple principles should be adopted at once. If support is generally low, then few principles should be adopted at a time. It is possible that some environments may not be suitable for Agile principles at all.

The next consideration is to think about the team you are managing. Some teams will adopt principles faster than others. Think about the team and how it is currently functioning and which functions will be easiest to adopt. Focus on those principles first then adopt others. It will take a few months for the changes to become accepted and possibly a few more months for the changes to become second nature. Also consider the location of the members of your team. Is everyone located in the same building or is your team collocated? Do a lot of team members work remotely? Some principles will be easier to adopt than others based on the geographical locations of the team. It's also helpful to understand if the team members are full time members of the team, or if members share responsibilities with other groups within the organization. If team members are part-time, some principles will be more difficult to adopt then others. If a principle or Agile practice isn't conducive to your team, it may be prudent to develop a hybrid or homegrown practice. The benefits of Agile can still be reached without full adoption of all practices and principles.

This chapter will introduce the concept of an agile IT security team and the getting started approaches and the concepts related to initiating a new agile effort. At a glance, here are the key concepts to be discussed in this chapter:

- The concepts and principle behind the Agile Manifesto
- How to offset resistance to the new agile process
- How to build trust with your team members to embrace the new agile process
- What level of change is appropriate for your team
- What ceremony is and gaging how much should be used

Agile focus

The focus of agile security is the concept of a whole team approach. In most organizations today, security professionals are siloed into different categories and various buckets in what I call a traditional approach to security. In the agile world, we focus on a collaborative, whole team approach. Security professionals in this arena must have expertise in all areas of security. They are held to a higher standard than in traditional efforts. Security professionals must understand basic endpoint, data, authorization, and application security. The whole team approach also means that every person takes responsibility for security. In the agile world, anyone could be asked to complete any task.

Agile team approach

The whole idea behind this is that any security breach has a detrimental effect upon the security team. The impact is typically shared among a team anyway, so it is better to work together on the most important items as they are identified. This is typically a big change for many companies, but it is one of the best ways to work efficiently and effectively in a rapidly changing environment. The agile approach's fundamental premise is that security professionals can easily look for new projects to address. The bullpen will list a number of issues, problems, and threats that require attention at any given time. There is no need for security professionals to wait for tasks to be handed to them.

In some organizations, it may make more sense to combine agile with other more traditional approaches. Yet, in some of the highest risk applications and data, it may be more sensible to continue with traditional security methods. The culture of some organizations is conducive for Agile IT security implementations. The principles behind the Agile Manifesto are as follows:

Principles behind the Agile Manifesto:
Organizational security by rapid, continuous delivery of security countermeasures
The frequent delivery of security controls and countermeasures
An acceptance of new security changes/challenges
Close, daily cooperation between sponsors and security
The belief that face-to-face conversation is the best form of communication
Continuous attention to technical excellence and good design
Simplicity
Self-organizing teams

Chart 2. Agile Manifesto Principles

Offsetting resistance

As you start to roll out Agile principles and practices to your team you're bound to meet up with resistance. This holds true for Agile or any idea you may have to change the company. No matter how great, super, and wonderful you think the new Agile principle will be, it's likely that someone somewhere is going to have a problem with the new practice. Your first instinct may be to calm your team member's fears with a heavy dose of logic. The problem is that most fears are not based on logic but emotions , which are very different. All the logic in the world will not help a person overcome his or her fear of heights, and the same holds true for fears of Agile. The secret is to embrace the resistance, and dig into the aspect of the resistance. The resistance may give you some insight about the teammate. You may also find out something you had not considered and alter your plans in some way. You may even change your plans completely.

A while back I was put on a project to implement an Intruder Detection System (IDS) and the network policies surrounding the project were unclear. This didn't bother me because I knew the IDS system needed to be put in place and the network signatures could be figured out as we proceeded. One of my team members, Fred, resisted heavy and insisted we get every policy and network signature figured out before we moved forward with the project. This, in the software development world, would be called a waterfall process. He was afraid I was going to be burnt if I didn't have total understanding before I started the project.

My first instinct was to explain how this was important to the company and it needed to get done. But my logic was lost on him. When I got more frustrated, I stopped and listened to his reason for resisting me. My team member, Fred, had a good point. If I didn't have all the policies and procedures in place, I could not request a resource from the resource pull I needed to pull from. It was a totally procedural thing from a group I had never worked with before. Working with Fred we were able to create a shell policy and requirements that were adequate to get the resource on site. Fred and I did not spend a lot of time creating a detailed, accurate requirements plan and in doing so the project changed a bit. In the long run we were ultimately successful and finished the project early. If I had not listened to Fred, I would have missed the dates because of procedural issues. For this, I thank Fred.

Whenever someone is asked to change to an Agile process, they are going to ask themselves if the change is possible. How much work is required to accomplish this task? What are the consequences if I ignore this request? What do I gain if I accept this request? Teammates will go through this process in their mind, so it's important to fill in the blanks without making the new Agile process seem like a demand. No one will embrace forced change, so it's important to make change seem natural and voluntary. Any resistance should be viewed as information only and not just complete resistance. Remember, resistance is a chance to better understand a team member, or is an opportunity to better understand your goal with additional information. Either way, you win, so embrace resistance.

Agile coaching

Agile coaching is an important aspect of the Agile process. The Agile process builds upon itself with the use of coaches. As team members build and use the Agile principles discussed in this book, they can teach other team members. Coaching can be challenging at first for peer-level coaching. Since most coaches don't have the power of management, the strength of the company is used to influence decisions. The coach must rely on his or her own abilities to help mentor others on the team. We will now look at four factors that a coach should consider.

First, a coach must consider what he or she brings to the table. What is their skill? What value does this hold? The value must be understood and conveyable so that the student can easily understand the value.

The second element is the concept of approachability. A student must feel as if the coach is looking out for the student's best interests. The student must feel like the coach isn't going to take poor or slow results to the manager. The student must feel comfortable coming to the coach for help in a penalty free environment as much as possible.

The third element is integrity; the coach must be consistent with the student. The coach must use good judgment and admit when he or she is wrong.

The fourth element is includeability. The coach must make the student part of the decision process to make the student feel like it is his or her decision. When coaching, the discussions should be more conversational and the student should interject. The student that is part of the overall decision process will be more likely to take the advice of the coach. Trust is something that takes time to build but can be faster recognized by taking into account all the factors mentioned here.

Trust exercise

In order for Agile to succeed in an organization, team members must feel safe. Safe to discuss difficult topics without feeling like negative consequences will come after a difficult discussion. A quick way to establish the overall trust of your team is to try this Agile exercise called the Trust Test. Have all the members of the team located in one room. Appoint one team member the chairman and have all the managers in the room leave. Each team member is given a 3" x 5" index card. Instruct them to write one of the following words on the card: "full", "strong", "marginal", or "low". A team member should write full on her card only if she feels like she can discuss the issue with anyone at any time. Strong means a team member feels she can discuss almost anything with a few reservations around extremely sensitive issues. Marginal means a team member has reservations in general about discussing topics but will at times. This team member is a person who picks battles carefully. Team members should write low if she goes with the flow and is mostly unwilling to challenge anyone in the organization.

The cards are collected by another team member, which are face down. The card collector will shuffle the cards to ensure anonymous answers. The card collector will read off the answers while the chairman tallies the votes on the whiteboard. This drill should be done regularly and hopefully as the Agile process continues the overall trust of the team should improve. Performing this test every few months is important to better understand the trust of the team. The hope of any Agile team over time is to see the trust improving. If the overall trust of the team is low, Agile will have a difficult time succeeding.

If trust is low, don't take it to heart. Understand that all organizations are different and the test itself is a good first step. Consider a retrospectiveness exercise to help build trust. Revisit recently finished projects and consider building a timeline. Have the start date, milestones, and finish dates listed on the whiteboard. Talk about what happened during the project, what was done well and what was done poorly.

Try to discuss how people felt and how we could have done things better during the project. It's important to focus on the positive because the things we did well, if improved, have a much better likelihood of improving the overall efficiency of the team. Let people discuss, learn, and build trust. This exercise should be considered a penalty free environment and solely done to improve the efficiency of the team, trust being the strong byproduct of people working and growing together.

Consider other team building exercises as well. Understand the organization's culture, what it allows and what kind of functions are not allowed. Organizational culture is discussed in *Chapter 4*. Use off site exercises whenever possible. Also consider taking the team out to a local event. Maybe take your team to a Whirly ball course or a Bocce ball course. Consider doing simple team exercises to help build a cohesive team and break down the walls between the team members.

Degree of change

When considering the Agile principles and concepts beneath them, it is also important to consider the degree of change associated with the project. The degree of change simply refers to the unknown elements of the project we are approaching. The degree of change is determined by estimating the amount of experience the company and the people on the team have implementing the countermeasure being planned. If the countermeasure has been implemented many times by an organization and the implementation method is well understood by the team members, the degree of change would be low. On the other hand, if the project is relatively new and the members of the team do not have significant experience implementing the countermeasure, the degree of change would be high. The degree of change is important to understand because it may impact on the principles we may or may not use. For example, if we are implementing an IDS system, say snort for the 50th time working pairwise, a concept I will discuss later, will not be productive. If we are working on connecting our SAP system to our enterprise identity and access assurance project for the first time and our team members have never written this type of interface, then we should heavily consider using the pairwise principle, along with other Agile Principles. Agile IT security is a way of reducing risk and the degree of change is simply another way of calculating risk; Agile IT security is a way of remediating the risk.

Agile ceremony

Another consideration point is ceremony; how much ceremony do you want in your project? Ceremony is the formalness of the agile culture. More formal, high visibility to the agile practices would constitute a high amount of ceremony. A low level of visibility and formal process constitutes a low level of ceremony.

When considering the level of ceremony to use for a given project, consider the length of the project, the degree of change, and the amount of Agile used before. In short, for low risk projects I tend to use the agile principle without much ceremony, and in some cases the team doesn't even know I'm using Agile principles. Also consider your culture and team members when considering the amount of agile ceremony you may want in your next project.

Summary

This chapter covered the steps that you would have to follow in designing and introducing a new agile practice into an organization. The approach taken will depend on the organizational structure and style. It is important to plan ahead to offset resistance and to coach team members to help ensure the success of your new security approach. Additionally, one should consider how fast and the amount of formalness the overall project should have.

In the next chapter, we will look at the Agile principles a team can use to increase the productivity of an Agile IT security team.

4
Agile Principles

Need to evolve

We are living in some of the leanest times in history. Every decision that an organization makes has an impact on the future. Motorola is a classic example of a company that was well respected and well organized. It introduced the Razr phone in the early 2000s and the company was generally performing well. Customers flocked to the Razr and it looked like Motorola had its finger squarely on the pulse of what the consumers wanted. A few bad decisions about product directions and Motorola's lack of willingness to evolve to where the market was going became evident. Suddenly, Motorola was in some serious financial trouble. Organizations need to move fast and the conservative approach of hanging behind technology to see where it goes is becoming more risky. In order to survive, organizations as well as IT security professionals need to be more agile when making decisions.

In this chapter, we will discuss the approaches available to the agile IT security team. It's important to realize that not all approaches may be used by a team, but a combination of the approaches that best fit the project, organization, and team. The approaches we will discuss are as follows:

- What to consider before deciding which agile principle to adopt
- How pairwise can help reduce implementation risk
- Why refractoring can help simplify a design
- How decomposition, agile spikes and small deliverables can help keep a project plan on time
- Why collective ownership can help an IT security implementation project be more successful

- Lean principles of simple design and minimizing waste
- The logic of why done means done
- How to track project divergence and velocity rates
- The value of understanding yesterday's weather
- Unique collaboration principles of agile security, which include the scrum master, agile planning, and standup meeting

Risk-driven security

Agile's first tenet is to plan with risk-driven security. We need to make frequent use of our bullpen to better understand the needs that are driving our activities. A discussion about the bullpen will soon follow in this book. The bullpen will give us the organization's risks and priorities that we will need to combine in order to deliver real business value and risk mitigation.

Hiring an agile professional

When forming an agile team, we need to find IT professionals who understand IT security core practices. This means that agile IT security professionals need to understand access management; database security; application design; endpoint, network, and server security; and physical security tooling. They need to understand the threats and risks that face an organization and be able to work well with other team members. Hiring good agile IT security professionals can be challenging because we need to consider more than just the role that the agile security professional is filling. When hiring a new team member, consider whether he or she will mesh well with the other members and whether he or she will embrace the whole team approach. One bad recruit can undo months of agile methodology progress.

Culture

What kind of culture does your organization promote and practice? Consider the security culture in your organization. Do people like, hate, or fear the security team? It is important to understand how your organization operates so that the types of security tools and policies that should be implemented may be determined. First, we need to consider the organization's values: What are they? Are employees aware of them and are they being carried out in daily activities? It is said that a strong corporate culture is indicated by the alignment of management with corporate values.

Conversely, weak culture is the lack of a line between organizational values and management. Take steps to better understand your corporate culture. Is it a "tough guy" culture? Perhaps it is more of a "work hard/play hard" culture, an "all in" culture, or maybe a "process-oriented" culture. The type of culture tends to align itself with the type of company in question. For example, a large software-based company may have a more "process–oriented" culture, whereas a stock investment account may have a "tough guy" mentality. It is important to understand the type of culture because it indicates which security measures will be required and those that will be outright rejected by the organization. Such intelligence becomes valuable when creating a blueprint to determine which tools are most appropriate to fill our specific gaps.

Changing culture

Culture is a collection of attitudes, values, beliefs, assumptions, and attitudes shared by a group of people. Cultures vary greatly from organization to organization and are similar to the neighborhoods in which we live. Many parallels can be drawn between neighborhoods and corporate culture. I have lived in a number of neighborhoods in my life, and each has been distinct. Some neighborhoods are very community-oriented; for example, if a mailbox was knocked down, many neighbors would come out of their homes to help me to repair it. If a fence needed painting, neighbors would help to paint the fence. On the other hand, I have also lived in communities where people were quite private and seemingly unfriendly. If a mailbox were to be knocked down in such a neighborhood, the mailbox's owner would repair it alone.

One of the most difficult aspects of Agile Security is changing the culture of the organizations for which we work. Culture was developed for a reason, whether good or bad. The first key to effect culture change is to gain management and executive agreement, at the highest possible level, about changing the culture. This can be similar to a sales job because some managers or executives may see change as disruptive. The higher the cultural change, the more profound the impact will be. The key is to focus on what can be controlled. Having an Agile Security team can be hugely significant, even if the entire IT department does not adopt the same culture.

One of the key aspects of changing culture is to identify the people who are willing to change and the people who are unwilling to change. The idea is to embrace the people who want to change and to create a hot spot, where the new highly collaborative team can work together toward some goal. A shared workspace for employees to work together and play together should be established: open workspaces that foster collaboration and team play, as well as areas for brainstorming and creative thinking. It is ideal to eliminate enmity and to increase trust among employees. Heroes who are embracing the new culture movement should be created within the organization.

Do not simply preach about the change and how great it will be, but lead by example and show people how to work together as a cohesive team. Coach team members in conduct befitting of the new culture. Give examples of other organizations or communities that use the new culture and how the new culture will improve the organization. Motivate people to want to change and provide the opportunity to change.

Lastly, not all people will be willing to change; tough decisions may need to be made for a better tomorrow. Demoting and terminating such employees may be the only options for stubborn culture change disbelievers. This will allow new members the chance to become part of the team. Remember that change will not happen overnight. Culture change is slow and takes a long time, so be sure to create a quarterly plan to review culture changes and to collaborate with others in order to ensure the changes are headed in the right direction.

Focus on strength

One of the keys of Agile Security is to focus on people's strengths. It seems commonplace for people to focus on an area of weakness in order to become well rounded. A person who is bad at organization may look for a course with *Franklin Planner*. Someone who is bad at programming may take a programming course at their local college. Working on weaknesses will never yield the same result as working on strength. When building on strength, the net gains are much greater. Let me illustrate this now.

Potential is measured by someone's natural ability X training. To illustrate, let's use a scale of 1-10. Let's say my organizational skills are naturally very low, say 2, so I take a few courses to help improve my natural ability to say, 4. I would find that my potential is now 8. On the other hand, if I am naturally very strong at strategic decision making, say I'm an 8, and I take some strategy courses and attend some seminars to put my training at a 4; my potential at strategic decision making would be a 32, which would make me really good at project coordinating. As you can see, a person's potential in a given area is much greater if a person focuses on their strength.

Another great example of this is Michael Jordan, who was arguably one of the greatest basketball players of all time and had natural ability. Michael also practiced day and night, which is a great example of natural ability and training working together. Conversely, Michael Jordan was not nearly as gifted when it came to baseball. Michael put an equal amount of effort practicing and training into baseball as he did for basketball. Unfortunately, Michael Jordan's brief career in the Chicago White Socks farm system was nowhere near as successful as his basketball career. Michael Jordan would have been better off spending time focusing on basketball from an efficiency perspective, and not overly focusing on his weakness.

Ironically, if we ask people what their three strongest skills are, most people would not know. Most people are aware of their weaknesses, but will seldom speak of them. A good agile team will focus on people's strengths and encourage growth in their areas of strength.

We will also use the Pairwise System to pair people with strengths and weaknesses together. Pairing the person who is strong in leadership with someone who is weak in leadership will help build organizational skills organically. The strong creative people should be coupled with the weak creative people to help build skills. Strong organizational skilled people should be paired with weak organizational skilled and so on. The idea is that a person weak in a skill will learn from someone strong in a skill, to see how the person works and operates and to be able to learn from a person that is naturally good at a skill.

The general categories for strength and weaknesses for IT security are creativity, organization, technical, and leadership. It is rare to find someone who is strong in all four categories. People who are equal in all four categories are typically not strong in any one category. People strong in all four categories are commonly called "Jack of trades, master of none". A good team will have people strong in the technologies we need to support our organization. Conversely, we also need a blending of people who have strong secondary skills such as creative, leadership, or organizational skills to support the organization. A good team will have people in each strength zone and will use people for their strengths to help keep the organization moving forward. Team members with good organization skills should keep us organized and on schedule, while creative people should be put on more challenging projects to help build a creative solution.

Leaders should be distributed among the different teams and not doubled up to avoid conflicts. Many sub-categories exist for the main categories. Hundreds of exams exist to help better understand yourself and your strengths. I would recommend having team members take the same exam so you can have a better understanding of everyone on the team. I recommend Strengths Finder 2.0, by Tom Rath. Rath talks about the advantage of focusing on strengths. Each book purchase includes a unique code to access an online exam, which will assess your individual strengths and give a description to better understand them.

Pairwise

Agile security professionals frequently work in paired teams, which makes a lot of sense. When people work together, they are forced to reconcile any differences to ensure that they share vision on a given assignment.

The ability to reconcile ensures better understanding of the tasks at hand and how to approach them. It is difficult for both members of the security team to have a misunderstanding when they are equally required to work together to organize thoughts. Such an approach also aligns itself with the concept of mutual ownership, which means that anyone can fix anything, anywhere when needed, which is another core function in agile.

Refractoring

Agile IT security professionals need to practice the concept of refactoring. Refactoring is the concept of simplifying a particular design. So, any time an agile IT security professional looks at a project or a task, his or her team always looks for ways to make the process simpler. Pairwise refractoring lays down the framework for this concept to blossom. Working in pairs allows everyone to mutually own all organizational security practices. Since all security professionals own all the solutions, it is no problem to refactor and simplify any previous or existing implementation or policy. Since we are working pairwise, we have the courage to tackle a tough refactoring assignment. One form of refactoring we may want to tackle is the actual layout of our hardware in the physical security space. We may have hardware scattered throughout the organization and refactoring may be a good option to unify the hardware in one place, thus reducing the number of areas we need to secure.

Small deliverables

In Agile, we focus on small deliverables. Small deliverables allow us to build momentum with the team and to see completion early and to level pressure over time. Small deliverables are preferred because if a project team has a year to deliver a solution, the early stages will move slowly, and be trumped by more pressing matters in the organization. As we move into the ninth and tenth months of a project, the 'oh no' factor kicks in and people start working frantically trying to deliver. This is similar to a college student trying to cram a semester worth of chemistry into her brain in just one night before finals. It typically does not work out that well. Usually, when we have long projects we end up missing the dates because the early months were not maximized. Small releases help a team better manage a large project time. By making a small release with faster delivery dates, we can put constant pressure throughout the project and not have extreme pressure at the end of a project. It is also helpful for stakeholders and owners to see progress as we move forward throughout the project. One of the Agile principles we will use to design small deliverables is the principle of decomposition, which is described next.

Decomposition

We want to look at delivering value as quickly and frequently as possible, which means that when we are scoping a new project, we will want to dig deeper in search of lower-level milestones and deliverables. This method is called "decomposing". This method of delivering smaller sections more frequently allows us to gain some advantages, such as less procrastination. It is human nature to procrastinate and work on the projects that are most interesting instead of the projects that are most critical. Two months into a one-year project, the sense of urgency to work on the project is significantly less than it is at the 11-month mark. That is when panic sets in and everyone starts working extra hard to compensate for the previous 11 months. Typically at this point, we are too far behind to catch up. An iterative approach in which we decompose the big task into smaller sub-projects is important so that we can deliver something every few weeks and maintain a high sense of urgency.

It is important to attack the most risky and undefined areas of any project first. This is a simple policy to expose any hidden difficulties and continue to address those problems that parallel the project. I think that the Identity and Access Management initiative was a long timeline that was created to deliver round-trip user administration. Sometimes, the most difficult systems are left for last, which often means that things will slip. Agile IT security would recommend attacking the high-risk areas first to get the process moving and keep the project on target. The refactoring process that employs more frequent delivery dates and milestones is an IT security professional's best friend during large-scale implementation.

Collective ownership

One principle of Agile IT Security is the principle of collective ownership. Everyone on the team owns every artifact, solution, and system. To achieve collective ownership, you must consider reducing the area of expertise or silo and work on having more generalized team members as much as possible. The value of having people specialize in generalizing is significant. By having people more generalized we can reduce risk, bottle necks, and improve flexibility. We in effect become more Agile. By reducing specialization, we reduce the need for as much documentation. We also reduce the bottle necks because multiple people on the team can do the same job that will allow a team to load balance work appropriately. It also reduces the risk of not having a skill set due to attrition. If someone leaves the company, we have other people who can fill that role.

Agile Spike

An Agile Spike is used whenever a team is faced with implementing a solution that is not fully understood. A team may be struggling with a number of issues such as the amount of traffic a solution will see or the number of users that a solution will have. A Spike is a pilot project that attacks risk. Agile Spikes are always time boxed to reduce the risk of scope creep and keep our project on schedule. Whenever a team is faced with uncertainty when implementing a solution, Agile teams will look to create a Spike. Essentially what we do is create model architecture of our solution with our best guesses on how the system will behave. Once the Spike is developed, we can watch and observe this model to better understand how the real solution will behave in production. Our goal is to understand how it is going to behave and learn any early lessons before we begin the real project.

Simple design

Security teams are under constant pressure to improve the security landscape of an organization. Usually, while faced with low budgets and resources, a primary practice in Agile IT Security is to keep our solutions as simple as possible. We also don't need to fully understand the solution or countermeasure in detail before we begin. We can learn and change as we build the solution. Look for models and solutions and how the industry at large is solving the problem at hand. Talk to experts in the field to see what solutions are recommended to solve the problem. Once we have a basic idea of what we want we can start to implement the solution. Keep in mind that the solution can and will change and we can always change the design along the way.

Simple design is based on a collaborative environment where all team members are allowed and encouraged to share ideas and concepts needed to implement a security solution. Each team member is unique and has a different perspective; collectively a team can create great simple designs. The idea is to design the solution just enough to get started and start working on the project. Over designing a solution is all too common in the early stages of design. Early design patterns should look for patterns and redundancies, which are the foundation of creating reusable processes in our solution. The traditional approach is to design everything upfront then execute the plan. The problem with this approach is that as we understand the solution and the problem better, early decisions are no longer relevant. With a simple design, we can quickly build a design to get started and revisit the design as needed.

Minimizing waste

Another principle of Agile is minimizing waste. Failure is a huge form of waste but is relatively unavoidable when delivering any new security practices. So, the best way to minimize waste is to fail fast. To fail fast, you must create an environment where failing is expected and encouraged. When team members are comfortable with the idea of failure, team members will be more vocal about failing. Being more vocal is a good thing because a team member who is struggling will not sit at his desk searching for an answer, but instead engage the greater team to help deal with his or her problem. This allows members of the team to help collaborate and change directions and minimize waste.

I was managing a team in which we needed to quickly create a SPNEGO connection with our point of contact server for a potential customer. After the SPNEGO connection was made, we needed to create a SAML federated connection with the other applications in the environment. This was a spike program to which we needed to be successful to win business for our organization. The difficulties were all based on the fact that we were setting up the spike environment in a test environment, in which the conditions were not at all representative of a real world environment. With multiple millions of dollars and a several year contract on the line, failure was not an option. We also had a short window to get the environment stood up. We had two engineers working on the configuration of the server and we failed twice trying to set up the environment. Both times we learned, collaborated, and came up with a new approach. On the third attempt, we made the connection and we had the federation working within a few days. Ultimately, we succeeded and the customer was impressed with our abilities to quickly stand up the environment.

Done means done

In Agile IT Security, we focus on the end state. All too often, implementers of security tend to revisit and reduce areas that work. In other cases, we get started working on multiple parts of the solution and we lose sight of finishing. The basic concept of this practice is being 100% complete with a task before moving on to the next task or sub component. This doesn't mean that everything needs to be perfect. In fact most teams operate under the Pareto Principle or the old 80/20 rule.

The Pareto Principle claims that 80% of the work typically takes 20% of the effort. And that the last 20% of the effort takes the majority of the time. This may seem like I am saying to complete a task in its entirety and fall victim to this 20% pit fall, which isn't Agile at all. But what I am saying is exactly opposite of that. The trick to 'done means done' is to understand where the 80% ends and try to complete the task before falling victim to the strenuous last 20%. So, the principle of 'done means done' is crucial to understanding how we can deliver 80% of the effort with 20% of the work needing to be done, and to avoid over developing and over complicating our solutions. As implementers, we sometimes try to over delve or make a solution perfect and work on the cool aspect of the solution. When 'done means done', we know exactly what we need to deliver a project and we understand our end state.

Project divergence rate

A manager I reported to when I first started in IT told me the best way to start a project is to "start running in a direction, any direction, and we will make changes as needed." Little did she know she was using an Agile principle long before the Agile software development practices were started. The only problem with "start running in a direction, any direction, and we will make changes as needed" is that how do you know when you're running in the right direction? Project Divergence attempts to determine the rate at which our understanding of a project is changing. This is accomplished by measuring the rate at which our requirements are changing.

To determine Project Divergence, we need to set up regular intervals or pulling rates to measure divergence. If we are determining divergence on a day-by-day basis, all we need to do is write down the number of changes that day versus the day before. What we are looking for is a point at which the change rate drops and is starting to level out. This is the point at which we are beginning to run in the right direction. Knowing this point in a project means we can focus more on executing and less on planning and meeting. The number on the Project Divergence chart is not as important as the trend of the chart. The value of the chart is to understand how your team is trending.

On most projects, I use the tried and true MS Word to keep track of my notes for a given project. Each day, I take my changes from my notebook and type them into MS Word in order to keep a current listing that I need to share with other members of the team. Each day after I make the changes, I can compare yesterday's changes to today's changes and look at how many items have changed in the document. This is easily done with the document comparing function built into MS Word. The number of things I see that change will then become my velocity number for that day. It's not an exact science, but it's a good measure of how well the team really understands the project at that given time.

Project Velocity rate

Project Velocity is a measurement of how much work is going into a project, similar to Project Divergence, which measures our basic understanding of the project at hand. Project Velocity measures the work effort involved in delivering the project. Project Velocity only takes into account the amount of effort given in delivering the project, not the planning effort. Project Velocity is determined by the total hours taken to complete a task divided by the number of hours estimated. The higher the number, the higher the velocity, and the faster the team is working. The lower the number, the lower the velocity, and slower the team is working. Poker planning is a great tool for estimating the amount of time needed to complete a task. This chart, which is similar in concept to the Project Divergence chart, should be used to understand the trending of work velocity. Don't get hung up on the number; stay focused on the trending of the team.

The Project Velocity rate and the Project Divergence rate should be read next to each other when possible. A good trend to see is that the the Project Divergence rate begins to drop as the Project Velocity rate begins to increase. This is good because it means we are starting to work hard at completing the project once the requirements are understood. This is also helpful in understanding how a given project is trending based on previous projects. Understanding how a team is doing in terms of understanding the requirements and the rate at with the team is working is helpful in understanding whether dates are going to be met.

Yesterday's weather

This is another planning technique used by Agile teams. The idea behind yesterday's weather is that if I was asked to predict tomorrow's weather based on the sole basis of today's weather, I would be right about three out of four days. When it comes to estimating how much time we need to complete a topic, Agile teams should use Project Velocity numbers based on our last project. The concept of yesterday's weather refers to the fact that an agile manager should only look at the last project and not the average of the last few projects. For example, when we complete the first part of the project and we are trending at 0.9 percent of the estimated time, we should plan on the same project velocity rate for the second part of the project.

Collaboration

Collaborating together as a team is one of the hallmark principles of Agile, mostly because of the simple idea that two brains are better than one. "If a solution is only reached when we grow tired of thinking, then let's never come up with a solution". I am not sure where I heard that, but I think it plays well into the concept of collaboration. Although if we never come up with any solutions we won't have jobs, but the premise is interesting. Collaborating comes in two forms, collaborating with a team in the same physical location and collaborating with a team in geographically different areas. Considerations should be made to help facilitate the collaborative process for both physically located and geographically distributed teams.

For physically located teams, time should be given to help improve the facility for collaboration. Knock down the walls, cubicles should be removed, and an open environment should be developed so people can freely communicate. Team rooms should be developed where team members can meet and discuss project plans, issues, and delivery dates. Team rooms should be surrounded by whiteboards and dry markers to encourage collaboration. White boards are great because you can write freely and also stick 3" x 5" inch index cards to them in a pinch. Sound deadening partitions should be used if possible to define your work space and to cut down on noise pollution from other teams. Other items can be incorporated that encourage creative thinking and collaboration such as a foosball table, toys, mascots, and the like, which are real additions to any collaborative environment. Also, have a master calendar in a central spot for marking important dates and events.

With teams that are physically located in different areas, collaboration takes a different form. Some of the basic tools you can use are email, instant messaging, and online team rooms. Your team room should be capable of facilitating an online white boarding session where multiple people can whiteboard at the same time. It should also have a centralized calendar for marking important dates and events. The online room should have an area where team members can co-author artifacts and work together on creating artifacts. Geographically distributed teams should have dial-in bridges to ensure collaboration can go on. Video conferencing can be used as well.

Scrum Master

Agile processes need Scrum Masters to help keep projects moving. I once heard a joke about a chicken and a pig that walked into a bar. The chicken said to the pig, "We should open a restaurant next door to this place, what do you think?" The pig said, "Good idea. What should we call it?" The chicken responded, "Ham and Eggs." The pig said, "No way. I would be committed, but you would only be involved".

The point of this joke is that we need people to be committed, rather than just involved, in the process. We need Scrum Masters to take ownership. The Scrum Master has a few jobs. First, he or she must make sure that the team does not overcommit to management. This is less difficult than with traditional project plans because agile refactors everything to the lowest common deliverables, allowing the Scrum Master to stay on track. The next step of the agile process is to meet with each member of the team on a daily basis. He or she asks three questions:

The Daily Scrum
What did you do yesterday?
What will you do today?
What is holding you back?

Chart 3. Scrum Masters' daily questions

It is expected that the Scrum Master will resolve any issues that arise from the daily agile process meeting as quickly as possible. If, for some reason, the Scrum Master cannot resolve the issue, then he or she will assign the task to another team member for resolution. The Scrum Master's main job is to ensure that everyone lives the agile values. A project manager or a team lead usually, although not necessarily, fills the Scrum Master's role. In the development world, such a role is also referred to as that of the Scrum Master, and his or her daily duties are referred to as the daily scrum.

When organizations have multiple and concurrent agile projects, it may make sense to have an agile master Scrum Master. The idea behind this role is to ensure that each Scrum Master is keeping up with his or her responsibilities. The idea here is that if we have large numbers of people, we should break up the teams into 6-8 members, each with multiple levels of Scrum Masters. The presence of multiple layers of Scrum Masters is a common practice in large-scale agile shops.

Agile planning poker

Once a project is underway and the project is refractored down to its smallest components, one of the ways we can estimate the time needed to compete a project is to use a trick I learned in the Agile software development community. It's time to play a little poker. Poker? Yes, poker. You will need a few things to begin the planning poker session: you will need an egg timer and planning poker card deck. Planning poker card decks can be found online or made by hand. Planning poker card decks are based loosely on the Fibonacci sequence or some variant of that model.

A traditional deck would have 0, ½, 1, 2, 3, 5, 8, 13, 20, 40, 100. This odd sequence is used to remind players that estimating is not an exact science but estimation. It's also designed to make the game more challenging because if you think a project is going to take 10 days, you will either need to play an 8 or be conservative and play 13. The odd sequence of numbers also makes for interesting conversations around the thought process behind each decision. Some decks will include other options like a question mark (?) or a coffee cup to indicate a break is needed. Decks can be found online and vary from place to place. Find a deck your team is comfortable with or make your own deck.

Planning poker always needs a dealer. The dealer should be a person who is the most knowledgeable about the project at hand. The dealer in planning poker typically does not play and only facilitates the planning poker session. Each team member is given a deck of cards with all values and holds the cards like a poker hand. The dealer, who is the most knowledgeable about the project, will give a short description about the project at hand. Team members are allowed to ask questions and discuss. Once the short discussion is over, each team member grabs a card from their hand and places it face down on the table. Concurrently, each team member turns over his or her card at the same time. Next, the members with the highest estimation and the lowest estimation get a few minutes to discuss why they estimated the time they did. This time is referred to as "soapbox time" and is typically time boxed with the egg timer. The time boxing is optional and is typically not used until after the first few rounds of discussions. Once the "soapbox" seasons are over, the team members pick up the cards and start a new round. The subsequent rounds are identical to the first round in that each member lays down a card with his or her estimation on it. Each team member concurrently flips over the card and the highest and lowest estimations get a few minutes to "soapbox". The game ends when all members agree upon the amount of time needed to complete the project. Or the project owner, the person who will be responsible for delivering the project sees a majority vote that seems reasonable.

Distributed teams can take advantage of planning poker as well. This can be achieved easily with a phone bridge and group instant messaging session. Each team member can instantly message the response into the same window and be verbally cued as to when to hit the *Enter* key. Specialized software exists to help facilitate planning poker session such as `www.pokerplanning.com`, which is a free online service that claims it will be free forever.

Standup meeting

Agile IT security implementers prefer to meet standing up. Most people spend too much of their days in meetings. It is important to conduct meetings so that we can communicate and collaborate, but it's not good to simply meet. A solid approach to make meetings more productive is to conduct it while all participants are standing. The idea is to boil long-winded conversation to its core. Most people feel bad and don't talk as much if they feel they are going to keep people standing for a long period of time. Some agile groups will limit meetings to 15-minutes, and hard stop the meeting after 15-minutes. While others will observe the 15-minute rule as more of a guideline and allow meetings to last a little longer if needed. When a meeting starts to run long, the obvious warning signs will begin to set in. People will stagger and sway and reveal body language that indicates they would like to wrap up the meeting.

Summary

This chapter covered the agile principles an agile IT security team uses to work more efficiently. The agile principle serves to shape the project design, team, people, and planning into the most efficient and effective patterns possible. The principles are used to shape people into collectively owning the IT security implementation project and serve to bring higher value to the organization. The design of the IT security team is completely up to the team manager and it is important to use only the principles that will be natural for the team to use. In the next chapter, we will discuss how to identify weak areas in the organization using agile gap analysis techniques.

5
Agile Risk-Driven Security

In this chapter on risk-driven security, we will look at the various ways we can use risk to understand an organization's security landscape. At a glance, will look at the following areas in this chapter:

- How to identify an organization's high value assets
- What risk-driven security is and how it can help an organization
- The bullpen and how to use it to find security gaps in a organization's infrastructure
- Best practices in determining which risks are the most impactful

Data value

Data is at the heart of most security and is typically the information that the black hats are after. In less frequent cases, such as in denial-of-service (DoS) or fuzzing attacks, black hats intend to interrupt commerce. But, for the most part, they seek data as it carries financial value on the black market. The value depends greatly on the information they gather. For example, names, addresses, and social security numbers will fetch less than a dollar per record, whereas names, addresses, social security numbers, dates of birth, and mothers' maiden names, will bring about a larger profit. Other information that will let terrorists know about weaknesses in a building or structure could potentially offer much more. Black hats can trade this information for gold bullion, which is literally the gold standard in hacking, through online escrow-based trading sites.

Structured data resides underneath a multitude of other layers, including the infrastructure, applications, and business functions, all of which can be compromised in some way to get at the underlying structured data. In this chapter, we will focus on structured data, which presents the higher risk, but the concepts will also apply to unstructured data. Before I discuss these threats, we should talk about the sensitive data that an organization contains. Every company has large amounts of data, some of which is more sensitive than other bits of information. What happens if this ultra-sensitive data gets into the wrong hands? Will it allow an entity or other terrorist cells to stage an attack on groups of people? Will individuals be physically harmed or killed if the information is leaked? What are the financial impacts upon the organization, in terms of recovery costs and compliance fines? Risk assessment will be discussed in later chapters when we get into the modern-day architecture that is emerging. It may be prudent to think about the data that resides under the architecture in order to understand how each of these risk areas affects your organization.

Data Vulnerabilities Vectors:

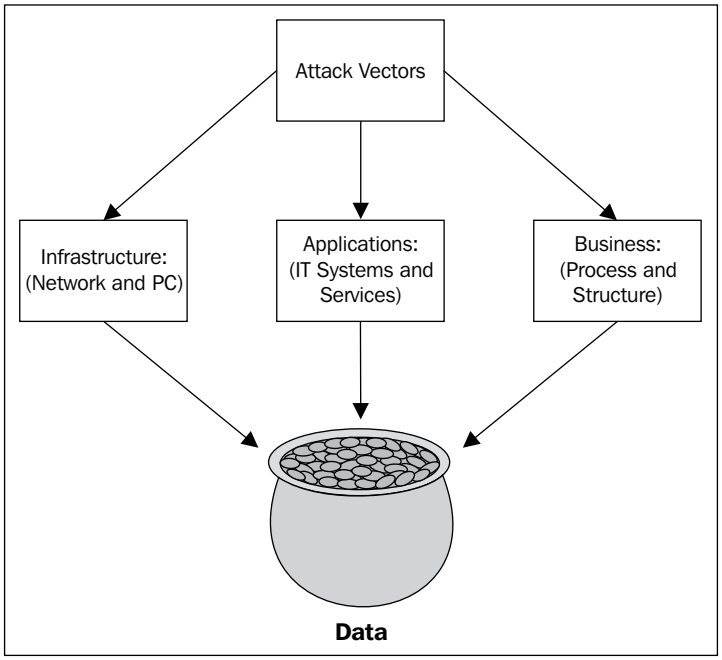


Figure 1. Data Vulnerabilities Points

Data-centric approach

In this chapter, we attempt to understand the current security posture within an organization. Our first order of business is to attempt to identify our current threats by taking a data-centric approach. We could take an application-centric approach, a business process approach, or a network approach. A data-centric approach is the easiest place to start because data is easy to find and, typically, we know where all the databases reside. To do this, we must begin to create a list of all data collections that we have in place today. We need to assess the network topologies, application topology, and business process models that are layered over a company's sensitive data. Some companies have detailed records that diagram the in-house networks, applications, and business processes. Other companies do not maintain them or have them at all. We need to list the data sources with the applications, networks, and business processes and understand their infrastructure. We may need to interview the system owners of the applications to better understand the data sensitivity and its architecture. But this sensitivity is one of the key aspects of any application. It can typically be determined through business owners, business continuity planners, and disaster recovery teams. If those organizations exist within the organization, you may want to reach out to them and find out what modeling has occurred. Chances are, one of those groups have already done this work.

Risk-driven security

Risk-driven security is the answer. We begin by writing down the names of our data sources on index cards and placing them in a stack. Try to estimate the order of importance and put the highest risk on the top and the lowest risk on the bottom. In some cases, we may just quest mate the order. In other cases, we may want to conduct some risk modeling for a more precise answer. We need to take into consideration both the data sensitivity and how vital it is to the operation of the company. The quick way is to determine some kind of a scale that assigns weight to each type of data. For example, name and address information is worth one point, and social security numbers and dates of birth are worth two points. Bank account numbers are worth five points. As you look at each data source, you can determine the importance of this data to run the company on a scale from one to seven. Then, multiply the sensitivity and the importance numbers to determine the data weight. Place that number on the lower right corner of the index card. Then, prioritize the index cards in lowest to highest order.

The bullpen

Find a large wall, white board, or poster board. I prefer poster boards because they can be moved, which is helpful if you are sharing the room with others and do not want to display your entire organization's security secrets for everyone to see. The poster boards are also easy to hide when they are not in use. This wall is called the Primary Target wall. Place the index cards on the wall by using tape, pins, or anything else that sticks. I prefer to use different colored pens so that it is easy to identify the primary targets. Make sure you space your cards out so that you have room to add more cards underneath the primary card and to the right of the index card when we get to the later steps. I like to arrange each data representation from the left to the right, according to the data weight that we derived earlier.

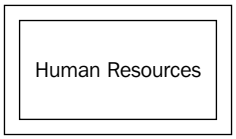


Figure 2. Picture of one of the identified data sources in the bullpen

To the right of each data source, write down the main aspects of the data-supporting infrastructure. This may include the application, the network, business processes, database, and any compliance needs. So, in the case of our human resources data, we have four cards to the right of the human resources card; the first would be for Web-based .NET. The next card would be labeled SQL Database, the next would say IIS server and Windows, and the last card would have Hipaa written on it.

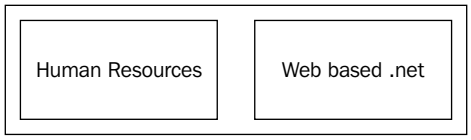


Figure 3. Picture of one of the identified data sources and infrastructure source in the bullpen

Next to the infrastructure cards, we would place our risk-based cards for each of the infrastructure cards. These cards will identify the given risk for each item, and we will need to conduct some risk analysis on each.

Once the security team has completed the mapping to the best of its ability, the security team should call in the system administrators for that given systems to validate the mapping. The system administrators should advise the security team of any misunderstandings in the architecture as well as any security countermeasure that are in place but the security team is not aware of. The opposite is true for any security countermeasure that the security team believes to be in place but removed, uninstalled, or disabled. The system administrators always have an unique insight into the system and sometimes have solutions that can be easily duplicated on other systems.

The next step is to align all the index cards next to each other as perfectly straight as possible. It may make sense to align the cards as you are working. You may need to re-align the cards at this point as well. Then, grab some tape, preferably clear scotch tape, but any other tape will do in a pinch as well. Grab your tape and stick it to the top of your first poster board, and run the tape across all of the cards without ripping the tape. Once all of your poster boards are taped together, you can fold up the poster boards in an accordion-like way. You can then store the business cards easily and later unfold the poster boards in the same positions as last viewed. This is extremely helpful for planning sessions that span multiple days.

After building a bullpen for a major retail customer, we were working with the administrators of a Point of Purchase (POP) system that was recently adopted in an acquisition. The administrators were telling us about the security measures that were in place when we realized the system had a sophisticated Data Loss Prevention (DLP) system in place. The system was a homegrown solution developed by the DBAs and had applicability to other systems. The DLP solution was database specific and would work on all the systems in house. However, we easily identified which other solutions could use the DLP solution and started projects to increase the usage of this tool throughout the organization.

Introduce your compliance team and compliance management team to the bullpen. Compliance teams have tools and procedures for systems affected by compliance. Compliance professionals will also lend great insight into pre-existing security countermeasures and procedures, which makes it important to get them involved in some collaborating around the bullpen.

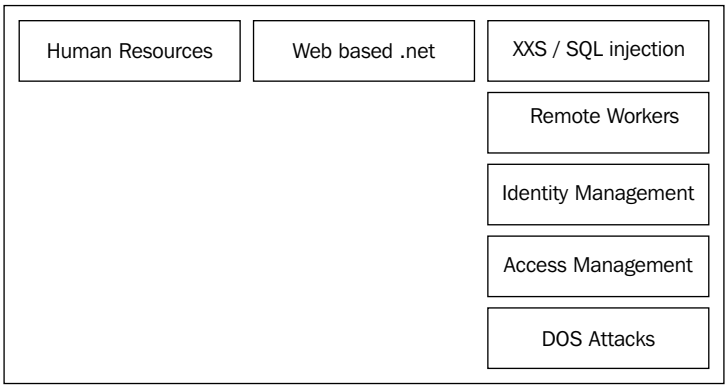


Figure 4. Picture of risk sources added to the bullpen

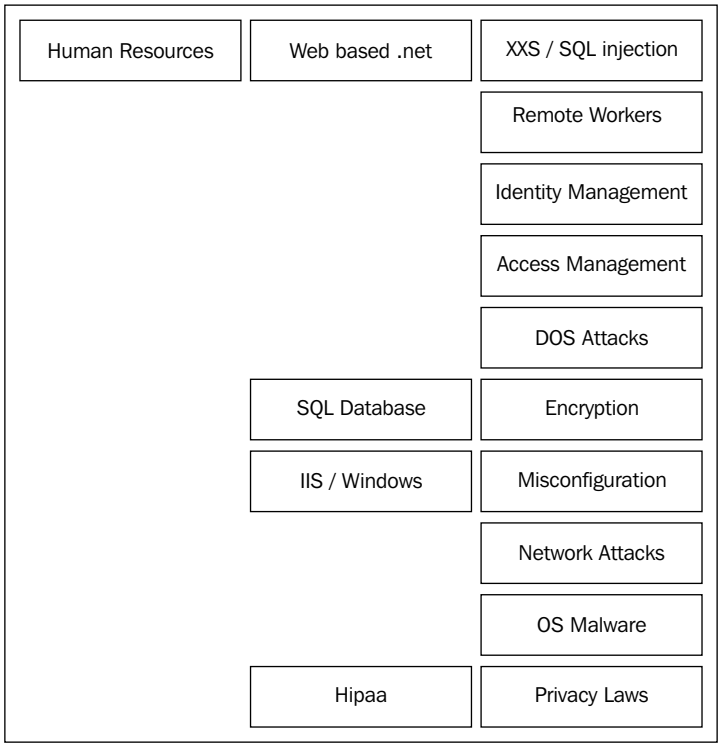


Figure 5. Picture of data source, multiple infrastructure, and risk sources in the bullpen

DREAD modeling

An Agile risk model is designed with simplicity in mind. Other risk modeling formulas exist and, if you are comfortable with the modeling that you are already doing, then by all means, continue with that model. At its heart, risk is a combination of future probabilities, which is impossible to quantify exactly. A simple risk matrix is adequate for most organizations that are not presently conducting any kind of risk management. The most important part of any risk matrix is consistency. We need to be consistent when calculating risk by the values we use. The DREAD model is preferred. DREAD is a way to classify, quantify, and prioritize the risk associated with each risk item. DREAD is an acronym for damage potential, reproducibility, exploitability, affect users, and discoverability.

Damage potential measures how much injury can result from a risk. All assets contain some kind of lost potential. Some points to consider while determining the scale you will use would be customer complaints, regulatory fines, investigation groups to assess incident resolution, and the costs to inform customers of a recent security incident. The second area is the organizational factors, such as how much time the company loses if the system or application goes down. What is the severity of impact during critical times? Lastly, what third environmental factors, such as laws, compliance and media, surround the given data and can damage the organization? Plot this value on a scale of one to ten with the following guidelines:

Damage Potential
0 = Nothing
5 = Individual user data is compromised
10 = Complete system/data destruction, business stops

Chart 4. Damage Potential Index

Reproducibility can be measured by how likely it is to happen and how frequently it will occur. Plot this data on a scale of one to ten according to the following guidelines. What is needed to complete this attack? Do we need to authenticate? If so, do we need administrator rights or can we exploit with regular or no rights? How many steps are needed to realize this risk? When considering the frequency, we are looking at the likelihood of being hacked, for example. We always consider it in terms of one month, one year, or every three years. Whatever we pick, we continue to use the same time.

The other area we need to ponder when considering risk is likelihood, such as how likely it is that the security risk will occur. We identify a timeframe once again. We also considered the chances of being successful, which has a great impact on the security tools that we already have in place. We may find that a system may be hacked 500 times a year, of which one in 100 attempts will be successful. That means we will have five incidents of this risk a year.

Reproducibility
0 = Very hard or impossible, low frequency, administrators would struggle to reproduce
5 = A few steps required, medium frequency, possible authentication
10 = Simple, high frequency, that is, just a web browser

Chart 5. Reproducibility index

Exploitability focuses on the skills and hacker tooling that are needed to exploit the risk. Again, we will plot this data on a scale of one to ten based on skills, tools, difficulty, and the complexity required to exploit the system. This area assumes that a black hat has found this risk and is attempting to gain information or bring the system down. What kind of skills will the hacker need? Will the black hat need programming skills? If so, to what level and languages are the programming skills required? Will the black hat need network knowledge? Will he need to create or use advanced tools in order to exploit the system? Another area to consider is the difficulty versus the complexity of exploiting a given system. For example, encryption is not very complex to hack, but it is very difficult. Attacking even 64-bit encryption can take from two months to 50 years to crack. Not that it's hard, but it takes a long time, unlike complexity, which is something that is very difficult to figure out.

Exploitability
0 = Advanced programming /networking, custom tools, high difficulty/complexity.
5 = Malware, easily performed, existing tools, medium difficulty/complexity.
10 = Just a Web browser, low, no difficulty/complexity.

Chart 6. Exploitability

What is or will be the affected user base? If a black hat finds and exploits this security risk, will some or all of our customers be affected? Will employees be affected by this exploit? Will any third-party users, suppliers, or partners be affected? Plot this data on a scale from one to ten.

Affected Users
0 = None
5 = Some users and employees
10 = All users, employees, and third parties

Chart 7. Affected users

How discoverable is this risk? Would a black hat require inside knowledge of our system? Can this information be guessed? Is the information ratably available? Keep in mind that the value of the data has to be greater than the time needed to hack the system. If the required amount of time is more expensive than the amount that the hacker can get on the open market for the data, the odds of that target being hacked are greatly reduced.

Discoverability
0 = Very hard to impossible; requires inside knowledge
5 = Requires monitoring or guessing
10 = Information is visible on the glass or easily discovered using a search engine

Chart 8. Discoverability index

The DREAD model for assessing risk works like this: Take each number and multiply it **(Damage Potential x Reproducibility x Exploitability x Affected Users x Discoverability)/5**. For example, (7x3x9x4x8)/5 = 1209.6, which we can use to compare this risk to other risks. Also, note that if any of the DREAD formula input values are 0, it will have a rating of 0, which makes sense from a risk perspective. Take this risk value and write it on the lower right corner of the index card. We will later compare this number to the other index cards in order to determine priority.

Bullpen solutions

We have developed threat models for each system that are used to run the business. Now, we must develop our wall of primary solutions further. It should list all tools that we currently have in-house to help secure our IT infrastructure. Solutions such as intruder detection systems, intruder prevention systems, firewalls, VPN solutions, port scanners, virus protection, and so on. In some cases, you will find solutions that are specific to an application or process. In those situations, it makes more sense to list those tools on the primary target wall, usually in different-colored ink, to avoid confusion. We have now created an agile bullpen, which is a central area in which we can see everything that we are working on and all tools we have to defend ourselves. The idea is to unify this information in one place in a highly collaborative environment where security professionals can discuss solutions and directions.

Summary

In this chapter, we discussed the approach an Agile IT security team can use to evaluate and find gaps in the current infrastructure. Using the above techniques, we discovered how to quickly identify our high value assets and find weaknesses in them. We discovered how to determine which gaps pose the most risk to an organization.

In the next chapter, we will look at applying the gaps discovered in the bullpen and look to build an Agile blueprint based on the findings in the bullpen.

6
Agile Blueprint

The agile blueprint is a plan we use to identify potential security holes in the organization. Without blueprinting, we have a bunch of systems and a bunch of applications and tools that are not necessarily mapped in any way, so we could have a tool that would bring benefit to multiple systems, but we only have it implemented in one of them. For example, we may have a number of Web-based applications for our customers and employees to use. We may also have a tool for scanning Web applications for only one Web-based vulnerability. We have gaps in our IT Security infrastructure that can easily be remediated if we could just identify them. The agile security blueprint is designed to do just that.

In this chapter, we will look to extend our bullpen with the use of agile blueprinting. In this chapter, we will:

- Understand how to create an agile blueprint
- Determine what needs to be considered when building an Agile blueprint plan
- Learn how to build a threat model for our high value assets
- Understand how to build an ill-use case model

Agile blueprinting

In Agile, we will look to the bullpen, which details all of our applications, data sources, security threats, and risks for the organizational IT system. We add another layer to identify tools or policies that we have to mitigate those risks. So our wall will look something like this: application, threat (with risk), and mitigation. We may have Web-based systems called human resources, so our first index card, the infrastructure card, would indicate that it is a Web-based tool and our next index card, the risk card, would indicate the threat, such as SQL injection or XXS. Our last card, the mitigation card, would indicate the mitigation device, such as a code scanner. We will then drop to our next risk element and list the mitigation device for that risk and so on down the index cards. In one example below, the bullpen identified HIPAA as a risk factor and the mitigation index card listed policy and the remediation tool. The idea is to list your areas of risk in one area for your applications and look for common vulnerabilities among them. Once these risk areas are identified, mitigating them becomes easier. This is especially true when we see a tool that we already own that can add additional value to the organization without extra cost or performance latencies.

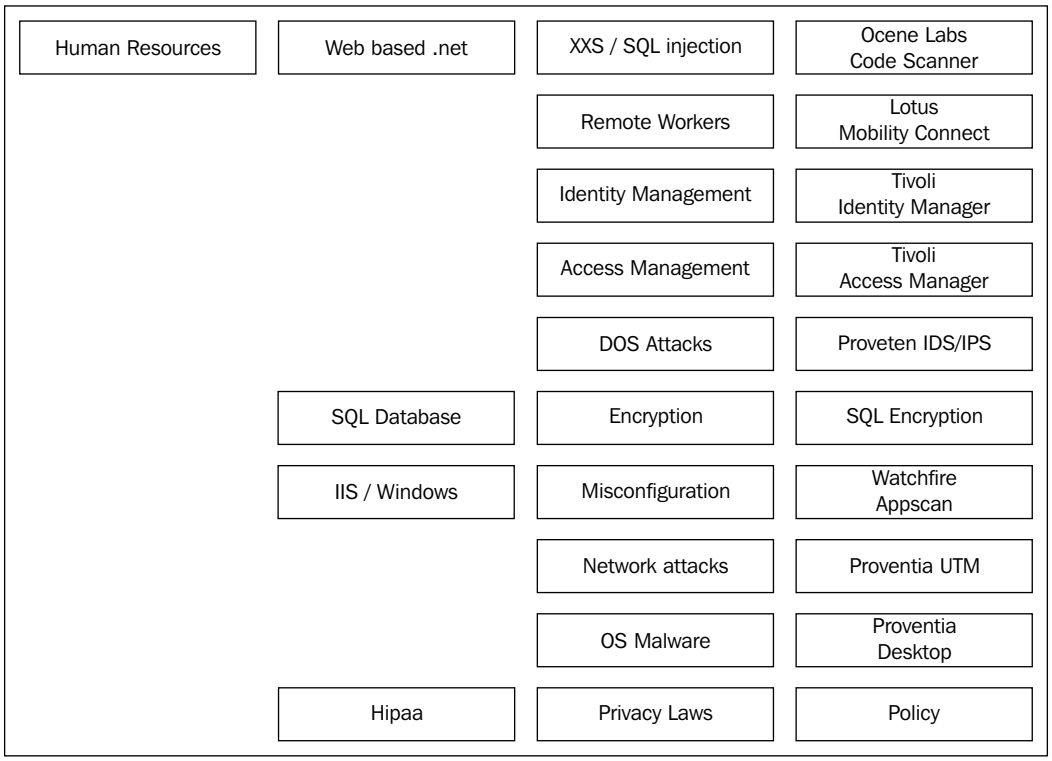


Figure 6. Bullpen with gap analysis

It may be easier to work at the sublayers of each of our data points. For example, let's say we discover that our most critical data and application is our billing system. We also know our billing system consists of a dynamic infrastructure as we use software as a service application to help it to function. We may identify off-boarding, terminated employees as a risk, so we will then identify our identity and access management solution as the tool we use to help mitigate that risk. We may identify risk in data backup and recovery, which may be minimized by our data backup and recovery system we already own for other backup processes. We may later find that other applications and data sources may be able to take advantage of existing security tools that we have not purchased for other applications.

Unfortunately, many corporate security tools are purchased to mitigate risk for a given application and not reapplied to other applications. Some companies could centralize security practices, a technique they do not take advantage of, which results in business owners making decisions on applications without looking at the corporate direction. This sometimes results in multiple tools with similar functions. Then, the company must pay for two tools as well as two sets of resources to maintain them. This scenario can also result from acquisitions and mergers, where a merged company used a different tool to conduct the same function. The consolidation of tooling and the possibility of finding other areas and organization that can reuse the company's existing applications is a big benefit to anyone within the organization.

Accounting for the past

Most companies have had some kind of security breach in the past. It is important to understand what security breaches have occurred, which ones caused the most damage, and which ones garnered media coverage. It is also important to identify which of the security breaches were caused by internal employees or external hackers. This knowledge will also help us to understand which tools and policies we need to help secure the organization from future attacks. The press is typically very unkind to multiple offenses from the same type of attack. Extra emphasis needs to be placed on anything that has already happened in the past. If a hacker has hacked into our organization and uncovered a series of credit card numbers, it is extremely important that we protect the organization from experiencing the same problem.

Threat modeling

We have now identified gaps in what may be considered a system's high-risk areas. Sometimes, certain systems have highly sensitive data that we feel needs to be protected by something besides current-day tooling or policy. Threat modeling fills this gap and we begin with our most valuable item on the primary target wall. Then, we consider all the ways a hacker could compromise this system and list each of the ideas on an index card to be added underneath each of the primary index cards. So, to use a non-IT example, let's say we have a combination safe. I would write "combination safe" on an index card and tape it to my primary wall. Then, I would think of all the ways I could gain unauthorized access to that safe, such as picking the lock. I would add, "pick lock" to an index card and tape it underneath the primary card. I could also bribe the owner of the safe, so I would add an index card with "bribe owner" under the primary card. It would also be possible to blow up the safe. I may add a card with "TNT" under the primary card. We could continue the threat model to include different ways of picking the lock, bribing the owner, and blowing up the safe, if needed. The idea here is to identify additional areas of security which are necessary in order to secure the system.

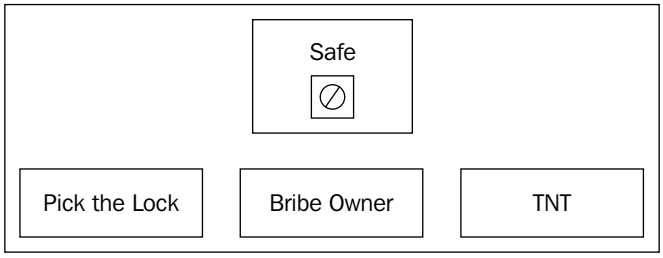


Figure 7. Picture of a simple threat model

Ill-use case

Another option that would appeal to visual people is to represent this information by using ill-use case models to depict the ways a system can be tampered with or altered in order to understand its vulnerability. The process is similar to threat modeling, except that we typically use UML modeling diagrams to portray the actor and the system. UML is not required, but ill-use cases were born from use cases, which are typically UML derived. I will use the example above to illustrate a threat model by using ill-use cases.

Example of an ill-use case:

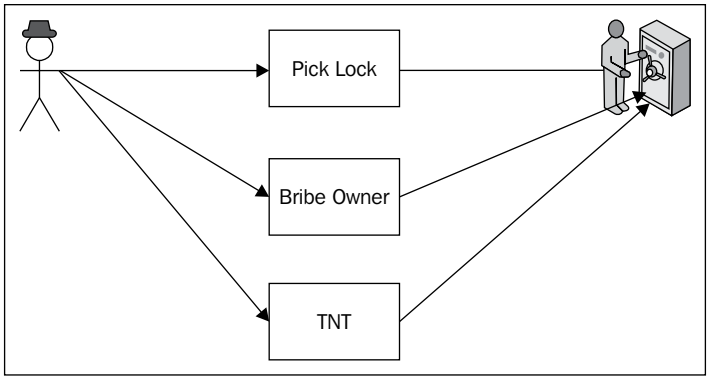


Illustration 1. Example of an ill-use case:

Some IT professionals prefer threat modeling and some prefer ill-use cases. The result is essentially the same for both. However, I recommend the use of just one or the other to ensure consistency between project and security professionals.

Summary

We looked at the various aspects of blueprinting in this chapter. We discovered how to use the bullpen to build an Agile blueprint. We looked the various considerations to take into account when building a blueprint. Additionally, we looked at other ways to determine risk in our high value assets such as threat modeling and ill-use case modeling.

In the next chapter, we will discuss the impacts of Agile on IT security.

7
Lean Implementation Principles

When we look at our implementation processes, we should look for areas where we can improve upon the process. The primary principle of lean is to find and eliminate any artifacts or processes that do not add value. This doesn't mean we should throw away all documentation and formal process, just the ones that add little or no value. This is a practice in which we look for waste, improve learning, and run our projects.

These principles in this chapter come directly from the lean manufacturing process that revolutionized the automotive industry, among other manufacturing industries. Lean manufacturing looks for ways to constantly improve the manufacturing process. Lean manufacturing principles align closely with IT security implementation. In this chapter, we will learn how to:

- Remove unneeded process from our project
- Improve the learning skills of each team member
- Learn when is the best time to make a decision
- Learn the value of delivering as early as possible
- Empower the team members to improve efficiency
- Allow team members to see the project for what it is

Eliminating waste

The first area to consider is the area of inventory. In manufacturing, inventory is the enemy. Inventory can spoil, get damaged, or become obsolete. The reduction or removal of inventory has a lot of benefits for any security implementation team. Inventory started and not done is IT security implementation waste and should be reduced to as little as possible. Just like in manufacturing, extra process, such as paperwork, reports, or red tape that doesn't add any value must be eliminated. If paperwork is mandatory for whatever reason, keep it short and sweet. Overproducing is another form of waste. Lean manufacturers look to remove overproducing because it causes waste. The same is true for IT Security implementations. Over producing in IT Security is simply adding unnecessary features to an implementation. I see this often in identity management implementations, people overly expounding on roles and entitlements to macroscopic detail. I have even seen organizations produce more roles than people in the company. These are all forms of waste. Lean manufacturing looks at transportation as a form of waste, and IT Security implementation also sees having the implementation team assigned too many project or teams as being a form of waste. The time needed for a team member to switch thoughts and move from one thought process to another is wasteful and should be reduced.

Waiting is an obvious form of waste. Delays happen for a number of reasons and should be reduced as much as possible. Also, an obvious form of waste is defect in the solutions that we deliver. Looking to eliminate defects and improperly working security solutions is a fundamental way of removing waste from the security implementation team.

To identify waste, consider the following formula. Make a list of the categories above and write down a few items a week that you observe your team doing. This can be anything from filling out reports, to switching tasks, to work started and not completed. Put the potential waste activity on the list in the appropriate category in a spreadsheet. Next to the name put a short description to help remember the issue in the future. Answer the following questions: how important is this activity to the direct manager? Answer on a scale of 1 to 10, how critical this activity is to the success of your job, 1 being not really important, 10 being extremely valuable. Next, ask yourself how important this activity is to the team members of the implementation team on a scale of 1 to 10. Also, ask how important this activity is to senior management and rate that on a scale of 1 to 10. Add up all the answers and derive a mean score.

For example, the weekly status reports are not really important to the team members (3), and important to the manager of the team (7), and important to the senior management (4). This would give weekly status reports a score of 14. While an activity such as time waiting for network down time will be low to everyone giving it a score of 3. Look for the lowest scoring activities and consider which task your team is spending the most time on. Look to remove or reduce this activity as much as possible.

Waist Activity	Description	Team Member	Manager	Executives	Total
Weekly Reports	This is the time spent producing weekly status reports for our management	3	7	4	14
Network Down Time	Time spent waiting on the network to come back online	1	1	1	3
Extra IP/SP Features	Time spent over developing the …	5	2	1	8

A team that really wants to become lean should consider advanced process analysis, which involves mapping processes in graphical charts. This process looks at every activity in a stream and looks for areas to combine or remove steps in the process. This process makes an organization more lean and agile and adds more value over time.

Amplify learning

When an implementation team starts to have problems and starts to miss deliver dates a quick response is to have the team do better documentation to force the team to have a better handle on the project before we begin. The management team will put more controls around how and when a task can be changed. This practice is bad and only slows a team down more. This makes a team less Agile and simply makes a bad situation worse. A strong principle of Agile IT Security is to amplify learning. Amplified learning can come from using Agile Spikes to understand unknown elements of an upcoming project.

An Agile Spike, as described in *Chapter 4*, is a quick time boxed project designed to reduce risk and amplify learning. This practice can help us learn fast. Also, the use of breaking down allows a team to work on many small projects at one time versus one large project. This smaller project allows for more feedback, which is another way to amplify learning. Another interesting way of amplifying learning is to use pairwise principles. The need for two people working together to synchronize is another way of amplifying learning. As complexity for a project grows, so does the need to synchronize and so does the opportunity to learn.

Decide as late as possible

As solutions become more complex to implement, so does the need for the solution implementation plan to change, which makes the principle of deciding as late as possible another aspect of Agile IT security. This is not an excuse to procrastinate, but a principle based on the JIT, Just In Time, process in manufacturing. Technical professionals have a need to understand every aspect of a project as quickly as possible. The tendency to start developing a detailed aspect of a solution is instinctual to technical folks. This behavior should be avoided because most of the times it pulls focus away from the design. This leads to premature designs and early decision making, which makes us far less Agile when implementing an IT Security countermeasure.

Some decisions must be made early while other decisions should be pushed out. How do we decide which decision should be made early and which decision should be made later? Decisions that significantly reduce breath are good questions to delay as long as possible. Let's say for example I need to build an identity feed from our outsourced human resource department. I have a few options, in which it could build a batch process, federate, or build a hybrid feed. Each decision will work, but when I decide on a solution, my decision breath is significantly reduced. This is a good decision to decide as late as possible on.

Deliver as fast as possible

Solution owners like rapid delivery, and who wouldn't? This may seem contradictory to the last concept to decide as late as possible. But deciding as late as possible helps teams stay flexible and helps teams deliver fast. This is also a direct descendent of JIT (Just in Time) delivery used in manufacturing. With speed comes efficiency; when decisions do need to be made, the likelihood of delaying is diminished because the speed of the project will force rapid decisions when the decision must be made.

This concept is typically achieved by breaking down the solution into smaller deliverables working on the sub projects as quickly as possible. Make a place, online or in a team room, where people can go to view the progress of projects. This should give team members a feel for what's been done, what needs to be done, and what is currently being worked on. When a bottleneck or problem arises that may slow down the delivery of a solution, Agile managers should be quick to build a bottleneck task force. The bottleneck task force's job is to break down the problems as quickly as possible to keep the project moving forward again.

Empowering the team

Use people to their full potential. In a traditional organization, employees are taught to listen to their managers. We are taught this from school and it is applied in the workplace. In the lean process, managers are taught to listen to their team members. This style helps the manager understand problems and issues and allows the manager to become a mentor to the employee and guide the thought process. Use motivation to empower the team. Understand what motivates each person (recognition, time off, flexible schedule, or money) and use that to motivate. Make sure employees feel like they belong. Make employees feel safe. Insecurity stifles productivity and creativity and should be avoided. Promote healthy competition in and outside the team. Healthy competition between teams makes jobs more enjoyable and fun. Competition with our competitors in the industry forms a bond within our team. Both forms of competition are good and should be encouraged. The need for people to see and understand the progress that is made is helpful in building team motivation. Use the concepts above to help empower your team.

See the Whole

The 'See the Whole' principle says that everyone on the team should understand all the principles and practices above. Lean principles are more than the sum of their parts, but the sum total of all the solutions working together in tandem. Team members should understand what the management team is trying to do and should not be left out of the equation. It is important to train people in the practices above, and to coach people on them to help team members work together to make these practices second nature habits to everyone on the team.

Summary

Lean manufacturing is a way of determining what aspects of an assembly line process is wasteful. Once lean identified the wasteful aspects of the process, we work to eliminate them. In addition to waste management, lean principles also give insight on how to manage the projects with better efficiency. Lean manufacturing is an extremely beneficial process used in manufacturing that improves productivity. The correlation to security implementations is extremely relevant and useful for an organization trying to roll out a new security countermeasure. When considering the adoption of Agile IT Security Implementation Methodology, lean principle should be understood and applied.

Consider the principles in *Chapter 4*, *Agile Principles*, and determine which principles work best in your organization and culture. Agile principles work well with other agile principles. A lean principle goes well with other lean principles. No reason exists why the two principles' practices cannot be mixed together, but some organizations tend to adopt more principles from either Agile or lean. Some organization will mix and match the principles together, it's up to you.

8
Agile IT Security Governance and Policy

Organizational policy mandates what in an organization needs securing and is the heart of security. IT security policies will be the driver for the majority of the security projects in most organizations. In this chapter, we will discuss how Agile IT Security can influence the process of policy generations and its effect on IT security governance. The processes in most organizations today surrounding IT security policy is typically restrictive and cumbersome. By applying the knowledge we gained in the previous chapters and considering the information in this chapter, we can learn to create a more effective method of developing security policies and better IT governance. This chapter will help to round out Agile IT security beyond security countermeasures implementation. In this chapter, we will learn:

- More effective ways to develop security policies
- To understand the value of security governance and how it works
- How to articulate the value of IT security countermeasures to help drive the business need for a new countermeasure
- How we can build secondary security policies for better business consumption

Developing security policy

The process for most traditional security practices for creating new security policies in organizations today is typically managed by one or two people. Those people are typically the Chief Information Security Officer CISO and/or the Security Officer, or other executive management. The organization compliance interest, hacker activity, and media typically help to dictate security policies extensively. It is also prudent to consider the internal practices of developing security policies. It is important to examine how new policies are developed to understand how we can do a better job of developing new policies.

Traditional security practices usually involve a Vulnerability Assessment team, which is an important team within the organization. The Vulnerability Assessment team is responsible for finding and maintaining the list of known vulnerabilities throughout the enterprise. Of course, this team must enumerate all the servers, operating systems, networks devices, and so on, throughout the organization. Finally, the responsibility of the Vulnerability Assessment (VA) team is to feed the CISO with known vulnerabilities and fix recommendations. The fix recommendations are typically known as policy recommendations. For example, the VA recommends you put fixpack x.x on that box, and that if you don't, the system will be breached because of this known vulnerability. Another example would be that these servers should be completely separated from other servers, with a firewall established between them.

The CISO or another governing body will decide which of the policies should be implemented and which should be ignored. Risk assessment formulas can be reviewed in *Chapter 5*. Not all organizations take a scientific approach to security risk formulas. The policies that will be adopted are then sent to the operations teams, security team, network team, and the server team for implementation.

The next team that becomes involved is the Intrusion Analysis team. Intrusion analysts are responsible for monitoring the logs and files of the organization to look for any violations of trust. When violations of trust are identified, the intrusion analysis team will report the breach to the Intrusion Response team. The Intrusion Response analyst will conduct a forensic analysis and clone necessary log files for evidence. Furthermore, the Intrusion Response analyst will do whatever is needed to remediate the immediate risk. The Intrusion Response team will then send a policy recommendation back to the CISO or another governing body. The CISO will once again determine whether the policy is to be implemented. The process for submitting new policies resembles what is shown in the following illustration.

Typical new policy process:

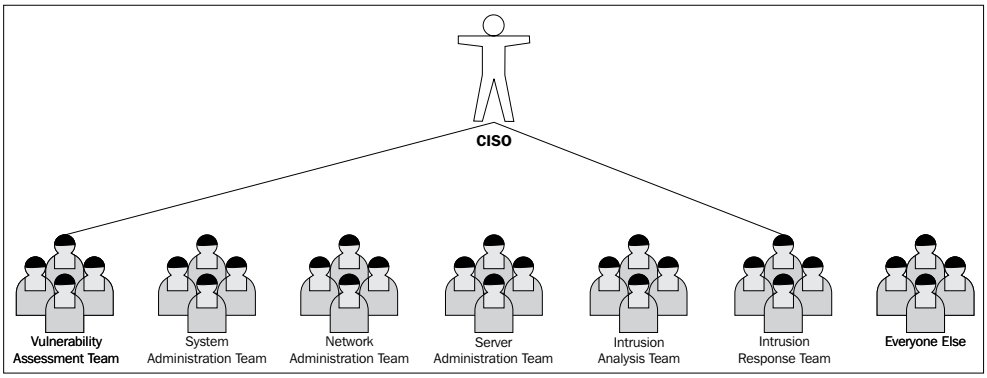


Illustration 1. Typical new policy process

Agile security organizations allow security policy to flow from anywhere in the organization. The best security professional can understand and fill any of the roles above held by the Vulnerability Assessment, Intrusion Analysis, and Intrusion Responses teams. This flexibility in understanding the entire security process allows analysts to make good policy recommendations. Furthermore, system administrators are encouraged to submit fix recommendations as well. We also encourage the flow of policy recommendations to come from anyone throughout the enterprise. The fundamental idea is that the more people we have thinking about security, the more secure we will be. The Agile approach to implementing new security looks like the following illustration. The bullpen, which is discussed in *Chapter 5*, can be a helpful focal point for policy decision-making. Good discussions and ideas can be generated by using the bullpen in conjunction with day-to-day operations, with the idea of generating new policies that will improve the security profile of the organization. The open structure of agile security organizations lends itself well to people working together in a creative, thought-invoking way, which allows for greater security.

Agile new policy process:

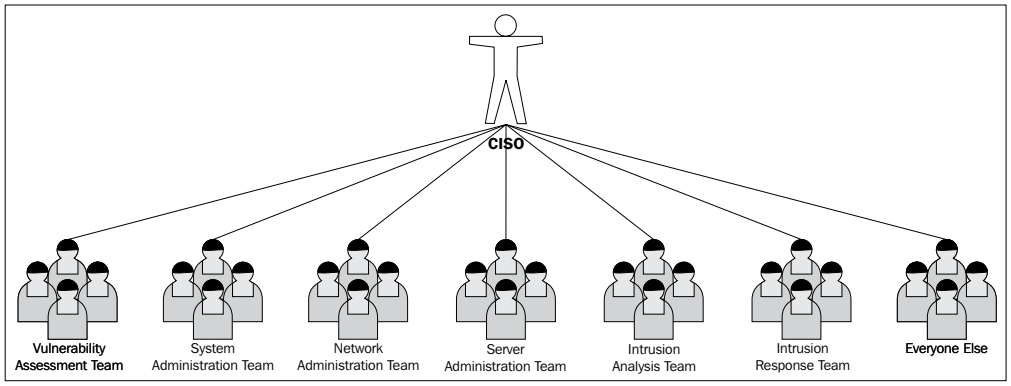


Illustration 2. Agile new policy process

Governance basics

General wisdom may deduce that, since we are talking Agile, an open approach to security, we should not concern ourselves with any kind of governance. This is untrue. In agile, we will try to categorize our security governance into three areas. The U.S. government is comprised of an executive branch, a legislative branch, and the judicial branch. The function of the executive branch is to ensure that laws are carried out and enforced to facilitate the federal government's day-to-day responsibilities.

The legislative branch, as a whole, is charged with passing the national laws the federal government. The judicial branch's primary function is to hear cases and sanction the offenders. Good governance is made up of the same three categories. In agile IT security, we should have three parties: one group to carry out the day-to-day activities of IT security, another to write rules and policies, and a third group to punish anyone who violates the system.

Some questions the legislative branch should be considering every so often to ensure the success of the governance project: Do people inside the organization know what an attack looks like? If so, would they know what to do to help prevent it? Organizations must spend some time in the education of security risks and policies. How often does the legislative branch meet? Do executive level members meet with the team? Is executive management happy with the job the security team is doing? Is security considered a prerequisite to any new projects or initiatives?

The executive team needs to be involved in any new project from the outset. Nothing is more difficult than trying to address security in the final hours of a new project. At that point, the budget is committed, the time has been spent, and the project is moving forward, no matter what. This is frustrating, because building the security in from the outset would have been much easier than addressing major problems a few weeks before the project is set to begin. Trying to stop this project will be impossible; slowing it down a little is all any security officer can do. This is an area in which agile IT security can help to change the mindset of an organization into working with security from the inception of a project.

The judicial branch should also have a team. The consequences for violating IT security should be rendered quickly and efficiently and the judicial branch should be the arena in which to resolve disputes. In short, the judicial branch interprets the laws made by the legislative branch and imposes any sanctions that apply.

Agile IT security professionals know that risk should not rely on the judgment of one person. In some IT organizations, the CTO or CIO is exactly that person. If you find this to be the case, it is important to align yourself as much as possible with any other projects going on in the organization, such as business continuity planning or disaster recovery. The reason is that, in most cases, IT security risks jeopardize the well-being of any organization. Something such as a denial-of-service attack would impact business continuity and could affect disaster recovery. Disaster recovery and business continuity do not typically roll up through a company's technology arm, which could provide another avenue for funding through the line of business.

Articulate security value

It is important for a governance team to be able to articulate the added value that the IT security department contributes. The governance teams will most likely involve executive management and it is important to be able to articulate this value.

It is good to quantify how much systems downtime was minimized because of security practices. Or to determine how conveniences such as self-service password resets are saving the company money by eliminating the need for a helpdesk professional to answer a password reset call. Or, perhaps, the volume of savings associated with automated provisioning of new users or deprovisioning of old users. I know this is a difficult task and we can never really understand the total impact of security. Sometimes, the answer resides in the simple truth that security is like insurance. We all have insurance for our cars, buildings, and lives to help protect the large investment we make in them. Similarly, IT security should be looked at in the same manner, by considering IT security as the insurance policies we need to protect the investment in the systems that run our organization. Either way, a good IT professional always has an answer to the question of the IT security department's value.

Agile second policy

In IT governance, it is important to consider policy. Policy is an important piece of our security posture. Security policy is the one item we have to address areas that we cannot secure with our tooling and countermeasures. Unfortunately, lawyers typically write policy for lawyers. When asked, most employees will say that they have never reviewed the security policies before. This is not that uncommon, since most policies are written for lawyers. In response, many agile IT security groups will write a second policy. So you may be wondering, what's agile thinking by writing two policies? Isn't extra paperwork exactly what we are trying to avoid? The answer is, not at all.

The second policy only focuses on areas that the tooling does not cover. Of course, a second policy will have a disclaimer that the master policy will always trump the secondary policy so that the lawyers do not get upset. The idea behind the second policy is that the security professionals can write it in simpler, more easy-to-understand language, and we can include examples and stories to help make the policy understandable for the average reader. One interesting aspect of policy is that the more policy you have, the less likely people are to read it. So it is definitely advantageous to have a second policy that focuses only on the gaps in our security portfolio that our employees really need to understand.

Summary

Agile IT Security goes beyond security implementations in an effort to help control the overall effects of security governance. By using the techniques discussed in this chapter, an organization can become more effective in the process of developing policies. An organization should use all the resources in the development of security policies. Agile security professionals should be able to articulate the value of any security countermeasures in business terms to help build the case for a new project.

Additionally, security organizations should consider writing secondary policies in simpler terms to help the consumption of policy by an organization's employees. To build on this chapter, the next chapter will discuss security policy and awareness programs that can be used to educate employees on security practices observed in an organization.

9
Security Policy and Agile Awareness Programs

This chapter is intended to help security professionals educate the employees of an enterprise. So much time is spent building countermeasures and securing the perimeter that end user security is often forgotten. As IT evolves and the boundaries of IT change with such advances as cloud, SAAS, and other shared infrastructures, it is important to educate employees on the best practices of security. This is especially the case to protect applications that are out of our physical hands. In this chapter, we will learn how to:

- Build security awareness in an organization
- Use the Ebbinghaus effect to your advantage in building security policy awareness
- Build an agenda for security awareness such as which different policies we need to have
- Use recognition awards and certifications to help educate employees in an organization
- Use Agile memory retention tricks to help increase retention

Security awareness

With all the different risks we face at a number of different layers—protocols, languages, and services on a day-in, day-out basis—it is funny to think that some of our most difficult challenges are educating and informing the organization's employees about security policies and practices to safe guard the company. I am sure that you just chuckled, and I bet you agree with me. Most employees feel that security is a waste of time and regret taking the effort to learn about it and become smarter workers. The truth is that security is everyone's job and we need everyone in the organization on our side.

There are two ways to stop black hats from doing bad things. The first tactic is tooling, but we will identify areas that we do not have tooling to help defend. So the only other way to deter a black hat is through the consequence itself. For example, some people may feel that surfing the Internet on the job is not a big deal. If we change and publicize the risk aspect of such a threat to a job-ending punishment, the likeliness that such behavior will happen will be diminished.

Ebbinghaus effect

Why is it so hard for employees to remember security policies and procedures? Because of Hermann Ebbinghaus! OK, maybe not because of Hermann Ebbinghaus, but his research will shed some light on the topic. Ebbinghaus, a Ph.D. in philosophy, was born in Germany in 1850. He dedicated his life to the understanding of memory and retention. Hermann Ebbinghaus' work later became known as the Ebbinghaus effect and marketers are well aware of his concepts. The Ebbinghaus effect states that 50% of all information is forgotten 24 hours after it was learned. It also states that 90% of all information is forgotten after one week. Ebbinghaus' findings are also known as the forgetting curve. Marketers try to combat this extreme falloff of memory by delivering short, consistent, repetitious messages. If you think of the company Aflac, you may quack. This was not done by accident. It was a marketing project used to reduce the Ebbinghaus effect. Sure, Aflac could have delivered a different message every time it created a commercial, but the marketing team knew that it would be a waste of time because no one would remember all the important details. Instead, the team focused on a simple message and the duck. This duck is an important component in reducing the Ebbinghaus effect. It is a memory enhancer that is designed to revert your memory back to the original message.

Policy awareness

We need to identify and ascertain whether people listen to and understand the current policies and procedures that pertain to security. At this point, we should understand what our tools are and what data we need to protect. We should also understand our policies to some level as well as what tools we have in place that render the policy irrelevant. In other words, if we have good tooling and plays with which to protect our applications, policy is not as important in terms of prevention.

The policy is obviously still important in the rare event that someone circumvents our control. The focus of our efforts should be around the areas of risk that lack protective tooling, for example, the likelihood of a social engineering attack. All companies are susceptible to this type of attack, and we need to understand how such an attack would affect the organization. A survey is one of the best ways to understand our current security posture with respect to the knowledge of our current employees. Tools such as SurveyMonkey.com are great for understanding security policy. Obviously, the key to a successful survey is to garner upper management support in advance and offer incentives to employees to complete it. Some areas to consider include a focus on corporate values and policy feedback to determine what people truly understand about them. This approach will help us to understand the overall strengths and weaknesses of our corporate culture. We want to assess whether employees are generally ethical or not. We would ask questions that focus on simple, day-to-day operations, such as "How many times a day do you check your personal email?" Some questions may center on data policies, such as whether or not employees think it is a big deal to copy information from the organization for a charity event, or how likely it is for a supervisor to request information via email.

How do you build a security awareness program in your organization? Infrastructures have infrastructure firewalls, applications have application firewalls, and businesses should have a firewall too—a human firewall. It is important to understand that the awareness program should reach outside of the company and confront one's awareness of personal Internet security. A breach in an employee's personal Internet use can have serious implications for the organization at which they work. The first topic that needs to be discussed in any security awareness program is why there is a need for a security program in the first place. Among the numerous reasons are: to increase security, protect privacy, reduce risk, and help protect the company and its employees from existing threats.

Password awareness

The next issue to discuss is the need for strong password and authentication practices. Time should be spent explaining how black hats can crack passwords based on computer programs and social networking practices. Following this paragraph are some examples of password best practices.

Next, organizational users should be educated about proper password etiquette, such as not leaving passwords written down in the open. Passwords and toothbrushes should be treated the same way: changed frequently and never shared.

Password best practices
Use both uppercase and lowercase characters
Use letters and special characters when available
Use pass phrases when available
Avoid using the same password over and over again
Avoid using personal information in your password
Avoid using words that can be found in the dictionary

Chart 9. Password best practices index

Another interesting concept is the use of special character buffering in passwords. This helps make easy passwords much more difficult for black hats to guess or crack. The use of a password buffering works by using a simple set of special characters over and over. So if my password was "Dog", I would pad the password with maybe "!@!@!@!@" or '#$#$#$#$' so my password would look like this "Dog!@!@!@!@". The idea is simple, the buffering is easy to add to any password and the special characters make it exponentially more difficult to guess or crack. The use of character buffering will help secure employees passwords' in the new shared infrastructure would we live in.

E-mail, social networking, and IM awareness

As modern-day e-mail, instant messaging, and social networking have made it easier for us to communicate and share information, they have also brought many threats, such as information leakage through viruses and backdoor programs. Awareness should be increased as to how easy it is to circulate digital information. Corporate users should be taught to trust their instincts when it comes to attachments and incoming requests for information. All incoming documents should be scanned before opening. Users should be taught that e-mail, instant messaging, and social networking can easily be spoofed by a black hat, so they should not be afraid to question incoming requests and attachments. Users should be made aware of the best practices for reducing spam, such as reducing the number of people and web pages that have access to your e-mail address, looking at the privacy policy of a website before inputting your e-mail address, looking at the available options for opting out of e-mail notifications whenever possible, and using filters and rules to reduce the number of e-mails received each day.

Employees should be encouraged to open a second e-mail account for non-business-related e-mails and should be made aware of policy restrictions regarding spamming other businesses or people. Some basic education around what a social networking site is may also be good for people who may not be exposed to this new technology. Users should be aware that actions performed on social networking sites may have implications for the corporations they work for. Employees should be encouraged to reduce the amount of information, especially personal information, they reveal on a social networking site. Users should also consider checking the privacy policy for that site and reducing the number of people who can access their information. It should be understood that information posted on any social networking site cannot be removed and is available for all to see. Finally, employees should also be educated on the benefits of using BCC, or blind carbon copy, when e-mailing others, for it helps protect other users' privacy and shows respect from the sender.

Social engineering, phishing, and hoax awareness

Awareness should be built around what a social engineering attack is and what a phishing attack is. A basic defense against these types of attacks is to make sure that employees are suspicious of unsolicited e-mails, instant messages, and phone calls that request information. Employees must understand that management would never ask for certain types of private information via phone or instant messaging and that there could be severe consequences for giving information via those channels. By increasing the stress level related to conveying information, you can decrease the chances of a social engineering attack being successful. Teach employees how to look for malicious websites, how to protect personal and private information by not giving it out via unsolicited requests, and to never follow links in an unsolicited e-mail. If an employee feels, after the fact, that he or she may have been the victim of a social engineering attack, teach the proper response actions to take, such as contacting the manager, contacting IT security, and/or changing passwords and logins. Some consideration may be given to contacting the police or the proper authorities. Employees need to be educated on what hoaxes, chain letters, and urban legends are. The colossal amount of time that can potentially be spent on such things means a great deal of wasted time to organizations. Employees should be taught what stress levels are involved in these types of communications, such as promises of extra luck, financial gains, or love. Teach the employees to ignore these types of requests and search snopes.com for the authenticity of such claims.

Privacy awareness

Employees should be taught the basics of whether they are using a secure layer encryption or an open unencrypted channel for communication, such as http vs. https. They should understand and read the privacy policy of a website before using it and be encouraged to access only reputable companies on the Internet, while limiting the exposure to and number of sites on which they do business in order to reduce their footprint. Many laws and regulations are still being worked out and are not yet fully understood. Caution should be employed when using the Internet for personal and business activities. Excitable data practices and policies surrounding the use and distribution of corporate data should be explained, along with the consequences of data breaches. Examples of how to encrypt data that is being transported on a portable device should be explained and understood by all mobile employees.

Physical awareness

Overlooking physical security is a common practice in many organizations. Without physical security, all the software security in the world will not protect you. Employees should understand which areas of the building are secure and not let people tailgate when entering a building or a secured area. Mobile employees should understand which devices contain sensitive information and are protected with appropriate passwords and encryption. Mobile devices and PDAs should be scrutinized as to which applications can be run on them and how to protect those devices in public areas. Sensitive information should not be viewed on mobile devices, such as laptops or PDAs, in public places due to shoulder surfing. Employees should understand the significance of backing up mobile devices regularly and a policy should be in place regarding laptops and PDAs that are stolen or lost, including the exact steps to be taken when such incidents occur.

Security infrastructure 101 awareness

Employees should be made aware of basic security infrastructure such as firewalls, entry detection systems, intruder prevention systems, and/or virtual private networking. Users should possess a basic high-level understanding of what these devices do and how they protect an organization from attacks. Employees should also understand what virus scanning does and how it protects the employee and the organization from malicious code that can compromise certain systems. Virus protection software can run automatically or, if needed, it can be run manually. Employees should understand how to update the current virus protection patch levels manually, if needed, and know what to do if a virus is found on their systems, including how to clean the system. Employees should be aware of which software packages are used by the company and which are not.

Attack recognition awareness

Employees need to understand when they're under attack and what it looks like, including what a system may do if a virus or worm is detected. Knowledge of what a social engineering attack, a denial of service attack, and a distributed denial of service attack are, is crucial. The network may start to slow down, particular websites may become slow, and employees may not be able to access any network resources. Users should be taught and trained as to the correct procedures for dealing with such an attack, such as contacting IT security and management about the problem.

Awareness certification

Some companies even take the extra step of distributing some kind of certificate to employees once the certification is complete. Having a certification path in place helps to reinforce the importance of this education to the employees and helps build employee pride in the program. Some organizations may consider requiring a public certification, such as Security|5, which is available through the EC-Council at `http://www.eccouncil.org`.

Memory retention

As agile IT Security professionals, we must find ways to increase the retention of our security seminars, workshops, and events. When conducting such events, we need to spend some time to determine what the high-level messaging will be. Instead of focusing on all the details that we can fit into the event, we should focus on delivering a clean, concise message. Another way to reduce the Ebbinghaus effect is to deliver the message repeatedly. By emphasizing the key messages over and over again, we can hope for better retention. Another idea to increase the retention of security events is to have a good slogan or catchphrase that people can easily remember. Catchphrases can be an important memory enhancer. Also, try to provide giveaways and incentive items in order to reduce the Ebbinghaus effect. Pens, buttons, t-shirts, or polo shirts are generally inexpensive and can help people to recall our slogans or catch phrases. Consider using short videos and brief messaging to deliver your points. Most younger-generation employees prefer short versus long messaging and they typically need to understand the messaging more than the seasoned employees. I have even seen some organizations take their slogans and incorporate them into the startup sound of the PC. We need to carefully select the messages we want to deliver and communicate them early and often.

Summary

In this chapter, we discussed the basics of building awareness for employees in an organization. We talked about the importance of security awareness in an organization. We discussed what the Ebbinghaus effect is and how it works. We learned how to use the Ebbinghaus effect to our advantage when educating employees in an organization. This chapter also discussed the need for various awareness programs such as password, email, social networking, IM awareness along with social engineering, phishing, and hoax awareness programs. We discussed creative ways of how to educate employees in these subjects. We discussed physical and architectural awareness programs. In the next chapter, we will discuss smart agile planning techniques and risk spreading.

10
Impact on IT Security

We'll learn the importance of agile structure as it relates to risk in an organization in this chapter. We will focus on the importance of spreading risk and the impact on IT. We will focus on disaster recovery and how it relates to security. We will also discuss the relationship security has with supply chain management.

Agile structure

Working smart is a critical aspect of agile IT security. Smart agile planning means that we shouldn't have the risk decisions being made by one executive, such as the CIO or CTO. Sending all IT risks through one senior executive is a significant danger to any company. A good security organization includes members from all areas of the business. Obviously, its members would report through the IT management structure, but security people from the line of business should report through the appropriate reporting chain. Furthermore, if possible, include representatives from the finance department that report through its specific departmental structure. The idea is to spread the burden of risk mitigation between the different members of the senior executive team. Risks such as DoS, or Denial of Services attacks, may be considered more sensitive to a line-of-business executive than an IT executive. The key idea is to spread the burden of risk remediation between the executives.

Spreading risk

In previous chapters, we discussed why it was unwise to allow all risk to lie on the shoulders of one individual, such as the CIO or CTO. What can you do if your organization is unable or unwilling to change?

The next best thing is to work with other teams that share responsibilities that are similar to IT security tasks. These teams include the privacy team, the compliance team, the disaster recovery team, and business continuity planners. The idea is to find out which initiatives they are working on and align IT security with the other team's risk. Each of the aforementioned teams are each put into place to remediate risk to an organization. Best practices call for the alignment of IT risk with business risk to help the executive management team to better understand the impact of the risks that the organization is facing.

Disaster recovery teams are tasked with understanding the impact of different types of disasters, such as pandemics, hurricanes, floods, or chemical spills. Disaster can take the form of cyber attacks and corporate espionage. It is important to work closely with this team, since IT security can play a major role in preventing and recovering from disasters. Most disaster recovery teams report through the line of business and may offer a good avenue for remediating risk through a different line of management.

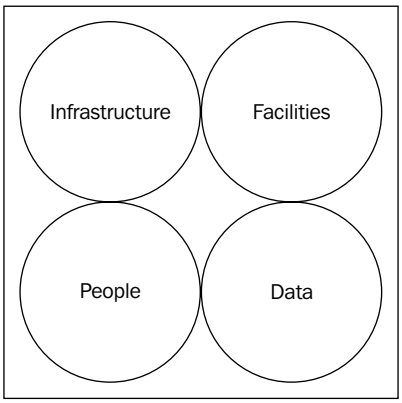


Illustration 4. Disaster recovery areas

A business continuity planner's job is to minimize downtime and keep an organization operational during a disaster. This task may seem similar to disaster recovery, but it is different in that disaster recovery looks for disaster and remediates the impact, while business continuity focuses on ways to maintain and keep the business running during times of disaster. The needs of the business continuity planner pertain to maintaining the organization and IT security and provide people with proper access and controls when faced with freakish disasters which, again, may offer additional avenues for risk remediation and budget.

I have come to think of IT security, business continuity, and business recovery all grouped together nicely in one umbrella called organization resiliency. Business continuity and disaster recovery in the past focused on the technology side of the business.

In the past, the concentration was more centered on how we get the business' networks and servers back up and running in a timely and expedient manner. The industry has shifted focus from infrastructure to also include facilities, people, and data. If any one of the four elements is missing, none of the other elements has any value.

Since it's rather impossible to plan for every possible disaster that could happen like earthquakes, chemical spills, pandemics, and alien invasions, an agile best practice in disaster recovery is to plan for a worst-case scenario. Plan for a complete system outage and identify the individual components of the disaster recovery plan for infrastructure, facilities, people, and data. The sub plans fit together in what can be called a framework. Organizations can implement the elements of the framework necessary to overcome the disasters at hand. Test out different scenarios against the framework to understand what additional backup contingency plans need to be added to the framework.

External black hats sense weakness and look for companies in a weakened or compromised state. When a black hat discovers an organization is having problems, it's an open invitation for black hats to start trying to penetrate it, because black hats understand that security measures may not be a priority system to get into place and recognize this opportunity to establish a foot hold inside an organization's network. Organizations may want to consider the transparency of any disaster until security is completely restored if possible.

Compliance and privacy

Compliance and privacy teams are tasked with making sure that the auditors are happy. They usually report through the line of business, which means that they operate under another management chain besides IT security. Compliance teams will not have security risks, but they will need to track security-related issues. Therefore, security risk plus audit risk may make projects, such as monitoring controls, easier to budget. In addition, this provides IT security with additional means of elevating risk remediation through the line of business.

Agile can be used in determining the current gaps in security compliance in terms of regulatory compliance. Depending on the regulatory compliance we are trying to adhere to, we can create a list of high level requirements on 3" x 5" index cards. It's always best to look at the regulatory compliance closely. Regulatory compliance regulations can vary significantly from organization to organization. For example, PCI compliance can vary depending on the dollar amount of the transaction an organization realizes. PCI has more regulations for organizations that do more credit card transactions than smaller organizations.

Once the requirements are understood, we can proceed to list them and build a bullpen. With all of our requirements written on 3" x 5" index cards, we can arrange them linearly across the top of the poster boards. This practice is similar to the bullpen we set up in Chapter 5, which should be referred to if the practices are not understood. Once the cards are up, we can write on 3" x 5" index cards all the systems that are impacted by the regulatory compliance. This system should be listed from top to bottom on the left side of the poster board. Once our matrix is in place, we can fill in the matrix collaboratively with the security counter measure we have in place to fulfill these requirements.

When looking at these counter measures, discussions should take place about how good of a job this counter measure is doing. Discussions should continue about how auditable the countermeasure is, if this countermeasure is performing its function well and is auditable. Then, time should be given to looking at the feasibility of using that same countermeasure for other systems that have gaps. On the other hand, if a system is doing a poor job or it is not easily auditable, then time should be spent on considering how to improve that countermeasure to become more auditable. In some cases, it may be prudent to replace the countermeasure with something more auditable. You may discover you have other tooling that can be easily replaced by other tooling. Once the gaps are understood, we can start to build a new project to fill them in.

Supply chain

Another area that merits consideration is the corporate supply chain, which is critical to a company's well-being. Today, most companies operate under the assumption of just-in-time inventory, which is a very efficient and effective way to run an organization. We currently face unprecedented times, considering the reduced amount of inventory that companies require in order to operate. Companies work so efficiently today that they do not need large amounts of inventory. But this trend also poses a threat from a resource perspective.

The first threat comes in terms of protecting the supply chain from black hats, since they could attempt to injure the organization through the supply chain. As IT systems lay foundations between suppliers and organizations, it is important to consider which suppliers are most critical to the organization and understand the security practices of the supplier. To do this, we will pick another wall and use it to list our suppliers on index cards. This wall will be called the Primary Suppliers wall. We will arrange the index cards according to the supplier's importance to our organization. Under each card, we may want to include additional cards that display which applications and IT services our organizations use to run our company. We may also want to identify alternative suppliers in case something happens to one of our critical suppliers. We may need to assist a particular supplier with our security practice efforts, or we may want to consider removing suppliers that have not implemented sound security practices. Therefore, we first need to find any information we have on our suppliers with respect to their importance in running the company. Again, we should look to disaster recovery teams and/or business continuity planners to assist us, since they have most likely conducted this type of planning for us. They can also help us to get a head start on the security effort.

Summary

In this chapter, we discussed the further implications of security , how security needs to work in conjunction with other entities both inside and outside of an organization. We learned how agile techniques could be extended to help coordinate the relationships with such things as disaster recovery and in the supply chain. When planning security in an organization the above consideration should be taken into account.

Next, we will discuss how to remove common barriers to a successful agile IT security rollout.

11
Barriers to Agile

With any change comes resistance. People typically do not like the unknown and tend to resist any type of organizational change. This chapter will focus on the common barriers to any Agile and how to offset this resistance. This brief chapter will discuss some key aspects to consider that will improve early success with Agile. In this chapter, we will discuss:

- How to make Agile culture more affluent
- The importance of training in Agile
- How to offset people's fears about Agile

Agile culture

Why do organizations fail at Agile? Agile is a cultural change, so although it may help to adopt just a few agile practices at a time, it is crucial to understand and implement the whole package. When you look at agile from a distance, you will notice that agile constitutes a blend of people, planning, and risks that are balanced in perfect harmony. It is that blending that helps the combination to become successful.

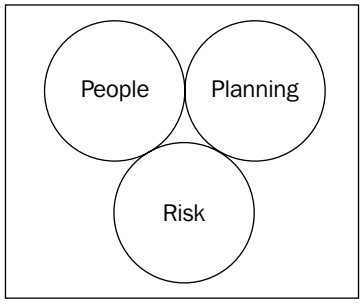


Illustration 5. Culture balance

Many IT security tends to revert to old patterns; in other words, old habits die hard. Such is true for people who are looking to move from a more traditional security practice that may have been in operation for years. Management and security professionals may be likely to resort to old tendencies when heading down the new path. Despite resistance to the new change, it is important to allow this philosophy some time to blossom.

Agile training

Lack of training is one barrier that prevents a company from being agile. Training security professionals on different security spheres is helpful to their personal development as well as that of the overall security team. Security professionals who are strong in endpoint security may want to take a class on another discipline, such as database or application security. Each security professional is responsible for the development of his or her own career, but is important for an organization to support employees in their efforts to learn and grow.

Agile fears

In the early phases of implementing agile, it is important to discuss the fears associated with moving to this new style of management. Teach the value of openness in communication and in vocalizing your fears. Do not force anyone to do anything that they do not want to do. Such force is not an agile practice and will only generate greater resentment toward the new practices.

Early in the agile process, it is important to make sure that everyone understands that the team is co-owning security business in the organization. Reemphasize that everyone is important and promptly celebrate each small success. This, for most organizations, will be a complete cultural change which people will resist. With patience and understanding, the fears will subside.

Summary

In this chapter, we discussed some consideration points to improve the success of Agile in any organization. Consider the organization's culture and how the principle and practices of this book will be received. Also consider training options and to help offset the fears people will have when changing the culture of the business unit. Spending a little time reducing fears can go a long way in the adoption process of Agile. In Chapter 12, we will discuss additional planning techniques using Mind Mapping.

12
Agile Planning Techniques

One of the simplest and easiest ways to take notes in a meeting is to mind-map them. Today, the leading technique with which to capture meeting notes is to record them with pen and ink. Some people attempt to type notes directly onto their PCs, but with all potential distractions, such as games, sports information, and news, most people who have the lid up on a PC are considered to lack participation in the conversation. Writing top to bottom worked well in college, when a professor was simply dictating his linear subject matter to his students and everything was well-organized and easy to consume. The problem in the corporate world is that information is presented in a circular fashion.

Mind-map example

Let's consider a sports conversation between Bob and Pete, coworkers at ABC Corporation, before a meeting on Monday. I will mind-map the conversation later in the chapter to demonstrate how it comes together. "Did you watch any of the games over the weekend, Pete?" asked Bob. "I did catch a few games, but I had a lot of yard work this weekend." Bob continued, "How about the Stingers hockey game on Friday night. Did you see that bad call in the third period? Those refs blew that call." Pete said, "I didn't see that game, but I did catch the Cougars football game on Saturday. Boy, that team is looking good. Did you see our new running back run for 200 yards?" "Yes, I did," said Bob. "It was an incredible game. He's lucky he has such good offensive line." "Wish we could say the same thing about the Dragons, our pro football team," added Pete. "Yep, injuries are killing 'em. You can't win if you don't stay healthy," said Bob. "Speaking of injuries, I saw on the Sunday news that the new rookie for the Stingers will be out for three weeks with a knee injury. Did you see him get hurt?" Pete asked. "Yes I did, and it was a clean play. He just came down funny on the knee, similar to that Big Fish baseball game when Smith stole second and came down funny on his knee," continued Bob. "Oh yeah, I remember that. Say, those Big Fish have an incredible pitching core this year…"

If we were taking notes on this conversation, it is clear that we would have difficulties listing the information from top to bottom. Since paper and ink are still the preferred note taking approach, consider a new alternative. Simply draw a large circle on your paper and write the main topic of the meeting in the middle. In the preceding example, the topic would be sports. As the conversation moved to a new topic, I would draw another circle with the new topic in it. According to our sports conversation example, I would write "hockey" and include another circle labeled "Stingers" next to that. Next to the Stingers circle, I would circle all the items about the team specifics, such as, in this case, "Bad calls." (See the following figure). After that, the conversation jumps back to sports, but now we are talking football. So we go back to our original bubble and we add a new, secondary bubble named "football" and a third-layer bubble named "Cougars." Next to "Cougar," we could put two bubbles that say, "good running game" and "good offensive line" (see figure 9). At the third level, we would put the pro team's name down and add another bubble with "injuries" in it. But now we head back to the hockey discussion and we simply add another bubble to the "Stingers" bubble to depict the jump back in the conversation. Lastly, we would add "baseball" and a third bubble with Big Fish" on it (see figure 10).

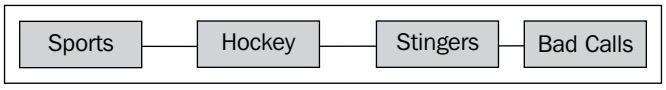


Figure 8. . Mind Mapping Example for Hockey

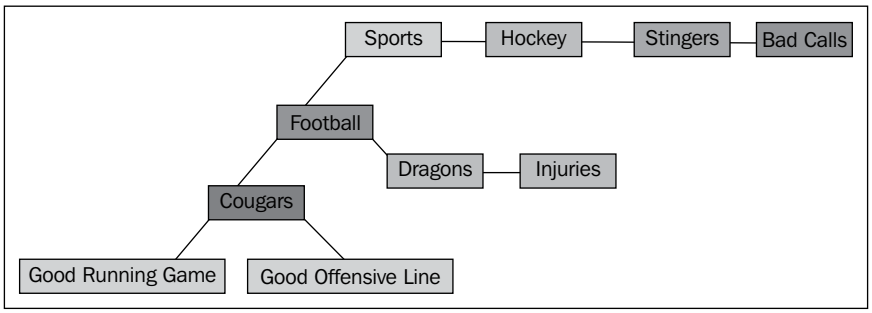


Figure 9. mind mapping example of hockey and football

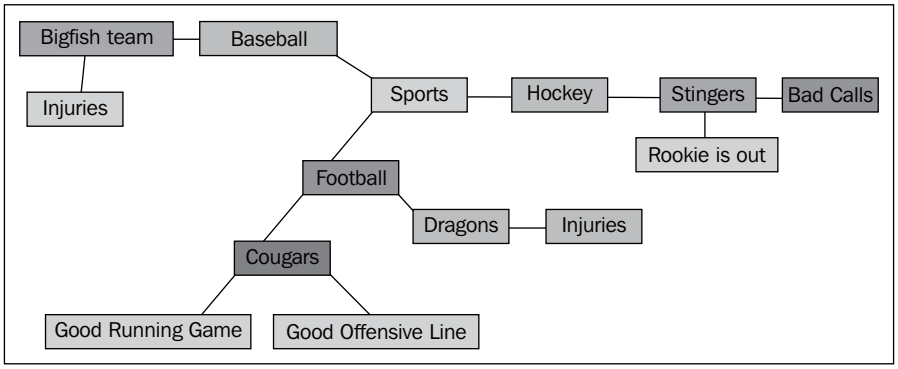


Figure 10. Example of mind mapping with hockey, football, and baseball

Mind-map tools

Mind mapping is a great tool with which to record the conversations you are having. As the conversation jumps, your notes jump as well. Mind mapping can be done on paper, but sometimes you may want to transfer your documents to a computer. Some specialized software is available for mind mapping. If you have it, Visio is a good mind mapping tool, as is `http://www.bubbl.us`. It is an easy-to-use, free process flow diagramming tool that works well for mind mapping. Also, freemind is a great mind mapping tool you can find at `http://freemind.sourceforge.net/` and runs on Windows, OS X, and Linux. A mind-mapping framework was developed for penetration testing. This framework is helpful in the reconnaissance phase of penetration testing and can be downloaded from here : `http://vulnerabilityassessment.co.uk/`.

Summary

Familiarity with mind mapping techniques and principles is great for any planning initiative. Mind mapping is easy and useful for all professionals who need to plan. Consider using mind mapping in your next planning session and see if your notes flow better. I am still amazed at how well the mind map helps me recall not just the important information but additionally the flow of the conversation. This has come in handy on a number of occasions.

In *Chapter 13*, we will discuss the Agile impact of compliance.

13
Compliance and Agile

Organizations seem to get in the most trouble with compliance in year two. The reason is that most companies address compliance only when there is the immediate danger of a Qualified Security Auditor, or QSA, about to walk through the door. The weeks or months preceding the auditor's visit, the members of the organization work valiantly to piece together a makeshift solution for the auditor. This practice is similar to a college student cramming for the final exam the night before.

This approach works for the first year because most QSA are lenient then. The QSA is less likely to fine someone on a first audit, but simply make note of it and check again the next year. Year two is when the organization determines that everything went fine the year before and decides to hold the status quo. The problem is when the QSA has discovered little progress has been made on compliance monitoring in year two and the QSA will typically come down hard on the organization. This practice is all too common and part of the compliance maturity.

Another interesting topic is how an organization should approach security. Should an organization secure itself with security-based practices and then address the security compliance regulation? Or should an organization secure itself with the compliance regulations in mind and build security practices around the regulation that it uncovers? Organizations are far better off when they design solid security practices and principles and have solid roadmaps for future security measures and implement additional security measures when needed to fill gaps. The other practice is a formula for weird security practices that ensure both small and big headaches. Agile IT Security is about sound security practices that protect the company from real user and auditor threats.

Agile compliance

In agile, we will look again to the bullpen, which we discussed in *Chapter 5* and *Chapter 6*. The bullpen details all of our applications, data sources, security threats, and risks for the organizational IT system. We add additional layers to identify tools or policies that we have to mitigate that risk. Now that we are looking at an application, threat (with risk), and mitigation we have the footprint to understand our organization compliance better. Once we have this overview on the poster boards on the walls we can review the various levels of infrastructure we have in place. We will want to review the architecture and countermeasures from the holistic compliance stand points. We will want to look at the bullpen from a couple of perspectives. First, we will obviously want to look for enterprise gaps in compliance across the enterprise. Auditors aren't necessarily looking for answers to every compliance issue, but rather that an organization understands the compliance gaps and is working to remediate them. The bullpen is also a point of reuse. We should study and understand the bullpen from a reuse perspective. When we identify a gap in a given application, we should look for other applications with a similar gap that has a resolution to it. When this type of relationship is found we should try to reuse the countermeasure. Reusing a countermeasure typically costs less than purchasing new countermeasures and reduces the skillsets required for a given organization.

Summary

Depending on the scope and complexity of your organizational compliance needs, Agile compliance techniques can be useful. For large scale compliance needs that involve numerous regulatory requirements, Agile techniques can easily be overwhelming. For small to medium scale compliance and organizations using Agile IT security for implementation, this is a natural progression for Agile. Compliance can be demanding, but process and structure is a key aspect of compliance. QSA professionals love to see process even if it's a lightweight Agile process.

14
Effective Agile IT Security

College students in the early 1900s were discouraged from entering the field of physics because most experts believed that the field was completely understood except for a couple of unanswered questions. Albert Einstein ignored the conventional wisdom and changed physics forever. I'm sure people feel the same way about IT Security: for the most part everything is figured out and not much else needs to be done. IT Security is a constantly evolving field. We have only just begun to understand the field of IT Security and I think this field needs to continue to incorporate new and unconventional wisdom.

As a penetration tester, I worked with numerous development teams that had adopted some form of Agile software development methodology. I worked most often with Extreme Programming, Scrum, or OpenUp, as well as several hybrid agile shops. I learned lessons, not only from my experience, but also from others in the agile community. Since my first experience with agile, I have watched the development communities prosper through the application of agile practices and methodologies.

At first, I was quite opposed to agile methodologies. Why wouldn't I be, when they were the opposite of everything that I had been taught in the formal processes world? My perspective was that the only way to accomplish a project was to surround it with a considerable amount of process and ceremony. I was comfortable with my highly process-oriented life. As a member of the IBM Rational team, I was certain that the Rational Unified Process (RUP) was the best and that nothing else would do. As I became more familiar with Extreme Programming and other methodologies and begin to work with teams to implement this process, I was amazed by its effectiveness and efficiency. What was more surprising was that the people who were involved in the process had a high degree of respect for the work they were doing and felt more self empowered. Thankfully, Rational introduced OpenUp to the open source community as its form of Agile.

Agile team success factors

Agile methodology attacks risk at its core. To identify risk early and often is the premise of agile security. Risk comes in many forms. Organizations face security threats every day. Risks are in the delivery and maintenance of the countermeasures. Agile IT Security focuses on the details of the steps in delivering and maintaining IT security measures while keeping the big picture in mind. This allows us to see the forest for the trees and deliver more value in our day-to-day activities.

Everyone on the team matters, and everyone counts. Some of the fundamental elements of Agile IT Security are involving the whole team to look at problems and generating solutions when problems arise. This means recognizing and talking about the team's fears. Agile teams adopt a whole team approach; this makes IT Security more fulfilling and team members work harder and feel happier and healthier. Agile teams conduct meetings standing up because this process will reduce the meeting to its core, which hurries up the meeting. Stand up meetings are strictly limited to 15 minutes. Stand up meetings increase the team's productivity and allow the team to deliver more value to the organization.

When executing tasks on an agile team it's important to work in pairs. Two people working on a task require mutual understanding and knowledge. If your goal is to reduce risk and secure the IT environment, this greatly reduces the likelihood of error. The pairwise methodology supports the whole team approach because two people on the team understand the countermeasures in place, which increases the breadth and depth of the team.

With the agile methodology, we approach projects from the aspect of refractoring. Refractoring is the reduction of new project deliverables—such as implementing access controls or correlating information between event sources—down to the smallest possible deliverables so we can show progress. Refractoring keeps motivation high because a section of the project is completed every few weeks. We do the same with security countermeasures, decomposing them from time to time to see if this project can be simplified. Every once in a while we look at the processes we use to secure our databases, for example, and determine whether there are any ways to make this process simpler, more secure, or better performing.

Agile risk success factors

We have agile tools in our toolbox for minimizing risk. The formula we use to calculate risk is damage potential, reproducibility, exploitability, affect users, and discoverability (DREAD). This formula determines risk and which security gaps should be addressed when.

Our bullpen is a key area in the agile methodology because this is where our team collaborates and identifies the gaps in our security posture. It's like our security dashboard that is designed to facilitate a whole team approach. I recommend reading the chapter on risk-driven agile security for more information on the bullpen and DREAD modeling.

Agile security recognizes that the security policy is a key way to reduce security risks that organizations face from internal threats. It's best to write a second policy that is easier for employees to understand. Heavy use of memory aids and simple reminders of security policy are some ways that we can improve employees' ability to keep these policies in mind.

Factors in the success of Agile countermeasures

The world of IT Security is changing with the new and emerging technologies such as cloud computing, Web 2.0, and increased network traffic. Mobile workers, contractors, and outsource teams are redefining the structure and layout of many organizations. IT security faces the risk of people, devices, and servers being located all over the world. New forms of attacks are focusing the business layer with social engineered attacks set to deceive our employees into giving up sensitive information and disrupting the workplace.

The traditional security countermeasures of server, network, and endpoint security are not enough to defend against modern-day threats. Additional identity countermeasures and database countermeasures, application countermeasures and physical security countermeasures need to be considered when planning our defensive posture. We should have tooling from all five areas of security to protect an organization from the threats and risks that it faces.

I feel the average IT security professional is outgunned, much like Davy Crockett at the Alamo. The number of vulnerabilities is growing at an alarming rate every year and an average IT security department budget is lucky to see any annual growth. With this modest, if any, increase in budget, organizations continue to ask the IT security team for increased protection from year to year. I wrote this book to present some of the techniques that I learned in software development as suggested best practices in the security world. My goal is to help IT professionals become more successful in delivering business value and to better align the IT security team with the organization's goals and directions. I hope that the fruits of my experience help to provide ideas for IT security groups around the world.

Most security professionals today need to be good in many areas in order to be successful, but hackers can succeed by only being good in one or two areas. Agile helps level the playing field for the good guys.

Summary

In this chapter, we reviewed the process of Agile IT security implementation methodology. This chapter reviewed the lessons learned in previous chapters. After reading this chapter, if you feel you didn't understand the concepts of a particular principle or practice, feel free to go back and review that chapter. The Agile principles discussed in this book are easy and simple to understand, but the applied approach can be different for some people. If you want to understand Agile better it may be best to join an Agile software development user group in your area. Although Agile software development is different than Agile IT security techniques, Agile principles and practices are similar and much can be learned from the Agile software development. A lot of the core concepts are the same, which makes software a great source of knowledge for Agile IT security.

Index

A

B

C

D

E

F

G

H

I

J

L

M

T

U

V

W

X

Y

Z

Thank you for buying
Agile IT security
Implementation Methodology

About Packt Publishing

Packt, pronounced 'packed', published its first book "Mastering phpMyAdmin for Effective MySQL Management" in April 2004 and subsequently continued to specialize in publishing highly focused books on specific technologies and solutions.

Our books and publications share the experiences of your fellow IT professionals in adapting and customizing today's systems, applications, and frameworks. Our solution based books give you the knowledge and power to customize the software and technologies you're using to get the job done. Packt books are more specific and less general than the IT books you have seen in the past. Our unique business model allows us to bring you more focused information, giving you more of what you need to know, and less of what you don't.

Packt is a modern, yet unique publishing company, which focuses on producing quality, cutting-edge books for communities of developers, administrators, and newbies alike. For more information, please visit our website: `www.packtpub.com`.

About Packt Enterprise

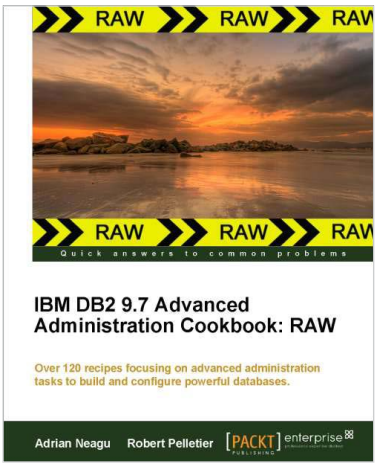
In 2010, Packt launched two new brands, Packt Enterprise and Packt Open Source, in order to continue its focus on specialization. This book is part of the Packt Enterprise brand, home to books published on enterprise software – software created by major vendors, including (but not limited to) IBM, Microsoft and Oracle, often for use in other corporations. Its titles will offer information relevant to a range of users of this software, including administrators, developers, architects, and end users.

Writing for Packt

We welcome all inquiries from people who are interested in authoring. Book proposals should be sent to author@packtpub.com. If your book idea is still at an early stage and you would like to discuss it first before writing a formal book proposal, contact us; one of our commissioning editors will get in touch with you.

We're not just looking for published authors; if you have strong technical skills but no writing experience, our experienced editors can help you develop a writing career, or simply get some additional reward for your expertise.

[PACKT] enterprise
professional expertise distilled
PUBLISHING



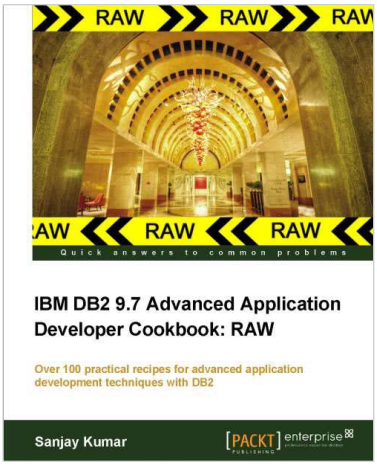
IBM DB2 9.7 Advanced Administration Cookbook : RAW

ISBN: 978-1-849683-32-6 Paperback: 420 pages

Over 120 recipes focusing on advanced administration tasks to build and configure powerful databases

1. Master all the important aspects of administration from instances to IBM's newest High Availability technology pureScale with this book and e-book.
2. Learn to implement key security features to harden your database's security against hackers and intruders.
3. Empower your databases by building efficient data configuration using MDC and clustered tables

---



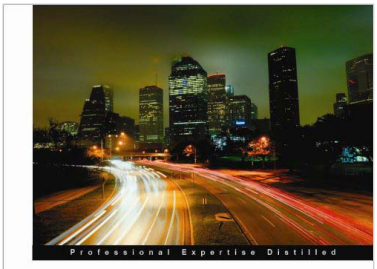
IBM DB2 9.7 Advanced Application Developer Cookbook : RAW

ISBN: 978-1-849683-96-8 Paperback: 380 pages

Over 100 practical recipes for advanced application development techniques with IBM DB2

1. Learn to design secured and robust database applications with this book and ebook.
2. Get to grips with all the important aspects of the DB2 application development life cycle starting with design and planning, moving through the development phase, and getting on to performance tips.
3. Master various new DB2 features for high quality application design.

[PACKT] enterprise
professional expertise distilled
PUBLISHING



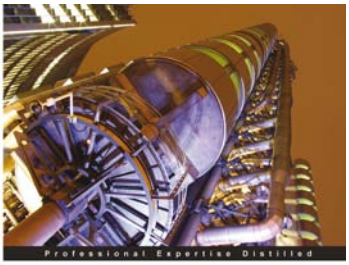
BMC Control-M 7: A Journey from Traditional Batch Scheduling to Workload Automation

ISBN: 978-1-849682-56-5 Paperback: 563 pages

Master one of the world's most powerful enterprise workload automation tools – BMC Control-M 7

1. Implement and utilize a world class enterprise batch scheduling and workload automation tool in the best possible ways with this book and e-book
2. Hands-on implementation and administration of a Control-M environment
3. Easily develop Control-M job flows to meet simple and complex business requirements

---



IBM WebSphere Application Server v7.0 Security

ISBN: 978-1-849681-48-3 Paperback: 312 pages

Secure your WebSphere applications with Java EE and JAAS security standards

1. Discover the salient and new security features offered by WebSphere Application Server version 7.0 to create secure installations
2. Explore and learn how to secure Application Servers, Java Applications, and EJB Applications along with setting up user authentication and authorization
3. With the help of extensive hands-on exercises and mini-projects, explore the various aspects needed to produce secure IBM WebSphere Application Server Network Deployment v7.0 infrastructures