Breaking into Information Security

Crafting a Custom Career Path to Get the Job You Really Want

Josh More

Anthony J. Stieber

Chris Liu

Technical Editor: Beth Friedman

ELSEVIER

AMSTERDAM • BOSTON • HEIDELBERG • LONDON
NEW YORK • OXFORD • PARIS • SAN DIEGO
SAN FRANCISCO • SINGAPORE • SYDNEY • TOKYO

Syngress is an Imprint of Elsevier

SYNGRESS®

Acquiring Editor: Chris Katsaropoulos
Editorial Project Manager: Anna Valutkevich
Project Manager: Priya Kumaraguruparan
Designer: Matthew Limbert

Syngress is an imprint of Elsevier
225 Wyman Street, Waltham, MA 02451, USA

Notices

Knowledge and best practice in this field are constantly changing. As new research and experience broaden our understanding, changes in research methods, professional practices, or medical treatment may become necessary.

Practitioners and researchers must always rely on their own experience and knowledge in evaluating and using any information, methods, compounds, or experiments described herein. In using such information or methods they should be mindful of their own safety and the safety of others, including parties for whom they have a professional responsibility.

To the fullest extent of the law, neither the Publisher nor the authors, contributors, or editors, assume any liability for any injury and/or damage to persons or property as a matter of products liability, negligence or otherwise, or from any use or operation of any methods, products, instructions, or ideas contained in the material herein.

British Library Cataloguing-in-Publication Data
A catalogue record for this book is available from the British Library

Library of Congress Cataloging-in-Publication Data
A catalog record for this book is available from the Library of Congress

ISBN: 978-0-12-800783-9

Author Biographies

JOSH MORE

Josh More started Eyra Security after spending more than 15 years in IT. He holds multiple security and technical certifications and serves in a leadership position on several security-focused groups. When taking a break from reducing IT and security risks for his company's customers, Josh enjoys reading, cooking, and photography.

ANTHONY J. STIEBER

Anthony J. Stieber has spent over 20 years in academia, banks, retail, information security, and insurance; designed enterprise security architectures; installed military and commercial firewalls; engineered medical diagnostic systems; reverse-engineered Internet stores; encrypted data warehouses; provided expertise for legal cases; spoken at international cryptography conferences; broken encrypted storage systems; studied as an apprentice locksmith; and became a published writer and recently a book co-author.

CHRIS LIU

Chris Liu has over 20 years of information technology experience, a CISSP, CISM, and no idea how he ended up where is. He has been a help desk technician, network administrator, quality assurance engineer, release manager, IT manager, instructor, developer, consultant, and product development manager, and is currently an information security professional. He is proof that careers sometimes only make sense in retrospect.

Acknowledgments

Many people helped us with the book; family and friends provided support and understanding for the many nights, evenings, and weekends we each worked on the book.

We are particularly grateful to all of those who contributed the information security stories in the book, and regret those we couldn't include:

Travis Abrams, Jim Chan, David Henning, Michael Huber, Heather Kohls, Mak Kolybabi, Joshua Marpet, John Meyers, Stephen Northcutt, Greg Sullivan, Jimmy Vo, Alan Waggoner

We support free and open-source software and its communities and believe it is critical to information security. GNU Emacs, GNU/Linux Knoppix, LibreOffice, Mozilla Firefox, Perl, VIM, and much other free open-source software were used in the production of this book.

CHAPTER

INTRODUCTION 0.1

INTRODUCTION

This book was written by three people with three vastly different experiences in information security. The book has been influenced by everyone with whom we've worked and by every book and article we've read. So this book is from the information security community. This book has illustrative stories of over a dozen people, describing their own information security stories. The number of people involved in this book is too many to count and goes well beyond those listed on the cover and in the Acknowledgments.

Information security is constantly changing, and we expect this book will also change to keep pace. The second edition will involve even more people and cover even more topics. We don't know what the third or even fourth editions will cover.

It is our aim that this book will grow, not just with our own careers, but yours as well. With that in mind, as you read this book, please feel free to tell us what has been important to you so we can include it in the next edition.

As authors of a community book, we feel that it is important that the book not only be from the community but also be part of the community. As such, we have earmarked a portion of the royalties of this book to be donated to the Hackers for Charity nonprofit organization.

WHO SHOULD READ THIS BOOK

This book is for anyone changing roles into or within the security community. While it will likely be of more interest to people trying to break into entry level information security, the book is written so that you may break into any role, not just at the beginning of your career. Whether you're just getting started as a security analyst or are becoming a penetration testing lead in charge of your team, there should be something in this book of interest to you.

HOW TO READ THIS BOOK

This book is a survey of the information security job market and community, not a direct path to success. Information security and technology changes quickly, so any direct advice given will quickly go out-of-date. Instead, we propose a different way of thinking about your career.

Careers often follow a path of three phases or "tiers" in which you first spend most of your time learning, then spend a large amount of your time doing what you've learned, and then you may focus on teaching others. This book follows these three tiers with a Learn/Do/Teach approach. While any one role will likely involve all three tiers, the proportions of Learn/Do/Teach will change as you progress.

To read this book, read the "core" of each first, going through Models, then Learn, then Do, then Teach. Each has descriptions of several information security roles. Feel free to jump around and read what the different roles involve. Once you know what work you want to do, consider which roles earlier in the process appeal to you the most. This should help you to create your own custom career path, which will both be more rewarding and more likely to succeed than anything any of us could lay out for you. For many people, career paths are dynamic, and change as roles or jobs change.

As you progress through your information security journey, keep your goals in mind, but also keep in mind that your goals may change. Both your goals and your environment determine the path your career will take. It may be that this path will not lead you to your new goals, so pay attention as things change and adjust accordingly.

CAUTION

Ethics and Career

The cautions in this book aren't just because you could be responsible for lost or damaged data, but also because you could lose your job and damage your career. You are responsible for your life, and your career is part of your life. You are responsible for your own career, and you are responsible for using your own judgment. Ethics matter in information security; any poor ethics and bad judgment will make a lot of trouble for yourself and others in your current or future jobs.

There is no guarantee that the path you choose will work for you, so if you find yourself at a dead end, consider other options. If you keep building your skills and remain persistent, you can get where you want to go.

This book provides a framework for thinking about your career. Careers move forward in fits and starts, so be prepared to fail fast, recover fast, and start over in another role as you move to where you want to go. This book is not a hard-and-fast guide, rather it is a steady and slow career guide. Your career will probably not be like any of ours, or anyone else's. If you see a path we didn't define, consider it, if it works for you, or doesn't, share it with the community and us. As the stories in the book show, there is no single true path to success.

NOTES FROM THE AUTHORS

We are three authors attempting to speak with one voice. It was not always possible; but where we had conflicts, we worked them out together. But there is also value in us each speaking individually.

WHY COMMUNITY? — JOSH MORE

Information security is a losing game. Our adversaries — the attackers — are better-funded than we defenders, and they have more time to cause problems than we have to fix them. This will not change. Throughout history, the cycle of attack and defense arms race has been built on the premise of "good enough." A wooden shield is a good enough defense until your enemies get metal lances and longbows. A stone castle wall is a good enough defense until your enemies get siege engines (or helicopters).

The fact is, someone is always going to lose because once we get to war, it's too late for a win-win situation. And with every win, the attackers are going to get a little bit better. They are tuning their tools and techniques every day, while far too many of us defenders are spending our time just catching-up. In order to survive, we have to learn as quickly as they do, and the only way to do that is to share knowledge.

This is what community is about. Our community is not perfect; but we are getting better at sharing. When I started, companies were loath to admit that they had experienced an attack, much less were breached. Today, we're seeing reports of major data breaches monthly. The more we talk about what everybody faces, the better we can work together. Knowing what we're thinking does give attackers an edge. However, on balance, the boost we get from sharing knowledge is greater than the increase in our risk.

And really, that's what it is about. As an information security professional, your job is about balancing risk. However, you will almost never be the sole decision maker. You will explain the risks as you see them and you'll have to understand those who see them differently. To win these battles and increase your chances of surviving the constant war, you'll need help. After you read this book, talk about your ideas with your local security groups, on mailing lists and blogs, and at conferences. Seek out those who disagree with you and learn how they think. Give feedback, so we can all improve.

This even applies to book authors. This book is written by three people with collectively over 50 years of experience in the industry. But still, we're just three. We've asked for help from a handful of others, but we're also asking for help from you. If you help us help others, we all get better. If we get better faster than the attackers, we can improve everyone's defense.

SECURE THINKING? — ANTHONY J. STIEBER

The biggest difference I have seen in being good at security, not just information security, is an attitude, a mindset, to think in ways that others don't. This isn't about being smart, imaginative, educated, technically skilled, or experienced, although those can help. I have met too many smart, imaginative, educated, technically skilled, and experienced people who can't imagine security problems. They are neither stupid nor ignorant, and they are very good at other things; but they aren't very good at security. Unfortunately, some of them are in the security industry.

For example, it doesn't occur to them that their system will be attacked by someone as smart, educated, and experienced as themselves. Perhaps this is an innate goodness in them, or a lack of empathy for someone else's goals. Successful defending means being able to think at least a little like the attacker, ideally before the attacker does. This doesn't require superhuman thinking, the ability to predict the future, or being a bad person—it just means thinking enough like an attacker *before* getting attacked.

If you can think about what an attacker might do at the same time you are trying to defend, you'll be better at security than those who can't.

Some defenders, such as security researchers and penetration testers, go further and even act like attackers. If you can think about two different and incompatible ideas at the same time, if you can ask that next question, if you have the empathy to think like an attacker, but have the sympathy to not be an attacker, then you can break into information security. Everything else you can learn, and teach others so they can do security better. If you can do this, then security could be right for you.

Empathize with your adversaries, and defeat them anyway.

IS SECURITY RIGHT FOR ME? — CHRIS LIU

As an instructor who has taught security both to college students and professionals, I have found that many people are interested in being information security professionals. Unfortunately, not as many people are interested in learning *how* to be a security professional. What do I mean by this? Simple: There are no shortcuts. You must get down and dirty with technical information. You need to become intimately familiar with bits and bytes that are boring and challenging at the same time. You need to be comfortable—very comfortable—with things not working the way you expect.

I have generally been able to spot those who will do well with security by the presence of a single attitude. Do they want to learn as much as they possibly can? Are they willing to explore stray paths and dead ends, but use those to learn from their mistakes? Or do they ask the question that gives it all away: “Do I need to know this for the test?”

Yes, hacking is cool. Being able to attack websites is neat. But being able to actually analyze a disparate set of data, and develop a cohesive vision of the target takes time, patience, and the ability to think outside the box. If a probe gives you an unexpected response, you need to be able to analyze that information and use it to create a new probe. If you can only run by the script, you will never get to the cool stuff.

Things are always going wrong when you are doing security. The script that worked the last time to attack a web server doesn’t work this time, even though it should. Well, you think it should. But you weren’t aware that this new client had a slightly different configuration that made this attack entirely irrelevant. Are you able to look at a long list of failed attempts and go, “Well, at least I know this won’t work here,” and develop a new strategy?

Are you able to learn, while you are doing? Are you able to teach while you are learning? If you can, then security could be the right fit for you.

If not, you may discover that security is more frustrating than cool for you.

TERMINOLOGY

Cyber, Hacking, and Information Security Growing Pains

Information security is a young and immature field, even the term “discipline” can’t really be applied yet, and the term “profession” is still debatable. Just being paid to do something doesn’t make it a profession, it also has to get done properly, and right now information security is often not even done. Information security also has many common terms without commonly accepted meanings or are highly ambiguous. Some terms and some meanings are even controversial. The meanings of ordinary words like “defect,” “exploit,” “threat,” “vulnerability,” and “weakness,” are still argued about. Some words are particularly controversial and are information security sub-culture shibboleths that will mark the speaker. For example, within some groups the word “hacker” means “computer criminal,” in other groups it means “computer genius,” and in other groups it means both.

Another common word is “cyber” and may mean “computers, the Internet, and command and control systems in general” or it can mean “I don’t know I’m ignorant about computers or security”. Cyber can also mean almost nothing.

To avoid the ambiguity of these words and others we’ve avoided them, except when used by others in context.

CHAPTER

MODELS 0.2

MODELS

> “Essentially, all models are wrong, but some are useful.”
>
> **— George E. P. Box**

Humanity has acquired more knowledge than can fit in a single human brain. To help us understand what’s going on, we continuously abstract concepts into other concepts. For example, most people don’t need to know the differences between an Adirondack, a Bofinger, and a caquetoire. For everyday life, the abstract concept of “chair” will suffice. We do the same in information security. Networking is abstracted with the seven-layer OSI model, the four-layer TCP/IP model, or just a single “is it working?” layer on which other even higher layers are placed.

The point of a model is to simplify the world and make it more understandable. Albert Einstein is often quoted as saying “Make it as simple as possible, but no simpler.” What Einstein actually wrote in the journal *Philosophy of Science* was:

> “It can scarcely be denied that the supreme goal of all theory is to make the irreducible basic elements as simple and as few as possible without having to surrender the adequate representation of a single datum of experience.”

The fact that Einstein chose to keep his lesson on simplification rather complex indicates the type of people who typically read the journal *Philosophy of Science* in the early 01930s. This book is aimed at a somewhat different crowd. We will primarily use the model: Learn/Do/Teach.

TIME MANAGEMENT

Five Digit Years

As this book is about time management and taking the long term view, we have adopted the practice of The Long Now Foundation of writing years with five-digit dates. While we do not realistically believe that much of this book will be applicable past the year 09999 — information security changes rather quickly, after all — we do feel that deliberately thinking in a longer term that most people are used to will help you to realize the importance of taking the long view as you plan your information security career.

NOTE

References and URLs

Full references for many items mentioned in this book, such as Albert Einstein's article in Philosophy of Science, can be found in the "Appendix: People & Quotes". Others references are in broad categories in roughly the order as presented in the book, such as Security Models and Time Management.

LEARN/DO/TEACH

The Learn/Do/Teach model is adapted from the medical community. The core idea is that you learn significantly better if, after the initial learning, you actively apply it. Then, after you've demonstrated understanding, you teach someone else. This gives you the opportunity to learn without causing harm, and provides a chain of knowledge stretching from generation to generation.

This book is organized similarly. The first section, Learn, is about the importance of learning and lists common entry-level information security jobs in which you typically learn the basics. Entry-level jobs are seldom fun, but only by using them to acquire a firm background can you expect to gain the experience needed to excel at higher levels. To use your experience from other jobs in information security you can perform "lateral" moves into the Do and Teach sections of the model.

The Do section of the model introduces the importance of getting your hands dirty, literally and figuratively. It lists common mid-career jobs and discusses how there can be gaps between formal "book learning" and "the school of hard knocks" on the job. We do not prefer one option over the other but those who understand the theoretical underpinnings and then also get deep experience tend to be more successful. The point is to maximize the amount of skill you gain in a specific amount of time. Thus, if you first Learn, then Do, you can gain a significant amount of demonstrable skill. These are the roles on which a business succeeds or fails, and that put you at the heart of information security. Many people are happy working these sorts of jobs for their entire career.

The Teach section brings it together. To close the loop and pass the learning to the next generation, you give back to the community through teaching. Teaching helps solidify your thinking and makes learning new things more efficient. As with the Do section, there are lateral paths to jump into the Teach of information security. The jobs listed in the Teach section include commonly accepted senior level jobs. However, what distinguishes Teach from Do is that each of these jobs has a teaching component where you are expected to help your colleagues improve over time.

Finally, there is a section on Boosting. This section is separate because it is optional, but has highly recommended suggestions for bootstrapping skills on your own time. Boosting defeats the "experience needed to get experience" trap and shows how to use the Learn/Do/Teach cycle outside of work to rapidly gain the skills needed to make that leap to your desired job.

INFORMATION SECURITY MODELS

Other information security models are also used in this book. These are not core such as Learn/Do/Teach but they are commonly used in information security.

(ISC)^2 COMMON BODY OF KNOWLEDGE DOMAINS

The International Information Systems Security Certification Consortium or (ISC)^2 is best known for its stewardship of the Certified Information Security Systems Professional (CISSP) certification. This certification tests on what (ISC)^2 calls the CISSP Common Body of Knowledge (CBK). The CBK is organized into domains which are "buckets" of information security knowledge concentrations which form the (ISC)^2 view of a well-rounded security practitioner. In 02015 the ten domains were reorganized into eight domains but with the same information: Security and Risk Management, Asset Security, Security Engineering, Communication and Network Security, Identity and Access Management, Security Assessment and Testing, Security Operations, Software Development Security. See the Appendix: Models for details.

The CISSP CBK approach involves several domains of study that may not necessarily apply to your specific job or interests. As such, this approach may be considered too broad for some individuals. The CISSP is generally considered "broad but shallow" which is well suited for generalists who can consult specialists, but not as well suited for those same specialists.

CIA TRIAD

A simple approach to information security is the CIA triad which has the three qualities of confidentiality, integrity and availability. Sometimes this is called ACI to avoid confusion.

- **Confidentiality**—Is the quality that only those who need access to or knowledge of the data will have either. Many reports you see of people intruding on networks involves a loss or breakdown of confidentiality. Lost laptops, poor database and web site access controls, insecure email, and weak cryptography are common ways to lose confidentiality. Examples include medical record and credit card exposure. Common terms for confidentiality loss include data breach, data theft, exposure, leak, and piracy.
- **Integrity**—Is the quality of how trustworthy the system or the data within the system is—in other words, how confident you are that the system will respond as it should and that it has not been tampered with. No access controls, poor web site access controls, and weak cryptography are common ways to lose integrity. Examples include incomplete or wrong medical records and ATM skimmers. Common terms for integrity loss include data corruption, defacement, modification, subversion and tampering.
- **Availability**—As you might expect, describes how available the system is. This idea comes from such fields as health care, where unavailable information, such as a heart-rate monitor, may result in someone's death, or financial systems, where lost transaction data means lost money. No data backups, equipment failure, no redundancy, and unexpected load are common ways to lose availability. Hard disk drive crashes, failed backups, power outage, and very high and aggressively low popularity are common ways to lose availability. Common terms for availability loss include business continuity loss, data destruction or loss, outage and denial-of-service. A denial-of-service (DoS) attack is specifically against availability by an adversary. Good security detects, prevents, and withstands DoS attacks. Conversely, if a security feature protects a system by preventing anyone from accessing

it, it creates a low availability regardless of the “good” or “bad” status of the individuals using it.

All of the CIA triad qualities can be adversely affected by merely accident and natural disasters, not just deliberate actions by an adversary. This does not lessen their status as information security concerns. In addition adversaries can take advantage of accidents and disasters and make them worse.

It is useful to rank the three CIA triad qualities in order of importance which will vary. Some environments have mostly equally balanced requirements, but others will be biased toward or away from one or two.

The CIA triad approach makes some security issues very simple to explain and helps determine which quality to prioritize. Health care often prioritizes availability, finance requires integrity, and defense organizations tend to focus on confidentiality. The CIA triad is a powerful tool for thinking about information security, but like all models, it is limited.

PARKERIAN HEXAD

In 01998, Donn B. Parker proposed a different way of looking at information security. This approach doubles the number of qualities from the CIA triad, thus squaring the number of possibilities, but it is not as difficult to work with as the eight domains of the (ISC)2 Common Body of Knowledge (CBK). The variables are:

- **Confidentiality**—As in the CIA triad.
- **Possession or Control**—Covers situations where a data element or system escapes the scope of controls placed around it. It may or may not involve a compromise of Confidentiality, but simply knowing that it has escaped allows you to treat it as “tainted” until verified as unaccessed or altered.
- **Integrity**—As described in the CIA triad.
- **Authenticity**—Covers the creation of data or systems. Possession or Control focuses on potential losses of Confidentiality, but Authenticity focuses on potential losses of Integrity. If you prove a trusted system as being authentic, it may be necessary to create additional controls around it.
- **Availability**—As described in the CIA triad.
- **Utility**—To disclose a bias on the part of the authors, Utility is a critical aspect to security that most models miss. Utility combines the concepts of usefulness and usability. Some security features fail “closed,” making it difficult or impossible for anyone to access the system. In such scenarios, Utility is zero and this can overwhelm any other issues at play, although it may very well be secure in that it is inaccessible. Utility can also be considered a measure of usability. Some security controls are extremely restrictive and users actively work against them to get their work done. Since security involves a mix of both technology and people, building an understanding of how people respond is critical to your model. An unusable system won’t be used and security that isn’t used isn’t secure at all.

SANS LEARNING FAMILIES

SANS is one of the best security training companies in the world, and offers dozens of different training classes. As with many academic approaches to learning, SANS groups the classes into categories. This approach is similar to the families discussed earlier. As of October 02015, the classes break down into several categories with some overlap between each:

- Cyber Defense—general information security covering the basics and and more advanced use of common defensive tools such as firewalls, anti-virus, monitoring, auditing, and system hardening.
- System Administration—focuses on information security for the operating system administrator.
- Digital Forensic Investigations and Media Exploitation—catching concerns after they've been exploited. This can involve reverse-engineering malware and carving memory and disk to obtain evidence. In short, forensics focuses on determining what happened, so appropriate plans can be made to address any issues.
- Penetration Testing—using and creating tools to break into live wired and wireless networks, web applications, mobile devices.
- Incident Response—often combined with Digital Forensics to form DFIR, this is near-real-time and real-time response to attack using forensics, reverse engineering, and any and all of the other tools available in the other categories.
- Management—for people who are responsible for the business as a whole. This category of security learning involves understanding things at a very high level, so appropriate decisions can be made.
- Secure Software Development—all aspects of development, regardless of language. Many developers have a different approach to security, and this category is aimed at catching security issues earlier, as the sooner you find an issue, the cheaper it is to fix.
- Intrusion Analysis—after the fact analysis including defensive techniques used elsewhere and forensics.
- Cyber Guardian—a high level subset of Cyber Defense.
- Audit and Legal—focus on specific issues around standards, regulations, and other things that people are required to do from a security perspective so people can demonstrate compliance.
- Industrial Control Systems—industrial control systems (ICS) and supervisory, control and data acquisition (SCADA) systems fit in here.

JOB REQUIREMENTS

The existing models represent the daily cycle of attack/defense very well and can also be used to discuss security issues in general at a very high level. However, they do not work well for personal career growth. These models are based mostly around Doing and not Learning or Teaching. This makes sense, since most businesses are focused around getting things done. Getting things done directly affects productivity and profit, which is what matters to businesses.

Individuals, however, care about more. We care about doing better, and we also care about understanding better, making people's lives better, doing things differently, and doing different things. A straightforward model doesn't capture the inherent "squishiness" of human nature. We need something different.

So let's look at getting hired.

A detailed look at being hired is available in Josh More's book dedicated to this topic: *Job Reconnaissance: Using Hacking Skills to Win the Job Hunt Game* also published by Syngress. We will not repeat all that material, though you may wish to read it for yourself. As a summary, consider why an organization hires someone. In general, people are hired to solve a problem. In for-profit businesses, they are often hired because the product of their work can be sold for more than it costs to generate. In nonprofit organizations, people are often hired because their presence helps organizations achieve goals better and more cost effectively than without them.

With that in mind, consider two otherwise identical candidates. Both have no higher education, but one has a certification and the other does not. Odds are the one with certification will get the first offer. Now consider two new candidates. Both have college degrees, but one has taken the time to explore something and has written and published a detailed HOWTO document about it. Again, who do you think will get the job?

Fundamentally, proving yourself the best option in the job market is like proving yourself the best anywhere else. You have to stand out, in a positive way. Often, the smallest differences will make surprisingly large impacts. A single project can set you apart in a field of people that haven't bothered to do a project. In a slate of people without degrees, having a degree of any sort will matter. However, against a slate of people with degrees, a degree that matches your selected field is far more important. So:

Table 0.2.1

A person with...	Wins over a person with...
No degree, but with certifications	No degree
A general degree	No degree, but with certifications
A focused degree	A general degree
A focused degree and certifications	A focused degree
A personal recommendation from a mentor	A focused degree and certifications
An interesting project to discuss	A personal recommendation from a mentor
Experience in the "good old boys" club	An interesting project to discuss
An interesting project they have led	Experience in the "good old boys" club
An awesome life story	An interesting project they have led

Clearly, this is extremely subjective and based on the limited experience of the authors, with job experience in a specific region of the world. This model also loses the intricacies of different types of degrees (associate, bachelor's, master's, doctorate) and ignores differences in people's interview skills. Instead we can model this as people accumulate points for each thing they've done. So:

Table 0.2.2

Experience	**Points**
Certification	1 certification = 5 points 2 certifications = 7 points 3+ certifications = 9 points
Degree	Associate = 20 points Bachelor's = 30 points Master's = 35–50 points, depending on job PhD = -10–100 points, depending on job
Personal recommendation from a mentor	Mentor not known to interviewer = 10 points Mentor known to interviewer = 30 points Mentor is the interviewer = 300 points
An interesting project to discuss	1 project = 20 points 2 projects = 30 points 3+ projects = 40 points
Experience in the "good old boys" club	Mentor not in the club = −10 to 10 points, depending Mentor in the club = 50 points
An interesting project they have led	1 project = 40 points 2 projects = 60 points 3+ projects = 80 points
An awesome life story	Story not pertinent to job = 20 points Story pertinent to job = 100–200 points

As you can see, having an awesome life story can trump pretty much everything here. Even if you may not be able to do the job as well as someone else, if those hiring you think you're awesome and want to work with you, you'll be considered the "best" for the job. If they think this, the hiring company will find some way to ignore requirements like degrees and certifications. You may have to get a degree or certification after you get the job, but you'll have broken in, and that's the goal. Some organizations will also support you in getting a degree or certification by paying for tuition, books, or exam fees.

Our job hiring model assumes there are common job types, and details those jobs, their requirements, and a rough path you may follow into the job you want. Ideally, you'll also create an awesome life story so maintain your story narrative as you move through your career. But having an awesome life is only half of it. The other half is that you must be able to tell your story.

This book will help you to find your new opportunities, maximize the outcome of each, and help you tell your story.

A NOTE ON EXPERIENCE — JOSH MORE

When I was young, being turned down for an ideal job because I lacked experience was both humiliating and angering. After all, I was smart and driven. I had the magic degree that was supposed to open doors. Running the whole “need experience to get experience” game was intensely frustrating.

Then, years went by and I found myself in the position of interviewing others. And I’ve got to tell you, the kids with degrees but no experience were arrogant idiots who had no idea how the real world worked. I hoped they’d get some experience because they’d be good after that, but I didn’t want to be the one to break them in.

One way to get the experience you need to get a job that requires experience is to ignore the 40-hour cap on what you “should” be doing. *Should* is one of the biggest career killers there is. If all of your competitors get caught in the “should trap” and you avoid it, you automatically land in a position above them. For example, you could work 40 hours per week, and learn 10 more hours a week on your own time, and get a basic entry-level job. Then add projects to your resume so at the next career leap, you have more experience than everyone else who only put in 40 hours a week. This allows you to take advantage of demand for experience. If you’re still in school, devote a specific time each week as well as time on school breaks towards developing the skills you need to give you a definite edge over everyone that doesn’t. Identify your limitations and work on them until they no longer damage you, then focus where you can excel.

DEGREES

This book assumes that a degree is a bachelor’s degree in the US collegiate system, with between three and five years of study with an emphasis in a specific field. The US also has associate degrees, which are usually two years of study, typically in more hands-on fields. US advanced degrees include master’s degrees of usually two to four years, law degrees of two years, and doctorates which are usually six years or more.

The two-year associate degree typically fulfills a prerequisite toward a specific type of job, and are usually only offered by community and vocational colleges. Such degrees are useful when competing against high school graduates without training. Associate degrees also give you useful theoretical underpinnings for your intended field. Typical associate degrees that are useful in information security are in networking or system administration. Associate degrees are relatively cheap to get and can be attained fairly quickly. Degree programs for working adults on evenings and weekends are often for associate degrees. Beware of low quality associate degree programs and schools, don’t trust the schools’ own marketing, guarantees, or claims of successful graduates find your own contacts of current students and graduates, ask your mentor, and current and future employers about both good and bad schools.

Bachelor’s degrees come in several flavors, but generally involve four years of study for a wider understanding for your subject matter as well as study outside of that area. A degree in physics, for example, will likely involve one or two classes each in math and physics for eight semesters. A degree in computer science would typically combine math and programming. There is also an expectation that you spend one to two classes each semester in unrelated disciplines such as literature, history, or psychology. The idea is that, by the end of your program, you have a more well-rounded understanding of the world and, along the way, have picked up the techniques you need to talk to people outside of your field.

Advanced degrees can involve between two and twelve years of study after your bachelor’s degree, as you focus deeper on a specific issue in your selected field. Some very specific jobs require these degrees but, increasingly, they are seen as a liability outside of dedicated academic or research-focused fields.

No one can see the future. The more narrowly you focus, the more you risk getting it wrong. If you guess wrong with an associate degree, you’re out two years of your life, and tuition. If you guess wrong with a PhD, you could lose a decade. However, if you guess right, you could land your perfect job and keep it for life. It’s a high-risk, high-stakes game.

Many people reading this book will have a bachelor's degree (or equivalent) or be working on it. Such a degree will give you an edge over everyone who doesn't have one, but also doesn't require quite as much time or money as an advanced degree. The cost of advanced degrees is rising, and their relevance is being questioned more, so it is likely that we'll be seeing fewer of these in the information security market. Master's degrees in information security are uncommon and not well tested yet in the market. However, at the time of this writing, they are becoming more popular so unlike other advanced degrees, we expect we'll be seeing more of them.

A bachelor's degree tells a hiring manager that you've managed to stay focused and work within a system for four whole years. It says that you can get things done and won't cause trouble in the company. That's what matters to them. Many don't even care what field your degree is in. However, you can demonstrate these things without a degree once you've put in enough work elsewhere, but it may be more difficult to get that work and it may take longer than an associate or bachelor's degree.

Degrees typically only open the first few doors in your career. Not having a degree will make it harder to get some jobs, but when you have around 10 years, experience, you'll find that not having one will matter less and less to what you want to do. If you develop a pattern of doing awesome things and are known in relevant communities, lacking a degree may not matter at all.

CERTIFICATIONS

Certifications are sometimes viewed as a cheaper alternative to a degree, usually both in time and financial cost. They show that you've learned something and, if you work hard, that you can do something with what you've learned. Some certifications are more rigorous than others. The Offensive Security Certified Professional (OSCP) certification involves hands-on work attacking test systems to verify that you can actually perform a penetration test. Others, like the (ISC)^2 CISSP exam, are multiple-choice exams with required minimum years of information experience.

There are two challenges around certifications—which to get first and which to get last. A lot of people, once they get a certification, feel the need to acquire more. For more hiring managers, the first certification is much more important than the fourth certification. More certifications may add to your personal growth—and you may choose to let one certification expire and replace it with a more advanced one. However, generally speaking, more than three certifications are going to cost you more time and money than they'll add to your earning power. Certifications can also be perceived as negatives by experienced information security professionals who have personally experienced poor certification standards.

Part of the problem is that certifications are nearly always pass/fail for minimum requirements. Some feel that time and money spent on certification and yearly maintenance is better spent on practical work experience and study which go beyond any certification. It's not unusual for experienced information security professionals to allow their certifications to lapse. Some of them think so little of information security certifications that they don't approve of using certifications anywhere except as an otherwise useless but necessary evil to get through HR departments. They won't mention their certifications, omit certifications from their business cards, don't put them in job requirements if they can, and would prefer if others did the same. But this is a personal decision and the right choice is up to you.

Only a handful of certifications are considered requirements in the industry, and which ones change over time. These certifications are generally not the most respected by actual information security practitioners. This seeming paradox is created by the economics of certification. Human Resources (HR) departments may require a particular certification, but HR departments don't know much outside of HR, especially in information technology and information security.

Large organizations like the US Department of Defense use certification as a standard of entry, and select one that their current employees in those roles can pass. So it must provide assurance, but also it must be possible to pass certification, otherwise it is not useful for those who accept certifications. The certification must also be well-enough known that the exam can be taken at any location that the organization has people. Ideally, it will also be vetted by a third party, such as the US ANSI or another standards organization. All of these requirements are expensive for the certification agency, so the certifications must also attract enough applicants that the certification fees cover the cost of the program. All of these requirements result in a market environment in which the best-known certifications are weakened (to increase passing rates). Some of the inexpensive certifications are extremely easy to pass, as they have to increase the number of applicants to cover the cost of managing the program. Thus, less-well-known, but more highly respected certifications are created in response to the belief that certifications "aren't what they used to be," as people want their certification to be something they can be proud of.

So how do you choose a certification? In our experience, your first certification should directly apply to your new job. A good listing of door-opening certifications can be seen on the US Department of Defense's Approved 8570 Baseline for different roles reproduced below with roles **in bold**:

Table 0.2.3

IAT Level I A + -CE Network+ CE **SSCP**	**IAT Level II** GSEC Security+ CE SSCP CCNA-Security	**IAT Level III** CISA CISSP (or associate) GCIH GCED CASP		
IAM Level I CAP GSLC Security+ CE	**IAM Level II** CAP GSLC CISM CASP CISSP (or associate)	**IAM Level III** GSLC CISM CISSP (or associate)		
IASAE I CISSP (or associate) CASP CSSLP	**IASAE II** CISSP (or associate) CASP CSSLP	**IASAE III** CISSP - ISSEP CISSP - ISSAP		
CNDSP Analyst GCIA CEH GCIH	**CNDSP Infrastructure Support** SSCP CEH	**CNDSP Incident Responder** GCIH CSIH CEH GCFA	**CNDSP Auditor** CISA GSNA CEH	**CNDSP Manager** CISSP-ISSMP CISM

As you can see, (ISC)^2 CISSP, EC CEH, CompTIA Security+ CE, and SANS GCIH are fairly popular certifications across the board, so these might be good choices for your first certification. They're well-known and likely won't work against you. They are not, however, technically equivalent. If you go to the specific certification outlines and training agendas, you'll see that they cover entirely different things. See the Appendix: Certifications for URLs.

But, if these certifications are equivalent from a hiring perspective, it makes sense to pick the one that will teach you the most and cost the least time and money. A good method of selecting which certification is best is to look at how much you'll learn. The more learning you do for a certificate the more valuable it will be over the long term, not because the certification itself has that much intrinsic value, but because the certification learning process will be with you for the rest of your life.

As an example to measure the personal value of a certification, compare the SANS GCIH to the EC CEH. In the two tables below, the certification objectives are placed one-per-line and a subjective guess is made as to how much new material would be learned in that area. For example, a developer reviewing the list might already have detailed understanding as to what a buffer overflow is and score it low at 10%. However, a network administrator might not know much about buffer overflows and score it high at 80%.

Once both certification lists have been scored, average the scores for how much new learning would be involved with each certification. In the examples below, we are assuming a skilled but moderately inexperienced person is comparing the two certifications.

Table 0.2.4

SANS GCIH New Learning for Skilled but Moderately Inexperienced	
	% New Material
Backdoors & Trojan Horses	80
Buffer Overflows	80
Covering Tracks: Networks	80
Covering Tracks: Systems	80
Denial of Service Attacks	10
Exploiting Systems using Netcat	50
Format String Attacks	80
Incident Handling Overview and Preparation	80
Incident Handling Phase 2 Identification	70
Incident Handling Phase 3 Containment	50
Incident Handling: Recovering and Improving Capabilities	50
IP Address Spoofing	80
Network Sniffing	10
Password Attacks	40
Reconnaissance	50
Rootkits	30
Scanning: Host Discovery	20
Scanning: Network and Application Vulnerability scanning and tools	40
Scanning: Network Devices (Firewall rules determination, fragmentation, and IDS/IPS evasion)	40
Scanning: Service Discovery	40
Session Hijacking, Tools and Defenses	80
Types of Incidents	70
Virtual Machine Attacks	90
Web Application Attacks	80
Worms, Bots & Bot-Nets	60
Average new material:	**57.6**

Table 0.2.5

EC CEH Learning for Skilled but Moderately Inexperienced	
	% New Material
Introduction to Ethical Hacking	10
Footprinting and Reconnaissance	50
Scanning Networks	20
Enumeration	50
System Hacking	80
Trojans and Backdoors	60
Viruses and Worms	60
Sniffers	10
Social Engineering	50
Denial of Service	10
Session Hijacking	80
Hijacking Webservers	80
Hijacking Web Applications	80
SQL Injection	70
Hacking Wireless Networks	70
Evading IDS, Firewalls and Honeypots	80
Buffer Overflow	80
Cryptography	40
Penetration Testing	50
Average new material:	**54.2**

For this person, the SANS GCIH certification at 57.6% new material is a somewhat better choice from a learning perspective compared to the EC CEH at 54.2% new material. But this is just an example. A simpler approach would just count the number of new enough items. What is "new enough" is also subjective, but using 60% or less has similar results: SANS GCIH is 13 and EC CEH is 11. Note that these numbers are just examples and would always be subjective and dependent upon each person.

A deeper but still subjective approach would review a complete study guide for each certification and measure new material by the number or percentage of pages, paragraphs or lines of new material. A short cut would only look for new material in the glossary or index. If practice certification exams are available for free or cheap, then take them, and whichever score is lowest determines the certification to pursue. Again, the goal is to learn, not to accumulate certifications.

If money is limited and you have to pay for it yourself, it may be wise to compare the certifications in terms of total dollars. Include exam preparation costs, the exam cost itself, any travel or time off needed to take the exam, and certification maintenance costs. Some certifications have effective maintenance costs of hundreds of US dollars a year. Some exams are only infrequently offered in some cities requiring possibly expensive travel. Others have reduced costs for retaking after a failing score.

There are three ways to pursue certification: class, self-study, or directly challenging the exam.

Table 0.2.6

SANS GCIH vs EC CEH Costs				
	Price	**Percent New Learning**	**Cost of Knowledge**	**Wasted Money**
EC CEH—Exam Only	$600	54.2%	$325.20	$274.80
EC CEH—Courseware + Exam	$825 + $600	54.2%	$772.35	$652.65
EC CEH—Class + Exam	$2,895 + $600	54.2%	$1,894.29	$1,600.71
SANS GCIH—Exam Only	$600	57.6%	$345.60	$254.40
SANS GCIH—Courseware + Exam	Not Available	Not Available	Not Available	Not Available
SANS GCIH—Class + Exam	$5,095 + $600	57.6%	$3,280.32	$2,414.68

Here, by calculating the amount of the money you're spending that goes only towards new knowledge (percent times price), you can determine cost of knowledge. You can then subtract this number from the total price to determine whether that is the best use for such money. The wasted money—money spent to learn things you likely already know—is the measure of how valuable the certification is to you. Clearly, if the person in this example thinks they can study on their own without purchasing the courseware, taking the GCIH in the Exam Only mode is the best way to go. However, if they need a class, perhaps the CEH class is a preferable option, given that it wastes almost $800 less than the GCIH option.

How these numbers break down will vary drastically based on the specific certifications you are comparing and your specific skill level in each. In some cases, such as defending a job you already have, you might already know the majority of what you'll be tested on, so new knowledge is low. If the price is low enough, it may be worth doing just to keep your job, even though most of it would fall under "wasted money," If, however, you want a challenge, it may be worth it to pursue a much more highly priced advanced certification, because the wasted money count is low, so most of your spending will go toward new learning.

In general, many people get their first certification in a way that is as easy as possible, simply to get that edge over otherwise comparable people. Subsequent certifications tend to be far more challenging; the direct financial effect is minimal compared to the joy of learning and the increased effectiveness gained.

STRIKING A BALANCE

In the end, you need to strike a balance. By considering where you actually are, you can decide how much effort it is worth to try to do more. This may involve investing time in your education, projects to boost your experience, or studying for certifications. As you get older, you will likely find more of your time going into interesting projects rather than education and certification, which provide diminishing returns for experienced professionals.

However, you will likely go through a period in your life where it's tempting to go for more and more schooling. After all, most of us spend 16 to 20 years of our lives in school. That's the safe option. If you graduate and don't immediately find a job that you "deserve" and are tempted to go to grad school or pursue another degree, consider whether you actually need it, or whether you're just discovering that the real world is harder than you were led to believe. Many information security professionals have considered advanced degrees. However, as shown in some of the stories included throughout this book, many information security professionals have found successful, useful, and rewarding careers without them. Sometimes you just have to roll up your sleeves and get to work.

It won't be easy. There will be pain and frustration. That's part of life.

However, information security has more pain and frustration than some other fields. Your organization's adversaries are well-funded by criminals and/or nation-states. Your bosses and customers don't understand the issues, and you will not have enough time and money to build the solutions you want. However, if you can get past this—something we've learned by experience—you can find yourself making real differences in people's lives and getting paid well to do it.

For us, it's worth it. If you think that applies to you, read on.

CHAPTER

MODEL FAILURES 0.3

BARRIERS

"The map is not the territory"

— Alfred Korzybski

Models are imperfect representations of reality. Problems can arise when you measure your progress against the abstract perfection that's inherent in the model you are using. This book uses a generic approach to gaining or improving employment within the information security industry. It will work in some environments, but will fail (sometimes spectacularly) in others. In general, though, working from a solid model will help you to identify and overcome barriers you may experience in the working world.

You can deal with barriers to the job you want by pushing through them or by subverting them within the system. Different barriers require different approaches.

BARRIER ENERGY

The first barrier you are likely to encounter is that of energy. Getting a new job isn't easy, especially in a field new to you. It's hard to work for an uncertain outcome. It's possible to expend too much effort for too little reward. Progress isn't straightforward, so each thing you do gives you a new or better tool to use in working toward your goal, and perhaps gets you one step closer. Success is not guaranteed. You're building skills, not checking off items on a guaranteed plan. There *are* no guaranteed plans; failure itself is learning, so gain energy from learning, and don't lose it from failure. Once you've started, it's often easier to keep going.

In the field of chemistry, there is a concept called "activation energy." This concept holds that some chemical reactions require a certain amount of starting energy to happen. A good example is that of combustion. A puddle of pure alcohol will just sit there evaporating to nothing. However, if you apply a flame, something rather different will happen—it will ignite. The same is true, metaphorically speaking, with doing the outside work needed to get a new job. Many people have trouble just sitting down and doing what they feel needs to be done. Often, the hardest tasks are shifted to the bottom of the "to do" list. Thus, you may find yourself organizing your digital music collection three times to avoid cleaning out the garage, or cleaning out the garage instead of looking for a new job. With learning tasks, a "hard" task tends to be one that involves more learning than others.

You may find that learning a brand-new skill is much harder than improving those you already have. Improving current skills is great if you want to move further up in your current job. It's not as good if you want to transfer into a different field. If you find yourself constantly putting off tasks because you're tired or because of something else that must be done, you may need to apply a large amount of activation energy to get yourself started or you need a catalyst. For example, as we worked on this book, in fact, there were some tasks that were just too hard for us to do individually. Instead, we set aside specific "sprint" days, where we'd hold one another accountable and push through the barriers blocking us.

A common task catalyst is the dedicated work day and dedicated work area. If you can afford the time, reserve four to eight hours and plan to do nothing but focus on your required task during that time. A dedicated work area and equipment is often helpful and sometimes required for security work. Don't include work area setup or tear down in that time period. Instead, do it ahead of time so you can immediately start work when the clock starts.

Minimize distractions:

Block out your schedule ahead of time and let people know so they won't interrupt you during dedicated time.

If your work space has a door, close it, if there's a window with distractions, block it.

Silence your phone, alarm clocks, and any other devices, including turning off vibrate. It may be easiest to turn them off, put them into airplane mode, or put them out of sight but where you'll easily find them again. There are apps for that.

If it's noisy, use ordinary, cheap, readily available ear plugs. Don't use ordinary ear phones which if turned to high volume to mask noise will also damage your hearing. Noise canceling ear phones help, but cannot block all noise, especially voices, will cost more, and are yet another technological distraction.

If you need to listen while working consider instead *noise isolating* earbuds, also known as canal phones or in-ear monitors (IEM), which seal inside the ear canal and passively block outside noise and sound. The cost is comparable to good ear phones but with much better isolation, especially with custom fitting by a professional audiologist.

Block visual distractions by running applications in full screen mode, especially text editors and word processors. Turn off task bars and notification areas. If you have multiple displays, but don't need them, turn them off.

For deep concentration avoid any background noise because any sound, even soothing music, is distracting.

If you don't need the Internet to accomplish your task, turn it off. If you think you do need the Internet, seriously reconsider and instead make a list of what you need and do it later. Defer your procrastination. If you find yourself constantly going to different social media sites, consider using web filtering to temporarily block those sites, or unplug and disconnect during this time. The goal is to focus and take less time to get into the deep concentration need for thoughtful work. If you are trying to light a puddle of alcohol on fire, a flame applied for a few seconds is more effective than shining a heat lamp for several hours. If you have an energy problem, focus the energy you do have as much as you can and you're much more likely to solve it.

This focusing limited energy is like borrowing from the future. You may find that the technique works well, so you do it over and over again, only to find that eventually it stops working for you. The problem is that you can't create energy from nothing. When you think you're being productive by cutting out all distractions, you are also likely cutting out those activities that allow you to rest and regain energy for the next round. If you start getting sick or extremely tired, you may just need to take some time off from the focused tasks to recover. If you don't, you risk burning out completely and losing focus for months.

However, most of the barriers you'll encounter will be external to you. These cannot be addressed by simply buckling down and pushing your way through. Most are surmountable, but you have to question whether they are worth fighting. What follows is an incomplete list of barriers that you may encounter on your way to your information security dream job.

HUMAN RESOURCES

Many organizations have a Human Resources (HR) department. HR is supposed to keep the organization out of legal trouble, handle interpersonal conflicts, and, to varying degrees, manage benefits. Many such departments also become involved in the hiring process. This typically happens because there are a lot of time-consuming steps involved in hiring someone, and it makes economic sense to move those tasks off the plates of the hiring managers, so they can manage people. However, it often results in a situation in which you have people unfamiliar with the actual job performing the initial filter for candidates. This trend within business is largely responsible for the "you need experience to get experience" trap.

Dealing with this barrier is particularly difficult if you are trying to move into a job you've never done before. HR personnel tend to follow the old "no one ever got fired for buying IBM or Microsoft" approach to decision making. If you look like a risk, you'll be passed over for the safer choice. Thus, you have two options. You can repackage yourself to look safer. This involves spending more time developing a work history in related fields, probably by doing additional work on a volunteer basis. However, the second option is more likely to be successful— bypass them altogether.

By taking the bypass approach, you find a way to meet the hiring managers directly and get far enough along in the process that by the time you meet HR, you've already been approved, avoiding the whole appraisal process. There are many ways to do this, most of which are detailed in Josh More's book *Job Reconnaissance*, also published by Syngress.

OLD BOYS' CLUB, RACISM, AND SEXISM

The phrase "old boys' club" refers to the tendency of people to want to work with people just like them. It brings up images of old white men sitting in private clubs smoking cigars and giving favors to their friends. However, the attitude itself is universal. Fundamentally, most people want to work with those who are like themselves or those they have worked with before, regardless of other differences. It's more comfortable. However, evidence strongly suggests that more diverse workplaces are more effective, more successful and more profitable. Pointing this out, of course, may get you kicked out of the club.

Whether the tendency to prefer their own type applies to previous employment, race, sex, or class, the first decision is always going to be the same. Will you be comfortable in a job where you have to either try to blend in or constantly fight the status quo? It's okay to not want to do either. That means that the organization isn't a good fit for you and that you're not a good fit for them. Find one of their competitors, add diversity there, and out compete the other firm.

If, however, you decide you want to try to blend in, learn the cultural markers. This is easier to do in some cases than others. In the United States, an individual who appears to be white, male, and reasonably (but not overly) educated will have a lot more options than someone who presents otherwise.

If you're comfortable hiding aspects of who you are from your co-workers, at least initially, that may make it easier to find a job. People with different political opinions, belief systems, sexual orientations, and gender expressions from the majority can often find reasonable employment and later, as they prove themselves an asset to the organization, slowly let more of their personality out. The cost to this approach is possibly never being fully comfortable in that workplace. It should also be noted that some people will simply not be able to blend in to all environments. Some environments won't be compatible with all people, you can learn how to speak and act in different ways and become familiar enough with specific interests to blend in.

There are increasingly more jobs where little to no physical presence is required, sometimes called full-time telecommuting or 100% telecommuting. This is common enough in the open source communities and some organizations that are conferences just so people can physically meet. These events are usually not mandatory, with most work still done remotely and over the Internet.

If you are a natural fighter, odds are that you've been doing it for a very long time. You probably had to fight your way through school and any previous job experience. It is unlikely we can give you any advice on how to fight that isn't insulting. The techniques you use to combat racism and sexism in the workplace will be uniquely personal and vary with your environment. So, instead of tips on how to fight, we instead suggest that you consider these things:

1. Security is about protecting others. If an organization recognizes and values that, you can find common ground in common protection. If it does not, the job will be undervalued. If you are personally undervalued because of irrelevant personal details, and are doing an undervalued job, think strongly on why you want to do it. It may be a great stepping stone, but it is very unlikely to ever become the job of your dreams.
2. Business is about making money. Most people in a high position in business recognize this, and the belief in the almighty dollar can trump other prejudices. As with protection, if the company can make more money with you than they can make without you, you have your "in" into the business.
3. An organization's mission is about accomplishing goals. Sadly, nonprofit organizations can be some of the most racist, sexist, and otherwise badly functioning organizations out there, largely because there is no counterbalancing profit motive. Nonprofits can also be some of the best places to work, for the same reason. If you can show that an organization can accomplish its goals better with you than without you, you better overcome these barriers.

These three points can get other people to fight for you. In a hiring situation, there may be a group of people influencing the decision, but usually only one person making it. If that person is on your side, they can fight for you. If they're not, but enough people are on your side, you may find yourself with a job offer anyway. This indirect fighting is far more likely to succeed than any sort of direct fighting.

Direct fights, such as involving the law, are expensive and less likely to work. Even if you do succeed with such a tactic, you'll always be known as the person who sued your way into the job. Once you get the job, continue the fight to keep it, advance, and to change the culture. Join organizations and community groups such as those listed in the Appendix, and discuss ways to improve acceptance and increase diversity.

Remember, at this phase of your professional development, your primary focus is on overcoming the barrier to entry, not fighting the constant uphill battle for universal equality. That fight can wait until you're in the door.

CORPORATE CULTURE

An organization’s corporate culture is something you may never understand fully, and almost certainly will not understand before working there. There are many books on corporate culture that may be worth reading, especially those for a particular company where you wish to work. Instead, we want to explore how corporate culture can prevent a job offer. Some companies have unique, unusual cultures that may be difficult to understand. Particularly notable organizations will have articles and books written about them; look for them and read them as part of your job reconnaissance. Current employees may be able to help, but ex-employees, part-time employees, and contractors who have experience outside the organization may be better able to explain the culture.

Sports Culture

In an organization with a sports culture, most employees will have played a sport in school or college. Sports metaphors will be used in most meetings. Everyone will have a favorite team and, most times, it will be the same local professional or college team. There will be an overall belief that, when asked, you should “be a team player” and “take one for the team.” You may be expected to work late nights or weekends or even be publicly called out in a meeting so the business can save an important client relationship.

If you are a person who would fit into such a culture, none of this sounds bad. If, however, you were never one of the “jocks” or “athletic supporters” (somewhat likely if you’re reading a book on information security careers), it may sound awful. However, if you still want to work in such an organization, you have to get in the door. To overcome this barrier, check out the Facebook and Twitter accounts of the people who work there. Figure out what sports they talk about and which sports teams are involved. Pick one team and learn the names of the important players. Go to YouTube and watch the highlights of some games. Learn the rules for the game and review the last few years of results. Don’t lie, don’t pose, don’t pretend to be a fan—the goal is be inoffensive, culturally sensitive, informed, and not give a blank stare when they make some sports reference. When asked about it, answer truthfully, such as “I don’t really follow sports much, I’ve just seen some <team> games.” This approach can help you blend in without lying or otherwise being dishonest.

This same approach can work for other corporate cultural interests, see below. You might even find a new interest yourself.

Education

Most organizations have one or more base education levels. Many small businesses expect everyone to have a college degree. Some startups involve everyone having a master’s degree or even a PhD, while others are made up of current college students or college drop-outs. Many large companies expect employees to have at least a high school degree. These are, however, just basic expectations and not every organization will fit the pattern. What will fit, however, is the overall expectation that you be able to communicate on the same level as the rest of the team. Generally speaking, the more educated a person is, the more abstractly they speak.

Cultures centered around people with less education tend to be more focused on day-to-day issues and less on the long-term goals. If you don’t fit such an organization’s model and still want to work there, you need to explain how you will make their day-to-day work easier. Find out what they don’t like to do and make it clear that you don’t mind doing that and perhaps you can improve the process. Cultures focused around greater levels of education will tend to speak more strategically. They want to achieve big dreams and by focusing on that you can blend in even if you don't have the expected degrees.

An exception to these "high" and "low" education-based organizations is that of artistic organizations which tend to be far more accepting and less focused on general education. The artistic part of an organization may be most of the company, or a sub-group such as marketing, or information technology. Some of these organizations have a high number of people interested in art, books, and film. Some interests are general, others are highly specific. Within information technology groups, science fiction fandom and computer gaming are common. However, if you—for example—mistake Star Wars for Star Trek, the response can be brutal. These organizations follow a basic educational path except that you are expected to be highly educated in the specific form of art that is preferred by the group.

Like a sports culture, you need not be an expert; just familiarize yourself enough with the core interest to be able to talk about it reasonably and also know when to admit your ignorance. If you need cultural references, Wikipedia is excellent for nearly all popular areas, even sports.

Military

Organizations made up of mostly ex-military personnel exist within the information security community. These tend to come in two types.

The first type consists of people who think the military had it right and are trying to run their organization like a military unit. These are the no nonsense highly driven people you often see depicted in war movies. You may think they're just a stereotype, but they do exist and some people thrive in that culture. If you have not served in the military, you will likely not fit in, regardless of how hard you try. Having served in a different military service or different country's military may also be a problem—not only is the culture going to be at least somewhat different, but it may even be antagonistic even if from the same country. For example the rivalry between the US Army and the US Navy is notorious.

The second type have a militaristic preference, but employ a number of those who have never served in the military. These organizations share a common background that is stronger than that of education or sports, but are more welcoming. You may encounter some terse speech and intolerance for mistakes, and a militaristic chain of command, but in general, these organizations are very goal-oriented and can be great places to work. If you wish to work for one of these, it really helps to appreciate the work that the military has done. If you have negative opinions about recent military involvements, it is generally best not to express them; you hear them expressed by someone who has actually served. Even then, be careful to keep criticisms to those who sent the military to war and not to those who actually fought. Militaries are not democracies, and neither enlistees or draftees get much choice in how they serve.

Academia and Health Care

Academia is, oddly, both extremely similar to and the exact opposite of militaristic cultures. Academic institutions are structured such that the most educated people, professors and doctors, are at the top and everything done by the organization exists to support them. This structure exists in healthcare, universities, and other organizations started by those doctors and professors. They are every bit as idealistic and driven as the military-based cultures, but lack a team focus. In academia, it is very much "everyone for themselves." Sure, you can make great friendships, but don't expect anyone to put themselves in figurative harm's way to help you. This is very different from a military organization, where putting yourself in literal harm's way for your team is part of the training and often part of the experience. Fortunately there is much less physical harm in academia, although the fighting may not be any less fierce.

To break into an academic culture, be prepared to defend yourself intellectually. Any time you need, not just want, to point out that someone is wrong, have the proof and be willing to enter a debate, using the scientific method where possible, because your opponent will also. If you can play the game without making those in power look bad or directly telling them no, you're in. Note that intellectualism and scientific method by name or by practice will not work in some other environments. It may not be understood, or worse, understood to be academic and consequently derided.

AGE

Ageism isn't talked about nearly as much as sexism or racism, but as the world's population gets older, the issues are growing. Older people tend to be more expensive employees because of their accumulated salary and compensation requirements and increased experience. This isn't exactly ageism but if a company can hire someone for half the cost of someone else, and believes them to be otherwise equivalent, of course they're going to go with the less expensive candidate, who will probably be younger and have less experience. The trick here is to make it clear that you're better than the other candidates (or be willing to work for less). Use the techniques discussed in Chapter 4.0 "Boosting" of this book, also see the branding suggestions in Josh More's book *Job Reconnaissance*.

Ageism can also work the other way: younger people may be wrongly assumed to lack the skills or experience to do the job simply because they are younger. Such assumptions are as ageist as assuming older people can't learn new skills or know new technologies.

To combat straight-up ageism, you must demonstrate that you can do the job. Information security isn't like working in a warehouse. If your brain works, you can do the work, even if your body is older. If you leverage your skills and experience and show that you can out think your competition, you should be able to get through the door. Some people report successfully using blending techniques as mentioned above.

CHAPTER

TIER 1—LEARN 1.0

LEARN/DO/TEACH

> "In theory there is no difference between theory and practice. In practice there is."
>
> **— Jan L. A. van de Snepscheut**

The key to this book and, we believe, to life in general, is the concept of Learn/Do/Teach. This concept was first developed in the medical field, where medical students first learn a medical procedure, then do the same procedure while guided by someone experienced, then teach a less experienced student the same procedure. Experiencing a concept directly multiple times from multiple perspectives results in better understanding and retention.

Unless you are extremely unusual, the first time you learn something, you are unlikely to learn it in any great detail. You may understand it roughly, but it is unlikely that you understand it well enough to have any level of mastery. Once you start doing something with it, however, you can rapidly find where your understanding fails. When theory meets reality, reality always wins, and it will be common for you to discover where the initial theory omitted some details. The more time you spend in Do, the greater you refine your understanding. Finally, in the Teach phase of learning, you get someone else through the Learn step. This further enhances your understanding, as you find areas where other people's understanding comes into conflict with yours, and you resolve the conflict by better understanding them and the material.

WHY LEARNING MATTERS

Learning is critical in a career. At one time, perhaps, people could do the same job every day for a lifetime and not have to learn anything new. However, so long as there is someone or something that will do the same work for cheaper, you are forced to improve, or learn a completely new job. Over time and at quantity, this tendency toward continual improvement sets person against person, company against company, and nation against nation. Fundamentally, the best Learners (as they become Doers, then Teachers) drive the economy ahead.

Traditional jobs are going away. It is important, however, to understand what "going away" means. The Earth is a closed system. We can't lose a significant number of jobs any more than we can lose a significant amount of water. While a small amount of either may vanish over time, this amount is negligible. Instead, when people speak of “lost jobs,” what they mean is “structural unemployment” where society either perceives less value for that particular work or there is a less costly way to achieve the same work. In the former, odds are that something has replaced it and things that used to be valuable simply no longer are. If jobs are “lost” during this process, they invariably reappear somewhere else in the economy.

For example, aluminum is now used for disposable, single-use beverage cans used to be a precious metal. As an example of conspicuous consumption, Napoleon III reserved a prized set of aluminum cutlery for special guests at banquets, while less-favored guests used gold knives and forks. However, cheap industrial scale bauxite ore refining, and mass production, made and anything made from it cheap. The latter case, however, is often tied to competition.

None of this means that in the case of a job shift, skills will be transferable. Quite to the contrary. In the same way that Earth is a closed system and species that cannot compete die out, skills that are no longer needed, due to improvements in technology or changes in what the market demands, will similarly die out. The trick is to constantly be adding new skills so when this, inevitably, happens, you are not effected as much as others.

This is a book about information security. It will, by necessity, occasionally drift into the philosophical, but fundamentally, we assume that you are reading this because you want to get into the field. Within information security, the good news is that, unlike elevator operators, ballast heavers, and scutchers, most jobs in information security are unlikely to go away because of technological advancement. However, specific jobs tied to particular vendors or very specific technology are subject to the fashions of the industry and organizations. An elevator operator who could only operate an Otis and not a Schindler or General Electric would have had a difficult time. The trick is to aim toward the general, while still being specific enough to be of value to prospective employers. The more specific you are, the more value you have to an employer. However, the trade-off is that such a career is limited. As technology advances, common tasks get automated, both in terms of attack and defense. So from a strategic perspective, it is better to focus on specific threats and automate their defense than to focus on an entire class of attacks that could eventually be managed by a standalone appliance.

Attackers and defenders are in an arms race. An attacker finds a vulnerability and creates an attack to exploit it. A defender may detect this attack, and if the defender is not completely destroyed, then recovers and implements a countermeasure, so the attacker is driven towards a new attack or even counter-countermeasure—repeating and moving ever faster with each cycle. This cycle affects all attackers and defenders and results in an ever-changing environment where advantages change moment to moment.

WHY LEARNING MATTERS TO YOU

Attaching yourself to a specific technology or even a specific area of compliance is short-sighted. If your chosen technology is proven to be ineffective, you may find yourself at a dead end—both unable to find more work in your field and unable to find time or money to learn something new.

This is why learning is critical to you. Only by constantly learning can you keep in front of the wave as old technologies drown in the marketplace and sink into disuse. Also, only through learning can you determine what new areas might exist in the future and position yourself to leap onto the new technology platform as your old one dies.

You can make a good living with older technologies. However, as those technologies become common, it results in job market crowding, which can drive down salaries and even result in unemployment due to too many people being available for a specific job.

By focusing less on specific technologies and more on learning, you become nimble, leaping from platform to platform as the market changes.

WHY LEARNING MATTERS TO COMPANIES

Learning is also critical to companies. Since the attackers are constantly learning and creating a new deluge of attacks, defenders must respond. An organization must decide whether to learn or to hire. When one organization learns and uses that knowledge to protect others, it basically functions as a product or service. However, products are easily analyzed and attackers put a disproportionate amount of effort into finding flaws. After all, if one product can be shown to be flawed, all organizations protected by that product are vulnerable.

A company is in the same situation you are. By connecting itself directly to a specific product, it faces the risk of that product eroding over time and leaving it vulnerable. However, by not doing that, it has less time to devote to targeting its customer base, resulting in lower profit margins.

By focusing more on learning, companies can take greater advantage of new technologies, either to improve efficiencies (costs) or to gain advantage (value).

HOW TO LEARN

There are many ways to learn within information security. As is common in technical fields, there are many who function well as self-learners. These people learn best from books, videos, or tech articles. Others learn better when being guided, either by taking classes or under direct mentoring of another. There are learning opportunities both online and offline. The key, once you identify what you need to learn, is to identify how you best learn and maximize your use of time.

Identifying what you need to learn is what the rest of this book is about. How to learn how *you* need to learn, however, is addressed here.

The first step is to identify how much you need to learn before it can be called "done." Few people are used to thinking this way, as school was highly structured and much learning after that worked under a model of "poke at things until they work, then stop." This is a perfectly functional model for learning about many things, but it doesn't serve you well when it comes to security. Security professionals talk a lot about the idea that people either have a security mindset or they don't—that security thinking is somehow inherent to a person. This is not necessarily the case.

In truth, thinking about security involves not just understanding how things are built, but also how they break, how they can be broken by people deliberately, and what other people would gain by breaking things. Thus, one way to measure how much learning is necessary is when you can answer the following questions:

- Do I understand why it works the way it does?
- Do I understand at least three ways it can fail?
- Can I list at least three ways someone else can benefit if it fails?

One approach to learning is to keep at a subject until these questions are answered. That gives you a minimum bound for learning. However, some things are harder to learn than others, so it may be helpful to place a maximum bound as well. First, decide the worth of what you learn in hours of your time. If it's only worth one hour then set a timer or alarm clock for one hour ahead. If you can answer the above questions in an hour, great. If you can't, then stop after an hour, since you already decided that it's not worth the cost to learn it. Don't fall into the trap of sunk costs, where even more is spent because so much was already spent.

MENTORS

A mentor is someone who can suggest, guide, and advise. There are many kinds of mentors, and a person can have more than one mentor. Even mentors will have mentors. A career mentor helps with the career path, such as what skills to develop, and what certifications, if any, to pursue. A skills mentor helps with the particular skills. Technical skills mentoring is important, but social/soft skills mentoring can be even more important, especially for the technically minded. Good social skills and soft skills will elevate the technically minded far above those who lack these skills.

Corporate and industry culture mentors can be very different. Corporate culture can vary considerably. Industries also have cultures which will be common in companies in that industry. Mentors can help with navigating the difficult aspects of these cultures. This means that a mentor from a different company but in the same industry can as useful, or even more than a mentor in the same company. The outside mentor can provide the perspective that all companies in the industry are alike in some way, or not, as it may happen.

A particularly experienced mentor may also provide direct longer-term instruction as part of a master/apprenticeship. Although the master/apprentice terms are somewhat archaic, the old traditional concept is coming back in some industries as an alternative to both formal, structured education and informal, unstructured on-the-job training. Traditional apprenticeships consisted of on-the-job-training under a master and would last for several years. At the end of the training the apprentice would become a journeyman who would typically leave the master and set up their own business. An all-too-common tradition was abuse of the apprentice by the master — having the apprentice be involved in work, but not involved in learning new skills, thus protecting the master's livelihood. Although modern-day formal apprenticeships are rare, the shorter term unpaid and often exploitative internships are common, should the opportunity arise, prospective employees should be aware of their rights.

Finding a Mentor

Some professional organizations and some companies have formal mentoring programs. If your organization or company doesn't have a mentoring program, consider starting one. Management support is particularly helpful here, but informal mentoring can still work and may be a natural extension of a work-related relationship. Mentoring programs may also be available through local young professional groups and trade associations.

A mentor needn't be in the same company or even the same industry. Local clubs, meetups, social networks, and other gatherings can be a place to meet a mentor. See the Appendix: Community for more details. Once found, identify how you will work together and what each of you will get out of the relationship. Some people work best with a face-to-face meeting each month. Others can work fine via email, telephone, or social media. In identifying the purpose of the mentoring, try to be more specific than “I want to learn from you.” Set goals, such as a 20% salary increase within two years, achieving a specific certification or job title, or even just laying out a general road map to get you where you want to go. Everyone works better with goals.

Being a Mentor

Mentors are similar but not identical to teachers or tutors; see the Teach chapter for more information on being mentor.

CLASSES

Classes work in a similar way as mentorship, but money has to be factored in as well as time. When you take a class, you must expend both time and money. Perhaps you have an employer willing to pay and you can discount the money aspect somewhat, but it is still going to take time. If you really want to break into a new job, you have to consider how much learning you can get for your time and financial resources.

One way to do this is to assign an estimate for the value of learning. Learning something new could result in a new job at more pay and greater happiness, or just reduce the stress you find in your current job. Before you invest in a class, consider first if there is any other way to learn what you need to learn. Then, figure out how well you will learn that way. Only then can you make a decision as to whether or not that approach makes sense.

Fortunately, legitimate classes provide an agenda or syllabus. Much in the same way we measured learning for certifications, we can measure classes for learning. As merely an example, let’s look at an old but representative version of SEC401 from the SANS Institute. This is one of the classic starter classes for getting into information security via a training/certification approach. The syllabus for day 4 is listed below:

SEC401 Syllabus—Day 4	
Network fundamentals • Network types (LANs, WANs) • Network topologies • Ethernet, token ring • ATM, ISDN, X.25 • Wiring • Network devices • Voice Over IP (VOIP) IP concepts • Packets and addresses • IP service ports • IP protocols • TCP • UDP • ICMP • DNS • IP behavior • TCPdump • Recognizing and understanding	• UDP • ICMP • UDP Behavior IOS and router filters • Routers • IOS • Routing • Routing protocols • Access control lists Physical security • Facility requirements • Technical controls • Environmental issues • Personal safety • Physical security threats • Elements of physical security

(Continued)

Cryptography • Need for cryptography • Types of encryption • Symmetric • Asymmetric • Hash • Ciphers • Digital substitution • Algorithms • Real-world cryptosystems • Crypto attacks • VPNs • Types of remote access • PKI • Digital certificates • Key escrow • Steganography • Types • Applications • Detection	PGP • Installing and using PGP • Signing data and what it means • Key management • Key servers Wireless • Common protocols • Common topologies • Misconceptions • Security issues • Securing wireless Operations security • Legal requirements • Administrative management • Individual accountability • Need to know • Privileged operations • Control types • Operation controls • Reporting

To assess the value of this class to you, score each item by how well you already know it and how you perceive its market value. Create a score list as seen below. Assume that the maximum any skill may be worth is $100 and the minimum is $0. The “Learning” column is a percentage of what’s left to learn. For example, if you feel that you know 80% of VoIP, then you have 20% left to learn. This approach may not be completely fair but, as a rule of thumb, if you have no idea what’s left to learn, just score a column as half value, such as $50 or 50% learning and be done. But if you don’t know what a term means, assume $100 or 100% learning value in that item.

Learning Objective	**Value**	**Learning**	**Learning Objective**	**Value**	**Learning**
Network types and topologies			Symmetric Cryptography		
VOIP			Asymmetric Cryptography		
TCP, UDP, ICMP			Hashing		
DNS			VPNs		
TCPdump			PKI		
Routing Protocols			Steganography		
Physical Security			PGP		
Technical Controls			Wireless		
			Legal Issues		

For illustrative purposes, let’s assume two different people are going through the exercise. Azal has been working in technology for a while, specifically in the networking area. Day 1 is likely to be review for him. Morgaine is just graduating college with an degree in mathematics. She is experienced in academic cryptography, but little else.

Here is Azal's estimate:

Learning Objective	Value	Learning	Learning Objective	Value	Learning
Network types and topologies	$50	0%	Symmetric Cryptography	$50	100%
VOIP	$75	20%	Asymmetric Cryptography	$50	100%
TCP, UDP, ICMP	$100	20%	Hashing	$75	100%
DNS	$100	50%	VPNs	$100	25%
TCPdump	$25	20%	PKI	$50	100%
Routing Protocols	$90	50%	Steganography	$50	100%
Physical Security	$20	75%	PGP	$50	100%
Technical Controls	$75	50%	Wireless	$75	25%
			Legal Issues	$25	50%

And here is Morgaine's:

Learning Objective	Value	Learning	Learning Objective	Value	Learning
Network types and topologies	$100	100%	Symmetric Cryptography	$100	10%
VOIP	$50	100%	Asymmetric Cryptography	$100	10%
TCP, UDP, ICMP	$70	100%	Hashing	$100	10%
DNS	$100	100%	VPNs	$75	80%
TCPdump	$50	100%	PKI	$100	20%
Routing Protocols	$50	100%	Steganography	$50	50%
Physical Security	$50	100%	PGP	$100	50%
Technical Controls	$50	100%	Wireless	$100	100%
			Legal Issues	$100	100%

Azal and Morgaine scored both value and learning differently. When estimating a market value for a skill, you're going to score what you already know higher simply because it's more familiar. If you're concerned with accuracy, have others review your estimates to bring the value more in line with reality. Your perceived percentage left to learn will also have a familiarity bias. Beginners often feel that the more they know, the less they have to learn. With experts, this tendency reverses itself as the more they know, the more they know there is to learn. Correcting for these biases is less critical if you are using the tool to choose which class to take or whether to learn on your own.

To calculate the value of a class, simply multiply each value by the amount left to learn and tally up the total score card. In this example, Azal estimates that days 1 and 4 would be worth $618.75 to him and Morgaine estimates them to be worth $905.00 to her. Suppose this class cost $750.00 (not true, but since we're just looking at two days from a six-day class, it's a reasonable simplification). Azal would not get as much out of it as he'd be paying, so he should look at other classes or other modes of learning. Morgaine would clearly get the value of the class plus $155 more and should thus consider it.

For other classes, the approach is the same: just run through this exercise for every class that interests you and passes a basic quality check. The one that has the highest value to cost ratio is the one you should consider first. Although the value numbers you've assigned are arbitrary, if the same arbitrary values are used consistently, the relative values of the classes will still be correct.

SELF-STUDY

The other classic method to bootstrap learning is to go through self-study. While it is often the cheapest and fastest way to learn, self-study can be difficult if you have problems focusing, completing, and limiting interruptions. To be successful at self-study, you must be able to set a plan and stick to it. A common failure mode for this style of learning is to lose time doing things that do not advance your goal. When learning in a class environment, the instructor is responsible for the structure of setting your goals and helping you maintain your forward progress. Without an instructor, you are on your own.

Many books and recorded lectures are available for much of information security, from which you can get some structure. Most "learn on your own" books and recorded lectures function much like a class, where you are presumed to start with a certain level of knowledge and gain more understanding with each portion that you complete. At the end, it is expected, you have complete knowledge and are done. However, unlike traditional classes, this approach has at best only self-assessment tests and quizzes, so you don't know what you've actually learned. This mode of self-study requires more rigor and self-honesty to be successful.

Not all learning materials, books, and recorded lectures will be useful to you, or worthwhile in general. Read reviews, check sources, ask around, especially your mentor, and fact check the material before investing much time or money.

Information technology and information security computer aided training/instruction (CAT/CAI) is almost uniformly poor. If you have free access, give it a try, especially for introductory topics, but seriously reconsider paying any money for it. However, if you have skills in curriculum development, and authoring tools then good information security CAT/CAI would be a valuable contribution to the community. See the Chapter 4.1 "Boosting—Author" and Chapter 4.7 "Boosting—Community Support" chapters.

If you are not taking a class-based approach to your learning, it is still wise to lay out a rough plan. You may know what you want to learn. This should help you design a "final exam" so you can verify what you learned. The goal may be something technical, such as setting up an Apache web server to perform a specific goal. It may be nontechnical, such as writing a report or paper for someone else to read. It may even be externally measured, such as passing a certification exam. Only by having a goal in mind, and a means of testing, can you truly call what you're doing self-study.

To be highly successful (as defined by learning a large amount in a relatively short period of time), you also have to track your days and weeks of study. These metrics help keep you from following dead-ends and tangents into areas that matter little to your end goal. One way to do this is to break apart your learning process into smaller pieces. Each piece should be roughly the same size—about a day's or a week's worth of work. Each piece should also have a set test at the end, so you can verify that you learned it sufficiently well to move along to the next item. Once you have your agenda down, you can start the self-study process.

PROJECTS/EXPERIMENTATION

If self-study doesn't work for you, either because you have difficulty learning that way or because you face constant interruption, you may learn best by doing. Many people learn best by blending the Learn and Do phases. However, if you learn by doing, your first work will necessarily be incomplete and flawed. The old programmer's adage of "write one to throw away" really applies well here, so plan to do more Learning and Doing.

Fundamentally, the issue is that people often learn best through mistakes. We typically learn from the mistakes of others; we're hard-wired for story, and stories of wild successes and massive failures are what stick with us. However, if we're blazing new ground or have to learn on our own, we must make our own mistakes.

There's nothing wrong with this. Scientific studies have shown that there is a measurement, called Error Positivity (EP), that shows how well people learn from mistakes. Those that have higher EP tend to learn more and better than those that do not. Since high EP correlates to positive attitude and a willingness to work but does not correlate to intelligence, there is a theory that this is why, over time, competent hard workers outperform their more intelligent but less hard-working competition.

However, if your learning process was built out of a series of mistakes, it's likely to not be sufficiently stable to build upon. In technology in general, we tend to build something until it works and then move on. This makes sense because it's a lot easier to determine when a thing is functional than when it is secure. However, if you want to break into the information security field, you have to build things that work, but also break them until you understand how they fail. This is a very different process and, as you're learning, can be dangerous because by definition you don't know what you're doing.

People who learn best by doing often do it while working for someone else. This may be on-the-job learning or learning done on your own but involving work-related systems. This may be work done when volunteering for nonprofit groups or open-source communities. But learning while doing is a risk to your company, to your customers, to your friends and to yourself.

Stop and ask yourself, "What did I learn from this?" and "How can what I built harm others?" and "How do I prevent harm?" Apply the answers to these questions and fix the mistakes you made as you went. In some cases, this may require a complete rebuild; in others, it may involve leveraging other skills you have to harden the environment that people will be using.

However you decide to minimize the risk to you and others, remember that security is about protection. It's also about helping some people achieve their goals, while limiting attackers' options. However, at the end of the day, if what you did doesn't improve the protection of someone else, you can't claim that it was security-related work.

BREAKING DOWN TO BREAK IN

The rest of this book focuses on specific jobs, roles, and tasks that are common in the information security industry. Each section within the Learning tier details a specific entry-level job—what it's like and how to get into it. We will detail not only the duties of the job, but also how and why the job might be less than enjoyable. Entry-level jobs are by definition stepping stones to something greater, but where you go is often a reflection of where you have been, so jobs that you hate will taint every job after as you'll be following a path based on work you dislike. Some paths will not be worthy of your goals.

To get where you want to go, you need to know where you are and what additional skills you need to acquire to get there. There is always more than one way to do something, and that is true for career management as well. You'll find some jobs to be dead-ends, where it feels as though there is no way forward. Jobs based on dying technologies may result in your feeling as if the market has dried up.

Remember, however, that your actual job is not reviewing firewall logs or deploying patches. It's protecting people. So long as you keep that in mind, you will find other ways to protect people. Maybe it will be with a new technology. Maybe it will be replacing the legacy technology in which you are currently an expert with newer technology. Maybe it will be something nontechnical, and you'll move into management.

As long as people are cheap, lazy, or stupid, you'll have work. The trick—the only trick, really—is not to be cheap, lazy, or stupid yourself. Fortunately, the opposites: thrift, hard work, and intelligence also provide work, and will usually be more pleasant. Learn about new technologies. Invest in and develop new skills. Hold yourself to high standards. Constantly improve. And, finally, always test your understanding. Do all these things with each job and, as you move through your career, you'll find the job you want and how to get it.

CHAPTER

TIER 1—LOG REVIEWER 1.1

INTRODUCTION

> "Once is happenstance. Twice is coincidence. The third time it's enemy action."
>
> **— Auric Goldfinger in *Goldfinger* by Ian Fleming, 01959**

The log reviewer role involves a periodic review of files that store critical data about what happens within your environment. These logs typically originate from applications or from devices like firewalls or servers. In more mature organizations, the logs will likely be stored centrally in a Security Information and Event Management (SIEM) system. These logs typically contain an ongoing description of what the system is currently doing—indicating whether the system started up properly, which specific events occurred, what problems were encountered, what failed, and finally whether the system shut down properly.

In many organizations, this role is entry-level and may be filled by untrained individuals or those in a junior role in system or network administration. In larger organizations, this is a full-time job that may be one of the roles in a dedicated team. In very well-developed security organizations, this team may be dedicated to a centralized (politically, if not physically) Security Operations Center (SOC) continuous 24/7/366 real-time monitoring and response. A well-developed information security organization will have a full staffed SOC with enough log reviewers to allow all critical security logs to be reviewed for all operating hours. Although "SOC" is a common information security term, it is not universal, and even very large organizations may not have a SOC by that name, or any single unit that does SOC functions. A network operations center (NOC) may include some SOC functions. In these cases, the term "SOC" represents the job functions, if not the organizational unit itself. It also represents a possible opportunity for an ambitious and resourceful person who could create a SOC where none previously existed. Lacking a SOC is also an opportunity for attackers and represents a risk to the organization.

At a technical level, most people never interact with logs or even know they exist. Logging is often an afterthought, even with information security systems. However, the logs can be a valuable way to troubleshoot a system, both during an incident and after the fact.

There tend to be two types of logs: debugging/informational and security. Information logs are used to track application behavior and troubleshoot; most developers and system administrators are used to working with them. Security logs are different. These logs store critical information about system use, such as when people log in and out of systems, when people fail to log in successfully, and what data access attempts fail or succeed based on access controls. These logs, such as audit logs and security event logs, can provide a great deal of information about security issues on a system, which is why attackers often alter or remove logs to hide and destroy evidence. Network security logs in particular can track which websites people visit and what sorts of network applications they run.

The job of a log reviewer is to ensure that any indications of attacks are detected and responded to within a reasonable amount of time where "reasonable" can include immediately. Logs need to be regularly reviewed, perhaps even in real-time. In addition, log reviewers may be responsible for managing and protecting logs so they are not not viewed by unauthorized people or modified in any way. The job can be boring, but it is also absolutely essential.

HOW TO BREAK IN

Log reviewing is often extremely boring, so it's often the first thing offloaded onto new or junior staff. Though logs are often dull, they can hide gems. It can be interesting to trace a 20-year-old bug as you analyze network traffic from one system to another to identify why a particular log entry keeps recurring.

Reviewing logs can be hazardous. You may discover information you shouldn't know, such as possible acquisition targets, just from seeing the browsing habits of senior managers. You may discover activities that shouldn't be occurring, such as employees selling illegal items or browsing to online gambling sites. Of course, you may also identify indications of compromise (IOC) in the attack's kill-chain that can help your organization get in front of an attack and determine how best to respond.

The role requires being trustworthy enough to not snoop where you shouldn't, but also intelligent enough to see and investigate things that look odd, or hinky. This is a great opportunity to learn. How you break in will depend on the sort of business you wish to target.

An otherwise fully staffed SOC may have positions available for those that start out in log reviewing. A SOC may require employees to have backgrounds in the systems they monitor, including non-security systems. These may include systems that support business operations, software development, system administration, accounting, customer support, and even marketing. Working in these areas can bring useful skills, knowledge, and business contacts to a SOC. Your experience in these non-security areas and understanding the technical underpinnings for how these systems function can help land your job in the SOC.

Smaller firms will not have a SOC, but log review is still required. This may be part-time and involve reviewing each system individually or looking at centralized SIEM log reports. If there isn't a centralized SIEM, your best way to break into the role is to spend some time implementing a central SIEM logging system and then tune it. If you are starting from scratch, look at the Security Onion, a free Linux-based system created and supported by Doug Burks. See the Appendix: Tools for details.

HAZARD WARNING

Policy Compliance

Activities such as setting up a centralized log service, any kind of monitoring, or any kind of server may have serious policy or political issues, and as mentioned above, moral issues. Make sure that such activities are within policy, within your job description, and known of and approved by your management in writing. Management approval needs to include conflict of interest resolution for monitoring of co-workers and direct management..

In a small organization a centralized SIEM system can be implemented on old hardware that may be lying around and, as you turn on the SIEM and tune it, you will find many internal activities that would be of interest to your boss. Tuning any alerts that occur may involve touching many different systems and implementing changes to things like web filtering and MS-Windows log settings, and removing unnecessary services on numerous servers.

If you wish to move to a new company in such a role, you will need experience. You can gain such experience from implementing a SIEM on your own, perhaps at a nonprofit or school near you, or your own home environment if you can't get approval elsewhere. This not only gives you the skill you need to get over the "need experience to get experience" hump, but also gives you an excellent story to tell in interviews.

HOW TO IMPROVE YOUR SKILLS

The contents of security logs can be highly counterintuitive. A sternly worded log warning message about dangerous activity could be an indication of compromise (IOC) or be a completely normal, but misleading, entry. Simply by looking at logs, real-time, or historical, a log reviewer will begin to understand what is common and what is not. Note that commonly occurring activities may actually be quite dangerous—just because it's frequent doesn't mean it should be occurring. Log reviewers learn the difference both for the general case, and for the specific environment they are reviewing. Context is critical.

By comparing log activity with known events, and by comparing logs of different systems, the log reviewer can start to build the skill of understanding the environment. A seasoned Log Reviewer will understand what a particular log entry really means, and if it's normal and innocuous, or subtle warning of serious security issues.

There are several areas in which you can grow your skills as a log reviewer.

The first, and most obvious, is tuning the logs within the logging system. Whether you're just using a standard syslog server or a much more complex enterprise log management system the system will need tuning. A tuned security logging system will log everything that is needed, log nothing that is not needed, and alarm, alert, and/or merely warn where needed. Every time a new server comes online or a major upgrade is installed, the logging system will detune and the logs will become "chatty." The trick behind tuning is to make a choice as to whether to adjust the source of the logs or the target.

For example, the log source may be a server that is running in debug mode because a developer needed some information, perhaps a lot more information. This mode is seldom turned off, but just resetting the log level to "informational" may reduce the logging to a more manageable level.

At the log target side, you may find that every Monday morning a legacy system sends ten thousand otherwise innocuous packets that trigger an alert. Legacy systems are notoriously difficult to change, but once you know why something is happening, if you can't stop it from happening, you can instead adjust the alerting system so that it knows that the problem isn't serious and shouldn't alert.

Another way to grow your skills as a log reviewer is to look at any custom logs that a developer may have implemented. These logs may contain extremely sensitive data, such as passwords or Social Security numbers, so you could get involved with developing a tokenization or masking system so the sensitive data is not compromised, but the logs remain useful. This is also an opportunity to learn software development and further expand business contacts.

Yet another way to grow your skills is to expand your logging capabilities. Many organizations are either running a central log collector or a network-based monitor. Usually only larger and more mature organizations do both. Expanding from network utilization and service monitoring to alerting on security events is often a very easy step to take. Implementing a log parsing and alerting system is also often something you can do to improve your position.

Finally, integrating trusted alerts into your ticketing system can improve efficiency and demonstrates a solid understanding of your environment.

RECOGNIZING WHEN YOU'RE STUCK

Security logs usually have recurring patterns, both short- and long-term, ranging from seconds to months. Some environments can take years to really understand, due to seasonal and yearly fluctuations. Although computers don't have seasons, the systems they are a part of, such as retail businesses, schools, and sports, can be directly affected by seasons.

Organizations often consider log review to be a low priority task not requiring a dedicated person or team. In these cases, log review might be even ignored and replaced by other tasks considered more urgent. This is an opportunity to move over to these other areas full-time. An organization that doesn't consider security log review and monitoring important is quite possibly an organization to learn from and then leave behind. Some organizations do consider security log review important enough to do consistently, but the organization may still not provide an advancement path.

After you've been through a few seasons of logs review you should have been able to grow your skills as mentioned above, but if not, then it may be time to move on.

HOW TO GET OUT

Log Reviewer may bring skills and knowledge into new areas. For example, by focusing on particular issues, a Log Reviewer may have valuable insight into Coding/Development, Patch Management, System Administration, and Network Administration. In some organizations Log Review may be so important that it requires a Subject Matter Expert.

This deep knowledge can be used to more quickly evaluate events and issues, and to start automating log review tasks.

1. Do: In a quiet, low-threat environment, a log reviewer may be able to quickly determine that no issues exist for the moment, so they can start growing skills in related areas.
2. Do: Partially automate the log review task with simple filters (Chapter 1.4 Coder/Developer).
3. Do: Fully automate with more complicated filters and custom code and reports (Chapter 1.4 Coder/Developer).
4. Learn: Dig deeper and learn more about the logs; notify what systems produce those logs (Chapter 1.5 System Administrator).
5. Teach: Find non-security issues in the logs and notify and help others in the organization (Chapter 1.9 Quality Tester).
6. Teach others how to be a log reviewer, as someone else will have to do it when you leave (Chapter 3.0 Teach, Mentoring section).

7. Volunteer at work or do personal activity for specific tasks outside of pure log review (Chapter 4.0 Boosting).
8. Write an article, or do a presentation on what you've done, learned, and taught.
9. Advance: Automate log review to the point you are no longer needed. Your new job, perhaps at a new company, won't be log reviewer, but instead will be a multi-classed log automation specialist, which combines log review and coder/developer and quite possibly adds system administrator and network administrator.

CRITICAL WARNINGS

Unlike many information security roles, the stress level of a Log Reviewer is simultaneously and paradoxically both high and low. Commonly described as "long periods of boredom punctuated by moments of sheer terror," the life of a Log Reviewer will have its up and downs. Most environments are relatively low in terms of threats, and most threats are not subtle, so the stress is usually low. However, one anomalous event may occur that will spur further investigation, causing stress levels to spike to extremely high levels. That the high stress can occur at any time without warning is itself stressful.

Log Reviewer can be a dead-end job and should be avoided by those who don't have the circumstances, patience, or discipline to benefit from its strong learning opportunities and potentially long wait for advancement.

Table 1.1 Role at a Glance—Log Reviewer

Hours	**Travel**	**Stress Level**	**Creativity**	**Flexibility**	**Stability**
0.5–8 hours/day	Usually none	Generally low	Low	High	High
General job duties	Reviewing logs. Writing log filter rules. Fulfilling log research requests.				
Learning	High – Reviewing the logs for an entire organization covers all aspects of their operations. You can learn about how firewalls work from the firewall logs and these logs potentially contain the details of all Internet access by everyone in the organization and details on everything on the Internet that connects to that organization. Anti-virus logs contain details on what malware the organization encounters and how users react to it. As you troubleshoot issues, you may find yourself learning about servers, networking systems and vulnerability management. The learning opportunity in this role is very high, but you will likely have to push your way into the learning as the role is often also culturally isolated and time limited.				
Advancement	Typically, people advance into the role as an entry level position or laterally from junior System Administrator or Network Administrator. Expect to advance to Security Assessment, Risk Assessment, and especially Incident Responder. Log Review can also serve as critical experience for Incident Responder, Security Architect, Security Consultant, or Security Management. Culturally, most organizations will provide little in the way of formal advancement paths for Log Reviewers. If you serve within a centralized Security Operations Center (SOC), you may be able to become a team manager, but generally speaking, you will have to either change employers or petition a superior to move out of the role.				

CHAPTER

1.2 TIER 1—PATCH MANAGEMENT

INTRODUCTION

> “All programmers are optimists.”
>
> **— Fredrick P. Brooks, Jr.**

Patch management is a component of configuration management where you are responsible for ensuring that patches are applied to computers as directed by a company’s policies and procedures. Generally, when a patch is released, individuals in Vulnerability Management roles will identify the urgency of releasing the patch based on the vulnerabilities it addresses and the systems to which it applies. Once the patches that need to be released are identified, those in Patch Management are responsible for ensuring that the patches are tested before release, if possible, and then applying the patches to the appropriate systems.

Depending on the devices being patched, various tools may be used to manage the process, such as Microsoft’s System Center Configuration Manager (SCCM) and SolarWinds Patch Manager. You may be responsible for a single platform (such as those workstations running Microsoft Windows) or multiple platforms.

This is the type of job that can be very monotonous in its duties but, if you take some initiative, it can provide you with many opportunities for learning.

HOW TO BREAK IN

Though technical skills are important, being well-organized and comfortable following set procedures is at least as important for this role. If there are no available mentors you will need a clear understanding of the organization’s platform and environment. Other organizations may have mentors who will train you into the role if you are able to demonstrate solid general technical skills.

In some organizations, this role can be used as an entry point into the IT field. There are many standardized tasks, and the primary job responsibilities are to monitor, document, and report, so interns or other entry-level people may be assigned this work under close supervision. People who are already enrolled in or have graduated from a computer-related degree program are often given preference, since it demonstrates an existing level of general technical skills.

For non-intern positions, showing how you provided support as a volunteer, or helped patch computers at a local nonprofit, are ways you can demonstrate experience in this area.

HOW TO IMPROVE YOUR SKILLS

Some people view patch management as a dry and boring field, but if you are willing to put in some extra effort it can be hugely beneficial for learning about the information security field. Nearly every patch addresses some form of security vulnerability—something that either has been, or could be, used to infiltrate systems.

Take the time, either at work or on your own, to deeply examine the patch and the vulnerabilities they address; they are a great repository of security knowledge. Understand not only the specific vulnerability that is being addressed, but the class of vulnerabilities it falls under, what causes the vulnerabilities, and how to prevent them. Learning the difference between a buffer overflow and SQL injection is important for the security professional. Early on, it is unlikely that you will be able to do so for every patch, but as you go on and develop your knowledge, it will become easier and easier. This same technique is used by penetration testers and security researchers to develop attacks against un-patched systems and to develop brand new attacks. Patches often only fix a specific vulnerability, not the class of vulnerabilities.

As you get better at understanding what sorts of attacks a patch prevents, you will also become better at implementing such attacks yourself, should you wish to eventually move into active auditing and penetration testing of systems.

Similarly, one of the key concepts of security is understanding that security is not about technical solutions addressing technical problems; security is about reducing risk to a level with which the business is comfortable. Take the time to learn why some patches are critical to your organization and others are not and you will learn the business drivers for security. Learn the different reasons that may make one buffer overflow patch critical for immediate installation, outside of the normal patch schedule, while another patch can easily be installed in the regular patch schedule, or not at all.

Well-run organizations will have processes for testing patches before their rollout. Take advantage of the fact that these processes have already been set up to learn about them. Identify the key characteristics of your organization’s processes, and work to learn enough about them that you could implement your own elsewhere, should it become necessary.

Finally, take the time to learn about your organization’s platform and environment.

RECOGNIZING WHEN YOU’RE STUCK

- You dread Patch Tuesday.
- You fear Exploit Wednesday even more.
- You don’t care if that patch really did get applied.
- You don’t get a sense of job satisfaction from having the best-patched environment you can manage.

HOW TO GET OUT

These roles are common at large organizations, so if you are looking to stay in the same role, you will usually be able to find a similar role elsewhere. Server Administration roles could be an advancement opportunity, as could a Security Coordinator role. If your organization splits patch management by platform, this could be an opportunity to switch platforms and retain your patch management experience. Alternatively, if your organization splits patches by internal organization structure this can be an opportunity to change job role and retain your platform experience. Other organizations split patches between servers and clients, or operating system and applications. There is no one best way, only what works for that organization.

If you have taken the time to learn deeply about security throughout your time in Patch Management, then advancement to Vulnerability Management, Auditor, or Security Assessment, depending on the skills that you have developed along the way, may be possible.

CRITICAL WARNINGS

If you are in this role too long, you will have a hard time leaving it. Patch management is often underfunded. Ideally, this role would involve you building your skills and automating patch processes within a test or lab environment. Then, when you have enough skills to be useful elsewhere, you get promoted and another person would come in and take over your role.

Sadly, this is often not the case. Many organizations don't have test environments. All production environments should have dedicated and safe test and experimentation environments. Patches can break a system, make it unstable, or take away key functionality. Sometimes key functionality is a security vulnerability and therefore a required feature! If an organization is not willing to spend the time and money to address these risks, you should be wary of staying there too long, as they are not supportive of helping your efforts to improve things.

Also, since it is an entry-level role, you should be working to get out of this role in one to three years. The organization and team you end up in will have a significant impact on your ability to grow. If you are left in your cube, only asked to perform your tasks, and never provided opportunities for cross-training and growth, that is another sign that the company is not serious and you probably need to plan to build skills on your own so you can move on to a better job sooner rather than later.

Table 1.2 Role at a Glance—Patch Management

Hours	Travel	Stress Level	Creativity	Flexibility	Stability
8 hours/day Overtime	Low to None	Low	Low	Low	High
General job duties	Testing patches on non-production systems. Setting up systems for automatic distribution of patches. Documenting the application of patches.				
Learning	Medium High. Leverage platform skill to security management skills. Leverage security skill to platform skills. Get exposed to vulnerability management and incident response. Many opportunities for the driven individual. Can be effectively avoided if you so desire, but don't do that.				
Advancement	Vulnerability Management, Auditor, Security Assessment or Incident Responder, depending on the skills that you have developed along the way.				

CHAPTER

TIER 1—HELP DESK 1.3

INTRODUCTION

> “Don’t Panic.”
>
> **— Douglas Adams**

The Help Desk is a common part of larger information technology environments. It provides a single physical place, phone number, email address, or other communications channel for what are typically the most basic and least experienced information technology people in the company. Consequently, it’s one of the most likely places an inexperienced person will start.

The help desk is where a “people person” is appreciated, but not always valued.

Some organizations separate the information security help desks from the information technology help desks, while others combine the functions. Either approach affords a good way to break into information security for someone with limited skills or experience. If you have a choice, the information security help desk might be a better place to start. But since information security is as much about the technology as about the security, an information technology help desk could also be a good start.

Consider these differences when choosing between a job at an information security help desk and an information technology help desk: Talk to people who work specifically in those areas in those organizations, talk to both your potential co-workers and your potential management. Find out how technical each is, how much flexibility in the job exists, and what you can learn, and where you can advance.

The stress level associated with this job is highly dependent upon the stress level of the incoming requests. For example, a help desk for a data recovery company may have highly stressed customers, so much that some data recovery companies provide specialized data loss bereavement counselors. Additional stress comes from help desk operators having limited to no ability to fix systemic problems, which will be highly stressful when there are many systemic problems.

Pay is usually hourly; work hours can be highly variable, and may include working nights, weekends, and holidays, yet still be part time and seasonal. Overtime is rare.

Help Desk employees usually operate in fixed shifts, but not necessarily as part of regular 8-hour days. Hourly employees are often limited to 40 hours a week or less, while seasonal variations, and extended operating hours may result in irregular hours, but which may provide high work hour flexibility to help desk operators.

Work flexibility is usually stable, but some organizations have seasonal workforce fluctuations. The Help Desk tends to be the largest group with the lowest seniority and least experienced staff, which may result in less stability.

Help Desk duties tend to be boring, with the same types of recurring customer problems—sometimes exactly the same problems, which becomes tedious, and possibly also stressful. The most interesting problems can't be solved at the lowest help desk tier and are moved up the chain to more experienced staff.

HOW TO BREAK IN

Help Desk is an entry-level position, and so is often a new hire position, perhaps an internship. However, in some cases it may be a horizontal movement from a non-technical or non-IT position. Non-technology companies may provide this horizontal movement for those who are interested in help desk jobs, and may also provide training. Someone with some basic technical skills but not enough to get hired directly may gain both experience and business contacts by becoming the local "computer person" that people go to before they call the official Help Desk. In some environments they can immediately provide better support because Corporate Help Desk employees are unfortunately notorious for providing poor service, with long waits, poor people skills, or irrelevant advice. Becoming the local technical "go to" person can develop from any non-technical position; such a person has the advantage of already knowing the organization.

HOW TO IMPROVE YOUR SKILLS

Most help desks have frequently asked questions and problems. Not all help desks have detailed and up to date documentation for those questions and problems. Some have no documentation, or even training. Either way, it's an opportunity to Learn/Do/Teach. Read the documentation, if any. Ask questions. Research and get the answers to your own questions and those asked of you. Document and solve problems. Update and write new documentation using what you've learned. Teach what you know to your co-workers and your customers. If you're starting from nothing then as you take calls, do metrics, take note of how long calls last, and what the most frequent problems and questions are. Set up a documentation environment; share with your co-workers.

In some cases a technology company may put senior people on help desk duty to supplement staff. These are opportunities for you to learn from experienced people and develop professional relationships. The help desk environment can be a highly structured and stressful one, with limited time to do anything but take calls and resolve trouble tickets. If this is the case, use this as an opportunity to ask others, senior or not, how they learned to cope with the environment. If it is not, you should cultivate relationships outside of the organization and build a team of people of which you can ask general questions. You will have to be careful not to let slip sensitive information, but a well selected team of colleagues can be very helpful.

RECOGNIZING WHEN YOU'RE STUCK

An organization is stuck when the same problems keep recurring which could be fixed with a better product or at least better documentation. A good Help Desk environment treats help desk tickets, resolved or not, as information for making better products and services. If the organization isn't learning from this, and you as a help desk operator aren't either, it's time to move on.

If you're learning and becoming a senior person, but you haven't had at least a minor promotion or had your hard work acknowledged in six months to a year of starting, it might be time to move on. Either the organization is using you for cheap labor, or you aren't working out in that environment. This doesn't mean you aren't suited for help desk, rather that that particular work situation may not be good for anyone, or only for certain people. Many people choose to move out of the help desk environment when the opportunity arises.

If you find yourself well suited to the role, you may find yourself promoted to Help Desk Lead and, eventually, into management. This is not a way to break in to information security, but it can be a very satisfying career nonetheless.

HOW TO GET OUT

Entry level Help Desk is close to the bottommost tier in the support structure, just above the end user. They are the ones who directly handle calls and requests, while more skilled or experienced help desk people will often be moved up to a higher tier. Each tier hands off work they cannot complete to the next higher tier. To move up a tier doesn't mean being able to handle all incoming requests without having to pass on to the next tier; it means being referred by others who are at a lower tier.

Table 1.3 Role at a Glance—Help Desk

Hours	Travel	Stress Level	Creativity	Flexibility	Stability
20–40 hours/day, variable	Usually none, although may have different job sites in the same area.	Variable	Low	High	High
General job duties	Directly answering the phone. Responding to email. Working trouble tickets.				
Learning	Variable. Help Desk operators may have a formal training program, either on the job, or a classroom type environment. Supervisors are often also trainers, and sometimes even mentor-like. Every incoming request could be a learning opportunity and will be for the new help desk operator. However, over time, perhaps quickly, the learning opportunities tend to drop and the same nearly identical requests keep arriving.				
Advancement	Help Desks can vary significantly. In larger environments the Help Desk supervisor positions may be available as a growth position. The Help Desk may have significant internal and external turn-over, which can be an opportunity for the new help desk operator to stay and retain. In some organizations they are considered the menial labor of the information technology organization. However the Help Desk can be an entry level position for an information technology career. Some organizations integrate the Help Desk and use it to find and feed talent into the entire support organization. A good organization will see talented and hardworking operators and want to retain them. In not so good organizations a help desk operator will have to figure this out on their own.				

CHAPTER

1.3.1 TIER 1—HELP DESK—STORY

JIM CHAN

I started my IT career in 01999 by attending Brown Institute in Mendota Heights. I finished their program with an A+ certification and landed a desktop support job at Target Corporation. During a hardware refresh project for the Finance executives, my boss asked me to work with the Information Security team to ensure the drives were erased properly. Target Corporation is huge on mentorship programs and development plans. I checked with my boss to see if I could schedule an informational interview with someone on the InfoSec team. My boss gave me the name of the Compliance and Monitoring manager and I scheduled an informational interview. During this informational interview, the Compliance and Monitoring manager said that they have an internship program. My current boss approved, and the internship program was a one-day-a-week shadow and Q&A session with various members of the Information Security team. This is where I learned an overview of IDS, Forensics, Architecture, Identity Management, etc. At the same time, I started attending night school to finish up my bachelor's degree. After finishing the internship program, there were no openings in the Information Security team. I continued the mentorship meetings and started attending open weekly "cryptolunch" lunch meetings with the Information Security team. After a few months, my mentor informed me of an entry-level user access management position on the Information Security team. I had finished my bachelor's degree before the opening came about, applied, interviewed, and got hired on.

The interpersonal relationships I developed through the internship program, having a bachelor's degree completed, and proactively getting involved with Information Security tasks and projects were the biggest contributors to breaking into the field.

CHAPTER

TIER 1—CODER/DEVELOPER 1.4

INTRODUCTION

> “What I really need is a droid who understands the binary language of moisture vaporators.”
>
> **— Uncle Owen**

Everyone who worked with computers was once expected to program them. Now it is common for people to work in information technology without learning any programming languages at all. In this section we will discuss those few remaining people who start out in the development world and then wish to break into information security.

If you are approaching information security from this perspective, you should have some familiarity with programming languages. Once you get in, you may find yourself reviewing existing code for security concerns. You may be creating brand-new programs or extending old ones. You may have a bridge role between non-security developers and the information security teams. Most information security roles involve some sort of bridging, but few roles bridge more different worlds than between non-security developers and information security.

This bridge role role will probably be at most 75% development and the rest discussions, explanations, and meetings. You should expect to gain deep skills in languages and to get very good at understanding how other people communicate, identifying application and business model issues, and prioritizing tasks.

HOW TO BREAK IN—PRELIMINARIES

Typically this role involves first knowing one language very well. A common but bad practice is that good security coder/developers must know a little bit about many different languages. Commonly chosen languages include: C, C ++, Java, Javascript, Perl, Python, and Ruby. Knowing a little about each means your work will be poor in all of them.

There are no shortcuts to learning, and this applies to learning programming languages just as much as other subjects. Perhaps you spend one month learning Perl. Then, when you hit limitations to the language, you spend a month learning Python. As you dig into security scripts, you learn some Ruby. Later, you may create modules to get your code to run efficiently, which means two to three months learning C and another two months in learning C ++. In a year of learning, you’ve learned only part of six to eight languages but none of them very well, and perhaps not well at all. In a worst case, you may only be able to write bad but otherwise useful code only by using all of the languages you know, instead of using the one language that would be best suited for the task. This is worse than the feared "spaghetti code," it's random layered pasta code.

Contrast this approach with the "learn one language very well, then move on" approach. The better you know your first language, the easier it is to pick up similar ones. Consider how much easier it is for English speakers to learn French and Italian after having thoroughly learned Spanish. The grammar, vocabulary, and logical structures of the latter languages are all similar, so picking up the differences is easier than learning any one of them from scratch. The same applies to programming. Once you learn one programming language thoroughly, you can use that knowledge to understand other programming languages. Documentation is easier to understand, because you've picked up the vocabulary around the language and can find what you need much more quickly. Troubleshooting code also becomes easier because you can quickly create proof-of-concept functions in first language to identify whether the issue is related to a bug or fundamental misunderstanding.

In general, you should start with the language you know best and spend upwards of six to eight months learning it as well as you possibly can. If you learn best from books (likely, as you're reading this one), read one or two beginner books, three to four intermediate books, and at least one advanced book. Select books with exercises to do, and never skip the exercises. The goal isn't to just read the book, the goal is to learn, and completing the exercises is your proof to yourself that you learned. The completed exercises are also a valuable reference for yourself and can be part of the portfolio you can present to prospective employers. A good example of book learning is the approach of Perl book publisher O'Reilly and Associates. If you start with *Learning Perl*, then move on to *Programming Perl*, *Mastering Regular Expressions*, and *Perl Best Practices*, then wrap up with *Advanced Perl*, you can jump from beginner to master in very short order. See the Appendix on Perl for details. Once you've mastered one language, you can pick up any intermediate book on C, Python, or Ruby and bootstrap yourself to an experienced level very quickly.

The same approach applies if you're learning from online materials, videos, or a classroom. Make sure that, before every new lesson, you understand the previous lesson as much as you can to minimize time lost to misunderstanding.

HOW TO BREAK IN—BEYOND THE BASICS

Once you have your basic skills down, it's time to leverage them and move in an information security direction. One common method is to join the team as a code assessor. These jobs are frequently advertised; in large companies they should be easy to find. In smaller organizations and some development teams, you may find a hybrid developer/security team lead role. For both of these, the basic approach is to apply for the position and then do well in the interview. This is, of course, easier said than done.

If you want to help a company start a development security practice, you have to come across as an expert. For this, it may help to develop some additional skill via one of the "boosting" paths near the end of the book. In general, working on a code similar to your target environment will do more for you than anything else. If you're targeting a company with legacy Microsoft ASP.NET code that it's converting into Java, you can quickly develop some experience. Find an older orphaned project written in ASP.NET that still has some legitimate value in terms of business logic on Microsoft's CodePlex.com, SourceForge.net, or similar open source hosting site. Then port it over to Java, documenting what you did and why you did it. This builds your portfolio and will give you the type of story you'll want to talk about in the interview, greatly increasing your chances of landing the job. Increasingly developer resumes are their open source code repositories.

Unfortunately, most hiring firms are obsessed with specific languages, and if you don't have what they want, you'll be rejected. If you follow the above advice and know one language very well, you should be able to pick up your new target language very quickly. This can help you turn an industrial disadvantage into an advantage, since if they're discriminating against you for not having the target skills, they'll be doing the same to everyone else. If you can develop the target skills quickly, you can move into a much smaller pool of applicants.

There are a lot of programmers out there. The number of those programmers that know information security is considerably smaller. If you can target a specific language at an organization where you want to work, you've cut the smaller pool into a tiny fraction, and increased your chances. For specific uncommon languages like Haskell and Smalltalk, you may be down to a pool of one—which means you get to name your price for the job.

HOW TO IMPROVE YOUR SKILLS

Once you have the job, you'll have to stay on top of things. Unlike other information security jobs, you can't just stay on top of security events and new tools. You also have to watch for code libraries. A very common failing for developers is to use an older library and also fail to keep it up-to-date. This allows security vulnerabilities and incompatibility to creep into a code base that is completely isolated from any actively developed code. Thus, there are several vectors along which you will need to grow:

1. New languages—There is a tendency in development teams to use the newest or sexiest languages. This is part optimization, part planned obsolescence, but mostly fad and fashion. The optimization is related to the tendency, when learning a new language, to know the flaws of what you've been using but not yet encounter the flaws of the new language. Many shifts between Java and Microsoft.NET, MS-Windows and Linux, and Perl, Python, Ruby and Microsoft PowerShell are due to this "grass is greener" tendency. Your job, as a security expert, is to remain familiar enough with operating systems and languages to advise on the hidden information security costs of changing.
2. Defending against laziness—There is a saying that all programmers are lazy. The truth is that programmers want to program; they get into the field because they enjoy creating, but hate repetition. Programmers love automating, but hate what can't be automated. So programmers tend toward more enjoyable tasks like writing new code, learning new languages, but ignore other critical tasks like updating libraries and refactoring code to address noncritical flaws. They especially avoid documentation and development work needed to support fundamental infrastructure changes. Your job will be to advocate important security changes without alienating the team or being an irritant.
3. Ongoing analysis—Just as developers tend towards laziness, you must combat that tendency within yourself. You will need to create a schedule and stick to it, so activities such as pre-release code scans, team code reviews, and library assessment continue to be done properly. You also need to find and use as appropriate new security tools. For example, when input fuzzing was new a different fuzzer was released about once a month. Each tested for different issues, so each fuzzer

could have been run against each application, with each run resulting in more known security issues that would need to be prioritized for the future. The same idea applies to automated code scanners, vulnerability testers, and web application firewalls.

4. Programming—You must continue to improve your development skills. This can mean attending user group meetings, hackathons, and "code camps" within the community, going to official training, conducting your own training, or simply exploring a new area of the language. Many developers have achieved truly excellent levels by simply reviewing a different operator or function every single day, so that eventually they become world-class experts in the language.

RECOGNIZING WHEN YOU'RE STUCK

The most common place to get stuck in a coder/developer job is when the business stops valuing your contributions. In economic recessions, security is often the first to go as companies refocus their development efforts on adding features to attract new customers.

Though it sounds like a long time, you may find it takes a year or two to address the easiest findings. Some people make their entire careers moving from company to company, just working on this low hanging fruit. Others stay after the first set, and focus on next issues. Then there are the architects that typically come through last and address high-strategy issues such as framework and infrastructure changes, after which the first set of people is needed again. That the architects are involved last is one of the many ongoing issues in information security. Learn which phase you prefer, so you can identify when these corporate shifts happen and move to the next company.

You can also identify when you're stuck if you spend several months just addressing one class of issue in the code base. While there are some systems that need a year's worth of work to mitigate SQL injection issues, most developers choose to implement a global input validator and make sure that all classes of input types are handled there. The same approach can apply to other common validation issues, such as cross site scripting, request forgery, and the less common injection attack types like LDAP. If you spend too much time addressing a single class of flaw, it often indicates that your organization is operating in a break/fix mode and not making strategic progress. If you can't remove issues in the code faster than the other developers are putting them in, it's probably time to leave.

HOW TO GET OUT

There are many levels of work for a coder/developer. If you stay in this area, you may find a career's worth of work. However, if you want to shift focus, you can find an easy jump to system and network administration, as those roles are often made a lot easier with some scripting. At a more advanced level, programming skills will be useful in penetration testing and quality assurance. If you focus on code assessments, you can find a place on an assessment team focusing on internal code at various companies or departments.

Just as you improve your skills before you get your specific coder/developer job, you can do the same thing when jumping into another role. Find out what languages are likely to be needed in your future job and spend some time at your current job learning them. Again, once you have the basics down, you can pick up the new languages or frameworks you're going to need surprisingly quickly.

CRITICAL WARNINGS

The biggest risk you take as a developer is getting caught in a dead-end job as an application expert. New developers are often given legacy projects that are painful to learn and even more painful to maintain. If you excel at this kind of work, it is likely that you will never get anything else. It's important to be involved in an organization's future rather than its past. Although there is more risk of failing on new ventures, there are also more learning opportunities. By pursuing the future, you are much less likely to get trapped where you don't learn anything new, but also feel you can't move because you're indispensable where you are.

- "Premature optimization is the root of all evil." – Donald Knuth
- What to expect: Occasional bouts of boredom and frustration adding spice to a daily work life that involves repeated tasks and time-driven goals.
- As you develop code: Remember that people other than you will have to work with it later, including your future self, so make sure the design, algorithms and paradigms are understandable by others and that your choices and reasoning are documented.

Combat laziness: Experiment with "code katas" and self-driven language exploration to maximize your professional development and avoid falling into ruts.

Table 1.4 Role at a Glance—Coder-Developer

Hours	Travel	Stress Level	Creativity	Flexibility	Stability
8–12 hours/day	Usually none	Low	High	Low	High
General job duties	Application design, working out requirements, writing specification, coding from specifications, documentation, some project management, and a lot of debugging and troubleshooting.				
Learning	This role has a great deal of learning potential, if you wish to pursue it. It is also sadly easy to get caught in a rut when you "take a break" because learning is hard. However, if you dig into it, you can learn how the internal structure of specific languages will constrain the thinking of developers, which in turn affects how different security layers can be implemented. Diving deeper, you can learn about memory management and gain a deeper understanding as to how memory protections function on a modern operating system. Looking laterally, as you use different libraries, you will learn how those libraries both speed development and restrict future options. Learning opportunities in this role is very high, but you'll have to drive yourself through the process as there will be few formal options available to you.				
Advancement	Typically, people advance into the role as an entry level position or laterally from System Administrator and Network Administrator. From this role, people can expect to advance to Security Assessment and Penetration Testing. These skills are particularly useful if you ever wish to get into custom exploit development, Security Architect, Security Consultant or Security Management. Culturally, most organizations will provide little in the way of formal advancement paths. You can realistically expect to move towards a team lead position if you stay in this role, eventually moving into manager. You can also find a lot of flexibility to move around within large organizations, as the development process will help you to familiarize yourself with the business logic in numerous departments.				

CHAPTER

1.5 TIER 1—SYSTEM ADMINISTRATOR

INTRODUCTION

> “You can be replaced by a small shell script.”
>
> **— Bill Hassell**

There are a many variations in the job of System Administrator. The flexibility of this job is generally dependent on the size of the organization you are working for. In some organizations, your job would be to keep a small handful of servers patched and troubleshoot them when they act up. In other organizations, you may be responsible for all the servers and, as needed, the workstations. Smaller organizations may require you to also involve yourself in networking and vendor review.

There are also two approaches to how one conducts system administration: reactive and proactive. While there is a spectrum on how you work within each approach, the two approaches are sufficiently different as to warrant separate discussion.

REACTIVE SYSTEM ADMINISTRATION

Reactive system administration is the most common approach to the work and is also known as “constant fire fighting.” Many organizations have enough problems that you can be kept extremely busy moving from figurative fire to figurative fire. You may be required to quickly build new systems for a project team that only recently thought to involve the infrastructure team in their activities. You may have to apply emergency patches to address vulnerability assessments and penetration tests. You will also have to troubleshoot why systems suddenly stop working, schedule emergency reboots, and build replacement systems for failing hardware. While failing hardware is much less of a problem in these days of virtual environments, it does still occur.

As a reactive system administrator, your goal will be to make things a little bit better every day. But your ultimate goal is to become a proactive system administrator. This is done by implementing more proactive processes and the supporting technology. However, you will seldom be given the time to do so, so expect a lot of additional work in the evenings and on weekends. When I (Josh More) was working such a role, a very effective strategy was to work a Sunday through Thursday week, which opened up Sundays for patching, research, and implementing new projects.

PROACTIVE SYSTEM ADMINISTRATION

Proactive system administration is becoming more popular in companies following Agile principles and in mature service firms. In these roles, your job is to create server templates from which new systems may be built, run automated management infrastructure (such as patching, monitoring and configuration management), and create scripts to automate common tasks. A lot of proactive administration involves monitoring scripts. Typical monitoring scripts start from a centralized system that simply checks for open ports on a periodic basis. However, in a true proactive environment, these scripts will evolve over time and start to model the actual business logic. This level of monitoring may move toward an operationally focused quality assurance team, but in most organizations it's part of system administration.

As a proactive system administrator, your goal will be to automate as much of your job as possible. This will involve creating and administrating the systems that administrate other systems. Tools like Microsoft's SCCM and WSUS are common, as are open source configuration management tools like Puppet. There are also many organization-specific proprietary tools that are used in this space. These jobs tend to become boring if you fail to continuously strive for better monitoring. However, going after better and better monitoring may eventually cause conflict with developers and network administrators, so you need to work carefully with such groups and not cause political issues.

A LITTLE BIT OF PROGRAMMING

System administration is not a programming role, but learning a little bit of programming can be extremely helpful. Modern Microsoft Windows systems often have PowerShell installed, which can be used to provide far more detail about system internals than more traditional methods of monitoring such as WMI and SNMP. Linux and UNIX systems have typically more language options, although Bash, Perl, and Python are the most common. While you do not have to be an expert in any of these languages, reaching the intermediate level will save you a great deal of time on your daily tasks and help you understand how such systems work internally.

HOW TO BREAK IN

System Administration jobs are the next easiest to get after Help Desk. Organizations often face a lack of system administrators as their current set of admins improve their skills and move on to different jobs or as the organization grows to require more systems. Breaking into such a role often requires a few basic skills and the ability to demonstrate that you can patch systems without causing unexpected problems. More advanced levels within the role would require you to demonstrate your ability to predict which changes would and would not be likely to cause problems and obtain the buy-in from affected business units.

If you're starting without any of those skills, it would be wise to develop skills with monitoring and management tools. Do a search on LinkedIn for your target company and look at the resumes of people who used to work there. Their skill lists will give you an idea as to what technologies they use. If they use open-source technologies, download a copy of Oracle VirtualBox or other virtual machine environment and build a small test environment so you can try out the tools. Document your learning process in a series of blogs and, if you can find a problem that others have had (by looking on forums), write a short post on how to resolve it. That will set you above the others competing for the same job.

If your target firm uses proprietary technologies like Microsoft Windows and management tools, its more difficult. If you can afford it, consider purchasing a year of the Microsoft Developer Network (MSDN) Operating Systems subscription. It's a bit expensive, but much cheaper than purchasing the software separately and it gives you licenses and download rights for all current Microsoft operating systems. From there, you can build any pure Microsoft environment you need. Then, if your target uses other management tools, try to sign up for a free 30-day trial with those vendors to familiarize yourself with the technology. This won't be as good as real experience, but it's much more than what the average job applicant does, so it should give you an edge.

HOW TO IMPROVE YOUR SKILLS

The most critical thing you can do to improve your skills is to invest in time management and metrics. Use a stopwatch or timer to track short tasks; use a diary or calendar to track longer tasks. System administration is about managing your time. Any daily work task that takes five minutes will cost over 20 hours a year, or over half of a work week. Even if it takes 20 hours to automate that daily five minute task, the automation will break-even in a year. Once automated, those hours can be used for other tasks.

Invest in time management to get dividends of more time. The better you understand how you spend the details of your day, the better you'll be able to determine where to focus your automation efforts.

Once you've identified where your time goes, you should review each task you do and ask yourself these two questions:

1. Does this involve so little thinking that it can happen automatically?
2. What does success look like and is there a way to write my script so that if the success condition isn't met what are the consequences: can the changes be undone automatically or I am alerted right away?

If you approach your skill growth in this way, you can develop rapidly and in ways that directly help your employer. From there, it's a matter of understanding your systems to the maximum extent possible. If you learn well from books, Microsoft Press books are very good explorations of Microsoft systems internals. For Linux systems, study them component by component, spend a month exploring the Apache web server, then a month exploring Apache modules like ModSecurity. One component per month is a good rule of thumb, but if you are extremely dependent on one component, such as Apache Tomcat or JBoss, spend more time.

Develop your skills at home in directions you'd like to go but that cannot yet be justified at work. If you are the local expert in a particular tool, the more valuable you become and the more chances you'll have to learn further.

RECOGNIZING WHEN YOU'RE STUCK

People typically get stuck in system administration roles when they or their employers are resistant to automation. By embracing automation, numerous options open up for other fields. If, however, you find yourself unable to automate for any reason and doing exactly the same thing day to day, week to week, and month to month, you may grow bored and your work will suffer.

As with log review, if you are successful with automation, you'll have time to invest in skill growth for other areas. If you enjoy development, you could look into coding monitoring scripts that mimic the way a user uses your systems. You can also do counter-coding and code repeating attack scripts to ensure that a system change doesn't inadvertently disable a security layer. Set up an internal wiki and start documenting; then, at a later date, tie that into your monitoring script so the documentation is automatically kept up-to-date. If your team does not have a good log review practice, you can look into addressing automated log review through system alerting, or adding a log management appliance to your system inventory.

Any of these skills will help boost your understanding of the field and increase your capabilities both on and off the job.

HOW TO GET OUT

Leaving a system administration job is easy. Once you have such a job under your belt, finding others like it is remarkably simple. There's almost always a need in the market, so there is a lot of movement within the system administration world. However, moving from system administration to a more security-focused job can be challenging. Hopefully, you used the time in the job to automate much of your environment so that, by now, you are skilled in at least one scripting language. This will help if you wish to refocus on the attack side of the industry and move into penetration testing.

Outside of attack, analysis is one of the better areas to move into, as your experience with systems will make it far easier to assess and remove false positives. Common analysis roles include Security Assessment, Risk Assessment, and Auditing. Also, if it turns out that you enjoy system administration and wish to continue the same sort of work, but more of a focus on security, you may wish to consider taking a Vulnerability Management role.

Any of these roles will be relatively easy leaps from where you find yourself after a few years of system administration.

CRITICAL WARNINGS

The biggest warning about system administration is that you will spend most of your days deep in the internals of live production systems. If you make a mistake, it could directly affect your company's profitability. In some industries, such as healthcare, it could have a direct impact on people's lives. The trite advice is "don't screw up," but this misses the great importance of the Learning stage of your career. If you don't screw up, you'll never learn. The trick is finding a way to screw up without causing huge problems for the organization or your career.

One common way to address this concern is to build a staging environment where you test everything before production. Some organizations do not provide funding for such an environment, and many of those that do will refuse to fund the systems needed to keep the staging environment completely synchronized with production. If you find an organization that wants you to do system administration but will not give you what is required to do it safely, and is, at the same time, very risk-averse, you may wish to pursue employment elsewhere.

- What to expect: System administration is boring work for lazy people, but exciting with many technical and social challenges for self-driven experimentalists.
- As you manage systems: Small changes may cause significant and critical problems within the environment. Expect sudden spikes of stress without warning mixed with long-ranging quiet periods. Also expect a significant amount of evening and weekend work and sometimes holidays, as many systems can't be worked on during the day.
- Combat laziness: Use automation everywhere you can to keep the boredom of your job from giving you a sense of complacency.

Table 1.5 Role at a Glance—Systems Administrator

Hours	**Travel**	**Stress Level**	**Creativity**	**Flexibility**	**Stability**
8–12 hours/day	Usually none	Low to High	Low to High	Low to High	High
General job duties	Resolving escalated trouble tickets. Troubleshooting. Writing documentation. Writing scripts. Backup and restoration. Specifying, reviewing, testing, installing new software and hardware. Patch management. Future platform planning, budgeting.				
Learning	Some – This job can be what you make of it. If you are very self driven, learning opportunities will abound. However, this is also a job in which many people feel comfortable "coasting" – going through life doing only the bare minimum needed to keep their job. If you approach it as per the latter, you will likely never get good enough to leap from system administration to security. However, if you're overly driven, you may make the "coasters" look bad, which could have political ramifications for you. As you learn, pay attention to who gives resistance to your ideas, why they say they are resisting and why they might actually be resisting. In general, if you can structure a safe test of a new process or technology and your ideas are still rejected, there may be an internal political issue that you don't know about.				
Advancement	From Network Administrator or Log Reviewer – Typically into an entry level position or laterally. Advancement or lateral move to Network Administrator, Security Assessment, Risk Assessment, Incident Responder, Auditor and Vulnerability Manager. It can also serve as critical experience for Security Architect, Security Consultant or Security Management. If you work with patching, you may be able to move to Vulnerability Manager or to a leadership role within the systems team. However, it is more likely that you will have to either change employers or petition a superior to move out of the role.				

CHAPTER

1.5.1 TIER 1— SYSTEM ADMINISTRATOR STORY

ALAN WAGGONER

My InfoSec story is not terribly interesting. In reality, it relates directly to my current position. Therefore, I took the initiative and used one week of my vacation time, paid for a Novell IntraNetWare 4.11 Administration class, and a hotel for a week. After completing the course, I paid for the certification test myself and became a CNA (Certified Novell Administrator). When a position opened up in the IT department I was able to demonstrate my personal commitment to earning the position.

A few years later, I took the position I currently hold (it was a one-person IT department at the time). I continue to do my best to keep up-to-date with current technologies and trends with a combination of reading, webinars/webcasts, and professional training. You can never stop learning when you hold an IT/security position. If you do, you quickly become irrelevant. Keeping an open mind and being able to be flexible is very important if you want to be successful. Also, like it or not, politics plays an important role if you want to procure funding for your projects and get management approval for your policy recommendations. The soft skills of person-to-person interactions can sometimes be just as important as the technical skills, so do not ignore bettering yourself in this regard as well.

I currently hold GSEC, GSLC, and GCED certifications from GIAC. I am active with the monthly OUCH! newsletter, and have contributed to the scripts for Securing the Human training videos. I have participated in the SANS open advisory board. I am also qualified as a SANS mentor, although I have not had the opportunity to mentor a class yet.

CHAPTER

1.6 TIER 1—NETWORK ADMINISTRATOR

INTRODUCTION

> “Never underestimate the bandwidth of a station wagon full of tapes hurtling down the highway.”
>
> **— Andrew S. Tanenbaum**

A Network Administrator’s specific job duties will vary based on the size and structure of the organization it is in. The focus will be on the transport infrastructure (routers, switches, firewalls, and wireless access points) used by the organization. In some cases, cable plant and network wiring will also be part of your job. In smaller organizations, there will often be additional job duties assisting with servers or even end-user workstations.

NETWORKING ROLE

In many organizations, especially larger ones, a Network Architect provides the high-level guidance for the enterprise network. A Network Engineer defines the specific configuration for each of the different pieces of hardware, and the Network Administrator makes day-to-day changes necessary to meet the business needs. For example, when a new computer is set up, as a Network Administrator you would be responsible for ensuring that the physical network port that computer will be connected to is configured appropriately. When a new application server is being set up, you would be responsible for modifying the access control lists (ACLs) on the firewall that are necessary for clients to be able to access this machine. Some organizations have dedicated network security staff just for firewalls.

In smaller organizations the Network Administrator may perform all of the above roles.

HOW TO BREAK IN

Breaking into networking varies by organization. Many organizations look for certifications such as a CCNA (Cisco Certified Network Associate) or less commonly a JNCIA–Junos (Juniper Networks Certified Associate Junos) as evidence of your skill set. There are many books and courses out there on getting network credentials. If you have a specific organization in mind before you start working toward your credential, check job skill requirements to focus your studies. Check the job postings of that organization, and search on-line forums for that organization’s email addresses. Ask around at user group meetings. Keep in mind that if you have strong demonstrable skills in one platform, many organizations will view that as adequate for moving to another platform. However, if you already have the skills on the platform they are using, you are a stronger candidate. For large organizations where you must get past an HR screen, a certification is often mandatory, unless you know a hiring manager well. If you can convince the manager that you have either the necessary skills or learning aptitude to succeed in the field, that may be enough to get you a position.

The most common mistake people make when working for their credential is focusing exclusively on what it takes to pass the test, rather than learning the material. Just because you can make it past the HR screen is no guarantee of success in the interview or on the job. Also, these are the skills that are providing the foundation of your security knowledge for breaking into the security field. Don't take short-cuts in getting certifications.

If coursework or self-study is not the best option, help desk roles at smaller organizations will often provide the opportunity to learn networking skills. Smaller organizations cannot afford dedicated help desk personnel so they will often be asked to assist beyond simply assisting end-users, and this may give you the opportunity to learn networking skills. For example, basic help desk troubleshooting may include network diagnostics down to the physical cable plant and network infrastructure layers.

HOW TO IMPROVE YOUR SKILLS

This role provides multiple ways to improve your skills. Whenever you are working under the direction of a more advanced role (such as a Network Engineer or Architect), take any opportunity you have to learn the reasoning behind their decisions. Do not be surprised to discover that sometimes the decision is business- or risk-based, rather than purely technical.

As you are going through your own job duties Evaluate why you are doing them for the organization. Are your tasks being done the most effective and efficient way possible? If not, can you identify an underlying business driver (such as a compliance requirement) that requires this way of doing business? This evaluation process, and understanding the business drivers for your role, will make you more effective in advanced roles. If this process is not required, develop an alternate, more efficient, method. Whether it is implemented or not, the processes involved will help develop your skill sets both technical and non-technical.

To grow your skills through courses and certifications, many vendors have certification tracks to advance your skills and career. Some vendors provide online labs in which to learn. If not create a lab network at home using used equipment from online auction sites to practice and study for these exams.

RECOGNIZING WHEN YOU'RE STUCK

If you do not advance, either formally or informally, within a few years, then you are likely starting to become stuck in this role. If three years into the job your duties match those you did after being on the job three months, then you are likely stuck and need to figure out how to best extricate yourself from the situation.

HOW TO GET OUT

Moving from one Network Administration role to another Network Administration role is generally easy. Having gained the experience, you are now much more attractive to other employers. If you enjoy networking, and have advanced your skills, moving up to a Network Engineer role is often an option. If being in networking is not for you, then often lateral moves into System Administration, Security Coordinator, or Coder/Developer are possible, depending on what other skills you have.

The key to getting out from your current situation are the skills you have developed while you were in the role. Did you work with your company's compliance team to develop auditing procedures? Did you develop techniques for automating changes using scripting languages? Did you run a standard scan developed by the security team after you made firewall changes? Those types of experiences will help provide opportunities for advancement into other roles.

CRITICAL WARNINGS

Organizations that do not provide any means to test changes on non-production systems are worrisome. While few will be able to have an exact duplicate of their production system in their test labs, good organizations will have labs for testing system upgrades and significant configuration changes. If you do not have the opportunity to learn, either through interactions with more advanced peers, or through experimentation in a test lab, then future advancement will be more difficult.

If you find yourself only doing exactly what you are told, without taking any time to investigate the environment or improve your skills, then you are very likely going to stagnate in this role and should consider lateral moves into areas involving tasks that you find more engaging.

Table 1.6 Role at a Glance—Network Administrator

Hours	Travel	Stress Level	Creativity	Flexibility	Stability
8–12 hours/day, after hours	Varies, often none	Generally Low with spikes	Generally Low	Low	High
General job duties	Making standard changes to network equipment based on predefined procedures.				
Learning	Can be high in the right organization. Some environments require creativity.				
Advancement	Many opportunities are available for those that take the time to advance their skill sets. Advancement may require changing employers, but advancement can occur within the same organization. If you stay in networking, you can advance to Network Engineer and from there to Network Architect (varying levels of Subject Matter Expert). For those who move into more security focused roles, potential moves into Vulnerability Management, Risk Assessment, or Incident Responder may be possible depending on the skills you have developed while working in this role, and the organization.				

CHAPTER

TIER 1—NETWORK ADMINISTRATOR 1.6.1

DAVID HENNING

My undergrad degree is in biology. I was even enrolled in a PhD program for molecular biology when I realized I needed a dramatic and risky career change. I was able to build enough skills to switch into sysadmin work during the dot-com boom of the late 01990s.

I had been working about two years as a sysadmin in Texas when I got my break to get into InfoSec. I had connections in DC through my existing job in Texas, which got my resume in front of my eventual boss who was looking for an assistant to do firewall admin and other work.

I was looking to move to the DC area. I'd had other interviews that hadn't worked out but in 01998 the job market was in my favor. Once at the job in DC, I was able to get support from management to attend SANS in Baltimore. I was able to volunteer time helping their IDNet demo where IDS vendors would plug in and show their product. This led to selecting a particular vendor for our first IDS at my job. From there, our group moved out to start up a security consulting and 24 × 7 monitoring group. I had an opportunity to work with some well-known folks in the industry that were part of the attrition.org site.

Then we all got unceremoniously bought and laid off by a bigger company. That hellacious story is too long to cover here but serves as a great learning experience. I was lucky again, had a neighbor move in next door to me who worked for another small consulting and monitoring shop. I worked for them for two years before getting tired of life on the road and found my current position where I'm going on 10 years. During that time I've continued to grow my skills, attend conferences, etc.

Maybe the best thing I've learned is not just the direction I think I want to go but also the directions I didn't want to go. Attending conferences and getting time to sit in small groups with the biggest names in the industry showed me aspects of pen testing, code development, and vulnerability discovery, which made me decide that was not going to be a successful path for my personal career. I currently manage a team of 3 analysts for an internal SOC.

CHAPTER

TIER 1—SECURITY COORDINATOR 1.7

INTRODUCTION

The role of Security Coordinator is broad; in some organizations, it is several distinct but closely related positions. Other titles that overlap this role include Security Analyst, Security Consultant, Security Facilitator, Security Liaison, Security Officer, Security Planner, Security Manager. For any of these names, "Security" can be replaced by or supplemented with "Risk," for a similar but risk- rather than security-focused role.

Security Coordinator is a full-time job, but project and management responsibilities in some organizations may require significantly more time. The Security Coordinator role is highly dependent upon the organization and bureaucracy of an organization and itself is part of that bureaucracy, but it can also be the point of flexibility within the security process.

Some organizations tightly define the Security Coordinator role, and the security policy may prevent them from performing some types of security or technical tasks.

Actual responsibilities will vary, but may include a mix of business analyst, project manager, document writer/editor, technician, engineer, architect, security/risk assessment, vendor relations, auditor, and even investigator. A Security Coordinator may perform these other roles or work directly or consume the deliverables of these specializations.

These roles usually have no direct reports or subordinates, even if they have the words "management" or "officer" in their titles. However, some organizations will directly or indirectly assign legal sign-off responsibility to the Security Coordinator so stress can be high.

Depending upon the organization's culture, geographic distribution, and specific responsibilities, travel can range from none to 100% based on the project and management aspects of the role and the wide range of expertise that may need to be consulted.

Some organizations combine information security and physical security in this role. Although integrated physical/information security is still unusual, it's a long term trend and can be an excellent way to cross over to the other side of security.

Security Coordinator can be a stepping stone to higher levels of management, or into any of the security or technical areas that touch this role in a particular organization.

HOW TO BREAK IN

In some organizations, Security Coordinator is a natural progression and even promotion from non-security and non-technical roles. For example, the Security Coordinator may be required to be very knowledgeable about the organization, which may preclude a deep security or technical background; if so, they will have to depend upon others for this expertise. In this case, having good management skills and expressing an interest in security can be enough to get started. Note that organizations with non-technical Security Coordinators may not value security as a career. Other organizations may expect a general specialist who can do everything, or at least try.

HOW TO IMPROVE YOUR SKILLS

All of the other roles are places for a Security Coordinator can learn.

Good writing skills are important for both writing and reviewing security documents, policies, procedures, security assessments.

Understand the roles of each of the different people have who work with the Security Coordinator.

Be able to possibly perform the tasks of at least one of the people who work with the Security Coordinator.

RECOGNIZING WHEN YOU'RE STUCK

Security Coordinators usually have to work with the existing security infrastructure. They may have little to no authority to change policy, purchase or implement new technologies, or accept risk. When these or other limitations of the role become frustrating, it may be time to look for new jobs and develop the applicable skills to address these issues.

HOW TO GET OUT

Because the Security Coordinator can be in the middle of the security process, there can be clear places to move on, both in lateral moves to peer roles and promotions to roles in technology, policy, and management. Because the Security Coordinator role can vary significantly, simply getting a new job at a new organization can make all the difference and be a way to grow.

Table 1.7 Role at a Glance—Security Coordinator

Hours	**Travel**	**Stress Level**	**Creativity**	**Flexibility**	**Stability**
8–10 hours/day	None to High	Medium to High	Low to High	Low	High
General job duties	Scheduling/running meetings and follow-up. Project management. Writing reports. Analysis. Budgeting.				
Learning	Whether an organization which requires the knowledge, or provides a team of experts, there is a wide range of things to learn.				
Advancement	Security Coordinator can be a stepping stone to higher levels of management, or into any of the security or technical areas that touch this role in a particular organization.				

CHAPTER

1.8 TIER 1—TRAINER-EDUCATOR

INTRODUCTION

> "*Educational* refers to the process, not the object. Although, come to think of it, some of my teachers could easily have been replaced by a cheeseburger."
>
> **— Terry Pratchett**

A Trainer-Educator is not an information security-specific role, but is particularly important in information security. Basic information technology skills are now so widespread and expected that most organizations do not bother teaching these skills. However, even basic information security awareness is frequently not known at all, and information security technical skills even less so. This is one of several reasons why information security is still immature, and is even getting worse. The Trainer-Educator is on the front-line to make information security better.

A Trainer-Educator may have a full-time job elsewhere and do training and educating as a second part-time job during evenings and weekends. A trainer-educator may also have another, possibly non-technical job. Travel time may be over weekends to allow for full work weeks of training. There may involve a lot of travel, even 100%, as it's usually more efficient for one person to travel to meet with multiple students. Although distance learning is becoming more common it still lacks the advantages of meeting face to face.

Stress levels are highly variable and depend upon management, resources, travel, and students.

Stability can be high, as some organizations and industries have mandatory yearly training requirements. But being only yearly may mean frequent travel to get to the students. Information security is constantly changing, so the need for teaching remains constant.

There can be a place for creativity, depending upon flexibility in developing course materials. Requirements can range from completely fixed legally mandated training materials that can't be altered or deviated from, while others give complete flexibility even in the same organization.

Flexibility can be limited due to fixed recurring class schedules.

Trainer-Educators may have very limited opportunity for advancement, as most organizations have small education departments. Dedicated training organizations may have more advancement possibilities. There is opportunity for the specialized Trainer-Educator independent security consultant.

HOW TO BREAK IN

The Trainer-Educator is well-established in information technology, but tends to exist in organizations dedicated to training and education. A background in training and education is a plus, but technical skills are required to work in any related job beyond basic end user instruction. Most information security Trainer-Educators have a technical background but spend part of their time doing training and education, typically in colleges and universities, and sometimes corporate training centers.

As with the Help Desk role, there is often an opportunity to get into the Trainer-Educator role by being the local "computer tutor" in the organization. Unlike the local "computer person" who fixes the computer and moves on, the "computer tutor" helps end users understand what went wrong and instructs them how to fix it in the future. This usually takes more time at first, but in the end can result in less time total spent fixing problems.

HOW TO IMPROVE YOUR SKILLS

What to learn depends on where you came from and where you want to go.

Trainer-Educators who got there by teaching users to avoid future problems may have good technical skills, but no formal training in how to train people. A common approach is one-on-one tutoring sessions. These can be very rewarding and effective for the one student and Trainer-Educator, but they are time-inefficient. With no other training, a computer tutor can attempt to develop the experience of these learning sessions into simple written course material, a video, or—with more technical knowledge—a website or automated course material.

Educator-Trainers from non-technical academic backgrounds have skills and experience in the learning and teaching process, but not the technical material. They may have to rely upon existing training material and fixed class plans. They can benefit from learning the underpinnings of what they teach—not just the material they use now, but more advanced material their students may later learn in more advanced classes taught by others. If they have Educator-Trainer co-workers they may be able to learn from them as teachers or mentors. All of the other sections in this book are also material to learn for the non-technical person or non-security technical person who wants move on to understanding more technical material.

RECOGNIZING WHEN YOU'RE STUCK

You're stuck when you dread facing a class with the same old material and you can't or aren't allowed to change it.

You're stuck when you can't see anything new in what you're teaching or what the students are asking in class.

If you're the local "computer tutor" person, you may get stuck as the "computer person" who only fixes the problems and can't get people to learn for themselves. If you're stuck here you could go elsewhere as elsewhere as a help desk engineer, network administrator or systems administrator depending upon your skills.

You're stuck when the Learn/Do/Teach cycle stops and there is no more learning, no new doing, no new teaching.

HOW TO GET OUT

If you grow tired of training others, see if there's something else you can train them to do. A lot of people who enjoy teaching wind up teaching forever and symptoms of burnout are often more about the topic you're working with and not the role itself. So see if there is anything new you could teach, if you have the option to build courseware or write books to rejuvenate your career and re-energize yourself.

If, after doing this, you decide that you still wish to move on, you should be well suited to any job involving team leadership or management.

Table 1.8 Role at a Glance—Trainer-Educator

Hours	**Travel**	**Stress Level**	**Creativity**	**Flexibility**	**Stability**
8–12 hours/day and after hours	None to High	Low to High	Low to High	Low	High
General job duties	Creating course materials: slide decks, presentation scripts, coordinating subject matter experts,				
Learning	Even a completely knowledgeable Trainer-Educator will learn a lot from students. A good organization will expect and provide for the training and education of Trainer/Educators.				
Advancement	May have very limited advancement as most organizations have small education departments. A dedicated training organization may have more advancement available.				

CHAPTER

TIER 1—TRAINER-EDUCATOR 1.8.1

STEPHEN NORTHCUTT

Stephen Northcutt was a network designer for the Department of Defense. The primary tool he used in 01986 was Autocad hosted on a Sun 3 workstation, since it was much faster than a tricked-out Windows system on a 386 chip with a graphics accelerator. One day, after lunch, he was grinding some coffee beans to caffeine up, for a really complex design problem. He looked over and noticed the disk access LED on his workstation was blinking furiously. Odd! He was grinding coffee beans, not pushing his workstation around. He typed “ps” and learned his workstation was compiling software. He realized he had been hacked; there was a flaw in sendmail. Stephen felt violated and decided to switch careers to security. He wrote the Shadow IDS, and later joined SANS as an instructor, author, and manager. He retired as a SANS instructor on December 14, 02013, after logging 1.5 million miles on United Airlines, plus flights on other airlines as well.

CHAPTER

1.9 TIER 1—QUALITY TESTER

INTRODUCTION

> "Quality is job 1.1"
>
> **— unknown**

Creating complex but still functional software is very difficult. The process to accomplish this has developed several names, such as quality testing, quality assurance, quality management, and more. Although software testing is, in effect, done by everyone who encounters that software, the formal role of quality tester is common in software engineering organizations. A quality tester's qualifications can be as simple as being the end user of the software in question, recognize flaws found and sufficiently document them. Consequently even those who otherwise lack technical skills might not merely be quality testers, they could even be excellent quality testers.

A Quality Tester can have a range from low to high technical skills. At the low end, a quality tester merely has to know how to use the software in question. This may involve highly specific and technical skills in the operation of that software. A Quality Tester doesn't have to know how the software works, or even how to write software; the critical skills for a Quality Tester are being able to consistently cause software to break and communicate to those who can then attempt to fix the problem. Hence there are opportunities for both the highly, and not so highly, technically skilled. In both cases, reproducibility and communications skills are important. If a flaw cannot be reproduced then any attempted fixes themselves cannot be tested.

Software quality testers need sufficient technical skill to operate the software and know when something has gone wrong. For some software this can be very technical and very specialized. Additionally, the tester needs good communication skills to describe what was done and what happened, so it can be reproduced by a software engineer. Most software development environment organizations have some way for end users to file bug reports. Although it may not result in a job, taking an active although unpaid role in quality testing is a way to gain experience. Some organizations will solicit external users for beta testing before a production release, typically for no compensation other than early access to products, and sometimes by getting a free or discounted commercial product.

HOW TO BREAK IN

Some organizations make heavy use of internal alpha testing, sometimes known as "dogfooding" as in "eating your own dog food." Where dogfooding is in place, there may be an expectation that all employees are part of the quality testing process. Dogfooding may also be part of the organization's culture to use it’s own products and services in which case it may actually lower quality by being blinded to the competition and ordinary users.

Some amount of quality testing is often an expected part of a coding/programming job, but dedicated quality testers typically work full time. If other software development staff work long hours, it’s likely that the same will be expected of quality testers.

Quality testing stress levels tend to reflect that of the rest of the software development organization, and team and management expectation. Organizations that emphasize high-quality products may stress quality testers as much or more than software engineers. Organizations that have less concern for quality may stress quality testers who want to do high-quality jobs.

Quality testing may be considered an adjunct to software engineering, either as a pathway to it, or out of it and into other areas; or quality testing may be a separate organization.

Creativity is critical to producing test cases, both in general and in detail. Even a small test item, such as “Application starts correctly,” can be subject to a wide range of subtle failures.

Some organizations expect all software engineers and even other technical or even non-technical areas to engage in quality testing. A separate Quality Testing role may not exist.

Some organizations use Quality Tester as a junior position within a software engineering department, where there may be quick advancement to the primary goal of the department of software engineering. Other organizations have dedicated departments for quality testing, where advancement is through generalization, specialization, or management.

HOW TO IMPROVE YOUR SKILLS

The core qualities of a quality tester are:

- Understanding the software.
- Reproducibility
- Communication

As a Quality Tester you can become an expert user of the system, which itself may be valuable. If its not a skill you can take to other jobs, how you learned to become an expert user is itself a powerful skill. Going beyond understanding as a user can involve reading the design documents, talking with the developers, and reading and understanding the source code. If possible, talk to end users about what they do and how they use the system. Join a user group for the system, and the tools you use or could use. Read about not just your organization’s products, but others as well. Learn the operating system and programming languages used for the system. If those programming languages aren’t suitable for test automation, learn a language that is suitable.

Reproducibility is critical in quality testing, if you can’t reproduce a problem, you may not be able to prove it exists. Study the scientific method. Learn about control and experimental groups, and why changing only one variable at a time is crucial. Precision, and your own software engineer skills to automate tests are important.

Communication skills are needed to read and completely understand requirements to be tested, writing test cases based on requirements, writing test scripts, both manual and automatic, and writing test reports which explain what was done are required.

RECOGNIZING WHEN YOU'RE STUCK

A dedicated quality tester may be stuck when the system to be tested itself is stuck. If it's not improving, if the same flaws keep showing up, if you've completely automated testing, and have nothing more to learn, if you could do the software engineer's job, but they won't let you, it's time to move on.

HOW TO GET OUT

Security product quality testing is similar to and sometimes identical to penetration testing. In some organizations, they are even the same group or will have clearly identified paths. A lateral move may be possible into pen tester, or perhaps a promotion into pen test lead. In other organizations Coder-Developer may be the normal career path. Depending upon the systems tested other roles may also be suitable such as System Administrator or Network Administrator.

Table 1.9 Role at a Glance—Quality Tester

Hours	**Travel**	**Stress Level**	**Creativity**	**Flexibility**	**Stability**
8–12 hours/day after hours	Low	Varies	High	High	Varies
General job duties	Writing test plans and contributing to software engineering process scheduling. Reviewing and troubleshooting test results. Reading source code. Detailed troubleshooting reports and possibly writing and submitting source code changes.				
Learning	A Quality Tester doesn't require a software engineering background, but will benefit greatly, and in becoming an experienced QA person, will understanding in many aspects of software engineering.				
Advancement	Some organizations use Quality Tester as a junior position within a software engineering department where they may be quick advancement to the primary goal of the department of software engineering. Other organizations have dedicated departments for quality testing where advancement is through generalization, specialization, or management.				

CHAPTER

TIER 1—QUALITY TESTER STORY 1.9.1

MAK KOLYBABI

I think if I had a defining moment it would be back in high school (probably 01999/02000). My girlfriend back then spent all her time on a Perl-based CGI webchat called Socko.net. Why didn't she just use IRC like a normal geek, I'll never know. This chat was run by another couple of kids our age somewhere in the USA. Anyways, the security on this thing was atrocious. I'd hang out on the chat to spend time with her—she lived across the city—but instead of actively talking, I'd spend my time poking at the chat itself, out of boredom.

The code for the chat was forked from something called JellyBeanz http://www.mindreader.com/chat/chat.cgib, and was very naive: no databases only files, minimal session management, poor anti-XSS, but this was all before I knew what a database was, to be honest. What I did know was HTML, QBasic, and Perl.

One day, the admins of the chat decided to add in session management code, but they did it completely wrong: you would be given a session ID when you first viewed the login page, but the site never associated an IP or username or any kind of state to the session ID. I explained the mistake to the admins, but was told that it wasn't considered a problem. So I wrote an HTML file with a <form> in it that had two <input> boxes: one for the username that you wanted to be on Socko.net, and the other for a valid session ID harvested from the login page.

I then showed this HTML file to one of my friends, who refers to himself even today as The Designated Asshole of any group he's part of. He then dedicated something like a week of his life— I think it was Christmas or summer vacation, so he was on his computer every waking moment—to round-the-clock trolling of this chat. Every time a user—even an admin—logged onto Socko.net, he was waiting and would use the HTML file to become them, change their nickname to SAXON <number>, which would boot them from the session, and then /kick them with the reason "stupid".

The end result of that week was that the admins bulletproofed their session management code, banned my asshole friend from the chat (deservedly), and I learned the importance of providing a PoC with bug reports and **not** sharing information with asshole script kiddies.

CHAPTER

1.a TIER 1—SUBJECT MATTER SPECIALIST

INTRODUCTION

> "A *philosopher* is a person who knows less and less about more and more, until he knows nothing about everything. A *scientist* is a person who knows more and more about less and less, until he knows everything about nothing."
>
> **— John Ziman**

A subject matter specialist is knowledgeable, skilled, and educated in some area. The area is not necessarily specific to information security but may instead be specific to information security in a particular area or vice versa. A specialist might be knowledgeable in information security applied to a particular industry. For example, financial services, health, medical, manufacturing, and retail are each quite different, but each typically uses identical information security technologies but applied in different ways.

Subject matter specialist hours can be long due to the work needed to maintain the specialization and being the only person who can do the job. The subject matter specialist may end up traveling a lot, possibly on short notice, for the same reasons.

Stress can be low because the job is well-understood and well-managed, or can be high due to high expectations and work load. Good management can make all the difference.

Stability can be high because no one else can do the job. However, some kinds of expertise age quickly and thus can also be unstable. For example, web application development knowledge can very quickly become out-of-date due to changing fashions and fads in languages, frameworks, and application servers.

HOW TO BREAK IN

Becoming a specialist is easy but time consuming: just know and admit to knowing more about an area than anyone else in the organization. Unless it's part of your current job, you'll have to do this learning separate from work. In the IT industry, there is a strong bias toward new tools and technologies, but most real work is done by legacy systems. Trying to be a specialist in the very latest technology can work, but not only does it require constantly keeping up-to-date, there may also be competition with the rest of the organization. For a very small subject area, this is easy: just find something that is so obscure that no one knows or even wants to know anything about it. However, if no one wants to know it, the information might not be particularly useful, and thus would be difficult to get paid for knowing it. But just because no one wants to know something doesn't mean it's not worth knowing. For example, in an organization dominated by a single vendor or technology, the organizational culture may look down upon any alternative vendors or technologies. Outside the organization, knowledge of these vendors or technologies may be nothing special, but within the organization such specialized knowledge could be the path to distinction despite being looked down upon. Choose your specialization carefully.

An overlooked, or unpopular area could be desperate for a specialist. A specialty might not be technology, but instead be a different part of the company, a geographic region, or even a particular work schedule. For example the finance department with a special COBOL environment that no other IT people want to admit even exists; backwater field offices that lack prestige; highly unpopular overnight, third shift, weekend, and holiday operations.

A useful amount of knowledge and skill usually takes quite a while to learn. But there are some areas where there are so few true experts that it takes little effort to become a relative expert. Finding these areas either within your organization or outside, is itself specialized knowledge. If you and others can't find the specialized knowledge in an organization, consider becoming the meta-specialist, the specialist who finds and knows the other specialists.

Some specializations are highly dependent upon the organization. For example, retailers often need people who know the Payment Card Industry Data Security Standards (PCI-DSS). It could be a particular information technology, an information security domain, technique, requirement, or an industry. In some organizations, information security is itself a candidate for subject matter specialist.

Consider how a specialization may fit your goals. Learn about that area and see if it's a fit for you. Find subject matter specialist or even better a subject matter specialist (SME) inside or outside your organization. Find a mentor/master. See the "How to Learn" section in this chapter for more details.

Learn from an existing specialist or expert in the organization. Determine what existing subject matter specialists or SMEs the organization may have. An organization chart, HR career path documentation, and the organizations mission, goals, and other direction can help. See Josh More's book *Job Reconnaissance* for more details. If there isn't a specialist or SME already, this is an opportunity to become that person.

Some organizations are too small, or have a corporate culture that favors generalists; they will not benefit enough from or will not appreciate a subject matter specialist. Some organizations can't afford the resource costs of specialist, even taking into account the benefits.

Find an area in the organization that has no subject matter specialist, but for which the organization has a use. Any of the job areas in this book are suitable for being a specialist, although these areas are broad enough that few organizations won't already have people who already are, or are becoming specialists—but if it's open, it will need to be filled, so take that opportunity. Usually, however, more specialization is needed. A specialist could specialize by industry, technology, vendor, or technique, or be performance-focused.

HOW TO IMPROVE YOUR SKILLS

Growing as a specialist can be by depth or breadth, or both. A specialist who goes for particular depth will learn even more about a smaller detailed area. For example, a database management administrator specialist (DBA specialist) is usually already a specialist in a particular database, so the next step might be another function of databases, such as security. However, security can itself be divided into further areas, such as Confidentiality, Integrity, and Availability. DBA specialists usually already have some attention paid to Availability, but can go deeper to High Availability, or further still to Fault Tolerant. This level of detail also usually requires special hardware, so that is a further specialization. At some depth, going wide again may be the right choice. This is highly dependent upon the organization, and future job direction.

Specialists may not have the opportunities to do hands-on work, but that's how they originally learned, so it can be important to go out of the way to Do. A Coder-Developer specialist could decide to port older code to a different language. Such porting doesn't have to be strictly a learning exercise with proficiency in both languages. It can also result in a useful product that is easier to maintain. Use your existing knowledge and keep it current by applying it in new ways. The DBA specialist may decide to build an entire database environment from scratch, using current practices. This should be easy for a DBA, but if it's been a while, things may have changed.

Specialists have a lot to offer, and Teaching others even a fraction of their expertise can be valuable. This can be done through personal instruction, mentoring, documentation, and cross-training. A Quality Tester specialist can Teach someone else, perhaps even in another role, such as Patch Management, to do regression testing of a patch and troubleshoot problems.

RECOGNIZING WHEN YOU'RE STUCK

It's best to recognize before becoming stuck, as specialists tend to be valued in their area but may not have sufficient qualifications to transition to another area and be as well-regarded, and thus comparably paid. Particularly bad is staying in one area long enough to become specialist merely by default, but without sufficient skills to go elsewhere.

HOW TO GET OUT

Some organizations may allow a subject matter specialist to move laterally, where aspects of their experience can be applied in new ways. For example, a database performance specialist might move into database security, or a backup and recovery specialist might get into forensics. Doing well in a specialization may also provide the credentials to go into a completely different area.

A specialist might move up to Subject Matter Expert (SME), see Tier 3, in the same specialty, this might otherwise be the same work but also might have to be at another organization.

Table 1.a Role at a Glance—Subject Matter Specialist

Hours	Travel	Stress Level	Creativity	Flexibility	Stability
8–12 hours/day after hours	Varies None to High	Low to High	High	Medium	High
General job duties	Research. Documentation. Mentoring Teaching				
Learning	The SME can be by definition limited in learning as there may be little more to learn. Although, few areas so small and so well known that there is truly nothing more to learn.				
Advancement	An obvious next step is Subject Matter Expert (SME), if that position is already filled then keep learning from the SME. If you can't learn any more in that position but can't get promoted to SME, consider Chapter 4.0 Boosting especially in your subject matter area to prepare for new opportunites. Also consider lateral moves to a complementary position within Tier 1, or leverage your subject matter knowledge to Tier 2.				

CHAPTER

TIER 2.0—DO

2.0

DOING

> "Life can only be understood backwards, but it must be lived fowards."
>
> **— Søren Kierkegaard**

The second tier of Learn/Do/Teach is, unsurprisingly, actually doing things. From a career perspective, if you never do anything, you don't have a career. Without a career, you can't prove yourself to prospective hiring managers, and the opportunity will instead go to those who can prove themselves. More importantly, if that learning is not applied, you will never know if you're right. Learning builds upon learning, and if you build knowledge on the top of faulty knowledge, the entire construction can fall down at the worst time.

By focusing on actually doing things, either with work or personal projects, you'll get feedback on incorrect assumptions. For example, bad network routing has immediate feedback; traffic simply will not flow. But in general, a defense cannot be tested until it meets an adversary, so you may wish to build adversary-style testing into your Do process. This will be easier for some topics than others, but the more you test as you go, the less time you'll spend going down dead ends.

TEST-DRIVEN DEVELOPMENT/SPRINTING

Developers often start with basic requirements such as "Application A must have feature B." This is easy to do when application A has few features. However, the more functionality you add to an application, the more you risk breaking an existing feature. To reduce the chances of this happening, some developers have embraced a concept called "test-driven development." With this approach, each application is assigned a suite of tests that must pass for the code to be accepted for any further development to occur. This can be applied to the Doing phase, although the particular testing technologies may not apply outside of the development world—but the core idea is the same.

When you are working on anything new, it is very likely that you don't understand it as well as you think you do. Verify your knowledge and skills later by creating a list of goals first. Such a list makes it easier to stay on track, to know when you've accomplished your goals, and know when you haven't.

This approach can be adapted into the time-based sprint method. The tasks needed to reach a goal are separated into sprints, with each sprint taking a fraction of the total time spent reaching a goal.

Between each sprint is a break to rest, test, review the past sprint, and to plan future sprints. Although some goals can be reached with a single sprint, this has several disadvantages. If the goal isn't reached, the time needed to accomplish the goal was wasted. Worse, it probably isn't clear when and where any problems occurred.

If, instead, the time spent is divided into multiple sprints, and each sprint is followed by a test, then when a test fails, it's clear when there is a problem. Once the problem is resolved, another sprint may begin.

By moving from Learning to Doing, your path constricts somewhat, but to greater advantage. Using sprints may feel awkward at first, but over time, you can get a lot more done in a lot less time by using a time clock. Start by building a small list of goals to accomplish in a day. At the end of the day, refer to your list to see which ones you got done. Then, as you improve at it, try the same thing for half a day, an hour, or 15 -minutes.

As you get better at these sprints, you'll be learning more quickly, but also creating work that is useful to others as each sprint is linked to a desired feature. This puts you in the position of building a portfolio to show others, as well as creating a group of hopefully grateful users. Both of these improve your leverage when trying to move into another job. It's important to track this over time, because, unlike other fields, information security is necessarily a multidisciplinary world.

INFORMATION SECURITY AND SILOS

The word "silo" is used a lot in business to indicate that a person or team is stuck in their area of expertise, and is unable to work with other people in other fields. In technology, for example, it is common for someone who has learned how to code in Java to avoid working with .Net. It's common for someone who works on the help desk to resist working with the server team, even if working together would allow both teams to solve a problem more quickly.

This tendency is, in part, due to corporate structures that tie salaries and bonuses to measurements that do not support cross-disciplinary work. However, it is also true that many people just get uncomfortable when forced to look at things in different ways. However, as an information security professional, you must resist this. To be successful over an entire career, you must know the basics of many different disciplines, such as being a database administrator, programmer, or project manager.

In the professional world, if you focus on a single skill group, you can get extremely good at it. If you don't specialize, you're unlikely to ever get as good at any given one as you would if that's all you did. A lot of people are singularly focused. After all, if you have to solve a COBOL problem, you want the best person in the lead. However, there isn't a lot of COBOL left, nor other very large but otherwise uncomplicated problems. In a world with blending boundaries and increasing complexity, you need to know a little about a lot instead of the other way around.

In the era we called Y2K (ironically brought about due to abbreviation of a date field), programmers with high skill levels in COBOL and other legacy systems were in high demand. However, Y2K was solved, and since then, there's been little for those legacy experts to do other than sit around and reminisce about the olden days. Large critical problems aside, it's often safer to hedge your bets and build a varied set of skills. That way, if the really big problems are solved or become irrelevant, you can still find work and continue to advance in your career.

This is particularly true for the information security field. In just the last decade, information security has moved from mostly firewalls and anti-virus to include patch management, log review, server hardening, workstation hardening, honeypot design and management, programming, compliance analysis, user training, penetration testing, social engineering, and so on. We've also lost the need, generally speaking, to manage Windows NT, Windows 2000, SGI IRIX, HP-UX, HP MPE, PR1MEOS, and the like. Those, so to speak, have died of old age.

Information security has largely become a multidisciplinary field. There will always be room for the best penetration testers, social engineers, and cryptographers in the world. The problem is, they're the best in the world, and to compete with them, you must also be the best in the world. But since people learn from doing, those people at the top tend to remain at the top because they get the most chances to practice their skills. So, not only is it hard to get the skills you need, it's also hard to get the experience to prove it. If you're aiming to displace those at the top, you face a very hard challenge to exceed their skills. The answer to this is to maximize your skills.

SKILLS

If we assume that any skill can be grown, the concept of boosting skills applies in the same way as boosting your career. Consider an approach if you want to become reasonably competent with firewalls, starting at zero, knowing nothing of networking. A reasonable breakdown of skill boosting may be:

1. Learn the network protocols including IP, TCP, UDP, and ICMP, where they're used and generally how they work and how they're vulnerable.
2. Learn basic IP routing and how IP subnets work and how to attack them.
3. Understand the different ways that firewalls can terminate and block connections and their limitations.
4. Create and implement a basic firewall policy that is aware of internal targets and also maps allowed connections.
5. Troubleshoot and attack a basic but otherwise unknown to you firewall policy.
6. Troubleshoot and attack an advanced but otherwise unknown to your firewall policy.
7. Migrate an advanced firewall policy from one type of firewall to another.
8. Attack and bypass basic and advanced firewall policies.
9. Defend against the previous attacks.

There is no maximum level. You can always learn something in greater detail, but every advance will take more effort. It could take a lot of effort to reach level 7 in the above list, but if there's no need for you to gain that particular skill, it may be wiser to invest that time in something easier to obtain, such as a patch management skill boost. Balance generalist and specialist skills with your current and future job requirements.

A common rule of thumb is that it takes 10,000 hours to become an expert in a subject. This is, of course, a gross oversimplification of the research; but since it suits our purposes, we're going to promote the concept. Thus, if you worked on something for just two hours each evening, it would take just over 13 years to become an expert. If you really focused and did four hours each evening and eight hours each weekend, you could get it done in half that time. Still, more than six years is a long time to wait, and by then that expertise might no longer be relevant.

Instead, focus along the easier paths. It's just as hard to find an expert in carrier-grade Cisco switch security as it is to find someone who is decently skilled in firewalls *and* Linux *and* patch management *and* social engineering. However, it might only take 1,000 hours in each of these areas to reach a level of "good enough." It can take less than a year to put in this kind of time, and you get to use that skill immediately. Spreading effort and skills around means greater employability sooner. For example, only carriers need carrier-grade Cisco switch security, but nearly all companies need at least one or more skills in firewalls, Linux, patch management, and social engineering.

THE RISK OF A MULTIDISCIPLINARY APPROACH

You'll probably be rather upset with yourself if after twenty years you had only basic skills in twenty different areas. While it is true that there aren't that many jobs for an expert in carrier-grade Cisco switch security, it is also true that it's hard to find jobs for someone who is competent but not expert in twenty different areas. To succeed, you must hit a happy medium for the job you want.

There are many career planning books on the market. Most of them say trite things like "you can't expect anyone to plan your career for you but you," "remember, the key is hard work and hard work is the key," and "those who fail to plan, plan to fail." While these sayings are true, the approaches they explore are woefully incomplete, especially in an emerging and highly technical industry. Instead, here are our two trite rules about careers:

1. Careers only ever make sense in hindsight. What you did in the past will make the most sense in the future.
2. Don't let Rule One make you stupid. Do what makes most sense when you're doing it.

We've structured this book in terms of jobs and job families. The Learning section jobs are generally entry-level ones. Once you've done an entry-level job or two, you're ready to move up. It's easier to move within a job family, but with some effort it is also possible to move into a higher-level job within another job family.

Story is key. There is no point in doing work to advance yourself if you can't tell your story. In almost all cases, a well-told but otherwise unimpressive accomplishment wins against a poorly told but impressive accomplishment. As you move through your career collecting projects and experience points, think about how you'll tell these projects' stories. Once you start thinking about your performance and how it affects others, you'll massively improve your ability to capitalize on your accomplishments. This allows you to treat your career, and therefore your path in information security, as a series of building blocks.

It is unreasonable to expect to become a team lead in penetration testing with absolutely no experience. However, there are many paths to such a job. For example, a log analyst may get a system administrator job doing server hardening. This may be followed by vulnerability analysis, then junior penetration testing demonstrating the vulnerabilities. Doing penetration testing for a while may result in a promotion to a team lead position. Another common path is to start in the help desk, go through organization layers, and become senior or supervisor help desk worker. This experience in understanding how people use systems can branch out into social engineering and eventually help penetration testing teams. Yet another approach is from development. Many developers start out extending and maintaining legacy systems. Over time, they write more and more code and develop ever-more-intricate applications. This experience primes them for application analysis, which allows an "in" into penetration testing.

Each of these examples shows a process of learning, and at certain points you have to tell a good story to achieve the next tier. In other words, it looks much like this:

Table 2.0.1 Information Security Career Paths

Standard Path	Help Desk Path	Developer Path
Log analysis	Help desk	Maintenance programming
Server hardening	Subject matter expert	Project development
Vulnerability analysis	Help desk supervisor	Product development
Penetration testing, junior	***Social engineering assistant***	Product management
Penetration testing, team lead	Social engineering, team lead	***Application security analysis***
	Penetration testing, team lead	Penetration testing, team lead

The turning points are at the steps ***in bold italics***, where you must successfully tell the story of your previous achievements to convince someone to let you make the lateral move into the role you want. It's likely but not certain that some of these moves will be in different organizations or even different industries. Promotions and raises are often easiest and best obtained by changing employers. Few employers grow and change in the same way as their employees.

OTHER CAREER PATHS

A lateral career path will take you outside your main field into something else. There are more ways to get into information security than we can cover in this book. Instead, we'll explore just nine different lateral paths here in Chapter 2.0 Do and five more in Chapter 3.0 Teach.

The first nine lateral paths are Physical Security, the Military, Law Enforcement, Legal, Sales, Project Management, Non-IT Engineering, Accounting, and Business Analysis. Each of these fields has its own way of thinking and will give you insights that people with more traditional paths won't have. Some of these fields of thought will work against you, but others will be extremely valuable.

For example in the military, there are things you Do and Don't Do not found in the private sector. Many military workers function more "by the book" than those in the private sector. This experience can result in more efficient operational practice, especially since "the book" may be an actual book of required formal processes and procedures. However, it can also cause problems when the book has to be written or revised as is common in information security. Militaries often depend upon empowered, strong, and decisive leadership to overcome issues like this, but this is often lacking in the private sector. Many people leaving a job doing military log analytics may consider their coworkers in a civilian Security Operations Center (SOC) as sloppy with inconsistent process and poor discipline. Start-up entrepreneurial environments promote an element of creativity often lacking in military circles.

What you Learn is reinforced by what you Do, so if you take a lateral path, put some careful thought into how you want your thinking to change and what you want to keep. Consider that your thinking about this will itself be biased by what you've Learned and Done, and may also need to change.

Each of these paths will be explored in greater detail in the chapter assigned to it.

BOOSTER PATHS

Building experience is hard and takes time, but there's only so much time in a day. While you can maximize your Learning and Doing time by focusing on one thing at a time and minimizing distraction, there is a practical limit to how much you can work.

The phrase "work/life balance" is popular today. However, if you want a job, your competition consists of people all along the work/life balance spectrum. If you want to win, you have to have Learned more and Done more than anyone else and be good at talking about it. To reach this level, you must have worked longer or harder at work or on your own time. If you don't want to wait to break in, you must plan and execute projects on your own time.

In other words, you need a boost. A boost path is a carefully chosen project that will reward you both in the present and in the future, for example, working as an author, a developer, an evangelist, a researcher, or a community supporter. This is because evening and weekend work tends to be better-defined and result in a higher experience-to-time ratio than your day job, where you also have to attend meetings, answer ignorant questions, and fight fires. It may also be much more enjoyable. Beware of conflict-of-interest issues between work and home. Some organizations have very specific rules on doing work at home, using work resources for personal use, or using personal resources for work.

Chapter 4.0 Boosting discusses this in greater detail.

HOW TO DO

We've talked a lot about the importance of Doing and what to Do, but not much on how to Do things. There are at least as many ways to do a thing as there are people on the planet; some are more right than others, and at least some of them are simply wrong. However, there are some fairly common issues that can get in your way as you try to get things done. There are various tips and tricks to dealing with such things. Many people seem, in the course of their careers, to discover the concept of time management and be astonished at their ability to suddenly be highly productive. This is not a time/work management book, so we'll trust in your ability to use an Internet search engine (and the Appendix of this book) if you need detailed help with that. Instead, we'll briefly discuss how to maximize the Do portion of your career in different ways.

DOING AT WORK

If you're like most people, most things you'll be doing will be for your primary employer. When you're new to the work, there won't be much to this, as you'll be told what to do and, if you're lucky, when to have it done. Sometimes your work will be more collaborative as you try to solve problems. As you gain experience, though, more and more of your work will be self-driven, and you may even find yourself telling others what to do. To maximize your experience, you have to know how your workplace functions.

Broadly there are two kinds of organizations: "do what I say" and "help me solve this problem."

Governments, militaries, law enforcement, prisons, manufacturing, retail, K-12 elementary schools, and other authoritarian organizations tend to be "do what I say" organizations.

Colleges, universities, museums, research institutions, and some nonprofits tend to be "help me solve this problem" organizations.

If you work in a "do what I say" sort of organization, think about the business importance of what you're doing. If you don't understand why it's important, then ask—but not at the beginning. Bosses in "do what I say" organizations seldom reward questioning. If you ask questions as you go and actively solicit feedback at the end, your work will improve faster and you can avoid this career-limiting move.

If you are working in a "help me solve this problem" organization, though, ask questions first. In these organizations, try to understand and discuss the problem as much as you can at the beginning, but be aware that some problems can't be understood until you experience them. Try to identify what's known, what's unknown, and what simply can't be known yet. Then, discuss with your co-workers on ways to identify what can't be known, and build a series of small explorations into the project plan. Most people don't do this, but if you do, you can not only rise above them but also maximize your experience gain.

If you are self-guided then create projects that help your organization where you get interesting stories you can tell later. By working within your job description, but constantly pushing the envelope, you can gain more experience both in doing things and in convincing others why they matter. This will make your leap into your next job easier.

Finally, if you are guiding the work of others, remember that each of them will be in one of the above categories. If they try to work without an understanding of why what they're doing matters, help them to understand. If they try to go off on their own, make sure to pull them back in so their assigned work that you are guiding them in gets accomplished. Ideally, you will both work together to achieve both your goals. If, however, this cannot be done, determine that their new goals are not incompatible with yours.

In the end, the work you do for your employer will be limited in both scope and Learn/Do potential. It may be necessary, but most tasks that are high in learning aren't cost-effective from a business perspective. Also, cost-effective tasks are often less interesting from a storytelling perspective than work done for other reasons. However, some organizations value cost-effectiveness and saving money can be a story all by itself, so pick your stories accordingly. You can generally learn more Doing things on your own time.

DOING ON YOUR OWN TIME

Working on your own time trades structure for flexibility. In a work environment, the structure you need is often provided for you. When you work on your own, you can easily get distracted, go down dead ends, and fail to polish the work once you're done. If you want to be successful, you have to keep yourself under control.

Start each project you do on your own with a fairly clear idea of what you're doing and why you're doing it. Some tasks never really end, but this helps you to identify when you're done, that you've done enough so you can get to the next task. Remember, it's okay to do something just to learn about it. However, if you can take it the extra step where it helps out someone else, you've turned it into a story. Having someone to help also sets a particular goal and provides someone to evaluate your work. Set aside a recurring time to do the work, or least think and plan about it. Having someone to help can also help set a schedule. Daily, weekly, or biweekly recurring times can work well. More frequent sessions mean less time to forget what happened in the previous session, but may require shorter sessions. Longer sessions may be more efficient, with less average overhead in starting and ending a session. Experiment and learn what works best for you.

It's tempting to only focus on the fun part of the work. However, the reason it's the fun part is probably because you already know how to do it. Learning is painful. Doing something while you're learning about it is even more painful. Fitness coaches have it right when they say to "embrace the pain" or "no pain, no gain." Only by doing what you don't know will you learn what you need and do what's worth doing. Remembering the goal will help to keep you on track, but preparing yourself for pain and frustration will help even more.

Anyone watching kids grow up will see a series of changes that are all prefaced with frustration and anger. When you learned to talk, you probably didn't just leap straight into complete sentences. When you learned to walk, you probably fell down a lot. The first time you went to school, learned to drive, went on a date, moved to a new place...each milestone is scary, frustrating and hard. This doesn't stop as we get older, we just get better at masking our pain when we talk to others. As adults, we get to feel the same pain we always have, but now we compare ourselves to people who seem to have it all so easy, making things even harder.

Remember that it's hard for everyone. All experts were once beginners. Your co-workers whose jobs you covet didn't always know what they know now. The trolls on the Internet who specialize in making beginners feel stupid are often unwilling to put in the amount of work needed to get where you want to go, and soon you'll surpass them.

When doing work on your own, your biggest enemy is yourself. The storytelling model goes both ways. While you're working toward being able to tell other people "I'm awesome," you may be telling yourself "I'm not awesome." The one true trick of being able to be productive while learning and doing things that are entirely new is to be aware of your internal monologue and add "...yet."

Yes, you don't have the job you want ... yet. You're not an expert ...yet. You're not awesome ...yet.

But you will be.

WORKING WITH OTHERS

It is likely that some of the work you'll be doing will involve working with others. In general, remember that each person has their own motivation. Many people are driven by money and only Do things when compensated. Some people work to make a difference in the world, and others, to increase their social status.

You must keep your goal in mind when working with others within a Learn/Do/Teach model. When you're Learning, are the others helping you or hindering you? If they're Teaching and you're still Learning, there may well be a good match. The same applies if reversed. Similarly, if you're both Learning or Teaching at the same time, you can discuss each others' ideas to make progress.

If all of you are Doing, consider whether or not you will get in each other's way. It is common when working collaboratively for each person to take the work that is easiest for themselves. This gets the work done as quickly as possible, but it also drastically weakens your stories and entrenches your current position, hindering your ability to move forward. Similarly, if you are Doing and someone else is Learning or Teaching, you may find them working toward understanding while you are working toward storytelling. This will not work well, and you may find yourselves fighting. The following chart shows these interactions:

Table 2.0.2 Learn/Do/Teach Interaction

	Learn	Do	Teach
Learn	yes	-	yes
Do	-	maybe	-
Teach	yes	-	yes

Basically, Learning and Teaching are compatible with one another, because they are education-focused tasks. Doing is only compatible with Doing, and even then you have to be careful. This is a cause of poor on the job training. When one person is trying to understand or get another person to understand, but that person only wants to get things done, there will be conflict.

Thus, when you have to work with others, consider where they are with respect to the project and see if their position and yours are compatible.

MAKING MISTAKES MATTERS

Doing work means making mistakes. This shouldn't come as a surprise. After all, you are translating theoretical knowledge into practical knowledge. Anything that you didn't understand completely will eventually result in an error. It is better to fail fast, before bad knowledge is applied for the long term. And yet, in an environment where a boss views *your* time as *their* money, any mistake feels to them like money lost. This puts a lot of pressure on people and, in effect, slows down the learning process—which, ironically, costs the bosses more money.

WHY MISTAKES MATTER

Fundamentally, a mistake shows you where you misunderstood something, so you can correct it. The faster you make mistakes, the faster you learn. It's that simple.

As an example, when one is writing a nonfiction book, one is expected to use a certain type of language. One is not supposed to be emotional. One is not supposed to swear. Violating this rule is frowned upon by editors and publishers. However, this is an emotional topic and one, I (Josh More) feel must be addressed emotionally. So, in the plainest language I can use...

Mistakes. Suck.

Many of us are perfectionists. Many of us are extremely intelligent. We know what we want and we know what we have to do to get it. After all, that is why you bought this book. Perhaps you're upset when you see other people getting the jobs you want. Perhaps you want to prove yourself or bring home more money. Perhaps you just want to do something more interesting. The key word here is "want." Wanting hurts. The more you want, the more pain you'll feel. Odds are that you are living with a constant level of emotional pain. It's not the same as living with a chronic disease like fibromyalgia or rheumatoid arthritis. However, it does mean that you have a certain threshold of pain that you just deal with every day.

Each mistake you make and internalize adds to that.

If you move from Learning to Doing at your maximum speed, at some point, you will exceed your limit. You will break. Your projects will fail or stall and you'll get stuck. This is, fundamentally, how humans work. And, in fact, the more superhuman you think you are (a common issue in IT and information security), the worse you'll break. It's like driving along a mountain road. The faster you go, the more likely you are to drive off a cliff. The exact people we want in this industry, smarter people, have this vulnerability—the more they know, the more multidisciplinary they are, the more comfortable they are driving quickly...the more catastrophically they will flame out. And in so doing, they'll learn their breaking point and what happens when they do, but it's much better to learn before crashing.

Unless you take the time to recover from and understand your mistakes, each one you make gets you closer to your breaking point.

THE VIOLENCE INHERENT IN THE SYSTEM

Information security is dangerous, but we don't like to talk about it. There are different types of Internet crimes. In commercial information security, we mostly think about adversaries as malware authors who are after our money or data. In education, the adversaries are often the institution's students. In the military, adversaries are foreign countries. The truth is that our attackers are all of these, and more. When you're first investigating an incident, the people who created the issue could be any of:

- A technical competitor, employed to steal your data.
- A technical activist, motivated to attack your infrastructure for ideological reasons.
- A technical thief, living off of what they can steal.
- A kid who doesn't know what they're doing.
- A kid who does know what they're doing, but doesn't care.
- A criminal, working for an organized crime syndicate.
- A warrior, working for a military organization.
- Someone you haven't even considered.

There is never a single response that is appropriate to the entire range of adversaries. This is one of the first areas where you can make a mistake. When you make assumptions about an adversary, you run the risk of the wrong response either from yourself or from them. If you're from a finance background and move to education, you should know that it's not appropriate to call the state police on a student who is exploring the network. However, that's not what your gut will tell you. Your gut "knows" who your enemy is, and will drive you emotionally, not rationally.

You can often pick up on things intuitively, resulting in a much faster analysis. The best analysts can't really explain why they think as they do. Sometimes things just feel "hinky" or just wrong. Know when you're operating intuitively and when you're operating rationally. This allows you to use your entire gamut of responses, from rescue to discussion to lawsuits to jail to military action, as appropriate.

WARNING

Violent Risks

While we hope that you are never engaged in physical violence, be aware that such a response is possible for militaries and organized crime syndicates. Know that we live in a world where attacks over the Internet can be digital, emotional, and physical. Some adversaries have killed. Some victims have died, either by their own hand or that of another. If you're going to play the game, understand what you may be called upon to do, and understand your limits before you must meet them.

As you shift from Learning to Doing, slow down. Realize that everything you try has a risk of failure, and think about the consequences of such a failure. Security is not about blocking the bad guys or taking control. Security is about protecting. The instant you take a risk that puts those you are protecting at risk, you're no longer a security practitioner. You become the enemy.

CHAPTER

TIER 2—PEN TESTER 2.1

INTRODUCTION

> "Men rise from one ambition to another: first, they seek to secure themselves against attack, and then they attack others."
>
> **— Niccolo Machiavelli**

Pen, or penetration tester. Getting paid to hack other people's systems. To many people, this is the dream job that attracts them to the information security field. The idea of getting paid to behave as a criminal is very enticing.

But it is hard. You must have strong technical skills. The minimum level of knowledge includes at least an understanding of networking, Microsoft Windows and Linux operating systems, Microsoft IIS, Apache, Tomcat, and nginx web and application servers, many testing tools including Metasploit, Nmap, BurpSuite, among others, and script writing. You will also need decent communication and social skills.

This list may seem daunting, but these are the skills the criminals possess. When pen testing an organization, one must be able to breach its technologies. Without a strong breadth of skills, your marketability and utility will be limited.

While pen tests used to be exclusively technical, social engineering has become an expected component of penetration tests. Being comfortable making phone calls, talking your way into buildings, or otherwise acting in various roles is required.

Beyond the technical, you will be expected to have other skills. After the pen testing you must provide understandable reports. This requires good writing and perhaps presentation skills, attention to detail, and a willingness to do the necessary drudge work. All reports should have non-technical executive summaries, but sometimes entire reports need to be written for and understandable by non-technical staff.

Most of pen testers are found at consulting companies; however, as regulations and standards increasingly require penetration tests, more and more organizations are adding internal pen testing teams.

HOW TO BREAK IN

The path varies, but the core process is generally the same: first develop general technical skills, then develop security knowledge, then develop pen testing skills. It's possible to break into systems with no technical skills or security knowledge, by using free and commercial tools and scripts. Someone with this minimal skill level is called a “script-kiddie,” because they are so inexperienced they can only run the scripts, but not read them, understand, or write new scripts.

Technical skills can be developed through the Tier 1 jobs in Chapter 1.0 Learn. The key is to develop the breadth of technical skills required for pen testing. If you only understand networking, then exploiting a Windows IIS web server will be challenging. If you only understand Windows and its platform, doing initial reconnaissance is going to be challenging, because customers will expect you to be able to break into any system. If you don’t know a system, you will be doing them a disservice if you imply they are secure just because you happen to be ignorant.

Take advantage of any opportunity you have to gain breadth of technical skills.

While developing technical skills improve your security knowledge. Understand not only the technical components of security, but the overarching security concepts that drive the security decisions in your organization.

HOW TO IMPROVE YOUR SKILLS

As security advances, more and more tasks are automated, and new tasks enter pen testing. Network reconnaissance used to be an entirely manual process; then tools such as Nmap semi-automated the process, and testers’ knowledge went from understanding less about network scanning (although some knowledge is still required), and more to understanding the tools.

This type of growth is continuous within the realm of pen testing. One of the best ways to improve your skills is to learn how to automate as many tasks as possible, leaving you able to focus on other areas during the pen test.

RECOGNIZING WHEN YOU’RE STUCK

Pen testing requires running forward merely to stay in place. New vulnerabilities are discovered daily, and must be incorporated into your testing frameworks. If you find yourself relying on older exploits, or older tools, then you are quickly working your way to irrelevance. If you are hearing about vulnerabilities from non-pen testers, then you are probably falling behind.

HOW TO GET OUT

The breadth of technical skills you have as a Pen Tester provides opportunities to move back into technically oriented roles (such as the various Systems Administration or engineering roles), should that direction be of interest. Similarly, moving to a role such as Security Architect (defense), Pen Test lead (management), Risk Assessment (less technical), or Auditor (business-focus) are possibilities. The key for any move is to emphasize how the skills you have developed will be appropriate in your new role.

Table 2.1 Role at a Glance—Pen Tester

Hours	Travel	Stress Level	Creativity	Flexibility	Stability
8–16/day	Low–High	Med–High	Moderate–High	Moderate	Moderate
General job duties	This job consists of performing scans, identifying vulnerabilities, performing social engineering and generating post-test reports. Along the way, you may find yourself creating new exploits, doing research and presenting at conferences. Basically, you have be able to keep pace with the attackers, which means doing what they do. When on engagements, you will need to work at odd hours, and frequently work long days in order to ensure appropriate testing is in place. Travel is low for internal team-based roles, high when working for a consulting company. International travel is common in this role. There is added stress. Did you find everything? What if someone comes in immediately after you? What if someone was already in, and you didn't find the way in that they did? Be prepared to have these questions floating around in your head. The high level of technical expertise required makes this somewhat stable. However, as pen-tests may be optional for some organizations, that does make the demand somewhat economy dependent.				
Learning	Learning is required in order to stay relevant in this role. You will be able to learn from both your peers and general research in preparation for each engagement.				
Advancement	Demonstrated technical and leadership abilities will present opportunities in other roles.				

CHAPTER

2.2 TIER 2—VULNERABILITY MANAGEMENT

INTRODUCTION

> "vulnerabilities arise at the boundary between two protection technologies"
>
> **— Ross Anderson**

The Vulnerability Manager works closely with other information security roles to determine, often in real time, the risk of any and all vulnerabilities that exist in an organization or a particular environment. Real-time response can be critical, as there may be a very limited time to understand, create, test, and deploy any solution before a vulnerability is actively exploited. In the case of zero-day attacks, the vulnerability is already being exploited against production systems. However, there is opportunity to engage in mitigation before the attacks occur by understanding higher level weaknesses. A vulnerability manager must balance the risks of continuing to operate vulnerable systems, even as they might be under attack, against the risks of shutting down systems and no longer being able to operate as an organization. As a role, vulnerability management may not be a dedicated position but instead be part of other positions up or down the management chain.

HOW TO BREAK IN

Vulnerability Management is, in a sense, a specialization of change management. However, it is focused on making a change that moves a vulnerability from a risk to an addressed issue—one that has been patched, mitigated, or is no longer a concern because the organization has already taken action.

The Vulnerability Manager can be a natural outgrowth from other management roles, such as Patch Management or Security Coordinator, and may be part of or a promotion from senior System Administrator, Network Administrator, Coder-Developer, Security Consultant, Security Facilitator, Security Assessment, Risk Assessment, or senior Incident Responder positions. Each of these positions feeds information to and works with vulnerability managers, providing a bigger picture of the issues of new, current, and past vulnerabilities to determine security posture and actions for the organization.

If there is no formal vulnerability management structure, any person in the positions listed above can step up when a new vulnerability is announced and take on the responsibility of collecting, organization, understanding, and suggesting mitigation paths for the organization. Many people did this with well-publicized vulnerabilities in the years 02014 and 02015 such as HeartBleed and ShellShock. One of the simplest tasks is to set up the first meeting, write the first memo, or take the first stance on next steps. However, follow-through is critical, both for the organization, and also to maintain momentum of change to get the vulnerability addressed and resolved.

There are opportunities from many other positions to become the informal vulnerability manager.

Each of these roles and positions feeds information into and receives information from a vulnerability manager. In doing so, these roles could grow into a vulnerability manager position:

Patch management may be the first to know of the vulnerability, through a patch release or notice if there is no patch. A patch is very often the only solution to a vulnerability, so such experience is critical to understanding the limitations of patching, including minimum time, risks of patches, and required outages.

System Administrators and **Network Administrators** are on the front line in dealing with vulnerabilities, including configuration changes that can avoid and perhaps prevent a vulnerability. They also contribute their own take on patch management.

Coder-Developers may be involved with the creation of a patch for the organization's own software. They can be the first to know of vulnerabilities and may also provide a patch or update at the same time.

Quality Testers and **Quality Assurance** provide input on risks of applying patches or configuration changes, the results of quality testing and assurances processes, the risks of not using such processes, and how the vulnerability can be prevented in the future. In some cases, they are the first to know of vulnerabilities.

Security Assessment provides input on the current known state of the environment and how the new vulnerabilities impact it. Then, after configurations changes and patches are completed, they follow up by checking for the same and any new vulnerabilities.

Risk Assessment brings all the risks together and provides input on what risks can be taken by the organization, which then goes to the coordinating vulnerability manager.

Incident Responders and **Log Reviewers** may be the very first to encounter the results of an attack such as from a zero-day vulnerability, and provide critical information for vulnerability management as to the current status.

Log Reviewers may provide information on what happened in past but previously undetected attacks.

HOW TO IMPROVE YOUR SKILLS

Every new vulnerability presents a new learning experience. When you deconstruct a vulnerability, you will often find it is made of many different issues, each one of which requiring study and analysis. The worst vulnerabilities tend to be from very different sources. For example, the top vulnerability of one week might be entirely client-based and not affect servers at all. Another might server only and have no effect on workstations.

Vulnerabilities are often the results of a failure of imagination, so understanding, managing, and fixing them requires creativity. For example, vulnerabilities often follow one another in an informal release cycle with creative thinking on the part of attackers helping to predict new vulnerabilities in the same class.

Research your environment and your industry, know what threats your organization faces both in terms of technology, your competition, and your customers' expectations.

Your organization's response should reflect it's goals and risk acceptance.

Communications is critical in an incident. All organizations will have incidents so effective and frequent internal and external communication can make the difference.

If your organization doesn't already have security architects doing long-term planning, this can be an opportunity for you to grow into that position. Be able to write a quick-to-read and yet easily understood summary of the vulnerability, how it affects the organization, what the organization should do, and what the consequences are if the organization does nothing.

RECOGNIZING WHEN YOU'RE STUCK

Are you doing everything you should but haven't been able to get the organization out of fire-fighting mode? Is each new vulnerability yet another crisis? Is your organization's vulnerability management process non-existent? Is there any prospect for a formal vulnerability management process? Does your organization have a formal process, but it's no longer improving?

Unless you thrive on fire-fighting and continual crisis management with no end in sight (some people do), then you may be stuck.

The stuck vulnerability manager may not even realize they are stuck in the position, since it is a synthesis of different roles. Being stuck means not being able to do the official job, and yet not being able to grow into a dedicated vulnerability manager.

HOW TO GET OUT

Vulnerability Manager may not exist as a dedicated position; consequently, it's possible to get stuck and have no way to get out because no one, even the person in the role, knows they're doing this role. If the role doesn't formally exist within the organization, it needs to be formally recognized. Once the role is realized, and the past work appreciated, then the formal role can be created.

To get out requires that the person stuck in the role realize it, and present it as a formal role in the organization and get it funded at least as a part-time role formally assigned to their position. In some organizations, this would automatically require that the role be assigned to a different person because the existing person already has a job; in others it could be a way to grow into a dedicated full-time position.

At the top of the management hierarchy, such as the Chief Information Security Officer (CISO), a vulnerability manager might be personally responsible for information security in the organization. Although this can be a stable position, there are well known examples of even a CISO, by title or not, leaving an organization because of a failure in vulnerability management. In some cases, as the top vulnerability manager in the organization, the Chief Information Officer (CIO) was fired, resigned, or otherwise left to pursue new opportunities.

Table 2.2 Role at a Glance—Vulnerability Management

<table>
<tr><th>Hours</th><th>Travel</th><th>Stress Level</th><th>Creativity</th><th>Flexibility</th><th>Stability</th></tr>
<tr><td>8/day</td><td>Low</td><td>Varies</td><td>Moderate</td><td>Low</td><td>High</td></tr>
<tr><td><b>General job duties</b></td><td colspan="5">This includes managing configurations, software engineering where applicable, iterative testing and tuning.
Stress can be high, especially when significant vulnerabilities are known. It's not unusual for this role and those this role works with to require overtime, weekend, and holiday work. This is usually the time available to mitigate vulnerabilities.</td></tr>
<tr><td><b>Learning</b></td><td colspan="5">Every new vulnerability is a new learning experience. Most vulnerabilities are a mix of issues and so there will usually be a lot to learn each time. The worst vulnerabilities that come up, perhaps even on a weekly basis, tend to be from very different sources. For example, the top vulnerability of one week might be completely client based with no server component with no mitigation or risk anywhere but on client systems. Another vulnerability might be entirely server based and similarly not have any client vulnerability. Some vulnerabilities are very vendor specific, while others cut across vendors due to common code or common design defects.</td></tr>
<tr><td><b>Advancement</b></td><td colspan="5">This is a management track role and can lead to any management roles, but especially within information security. It can lead to any of the other information security management positions. This position can grow into the Chief Information Security Officer (CISO) or higher if it doesn't exist.</td></tr>
</table>

CHAPTER

2.3 TIER 2—SECURITY ASSESSOR

INTRODUCTION

When approaching information security, one must remember that not every solution works for every company. A large organization may be able to bring in a team to do a full penetration test, including testing physical controls, compliance with policy (social engineering), and detailed analysis of legacy systems. The cost for this type of single test could easily outstrip the entire annual profits of a small company. Specialists are needed to fill this gap and help a small company understand its security profile while helping it identify and evaluate the steps required to protect its infrastructure.

Doing Security Assessment requires a different approach than pen testing. The Security Assessor is working from the perspective of the attacker; rather, their process involves more discussion and interaction with the organization.

To help explain this role, here is an example engagement.

The Security Assessor is brought in by a small organization to review its security controls. During the initial kickoff meeting, they discuss the specific goals that they hope to achieve. In this example, the goal will be to build a general profile of the adequacy of the security around the main infrastructure: network, servers, and end-user computers.

Using this as a basis, the assessor would then begin asking questions of those responsible for this infrastructure. Depending on the size of the organization, this could be a single individual or a small team. The assessor would identify the controls in place around the network. For example, they would determine whether default usernames and passwords have been changed, if systems are patched regularly, and if the configurations used are appropriate for the environment. Questions regarding how end-user machines are configured and if any malware protection software is installed will also be important for determining the company's overall profile.

After talking to the technical team, the assessor will discuss policies and procedures with HR. Is the technical team notified of staff departures so that accounts can be disabled? Are there policies in place to protect the company regarding appropriate use of the equipment? The organization's industry and requirements to meet various regulations and standards will influence the questions that must be examined and discussed.

After gathering all of this information, the Security Assessor would produce an analysis report describing the findings and recommendations as to how to address any shortfalls that were found.

This role requires a wide range of technical and business skills. The ability to understand a wide range of equipment and operating systems is required to perform a complete and thorough analysis of the client's security profile.

HOW TO BREAK IN

Because of the breadth of skills required to perform these assessments, the Security Assessor must have gained knowledge in a wide range of technical and business areas. Working as a Wildcard, as described in Chapter 2.7, is a good way to develop these skills due to its generalist nature. Similarly, if you move between roles such as Network Administrator and System Administrator, this can provide similar breadth of expertise.

Since this is a consulting role, having some form of security certification may be required. It is significantly easier to sell the skills of a consultant to a client when they have some level of certification. A CISSP from $(ISC)^2$ or a CompTIA Advanced Security Practitioner can demonstrate security expertise. Other certifications from GIAC or ISACA are appropriate as well. Which will be required will depend upon the market you are in and your target audience.

HOW TO IMPROVE YOUR SKILLS

This role provides many opportunities for growth in its regular job duties. Since no one can know every piece of equipment or software that an organization may have, you will often have to learn about the site's infrastructure while on the job. Some of this investigation will need to be done after hours, or at least not in front of the client, as they will expect a certain level of expertise. Because of this, work weeks of well over 40 hours are common.

Depending on the direction you wish to move your career, you will want to advance your skills in those areas while doing assessments. If you wish to move into auditing, then improving the questions asked, your ability to ask questions and process answers, and your work with the individuals will be important. If you wish to move into more technical roles such as pen testing, then diving more deeply into the technical facets of the assessment will be an area of growth for you.

RECOGNIZING WHEN YOU'RE STUCK

If you really enjoy working with the smaller clients and using a wide breadth of skills, then this may be an end role for you. However, many people view this as a stepping stone role to a more advanced security position. If you intend to advance your career, staying in the role for more than a few years may be indicative of being stuck in the role. If you are not advancing your skills and getting more advanced assessment opportunities, or worse, getting simpler ones, that is indicative of it being time to look at your next career change.

HOW TO GET OUT

Depending on how you have advanced your skills, a move to Pen Test or Auditor could be a logical step for a Security Assessor. For more lateral moves, moving to Risk Assessment, Vulnerability Management, or Wildcard are options, even if you have not advanced your skills while in this role.

Table 2.3 Role at a Glance—Security Assessor

Hours	Travel	Stress Level	Creativity	Flexibility	Stability
8 + /day	High	Moderate	Moderate	Low	Low
General job duties	Evaluating a small organization’s security profile. Providing reports to clients. As a consultant you will be under pressure to ensure that you have enough billable hours, while the clients will be trying to get as much as possible from you during your engagement. Because each organization will vary, you will need to adjust your questions and evaluations based on what you observe, rather than exclusively following a script. You will be tied to the client’s calendar, and working to ensure that you get the necessary billable hours a week. As these types of assessments are often optional for small organizations, the demand will vary depending upon current economic conditions.				
Learning	Lots of opportunity to learn through the job duties, if you are willing to leverage some time after hours during your engagements.				
Advancement	The experience gained through this type of position can be used to move into a wide range of positions for the properly motivated.				

CHAPTER

TIER 2—RISK ASSESSOR 2.4

INTRODUCTION

The Risk Assessor, as part of Risk Management, is responsible for assessing the overall risk profile of an organization's IT systems. Every system is associated with some level of risk. It is the job of the Risk Assessor to determine how large an impact a security compromise would have upon an organization, and work with Vulnerability Management to mitigate risks to a company-acceptable level. The organization may or may not know what systems are the most important. That may sound unreasonable, but without going through the process of analyzing the overall risk profile of an organization's systems, the company may not realize its level of dependence. It is not uncommon, for example, to find a dependency on a single system that is running out of sight, hidden under the desk of a developer.

While the specific tasks that are used to identify an organization's risk vary based on the framework used, they all have the same ultimate goal of identifying the impact to the company, should a system be compromised. The most commonly accepted categories for ways in which a system may be compromised include losses of confidentiality (sensitive data is no longer controlled), integrity (data has been inappropriately manipulated), and availability (a system is no longer available to use).

The nature of that risk will vary from organization to organization. For an online retailer, keeping the website up and running (ensuring availability) may be the highest priority. For an insurance company, keeping patients' medical records secure (ensuring confidentiality) may be the highest priority. It is important to understand that the importance of a system is not a technical decision, but a business decision.

Once the impact on the organization has been determined, the possible threats must be identified. Which is more likely to take down your system: a hurricane, an Internet denial-of-service attack, or some other threat? Each threat has its own distinct set of mitigations (or controls) that reduce the damage to the organization. Understanding which mitigations are the most likely to work, most cost-effective to implement, or least damaging to the organization will determine the order in which the controls are implemented.

HOW TO BREAK IN

Risk Assessment involves understanding not only the business, but also the systems that the organization uses. Being able to spend time talking to a business manager about the costs of potential loss (how much will it cost the company per minute when the website is down) and then talk to the technical people (such Vulnerability Management) about implementation costs will be key skills for this role.

Many Auditor skills are also used in Risk Assessment. Auditors tend to focus on standards or regulations, such as PCI-DSS, US Health Insurance Portability and Accountability Act (HIPAA), or US Children's Online Privacy Protection Act (COPPA). Risk Assessors tend to focus on business needs. If those in Audit are comfortable determining relevant standards they are evaluating systems against, then they are well set for a move into Risk Assessment, should they so desire.

There are certifications, such as the ISACA Certified in Risk and Information Systems Control (CRISC), that may be required by some organizations; however, being able to relate your current experience to the job tasks of identifying risks and being able to recommend/investigate controls will often be more important.

HOW TO IMPROVE YOUR SKILLS

This type of role requires growth simply to keep pace. Business drivers change. New threats emerge. New vulnerabilities are discovered. New controls become available. Maintaining your knowledge of the current state of the business is required to continuously do your job effectively. The advantage of the growth associated with this role is that it can position you well for other positions, should you be interested in changing.

RECOGNIZING WHEN YOU'RE STUCK

If your role as risk assessor is one where you are a check box in some process—"ensure risk assessment done"—rather than an integral part of a business development, then you are in an organization that does not value this role. This type of role should be a dynamically engaged one within the company.

On the personal side, if you find that you are not taking the time to seek out as much information as possible as part of the process, or only working with the minimum amount of information necessary to do a perfunctory review, then you are not engaged in your role, and should investigate other positions. Consider either a different role or changing the organization for which you are working.

HOW TO GET OUT

This is a growth role as more organizations move to risk-based analysis for the implementation of controls. Frameworks such as the US NIST SP-800 document series are becoming standard in the US government for security design and implementation. As such, your opportunities in other organizations should be plentiful.

Table 2.4 Role at a Glance—Risk Assessment

Hours	**Travel**	**Stress Level**	**Creativity**	**Flexibility**	**Stability**
8/day	Varies	Moderate	Low	Low	High
General job duties	Performing risk assessments. Analyzing available controls. Developing risk reports. Travel will vary based on whether the risk assessor is a consultant or working for a specific organization. The need to work with a wide range of people within the organization, especially business managers who view your job as a necessary (or unnecessary) component of their job will make this job stressful. Explaining to technical people that while a technical control is indeed very exciting, but does not meet a business need can also be challenging at times.				
Learning	This position requires constant learning in order to stay abreast of the current risks.				
Advancement	Many opportunities. Potential for moves into management positions such as CISO, chief privacy officer (CPO) or similar is potential. Those with exceptional technical skills have the potential to move into Pen-Testing and Security Architect. Most other roles in this tier will be available if you develop your skills in that direction.				

CHAPTER

2.5 TIER 2—AUDITOR

INTRODUCTION

> "Sed quis custodiet ipsos custodes?"
>
> **— Juvenal**

In some ways, the role of an Auditor is simply to ensure that an organization is doing what it is supposed to—either according to an outside standard, or its own policies and procedures. Though simple in concept, in practice it requires a range of skills, including attention to detail, the ability to work with a wide range of abilities, and the ability to understand the underlying areas that the auditor is responsible for auditing.

There are two general categories in which Auditors can be grouped: internal (those employed by the organization) and external (those employed by an outside firm or from a regulating body).

Internal auditors are responsible for finding problems as soon as possible. During system development, they help internal groups understand what they will be looking for after the system goes live, helping to ensure that systems that are developed will meet the standards. In many cases, after running internal audits, they will provide suggestions as to how to address the issues that they find.

External auditors that are hired by a company are usually responsible for certifying that an organization is meeting a certain standard. For example, an organization may be required to be certified as an ISO-27001 organization to be eligible as a supplier in certain industries. The company will contract with an organization that is accredited to do ISO-27001 audits, which will supply the auditors to do the evaluation. In some cases, auditing organizations also have consulting divisions that can provide services to help ensure that an organization will meet the required standard.

External auditors from regulating bodies such as the US FDA or US FDIC are solely responsible for ensuring that an organization is meeting the regulatory standards that agency is responsible for enforcing. Depending on the regulatory body they represent, they may not provide remediation recommendations. They do, however, expect responses back describing how the issues will be addressed.

When embarking upon an audit, the first step is to identify what you are auditing against. Is the audit to ensure compliance with PCI-DSS, or is it to confirm that an organization is actually at Level 4 of the Carnegie-Mellon University Software Engineering Institute Capability Maturity Model (CMU SEI CMM)? Each standard has a different area of focus, and the auditor must be familiar with the underlying standard. Within a given standard are certain expectations for performance. For example, Level 3 of the CMM includes an expectation that processes are documented and established. However, the exact implementation for how those processes are documented and managed can vary and still be acceptable within the model. The Auditor must be capable of determining whether or not the required characteristics are present within the company that is the target of the audit. While the Auditor may have guidance documents to use to look at, it is key that they can actually comprehend both the standard and the system under audit.

One thing to be aware of as an Auditor is that you will rarely be directly involved in the development process. While you may be asked to help provide guidance, Auditors are not responsible for the actual development themselves. Because of this, those who have a desire to be builders should not enter this role, and instead investigate Engineering, Architect, or Subject Matter Expert roles instead.

HOW TO BREAK IN

There are many paths to auditing, but a common component is solid understanding of the areas being audited. If you are auditing a company's financial statement, you need to understand accounting principles. If you are auditing a company's security controls, you need understand the security concepts associated with information systems. This will likely involve a great deal of reading. Auditing the PCI-DSS, for example, will involve being intimately familiar with over 200 different audit points and understanding them from both the implementation and auditing guidance documents. Additionally, you have to be familiar with over 20 different support documents. Auditing US HIPAA is similar in scope, but auditing against ISO and US NIST standards and guidelines could well involve referencing ten times as much documentation.

Auditing is a two-way process, involving both the group doing the auditing and the group being audited. Because of this, you can gain experience by being part of the team that is responsible for responding to audits. As a Vulnerability Manager, Risk Assessor, or Quality Assurance Engineer, you may be asked to be part of an audit response team. Go through enough audits, and you will be familiar enough with the process to become an auditor yourself. In this case, the most common transition is to a role on an internal audit team, especially within the same organization in which you have been working to respond to audits.

There are IT audit certifications (such as CISA from ISACA, Systems and Network Auditor from GIAC, and CIA from The Institute of Internal Auditors). Each has its own set of requirements regarding tests and previous experience. Which one will work best will depend upon your circumstances. For example, if you can show your audit experience as an IT Manager or other role that includes auditing, then a CISA or CIA may be appropriate for you. If you have no experience, then the SANS GIAC: Systems and Network Auditor certification may be a way to differentiate yourself from other candidates with no certifications. Some positions, especially those with companies that provide audit services, will require certification.

HOW TO IMPROVE YOUR SKILLS

The more standards you can audit against, the more valuable you are. If there are opportunities to be part of teams that are doing different audit types, take advantage of them. Some audits (for example, CMU SEI CMMI or PCI-DSS) require specific certifications. Take advantage of any opportunity that may present itself to get those certifications.

As an internal auditor, become a part of development teams where auditors help ensure that new development meets the appropriate standards. Also work with other departments within the organization to identify not just which standards the organization is held to today, but which ones are anticipated for the future. Being able to provide guidance for projects so they will hit multiple audit areas will be highly valuable in terms of future career growth.

RECOGNIZING WHEN YOU'RE STUCK

If you are an internal auditor and are not involved in the process of helping develop systems, and the opportunity exists within your group, ask yourself why. If it is because you have no interest in expanding skills or learning about new technologies, then you are stuck. If you are not keeping up with the new standards as they come out, that could lead to the point where your skills are not relevant any more, and you need to think about your career path.

HOW TO GET OUT

There are many opportunities to move to other organizations, as the demand for auditors is growing. If you are no longer interested in Auditing, a move to management, Risk Assessment, Vulnerability Management, or Quality Assurance is reasonable, depending on the other skills that you have.

Table 2.5 Role at a Glance—Auditor

Hours	**Travel**	**Stress Level**	**Creativity**	**Flexibility**	**Stability**
8 + /day	Varies	Moderate	Low	Low	High
General job duties	Running audits against standards. Recommending remediations for issues found (as an internal auditor). If you work as a consultant, remote engagements can be significantly more than 40 hours/week. Similarly with travel, if you are a consultant, expect high level of travel. As an internal auditor, travel level will often be low. Your job is to find where people are not doing what they should be. As such, your presence is rarely something that people look forward to, and the interactions may be stressful.				
Learning	Many opportunities. You will need to learn about standards. As you audit companies, you will learn something new in each audit.				
Advancement	Potential advancement to management (CISO, CPO). Lateral moves to Risk Assessment, Vulnerability Management or Quality Assurance are available as well.				

CHAPTER

TIER 2—INCIDENT RESPONDER 2.6

INTRODUCTION

> "Months of boredom punctuated by moments of terror."
>
> **— New York Times**

The title of Incident Responder brings to mind images of chemical spills, earthquakes, tornado damage or other natural disaster. While there can be dramatic security incidents, day-to-day incident response is much less glamorous but still essential.

While an incident is going on, one must be able to collect and analyze the data that is being generated by the systems that the attacker may have compromised.

When collecting data that may be used in a legal case, maintaining an impeccable chain of custody is required to ensure legal admissibility. You must understand the process of collecting the data, making forensic copies (so that you do not alter evidence), and then analyzing the data collected to be able to understand the scope of the incident.

In situations where maintenance of evidence is not required, the focus will be on analysis of data. The business priorities will dictate whether or not the collection of evidence is expected.

Analyzing forensic evidence requires a range of skills, and can be an interesting and challenging security role.

If you are dealing with a skilled attacker, they may hide data through the use of steganography—the process of hiding information within another file, such as an image file. This makes exfiltration, getting the data out of the organization and past data loss prevention systems, easier. Similarly, attackers often delete data after they leave. Being able to analyze a hard drive to identify and extract deleted files is an important skill to have.

Beyond hard drive analysis, there are additional skills that are important for an Incident Responder. The process of network forensics, where you analyze network traffic and log data to determine the characteristics of an attack, is becoming a larger component of incident response. The use of virtual machines in the analysis of malware and systems is also becoming increasingly important. It used to be that when you had a computer you needed to analyze, you would have to find a similar computer to run the acquired forensic image. Now, a virtual machine makes the need to find a similar model no longer necessary.

HOW TO BREAK IN

The tasks of collecting and analyzing forensic data require a level of specialized knowledge that requires study and training. This knowledge can be gained through self-study, assisting others in your organization, or formal training. There are certification programs, as well as Associate degrees in Information Security, which include Digital Forensics as a component of the coursework.

The certification or degree, in combination with a demonstrated technical ability in some other technical role, will often be enough to get a lower-level Incident Responder position. For an advanced position, extensive network and system knowledge is necessary. You may be able to find an organization that would be willing to train you if you can demonstrate solid general technical skills, as well as an excellent attention to detail.

HOW TO IMPROVE YOUR SKILLS

How you improve will depend upon what skills you entered with. You never start with a full set of Incident Response skills. If you start strong with forensic collection skills, work to expand your analysis skills. As new technologies enter the market, you will need to learn how to analyze data that is stored using them (analysis of a solid state hard drive is different than analysis of a magnetic plate hard drive). Understanding the techniques that attackers use to hide their tracks will be necessary so that you do not miss them. As these techniques are continually changing, this provides you with an opportunity for continuous growth and improvement.

Similarly, the tools used for forensic analysis are continually changing, it is important to stay up-to-date with the new tools and how they can assist you in your job duties.

Beyond forensics, most incident response teams are composed of individuals who have additional responsibilities outside of incident response, such as Systems and Network Engineers. When working with them, learn as much as you can about their areas of expertise so you can expand your knowledge.

RECOGNIZING WHEN YOU'RE STUCK

As with most of these positions, the clearest sign that you are stuck is that you are not expanding your skill set. However, unlike some positions where maintaining the status quo may be adequate to continue being employed, as an Incident Responder, failure to update your skill set continuously could be a slow path to irrelevance and unemployment. Improving your skills is critical in this type of position, and if you are not in a position that allows you to do so, or have no interest in doing so, then you need to find an alternative position.

HOW TO GET OUT

The skills that you develop—being able to maintain a strong attention to detail and the ability to be very precise in your work—will, with the right organization, allow you the opportunity to move to similar roles, such as Security Assessment, Vulnerability Management, Risk Assessment, or Auditor.

Table 2.6 Role at a Glance—Incident Responder

Hours	**Travel**	**Stress Level**	**Creativity**	**Flexibility**	**Stability**
Varies	Varies	Varies	Low–Moderate	Low	Moderate
General job duties	Forensic analysis of storage media and network traffic. Travel level will be dependent upon whether or not you are a consultant, or working for a single organization. Forensic analysis requires the ability to think about the possibilities associated with a task, but standard procedures are often required in certain situations. As you do not get to dictate when you are attacked, your flexibility is highly limited. Stress will be high during the active attack, lower when doing background tasks.				
Learning	Learning is an essential component to maintain your jobs skills for this role.				
Advancement	Within the same tier to Security Assessment, Vulnerability Management, Risk Assessment or Auditor. Potential advancement to Security Consultant or Security Management.				

CHAPTER

2.6.1 TIER 2—INCIDENT RESPONDER—STORY

JOHN MEYERS

My name is John Meyers and I currently work for Hewlett Packard as an Incident Response/Forensic Specialist. I am part of a team that investigates computer security incidents for HP's trade clients. Information Security is my third career, and ten years ago I did not think I would be working Incident Response for one of the largest IT companies in the world.

Ten years ago, I was working as an Industrial Electronics Technician for a consumer goods manufacturer. Due to an acquisition, the plant I worked at was slated to be shut down. Since the company was moving manufacturing out of the US, I was eligible for retraining under a Pennsylvania Jobs program. I decided to take advantage of retraining and went back to school at the local community college. I enrolled in the Associate Degree Program in Information Systems, Network & PC.

While in school, I got a part-time job working in the College's IT department. It was a valuable experience for me and also gave me real-world experience in addition to the education. Because I worked in the IT department I got to know the faculty. One of the instructors was a SANS alumnus and this was the first time I heard about the SANS Institute. This was when I first became interested in Information Security.

I graduated in May of 02005 and in July 02005 I started work at a nonprofit in a help desk/System Administrator role. My interest in security led me to take the SANS course Introduction to Information Security. I received my GISF certification in March of 02006. I had been a subscriber to the SANS newsletter and one day I received an email from SANS that they were going to have a local mentor teach the SANS Hacker Exploits, Techniques, and Incident Handling class in Pittsburgh the summer of 02006. This was an opportunity I could not pass up, and I signed up for the class.

The class was great and I earned my GCIH certification in October 02006. After that, I brought what I learned about security from SANS back to my job. I started looking for ways to improve the security of the network I administered. Some of the changes I made were to standardize the builds of the PCs, install an IDS system, and implement Software Restrictions Policies.

All of this just whet my appetite for security. I taught myself Linux in my free time at home and played around with the tools I learned about in the SANS Hacker Exploits class like Metaspoit and Nessus. I got involved with SANS Mentor Program. Basically I tried to learn skills that would lead to some kind of Incident Handler job.

I finally got an opportunity to work for a managed security services company called Solutionary in January 02010. I would be working in their Security Operations Center. While at Solutionary I took every opportunity to learn new skills. Eventually I worked my way up to shift lead.

Then I saw an ad for my dream job, Incident Response at HP. I applied and was hired in January of 02011. Now I get to work with some of the most talented people I know on some of the most complex problems in Information Security. Being an Incident Responder for lots of large companies, HP's clients, provides a breadth of experience that is invaluable.

Well, that is my story. I went from losing my job as an electronics technician to working Incident Response for HP in eight and a half years. It was a lot of hard work, self-education, and luck. I'm sure I took the road less traveled to get to where I am, but I'm here now and having a good time.

CHAPTER

2.7 TIER 2—WILDCARD

INTRODUCTION

> "Specialization is for insects."
>
> **— Robert A. Heinlein**

As a Wildcard, Jack of All Trades (JoAT), or Go-To Technical Person you may be the only technical person at a small organization and if not the only such person, you may still be responsible for all technical architecture. The Wildcard is responsible for all of the technical architecture at a small company. From servers to phone system, website to end-user computers, it is your responsibility.

This type of position has three main job responsibilities: end-user support, current system maintenance, and infrastructure improvements. The ratio of the three will vary greatly from organization to organization and from day to day. If malware gets loose within the network, the Wildcard will be spending time fixing machines. When patches come out, system maintenance will be the focus. When the chief financial officer (CFO) wants new accounting software, that will take up a large fraction of the available time.

More than any other role, each day brings new challenges in many areas. You will need to be comfortable working alone, without any other experts to turn to within the organization. To meet the crises face on, you must be able to research solutions on the Internet and fully leverage your own problem-solving skills.

Since this position frequently exists at small companies and nonprofits where money may be limited, your biggest challenge will frequently be the budget. More often than not, you will not have the option of solving problems by spending money. Your creativity and knowledge will be required to keep the organization running. However, this has advantages as well. The approval process is usually simple, involving only a single person (owner or business manager). Additionally, many of the projects selected will be ones that you designed and developed. Rather than being handed a specification and told to build it, you get to design the specification and then implement it, but using little or no money.

To succeed in this role, you must be a self-starter and able to work independently.

HOW TO BREAK IN

These roles often grow out of another position within the organization. When the company was too small to support a dedicated full-time technical person, you were the person people turned to when things went wrong. You were able to support the company a bit at a time, and as the role grew, you grew with it.

For those situations where you would be coming into an existing role, the key is going to be to demonstrate as wide a range of relevant skills as possible, along with the ability to learn new skills. If you have a role that only has a few skills, work on getting certifications and other credentials in other areas to demonstrate a breadth of skills.

HOW TO IMPROVE YOUR SKILLS

This role is a classic "learn by doing" role. You will be learning by solving the new problems that arise, building new infrastructure, and helping others as their needs change. Because you are the only one with technical expertise at the organization, you will be able to drive many projects to improve the infrastructure, but probably not at the same time. Additionally, as you will be working with everyone in the company, you will develop a level of understanding about many non-technical components as to how that business is run—knowledge and skills that have value in future roles.

RECOGNIZING WHEN YOU'RE STUCK

If you spend no time working to improve the infrastructure, you are in danger of being stuck in this role, with your skill set stagnating. Even if your company cannot afford to pay for any more infrastructure, you should be able to investigate free solutions for ways to improve the organization.

HOW TO GET OUT

This role can be used as a stepping stone to a wide range of other technical roles at larger organizations. Larger organizations tend to look for specialists rather than generalists such as the Wildcard, so you will need to demonstrate a depth of understanding in a specific technology or, if you can make it past the HR screen, the ability to learn rapidly to meet the specific needs of the organization.

Table 2.7 Role at a Glance—Wildcard

Hours	**Travel**	**Stress Level**	**Creativity**	**Flexibility**	**Stability**
8+/day	Low	Varies	High	Moderate	Moderate
General job duties	Everything technical related. Help desk, server management, network management, website management, and anything else that seems like it should be your job. There will be periodic after hours work so as to not disrupt the work of others during system upgrades. Since you are the go-to expert, you will get to devise the solutions to the problems that arise. Creativity will be a key skill because your budget will be smaller than in a large corporation. The stress level will be very dependent upon the company, work load, and your supervisor. This can be a dream job with enough to keep you busy, but not enough to be overwhelming, a job that is boring in general, or a one where one person really should be three people. In some environments you will have rigid hours or regulations that you must work within, but in many others you will have a high degree of flexibility, as long as you get the job done. Stability is generally high as the company will be completely dependent upon you for their technical needs. However, since this is usually at a small company, the stability of the company will be the most important factor in the position's stability.				
Learning	High. If something new comes up, you will have to learn how to deal with it, rather than turn to the other expert in house.				
Advancement	Few options within the same company. The opportunities to advance will not be within the same company, but when you want to move on to be more specialized at a larger company.				

CHAPTER

2.7.1 TIER 2—WILDCARD—STORY

TRAVIS ABRAMS

Like most people in InfoSec, I started out in IT. I started out managing desktops for a large law firm; we had over 3,000 desktops. I took on the responsibility of managing the anti-virus for the desktops, because no one else wanted to do it, among other tasks. This was 01997 and we had no central management console, so I would email the weekly DAT file to users and ask them to click on it! It was absurd then and even more so today. We were asking users to run the executable in their email from us but not click on files from anyone else!

One day, I was walking out of my work area into the main hallway at the same time the CIO walked out of his office. I am not sure he even knew my name. I was the first one he saw and he said, "Hey, I just got a call and someone said there is something wrong with our training website. Go fix it." It turned out the site had been defaced and I had to argue with the site owner that we had to take it offline, patch it, and them put it back up. It took time to convince them this was the only way to correctly fix it but I was successful.

After this I was known as the security guy. The organization recognized the need for security and invested in training for me. I was able to convince them of the need for a centrally managed AV solution, regular vulnerability assessments, etc. A few years later I was officially given the title of Security Manager, which was a first for this organization. I eventually left but I have been in security ever since.

My recommendations for a security career are:

1. If you do not have experience in IT and want to go into the technical side of security; first get a job in IT. Having managed servers, desktops, etc. has only enhanced my security career. The only exception is if your goal is to be an auditor. This does not necessarily require a technical IT background.
2. Be indispensable and willing to do what others may not want to.
3. Take advantage of opportunities. The reality is that many times companies will not invest until something adverse happens. For example, after our server was defaced I was able to convince the company to have a pen test performed by a third party.

CHAPTER

TIER 2—ADVANCED HELP DESK—HELP DESK SUPERVISOR

2.8

INTRODUCTION

> "Hello, IT. Have you tried turning it off and on again?"
>
> **— Roy Trenneman**

Advanced Help Desk, or Help Desk Supervisor in Tier 2 support, are those who have developed enough skills to address the problems that cannot be solved through a standard verbal script. After the ticket has been submitted, and the first level of support has attempted to gather as much material as possible about an issue they cannot fix, it will often be escalated to the second level of support engineers. These engineers are capable of working outside of a script to address problems that come up.

Even if they are not able to solve the problem themselves, Advanced Help Desk are able to work with end users to isolate where the problem is occurring. They can take a problem like "the Internet is not working" and turn it into "DNS can't properly resolve external domain names." They may not be able to fix the DNS infrastructure, but they provide the Systems Engineers with a clearly defined problem to address.

This position requires an intermediate level of technical understanding and skills. More importantly, it requires the ability to think critically under pressure. The pressure will vary, depending on the requester. If the chief executive officer (CEO) is contacting you, the pressure is higher. Think of this as training for your security career; clear thinking and comfort under pressure is critical when dealing with security situations.

In many organizations, the Help Desk Supervisor will have the technical responsibilities associated with this role, as well as management responsibilities for those individuals who are working on a given shift. The Help Desk Supervisor would then report to a Help Desk Manager that is responsible for all levels of shifts for that help desk.

HOW TO BREAK IN

People in this role often advanced from an entry-level help desk position. To get into this type of position, you must demonstrate three keys skills: solid general technical skills, excellent customer service skills, and solid problem-solving skills. Working as a help desk technician can give you many of these skills, but you may find that you need to develop your technical skills beyond what is required to at Tier 1. If you have spent time working as the go-to person for problems within your office, even though your main job was in other areas, you may have been working as an Advanced Help Desk Technician, even though you did not realize it. You may also be on your way to being a Wild Card.

For those people who have spent time as a Patch Manager or Security Coordinator, there may be a desire to move into a more people-focused role, or gain experience in a wider range of technologies.

HOW TO IMPROVE YOUR SKILLS

When working as an Advanced Help Desk Technician, you will find that you are exposed to a wide range of technologies. Lots of things break—different things at different times. As you work to isolate the problem, you will be learning more about networking, servers, workstations, and the Internet. The key is to learn as much as you can about the problems that you find. Talk to the engineers who address the problem and confirm that your diagnosis of the problem was correct. Learn about other ways you could have gone about diagnosing the problem. Were there other tools you could have used? Were there other questions that you should have asked? If you can manage to watch an engineer address the problem, that opportunity would be great, but it rarely presents itself.

The other key component to be alert to while in this role is how people interact with the help desk, and what they are willing to entrust you with. Most companies have strict policies against sharing passwords, yet people often volunteer them (always say no). Many of today's pen tests include a social engineering component. Take the time to learn how people interact with you, and think about how you could exploit it (only in the name of good!) as part of a security test. With solid people skills, you could become part of a pen test team that includes a social engineering component.

RECOGNIZING WHEN YOU'RE STUCK

If it took you more than one or two years to get to this Tier 2 role, you're probably already stuck. If you quickly got to this role in months from Help Desk, but haven't gotten further since, you may also be stuck. As with many other roles, if you find yourself doing just the minimum in order to get by, then you are stuck. If the corporate culture doesn't allow you the time to expand your technical skill set, or you just don't feel the urge, it is time to start looking for other opportunities.

HOW TO GET OUT

This is the type of role where moving between companies is generally easy, due to the high turnover in the field. However, since you want to move into security fields, ideally you want to use your general technical skills as an entry point to get more high-value technical skills. Depending on your skills, you may be able to become the sole technical person for a small company—someplace where they need a person who will be comfortable working with end users, and at the same time manage all of the back end systems also known as the Wildcard described in Chapter 2.7.

CRITICAL WARNINGS

Positions that don't provide any training or growth opportunities will stagnate your career. Look for positions and opportunities where you are able improve your skills through exposure to other roles.

If an organization has a habit of sharing credentials, be wary. Usernames and passwords should be kept confidential and never shared.

Table 2.8 Role at a Glance—Advanced Help Desk—Help Desk Supervisor

Hours	Travel	Stress Level	Creativity	Flexibility	Stability
8/day	Low	Medium	Low	Low	Moderate
General job duties	Isolating problems Some break/fix responsibilities in some organizations Developing decision trees As those who call a help desk are generally having problems. There is the inherent stress associated with frustrated individuals.				
Learning	Most learning will be done from working with the various technical people that you work with as you develop decision trees. Some business process education will be part of this role as well.				
Advancement	Supervisory roles are one possible direction to move from this role. Moving to more technically focused roles such as Server Administration/Engineering roles or Network Administration/Engineering roles. While Server Administrator and Network Administrator roles may be more entry-level, others will be there that will help you advance your career.				

CHAPTER

2.9 TIER 2—SECURITY FACILITATOR

INTRODUCTION

A Security Facilitator, also known as a Delivery Lead, Project Coordinator, or Project Manager, is responsible for ensuring that projects reach completion. Project management is a specialty in and of itself, with varying ranges of formality to the process. The larger the project, the greater the requirement to have formalized and agreed-upon processes for managing each stage. As the Security Facilitator, it is your responsibility to ensure that all of the appropriate steps are followed. Identify and, where possible, work with management to mitigate those risks. One of the most common risks that a Security Facilitator will face is dealing with resource allocation issues. Working to identify as soon as possible when there are either not enough people, or people with the right skills, and working with management to address these issues, is a significant part of a Security Facilitator's role.

One component of this that can be very frustrating is that a Security Facilitator will often be tasked with managing a project and ensuring it reaches completion, but not be given the authority to ensure that the appropriate people or funds are assigned to the project. This will frustrate some, but for those with negotiation and communication skills, this can be an excellent fit.

While the general tasks associated with project management are common across all projects, the exact mechanism for doing so will vary. For example, all projects must identify the requirements that, upon completion, indicate that the project is done. The methodology for gathering those requirements will vary based on the framework in use for that specific project.

One of the bigger challenges associated with security projects is that what may appear to be a simple technology implementation may often impact many or all areas of the organization. This potential impact will require working with a wide range of people with a wide range of technical abilities. Being able to communicate with this wide range of individuals will be one of the key success factors for someone in this role.

What may be apparent at this point is that this is not primarily a technical role. You will not be responsible for doing any of the actual implementation yourself, though you may be asked to help in some cases. If you are someone who requires daily working with technology at a deep level on a daily basis, this role is not for you. This role focuses more on organization and communication. That said, there is still a need to be able to understand, at least at a high level, the underlying technology. Doing so will help as you gather people to help you, and helps to ensure that you ask for the right people.

HOW TO BREAK IN

This is the type of role where each previous small success will lead to the next bigger role. Open the door to larger projects by managing a smaller project for which you and a co-worker are responsible, demonstrate the ability to report accurate status reports to management, and keep the project both on time and under budget. Your role can be either as the technical expert or the business expert, but demonstrating the ability to understand an area that is not your specialty will be a key for future advancement.

Consider an example where you, as the technical expert, have been asked to work with someone in procurement who needs to set up a new application, and make the appropriate infrastructure changes to get a new application work with one of your company's suppliers. You will need to understand what the application needs to run, if it requires special network access, if other people may be using the application in the future, and the ramifications of the changes to the infrastructure this will require. You will work with the appropriate people, using your change process to get the solution.

This is a simplistic example, but the path to larger roles start with learning smaller roles, and demonstrating your ability to succeed.

For larger projects, and in some organizations, various project management certifications may be necessary to reach certain levels. Some certifications only require coursework and passing a test (such as a Certified Scrum Master), while others require a certain number of hours of work experience in the role.

HOW TO IMPROVE YOUR SKILLS

The most growth is going to be had by being involved in projects that are in as wide a range of fields as possible. If, for example, you have managed a mobile device management implementation, a new firewall rollout, and a new two-factor authentication subsystem, your breadth of experience will set you apart from others in the field.

RECOGNIZING WHEN YOU'RE STUCK

If the projects you are managing are getting smaller, have a lower visibility within the company, or have trouble getting the resources necessary to complete them, you are stuck. If your projects focus on maintaining legacy applications rather than implementing new technologies or mission critical tasks, then you are also starting to stagnate in this role. Those tasks are fine as you are starting in the role, but you want to move into projects involving newer technology as you advance.

HOW TO GET OUT

If you want to change organizations, there will be opportunities, as this role (as of this writing) is in moderately high demand. If you wish to change roles, a lateral move to Business Analyst, Patch Management, or Vulnerability Management will often be a logical step, as many of the underlying skills are similar. For more dramatic changes, a move to Risk Assessment or Security Assessment may be feasible, depending on your skill set.

Table 2.9 Role at a Glance—Security Facilitator

Hours	Travel	Stress Level	Creativity	Flexibility	Stability
8–10 hours/day	Low	Low	Moderate	High	High
General job duties	The job duties for a security facilitator will vary widely with what you are helping to facilitate. Generally, there will be a lot of reading and a lot of pulling together different documents, emails, and specification lists to make sure that everything is getting done as it should.				
Learning	This is a very learning-focused job as you will largely be doing work to help people who have less time than you. This means that, while they can guide you somewhat, you will have to do a lot of learning on your own.				
Advancement	Security Facilitators tend to move on to being assessors or project managers.				

CHAPTER

TIER 2—SECURITY FACILITATOR—STORY 2.9.1

JIMMY VO

My story begins at age 10, where I received my first computer. In that day, it was a top-of-the-line Dell, with a blazing 333 MHz with 64MiB of RAM running Windows 98. I ended up spending a lot of time in front of the computer, mostly waiting for an AOL dial-in number that wasn't busy. Since then, I was fascinated by computers, taking them apart, making webpages, playing games and various geek activities. I didn't do anything remarkable like many other information security professionals did when they were younger like write crazy applications or break into something they weren't supposed to. Being in front of a computer was a hobby. Fast-forward to my high school years where I excelled (compared to the other students) in the programming classes. Believe it or not, I took two programming classes in high school. BASIC and visual basic, nothing crazy.

It was a natural that I wanted to work with technology. I applied to colleges and was accepted to Montclair State University to study Information Technology. At MSU, I had my first tech job, working the help desk where I help people with their connectivity and email problems. I transferred colleges and finally graduated from The Richard Stockton College of NJ after 5.5 years with my Computer Science degree. I recall finishing up my last semester, with little experience, and thinking the job hunt would be easy. Boy was I wrong.

I learned many lessons during my college days that I wish I would have known sooner. I wish I would have been more involved, networked more, and spent more time working jobs that added relevant experience. I luckily landed my first full time job as the Systems Analyst at a small company. It was at this time I really focused on self-improvement. I participated in Toastmasters and spent a lot of time building my professional network, focusing on my online presence (Twitter) and getting all of the knowledge I could. I also decided to pursue my master's in Computer Information Systems with a Security focus at Boston University. I occasionally applied for security jobs during my time at this organization; however, I was unexperienced and not hirable (even with a master's degree, take note).

I was a systems analyst for a couple of years until I approached my boss and came up with a plan to transition into a more security-based role. With a lot of convincing and a solid plan I was able to focus a bit on security planning. Now I was getting my hands on various domains of security, such as policy writing, bit of risk management, security ops, vendor management, etc. Being the only security guy and extremely inexperienced, I identified that I wasn't growing as a security professional. I needed to be around a smart team of experienced guys to mentor and help me grow.

When I decided to hit the job market, I did it in a different way than searching job boards. I reached out to my professional network and Twitter followers. I was able to land multiple interviews with organizations without having to deal with the application process because of Twitter. After months of interviewing, I was picked up by a consulting firm. All. Because. Of. Twitter. Let that sink in.

The Cliff Notes version of my long-winded story: be passionate, invest in yourself, meet people, hustle—and good things will happen.

CHAPTER

TIER 2—POLICY ADMINISTRATOR

2.a

INTRODUCTION

Policies are the core of any organization’s security infrastructure. Clearly defined policies provide guidance to those that are responsible for implementing security controls across the company.

While the above statement may appear simple, the management of those policies takes significant effort, especially in larger organizations. The process of gathering the policies, ensuring that they are all recorded properly per the organization’s standards, reviewed periodically, and implemented properly can become a full-time responsibility. Some organizations, rather than have dedicated individuals who only focus on this role, will blend the responsibilities in with other roles, but the tasks are still necessary.

The Policy Administrator is not responsible for setting policy. Management and subject matter experts with expertise in the appropriate regulations are commonly responsible for setting the policy. Only people in certain roles being allowed access to a certain module within the enterprise resource planning (ERP) system, no visiting of certain types of websites, or only people within certain roles being allowed to view logs on the server are all parts of security policies.

All of these policies would be documented, and provided to the appropriate technical administrator (such as Network Administrator or System Administrator) for implementation. Depending on the organizational structure, the person responsible for collecting the policies might be responsible for the implementation as well but in other organizations this would be a conflict of interest.

Documentation of the implementation of the policy is necessary, and usually done through some type of change management system.

Finally, in well-run organizations, policies will be reviewed periodically for continued appropriateness. As the business changes, it is important to ensure that the group that has access to a system is still appropriate given the current organizational structure.

HOW TO BREAK IN

The method for breaking into Policy Administration is going to depend on the target organization’s structure. If the organization is structured so that individuals have more vertical responsibilities, technical skills are going to be required. In this example, an individual would be responsible for gathering and implementing all policies around Active Directory. This type of role requires a combination of business and technical skill sets that would be necessary to demonstrate that you either possess or are capable of learning quickly. For this type of role, experience as a Systems Administrator, Network Administrator, or similar technical roles may be required.

If an organization is more skill-focused, working with business process owners and being able to work with technical people will be key. This role is going to be more focused on documenting and collecting, across a wide range of technologies, and leaving the implementation to the technical experts. Being able to work with the technical experts, to be able to ensure that the desired policies are possible for a given infrastructure, is important. For that kind of organization, Auditor, Risk Assessment, Security Assessment, or similar roles are potential previous experience.

HOW TO IMPROVE YOUR SKILLS

This role provides ample opportunity to grow in a variety of directions. By working with business process owners, you will develop an understanding of business needs and drivers. Understanding this is a key success factor in moving on to positions with more responsibility.

Whether you are responsible for implementing solutions, or working with those who implement solutions, your technical skills should improve as you go through the processes. Working to ensure that you are using the best method for the platform that you are on, or working to streamline the implementation, are important skills for future advancement.

RECOGNIZING WHEN YOU'RE STUCK

This role can turn into exclusively a bureaucratic paper-pusher role, where you gather and document. While this can be a useful stage in your career, especially if you are new to learning about business processes, if you find that you are no longer learning about the business then you have probably maximized your growth potential within this role and it is time to start investigating other roles.

HOW TO GET OUT

The policy administrator has business, documentation, and audit skills. Auditor would be a potential move, especially for an individual who has demonstrated the ability to audit an existing policy set to ensure it is appropriate for the current business model. Risk Assessment may be possible if you have developed your technical skills to understand the underlying technologies. Strong business and documentation skills with the front line experience in security may make others roles available such as a Tier 2 Trainer-Educator.

Table 2.a Role at a Glance—Policy Administrator

Hours	**Travel**	**Stress Level**	**Creativity**	**Flexibility**	**Stability**
8/day	Low	Low	Low	Low	Moderate
General job duties	Working with business process owners to identify the appropriate permissions for various roles. Ensuring that policies are implemented appropriately. This position is usually based out of the location where it is most needed, and all work with other offices can be done remotely. This position is rarely dealing with crisis level issues. If an individual is determined to have permissions that they should not have, then well documented histories should protect the policy administrator. If, however, policy decisions are not well documented, then stress levels can be higher. Creativity and flexibility are limited due the need to be guided by the business.				
Learning	Moderate early in the role, decreasing over time. When first in this role, you will have the opportunity to learn about many business processes. As time goes on, the learning will decrease.				
Advancement	Taking the time to learn about a range of business processes provides the opportunity to move into other roles where technology and business process intersect.				

CHAPTER

2.b TIER 2—TRAINER-EDUCATOR

INTRODUCTION

At Tier 1, the Trainer-Educator generally is mostly limited to using materials developed by others—some centralized organization produces a standard set of documentation that you work from. At the more advanced Tier 2 Trainer-Educator level, you are responsible for creating your training materials for you and others. For example, education providers that focus on training professionals in one-week boot camps would use these materials for their training sessions. Some individuals work as consultants for training companies whose clients do not want to bring full-time personnel in house, but still have regular needs for trainers.

In some cases, an advanced trainer who has been hired to provide some level of customized training will be brought in for a training company. The contractor would take the standard core materials that the training company uses, and adjust them to meet the requirements of the client. These trainers must be able to work independently with very little guidance beyond, "The client wants you to focus on….."

Advanced Trainer-Educators need to have combined subject matter expertise and teaching ability. They must be able to go outside of the training materials when asked, and be able to adjust to a wide range of students in the class simultaneously.

HOW TO BREAK IN

Some form of training experience will be required. If you have worked as a trainer where you worked exclusively from others' materials, that can help provide the foundation, but a demonstrated ability to develop training materials is necessary. Such examples might include adjusting course materials for a course that you were teaching at the local community college, creating course work at a company where you work, or coming up with materials to teach a technical subject at a local community center.

Similarly, in order to develop these materials, a comprehensive understanding of the material that you will be teaching is required. Depending on the subject matter, this knowledge could be gained from hands-on experience or through coursework. When developing new materials, you will often find there are small gaps in your knowledge, where you did not fully understand why you were doing what you were doing. As you prepare to teach the material, the why becomes critical, as you will be called upon to display that knowledge frequently while teaching.

In order to teach certain subjects, it will often be expected that you have certifications in the field. This is especially true when you are teaching courses targeting preparation for a specific credential. It will generally be required that you have that credential in order for you to be hired to teach that material.

HOW TO IMPROVE YOUR SKILLS

When teaching in a technical field, it is critical that you maintain a current knowledge set. Whether it is in security, networking, or server management, the new material is something that you must keep up with in order to maintain your marketability within a given market.

Similarly, it is valuable to increase the breadth of courses that you can teach. If you can teach security and networking, you will be more employable than a trainer who only can teach one of those subjects. When working for an organization as a trainer, you will often have access to the training material for other courses for free or significantly reduced costs. Take advantage of that to improve your skills in other areas.

RECOGNIZING WHEN YOU'RE STUCK

If you keep teaching the same material without updating, you need to evaluate whether teaching continues to be an appropriate role for you. If you find that the classes that you are teaching have gone from being cutting-edge classes to being legacy technologies, you are in danger of becoming marginalized and not employable.

HOW TO GET OUT

If you have maintained a solid technical understanding of the material, you can often move into the roles that you are training others to become. Be forewarned, though, that if you have spent an extensive time teaching, it may be harder to break back into technical roles, as some may question whether you have lost the ability to Do. That will be less of a hurdle if you can demonstrate that you have developed materials, such as labs, that revolve around Doing.

Table 2.b Role at a Glance—Trainer-Educator

Hours	**Travel**	**Stress Level**	**Creativity**	**Flexibility**	**Stability**
8/day	High	Low–Moderate	Moderate	Varies	Low
General job duties	Presenting coursework, developing coursework If you teach boot camps, long days where 12 hours would exist. Travel may be high, due to going to different locations to do the training. Travel will be less frequent if you are in a major metropolitan area. Stress will vary based on the classes. Also, if there is a spike in stress, it is more likely to be short lived, just the duration of the class. Stability is low since in economic downturns, training is often the first to go, so demand for trainers is much more tied to economic cycles. Depending on the organization and training environment, you will have varying degrees of latitude in what you create for training material. Your flexibility will vary greatly based on whether you are an independent or working for an organization.				
Learning	It needs to be plentiful, either through the organization, or working independently to ensure that you are familiar with the current trends and technologies.				
Advancement	Not much. There may be opportunities to manage trainers, but there will not be many of those opportunities. Sometimes training classes can lead to consulting in other areas, but that varies dramatically based on the material and the trainer.				

CHAPTER

2.c TIER 2—QUALITY ASSURANCE

INTRODUCTION

> "There are two ways to write error-free programs; only the third one works."
>
> **— Alan Perlis**

It is important to clarify the difference between Quality Assurance and Quality Control. As a Tier 1 Software Tester, you are doing quality control. You are making sure that the resulting product (software) has the functionality expected of it. Quality Assurance is making sure you are using proper processes to create that product. In some cases, positions that are advertised as Quality Assurance positions are actually Quality Control positions.

That said, as software development matures, the need for true Quality Assurance engineers is becoming more common. Many organizations need to ensure that their development is done to ISO-9000 standards or at certain levels of the SEI Capability Maturity Model (CMM). Fully describing those standards is outside the scope of this book, but they are ways of approaching software development so that the product produced meets the various system and functionality requirements. This involves ensuring requirements are documented, being able to demonstrate that all of the requirements are met, and having clearly defined processes for addressing issues that are discovered. Some standards also require the lessons learned while doing the development be used to improve the processes for subsequent development.

You will find that different organizations have very different levels of process associated with their software development. Organizations with poor processes will give software testers an approximate idea of what the software is supposed to do, based on very loose engineering requirements, and ask them to test the software. In this case, no quality assurance is being done. When an organization is managing an end-to-end requirements management system, where requirements are traced to test plans that are traced to test results, a true Quality Assurance system is in place.

There are other process models (such as Six Sigma or US Department of Defense standards) that can be used in association with software development as well. Which ones are in use in your organization will depend on the business culture and the regulating body to which your business reports (US EPA, US FDA, US Department of Defense, or others).

Understanding the software development processes is a key component to securing your organization. All too often, security is an afterthought in development, leaving systems vulnerable to attack. If you go into a role as a Chief Information Security Officer or similar role, you will have responsibility for ensuring that software developed internally is secure. Understanding the processes that support secure development is extremely valuable.

HOW TO BREAK IN

This position is one that can be advanced to from a Quality Tester position. While working as a tester, understand all of the various components associated with the system that your company is using. There are many common facets to all of the systems (requirements management, documentation of the various stages in the process), and understanding those will be critical for advancement within the field.

After you have learned the general concepts associated with a standardized software development process, lower-level Quality Assurance positions will be available to you. For upper-level positions, understanding of one or more of the standards that organizations must meet is required.

If you are a Systems Administrator, become the administrator for the systems that are responsible for supporting the processes at your company. This will give you the opportunity to learn about the standards from an implementation perspective, improving your marketability.

No matter what your role, if Quality Assurance is your goal, try to get involved in any team that is responsible for improving process. This will give you the opportunity to understand both the business drivers and the standards.

HOW TO IMPROVE YOUR SKILLS

The best way to improve your skills is through your day-to-day tasks. Be involved in the process of developing processes and systems.

RECOGNIZING WHEN YOU’RE STUCK

If you find yourself simply approving changes in processes that others are making, and not being a driver for change yourself, you are stuck. The key to growth is to be part of driving the changes that an organization implements. Have you created or improved processes? Are you making difference in the quality in the organization? Is quality at least not going down? If not, despite your best efforts, the problem may not be you, but the organization.

HOW TO GET OUT

If you are stuck because the organization does not give you the opportunity to be part of the changes, moving to a different company but in the same role is a logical step. If you find that you are not interested in the tasks any more, moving to Auditor is a logical step. Other roles may be feasible as well, depending upon the skills you have developed along the way.

Table 2.c Role at a Glance—Quality Assurance

Hours	Travel	Stress Level	Creativity	Flexibility	Stability
8/day	Low	Moderate	Low	Low	High
General job duties	Developing processes in order to meet standards. Ensuring that current processes meet with external standards. Changing processes as business needs change. Optimizing processes based upon lessons learned from other projects. There will be stresses in ensuring that the business processes are holding to the standards. One of the common stresses in this role is understanding how to enforce a standard when the business unit responsible for the process is unwilling or unable to work to the standard, but it is generally a stable position. There is little creativity or flexibility in this role due to the external forces that drive this role.				
Learning	There is a significant opportunity to learn about business process within this role. Additional opportunities to learn from peers and auditors that you work with are available.				
Advancement	Lateral to Auditor. Management opportunities may present themselves as well.				

CHAPTER

TIER 2—SUBJECT MATTER EXPERT 2.d

INTRODUCTION

An information technology subject matter expert (SME), pronounced either as "ess em ee" or "smee," goes beyond the subject matter specialist. The SME may be the best in the company, and possibly the best in the field, or even in the top 20 in a software engineering organization. But an SME might just be someone who has some specialized knowledge that no one else has on the team. The particular meaning depends upon the organization and organizational culture.

The SME is the person that everyone in the organization turns to when there is an issue in that subject. They are the final person to work with. If you are the SME, and you do not have the answer, you will be expected to be able to do the research necessary to come up with the answer. For very specialized subjects, it may be part of one's overall job responsibilities.

HOW TO BREAK IN

Find the existing SMEs in any subject in the organization and learn from them, and refer other people to them. If you eventually learn enough in a subject, you may become an SME if either of you leaves the organization. There may be no master to learn from directly, but there might still be a mentor or master within or without the organization to learn from. For example, if a consultant comes in to work with a system, learn as much as you can from this person. If an organization has no SME position, that's an opportunity for *you* to become that SME, officially or not. If someone asks a question in a particular area, volunteer to find the answer. Do the deep research, find an expert if you can, go beyond just that question, and provide the answer and be able to answer the next question before it's even asked.

Some organizations have a formal or semiformal role of SME, or may be willing to create one; others may merely have a culture in which people know who the best are in any particular area.

An SME role might come about by accident; by simply spending the time as a normal part of a full-time job, someone may discover themselves to be considered the expert. However, by consciously choosing a particular area of focus, the prospective SME can accomplish the goal faster, in a distinctively useful area. Information technology is sufficiently broad that there are few SMEs in any organization. By choosing an area that no one else in the organization knows or knows well, a prospective SME can quickly become the expert. One example of this would be the management of a newly released product. By definition, there is no one who understands that tool. By taking ownership and responsibility for that product, you can quickly become the SME in the organization. To be recognized as an SME outside your organization will probably take more time, but as it's now part of your job, you can devote work time to benefit your organization and yourself.

It takes thousands of hours to become an expert. However, to be considered an expert in the local area can merely be knowing more than anyone else available.

HOW TO IMPROVE YOUR SKILLS

Growth can be upward (enhancing skills in your specific area) or outward (expanding your area of expertise). As the SME, growing upward will require independent, self-guided research. For gaining breadth, take advantage of the projects that you are working on with others. As an SME, you will frequently be part of larger teams; take advantage of that to expand your skill set.

RECOGNIZING WHEN YOU'RE STUCK

If you do not enjoy the tasks you are responsible for, and you likely won't learn to go any deeper, especially if the organization doesn't want you to, then you are stuck. It's time to move on.

HOW TO GET OUT

Being stuck as an SME may not be a bad thing, and can be a suitable end-point for a career. However, transitioning from being an SME can be difficult, as they tend to be highly valued in their area and may not have sufficient qualifications to get into another area and be as well-regarded, or as well-paid.

One way to get out is to develop additional skills needed to document and train others to do it. Another way is to work yourself out of a job by establishing automated systems and processes.

Some organizations may require an SME who has been within the organization a long time to move laterally, so others can gain knowledge in that area. Additionally, there may be opportunity for promotions where aspects of their experience can be applied in new ways.

Table 2.d Role at a Glance—Subject Matter Expert

Hours	Travel	Stress Level	Creativity	Flexibility	Stability
8+ hours/day	Varies	Moderate–High	Moderate	Moderate	High
General job duties	Project and program management (planning, scheduling, budgeting) Research. Mentoring. Writing whitepapers, documentation. Reviewing others' work. Stress can be lower because the job is well understood and well managed, but stress can be high due to high expectations. Can be very stable because no one else can do the job. The travel varies. In companies with many locations, there may be a high amount of travel due to being the only person who can do the job. There is only moderate creativity and flexibility. As the expert in your area, you are allowed significant freedoms with that area. However, being able to branch out can be challenging.				
Learning	The SME can be, by definition, limited in learning as there may be little more to learn from others. Although, few areas so small and so well known that there is truly nothing more to learn.				
Advancement	An SME may be the highest level available in an organization for that particular subject area. Some organizations have special non-managerial career paths for technical staff. This can include addition to titles such as: Senior, Principal, Chief, or "Vice President of". These can sometimes be paths to broader knowledge, not unlike a post-graduate education, but consider it on the job training.				

CHAPTER

TIER 2—SUBJECT MATTER EXPERT—STORY

2.d.1

MICHAEL HUBER

I mentor in the SANS program in the more traditional sense informally and formally in the local community in NC. I am informally assisting three new professionals in entering the InfoSec field from their current positions.

So my recommendations are based on my path that I took as well as learning from the mistakes I made earlier in my career to help others in not hitting the same pitfalls.

My path into information security started while I was still in college in 01990 when I started working on removing viruses from Windows machines as well as bugs from the UNIX world. In my first ten years I dedicated myself to understanding both Windows and *nix worlds in all their complexity and to ensure that I became a subject matter expert in those areas. My focus was to understand down to the bits and bytes how and why these operating systems functioned to better protect them. I then moved on and in my next ten years I focused on understanding the non-technical portions of the information security profession. I still keep my hand in the technical arena with a home lab but my day job is one that is focused on risk mitigation and I have certifications in both the Information Security and Risk fields. I have added teaching and mentoring in the last ten years and giving back to the community.

My recommendation to someone starting out is to become the SME for your technical love (Windows, *nix, Macintosh, mobile, cloud) and understand how it functions down to the machine level, and then branch out from there. Keep focused on business processes and risk mitigation throughout your career. Good resources are SANS (301, 401, 501) when you start out, get CISSP and CRISC once you have three to five years under your belt of hands-on in the field while continuing to grow your skills through the many other classes and certifications that SANS, ISACA, (ISC)^2, and the European Council have available, depending on what part of the profession you love. Love what you do and you will go far.

CHAPTER

2.e TIER 2—LATERAL: PHYSICAL SECURITY

INTRODUCTION—HOW THIS APPLIES

Traditionally, physical security and information security were completely isolated roles with no interaction. As the tools used for physical security increasingly involve a technological component, and the assets protected by information systems increase in value, the segregation has blurred, and increasingly the departments have become integrated.

In the 01960s, nearly all security was physical. While there were some electronic systems, and they may have been protected with passwords, the primary control was the location of the terminals, where the wires were accessible from, and the simple fact that electronics were essentially non-portable. As computers increased in processing power for their unit size, connectivity increased, and more assets were stored electronically, the need for information security increased. Simultaneously, those responsible for physical security began to use tools that were connected and computer-based.

Today information security specialists must consider physical security (how exactly will the server room be secured), and physical security specialists must ensure that their information systems are protected (how will the physical access control system network and servers be secured). Will the server room be secured with the physical access control system? Could a network vulnerability compromise the server room physical security which could then compromise the physical access control system which then compromises everything else in a cascade failure?

The key to remember is that in all cases, the goal is to reduce risk. The quality and types of locks used for a room are going to be dependent upon the value of the items that they are protecting. The evaluation process used to determine what controls are necessary on that room are essentially the same as the process for determining what technical controls will be put in place to protect a server.

Physical security is enough a part of information security that (ISC)^2 includes a whole domain on physical security as part of the CISSP exam. Similarly ASIS certification considers information security to be a part of information security and includes a section information security in its exam.

WHAT SKILLS THIS GIVES YOU

Physical security and information security both rely on the same processes for risk assessment: identifying the threats, understanding the vulnerabilities, and putting controls in place to bring the level of risk to the point that the organization is comfortable with it. Though the details may differ, the processes are similar enough that they can be transferable between the two areas.

Similarly, many penetration tests now include a physical access component. Are the pen testers able to gain access to the building when they should not have been? Doing so will often give them additional access to information assets, and so physical access tests are included when an all-encompassing pen test is run. Being familiar with the underlying physical security processes, procedures, and tools will increase the value of the pen tests that you run.

WHAT SKILLS YOU MIGHT STILL NEED

The move into physical security from information security will require you to become familiar with the tools and terminology used in that field. Understanding such things as the concepts behind crime prevention through environmental design, what features make locks pick-resistant, and how to determine wall and door fire ratings requires understanding of new terms and concepts, and new definitions of old terms.

The move from physical to information security is similar. ASIS International physical security certifications, Associated Locksmiths of America (ALOA) certifications, or locksmithing apprenticeship and experience all will give you a solid grounding. Information security people can learn them in similar ways from physical security people, or from other information security people who learned physical security the hard way.

HOW TO FRAME YOUR SKILLS

For those interested in making the transition, you will need to focus on the areas where the two areas of security overlap: risk assessment. Demonstrating your ability to perform your skills, and apply them to the areas of information security that overlap with physical security will provide you with some common ground for discussion. Despite that, this type of transition will not be easy, as you will be competing against those who come from a background specializing in physical security.

DIFFERENCES BETWEEN WHERE YOU ARE AND INFORMATION SECURITY

CULTURE

Traditionally, the physical security specialist often came from a military or police background, because those professions provided training and experience that could be transferred to the private sector. The rigid structure found in those environments is very different from the freewheeling, independent environments that many information security professionals are used to. But as information security becomes a fundamental component of business, it will become more formalized, while those working in physical security are beginning to see the value in out-of-the-box thinking that is more traditionally seen in information security. There will be an increasing convergence between physical security and information security, as they become sub-specialties of a future holistic full-security discipline. A move in this direction is already apparent in companies that have a single converged security department, and is further demonstrated by the fact that the $(ISC)^2$ CBK contains physical security and ASIS has an information security domain.

TERMINOLOGY

Risk, threat, vulnerability, and other terms used for risk assessments will have different nuances based on the background that you come from. That really should not be a surprise, as you will find that even within information security, different standards organizations have different definitions. You will need to become familiar with the specific definitions for those you will be working with, to ensure that you are working from the same vocabulary.

CHAPTER

2.f TIER 2—LATERAL: MILITARY

INTRODUCTION—HOW THIS APPLIES

The military includes such a wide range of skills that there is no way to cover them all here. However, as the military does provide an entry to information security for some, it is worth discussing.

Military service involves a wide range of roles. From the foot soldier that so many associate with the military to the quartermaster to the cook—all require their own sets of skills. As of the time of this writing, one of the biggest concerns within the military was that there were not enough information security experts to meet demand. Because of this, you may find your entry into information security through military service.

Keep in mind that the roles available to you within the military will be dependent upon your existing skills and what skills and roles are needed by that particular military organization. Being able to demonstrate technology skills (either through a degree or certification) will make it more likely that you will be able to convince your recruiter to slot you to one of those specialized roles. But there are no certainties, and recruiters may make unrealistic promises that cannot be kept.

Each branch of the military has its own roles. You can investigate to get more specific details not only about what roles are available, but what is required to get into that role. A quick Internet search of the branch you are interested in along with "information technology" in the search will get you to the various options available.

WHAT SKILLS THIS GIVES YOU

Beyond the specific skills you gain for the role you enter, you will learn discipline and teamwork. You will rely on others, just as others will rely on you to achieve a common goal—but one you will often quite deliberately not fully know or understand, but are able to work toward because of military discipline.

Many employing organizations view former military service favorably because of these skills and experience, and some have special programs for military veterans and current military people.

WHAT SKILLS YOU MIGHT STILL NEED

What options are available to you in the military will depend upon what skills you bring when enlisting. Civilian life and corporate culture are very different from the military. For example, some civilian organizations have specific uniforms and formal requirements, but most do not and it will be awkward to wear a three- piece suit when everyone else is wearing jeans and t-shirts.

More importantly, you will need to develop a feel for when processes need to be followed and when an ad-hoc approach is warranted. Civilian jobs will have less structure and more personal responsibility and flexibility. This is, however, a sword that cuts both ways. The same structure that restricts you, also protects you, and as that structure lessens in the corporate world, you will be exposing yourself to more risk as well as greater responsibility.

HOW TO FRAME YOUR SKILLS

There are multiple ways to enter the military. For young adults and students, a high school military academy or reserve officer training corps during college may provide opportunities. Which path you choose will determine how to best present yourself. This is a broad topic and outside the scope of this book.

DIFFERENCES BETWEEN WHERE YOU ARE AND INFORMATION SECURITY

CULTURE

The military's rigid structure and information security's renegade image are definitely at odds. However, as information security becomes a more integrated part of business, the need for professionals who are comfortable working within the system has continually increased. While former military may or may not be comfortable at the anarchy seen at security conventions such as DEF CON, they certainly are finding civilian employment.

TERMINOLOGY

The military frequently has its own set of terms that civilians may or may not be familiar with. If you move to civilian information security you will need to understand how military jargon maps to civilian terms. Some civilian information security terms came from the military, like "perimeter," and "defense in depth," but others aren't quite as common yet, and might be misunderstood or have poor connotations such as "kill-chain" or "exclusion zone."

CHAPTER

2.g TIER 2—LATERAL: LAW ENFORCEMENT

INTRODUCTION—HOW THIS APPLIES

Like other broad fields, Law Enforcement encompasses a wide range of skills and abilities. While your local police officer is probably your image of Law Enforcement, there are many roles beyond, especially in larger organizations. This breadth of potential roles expands even further if you include federal law enforcement agencies.

The most obvious (and first) overlap between information security and law enforcement is in the area of computer forensics. The Incident Responder role will need to have some level of computer forensics skills, and larger teams often have a dedicated forensics person. That said, as information security expands, so do the opportunities within law enforcement. The US FBI now has a wide range of career opportunities in information security as does the US Department of Homeland Security. The US Secret Service will form Electronic Crimes Task Forces, which include establishing partnerships with a wide range of organizations. See the Appendix for links to these and other US government resources.

This breadth of opportunity within law enforcement provides entry points for those individuals who are comfortable working in a highly structured and regulated environment.

WHAT SKILLS THIS GIVES YOU

For those roles that are not information security-related, concepts in physical security will be at least part of your duties. Law enforcement skills and experience include formal process and procedures, crisis management, psychology, human nature, and de-escalation, and the justice system, all of which can be valuable in a holistic view of information security.

WHAT SKILLS YOU MIGHT STILL NEED

The most likely gap in your skills will be in your technical skills. This gap can be filled through formal or independent study.

HOW TO FRAME YOUR SKILLS

For those in law enforcement hoping to move into information security, the easiest path often will be within your organization or to a similar organization. If, through some form of study or certification program, you can demonstrate an aptitude for learning computer technical skills, lateral opportunities within your organization are likely the simplest road to information security. Take advantage of the fact that you are already established within the organization.

DIFFERENCES BETWEEN WHERE YOU ARE AND INFORMATION SECURITY

CULTURE

The cultural differences between law enforcement and the information security world has changed over time. While conferences such as DEF CON used to be primarily aimed at individuals, with an occasional law enforcement individual, they now are seen as important educational opportunities for information security professionals from all backgrounds.

There will still be distrust in some circles regarding those coming from a law enforcement background, but that should not limit one's career opportunities.

TERMINOLOGY

The biggest challenge will be ensuring that a common set of definitions is used, especially in the area of risk assessment and risk management, as those terms vary.

CHAPTER

2.g.1 TIER 2—LATERAL: LAW ENFORCEMENT—STORY

JOSHUA MARPET

Although all information security stories are unique, Joshua's is unusual. Although Joshua has a background in information technology, he was also a cop, a horse show announcer, a blacksmith, a certified horse dentist (still is), a bouncer on New Orleans' Bourbon Street during the Super Bowl and Mardi Gras, and waiter. To the best of his knowledge, he was the only Jewish jail guard in the United States Deep South. Joshua also has a BA in psychology.

His work in information technology was as a sysadmin, forensics, pen testing, and infosec entrepreneur. As he says, "If it's a job, I've done it;" and in those jobs, he says, "There's always got to be someone I can learn from."

He has had several turning points in his career, but the one that got him into information security was when he was working as a system administrator when his co-worker accidentally shut down one of the largest payroll systems in the US. The co-worker just didn't expect the shutdown command to work for non-privileged users. That was the moment Joshua decided to go into information security.

Joshua was in the dot com boom and bust of the late 01990s and early 02000s, and made the mistake of accepting stock options instead of actual pay. So he took his stock options and ended up as a Louisiana cop during Hurricane Katrina in 02005, and learned the hard way how to be prepared for disaster recovery in the devastating damage and floods. He stayed behind, but evacuated his girlfriend and children to New Jersey along with his mother.

Weeks later, he was able to tell his family that he and the house had survived the hurricane and they could come back to Louisiana. But his family had already settled down in New Jersey and wanted to stay. Even then, in the aftermath of Hurricane Katrina as he tried to leave and get to New Jersey, he had beat up someone to keep his truck (he had an ASP baton, they didn't, he won).

He did get to New Jersey and was reunited with his family. But he couldn't be a cop in New Jersey—he was too old at age 33—so he got back into computers, system administration, and various jobs.

Since then, he's learned digital forensics and pen testing, spoken at 30 to 40 conferences including Black Hat and DEF CON, and does security advisory services.

He got fired for doing a DEF CON presentation about shooting a camera from a flare launcher, that was later seen by someone at his employer, but he only found out the reason a year later. That firing prompted him to start his own company, and he's glad he did. He founded Guarded Risk. He says, "That exit story got me to a new origin story."

In the past year, he's been doing startups but this time he's getting paid by "taking stock options in addition to this thing called money."

He has a client list and is making money. His business is building and growing: “you can see the growth that’s real magic.”

Some advice from Joshua:

“It’s a few different things that make a difference in infosec, who you know, what you know, and who knows what you know.”

On being an entrepreneur: “You’d better have a lot of money squirreled away or be prepared to be poor.”

He recommends having friends, helping people, and volunteering. He’s on staff at several information security conferences where he’s known as the “infosec megaphone” and the “infosec therapist.”

CHAPTER

2.h TIER 2—LATERAL: LEGAL—LAWYERS

INTRODUCTION—HOW THIS APPLIES

Information security and the legal professions used to have no overlap, well except when a lawyer defended or prosecuted an information security criminal case. Today, with the number of laws guiding the use of data, federal requirements for minimum levels of protection for a wide range of data types, and business agreements with third-party suppliers (such as cloud providers), information security is becoming an area of focus in the legal field.

The regulation of information security does not mean that lawyers are starting to become educated in the information security field. However, each side is becoming more aware of the issues the other faces, and the two fields are more likely to be working together. For example, when setting up an arrangement with a cloud provider, the legal team may work with an information security professional to determine what types of auditing and reporting would be necessary to meet certain standards the company must abide by.

The regulation of the financial industry has also led to greater interaction between legal and information security teams. Determining how to meet requirements for traceability, information isolation, and compliance will often require cooperation between both skill sets.

WHAT SKILLS THIS GIVES YOU

Being a lawyer will help you understand the drivers behind certain information security positions. The ability to evaluate and interpret the regulations surrounding the data a company manages is a key skill at the upper levels of information security. Because of this, lawyers are more likely to move to management or upper management positions such as chief information security officer (CISO) or chief privacy officer (CPO). In fact, the demands of the CPO position often make someone with a legal background an excellent fit.

WHAT SKILLS YOU MIGHT STILL NEED

Because of the way that the legal field and information security overlap, the fit is either going to be very good (such as the Chief Privacy Officer example mentioned above), or very poor (for example, as a pen tester) where strong technical skills are required. Thus, transitions from Lawyer to information security are generally constrained to those positions that focus more on dealing with regulatory compliance and other legal-focused drivers.

To move from information security into the legal field, exact requirements will vary from state to state. In most cases, passing a bar exam is required, but that is not true in all cases. That discussion is left for other texts. The legal field isn't just about lawyers and law degrees and passing the bar exam, considerable work in the legal field is done by paralegals and legal secretaries. Law firms also often directly employ non-legal staff for specialized knowledge such as technology, forensics, and investigators.

HOW TO FRAME YOUR SKILLS

While it is understood that those who are capable of filling a CISO- or CPO-level position are unlikely to use this book for career guidance, the process for moving into a new position is going to be similar to the general process all job hunters go through: find an organization that needs skills and demonstrate how your skills are a good fit for their needs. In the case of CPO, this would include demonstrating an understanding of the regulations the organization is held to, and how the applicant can help the organization achieve compliance with those regulations through their past work experience.

DIFFERENCES BETWEEN WHERE YOU ARE AND INFORMATION SECURITY

CULTURE

Law and information security both must deal with an ever-changing landscape. However, the process for change within the legal profession is much more methodical and event-driven. New laws and regulations generally come into being as a reaction to events that have occurred in the past, and when a new law comes into place, it may take years for the full impact of the law to be felt or even understood. The cycle of a new law coming into place, issues being found, and new laws being enacted can be a process that extends over years, or even decades.

In contrast, the action and reaction between a new exploitation technique and new security feature will often have a life cycle time of weeks or months. Though this life cycle time is increasing as infrastructures become more ingrained (and there are examples such as Internet Border Gateway Protocol (BGP) security that are taking decades to fix), response times are more often likely to be rapid, especially if your organization is at risk.

A similar split occurs when dealing with active attacks. If you are involved in a lawsuit, you can appeal deadlines, ask a judge to impose rules upon your adversary or avoid rules from the adversary, and have months to evaluate your course of action.

When dealing with an active security incident, there is no such arbiter, no such flexibility of time. The attacking entity will do whatever they can, without any external restrictions beyond the technology. In contrast, you may be limited in that you are bound by legal actions. For example, if a botnet is attacking you with a distributed denial-of-service attack, there is no law that allows you to disable the attacking computers. You may be able to work with Internet service providers (ISP) to limit activity, but you cannot directly counterattack.

TERMINOLOGY

As always, when two drastically different areas are working together, it is important to ensure that the vocabulary in use is the same. Regulations frequently define the vocabulary in use, and the information security professional must ensure that they are using the appropriate terms for the environment in which they are working.

CHAPTER

2.i TIER 2—LATERAL: SALES

INTRODUCTION—HOW THIS APPLIES

The business world runs on sales. Most jobs are classified as “overhead” or cost center which costs money or “revenue generating” or profit center which makes money. Most information security roles outside of consulting firms are viewed as overhead. Sales, however, brings in the money—the lifeblood of all organizations. While a move from sales to an information security position will be challenging (because of the technical skills required), the move from information security to sales does have some paths available, should that be the direction you wish to go.

Sales is not for introverts. If you thrive working head-down in your cubicle, focusing on the technical task at hand, then the daily introduction to new people with a wide range of agendas will be challenging, even stressful, for you. You need to be prepared, and comfortable with, walking into situations where the people on the other side of the table may be merely curious, potential allies, or outright antagonistic, and not knowing which when you sit down.

If you thrive on the challenge of “hooking” the client, and turning a “maybe” into a big sale, Sales is a path to consider.

For technical people, the path to sales will generally depend upon the size of the organization you are in. If you are in a smaller organization, sales may be one of your duties, simply because the company does not have anyone who is dedicated to sales. If you naturally become the leader at the sales meetings, that may end up becoming your full-time responsibility.

There are many sales teams that are made up of a combination of sales only (those specializing in the person-to-person interactions) and sales engineers (those specializing in providing demonstrations, answering technical questions, and describing architecturally how the product or service would be a good fit for the organization). As a technical person interested in moving into sales, the sales engineering role will provide you an opportunity to learn about sales in general and, if you display an aptitude for sales, an opportunity for moving into a sales-only role. One warning, though: good sales engineers are hard to find. If you are very good in that role, the company you are working for may not be willing to have you move out of the technical role.

WHAT SKILLS THIS GIVES YOU

Sales is all about people skills. Your ability to understand what others want and match that with what your company can provide is the core of sales. You must possess a combination of communication and negotiation skills to be successful. While these are skills that are useful in any role, they are required for sales.

WHAT SKILLS YOU MIGHT STILL NEED

For any given product or service, you will need to understand the specific market. Why are people interested in this type of product? What drives people to purchase your product rather than a competitor's? What product features are going to get you in the door, and which are going to close the sale? Understanding the value of these questions, and the answers to them, is going to aid your success in the field.

HOW TO FRAME YOUR SKILLS

Demonstrate your abilities in the working with others. If you have spent time doing some type of customer service role (e.g., help desk), emphasize your ability to work with a wide range of individuals, identify their specific skill sets, and provide them solutions to their problems. The same pattern is used with sales, except you may first need to convince the potential client that they have a need.

Additional relevant skills could be displayed in your ability to work with sales representatives. If you have experience on the client side of the table working with sales representatives, use that to demonstrate your abilities in sales. If you are an experienced negotiator working with vendors, that gives you insight into what drives those individuals, allowing you to focus on their specific needs.

DIFFERENCES BETWEEN WHERE YOU ARE AND INFORMATION SECURITY

CULTURE

Technical skills are all about building the thing. Sales is all about building the relationship. This creates a different dynamic in how you work to solve the problem. Depending on the environment, sales can be much more antagonistic with others that you are working with to reach the end goal. When battling a technical problem, generally everyone you are working with will have the same goal. In sales, that is rarely the case.

In sales you will find that your peers are not necessarily your allies. In many environments, managers explicitly pit sales representatives against each other in an effort to drive the overall top line. As such, there is frequently an underlying wariness associated with helping your peers (or receiving help). If you are not comfortable working in a cutthroat environment such as this, you will either need to be very choosy in the company you work for, or this may not be the field for you.

TERMINOLOGY

Be prepared to learn a new language. You'll likely be comfortable with technical product terms, the language of business will become more important to you. A simplistic example is the concept of return on investment (ROI). Many business decisions to purchase will be based on the need to have an ROI above a certain point; be prepared to ask to demonstrate how this product will provide a return to the company making the purchase.

That is a basic example, and the business language around your product will vary. Working as a Sales Engineer, or having been responsible for making these purchases, will often provide insight into the language of sales in your industry.

CHAPTER

2.j TIER 2—LATERAL: PROJECT MANAGEMENT

INTRODUCTION—HOW THIS APPLIES

Project Management: the art and science of herding cats. While that may not be the formal definition, if that is your role, it will feel that way at some point. This role is similar to that of Security Facilitator, with the difference being one of degrees. A Security Facilitator will generally be responsible for monitoring a few (3–5) people, while a Project Manager has responsibility for projects that could potentially include hundreds of people. In those cases where a project spans such a large number of people, there will frequently be multiple tiers of project management, rolling up to a top-level manager.

These large-scale projects span large numbers of disciplines and will invariably impact the organization, either positively or negatively. Because of the scope and impact, these people may report to C-level executives within the organization. They also will have more authority to draw in resources from within the organization.

WHAT SKILLS THIS GIVES YOU

The organizational and project tracking skills required to manage large projects are valuable for managerial roles such as CISO. The people skills, sometimes referred to as soft skills, used for project management are relevant to any career path, though they are used more in managerial and lead roles.

WHAT SKILLS YOU MIGHT STILL NEED

The most likely gap will be in security-specific knowledge. While project managers will be good at understanding and coordinating a project, the underlying business process and technical expertise is left to others. Often, project managers who work closely with those with operational knowledge will develop some skills in those areas, but rarely enough to move into those areas. Rather, this provides them with a greater understanding of what those people do when it comes to managing people in those areas.

HOW TO FRAME YOUR SKILLS

When moving into managerial roles, the people and project management skills that you have developed will be the most useful in demonstrating your ability to take on these other roles.

DIFFERENCES BETWEEN WHERE YOU ARE AND INFORMATION SECURITY

CULTURE

When moving from project management to information security, the cultural differences will vary dramatically, depending on the roles or organization you move into. The managerial roles at large organizations will have little variance from your current role, but at a smaller, more technology-focused organization, the lack of formal process could be jarring.

TERMINOLOGY

Project Management has its own terms and definitions that are used for describing the state of a task and the project as a whole. As this vocabulary has essentially no overlap with information security terminology, the task is to learn new terms, rather than understanding the difference in how the same terms are used.

CHAPTER

2.k TIER 2—LATERAL: NON-IT ENGINEERING—ARCHITECTURE—SCIENCE

INTRODUCTION—HOW THIS APPLIES

STEM—science, technology, engineering, and math—is a general term for areas of education and jobs that are critical for overall technological advancement of a society. This covers such a broad range of fields that it is impossible to describe the details associated with the various fields here. However, there are some common characteristics that can be discussed.

These fields all require some level of degree in order to enter them. A bachelor's degree will be necessary to function at the Do level—implementing solutions, and using their knowledge as part of a team. For leadership roles, a master's degree is often required. To be in charge of any form of research, a PhD will generally be required. This need for a degree is in stark contrast to information security. Though formal education is becoming more available for information security professionals, it is entirely possible to have a complete career without ever attending an accredited university.

Many security professionals in the industry over 15 years originally came from science backgrounds. In many cases, after getting their degree, they did not enter the workforce as a scientist, but rather had some level of computer background that they developed on the side, possibly while in college. For those with only a bachelor's degree, the job opportunities in computers were much more plentiful than those in their major, so they went into information technology roles. From there, they progressed to information security roles.

If you are interested in seeing an early move from sciences into information security, take a look at *The Cuckoo's Egg* by Clifford Stoll. It is a classic and entertaining book on information technology infrastructure and security in the 01980s. While the technology has advanced significantly since then, the underlying processes involved have not changed much: Find anomalies, be creative, investigate until they can be explained. Although most astronomers don't expose Cold War spy rings, Clifford Stoll did.

WHAT SKILLS THIS GIVES YOU

The domain-specific knowledge found in science and engineering will rarely be applicable to information security. In contrast, the general skills—problem solving, detailed record keeping, plan development and the like—are all very appropriate for information security.

In many cases these roles will include computer skills such as writing code or basic computer administration. These can be a foundation for computer security.

WHAT SKILLS YOU MIGHT STILL NEED

For those moving into information security, the role you are interested in moving into, and what you are moving from, will dictate what skills you need. Auditor-type roles may be easier to move into if you have experience doing scientific or engineering audits. Your familiarity with those processes will reduce the amount of information security-specific skills you need to develop. If you are moving from chemistry to pen testing, the path will take longer, as you will need to develop many information security-specific skills.

For those moving out of information security, a degree in the field that you wish to move into will probably be required. In fact, this author has no examples of a degree not being required, but I hesitate to say always, as there may be some exception out there I am not aware of.

HOW TO FRAME YOUR SKILLS

Moving into information security will require you to demonstrate a minimum level of technical ability to do the required tasks. Additionally, you will need to demonstrate the ability to learn and develop rapidly within your new field. Give examples of your technical skills: the program you wrote to collect data from a sensor, the RAID array you managed to store the data, or any other examples of information technology-related tasks that you have successfully dealt with in your career.

DIFFERENCES BETWEEN WHERE YOU ARE AND INFORMATION SECURITY

CULTURE

The most likely difference you will encounter will be the degree of uncertainty that information security sees as compared to science fields. While engineering and sciences all have uncertainty, it is generally treated as a variable to be included in the analysis, and used to calculate the margin of error you must work within. Conversely, in information security you sometimes have to accept that complete unknowns exist and work as best as possible to contain them, never having complete control over the variables.

Information security best practices may call for certain policies and procedures, but if those are not feasible for a given corporate culture, you will often have to accept that the business is going to accept that risk, and not implement them. This lack of control can be uncomfortable; you will need to determine if it is comfortable for you.

TERMINOLOGY

Every discipline has its terminology and communication styles. Most engineering and scientific disciplines have had decades, if not centuries, to work to build this common language. In information security, the language is less standardized because it's so immature, and much more dynamic, to meet the still changing needs of the field. In a mature field like chemistry, an atom with two protons is always helium, and has the international standard notation of $_2$He. In information security, neither concepts nor terms are standardized. The plan for keeping an organization running could be a Business Continuity Plan (BCP) which is common in business, a Continuance of Operations Plan (COOP), common in government, or a Disaster Recovery Plan (DRP) as a general plan for disasters. Risks, threats, and other terms will have different definitions, and you will need to be able to adjust based on your audience.

CHAPTER

2.1 TIER 2—LATERAL: ACCOUNTING

INTRODUCTION—HOW THIS APPLIES

Accountants manage the money. Whether it is the going in, the going out, or the movement within the organization, accountants are responsible for ensuring that the paths are completely traceable and appropriately assigned. Your level within the organization will determine the scope of this authority. For example, an entry-level position may only be responsible for a single process, such as dealing with the accounts payable for a single group or department. As you rise within the company, the scope of your responsibility will grow as you manage those doing the day-to-day work.

Being in accounting, with any level of authority, will require being a certified public accountant (CPA). The exact requirements vary by state, and are beyond the scope of this book.

In many organizations, accounting and information technology are closely aligned. The CIO and CISO may report to the CFO. Also, since the financial information systems are among those that need the highest security within an organization, it is common for those who work in information security to work closely with at least some of those who work in finance.

One of the main reasons for this close relationship is the audit and regulatory requirements the financial division is required to deal with. For public companies, regulations such as the US federal Sarbanes-Oxley law (SOX) include information security components to ensure that financial information of the company is appropriately protected. There may be other standards, such as the Payment Card Industry Data Security Standards (PCI-DSS), that overlap between the financial and information security groups, creating the need for a close working relationship between the two areas. Even smaller companies with no regulatory requirements will have financial audits that will frequently include an information security component.

WHAT SKILLS THIS GIVES YOU

The attention to detail, understanding of the documentation, and audit experience associated with Accounting is very appropriate for information security. Depending upon your target role, Accounting can thus have some clear paths to information security. Those preparing for financial audits may find themselves involved with information security components of those audits. From there, they add the information security-focused audits such as PCI-DSS to their repertoire, and thus become information security auditors.

Similarly, at the higher levels, the move from finance into roles such as CISO or CPO is often appropriate, and requires more understanding of general processes and company needs than an understanding of the specific day-to-day tasks.

WHAT SKILLS YOU MIGHT STILL NEED

While the transitions mentioned above, focused on process and procedure, can have relatively seamless transitions, moves to more technical roles will require developing appropriate skills. These skills will generally not be attainable as part of your regular job duties, so you will have to develop them on your own based on your specific goals.

HOW TO FRAME YOUR SKILLS

As you can see from the previous discussion, your path will dictate how to frame your skills. Moving between closely aligned fields will require the ability to work within an information security framework rather than a financial one. Demonstrate an understanding of the frameworks in general, and any specific information security knowledge you may have.

DIFFERENCES BETWEEN WHERE YOU ARE AND INFORMATION SECURITY

CULTURE

Making the lateral move from Accounting to Audit, CISO, or CPO will generally not have much difference in the culture between where you are and the new role. However, the type of people that will be working with you or for you will change somewhat. In accounting, there generally is little appreciation for someone who comes up with their own accounting method, and implements it. The accounting field is based on standards. An accounting manager needs no particular creativity, except when someone is trying to commit a crime.

In contrast, creative information security professionals can be some of the most valuable, as their problem-solving abilities can help find issues that would not have been found using standard practices. Similarly, the enthusiasm that many information security professionals have regarding the security process may be unfamiliar to those in accounting, who are less likely to feel passion for a specific accounting method. Being able to work with them is important for those who move into information security management.

TERMINOLOGY

Fortunately there are fewer cases of the same vocabulary used for different purposes as there are between information security and other fields. Because of that, it is more a case of learning the information security vocabulary, rather than the differences between the two fields.

CHAPTER

2.m TIER 2—LATERAL: BUSINESS ANALYST

INTRODUCTION—HOW THIS APPLIES

Whenever a project begins, there are general high-level guidelines put in place by the management. “Roll out a new ERP System by the end of next year.” “Have the new phone system in place by the end of the next quarter.” Those high-level goals, while enough to get funding for resources, do not provide detailed requirements for what people need to be working on. The Business Analyst is responsible for taking those high-level requirements and, through discussions with various business process owners, generating a detailed set of requirements that the technical team can use to guide its implementation.

Identifying exactly what a business wants and needs takes work. Existing business processes may be used as models, but are often poorly documented or defined. An engineer cannot implement a task such as: “Joe fills out a form.” It is the Business Analyst’s job to identify what information is collected by the form, and if there are prerequisite activities that need to take place. Could the contents of the form change based on different circumstances? Finally—and this is the component that often is the biggest challenge—why is this information being collected? Sometimes the reasons can be clearly identified. Sometimes the answer is “Well, that’s my job,” at which point the impacts of that process on other areas need to be determined.

Whenever new processes are implemented, there is a drive to improve upon the current system. This then drives the question of what would be an improvement. If there are five people in a room, there will likely be at least seven different suggestions to improve the process presented. Yes, some people will have multiple ideas. The business analyst will be responsible for working with the business to produce a single model, and from there creating a set of requirements for the technical team to work from.

This generation of technical requirements means that a good business analyst will need to be able to translate from business to technical and back again. The more the business analyst can understand about the technical side, the greater the likelihood that the requirements generated will be things that can actually be implemented.

WHAT SKILLS THIS GIVES YOU

In many ways, a security professional functions as a business analyst. Security professionals need to be able to understand the business drivers behind their technical implementations, so they are not mitigating risks that the business does not care about, or leaving risks unaddressed that the business needs to address. The ability to work with the business leaders to understand the business drivers behind a project is an important skill for advanced security professionals.

WHAT SKILLS YOU MIGHT STILL NEED

Successful business analysts who want to get into information security will probably need to improve their background. Consider the security ramifications of what you have already worked on. For example, ERP systems almost always have security issues. If your ERP vendor doesn't understand security, there may not be much to find, so instead dig into the system components: the database, application, operating system. Look at vulnerabilities that directly affect a system that you already understand. For higher level learning, the broad but shallow (ISC)^2 CISSP is a start, not necessarily as a certification, but as a framework for learning.

MOVING FROM BUSINESS ANALYSIS TO A SECURITY FIELD

Security professionals, especially those who work in the risk mitigation areas such as Risk Assessment, Security Assessment, and Vulnerability Management, need to have a deeper understanding of the underlying technical issues than most Business Analysts have.

MOVING FROM SECURITY TO BUSINESS ANALYSIS

While security professionals often have an understanding of business processes, their ability to model business processes and identify all of the key components associated with a business process often needs improvement. Security processes are only touched upon at the high level, and deeply at the technical level, and the middle area is not focused upon as much.

HOW TO FRAME YOUR SKILLS

For those business analysts looking to move into security, focusing on your ability to understand both business and technical issues is extremely valuable in the security field. Furthermore, the ability to demonstrate the ability to learn technical skills is important for many security positions, and having examples of where you have done this in the past will be important for those wishing to transition to those areas.

DIFFERENCES BETWEEN WHERE YOU ARE AND INFORMATION SECURITY

CULTURE

Security roles exist across a wide range of corporate cultures. Both maverick security personnel and military-style precision-based security engineers can find their place, thrive, and excel. Business analysis culture tends more towards the well-defined process and procedure end of that spectrum, so finding a security position will require either being willing to limit where you look, or be willing to expand the cultural environments in which you are comfortable working.

TERMINOLOGY

It's necessary to understand the security field's many specific terms and concepts related to the analysis of risk. The process of analyzing threats, vulnerabilities, and exposures, while similar to dealing with business processes, involves enough outside actors that there will be some necessary adjustment.

CHAPTER

TIER 3—TEACH 3.0

WHY TEACHING MATTERS

By now, you have learned things and done things. After a while, though, you get burnt out or, as author Ursula K. Le Guin once put it, "done with doing." In business, this may manifest as feeling a general lack of direction. When you are learning, your teacher gives you direction. When you are doing, you should have goals. However, once you achieve your goals, where do you go next? Teaching helps answer this question.

By teaching others, you may find yourself learning better than ever before. Your students will ask new questions you never considered which include new ways to be wrong and to learn, but also new ways to be right. As they learn, you'll also learn and see new ways for you to learn your material. Most importantly, working with someone else as you explore a field will lead you to explore nooks and crannies that you otherwise would have overlooked.

To complete your learning process, you must teach someone else what you know.

Teaching can serve as a business force multiplier. In physics, a force multiplier, such as a lever or wedge, increases the amount of force you can place on an object. In military terms, a force multiplier makes a military force more effective. For example, a fleet of drone craft can simplify reconnaissance to where a handful of analysts can do the work of hundreds. Information security can also use force multipliers. We must improve the effectiveness of both our tools and our people. By teaching others how to do security, you learn your tools better, and find new tools and ways to combine them so they are greater than the sum of their parts. By teaching others how to learn security, you increase the number of people on your side. By teaching effective techniques, you all spend less time, and become force multipliers yourselves.

To oversimplify the complex attack/defense landscape, there are attackers and defenders. If you are reading this book, you may be most interested in defense. This is a difficult path to choose, as attackers tend to be better funded for their goals and less restricted by laws and ethics. When you choose defense, you must accomplish your tasks with fewer resources, as the attackers focus on multiplying their forces with constantly improving tools. If you choose a mentorship path, as you teach more and more people, the number of people on defense begins to overwhelm the attackers. While it's true that a small number of people with quickly improving tools will win over those who move less quickly, a large number of people with a diverse set of slowly improving tools will win over faster attackers, as it takes a lot more human-centered analysis to attack such a wide group of defenses.

These concepts can be generalized into the Observe, Orient, Decide, Act (OODA) loop, which was originally developed for training fighter aircraft pilots—which, incidentally, are both primarily defensive and also creative. For example, observation includes the first knowledge of a vulnerability such as through a security advisory, patch release, or incident. Orient is understanding the consequences of an advisory, getting prepared for patch deployment, or responding to an incident. Decide is taking the output of observation and orientation and making a decision; not all advisories are applicable, not all patches should be applied, and not all incidents will have consequences. Act follows through on the decision, and must also be done to completion if the decision was correct at all. A shorter, faster, tighter OODA loop allows for faster response, more frequent correction, and better results. An OODA loop that runs inside of your adversaries' own OODA loop will mean your response occurs before your adversaries even know their attack didn't work as expected.

Research your environment and your industry, know what threats your organization faces both in terms of technology, your competition, and your customers' expectations.

SHORT-TERM TEACHING

There are at least as many different ways to teach people as there are people. However, many of them break down into short- and long-term teaching. With short-term teaching, you are likely working with one or more people on specific projects. This can be as short as an evening tutorial or as long as a few months helping people learn well enough to transition from Learning to Doing. You will be helping people learn either online or offline. These modes are very different, because only with offline teaching will you be physically in front of your student(s) and able to tell more immediately when they're failing to understand. Online teaching and learning are more difficult.

ONLINE TEACHING

Teaching online comes in several flavors. You can have a formal relationship with your students or, as is far more common, a highly informal one. Formal online teaching will be much like the offline teaching described below. You will define a syllabus, set a schedule, and run through the process with one or more students. The biggest difference between online and offline learning is that online learning requires you check in more often.

Checking in is critical in all modes of teaching, but it's even more important when working online. If you can't directly interact with your students, you will only know how well they're doing when they fail (or pass) a quiz. Quizzes and tests exist both for you to gauge how well your students are doing and how well you are teaching them. When teaching online, even formally, it is vital to get frequent feedback from your students as to their understanding. This can be done formally, by tests and quizzes, but it should also be done during the process of teaching, as you ask "do you understand?" and "does that make sense?" Ask people to explain things back to you or, as often works the best, have them implement something that illustrates some small piece of what you are trying to teach, such as in class laboratory exercises.

Teaching less formally online is vastly different. In this mode, you may be finding someone on Twitter with an immediate problem, someone on IRC that needs a bit of advice, or a long-term friend on a mailing list that needs help with direction in life. What makes each of these scenarios different is that you won't have any sort of prep time or structure. Instead, what you get and must notice are teachable moments.

A teachable moment is a question or statement that indicates a student is both in need of assistance and open to new ideas. It may take some time to notice teachable moments, and as you practice you'll make mistakes and anger people. It takes time to get good at this sort of teaching, but it doesn't have to require emotional involvement. If you get hurt every time someone misunderstands or demonstrates their unwillingness to learn, you'll never build the experience needed to get truly good at teaching and, therefore, will never experience the rewards of doing it well. Take advantage of the physical distance to provide emotional distance as needed.

You may have an opportunity to do one-to-many teaching in an online mode. Other than classes, which are addressed in greater detail in the next section, the Internet provides opportunities for you to teach via forums, mailing lists, webinars, and videos. Forums and mailing lists are text-based communication, so when engaging in these, realize that you're not just teaching for now, but also the future. Online media survives for a long time, and it is not uncommon to receive feedback on items you wrote and forgot about years ago. Take the time to write up your thoughts; proof your work and check it for spelling and grammar. Then do a fact check and test and take a short rest break and then do a final review before sending. Just taking one extra hour between writing and hitting the Send button can be the difference between creating a throwaway post and creating knowledge that will stand the test of time.

The same logic applies to multimedia pieces like webinars and online videos. Both require preparation. Webinars and videos often start with rough scripts. Webinars often take these scripts and turn them into presentations. Videos take the scripts, flesh them out, and become recordings. In both cases, it is sadly common to create the concept, perform the script, and be done. However you can do much better with some practice. Following the old programmer's adage of "build one to throw away," if you run a practice webinar or do one video recording, you can assess your performance and find ways to improve it. Few performances are perfect the first time; one rule of thumb states that it takes ten performances to start to become comfortable with that particular material.

Let us be clear. There is little in the working world that is less pleasant than watching your own performance while picking it apart and identifying where you did terribly, except perhaps watching someone else do it. Many people start this process and quit after one try. However, those that push all the way through will gain experience quickly, as each cycle will result in more change. You don't want to practice so much that your performance becomes robotic (a common failing of the approach of making a word-for-word script). However, a bit of formal structure will make your message much easier to understand and will make your teaching much more effective.

A more formal teaching plan will give more of these benefits in a shorter amount of time. However, formal teaching opportunities can be rare. To fill in the gaps, to gain experience more quickly, and to experiment, actively seek out teachable moments. As with everything else, the less time you waste, the faster you'll move forward. The more time you put into teaching, the better you make the world and the more you accelerate defensive capabilities.

OFFLINE TEACHING

As noted earlier, offline teaching has the primary benefit of being able to see, in real time, how you are affecting your students. Maybe you're just sitting down with someone at a conference, one-on-one, discussing your thoughts. Perhaps you're leading a discussion in a user group setting. You could be giving a live presentation to a room full of beginners, your co-workers, or even business owners.

You can choose to be formal or informal. However, since you'll be right in front of your student(s), you'll get immediate feedback from them. It should be immediately obvious when you say something that is confusing or, worse, factually incorrect. This both raises the stakes and allows you to improve significantly faster, as constant feedback results in faster learning cycles for you. It's okay to make mistakes—this is how to learn.

As you go through this process, remember what works and what doesn't, taking notes if necessary. Record yourself, if you can, and ask your students if you can record them. Your worst errors will likely stick in your mind and prevent you from making them again. Highly successful communication methods are harder to remember. If you're lucky, phrasings that resonate particularly well will stick with you and be available for future discussions. If you're like most people, though, they won't.

However, even if you can't record yourself or remember specific elements that work or fail, you can still maximize your success by pre-thinking what and how you want to teach. Odds are that, while you have a lot to teach, the material falls into specific areas into which you've given a great deal of thought. If that's the case, you can think about the type of people you are likely to teach and come up with metaphors that can help them to understand.

This book lacks the room for a full course on metaphor mapping. See the Appendix for several books on the topic or chapter 8 of *Job Reconnaissance* by Josh More. The nutshell version is to think about who you need to teach and what their lives are like. By choosing metaphors that match with their experiences, you will decrease their difficulty in learning. You will understand them better, they will perceive you as being more like them, and—as a bonus—you will look a whole lot smarter. If you're teaching a room full of college students, it would be a fairly safe bet to assume that they have much greater familiarity with modern technology and culture than a similar room full of senior citizens. Some college age students respond well to metaphors that reference video games, and may not need explanations of how the Internet works. In contrast, senior citizens may need help with some technical concepts, but having personally lived through several periods of war, economic depression, and political and social change, will likely find relevance in topics framed as organized attacks and the importance of defending the helpless.

This is just one example of metaphor mapping, based on generational differences. You can find similar metaphor divergence along lines of class, nationality, ethnicity, sex, race, and gender. The more you think about your audience's responses to different concepts framed in different ways, the more you build up your own metaphor "library." With a collection of metaphors that you know will work or fail depending on your audience, you can build toward a more targeted ad hoc method of teaching. This can give the appearance of having a very informal structure, to which people respond to quite well, while still having the formal underpinning that helps to maximize success.

LONG-TERM TEACHING

Long-term teaching is exactly like short-term teaching, except that you have a lot more chances to fail. Since one teaching session builds on the next, and mentorship and community involvement don't follow set curricula, there is ample opportunity for one misstep early in the process to build upon others, until the entire process falls down in a massive cascade of failure.

If that seems a bit daunting, it should. Long-term teaching is the best way to amplify your thoughts through many people. *This means that if you're wrong, and give bad advice, this mistake will be amplified through the community, and could last for years...even decades. Be careful.*

However, that's not to say that you shouldn't try. It is very difficult and extremely rare for any person to hold the same position—and be correct—their entire life. It's far more common for people to realize that they've been wrong and try to correct. It is, of course, difficult to correct your failure without losing face, which keeps people locked into dubious positions for far too long. For example, both Bill Gates and Steve Jobs were often wrong, even publicly, but when they realized it they both corrected it, admitting their mistakes by changing products and company direction. To truly maximize the benefits of teaching as a force multiplier, you must maximize the time you spend on effective methods and avoid the dead ends that result from people retrenching and defending their positions. Recognize and admit when you're wrong and teach others to do the same.

Admit your failures as quickly as possible, and take feedback so you can correct yourself when you're wrong. This is, of course, incredibly difficult to do. When you start teaching, you must not stop the Learn or Do phases. When you stop learning from others and testing your ideas, you lose effectiveness as a teacher.

MENTORING

By this point in your career, you should have good time management. It's unlikely that you'll be a master at it—this book, for example, is being delivered years late (sorry, Syngress). However, it is quite likely that you will have had enough time management failures in your life to recognize them in others.

This ability puts you in a position of helping others learn not just the intended material but also how to learn more effectively. This means you must be comfortable putting them in a position to fail faster, fail smaller, and, hopefully, gain confidence. When you see your people wasting their time, try to point it out to them. More than just reminding them of deadlines or playing boss, try to serve as a true mentor. To be an effective mentor, your students must trust you. This will only happen when they believe they can make mistakes without upsetting you, that you'll do all that's reasonable to help them fix their mistakes, and that you truly care about them as people. Mentorship is an extremely rare and difficult skill. Many people in so-called mentoring programs fail to be good mentors, mostly because they were never mentored in mentoring. Break the vicious cycle and make a virtuous cycle. Be the best mentor you can be, if it's not good enough, stop.

Thus, when you see people veer off task, try first to understand why. Do they not understand? Do you not understand? Are they avoiding hard work? Are they chasing a shortcut that you know isn't there? Once you understand them, work on getting them to understand you. Walk them through your own learning processes. Walk them through the learning processes that you've seen others go through. Initial sessions will feel painfully slow because you aren't teaching your topic yet; you're teaching trust and also learning to trust them. Build trust, so you're both comfortable making mistakes in front of one another. Start by admitting your mistakes as you make them and freely admit ignorance. Talk about past mistakes, major and minor as you feel comfortable, so your students will be both comfortable with you and comfortable with themselves about admitting their own mistakes.

As you go through this process, let them question you; it wasn't that long ago that you were Learning. Encourage them to question you, both by asking them, but also when they do ask a question. Don't only say encouraging words like "That's a good question." Also briefly answer the question, and if they want to know more, say more. If your body language skills are poor and you can't tell without asking them, admit that, and ask them directly. Be careful of both the content of answers, but also style and manner. If you habitually answer questions in ways that aren't useful or bother them, your students will habitually stop asking questions and then they will Learn less or might stop Learning. If their questions lead you in a direction you've not yet explored, admit your ignorance, and explore it together. This closes the loop between Learning and Teaching; and once the loop is closed, the faster you can cycle through the Learn/Do/Teach cycle. The faster you cycle, the faster your entire community can learn and, eventually, begin teaching on their own.

This is the force multiplier: There is no end to your journey, because there is no end to learning, doing, and teaching. Instead, as an entire community pulls together, it learns together and begins working together, and begins to out compete slower communities. Review the teaching roles that are common in the industry, and think about more than what you can get out of it. Think about what you can give back to your community and how you can make us all better.

CHAPTER

TIER 3—PEN TEST LEAD 3.1

INTRODUCTION

Penetration testing was discussed earlier, and as there is no need to reiterate what has already been said, this section will focus specifically on the leadership role. The lead on a pen test team is both a contributor and manager. In addition to using your skills to help assess systems, you will have to prioritize the work of others and be front and center if something goes wrong during the test.

In some ways, it's just like being a regular penetration tester. You have to understand systems and networks, stay on top of the latest attacks, vulnerabilities, and tools, run tools, analyze output, and write understandable reports. However, in this role you become the face of the team to the client. You will be the one they contact to scope the test, to report oddities, and to question the report. You will have to mediate disputes between your own team members and help to foster understanding between the client and your team.

In short, the jump from penetration tester to penetration testing lead is much like the jump from one who does system or network administration to one who manages a team of those people. You will need technical skill so your people respect you, business knowledge so your clients respect you, and soft skills so both will understand you and you will understand them, and so everyone will understand and do what is needed.

HOW TO BREAK IN

Breaking in to this role requires that you be able to demonstrate both technical and business skills. This can often happen as you are testing and eliminating false positives. The better you get at identifying what systems exist on the network, how they support the business, and how to better protect them, the more easily you will be able to explain this to others. You may find yourself taking a more public role, meeting with clients and giving presentations.

You can also break in at a technical level if you show aptitude for learning the business skills. Often, the most technical person on a penetration testing team winds up as the lead because they can quickly identify and resolve aspects of the test that are blocking others on the team. This is a useful skill, but without external communication skills, the team will only have half a leader. If you are in such a role and want to eventually convert this role into another, you should focus on your business and soft skills.

COMMON PATHS

The most common path, by far, is promotion from within. One penetration tester who is better/faster/calmer than others on the team will often rise to become a leader. The best way to do this is to work with your manager to identify where you lack skills, and develop them as best you can. You may need to transfer between teams or companies to make the final move as with any management position.

However, if you have shown decent technical skill in other roles, such as security assessment or vulnerability management, and have a high level of skill in communicating with non-technical people, as well as a decent business understanding, you may find opportunities to jump over the basic penetration testing role and go straight into the lead.

HOW TO IMPROVE SKILLS—YOURS AND OTHERS

This is a management role, so improving the skills of others is an important component to the work. This involves first understanding where they are strong and where they are weak. Then, you need to identify which areas of weakness actually matter to the team. For example, if you have a tester who is great at networking but completely lost when it comes to HTTP-based attacks, it only matters if you don't have someone else on the team to cover that area of work.

Outside of the technical, learning to manage personality conflicts and helping people to better communicate their desires and frustrations with their teammates will go a long way to helping them grow.

This is very interesting work if you like working with people. If you don't much like people, it's probably not the job for you. Network with your peers. There aren't a lot of people doing penetration testing, and the attackers are better at working together than we are. To keep up, you have to share your cool tricks in hopes of getting some back in turn. Be prepared to argue about scope. Many people artificially reduce scope to make it easier to pass. As a lead, it is your job to tell them when the scope is no longer sufficient to provide legitimate business value.

Finally, you must always keep in mind that their most likely avenue for personal growth may be to take over your job. On one hand, you don't want to stifle your team, but on the other, you don't want to lose your job. Managing this issue can be quite challenging. However, one thing you may wish to consider is to designate a different team member as a "test lead" for a single penetration test so they get to experience running a team and decide whether or not they want that responsibility. If you can let your competitors leave for jobs like yours, with your blessing, and keep those who realize they don't like that sort of work, you can wind up in a very stable job with a team of people who trust you implicitly because you gave them a chance when they wanted it.

RECOGNIZING WHEN YOU'RE STUCK

You can get stuck as a penetration testing lead in several ways. You can get stuck within a company that has no role above "lead" in your area. You can get stuck with a bad team. You can get stuck with bad clients.

If your company leaves you no advancement opportunities, your best bet is to try to find another job elsewhere. You can take a penetration testing lead job at another company, but your skills may also translate into doing auditing, architect work, or as a more general manager for information security or IT.

If you find yourself stuck with a bad team and you want to keep working there, you either have to start shifting those people out or change their attitudes. This can be done by firing them, giving them unpleasant tasks in the hopes they leave, or finding and addressing their concerns. The former is the easier, but more expensive method as you have to put up with getting a lower quality of work for what the organization is paying on top of having to pay for severance and unemployment insurance. The latter is difficult, and can also be quite expensive, both in time and money. A lot of leaders just choose to suffer through. The middle path, trying to force people out by giving them unpleasant work can work, but can also create a team based on bullying and punishment and the ill effects may last well after the bad people have left. Before you select any of these options, think critically about how bad the work will be if you choose to tolerate the situation, and see if you can justify it.

Finally, if you get stuck with bad clients, you have to educate your sales team (if you're selling the service) or the business unit managers (if you're performing testing internally). Most commonly, people react poorly to penetration testing because they think of the testing team as the enemy. If you approach them from a position of asking for help instead of telling them what to do, their attitudes may well change.

ROLE AT A GLANCE—PENETRATION TESTING LEAD

Table 3.1 Role at a Glance—Penetration Testing Lead

Hours	**Travel**	**Stress Level**	**Creativity**	**Flexibility**	**Stability**
8–10 hours/day	Low to Moderate	Moderate	Moderate	High	High
General job duties	As with the Pen Tester role has most of the same tasks of performing scans, identifying vulnerabilities, performing social engineering and generating post-test reports. But common tasks are more likely to be delegated to Pen Testers, while the experienced Penetration Testing Lead works with and presents to management and may do more advanced tasks such as create new exploits, research, and presenting at conferences. Mentorship and teaching are part of being a team lead.				
Learning	There is a fair amount of learning in this role, but there will be some rote. A lot of tests will involve similar setup and reconnaissance activities. You will have to stay on top of changes in the industry, but as a leader, you must also trust your people to stay on top of their areas. There may be less technical learning in this role than as a regular penetration tester. However, it would be wise to replace that lost learning with a focus on practicing soft skills and learning business theories.				
Advancement	There are often no direct advancement opportunities for a penetration testing lead. You may choose to move laterally into Security Management or an advanced level of assessment or auditing.				

CHAPTER

3.2 TIER 3—SECURITY ARCHITECT

INTRODUCTION

> “If builders built buildings the way programmers wrote programs, then the first woodpecker that came along would destroy civilization.”
>
> **— Gerald Weinberg**

The Security Architect role is broad and may have other names, such as Lead Engineer, Senior Engineer, Chief Systems Programmer, or Computer Scientist. In some companies, “Architect” is simply a more senior engineer or systems analyst.

Architecture is about the details at a high level. To use the building architect metaphor, architects don’t individually place the bricks and mortar; instead, they need to communicate to the bricklayers exactly what bricks to use, how to place them, what, how much, and where the mortar goes, how it is faced. If they don’t communicate this information, they need to understand exactly what the bricklayers will do by default. The goal is to achieve an end result that is not just pleasing, but will also work and stand the test of time.

Architects may have a mix of technical, managerial, and political duties. Although all roles typically have at least some political aspects, the architect role often negotiates with technical and non-technical peers, both inside and outside of information security. An architect may be a trusted advisor to non-technical staff.

Security Architects research, specify, design, and document security architectures. They may also be directly involved with implementation, quality assurance, testing, and user acceptance. To do this, Security Architects usually work closely with information technology architects, engineers, management, users, and vendors.

Security architects have to see the big picture: not just how the system can work, should work, and will work, but also how it will interact with all of the other existing and future systems that haven’t even been conceived of yet. The ideal security architect is beyond Subject Matter Expert, and expert in all things. Most security architects must specialize or rely upon others with subject matter expertise. Either path requires superior communications skills and ability to work with a team.

Some information technology organizations use the “architect” role as a top-level technical but non-managerial position. Other organizations use “architect” as the highest purely technical position, with the highest technical and managerial position being a C-level position such as CIO, or CISO, who may also be the senior information security architect, in duties if not title. However, C-level positions tend to be much more managerial and political in nature and may not be deeply technical. They may heavily rely upon technical architects, engineers, and subject matter experts.

In most organizations, the architect position is a full-time job, but not necessarily full-time information security. Particularly large organizations will have teams of architects, and perhaps teams of dedicated security architects. In very well-developed security organizations, there will be teams of specialist security architects who work closely with other specialist architects and specialist engineers for particular security controls in operating systems, software engineering, networking, and databases.

The information security architect is usually directly involved in the requirements, specifications, design, architecture, and high-level implementation of information security systems.

HOW TO BREAK IN

A senior engineer can break into security architecture through simple promotion within the same organization, or by joining another organization as an architect. However, architecture is different from engineering in that engineers solve a problem, but architects set and define the problem and how it could be solved. An engineer may solve the problem for the short term, but an architect solves the problem for the long term.

Smaller firms may have no architects, but that doesn't mean they don't need architecture. The senior engineer, who in small organizations is the sole engineer, must also be an architect if solutions are to last longer than the engineer's employment.

Software engineers often wonder if their solutions are still in use after they leave an organization, but software architects know that their solutions must still be in use; otherwise they've failed and the organization will have lost time and money creating a new solution.

HOW TO IMPROVE YOUR SKILLS

Communication skills for this role include lots of reading: very deep technical documents, high-level but still technical documents, business plans, HR policy, information security policy, and standards documents. So to improve yourself, you need to dedicate a certain amount of your time to just learning what others are doing.

Architects need strong and clear communications skills, and enough political savvy and social awareness to know when and what not to communicate. Social awareness and diversity training are particularly useful for architects because of the wide range of organizations where they will work.

Both formal and informal writing skills are critical. A poorly written informal email can cause immediate political issues and, if not caught, long-term technical issues, followed by more political issues. Architects may write both informal and formal short- and long-form documents. Some organizations require lengthy architectural documentation.

Speaking skills are important, both informally in small meetings, and also in formal settings such as all-hands organizational meetings with a set agenda, time limits, and presentation slides. Public speaking, and organizations such as Toastmasters and the affiliate Techmasters "Toastmasters for geeks" groups are one way to do all three of Learn/Do/Teach. See also the Boosting section of this book.

Different people learn and comprehend complex subjects in different ways. Although writing is critical, even poor diagraming, drawing, and presentation material skills can be useful. Good skills in this area will make you a better communicator, and being an artist will make you stand out.

RECOGNIZING WHEN YOU'RE STUCK

Architects who yearn to write code, wire up a server cluster, or break into systems may decide they would rather be a Senior Engineer in software development, systems administration, and penetration testing.

If an architect would prefer to be coordinating and planning than researching and designing, perhaps project management would be an appropriate career path.

HOW TO GET OUT

There are many paths for an architect to move on and still stay an architect. Security architects tend to be generalists, and could move into any of the other technical areas; however, there is plenty of room for specialization in larger organizations, or for the independent consultant or entrepreneur.

Some organizations provide a technical manager path for architects where they still do architecture but also are managing junior architects and engineers.

CRITICAL WARNINGS

Engineers with obsolete skills but are too senior to fire are sometimes promoted to architect. Even when this is not the case, an architect's hands-on skills can quickly become obsolete because engineers and technicians do that work. Some organizations require that for separation of duties architects do no hands-on work. Architects risk becoming irrelevant unless they keep up with and use current technology. If your organization can't or won't provide hands-on experience or training, then you'll need to do this on your own time. This may also be an opportunity to train for your next job beyond architect or into a specialization.

Table 3.2 Role at a Glance—Security Architect

Hours	**Travel**	**Stress Level**	**Creativity**	**Flexibility**	**Stability**
8+ hours/day	Moderate to High	Moderate to High	High	High	High
General job duties	Technical analysis, reading and sometimes summarizing papers, presentations, writing papers, preparing for presentations, advocacy, evangelism.				
Learning	High				
Advancement	Medium				

CHAPTER

TIER 3—LEAD AUDITOR

3.3

INTRODUCTION

> "He who knows does not speak; he who speaks does not know."
>
> **— Laozi**

The Lead Auditor is a bridge between technical and non-technical, between management and non-management. Lead Auditor is a full-time position and will usually be a formal or informal leader of a dedicated audit team. As with the Auditor position described in Tier 2, there are internal and external Lead Auditors. The role of "Lead" varies considerably. In some organizations, Lead is a formal management title and role at the start of the management track, and may have formal direct reports. In other organizations, Lead is not formally part of management, but is a promotion to a more senior position. Still other organizations use Lead as an informal title that isn't technically a promotion and has no recognition outside of the immediate management. The Lead position may even be totally informal and merely be by consensus or simple seniority.

The Lead Auditor may have the authority to decide what is being audited against or, if this is determined elsewhere, to what detail and depth the auditing is done. The Lead Auditor may also be the primary author, editor, and approver of the final audit report.

HOW TO BREAK IN

Becoming a Lead Auditor can range from as simple as just hanging around long enough to gain seniority, or it may require substantial investment in time and effort to get into the management track. A Lead may even simply be the only person in that role. However, a fully-functioning Lead Auditor needs good communications, management, and technical skills. Auditors do not need coding skills, but if they are auditing a development environment, they do need to know enough to recognize bad environments and poor excuses. A skilled and experienced auditor in a large organization with decentralized auditing may be able to transfer into a Lead Auditor position in another part of the organization, or get a new job as Lead Auditor in another organization. Certifications may be useful, or even mandatory (official or not), so find out what certifications the Lead Auditors you know have. Propose certification study groups to gauge interest. If there is no interest from your peers or management, perhaps that certification isn't useful in that organization, but it may be useful elsewhere.

HOW TO IMPROVE YOUR SKILLS

Auditors who have information technology backgrounds may be missing the communications skills and knowledge needed by a Lead Auditor. Other roles that include writing can be a good background, such Trainer-Educator. See also the Boosting Author chapter. If you aren't completely familiar with your particular industry's auditing requirements, learn them. Prepare for jobs in other industries by learning those industries' auditing requirements as well. Although not always a requirement, ISO 27000 is a common knowledge base.

Auditors from outside of information technology, such as in accounting, finance, and business, can grow through technical Learn/Do/Teach by studying or transferring into information technology roles, but not usually the lowest Tier 1 roles such as Help Desk and Log Reviewer. The Security Assessment and Risk Assessment roles are particularly well-suited for study or transfer before the Lead Auditor role. Some Lead Auditor roles require highly industry-specific training and qualifications, such as the Payment Card Industry—Data Security Standards (PCI-DSS QSA) which is common in financial services and retail. Although not always expected, some Lead Auditors may need to personally present audit reports, so public speaking may be useful, also present in Trainer-Educator, and in Boosting—Speaker.

RECOGNIZING WHEN YOU'RE STUCK

Auditing is a crucial part of any organization's success. If your organization doesn't learn from and improve with your best auditing practices, even as the Lead Auditor, it's time to leave. Becoming a jaded, cynical Lead Auditor isn't going to help the organization and it's not going to help your career; get out before that happens.

HOW TO GET OUT

A successful Lead Auditor will have strong communications and management skills, and at least some technical skills. The Security Assessment and Risk Assessment roles can be good lateral moves. A particularly technically proficient Lead Auditor may be able to move into Vulnerability Management, and possibly Pen Testing.

Table 3.3 Role at a Glance—Lead Auditor

Hours	Travel	Stress Level	Creativity	Flexibility	Stability
8–10 hours/day after hours	Varies	Moderate	Low	Low	High
General job duties	Review standards, set specific requirements. Manage/supervise other auditors. Write reports, present summaries. Run Audits As with consultants in general, may have high load and travel. As with auditors in general, auditors can be exposed to the stress of those being audited.				
Learning	Many opportunities especially for external auditors or internal auditors in large organizations.				
Advancement	Can be a step towards full management positions in infosec and elsewhere.				

CHAPTER

TIER 3—LEAD SECURITY-RISK ASSESSOR 3.4

INTRODUCTION

The Lead Security-Risk Assessor role is that of management of the combination of Security Assessment and Risk Assessment. It may actually be two separate roles or, in smaller organizations, a single role that has separate or combined security and risk assessment components. This role also is the interface between technical skills of the risk assessors and those whose systems and environments are being assessed. See the separate Tier 2 chapters of Risk Assessment and Security Assessment for specific details.

Both as a lead and as a bridge between roles, communications skills are very important. Technical skills are also very important, both to properly understand the technicalities of the security as it exists, and how to assess the component risks of the larger problem. Although these may appear to be very similar, or even identical, tasks, understanding this difference is a critical aspect of understanding this role. Security weaknesses, vulnerabilities, threats, and attacks are different from, and can even have some independence from, the actual risk. For an extreme example, consider a weak, vulnerable system under attack. It may actually have very low risk, because the consequences of the system being compromised are very low. Also, you cannot view all compromises as the same. Some systems, such as honeynets, are designed to be compromised so the defenders can learn from what the attackers do. An assessor that cannot see the subtleties in how organizations use their systems is not useful to their clients.

Those who perform this role know where the bodies are buried, or at least who's most likely to be buried next. Their job is to keep it from happening.

HOW TO BREAK IN

The Lead Security-Risk Assessor may be a peer or possibly more senior position from Security Architect, Lead Auditor, and Pen Test Lead. Each can be a leading step. Technical, management, and communications skills are all very important, but most especially critical is an understanding of the organization and the industry of which it is a part. At this level, unlike most others, comparisons and contrasts with competitors are likely to be important. The particular risk of an occurrence may be less important than risk comparison—the likelihood it will happen first or more often to your organization than to your organization's competitors. Experience at a competing or very similar organization can be extremely useful.

HOW TO IMPROVE YOUR SKILLS

At this level, communications—publishing—and political boosting become particularly important. Writing just one published book or pamphlet, having even a semi-successful business, or being a well-known industry speaker, even if only locally, can be a huge difference. Evangelizing the right area inside or outside the organization brings knowledge, and others' awareness of that knowledge. Leadership community contributions provide political experience and social skills building.

RECOGNIZING WHEN YOU'RE STUCK

If you're not making a difference in the quality of information security in your organization at this level; if your recommendations aren't at least sometimes heeded; if your internal mentoring rightly leads people away from the organization, never to return; if you keep running into the same problems year after year—then it may be time for you to move on. Although information security has many recurring cycles, doing it in another organization and perhaps another industry may be the way to growth, or at least to avoid stagnation.

HOW TO GET OUT

The Lead Security-Risk Assessor position is a step into the management track; however, it can also lead to peer roles such as Lead Auditor, Security Architecture, or, for the more technically minded, Pen Testing Lead, Tiger Team, or going independent as a Security Consultant. Try any of the Boosting categories you haven't done yet, or pursue those you did well at and enjoyed.

Table 3.4 Role at a Glance—Lead Security-Risk Assessor

Hours	**Travel**	**Stress Level**	**Creativity**	**Flexibility**	**Stability**
8–10 hours/day after hours	Varies	High	High	Moderate	High
General job duties	Security, audit, and risk document writing, editing, reviewing. Executive presentations and summaries.				
Learning	As a process choke-point and information funnel, there is much to learn.				
Advancement	A step towards management track, or to parallel technical and non-technical roles.				

CHAPTER

TIER 3—TIGER TEAM MEMBER—TIGER TEAM LEAD (RED TEAM) 3.5

INTRODUCTION

“Tiger Team,” sometimes also known as a “Red Team,” is a fancy name for the group of people who scramble into action. Typically, incident response teams are local to an organization and are both understaffed and underfunded. This all changes when you get to the world of tiger teams. A tiger team can be run as a service to other companies or run within a single, typically large, company.

These teams will be activated during a particularly nasty incident and will often be sent into the line of fire. Working on a tiger team will often involve working more than 24 hours at a time for a week or two.

On the positive side, you will be working with the best people in the world as you dig into network and forensics dumps, set traps, isolate attackers, and work to figure not only what happened and prevent it from happening again, but also (in some cases) how to retaliate. The work will be constantly new and challenging.

The negatives, however, are not inconsiderable. More than any other role, a tiger team works hard and long hours. Pulling an all-nighter is not just possible, but common. You may find yourself putting in a full week’s worth of work in a mere two days, and then have to keep working. Much as emergency room and military doctors sleep for two hours at a time, you may find yourself working in this mode as you trace issues and work with your team to figure out what’s going on. Stress will be extremely high, so if you don’t handle it well, this may not be the best choice for you. However, if you thrive in stressful, challenging environments, there’s little better.

HOW TO BREAK IN

These jobs usually open up when a new team is being developed or when an existing tiger team member has burned out and decided to go on to something else. Breaking into the role will require either being very convincing when creating a new team or being very aware of the status of teams you wish to join, so you can get on the hiring list when they replace roles or expand.

To identify potential groups with internal tiger teams and companies that focus on this sort of work, get involved in internal incident response groups. By helping the community, you become better known. Then, when opportunities arise, people will reach out to you.

As with security consulting, expect interesting and challenging work, but often, far too much of it. Being able to handle a constantly changing work environment is critical. This type of work is all crisis all the time. Develop the ability to identify which issues need to be handled immediately and which ones can wait. Learn the scientific method and to develop and test hypothesis very quickly.

Identifying and eliminating entire chains of thought can drastically reduce your levels of uncertainty. Readings in the hard sciences and scientific method can help you develop this skill. See the Appendix for references.

COMMON PATHS

You will not succeed as a tiger team member until you collect a lot of experience. You can go extremely deep into a specific field of study, such as penetration testing or incident response, or become a subject matter expert in something like networking or cryptography. Alternatively, you can try to become good, if not excellent, in multiple fields. The former approach makes you critical on the team when that phase of the investigation is ongoing, but it may become somewhat boring when your part is done. The latter approach may not allow you to contribute as much as others, but you will be able to contribute at a more modest level for a longer period of time during each incident.

RECOGNIZING WHEN YOU'RE STUCK

You'll never get stuck in this role. So long as attackers are given incentives to come up with new attacks (i.e., you have something worth taking), life will be a constant flow of new and interesting things to work on, either brought to you by clients, or found on your own as you do research. Instead, you may find yourself burning out.

A good tiger team will expect you to work hard when you're working, but also give you ample flex time to make up for it. Some groups require a minimum of 40 hours per week, and when you hit that number, if your specific skills are no longer required to contain the incident, you're off the rest of the week. This can result in some people working only two days per week, though three to four days per week is more common. Other organizations expect you to give your all for each week you work, but give you eight to twelve weeks of paid vacation to schedule your off hours. To avoid burning out, take advantage of your time off, and develop a non-computer and low-stress hobby, so your time off is actually relaxing.

If you find yourself burning out, ask first for a leave of absence. Typically, these sorts of jobs pay well enough that you should have sufficient savings to cover a month or two of unpaid vacation time. Rest, recover, and seriously consider whether you want to go back. If not, it shouldn't take long to leverage your tiger team experience into work as a consultant, manager, or security engineer.

WHEN OTHERS ARE STUCK

If you are mentoring or working with someone who is burning out in their role, advise them to do the same as if you were stuck yourself. They need a break, preferably a nice long one. Have them rest and take a bit of distance to really think things through. It takes about two weeks of not doing something to really start to get a sense of clarity. Make sure they take that time.

Table 3.5 Role at a Glance—Tiger Team Lead

<table>
<tr><th>Hours</th><th>Travel</th><th>Stress Level</th><th>Creativity</th><th>Flexibility</th><th>Stability</th></tr>
<tr><td>18–24 hours/day
More time off</td><td>Considerable</td><td>High</td><td>High</td><td>Low</td><td>Low</td></tr>
<tr><td><b>General job duties</b></td><td colspan="5">Nearly everything elsewhere, and more. Exceptions can include higher management, and long term planning such as architecture. Because of the secrecy of incident response even when there are other teams who would otherwise do some work, the Tiger Team may have to do it on their own, anyway.</td></tr>
<tr><td><b>Learning</b></td><td colspan="5">High—At the upper end of this role, you get to see a lot of interesting attacks and will spend a lot of time doing reverse engineering. However, in order to get to that point, you will likely have to go through a lot of “same old, same old” incidents. If you work with good people who only call the tiger team when absolutely needed, learning density will be extremely high. However, most people aren’t quite so careful, so expect to see a lot of the same thing over and over again as you ramp up within this role.
As this is a high-stress environment, you have to be sufficiently disciplined to cope with the immediate issue—contain the incident, prevent recurrence, and complete the documentation. Then, after everything is done, take the time to fill in your learning with true understanding. Consider duplicating the attack in a lab, writing up a public paper, or engaging in forensics against collected images. By closing the loop, you maximize the learning opportunities and increase your own effectiveness for future issues.</td></tr>
<tr><td><b>Advancement</b></td><td colspan="5">While tiger team members may advance to a leadership position, there are few advancement opportunities for a tiger team lead, as you’re already at the top.
That said, very few people can tolerate the extremely high stress levels of this role, and typically make a lateral move to security consultant, penetration testing lead, or security management after a few years. While this can be a very lucrative and engaging role for a long time, as your life’s priorities change, you may find that you’d rather make less money in exchange for having reliable time off, weekends, and a better work/life balance.</td></tr>
</table>

CHAPTER

3.6 TIER 3—SECURITY CONSULTANT

INTRODUCTION

> "A consultant is someone who borrows your watch to tell you the time, and then keeps the watch."
>
> **— Carl Ally**

Being an independent security consultant is a very interesting job. Basically, your role is to be the smart person in the room. However, to be successful, you must know when to lead others as the smart person and when to guide other people, allowing them to be the smart ones in front of their co-workers. Knowing the difference will be the key to success and, sadly, is often learned from failure.

This role is one that many people want, as it is perceived as lucrative and consisting of only the "good parts" of working in security. Most security consultants don't get involved in the day-to-day operations of a business and are sheltered from politics. However, people who want to effect change but can't because of the political sheltering may get frustrated.

The hidden aspects of the job also involve endless negotiation over different ways to solve problems, continual reassessment, and project lengths in years rather than days or months. Also, if you're an independent consultant, you must line up new projects while you work on existing ones, bill your own time, and sometimes work on many emergencies at the same time. The stress level as a security consultant can be higher than that of any other information security role.

CONSULTING VERSUS CONTRACTING

It should be mentioned that many people with the title of "security consultant" are actually functioning as security contractors. There is nothing wrong with such work, but if you are not weighing different courses of actions and interacting with both the people doing the work and those making the decisions, you are not actually consulting. A contracting role will be like any of the other roles described within this book, except that you will technically be employed by a contracting company rather than the organization for whom you are doing the work. This can, in some ways, be even more lucrative than true consulting. What it lacks, however, is the strategic component that consulting involves.

DEPENDENCY

Most consulting functions in one of two modes. Either you are employed by someone or you are running your own business. The fundamental difference here is whether you are the one finding new clients. If you are responsible for both finding and servicing new clients, you will have a lot more freedom to do what you want, in exchange for having more stress. If you are given projects to work on, you can have more confidence that there will be work when your project ends, but you often trade the ability to work on what you want and accept a somewhat lower salary to gain this level of job security.

Typically, most consultants work for quite a while within an existing consultancy before starting their own practice.

NOTE

Clients and Customers

In consulting terms, a "client" is the organization or person for whom you are providing services. A "customer" is someone for whom you provide goods. Typically, as a consultant, you would work for clients, each of which would have customers. Some of them may have clients as well, but generally, if you are working in this role, you would never have a customer.

THE MONEY

We have not focused much on income in this book. That's because, although we like to get paid, we do this because we love information security. We're not in information security for the money, but we will do it for money. Frankly speaking, at the time of this writing, almost any job in security will pay better than most other information technology options for whatever level you happen to be at. An exception is made for this role. While salary surveys are unreliable to begin with, they are completely useless for consultants. Not only do regular contractors get lumped into this category, but so do business owners. One person may have a job title of "consultant" and be making USD $40,000 per year in 02015 doing contracted help desk work. Another may have the same title and oversee a team of 20 other consultants, as well as doing their own work, for a total take-home of over a million US dollars yearly.

If you are driven largely by a desire for income instead of quality of work or quality of life, you are unlikely to be successful in this role. To truly succeed as a consultant, you must both care for your client and be able to help them find the right solution—for them, not for you. Greedy consultants often drop out of the business because clients talk to one another and word will spread.

HOW TO BREAK IN

There is a saying in consulting: "Fake it 'til you make it." Distressingly, this is true. The only way to succeed at being a consultant is to be a consultant. You will need some experience to maximize your chances of "faking it," but at some point, you must be willing to step into an unfamiliar situation and confidently suggest solutions that you are not certain will work. Critical skills in this role involve not only being able to understand and simplify highly complex situations and environments, but also being able to rapidly understand when things are not going as well as they ought and being able to make on-the-fly corrections.

The most stringent requirement for a security consultant is to be able to project a sense of trustworthiness. The second most stringent requirement is being able to follow through. This means years of having, and losing, fights that you don't really need to be having. After all, until you've lost some fights, you can't really develop the intuition for which fights are and are not worth having.

Instead of the classic advice "pick your battles," the path to becoming a security consultant is "pick every battle." The security consultant is a mercenary, and mercenaries are hired to win wars. Only through fighting and losing battles will you get good enough to fight for your clients and develop the skills to fight and win wars. This will, obviously, result in some uncomfortable situations.

COMMON PATHS

Common paths toward becoming a security consultant involve having done work as a Penetration Tester, Security Assessor, or Risk Assessor and grown tired of providing only tactical guidance. However, before the jump can be made, you may need to develop your presentation skills and move from having discussions with the IT-level people to presenting to the CEO, CIO, and CFO—and, in some cases, the board of directors. To do this, you may also want to spend some time as a security facilitator, project manager, trainer, or other role that focuses more on the softer skills than on traditional heavy technology. Typically, you will need an equal blend of hard and soft skills to succeed in this role.

LESS COMMON PATHS

Though less common, people do sometimes become security consultants after working as a lead auditor or security architect. It is rare, but academics and entrepreneurs can also become highly successful security consultants, though often in a specialist role. Specialized security consulting, such as ISO 27000 compliance, PCI-DSS QSA, or developing and implementing sand-boxing or intelligent analytics can be extremely lucrative. However, you will constantly be fighting people offering to do the work for less money and, as the service becomes a commodity, you will likely lose such work altogether.

HOW TO IMPROVE SKILLS—YOURS AND OTHERS

Unlike other jobs, this role involves a constant stream of learning opportunities. Every problem you solve will be a chance to grow. Moreover, since you will be functioning strategically, you will have constant opportunity to teach others. Whether you are helping a company become PCI-DSS compliant that runs from the lowest to the highest levels, implementing a Security Operations Center (SOC), or just reviewing and redrafting its security policies, you will have to learn a lot about your project and then convert that knowledge into a form your clients can use.

You will be writing, presenting, talking, creating graphics, writing code, reviewing deep technical internals, and monitoring the industry as a whole. More than any other role in information security, consulting will involve something new every single day.

RECOGNIZING WHEN YOU’RE STUCK

> “A slow sort of country!” said the Queen. “Now, here, you see, it takes all the running you can do, to keep in the same place. If you want to get somewhere else, you must run at least twice as fast as that!”
>
> **— the Red Queen**

Unlike practically every other role in information security, you simply cannot get stuck as a security consultant. You can leave at any time, and you can change your practice whenever you like. In fact, to be successful, you must constantly reinvent yourself to keep pace with what the attackers are doing.

Instead, your risk is burnout. Change can come so quickly, and so much can rest on you, that you might not be able to take it. If you find yourself snapping at people, missing obvious issues and generally feeling tired all the time, it is time for a vacation. If you still feel that way when you get back, take another one. If, after that, you still can't handle the stress, it may be time to move into another role.

WHEN OTHERS ARE STUCK

In a teaching role, you must pay attention to the well-being of others. As a consultant, you may be working with a different team every day. This means you'll be in a good position to identify when people are experiencing stress and help address these issues. Commonly, you will work with operational people—keeping systems functional, helping to identify concerns, working through issues, and so on. However, you will report to management. This means that if the people you work with are experiencing personal problems, deserve a raise, need some help, need a break, or—in some cases—need to be fired, you can work with the organization to make these things happen.

Your loyalty must always be to the organization, even if you are working with friends. If it is in the best interest of the organization that someone you like, but who can't do the job, must be let go, that's what you must recommend. Similarly, if they're about to lose someone if they don't give them a raise, the organization's management needs to know that, even if the person would be likely to find better employment elsewhere. Communicate clearly and frequently to make your position understood by management, co-workers, and your friends.

RULES OF THUMB

- Extremely interesting and challenging work, but often, far too much of it.
- Resiliency is more important than being right. You will be wrong. You will fail. How you handle that situation will matter far more than the failure itself.
- Be sure to set aside enough time to stay on top of changes in the industry, so you can provide the best service and minimize the chances of being caught flat-footed by your clients.

Table 3.6 Role at a Glance—Security Consultant

Hours	**Travel**	**Stress Level**	**Creativity**	**Flexibility**	**Stability**
12–18 hours/day	Considerable	High	High	High	Moderate
Learning	Lots—more than any other role, there will be ample opportunities to learn. At times, this will be overwhelming. The trick will be managing the learning opportunities and balancing them against the work that must be done to get paid, the work that must be done to set up new jobs, and focusing enough on the boring stuff to make sure everything gets done properly. As you go through this process, be sure to increase all aspects of your learning, not just the stuff you like or you're best at. The two keys to being successful are: • Staying on top of changing technologies and environments, being the best you can. • Addressing your weakest areas to limit the amount of damage they cause.				
Advancement	There are no advancement opportunities for a security consultant. You're already at the top. You may choose to move laterally into Security Management or an advanced level of assessment or auditing. However, most people who find they can succeed as a consultant tend to stay in that role.				

CHAPTER

3.7 TIER 3—SECURITY MANAGEMENT (CSO, CISO, CPO)

INTRODUCTION

> "Organizations Get the IT They Deserve."
>
> **— Phillip J. Windley**

Odds are that you've already worked for enough managers to have an idea what management involves, but your understanding is probably incomplete. Most people don't realize until they become a manager just how much involves doing what previously they consider work. As a manager, you are responsible for making sure the people under you are happy and, at the same time, that the people to whom you report are also happy. However, as a security manager, you also must make sure that all of the people your systems and processes protect are kept protected. As hard as it is to serve two sets of people, it's extremely difficult to serve all three of these masters simultaneously.

On top of that, at the C-level, you are responsible to the investors or owners of the organization and, in some cases, personally responsible for legal requirements, the violation of which could result in jail time.

Though these roles have different names, ranging from Information Security Manager to Chief Security Officer (CSO) and Chief Privacy Officer (CPO), they all involve a split understanding of security and basic management principles. However, this means that you may often find yourself torn between what's right for a company's bottom line and what's needed to protect the data you store. This role involves not just fighting the bad guys, but also fighting your own people as you argue for long-term investment over short-term return.

HOW TO BREAK IN

A good way to break into this job is to become a team lead from another job. Management responsibilities tend to aggregate with experience. Thus, it's common to find yourself running a small incident response team or penetration testing team and then gradually take over security responsibilities from the IT Director or CIO. As time goes by, you may find yourself promoted to CISO or even CSO. The process can take a very long time, but can be accelerated in environments with high turnover.

In many cases, the CISO/CSO/CPO is viewed by the rest of management as an unpleasant necessity; people tend to fall out of the jobs either because they try to do too much too quickly or they do too little and suffer from unexpected security issues, or get blamed for the security issues they tried to prevent but were stopped. In these environments, having the role is like having a target painted on your back. The jobs are easy to get, because no one really wants them, but they're also easy to get fired from. Thus, a typical avenue into the role is to get a job at a high-turnover organization and then, when your tenure comes to an end, find one in a more stable environment.

HOW TO IMPROVE SKILLS—YOURS AND OTHERS

This could be a job you stay in for the rest of your career if you are willing to constantly improve your management skills. Read many business books, current business literature and journals, and possibly take management classes. Some books and classes are bad, but you'll need to know about them because other managers may rely on them. You will also need to learn more about finance to make the proper arguments for or against a purchase. Being able to "talk the talk" will be required for you to be successful when negotiating with board members and your management team.

However, that is not enough on its own. Many technical people will not respect a manager who can't understand what they're talking about. You will have to maintain your skill level to a certain extent while also avoiding improvement opportunities that do not benefit your organization. Being able to win the respect of those you manage even when you lack the technical skills will require understanding a lot more about human psychology and behavior so you can remove roadblocks and ensure that they feel appreciated without them taking advantage of you in the process.

RECOGNIZING WHEN YOU'RE STUCK

You can get stuck as a manager when there are no further prospects for promotion. Some managers get burned out at this stage or start job-hopping. However, if you truly enjoy management, you may not want to move anywhere else. You can keep improving your people skills and have a greater challenge than anything technical for the rest of your life.

If you find yourself disliking your job, seriously explore the possibility that you might lack the skills to properly deal with your bosses or subordinates. If that's the case, and you're up to the challenge, you may find your job improving as you improve your own ability to understand and respond to those issues you face that involve people.

RULES OF THUMB

1. Non-technical but challenging work if you can make the transition.
2. Never stop reading. There are more business books than you can read in your lifetime and many of them have very useful information for this role. Ask people what books they found useful. Those books that aren't directly useful to you will help you better understand those who do find them useful, or think they are.
3. Get good at fighting. A weak manager gets nothing done. A strong manager who only tells others what to do is equally ineffective. Know when and how to fight and when to negotiate to get what you need.

Table 3.7 Role at a Glance—Security Management

Hours	Travel	Stress Level	Creativity	Flexibility	Stability
8–10 hours/day	Low	Moderate	Low—Technical High—People	Moderate	Moderate
Learning	Most of the learning in this role involves how to better work with people. If you're used to working with technology, this may be a very different form of learning than what you're used to. Expect to read a lot of business books and learn from attending management seminars. If this sounds boring to you, this may not be the right role for you.				
Advancement	There are often no direct advancement opportunities from here unless you want to go higher in terms of management. Many people like to just stay at this level forever.				

CHAPTER

TIER 3—LATERAL: CPA 3.8

INTRODUCTION

What is Certified Public Accountant (CPA) doing in this list of information security roles?

It's a fair question. The answer is four simple letters—SSAE. The Statement on Standards for Attestation Engagements number 16, or SSAE-16, is a form of external audit that involves security and privacy analysis of a business. This is a more stringent form of the better-known SAS-70 reporting process. This is a standardized process for a company to be audited by an outside firm to produce a report for its customers. This is often done by commercial data centers and other hosting providers, but is becoming more common in general business-to-business service providers.

The catch? This work can only be done by CPAs.

HOW TO BREAK IN

You break in to work as a CPA by passing the CPA exam, after you meet the preliminary requirements. These vary by state, but in general, it can take a bit longer than a bachelor's degree. Odds are that if you're reading this book, you're really not that interested in transferring out of security into being a CPA.

HOW TO BREAK OUT

If you are already a CPA, however, you may find this an interesting opportunity to start moving into security work. Most of it will be on the auditing side of things, but it wouldn't be very hard to find a security or IT consulting firm that has a client who wants an inexpensive SSAE-16. You basically trade the cost of a full SSAE-16 engagement for one that covers your costs and allows you to learn. The more you do, the more experience you build, and you may find yourself moving towards less CPA-focused auditing and into compliance realms like PCI-DSS or US HIPAA. At that point, you are a basically an Auditor in the Doing phase of your career and the rest of this book applies just as it would to someone who started in information security.

DEALING WITH DIFFERENCES

You will experience many differences between information security and accounting work. In accounting, there is some uncertainty, but a lot of it can be resolved with some effort. In general, information security has a lot more uncertainty and many areas that simply cannot be fixed. You will have to become much more comfortable with uncertainty and grow to accept the fact that best practice doesn't always apply. This can be a difficult adjustment for people who like the relative certainty of numbers, but if you can make it, many more options will open up to you.

CHAPTER

3.a TIER 3—LATERAL: GENERAL MANAGEMENT

INTRODUCTION

A general management job will consist of everything detailed in the Security Management job, except that you won't just be dealing with security. As with the security-focused role, there will be a lot of people- and business-based learning opportunities that stretch your understanding of psychology and economics. There will, however, be even less technical need than in the security-focused role. You will, instead, be largely focused on keeping people's performance metrics high and making sure that projects move properly toward completion.

HOW TO BREAK IN

If you wish to move out of security into general management, it should be relatively easy if you've already done some work as a security manager. If all you have is a lead position, you will likely need more experience before anyone would seriously consider you for a management position. As with anyone trying to move from a minority position to a majority one, you will have to be significantly better than the other candidates to overcome the perceived risk of hiring a non-manager for a manager role.

Focus on maximizing your learning, and demonstrating what you've learned by teaching other managers how to better manage. This can be done formally through some type of MBA program (various options exist) or informally through a program like the one outlined in the book *The Personal MBA*. The more you can understand about how both people and businesses work, the more successful you will become.

HOW TO BREAK OUT

If you find yourself in a general management job and wish to move into information security, your challenge will be greater. As you should know by this point in the book, information security is weird. Your managerial skills will be useful, but you will also need to be able to evaluate risk and identify when people are feeding you incomplete information or active misinformation to get you to agree to their view of things. Hanlon's razor is useful: Don't ascribe to malice that which can be attributed to stupidity, but don't rule out malice. Also consider causes like misinformation and ignorance.

To succeed, you will need to improve your overall technical and risk management skills. Though we have avoided recommending such solutions elsewhere in this book, the use of "boot camp" classes will go a long way to help. Some security for managers courses are available and focus on expanding your skills to a point where you can talk about real issues in ways that will be understandable to everyone on your team. Beware of out of date, mono-cultural, and outright wrong course material. For example, if you start using the terms “hacker” or “cyber” in ways that don’t match your team’s use of those terms, you risk making yourself appear to them as actually stupid, rather than merely misinformed or culturally tone-deaf.

It may also be wise to get hands-on technical experience. In the Resources section, you will see lists of self-study systems and websites, some of them in a game-based format. Start by playing some of the games to understand what attackers do and what defenders face. Showing that you can, at least in a small way, walk the walk will go a long way toward getting you into the meritocracy of information security. If you have some technical knowledge but also understand your technical limitations then you can greatly impress your team, most of whom will not expect any technical understanding from a manager. However, don’t try to fake it, they may be able to immediately tell, and will not be impressed with lying, and even less so, if you fool them longer.

DEALING WITH DIFFERENCES

The biggest difference you will face between security management and general management is that security management is very methodical and process-driven until there’s an incident, and then it’s all about flexibility and thinking of different ways to address concerns. All of your regular business management skills will apply, and there will be little difference in terminology once you learn the basics. The challenge will be in helping your technical people understand the business drivers around cash flow and investment, and helping your non-technical people understand the technical risk of not taking action. Then, with every incident, you will need to readjust to the new reality and react appropriately.

CHAPTER

3.b TIER 3—LATERAL: TECHNICAL ARCHITECT

INTRODUCTION

You're an Information Technology Architect. You're experienced, you're skilled, you're a Subject Matter Expert, you've been there and done that. You didn't just get the T-shirt, you got all the T-shirts and then you wrote the book. Information security is your next frontier, and you want to go where you haven't gone before.

You've got security awareness. You learned the hard way and learned from others' mistakes. You know that information security is different; that it's not just about getting things done, it's about making sure that *attackers* don't get *their* things done.

You get it. Now you're ready to break into information security and hit the ground running.

If you believe all that, then you're also going to need humility.

This book mostly assumes that if you're going into information security, you already know the basics or are going to learn it elsewhere. Except this chapter.

Information technology is hard. Just getting something to work at all takes time, thought, and energy. But once it's working it is almost certainly not secure.

Information security is even harder than information technology, because making it work *securely* requires the original effort just to make it work and then much more. Making it securely work consistently is even more work. And if it it's not consistently secure, then it's not secure. Information security often follows the Perverse Pareto Principle, the first 20% of the problem is solved with the first 80% of the work, the remaining 80% of the problem is solved using another 80% of the work, for 160% total work. The work required adds to more than 100%, because the amount of work was incorrectly estimated.

The best information security people fail constantly. Major corporations both deliver and use defective information security products. Experienced CISOs tend to be those who got fired because their previous organization got broken into, usually publicly. The best cryptologists in the world are routinely defeated during their own presentations, or worse, a few years later, when people are depending on the security of their products. There are few successes and many failures in information security. The worst failures can't be covered up; although most organizations still won't talk openly about either the cause or ramification of the failure. The best successes often don't talk about it, either, to avoid attracting unwanted attention.

Information security is so hard that nearly everyone fails and keeps failing. Success is often just failing less—sometimes it's failing less than others, or failing just less enough that there's still some reason to keep going.

There are many places to learn more about this. Bruce Schneier's essay "Inside the Twisted Mind of the Security Professional" is a place to start. See the Appendix: People for a URL.

Information security is still a young field in business, even younger than information technology. Most organizations still haven't figured out how important information security is and where it should go, if anywhere. Some organizations put information security in with technology, but because of conflict of interest issues, others put it in finance under the CFO, or audit and compliance, or in with corporate security (those people with the guards and guns). Some organizations completely outsource their information security.

If you can't find an information security mentor, look for other information security people who are willing to talk to you. Information security has the same specialties as information technology, and more. Finding information security people, books, organizations, and other resources for your technology will help, but don't get hung up there, either, because thinking security is as important as knowing security. As a prospective security architect, it helps to have some understanding of all of security, not just a narrow specialization.

If you don't think your specialty has information security requirements there, look more closely. Common security practice is usually barely adequate in any specialty and even state-of-the-art systems usually have information security problems.

Part of the problem of information security is that the other specialties don't know much about security, and don't even think about it. As you learn more, you'll probably also find information security for your specialty is also inadequate; it may work but run counter to normal practices, or it may be expensive or extremely difficult to use. This is, unfortunately, completely normal in information security.

That's the bad news.

The good news is you've already got a career of experience and knowledge that's valuable in information security. Your ability to think outside the information security box can make the difference. You know how to communicate with those in your specialty; you understand them because you are them, for now. You may find that, as you learn more about information security, that your co-workers and colleagues don't understand information security. You may also find that information security doesn't understand your specialty, either. But *you* can be the bridge of that gap between information technology and information security.

HOW TO BREAK IN

As an established technical architect, use your existing contacts to get an information security mentor. You may need to move through a few degrees of separation to find your mentor. Don't get too hung up on your contacts' or your mentor's information security job titles. Even in the same organization, the information security people may have very different or even oddly inappropriate titles. A senior architect outside information security may merely be a business analyst in information security. Many organizations still don't have CISOs, so the top information security job might be "Computer Scientist" or as junior-sounding as Assistant Vice President.

A Technical Architect interested in moving into information security can be an ideal candidate. However, the nature of information security is often more adversarial, and may be thought of as negative thinking. The wealth of experience and technical knowledge a Technical Architect brings to information security is extremely valuable, but making use of this also requires an information security mindset that can be difficult to understand. In large enough organizations, the Technical Architect can join an existing information security group and have a peer group of experienced security architects to learn from and share knowledge. It also requires some amount of humility and flexibility to learn just how ignorant they were about security and how insecure their architectures are.

If your current organization is big enough, there will be an information security group that you can start learning from and perhaps plan a lateral promotion into. Larger organizations will have training and education opportunities for you that may require nothing more than signing up and showing up. Organizations with good lateral mobility will make this easy. However, other organizations may make it difficult to make this move, either in general, or because of isolation within the information security group. If it's too difficult, then making such a move could even be politically limiting within the organization, so you may have to move to another organization entirely; make this your backup plan.

Small organizations won't have an information security group. This is bad because you won't have anyone to learn from internally, but *you* can create, officially or not, that information security group. It's better if you can get management buy-in, but if you don't, you just need to become your organization's information security subject matter expert. This is a lot of work, but if your organization is so small that it doesn't already have such a group, either it's doing really badly, or there will be less to learn.

If you want to become a generalist Security Architect (see that chapter) then use the entire rest of this book to start to learn about information security groups, and their roles and responsibilities. However, you can also focus on information security in your specialty—this book is just the beginning.

In some ways you'll be starting from scratch, and in some ways it will be worse, because you'll have to unlearn some things and yet have to retain those same things, because you'll need them to keep relating to your specialty.

HOW TO IMPROVE YOUR SKILLS

If you are going the generalist information security route, all the skills in this book and beyond are applicable. However, if you're going to focus on your current specialty, the approach is dependent upon that field. You'll probably find a lot in the information security field that pertains to your field, and possibly some within the field itself. There might be a security certification, or a concentration available. Having certifications tend to be less useful to those with experience, but the study materials may still be useful as long as they aren't focused solely on passing the exams.

CRITICAL WARNINGS

If your current organization or specialty doesn't have security awareness, you may run into opposition. If you are not politically or socially aware, you may inadvertently limit your opportunities, possibly resulting in you having to leave your organization or even your specialty. Even without these difficulties, you may find it difficult to continue to relate to your specialty and your field's colleagues.

If you're already a Technical Architect and going into Security Architecture, you'll see many similarities in architectural concepts. However, Information Technology and Information Security are different enough that what passes for Security Architecture is very different from Technical Architecture. Because information security is even less mature than information technology, you may find that security architecture is similarly immature in your organization, and even the field.

As a Security Architect, you might find that what you do is mostly ignored and considered irrelevant, or you might find that what you're really doing is security engineering because there are no information security engineers in the organization, only information technology operators who don't really know security. You might end up doing much more hands-on work than you are used to, or want to do.

TERMINOLOGY

There are confusing and subtle differences in terminology between information technology and information security. For example, "token" has a lot of meanings in general; some information security-specific meanings are even more confusing. Information security lacks many common definitions; an ongoing argument is the difference among a weakness, a defect, an error, or a vulnerability. What "random" means to most people, or in information technology, is different from information security, and different again in cryptography. Common words like "clear," "erase," "reset," "sanitize," "scrub," "wipe," and—less commonly—"zeroize" can be synonymous or each can have similar but precisely different meanings. The information security intent of the phrase "security through obscurity" can trip up even long-time information technology professionals.

CHAPTER

3.c TIER 3—LATERAL: ENTREPRENEUR

INTRODUCTION

Odds are that you know what an entrepreneur is. If you've been in the security field for a while, you know that entrepreneurs have partially formed ideas and make people do a lot of work, pull lots of late hours, and typically run in the wrong direction most of the time. If you're an entrepreneur coming from outside of the information security space, you know that entrepreneurs are brilliant innovators who are seldom understood in their time and have to constantly juggle funding concerns with vision and planning to hit the market just right at just the right time.

Clearly, there may be communication issues when trying to be an entrepreneur in the information security field.

Fundamentally, the field is ripe for innovation and companies come and go (or are acquired) with rapidity. There is ample room for an entrepreneur to make lots of money in information security—or lose it all. It is a high-risk, high-reward industry at present, which is ironic, since security should lower risks.

HOW TO BREAK IN

If you want to function as an entrepreneur in information security, you will either be an information security practitioner who wants to launch a company, or you are already an entrepreneur and want to move into a more lucrative field.

If you are the former, odds are that you understand a core problem and have a good solution in mind. You may need funding to create a proof-of-concept device or application. You may already have that and need help finding customers. You may just have a rough idea that you want to be your own boss and nothing beyond that. In all of these cases, the key is to find a partner. Most business ventures fail; the best way to keep that from happening to you is to get someone you trust to help you out. Some people work best with small "mastermind" groups, staying completely independent. Others work best by signing over a percentage of the business to someone else and trusting them to handle the business side of things while you focus on security or technology. Still others go through a fundraising process and get investors who function as a partner.

Conversely, if you are non-technical or not a security expert and wish to become an entrepreneur in the field, the critical element is similar. You are going to need help. The field is rife with people who have good ideas but little business sense. If you can't figure out who is trustworthy and who is not, you could lose your investment very quickly. It is often best to put together a small advisory team to help you assess the suggestions you get from potential partners or employees. The people to pick for this advisory team should have experience both in the technical side of things and on the business side. Current and former consultants are often good choices, as are current and former security managers.

However, each entrepreneurial journey is different, so there is little we can offer you here. Instead, you should read some of the many books out there about starting businesses. Read more than one book and find the books and paths that fit you. It would be wise to go through the basics, as described in the Tier 1—Learn section, and get some level of knowledge of the space before you jump in, but once you're there, success will largely be about your ability to choose good people.

CHAPTER

3.c.1 TIER 3—LATERAL: ENTREPRENEUR—STORY

GREG SULLIVAN

Greg Sullivan was a double math/music major. His goal all along was to make enough money in computers to go back to making music for a living. In the 01980s and 01990s, he did business software.

> I remember our CTO for the company gave a talk to the whole company. He said, “Here’s the deal with email, don’t press send unless you are willing to stand on top of the building and shout the message with a megaphone. Once you press send you expose it to the Internet, where it’s available to anyone with the means and the intent.” That was a big eye-opener. I can remember where I was standing, I can remember where he was standing. I’ve never been able to let that go. I believed him.
>
> My biggest challenge has been convincing people to protect their data.

So Greg became an information technologist with a passion for security.

Part of his introduction to information security was the blatant willingness of kids with a sense of entitlement to not have to pay for content. He likens theft of copyrighted content to crashing into a convenience store and taking things off the shelf. He wanted to make a contribution to copy protection.

> The bad guys operate under business drivers and economic models like the rest of us do. If we don’t drive up the cost of their business then we leave ourselves vulnerable. The only thing our adversaries are constrained by is their imagination because we digitize everything.

Sullivan created a company that grew from an apartment-based business developing custom software to businesses. A decade later, he restructured to embrace the Internet and help his clients leverage that new technology. His company grew to several hundred employees focused on the insurance, banking and securities industries, with operations in the U.S., Netherlands and the United Kingdom. He sold that company and looked for other opportunities.

Today, Greg is the CEO of Global Velocity, a data security startup that focuses on data management. It’s smaller that his previous company, but it’s growing. As an entrepreneur, Greg constantly deals with new resource constraints, but gets to solve brand new problems every day.

CHAPTER

TIER 3—LATERAL: ACADEMIA 3.d

INTRODUCTION—HOW THIS APPLIES

> "Academic politics is the most vicious and bitter form of politics, because the stakes are so low."
>
> **— Wallace Stanley Sayre**

For this chapter, Academia refers to positions in accredited four-year colleges or universities. Positions at two-year technical colleges are closer to that described in the Trainer-Educator chapter. The two roles differ in both requirements and duties.

To start, let's set forth the various levels within academia. At the most basic level is an Instructor. This is someone who is usually hired for a very limited term to teach a single course, though sometimes more. This person will often be using someone else's curriculum, while adding small amounts from their own experience. This role is essentially the same as the Trainer-Educator. Because of its similarity to that role, and its distinction from the other levels, it will not be discussed here.

This brings us to a tenure track position within the college or university. These are full-time positions where individuals are part of some department. People with roles in information security are usually in the Computer Science Department, though other areas may be appropriate as well, depending upon the individual's skill set.

As a professor you will have responsibility for doing research and generally teaching one or more courses per semester, though some professors who have enough outside funding can exclusively do research. How academia prioritizes the two can be summed up with the phrase "publish or perish." Professors must publish some number of publications during their time as a professor, if not, they will not be able to continue in the field. Each scientific community has its own standards for publishing quantity and frequency. For those in tenure track-positions, but who have not achieved it, publishing is essential for receiving tenure. Unfortunately, this publishing expectation is often in conflict with both scientific rigor and their teaching responsibilities. Depending upon the expectations of the college, a professor may end up focusing on research at the expense of preparing for classes.

The challenge of not being able to spend time focusing on updating curriculum is much more drastic in information security than it is in other areas. As an extreme example if your Ancient Greek History curriculum is five years behind, some advances might be missed, but your students will walk out of your course with an otherwise complete and solid understanding of Ancient Greek History. If you are five years behind in your Information Security curriculum, your students will be at a severe disadvantage compared to those taught with up-to-date material. At least some, if not most of the material will be brand new in the past five years. The tools they know how to use may be no longer used anywhere else, the vulnerabilities they protect against may now be irrelevant, and new threats will be unknown to them.

To avoid teaching obsolete material you may even find it necessary to write your own course books, as cutting-edge information security researchers are nearly always too busy to create up-to-date teaching material.

Finally, in academia, there is an expectation of an advanced degree (generally a PhD) in the subject matter you are teaching. As a professional, degrees are helpful in some cases, but in other degrees may actually be viewed as negative due to a lack of hands-on experience within the field.

WHAT SKILLS THIS GIVES YOU

The benefits of teaching have been described in the Trainer-Educator role. For those with the research experience within computer science in general, and information security specifically, their knowledge of those areas can be significantly beyond anyone working professionally in the field, as their discoveries may not have real-world implications yet. Similarly, the skills necessary to do academic research are very similar to those for investigating security breaches or auditing: attention to detail, keeping of meticulous notes, problem solving, and understanding of security principles.

When doing security research in academia, you must be able to defend your research to people who have an in-depth understanding of security in general. Doing so develops your communications skills, both in analyzing the questions being asked of you, and being able to express your thoughts and results. Advanced research will be expected to extend the state of the art and of human knowledge in general.

WHAT SKILLS YOU MIGHT STILL NEED

The ability to handle unexpected and nonstandard situations may be challenging for those moving from academia to a corporate environment. While doing research in academia, you often need to break down an overarching theme into smaller, more manageable pieces, then take a single piece and focus exclusively on that challenge. Or there may a situation while doing research where you discover an interesting side branch that does not directly relate to your current research. In academia, you have the luxury of noting that on the side and then, if you have the time, coming back to that research component. In contrast, there are many times when dealing with security incidents you are not going to be able to ignore certain side paths in order to appropriately assess the situation. Being able to manage that growing challenge, either by yourself, or by bringing in additional resources, will be an important part of your corporate job.

HOW TO FRAME YOUR SKILLS

As an Academic, your research and communications skills will provide value within the corporate environment. Demonstrating an ability to solve problems and communicate solutions will be helpful in moving between the areas. If you can react quickly to changing situations, that will serve you well in moving into areas such as Incident Responder or Pen Tester.

DIFFERENCES BETWEEN WHERE YOU ARE AND INFORMATION SECURITY

CULTURE

Academia is often more focused on getting it right, as compared to the corporate environment of getting things done. Having a flaw in a research paper is entirely inappropriate, while having a less than idealized process may be perfectly acceptable. Being able to adjust to differences in expectations will be important for moves to or from the field. Additionally, while academia would like to view itself as pure meritocracy, the political components of working within the field can be challenging to those who were more used to focusing on the tasks of their job.

Information security often rewards rebellion and going off on your own more than academia will.

TERMINOLOGY

Academics tend to be more precise in their language use than most, forgoing slang terms that may be used as part of the common culture. That said, academic and corporate vocabularies are merging, as last year's hypothetical research paper has become this week's major security breach.

CHAPTER

BOOSTING 4.0

INTRODUCTION

“Boosting” is what we call the process by which you can gain skills outside of your regular job. Boosting isn’t required to land a good job in information security, but it can help a lot. The goal isn’t to demonstrate why you are the *best* choice for an opening, it’s to demonstrate why you are a *better* choice than your competition. If even one of your competitors has been spending time at home to improve their skills and you have not, they’ll have the edge. And in this industry, an edge is all it takes.

Such edges include writing and getting published, software development and getting distributed, evangelist/advocate, original research, public speaking, community involvement, and volunteering at conferences and user groups.

As you consider whether or not to engage in boosting, consider two related factors.

First, there is the issue of work/life balance. Families demand your time. Any time spent on advancing your skills in your off hours will take time away from those you love. This can be done in short project-based runs, where you invest four weeks to build a skill and then move back for family time. Some do it full time, and the family knows that two hours each day (often, while they sleep) you are learning. Others don’t do it at all. This is not a decision to be made lightly. Your decision may be to have less to offer at work than others. There’s nothing wrong with this. You have to be who you are and what your family needs.

The second issue is burnout. Some people have no trouble working 80 hours a week—60 hours at their regular job and 20 hours on their own projects. For others, just the required work hours are difficult. Additionally, different people need different amounts of sleep. Eight hours of sleep per night is considered ideal by many. But some people need six hours or fewer while others need nine hours or more. Health and stress may affect your sleep requirements. Before you decide to pursue a boosting path, consider where your limits are and build a plan with that in mind. Don’t lose sleep; it is a false economy, don’t plan to catch up on sleep later, sleep when you need to, including naps. If you are already short of sleep, do not lose more sleep; instead, find other places to borrow time from. But don’t skip on exercise. If you don’t get some sort of exercise already, you probably should; not exercising is another false economy.

If your plan requires just one hour of your day, but your day has no time available, something has to give. Most people can give up an hour of television or other entertainment, but if your plan requires you to give up an hour of sleep or exercise, you shouldn’t do it. Similarly, adding an hour sounds trivial, but if you’re already physically and mentally exhausted at the end of each day, this one hour may push you over the edge into burnout. A day has only 24 hours, so an hour is slightly more than 4% of a day. Consider having 4% less money to put into perspective the loss of an hour a day.

Consider also that if just 4% of a day makes the difference then you have a 4% margin or less for error, change, or anything else. You are at the edge of burnout, just 4% more effort of anything may be disaster. If your manager doesn’t see this, and doesn’t appear to care when you bring it up, you’ve got another reason to get a new job. Examine what at work and home is pushing you to your physical, mental, and time limits. You may need to spend more time on other things, such as more or better sleep, rest, mediation, exercise, or diet. Talk to your physician, counselor, or trusted friend.

The phrase “quality time” is trite but is useful. Gain time by being mindful of time spent everywhere; budgeting time is even more important than budgeting money, as time can’t be accumulated, nor is interest easily gained. Keep careful notes of where you spent your time so that at the end of the day you know what you did and how long it took. Use these metrics to find lost time and to plan better. Time can be gained from many places. You might find that every day you spend 5 minutes just looking for your wallet, phone, or keys. One way to gain that time back is to develop an every day carry (EDC) habit. You already have an EDC, unless you don’t regularly wear clothing or anything else so if you’re wasting time looking for anything every day, then you’ll save time with a good EDC habit. Your EDC are the things you always need or have close at hand. Your EDC is a way to be organized and prepared, so that when time is unexpectedly available or unavailable, it is not wasted.

Typical EDCs include clothing, wallet, bag, or other containers that organize, hold, and keep everything else from being lost. Include in your everyday carry a book or other learning material as well as writing materials. If this seems like too much to carry, the phone, tablet, and other computing devices already part of your EDC may do it already.

Mobile devices that function as eBook readers or music players, connect to social media, and provide Internet access are common, but can be expensive both in initial purchase and ongoing use. In addition, such tools can be their own distraction; instead a special-purpose eBook reader or music player, despite being less functional, may pay dividends in time by being easier to use, cheaper, and in particular being less of a distraction.

Time spent traveling may be usefully spent thinking about ongoing problems or listening to audio-books, conference recordings, or podcasts. Recordings can be played faster without pitch distortion with the right software, allowing you to get even more learning out of limited time. Time spent waiting may be spent reading or participating in social media. Some types of exercise also allow these learning activities. Many menial but necessary tasks can be done just as well as listening and sometimes even reading.

TIME MANAGEMENT

Procrastination

“People commonly use the word “procrastination” to describe what they do on the Internet. It seems to me too mild to describe what’s happening as merely not-doing-work. We don’t call it procrastination when someone gets drunk instead of working.” — Paul Graham, The Acceleration of Addictiveness

It used to be that being really productive required giving up television, but today that’s not sufficient. Both smart phones and the Internet have many distractions, and are all the more so because they are truly useful. Don’t allow your tools to be your undoing. Beware of “productivity porn” and even “EDC porn” in which researching and implementing new ways to be productive distracts, while allowing the delusion of being productive.

You may not know your limits without reaching them, perhaps many times. When you begin to feel overly stressed and you're not making the progress you need, take a week off and see if you feel better. If so, consider whether or not you're doing too much, and rework your plan to reevaluate your goals and process.

SEPARATE CYCLES

These options to boost your skills operate on a completely separate Learn/Do/Teach cycle. These cycles are unlikely to involve your employment, so they'll be smaller and faster. Learning in a group can be slow, and is highly dependent upon the group and the teacher if any. Faster learners will outpace the others and may grow dissatisfied and leave. However, group learning also is an opportunity to learn from others, especially the teacher. Organizations also provide additional structure to verify that when you're in a Doing phase, things are getting done properly. This is good from a verification perspective, but bad from a speed perspective. Finally, some organizations are actively hostile toward the Learning and Teaching phases. In their view, time spent learning or teaching is time spent not getting things done. Avoid these organizations unless you're a consultant and are willing to work under these conditions and charge them for their poor practices.

Thus, to maximize your effectiveness, think differently about your Learn/Do/Teach cycle speed at work and at home. You can often achieve more learning and a much faster pace on your own. However, you can also get misled much more easily. As you review your Boosting plan, be sure to double-check that you're not just learning but learning the right things.

EXPLORATIONS

Many times, you just don't know what you need to learn. This is particularly common before you get into your field of choice. If you're a penetration tester and want to become a team lead, odds are that you know where at least some of your faults are. However, if you're spending your day job doing workstation support and you really want to design firewalls, you may know some of the basics, but the gap between where you are and where you want to be is much wider, and wider gaps have many more hidden challenges.

Think of boosting as building bridges between the person you are and the person you want to be. In the physical world, bridging a broad valley is much more difficult than stepping over a small gap. As you build the bridge, you need anchor points and a plan and a straightforward path. Bigger and longer bridges across wider and deeper gaps will involve footings spread across the length of the bridge as well as exploration of the area below the bridge to discover unseen challenges. Some bridges may fail because the land where a footing went is sandy or marshy. Other footings fail because the land is too hard to drill into. Sometimes you'll move the bridge somewhere else. Sometimes you'll have to move many bridges.

The same is true for creating new paths for learning. Sometimes you just have to experiment so you can feel out the area in which you want to improve your learning. This can involve testing new operating systems, new tools, or visiting different user groups, and meeting new people. If you don't know precisely where you need to go, take a week or two and just explore. The time spent in this activity is far from lost. Instead, it will pay off many times in the time saved when you start working on real projects.

DISADVANTAGES OF BOOSTING

There are many advantages to boosting; however, it is not an always beneficial process. In addition to the potential waste of time if you misjudge the need for the skill you target, remember that the time you spent is time you cannot spend on other things. This may not mean much if the next pressing issue in your life is that you really want to get caught up on watching your DVD collection. However, if it means missing your child's winning goal in the hockey finals, giving up a chance to take on a leadership role in your church, or not being there when a dying parent needs you, then the cost is high.

Taking a less dramatic view, a narrow focus can disadvantage you by shrinking the diversity of your education. By focusing ever more narrowly on your career goals, you box yourself in. At the time this book is being written, there are many unemployed COBOL programmers; Java programmers may be next. This is a delicate trade-off to make, so approach it with the understanding that once you get where you want to be, shift your focus to improve your thinking and learning diversity. The more diversity you have, the better you will be able to see interesting patterns and also understand and communicate those patterns to others. Well-rounded security practitioners understand their field, but also have insights into other fields, both technical and not, such as current events, business, cooking, art, literature, economics, and history. These insights can prove critical in addressing a real-world attack.

Many of these paths involve doing public work. For many, this won't be a concern, but some people work within restrictive organizations, and being seen to publicly endorse a specific project, such as Ubuntu Linux, may have negative consequences. More often, organizations take a view of not caring what you do outside of work, so as long you don't get paid. Be sure to familiarize yourself with your employer's restrictions before you commit yourself to any of the ideas in this section.

Finally, there are some projects that are more likely to have negative repercussions than others. For example in open-source, both the OpenSSL cryptographic library and the Tor anonymization service provide a great deal of value to others. However, Tor is far more politicized. If you work for an organization dealing with sensitive data, such as a military organization, the discovery that you're working on what can be a data exfiltration tool could both harm your work and your Tor project effort much more than less politicized work.

Consider what others would think about your involvement in any particular project. If you think it is likely to work well for you, go for it. However, if there's some concern, you may wish to consult with your supervisor to get permission—in writing—first.

CHAPTER

BOOSTING—AUTHOR (BLOGS, MAGAZINES, BOOKS)

4.1

INTRODUCTION—WHAT THIS IS

Writer. Author. Blogger. All have the same general goal: generating content, in some form, that someone else sees as valuable. The choice of medium (electronic or paper) will impact the format of that content, but keeping a focus on the content is where being an Author will improve your career prospects.

As a book author, you will be responsible for generating content to a quantity, quality, and format dictated by the publisher. If you are coauthoring a book, you must be able to collaborate with the other authors in order to ensure a cohesive book.

For those that are more independent-minded, you can create a blog. Keep in mind that even if you have developed a following, it will take work to maintain and generate traffic to your blog. That said, it is your blog. You can do with it as you please subject to your country's laws. You get full credit for the content generated. You also get the full blame for a lack of quality, or worse, should you plagiarize others' material, the legal issues.

For those who are more community minded, forums such as StackExchange provide a way to contribute to the community by addressing specific questions. You may be able to talk about the points and scores you have in interviews. Community involvement shows an interest and ability to help others, important characteristics for many organizations. While many may not consider this actual authorship, many of the advantages of authorship are present here. Your material is available. You can point prospective employers to the site and identify who you are, and they can evaluate your writing and knowledge.

For those interested in boosting their career through authorship, that is what it is all about: making your excellent knowledge available for others to appreciate.

WHY YOU MIGHT WANT TO DEVOTE TIME TO THIS

All authorship can be used to generate your brand. How much that will help you will vary based on the content and reputation of the items authored. Getting authorship for a book or article is generally viewed more highly, as it indicates that some external entity (the publisher) viewed your work as valuable. The simple fact that someone was willing to pay you for your efforts will raise the perception of value, for those who have not read the material.

For those who have read the material, the value is in the content. If your content does not display an understanding of the topic, is plagiarized, or is filled with errors, the authorship could potentially hurt your prospects. Do what you can to generate high-quality material.

HOW THIS MIGHT COST YOU

Writing is a high-paying career for only a very few individuals. Some individuals can generate a site with which they can support themselves (Brian Krebs is a classic example), but this takes significant experience, knowledge, and effort. If you want a high hourly rate for your writing time, you are going to have to spend a significant amount of time working to develop a following. Even putting forth that effort has no guarantee of success.

Those who have achieved some level of success at writing will then be introduced to the trials and tribulations of deadlines. Once someone starts paying you for your efforts, there is an expectation that the work will be produced by the agreed-upon time. Since writing is unlikely to be your regular job, this will take away from your personal time. Many times the deadlines are critical to the publisher, so failure to meet them may reduce your opportunities with them in the future.

Finally, if you plagiarize others' materials, you can permanently damage your personal brand. With the search abilities of the Internet, you can assume that if you do so, sooner or later someone will find you out.

HOW TO GET STARTED

There are books and articles aplenty on how to become an author; find one that works for you. Focus on material that you know. Generate quality content.

If you can find someone who is working on a book and you have an opportunity to coauthor with them, take full advantage of that opportunity. Opportunities such as that do not present themselves very often.

WHEN YOU MIGHT WANT TO STOP

Writing regularly is hard. Taking a blank slate, and producing valuable content takes a level of dedication that can be hard to maintain. The number of people who are working on a book is many orders of magnitude larger than those who have completed a book. If the time and effort necessary for you to generate the content is too high, then it is time to exit.

It is far better to stop generating content than to generate content that is of such low quality that others do not find value in it and you hurt your brand.

WHAT SKILLS THIS GIVES YOU

The ability to write well is a valuable skill by itself. Being able to clearly present concepts in written form is useful for any information security role, whether it be technologist, manager, or organizer.

Beyond the writing skills, doing the research necessary to produce material will be valuable as well. Not only will you develop your research skills, but you will be learning more details about an area that interests you.

WHAT SKILLS YOU MIGHT STILL NEED

One of the biggest challenges for those who want to become authors is understanding the time and effort involved in producing quality material. The process of generating the content will often involve multiple revisions, or at least multiple reviews as you go through the material.

CHAPTER

BOOSTING—DEVELOPER (OPEN SOURCE)

4.2

INTRODUCTION—WHAT THIS IS

The job role of developer has already been discussed in the Developer chapter. However, for those who like to write code, or want to develop their coding skills, being a developer as part of an open-source project is a great way to build and maintain those skills.

Though you may be unfamiliar with what open source is, you probably use open-source software. The Linux kernel and Mozilla Firefox are two of the most widely used open-source software, while open source libraries such as Info-ZIP and OpenSSL are found in ubiquitous commercial software and hardware. This means that anyone is able to get a copy of the source code, and build or modify the code themselves. That said, there are frequently limitations put upon the changes that you make. Many open-source licenses require you to release your source code, should you wish to distribute anything based on their code. More information on open-source licenses and the idea of free software can be found at the Free Software Foundation's website.

There are literally thousands of active community open-source projects out there. And there are thousands of open-source projects that were started but are no longer being actively supported.

WHY YOU MIGHT WANT TO DEVOTE TIME TO THIS

Working on open-source software is a great way to develop your programming skills. If you are not already a professional developer, it provides an opportunity to learn about the development process. For those who are professional developers, it provides an opportunity to network and learn from a diverse peer group that you would not have access to otherwise.

For those in the security field, fixing security bugs is one of the best possible ways to understand the types of coding mistakes people make that lead to security vulnerabilities. By fixing bugs, you learn what attackers look for and you learn what their exploit code does to take advantage of the flaws in the system. This process will teach you multiple languages and multiple coding styles. If, in the future, you transition to penetration testing, you can take your skills and pivot to exploit development, writing the code to take advantage of the weaknesses your team finds.

Beyond the security-specific knowledge you may gain, working on projects you are interested in provides an opportunity to network with a wide range of people with similar interests. This may open doors to other career opportunities that you might not have otherwise.

Finally, providing enhancements to open-source projects is one way to truly demonstrate your skills to prospective employers. It is one more way to differentiate yourself from the other candidates.

HOW THIS MIGHT COST YOU

Working on open-source projects is guaranteed to cost you time: unpaid, no-glory time. If you are someone who enjoys the guts of code, that time may be time well spent, regardless of the outcome. There are relatively few paid open-source project jobs. They are usually commercial open source support companies but also a few funded open source organizations, and also large organizations that are heavily dependent upon open source. Some examples are the FSF, Google, Mozilla, and Red Hat. However, jobs that make some use of open-source are very common with few IT jobs not using open source somewhere. However, a few organizations still officially avoid and even ban open-source software to avoid perceived security open-source licensing issues.

Most open-source projects never catch on, get replaced by other projects, or fall apart due to bickering of the project team, just as most companies eventually go out of business. If this were to happen, there would be less value to your resume, but the experience you gain would be valuable no matter what happens to the project and no less than if your previous employers went out of business.

HOW TO GET STARTED

When working on an open-source project, there are three key pieces to consider: Project, Community, Technology.

First, are you interested in that project? If you love graphics programs, GIMP may be the perfect program for you, but if you couldn't care less about gray scale and transparency layers, it is unlikely to keep your attention. Focus on projects where you are excited about the finished project. That will help drive you to produce the highest quality code you can.

Second, is the community one you want to be part of? Every project has its own management style and community. Some projects are driven by a committee, while others have a single person who has final say with large numbers of people contributing to the code. Some projects are welcoming to inexperienced developers. Others have little patience for those who do not have a deep understanding of coding projects. If the community personality does not fit your personality, you are unlikely to be interested in staying engaged with the project.

Finally, there is technology. If you are interested in developing your PHP skills, working on the Linux kernel is not going to be a good project for you. Find a project that requires the skills you want to develop.

WHEN YOU MIGHT WANT TO STOP

Coding after hours can be draining. The idea of getting home from a long day at work staring at a computer, just to stare at another computer screen, may get to the point where it is no longer worthwhile. Sometimes the project's leadership or community may change to the point where it is no longer a group you are comfortable interacting with.

Stopping may mean leaving one project and moving on to another, or it may mean stopping doing development altogether.

WHAT SKILLS THIS GIVES YOU

At a minimum, your development skills will improve. Additionally, you may develop soft skills associated with working in a large diverse team. Potentially you will develop an understanding of security vulnerabilities within code.

WHAT SKILLS YOU MIGHT STILL NEED

Your development skills may or may not be to the level that are necessary for the project that you want in the role that you are interested in. Finding a role and project that matches your skill set may take some effort.

CHAPTER

4.3 BOOSTING—DEVELOPER/ ENTREPRENEUR (CLOSED OR OPEN SOURCE)

INTRODUCTION—WHAT THIS IS

> “One percent inspiration, 99 percent perspiration.”
>
> **— Thomas Edison**

You have the idea for the next Facebook, Google, Linux, or Flappy Bird. Well, maybe the next Flappy Bird. Time to start writing code. Writing in the morning. Writing in the evening. Writing in the lunch hour. You can be a devpreneur—a developer entrepreneur who creates a business with just software. However, as a booster, you have a day job, so make sure not to use business resources. Check your employer’s policy on doing your own work on your own time with your own equipment. Most companies have rights to anything you do on their time on their equipment, but some claim anything you thought of while you were employed, and you do not want to do all of that work for their benefit or risk legal issues should there be any questions.

There are periods of time when there are massive spikes in Devpreneurs. In the 01980s, the platform was the personal computer. In the mid-01990s, it was the Internet accessed with personal computers. Later in the 02010s, it was smart phone, tablet, and mobile Internet applications. Each time, a new market was created by the emergence of a new, ubiquitous platform. After the initial surge, the market began to saturate, and finding opportunities became harder and harder, but they still exist. New platforms and technologies continue to be developed, such as smart watches, smart glasses, augmented reality, full speech control, and more.

If you have a passion for creating, this can be a chance to enjoy that and learn, with the potential for self-employment.

WHY YOU MIGHT WANT TO DEVOTE TIME TO THIS

Owning and running an independent company has great appeal for many. This drives many to work on creating their product. No matter what the outcome, the process is going to be educational. You will learn about development, market forces, and coding. Expect to learn about software vulnerabilities, the public relations and marketing response to those vulnerabilities, having to fix bugs, and generating interim releases.

Do not underestimate the value of going through this process. Whether a success or failure, there is much to be learned in starting your own company.

HOW THIS MIGHT COST YOU

The largest commodity this process will cost you is time. Depending on the development platform, there may be additional costs for hardware, server hosting, tool and developer registration fees, or even legal costs, depending on what you are developing. Standard costs for a business may also apply such as Internet domain name registration, email, web site, dedicated phone number, PO Box, and business cards, if you know anyone who still uses those, etc.

If you move to exclusively focusing on developing and selling your product, you will frequently have to rely on savings while the product ramps up. Additionally, if it does not succeed, moving back into the job market may be challenging.

HOW TO GET STARTED

You might think you should start by writing code. Actually, that may not be the best starting point. Investigate the market, determine if the concept has already been developed, and determine approximately how much effort it will take to produce the product you want. A common used software engineering concept is the minimum viable product (MVP), or rephrased “What is the simplest thing that could possibly work?”

If you have a truly viable product idea, and the ability, you may even be able to get funding and support from an incubator company in exchange for sharing future profits.

Often someone else has tried the idea before. Sometimes, it is part of a suite offered by a major player. You will have to decide if you can build it better, or if it is better to not attempt to enter the market.

WHAT SKILLS THIS GIVES YOU

You will start by developing your coding skills. After that, you will develop your marketing skills. After that you will develop your business skills.

That is, of course, an idealized progression. Most of your new skills will be developed as you build your product. Technical people often miss the importance of marketing. You should be marketing yourself and your product early in the process.

WHAT SKILLS YOU MIGHT STILL NEED

You’ll need all of the skills above that you do not already have. If you are fortunate enough to have your business take off and grow, be prepared to move away from technology and into a more business-oriented role. A common strategy is to start the company with a partner who complements the other’s skills, talents, and interests. For example if you don’t want to go into the business side, then choose a partner who will. A classic example of this partner relationship is Steve Wozniak (software and hardware) who co-founded with Steve Jobs (marketing and business) what became Apple Inc.

CHAPTER

4.4 BOOSTING—EVANGELIST (SECURITY, PRIVACY)

INTRODUCTION—WHAT THIS IS

Evangelism has been around for as long as people have been able to communicate. I would not be surprised if different groups had advocated for different means of starting fires. When someone feels passionate about a subject, they will work to bring others to it, either individually, when meeting with others, or in group settings. Technology evangelists have been around for generations. In some cases, they end up getting hired by the company they are passionate about such as Guy Kawasaki at Apple, Inc., and in others, they create an organization and a technology platform around their beliefs such as Richard Stallman at the Free Software Foundation (FSF).

In modern computing, evangelism has spread to include the computer security, privacy, and legal arenas. The platforms for evangelism have also changed as social media and the Internet have allowed individuals to have a much broader reach.

WHY YOU MIGHT WANT TO DEVOTE TIME TO THIS

If you want to change the world, or some small part of it, you will need to evangelize to get others who in turn get others to work toward the goal you are trying to achieve.

HOW THIS MIGHT COST YOU

If there is a need to evangelize for your point of view, that means there are others who disagree with you. After all, if everyone agreed, why bother taking the effort to espouse your point? You will thus need to be ready to defend your position against others. Is conflict something you are comfortable with? If not, then sleepless nights are potentially in your future. Also, because successful evangelism results in your cause being taken up by others, you may get caught in blowback from what others do. For example, you are evangelizing for privacy and some rogue group defaces a website to also evangelize for privacy, you could get branded as “one of those privacy nuts” even if you did not participate in the action.

Of course, if you do anything illegal as part of your advocacy, there may be legal challenges for you as well.

HOW TO GET STARTED

Evangelism is built around the platform that you have. If you have a LinkedIn or Facebook account use it to express your views. However, consider the medium for your message. For example, if you're a privacy evangelist, consider the irony of using anti-privacy platforms. Others may be able to speak at user group meetings or similar groups to express their views.

Be careful about doing evangelism at work. Depending on your work rules, what you are evangelizing for, and the laws applicable to your organization, evangelism can be grounds for discipline at work.

As with the Speaker boosting role, the more exposure you get, the more likely you are to get a larger platform. It's always necessary to work to expand your social circle.

WHEN YOU MIGHT WANT TO STOP

If what you are evangelizing for happens (or evangelizing against has happened yet), then victory is yours. Congratulations. However, before that happens, you may become drained by the combativeness of the situation, which will induce you to leave the subject. You may also change your views. New developments may change whether what you are advocating for is still necessary.

WHAT SKILLS THIS GIVES YOU

Written or verbal expression of your views is the core of evangelism. You will need to communicate in a clear and convincing manner. These skills are useful in many career tracks.

WHAT SKILLS YOU MIGHT STILL NEED

A thick skin, the ability to not let others' words affect you, will be important in this role. As you grow, you may need to develop your speaking skills.

CHAPTER

4.5 BOOSTING—RESEARCHER (SECURITY, VULNERABILITY, ETC.)

INTRODUCTION—WHAT THIS IS

> "Research is what I'm doing when I don't know what I'm doing."
>
> **— Werner von Braun**

To extend the boundaries of security, people investigate them. There are many areas, from reverse-engineering malware to identifying vulnerabilities in web browsers.

You get to name the bugs you find, unless it's so serious that others give it more popular names Heartbleed and Shell Shock. You know you have found a serious bug if it winds up being named, either by the community or the media. If this happens, you may find yourself working some lucrative consulting contracts. That said, those are rare situations, as most research results in little to no financial reward. Some bugs have financial bug bounties offered by the organization responsible for that vulnerable system (see the Appendix), but most don't. Most rewards are going to be in a sense of self-fulfillment, and potentially in speaking opportunities.

WHY YOU MIGHT WANT TO DEVOTE TIME TO THIS

Doing research is one of the single best ways to truly understand a topic. The knowledge and experience required to become a good security researcher will set you apart from nearly everyone in the industry.

It can be immensely rewarding. Finding a security vulnerability that no one else has reported is something to be proud of. If it's sufficiently novel, you could be asked to do talks on your research, increasing your personal brand.

HOW THIS MIGHT COST YOU

This will cost you time. Lots of time. And if you are not careful to stay within legal research, you could face the ultimate time loss: prison.

As long as you stay legal and do not plagiarize, the risks are essentially in time spent. If you learn anything, the time is not wasted.

HOW TO GET STARTED

Whatever the area you are planning to move into, you must find it interesting. Whether it is website vulnerabilities, software vulnerabilities, cryptography, malware, or some other area, the work is going to be hard, at times frustrating, and occasionally grueling.

WARNING

Identify what is legal to do in your jurisdiction. Attempting to break into websites is usually illegal. Make sure you understand any website's policy regarding security research. You must be careful to never violate others' privacy. Doing so may expose you to criminal liability. Similarly, many software licenses do not allow reverse-engineering. Make sure you understand ethical vulnerability disclosure guidelines. There are many different views on this, and you will have to decide what you are comfortable with. But, once again, whatever you do, ensure that you are within the laws of your jurisdiction.

In short, do not do *anything* until you understand what you can and cannot legally do.

The next step in the research will be to develop your background knowledge in the subject. Take a look at previous research. Many conferences publicly post presentations that have been given, such as DEF CON and Black Hat. You can get at cutting-edge research by looking at the more recent presentations, and develop a historical perspective by looking at older presentations.

Start small. Develop a lab with the equipment and tools where you can experiment safely and legally and consistently. Having dedicated space where you can keep your equipment and tools visible and available is a part of time management. But don't spend a week at the lab to save a day at the library, research both the historical and current industry literature. Reproduce others' work in your lab. Use the scientific method. This will give you an understanding of the process, tools, and skills. See the Appendix for references.

Start building from there, then move into unexplored areas.

WHEN YOU MIGHT WANT TO STOP

Making money as a security researcher is hard. While bug bounties are sometimes offered, very few individuals are able to make a living at it. If you are not making money at it, your motivation is going to be your interest in the subject matter. Once you lose interest and are no longer keeping up with the new advances, it is probably time to move on.

WHAT SKILLS THIS GIVES YOU

After spending significant time as a researcher, you will have a great understanding of that facet of security. Beyond that area, the overlap and extension into other areas can move you on the path to being an expert in this field.

WHAT SKILLS YOU MIGHT STILL NEED

Describing the skill gaps is not really feasible, given the range of possible security research. In almost all cases, a strong understanding of software development will be required, but what that means will vary from area to area.

CHAPTER

4.6 BOOSTING—SPEAKER (LOCAL EVENTS, PODCASTS, WEBCASTS, ETC.)

INTRODUCTION—WHAT THIS IS

> "All the great speakers were bad speakers at first."
>
> **— Ralph Waldo Emerson**

Presenter. Speaker. Podcaster. All roles in which you are using your voice to transmit knowledge to someone else. With this booster, you have the opportunity to get out and share your hard-earned knowledge and experience with someone else, potentially with many others.

For those who prefer not to speak in front of people, at least to start, a recorded medium may be the best. Keep in mind that if you do well at podcasting, you will be asked to make live presentations. That additional exposure and opportunity to build your brand will be valuable, so think about doing so.

Whatever your medium, it will all be about the preparation. How much time you spend finding background material, and providing depth and breadth to your topic will impact the quality of your final product. Being able to provide a unique combination of fact, opinion, and speculation, making sure you identify which is which, will help make you an interesting speaker.

WHY YOU MIGHT WANT TO DEVOTE TIME TO THIS

Once you have presented well, additional opportunities to speak will often become available. Someone at a small conference saw you speak, and now they are looking for speakers at a large conference, and you come to mind. You can grow your brand and get additional opportunities this way. A good speaking engagement can provide new consulting or other career opportunities.

HOW THIS MIGHT COST YOU

Depending on your topic, and your format of choice, you could end up being branded in a certain category. If you are not comfortable being thought of as "the <x> person," where <x> is your topic of choice, speaking is probably not a good booster for you. Additionally, once you put yourself out there as a subject matter expert in a specific field, you are going to be scrutinized significantly. If you are not comfortable backing up your claims every time you present, speaking is probably not a good option.

HOW TO GET STARTED

As with all boosters, find a topic you are excited and passionate about. Speaking is all about presentation. It is easier to present well if you feel strongly about the subject. Learn proper citation techniques for the facts and information you use. Before you start understand how to give them appropriate credit.

Find small user groups. Smaller group meetings often have trouble finding speakers, and will be open to volunteer presenters. Do this as often as you can. This will help to build your brand, and potentially provide opportunities at larger engagements.

Put some speeches online. YouTube or other similar sites can provide additional exposure. They also provide a way for you to give an example of your work to those who might be interested in engaging you.

WHEN YOU MIGHT WANT TO STOP

As with any public figure, the more exposure you get, the greater the number of detractors you will have. If dealing with these detractors is no longer something you are willing to do, then lowering your profile by moving out of speaking may be the best option for you. If you are no longer able to find topics that interest you, it is often better to move to other areas, rather than lowering the overall quality of your speaking efforts.

WHAT SKILLS THIS GIVES YOU

Good speaking skills are valuable in any environment. Additionally, you will develop your networking skills, and potentially your debating skills as well.

WHAT SKILLS YOU MIGHT STILL NEED

You will often need to develop your skills in the area that you are talking about. For example, when you first start talking about malware, your reverse-engineering skills may need to be improved to allow you to understand more about the variants you find in the field.

CHAPTER

4.7 COMMUNITY SUPPORT (DOCUMENTATION, BUG PRIORITIZATION, PROJECT MANAGEMENT)

INTRODUCTION—WHAT THIS IS

While we discussed previously the booster of Developer (Open Source), there are other ways to help the open-source community.

Documentation for open-source projects tends to need updates, tweaks, or additional sections written, and sometimes is poor quality or doesn't exist. If you are interested in learning deeply about a piece of open-source software, helping write documentation is one of the best ways to do so.

Testing of the code is also another way to develop skills with software. Being part of the development process by testing software is another way to help an open-source community while learning about software.

After you have established yourself within a development community, you can often contribute in other manners. Helping to reproduce and determine the severity of bugs, thus helping determine their priority, or even helping with coordinating of the project as a whole, may be an option.

WHY YOU MIGHT WANT TO DEVOTE TIME TO THIS

For those who want to help build open-source software, but are not development-focused, this can be an ideal way to be part of the community. It also provides a way to build your skills and experience with the development process.

HOW THIS MIGHT COST YOU

The primary cost will be time. Most open-source projects look upon anyone willing to put in the effort to try to help as a welcome addition. There are exceptions, so find a community that fits with your personality.

HOW TO GET STARTED

The first step is to find a project you are interested in. Without that core interest, it will be difficult for you to maintain focus. Once you have the project identified, investigate the community. Ensure that it will be a good fit for your personality.

Now that you have identified the project, you can start helping. Each project will have its own take on how to integrate volunteers into the system.

WHEN YOU MIGHT WANT TO STOP

Community support will be either energizing, as a result of seeing your efforts moving into a production system, or demotivating, because you struggle with the project's management chain, or you feel as if you have no support from others in the community. Some people are able to push through the latter, but you will need to determine your limit. Additionally, projects may split—fork, in open-source parlance. If that happens, you will need to determine which project to join, or decide if it is time to leave.

WHAT SKILLS THIS GIVES YOU

Working in these roles provides you with the opportunity to develop your skills in the area of your choice. You can become a better writer, tester, or project manager, depending on the role you take on. Additionally, you'll develop your communication soft skills working within a group, no matter how strong they were when you started.

WHAT SKILLS YOU MIGHT STILL NEED

As you develop a reputation for success within the community, you will likely find that you are asked to assist in areas outside your area of expertise. Be honest and acknowledge what you can and cannot do; if the group is still willing to have you work in that area, you now have the chance to build those skills. For example, if you come in as a tester, you may be asked to help with documentation. If that is not your strong point, you have the option to refuse, or you can take advantage of the opportunity to work on your skills in that area. As this is purely a volunteer organization, being willing to branch out and succeed in new areas will help you develop a reputation as someone who gets the job done.

CHAPTER

4.8 CONFERENCE SUPPORT (FOUNDING, ATTENDING, VOLUNTEERING, RUNNING, LEADING)

INTRODUCTION—WHAT THIS IS

Conferences take significant effort. Even a small 80-person one day conference could take 100 to 150 hours of effort, depending on conference length, complexity, speakers, and facilities available to host the conference.

Conferences by local user groups or similar organizations often have trouble getting people to help. By volunteering or even taking a lead role, you help make these events happen.

Your role could be taking care of presenters, attendee enrollment, or food service and support. Those with more experience will tend toward leadership roles, such as managing a team of volunteers.

WHY YOU MIGHT WANT TO DEVOTE TIME TO THIS

Do you enjoy working with people? Do you like contributing to your local security community? Do you want to learn how to run conferences? If your answer is yes, then you just answered why you might want to devote time to this. For any conference, there will be stressful moments and scrambling leading up to and during the event, but the sense of accomplishment after it is done is a great feeling.

Doing this work will also provide you with an opportunity to network with others in the field, and perhaps have a chance to meet the presenters. However, remember that your job is so others can hear the presenters, not monopolize them for yourself.

HOW THIS MIGHT COST YOU

As with most volunteer efforts, the main cost will be time. There is the additional risk that if the conference ends up being viewed as run poorly, whether due to the errors of others or your own, it could taint people's views of your abilities.

HOW TO GET STARTED

Find a conference, especially one run by a small local user group or similar organization. They are the most likely to need, and be receptive to, those without experience helping them. From there you can expand your role in future conferences as you gain experience.

WHEN YOU MIGHT WANT TO STOP

You will take on one project, see it to the end, then continue on. Leaving a conference prematurely for any reason, so that others will need to cover for you, is a very effective way to taint your brand, not to mention just plain rude. Take some time after each one, and decide whether or not it is something that you are interested in doing again. Wait about a week after the conference to do your self-evaluation. Sooner, and the exhaustion from the conference may taint your view; later, and the event may not be fresh enough in your mind to evaluate completely.

WHAT SKILLS THIS GIVES YOU

Your organizational skills, project management, and people skills will all be improved by going through this process. Dealing with the various issues that come up will also help you with your crisis management skills, adjusting to situations on the fly—though hopefully everything that comes up is only an issue, not a crisis.

WHAT SKILLS YOU MIGHT STILL NEED

For those who have never worked in a conference, it can be overwhelming. There are multiple moving parts that come together at the last minute to make a whole experience for the attendees. Learning how to deal with those situations can be uncomfortable for some, since it can often feel as though you are out of control of the situation.

As you gain experience, you may find yourself founding or leading a user group or conference. The biggest skills you will need will focus on organization and communication. Try not to bite off more than you can chew and put a lot of effort into making sure that your support staff feels valued and that their work is worthwhile. It is all too common for a conference or user group to collapse when a handful of people who were carrying the weight get burned out and no one steps forward to keep things moving forward.

CHAPTER

4.9 USER GROUP SUPPORT (FOUNDING, ATTENDING, VOLUNTEERING, RUNNING, LEADING)

INTRODUCTION

> "80 percent of life is showing up."
>
> **— Woody Allen**

Information security user groups vary a lot, but most are volunteer-run and semi-formal, and meet monthly at a fixed time and place to study some part of information technology. They can be exactly like other community groups, such as book clubs, knitting groups, motorcycle clubs, or local sporting activities. Some are more organized and formal and have strict meeting times, dress codes, membership fees, and requirements, while others are more informal and may be free to join and open to all.

To benefit from this boost doesn't require that these groups be information security-related, although it helps, as the skills you will gain – communication, marketing, presenting, etc., will all be applicable to the day job.

There are several roles that can be present in a user group. Common roles are: president, secretary, treasurer, membership, scheduler, facility liaison, and newsletter editor. However, the single most important thing in running a user group is having a time and a place to meet. Even a leader isn't needed as long as people know when and where to go. What people do at the meeting is important, but might not be structured. Some user groups are really just social clubs where people hang out and talk and—depending on the meeting location—eat or drink.

WHY YOU MIGHT WANT TO DEVOTE TIME TO THIS

As with conferences, a user group is an opportunity to work with people, contribute, and learn. However, unlike conferences, user groups tend to have shorter but more frequent meetings, carry lower expectations, cost less money (if any), and involve lower stress. Since they involve groups of people who have common interests, they are a great way to meet people and socially network, and may also provide opportunity to Learn, Do, and Teach. Some groups are very structured, and have recurring classes, sponsor industry training and certification, and even run their own conferences.

HOW THIS MIGHT COST YOU

The most important initial contribution you can make to a user group is simply to show up. Showing up, especially for a smaller group, may be both noticed and appreciated. Showing up consistently is even more important, especially if you wish to be a contributing member or an organizer. Your time commitments for a meeting might be as short as an hour, or might take several hours, including travel time to and from the meeting location.

Getting more involved with a user group usually takes additional administrative time outside of meetings, although some user groups are so informal and small that it's built into the existing meeting. Most groups have a small core of organizers who actually do most, if not all, of the work of running the group. Organizations are often thankful for any volunteers who will consistently show up or provide help at other times. The organizers may meet shortly before or after the regular user group meetings, which helps reduce travel time, but may also have separate special meetings at other locations, remote meetings over teleconference lines, or communicate through other electronic means.

HOW TO GET STARTED

If you can't find an information security group in your area, check the IT groups in your area, even if they aren't an information technology you know or have an interest in. They may have special interest groups (SIG) for information security. If no SIGs are available, attend the IT groups anyway for the networking, and Learn about that area; perhaps you'll find some other information security group you missed earlier. If, after asking around you still haven't found one, start a security SIG in an existing group. If they haven't already asked you to start one, you may not have asked around enough. If you can't find a group you can extend, start your own. There are existing resources, so you don't have to do all the work on your own. Several international information security organizations have provisions for local chapters, so you can be the one to start the local chapter. Some are extremely easy to start because the central information security organization is very informal; others might require more work.

WHEN YOU MIGHT WANT TO STOP

If you don't have time to show up to meetings, you're mostly stopped. However, it's critical to let people know you can't attend or make commitments if you're a contributing member, and especially if you are an organizer.

If you can no longer Learn, Do, or Teach, it may be time to move on. Note, however, that the social aspects of user groups are none of these things but still have their own benefit.

Bad politics in a user group is reason for people to leave and never return. If you're a new arrival to a group with troubled dynamics, it might be an opportunity to get involved. Beware, though—it can be especially bad for your brand to take sides and get entangled in such conflicts. The politics can be especially vicious in a user group since, as the truism goes, the fighting is never so fierce as when the stakes are so low.

If it's not fun or sociable for you, others may not find your contributions enjoyable, either. Although a very technically minded user group may be willing to acquire the learning, doing, and teaching you offer, even if it isn't particularly fun, most user groups won't. The social aspects of a user group are strong and should not be underestimated.

Some organizations have formal mechanisms to move old leaders out to make room for others, and others don't. Organizations often have a hard time recruiting and maintaining organizers. As with any job, paid or not, it's much better to move out on your own power than get fired from the group in a coup d'état.

If you want to move on and there is someone who could take your place, this is an opportunity to be a mentor and Teach, so that another may Learn, and the organization will move on and survive without you, which is better for both you and the user group.

WHAT SKILLS THIS GIVES YOU

User groups need ongoing and consistent organizational and social skills; otherwise, they die. The most important things are to show up, get other people to show up, and have everyone leave happy enough that they'll show up next time and maybe bring friends. Doing this may require other skills, such as finding good speakers, teachers, or interesting activities—or doing them yourself, using the other boosting skills of presenter. Some user groups will be interesting enough that they will be self-sustaining without special work, as long as they have a meeting time and place. But most won't be self-sustaining and will need at least some ongoing help. If you start a user group that lasts more than a year, you'll have accomplished something. But if your user group survives your leaving it, then you have *created* something.

User groups can have several expenses: meeting spaces, meeting equipment and storage, meeting food and beverages, newsletter printing and distribution, advertisement posters and cards, website, a library, or special equipment.

User groups with fund raising activities will also require budgeting, accounting, and auditing skills. If the user group has no funds, you won't learn those skills, but you will learn how to do something for almost nothing. User groups with formal nonprofit status have legal and tax record-keeping requirements that you will have to learn or delegate to someone who does.

WHAT SKILLS YOU MIGHT STILL NEED

Generally there aren't that many information security skills to be learned from *running* a user group, unless you make it a core part of what the user group does. A user group could teach almost anything, but it probably won't teach everything, so whatever the group doesn't have as part of its goal, you won't learn, except what you get from running the group itself. Not all user groups are about the technical skills; some are really just social clubs for like-minded people to hang out. What the user group wants and what it's even capable of are totally dependent upon the organizers' and members' time, interest, and resources. Make your user group what you will of it.

Be aware that running a user group is enough work that you may not have time to either learn or teach that much in the actual activities of the user group. It's entirely possible to be so busy that you'll only learn how to run a successful user group and not benefit at all from what the user group itself is teaching its members. However, the skills to run a successful user group are quite valuable, and you'll probably also gain valuable social network contacts.

CONCLUSION

> "If you want to build a ship, don't drum up people together to collect wood and don't assign them tasks and work, but rather teach them to long for the endless immensity of the sea."
>
> **— Antoine de Saint Exupery**

Congratulations on making it through the book! We hope this means you love information security as much as we do and perhaps you've found the book to be interesting if not useful.

In addition to Learn/Do/Teach two more big messages of this book are:

1. As a community, we only get better when we work together and share information. Information wants to be free, although the book is not free, please share the book's ideas with anyone interested.
2. There are many paths and careers that only make sense in hindsight. Be sure to experiment and take risks, as that's the only way to truly improve.

If you've read the book straight through from beginning to end, then you have reached the conclusion but not the end. You have Learning, Doing, and Teaching to do next.

If you didn't markup any pages, take notes, or plan your next steps, go back and do that now.

Really, go do that, now.

On the gripping hand, if you've read the book and Learned, Done, and Taught while you read it, that's great! You're already working on your information security career and may be looking forward to your next job. Keep Learning, Doing, and Teaching!

And if this book didn't work for you, please Teach us. We really do want to make this book better.

Information security is what we do, and we love doing it. Our hope is that you love it, too, and will keep loving it as part of your new career.

There are many paths into, through, and out of information security. We hope you'll share your journey with all of us in the community.

Though this is the end, this book is not about endings, but beginnings. We can't tell you what will make you happy, but I hope we've helped you figure out what will make you happier while on your journey.

Now go out and Learn for yourself, Do work, Teach others, and get your next job!

When you're done with this book, please lend or gift it to someone who would also love to break into information security.

APPENDIX

APPENDIX: INTRODUCTION

> "Cool URIs don't change."
>
> **— Sir Tim Berners-Lee, inventor of the World Wide Web**

All references were accurate and up to date at the time of submission. Which means if you are reading this right now in a paper book they are almost certainly out of date. If you're reading this in an ebook, there was a brief time before you started reading it when it wasn't out of date. URLs break, books go out of print, and new editions have new ISBNs. Organizations change names (or meanings), and sometimes go away. We hope at least some of these are cool enough to not change.

Use these resources names as search keywords but be careful of what search results you get because you're responsible for what you find and use.

SECURITY MODELS

ACI (CIA) triad — https://en.wikipedia.org/wiki/Information_security#Key_concepts
Parkerian Hexad — 01998, Donn B. Parker, https://en.wikipedia.org/wiki/Parkerian_hexad
(ISC)^2 Common Body of Knowledge (CBK) as of 02015-06 — https://www.isc2.org/cissp-domains/

JOB RESOURCES

In addition to the below, some organizations, national and local, have their own job postings. Some conferences have active job recruitment and job posting areas. See below Appendix: Communities.

Association of Information Technology Professionals (AITP) Career Center — http://jobs.aitp.org/
DAMA International, The Global Data Management Community, Careers
http://dama-jobs.careerwebsite.com/
International Association for Cryptologic Research (IACR) — http://iacr.org/jobs/
US Government Clearance Jobs — https://www.clearancejobs.com/
US Department of Homeland Security— http://www.dhs.gov/join-dhs-cybersecurity
US FBI — http://www.fbijobs.gov/cybercareers/
US Secret Service — http://www.secretservice.gov/ectf_about.shtml
Y Combinator funded startup jobs — https://news.ycombinator.com/jobs

GENERAL BOOKS AND REFERENCES

NIST IR-7298 Rev. 2, Glossary of Key Information Security Terms
http://nvlpubs.nist.gov/nistpubs/ir/2013/NIST.IR.7298r2.pdf

The Cuckoo's Egg by Cliff Stoll, 01989— ISBN:0-385-24946-2
The Security Principles of Saltzer and Schoeder — http://emergentchaos.com/the-security-principles-of-saltzer-and-schroeder see also Jerome Saltzer and Michael Schoeder "The Protection of Information in Computer Systems" http://www.cs.virginia.edu/~evans/cs551/saltzer/
The Personal MBA: A World-Class Business Education in a Single Volume — Kaufman, Josh, 02011 — ISBN:978-0-14-197109-4

TIME MANAGEMENT

The Long Now Foundation — http://longnow.org/ https://en.wikipedia.org/wiki/Long_Now_Foundation
Getting Things Done: The Art of Stress-Free Productivity by Allen — ISBN:0142000280
Personal Kanban: Mapping Work | Navigating Life by Benson and Barry — ISBN:1453802266
The 7 Habits of Highly Effective People by Covey — ISBN:0743269519
Pomodoro Technique — http://www.pomodorotechnique.com/ https://en.wikipedia.org/wiki/Pomodoro_Technique
Scrum and sprints — https://en.wikipedia.org/wiki/Scrum_(software_development) https://en.wikipedia.org/wiki/Scrum_%28software_development%29#Sprint
Paul Graham — The Acceleration of Addictiveness — http://paulgraham.com/addiction.html https://en.wikipedia.org/wiki/Minimum_viable_product
Simplest Thing That Could Possibly Work — http://c2.com/cgi/wiki?SimplestThingThatCouldPossiblyWork
Yak shaving — https://en.wiktionary.org/wiki/yak_shaving
The New York Times — Current History of the European War, 01915

STORY TELLING AND METAPHOR

I Is an Other: The Secret Life of Metaphor and How It Shapes the Way We See the World by Geary — ISBN:9780061710292
Images of Organization by Morgan — ISBN:1412939798
Made to Stick: Why Some Ideas Survive and Others Die by Heath and Heath —ISBN:1400064287
Master Metaphor List v2 by George Lakoff
Jane Espenson, and Alan Schwartz, 01991 — http://araw.mede.uic.edu/~alansz/metaphor/METAPHORLIST.pdf
Metaphors We Live By by Lakoff and Johnson — ISBN:0226468011
Pixar Story Rules by Emma Coats — http://twitter.com/lawnrocket http://filmmakeriq.com/2012/08/the-pixar-story-rules http://www.pixartouchbook.com/blog/2011/5/15/pixar-story-rules-one-version.html
Gripping hand — https://en.wikipedia.org/wiki/The_Gripping_Hand#Notes_and_references http://www.catb.org/jargon/html/O/on-the-gripping-hand.html

SCIENTIFIC METHOD

https://en.wikipedia.org/wiki/Scientific_method
https://en.wikipedia.org/wiki/List_of_fallacies
https://en.wikipedia.org/wiki/List_of_cognitive_biases

Richard Feynman — http://www.richardfeynman.com/ 01974 “Cargo Cult Science, Some remarks on science, pseudoscience, and learning how to not fool yourself. Caltech’s 1974 commencement address.” http://calteches.library.caltech.edu/51/02/CargoCult.pdf ; Engineering and Science 37 (7); Space Shuttle Challenger Disaster investigation, *What Do You Care What Other People Think?*, 01988, ISBN 0-393-02659-0, 02001 paperback: ISBN 0-393-32092-8;

Thomas Kuhn — *The Structure of Scientific Revolutions,* 01962 ISBN:9780226458113 https://en.wikipedia.org/wiki/The_Structure_of_Scientific_Revolutions

APPENDIX: COMMUNITIES

Association of Information Technology Professionals (AITP) — http://www.aitp.org/
DAMA International, Data Access Management Association (DAMA), The Global Data Management Community — http://www.dama.org/
Hackers for Charity — http://www.hackersforcharity.org/
FBI InfraGard — http://www.infragard.net/
Free Software Foundation — https://www.fsf.org/
Information Systems Security Association (ISSA) — https://issa.org/
Open Web Application Security Project (OWASP) — https://www.owasp.org/
Society of Hispanic Professional Engineers (SHPE) — http://www.shpe.org/

MAILING LISTS

The best mailing lists are private. The SANS Advisory Board is available to those who take and pass a SANS course with a high enough score.

Full Disclosure and BugTraq RSS feeds, and more — http://seclists.org/
The Forum of Incident Response Teams (FIRST) private organization requires sponsorship and vetting that to join — http://www.first.org/
GPWN, SANS alumni only — https://lists.sans.org/mailman/listinfo/gpwn-list
PaulDotCom list — http://mail.pauldotcom.com/cgi-bin/mailman/listinfo/pauldotcom
SANS DFIR, job postings — https://lists.sans.org/mailman/listinfo/dfir
SANS @RISK — http://www.sans.org/newsletters/at-risk/
SANS Newsbites — http://www.sans.org/newsletters/newsbites/
SANS OUCH! — http://www.securingthehuman.org/resources/newsletters/ouch/

CONFERENCES AND COMMUNITIES

In addition to these, there are communities in some web sites and blogs, and national organizations often have local chapters, if there isn't one, start it yourself, see the chapter "Boosting 4.9— User Groups".

Ada Initiative — http://adainitiative.org/
http://geekfeminism.wikia.com/wiki/Feminist_and_women%27s_hackerspaces
AdaCamp — https://adacamp.org/
Black Hat, Las Vegas, NV USA, Europe, Asia — http://www.blackhat.com/
Security BSides, worldwide local volunteer run alternative information security conferences during other conferences, job recruitment — http://www.securitybsides.com/
CanSecWest, Vancouver, BC, Canada — http://www.cansecwest.com/
CarolinaCon, Raleigh, NC, USA — http://www.carolinacon.org/
Chaos Communications Congress, Hamburg, Germany — http://www.ccc.de/congress/
Chaos Communications Camp (CCC), Germany — https://events.ccc.de/camp/
DEF CON, Las Vegas, NV, USA — http://defcon.org/
DEF CON Forums — https://forum.defcon.org/
DEF CON Groups — https://defcongroups.org/
DerbyCon, Louisville, KY, USA — https://www.derbycon.com/

Hackers On Planet Earth (HOPE), New York, NY, USA — http://hope.net/
Hack-Tic events, Netherlands — http://www.hacktic.nl/ https://en.wikipedia.org/wiki/Hack-Tic
(ISC)^2 Twin Cites Chapter Information Security Events (MN & North America) — http://isc2tc.org/
MeetUp — search for "security", "infosec", "software", "development", "nerd", "geek", "hacker", security vendor and product names in your area http://www.meetup.com/
Notacon, Cleveland, OH, USA — http://www.notacon.org/
PhreakNIC, Nashville, TN, USA — http://phreaknic.info/
Reddit —
https://www.reddit.com/r/HowToHack http://www.reddit.com/r/hacking/
https://www.reddit.com/r/netsec https://www.reddit.com/r/security
https://www.reddit.com/r/AskNetsec https://www.reddit.com/r/infosec
https://www.reddit.com/r/malware https://www.reddit.com/r/pwned
https://www.reddit.com/r/ReverseEngineering http://www.reddit.com/r/netsec/
ShmooCon Washington, DC, USA — https://www.shmoocon.org/
SkyDogCon — Nashville, TN, USA http://skydogcon.com/ http://skydogcon.blogspot.com/
StackExchange, Information Security — https://security.stackexchange.com/
ThotCon, Chicago, IL, USA — http://www.thotcon.org/
Toastmasters — http://www.toastmasters.org/ https://en.wikipedia.org/wiki/Toastmasters
ToorCon, San Diego, CA, USA — http://toorcon.org/
USENIX — Cyberlaw; HotBots; Workshop on Hot Topics in Security (HotSec); LISA; Symposium on Usable Privacy and Security (SOUPS); Usability, Psychology, and Security (UPSEC); Women in Advanced Computing Summit (WiAC); Workshop on Offensive Technologies (WOOT), USENIX Security Symposium, world wide, job recruitment — http://usenix.org/conference/

APPENDIX: SOFTWARE TOOLS

Amazon Web Services (AWS) Free Usage Tier — http://aws.amazon.com/free/
Duck Duck Go privacy oriented search engine — https://duckduckgo.com/
EFF HTTPS Everywhere and Privacy Badger for Mozilla Firefox — https://www.eff.org/
GnuPG, S/MIME and OpenPGP cryptography, encryption — http://www.gnupg.org/
LibreOffice — http://www.libreoffice.org/
Nessus— http://www.tenable.com/products/nessus
nginx web server — http://nginx.org/
nmap, network mapper and security scanner— http://nmap.org/
Mozilla Firefox with NoScript security add-on — https://www.mozilla.org/
OpenSSL cryptographic library — http://www.openssl.org/
Oracle VirtualBox — https://www.virtualbox.org/
PRISM Break privacy tools — https://prism-break.org/
Puppet, configuration management — https://puppetlabs.com/
SecTools.Org: Top 125 Network Security Tools — http://sectools.org/TOR, The Onion Router anonymity software and network — https://www.torproject.org/

Programming and Software Tools

Apache Tomcat, JBoss, ModSecurity — http://apache.org/
Burp Suite, web application Proxy, Spider, Scanner, Intruder, Repeater, Sequencer — https://portswigger.net/burp/
GNU Emacs “the extensible self-documenting text editor” — https://www.gnu.org/software/emacs
Fuzzers — http://crashme.codeplex.com/ http://pages.cs.wisc.edu/~bart/fuzz/ https://en.wikipedia.org/wiki/Fuzz_testing
SourceForge — http://sourceforge.net/
GitHub — https://github.com/
Microsoft

- Microsoft Developer Network (MSDN) — https://msdn.microsoft.com/
- PowerShell — http://microsoft.com/powershell
- CodePlex — https://www.codeplex.com/

Perl

- http://www.perl.org/
- *Learning Perl* by Randal L. Schwartz, brian d foy, Tom Phoenix — http://www.oreilly.com/ ISBN:978-1-4493-0358-7 (print) ISBN:978-1-4493-0458-4 (ebook)
- *Programming Perl* by Tom Christiansen, brian d foy, Larry Wall, Jon Orwant — http://www.oreilly.com/ ISBN:978-0-596-00492-7 (print) ISBN:978-1-4493-9890-3 (ebook)
- *Mastering Regular Expressions* by Jeffrey Friedl, O’Reilly Media — http://www.oreilly.com/ ISBN:0-596-52812-4
- *Perl Best Practices* by Damian Conway — http://www.oreilly.com/ ISBN:0-596-00173-8
- *Advanced Perl* — http://www.oreilly.com/
- Perl Mongers, community — http://www.pm.org/
- Perl Monks, communit — http://www.perlmonks.org/

Python
  https://www.python.org/
  Python the hard way — http://learnpythonthehardway.org/
  Python tools for penetration testers — http://dirk-loss.de/python-tools.htm
Ruby
  https://www.ruby-lang.org/
  http://learnrubythehardway.org/
Vim, Vi Improved — http://www.vim.org/

GNU/LINUX DISTRIBUTIONS

Debian — https://www.debian.org/
Kali Live distribution for digital forensics and penetration testing — https://www.kali.org/
Knoppix Live distribution for general use with hundreds of pre-installed software packages — http://www.knopper.net/knoppix/index-en.html
Security Onion, SIEM, intrusion detection, network security monitoring, and log management security tool system by Doug Burk — https://github.com/security-onion-solutions/security-onion contains Snort, Suricata, Bro, OSSEC, Sguil, Squert, Snorby, ELSA, Xplico, NetworkMiner, and many other security tools. http://securityonion.net/
Slackware — http://www.slackware.com/
The Amnesiac Incognito Live System (TAILS) privacy and anonymity — https://tails.boum.org/

APPENDIX: SELF STUDY

BugCrowd A comprehensive, up to date list of bug bounty and disclosure programs from across the web curated by the Bugcrowd researcher community. — https://bugcrowd.com/list-of-bug-bounty-programs/

Progressive Games

crackme — https://en.wikipedia.org/wiki/Crackme
code katas — https://en.wikipedia.org/wiki/Code_Kata
Matasano/Square Embedded Security Capture the Flag Challenge — https://microcorruption.com/
Starfighter CTF, programming competion instead of technical interviews or resumes — http://www.starfighters.io/ http://www.kalzumeus.com/2015/03/09/announcing-starfighter/
EnigmaGroup, wide range of excercizes — http://www.enigmagroup.org/
GameOver, insecure web applications — http://sourceforge.net/projects/null-gameover/
SecuraBit Gh0st PenLab, CTF — http://www.gh0st.net/
Google Gruyere, Web Application Exploits and Defenses, small, cheesy web application codelab — http://google-gruyere.appspot.com/
Hacker Challenge — http://www.dareyourmind.net/
Hacker Test, JavaScript, PHP, HTML — http://www.hackertest.net/
Hacking-Lab, CTF and mission style challenges for the European Cyber Security Challenge — https://www.hacking-lab.com/
Hack.me, vulnerable web applications, code samples and CMS's online — https://hack.me/ http://www.elearnsecurity.com/
HackThis, JavaScript, SQLi, Coding, Crypt, Captcha, Forensics, community — http://www.hackthis.co.uk/
Hack This Site, Programming, JavaScript, Forensics, Stego, Irc — https://www.hackthissite.org/
Hax.Tor, 02006 many levels deprecated, — http://hax.tor.hu/
hackxor, virtual machine image like WebGoat but with a plot — http://hackxor.sourceforge.net/cgi-bin/index.pl
OverTheWire, SSH shell access — http://www.overthewire.org/wargames/
p0wnlabs, free sample challenges forensics, password cracking, OpenVPN, Metasploitable, WebGoat, OWASPBWA, pay challenges — http://www.p0wnlabs.com/free
pwn0, VPN access — https://pwn0.com/home.php
Root Me, hundreds of challenges, virtual machines — http://www.root-me.org/?lang=en
Security Treasure Hunt, web vulnerability, forensics — http://www.securitytreasurehunt.com/
Smash The Stack, SSH shell access — http://www.smashthestack.org/
sqli-labs, a platform to learn SQLi — https://github.com/Audi-1/sqli-labs
TheBlackSheep and Erik, Programming, JavaScript, PHP, Java, Steganography, and Cryptography — http://www.bright-shadows.net/
ThisIsLegal, hacker wargames — http://thisislegal.com/
Try2Hack, — http://www.try2hack.nl/
WabLab, SQL, web application — http://www.wablab.com/hackme
VulnApp — http://www.nth-dimension.org.uk/blog.php?id=88

Network Targets

US NIST Computer Forensic Reference Data Sets (CFReDS) — http://www.cfreds.nist.gov/
Damn Vulnerable Linux, 02010 — http://sourceforge.net/projects/virtualhacking/files/os/dvl/
Handler Diaries, Digital Forensics and Incident Response — http://blog.handlerdiaries.com/
Kioptrix, virtual machine challenges — http://www.kioptrix.com/blog/test-page/
LAMPSecurity, vulnerable virtual machine images to teach linux,apache,php,mysql security — http://sourceforge.net/projects/lampsecurity/
Metasploitable, intentionally vulnerable Linux virtual machine — http://sourceforge.net/projects/virtualhacking/files/os/metasploitable/
Metasploitable2, intentionally vulnerable Linux virtual machine — http://sourceforge.net/projects/metasploitable/files/Metasploitable2/
GoatseLinux: It's Wide Open, 02009 — http://neutronstar.org/goatselinux.html
pWnOS — http://www.pwnos.com/
RebootUser Vulnix, vulnerable Linux host with configuration weaknesses rather than purposely vulnerable software versions. The goal; boot up, find the IP, hack away and obtain the trophy — http://www.rebootuser.com/?page_id=1041
UltimateLAMP, PHDays iBank CTF — http://www.amanhardikar.com/mindmaps/practice-links.html
Vulnserver, vulnerable Windows based threaded TCP server application — http://www.thegreycorner.com/2010/12/introducing-vulnserver.html

Web Targets

Metasploit Unleashed, free training from Hackers for Charity — http://www.offensive-security.com/metasploit-unleashed/Main_Page
Metasploitable, use with Metasploit Unleashed http://www.offensive-security.com/metasploit-unleashed/Metasploitable
Backtrack Tutorials — http://www.backtrack-linux.org/tutorials/
Hack This Site, Programming, JavaScript, Forensics, Stego, Irc — http://www.hackthissite.org/
BodgeIt Store, a vulnerable web application for those new to pen testing — https://github.com/psiinon/bodgeit
Butterfly Security, web application and PHP vulnerabilities and mitigation — http://sourceforge.net/projects/thebutterflytmp/
CryptOMG, common cryptographic flaws CTF — https://github.com/SpiderLabs/CryptOMG
Damn Vulnerable Web App (DVWA), PHP/MySQL — http://www.dvwa.co.uk/
Damn Vulnerable Web Services (DVWS) — http://dvws.professionallyevil.com/
Exploit KB Vulnerable Web App, SQLi, PHP, MySQL — http://exploit.co.il/projects/vuln-web-app/ https://sourceforge.net/projects/exploitcoilvuln
Foundstone Hackme Bank, MS-Windows, 02006 — http://www.mcafee.com/us/downloads/free-tools/hacme-bank.aspx
Foundstone Hackme Books, MS-Windows, Java, 02006 — http://www.mcafee.com/us/downloads/free-tools/hacmebooks.aspx
Foundstone Hackme Casino, MS-Windows, 02006 — http://www.mcafee.com/us/downloads/free-tools/hacme-casino.aspx

Foundstone Hackme Shipping, MS-Windows, Adobe ColdFusion, MySQL, 02006 — http://www.mcafee.com/us/downloads/free-tools/hacmeshipping.aspx

Foundstone Hackme Travel, MS-Windows client/server SQL — http://www.mcafee.com/us/downloads/free-tools/hacmetravel.aspx

LAMPSecurity, vulnerable virtual machine images to teach linux,apache,php,mysql security — http://sourceforge.net/projects/lampsecurity/

Magical Code Injection Rainbow (MCIR), SQLol, XMLmao, ShelLOL and XSSmh — https://github.com/SpiderLabs/MCIR

Moth, VMware image with vulnerable Web Applications and scripts — http://www.bonsai-sec.com/en/research/moth.php

NOWASP / Mutillidae 2, vulnerable web-application for Linux and Windows using LAMP, WAMP, and XAMMP, pre-installed on SamuraiWTF, Rapid7 Metasploitable-2, and OWASP BWA — http://sourceforge.net/projects/mutillidae/ http://www.irongeek.com/i.php?page=mutillidae/mutillidae-deliberately-vulnerable-php-owasp-top-10

OWASP Bricks, vulnerable web application built on PHP and MySQL exploitable using Mantra and ZAP — http://sourceforge.net/projects/owaspbricks/

OWASP Broken Web Apps, vulnerable web applications on a Virtual Machine — https://www.owasp.org/index.php/OWASP_Broken_Web_Applications_Project

OWASP Broken Web Applications Project (BWA), vulnerable web applications on VMware virtual machine — http://code.google.com/p/owaspbwa/

OWASP Security Shepherd, web and mobile application security training platform — https://www.owasp.org/index.php/OWASP_Security_Shepherd

OWASP SiteGenerator, dynamic websites based on XML files and predefined vulnerabilities — https://www.owasp.org/index.php/Owasp_SiteGenerator

PuzzleMall, Java/JSP, Apache Derby, Temporal Session Race Conditions (TSRC) and Layer Targeted AdoS, 02011 — http://code.google.com/p/puzzlemall/

SecuriBench, Java, SQL injection attacks, Cross-site scripting attacks, HTTP splitting attacks, Path traversal attacks — http://suif.stanford.edu/~livshits/securibench/

SocketToMe, PHP, chat, a simple number guessing game and a few other hidden features — http://digi.ninja/projects/sockettome.php

WackoPicko, part of OWASP BWA Project — https://github.com/adamdoupe/WackoPicko "Why Johnny Can't Pentest: An Analysis of Black-box Web Vulnerability Scanners" http://cs.ucsb.edu/%7Eadoupe/static/black-box-scanners-dimva2010.pdf

WebGoat.NET — https://github.com/jerryhoff/WebGoat.NET/ https://www.owasp.org/index.php/WebGoat_User_Guide_Table_of_Contents

WebSecurity Dojo, self-contained training environment for Web Application Security penetration testing xubuntu 12.04 — http://sourceforge.net/projects/websecuritydojo/files/ http://dojo.mavensecurity.com/

OWASP Zed Attack Proxy - Web Application Vulnerability Examples (WAVE), for testing OWAP ZAP — http://code.google.com/p/zaproxy/downloads/detail?name=zap-wave-0.1.zip

Hewlett-Packard Fortify WebInspect product demo Zero Bank — http://zero.webappsecurity.com/

APPENDIX: CERTIFICATIONS

ASIS Certified Protection Professional (CPP), Professional Certified Investigator (PCI), Physical Security Professional (PSP) — https://www.asisonline.org/
Cisco Certified Network Associate (CCNA) — https://learningnetwork.cisco.com/community/certifications/
CompTIA Security+ CE — http://certification.comptia.org/Training/testingcenters/examobjectives.aspx (registration required)
EC Council CEH — http://www.eccouncil.org/Certification/professional-series/ceh-course-outline
ISACA Certified in Risk and Information Systems Control (CRISC) — https://www.isaca.org/
(ISC)^2 CISSP — https://www.isc2.org/exam-outline/default.aspx (registration required)
JNCIA -Junos (Juniper Networks Certified Associate Junos) — http://www.juniper.net/us/en/training/certification/
Offensive Security OSCP — http://www.offensive-security.com/
SANS GIAC Certified Incident Handler (GCIH) — http://www.giac.org/certification/certified-incident-handler-gcih
US Department of Defense Approved 8570 Baseline — http://iase.disa.mil/iawip/Pages/index.aspx

APPENDIX: NEWS

Privacy Rights Clearinghouse — https://www.privacyrights.org/
CSO Online — http://csoonline.com/
Freedom to Tinker — https://freedom-to-tinker.com/
Cambridge University— https://www.lightbluetouchpaper.org/
Dark Reading — http://www.darkreading.com/
InfoWorld Security Channel — http://www.infoworld.com/category/security
BankInfoSecurity — http://www.bankinfosecurity.com/
Wired Threat Level — http://www.wired.com/category/security
The Risks Digest — https://catless.ncl.ac.uk/Risks/
Ycombinator Hacker News — https://news.ycombinator.com/

APPENDIX: PEOPLE

Douglas Adams — Hitchhiker's Guide to the Galaxy, http://douglasadams.com/
Carl Ally — 01965, http://www.barrypopik.com/index.php/new_york_city/entry/a_consultant_is_someone_who_borrows_your_watch_to_tell_you_the_time_and_the/
Ross Anderson — http://www.ross-anderson.com/ *Security Engineering* http://www.cl.cam.ac.uk/~rja14/book.htm
Programming Satan's Computer — http://www.cl.cam.ac.uk/~rja14/Papers/satan.pdf
Steve Bellovin — https://www.cs.columbia.edu/~smb/blog/
Richard Bejtlich — http://taosecurity.blogspot.com/ http://taosecurity.blogspot.com/search/label/bestbook
Tim Berners-Lee — http://www.w3.org/People/Berners-Lee/ https://en.wikipedia.org/wiki/Tim_Berners-Lee
Matt Blaze — https://twitter.com/mattblaze/ https://en.wikipedia.org/wiki/Matt_Blaze
Fredrick P. Brooks, Jr., The Mythical Man-Month, Essays on Software Engineering, 01975
George E. P. Box — https://en.wikipedia.org/wiki/George_E._P._Box
Albert Einstein — "On the Method of Theoretical Physics" *Philosophy of Science*, Vol. 1, No. 2 (April 01934), pp. 163–9, p. 165
Robert A. Heinlein, Time Enough for Love: the Lives of Lazarus Long, 01973
Mak Kolybabi — https://twitter.com/mogigoma http://mogigoma.com/
Alfred Korzybski "A Non-Aristotelian System and its Necessity for Rigour in Mathematics and Physics", a paper presented before the American Mathematical Society at the New Orleans, Louisiana, meeting of the American Association for the Advancement of Science, December 28, 1931. Reprinted in Science and Sanity, 1933, p. 747–61. https://en.wikipedia.org/wiki/Alfred_Korzybski https://en.wikipedia.org/wiki/Map-territory_relation
Ursula K. Le Guin — http://www.ursulakleguin.com/ https://en.wikipedia.org/wiki/Ursula_K._Le_Guin
Paul Graham, time management, The Acceleration of Addictiveness — http://paulgraham.com/addiction.html
Jeremiah Grossman — http://www.whitehatsec.com/
Hanlon — https://en.wikipedia.org/wiki/Hanlon%27s_razor
C.A.R. Hoare, The Emporer's New Clothes — https://en.wikiquote.org/wiki/Tony_Hoare#The_Emperor.27s_Old_Clothes
Juvenal — aka Decimus Iunius Iuvenalis c. 00055/00140
Dan Kaminsky — http://dankaminsky.com/
Guy Kawasaki — http://www.guykawasaki.com/
Brian Krebs on Security — http://www.krebsonsecurity.com/ http://krebsonsecurity.com/category/how-to-break-into-security/
Donald Knuth — http://cs.stanford.edu/~uno/ https://en.wikiquote.org/wiki/Donald_Knuth
Niccolo Machiavelli — Discourses on Livy, Book 1, Chapter XLVI

Josh More — http://www.starmind.org/ *Job Reconnaissance: Using Hacking Skills to Win the Job Hunt Game* by Josh More — Syngress ISBN-13:978-0124166011
http://www.starmind.org/2012/04/07/so-you-want-a-new-job-adapted-from-a-presentation/
http://www.starmind.org/2012/01/13/security-certification-23-learning/

Red Queen — Through the Looking-Glass by Lewis Carroll, 01871

Stephen Northcutt — http://www.sans.edu/about/governance/administrators#stephen-northcutt

Gunnar Peterson — http://1raindrop.typepad.com/

Terry Pratchett — https://en.wikiquote.org/wiki/Terry_Pratchett#Usenet

Thomas Ptacek — http://sockpuppet.org/ https://news.ycombinator.com/user?id=tptacek

PaulDotCom — Security Podcast http://pauldotcom.com/

Alan Perlis — "Epigrams on programming". ACM SIGPLAN Notices (New York, NY, USA: Association for Computing Machinery) 17 (9): 7–13. doi:10.1145/947955.1083808

Antoine de Saint Exupery — https://en.wikiquote.org/wiki/Antoine_de_Saint_Exupery

Bruce Schneier — https://www.schneier.com/ So You Want to Be a Security Expert by Bruce Schneier — http://www.schneier.com/blog/archives/2012/07/how_to_become_a_1.html
Inside the Twisted Mind of the Security Professional by Bruce Schneier — http://www.schneier.com/essay-210.html

Rick Smith — http://www.cryptosmith.com/

Gene Spafford — http://blog.spaf.us/

Richard Stallman — https://www.stallman.org/ https://en.wikipedia.org/wiki/Richard_Stallman

Cliff Stoll — "The Cuckoo's Egg: Tracking a Spy Through the Maze of Computer Espionage", 01989 ISBN:0-385-24946-2

Andrew S. Tanenbaum — *Computer Networks*, 4th ed., p. 91; http://www.cs.vu.nl/~ast; https://en.wikipedia.org/wiki/Andrew_S._Tanenbaum

Roy Trenneman — The IT Crowd, Channel 4 Television, UK
http://www.channel4.com/programmes/the-it-crowd

Jan L. A. van de Snepscheut — https://en.wikipedia.org/wiki/Jan_L._A._van_de_Snepscheut

Gerald Weinberg — http://www.geraldmweinberg.com/
https://en.wikiquote.org/wiki/Gerald_Weinberg

David Wheeler— https://en.wikipedia.org/wiki/David_Wheeler_%28computer_scientist%29
https://en.wikipedia.org/wiki/Fundamental_theorem_of_software_engineering

Phillip J. Windley, CIO of the state of Utah, USA 02001/02002 — http://phil.windley.org/
http://www.windley.com/archives/2005/05/organization_ge.shtml
http://www.cio.com.au/article/164604/just_desserts_/

Ira Winkler — https://twitter.com/irawinkler http://xcompanionguide.com/
http://www.securementem.com/about-us/

John Ziman — *Knowing Everything about Nothing: Specialization and Change in Scientific Careers,* 01987 ISBN:0-521-32385-1; https://en.wikipedia.org/wiki/John_Ziman

Subject Index

Congratulations on as thoroughly reading the book as we tried to be
in writing it! You can use this space for planning your custom career path,
writing your story, and for notes. — The Authors