مهندسی رمزنگاری

اصول طراحی و کاربردی عملی

نیلز فرگوسن
بروس اشنایر
تادایوشی کونو

ترجمه
احد درفشی چارطاق

مهندسی رمزنگاری

اصول طراحی و کاربردی عملی

نویسندگان

نیلز فرگوسن

بروس اشنایر

تادایوشی کونو

ترجمه

احد درفشی چارطاق

سرشناسه : کوهنو، تادایوشی
Kohno, Tadayoshi
عنوان و نام پدیدآور : مهندسی رمزنگاری : اصول طراحی و کاربردی عملی/ نویسندگان تادایوشی کونو، نیلز فرگوسن، بروس اشنایر ؛ ترجمه احد درفشی‌چارطاق.
مشخصات نشر : تبریز: انتشارات مرتضی دشت، ۱۳۹۳.
مشخصات ظاهری : ۴۵۶ ص.
شابک : ۳۲۰۰۰۰ ریال: 978-600-94388-5-3
وضعیت فهرست نویسی : فیپا
یادداشت : عنوان اصلی: Cryptography engineering : design principles and practical applications ., c2010.
موضوع : رمزنگاری
موضوع : کامپیوترها -- ایمنی اطلاعات
شناسه افزوده : فرگوسن، نیلس
شناسه افزوده : Ferguson, Niels
شناسه افزوده : اشنایر، بروس، ۱۹۶۳ - م.
شناسه افزوده : Schneier, Bruce
شناسه افزوده : درفشی چارطاق، احد، ۱۳۶۲ -، مترجم
رده بندی کنگره : QA۷۶/۹/ ر۸ک۹ ۱۳۹۳
رده بندی دیویی : ۰۰۵/۸
شماره کتابشناسی ملی : ۳۶۱۸۴۶۵

نام کتاب: مهندسی رمزنگاری اصول طراحی و کاربردی عملی
نویسندگان: تادایوشی کونو، نیلز فرگوسن، بروس اشنایر
مترجم: احد درفشی چارطاق
چاپ اول: پائیز ۹۳
انتشارات: مرتضی دشت
شمارگان: ۱۰۰۰ نسخه
قیمت: ۳۲۰۰۰۰ ریال
شابک: ۹۷۸-۶۰۰-۹۴۳۸۸-۵-۳

دفتر پخش: تبریز، خیابان امام، چهارراه شهید بهشتی، اول خیابان حافظ ساختمان آذر ۱۲۵ طبقه دوم شمالی- ۰۴۱۳۵۵۴۸۸۹۱ - ۰۹۱۴۴۰۱۷۹۳۰

بیشتر کتاب‌ها در مورد اینکه رمزنگاری چیست صحبت کرده‌اند و اغلب کل مطالب مربوط به رمزنگاری در آنها پوشش داده شده است. اینکه طرح‌های رمزنگاری حال حاضر چه چیزهایی هستند و پروتکل‌های رمزنگاری چگونه بوجود آمدند مانند SSL/TSL و کارکرد آنها. مانند کتاب قبلی *بروس اشنایر، رمزنگاری کاربردی*. این چنین کتاب‌هایی به عنوان مرجعی در خدمت کسانی است که می‌خواهند با رمزنگاری کار کنند. اما کتاب‌هایی این چنین یک گام از نیازهای مهندسین رمزنگاری و امنیت را که در عمل به آن احتیاج است را در خود ندارند. مهندسین رمزنگاری و امنیت می‌خواهند بدانند که چگونه پروتکل‌های رمزنگاری حال حاضر کار می‌کنند و نحوه استفاده آنها را بدانند.

برای دانستن چگونگی استفاده از رمزنگاری، باید یاد بگیرید که چگونه مانند یک رمزنگار فکر کنید. این کتاب به شما کمک می‌کند تا به این هدف برسید. این کار را با غوطه‌ور ساختن شما در این زمینه انجام خواهیم داد. بجای اینکه بطور گسترده در مورد تمام پروتکل‌های رمزنگاری صحبت کنیم، عمیقا در طراحی و تجزیه و تحلیل خاص پروتکل‌ها شیرجه می‌زنیم. دست در دستان شما همراه شما خواهیم بود تا در مورد چگونگی طراحی پروتکل‌های رمزنگاری آشنا شویم. در مورد دلیل تصمیم‌گیری‌های طراحی که انجام داده‌ایم آنها را با شما به اشتراک خواهیم گذاشت و به مشکلات بالقوه طول مسیر اشاره خواهیم کرد.

با یادگیری اینکه چگونه همانند یک رمزنگار فکر کنید، خواهید آموخت که چگونه به یک کاربر باهوش رمزنگاری تبدیل شوید. قادر خواهید بود تا به ابزارهای رمزنگاری موجود نگاهی بیندازید و نحوه استفاده از آنها را یاد بگیرید. همچنین به درک بهتر چالش‌های موجود در رمزنگاری خواهیم پرداخت و به چگونگی غلبه بر این چالش‌ها فکر خواهیم کرد.

همچنین این کتاب به عنوان دروازه‌ای برای یادگیری امنیت کامپیوتر می‌باشد. در بسیاری جهات امنیت کامپیوتر یک مجموعه سطح بالایی از رمزنگاری است. امنیت کامپیوتر و رمزنگاری در مورد طراحی و ارزیابی اشیاء (سیستم‌ها و الگوریتم‌ها) می‌باشند تا هنگامی که دشمن حضور پیدا می‌کند به شیوه‌های خاصی رفتار کنند. در این کتاب به شما خواهیم آموخت که در زمینه رمزنگاری چگونه به دشمن فکر کنید. زمانیکه شما همانند دشمنان فکر کنید، می‌توانید همان طرز فکر را به امنیت سیستم‌های کامپیوتری اعمال کنید.

فهرست در یک نگاه

فهرست مطالب

فصل اول: مفهوم رمزنگاری

فصل دوم: مقدمه‌ای بر رمزنگاری

فصل نهم: تولید تصادفی

فصل دهم: اعداد اول

فصل یازدهم: دفی هلمن

فصل دوازدهم: RSA

فصل شانزدهم: ساعت

فصل هفدهم: سرورهای کلید

فصل بیستم: نکات PKI

فصل بیست و یکم: ذخیره رمزها

نمونه‌ای از برنامه درسی

راه‌های متعددی برای خواندی این کتاب وجود دارد و می‌توانید به عنوان یک راهنمای شخصی برای مهندسی رمزنگاری مطالعه کنید یا اینکه در طول یک ترم آنرا مطالعه کنید. می‌توانید به عنوان یک دوره فشرده ۶ هفته‌ای رمزنگاری برای امنیت کامپیوتر استفاده شود. یا اگر زمان اجازه دهد می‌تواند به عنوان یک دوره کامل ترمی رمزنگاری ارائه شود. به منظور تسهیل در استفاده از کلاس، چند برنامه درسی را در زیر ارائه می‌کنیم.

برنامه درسی زیر برای یک دوره فشرده ۶ هفته‌ای در رمزنگاری است. در این دوره ۶ هفته‌ای، فرض می‌کنیم که مطالب فصل ۱ بطور جداگانه مورد بحث قرار گرفته است.

هفته اول: فصل‌های ۲، ۳ و ۴

هفته دوم: فصل‌های ۵، ۶ و ۷

هفته سوم: فصل‌های ۸، ۹ و ۱۰

هفته چهارم: فصل‌های ۱۱، ۱۲ و ۱۳

هفته پنجم: ۱۴، ۱۵، ۱۶ و ۱۷

هفته ششم: ۱۸، ۱۹، ۲۰ و ۲۱

برنامه درسی زیر برای یک دوره ۱۰ هفته‌ای مهندسی رمزنگاری می‌باشد:

هفته اول: فصل ۱و ۲

هفته دوم: فصل‌های ۳ و ۴

هفته سوم: فصل‌های ۵ و ۶

هفته چهارم: فصل‌های ۷ و ۸

هفته پنجم: فصل‌های ۹ و ۱۰

هفته ششم: فصل‌های ۱۱ و ۱۲

هفته هفتم: فصل‌های ۱۳ و ۱۴

هفته هشتم: فصل‌های ۱۵، ۱۶ و ۱۷

هفته نهم: فصل‌های ۱۸، ۱۹ و ۲۰

هفته دهم: فصل ۲۱

برنامه درسی زیر برای یک ترم ۱۲ هفته‌ای است. همچنین در این دوره می‌توان به موارد پیشرفته‌تری از رمزنگاری و امنیت کامپیوتر در طول ترم ارائه کرد.

هفته اول: فصل ۱و ۲

هفته دوم: فصل‌های ۳ و ۴

هفته سوم: فصل‌های ۵ و ۶

هفته چهارم: فصل‌های ۷

هفته پنجم: فصل‌های ۸ و ۹

هفته ششم: ادامه فصل ۹ و ۱۰

هفته هفتم: فصل‌های ۱۱ و ۱۲

هفته هشتم: فصل‌های ۱۳ و ۱۴

هفته نهم: فصل‌های ۱۵ و ۱۶

هفته دهم: فصل‌های ۱۷ و ۱۸

هفته یازدهم: فصل‌های ۱۹ و ۲۰

هفته دوازدهم: فصل ۲۱

این کتاب انواع مختلفی از تمرینات را ارائه کرده است و خوانندگان این کتاب را تشویق می‌کنیم تا تمرینات این کتاب را حل کنند. یکسری تمرینات ابتدایی برای آزمایش درک خود از خواص فنی رمزنگاری ارائه شده‌اند. با این حال، از آنجا که هدف ما در این کتاب اینست که به شما کمک کنیم تا در مورد رمزنگاری سیستم‌های واقعی چگونه فکر کنید، یک سری تمرینات پیشرفته‌تری ارائه کردیم (بخش ۱-۱۲ را مشاهده کنید). رمزنگاری بصورت ایزوله شده نمی‌باشد، رمزنگاری تنها بخشی از یک اکوسیستم بزرگتر متشکل از سخت‌افزار و نرم‌افزار سیستم‌های دیگر، مردم، اقتصاد، اخلاق، تفاوت‌های فرهنگی، سیاست و قانون و خیلی موارد دیگر است. تمرین‌های پیشرفته ما طوری طراحی شده‌اند که به صراحت شما را مجبور می‌کند تا در مورد رمزنگاری در زمینه سیستم‌های واقعی و اکوسیستم اطراف آن فکر کنید.

تمرینات این کتاب به شما این فرصت را می‌دهند تا به صورت مستقیم با تمرین فکر کردن به سیستم واقعی توانایی خود را بالاتر ببرید. علاوه‌بر این، ترکیب و اتصال این تمرینات در سراسر این کتاب، باعث می‌شود که رشد دانش خود را در اثر مطالعه فصل به فصل این کتاب مشاهده کنید.

بخش ۱

مقدمه

فصل اول: مفهوم رمزنگاری

فصل دوم: مقدمه‌ای بر رمزنگاری

فصل
۱
مفهوم رمزنگاری

رمزنگاری عبارت است از هنر و علم رمزگذاری است. امروزه این علم بسیار گسترده شده و احراز هویت، امضاهای دیجیتالی و بسیاری از عملکردهای امنیتی پایه را در بر می‌گیرد. با اینحال، رمزنگاری هم علم و هم هنر محسوب می‌شود. برای ایجاد سیستم‌های رمزنگاری خوب به یک پیشینه علمی مناسب و به حد کافی افسونگری نیاز است که ترکیبی از تجربه و ذهنیت درست برای تفکر درباره مشکلات امنیتی است. طراحی این کتاب به گونه‌ای است که در ترویج این عوامل حیاتی به شما کمک می‌کند.

رمزنگاری زمینه بسیار متنوعی است. در یک کنفرانس پژوهشی در زمینه رمزنگاری به جبر پیشرفته، اقتصاد، فیزیک کوآنتوم، حقوق مدنی و جنایی، آمار، طراحی چیپ‌ها، بهترین بهینه‌سازی‌های نرم‌افزار، سیاست، طراحی واسط[1] کاربر و هر چیزی که میان آنها وجود دارد باید پرداخته شود. این کتاب از بین روش‌های مختلف تنها بر بخش بسیار کوچکی از رمزنگاری یعنی جنبه عملی آن متمرکز شده است. هدف ما در این کتاب آموزش نحوه پیاده‌سازی رمزنگاری در سیستم‌های واقعی است. به عبارت دیگر، این کتاب در سطح بسیار گسترده‌تر به شما کمک می‌کند تا در مهندسی امنیتی و پرورش توانایی خود برای تفکر درباره رمزنگاری و مسایل امنیتی مانند یک متخصص امنیتی تجربه کسب کنید. این درس‌های گسترده به شما کمک می‌کند با موفقیت با چالش‌های امنیتی دست و پنجه نرم کنید، خواه این چالش‌ها مستقیماً به رمزنگاری مربوط باشند یا خیر. تنوع این حوزه، رمزنگاری را تبدیل به یک زمینه جذاب کاری ساخته است. رمزنگاری در واقع ترکیبی از زمینه‌های مختلف و وسیع است. همواره چیزهای جدیدی برای آموختن وجود دارد و ایده‌های تازه از هر سو به ذهن می‌رسند و یکی از دلایل دشوار بودن رمزنگاری همین موضوع است. درک رمزنگاری به طور کامل غیرممکن است و هیچ کس در دنیا وجود ندارد که همه چیز را درباره آن بداند. حتی هیچ کس وجود ندارد که از بیشتر جوانب آن آگاه باشد. ما نیز به طور حتم همه چیز آن را نمی‌دانیم.

مقدمه عبارت است از هر آن چه که باید درباره موضوع این کتاب بدانید. بنابراین، نخستین درس رمزنگاری این است که باید ذهنی انتقادی داشته باشید. کورکورانه به چیزی اعتماد نکنید، حتی اگر آن چیز چاپ شده باشد [و مثلاً در کتاب‌ها وجود داشته باشد]. به

---

[1] Interface

زودی خواهید دید که داشتن چنین ذهن انتقادی محتوای اساسی آن چیزی را تشکیل می‌دهد که آن را «شکاکیت حرفه‌ای» می‌نامیم.

۱-۱ نقش رمزنگاری

رمزنگاری به خودی خود چیز نسبتاً بی‌فایده‌ای است و باید به عنوان بخشی از یک سیستم بسیار بزرگتر در نظر گرفته شود. می‌توان رمزنگاری را با نقشی که قفل در دنیای فیزیکی دارد مقایسه کرد. قفل به خودی خود وسیله‌ای بی‌فایده است و باید به صورت بخشی از یک سیستم بزرگتر باشد. این سیستم بزرگتر می‌تواند دری بر روی یک ساختمان یا یک رشته زنجیر، یک گاو صندوق یا چیز دیگری باشد. این سیستم بزرگتر حتی به افرادی تعمیم می‌یابد که قرار است از قفل استفاده کنند. این افراد باید به یاد داشته باشند که در را قفل کرده و کلید را در اطراف آن جا نگذراند، به صورتی که هر کس بتواند آن را پیدا کند. همین موضوع درباره رمزنگاری نیز صادق است، رمزنگاری فقط بخش کوچکی از یک سیستم امنیتی بسیار بزرگتر است.

هر چند رمزنگاری فقط بخش کوچکی از سیستم امنیتی است، اما بخش بسیار حیاتی از آن محسوب می‌شود. رمزنگاری بخشی است که باید برای برخی افراد دسترسی ایجاد کرده و از دسترسی بعضی دیگر جلوگیری کند. این کار با ترفندهای بسیاری همراه است. بیشتر بخش‌های سیستم امنیتی مانند دیوار و نرده هستند و طوری ساخته می‌شوند که دیگران نتوانند به فضای مورد نظر راه پیدا کنند. رمزنگاری در اینجا همان نقش قفل را دارد و باید بین دسترسی «خوب» و دسترسی «بد» تمایز ایجاد کند. این کار بسیار دشوارتر از این است که صرفاً به دیگران اجازه ورود به جای مورد نظر را ندهیم. بنابراین، رمزنگاری و عناصر پیرامونی آن نقطه طبیعی حمله به هر سیستم امنیتی را تشکیل می‌دهند.

این بدان معنا نیست که رمزنگاری همیشه نقطه ضعف یک سیستم است. در برخی موارد، حتی رمزنگاری بد می‌تواند بسیار بهتر از مابقی سیستم امنیتی باشد. احتمالاً در ورودی اتاق‌های بانک را در فیلم‌ها دیده باشید و فولاد سخت شده با ضخامت ۱۰ اینچ به همراه پیچ‌های بزرگ که برای محکم کردن آن به کار می‌رود را می‌شناسید. این اتاق‌ها قطعاً تأثیرگذار هستند و اغلب معادل کوچکی از چنین درهایی را در خیمه‌ها دیده‌ایم. افرادی که مقابل چنین خیمه‌ای قرار می‌گیرند به جای این که وقت خود را به نگاه کردن به خیمه صرف کنند، درباره مقدار ضخامتی که درِ آن باید داشته باشند با هم بحث می‌کنند. اغلب ساعت‌ها وقت خود را صرف پیدا کردن طول دقیق کلید سیستم‌های رمزنگاری می‌کنیم ولی به آسیب‌پذیری که از طریق سربار به یک برنامه کاربردی تحت وب وارد می‌

شود اصولا توجهی نمی‌کنیم. نتیجه این کار قابل پیش‌بینی است، مهاجمان سربار بافر را پیدا کرده و رمزنگاری هرگز مزاحمتی برای حمله آنها فراهم نخواهد کرد. رمزنگاری تنها زمانی واقعاً مفید است که مابقی سیستم نیز به صورت مؤثر در برابر حملات ایمن شده باشند.

با اینحال، دلایلی وجود دارد که نشان می‌دهد حتی در سیستم‌هایی که ضعف‌های دیگری نیز دارند، انجام درست رمزنگاری اهمیت دارد. ضعف‌های مختلف به روش‌های متفاوت برای مهاجمان مختلف مفید واقع می‌شوند. به طور مثال، شانس کمی برای ردیابی مهاجمی که در سیستم رمزنگاری رخنه می‌کند وجود دارد. هیچ مسیر حمله‌ای وجود ندارد، چون دسترسی مهاجم فقط مانند یک دسترسی «خوب» به نظر می‌رسد. این امر در دنیای واقعی با شکستن در قابل مقایسه است. اگر سارق از اهرم برای شکستن در استفاده کند، حداقل خواهید دید که شکستن دری اتفاق افتاده است. اما اگر او قفل را از جا در بیاورد، ممکن است هرگز متوجه اینکه دزدی اتفاق افتاده است نشوید. حالت‌های بسیاری در حملات وجود دارد که ردی از حمله به جا می‌ماند، یا به روش‌های مختلف مزاحمتی برای سیستم پدید می‌آید. حمله به بخش رمزنگاری ممکن است سریع و نامحسوس باشد و به مهاجم اجازه می‌دهد بارها و بارها به سیستم بازگشته و نفوذ کند.

۱- ۲ خصوصیت ضعیف‌ترین حلقه زنجیر

جمله زیر را با یک فونت بسیار بزرگ چاپ کرده و در بالای مانیتور خود بچسبانید:

«قدرت یک سیستم امنیتی فقط به اندازه ضعیف‌ترین حلقه ارتباطی آن است.»

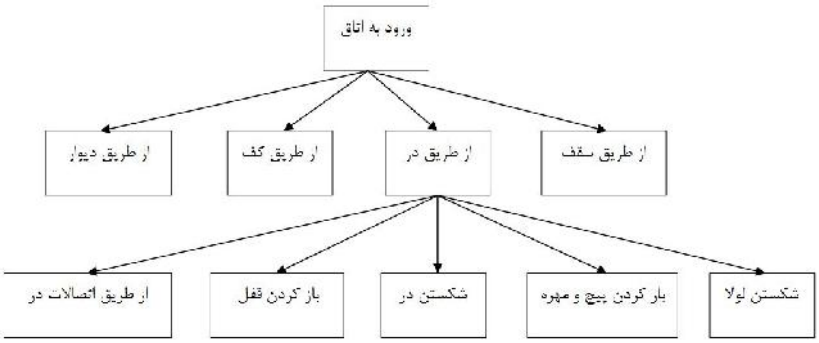
هر روز به این نوشته نگاه کنید و سعی کنید مفهوم آن را درک کنید. خصوصیت ضعیف‌ترین حلقه ارتباطی یکی از دلایل اصلی است که نشان می‌دهد چرا درست ساختن سیستم‌ها تا این حد دشوار است.

هر سیستم امنیتی شامل بخش‌های زیادی است. باید فرض کنیم که طرف مقابل ما باهوش است و قصد دارد از طریق ضعیف‌ترین بخش سیستم ما حمله خود را انجام دهد و اهمیتی ندارد بخش‌های دیگر چقدر قوی هستند. درست شبیه یک زنجیر، ضعیف‌ترین دانه قبل از همه شکسته می‌شود و مهم نیست بقیه دانه‌های موجود در زنجیره چقدر قوی باشند.

نیلز در یک ساختمان اداری کار می‌کرد که در آن تمام درها هر شب قفل می‌شد. چنین جایی بسیار امن به نظر می‌رسد، نه؟ تنها مشکل این بود که ساختمان سقف خوبی

نداشت و با کنار زدن پنل‌های سقف و بالا رفتن از هر در و دیواری می‌شد به آن نفوذ کرد. اگر پنل‌های سقف را کنار می‌زدید، کل زمین کف شبیه به مجموعه‌ای از اطاقک‌های درداری که قفل شده بودند به نظر می‌رسید. مطمئناً قفل کردن درها کار را برای سارقین بسیار سخت‌تر می‌کرد، اما این کار کنترل اتاق‌ها را نیز برای نگهبانان در طی گشت‌های شبانه‌شان دشوار می‌کرد. اصلاً مشخص نبود امنیت کلی اداره با قفل کردن درها بیشتر شده است یا کمتر. در این مثال، خصوصیت ضعیف‌ترین حلقه زنجیر باعث می‌شد قفل کردن درها چندان مؤثر نباشد. این کار ممکن بود توان یک حلقه ارتباطی خاص (در) را بیشتر کند اما حلقه دیگری وجود داشت (سقف) که هنوز ضعیف بود. اثر کلی حاصل از قفل کردن درها در بهترین حالت بسیار کم بود و اثرات جانبی منفی آن می‌توانست به خوبی تأثیرات مثبت آن را بالا ببرد.

برای بهبود امنیت یک سیستم، باید ضعیف‌ترین حلقه ارتباطی را تقویت کرد. اما برای انجام این کار، باید بدانیم حلقه‌های ارتباطی چه چیزی هستند و کدام یک از آنها ضعیف می‌باشند. این کار با استفاده از یک ساختار درختی سلسله مراتبی به خوبی انجام می‌شود. هر بخش از یک سیستم حلقه‌های چندگانه‌ای دارد و هر حلقه به نوبه خود زیر حلقه‌های ارتباطی دارد. می‌توان این حلقه‌های ارتباطی را با آنچه که آن را درخت حمله می‌نامیم سازماندهی کنیم [۱]. در شکل ۱-۱ مثالی از این مورد نشان داده شده است. در واقع می‌خواهیم به یک اتاق محافظت از اشیاء قیمتی نفوذ کنیم. حلقه‌های ارتباطی سطح اول دیوارها، کف زمین، در و سقف هستند. نفوذ از طریق هر یک از آنها ما را به اتاق می‌رساند. بیایید به طور جزئی‌تر نگاهی به در داشته باشیم. سیستم در حلقه‌های ارتباطی خاص خودش را دارد: ارتباط بین چارچوب در و دیوارها، قفل، خود در، پیچ‌هایی که در را به چارچوب در وصل کرده‌اند و لولاها. می‌توان کار را با بحث از خطوط منفرد حمله به قفل ادامه داد، یکی از آنها به کلید می‌رسد، که به نوبه خود به کل درخت منجر می‌شود که در سرقت می‌توان کلید را به طریقی به دست آورد.



شکل ۱- ۱ مثال حمله درختی به اتاق

می‌توان هر حلقه ارتباطی را تجزیه و تحلیل کرد و آن را به حلقه‌های دیگری تقسیم کرد تا به تک تک عناصر دست پیدا کنیم. انجام این کار برای یک سیستم واقعی ممکن است کار بسیار زیادی ببرد. اگر نگران مهاجمی باشیم که می‌خواهد الماس‌های نگهداری شده در اتاق را سرقت کند، در این صورت شکل ۱-۱ نیز فقط یک تکه از درخت حمله بزرگتر خواهد بود؛ یک مهاجم می‌تواند فردی را فریب دهد و از او بخواهد الماس‌ها را از اتاق خارج کرده و به محض خروج آنها را بدزدد. درخت‌های حمله با نشان دادن خطوط احتمالی حمله دید خوبی فراهم می‌کنند. تلاش برای ایمن ساختن یک سیستم بدون انجام چنین تجزیه و تحلیلی در اکثر موارد به کار بی‌فایده‌ای می‌انجامد. در این کتاب، فقط بر روی عناصر محدودی کار می‌کنیم که می‌توان آنها را از طریق رمزنگاری حل کرد و صرفاً درباره درخت‌های حمله آنها صحبت نمی‌کنیم. اما باید اطمینان داشته باشید که نحوه استفاده مهاجم از درخت را برای مطالعه یک سیستم بزرگتر و ارزیابی نقش رمزنگاری را در سیستم درک کرده‌اید.

خصوصیت ضعیف‌ترین حلقه ارتباطی به طرق مختلف بر کار ما تأثیر می‌گذارد. به طور مثال، تصور این مطلب که کاربران کلمات عبور مناسبی را در اختیار دارند چیز وسوسه-انگیزی است اما در عمل چنین نیست. آنها اغلب کلمات عبور ساده و کوتاه را انتخاب می-کنند. کاربران ممکن است هر طولی را که سیستم‌های امنیتی اجازه دهند برای کلمات عبور خود انتخاب کنند. نوشتن یک کلمه عبور بر روی یک یادداشت چسبی و چسباندن آن بر روی مانیتور فقط یکی از مواردی است که ممکن است کاربران انجام دهند. شما هرگز نمی‌توانید از چنین مسایلی چشم‌پوشی کنید چون این مسایل همواره بر نتیجه نهایی تأثیر می‌گذارند. اگر سیستمی را طراحی کرده باشید که هر هفته به کاربران یک کلمه عبور تصادفی ۱۲ رقمی بدهد، در این صورت مطمئن باشید که کاربران پسوردهای خود را بر

روی مانیتور خواهند چسباند. این همان حلقه ارتباطی ضعیف است و برای امنیت کلی سیستم بد خواهد بود.

قاطعانه می‌توان گفت همه چیز را باید قوی تعبیه کرد اما ضعیف‌ترین حلقه ارتباطی بی‌فایده است. اما در عمل، همه چیز به این روشنی و صراحت نیست. مهاجم ممکن است نداند ضعیف‌ترین حلقه کدام است و به یک حلقه ارتباطی نسبتاً قوی حمله کند. ضعیف-ترین حلقه ارتباطی ممکن است برای مهاجمان مختلف متفاوت باشد. قوت هر حلقه ارتباطی به مهارت مهاجم و ابزارها و دسترسی به سیستم بستگی دارد. حلقه ارتباطی که ممکن است مهاجم از آن سوءاستفاده کند نیز می‌تواند به اهداف مهاجم بستگی داشته باشد. بنابراین، اینکه کدام حلقه ارتباطی ضعیف‌تر است به موقعیت بستگی دارد. بنابراین تقویت هر حلقه ارتباطی که ممکن است در یک موقعیت خاص ضعیف‌ترین حلقه باشد، ارزشمند است. علاوه بر این، تقویت حلقه‌های ارتباطی چندگانه کار ارزشمندی است، به صورتی که اگر یک حلقه نتوانست کار خود را انجام دهد سایر حلقه‌ها همچنان بتوانند امنیت لازم را فراهم کنند، خصوصیتی که به عنوان دفاع در عمق[۲] شناخته می‌شود.

۱-۳ تنظیمات خصومت‌آمیز[۳]

یکی از بزرگترین تفاوت‌های میان سیستم‌های امنیتی و اکثریت قریب به اتفاق انواع دیگر مهندسی، تنظیمات خصومت‌آمیز است. بیشتر مهندسان ناچارند با مشکلاتی مانند طوفان، گرما و هواخوردگی و پارگی مقابله کنند. تمام این عوامل بر طراحی‌ها تأثیر می-گذارند، اما اثر آنها برای یک مهندس با تجربه نسبتاً قابل پیش‌بینی است. اما در سیستم-های امنیتی چنین نیست. طرف‌های مقابل ما باهوش، هوشمند، بداندیش و انحرافی هستند و کارهایی انجام می‌دهند که هیچکس فکرش را هم نمی‌کند. این افراد با قوانین پیش نمی‌روند و کاملاً غیرقابل پیش‌بینی‌اند و این باعث می‌شود محیط کار کردن بسیار دشوارتر شود. بسیاری از ما فیلمی را که در آن پل معلق تاکوما، با بادی که پی در پی می‌وزد و تلوتلو و پیچ و تاب می‌خورد و در آب سقوط می‌کند را دیده‌ایم. این قطعه فیلم بسیار مشهور است و فرو ریختن آن به مهندسین پل آموزش داده می‌شود و درسی ارزشمند به حساب می‌آید. برای پل‌های معلق باریک می‌توان حالت رزونانسی در نظر گرفت که در آن باد مداوم می‌تواند کل ساختار را به نوسان بیاندازد و نهایتاً آن را بشکند. اما چطور می‌توان

[2] Defense in depth

[3] The Adversarial Setting

مانع از وقوع چنین چیزی در پل‌های جدیدتر شد؟ قویتر ساختن پل‌ها به صورتی که بتوانند در مقابل نوسانات مقاومت کنند بسیار پرهزینه خواهد بود. متداول‌ترین روش مورد استفاده تغییر ایرودینامیک پل است. در این روش کف پل ضخیمتر ساخته می‌شود و باعث می‌شود نسبت به وزش بادی که آن را به بالا و پایین فشار می‌دهد محکم‌تر عمل کند. گاهی از نرده‌گذاری به عنوان از بین برنده این حالت استفاده می‌کنند که باعث می‌شود کف پل کمتر شبیه به بالی که در باد به بالا و پایین می‌رود عمل کند. این کار مفید واقع می-شود، چونکه باد تا حدی قابل پیش‌بینی است و رفتار آن در یک تلاش فعال برای تخریب پل تغییر نمی‌کند.

یک مهندس امنیتی باید بادهای سخت را مورد نظر قرار دهد. اما اگر باد به جای اینکه از طرفین بوزد از بالا به پایین بوزد چه؟ و اگر جهت‌های آن با یک فرکانس مطابق با رزونانس پل تغییر کند چه؟ مهندسین پل این سبک سخن گفتن را منسوخ شده می‌دانند: «ساده نباش، باد به این صورت نمی‌وزد.» چنین چیزی قطعاً کار مهندسین پل را بسیار ساده‌تر می‌سازد. رمزنگاران از چنین نعمتی برخوردار نیستند. سیستم‌های امنیتی از سوی مهاجمان باهوش و شرور مورد حمله قرار می‌گیرند و ما باید هر نوع حمله‌ای را مد نظر قرار دهیم. تنظیمات خصومت‌آمیز فضای بسیار خشنی برای کار کردن فراهم می‌کند. هیچ قانونی در این بازی وجود ندارد و شبیه به این است که کف پل در اینجا بر خلاف نظر ما عمل کند. ما از «مهاجم» در یک حس انتزاعی صحبت می‌کنیم و نمی‌دانیم او کیست، از چه چیزهایی خبر دارد، هدف او چیست، کی حمله می‌کند یا منابع او چه چیزهایی هستند. از آنجا که حمله ممکن است مدت طولانی پس از طراحی سیستم اتفاق بیفتد، مهاجم از مزیت پنج یا ده سال تحقیق بیشتر برخوردار است و می‌تواند از فناوری آتی استفاده کند که اکنون در دسترس ما قرار ندارد؛ و با برخورداری از تمام این مزایا، مهاجم می‌تواند تنها یک نقطه ضعف در سیستم ما پیدا کند، در حالی که ما مجبوریم از تمام حوزه‌های سیستم خود محافظت کنیم. با اینحال، مأموریت ما ساختن سیستمی است که بتواند همه این حملات را تحمل کند و این یک عدم تعادلی اساسی بین مهاجم به سیستم و مدافع آن پدید می‌آورد و این همان چیزی است که دنیای رمزنگاری را بسیار جذاب می‌کند.

۱-۴ شکاکیت حرفه‌ای

برای کار کردن در این زمینه، باید خود را [همراه با ذهنیت] انحرافی تربیت کنید. شما ناچارید شبیه به یک مهاجم شرور برای یافتن ضعف‌های کار خود فکر کنید. این امر بر مابقی زندگی شما نیز تأثیر می‌گذارد. کسانی که بر روی سیستم‌های رمزنگاری کار می‌کنند

این موضوع را تجربه کرده‌اند. به محض این که شروع به فکر کردن درباره نحوه حمله به سیستم می‌کنید این نحوه تفکر را بر سایر چیزهای دور و بر خود نیز اعمال خواهید کرد و ناگهان متوجه می‌شوید که چقدر می‌توانید مردم را فریب دهید و آنها چقدر قادر به فریب شما هستند. رمزنگاران شکاک‌های حرفه‌ای هستند. البته باید شکاکیت حرفه‌ای خود را از زندگی واقعی‌تان جدا کنید به صورتی که کاملاً هم در حالت دیوانه‌واری قرار نگیرید. اغلب ما به صورتی این کار را مدیریت می‌کنیم که عاقل بودن ما محفوظ بماند و روی موضوعات درست فکر کنیم.[۴] در واقع فکر می‌کنیم این شکاکیت حرفه‌ای تفریح زیادی به همراه می‌آورد. پرورش این تنظیم ذهنی به شما در مشاهده اشیاء مربوط به سیستم‌ها و محیط اطراف شما کمک می‌کند که بیشتر افراد دیگر به آن توجه نمی‌کنند.

شکاکیت در کار ما بسیار مفید واقع می‌شود. فرض کنید بر روی یک سیستم پرداخت الکترونیکی کار می‌کنید. چندین طرف در این سیستم دخیل هستند، از جمله مشتری، فروشنده، بانک مشتری و بانک فروشنده. محاسبه اینکه رفتارها به چه صورت خواهند بود کار دشواری است، بنابراین ما از مدل شکاکیت استفاده می‌کنیم. برای هر طرف فرض می‌کنیم که سایر طرف‌ها بخشی از یک دسیسه و توطئه برای کلاهبرداری از آن طرف هستند و ما نیز فرض می‌کنیم که مهاجم ممکن است اهداف دیگری داشته باشد، مانند افشاء محرمانه بودن تراکنش‌های یک طرف یا از بین بردن دسترسی یک طرف به سیستم در یک زمان حیاتی. در صورتی که سیستم رمزنگاری شما بتواند مدل شکاکیت را باقی نگه دارد، دست کم شانس مبارزه‌ای برای بقا در دنیای واقعی را دارد. ما به همین صورت مبادله‌ای به شکاکیت حرفه‌ای و مدل شکاکیت را به عنوان ذهنیت امنیتی رجوع می‌کنیم.

۱-۴-۱ منافع بیشتر

به محض اینکه حس شکاکیت حرفه‌ای را پرورش دادید، دیگر هرگز به سیستم‌ها به یک صورت نگاه نخواهید کرد. این تنظیم ذهنی از طریق حرفه‌تان به شما سود می‌رساند، صرف نظر از اینکه تبدیل به یک رمزنگار بشوید یا خیر. حتی در صورتی که رمزنگار هم نشوید، ممکن است روزی ببینید که خودتان بر روی یک طراحی، پیاده‌سازی یا ارزیابی نرم‌افزار یا سخت‌افزار کامپیوتری جدید سیستم‌ها کار می‌کنید. اگر ذهنیت امنیتی داشته باشید، مستمراً درباره آنچه که مهاجم ممکن است با سیستم شما بکند فکر خواهید کرد.

[۴] – اما به خاطر داشته باشید این واقعیت که شما شکاک نیستید به این معنا نیست که آنها هم به دنبال افشاء سیستم شما نیستند.

این موقعیت به صورت جالبی شما را در موضعی قرار می‌دهد که به ناچار مشکلات امنیتی بالقوه این سیستم‌ها را به زودی شناسایی کنید. ممکن است خودتان همیشه قادر به برطرف کردن تمام مشکلات امنیتی نباشید اما در همین حد هم خوب است. مهمترین چیز درک مشکلات امنیتی است که وجود دارد. وقتی این کار را انجام دادید، یافتن کسانی که بتوانند مشکلات را برطرف کنند کار سختی نخواهد بود. اما بدون ذهنیت امنیتی، ممکن است هیچ گاه متوجه نشوید که سیستم شما مشکلات امنیتی دارد و در نتیجه به وضوح نتوانید از آن در برابر این مشکلات به یک روش اساسی محافظت کنید.

فناوری‌ها بسیار سریع تغییر می‌کنند. این بدان معنا است که برخی مکانیزم‌های امنیتی مطرح امروزی ممکن است در طی ۱۰ تا ۱۵ سال آیند از رده خارج شده باشند. اما اگر بتوانید چگونه فکر کردن درباره مسایل امنیتی را یاد بگیرید و حس قدردانی نسبت به طرف‌های مقابل داشته باشید، در این صورت می‌توانید ذهنیت امنیتی را با مابقی زندگی خود همراه کرده و آن را در فناوری‌های جدیدی که با آن سر و کار دارید اعمال کنید.

۱-۴-۲ بحث درباره حملات

شکاکیت حرفه‌ای ابزار اساسی در تجارت است. در هر سیستم جدیدی که با آن مواجه هستید، نخستین چیزی که باید به آن فکر کنید این است که چگونه می‌توان به آن سیستم نفوذ کرد. هر چه زودتر نقطه ضعف سیستم را بیابید، زودتر هم سیستم جدید را یاد می-گیرید. هیچ چیز بدتر از این نیست که سال‌ها بر روی یک سیستم کار کنید و بعد فقط یک نفر بیاید و بگوید: «اما اگر من از این طریق به این سیستم حمله کنم چه؟» واقعیت این است که اصلاً دوست ندارید چنین لحظه تأسف برانگیزی را تجربه کنید.

در این زمینه، ما تفکیک بسیار جدی بین حمله کردن به کار یک نفر و حمله شخصی به یک نفر قایل می‌شویم. هر کاری یک بازی بی‌طرفانه است. اگر کسی چیزی را پیشنهاد کند، یک دعوت خودکار برای حمله به آن خواهد بود. در صورتی که شما به یکی از سیستم‌های ما نفوذ کنید، ما این حمله را ستایش کرده و به همه خبر می‌دهیم که چنین چیزی رخ داده است.[۵] ما همواره به دنبال ضعف‌های موجود در هر سیستمی هستیم چون این تنها راه برای یاد گرفتن نحوه ایمن‌ترسازی سیستم‌ها است. چیزی که باید بیاموزید این است که یک حمله به کار شما لزوماً به معنای حمله به شخص شما نیست. همچنین، وقتی

[۵] - بسته به حمله، ممکن است خودمان را برای نیافتن ضعف‌هایمان سرزنش کنیم و این مسأله دیگری است.

شما به یک سیستم حمله می‌کنید همیشه مطمئن هستید که در حال انتقاد از سیستم هستید و نه طراحان آن. حملات شخصی در رمزنگاری با واکنش منفی مشابه از هر طرف دیگر مواجه خواهند شد. اما بدانید که این پذیرش حملات ممکن است به دیگرانی که بر روی یک سیستم کار می‌کنند تعمیم پیدا نکند، به خصوص در صورتی که آنها با حوزه رمزنگاری و امنیت در کامپیوتر آشنا نباشند. بدون کسب تجربه در جامعه امنیتی، انتقاد از کار دیگران به عنوان حمله شخصی با تمام مشکلات حاصل از آن کار آسانی است. بنابراین پرورش یک رویکرد دیپلماتیک اهمیت دارد، حتی در صورتی که این رویکرد به دست آوردن پیام را دشوار سازد. حتی در صورتی که طراحی پایه اساساً معیوب باشد، مبهم عمل کردن و گفتن چیزهایی از قبیل «ممکن است ازجنبه‌های امنیتی مسایلی وجود داشته باشد» ممکن است حاصلی نداشته باشد، چون این کار با واکنش غیرصریحی مانند «بله، آن را برطرف خواهیم کرد» مواجه می‌شود. تجربه نشان داده است بهترین راه برای یافتن پیام به صورت فنی بیان معایب به صورت خاص و گفتن چیزی شبیه به این است: «اگر این و این را انجام دهید، مهاجم می‌تواند این کار را بکند.» اما چنین عبارتی ممکن است از طرف گیرنده آن ملایم به نظر نرسد. در عوض، می‌توان با این جمله شروع کرد: «اگر کسی این کار را بکند، به نظر شما چه اتفاقی می‌افتد؟». پس از آن می‌توان طراحان سیستم را در بحث درباره خود حمله آزاد گذاشت. همچنین ممکن است تعریف آنها را درباره توان باقیمانده سیستم‌شان مد نظر قرار داده، چالش‌های موجود برای ساختن سیستم‌های امن را زیر نظر بگیرید و به آنها پیشنهاد کمک به برطرف کردن مشکلات امنیتی را در صورت وجود ارائه دهید.

بنابراین، بار دیگر که کسی به امنیت سیستم شما حمله‌ای وارد کرد، سعی کنید آن را حمله‌ای شخصی در نظر نگیرید و مطمئن باشید که وقتی شما به یک سیستم حمله می‌کنید فقط بر روی فناوری تمرکز دارید، در پی انتقاد از افرادی که در پس آن قرار دارند نیستید و نسبت به این واقعیت که طراحان ممکن است با فرهنگ انتقاد سازنده در جامعه امنیتی آشنا نباشند حساس هستید.

۱-۵ مدل تهدید

هر سیستم ممکن است مورد حمله قرار گیرد. چیزی به عنوان امنیت کامل وجود ندارد. نقطه کلی یک سیستم امنیتی فراهم ساختن دسترسی برای برخی افراد و اجازه دسترسی ندادن به دیگران است. نهایتاً همواره ناچارید به نحوی به عده‌ای اعتماد کنید و همین افراد ممکن است به سیستم شما حمله کنند.

آگاهی از اینکه چه تلاشی برای حفاظت از سیستم خود می‌کنید و تلاش شما برای حفاظت در مقابل چه کسانی است اهمیت بسیاری دارد؛ و اینکه چه چیز ارزش و اهمیت دارد و تهدیدها چه هستند. اینها شبیه به سؤالات ساده‌ای هستند اما مشکلات بسیار جدی‌تر از آنچه فکر می‌کنید به وجود می‌آورند. از آنجا که این سؤالات واقعاً چیزی جز امنیت کامل نیستند، وقتی می‌گوییم یک سیستم «امن» است، در واقع می‌خواهیم بگوییم این سیستم سطح امنیت کافی برای دارایی‌های مورد علاقه ما در مقابل طبقات خاصی از تهدیدات فراهم می‌سازد. در اینجا باید امنیت یک سیستم را تحت مدل تهدید طراحی شده ارزیابی کنیم.

بیشتر شرکت‌ها از LAN خود با استفاده از فایروال محافظت می‌کنند، اما بسیاری از حملات آسیب‌زا از طرف افراد داخلی و خودی انجام می‌شوند و فایروال به طور کامل در برابر افراد خودی محافظتی از سیستم به عمل نمی‌آورد. مهم نیست فایروال شما چقدر خوب کار می‌کند، مهم این است که در برابر کارمندان شرور قادر به محافظت از سیستم نیست و این در مدل تهدید ناجور و نامناسب محسوب می‌شود.

مثال دیگر SET است که پروتکلی برای فروش آنلاین با کارت اعتباری است. یکی از ویژگی‌های این پروتکل این است که شماره کارت اعتباری را رمزگذاری می‌کند به صورتی که فرد با استراق سمع نیز نمی‌تواند آن را کپی کند و این ایده خوبی است. ویژگی دوم این است که این شماره حتی برای فروشنده هم قابل مشاهده نباشد. این ایده به خوبی روش اول کار نمی‌کند.

خصوصیت دوم موفق عمل نمی‌کند چون برخی فروشندگان از شماره کارت اعتباری برای پیدا کردن سوابق مشتری یا هزینه کردن اضافه بها استفاده می‌کنند. سرتاسر سیستم-های تجاری بر این فرض بنا شده‌اند که فروشنده به شماره کارت اعتباری مشتری دسترسی دارد و در این شرایط SET تلاش می‌کند این دسترسی را حذف کند. وقتی *نیلز* در گذشته با SET کار می‌کرد، ایده‌ای برای ارسال دوباره شماره کارت اعتباری وجود داشت که یک بار برای بانک رمزگذاری شود و یک بار برای فروشنده به صورتی که فروشنده نیز بتواند آن را دریافت کند. (البته ما بررسی نکرده‌ایم این وضعیت هنوز هم وجود دارد یا خیر).

اما با به کار بردن این ایده، SET کل مشکلات را حل نمی‌کند. بیشتر شماره‌های کارت-های اعتباری که به سرقت می‌روند در حین انتقال بین مشتری و فروشنده سرقت می‌شوند و از پایگاه داده‌های فروشنده به سرقت می‌روند و SET فقط از اطلاعات در حین انتقال محافظت می‌کند.

SET مشکل جدی‌تری به وجود می‌آورد. سالها قبل بانک *نیلز* در هلند یک SET ارائه کرد که کارت اعتباری را فعال می‌کرد. امنیت ارتقاء یافته برای فروش‌های آنلاین یکی از نقاط فروش اصلی بود. اما این عمل، باگی را تولید می‌کرد. این روش برای سفارش آنلاین با یک کارت اعتباری معمولی کاملاً امن بود. شماره کارت اعتباری شما شماره سری نیست و شما آن را به هر فروشنده‌ای که از او خرید کنید ارائه می‌کنید. آنچه که واقعاً سری است امضای شماست و همان چیزی است که احراز هویت تراکنش را انجام می‌دهد. در صورتی که فروشنده شماره کارت اعتباری شما را افشا کند، می‌تواند هزینه‌های نامعتبری را بر شما تحمیل کند، اما تا زمانی که هیچ امضای دست نویسی (یا کد PIN) وجود نداشته باشد هیچ شاخصی از پذیرش تراکنش وجود نخواهد داشت و در نتیجه هیچ پایه و اساس قانونی برای هزینه اعمال شده وجود ندارد. در اغلب حوزه‌های قضایی به راحتی می‌توان شکایت خود را مطرح کرده و پول خود را دریافت کنید. با استفاده از SET، شرایط فرق خواهد کرد. در SET از یک امضای دیجیتال (که در فصل ۱۲ تشریح شد) توسط کاربر برای احراز درستی تراکنش استفاده می‌شود. این کار نسبت به استفاده از شماره کارت اعتباری امن‌تر است. اما فکر کنید اکنون خود کاربر مسئول هر تراکنش ارائه شده توسط نرم‌افزار SET بر روی PC خودش است. این امر قابلیت‌های زیادی را برای کاربر فراهم می‌سازد. اما اگر کامپیوتر شخصی به ویروس آلوده شود و نرم‌افزار SET را از کار بیاندازد چه؟ این نرم‌افزار ممکن است تراکنش اشتباهی را امضا کرده و باعث از دست رفتن پول کاربر شود.

بنابراین از دیدگاه کاربر، SET امنیت بدتری نسبت به کارت اعتباری ارائه می‌کند. کارت‌های اعتباری اصلی در فروش آنلاین امن محسوب می‌شوند چون کاربر همیشه می‌تواند پول خود را از تراکنش جعلی دوباره پس بگیرد. استفاده از SET افشاء کاربر را محتمل‌تر می‌سازد. بنابراین هر چند سیستم پرداخت کلی با امنیت بیشتری ساخته شده است، اما SET خطر باقیمانده‌ای را از فروشنده و یا بانک به کاربر منتقل می‌کند. این روش مدل تهدید کاربر را از «در صورتی پولم را از دست می‌دهم که امضای مرا به خوبی جعل کنند» به «در صورتی پولم را از دست می‌دهم که امضای مرا به خوبی جعل کنند یا ویروس قوی کامپیوتر شخصی‌ام را آلوده کند.» تغییر می‌دهد.

مدل تهدید مهم است. به هنگام آغاز کار روی پروژه‌های امنیت رمزنگاری، بنشینید و به این فکر کنید که دارایی‌هایتان چه هستند و در مقابل چه تهدیدهایی می‌خواهید از آنها محافظت کنید. وقوع یک اشتباه در تجزیه و تحلیل تهدیدهایی که با آن روبرو هستید می‌تواند کل پروژه را بی‌اعتبار کند. در این کتاب درباره تجزیه و تحلیل تهدید زیاد صحبت

نمی‌کنیم، چون در اینجا درباره حوزه محدود رمزنگاری بحث می‌کنیم، اما در هیچ سیستم واقعی نباید هرگز تجزیه و تحلیل تهدید را برای هر طرف از نظر دور نگه دارید.

۱-۶ رمزنگاری راه حل نیست

رمزنگاری راه حل مشکلات امنیتی شما نیست. البته ممکن است بخشی از راه حل یا بخشی از مسأله باشد. در برخی موقعیت‌ها، رمزنگاری با بدتر کردن مشکل آغاز می‌شود و به هیچ وجه معلوم نیست که استفاده از آن پیشرفتی به وجود آورد. بنابراین، استفاده درست از رمزنگاری باید با دقت مد نظر قرار گیرد. بحث قبلی ما درباره SET مثالی از این مورد بود.

فرض کنید فایل سرّی بر روی کامپیوتر شخصی خود دارید که نمی‌خواهید دیگران آن را بخوانند. در این حالت می‌توانید از سیستم فایل خود در برابر دسترسی مجاز محافظت کنید یا می‌توانید فایل را رمزگذاری کرده و از کلید محافظت کنید. این فایل اکنون رمزگذاری شده و انسان طبعتا کنجکاو است که بفهمد این فایل چیست و ممکن است نتوانید به صورت خوبی از آن محافظت کنید. ممکن است آن را بر روی مموری USB نگهداری کنید و نگران از بین رفتن یا سرقت مموری USB نیز نباشید. اما کلید را کجا می‌توان ذخیره کرد؟ کلید خوب کلیدی است که آنقدر طولانی باشد که نتوان آن را به خاطر سپرد. برخی برنامه‌ها کلید را بر روی دیسک ذخیره می‌کنند، یعنی مکانی که فایل سرّی در اولین محل ذخیره شده است. اما حمله‌ای که بتواند فایل سرّی را در اولین موقعیت بازیابی کند کلید را نیز می‌تواند بازیابی کند که به نوبه خود می‌توان از آن برای رمزگشایی فایل استفاده کرد. علاوه بر این، نقطه جدید از حمله را نیز معرفی کردیم و آن این بود که اگر سیستم رمزگذاری ناامن باشد یا مقدار تصادفی بودن در کلید بسیار پایین باشد، در این صورت مهاجم می‌تواند به سیستم رمزگذاری نفوذ کند و نهایتاً امنیت کلی کاهش می‌یابد. در نتیجه، تمام راه حل صرفاً رمزگذاریِ فایل نیست. این کار ممکن است بخشی از راه حل باشد اما به خودی خود ممکن است مسایل بیشتری ایجاد کند که باید آن مسایل را نیز حل کرد.

رمزنگاری کاربردهای بسیاری دارد و بخش حیاتی بسیاری از سیستم‌های امنیتی خوب محسوب می‌شود و اگر به صورت خوب و درستی انجام نشود ممکن است سیستم را ضعیف‌تر نیز بکند. در بسیاری از موقعیت‌ها، رمزگذاری فقط احساسی از امنیت ایجاد می‌کند و نه امنیت واقعی. رمزگذاری تلاشی برای متوقف کردن است و این همان چیزی است که اغلب کاربران می‌خواهند، یعنی احساس امنیت. استفاده از رمزگذاری حتی در صورتی

که سیستم حاصل واقعاً امن هم نباشد، به روش‌های مختلف می‌تواند سیستم را با استانداردها و قوانین معین وفق دهد. در موقعیتی شبیه به این (که همگی بسیار رایج هستند)، هر افسونی که مشتری بدان معتقد باشد، احساس مشابهی از امنیت را ایجاد خواهد کرد و به همین خوبی عمل خواهد کرد.

۱-۷ رمزنگاری بسیار دشوار است

رمزنگاری بسیار سخت است. حتی سیستم‌هایی که طراحی تخصصی مناسبی دارند چند سالی نمی‌گذرد که می‌توان در آنها نفوذ کرد و معمولاً از وقوع چنین اتفاقاتی تعجب نمی‌کنیم. خصوصیت ضعیف‌ترین حلقه ارتباطی و تنظیمات خصومت‌آمیز زندگی را برای رمزنگاران یا هر مهندس امنیتی بسیار دشوار ساخته است. مشکل قابل ملاحظه دیگر کمبود تست کردن است. هیچ روش شناخته شده‌ای برای تست کردن و اطمینان از اینکه آیا یک سیستم امن است یا خیر وجود ندارد. به طور مثال، در انجمن تحقیقات رمزنگاری آنچه که سعی می‌کنیم انجام دهیم انتشار سیستم‌هایمان و فراهم کردن امکان مشاهده آنها برای دیگر متخصصان است. توجه داشته باشید که بخش دوم به طور خودکار رخ نمی‌دهد، سیستم‌های منتشر شده بسیاری وجود دارند که هیچ کس پس از انتشار حتی نگاهی هم به آنها نیانداخته است و فرآیند کنفرانس و بازبینی مجلات به تنهایی برای شناسایی اولیه تمام مسایل امنیتی بالقوه با یک سیستم قبل از انتشار آن کافی نیست. حتی با بررسی‌های دقیقی که بر روی این سیستم‌ها انجام می‌شود، معایب امنیتی باز هم ممکن است سال‌ها از دید متخصصین مخفی باقی بماند.

برخی حوزه‌های کوچک در رمزنگاری وجود دارد که ما به عنوان یک انجمن تا حد خوبی آنها را درک می‌کنیم. البته این به معنای ساده بودن این حوزه‌ها نیست. بلکه به این معنا است که ما چندین دهه بر روی آنها کار کرده‌ایم و تصور می‌کنیم از مسایل اساسی آنها آگاهی داریم. این کتاب بیشتر به همین حوزه‌ها می‌پردازد. آنچه که سعی کرده‌ایم در این کتاب انجام دهیم گردآوری اطلاعاتی درباره طراحی و ساخت سیستم‌های رمزنگاری عملی و جمع کردن آنها در یک محل است.

بنا به دلایلی به نظر می‌رسد بسیاری از افراد هنوز فکر می‌کنند رمزنگاری کار ساده‌ای است، در صورتی که اینطور نیست. این کتاب به شما در درک چالش‌های موجود بر سر راه مهندسی رمزنگاری کمک کرده و به شما کمک می‌کند تا بر این چالش‌ها فائق آیید. اما دست از کار نکشید و سعی کنید ماشین رمزنگاری جدید یا دیگر سیستم‌های اساسی امنیتی خود را به خوبی بسازید. در عوض، از آنچه که در اینجا یاد می‌گیرید استفاده کرده و

با دیگران، به خصوص متخصصان رمزنگاری باتجربه، دست به کار شوید و سیستم جدید خود را طراحی و تجزیه و تحلیل کنید. حتی ما با وجود سال‌ها تجربه در رمزنگاری و امنیت، از دیگر متخصصان امنیتی برای بازبینی سیستم‌هایی که طراحی می‌کنیم سؤال می‌کنیم.

۱-۸ رمزنگاری بخش آسانی است

با وجود اینکه رمزنگاری به خودی خود کار دشواری است، اما هنوز یکی از بخش‌های آسان یک سیستم امنیتی به حساب می‌آید. یک عنصر رمزنگاری، همچون یک قفل، مرزها و الزامات نسبتاً تعریف شده خود را دارد. تعریف یک سیستم کاملاً امنیتی به صورت واضح کار بسیار دشواری است، چون جنبه‌های بسیاری را شامل می‌شود. برخورد با مسایلی مانند فرآیندهای سازمانی برای اعطاء دسترسی و فرآیندهای استفاده شده برای کنترل که دیگر فرآیندها آنها را دنبال می‌کنند بسیار دشوار است، چرا که موقعیت همیشه در حال تغییر است. مشکل بزرگ دیگر در امنیت کامپیوتر کیفیت بسیاری از نرم‌افزارها است. اگر نرم‌افزار بر روی ماشینی نصب شده باشد که باگ‌های متعددی داشته باشد که به چاله‌های امنیتی منجر می‌شوند، نرم‌افزار امنیتی دیگر کارساز نخواهد بود.

رمزنگاری بخش آسانی است، چون افرادی وجود دارند که از نحوه انجام یک کار خوب و قابل قبول آگاهی دارند. متخصصانی را می‌توان برای طراحی یک سیستم رمزنگاری برای شما به کار گرفت. استخدام این افراد هزینه کمی در بر ندارد و اغلب از کار کردن با این سیستم‌ها به زحمت می‌افتند. آنها بخش‌های دیگری از سیستم را تغییر می‌دهند تا به خصوصیات امنیتی مطلوب دست پیدا کنند. با اینحال، بنا به دلایل عملی مختلفی که وجود دارد، رمزنگاری مشکلاتی را به وجود می‌آورد که می‌دانیم چطور باید آنها را حل کرد و این کتاب دیدی برای حل این مسایل به شما می‌دهد. مابقی سیستم امنیتی شامل مشکلاتی است که نمی‌دانیم چطور باید آنها را حل کرد. مدیریت و ذخیره کلید برای هر سیستم رمزنگاری حیاتی است، اما بیشتر کامپیوترها هیچ جای امنی برای ذخیره کلید ندارند. کیفیت پایین نرم‌افزار نیز مشکل دیگری است. امنیت شبکه نیز به مراتب سخت‌تر است و وقتی کاربران را به این مجموعه اضافه کنیم، مشکل باز هم سخت‌تر خواهد شد.

۱-۹ حملات عمومی

درک اینکه برخی مشکلات امنیتی را نمی‌توان حل کرد امر مهمی است. جعبه سیاه یا حملات عمومی بر ضد انواع مختلف سیستم‌ها وجود دارند. یک مثال کلاسیک از این مورد

حفره آنالوگ[۶] در سیستم‌های مدیریت حقوق دیجیتال[۷] (DRM) است. این سیستم‌های DRM سعی می‌کنند نسخه‌برداری از مواد دیجیتالی، مانند عکس، آواز، فیلم یا کتاب را کنترل کنند. اما هیچ فناوری، و در غیر این صورت هیچ رمزنگاری نمی‌تواند در برابر یک حمله عمومی از سیستم محافظت کند. به طور مثال، یک مهاجم می‌تواند عکسی از یک صفحه نمایش کامپیوتر بگیرد و نسخه‌ای از عکس ایجاد کند یا از یک میکروفن برای ثبت دوباره آواز استفاده کند.

شناسایی اینکه حملات عمومی بر ضد سیستم چه حملاتی هستند اهمیت خاصی دارد. در غیر این صورت، ممکن است وقت زیادی را برای برطرف کردن یک مشکل لاینحل صرف کنید. به همین صورت، وقتی یک نفر ادعا می‌کند که سیستمی را در برابر یک حمله عمومی ایمن ساخته است، باید نسبت به گفته او مشکوک بود.

۱-۱۰ امنیت و سایر معیارهای طراحی

امنیت هرگز تنها معیار طراحی برای یک سیستم نیست، بلکه یکی از این معیارها محسوب می‌شود.

۱-۱۰-۱ امنیت در مقابل عملکرد

پل ساخته شده بر روی «فیرث آو فورث[۸]» در اسکاتلند را باید اعجاز مهندسی قرن نوزدهم دانست که در مقایسه با قطارهایی که بر روی آن حرکت می‌کنند، به طرز شگفت-آوری بزرگ (و در نتیجه پرهزینه) است. این سازه به طرز باورنکردنی فراتر از مهندسی ساخته شده که به چشمان خود نمی‌توانید اعتماد کنید. در عین حال که طراحان کار خود را به خوبی انجام داده‌اند، اما با مشکلاتی مواجه بوده‌اند که پیش از آن به خوبی حل نشده بودند و آن عبارت است از ساختن یک پل فولادی بزرگ. این مهندسان کار خود را به صورت حیرت‌آوری خوب انجام داده‌اند و به طرز تماشایی در انجام این کار موفق شده‌اند و پل آنها امروزه هم هنوز پس از گذشت یک قرن مورد استفاده قرار می‌گیرد. به همین دلیل است که مهندسی آنها خوب به نظر می‌رسد.

---

[6] Analog hole

[7] Digital rights management

[8] The Firth of Forth

در طی سال‌ها طراحان پل به خوبی فرا گرفته‌اند چگونه پل‌هایی بسازند که کم‌هزینه‌تر و کارآمدتر باشند. اما اولویت اول همیشه ساختن پلی بوده است که امن بوده و به خوبی کار کند. بازدهی در قالب کاهش هزینه دومین مسأله است.

تا حد زیادی این اولویت را در صنعت کامپیوتر مطرح کرده‌ایم. هدف مقدماتی طراحی اغلب شامل تقاضاهای بازدهی بسیار بالا است. نخستین اولویت همیشه سرعت است، حتی در زمینه‌هایی که در آنها سرعت اهمیتی ندارد. در اینجا سرعت را می‌توان سرعت خود سیستم در نظر گرفت یا سرعتی که سیستم به فروش می‌رسد. این سرعت باعث کاهش هزینه‌های امنیتی می‌شود. نتیجه عموماً سیستمی است که تا حدی کارآمد است، با اینحال بقدر کافی امن نیست.

جنبه دیگری نیز در داستان پل «فیرث آو فورث» وجود دارد. در سال ۱۸۷۸، *توماس باخ* دومین پل طولانی جهان را بین «فرث آو تای[۹]» در *دوندی* ساخت. *باخ* از طراحی جدیدی استفاده کرده بود که در آن چدن و آهن فرفورژه با هم ترکیب شده و این پل را به صورت شاهکاری از مهندسی مطرح کرده بود. در شب ۲۸ دسامبر ۱۸۷۹، کمتر از دو سال بعد، این پل در یک طوفان عظیم در حالی که قطاری با ۷۵ نفر مسافر از روی آن عبور می‌کرد فرو ریخت و تمام مسافران کشته شدند و این یک فاجعه مهندسی در آن زمان محسوب می‌شد.[۱۰] بنابراین وقتی پل «فیرث آو فورث» چند سال بعد طراحی می‌شد، طراحان نه تنها برای امن‌تر ساختن آن بلکه برای اینکه در انظار عموم هم امن‌تر جلوه کند، فولاد بیشتری در بدنه آن به کار بردند. همه می‌دانیم که مهندسان گاهی طراحی‌های اشتباهی انجام می‌دهند، به ویژه وقتی چیز جدید می‌سازند و وقتی اشتباهی مرتکب می‌شوند، اتفاقات بد نیز رخ می‌دهند. اما در اینجا درس خوبی از مهندسان ویکتوریایی می‌گیریم و آن اینکه اگر چیزی که ساخته‌ای درست عمل نکرد، برگرد و به صورت سازنده‌تری عمل کن. صنعت کامپیوتر تا حد زیادی این درس را از یاد برده است. وقتی شکست‌های امنیتی بسیار جدی در سیستم‌های کامپیوتر خود داریم و به دفعات زیاد با آنها روبرو می‌شویم، کار سختی نیست که به شدت بر روی آن کار کنیم و آنها را به عنوان چیزهایی که در این سیستم مقدر شده بپذیریم. به ندرت پیش می‌آید که به تخته طراحی برگردیم و

---

[9] The Firth of Tay

[۱۰] – ویلیام مک گوناگال شعر معروفی درباره فاجعه پل تای سروده است که با این ابیات تمام می‌شود: «خانه‌های‌مان را محکمتر می‌سازیم/ تا شانس کشته شدن‌مان کمتر باشد.» این توصیه همچنان به قوت خود باقی است.

چیزی را طراحی کنیم که سازنده‌تر باشد. ما فقط وصله‌هایی به اشکالات قبلی می‌زنیم و امیدواریم مشکل به این ترتیب برطرف شود.

اکنون کاملاً برایتان واضح است که هر بار امنیت را بر بازدهی برتری می‌دهیم. اما زمانی را که CPU بر روی امنیت صرف می‌کند، چقدر است؟ پاسخ این است: تقریباً تمام زمان خود را. در واقع در صورتی که سیستم جایگزین سریعتر اما ناامن باشد، برای ما مهم نیست که ۹۰٪ چرخه‌های CPUما بر روی یک سیستم امنیتی قابل اعتماد صرف شود. کمبود امنیت در کامپیوتر علت تأخیر واقعی برای ما و اغلب کاربران است. به همین دلیل است که هنوز مردم مجبورند تکه کاغذهایی را با امضا ارسال کنند و به همین خاطر است که باید از ویروس‌ها و دیگر حملاتی که متوجه کامپیوترهایشان است نگران باشند. کلاهبرداران دیجیتالی آینده چیزهای بیشتری می‌دانند و بهتر تجهیز شده‌اند و امنیت کامپیوتر مشکلی بزرگ و بزرگتر می‌شود. ما هنوز شاهد شروع موج جنایی دیجیتال هستیم و باید کامپیوترهای خود را بسیار بهتر از این ایمن کنیم.

مطمئناً راه‌های بسیاری برای دستیابی به امنیت وجود دارد اما همچنان که *بروس* به صورت گسترده در مستند «اسرار و دروغ‌ها» نشان داده است، امنیت خوب همیشه ترکیبی از جلوگیری، ردیابی و واکنش است [۲]. نقش موجود برای رمزنگاری در وهله اول به بخش جلوگیری مربوط می‌شود که باید بسیار خوب عمل کند تا مطمئن شویم بخش‌های ردیابی و واکنش (که می‌تواند و باید شامل مداخله دستی باشد) در هم نشکنند. با اینحال، می‌توان از رمزنگاری برای ارائه مکانیزم‌های امن‌تر مانند پیگیری[11] رمزگذاری قوی‌تر استفاده کرد. رمزنگاری همان چیزی است که این کتاب درباره آن بحث می‌کند، بنابراین بر روی همین جنبه تمرکز می‌کنیم.

بله، می‌دانیم که ۹۰٪ هنوز هم درصد بالایی است اما راه‌های امیدی نیز وجود دارد. نخست به خاطر بیاورید اگر مورد جایگزین، یک سیستم ناامن باشد، در این حالت می‌خواهیم ۹۰٪ از زمان CPU ما صرف امنیت شود. خوشبختانه، در بسیاری از موارد هزینه‌های امنیت را می‌توان از کاربر پنهان کرد. ما فقط می‌توانیم حدود ۱۰ کاراکتر در ثانیه تایپ کنیم (در شرایط خیلی خوب) و حتی ماشین‌های کند مربوط به یک دهه قبل نیز هیچ مشکلی برای حفظ این شرایط ندارند. ماشین‌های امروزی بیش از هزار بار سریعتر شده‌اند. در صورتی که برای امنیت از ۹۰٪ زمان CPU استفاده کنیم، به نظر می‌رسد افزایش سرعت کامپیوتر یک دهم قبل خواهد بود. این افزایش ده درصدی به اندازه سرعت

[11] Audit log

کامپیوترهای پنج سال قبل است و این کامپیوترها از حد لازم برای انجام کار ما نیز سریعتر هم خواهند بود. ممکن است همیشه مجبور به گذراندن چرخه‌های زیاد بر روی امنیت نباشیم. اما این چیزی است که خواهان آن هستیم و نکته همین‌جا است.

موقعیت‌های کمی وجود دارد که در آنها منتظر انجام عملیات در کامپیوتر هستیم. این موقعیت‌ها شامل انتظار برای صفحات وب، چاپ داده‌ها، شروع برنامه‌های خاص، بوت شدن ماشین و غیره هستند. یک سیستم امنیتی سبب نخواهد شد که این فعالیت‌ها به کندی انجام گیرند. کامپیوترهای مدرن آنقدر سریع هستند که در آنها محاسبه چگونگی استفاده از چرخه‌ها به یک روش کارآمد کار بسیار سختی است. برای بخش‌هایی نظیر تصاویر صفحه نمایش و انیمیشن‌های سه بعدی و یا حتی تشخیص صدا از ترکیبات آلفا استفاده می‌کنیم، اما بخش‌هایی از همین نرم افزارها که محاسبات پیچیده را انجام می‌دهند هیچ نوع ارتباطی با بخش‌های امنیتی سیستم ندارند، بنابراین اگر سیستم امنیتی داشته باشیم، همین سیستم امنیتی سبب کند شدن بخش‌های محاسباتی از برنامه‌ها نخواهد شد.

از هم اکنون واضح است که اولویت‌های ما اول امنیت است، و عملکرد، در پایین فهرست اولویت‌های ما قرار دارد. به طور قطع، ما فقط می‌خواهیم که سیستم تا حد امکان کارآمد باشد اما نه به قیمت از بین رفتن امنیت. قابل درک است که این فلسفه طراحی در دنیای واقعی همیشه امکان‌پذیر نیست. اغلب واقعیت‌های بازار، مغلوب امنیت می‌شود. به ندرت می‌توان سیستم‌ها را از نقطه صفر توسعه داد و اغلب باید آنها را به صورت افزایشی یا پس از تولید، ایمن ساخت. سیستم‌ها باید قابل پشتیبانی با سیستم‌های ناامن موجود باشند. فلسفه طراحی این کتاب در درجه اول امنیت است و این همان چیزی است که می‌خواهیم در سیستم‌های تجاری بیشتر به آن توجه شود.

۱-۱۰-۲ امنیت در مقابل ویژگی‌ها

پیچیدگی، بزرگترین دشمن امنیت است و در اکثریت مواقع به صورت ویژگی‌ها یا گزینه‌هایی مطرح می‌شود.

در اینجا استدلال اساسی را ارائه می‌کنیم. برنامه کامپیوتری را با ۲۰ گزینه مختلف تصور کنید که هر یک از این گزینه‌ها ممکن است فعال یا غیرفعال باشند. چنین شرایطی بیش از یک میلیون پیکربندی خواهد داشت. برای اینکه برنامه بتواند کار کند، باید رایج-ترین ترکیب از گزینه‌ها را تست کنید. برای امن ساختن برنامه باید تمام این یک میلیون پیکربندی ممکن را که برنامه می‌تواند داشته باشد بررسی کرده و کنترل کنید که هر پیکربندی در مقابل اشکال احتمالی حمله، امن باشد. البته انجام چنین کاری محال است و

اغلب برنامه‌ها به صورت قابل ملاحظه‌ای بیش از ۲۰ گزینه دارند. بهترین روش برای اعتماد به امن بودن یک برنامه، ساده نگه داشتن آن است.

یک سیستم ساده لزوماً یک سیستم کوچک نیست. شما می‌توانید سیستم‌های بزرگی بسازید که در عین بزرگی، ساده هم باشند. پیچیدگی مقیاسی است از تعداد مواردی که در هر نقطه با هم تعامل می‌کنند. در صورتی که اثر یک گزینه به یک بخش کوچک از برنامه محدود باشد، نمی‌تواند با گزینه‌ای که اثر آن به بخش دیگری از برنامه محدود است برهم کنش داشته باشد. برای ایجاد یک سیستم بزرگ و ساده، باید بین بخش‌های مختلف سیستم، واسطِ بسیار شفاف و ساده‌ای ایجاد کنید. برنامه‌نویسان این کار را ماژولاریزاسیون می‌نامند که تماماً مهندسی نرم‌افزار پایه است. یک واسط ساده خوب، بقیه سیستم را از جزئیات یک ماژول جدا می‌کند. تمام گزینه‌ها یا ویژگی‌های ماژول نیز از بقیه سیستم جدا می‌شود.

یکی از مواردی که سعی داشتیم در این کتاب انجام دهیم تعریف واسط‌های ساده برای اعداد اول رمزنگاری است. هیچ ویژگی، هیچ گزینه، هیچ مورد خاصی به غیر از تعریف ساده‌ای که با آن پیش می‌رویم برای یادآوری وجود ندارد. برخی از این تعاریف جدید هستند و ما در حین نوشتن این کتاب آنها را بیشتر شرح و بسط می‌دهیم. این تعاریف به ما در شکل‌دهی تفکر درباره سیستم‌های امنیتی خوب کمک می‌کنند و امیدواریم به شما نیز کمک کنند.

۱-۱۰-۳ امنیت در مقابل سیستم‌های در حال تکامل

یکی از بزرگترین مشکلاتی که برای امنیت وجود دارد این است که سیستم کامل حتی بعد از اینکه مکانیزم‌های امنیتی پایه‌ای، کار گذاشته شدند همچنان تکامل می‌یابند. این بدان معناست که طراحان مکانیرم امنیتی، نه تنها به شکاکین حرفه‌ای و در نظر گرفتن طیف وسیعی از مهاجمان و اهداف تهاجمی نیاز دارند، بلکه به پیش‌بینی و آمادگی سیستم جهت استفاده‌های آتی از سیستم نیز نیاز دارند. این امر چالش‌های بزرگتری را ایجاد می‌کند و مسأله‌ای است که طراحان سیستم‌ها باید به خاطر داشته باشند.

۱-۱۱ منابع بیشتر برای مطالعه

تمام کسانی که به رمزنگاری علاقمند هستند باید کتاب «کد بازکن‌ها»[12] نوشته کان را بخوانند [۳]. این کتاب تاریخچه‌ای از رمزنگاری از دوران باستان تا قرن بیستم ارائه می‌کند. مثال‌های بسیاری از مشکلاتی که مهندسان سیستم‌های رمزنگاری با آن مواجه هستند در آن کتاب ارائه شده است. از جمله کتاب‌های بسیار جالب دیگر، «کتاب کد» است [۴]. کتابی که در دست دارید به روش‌های مختلف دنباله‌ای از اولین کتاب بروس، «رمزنگاری کاربردی» است [۵]. رمزنگاری کاربردی طیف وسیع‌تری از موضوعات مختلف را پوشش می‌دهد و شامل مشخصات تمام الگوریتم‌های مورد بحث آن است. با اینحال، در آن کتاب به جزئیات مهندسی که ما در این کتاب راجع به آن صحبت کردیم، پرداخته نمی‌شود.

برای دریافت واقعیت‌ها و نتایج دقیق نمی‌توانید از کتاب رمزنگاری نوشته *منزس، وان اورچوت* و *وانستون* صرف نظر کنید [۶]. این کتاب دائره‌المعارفی از رمزنگاری است و کتاب مرجع بسیار مفیدی محسوب می‌شود، اما درست همچون یک دائره‌المعارف، این کتاب هم منبع خوبی برای یاد گرفتن رمزنگاری نیست.

اگر به نظریه رمزنگاری علاقمند باشید، سری عالی از متون مرتبط در زمینه «اصول رمزنگاری» نوشته *گلدریخ* وجود دارد [۷،۸]. کتاب بسیار خوب دیگر «مقدمه‌ای بر رمزنگاری مدرن» نوشته *کتز و لیندج* است [۹]. همچنین یادداشت‌های دوره دانشگاهی بسیار عالی وجود دارد که به صورت آنلاین در دسترس است مانند یادداشت‌های دوره‌ای از *بلار و روگاوی* [۱۰].

کتاب قبلی بروس «اسرار و دروغ‌ها» [۲] توصیف خوبی از امنیت کامپیوتر به صورت عمومی و نحوه انطباق رمزنگاری با این تصویر بزرگتر است؛ و درباره مهندسی امنیت هیچ کتابی بهتر از کتاب «مهندسی امنیت» *روز اندرسون* وجود ندارد [۱۱]. هر دو کتاب برای درک زمینه رمزنگاری مهم و اساسی هستند. منابع آنلاین خوبی وجود دارند که هماهنگ با مسایل اخیر رمزنگاری و امنیت کامپیوتر پیش می‌روند. ما پیشنهاد می‌کنیم به خبرنامه کریپتوگرام[13] *بروس* به آدرس

http://www.schneier.com/crypto-gram.html

مراجعه کنید و بلاگ *بروس* را در این آدرس مطالعه کنید:

.http://www.schneier.com/blog/

---

[12] Code breakers

[13] Crypto-Gram

۱-۱۲ تمرین‌هایی برای مبحث شکاکیت حرفه‌ای

گفته می‌شود یکی از بهترین راهها برای یاد گرفتن یک زبان خارجی این است که در آن غرق شوید. اگر می‌خواهید فرانسه بیاموزید باید به فرانسه سفر کنید. این کتاب برای غرق کردن شما در زبان و ذهنیت رمزنگاری و امنیت کامپیوتر طراحی شده است. تمرین-های زیر به شما کمک می‌کند بیشتر با این مبحث درگیر شده و مثلاً وقتی عناوین اخبار را می‌خوانید یا با دوستان درباره رویدادهای جاری فکر می‌کنید یا شرح یک محصول جدید را بر روی وبسایت‌های مربوط به فناوری می‌خوانید به صورت منظمی وادار به فکر کردن درباره امنیت شوید. فکر کردن درباره امنیت دیگر کار سخت و طاقت‌فرسایی نیست، به خصوص زمانی که به طور خاص وظیفه اختصاص یافته به شما فکر کردن به آن باشد. حتی ممکن است وقتی سگ خود را برای گردش بیرون برده‌اید، یا در حال دوش گرفتن یا حین تماشای فیلم شروع به فکر کردن درباره امنیت باشید. به طور خلاصه، تنظیمات ذهنی متناسب با شکاکیت حرفه‌ای را در خود توسعه داده و شروع به فکر کردن به امنیت همچون متخصصین امنیت خواهید کرد.

همچنین برای دست اندرکاران امنیت کامپیوتر (و در واقع برای تمام دانشمندان علوم کامپیوتر) آگاهی از مسایل گسترده مفهومی در حول و حوش فناوری از اهمیت بسیاری برخوردار است. فناوری‌ها به صورت ایزوله وجود ندارند بلکه جنبه کوچکی از یک اکوسیستم بزرگتر هستند که مردم، اقتصاد، اخلاق، تفاوت‌های فرهنگی، سیاست، قانون و مانند آنها را شامل می‌شوند. تمرین‌های امنیتی، فرصتی را برای بحث و کشف این مسایل بصورت وسیع‌تر را فراهم می‌کند، چرا که مسایلی مرتبط با امنیت هستند.

پیشنهاد می‌کنیم به طور منظم به تمرین‌های زیر مراجعه کرده، سعی کنید این تمرین-ها را تا حدی که می‌توانید مکرراً انجام دهید. به طور مثال، ممکن است یک ماه به طور مرتب این تمرین‌ها را هر هفته انجام دهید یا پس از اتمام هر فصل از این کتاب این کار را انجام دهید، خلاصه اینکه به هر ترتیبی که بیشترین تکرار را داشته باشد این کار را انجام دهید. در ابتدا ممکن است تمرین‌ها پرزحمت و خسته کننده به نظر برسند، اما انجام این تمرینها به عهده شما گذاشته شده و به زودی هر وقت که با عناوین اخبار مرتبط با امنیت روبرو می‌شوید یا با محصول جدیدی روبرو می‌شوید، می‌بینید که به طور خودکار در حال انجام این تمرین‌ها هستید. این همان ذهنیت همراه با شکاکیت حرفه‌ای است. علاوه بر این، اگر در حین خواندن این کتاب، به انجام این تمرین‌ها ادامه دهید، متوجه توان خود در

ارزیابی خصوصیات تکنولوژیکی از سیستم‌ها که در طی زمان تکامل می‌یابد نیز خواهید شد.

انجام این تمرین‌ها را به همراه یک دوست به شما پیشنهاد می‌کنیم. بحث کردن درباره مسایل مربوط به امنیت با دیگران می‌تواند بسیار آموزنده باشد و به زودی متوجه می‌شوید که مسایل مربوط به امنیت به طرز باورنکردنی زیرکانه بوده و مسلط شدن به نقاط ضعف بحرانی بسیار آسان است. واضح است که اگر یادگیری شما از طریق کلاس و انجام تمرین-های رسمی نباشد، ممکن است ترجیح دهید تمرین‌ها را در ذهن خود انجام دهید تا اینکه واقعاً گزارش‌های مکتوبی از آن را ارائه دهید. مع ذلک، به شما پیشنهاد می‌کنیم دست کم یک بار گزارش مکتوبی از مسایل حل شده تهیه کنید، زیرا که از طریق حل مسایل مربوطه، خود را وادار کرده تا به طور کامل به امنیت فکر کنید.

۱-۱۲-۱ تمرین‌های مربوط به رویدادهای جاری

در این تمرین‌ها باید برخی رویدادهایی را که در حال حاضر در اخبار به گوش می‌رسد به صورت انتقادی تجزیه و تحلیل کنید. رویدادی را که انتخاب می‌کنید باید به نوعی به امنیت کامپیوتری ربط داشته باشد. شاید مکانیزم‌های امنیت کامپیوتری توسعه یافته، با این رویداد جدید ناسازگار باشد. ممکن است این رویداد مشوقی برای طراحی مکانیزم‌ها یا سیاست‌های امنیتی جدید باشد.

مطالبی که درباره رویدادهای جاری فوق‌الذکر می‌نویسید باید کوتاه، مختصر، با پشتوانه فکری بسیار خوب بوده و به خوبی نوشته شود و برای نوشتن آن مخاطب عام را مد نظر قرار دهید. هدف شما باید نوشتن مقاله‌ای باشد که مطالب مورد نظر و درک زمینه امنیت کامپیوتر را به خواننده یاد داده و نحوه ورود به زمینه‌های وسیعتر را به او بیاموزد.

باید رویداد جاری را به اختصار بیان کنید، درباره چگونگی وقوع این رویداد بحث کنید، به این فکر کنید که قبل از وقوع این رویداد چه کار متفاوتی را می‌شد (شاید برای جلوگیری، ردیابی یا اصلاح پیامدهای رویداد) انجام داد، مسایل وسیع‌تر مربوط به رویداد جاری (مانند مسایل اخلاقی یا مسایل اجتماعی) را تشریح کرده و واکنش‌های احتمالی نسبت به رویداد جاری را پیشنهاد کنید (مانند اینکه عموم مردم، سیاست‌گذاران، رسانه‌ها یا دیگران چه واکنشی را ممکن است نشان بدهند).

۱-۱۲-۲ تمرین‌های مربوط به مرور امنیت

این تمرین‌ها، سبب توسعه امنیتی ذهن شما در زمینه محصولات یا سیستم‌های حقیقی می‌شود. هدف شما از مرور موارد امنیتی، ارزیابی مسایل بالقوه امنیت و حریم خصوصی در فناوری‌های جدید، ارزشیابی دقت این مسایل و بحث درباره نحوه پرداختن به مسایل امنیتی و حریم خصوصی است. این بررسی‌ها باید به طور عمیق در فناوری‌هایی که به بحث درباره آنها می‌پردازید منعکس شده و بطور قابل ملاحظه‌ای در تمرینات طولانی مدت شما قابل مشاهده باشد.

هر مورد از بررسی‌های امنیتی باید شامل موارد زیر باشد:

- خلاصه‌ای از فناوری که در حال ارزیابی آن هستید. ممکن است ارزیابی یک محصول خاص را ترجیح دهید یا کلاسی از محصولاتی با اهداف مشترک (مانند مجموعه تمام ابزارهای پزشکی قابل کاشت). این خلاصه باید در سطح بالا بوده و حدود یک یا دو پاراگراف باشد. جنبه‌های فناوری را که به مشاهدات شما در عناوین زیر است بیان کنید.

برای این تمرین‌ها، بیان برخی مفروضات درباره نحوه کار محصولات قابل قبول است. با اینحال، در صورتی که فرضیاتی را درباره یک محصول در نظر بگیرید، باید روشن کنید که چنین مفروضاتی را داشته‌اید و به صراحت بگویید که این مفروضات چه هستند.

توانایی شرح یک محصول به صورت شفاف و مختصر (حتی با مفروضاتی که به صراحت بیان شده‌اند) بسیار اهمیت دارد. در صورتی که بقدر کافی فناوری لازم برای ارائه یک خلاصه شفاف و روشن را درک نکرده باشید، احتمالاً فناوری لازم برای ارزیابی امنیت و حریم خصوصی آن را نیز درک نکرده‌اید.

- حداقل دو چیز با ارزش و برای هر کدام یک هدف امنیتی مرتبط بیان کنید. توضیح دهید چرا اهداف امنیتی مهم هستند. شما باید حدود یک یا دو جمله برای آن چیز با ارزش/ هدف بنویسید.
- دست کم دو تهدید احتمالی را بیان کنید که در آنها یک تهدید به عنوان اقدامی از سوی دشمن با هدف نفوذ به آن دارایی تعریف می‌شود. برای هر تهدید مثالی از طرف دشمن بزنید و حدود یک یا دو جمله برای هر طرف متخاصم در تهدید بنویسید.

- حداقل دو مورد از ضعف‌های بالقوه را بیان کنید. با استفاده از یک یا دو جمله برای هر ضعف جواب خود را ثابت کنید. برای رسیدن به هدف‌های مورد نظر این تمرین‌ها، نیازی به بررسی صحت و سقم این که آیا ضعف‌های بالقوه ضعف‌های واقعی هستند یا خیر ندارید.
- دفاع‌های بالقوه در برابر این معایب را بیان کنید. دفاع‌های بالقوه‌ای که سیستم می‌تواند استفاده کند یا ممکن است تا به حال برای پرداختن به ضعف-های بالقوه‌ای که در عناوین قبلی شناسایی کردید استفاده شود را بیان کنید.
- ریسک‌های همراه با دارایی‌ها، تهدیدها و ضعف‌های بالقوه‌ای را که تشریح کردید ارزیابی کنید. به صورت حدودی فکر می‌کنید این ترکیب دارایی‌ها، تهدیدها و ضعف‌های بالقوه چقدر جدی هستند؟
- نتیجه‌گیری. نظرات همراه با پشتوانه فکری را در جواب‌های خود ارائه کنید. همچنین درباره مسایل «تصویر بزرگتر» بحث کنید (اخلاق، همانندی که فناوری به آن خواهد رسید و مانند آن).

مثال‌هایی از مرورهای مربوط به مسایل امنیت به صورت آنلاین به این آدرس در دسترس است: http://www.schneier.comlce.html.

تمرین‌ها

تمرین ۱-۱: یک درخت حمله برای سرقت خودرو بنویسید. می‌توانید درخت حمله خود را بصورت یک شکل (همانند شکل ۱-۱) یا بصورت لیتس از شماره‌ها (مانند ۱و ۱-۱ و ۱-۲ و ۱-۲-۱و ...) نشان دهید.

تمرین ۱-۲: یک درخت حمله برای وارد شدن به باشگاه ورزشی بدون پرداخت هزینه ایجاد کنید.

تمرین ۱-۳: یک درخت حمله برای بدست آوردن غذا بدون پرداخت هزینه طراحی کنید.

تمرین ۱-۴: یک درخت حمله برای یادگیری نام و کلمه عبور حساب بانکی یک شخص طراحی کنید.

تمرین ۱-۵: یک درخت حمله برای خواندن ایمیل‌های یک فرد طراحی کنید.

تمرین ۱-۶: درخت حمله‌ای طراحی کنید که یک شخص نتواند ایمیل خود را بخواند.

تمرین ۱-۷: یک درخت حمله طراحی کنید که ایمیلی برای یک فرد ارسال کند. بطوریکه شخص گیرنده تصور کند آن ایمیل واقعا از سمت فرستنده ارسال شده است در حالیکه اینگونه نیست.

تمرین ۱-۸: محصول یا سیستمی را پیدا کنید که در ۳ ماهه گذشته تولید یا منتشر شده است. بررسی‌های امنیتی را برروی این محصول همانند آنچه در بخش ۱-۱۲ گفته شد انجام دهید. یک قسمت از آن را انتخاب کرده و درخت حمله‌ای برای آن طراحی کنید.

تمرین ۱-۹: با ارائه یک مثال عینی از طریق رسانه‌ها یا تجربیات شخصی خود، که در آن یک سیستم از طریق ضعیف‌ترین لینک آن به خطر افتاده است و به آن حمله شده است. سیستم را توضیح دهید. آنچه را که از ضعیف‌ترین حلقه مشاهده می‌کنید را توضیح دهید و بگوئید که چرا و چگونه این سیستم به خطر افتاده است.

تمرین ۱-۱۰: یک مثال واقعی را توضیح دهید که در آن با افزایش امنیت در برابر نوع خاصی از حمله، احتمال حمله دیگر را افزایش می‌دهید.

مراجع

[1] Bruce Schneier. Attack Trees. Dr. Dobb's Journal, 1999. Also available from http://www.schneier.com/paper-attacktrees-ddj-ft.htrnl.[Page5]

[2] Bruce Schneier. Secrets and Lies: Digital Security in a Networked World. John Wiley & Sons, Inc., 2000. [Pages 16, 18]

[3] David Kahn. The Codebreakers, The Story of Secret Writing. Macmillan Publishing Co., New York, 1967. [Page 18]

[4] Simon Singh. The Code Book: The Science of Secrecy from Ancient Egypt to Quantum Cryptography. Anchor, 2000. [Page 18]

[5] Bruce Schneier. Applied Cryptography, Second Edition, Protocols, Algorithms, and Source Code in C. John Wiley & Sons, Inc., 1996. [Pages 18, 323]

[6] Alfred J. Menezes, Paul C. van Oorschot, and Scott A Vanstone. Hand-book of Applied Cryptography. CRC Press, 1996. Also available from http://www.cacr.math.uwaterloo.ca/hac/.[Pages18.243]

[7] Oded Goldreich. Foundations of Cryptography: Volume I, Basic Tools. Cambridge University Press, 2001. Also available from http://www.wisdorn .weizrnann. ac. i1/oded/foc-book.html. [Page 18]

[8] Oded Goldreich. Foundations of Cryptography: Volume 2, Basic Applications. Cambridge University Press, 2001. Also available from http://www.wisdom.weizrnann.ac. i1/oded/foc-book.htrnl. [Page 18]

[9] Jonathan Katz and Yehuda Lindell. Introduction to Modern Cryptography: Principles and Protocols. Chapman & Hall/CRC, 2007. [Page 18]

[10] Mihir Bellare and Phillip Rogaway. Introduction to Modern Cryptography, 2005. Available from http://cseweb.ucsd.edu/users/rnihir/ cse207 /classnotes .htrnl. [Page 18]

[11] Ross J. Anderson. Security Engineering: A Guide to Building Dependable Distributed Systems. John Wiley & Sons, Inc., 2008. [Page 18]

فصل
۲۳
درگیر کردن متخصصین

موضوع عجیبی پیرامون رمزنگاری وجود دارد: همه فکر می‌کنند که برای طراحی و ساخت سیستم خود، اطلاعات کافی دارند. هیچ‌گاه از یک دانشجوی سال دومی فیزیک درخواست نمی‌شود تا تاسیسات انرژی هسته‌ای طراحی کند. مسلما به یک پرستار کارآموز که ادعا می‌کند به روش جدیدی در جراحی قلب دست پیدا کرده است، اجازه نمی‌دهیم تا ما را عمل کند. با این وجود، برخی افراد با خواندن یک یا دو کتاب فکر می‌کنند قادرند تا سیستم رمزنگاری خود را طراحی کنند. بدتر اینکه گاهی می‌توانند مدیریت، سرمایه‌داران فعالیت‌های اقتصادی و حتی برخی ارباب رجوع را قانع کنند که طرح آن‌ها، شسته رفته-ترین طرح بعد از نان تست است.

کتاب اول *بروس* تحت عنوان "رمزنگاری کاربردی" [۱، ۲] در میان رمزنگاران هم از شهرت برخوردار است و هم بدنامی. شهرت این کتاب از این بابت است که ده‌ها هزار نفر را متوجه رمزنگاری کرد و بدنامی آن از بابت سیستم‌هایی است که این افراد به تنهایی طراحی و اجرا کردند. ۸۰۲.۱۱، استاندارد شبکه بیسیم، نمونه‌ی اخیر این سیستم‌هاست. طراحی اولیه شامل یک مجرای امنیتی بنام WEP می‌باشد که ارتباطات بیسیم را رمزنگاری و تایید می‌کند. کمیته بدون هیچ (شخص) رمزنگاری، این استاندارد را طراحی کرد. نتایج ازلحاظ امنیتی افتضاح بود. تصمیم بر استفاده از الگوریتم رمزنگاری RC4 ایده خوبی نبود اما به خودی خود نیز یک نقص محسوب نمی‌شد. با این حال، RC4 یک رمز جاری و نیازمند یک نانس[1] منحصر به فرد است. WEP، ارقام دودوئی کافی به نانس اختصاص نداد با این پیامد که همان مقادیر نانس بایستی مجددا استفاده شوند که در عوض باعث شد بسته‌های بسیاری با جریان کلیدی یکسانی رمزنگاری شوند. این کار دارایی‌های رمزنگاری رمز جاری RC4 را مغلوب کرد و اجازه داد تا یک ضارب کوچک این رمزبندی را بشکنند. نقص زیرکانه‌تر این نبود که کلیدهای سری و نانس را قبل از استفاده از آن به عنوان کلید RC4 هش کند که نهایتا منجر به حملات بازیافت کلیدی شد [۳]. جمع مقابله‌ای CRC[2] برای تایید کردن استفاده شد اما بدلیل خطی بودن محاسبات CRC، اصلاح هر بسته بدون شانس ردیابی (با استفاده از جبر خطی)، امری بدیهی بود. همه کاربران

---

[1] Nonce

[2] Cyclic Redundancy Check

شبکه از یک کلید همگانی استفاده می‌کنند در نتیجه آپدیت (به روز کردن) این کلید با دشواری‌هایی همراه است. پسورد (کلمه عبور) شبکه مستقیما به عنوان کلید رمزنگاری برای تمام ارتباطات استفاده شد، بدون اینکه از هیچ پروتکل مذاکره‌ای کلیدی استفاده شود. و سرانجام، رمزنگاری از طریق پیش‌فرض خاموش شد به این معنا که اکثر اجراها برای روشن کردن رمزنگاری در اولین مکان، هرگز خود را به دردسر نمی‌انداختند.

طراحی یک جایگزین برای WEP کار آسانی نبود، زیرا باید به سخت‌افزار موجود اضافه می‌شد، هرچند راه دیگری نیز وجود نداشت؛ استاندارد اصلی از امنیت بالایی برخوردار بود و جایگزین آن WPA شد. داستان WEP استثنا نیست. فشاری بیشتر از اکثر طرح‌های رمزنگاری تحمل کرد زیرا ۸۰۲.۱۱ محصول موفقی است هرچند ما شاهد موقعیت‌های مشابه زیادی در سیستم‌های دیگر بوده‌ایم. *بروس* در مقام یک همکار سابقا بیان کرد که:

"جهان مملو از سیستم‌های امنیتی ضعیفی است که بدست افرادی طراحی شده که کتاب رمزنگاری کاربردی را خوانده‌اند". مهندسی رمزنگاری نیز می‌توانست همین تاثیر را داشته باشد. همین امر، این کتاب را بسیار خطرناک می‌کند. برخی افراد این کتاب را می-خوانند و سپس اقدام به طراحی یک پروتکل یا الگوریتم رمزنگاری می‌کنند. پس از اتمام کار، نتیجه در نظرشان خوب جلوه می‌کند و حتی کار هم می‌کند اما آیا از امنیت برخوردر است؟ ممکن است ۷۰ درصد و اگر خیلی خوش شانس باشند، ۹۰ درصد درست کار کرده باشند. اما اگر درست هم کار کرده باشند، هیچ جایزه‌ای در رمزنگاری به ایشان تعلق نمی-گیرد. قدرت یک سیستم امنیتی به اندازه ضعیف‌ترین ارتباط‌هایش است؛ همه چیز باید درست کار کند تا یک سیستم از امنیت برخوردار باشد و این چیزی است که مسلما با خواندن کتاب بدست نمی‌آید.

اگر این احتمال وجود دارد که خواندن کتاب منجر به ساخت سیستم‌های ضعیف می-شود، حال این سوال پیش می‌آید که چرا این کتاب را تالیف کردیم؟ افرادی که می‌خواهند طریقه طراحی سیستم‌های رمزنگاری را یاد بگیرند، بایستی از جایی آن را بیاموزند، درنتیجه ما این کتاب را تالیف کردیم و کتاب مناسب دیگری نیز مد نظر نداشتیم. با وجود اینکه این کتاب، کتاب راهنما نیست اما آنرا مقدمه بر این زمینه درنظر بگیرید. به علاوه، این کتاب برای دیگر مهندسین شاغل در یک پروژه نیز می‌باشد. تمام بخش‌های یک سیستم امنیتی از اهمیت بالایی برخوردارند و هرکس که در پروژه‌ای مشغول است، باید از

درک پایه‌ای مسائل و تکنیک‌های امنیتی آن پروژه برخوردار باشد. این مسائل، برنامه نویس‌ها، آزمونگرها، نویسندگان فنی، مدیریت و حتی فروشندگان را شامل می‌شود. هر شخص باید در حد لازم مسائل امنیتی را درک کند تا کار خود را به طور صحیح و شایسته انجام دهد. امیدواریم این کتاب بتواند زمینه کافی از جنبه کاربردی رمزنگاری ارائه دهد. همچنین امیدواریم توانسته باشیم حس پارانویای حرفه‌ای را در شما القا کرده باشیم. اگر این را متوجه شده باشید، مطالب زیادی را یادگرفته‌اید. می‌توانید پارانویای حرفه‌ای را در تمام جوانب کارتان بکار برید. زمانی که پروتکل خود را طراحی می‌کنید و یا به طرح شخص دیگری نگاه می‌کنید، کاملا مردد هستید و این می‌تواند به افزایش امنیت کمک کند.

به شما توصیه می‌کنیم در صورت امکان از متخصصین رمزنگاری استفاده کنید. در صورتی که پروژه شما شامل رمزنگاری است، می‌توانید از بینش یک طراح رمزنگار با تجربه کمال استفاده را بکنید. در ابتدا یکی را در پروژه خود وارد کنید. هرچه زودتر با یک متخصص رمزنگاری مشورت کنید، پروژه در دراز مدت برایتان آسان‌تر و کم هزینه‌تر خواهد بود. بارها از ما برای پروژه‌های در دست اقدام دعوت به عمل آمده تا در بخش‌هایی که مدت‌ها پیش طراحی و اجرا شده‌اند، روزنه‌ای ایجاد کنیم (راهی باز کنیم). نتایج پایانی همیشه گران هستند چه از لحاظ تلاش، زمانبندی پروژه و هزینه و چه از لحاظ امنیت کاربر محصول نهایی.

رمزنگاری بطور صحیح بسیار سخت‌تر از حد تصور است. حتی سیستم‌هایی که توسط متخصصین طراحی می‌شوند نیز مرتبا با شکست مواجه می‌شوند. مهم نیست که چقدر باهوش هستند و یا چقدر تجربه در زمینه‌های دیگر دارید. طراحی و اجرای سیستم‌های رمزنگاری نیازمند تجربه و دانش تخصصی است و تنها راه کسب تجربه نیز انجام مکرر آن کار است. البته اشتباهاتی نیز در آن صورت می‌گیرد. پس اگر متخصص نیز دچار اشتباه می‌شود، چرا از او استفاده کنیم؟ به همان دلیل است که شما از یک جراح لایق می‌خواهید تا شما را عمل کند. این بدان معنا نیست که آن‌ها دچار اشتباه نمی‌شوند بلکه دارای اشتباهات کمتر و کم اهمیت‌تر می‌باشند. آن‌ها همچنین بسیار محافظه کارانه عمل می-کنند طوری که اشتباهات کوچک منجر به مصیبت یا فاجعه نشوند؛ آن‌ها می‌دانند چطور عدم موفقیتشان خیلی بد جلوه نکند. اجرای سیستم‌های رمزنگاری نیز تقریبا به اندازه طراحی آنها، نیازمند تخصص است. طراحان رمزنگاری وجود دارند که شما می‌توانید آن‌ها را

استخدام کنید. اما بدست آوردن اجراکنندگان رمزنگار بسیار مشکلتر است، زیرا شما به تعداد زیادی از آن‌ها نیاز دارید. یک طراح می‌تواند برای ۱۰ تا ۲۰ فرد اجرا کننده، اشتغال-زایی کند. اکثر مردم به اجرای رمزنگاری به چشم تخصص نگاه نمی‌کنند. برنامه‌ریزان از برنامه‌ریزی تحت پایگاه داده‌ها تا کار GUI و اجرای رمزنگاری حرکت می‌کنند. اینکه برنامه-ریزی تحت وب و کار GUI هر دو تخصصی هستند، واقعیت دارد اما یک برنامه‌ریز با تجربه می‌تواند با کمی مطالعه به نتایج قابل قبولی دست یابد. این موضوع برای اجرای رمزنگاری صدق نمی‌کند زیرا در اجرای رمزنگاری همه چیز باید درست باشد و یک نفوذ کننده نیز وجود دارد که سعی در خراب کاری دارد.

بهترین شیوه اجرای سیستم‌های رمزنگاری، بکارگیری برنامه‌ریزان لایق و آموزش به آن‌ها در زمینه مربوطه می‌باشد. این کتاب می‌تواند بخشی از آموزش آنان باشد اما آموزش آنان بیشتر نیازمند تجربه و نگرش پارانویای حرفه‌ای صحیح می‌باشد. دقیقا مانند مهارت-های تخصصی فناوری اطلاعات، این کار نیز سال‌ها طول می‌کشد تا فردی در آن خبره شود. در مدتی که طول می‌کشد تا این تجربیات بدست آیند، شما باید بتوانید این افراد را زمانی که به تجربه لازم می‌رسند، نگه دارید. این مسئله دیگری است و ما حل آن را با مسرت به افراد دیگر واگذار می‌کنیم. شاید فرهنگ پروژه از این کتاب یا هر کتاب دیگری، مهمتر باشد. "امنیت حرف اول را می‌زند" نباید تنها یک شعار باقی بماند؛ بلکه بایستی به اساس پروژه و تیم پروژه متصل باشد. همه باید در هر زمانی با امنیت زندگی کنند، نفس بکشند، صحبت کنند و تفکر کنند. رسیدن به این امر بسیار سخت اما ممکن می‌باشد. *دیگی کش* در دهه ۱۹۹۰ یک تیم این چنینی داشت. صنعت هواپیمایی دارای فرهنگ نافذ امنیتی مشابهی بود. این چیزی است که در کوتاه مدت حاصل نمی‌شود بلکه می‌توانید در راه آن سعی و تلاش خود را بکنید. این کتاب صرفا مقدمات اولیه پیرامون مهمترین مسائل امنیتی است که برای افراد فنی‌تر و متخصص‌تر تیم طراحی شده است.

طبق نوشته *بروس* در کتاب اسرار و دروغ‌ها، "امنیت یک فرایند است نه فراورده". شما علاوه بر فرهنگ امنیت به یک فرایند امنیت نیز نیاز دارید. صنعت هواپیمایی از فرایند وسیع امنیتی برخوردار است. بیشتر صنعت‌های فن‌آوری اطلاعات بدون درنظر گرفتن فرایند نرم‌افزارهای باکیفیت، فرایندی برای تولید نرم افزار ندارند چه برسد به فرایند نرم-افزارهای امنیتی. نوشتن یک نرم افزار امنیتی خوب بسیار فراتر از وضعیت کنونی هنر در صنعت ما می‌باشد. با این وجود، این بدان معنا نیست که ما باید تسلیم شویم؛ اخیرا نیز

پیشرفت‌هایی صورت گرفته است. با افزایش اهمیت فن‌آوری اطلاعات برای زیرساخت‌ها، آزادی و امنیت ما، باید در جهت پیشرفت و بهبود امنیت سیستم‌های خود حرکت کنیم و تمام تلاش خود را معطوف آن کنیم.

مراجع

[1] Bruce Schneier. Applied Cryptography, Protocols, Algorithms, and Source Code in C. John Wiley & Sons, Inc., 1994. [Page 323]

[2] Bruce Schneier. Applied Cryptography, Second Edition, Protocols, Algorithms, and Source Code in C. John Wiley & Sons, Inc., 1996. [Pages 18, 323]

[3] Scott Fluhrer, Itsik Mantin, and Adi Shamir. Weaknesses in the Key Schedule Algorithm of RC4. In Serge Vaudenay and Amr M. Youssef, editors, Selected Areas in Cryptography, 8th Annual International Workshop, SAC 2001, volume 2259 of Lecture Notes in Computer Science. SpringerVerlag, 2001. [Page 324]