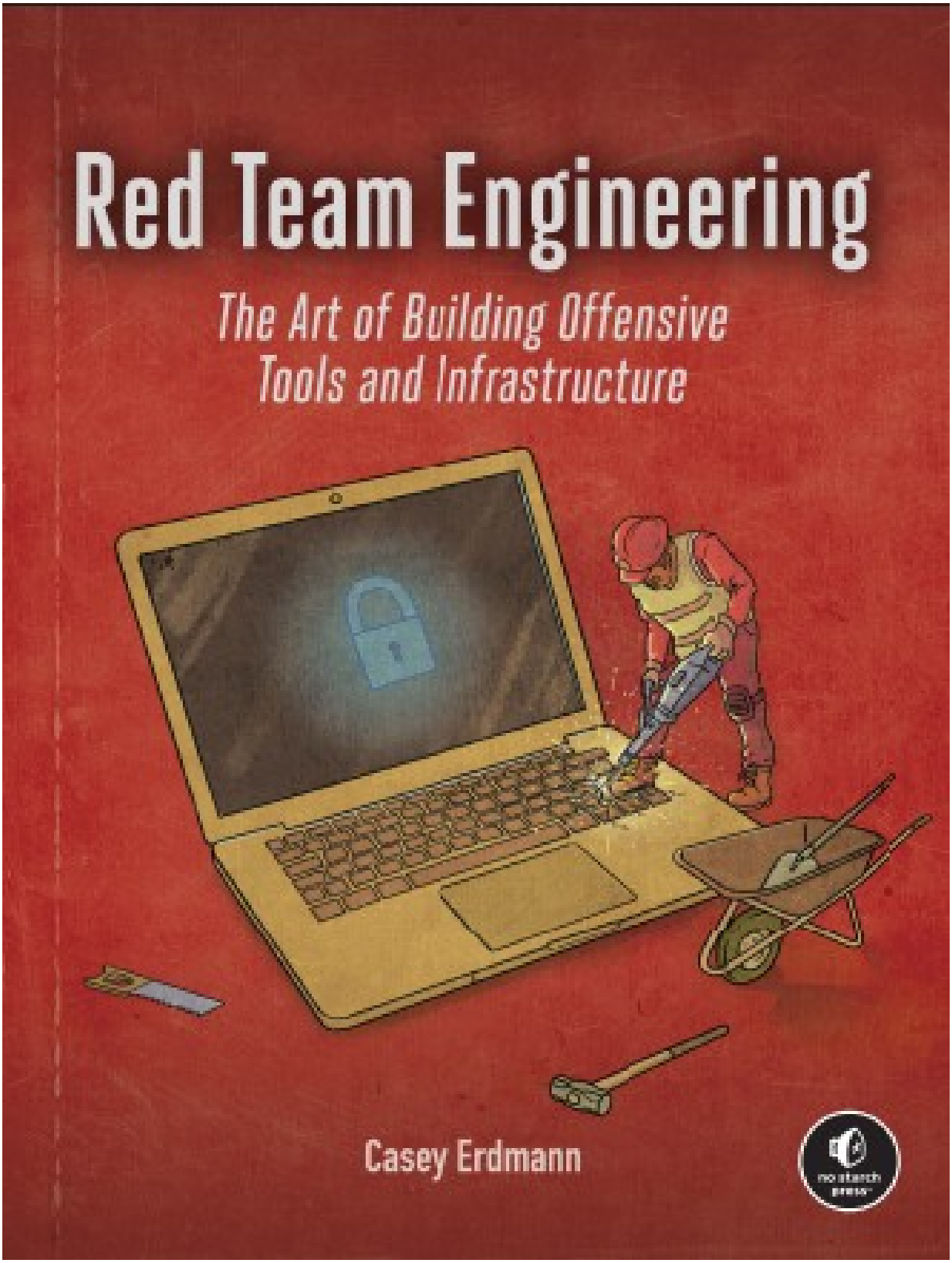


مقدمه ای بر مهندسی ردتیم

مهندسی ردتیم (RED TEAM ENGINEERING)

هنر ساخت ابزارها و زیرساخت‌های امنیت تهاجمی

@Sec_Book_Mind

مترجم : typhon

کانال تلگرام: @Sec_Book_Mind

مقدمه



هکر یه خلافکار نیست؛ هکر بیشتر شبیه یه هنرمنده که می‌تونه از یه ابزار یا محصول، جوری استفاده کنه که لزوماً سازنده‌ش براش در نظر نگرفته. ممکنه برای خلاقیت و تجربه‌کردن هک کنی، دنبال حل یه مشکل باشی، بخوای به پیشرفت و منفعت جامعه کمک کنی یا حتی، بله، ازش برای آسیب‌زدن استفاده کنی.

درسته که تو گذشته خیلی از هکرها مهارت‌هاشون رو تو محیط‌های غیرقانونی و خرابکارانه یاد می‌گرفتن، اما خوشبختانه دیگه اوضاع مثل قبل نیست. امروزه دسترسی به دانش و مهارت‌های هک خیلی راحت‌تر شده و اگه این دانش به شکل اخلاقی و درست به کار گرفته بشه، می‌تونه هم به پیشرفت فناوری کمک کنه و هم برای کل جامعه مفید باشه. دلیل اینکه خیلی از ما هک رو به‌عنوان شغل و حرفه‌مون انتخاب کردیم هم همینه؛ و البته یکی از دلایل نوشتن این کتاب هم همین بوده.

این کتاب بهت کمک می‌کنه مهارت‌هایی رو به دست بیاری که برای تبدیل‌شدن به یه هکر موفق بهشون نیاز داری. یاد می‌گیری چطور از دانش فعلی‌ات تو زمینه برنامه‌نویسی، مدیریت سیستم‌ها و مهندسی زیرساخت استفاده کنی و اون رو وارد دنیای امنیت تهاجمی (Offensive Security) کنی.

فرقی نمی‌کنه متخصص امنیت باشی، محقق امنیتی یا صرفاً یه متخصص IT کنجکاو؛ این کتاب قراره ابهامات مربوط به نحوه اجرای عملیات حرفه‌ای هک رو برات برطرف کنه و ابزارهایی رو در اختیارت بذاره که برای انجام چنین عملیاتی بهشون نیاز داری.

@Sec_Book_Mind

مفاهیم اصلی امنیت تهاجمی (Offensive Security)

قبل از اینکه وارد مباحث اصلی بشیم، بهتره یه نگاهی به اصطلاحات مهمی بندازیم که تو سراسر این کتاب بارها باهاشون سروکار خواهی داشت:

Command and Control (C2) — فرماندهی و کنترل

مجموعه‌ای از ابزارها و تکنیک‌ها که متخصص‌های امنیت تهاجمی ازشون برای کنترل هم‌زمان چندین کامپیوتر استفاده می‌کنن. C2 همچنین به اون‌ها اجازه می‌ده دسترسی‌شون به یه سیستم نفوذشده رو حفظ کنن تا بتونن حملات پیچیده‌تری رو اجرا کنن.

Fingerprinting — شناسایی اثر انگشت

فرایند جمع‌آوری اطلاعات و شناسایی ویژگی‌های یه اپلیکیشن یا سرویس. متخصص‌های امنیت تهاجمی از Fingerprinting استفاده می‌کنن تا بفهمن یه سرویس از نظر فنی چه ویژگی‌هایی داره و آیا می‌تونه برای اهدافی مثل نفوذ به یه سیستم به دردشون بخوره یا نه.

Fuzzing — فازینگ

فرایند خودکاری که توش ورودی‌های تصادفی به یه اپلیکیشن یا سرویس ارسال می‌شن تا آسیب‌پذیری‌ها، باگ‌ها و اطلاعات مرتبط دیگه شناسایی بشن.

Offensive Security — امنیت تهاجمی

حوزه‌ای از امنیت سایبری که توش متخصص‌ها استخدام می‌شن تا به یه هدف مشخص حمله کنن و از این طریق، وضعیت کلی امنیت اون سازمان یا مشتری رو ارزیابی و به بهبودش کمک کنن. این

@Sec_Book_Mind

حوزه شامل موقعیت‌های شغلی مختلفی مثل پنتسترها، اعضای ردتیم، محقق‌های امنیتی و تحلیلگرهای آسیب‌پذیری می‌شه.

Password Spraying — حمله پسورد اسپری

تو این روش، مهاجم برای ورود به یه اپلیکیشن، چندین نام کاربری مختلف رو با استفاده از یه رمز عبور رایج امتحان می‌کنه.

بعد از اولین دور تلاش‌ها، متخصص امنیت معمولاً یه مدت صبر می‌کنه و بعد سراغ امتحان‌کردن یه رمز عبور رایج دیگه می‌ره تا شناسایی نشه یا حساب‌های کاربری به خاطر تلاش‌های ناموفق قفل نشن.

این چرخه همین‌طور ادامه پیدا می‌کنه تا زمانی که یا تمام نام‌های کاربری و رمزهای عبور رایج موجود تو لیستش امتحان بشن یا بالاخره موفق بشه به یکی از حساب‌ها دسترسی پیدا کنه.

Penetration Testing — تست نفوذ

فرایند شناسایی و اکسپلویت‌کردن حداکثر تعداد ممکن از آسیب‌پذیری‌های فنی برای یه مشتری، با این هدف که نقاط ضعف سیستم مشخص بشن و راهکارهایی برای برطرف‌کردنشون ارائه بشه.

برخلاف ردتیمینگ (که در ادامه تعریفش می‌کنیم)، تست نفوذ معمولاً روی رسیدن به یه هدف مشخص متمرکز نیست و اغلب شامل فعالیت‌های مهندسی اجتماعی هم نمی‌شه.

@Sec_Book_Mind

Red Teaming — ردتیمینگ

رویکردی برای پیدا کردن آسیب‌پذیری‌ها و عبور از لایه‌های دفاعی یه سازمان، با هدف ارزیابی وضعیت امنیتی اون سازمان.

توی این روش، اعضای ردتیم نقش مهاجم رو بازی می‌کنن. معمولاً یه هدف مشخص بهشون داده می‌شه؛ مثلاً دسترسی به ایمیل مدیرعامل. همچنین مجوز کامل دارن که با محدودیت‌های نسبتاً کمی برای رسیدن به اون هدف تلاش کنن تا تهدیدهایی رو که سازمان نگرانشونه، شبیه‌سازی کنن.

در نقطه مقابل، بلو تیم‌ها (Blue Teams) قرار دارن که مسئولیتشون دفاع در برابر تهدیدها و واکنش به حوادث امنیتیه.

Shell — شل

یه اصطلاح کلی برای هر نوع نشست تعاملی خط فرمان (Command Line) داخل سیستم هدف.

یه شل معمولی می‌تونه همون ترمینال لینوکس یا Command Prompt ویندوز باشه.

Web Shell هم با استفاده از یه زبان برنامه‌نویسی وب مثل PHP یا ASP.NET ساخته می‌شه و از طریق یه اپلیکیشن وب و داخل مرورگر، امکان دسترسی به نشست خط فرمان سیستم اصلی رو فراهم می‌کنه.

Console هم به اتصالی گفته می‌شه که از طریق پورت سریال برقرار می‌شه.

Social Engineering — مهندسی اجتماعی

@Sec_Book_Mind

استفاده از تکنیک‌های روان‌شناختی و دست‌کاری رفتار انسان برای وادارکردن یه هدف به افشای اطلاعات یا انجام یه کار مشخص.

حملات مهندسی اجتماعی شکل‌های مختلفی دارن؛ مثلاً ساخت وب‌سایت‌های جعلی یا ارسال ایمیل‌های مخرب (Phishing)، تماس با کاربرهای بی‌خبر و فریب‌دادنشون برای افشای اطلاعات حساس (Vishing)، یا درخواست از یه نفر برای نگه‌داشتن در یا کشیدن کارت شناسایی، با هدف ورود غیرمجاز به یه محیط فیزیکی امن (Tailgating).

Virtual Private Network (VPN) — شبکه خصوصی مجازی

یه تونل نقطه‌به‌نقطه که ترافیک شبکه بین کامپیوتر و یه شبکه راه دور (که معمولاً عمومی هست) رو رمزنگاری می‌کنه.

VPNها تو عملیات امنیت تهاجمی کاربردهای مختلفی دارن؛ از جمله دورزدن محدودیت‌های شبکه، مخفی‌کردن الگوی ترافیک و انتقال امن داده‌ها.

البته اصطلاحات و مفاهیم مهم دیگه‌ای هم تو فصل‌های مربوط به خودشون تعریف می‌شن، اما این موارد قراره بارها تو طول کتاب تکرار بشن؛ پس مهمه که درک درستی ازشون داشته باشی.

ملاحظات اخلاقی در ردتیمینگ

یه نکته خیلی مهم رو باید درک کنی: ردتیمینگ به‌هیچ‌وجه مجوز انجام فعالیت‌های مجرمانه نیست؛ حتی اگه اون فعالیت‌ها از نظر فنی خیلی مؤثر باشن.

@Sec_Book_Mind

مثلاً تهدیدکردن یه نفر با اسلحه برای گرفتن رمز عبورش شاید روش فوق‌العاده مؤثری باشه، اما هم غیرقانونیه و هم از نظر اخلاقی کاملاً غیرقابل قبوله. هیچ مشتری سازمانی یا تیم حقوقی‌ای هم چنین عملیاتی رو تأیید نمی‌کنه.

ممکنه یه مجرم واقعی از چنین روشی استفاده کنه، اما آزمایش‌کردن این روش هیچ ارزشی نداره؛ چون از مرزهای قانونی و اخلاقی عبور می‌کنه.

بااین‌حال، روان‌شناسی پشت این روش، یعنی استفاده از ترس برای متقاعدکردن افراد، چیزیه که اگه با درخواست مشتری و قوانین و محدوده عملیاتی (Rules of Engagement) مطابقت داشته باشه، می‌تونی به شکل اخلاقی تو یه عملیات ردتیم ازش استفاده کنی.

برای مثال، می‌تونی یه ایمیل فیشینگ برای کاربر بفرستی و بهش اطلاع بدی که حساب کاربریش به خطر افتاده و باید هرچه سریع‌تر روی لینک فیشینگ کلیک کنه و رمز عبورش رو تغییر بده.

این کار حس فوریت و ترس رو تو کاربر ایجاد می‌کنه و ممکنه باعث بشه به درخواستت واکنش نشون بده؛ بدون اینکه واقعاً آسیبی بهش وارد کنی.

مرزهای اخلاقی ردتیمینگ تو صنایع مختلف تفاوت زیادی با هم دارن. برای مثال، قوانین و محدوده عملیاتی یه ردتیم نظامی با ردتیمی که قراره امنیت یه شرکت مالی رو ارزیابی کنه، کاملاً متفاوته.

هیچ نسخه واحدی برای تشخیص اینکه چه کاری درسته و چه کاری غلط وجود نداره. این شرایط و ماهیت صنعت موردنظر هست که مشخص می‌کنه تو یه پروژه امنیتی چه کارهایی لازمه، چه کارهایی مجازن و چه کارهایی از نظر اخلاقی قابل قبول هستن.

@Sec_Book_Mind

یادت باشه هدف اصلی یه ردتیم فقط رسیدن به هدف تعیین‌شده نیست؛ بلکه در نهایت باید به بهبود امنیت هم کمک کنه.

در اصل، اعضای ردتیم متخصص‌های امنیتی‌ای هستن که برای قوی‌ترکردن سیستم‌های دفاعی تلاش می‌کنن.

برای درک بهتر این موضوع، امنیت یه خونه رو در نظر بگیر. اگه تو یه خونه زندگی می‌کنی، احتمالاً روی درهاش قفل داری و شاید حتی یه سیستم دزدگیر هم نصب کرده باشی. این تجهیزات برای جلوگیری از فعالیت‌های مجرمانه و باخبرکردنت از اتفاقات مشکوک طراحی شدن.

درسته که می‌شه قفل‌ها رو باز کرد و سیستم‌های دزدگیر هم صرفاً هشدار می‌دن، اما هر دوی این‌ها زمان ارزشمندی برات می‌خرن تا بتونی با پلیس تماس بگیری یا از خودت دفاع کنی.

حالا فرض کن پیچ‌های قفل از بیرون کاملاً در دسترس باشن و هر کسی بتونه به‌راحتی قفل رو باز کنه و از در جداش کنه. یا مثلاً سیم‌کشی سیستم دزدگیر ایراد داشته باشه و نتونه به‌موقع به پلیس هشدار بده.

مسلماً دوست داری قبل از اینکه یه سارق متوجه این ضعف‌ها بشه و به خونه‌ات دستبرد بزنه، خودت از وجودشون باخبر بشی، نه؟

این مثال نشون می‌ده چرا باید تجهیزات و کنترل‌های امنیتی رو واقعاً آزمایش کنیم، نه اینکه کورکورانه بهشون اعتماد داشته باشیم.

اینجاست که ارزش واقعی ردتیم مشخص می‌شه.

@Sec_Book_Mind

وقتی یه هدف مشخص به ردتیم داده می‌شه، اعضای تیم تلاش می‌کنن با استفاده از هر روش منطقی و قابل قبولی به اون هدف برسن؛ درست مثل کاری که مجرم‌های دنیای واقعی انجام می‌دن.

بعد از پایان ارزیابی، تیم به سازمان کمک می‌کنه تا کنترل‌ها یا فرایندهایی رو که عملکرد درستی نداشتن، اصلاح کنه. همچنین راهکارهای لازم برای برطرف‌کردن ضعف‌های امنیتی شناسایی‌شده رو ارائه می‌ده تا یه مجرم واقعی نتونه همون آسیب‌هایی رو وارد کنه که ردتیم در جریان ارزیابی بهشون دست پیدا کرده.

تست‌های فنی مثل تست نفوذ هم ارزش مشابهی دارن، اما تمرکزشون فقط روی آسیب‌پذیری‌های فنیه. ردتیمینگ علاوه بر مشکلات فنی، ایرادهای موجود تو فرایندهای کسب‌وکار، ضعف‌های امنیت فیزیکی و خطاهای انسانی رو هم شناسایی می‌کنه.

عملیات ردتیم (Red Team Operations)

عملیات ردتیم در اصل همون عملیات زیرساختی (Infrastructure Operations) هستن؛ البته چندتا تفاوت و نکته ظریف وجود داره که باید حواست بهشون باشه.

برای راه‌اندازی زیرساخت ردتیم می‌تونی از همون ابزارها و فناوری‌هایی استفاده کنی که تو عملیات معمول زیرساختی به کار می‌برن.

مثلاً می‌تونی از Salt Project به‌عنوان C2 (که تو فصل V درباره‌ش صحبت می‌کنیم) استفاده کنی، با Terraform سرورها رو راه‌اندازی کنی و حتی از Serverless Framework برای پیکربندی C2 Redirectorها کمک بگیری.

@Sec_Book_Mind

تفاوت اصلی بین IT سنتی و عملیات امنیت تهاجمی، نه تو ابزارهاست و نه تو مهارت‌های پایه‌ای موردنیاز؛ چون همون‌طور که گفتم، این‌ها عملاً یکی هستن. تفاوت واقعی تو نحوه استفاده از این ابزارها و مهارت‌هاست.

برای مثال، پیکربندی یه سرور C2 رو در نظر بگیر. یه مهندس پلتفرم معمولی هیچ‌وقت مجبور نیست به چنین موضوعی فکر کنه.

اما تو دنیای امنیت تهاجمی، ممکنه بخوای از طریق HTTP/HTTPS، TCP، UDP یا حتی Domain Name System (DNS) یه شل دریافت کنی. هر کدوم از این پروتکل‌ها هم نیازمندی‌ها و مراحل پیکربندی خاص خودشون رو دارن.

به‌خصوص DNS به تنظیمات غیرمعمولی برای دامنه نیاز داره تا سرور C2 بتونه از طریق اون دستور دریافت کنه. در واقع، دریافت دستور از طریق کوئری‌های DNS اصلاً کاربرد اصلی این پروتکل نیست؛ اما این دقیقاً همون روحیه هکریه که باعث می‌شه از فناوری‌های موجود خلاقانه استفاده کنی و راه‌های جدیدی برای به‌کارگرفتنشون پیدا کنی؛ طوری که هم پایدار باشن و هم درست کار کنن.

عملیات ردتیم فراتر از مهندسی معمول زیرساخت، به یه سری ملاحظات خاص دیگه هم نیاز دارن. مثلاً:

- پیکربندی مسیریابی شبکه برای خارج‌کردن داده‌ها از سیستم هدف (Data Exfiltration) یا واردکردن داده به اون (Data Infiltration).
- مدیریت مجوزها یا دسترسی به دیتابیس‌های مشخص برای استفاده از ابزارهای مخصوص امنیت تهاجمی.

@Sec_Book_Mind

•در نظرگرفتن تدابیر امنیتی و رمزنگاری قوی‌تر برای محافظت از داده‌های مشتری.

تو امنیت تهاجمی با سناریوهای نسبتاً خاصی روبه‌رو می‌شی که یه مهندس پلتفرم یا مدیر سیستم معمولی احتمالاً هیچ‌وقت نیازی به درنظرگرفتنشون نداشته باشه.

برای مثال، Gophish رو در نظر بگیر. می‌تونی با ترکیب Gophish و Evilginx، قابلیت‌هایی برای اجرای تکنیک‌های دورزدن احراز هویت چندمرحله‌ای (MFA) بهش اضافه کنی. یا اینکه با استفاده از تکنیک Reverse VPN Tunneling که تو فصل ۶ نشون داده شده، یه Drop Box راه‌اندازی کنی.

این نوع پیاده‌سازی‌ها تفاوت زیادی با کاربرد معمول این ابزارها تو یه محیط زیرساختی سنتی دارن.

البته استفاده از این ابزارها اجباری نیست. چیزی که بیشتر از هر چیز اهمیت داره، اینه که یه مهندس ماهر باشی و با ذهنیت امنیت تهاجمی به مسائل نگاه کنی؛ همین موضوعه که باعث می‌شه بتونی به‌خوبی از عملیات یه ردتیم پشتیبانی کنی.

برای همراهی با کتاب به چه چیزهایی نیاز داری؟

برای اینکه بتونی تمرین‌های این کتاب رو انجام بدی، باید یه محیط توسعه (Development Environment) برای خودت آماده کنی. بالاخره هدف اصلی، یادگیری برنامه‌نویسیه!

تمام ابزارهایی که تو این کتاب استفاده می‌شن متن‌باز (Open Source) هستن؛ به‌جز چند مورد از ابزارهای خودکاری که در پایان بعضی فصل‌ها صرفاً به‌عنوان ابزارهای جانبی و قابل‌توجه معرفی می‌شن.

@Sec_Book_Mind

تمام ابزارهای اصلی توسعه نرم‌افزار نه‌تنها متن‌باز هستن، بلکه روی سیستم‌عامل‌های مختلف هم اجرا می‌شن (Cross-Platform). یعنی می‌تونی روی هر سیستم‌عاملی که باهاش راحت‌تری، کدنویسی و توسعه رو انجام بدی.

اگه تصمیم داری تمرین‌های هر فصل رو هم‌زمان با خوندن کتاب انجام بدی، به این موارد نیاز داری:

- سیستم‌عامل Ubuntu Linux 22.04 LTS یا نسخه‌های جدیدتر
- PHP
- Go
- PHP MySQL
- MySQL Server
- Python 3.x
- pip
- ماژول‌های pip: شامل requests و pysmb
- Visual Studio (VS) Code: https://code.visualstudio.com/download

این نکته رو هم در نظر داشته باش که سرعت پیشرفت نرم‌افزارها خیلی بیشتر از سرعت به‌روزرسانی کتاب‌هاست!

دانش و مهارت‌هایی که از این کتاب به دست میاری، احتمالاً تا چندین دهه آینده هم به دردت می‌خورن؛ اما ابزارهای دقیقی که تو کتاب معرفی شدن، قطعاً با گذشت زمان تغییر می‌کنن.

@Sec_Book_Mind

اگه هنگام انجام تمرین‌های هر فصل نتونستی یه ابزار مشخص رو نصب کنی، سعی کن در صورت امکان از نسخه جدیدتر یا ابزار جایگزین اون استفاده کنی.

اگه هم تصمیم گرفتی از یه پیکربندی کاملاً متفاوت استفاده کنی، دیگه مسئولیت راه‌اندازی و تنظیماتش با خودته؛ البته این هم خودش یه چالش مفید برای یادگیریه.

یادگیری یه چیزه، اما حرفه‌ای‌شدن یعنی بتونی هر وقت لازم بود یا خودت خواستی، شرایط رو با نیازهای جدید تطبیق بدی و در نهایت به همون نتیجه برسی.

اگه از Ubuntu استفاده می‌کنی، می‌تونی با اجرای دستور زیر بخش زیادی از پیش‌نیازها رو به‌راحتی نصب کنی:

```
$ apt install php php-mysql mysql-server golang-go python3 python3-pip vim
```

ابزار Vim هم تو این دستور قرار داده شده تا اگه دوست نداری از VS Code استفاده کنی، یه ویرایشگر متن جایگزین داشته باشی.

البته هر ویرایشگر متنی که باهاش راحتی قابل استفاده‌ست؛ فقط کافیه بتونی باهاش کد بنویسی!

تو این کتاب از چندین زبان برنامه‌نویسی استفاده می‌کنیم تا نشون بدیم چطور می‌شه یه کار مشخص رو تو شرایط مختلف و با زبان‌های متفاوت انجام داد.

هر زبان برنامه‌نویسی نقاط قوت و قابلیت‌های خاص خودش رو داره، اما در نهایت همه‌شون یه هدف مشترک رو دنبال می‌کنن: ارسال دستورالعمل‌هایی که سیستم‌عامل یا سخت‌افزار کامپیوتر بتونه اجراشون کنه.

@Sec_Book_Mind

تا وقتی یه زبان برنامه‌نویسی بتونه این کار رو انجام بده، می‌تونی ازش برای ساخت هر ابزاری که می‌خوای استفاده کنی.

تو این کتاب، PHP زبان اصلی ما برای توسعه بخش سمت سرور (Server-Side) اپلیکیشن‌های وب خواهد بود. برای ذخیره و مدیریت داده‌های این اپلیکیشن‌ها هم از MySQL استفاده می‌کنیم؛ مخصوصاً وقتی قراره اپلیکیشن‌های فول‌استک (Full-Stack) کوچیک خودمون رو بسازیم.

برای توسعه بخش فرانت‌اند این اپلیکیشن‌ها هم از Hypertext Markup Language (HTML)، Cascading Style Sheets (CSS) و JavaScript استفاده می‌کنیم.

وقتی نوبت به ساخت اسکریپت‌های مربوط به حملات احراز هویت و بعضی از برنامه‌های مرتبط با C2 برسه، Python حسابی به کارمون میاد.

از Go هم برای ساخت ابزارهای مرتبط با C2 و Cross-Compile کردن باینری‌ها در فرایند توسعه Implant استفاده خواهیم کرد.

این زبان‌ها برای انجام کارهایی که گفتیم انتخاب‌های خیلی خوبی هستن، اما تنها گزینه‌های موجود نیستن و قطعاً هر کدوم محدودیت‌ها و نقاط ضعف خودشون رو هم دارن.

بعد از تموم‌کردن هر فصل، بد نیست خودت رو به چالش بکشی و سعی کنی پروژه‌ای رو که ساختی، از صفر تا صد با یه زبان برنامه‌نویسی دیگه هم پیاده‌سازی کنی.

مثلاً سعی کن سرور C2 و Implant خودت رو به‌طور کامل با PHP بسازی.

شاید نتونی Implant رو به‌سادگی Go به شکل یه فایل باینری قابل اجرا منتشر کنی، اما شک نکن که می‌تونی یه راه برای انجامش پیدا کنی!

@Sec_Book_Mind

ساختار این کتاب چطوریه؟

این کتاب از سه بخش و ۱۲ فصل تشکیل شده.

بخش اول به جنبه توسعه نرم‌افزار در امنیت تهاجمی می‌پردازه، بخش دوم روی مهندسی زیرساخت تمرکز داره و بخش سوم هم با سناریوهای دنیای واقعی به پایان می‌رسه.

ایده کلی اینه که اول مهارت‌های برنامه‌نویسی‌ات رو تقویت کنی و با ابزارهایی که برای اجرای عملیات امنیت تهاجمی بهشون نیاز داری آشنا بشی. بعد تو فصل‌های مربوط به مهندسی، همین مهارت‌ها و ابزارها رو تو تمرین‌های مختلف و سناریوهای شبیه‌سازی‌شده امنیت تهاجمی به کار بگیری.

فصل ۰: ابزارهای حرفه هک (Tools of the Tradecraft)

تو این فصل با دیدگاه متفاوتی نسبت به ردتیمینگ آشنا می‌شی و می‌بینی چه تفاوتی با سایر روش‌های تست امنیت تهاجمی داره.

همچنین بررسی می‌کنی که ابزارهای امنیت تهاجمی واقعاً چی هستن و نگاهی به کد بعضی از ابزارها و مجموعه‌ابزارهای متن‌بازی می‌ندازی که مهاجم‌های دنیای واقعی ازشون استفاده می‌کنن.

هدف این فصل اینه که با این دیدگاه آشنا بشی که بدافزار هم در اصل یه نرم‌افزاره؛ و از همین ابتدا بدونی ابزارهایی از این دست، پشت صحنه چطور کار می‌کنن.

@sec_Book_Mind

بخش اول: توسعه در امنیت تهاجمی (Offensive Security Development)

فصل ۱: اکسپلویت‌های اپلیکیشن‌های وب (Web Application Exploits)

تو این فصل یاد می‌گیری چطور اپلیکیشن‌های وب اختصاصی بسازی و ازشون تو سناریوهای مختلف امنیت تهاجمی استفاده کنی.

با ساخت صفحات فرود (Landing Page) ساده برای کاربردهای مختلف آشنا می‌شی و یاد می‌گیری چطور با JavaScript، PHP و Structured Query Language (SQL) یه اپلیکیشن فول‌استک بسازی که بتونه برای حملاتی مثل فیشینگ و سرقت اطلاعات ورود کاربران (Credential Harvesting) استفاده بشه.

فصل ۲: حملات احراز هویت (Authentication Attacks)

این فصل به نحوه عملکرد حملات Brute-Force و Password Spraying می‌پردازه.

@Sec_Book_Mind

اول ابزارهای اختصاصی خودت رو برای بررسی ترافیک وب و پروکسی‌کردن داده‌ها می‌سازی. بعد یاد می‌گیری چطور این داده‌ها رو خودکارسازی کنی و ازشون برای اجرای حملات احراز هویت استفاده کنی.

تو هر سناریوی این فصل، به یه پروتکل شبکه یا اپلیکیشن مشخص متصل می‌شی، ابزاری برای Brute-Force کردن صفحه ورودش می‌سازی و بعد یه ابزار دیگه توسعه می‌دی که بتونه حمله Password Spraying رو علیه همون هدف اجرا کنه.

فصل ۳: توسعه و توزیع بدافزار اختصاصی (Custom Malware Development and Distribution)

تو این فصل با نحوه ساخت بدافزارهای اختصاصی آشنا می‌شی.

یاد می‌گیری چطور با زبان‌هایی مثل Go برنامه بنویسی که انتشار بدافزار به شکل فایل‌های اجرایی رو راحت‌تر می‌کنن.

با استفاده از این دانش، یه سرور C2 و Implant اختصاصی می‌سازی و در نهایت یه Ransomware سفارشی توسعه می‌دی که بشه با رعایت ملاحظات ایمنی، تو عملیات‌های واقعی هم ازش استفاده کرد.

بخش دوم: مهندسی امنیت تهاجمی (Offensive Security Engineering)

فصل ۴: خودکارسازی راه‌اندازی زیرساخت امنیت تهاجمی (Automating Offensive Security Infrastructure Deployment)

از این فصل وارد دنیای مهندسی زیرساخت برای امنیت تهاجمی می‌شی.

@Sec_Book_Mind

اینجا با موضوعاتی آشنا می‌شی که برای پیکربندی زیرساخت موردنیاز در عملیات امنیت تهاجمی بهشون نیاز داری.

مفاهیمی مثل زیرساخت به‌عنوان کد (Infrastructure as Code یا IaC) رو با استفاده از Terraform، Serverless Framework و Amazon Web Services (AWS) یاد می‌گیری و بعد محیط‌های اختصاصی خودت رو می‌سازی تا برای فصل‌های بعد آماده باشی.

فصل ۵: استفاده از مبانی شبکه در پیاده‌سازی C2 (Applying Network Fundamentals to C2 Implementation)

این فصل یه مرور کوتاه روی نحوه کار شبکه‌هاست، اما این بار از زاویه دید امنیت تهاجمی.

با مدل Open Systems Interconnection (OSI) آشنا می‌شی و چندین ساختار مختلف شبکه رو بررسی می‌کنی تا بهتر متوجه بشی ترافیک شبکه چطور بین بخش‌های مختلف جابه‌جا می‌شه.

بعد از این دانش استفاده می‌کنی تا با کمک فریم‌ورک Metasploit، یه سرور C2 و Implant رو پیکربندی کنی.

در پایان این فصل ابزار و دانشی رو به دست میاری که بهت کمک می‌کنه تو هر سناریو، پیکربندی مناسب C2 رو انتخاب کنی.

فصل ۶: تونل‌سازی معکوس VPN (Reverse VPN Tunneling)

تو این فصل وارد کاربردهای عملی‌تر مدیریت و کنترل جریان ترافیک شبکه در عملیات امنیت تهاجمی می‌شی.

@Sec_Book_Mind

این فصل ادامه مباحث فصل پنجم رو دنبال می‌کنه و بهت یاد می‌ده چطور یه تونل معکوس از طریق VPN راه‌اندازی کنی تا بتونی به شبکه‌ای دسترسی پیدا کنی که در حالت عادی از بیرون قابل دسترسی نیست.

فصل ۷: مدیریت زیرساخت در عملیات امنیت تهاجمی (Managing Infrastructure for Offensive Security Operations)

این فصل مفاهیم گسترده‌تری رو درباره مدیریت زیرساخت در عملیات امنیت تهاجمی معرفی می‌کنه.

با Salt Project کار می‌کنی تا پیکربندی سیستم‌ها رو خودکار کنی و وضعیت موردنظرت رو روی چندین هاست مختلف اعمال و حفظ کنی.

همچنین با نقش‌ها و مسئولیت‌های مختلف اعضای یه ردتیم آشنا می‌شی.

بخش سوم: امنیت تهاجمی در دنیای واقعی (Offensive Security in the Real World)

فصل ۸: اکسپلویت‌کردن با Metasploit (Exploitation with Metasploit)

تو این فصل مهارت‌هایی رو که تا اینجا یاد گرفتی، تو قالب یه سناریوی عملی و نزدیک به دنیای واقعی کنار هم به کار می‌گیری.

در جریان یه عملیات شبیه‌سازی‌شده، هم نقش اپراتور و هم نقش مهندس تیم امنیت تهاجمی رو بر عهده می‌گیری و با استفاده از دانش خودت درباره C2 و تکنیک‌های ساده اکسپلویت از طریق Metasploit، یه سرور رو به خطر می‌اندازی.

@sec_Book_Mind

فصل ۹: راه‌اندازی Drop Box (Deploying a Dropbox)

این فصل دومین سناریوی عملی کتاب رو ارائه می‌ده که بر اساس مباحث فصل ششم طراحی شده.

اینجا یه حمله جدید اجرا می‌کنی؛ نه‌تنها سرور Reverse VPN رو که تو فصل قبل ساخته بودی پیکربندی می‌کنی، بلکه از زیرساخت C2 خودت هم کمک می‌گیری تا فرصت‌های مختلف بعد از نفوذ (Post-Exploitation) رو با استفاده از فریم‌ورک PowerShell Empire C2 بررسی کنی.

فصل ۱۰: حمله فیشینگ با استفاده از C2 Redirectorها (Phishing Attack with C2 Redirectors)

این فصل تو رو به سناریوی نهایی کتاب می‌رسونه؛ جایی که قراره یه عملیات پیچیده امنیت تهاجمی رو از ابتدا تا انتها مدیریت کنی.

یه سرور فیشینگ و یه سرور C2 برای اجرای یه تمرین فیشینگ شبیه‌سازی‌شده راه‌اندازی می‌کنی تا ابزارهایی رو که در طول کتاب ساختی آزمایش کنی و مفاهیم مهندسی اجتماعی‌ای رو که تا اینجا یاد گرفتی، به کار بگیری.

فصل ۱۱: پیکربندی C2 چندنفره (Multiplayer C2 Configuration)

تو فصل آخر با دو مجموعه‌ابزار دیگه C2 آشنا می‌شی که قابلیت‌های بیشتری برای کار تیمی دارن: Teamserver و Sliver C2.

این ابزارها از نظر امکانات و کاربرد با هم تفاوت‌هایی دارن، اما هر دو برای شبیه‌سازی گروه‌های تهدیدکننده مشخص یا اجرای عملیات‌های پیچیده به‌صورت تیمی مفیدن؛ اون هم بدون اینکه اعضای تیم مجبور باشن مستقیماً یه رابط C2 مشترک رو با هم به اشتراک بذارن.

@Sec_Book_Mind

بعد از خوندن این فصل، گزینه‌های بیشتری برای ادامه مسیر و گسترش مهارت‌هات در اختیار خواهی داشت تا بتونی کم‌کم اون‌ها رو تو دنیای واقعی به کار بگیری.

خب، دیگه وقتشه دست‌به‌کار بشی و اولین کدهای خودت رو برای ساخت ابزارهای امنیت تهاجمی بنویسی!

@Sec_Book_Mind