سیستم‌شناسی و فازهای آزمایش نفوذپذیری استاندارد[1] در پیاده‌سازی حملات هدفمند و مکانیزم‌های دفاعی

پژوهشگر: نادیه فلاحتی[2]، پیام عسکرپور[3]، آرش کامجو[4]، سعید شهبازی[5]، امیرعلی عظیمی[6]، محمد امین مسلمی[7]، رضا احمدی[8]، مبین علی‌اکبری[9] و امیرحسین سهرابی[10]

استاد راهنما: میلاد کهساری الهادی[11]

دوره اول مهندسی معکوس و تحلیل باینری[12]

تاریخ ارائه: 13-8-1401

[1] Systemology and PTES in Implementation of Targeted Attacks and Defense Mechanisms

[2] nadiyeft@yahoo.com

[3] payam59@gmail.com

[4] arash.kamjoo.cs@gmail.com

[5] saeid.shahbazi@hotmail.com

[6] maz28787@gmail.com

[7] ma.moslemi.cs@gmail.com

[8] ahmadi.reza.cs@gmail.com

[9] mobin.aliakbariii@gmail.com

[10] a.sohrabi.cs@gmail.com

[11] m.kahsari@gmail.com

[12] https://aiooo.ir

فهرست

سیستم[1] چیست؟

تعریف‌های بسیار متنوع و متفاوتی از سیستم و تفکر سیستمی وجود دارد. با وجودی که تقریباً همه‌ی ما می‌توانیم به صورت حسی مفهوم سیستم را درک کنیم، اما عملاً نمی‌توانید یک روایت مشترک از تعریف سیستم در میان محققان و متفکران سیستمی بیابید.

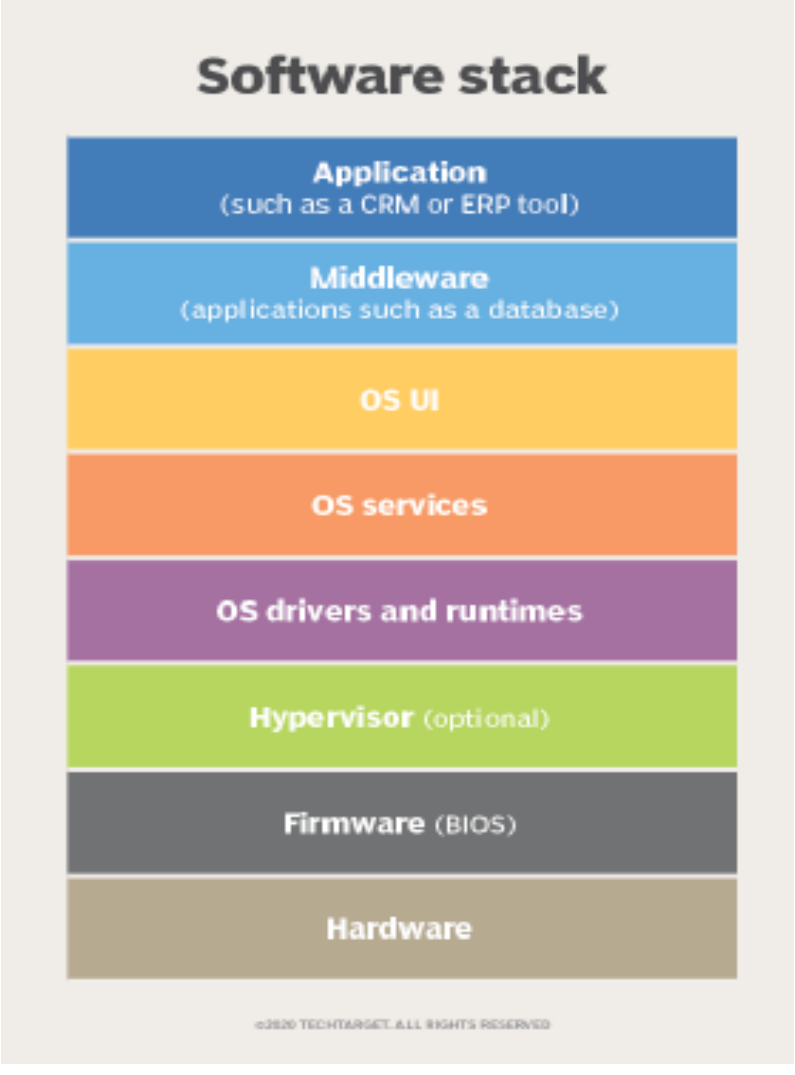
به هر روی، یک سیستم مجموعه‌ای سازمان یافته از قطعات، مولفه‌های اجرایی، عناصر عملیاتی یا مفاهیم انتزاعی هستند که برای دستیابی به یک هدف در تعامل یا مرتبط با هم دیگر عمل می‌کنند. یک سیستم دارای ورودی‌های مختلفی است که فرآیندهای خاصی را طی و در نهایت خروجی‌های مشخصی تولید کرده و هدف سیستم را محقق می‌کنند. شایان ذکر است، اگر بخشی از یک سیستم تغییر کند، ماهیت سیستم نیز تغییر می‌کند. اگر بخشی از سیستم درست کار نکند، در نهایت کل سیستم دچار مشکل و توقف خواهد شد. در زیر، برخی از ویژگی‌های مهم یک سیستم آورده شده است:

1. سیستم از اجزای متعدد تشکیل می‌شود.
2. اجزای سیستم با یکدیگر در ارتباط و تعامل هستند.
3. برای سیستم‌ها می‌توان رفتار تعریف کرد. با حذف هر یک از اجزای سیستم، رفتار کلی سیستم تغییر می‌کند.
4. معمولاً برای سیستم یک مرز تعریف می‌شود.
5. برای بسیاری از سیستم‌ها می‌توان هدف تعریف کرد.
6. سیستم‌ها را می‌توان به دو دسته‌ی باز و بسته تقسیم کرد.

به عبارت دیگر یک قطعه را به تنهایی به عنوان یک سیستم در نظر نمی‌گیریم. وقتی دو یا چند المان در کنار یکدیگر قرار می‌گیرند و مجموعه‌ی آنها – از دید ما – هویت پیدا می‌کند می‌توانیم بگوییم یک سیستم به وجود آمده است. به عنوان مثال، یک ماشین یک سیستم است. اگر کاربراتور را بردارید، دیگر ماشینی ندارید که کار کند. یک انسان هم یک سیستم است، اگر قلب او را بردارید، دیگر زنده نخواهد بود پس انسانی وجود نخواهد داشت. چون هم ماشین و هم انسان تشکیل شده از مجموعه‌ای زیرسیستم هستند که باید به شکل صحیح و درست کار کنند تا ماشین و انسان عملیاتی باقی بمانند. همچنین هر چیزی را که در قالب یک سیستم در نظر بگیریم، در ادامه تهدیداتی نیز متوجه آن خواهد بود. تهدیدات متوجه یک سیستم یا داخلی یا خارجی هستند. تهدیدات داخلی یک سیستم منشا

[1] System

درونی و تهدیدات خارجی منشا خارجی دارند که این تهدیدات می‌توانند هویت یک سیستم را در معرض خطر قرار بدهند.



تصویر 1: زیرسیستم‌های تشکیل‌دهنده کامپیوتر

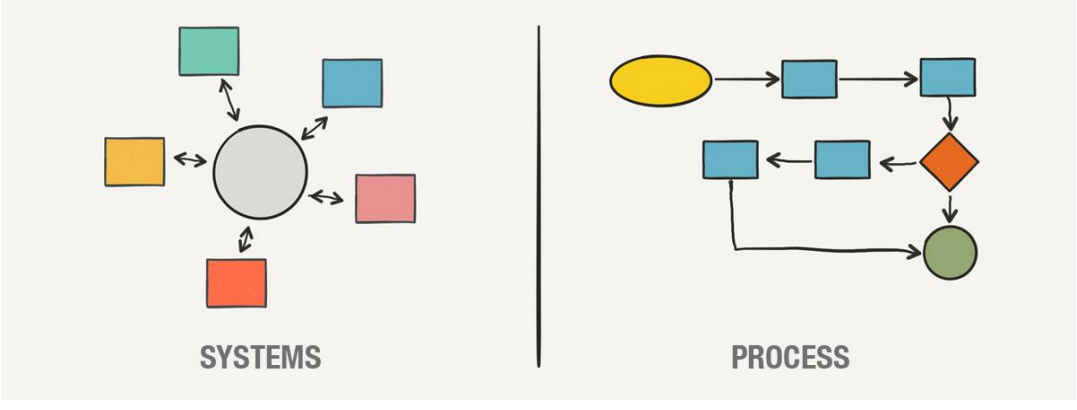
با توجه به این تعریف، یک کامپیوتر نیز در قالب یک سیستم تعریف می‌شود زیرا از اجزای مختلفی تشکیل می‌شود که با هم کار می‌کنند تا با استفاده از آن مجموعه وظایف محاسباتی و عملیاتی و کنترلی را انجام بدهیم. همانطور که در تصویر 1 نمایش داده شده است، هر کامپیوتر دارای اجزای مختلفی است و هر کدام از آن اجزا به صورت فردی برای هدف خاصی طراحی شده‌اند.

شایان ذکر است، هر کدام از اجزای کامپیوتر خود یک سیستم هستند، به همین دلیل آن سیستم‌ها خود به عنوان زیرسیستم‌های تشکیل‌دهنده کامپیوتر شناخته می‌شوند. وقتی تمامی آن زیرسیستم‌ها با هم به تعامل صحیح و درست می‌پردازند، یک ابرسیستم خلق می‌کنند که به عنوان کامپیوترها اکنون شناخته می‌شوند.

یک ابرسیستم مانند کامپیوتر شامل سخت‌افزار (زیرسیستم‌های فیزیکی)، نرم‌افزار (زیرسیستم‌های انتزاعی)، و یک یا چند کاربر می‌شوند که نحوه عملکرد زیرسیستم‌های کامپیوتر را کنترل و مدیریت می‌کنند. زیرسیستم سخت‌افزاری رایانه شامل پردازنده، حافظه اصلی، دیسک سخت، صفحه کلید، منبع برق و ... می‌شود. زیرسیستم نرم‌افزاری کامپیوتر شامل سیستم‌عامل و هر برنامه دیگری است که بر روی سخت‌افزار نصب می‌شوند تا بتوانیم آن زیرسیستم‌های فیزیکی را کنترل کنیم و مورد استفاده قرار بدهیم. همانطور که در تصویر 1 قابل مشاهده است، به صورت خلاصه، اجزای اساسی کامپیوتری شامل سخت‌افزار، نرم‌افزار، سفت‌افزار و انسان‌افزار می‌شود.

تفاوت بین سیستم و فرآیند[1]

در برخی از شرایط ممکن است افراد بین سیستم و فرآیند نتوانند تمییز قائل شوند و تصور کنند که هر دو اشاره به یک مفهوم دارند. در حالیکه سیستم و فرآیند دو مفهوم مجزا از هم هستند. در حالت کلی، سیستم مجموعه یا ترکیبی از اجزای انتزاعی یا فیزیکی است که یک کل پیچیده یا واحد را تشکیل می‌دهند.



تصویر 2: تفاوت بین یک فرایند و یک سیستم

به عنوان مثال، کامپیوترها نمونه خوبی از یک سیستم هستند. یک کامپیوتر این هدف را دارد که یک کاربر بتواند با آن عملیات محاسباتی، کنترلی، عملیاتی و ... انجام بدهد. از همین روی، یک سیستم کامپیوتری ارزش‌های متعددی برای ما از جمله ارائه یک منبع برای انجام محاسبات سنگین به صورت خودکار، انجام مسیریابی برای کشتی‌ها و هواپیماها و ...، امکان ایجاد ارتباط بین انسان‌ها و... ایجاد می‌کند. اما اگر همه زیرسیستم‌های تشکیل دهنده کامپیوتر را نداشته باشیم یا برخی از زیرسیستم‌ها به درستی کار نکنند، سیستم کامپیوتری ممکن است متوقف شود.

[1] Process

به آن زیرسیستم‌ها همچنین فرآیند گفته می‌شود چون یک وظیفه‌ای را در تشکیل ابرسیستم کامپیوتر بر عهده دارند. به همین دلیل، وقتی در مورد فرآیندها صحبت می‌کنیم، اشاره به تمام فعالیت‌های درون سیستمی داریم. علاوه بر این، می‌توانیم فرآیندها را به عنوان بخش کوچکتری از یک سیستم بزرگتر بدانیم. مهم است که فرآیندها در کاری که انجام می‌دهند مؤثر باشند تا سیستم بتواند به طور مؤثر اجرا شود. خلاصه، فرآیندها دنباله‌ای از فعالیت‌هایی هستند که برای ایجاد یک نتیجه خاص در قالب یک ابرسیستم در نظر گرفته شده‌اند. فرآیندها در قالب ابرسیستم کامپیوتر جریان‌های اطلاعاتی و منابع سخت‌افزاری و نرم‌افزاری را به یکدیگر پیوند می‌دهند تا ارزشی را ایجاد و ارائه دهند.

تعامل و ارتباط در نگاه سیستمی

یکی از مهم‌ترین ویژگی‌های یک سیستم این است که اجزای آن با هم تعامل دارند و بر روی یکدیگر تاثیر می‌گذارند. هر یک از اجزای یک سیستم را که انتخاب کنید، می‌توانید تشخیص دهید که بر روی کدامیک اجزای سیستم تأثیر می‌گذارد. همچنین با بررسی سیستم می‌توانید بگویید که آن جزء خود از چه بخش‌هایی از سیستم اثر می‌پذیرد. تا وقتی تعامل وجود ندارد صرفاً یک مجموعه داریم و نه یک سیستم. شاید این توضیحات برای شما بحث مهارت کار تیمی را تداعی کند.

یکی از بحث‌های رایج در کار تیمی، بررسی تفاوت تیم و گروه است. اگر بخواهیم از ادبیات سیستمی استفاده کنیم می‌توانیم بگوییم گروه، تیمی است که تعامل و تأثیرگذاری میان اعضای آن حذف شده است. یا بالعکس می‌توانیم بگوییم یک گروه از انسان‌ها را وقتی می‌توانیم به یک تیم تبدیل کنیم که با یکدیگر تعامل داشته باشند و بر روی کار یکدیگر تاثیر بگذارند و از یکدیگر تأثیر بپذیرند. وقتی اجزای سیستم با هم تعامل دارند و بر روی یکدیگر تأثیر می‌گذارند، به تدریج چیزی به نام رفتار سیستم به وجود می‌آید و پدیدار می‌شود. این رفتار، چیزی فراتر از ویژگی‌ها یا تغییرات یک بخش خاص است و از ترکیبِ تعامل همه‌ی اجزا تشکیل می‌شود.

مرز از نگاه سیستمی

تقریباً هر سیستمی که انتخاب کنید و در موردش حرف بزنید، مرز هم دارد؛ مگر اینکه بخواهید تمام عالم هستی را به عنوان یک سیستم بگیرید. مرز سیستم یک واقعیت بیرونی نیست. بلکه ما انسانها مرزها را تعریف می‌کنیم. به عنوان مثال، نظام جمهوری اسلامی ایران یک سیستم است که مرزی مشخص دارد و تمامی مسائل این سیستم در مرزی مشخص می شود که برای ایران تعریف شده است. فراتر از آن مرز، هر مسئله ای وجود داشته باشد، مرتبط با ابرسیستم جمهوری اسلامی ایران نخواهد بود. به هر صورت، بسیاری از راه‌حل‌های غیرسیستمی از آنجا ناشی می‌شوند که ما مرزهای سیستم خود را بسیار محدود تعریف می‌کنیم.

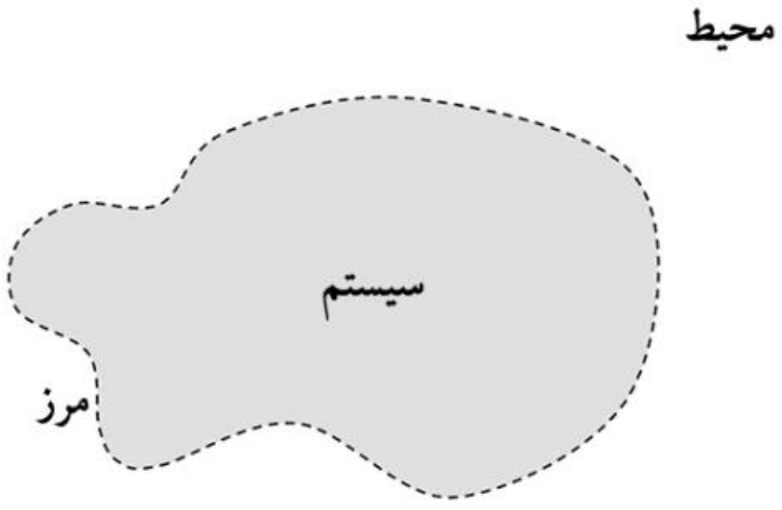
هدفمندی سیستم

همچنین شایان ذکر است، برای همه‌ی سیستم‌ها هدف تعریف نمی‌شود و سیستم حتماً نباید هدف داشته باشد. با همین استدلال (و استدلال‌های مشابه) کسانی که نظریه سیستم‌ها را به شکل عام می‌خوانند و دنبال می‌کنند، ترجیح می‌دهند این مورد را به عنوان یکی از ویژگی‌های یک سیستم مطرح نکنند. اما برای کسانی مثل ما که تفکر سیستمی را برای استفاده در درس‌های امنیت و اطلاعات می‌خوانند و می‌آموزند، مناسب‌تر است هدف و هدفمندی را هم به عنوان یک ویژگی سیستم در نظر بگیریم و فرض کنیم بر بسیاری از سیستم‌ها می‌توان یک یا چند هدف تعریف کرد.

ما اگر می‌گوییم یک کامپیوتر یا یک کشور را می‌توان یک سیستم در نظر گرفت و بعد می‌خواهیم به عنوان تحلیل‌گر سیستمی در مورد موفق بودن یا نبودن این سیستم‌ها صحبت کنیم، باید این موفقیت را در مقایسه با یک یا چند هدف بسنجیم. بنابراین معمولاً در تحلیل سیستم‌ها از هدف سیستم صحبت می‌شود. البته همه می‌دانیم که این ما هستیم که به عنوان تحلیل‌گر و ناظر برای سیستم هدف تعریف می‌کنیم.

سیستم باز و بسته

بعضی سیستم ها با دنیای اطراف خود در ارتباط هستند و یک سیستم باز را شکل می‌دهند. در حالی که ممکن است سیستم‌هایی را ببینیم که بسته هستند و هیچ رابطه‌ای با دنیای اطراف خود ندارند. از نظر تئوری، تنها سیستم بسته‌ی واقعی کلِ جهان است که دیگر چیزی در بیرون آن نیست که با آن تعامل داشته باشد و سایر سیستم‌ها همگی به نوعی سیستم باز محسوب می‌شوند.



تصویر 3: سیستم‌های باز و بسته

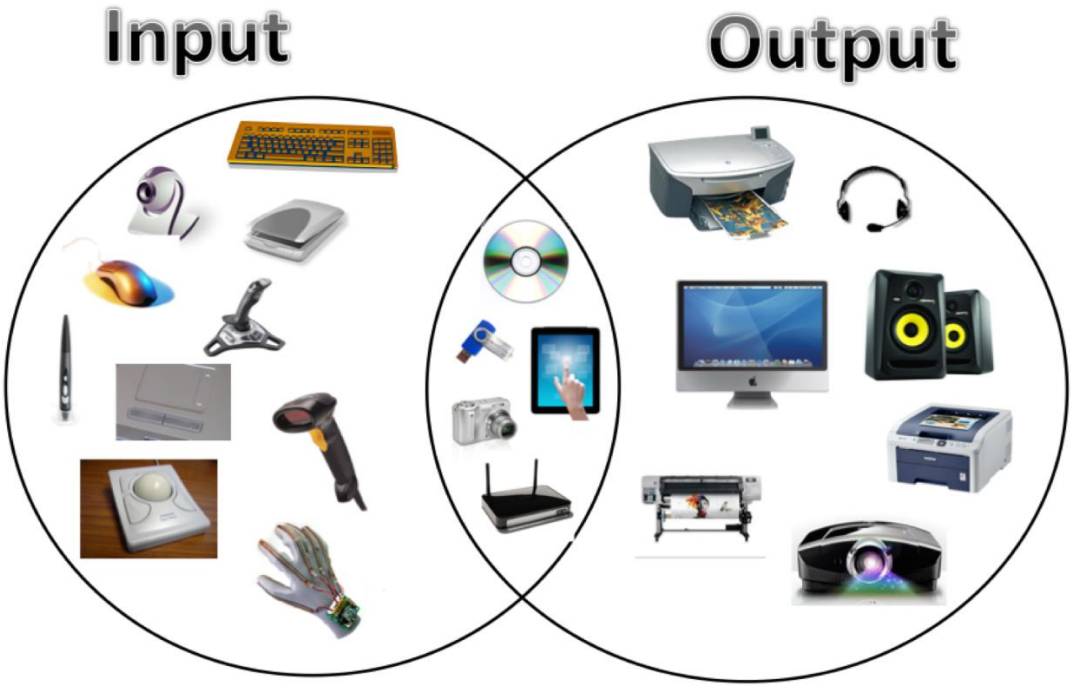
وقتی در تحلیل سیستم‌ها از اصطلاح سیستم بسته صحبت می‌کنیم منظورمان سیستمی است که تعامل آن با محیط، اندک است؛ یا اینکه ما تصمیم گرفته‌ایم از تعاملات آن با محیط صرف‌نظر کنیم.

یک سیستم چطور کار می‌کند؟

هر سیستم حداقل یک ورودی، مجموعه‌ای پروتکل با محوریت پردازش، و همچنین یک خروجی یا یک سرویس دارد. ورودی‌ها اقلامی هستند که توسط فرآیندهای مختلف در سیستم برای دستیابی به هدف کلی سیستم مورد استفاده قرار می‌گیرند. در سیستم‌های کامپیوتری، ورودی داده خام[1] است. این ورودی‌ها از دستگاه‌های مختلف دریافت می‌شوند و به سیستم کامپیوتری برای پردازش عبور داده خواهند شد. در ادامه برخی از دروازه‌های ورودی داده‌های خام به یک سیستم کامپیوتری معرفی شده است:

- **کیبورد:** کیبورد در دسته‌بندی ورودی‌ها قرار می‌گیرد. از کیبورد برای وارد کردن متن، شماره و به صورت کلی یک رشته از حروف استفاده می‌شود.
- **ماوس:** بعد از کیبورد، ماوس به عنوان یکی دیگر از راه‌های ورودی به کامپیوتر شناخته می‌شوند. از ماوس برای جابه‌جایی و ویرایش فایل‌ها استفاده می‌شود. البته به این موضوع باید توجه داشت که ماوس‌ها فقط در محیط‌های گرافیکی قابل استفاده هستند.
- **پویشر:** از پویشرها برای تبدیل پرونده‌های کاغذی به فایل‌های کامپیوتری استفاده می‌شود. برای مثال، می‌توانید با استفاده از پویشر صفحات یک مقاله را پویش کنید و عکس هر صفحه از مقاله را در کامپیوترتان دریافت کنید.
- **مودم:** مودم ها می توانند هم در دسته بندی ورودی ها، و هم در دسته بندی خروجی ها قرار بگیرد زیرا با استفاده از آن می توان در سطح شبکه های کامپیوتری بین ماشین های کامپیوتری به مبادله داده و اطلاعات پرداخت.
- **سنسورها:** در سیستم‌های کنترل صنعتی، تجهیزات PLC به منظور تصمیم‌سازی و کنترل فرایند تولید از سنسورها اطلاعات فیزیکی و محیطی را دریافت می‌کنند، و به یک PLC به منظور پردازش عبور خواهند داد.
- **میکروفن:** میکرفون‌ها برای فرستادن فایل‌های صوتی به کامپیوتر استفاده می‌شوند.
- **وب‌کم:** وبکم‌ها و همچنین دوربین‌ها دروازه دیگری از ورودی‌های تصویری به یک کامپیوتر به شمار می‌روند. اطلاعات تصویری محیط توسط وب کم به کامپیوتر ارائه می‌شود.

[1] Raw Data



تصویر 4: ورودی و خروجی‌های یک سیستمی کامپیوتری

فرآیندها مجموعه‌ای از فعالیت‌هایی هستند که توسط کامپیوتر انجام می‌شود که ورودی‌های مختلف را برای دستیابی به هدف تعریف شده خود دستکاری و پردازش می‌کند. به عنوان مثال، وقتی از میکروفون داده خامی در قالب صوت توسط کامپیوتر دریافت می‌شود، می‌تواند آن را پردازش و در ادامه توسط اسپیکر پخش کند یا وقتی توسط کارت شبکه داده های خام در قالب پاکت‌های شبکه را دریافت می‌کند، آن داده‌ها را پردازش کند و در صفحه نمایشگر نمایش بدهد. در تصویر 4، تجهیزات ورودی و خروجی یک سیستم کامپیوتری نمایش داده شده است. در ادامه دروازه‌هایی که در سیستم‌های کامپیوتری خروجی پردازش و یا دستکاری داده‌های خام ارائه می‌شود، آورده شده است:

- **مانیتور:** اصلی‌ترین و مهم‌ترین خروجی هر کامپیوتر، مانیتور می‌باشد. اگر که این قطعه دچار مشکل شود، دیگر نمی‌توان از کامپیوتر استفاده کرد. مانیتور یا همان صفحه نمایش کامپیوتر، ها داده های پردازش شده توسط سی پی یو را به صورت گرافیکی به کاربر نشان می‌دهد.
- **اسپیکر:** بعد از مانیتور ها، اسپیکرها از اهیمت بسیار بالایی هنگام استفاده از کامپیوتر برخودار هستند. از اسپیکر ها به عنوان خروجی فایل های صوتی و صدا های مختلف استفاده می‌شود. از آنجاکه وظیفه این قطعه پخش کردن صدا است، شما می‌توانید آن را با هدفون و یا هنذفری جایگزین کنید.
- **پرینتر:** همه افراد با پرینتر ها آشنا هستند. این وسایل ارزشمند، اسناد و فایل های تصویری و متنی کامپیوتری را برروی کاغذ چاپ می‌کنند و آن را به اسناد کاغذی تبدیل می‌کنند. این دستگاه را می توان مخالف پویشر دانست.معمولا از پرینتر ها مانند پویشر ها در مکان هایی استفاده می‌شود که روزانه تعداد زیادی فایل را پرینت یا چاپ می‌کنند.

- **ویدیو کارت:** ویدیو کارت، از جمله خروجی هایی است که در کیس قرار دارد. وظیفه ویدیو کارت این است گرافیک های پردازش شده را به سمت مانیتور هدایت کند. در واقع این وسلیه مکمل مانیتور است. یکی از مدل های مشهور ویدیو کارت ها، کارت های گرافیک هستند که علاوه بر پردازنده اصلی کامپیوتر، آنها نیز پردازنده ای دارند و تغییراتی برروی گرافیک ها ایجاد می‌کنند.
- **عملگر[1]:** وظیفه عملگر یا اکتواتور که نوعی موتور است در شیرهای کنترل، تأمین نیروی لازم جهت باز و بسته کردن شیر و قراردادن آن در موقعیت مطلوب و متناسب با سیگنال ارسالی از طرف کنترلر است. اکچوایتورها در انواع مختلف پنوماتیکی، الکتریکی یا خود عملگرموجود بوده وبا توجه به وضعیت سامانه انتخاب و استفاده می‌گردند.

در نهایت بعد شناخت ورودی، قواعد پردازش و خروجی به مسئله اهداف با محوریت سیستم‌های کامپیوتری خواهیم رسید. اهداف، نتایج نهایی هستند که سیستم می‌خواهد به آن دست یابد. در نگاه ما همه سیستم‌ها هدفمند هستند که می توانیم با خروجی که تولید میکنند، آن اهداف را مورد بررسی قرار بدهیم. اهداف کلی یک ماشین کامپیوتری معمولاً بر اساس مأموریت یا هدف آن توصیف می شود. به عنوان مثال، وقتی هدف یک ماشین کامپیوتری مسیریابی است، وقتی ورودی های مختصات را از ماهواره توسط آنتن های خود دریافت می کند، بعد پردازش باید بر روی نقشه مسیری را مشخص کند. اگر چنین کاری را انجام ندهد، آن ماشین کامپیوتری در مقایسه با هدف خود صحیح کار نمی کند.

متدولوژی سیستم‌های نرم[2]

متدولوژی سیستم‌های نرم، براي اولين بار در سال 1972 توسط پیتر چکلند در دانشگاه لنکستر مطرح شد. نخستین مقاله در باب SSM با عنوان" به سوی یک متدولوژی سیستم محور برای حل مسائل دنیای واقعی" در سال 1972 در مجله مهندسی سیستم‌ها توسط پیتر چکلند ارائه شد. متدولوژی سیستم‌های نرم یک فرآیند سازمان یافته و انعطاف‌پذیر برای مواجهه با موقعیت‌های مسئله‌زا است.

شایان ذکر است، مهندسی سیستم‌ها سرچشمه ومنشا علمی متدولوژی سیستم‌های نرم است. رویکرد مهندسی سیستم: انتخاب وسیله‌ای مناسب برای دستیابی به هدفی مشخص و شفاف که از ابتدا تعیین شده است. پیتر چکلند سیستم‌ها را به پنج دسته سیستم طبیعی، فیزیکی طراحی شده، انتزاعی طراحی شده، فعالیت انسانی و

[1] Actuator

[2] Soft systems methodology

متعالی تقسیم می‌کند. ولی در حالت کلی، تمامی سیستم‌ها موجود را می‌توانیم به صورت زیر تقسیم‌بندی یا کلاس‌بندی کنیم:

- سیستم‌های فیزیکی[1] که خود شامل زیرسیستم‌های زیر می‌شود:
  - سیستم‌های اورگانیکی[2] مانند انسان
  - سیستم‌های مکانیکی[3] مانند ماشین
  - سیستم‌های سایبرنتیک[4] مانند کامپیوتر
  - سیستم‌های بیولوژیکی[5] مانند قلب
  - سیستم‌های زیستی[6] مانند شکار
- سیستم‌های متافیزیکی[7] که خود شامل زیرسیستم‌های زیر می‌شود:
  - سیستم‌های منطقی[8]
  - سیستم‌های ریاضی[9]
  - سیستم‌های شیمیایی[10]
- سیستم‌های ماورایی[11]

با این حال، تمامی این سیستم‌ها، فارغ از اینکه در چه کلاسی قرار دارند، دارای تهدید خواهند بود. همانطور که پیش از این ذکر شد، هر چیزی را که در قالب یک سیستم در نظر پرفته شود، در ادامه برای آن تهدیدات داخلی و تهدید خارجی تعریف می‌شود. در ادامه این مقاله، به تشریح برخی از این تهدیدها و همچنین استانداردی به منظور سنجش امنیت سیستم‌های کامپیوتری خواهیم پرداخت.

---

[1] Physical Systems
[2] Organically Systems
[3] Mechanical Systems
[4] Cybernetic Systems
[5] Biological Systems
[6] Ecological Systems
[7] Metaphysical
[8] Logical Systems
[9] Mathematical Systems
[10] Chemical Systems
[11] Transphysical

تهدید[1] چیست؟

تهدید امنیتی به عنوان خطری تعریف می‌شود که به طور بالقوه می‌تواند به سیستم‌ها آسیب برساند. علت می‌تواند فیزیکی باشد، مانند سرقت کامپیوتری که حاوی داده‌های حیاتی است. علت نیز می‌تواند غیر فیزیکی مانند حمله یک بدافزار باشد. در امنیت سیستم‌های کامپیوتری، تهدید یک اقدام یا رویداد منفی بالقوه است که توسط یک آسیب‌پذیری تسهیل می‌شود که منجر به تأثیر ناخواسته بر یک سیستم یا برنامه رایانه خواهد شد.

یک تهدید می‌تواند یک رویداد منفی عمدی (به عنوان مثال نفوذ به سیستم‌های کامپیوتری موجود در یک پتروشیمی) یا یک رویداد منفی تصادفی (مثلاً احتمال خرابی کامپیوتر مانند سوختن منبع تقذیه) باشد. در ادامه برای تمامی سیستم‌های سایبرنتیک از جمله کامپیوترها تهدیدات زیر را می‌توان در نظر گرفت. شایان ذکر است، مواردی که در ادامه ذکر شده است، تهدیدات اصلی بر علیه امنیت سیستم‌های سایبرنتیکی است:

1. بدافزارها
2. حملات کانال جانبی
3. حملات منع سرویس
4. حملات مردی در میان
5. تهدیدات پیشرفته مستمر
6. حملات مهندسی اجتماعی
7. حملات جستجوی فراگیر گذرواژه ها
8. حملات آلوده‌سازی زنجیره تامین تجهیزات

در ادامه هر یک از حملات با محوریت سامانه‌های سایبرنتیک مورد تشریح قرار خواهد گرفت. شایان ذکر است، هرکدام از این حملات (تهدیدات بر علیه سیستم‌های کامپیوتری) دارای گونه‌های متفاوتی است. به عنوان مثال، وقتی در مورد تهدید بدافزارها بر علیه سیستم‌های کامپیوتری صحبت می‌کنیم، با انواع بدافزارها رو به رو هستیم که هر کدام با هدف خاصی توسعه داده می‌شوند یا وقتی در مورد حملات تحت شبکه صحبت می‌کنیم، انواع حملات در این لایه از جمله شنود ارتباطات تا حرکت عمقی[2] در سطح شبکه‌های داخلی وجود دارد.

---

[1] Threat

[2] Lateral Movement

تهدید بدافزارها[1]

مهاجمان از روش‌های زیادی برای وارد کردن بدافزار به دستگاه اهداف خود استفاده می‌کنند. مهاجمان برای آلوده کردن اهداف خود به بدافزار از حملات مهندسی اجتماعی تا استفاده از آسیب‌پذیری‌های روز صفر استفاده می‌کنند. در حملات مبتنی بر مهندسی اجتماعی، مهاجمان با استفاده از مسائل روانشناختی انسان‌ها، اهداف خود را وادار می‌کنند که اقدامی مانند کلیک روی لینک یا باز کردن یک پیوست را انجام دهند.

در موارد دیگر مهاجمان، از آسیب‌پذیری‌های موجود در مرورگرها یا سیستم‌عامل‌ها برای نصب بدافزار بدون اطلاع یا رضایت کاربر استفاده می‌کنند. هنگامی که بدافزار نصب شد، می‌تواند فعالیت‌های کاربر را نظارت کند، داده‌های محرمانه را برای مهاجم ارسال کند، به مهاجم در نفوذ به اهداف دیگر در شبکه کمک کند، و حتی باعث شود دستگاه کاربر وارد یک شبکه از بات‌نت‌ها شود و توسط مهاجم برای اهداف مخرب دیگر استفاده شود. بدافزارها شامل موارد زیر می‌شوند:

- **ویروس[2]:** مشابه یک ویروس واقعی، این نوع بدافزار خود را به فایل‌های سالم رایانه‌تان متصل می‌کند و سپس تکثیر می‌شود، پخش می‌شود و فایل‌های دیگر را نیز آلوده می‌کند.
- **کرم‌ها[3]:** کرم‌ها شبیه ویروس‌ها نوعی آلودگی هستند که برای آلوده کردن سیستم‌های دیگر تکثیر می‌شوند. با این حال، برخلاف ویروس‌ها، کرم ها برای انتشار نیازی به میزبان ندارند. کرم‌ها خودکفا هستند و می‌توانند با استفاده از اکسپلویت‌های زیرودی یا روز اول خود را گسترش بدهند.
- **تروجان‌ها[4]:** تروجان نام خود را از داستان اسب تروا گرفته است. تروجان‌ها به عنوان یک نرم‌افزار بی‌ضرر ظاهر می‌شوند و می‌توانند آغازگر حملات مختلفی به سیستم باشند. برخی از تروجان‌ها توسط کاربر و برخی دیگر بدون دخالت کاربر کار می‌کنند.
- **جاسوس‌افزار[5]:** نوع دیگری از بدافزار جاسوس‌افزارها هستند که به‌طور مخفیانه روی یک سیستم یا دستگاه نصب می‌شوند و فعالیت‌ها را برای جمع‌آوری اطلاعات رصد می‌کنند.

---

[1] Malware Threats

[2] Virus

[3] Worms

[4] Trojans

[5] Spyware

- **ربات‌ها[1]:** ربات‌ها اغلب برای خودکارسازی وظایف و تعامل پویا با بازدیدکنندگان وب‌سایت استفاده می‌شوند. متأسفانه، آنچه برای خیر استفاده می‌شود اغلب می‌تواند برای شر مورد استفاده قرار گیرد و ربات‌ها نمونه بارز آن هستند. بات‌نت‌ها مجدداً به یک سرور متصل می‌شوند و خود منتشر می‌شوند، که آن‌ها را به ویژه برای به خطر انداختن تعداد زیادی دستگاه مفید می‌کند. این یک تاکتیک رایج در حملات منع سرویس توزیع شده یا DDoS است. زیرا ماشین‌های آلوده که به عنوان ربات یا حتی زامبی[2] شناخته می‌شوند، می‌توانند برای ارسال ترافیک سنگین به یک ماشین مورد استفاده قرار بگیرند.
- **باج‌افزار[3]:** این دسته از بدافزارها اقدام به رمزگذاری بر روی فایل‌های موجود بر روی دیسک می‌کنند، با این هدف که پس از پرداخت هزینه، دسترسی را برای مالیک آن اطلاعات بازیابی کنند.
- **روت‌کیت‌ها[4]:** روت‌کیت‌ها شکلی از بدافزار در سطح کرنل سیستم‌عامل (درایورها یا ماژول‌های سیستم مخربی) هستند که بعد از راه‌اندازی سیستم‌عامل اجرا می‌شوند و می‌توانند با سطح دسترسی بالا (سطح دسترسی سیستم[5] در ویندوز و سطح دسترسی روت[6] در لینوکس) اقدام به دستکاری ساختمان داده‌های سیستم‌عامل کنند. به عنوان مثال، می توانند یک پروسه عملیات را مخفی کنند، سوکت بازی را مخفی کنند، توابع مهم سیستمی را دستکاری و هوک کنند و ...
- **بوت‌کیت‌ها[7]:** بوت‌کیت نوعی دیگر از کیت های مخرب هستند که در سیستم هایی که از فریمور BIOS استفاده می کنند، می توانند با اعمال تغییرات بر روی Master Boot Record و Volume Boot Record دیسک سخت اقدام به دستکاری فرایند بارگزاری سامانه کنند و به صورت کامل ماشین را تحت اختیار خود بگیرند. بوتکیت ها مبتنی بر نوع فریمور ماشین از جمله BIOS یا UEFI عملکرد و معماری متفاوتی دارند، ولی با این حال کیت هایی هستند که قبل از بارگزاری کرنل سیستم‌عامل لود می شوند و می توانند اقدام به دستکاری در سطوح مختلف سیستم کنند.
- **کیلاگر[8]:** برنامه‌های کیلاگر هر آنچه را که می‌نویسید و یا بر روی آن کلیک می‌کنید، ردیابی و ضبط می‌کنند. در نهایت اطلاعات ضبط شده را به سرور کنترل و فرماندهی بدافزار ارسال خواهند کرد. کیلاگرها خود می‌توانند به عنوان یک کلاس تهدید بدافزار زیرمجموعه جاسوس‌افزارها قرار بگیرند.

---

[1] Robots

[2] Zombie

[3] Ransomware

[4] Rootkits

[5] System Privilege

[6] Root Privilege

[7] Bootkits

[8] Keylogger

- **بدافزار پاک‌کننده[1]:** هدف آن از بین بردن داده‌ها یا سیستم است که معمولا برای ارسال یک پیام سیاسی یا مخفی کردن فعالیت‌های هکر پس از استخراج داده‌ها در نظر گرفته شده‌اند.
- **بدافزار بدون فایل[2]:** بر خلاف بقیه بدافزارها که بر روی یک فایل نوشته شده و روی دیسک هستند، بدافزار بدون فایل، صرفاً در حافظه هستند و در حالت ایده‌آل پس از اجرا هیچ اثری از خود باقی نمی‌گذارند، بنابراین پیلود مخرب بدافزار در حافظه رایانه وجود دارد و این به این معنی است که هیچ چیز مستقیماً روی دیسک سخت نوشته نمی‌شود. برای یک مهاجم، بدافزار بدون فایل دو مزیت عمده دارد:
  - هیچ فایلی برای آنتی ویروس سنتی برای شناسایی وجود ندارد.
  - روی هارد دیسک چیزی برای فارنزیک وجود ندارد.

حملات مهندسی اجتماعی[3]

هکرها با انجام حملات مهندسی اجتماعی که بر پایه سواستفاده از مسائل روانشناختی انسان‌ها است، اقدام به بهره‌برداری‌های مخرب از اهداف خود مانند اقناع قربانی به کلیک بر روی یک لینک یا دانلود یک فایل مخرب می‌کنند. حملات مهندسی اجتماعی عبارتند از:

- **فیشینگ[4]:** مهاجمان اقدام به انجام مکاتبات جعلی که به نظر می‌رسد از منابع قانونی است با کاربران می‌کنند، که این کار معمولاً از طریق ایمیل انجام می‌شود. ممکن است این ایمیل کاربر را به انجام یک عمل مهم ترغیب کند یا کاربر را ترغیب کند که بر روی لینکی که کاربر را به یک وب‌سایت مخرب هدایت کند، کلیک کند و باعث شود اطلاعات حساس را به مهاجم تحویل دهد یا خود را در معرض دانلودهای مخرب قرار دهد. ایمیل‌های فیشینگ ممکن است شامل یک پیوست آلوده به بدافزار باشند.
- **فیشینگ هدفمند[5]:** نوعی از فیشینگ است که در آن مهاجمان به طور خاص افراد دارای امتیازات امنیتی یا با نفوذ، مانند مدیران سیستم یا مدیران ارشد را هدف قرار می‌دهند. خود این حمله شامل زیرمجموعه ای از حملات دیگر از جمله Vishing Attack و Whale Attack و ... می‌شود.

---

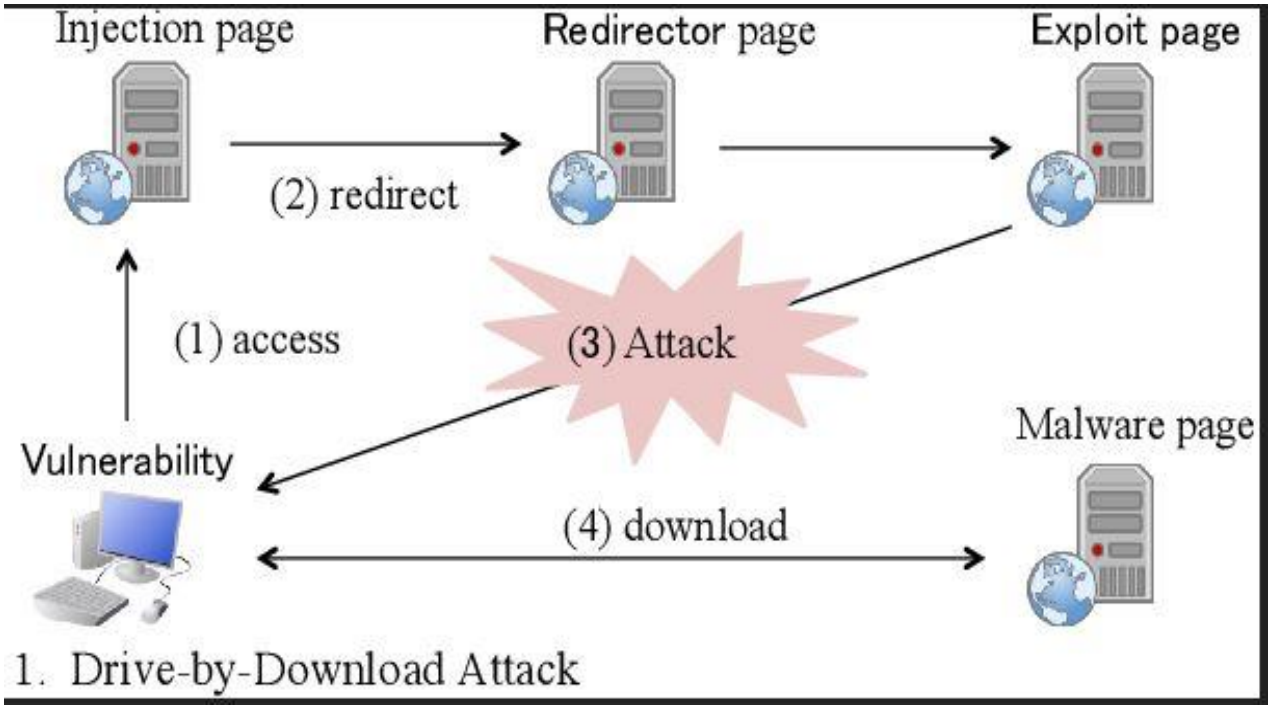
[1] Wiper Malware

[2] Fileless Malware

[3] Social Engineering

[4] Phishing

[5] Spear phishing



تصویر 5: گراف حملات مبتنی بر راهبری برای دانلود

- **راهبری دانلود[1]:** مهاجمان می‌توانند وب‌سایت‌ها را هک کنند و اسکریپت‌های مخرب را در کدهای برنامه تحت وب قرار دهند. هنگامی که کاربران از صفحه بازدید می‌کنند، با اجرای آن اسکریپت VBS یا JS مخرب، بدافزار مستقیماً روی رایانه آن‌ها نصب می‌شود یا اسکریپت مخرب مذکور، هدف را به یک سایت مخرب هدایت می‌کند که دانلود بدافزار را انجام می‌دهد. حمله راهبری برای دانلود بدافزار به آسیب‌پذیری‌های مرورگرها متکی هستند. در تصویر 5، ساختار این حملات نمایش داده شده است.
- **بهانه‌سازی[2]:** زمانی اتفاق می‌افتد که یک عامل تهدید برای دسترسی به اطلاعات سطح بالا به هدف دروغ می گوید. این حمله ممکن است شامل یک عامل تهدید باشد که وانمود می‌کند که با درخواست داده‌های مالی یا شخصی می‌خواهد هویت هدف را تأیید کند.
- **ترس‌افزار[3]:** یک عامل تهدید قربانی را فریب می‌دهد تا فکر کند به طور غیرعمدی محتوای غیرقانونی را دانلود کرده است یا رایانه او به بدافزار آلوده شده است. در مرحله بعد، عامل تهدید به قربانی راه‌حلی برای رفع مشکل جعلی ارائه می‌دهد و قربانی را فریب می‌دهد تا بدافزار را دانلود و نصب کند.

[1] Drive-by download

[2] Pretexting

[3] Scareware

حملات زنجیره تامین تجهیزات[1]

حملات زنجیره تامین تجهیزات یک حمله علیه سازمان یا تولیدکننده است. این حملات می‌توانند علاوه بر جنبه سایبری، جنبه فیزیکی (نفوذی) داشته باشد تا زنجیره تامین تجهیزات نرم‌افزاری و سخت‌افزاری مورد نفوذ قرار گیرد. زنجیره تامین تجهیزات سخت‌افزاری و نرم‌افزاری، شبکه‌ای از تمام افراد، سازمان‌ها، منابع، فعالیت‌ها و فناوری‌های دخیل در ایجاد و فروش و تهیه یک محصول است. مهاجمان با هدف قرار دادن یک مولفه در این زنجیره اقدام به کارهای مخرب از جمله قراردادن درپشتی بر روی محصولات سخت افزاری و نرم‌افزاری می کنند تا محصول بعد از استقرار در یک محیط مشخص به هکرها راهی به منظور نفوذ به آن مجموعه را ارائه بدهد.

به عنوان مثال، مهاجمان در انجام یک حمله زنجیره تامین نرم‌افزار، از اعتمادی که سازمان‌ها به فروشندگان شخص ثالث به ویژه در به‌روزرسانی‌ها و وصله‌های نرم‌افزاری دارند، سوء استفاده می‌کنند. به عنوان مثال حمله بر علیه Solarwinds از این جمله حملات بود. این امر به ویژه برای ابزارهای نظارت بر شبکه، سیستم‌های کنترل صنعتی، ماشین‌های هوشمند و سایر سیستم‌های فعال در شبکه با service accountها، صادق است. حمله می‌تواند در بسیاری از مکان‌ها علیه یکپارچه‌سازی مداوم و تحویل مداوم (CI/CD) در چرخه عمر نرم‌افزار یا حتی مقالهخانه‌ها و مولفه‌های جانبی مانند Apache و Spring انجام شود. انواع حملات زنجیره تامین نرم‌افزار شامل موارد زیر می‌شود:

- آلوده‌سازی زیرساخت‌های توسعه و آزمایش نفوذپذیری نرم‌افزار
- آلوده‌سازی دستگاه‌ها یا حساب‌های متعلق به فروشندگان شخص ثالث با دسترسی سطح بالا
- برنامه‌های مخرب امضا شده[2] با گواهی دیجیتال[3] یا شناسه توسعه دزدیده شده
- درپشتی مستقر در سخت‌افزار یا فریمور[4]
- بدافزار از پیش نصب شده روی دستگاه‌هایی مانند دوربین، USB و تلفن همراه

در بخش بالا برخی از حملات رایج در این کلاس آورده شده است که به عنوان تهدیدات جدی بر علیه امنیت سیستم‌های کامپیوتری مطرح هستند. شناسایی این حملات به شدت دشوار است.

---

[1] Supply Chain Attack

[2] Signed

[3] Certificate

[4] Firmware

حملات پایدار پیشرفته[1]

یک تهدید پایدار پیشرفته (APT) از تکنیک‌های هک مداوم، مخفیانه و پیچیده برای دسترسی به یک سیستم و ماندن در داخل آن برای مدت طولانی، با عواقب مخرب بالقوه استفاده می‌کند. به دلیل تلاش بسیار زیادی که برای انجام چنین حمله‌ای لازم است، حملات از نوع APT معمولاً در اهداف با ارزش بالا، مانند دولت و شرکت‌های بزرگ، با هدف نهایی سرقت اطلاعات یا نابودی سیستم‌های حیاتی در یک دوره زمانی طولانی انجام می‌شود.

با این حال، این بدان معنا نیست که مشاغل کوچک و متوسط می‌توانند این نوع حمله را نادیده بگیرند. مهاجمان APT معمولا از شرکت‌های کوچکتری که به زنجیره تامین هدف نهایی متصل هستند به عنوان راهی برای دسترسی به سازمان‌های بزرگ استفاده می‌کنند. چنین شرکت‌هایی معمولاً سیستم دفاعی ضعیف‌تری دارند و مهاجمان APT از این شرکت‌ها به عنوان یک پله برای هدف خود استفاده می‌کنند. در ادامه این مقاله، به طراحی یک حمله APT با محوریت استاندارد آزمایش نفوذپذیری صحبت خواهیم کرد.

حملات منع سرویس توزیع شده[2]

هنگامیکه که یک وب سرور مانند Apache یا IIS7 دارای آسیب‌پذیری است که با دریافت حجم زیادی داده دچار خرابی و توقف عملیات شود، مهاجمان می‌توانند از آن حملات استفاده کنند و یک حمله منع سرویس بر علیه سرویس Apache یا IIS انجام بدهند.

حمله منع سرویس توزیع شده (DDoS) گونه‌ای از حملات منع سرویس‌دهی هستند که در وسعت خیلی گسترده تری صورت می گیرند. در این نوع حملات مهاجمان تعداد زیادی از رایانه‌ها (اصطلاحا به عنوان زامبی شناخته می‌شوند) آلوده می‌کنند و از آن‌ها در یک حمله هماهنگ علیه اهداف خود استفاده می‌کنند. حملات منع سرویس توزیع شده (DDoS) اغلب در ترکیب با سایر تهدیدات سایبری استفاده می‌شوند. روش‌های حملات منع سرویس توزیع شده شامل موارد زیر می‌شود:

- **بات‌نت‌ها:** بات‌نت‌ها یا ربات‌ها یا زامبی‌ها ماشین‌های آلوده هستند که هکرها آن‌ها را تحت کنترل دارند. مهاجمان از این ربات‌ها برای انجام حملات منع سرویس توزیع شده استفاده می‌کنند. بات‌نت‌های بزرگ می‌توانند شامل میلیون‌ها دستگاه باشند و می‌توانند حملاتی را در مقیاس ویرانگر انجام دهند.

---

[1] Advanced persistent threats (APT)

[2] Distributed denial of service (DDoS)

- **حملات اسمورف[1]:** این حمله که از کلاس حملات منع سرویس توزیع شده است، با محوریت آسیب‌پذیری‌های موجود در پروتکل‌های IP و ICMP انجام می‌شود. به عنوان مثال، در این حمله مهاجم درخواست‌هایی مبتنی بر پروتکل ICMP را به آدرس IP قربانی ارسال می‌کنند. درخواست‌های ICMP از آدرس‌های IP جعلی تولید می‌شوند. مهاجمان این فرآیند را خودکار می‌کنند و آن را در مقیاسی انجام می‌دهند تا سیستم هدف را تحت تأثیر قرار دهند.
- **حمله گسیل درخواست‌های Syn در ارتباطات TCP[2]:** این حملات سیستم هدف را با انبوهی از درخواست‌های ایجاد ارتباط از نوع SYN مورد هدف قرار می‌دهد. هنگامی که سیستم هدف تلاش می‌کند تا اتصال را تکمیل کند، دستگاه مهاجم پاسخ نمی‌دهد و سیستم هدف را مجبور به time out می‌کند. این به سرعت connection queue را پر می‌کند و از اتصال کاربران واقعی جلوگیری می‌کند.

حملات مردی در میان[3]

در حمله مردی در میان، مهاجمان این فرض کاربر را که مستقیم به سرور مورد نظر وصل شده را شکسته و خود را بین کاربر و سرور هدف قرار می‌دهند. هنگامی که مهاجم ارتباطات را رهگیری کرد، ممکن است بتواند اعتبار یک کاربر را به خطر بیاندازد، داده‌های حساس را سرقت کند و پاسخ‌های مختلف را به کاربر بازگرداند. انواع حملات MitM به شرح زیر است:

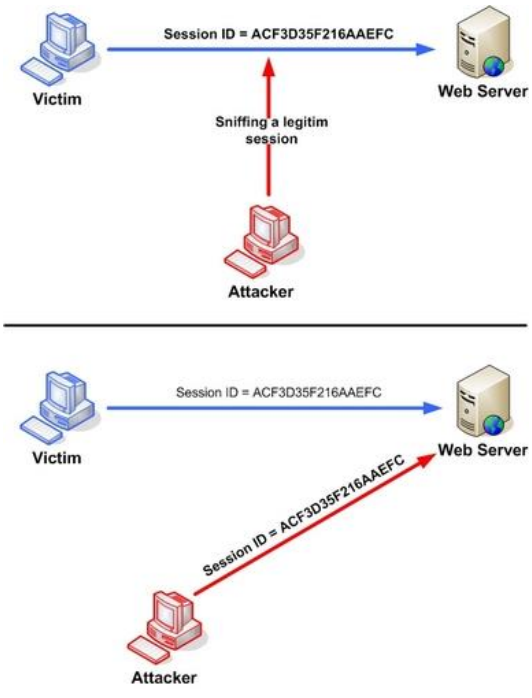
- **هایجک جلسه[4]:** در این نوع حمله، مهاجم جلسه بین کلاینت و سرور را هایجک یا به عبارت دقیق‌تر ربایش می‌کند. کامپیوتر مهاجم آدرس IP خود را جایگزین آدرس IP کاربر می‌کند. در این حالت سرور متوجه مهاجم نشده و جلسه ارتباطی را ادامه می‌دهد. در نتیجه مهاجم می‌تواند به داده‌ای که از سمت سرور به سمت کلاینت می رود دسترسی بگیرد و در نهایت آن را به سمت کلاینت مجدد مسیردهی کند. در تصویر 6، ساختار این حمله نمایش داده شده است.

[1] Smurf attack

[2] TCP SYN flood attack

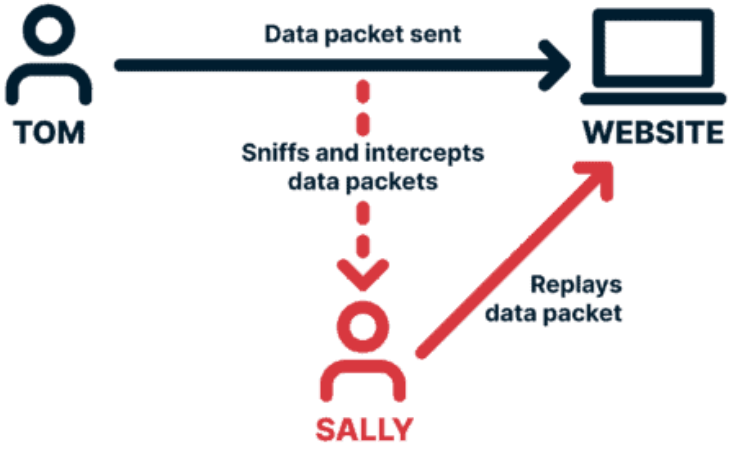
[3] Man-in-the-middle attack (MitM)

[4] Session Hijacking



تصویر 6: مثالی از حمله Session hijacking

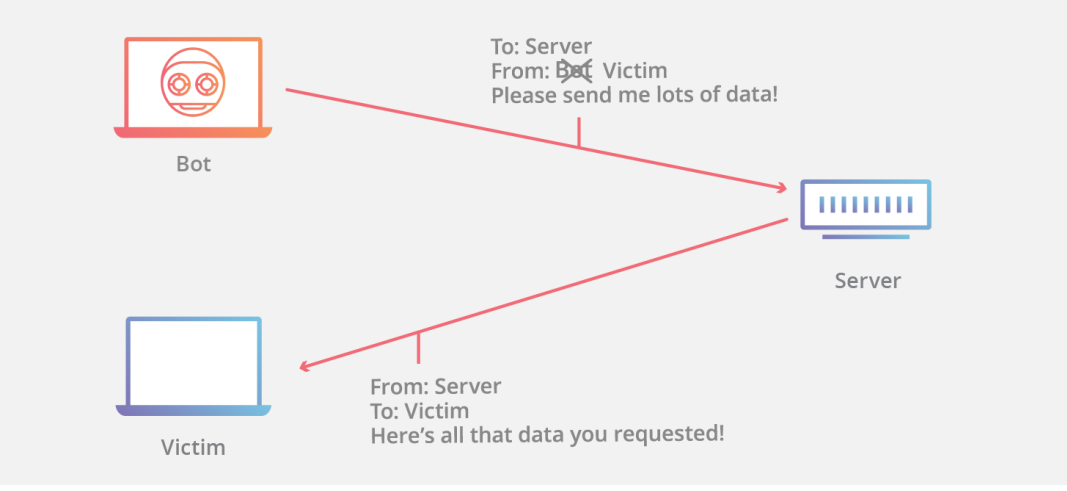
- **حمله پخش مجدد[1]:** یک مجرم سایبری ارتباطات شبکه را استراق سمع می‌کند و پیام‌ها را در زمان دیگری مجددا ارسال می‌کند و وانمود می‌کند که کاربر است. حملات پخش مجدد تا حد زیادی با افزودن timestamp به ارتباطات شبکه کاهش یافته است.



تصویر 7: مثالی از حمله پخش مجدد

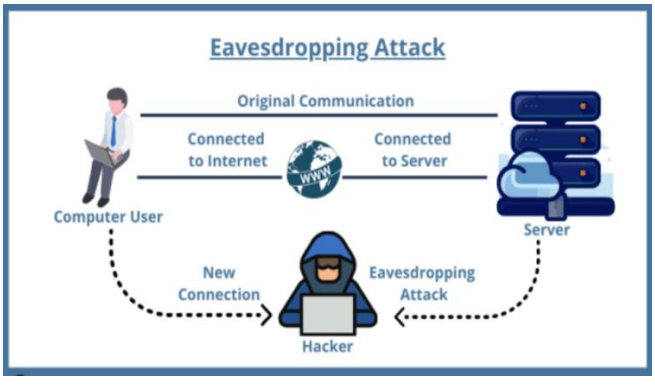
[1] Replay attack

- **جعل پروتکل اینترنتی[1]:** یک مهاجم سیستم را متقاعد می‌کند که با یک نهاد معتبر و شناخته شده مطابقت دارد. بنابراین سیستم دسترسی مهاجم را فراهم می‌کند. مهاجم IP خود را با IP یک میزبان قابل اعتماد عوض می‌کند. در تصویر 8، ساختار این حمله نمایش داده شده است.



تصویر 8: مثالی از حمله IP spoofing

- **حملات شنود ارتباطات[2]:** مهاجمان از ارتباطات غیرایمن تحت شبکه برای دسترسی به اطلاعات منتقل شده بین مشتری و سرور استفاده می‌کنند. از آنجایی که در چنین ارتباطاتی، ترافیک شبکه رمزنگاری نمی شود، مهاجم می تواند به سادگی این ارتباطات را شنود کند و به اطلاعاتی که در حال گذر از شبکه است، دسترسی بگیرد. در تصویر 9، این مسئله نمایش داده شده است.



تصویر 9: مثالی از Eavesdropping attack

[1] IP spoofing

[2] Eavesdropping attack

- **حملات پروتکل بلوتوث[1]:** از آنجایی که بلوتوث اغلب در حالت بی‌وقفه باز و روشن است، حملات بسیاری، به‌ویژه علیه تلفن‌ها، وجود دارد که طریق اتصالات بلوتوث باز انجام می‌شود. معمولاً هدف از انجام این حمله جمع‌آوری اطلاعات، شنود ارتباطات، اخذ دسترسی و در نهایت استقرار یک بدافزار بر روی دستگاه آسیب‌پذیر است. به عنوان مثال، حمله BlueBorne یکی از مشهورترین حملات بر علیه پروتکل Bluetooth است.
- **حملات بر علیه گذرواژه‌ها[2]:** یک هکر می‌تواند با شنود یک ارتباط تحت شبکه، جستجوی فراگیر گذرواژه‌ها[3]، استفاده از مهندسی اجتماعی، حدس زدن یا دسترسی به پایگاه داده، به اطلاعات گذرواژه یک حساب کاربری دسترسی پیدا کند. مهاجم می‌تواند رمز عبور را به صورت تصادفی یا سیستماتیک در انجام حملات جستجوی فراگیر گذرواژه نیز حدس بزند. انواع حملات رمز عبور عبارتند از:
  - **جستجوی فراگیر گذرواژه:** یک مهاجم از نرم‌افزار یا یک اسکریپت که خودش نوشته است، برای امتحان گذرواژه‌های مختلف استفاده می‌کند، به امید اینکه گذراواژه صحیح را حدس بزند. این نرم‌افزار می‌تواند از منطقی برای امتحان گذرواژه‌های مربوط به نام فرد، شغل، خانواده و غیره استفاده کند.
  - **حمله مبتنی بر دیکشنری[4]:** در یک دیکشنری گذرواژه‌های رایج برای دسترسی به رایانه و شبکه نوشته شده است. با استفاده از این لیست مهاجم گذرواژه‌های احتمالی را آزمایش می‌کند.
  - **عبور با هش[5]:** یک مهاجم بجای آزمایش گذرواژه‌ها از الگوی هش آن گذرواژه‌ها استفاده می‌کند و سعی در دسترسی به فایل، سرویس و یا سیستم قربانی دارد. در این روش نیازی به رمزگشایی هش‌های از قبل پیدا شده نیست و با همان هش‌ها می‌توان سعی در دسترسی به بقیه اهداف کرد.
  - حمله تیکت طلایی[6]: حمله تیکت طلایی به همان روشی شروع می‌شود که حمله عبور با هش شروع شروع خواهد شد. در این حمله، با توجه به پروتکل Kerberos در ساختار اکتیو دیرکتوری ویندوز مهاجم از هش برای دسترسی به مرکز توزیع کلید[7] به منظور جعل هش ticket-granting-ticket(TGT) استفاده می‌کند.

---

[1] Bluetooth attacks

[2] Password attacks

[3] Brute force Attacks

[4] Dictionary Attacks

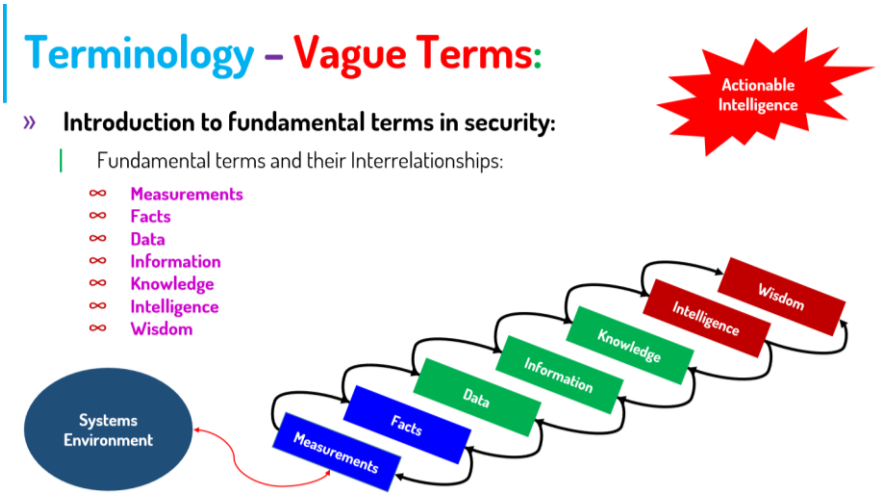
[5] Pass the hash Attacks

[6] Golden Ticket Attacks

[7] Key Distribution Center

اصطلاحات اساسی در امنیت سیستمی

در ادامه این مقاله، به منظور فهم استانداردهای مطرح آزمایش نفوذپذیری شبکه‌ها و سیستم‌های کامپیوتری از منظر سیستمی مانند استاندارد PTES، اقدام به بررسی و مطالعه اصطلاحات و واژگان تخصصی خواهیم کرد تا با ادبیات این حوزه آشنا شویم.



تصویر 11: هرم واژگان تخصصی در امنیت و اطلاعات

در تصویر 11 که بخشی از اسلایدهای دوره تجزیه و تحلیل جامع بدافزار میلاد کهساری است، هرم تخصصی واژگان در حوزه امنیت سیستمی نمایش داده شده است. این اصطلاحات (واژگان تخصصی) در ادامه تشریح شده است:

- **اندازه‌گیری[1]:** اندازه‌گیری، کمی کردن ویژگی‌های یک شی یا رویداد در یک محیط است که می‌توان از آن برای مقایسه آن با اشیاء یا رویدادهای دیگر استفاده کرد. شایان ذکر است، اندازه گیری پدیده ها در محیطهای مختلف تفاوت های اساسی با یکدیگر دارند. به عنوان مثال، واحد اندازه‌گیری جاذبه از واحد اندازه‌گیری قدرت تخریب بمب هیدروژنی متفاوت است. زیرا محیط و زمینه آن دو با هم متفاوت است.
- **واقعیت[2]:** واقعیت اطلاعات یا برداشت‌هایی هستند که ثابت شده اند. به عنوان مثال، جاذبه یک واقعیت هست، زیرا هر جای کره زمین قدم بگذارید آن وجود دارد و نمی توان آن را رد کرد یا نادیده گرفت. یا فرمول $E = mc^2$ یک واقعیت است، زیرا هم‌ارزی جرم و انرژی را اثبات می‌کند.

[1] Measurement

[2] Facts

- **داده خام[1]:** داده‌های خام در تمامی سطوح وجود دارند و هیچ اهمیتی فراتر از وجودش (به خودی خود) ندارد. می‌تواند به هر شکلی وجود داشته باشد، قابل استفاده یا غیر قابل استفاده باشد. خودش معنایی ندارد. بر روی داده خام، هیچ پردازش و دسته‌بندی اتفاق نیافتاده است. به عنوان مثال، توالی 14010630 یک فرم از داده های خام عددی است و به خود خود اهمیت و معنا ندارد.
- **اطلاعات[2]:** دادههای خام وقتی پردازش می شوند، و همچنین در یک زمینه مناسب قرار می گیرند، تبدیل به اطلاعات می شوند. به عنوان مثال، وقتی داده خامی مانند 14010630 را در قالب یک تاریخ بررسی کنیم، معنا و اعتبار پیدا خواهد کرد زیرا به یک روز اشاره دارد. این معنا می‌تواند مفید باشد، اما لزوما همه جا قابل استفاده نیست. داده‌های خام که برای مفید بودن پردازش می‌شوند، در نهایت به سؤالاتی از جمله "چه کسی"، "چه"، "کجا" و "چه زمانی" در یک زمینه مشخص پاسخ می‌دهند. اطلاعات پیامی است که حاوی معنا، دلالت یا ورودی مربوط به تصمیم و/یا اقدام است. اطلاعات هم از منابع فعلی (ارتباطات) و هم از منابع تاریخی (داده‌های خام پردازش شده یا تصاویر بازسازی شده) به دست می‌آید. در اصل، هدف از اطلاعات کمک به تصمیم گیری و/یا حل مشکلات یا تحقق یک فرصت است. اگر خیلی خلاصه بخواهیم اطلاعات را تعریف کنیم می‌توانیم بگوییم اطلاعات، داده‌های خامی هستند که پردازش شده‌اند. مثلا مشخص شده که آن عدد اشاره به تاریخی دارد که کلاس تحلیل بدافزار میلاد کهساری برگزار شده است.
- **دانش[3]:** مجموعه مناسبی از اطلاعات است، به گونه‌ای که قصد آن مفید بودن است. اطلاعات زمانی تبدیل به دانش خواهد شد که بتواند به شناخت یا دانستن چیستی، ظرفیت عمل (دانش چگونگی) و درک (دانستن چرایی) یک پدیده به ما کمک کند. آن زمان می توانیم مدعی شویم که نسبت به چیزی دیگر فقط اطلاعات نداریم، بلکه به دانش در مورد آن رسیده‌ایم.
- **اینتلیجنس[4]:** اینتلیجنس مستلزم توانایی درک محیط، تصمیم گیری و کنترل عمل است. سطوح بالاتر اینتلیجنس ممکن است شامل توانایی تشخیص اشیا و رویدادها، ارائه دانش در یک مدل جهانی و استدلال در مورد برنامه آینده باشد. در اشکال پیشرفته، اینتلیجنس ظرفیت شناسایی و درک، انتخاب عاقلانه و عملکرد موفقیت آمیز تحت شرایط مختلف برای بقا، شکوفایی و تولید مثل در یک محیط پیچیده و اغلب خصمانه را فراهم می‌کند.

---

[1] Data

[2] Information

[3] Knowledge

[4] Intelligence

- **خرد[1]:** خرد یعنی توانایی پیش‌بینی داریم. مثلا می‌توانیم پیش‌بینی کنیم که یک فرد در یک موقعیت خاص چه عکس‌العملی از خودش نشان می‌دهد. هنگامی که نسبت به یک پدیده به اینتلیجنس برسیم، و آن اینتلیجنس در روندهای متعدد صادق باشد، دیگر به خرد درباره آن پدیده و یا سوژه رسیده‌ایم.

مبانی مطلق در آزمایش نفوذپذیری

در آزمایش نفوذپذیری[2] ما به عنوان یک متخصص امنیت باید تمام روش‌ها و یا حفره‌های امنیتی را که یک هکر می‌تواند از آنها برای دور زدن[3] کنترل‌های امنیتی سوء استفاده و به سیستم‌های کامپیوتری نفوذ کند، شبیه‌سازی کنیم و در صورت وجود ضعف‌های امنیتی جهت رفع آن‌ها اقدامات مناسب را انجام دهیم تا از نفوذ هکرها و افراد بیگانه به سیستم جلوگیری شود. قابل ذکر است، هدف یک هکر می‌تواند هر چیزی باشد، از سیستم‌های بزرگ گرفته تا سیستم‌های کوچک از جمله سیستم‌های دولتی، بانک‌ها، کامپیوترهای شخصی، وبسایت‌ها، سرویس‌های شبکه و غیره... در اینجاست که باید به عنوان یک متخصص امنیت (هکر کلاه سفید) وارد میدان شوید و تمام راه‌های ممکن برای رخنه به سیستم توسط یک نفوذگر (هکر کلاه سیاه) را کشف کنید و سپس مهار سازید.

شایان ذکر است، باید به این نکته توجه شود، عملیات آزمایش نفوذ عملا بیش از یک عمل ساده پویشگری با استفاده از ابزارهای خودکار و سپس تهیه گزارش است. این موضوع را گام به گام درک خواهید کرد. خوشبختانه درحال حاضر دید مدیران به موضوع امنیت عوض شده است و دیگر با دید سطحی به آن نگاه نمی‌شود. به همین دلیل شاهد هستیم برای آزمایش‌های نفوذ استاندارد دستورات کاری معینی وضع می‌شود که Penetration Testing Execution Standard یا PTES نمونه‌ای از آن است که منشوری از اطلاعات افراد مختلف درباره یک آزمایش نفوذ واقعی است. به هر حال اگر شما در حوزه امنیت تازه کار می‌باشید یا با فازهای PTES ناآشنا هستید و اگر میل به اطلاعات بیشتر درباره استاندارد آزمایش نفوذ دارید می‌توانید به وبسایت Pentest-Standard.Org رجوع نمایید و مطالب موجود در آن را مورد مطالعه قرار بدهید. با این حال توضیح مختصری درباره تمامی فازهای PTES در این مقاله آورده شده است.

---

[1] Wisdom

[2] Penetration Testing

[3] Bypass

فازهای آزمایش نفوذپذیری استاندارد

فازها مختلف PTES بدین دلیل تعریف شده‌اند که بتوان بواسطه آن‌ها تمام مسائل امنیتی را ارزیابی کرد. این فازها دارای مزیت‌های بسیاری هستند، به‌طوریکه اگر سیستم با مشکل امنیتی مواجه شد، بتوان به‌راحتی آنها را کشف و مهار کرد. اما باید همیشه به خاطر داشت که "در دنیای رایانه‌ها، امنیت هیچگاه صد در صد نیست" لذا هیچگاه نمی‌توانید به مشتری امنیت صد در صدی را تضمین دهید. اما همواره سعی کنید تمام ضعف‌های امنیتی شناخته شده را برطرف کنید تا سطح امنیت به حد قابل قبولی برسد. به هر حال فازهای یاد شده بسته به نوع سازمان یا سیستم مخاطب که مورد آزمایش نفوذ قرار می‌گیرد، به هفت قسمت با سطوح و زیرلایه‌های مختلف تقسیم می‌شود که در ادامه آن‌ها را مختصراً مورد بررسی قرار خواهیم داد.

فاز اول: توافق‌های قبل از قرارداد[1]

منظور از توافق‌های قبل از قرارداد مسائل و مواردی است که قبل از بستن قرارداد باید برای کارفرما توضیح دهید و روی آن‌ها توافق کنید و نظریات احتمالی کارفرما را نیز جویا شوید و آنها را در بعضی از مسائل وارد برنامه آزمایش نفوذپذیری کنید. هنگامی که می‌خواهید سیستمی را مورد آزمایش قرار بدهید، باید با مشتری قرارداد بسته و حوزه کاری خود را شرح دهید. سپس شرایطی را که کارفرما برای شما شرح می‌دهد وارد قرارداد نمایید. تعهدهای مطرح شده در قرارداد بسیار مهم هستند، زیرا این مرحله نیز به عنوان فرصتی است تا به مشتری بفهمانید در یک آزمایش نفوذ چه عملیاتی انجام می‌شود و چه انتظارهای از شما می‌تواند وجود داشته باشد. بیشتر تیم‌های حرفه‌ای که در زمینه آزمایش نفوذپذیری فعالیت می‌کنند برای بستن قرارداد و انجام مذاکرات دارای یک شخص خاص هستند که به مسائل حقوقی نیز آشنا است و انجام مذاکرات بر عهده او می‌باشد.

فاز دوم: جمع‌آوری اینتلیجنس[2]

در قسمت جمع‌آوری اینتلیجنس، تمام اطلاعات مورد نیاز برای حمله با استفاده از شبکه‌های اجتماعی، گوگل هکینگ، پویشگری و غیره به دست می‌آید. یکی از مهم‌ترین مهارت‌های یک متخصص امنیت یا هکر، توانایی بدست آوردن اطلاعات در مورد هدف است، از قبیل اینکه قربانی چگونه رفتار می‌کند، چه اطلاعاتی در اختیار دارد و نهایتا این‌که چگونه باید به آن حمله کرد. این اطلاعات دید مناسبی راجع به انواع کنترل‌های امنیتی مورد استفاده در سیستم هدف

---

[1] Pre-engagement

[2] Intelligence Gathering

به متخصص ارائه می‌دهد. مثلا اینکه سیستم‌عامل هدف چیست؟ آیا ماشین هدف قابل بهره‌برداری[1] است؟ آیا سیستم هدف از دیوار آتش[2] استفاده می‌کند؟ معماری شبکه[3] آن به چه شکل است و غیره.

فاز سوم: مدل‌سازی تهدیدات یا هدف[4]

در مدل‌سازی تهدیدات هنگام مطالعه یک تهدید، یا در مدل‌سازی هدف هنگام طراحی یک حمله APT، بر پایه اطلاعات بدست آمده از فاز جمع‌آوری اطلاعات، به صورت قدم به قدم به تجزیه و تحلیل و شناسایی آسیب‌پذیری‌های موجود پرداخته می‌شود. هدف می‌تواند شبکه، سرویس، برنامه‌های کاربردی و غیره باشد. نوع اطلاعاتی را که برای مورد حمله قرار دادن هدف مورد نیاز است، در این فاز دریافت می‌کنیم. به عنوان مثال یک کامپیوتر شخصی را به عنوان دشمن شبیه‌سازی کنید. در فاز جمع‌آوری اطلاعات باید تمام جنبه‌های سیستم هدف مورد بررسی قرار گیرد.

از قبیل اینکه سیستم‌عامل هدف چیست؟ چه برنامه‌هایی روی آن نصب شده است؟ آیا سیستم دارای دیوار آتش و ضدویروس است و امثالهم. پس از جمع‌آوری این اطلاعات مهم وارد قسمت سوم یعنی مدل‌سازی تهدیدات یا هدف می‌شوید. در این فاز باید بررسی کنید که آیا سیستم‌عامل هدف اصلا قابل نفوذ است یا خیر؟ اگر در این فاز سیستم‌عامل قربانی را قابل نفوذ تشخیص دادید، باید بهترین الگو برای حمله به قربانی را مشخص کنید. این فاز یکی از فازهای طاقت فرسای PTES است.

فاز چهارم: تحلیل آسیب‌پذیری‌ها[5]

بعد از یافتن بهترین الگو برای حمله، باید چگونگی دسترسی به هدف را مد نظر داشت. در طی فاز تحلیل آسیب‌پذیری باید اطلاعات خود را با اطلاعاتی که در مراحل قبل بدست آوردید ترکیب کرده تا بهترین الگوی حمله را در اختیار داشته باشید. فاز تحلیل آسیب‌پذیری، شامل پویش درگاه‌ها و آسیب‌پذیری‌ها، جمع‌آوری داده‌ها توسط Banner Grabbing و اطلاعات جمع‌آوری شده در طی فاز اول می‌باشد.

---

[1] Exploiting

[2] Firewall

[3] Network Architecture

[4] Threat or Target Modeling

[5] Vulnerability Analysis

فاز پنجم: اکسپلویت آسیب‌پذیری[1]

اکسپلویت آسیب‌پذیری در یک سیستم شاید یکی از هیجان انگیزترین بخش‌های آزمایش نفوذ باشد. به طور کلی برنامه‌های موجود روی سیستم‌های کامپیوتری مجموعه‌ای از قوانین را دنبال می‌کنند تا یک عمل خاص (مد نظر برنامه‌نویس) را انجام دهند. اکسپلویت یک برنامه در واقع راهی است جهت انجام کارهای مخرب (مد نظر هکر) روی سیستم قربانی؛ حتی اگر سیستم از اجرای آن فرمان‌ها منع شده باشد. نفوذگر می‌تواند از حفره‌های امنیتی (در صورت وجود) در برنامه بدین منظور سوء استفاده کند و دستورات و فرامین مطلوب خود را روی سیستم قربانی اجرا نماید. یافتن این حفره‌ها نیازمند یک ذهن خلاق است و شما باید از وجود یا عدم وجود این حفره‌ها در سیستم قربانی اطمینان حاصل کنید تا در موفقیت آزمایش نفوذ به مشکل بر نخورید.

فاز ششم: پس از اکسپلویت[2]

این فاز بعد از نفوذ به یک یا چند ماشین و گرفتن دسترسی از آنها آغاز می‌شود و تمامی فرآیندهایی که هکر یا نفوذگر بعد از نفوذ به ماشین قربانی انجام می‌دهد، از قبیل دانلود فایل، بارگزاری بدافزار در ماشین هدف و... جزوء فعالیت‌هایی است که در این فاز صورت می‌گیرند. در این فاز علاوه بر نکات ذکر شده، ارزش ماشین‌های مورد نفوذ قرار گرفته شده در راستای اینکه هکر دسترسی خودش را روی آنها حفظ کند یا خیر تعیین می‌شود. حالا سوال پیش می‌آید که کامپیوترها چه ارزشی می‌توانند داشته باشند؟ در جواب این سوال می‌توان گفت، ارزش هر کامپیوتر بر مبنای حساسیت اطلاعات ذخیره شده در آن است. این فاز یکی دیگر از فازهای مهم برای نفوذگر است، زیرا روش‌های مورد استفاده در این بخش به نفوذگر برای شناسایی و مستندسازی حساسیت داده‌های ماشین قربانی کمک می‌کند.

فاز هفتم: گزارش‌نویسی[3]

این فاز به مراتب مهم‌تر از بقیه فازها است، زیرا باید به کارفرما گزارش کامل کار خود را از آزمایش نفوذپذیری تحویل می‌دهید. در این گزارش شرحی از کارهای انجام شده (عملیات نفوذ)، چگونگی انجام آنها و از همه مهم‌تر چگونگی مقابله با نفوذهای انجام شده و برطرف نمودن آسیب‌پذیری‌های کشف شده ارائه می‌شود.

---

[1] Vulnerability Exploitation

[2] Post-Exploitation

[3] Reporting

انواع آزمایش‌های نفوذپذیری[1]

تا این قسمت از مقاله تا حدودی با فازهای استاندارد آزمایش نفوذپذیری آشنا شدید. اکنون دو نوع از مهم‌ترین الگوهای آزمایش نفوذپذیری را مورد بررسی قرار می‌دهیم، یعنی آزمایش نفوذپذیری پنهان که هکرهای کلاه سیاه آن را مورد استفاده قرار می‌دهند و آزمایش نفوذپذیری آشکار که متخصصین امنیت یا هکرهای کلاه سفید از آن استفاده می‌کنند. قابل ذکر است، تا به حال تعریف دقیق و جامعی برای این مسائل بدست نیامده است و همچنین هر دو روش مذکور دارای مزایا و معایبی هستند که در ادامه به آنها خواهیم پرداخت.

Knowledge Type	Network Knowledge	Benefits
Black-Box (Covert)	**No knowledge** of your infrastructure	• Focused on the impact an adversary can cause
White-Box (Overt)	Almost **complete knowledge** of your infrastructure (IPs, diagrams, etc.)	• Faster than Black Box • Cheaper than Black Box • A more comprehensive evaluation of the network
Grey-Box	**Some knowledge** of your infrastructure	• Blend of speed vs adversarial simulation

تصویر 12: انواع رویکرد آزمایش نفوذپذیری

آزمایش نفوذپذیری آشکار[2]

آزمایش نفوذپذیری آشکار با اطلاع و آگاهی کامل سازمان مورد نفوذ انجام می‌پذیرد و هدف آن برطرف ساختن ضعف‌های امنیتی سازمان در قالب یک قرارداد است. در این الگوی آزمایش نفوذپذیری، متخصص امنیت وظیفه دارد که سیستم سازمان مشتری را مورد آزمایش نفوذپذیری قرار داده و آسیب‌پذیری‌های امنیتی موجود در آن را

[1] Types of Penetration Tests

[2] Overt Penetration Testing

شناسایی و رفع نمایید. در نهایت یک گزارش از کار خود به کارفرما ارائه بدهد. مزیت این نوع آزمایش نفوذپذیری در این است که دقدقه‌ای راجع به مسدود شدن خود یا پیگیری‌های احتمالی قضایی و مسائلی از این دست را ندارید و می‌توانید با فکر آزاد و دسترسی مستقیم، روی سیستم‌ها آزمایش نفوذپذیری انجام دهید. اما نکته این است که همواره متخصصین امنیت استدلال می‌کنند که بر پایه این نوع آزمایش نفوذپذیری، امنیت برنامه ها به خوبی حصول نمی‌شود، به همین دلیل این روش را موثر نمی دانند. در حالت کلی وقتی زمان محدود است و از طرفی مراحل PTES (مثلا فاز جمع‌آوری اطلاعات) در ظرف زمانی مناسب نمی‌تواند انجام پذیرد، این نوع آزمایش بهترین گزینه برای سنجش امنیت است. مسلم است که نمی‌توان امنیت صد در صد برنامه‌ها را تضمین نمود. ولی به هر صورت مزیت این نوع آزمایش صرفه جویی در وقت و طی کردن مراحل استاندارد آزمایش امنیت بدون دردسر و محدودیت می‌باشد.

آزمایش نفوذپذیری پنهان[1]

بر خلاف آزمایش نفوذپذیری آشکار، این نوع آزمایش بر اساس طراحی و شبیه‌سازی حملات واقعی هکری انجام می‌شوند. در این حملات یک نفوذگر بدون آگاهی خاصی از قربانی حمله را صورت می‌دهد و کنترل‌های امنیتی را دور زده و قربانی را مورد نفوذ قرار می‌دهد. از مزیت این نوع آزمایش نفوذپذیری می‌توان به سنجش تیم های واکنش سریع[2] و میزان مقاومت و ایمنی سیستم در برابر حملات سرزده اشاره نمود. آزمایش‌های پنهان می‌تواند از بُعد مالی و زمانی پرهزینه باشند و همچنین بدیهی است که هکر نیاز به مهارت های بیشتری برای انجام این حملات دارد. متخصصین نفوذگری معمولا این روش را انتخاب می‌کنند، چونکه به یک حمله واقعی بیشتر شبیه است. آزمایش‌های نفوذ پنهان بر توانایی فرد در بدست آوردن اطلاعات از قربانی متکی هستند. از طرفی معمولا در تلاشی که برای این انجام می دهید، احتمالا تعدادی آسیب‌پذیری و متقابلا راه هایی را برای اکسپلویت کردن آسیب‌پذیری‌ها و نهایتا دسترسی به سیستم قربانی پیدا خواهید کرد. این راه ها شناخته شده نیستند، اما شما می‌توانید آنها را کشف نموده و سپس از آنها سو استفاده یا در راه برقراری امنیت استفاده کنید.

اصطلاحات رایج در هکینگ[3]

در این مقاله از واژگان مختلفی استفاده می‌شود که در صورت مواجه با آنها برای اولین بار درکشان شاید پیچیده باشد. لازم به توضیح است که اکثر این واژگان معمولا در علوم امنیت سیستم‌های کامپیوتری استفاده می‌شوند.

---

[1] Covert Penetration Testing

[2] Incident Response

[3] Terminology

اکسپلویت (کد بهره‌برداری خودکار)

به کدی که یک هکر می‌نویسد و هدف اصلی آن بهره‌برداری از آسیب‌پذیری‌ها در سیستم‌ها، برنامه‌های کاربردی، سرویس‌ها و غیره است، اکسپلویت گویند. از اکسپلویت‌هایی که معمول برای بهره‌برداری از آسیب‌پذیری‌های امنیتی توسعه می‌یابند، می‌توان به سرریز بافر[1] در برنامه‌های کاربردی، SQL Injection بر روی برنامه‌های مبتنی بر وب، خطاهای پیکربندی در سرویس‌ها اشاره نمود. همچنین به دلیل اینکه واژه فارسی این جمله یعنی Exploiting زیاد در دنیای امنیت رایج نیست و درک آن مشکل است، ما در این مقاله از خود واژه لاتین یعنی اکسپلویت استفاده خواهیم کرد.

پیلود[2] (محموله اجرایی اکسپلویت)

کد اصلی را که در حین اکسپلویت شدن سیستم، عمل خاصی را روی سیستم قربانی انجام می‌دهد، پیلود می‌گویند. پس از اجرای موفقیت‌آمیز این کد کنترل سیستم هدف (بسته به نوع آن) در اختیار نفوذگر قرار می‌گیرد. معمولا پیلود در درون کد اکسپلویت جاسازی می‌شود. در فریمورک متاسپلویت در حالت کلی دو نوع پیلود وجود دارد که با نام‌های Reverse Shell و Bind Shell شناسایی می‌شوند. تفاوت اساسی که بین این دو پیلود وجود دارد، در نوع برقراری ارتباط با سیستم قربانی است. پیلود Reverse Shell یا پوسته معکوس ارتباط با سیستم قربانی را با یک اتصال بازگشتی (Connect-Back) و ارائه خط فرمان وظیفه خود را انجام می‌دهد، اما در آن طرف ماجرا، پیلود Bind Shell یک خط فرمان یا cmd را به یکی از درگاه‌های ماشین اکسپلویت شده پیوست می‌کند و در نهایت شما می‌توانید به آن درگاه متصل شوید و از طریق آن بر روی ماشین قربانی فرآمین خود را اجرا کنید. قابل ذکر است، معروف‌ترین پیلودی که در متاسپلویت وجود دارد Meterpreter یا مفسر متاسپلویت نام دارد. این پیلود با در اختیار قرار دادن خط فرمان سیستم‌عامل قربانی (پوسته فرمان) به هکر عملیات خود را انجام می‌دهد.

شلکد[3]

شلکد مجموعه‌ای از دستورات به زبان اسمبلی است که در کد اکسپلویت آسیب‌پذیری‌ها با هدف ارائه دسترسی از پوسته سیستم‌عامل قربانی به مهاجم مورد استفاده قرار می‌گیرند. قابل ذکر است، پیلودهای اجرایی که فقط و فقط

---

[1] Buffer Over-flow

[2] Payload

[3] Shellcode

به منظور در اختیار قرار دادن پوسته فرمان (Shell) سیستم قربانی به هکر طراحی شده باشند، همچنین شلکد نامیده می‌شوند.

ماژول[1]

در این مقاله ماژول به عنوان یک برنامه یا قطعه کد فرض می‌شود که می‌توان از آن در متاسپلویت استفاده کرد. گاهی اوقات لازم است که از یک ماژول برای حمله به اجزای یک برنامه استفاده کنید. یا شاید لازم باشد از ماژول‌ها برای اعمالی مثل پویش استفاده کنید. به طور کلی ماژول‌های تعاملی (Interactive) نقطه قدرت متاسپلویت هستند، زیرا انجام بسیاری از کارها را تسهیل می‌بخشند.

شنونده[2]

در متاسپلویت گاهی اوقات لازم است که سیستم قربانی پس از اکسپلویت شدن به سیستم مهاجم متصل شود و اطلاعاتی را ارسال و یا در موارد خاص دریافت نماید. در این موارد یک شنونده روی سیستم مهاجم اجرا می‌شود تا سیستم قربانی بتواند به آن متصل گردد. از کاربردهای این ویژگی می‌توان به اتصال به سیستم قربانی به صورت Connect-Back اشاره کرد که در آن سیستم قربانی به سیستم مهاجم متصل شده و مثلا یک پوسته فرمان را در اختیار قرار می‌دهد.

فازر[3]

فازرها برنامه‌هایی هستند که به صورت پویا و ایستا برنامه‌های کاربردی را تجزیه و تحلیل کرده و آسیب‌پذیری‌های موجود در برنامه را گزارش می‌دهند. اساس کار فازرها، دادن مقادیر زیاد، کم و خاص به ورودی‌ها و توابع برنامه کاربردی به صورت تصادفی است. در نتیجه ارسال این مقادیر، اگر برنامه مقصد کرش کند یا خراب شود، گواه این است که برنامه دارای آسیب‌پذیری است.

---

[1] Module

[2] Listener

[3] Fuzzer

پویشگرهای آسیب‌پذیری (حفره‌های امنیتی)[1]

بسیاری از سایت‌ها و پورتال‌هایی که امروزه دیفیس[2] می‌شوند به دلیل ضعف‌های امنیتی است، مثلا ممکن است حمله از طریق SQL Injection صورت پذیرد (که ناشی از برنامه نویسی ضعیف است و به هکر اجازه اجرای دستورات SQL را برای انجام مقاصد خود می‌دهد) و یا وجود یک ضعف روی وب سرور مسبب نفوذ هکرها شود.

یک پویشگر آسیب‌پذیری، ابزار خودکاری است که برای شناسایی آسیب‌پذیری‌های (حفره‌های امنیتی) موجود در سیستم‌های مبتنی بر وب یا برنامه‌های کاربردی استفاده می‌شود. این پویشگرها راه های مختلفی را به طور خودکار برای تشخیص آسیب‌پذیری‌ها و جمع‌آوری اطلاعات از هدف استفاده می‌کنند، بدون اینکه شما در این رویه دخالتی داشته باشید. کار اصلی این پویشگرها ارزیابی امنیت و در اختیار گذاشتن اطلاعات امنیتی سیستم مقابل برای ما در قالب یک گزارش است.

این اطلاعات شامل مواردی از قبیل نوع سیستم‌عامل، نسخه سیستم‌عامل، آسیب‌پذیری‌های برنامه‌های مبتنی بر وب، درگاه‌های باز روی سیستم قربانی و نیز تمام سرویس های در حال اجرا روی آن می‌شود. وقتی سیستم هدف را پویش می‌کنید، پویشگر از روش‌های مختلفی برای تشخیص وجود آسیب‌پذیری‌ها استفاده می کند. در پویشگرهای مدرن محاسبات و عملیات مهمی برای به حداقل رساندن False-Positive ها انجام می‌شود. از همین روی بسیاری از سازمآن‌ها این برنامه‌های آماده را برای ارزیابی امنیت سیستم‌های خود انتخاب یا خریداری می‌کنند و نهایتا در صورت وجود آسیب‌پذیری ضعف‌های امنیتی را رفع می نمایند.

نکته قابل ذکر این است که تا جای امکان بهتر است از چنین ابزارهای استفاده نشود، چون اگر کارها را خودتان به صورت دستی انجام دهید، تکنیک هایی را که پویشگرهای آسیب‌پذیری استفاده می‌کنند می آموزید. این تکنیک ها برای شما می‌توانند بسیار ارزشمند باشند. درهرصورت اگر تمایل دارید که از چنین ابزارهایی استفاده کنید باید بسیار مراقب باشید، چون علیرغم استفاده از الگوریتم های پیشرفته و تکنیک های روز بهرحال درصدی از خطا در این ابزارها وجود دارد.

شایان ذکر است، در اصل زیبایی و هنر در آزمایش نفوذ انجام تمام کارها به صورت دستی است و اینکه حمله و گرفتن دسترسی از سیستم‌ها مرتبط به علم و دانش شما باشد نه استفاده از ابزارهای از پیش آماده. وقتی فرد به یک متخصص امنیت تبدیل شود به ندرت از چنین ابزارهای استفاده می نماید و بیشتر متکی بر دانش خود خواهد بود.

---

[1] Vulnerability Scanners

[2] Deface

به هر حال اگر شما در ابتدای راه قرار دارید یا می‌خواهید واقعا یک روش رسمی و همگانی را برای آزمایش نفوذ بکار گیرید، حتما فازهای PTES را مطالعه نمایید. بدین ترتیب می‌توانید مطمئن باشید که در هر گام از آزمایش نفوذ یک فرآیند کامل و غیرتکراری را پیش می گیرید.

پیاده‌سازی یک حمله مبتنی بر فازهای PTES

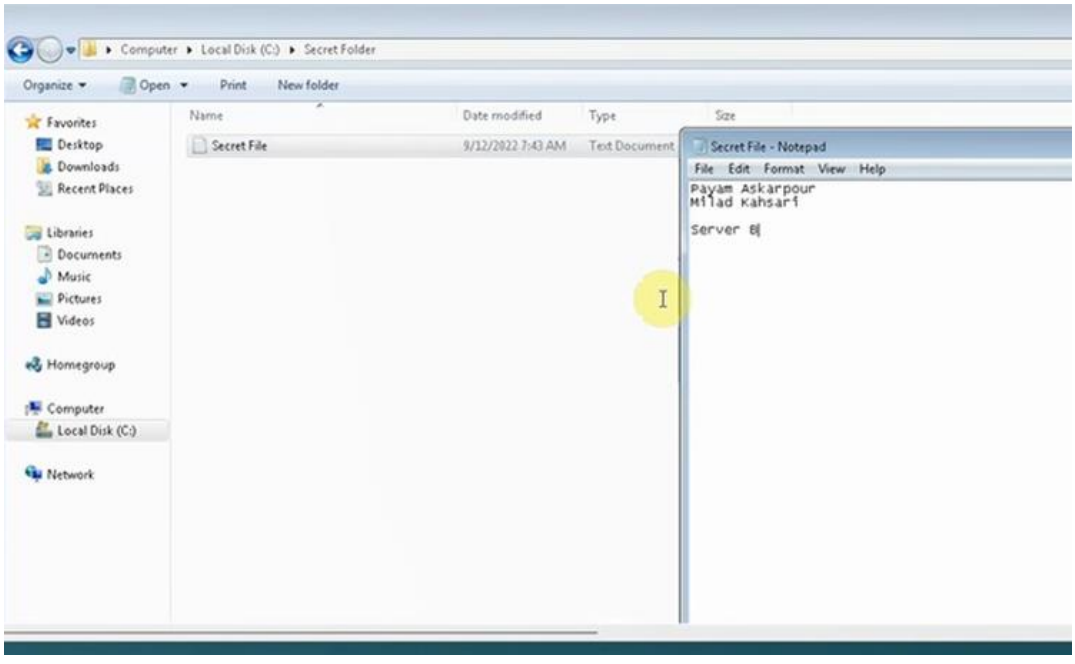
آزمایش نفوذپذیری با هدف حمله به سیستم‌عامل ویندوز و دسترسی به فایل‌های محرمانه از طریق حمله از نوع RCE[1] وانجام lateral movement صورت گرفته است. منظور از انجام lateral movement نفوذ به یک سیستم و قراردادن آن سیستم به عنوان پایگاه حملات بعدی جهت نفوذ به سایر سیستم‌های غیر قابل دسترس به صورت مستقیم است. در این حمله از سه سیستم‌عامل استفاده شده که سیستم‌عامل مهاجم Kali Linux بوده و از طریق آسیب‌پذیری MS17-010 موجود در یک سیستم‌عامل ویندوزی مهاجم اقدام به نفوذ به آن ماشین کرده است. در گام بعد، مهاجم آن را پایگاه حمله به سیستم‌عامل دیگری قرار داده است. از طریق این نوع حمله، امکان اجرای کدهای مخرب را بر روی سیستم‌عامل هدف فراهم می‌نماید.

آسیب‌پذیری که از طریق آن این حمله صورت پذیرفته از جمله خطرناک‌ترین آسیب‌پذیری‌هایی بوده که به مدت طولانی بدون شناسایی مانده و در تمامی نسخه‌های سیستم‌عامل ویندوز تا زمان کشف آن یعنی سال 2017 قابل exploit بوده است. نفوذگر با استفاده از این آسیب‌پذیری به راحتی به کلیه امور سیستم احاطه داشته و هرگونه تغییری را در سیستم می‌تواند پیاده‌سازی کند. این آسیب‌پذیری با شماره MS17-010 / CVE-2017-0146 معرفی گردید که وصله‌های مرتبط با آن نیز منتشر شده است، ولی با توجه به عدم به‌روزرسانی به موقع سیستم‌عامل توسط بسیاری از کاربران همچنان تا مدت زیادی شاهد وقوع حملاتی با استفاده از این آسیب‌پذیری بودیم. لازم به ذکر است این آسیب‌پذیری توسط NSA به مدت زیادی استفاده شده بود.

در این سناریو هدف دسترسی به هر دو سیستم‌عامل ویندوزی آسیب‌پذیر است، به منظور دسترسی به اطلاعات محرمانه دو پوشه‌ای که به عنوان flag درنظر گرفته‌ایم. این flag در درایو C در یک پوشه با نام Secret Folder که حامل فایل از نوع TXT با نام Secret File با محتوای نام Payam Askarpour و Milad Kahsari قرار دارد. در تصویر 14، محتوای این فایل و همچنین نام پوشه نمایش داده شده است.

---

[1] Remote code execution



تصویر 14: پوشه حاوی اطلاعات محرمانه که به عنوان flag در این آزمایش در نظر گرفته شده است.

دامنه آزمایش نفوذپذیری

هدف این گزارش انجام آزمایش نفوذپذیری با استفاده متدولوژی استاندارد و ساختارمند PTES[1] است. با استفاده از فازهای هفتگانه آزمایش نفوذپذیری PTES اقدام به بررسی امنیت سامانه های هدف شده است و در نهایت با محوریت رویکرد نفوذ از طریق آسیب‌پذیری MF17-010 و همچنین برقراری ارتباط روی پورت 445 (پروتکل SMB)با ماشین هدف گزارش ارائه خواهد شد. در این حمله، به یک سیستم‌عامل ویندوزی نفوذ خواهد شد و سپس از آن برای انجام حمله به هدف ثانویه در شبکه هدف استفاده خواهد شد. در جدول 1 اطلاعات این ماشین‌ها آورده شده است:

Operating System	IP Address
Kali Linux	192.168.200.129
Windows 7	192.168.200.130
Windows 7	192.168.200.131

جدول 1: مشخصات ماشین‌های استفاده شده در این آزمایش

[1] Penetration Testing Execution Standard

ابزارها و ماژول‌ها

ابزارهایی که در این آزمایش نفوذپذیری مورد استفاده قرار گرفتند به شرح زیر هستند:

- ابزارهای عمومی شبکه مانند commandهایی مانند ping، ip و ...
- ابزار ترسیم‌گر شبکه Nmap
- فریمورک متاسپلویت (Metasploit Framework)
- انکدر متاسپلویت msfvenom

ماژول‌ها و اسکریپت‌های به کار برده شده از چارچوب متاسپلویت به شرح ذیل هستند:

- auxiliary/scanner/smb/smb_ms17_010
- exploit/windows/smb/ms17_010_psexec
- auxiliary/scanner/portscan/tcp
- post/multi/manage/autoroute

روش انجام آزمایش نفوذپذیری نفوذ

این آزمایش نفوذپذیری نفوذ بر اساس استاندارد PTES، بر اساس فازهای 1 تا 6 به صورت ذیل انجام شد:

- تعاملات قبل از آزمایش نفوذپذیری
- جمع‌آوری اطلاعات در مورد هدف
- مدل‌سازی هدف
- تحلیل آسیب‌پذیری‌ها
- بهره‌برداری از آسیب‌پذیری
- پس از بهره‌برداری از آسیب‌پذیری

فاز تعاملات قبل از آزمایش نفوذپذیری

در این مرحله نسبت به تعیین دامنه آزمایش نفوذپذیری / حمله و انجام توافقات لازم جهت اجرای دقیق پروژه اقدام باید صورت گیرد. یکی دیگر از مواردی که در این مرحله مشخص خواهد شد، راه‌اندازی اولیه آزمایش نفوذپذیری شامل آماده‌سازی سیستم‌عامل و ابزار و نرم‌افزارهای لازم می‌باشد. لذا برای انجام این آزمایش نفوذپذیری نسبت به ایجاد یک آزمایشگاه ساده اقدام کردیم. در این آزمایشگاه سه ماشین جهت انجام آزمایش نفوذپذیری نصب گردید.

یک سیستم‌عامل کالی لینوکس و دو سیستم‌عامل Windows7 (نسخه با آسیب‌پذیری) نصب و شبکه مجازی بین دو کامپیوتر که از این به بعد آنها را به ترتیب سیستم A و سیستم B خطاب خواهیم کرد، ایجاد شد. در این فرآیند آزمایش نفوذپذیری، ابتدا از نرم‌افزار Nmap که ترسیمگر معماری و ساختار شبکه است برای انجام پویش سیستم‌عامل مقصد استفاده شده است. همچنین جهت انجام تحلیل آسیب‌پذیری و اجرای حمله و Exploit نمودن آسیب‌پذیری‌های موجود از ابزار MSF استفاده نمودیم.

فاز جمع‌آوری اطلاعات

در مرحله جمع‌آوری اطلاعات، از اولین اقداماتی که صورت می‌گیرد پویش سیستم‌عامل هدف به منظور یافتن آسیب‌پذیری با استفاده از نرم‌افزار Nmap می‌باشد. در این پویش به دنبال پورت‌های باز و نسخه برنامه‌ها و سرویس‌های عملیاتی در سطح شبکه خواهیم گشت تا به احتمال زیاد بتوانیم پس از آن از آسیب‌پذیری موجود در سرویس استفاده کرده و سیستم را با استفاده از آن آسیب‌پذیری مورد استفاده قرار بدهیم.

لازم به ذکر است با توجه به پویش انجام شده سیستم‌عامل هدف، نوع سیستم‌عامل Windows 7 تشخیص داده شد. با استفاده از ابزار Nmap یک پویش از نوع TCP SYN که با توجه به half-open بودن sessionها ماهیتاً مخفیانه[1] می‌باشد، انجام دادیم که در آن OS detection نیز انجام می‌دهیم و حتی قید کردیم که در صورت قابل پینگ نبودن یک هاست آن را پویش نماید. در تصویر 15 این مسئله نمایش داده شده است.

```
root@kaliP:~# nmap -sV -sS -T4 -O -Pn -A -v 192.168.200.130
Starting Nmap 7.92 ( https://nmap.org ) at 2022-09-12 10:33 CDT
```

تصویر 15: TCP SYN scan سیستم A

در پایان پویش تعدادی پورت باز پیدا شد و پورتی که ما درحمله تصمیم گرفتیم که مورد استفاده قراردهیم، پورت 445 است که در پس زمینه آن سرویس SMB در حال اجراست. نوع سیستم‌عامل هم Windows 7 مشخص گردید و روی آن پورت‌های بسیاری باز تشخیص داده شد. در تصویر 16 این مسئله نمایش داده شده است.

```
445/tcp   open  microsoft-ds   Windows 7 Enterprise 7601 Service Pack 1 microsoft-ds (workgroup: WORKGROUP)
515/tcp   open  printer        Microsoft lpd
554/tcp   open  rtsp?
2103/tcp  open  msrpc          Microsoft Windows RPC
2105/tcp  open  msrpc          Microsoft Windows RPC
2107/tcp  open  msrpc          Microsoft Windows RPC
2869/tcp  open  http           Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
3306/tcp  open  mysql          MariaDB (unauthorized)
3389/tcp  open  ms-wbt-server?
| ssl-date: 2022-09-12T15:37:36+00:00; -19s from scanner time.
| ssl-cert: Subject: commonName=IE8Win7
| Issuer: commonName=IE8Win7
```

تصویر 16: تصاویر پورت‌های باز سیستم

[1] Stealth

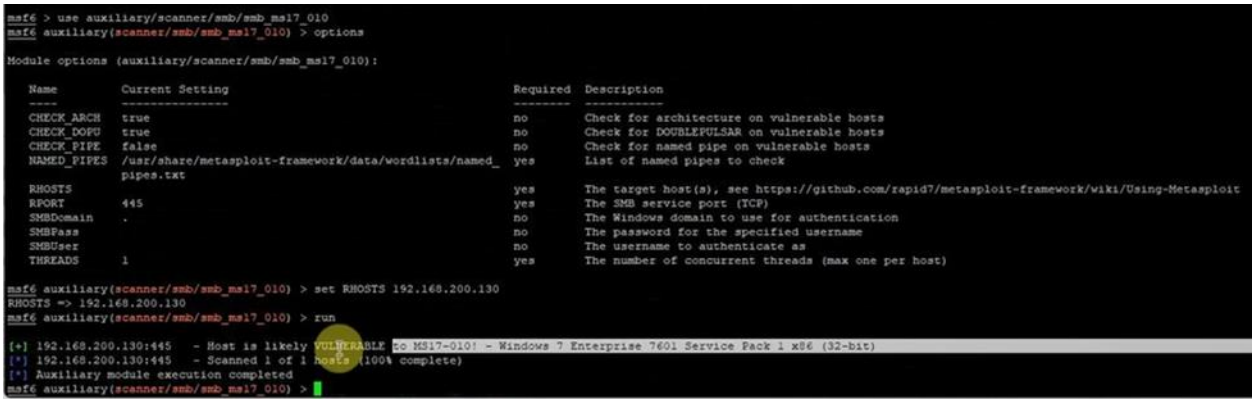
در این مرحله جمع‌آوری ساده اطلاعات از ماشین های عملیاتی موجود در شبکه هدف با موفقیت انجام پذیرفت. حال باید با بررسی و تحلیل این اطلاعات یک وکتور حمله برای هدف قرار دادن شبکه مورد استفاده قرار بدهیم. در فاز بعد به تحلیل خروجی فاز قبلی خواهیم پرداخت.

فاز مدل‌سازی هدف

در این مرحله چون ما بر اساس آزمایش پیش رفته‌ایم و حمله واقعی نیست شاید تعریف مدل خیلی مفهومی نداشته باشد، ولی در واقعیت زمانی که ما می‌خواهیم به هدفی حمله کنیم، مشابه آن سیستم با همان سرویس‌ها را آماده خواهیم کرد و سپس اقدام به بررسی نقاط ضعف سیستم هدف خواهیم کرد تا بتوانیم Vulnerability بر روی آن محیط شناسایی کنیم و بعد از اطمینان به هدف اصلی مان بر می‌گردیم تا آن را با استفاده از آن آسیب پذیری مورد حمله و نفوذ قرار بدهیم و بعد فازهای دیگر عملیات را پیش ببریم.

فاز تحلیل آسیب‌پذیری‌ها

جهت تحلیل آسیب‌پذیری‌های سیستم های هدف، از چارچوب Metasploit یا MSF نسخه community که بر روی سیستم‌عامل Kali قرار دارد، استفاده نمودیم. با توجه به نتیجه اطلاعات جمع آوری شده در فاز جمع‌آوری اطلاعات و اطمینان از باز بودن پورت 445 از یک سو و از سوی دیگر نوع سیستم‌عامل هدف بررسی نمودیم که آیا هدف ما دارای آسیب‌پذیری MS17-010 هست یا خیر.

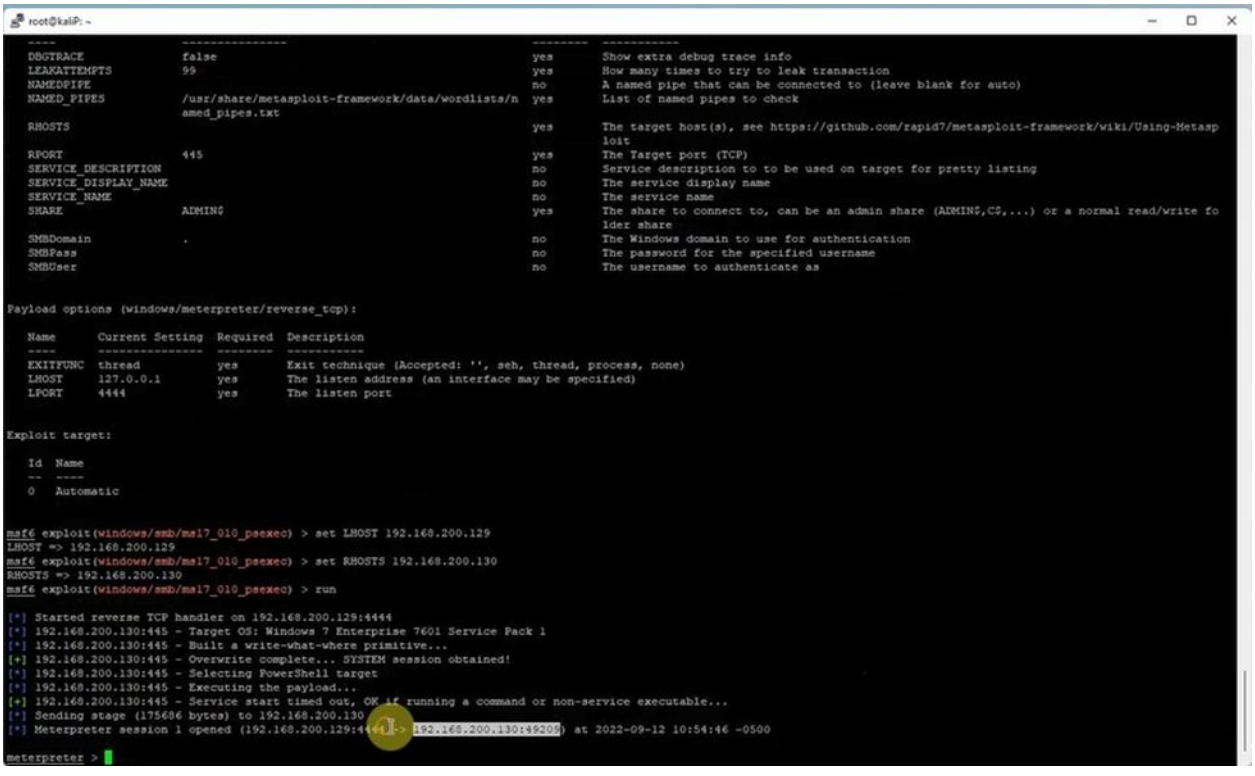


تصویر 16: تحلیل آسیب‌پذیری Windows A

همانطوری که در تصویر 16 مشاهده می‌کنید، سیستم‌عامل هدف از نوع ویندوز 32 بیتی نسخه 7 تشخیص داده شده است که در برابر آسیب‌پذیری MS17-010 نفوذپذیر است.

فاز بهره‌برداری از آسیب‌پذیری

در این مرحله جهت استفاده از آسیب‌پذیری MS17-010 توسط دستور search، exploit مرتبط را یافته و انتخاب نمودیم. با انجام تنظیمات مورد نیاز بر روی exploit مذکور همانگونه که در تصویر 17 قابل مشاهده است، حمله خود را با دستور run آغاز کردیم.



تصویر 17: حمله به Windows A

پس از اتمام حمله همانگونه که در تصویر مشاهده می‌گردد، یک meterpreter shell از نوع reverse tcp بر روی هدف تحت عنوان session 1 دریافت کردیم. پس از در اختیار گرفتن session مذکور اقدام لازم جهت دستیابی به اطلاعات محرمانه سیستم A صورت پذیرفت. مراحل مورد نیاز جهت تسخیر flag تنظیم شده با نام Secret file.txt در پوشه Secret folder در تصویر 18 قابل مشاهده است.

تصویر 18: تصویر دریافت دسترسی به اطلاعات محرمانه (آزمایشگاهی)

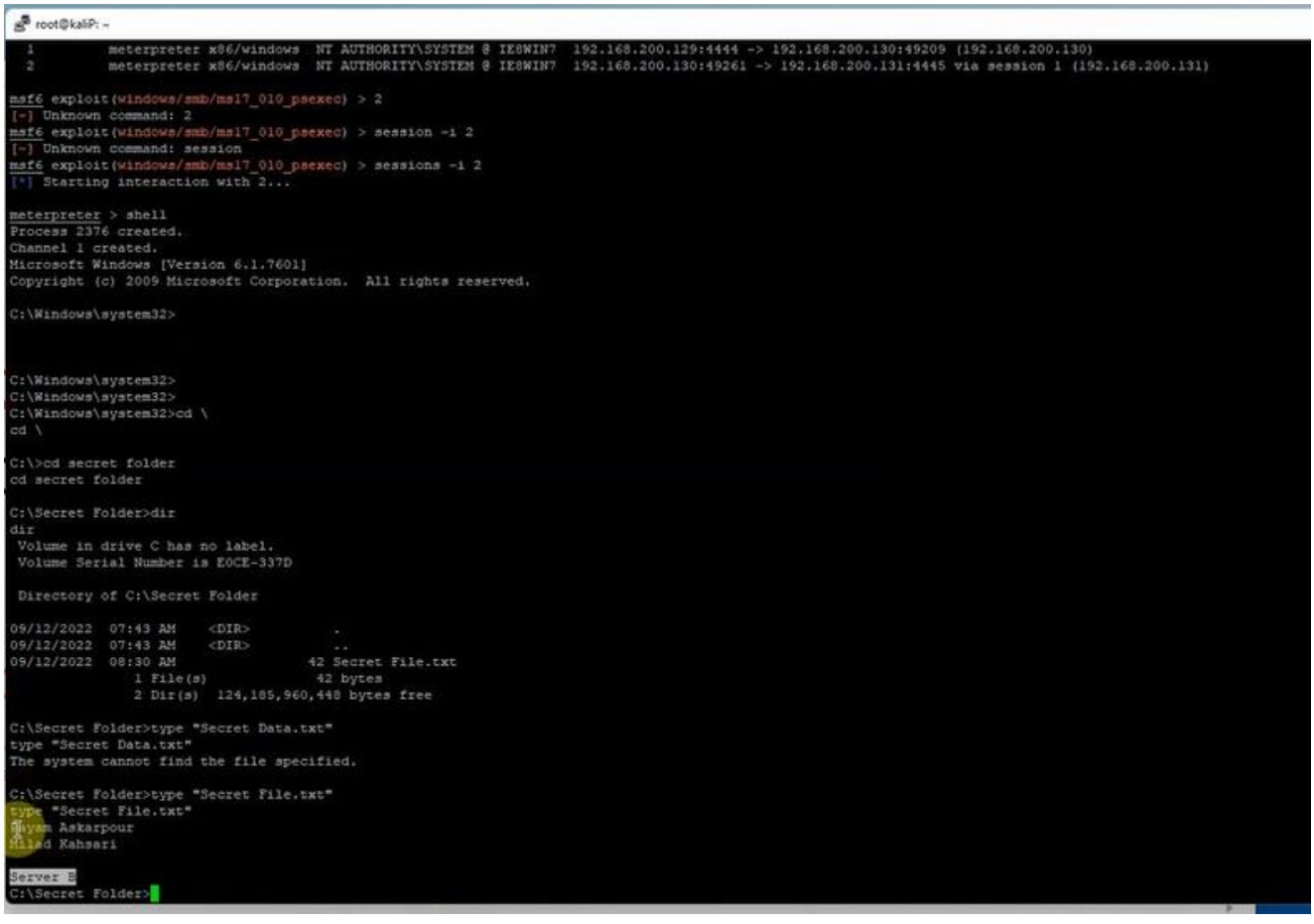
فاز پس از بهره‌برداری از آسیب‌پذیری

در این مرحله معمولا از نصب backdoor، جهت تداوم دسترسی به سیستم‌عامل هدف استفاده می گردد. یکی دیگر از عملیات مهم در این مرحله پاک سازی رد پا و لاگ های سیستم‌عامل قربانی می باشد. در این مرحله با توجه به دسترسی اخذ شده از قربانی میزان دقیق شدت آسیب‌پذیری ها و ارزش داده ها تخمین زده شده و همچنین امکان انجام lateral movement جهت نفوذ به سایر سرورهای آسیب پذیر بررسی می گردد.

انجام Lateral Movement

جهت اجرای عملی این مرحله از سیستم A هدف بعدی به نام سیستم B را پینگ نمودیم. با توجه به برقراری ارتباط شبکه بین سیستم A و B، اقدام به پویش سیستم B از طریق سیستم A نموده و پس از اطمینان از باز بودن پورت 445 در هدف دوم، آنرا برای آسیب‌پذیری MS17-010 پویش کرده و سپس Exploit نمودیم. لازم به ذکر است جهت انجام این امر از Pivot یا Lateral Movement با استفاده از قابلیت autoroute در چارچوب MSF نمودیم.

همانطور که در تصویر ذیل مشاهده می‌گردد در سیستم دوم نیز فایل Flag مربوطه با محتوای محرمانه Capture شده و نمایش داده شده است. نکته قابل توجه در این مرحله استفاده از payload با قابلیت اتصال مستقیم بوده و بر خلاف مرحله اول برای تنوع و نمایش قابلیتها از ارتباط direct استفاده گردید. ضمناً تمامی ارتباطات با هدف دوم از مبدا مهاجم از طریق تانل برقرار شده در Session 1 استفاده شده است.



تصویر 19: تصویر CTF هدف دوم

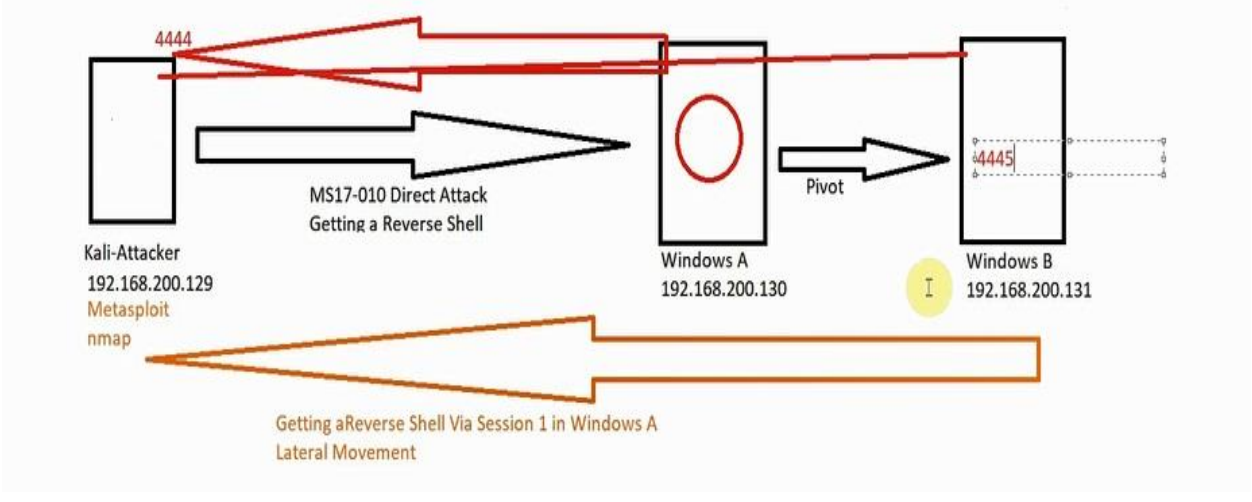
قرار دادن Backdoor

در این مرحله می‌توان با استفاده از ابزار msfvenom، backdoor مورد نظر خود را تولید و با استفاده از session های ایجاد شده و ارتباط مبتنی بر pivoting ای که بر روی سیستم B از طریق سیستم A داریم، backdoor را بر روی سیستم B قرار می‌دهیم. به عنوان مثال می‌توان payload تولید شده برای backdoor را از نوع reverse_tcp در نظر گرفت و با استفاده از قابلیت اضافی port forwarding ای که در session های meterpreter پیاده سازی شده است، ارتباط backdoor از روی سیستم B و از طریق سیستم A به حمله کننده را میسر نمود. برای این منظور می توان از دستور portfwd در meterpreter session گرفته شده بر روی سیستم A استفاده کرد که لازمه آن این است

که بر روی سیستم حمله کننده پیش از این listener یا multi-handler مورد نیاز که بر روی port forwarding ای که بر روی سیستم A مشخص می گردد، در حالت listen قرار داشته باشد.

پاک‌سازی لاگ‌ها

در پایان با استفاده از قابلیت‌های پیاده‌سازی شده در meterpreter sessionها می‌توان در خصوص پاکسازی رد اقدامات انجام شده و event های مربوطه بر روی سیستم های مورد حمله اقدام نمود. دستور مورد استفاده در این خصوص clearev می‌باشد که به صورت اتوماتیک eventهای مرتبط را پاک می‌نماید. کل سناریوی انجام شده و اقدامات صورت گرفته در این گزارش را می‌توان با تصویر زیر خلاصه نمود:



تصویر 20: نمای کلی حمله

نتیجه‌گیری

در این مقاله، بعد از بررسی ماهیت سیستم‌ها و تفاوتی که بین آن‌ها وجود دارد، اقدام به بررسی استاندارد PTES برای ارزیابی و شناسایی نقاط ضعف یک سیستم کردیم. در گام بعد، با استفاده از فریمورک و استاندارد PTES اقدام به پیاده‌سازی یک حمله بر روی یک محیط آزمایشگاهی کردیم تا با جزئیات بیشتر با شیوه پیاده‌سازی حملات هدفمند آشنا شویم.