مراحل کل یک تهاجم

مرحله اول: کسب اطلاعات و شناسایی مقدماتی سیستم.

مرحله دوم: پویش و جستجو در شبکه بدنبال محل مناسب برای نفوذ.

مرحله سوم: نفوذ و راهیابی به سیستم (متداولترین O.S یا APP ها یا شبکه و...).

مرحله چهارم: تثبیت نفوذ و سیطره بر شبکه و سیستم.

مرحله پنجم: پوشش مسیرها و پنهان کردن ردپا و ردگم کردن.

گام اول: شناسایی مقدماتی

در اولین گام باید مجموعه ای از اطلاعات در خصوص شبکه هدف بدست آورد و مشخصات فنی و عمومی آن را شناسایی کرد.

مثال میدان جنگ: چگونه می توان بدون آگاهی از موقعیت جغرافیایی دشمن، محل استقرار نیروها، حجم نیروها، مهارت و ... حمله کرد.

یا سارق بانک .

روشهای شناسایی مقدماتی:

الف) مهندسی اجتماعی

برقراری ارتباط با یکی از کارمندان و طرح دوستی

از طریق منشی یا روابط عمومی

از طریق تماس با گروه پشتیبانی فنی (Technical Support Team)

مثلاً فردی به مدیر سیستم زنگ میزند و بیان می کند که با **user :A** و **pass :X** نمی تواند متصل شود و می خواهد از او به رئیس شکایت کند. در صورتی که مدیرسیستم فریب بخورد به تماس گیرنده کلمه عبور جدیدی خواهد داد!

راه حل الف) آگاهی دادن و آموزش به دست اندرکاران شبکه

برقراری نظم اداری و طی کردن روال قانونی کارها و...

ب) دسترسی مستقیم و فیزیکی

+ شما با ID خودتون login کردید ولی پشت میز نیستند. امکان نصب برنامه , کپی , Del ...
+ پورت سخت افزاری آن را روی hub یا سوییچ یا laptop
+ دزدیدن CD یا دیسکت یا باز کردن هارد
+ چسباندن ID وPass کنار مانیتور

راه حل ب) نگهبانی قوی

کارت شناسایی

دوربین مدار بسته

ابزارهای احراز هویت حرفه ای (اثر انگشت ,...)

Screen Server مبتنی بر Password

ج) آشغالگردی Dumpster Diring

کاغذ باطله، دیسکت خراب، هارد سوخته

مثال: سال 2000 شرکت اوراکل چند میلیون دلار هزینه کرد تا در آشغال های مایکروسافت دنبال کاغذ باطله و کد برنامه و ... باشند.

راه حل ج) کاغذ خرد کن

انهدام زباله های اداری به نحو مناسب CD، دیسکت و ... .

د) جستجو در وب و اینترنت. STFW (Search The Fine Web sites)

وقتی یک شرکت تاسیس می شود و مثلاً یک Domain Name ثبت می کند، اطلاعات زیادی ارائه می کند که می توان آنها را بدست آورد.

مثلاً: آدرس های IP ثبت شده

ایمیل افراد یا حتی ID چت

نوع سرویس های عرضه شده

یا در سایت مؤسسه :

شماره تماس کارکنان

شرکت های تجاری

یا حتی تکنولوژی و O.S. و ... .

یا AltaVista (یک آدرس می گیرد سایتهای که به آن صفحه لینک داده اند را استخراج می کند)

مقابله با د)

آموزش کارکنان

عدم استفاده از اطلاعات سری و محرمانه در وب سایت مؤسسه

خودمان بگردیم و نباید اطلاعات محرمانه از خود را از طریق اینترنت پیدا کنیم .

ه) بانک اطلاعاتی who is

این بانک های اطلاعتی , اطلاعاتی نظیر آدرس های IP , Domain Name ها و روش برقراری تماس با یک فرد مسئول در شبکه را ارائه می دهند.

برای پیدا کردن مشخصات صاحبان یک آدرس با پسوند .com , .net , .org می توان به سایت www.internic.net/whors.html رجوع کرد.

اما برای پیدا کردن مشخصات صاحبان سایر آدرس ها با پسوندهای کشورها می توان به www.allwhois.com/home.html رجوع کرد.

نکته: از طریق سایت هایی مانند ARIN می توان فهمید که یک آدرس IP متعلق به چه شرکت یا سازمانی است.

آمریکا – آفریقا	www.Arin/whors/arinwhors.html
اروپا	www.ripe.net
آسیا	www.apnic.net

راه مقابله با ه) راه مقابله ندارد چون قانونی است.

گام دوم: پویش و جستجو به دنبال رخنه ای برای نفوذ

در مرحله اول نفوذگر مقداری اطلاعات مقدماتی در ارتباط با شبکه هدف بدست آورده است :

شامل: تعدادی شماره تلفن از خطوط دسترسی به شبکه

تعدادی آدرس IP از شبکه هدف

تعدادی Domain Name

و ... .

در این مرحله نفوذگر به دنبال یک راه برای نفوذ می گردد .

الف) جستجوی مودم های شبکه یا War Dialing

نکته: اغلب کاربران نا آگاه برای اینکه از منزل بتوانند به PC اداره متصل شوند یک مودم بر روی PC اداره نصب می کنند و با استفاده از آن بتوانند کارهای خود را انجام دهند .

روال کار: یک نرم افزار سرویس دهنده بر روی PC اداره نصب می کنند مانند PC any where و مودم و تلفن را وصل می کنند. حال از راه دور با این ماشین ارتباط برقرار می کنند و همانند یک کاربر معمولی در محل از آن سرویس می گیرند.

حال نفوذگر با در اختیار داشتن فهرستی از شماره تلفن های هدف , شروع به پویش آنها می کند , برای این کار از یک ابزار خودکار استفاده می کند تا پی در پی با این شماره ها تماس بگیرد به این ترتیب اگر یک سیگنال متعلق به یک مودم را یافت , شماره تلفن آن را ذخیره می کند.

به این کار War Dialing گویند.

سپس نفوذگر پس از شناسایی این خطوط بر روی آنها متمرکز می شود تا موارد زیر را بدست آورد :

1- نوع مودم و پروتکل ارتباطی آن
2- سرویس دهنده آن مودم
3- کلمه عبور برای اتصال به آن مودم

به این کار Demon Dialing گویند.

نکته: در واقع یک War Dialer در بین مجموعه بزرگی از شماره تلفن ها برای یافتن یک مودم فعال جستجو می کند اما یک Demon Dialer برای یافتن کلمه عبور و راهی جهت نفوذ به یک مودم و شماره شناسایی شده، جستجو می کند.

ابزارهایی مثل THC-Scan (thc.inferno.tusculum.edu)، X-Dial , X-Dialer و ... .

یک War Dialer وقتی شماره ای را می گیرد با 3 حالت مواجه می شود ;

1) خط اشغال است. پس بعداً باید دوباره شماره گیری شود.
2) خط آزاد است. اما یا کسی برنمی دارد یا یک فرد گوشی را برمی دارد.
3) خط آزاد است و به مودم متصل است.

مقابله با الف)

+ وجود قوانین سخت گیرانه و مدرن جهت استفاده از مودم ها و خطوط تلفن.
+ نظارت بر روی مودم ها , می توان آنها را به طور متمرکز یک جا قرار داد.
+ استفاده از کلمات عبور طولانی و مشکل.
+ در نهایت خودمان یک War Dialing انجام دهیم تا مودم های فعال را کشف کنیم. War Dialing فعلاً کنار بذاریم

ب) نقشه برداری از شبکه

پس از اینکه اطلاعات اولیه از شبکه بدست آمد می توان اطلاعات دقیق تری بدست آورد. یکی از مهمترین مسائل، شناخت ساختار و توپولوژی شبکه است. در حقیقت یافتن معماری شبکه و آرایش سرورها , دیواره های آتش، مسیر یابها و ... .

در این حالت اگر نفوذگر از درون شبکه اقدام کند، احتمال موفقیت او بیشتر است اما اگر بخواهد از بیرون شبکه اقدام کند باید هزینه بیشتری صرف کند .

برای نقشه برداری از یک شبکه مراحل زیر را باید انجام داد:

1) **تشخیص ماشین های فعال**:

برای این کار می توان از Ping استفاده کرد. فرض کنید در گام اول، نفوذگر آدرس IP یک شبکه را از کلاس C و 192.150.40.0 تشخیص داده است بنابراین 254 ماشین موجود در شبکه را Ping می کند. البته ممکن است Firewall اجازه عبور بسته های Icmp را ندهد.

در این حالت می توان بسته های TCP از نوع SYN بر روی پورت 80 ارسال کرد و منتظر پاسخ SYN=1 , Ack=1 ها ماند یا حتی بسته های نا متعارف TCP که باعث Rst=1 پاسخ می شوند ارسال کرد.

2) **Trace Route برای کشف توپولوژی**:

برای این کار مهاجم یک بسته با Ttl=1 ارسال می کند , در نتیجه اولین مسیریاب آن را حذف می کند و در یک پیغام به مبداء ارسال می کند در این پیغام آدرس مسیریاب حذف کننده وجود دارد.

سپس Ttl=2 , Ttl=3 , ...

در ویندوز فرمان Tracert و در لینوکس Traceroute وجود دارد.

این کار را برای تمام ماشین های فعال انجام می دهیم.

در پایان نفوذگر با تحلیل نتایج Trace route یک نقشه تقریبی از شبکه دارد.

البته برای این کار نرم افزارهایی نیز وجود دارد. مانند: Cheops www.marko.net/cheops/

راه حل جلوگیری از ب)

+ بر روی ماشین هایی که لزوم ندارد، ICMP را نیز فعال می کنیم.
+ بر روی Firewall نیز پیغام های ICMP از بیرون شبکه , حذف شوند.

ج) تعیین پورت های باز بر روی یک ماشین

این مرحله بعد از نقشه برداری از شبکه انجام می شود.

در این حالت نفوذگر ماشین های فعال و توپولوژی شبکه را می شناسد.

حال نفوذگر می خواهد بداند هر ماشین چه وظیفه ای بر عهده دارد و چه خدماتی ارائه می کند.

می دانیم هر ماشین می تواند 65535 پورت TCP و 65535 پورت UDP داشته باشد.

هر گاه یک پروسه در حال اجرا و گوش دادن به یک پورت باشد، اصطلاحا آن پورت باز است.

هر پورت باز روی ماشین در حقیقت یک درب ورودی به آن ماشین محسوب می شود.

حال اگر پورتهای TCP و UDP را به منزله درهای ورودی یک ماشین فرض کنیم، عمل Port Scanning به عنوان در زدن می باشد تا کشف کنیم آیا پروسه ای پشت این در هست یا خیر.

برای این کار ابزارهایی وجود دارند با عنوان Port Scanner که فهرستی از پورت ها را بررسی و پویش می کند و از باز یا بسته بودن آنها آگاه می شود. بهترین آنها Nmap می باشد. www.insecure.org/Nmap

جهت پویش پورت های باز، مکانیزم های گوناگونی وجود دارند:

1) **Polite scan** **پویش مؤدبانه**

در این حالت نرم افزار پویشگر یک ارتباط کامل و دو مرحله ای TCP با یک پورت خاص برقرار می کند و اگر این اتصال برقرار شود , پورت مربوطه کاملا" باز است.

ابتدا پویشگر یک بسته SYN=1 , Ack=0 برای ماشین هدف ارسال می کند.

پس منتظر یک پاسخ SYN=1 , Ack=1 می ماند، اگر پاسخ برگشت، پورت باز است و بسته Ack ارسال می شود، حال ارتباط برقرار است و پویشگر بلافاصله یک بسته Fin=1 ارسال می کند و ارتباط خاتمه می یابد.

نکته: دقت کنید که عدم بازگشت SYN=1 , Ack=1 به معنای بسته بودن صد در صد پورت نیست بلکه اگر پورت بسته باشد باید پیغام Rst=1 برگردد. (یا حداقل ICMP Port unreachable).

نکته: این نوع پویش بیشتر جهت اهداف مدیریتی استفاده می شود و نفوذگران به دو علت از آن استفاده نمی کنند:

1) بسیاری از سرویس دهندگان به محض تکمیل شدن 3 مرحله برقراری ارتباط TCP، مشخصات این ارتباط را ثبت می کنند، لذا به سادگی آدرس نفوذگر کشف می شود.
2) تکمیل این 3 مرحله بسیار وقتگیر است و باید به ازای هر پورتی انجام شود.

2)TCP SYN Scan

این روش شامل 3 مرحله زیر است:

1) پویشگر یک بسته ی syn به سمت ماشین هدف ارسال می کند.
2) پویشگر مدت زمان مشخص برای بازگشت syn-ack صبر می کند بازگشت این بسته نشان می دهد پورت باز است.

3) هنگامی که بسته ی syn-ack برگشت پویشگر بلافاصله یک بسته ی reset ارسال می کند و هیچ ارتباطی برقرار نمی شود.

در واقع در این روش فقط دو مرحله از دست تکانی سه مرحله ای انجام می شود.

به این ترتیب نفوذگر در یک حاشیه ی امنیت قرار دارد زیرا مشخصات یک ارتباط نیمه کاره ثبت نمی شود و در ضمن سرعت عمل بالا می رود.

3) پویش به روش نقض اصول پروتکل:

با دقت در 2 روش پویش پیشین متوجه می شویم هر 2 در واقع با ارسال بسته ی syn منتظر syn _ack می مانند.

اما در این روش در مرحله اول بسته ای ارسال می شود که معمول و متعارف نیست. برای این کار معمولا از 3 نوع بستهء زیر استفاده می شود:

a) TCP FIN Scan

در این روش بدون هیچ مقدمه ای یک بسته ی TCP FIN بر روی یک پورت مشخص ارسال می کنند در این حالت اگر پورت مورد نظر بسته باشد یک rspsl برگشت داده می شود اما در غیر این صورت و اگر پورت باز باشد هیچ بسته ای برگشت داده نمی شود. دقت کنید عدم برگشت پاسخ یعنی احتمالا باز است.

b) Null scan

در این روش یک بسته ی TCP بر روی یک پورت ارسال می شود که هیچ یک از بیت های SYN و RST و ACK آن برابر یک نیستند. این بسته طبق تعریف TCP هیچ مفهومی ندارد و اگر پورت باز باشد حذف می شود و اگر بسته باشد یک RST بر می گرداند. باز هم عدم بازگشت پاسخ یعنی احتمالا باز است.

c) Xmas Free

در این حالت پویشگر یک بسته ارسال می کند بایت های fin = 1 و urg =l و push =l که این بسته نیز طبق تعریف TCP هیچ مفهومی ندارد و اگر پورت باز باشد هیچ پاسخی نمی آید اما اگر پورت بسته باشد یک بسته با rsp=l برمی گردد.

✧ **نکته**: این 3 روش در همه ی o.s. ها به خوبی عمل می کنند به جز ویندوز زیرا بر خلاف اصول TCP در ویندوز هر بسته ی نامتعارفی هم که دریافت شود یک بسته RST برمی گردد.

4) TCP Ack Scan

در این روش، یک بسته ی syn-ack بدون مقدمه به سوی یک پورت در ماشین هدف ارسال می شود (دقت کنید به طور معمول این بسته در پاسخ به یک بسته ی syn ارسال می شود.) حال وقتی این بسته به یک پورت باز برسد آن را حذف می کند اما اگر پورت بسته باشد یک بسته ی RST = 1 بر می گردد.

✧ **نکتهٔ مهم**: این روش یک مزیت عمده نسبت به روش های پیشین دارد و آن این که این بسته ها می توانند از دیوار آتش عبور کنند دقت کنید که به طور متعارف یک دیوار آتش از یک شبکه داخلی که هیچ سرویسی را به خارج ارائه نمی دهد محافظت می کند و اجازه نمی دهد هیچ بسته ی syn به شبکه وارد شود. زیرا بسته ی syn به معنای درخواست

سرویس گیری می باشد و به این ترتیب مکانیزم های polite scan و TCP syn scan قادر نخواهند بود از دیوار آتش عبور کنند اما بسته های syn-ack می توانند به شبکه وارد شوند.

به این ترتیب اگر بعد از ارسال یک syn-ack یک RST دریافت شد به این معناست که بر روی firewall آن پورت آزاد است اما بر روی ماشین مقصد بسته است. با این روش می توان سیاست firewall را شناسایی کرد. اما اگر پاسخی برنگشت یا پورت کاملا باز است هم در fire wall هم در ماشین مقصد و یا firewall آن را فیلتر کرده.

5) FTP Bounce scan

نکته: در سرویس های ftp سنتی یک سرویس به صورت زیر ارائه شده است

یک کاربر می تواند ضمن برقراری یک ارتباط TCP از سرویس دهنده بخواهد که یک فایل را برای یک ماشین ثالث ارسال کند. فرض کنید یک کاربر از روی یک ماشین با پهنای باند پایین به یک سرویس دهنده ی قوی متصل است اما چون پهنای باند وی کم است از سرویس دهنده می خواهد که یک فایل را بر روی یک ماشین دیگر ارسال کنند. File forwarding

حال سناریوی زیر را در نظر بگیرید:

نفوذگر با یک سرویس دهنده ی ftp یک ارتباط TCP برقرار می کند و از وی می خواهد فایل را برای یک پورت خاص از ماشین هدف ارسال کند در این حالت اگر پورت مورد نظر بر روی ماشین هدف بسته باشد سرور ftp به نفوذگر گزارش می دهد و اگر پورت مورد نظر باز باشد به نفوذگر گزارش می دهد که پورت باز است اما انتقال فایل میسر نیست (چون ماشین هدف منتظر این ftp نبوده) با این ترتیب می توان پورت های دیگر ماشین هدف را پویش کرد.

نکتهٔ مهم: این روش یک مزیت بسیار بزرگ دارد و آن این که نفوذگر کاملا مخفی می ماند زیرا ماشین هدف حتی اگر همه ی این اتفاقات را ثبت کند فقط مشخصات سرویس دهندهٔ ftp را ثبت می کند که در حقیقت یک واسطهٔ خودی می باشد.

6) استفاده از بسته های udp برای پویش

یادآوری: udp یک پروتکل بدون اتصال است و بسته ها ممکن است به کلی گم شوند و یا از بین روند در ضمن در سرآیند آن فلگ های SYN و RST - ACK – FIN و .... وجود ندارد.

برای پویش پورت های udp می توان یک بسته را برای یک پورت خاص ارسال کرد و اگر پاسخ ICMP port uncechable آمد می توان گفت صد در صد پورت بسته است در غیر این صورت هیچ چیز نمی توان گفت.

نکته: یک نکتهٔ کلی در مبحث port scaning انتخاب شماره ی مناسب برای پورت مبدا می باشد. زیرا این بسته باید قادر باشد از دیوار آتش عبور کند و دیواره های آتش بسته های با برخی از شماره پورت ها را حذف می کند.

برخی از شماره پورت های مبدا که که شانس عبور از دیواره ی آتش را دارند عبارت از:

Tcp 80	برای وب
Tcp 25	برای smtp
Tcp 20	برای ftp کانال داده

Tcp 21 برای ftp کانال فرمان

udp 53 برای dns

نکتهٔ مهم: یک نفوذگر دوست ندارد آدرس IP او فاش شود و از طرفی برای پویش پورت های باز باید حتما از آن استفاده کند تا قادر باشد پاسخ های بسته های ارسالی را دریافت کند.

برای حل این مشکل نفوذگر هر بار که می خواهد بسته ای برای یک پویش پورت ارسال کند چند بسته اضافی را با آدرس هایIP را جعل نیز ارسال می کند به این ترتیب حتی اگر این وقایع در فایلی ثبت شوند مدیر امنیت با مجموعه ای از آدرس های IP مواجه می شود که فقط یکی از آن ها متعلق به نفوذگر است. به عنوان مثال فرض کنید یک نفوذگر به ازای هر بسته یTCP برای پویش پورت که ارسال می کند 50 بسته ی با آدرس IP مبدا جعلی نیز ارسال کند در این صورت در فایل ثبت وقایع 51 آدرس ثبت می شود که فقط یکی از آن ها نفوذگر است. دقت کنید که پاسخ هر 51 بسته ارسال می شود.

راه مقابله با ج) پویش پورت های باز

- بهترین روش این است که پورت های باز ماشین های شبکه را که بلا استفاده اند بسته شوند.
- همچنین می توان پورت های ماشین ها توسط خودمان از بیرون پویش داده شوند.
- از دیواره های آتش stateful استفاده کنید. به این ترتیب خیلی از حمله های یورش پورت از کار خواهند افتاد. رجوع شود به مبحث fire wall

د) تعیین سیستم عامل ماشین هدف با TCP stack finger

نفوذگر تمایل دارد که بداند سیستم عامل نصب شده بر روی ماشین هدف چه نوعی است. به این ترتیب می تواند حملات خود را شایسته تر انجام دهد.

در پروتکل TCP/ IP تمام اتفاقات متعارف تعریف و مشخص شده است اما برای اتفاقات نا معلوم TCP/ IP استاندارد هماهنگی ندارد و هر سیستم عامل روش خاص به کار می برد. به این ترتیب می توان نوع سیستم عامل را از راه دور حدس زد.

مثلا در win وقتی به یک پورت بسته یک Ack ناگهانی ارسال شود یک RST برگشت داده می شود اما در unix خیر.

تا اینجای کار مهاجم اطلاعات زیر را در مورد شبکه هدف بدست آورده است:

1. فهرست ماشینهای فعال در شبکه، مثلا با استفاده از Ping
2. ساختار و توپولوژی شبکه، مثلا با استفاده از Traceroute
3. فهرست پورتهای باز بر روی ماشین‌های فعال، مثلا با استفاده از NMap.
4. نوع سیتم عامل هر ماشین.
5. فهرست پورتهای باز و بسته firewall، مثلا با استفاده Firewalk

نکته: اینکه مهاجم بداند یک پورت بر روی یک ماشین باز است یا بسته، به خودی خود هیچ سودی برای او ندارد، بلکه مهاجم باید یک نقطه ضعف در برنامه‌ای که به آن پورت گوش می‌دهد، پیدا کند.

ابزار پویشگر نقاط آسیب پذیر (Vulnerability Scanner)

نفوذگر برای شناسایی نقاط آسیب پذیر از ابزارهای پویشگر نقاط آسیب پذیر استفاده می کند. ایده کلی این ابزار بصورت زیر است:

به یک سیستم هدف متصل شده و با آن ارتباط برقرار می کنند و در طی این ارتباط متوجه نقاط آسیب پذیر آن می شوند. معمولا جهت این کار، برای آن پروسه داده های نامتعارف ارسال می کنند تا نقاط آسیب پذیر آن پروسه را در مواجهه با این داده ها کشف کنند.

ابزار پویشگر نقاط آسیب پذیر معمولا دنبال ضعف هایی مانند ضعف های زیر هستند:

1. ضعف در پیکربندی پیش فرض یک نرم افزار. بعنوان مثال بسیاری از برنامه ها از User و Password پیش فرض استفاده می کنند.
2. خطا در پیکربندی یک نرم افزار.
3. ضعف ها و آسیب پذیری های شناخته شده و مشهور.

برخی از نرم افزارها دارای bugهای ذاتی هستند که پس از عرضه و استفاده گسترده از آن، کشف می شوند. تولیدکنندگان نرم افزار سعی می کنند با ارائه Patchهایی این نقاط ضعف را برطرف کنند، اما برخی از کاربران بنابر دلایلی این Patchها را نصب نمی کنند.

از ابزارهای پویشگر نقاط آسیب پذیری ها می توان به Nesus، Retina و SAINT اشاره کرد.

Service Pack: معمولا مجموعه ای از Patchها جمع اوری شده و تحت عنوان service pack ارائه می شود.

مقابله با سوء استفاده از نقاط آسیب پذیر:

1. باید بطور مداوم اشکالات و bugهای نرم افزارهای مورد استفاده را اصلاح کرد و Patchها را نصب کرد.
2. استفاده از ابزارهای پویشگر نقاط آسیب پذیر برای شناخت ضعف های سیستم خود و برطرف کردن آنها.

گام سوم: نفوذ و راهیابی به سیستم

در مراحل قبل، نفوذگر مشخصات و اطلاعات جامعی درباره سیستم هدف بدست آورده است. حال یک سوال پیش می آید که اگر نقاط آسیب پذیر یک سیستم زیاد باشد از کجا باید حمله را شروع کنیم؟ از نقاط آسیب پذیر برنامه کاربردی یا نقاط آسیب پذیر سیستم عامل یا نقاط آسیب پذیر لایه شبکه؟

پاسخ این پرسش تا حدود زیادی به اهداف حمله بستگی دارد. مثلا برای حملات DOS حمله به سیستم عامل کارآمد است ولی برای سوء استفاده از حساب کاربران، حمله به برنامه های کاربردی موثر خواهد بود.

حمله علیه کلمات عبور (Password Attack):

همانطور که قبلا گفته شد، استفاده از کلمه عبور یکی از راه های تصدیق اصالت می باشد. در بسیاری از فقط کلمات عبور هستند که از داده های محرمانه و حساس حفاظت می کنند و فاش شدن یک کلمه عبور، گاهی ممکن است باعث شکسته شدن حریم امنیتی کل سیستم شود.

در اکثر سیستم ها برای راحتی کاربران به آنها اجازه داده می شود تا خودش یک کلمه عبور مناسب انتخاب کند، لذا کاربر معمولا برای راحتی کار خود کلمه ی عبوری انتخاب می کند که به خاطر سپردن آن راحت باشد. مثلا نام خود، شماره شناسنامه، تاریخ تولد، شماره تلفن و... غیره، که همین مساله معمولا در حملات نفوذگران علیه کلمات عبور مورد استفاده قرار می گیرد.

معمول ترین روش های حمله به کلمه های عبور سیستم ها عبارتند از:

1. **حمله به کلمه عبور پیش فرض سیستم ها:**

بسیاری از سیستم عامل ها، سرویس دهنده ها و مسیریاب ها، تعدادی کلمه عبور دارند که توسط سازنده تعریف شده اند و به کاربران اجازه می دهد برای اولین بار سیستم را نصب و پیکربندی کنند. البته پس از پیکربندی و کارهای اولیه این passwordها باید عوض شوند، اما یک مسئول مشغول و یا بی اطلاع یا تنبل ممکن است این کلمه عبور پیش فرض را تغییر ندهد، بنابراین نفوذگر می تواند با اطلاع از این کلمات پیش فرض آنها را امتحان کند. در آدرس Security.nerdnet.com می توانید بسیاری از این کلمات عبور را پیدا کنید.

2. **حدس زدن کلمه عبور با استفاده از آزمون و خطا:**

در این روش نفوذگر با استفاده از یک برنامه کوچک که بطور مکرر کلمات عبور مختلف را جهت ورود به سیستم امتحان می کند. این برنامه ها معمولا یک فرهنگ لغت غنی در اختیار دارند که کلمات معنی دار را از آن انتخاب می کنند. اگر کلمات با معنی نتیجه بخش نبود، شروع به ترکیب کلمات می کند (مانند Alex22) و در نهایت کاراکترهای تشکیل دهنده کلمه عبور را بطور تصادفی انتخاب می کند و تمام حالتهای ممکن را امتحان می کند (Brute force) که البته استفاده از ای روش آخر به زمان بسیار زیاد و پهنای باند بالا احتیاج دارد

روشهای مقابله با حدس زدن کلمه عبور با استفاده از آزمون و خطا:

1. عدم استفاده از کلمات کوتاه و با معنی. بدین ترتیب نفوذگر به زمان زیادی نیاز دارد.

2. در برخی از سیستم ها، هرگاه تعداد دفعات تلاش ناموفق برای ورود به سیستم از تعداد مشخصی تجاوز کرد، حساب مربوطه غیرفعال می شود. بدین ترتیب نفوذگر نمی تواند بعنوان مثال بیش از 3 یا 5 کلمه را امتحان کند.

نکته: البته با این کار بطور ناخواسته مقدمات یک حمله از نوع DOS را فراهم کرده ایم، به این ترتیب که نفوذگر می تواند حسابهای کاربری استفاده کنندگان مجاز شبکه را غیرفعال کند و در نتیجه افراد مجاز هم دیگر نمی توانند به شبکه وارد شوند.

3. در برخی از سیستم ها هرگاه یک کلمه عبور اشتباه باشد، بطور عمدی دریافت کلمه عبور بعدی به تاخیر خواهد افتاد، به این ترتیب امتحان کردن کلمه‌های عبور مختلف زمان بسیار زیادی نیاز دارد و عملا غیر ممکن است.

4. با استفاده از یک IDS می توان این نوع حمله ها را کشف کرد. بدین ترتیب که هرگاه تلاش های یک فرد برای ورود به سیستم از تعداد دفعات مشخصی فراتر رفت، گزارش داده شود.

3. شکستن کلمات عبور (Password Cracking) به روش علمی

حدس زدن کلمه عبور با استفاده از آزمون و خطا احتمال موفقیت کمی دارد، به همین خاطر از روشهای دیگری موسوم به **Password Cracking** (شکستن کلمه عبور) استفاده می شود.

مکانیزم های شکستن کلمات عبور بر دو اصل استوار است:

- **در همه سیستم ها، کلمات عبور در جایی ذخیره و نگهداری می شود.**
- **در بسیاری از سیستم ها، کلمات عبور بر روی خطوط ارسال می شوند.**

همه طراحان سیستم با این مساله مواجه هستند که حتما کلمات عبور باید در جایی ذخیره شوند تا بتوان با جستجوی در میان آنها، هویت کاربر را احراز کرد. البته در عمل کلمات عبور بصورت رمز شده یا Hash شده در فایلی نگهداری می شوند، اما همین فایل رمز شده نیز برای نفوذگر بسیار کارآمد است.

در ابتدا نفوذگر باید فایل حاوی کلمات عبور رمز شده را بدست آورد. بعنوان مثال در Windows کلمات عبور در پایگاه داده SAM (Security Account Manager) نگهداری می شوند (این فایل در XP در مسیر %windows%\system32\security\sam قرار دارد).

برای بدست آوردن این پایگاه داده روشهای مختلفی وجود دارد:

- از Registery ویندوز
- از کپی پشتیبان در مسیرsystm\repair
- از دیسک های پشتیبان برای بازیابی
- بسته های Challenge/Response روی شبکه

روش کار به این صورت است که ابتدا یک کلمه عبور به صورت حدسی تولید می شود، سپس طبق الگوریتم hash رمز می شود، حال این متن رمز شده در فایل رمز جستجو می شود، اگر پیدا شد کار تمام است در غیر این صورت یک کلمه جدید امتحان می شود .

نکته: در این روش تولید و حدس زدن کلماتی که امتحان می شوند بسیار مهم است . به عنوان مثال می توان از یک فرهنگ لغت شروع کرد. اما کافی نیست زیرا اکثر کاربران کلمات با معنی انتخاب نمی کنند .پس می توان کلمات موجود در فرهنگ لغت را با اضافه کردن پسوند ها و پیشوند هایی امتحان کرد . اما در نهایت می توان از روش Brute Force استفاده کرد، به این ترتیب که تمام ترکیب های مختلف را امتحان کنیم که البته بسیار بسیار زمانگیر است.

حسن روش سوم، شکستن کلمات عبور :

تمام محاسبات می تواند در کامپیوتر نفوذگر انجام شود و در نتیجه توان محاسباتی بالایی در اختیار است بدون تاخیر در شبکه.

در ضمن محدودیت هایی نظیرمحدودیت تعداد تلاشهایی ناموفق و یا تاخیر عمدی بین دو تلاش ناموفق وجود ندارد.

Lopht crack برای win (www.lophe.com/lophtcrack/)

راه های مقابله با Passwords cracker ها:

1- به کاربران اجازه ندهیم کلمات عبور معنی دار(می توان قبل از قبول کلمه عبور در یک Dictionary جستجو کنیم) و کوتاه انتخاب کنند .

2- در نظر گرفتن طول عمر برای Password ها

طول عمر خیلی کوتاه نباشد زیرا کاربر کلمات با معنی استفاده می کند تا بتواند به خاطر بسپارد یا جایی می نویسد.

3- استفاده از password generator و یعنی کلمه عبور را کاربر انتخاب نکند و سیستم خود یک کلمه تولید کند و در اختیار کاربر بگذارد.

4- احراز هویت بدون کلمه عبور معمولی مثلا smart card

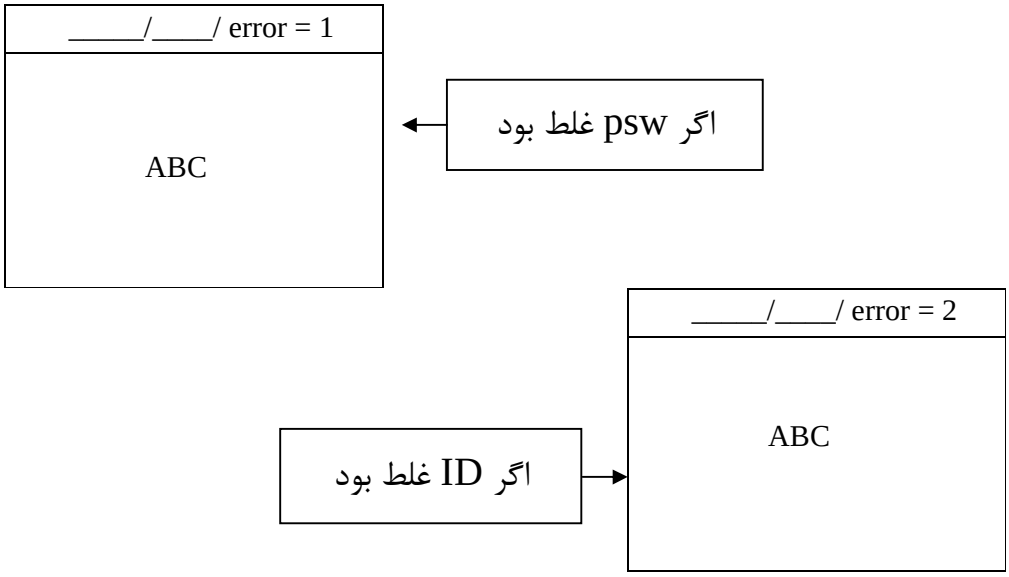
5- محافظت از فایل حاوی کلمات عبور رمز شده .

حمله به وب به روش درو کردن حساب کاربری: Account Harvesting

در این روش نفوذگر تلاش می کند مشخصه های کاربری(User ID) و کلمات عبور کاربران وب را ربوده و از آنها استفاده کند و برای انجام این کار از ضعف در برنامه های کاربری وب استفاده می کند . دقت کنید در این حالت نفوذگر حتی User ID را نیز در اختیار ندارد و باید آن را هم در صورت نیاز بدست آورد.

یک نفوذگر برای بدست آوردنUser ID های مربوط به یک برنامه کاربردی وب از ضعف در برنامه نویسی طراح آن استفاده می کند.

به این ترتیب که برخی از سایت ها هنگامی که افراد قصد ورود به سایت را داشته باشند ، برای هرکدام پیغام متفاوتی ظاهر می کنند. حتی اگر ظاهر صفحه نیز یکسان با شد باز هم ممکن است رد پایی از این مسئله وجود داشته باشد.



به این ترتیب نفوذگر می تواند یک برنامه کوچک بنویسد تا به جای مرورگر با برنامه وب سرویس دهنده ارتباط بر قرار کند و با استفاده از یک فرهنگ لغت، ID های مختلف را کشف کند به این ترتیب که کلمات مختلف را با یک psw امتحان می کند و بر اساس پیغام خطای رسیده به درو کردن User ID ها می پردازد.

پس از شناساییID های معتبر، شروع به کشف Password ها می کند.

برای مقابله با این مشکل باید به عنوان برنامه نویس وب، باید مراقب بود پیغام های خطا هیچ نماد روشن و با معنی در خود نداشته باشند.

حمله به وب به روش تعقیب نشست وب: Web session tracking

مقدمه و یاد آوری :

اصطلاحا وب State less است. یعنی هنگامی که یک صفحه برای شما ارسال شد سرویس دهنده همه چیز را فراموش می کند و دیگر شما را به جا نمی آورد این مسئله یک مشکل بزرگ است. به عنوان مثال سناریوی زیر را در نظر بگیرید شما برای چک کردن E-mail های خود، به Yahoo مراجعه می کنید و پس از وارد کردن User ID و Psw خود، وارد Mail box خود می شوید. از قضا تعداد زیادی Mail در Box شما وجود دارد و Yahoo این ایمیل ها را ده تا ده تا برای شما ارسال می کند. شما پس از بررسی اجمالی ده ایمیل اول، علاقه دارید بقیه را نیز مشاهده کنید لذا کلید Next یا Continue را می زنید حال این سوال پیش می آید، ، با توجه به State less بودن وب، یاهو از کجا بداند شما چه کسی هستید؟ زیرا شما چند دقیقه پیش User ID وPsw را وارد کرده اید. به علاوه مثلا یاهو ازکجا بداند چند ایمیل را برای شما نمایش داده است و باید از کجا ادامه دهد ؟ برای رفع این مشکلات از مفهوم Session ID استفاده می کنند. در اکثر برنامه های کاربردی وب ،پس از آنکه کاربر ID و Psw خود را وارد کرد و هویت آن محرز شد، برنامه کاربردی یک مشخصه منحصر به فرد و غیر تکرار به نام Session ID برای وی تولید می کند تا از آن به بعد عملیات کاربر با آن مشخصه دنبال شود. به عبارت دیگر برنامه تحت وب با استفاده ازSession ID به مرورگر کاربر اعلام می کند هویت شما مورد تایید است لطفا این کد(Session ID) را گرفته و هرگاه خواستید کاری انجام دهید آن را به من بر گردانید، تا اولا بتوانم شما را بشناسم، ثانیا درخواست های قدیم شمارا بدانم و درخواست جدیدتان را دنبال کنم.

✧ نکته: برنامه کاربردی وب به سه روش Session ID راپیاده سازی می کند:

1- ارسال Session ID به عنوان بخشی از URL

2- تعریف عناصر مخفی درون صفحه وب

3- استفاده از کوکی Cookie

حال نکته مهم این است که یک نفوذگر ممکن است یک نشست با یک برنامه کاربردی برقرار کرده و یک Session ID معتبر دریافت کند، پس آن را عوض کرده و خودش را در قالب یک کار بر دیگر جا بزند و از آن به بعد عملیات او را ادامه دهد دقت کنید این گونه حمله ها اگر سنجیده باشند چقدر مهلک هستند به عنوان مثال کار بران یک بانک را در نظر بگیرید (S.ID معتبر مهم است)

مکانیزم حمله در این حالت به صورت زیر است:

1- نفوذگر باید با برنامه کاربری ارتباط بر قرار کرده و یک نشست مجاز بر قرار کند پس مقدار Session ID اختصاص داده به خودش را مشاهده می کند. (طریقه مشاهده SESSION ID در ادامه تشریح می شود)

در این مرحله نفوذگر بررسی میکند که ساختار و آرایش SESSION ID چگونه است. مثلا چند کاراکتر است، چه نوع کاراکتر هایی دارد و ...

2-پس نفوذگر یک برنامه کوچک می نویسد و چندین بار با ID و Psw خودش به آن برنامه کاربردی وارد می شود و یک SESSION ID جدید دریافت می کند و این SESSION ID ها را در جایی ذخیره می کند.

3- در این مرحله نفوذگر SESSION ID های جمع آوری شده را مورد تحلیل های آماری قرار می دهد تا فرمول تولید آن را بیابد، شاید بتواند SESSION ID یک کار بر دیگر را حدس بزند.

4- در نهایت با حساب شخصی خود به آن بر نامه وارد شده و پس از در یافت یک SESSION ID آن را با مقدار متعلق به یک کاربر دیگر عوض کرده و در کنار وی سرویس می گیرد.

روش های بدست آوردن و استخراج SESSION ID خود:

همانطور که قبلا ذکر شد برنامه های کاربردی وب ، به سه روش SESSION ID را پیاده سازی می کند:

1- **ارسال Session ID به عنوان بخشی از URL**:

در این حالت کار وی بسیار ساده است و فقط باید به URL مراجعه کرده و فیلد مربوط به SESSION ID را تشخیص دهد.

مثال :

http://www.abc/abc.aspy.session ID= 1234
http://www.abc/abc.aspy.ssD= 1234
http://www.abc/abc.aspy.bob=1234

نام متغیر SESSION ID دست برنامه نویس است و می تواند متفاوت باشد.

2- **ارسال SESSION ID به صورت تعریف عناصر مخفی درون صفحه وب**:

در این حالت نفوذگر باید کد Html صفحه وب مربوط را مشاهده کرده و دنبال فیلد های Hidden بگردد

مثل :

<input type =”hidden” name =”session” value = “1234”>

3- **استفاده از کوکی**

در حالت کلی 2 نوع کوکی داریم:

1- کوکی دائمی

2- کوکی موقت

کوکی دائمی به صورت متنی در یک فایل ذخیره می شود که این فایل با هر ویرایشگری قابل رویت است. مثلا" هرگاه نمی خواهیمusername و password را همیشه وارد کنیم remember me را تیک می زنیم و آن صفحه در دفعات بعدی، اطلاعات ما را نگهداری می کند. این از کوکی دائمی استفاده می کند. به این ترتیب نفوذگر به راحتی فایل کوکی را با یک ویرایشگر ساده باز کرده و مقدار آن را تغییر می دهد . در IE کوکی ها به تفکیک هر سایت و با اسامی مختلف در یک شاخه به نام cookies ذخیره میشود. در Netscape همه در یک فایل به نام coockies.txt ذخیره می شوند.

اما کوکی موقت در حافظهء مرورگر بر روی RAM (نه دیسک) ذخیره می شوند و با بسته شدن مرورگر از بین می روند (بر خلاف کوکی های دائم). در این حالت تغییر Session ID به راحتی امکان پذیر نیست .برای این کار می توان از ابزار های موجود استفاده کرد .مانند Achilles که می تواند هر گونه Session ID را تغییر دهد. Achilles در واقع به صورت یک proxy server کوچک پیاده سازی شده است. نفوذگر تنظیمات proxy را set می کند و این برنامه بعنوان واسطی بین مرورگر و دنیای اینترنت قرار می گیرد.

مقابله با حملهٔ Session Tracking

1- اطلاعات مربوط به Session ID را باید قبل از ارسال رمز کرد.
2- Session ID به قدر کافی طولانی باشد.
3- روال تولید Session ID باید مناسب و حتی الامکان پویا باشد.
4- در نهایت میتوان Achilles را علیه سایت خود امتحان کرد.

نکته: پس از اتمام کار با یک برنامهٔ کاربردی وب باید از آن log out کرد. تا Session ID تخصیص داده شده نامعتبر شود. در غیر اینصورت Session ID معتبر باقی می ماند. البته طراح سایت می تواند یک طول عمر برای Session ID در نظرگیرد.

حمله به وب به روش SQL Piggybacking

در اکثر برنامه های کاربردی وب از بانک اطلاعاتی استفاده می شود. فارغ از نوع بانک اطلاعاتی آنها ، این برنامه ها ورودی را از کاربر دریافت کرده و بر اساس آنها یک Query از بانک خود می گیرند. اغلب این بانک ها برای فعل و انفعال با بانک اطلاعاتی از زبان SQL استفاده می کنند .روش کار به این صورت است که برنامه، محتویات یک فیلد را که کاربر وارد کرده است را در جلوی یک دستور SQL چسبانده و به بانک ارسال می کند. حال یک نفوذگر حرفه ای این فیلدها را به گونه ای پر می کند که این Query به یک فرمان مهلک تبدیل شود.

نکته: در زبان SQL کاراکتر هائی نظیر ' یا '' یا , یا ; عملکرد های خاصی دارند. وجود هر یک از این علامات در یک Query ممکن است یک عملکرد غیر طبیعی به دنبال داشته باشد. نفوذگر می تواند با مقداردهی فیلد ها با کاراکترهای خاص و مقادیر مشخص و ارسال آن به سمت برنامهٔ کاربردی نقاط ضعف برنامه را کشف کرده و با استفاده از فرامین قابل ارسال SQL با بانک اطلاعاتی آنگونه که می خواهد فعل و انفعال کند.

مقابله با SQL Piggybacking

1- به هیچ وجه نباید از داده های ارسالی توسط کاربر مستقیم Query گرفت. و قبل از انجام هر کاری باید ابتدا داده ورودی کاربر بررسی شود. داده را باید pars کرد . مثلا"first name نباید عدد داشته باشد، سن نباید کاراکتر غیرعددی داشته باشد.حتی الامکان از combo یا شبیه آن استفاده کنیم.

مثال از کاربرPassword وID گرفته می شود.

```
Select * From UsersTable Where (ID=''A'' and PSW=''P'')
```

حال نفوذگر بجای نام کاربری 'B = ID or 'P را وارد می کند، بنابراین Query تبدیل می شود به:

Select * from UsersTable where (ID=” A” and psw =”P” or 1=1)

که همواره درست است

حمله های در سطح لایهٔ شبکه:

حمله در سطح لایهء شبکه گاهی مخرب تر و خطرناک تر از حملات در لایه های بالاتر است . زیرا در این حالت نفوذگر آزادی عمل بسیار بیشتری دارد.

استراق سمع در لایهٔ شبکه :

یک**Sniffer** برنامه ای است که ترافیک جاری بر روی شبکه را جمع آوری و استراق سمع کرده و بخشهای مفید آن را در اختیار نفوذگر قرار می دهد.Sniffer ها هیچ گونه تغییری در اطلاعات ایجاد نمی کنند و عملیات آنها لذا به سادگی قابل کشف نیست.

در شرایط معمولی کارت های شبکه فقط فریم هایی را از روی کانال دریافت می کنند که فیلد آدرس مقصد در آن فریم، با آدرس کارت شبکه یکسان باشد. اما برخی از کارت های شبکه از کارت های شبکه از حالتی پشتیبانی می کنند به نام promiscuous mode یا حالت بی قید . در این حالت بدون توجه به فیلد آدرس مقصد هر فریم، آن را می پذیرد.

نرم افزارهای Sniffer ابتدا کارت شبکه را در حالت بی قید تنظیم می کنند و بدین نحو تمام ترافیک جاری بر روی کانال را دریافت می کنند، سپس می توانند از بین اطلاعات دریافتی مورد نظر نفوذگر را گلچین کنند

استراق سمع از Hub (Passive Sniffing):

در بسیاری از شبکه های محلی مبتنی بر اترنت، به دلیل قیمت مناسب از هاب استفاده می شود. دقت کنید که یک Hub تمام اطلاعات دریافتی را به صورت broadcast بر روی تمام خروجی ها ارسال می کند. در چنین ساختاری هرگاه یکی از ماشین های متصل به hub، یک sniffer را اجرا کند به راحتی قادر است کل فریم های ارسالی همهٔ ماشین ها را دریافت کند.

استراق سمع از سوئیچ (Active sniffing):

یک سوئیچ بر خلاف یک هاب ، پس از دریافت یک فریم، با توجه به آدرسMAC مقصد، آن را فقط به سمت مقصد ارسال میکند. به این طریق یک ماشین متصل به سوئیچ قادر نیست ترافیک دیگر ایستگاه ها را sniff کند. به این ترتیب نرم افزار های passive sniffer قادر نیستند بقیهٔ فریم های شبکه را در اختیار نفوذگر قرار دهند. البته برای شنود از switch نیز روشهایی وجود دارد:

روش اول: **ارسال سیل آسای فریم ها و از کار انداختن سوئیچ ها**:

سوئیچ ها تا وقتی که از طریق کابل در یک کارت شبکه متصل نشوند هیچ اطلاعی از آدرس سخت افزاری کارت های شبکه ندارند. اما در هر سوئیچ مقداری حافظه وجود دارد تا با دریافت یک فریم از روی یک کانال، آدرس مبداء آن فریم را یاد گیرد.

در این روش نرم افزار sniffer تعداد زیادی فریم با آدرس مبداء تصادفی و بی ربط برای سوئیچ ارسال می کند و سوئیچ نیز تمایل دارد این آدرس های جدید را نیز فرا گیرد و به این ترتیب حافظهء سوئیچ پر خواهد شد.

برخی از سوئیچ ها به محض پر شدن حافظه، سریعاً دست ها را بالا برده و همانند هاب عمل می کنند که در اینصورت sniffer به مقصود خود می رسد اما برخی دیگر از سوئیچ ها شروع به بازنویسی اطلاعات جدید بر روی اطلاعات قدیمی می کنند که در اینصورت نیز برای آدرس های حذف شده ارسال شود، از آنجا که سوئیچ آدرس مقصد را نمی داند آن را برای همه ارسال می کند. این روش به خوبی در نرم افزار DSniff پیاده سازی شده است.

مقابله با استراق سمع:

1- همهء ارتباط را حتی الامکان به صورت رمز شده انجام داد . مثلاً Http

2- در ضمن دقت کرد که استراق سمع از سوئیچ بسیار سخت تر از هاب است و باید حتی الامکان از سوئیچ استفاده کرد.

استراق سمع بسته ها از طریق Arpspoofing

یادآوری 1:

در شرایط عادی هنگامی که یک ایستگاه بخواهد یک بسته را برای یک ماشین که در شبکه داخل وجود ندارد (او آدرس را نمی داند)، ارسال کند، آن بسته را باید تحویل یک مسیریاب پیش فرض (که با دنیای خارجی در ارتباط است) بدهد. به این ترتیب هر ماشین باید آدرس یک مسیریاب پیش فرض در شبکه را بداند.

یادآوری 2:

پروتکل Arp: این پروتکل در واقع یک نگاشت بین آدرس های IP و آدرس های MAC برقرار می کند.

تکنیک Arp Spoofing

در این حالت ماشین نفوذگر یک بسته پاسخ Arp، ارسال می کند و آدرس فیزیکی خود را به عنوان آدرس فیزیکی دروازه پیش فرض اعلام می کند و ماشین فرمان آدرس جدیدی را به جای آدرس قبل درج می کند. به این ترتیب تمام بسته های IP که باید از شبکه خارج شوند، ابتدا تحویل نفوذگر می شوند.

دقت کنید ماشین نفوذگر در این حالت باید عمل IP forwarding را نیز انجام دهد و بسته هایی را که به عنوان دروازه پیش فرض تحویل وی می شوند را به سمت دروازه واقعی هدایت کند تا قربانیان متوجه این استراق سمع نشوند. نرم افراز DSniff این تکنیک را به خوبی پیاده می کند.

مقابله با Arp Spoofing

می توان جدول ARP را از حالت پویا در آورده و آن را به صورت دستی تنظیم کرد. این کار برای شبکه های بزرگ بسیار دشوار است.

حمله DNS Spooring

یادآوری: سیستم DNS آدرس های نمادین حوزه را به آدرس های IP نگاشت می کند. هنگامی که کاربر آدرس IP متناظر با یک نام حوزه را نیاز دارد و یک تقاضا به سرویس دهنده محلی DNS ارسال می کند.

مراحل انجام این حمله به صورت زیر است:

ابتدا نفوذگر باید منتظر یک درخواست DNS بماند. برای دریافت این درخواست اگر شبکه مبتنی بر هاب باشد که هیچ کاری نیاز نیست اما اگر شبکه مبتنی بر سوئیچ باشد ابتدا نفوذگر باید با تکنیکی مشابه Arp spoofing خود را به جای سرویس دهنده محلی DNS جا بزند. در این حالت ماشین قربانی سعی می کند آدرس IP ی مربوط به یک نام www.example.com را به دست آورد بنابراین یک بسته پرس و جوی DNS به سمت سرور محلی DNS برروی شبکه ارسال می کند. در این حالت نفوذگر این درخواست را دریافت کرده و IP خود را به جای IP مورد نظر ارسال می کند. به این ترتیب مرور گر قربانی از این به بعد یک ارتباط TCP با ماشین نفوذگر برقرار کرده و تقاضای صفحه اصلی سایت را دارد که در این حالت مرورگر می تواند صفحات جعلی خود را برای کاربر ارسال کند و اطلاعات مهمی از وی دریافت کند مثلا ID و Password.

IP Spoofing

در بسیاری از حملات آدرس IP ماشین مبدا به صورت جعلی درج می شود که به این تکنیک IP Spoofing گویند. این کار می تواند دلایل متعددی برای نفوذگر داشته باشد. مثلا اینکه با این کار امکان تعقیب و کشف ماشین وی در طرف مقابل گرفته شود و یا می توان از فیلترهایی که به آدرس IP حساسیت دارند عبور کرد و ...

✧ **نکته مهم**: دقت کنید که ارسال بسته های با آدرس IP جعلی هیچ پاسخی را به سمت نفوذگر به دنبال ندارد و همه پاسخ ها برای ماشینی که IP آن توسط نفوذگر استفاده شده است، می روند.

✧ **نکته**: تکنیک IP Spoofing بیشتر به 2 منظور انجام می گیرد:

1- فریب یک ماشین برای اخلال در عملکرد عادی آن. (مانند DNSspoofing, ARPspoofing)

2- حملات از نوع بمباران جهت تخریب و اختلال در کار یک سرور.

حمله از طریق IP Spoofing و Source Routing

یادآوری: در پروتکل IP و با استفاده از بخش option، می توان در سرآیند بسته های IP، کل یک مسیر یا بخشی از آن که بسته باید طی کند را مشخص کرد و مسیریابی که این بسته را دریافت می کند، آن را بر اساس همان مسیر موجود در سرآیند آن هدایت می کند.

مثال از این نوع حمله:

فرض کنید نفوذگر بسته TCP SYN به سمت ماشین هدف ارسال می کند که در آن آدرس مبدا بسته، به دروغ آدرس ماشین دیگری است(مثلا دوست قربانی). پس در سرآیند این بسته، مسیری را قرار می دهد که از ماشین وی می گذرد، به عبارت دیگر نفوذگر آدرس خود را به عنوان یک مسیریاب در سرآیند بسته قرار می دهد.

ماشین قربانی با دیدن آدرس IP دوست خود به عنوان مبدا، SYN-ACK برای وی ارسال می کند و این بسته با توجه به آدرس موجود در سرآیند، از ماشین نفوذگر می گذرد و نفوذگر به راحتی مرحله سوم برقراری ارتباط TCP را کامل می کند. به این ترتیب نفوذگر به راحتی خودش را به جای دیگری جا زده و یک ارتباط TCP با قربانی برقرار می کند.

راه مقابله با این حمله:

امروزه به خاطر این نوع حملات و همچنین دلایل دیگر، اکثر مسیریاب ها اهمیتی به Source Routing نمی دهند و با گرفتن هر بسته ای، کار خود را انجام می دهند!

✧ **نکته**: تعدادی از حملات متکی به IP Spoofing در مبحث D.O.S بررسی می شوند.

حملات D.O.S (Denial Of Service)

هدف بسیاری از حملات، ایجاد اختلال و وقفه در سرویس دهی یک ماشین در شبکه است. به این نوع حملات Denial Of Service گویند. حملات D.O.S معمولا هزینه بالایی دارند. به عنوان مثال اغلب به پهنای باند بالایی نیاز دارند.

در حالت کلی حملات D.O.S به دو دسته تقسیم می شوند:

1- حملاتی که منجر به توقف کامل یک سرویس شوند.

2- حملاتی که منابع یک سرور را اشباع و تلف می کند.

در حملات نوع اول مستقیما به پروسه ی سرویس دهنده حمله می شود و باعث از کار افتادن آن پروسه می شود. اما در حملات نوع دوم منابع یک سرویس دهنده مانند حافظه وCPU به گونه ای تلف می شود که حتی با وجود فعال بودن سرویس دهنده، کاربران اصلی امکان سرویس گرفتن از آن را نداشته باشند.

اختلال در کار سرویس دهنده از درون

حمله از درون به این معناست که نفوذگر می تواند یک حق دسترسی مجاز بر روی یک ماشین شبکه فراهم کند. البته برای اینکار لزوما نیازی به حضور فیزیکی نیست.

یادآوری: برای بدست آوردن چنین مجوزی میتوان از مراحل قبلی استفاده کرد، مانند حمله به password یا Sniffing یا هر روش دیگری.

در نهایت نفوذگر به این ترتیب می تواند حداقل کارهایی را انجام دهد.به عنوان مثال اگر با مجوز Admin یا root به یک سیستم وارد شود، می تواند بطور مستقیم به یک پروسه ی در حال اجرا پایان دهد. یا مثلا تنظیمات را به گونه ای تغییر دهد که کاربران قادر به دسترسی به سیستم نباشند.حتی اگر نفوذگر مجوز سطح بالا در اختیار نداشته باشد با مجوز کاربر عادی می تواند برنامه ای را نوشته و آن را اجرا کند تا سرویس دهنده را سرگرم کند و تقاضاهای بسیاری را برای آن ارسال کند.

بمباران:

اصطلاحا به ارسال پی در پی تقاضا برای یک فرایند به نحوی که نتواند به تقاضاهای واقعی پاسخ دهد، بمباران گویند.

یک راه دیگر این است که نفوذگر بدون اینکه کاری با پروسه ی سرویس دهنده داشته باشد، منابع ماشین را در اختیار بگیرد. مثلا برنامه ای بنویسد که زمان پردازنده یا حافظه یا دیسک یا پهنای باند شبکه ی زیادی معرفی می کند.

راه مقابله با این نوع حملات

- تنظیم دقیق سطوح دسترسی افراد. (مثلا کاربران نباید مجوز سطح Admin داشته باشد.)
- اختصاص مناسب منابع به افراد.

- استفاده از منابع کافی وزیاد برای سرور. در این صورت قبل از اینکه سیستم از کار بیفتد شاید بتوان حمله را کشف کرد.
- استفاده از IDS.

حملات DOS ازبیرون

بمباران با استفاده از بسته ها ی ناقص و دارای اشکال

در این حالت برای در هم شکستن یک سرویس دهنده، با بسته های مشکل دار آن را بمباران می کنند. تعدادی از این حملات به قرار زیرند:

1. **حملهٔ Land**:

یکی از مهلک ترین حمله ها است و علیهTCP انجام میشود. در این حمله تعداد زیادی بسته با مشخصات زیر به هدف ارسال می شوند:

- فیلد های پورت مبدا و پورت مقصد هردو با یک مقدار تنظیم می شوند واین مقدار یکی از پورتهای باز ماشین هدف است .
- فیلدهای آدرس IP مبدا و مقصد نیز هر دو با یک مقدار و آن هم با آدرس IP ماشین هدف تنظیم می شوند.

در این حالت ماشین مقصد هنگامی که این بسته را دریافت کرد و پاسخ آن را تولید کرد آن را ارسال می کند، اما این بسته به خود ماشین برمی گردد و این روند ادامه می یابد.

حمله از نوع Land را اگر بر روی چندین پورت باز انجام دهیم، با عنوان Latierra شناخته می شود.

2. **حملهٔ Ping of Death**:

اندازه ی بسته های Ping حداکثر 64 KByte می باشد و بسته های بزرگتر از آن غیر مجاز هستند. حال اگر به یک ماشین یک بسته ی Ping با اندازه ی بیشتر از 64 KByte ارسال شود، چون در پروسه ی ICMP هیچ تمهیدی برای آن در نظر گرفته نشده است، پروسه ی ICMP به راحتی مختل می شود و عملکرد صحیح خود را از دست می دهد.

3. **حملهٔ Jolt2**:

یادآوری: بسته های IP می توانند به قطعات کوچکتری تقسیم شوند و فیلد fragment offset در این قطعات، جایگاه قطعه ی جاری را در بسته ی اصلی نشان می دهد.

در این حمله مهاجم تعداد زیادی از این قطعات را برای هدف ارسال می کند اما در بین این قطعات هیچ قطعه ای با Fragment-Offset=0 وجود ندارد. بنابر این پروسه ی IP درمقصد برای تصمیم گیری راجع به هر بسته باید ابتدا آن را بازسازی کند و برای بازسازی آن باید همه ی قطعات را در حافظه نگهدارد، به این ترتیب تمام حافظه ای که در اختیارIP است تمام می شود و هیچ بسته ی جدیدی نمی تواند دریافت کند.

راه مقابله با این نوع حملات:

- شناسایی نقاط ضعف O.S و به روز کردن آن.
- استفاده از یک دیواره ی آتش قوی. مثلا Land از IP مبدا و مقصد یکسان استفاده می کند که توسط Firewall قابل کشف است.

4. **حمله از نوع SYN Flood**:

هدف این حمله تلف کردن منابع سیستم از راه دور می باشد.

یادآوری: یک ارتباط TCP طبق یک دست تکانی سه مرحله ای شکل می گیرد .

اولین مرحله: مشتری یک بسته یsyn=1 برای سرور ارسال می کند.

دومین مرحله: سرور یک بسته ی syn=1, Ack=1 برای مشتری ارسال می کند.

سومین مرحله: مشتری یک Ack=1 برای سرور ارسال می کند .

نکته: سرویس دهنده پس از انجام مرحله ی دوم باید مشخصات این تقاضا را در جایی ذخیره کند تا پس از مرحله ی دوم از آن استفاده کند. در واقع سرویس دهنده مقداری از فضای حافظه ی خود را صرف ذخیره ی برخی اطلاعات مربوط به این درخواست می کند و چیزی بین 40 تا 360 ثانیه این اطلاعات را نگه می دارد و اگر مرحلهٔ دوم تکمیل نشده است آنگاه این اطلاعات را دور می ریزد

در حمله ی SYN Flood نفوذگر در یک حلقه ی بی نهایت ، تعدادی بسته ی syn تولید می کند و با آدرسهای IP مبدا دروغین برای سرور ارسال می کند. و سرور به ازای هر درخواست باید مشخصات آن را در حافظه خود ذخیره کند. حال اگر نفوذگر بتواند با سرعت مناسبی این بسته را برای سرور ارسال کند، حافظه ی سرور پر می شود و عملا قادر به دریافت درخواستهای واقعی نیست.

راه مقابله با SYN flood

_ استفاده از پهنای باندها و حافظه‌ی بالا برای سرور

_ می‌توان زمان نگهداری مشخصات یک ارتباط نیمه‌تمام را کوتاه‌تر کرد.

_ در linux اصلا چنین صفی تشکیل نمی‌شود.

بلکه اطلاعات لازم به‌طور محرمانه و رمز شده در بسته‌ی Syn-Ack که سرور برای مشتری ارسال می‌کند قرار داده می‌شود و این اطلاعات طی مرحله‌ی سوم دوباره برای سرور بازفرستاده می‌شوند.

5. **حمله‌ی Smurf**

یادآوری: آدرس‌های IP شامل دو بخش آدرس شبکه و آدرس میزبان هستند. و اگر تمام بیت‌های فیلد آدرس میزبان 1 باشند، بسته باید توسط تمام ماشین‌های آن شبکه دریافت شوند.

مثال:

A 10.265.165.255
B 131.131.155.255
C 192.168.15.255

روش حمله:

نفوذ یک یک بسته‌ی ping را با آدرس مقصد فراگیر برای همه‌ی ماشین‌های شبکه ارسال می‌کند. این بسته قاعدتا توسط همه‌ی ماشین‌های شبکه دریافت می‌شود و هر کدام از آن‌ها به صورت جداگانه سعی می‌کنند پاسخ آن را ارسال کنند.

حال فرض کنید نفوذگر در فیلد آدرس مبدا بسته‌ی ping اولیه، آدرس ماشین قربانی را قرار داده در نتیجه به ناگاه ماشین قربانی با سیل بسته‌های پاسخ مواجه می‌شود.

تعداد بسته‌هایی که ماشین قربانی دریافت می‌کند برای تعداد ماشین‌های شبکه‌ی محلی است. حال اگر نفوذگر در هر ثانیه چندین بسته‌ی ping ای چنین برای شبکه ارسال کند، ماشین قربانی از کار خواهد افتاد.

مقابله با Smurf

_ مسیریاب نباید اجازه دهد کسی از بیرون شبکه، یک بسته‌ی broadcast برای داخل شبکه ارسال کند.
_ حذف بسته‌های ICMP در مسیریاب
_ غیر متعال کردن ICMP برای ماشین‌های شبکه.

6. حمله‌ی Fraggle

یادآوری: در UDP پورت شماره‌ی 7 برای echo در نظر گرفته‌شده است یعنی هر داده‌ای که برای آن ارسال شود، عینا برای مبدا برگشت داده می‌شود.

روال حمله:

نفوذگر یک بسته‌ی UDP بر روی پورت 7 و با آدرس مقصد broadcast شبکه ارسال می‌کند و آدرس مبدا را به ماشین قربانی SET می‌کند. به این ترتیب تمام ماشین‌های شبکه این بسته را دریافت کرده و همان را دوباره برای ماشین قربانی ارسال می‌کنند.

7. راه مقابله با Fraggle

_ مسیریاب نباید اجازه دهد از بیرون شبکه یک بسته‌ی broadcast برای داخل شبکه ارسال شود.
_ بستن پورت 7 مربوط به UDP بر روی ماشین‌ها

حملات D.D.O.S (Diseributed Denial of Service)

✧ **نکته**: در اغلب حملات D.O.S نفوذگر باید منابع و مخصوصا پهنای باند بالایی در اختیار داشته باشد و هزینه‌ی بالایی تقبل کند . در ضمن ممکن است به خاطر طولانی شدن حمله (مانند Syn flood) هویت نفوذگر بالاخره آشکار شود.

حملات D.D.O.S اولین بار در سال 1999 استفاده شدند و دارای خصوصیات و اهداف کلی زیر هستند:

1-هزینه‌ای برای نفوذگر ندارند.

2-هویت نفوذگر پنهان می‌ماند.

3-احتمال موفقیت آن‌ها بسیار بالاست.

4-مبارزه با آن‌ها بسیار مشکل است.

✧ **نکته**: نفوذگر در حملات D.D.O.S سعی می‌کند از ماشین‌های زیادی در سراسر اینترنت برای هدف خود استفاده کند.

تعریف: به ماشین‌هایی که بدون اطلاع صاحبش و توسط یک نفوذگر به عنوان ابزاری برای حملات D.O.S مورد سوء استفاده قرار می‌گیرند، ماشین‌های **زامبی(zombie)** می‌گویند.

روال کلی حملات D.D.O.S به این صورت است که نفوذگر با استفاده از روش‌هایی، نرم‌افزارهای zombie را بر روی ماشین‌های کاربران ناآگاه نصب می‌کنند. این نرم‌افزارهای مانند تروجان در قالب برنامه‌های رایگان و زیبا و یا هر چیز دیگری توزیع می‌شوند. به این ترتیب هرکدام از این برنامه‌ها می‌توانند بخشی از حمله‌ی D.O.S را بر عهده بگیرند. این برنامه‌ها در ماشین مورد نظر اجرا شده و منتظر صدور فرمان از جانب نفوذگر می‌باشند.

حمله‌ی TFN2K (Tribe Flood Network 2000)

در این حمله نفوذگر ماشین‌های زامبی را در غالب گروه‌هایی تقسیم‌بندی می‌کند و برای هر گروه یک سرگروه انتخاب می‌کند. به این ترتیب نفوذگر فرمان حمله را فقط به سرگروه‌ها اعلام می‌کند و سرگروه‌ها برای ماشین‌های زامبی فرمان صادر می‌کنند.

حمله‌ی TFN2K از مکانیزم‌های UDP flood یا Syn flood یا smurf یا حمله‌ی Targa (ارسال سیل‌آسای بسته‌های IP ناقص) یا ترکیبی از این حالت‌ها استفاده می‌کند.

✧ **نکته**: یکی از شگفت‌انگیزترین مسائلی که در TFN2K وجود دارد این است که این نرم‌افزار در ماشین‌های زامبی هیچ پورت جدیدی باز نمی‌کند. بنابراین صاحب ماشین زامبی نمی‌تواند با نرم‌افزارهای اسکن پورت‌ها، مشکوک شود.

✧ **نکته**: نحوه‌ی صدور فرمان حمله‌دهنده از موارد بسیار هوشمندانه‌ی TFN2K می‌باشد. نفوذگر برای ارسال دستور حمله از بسته‌های پاسخ ping (echo reply packet) استفاده می‌کند. دلیل این استفاده نیز این است که اکثر مسیریاب‌ها و دیواره‌های آتش اجازه می‌دهند این بسته وارد شبکه شود ، زیرا فرض می‌کند این بسته در پاسخ به دستور ping به شبکه بازگشته است.

✧ **نکته**: در TFN2K، حتی آدرس مبدا صدور بسته‌ی پاسخ ping نیز جعلی تنظیم می‌شود تا ماشین نفوذگر و ماشین‌های سرگروه کشف نشوند.

✧ **نکته**: در TFN2K ماشین‌های زامبی نیز هنگام حمله از آدرس‌های جعلی استفاده می‌کنند تا دیرتر کشف شوند.

✧ **نکته**: به سه دلیل پیگیری ماشین‌های زامبی هیچ سودی ندارد:

1- تعداد آن‌ها بسیار زیاد است.

2- در سرتاسر دنیا پراکنده‌اند و به ISP های متفاوتی متصل هستند. بنابراین پیدا کردن آن‌ها (در کشورهای مختلف و با قوانین مختلف) بسیار زمان‌گیر و عملا غیر ممکن است.

3- آن‌ها بی‌گناهند.

✧ **نکته**: نفوذگر می‌تواند در TFN2K حتی نسخه‌ی نرم‌افزار نصب شده در ماشین زامبی را به روز کند.

✧ **نکته**: بعد از پایان حمله، نفوذگر ماشین‌های زامبی را وادار می‌کند تا نرم‌افزار TFN2K را پاک کند. (خودکشی) و یا حتی کل اطلاعات دیسک سخت ماشین زامبی را پاک کند(انفجار قرارگاه)

راههای مقابله با D.D.O.S

- مسؤل شبکه باید ماشین های شبکه را آزمایش کند تا احیاناً به عنوان زامبی استفاده نشوند.

- باید در مسیریابها از نرم افزارهای Anti-spoof دو طرفه استفاده کرد. این نرم افزارها اجازه نمی دهند یک ماشین از داخل شبکه، با آدرس مبدأ برای بیرون شبکه داده ارسال کند. بنابراین ماشین زامبی نمی تواند عملاً در حمله مؤثر باشد.

گام چهارم حمله: سیطره بر شبکه و سیستم و تثبیت نفوذ

پس از آنکه نفوذگر به نحوی به شبکه نفوذ کرد باید نفوذ خود را حفظ کرده وسیطرهٔ خود را تداوم بخشد، یعنی راهی مهیا کند تا در دفعات بعدی بتواند به راحتی وارد سیستم شود بدون آنکه دوباره مجبور باشد مراحل حمله را مجددا طی کند. برای این کار نفوذگر از نرم افزارهای مخرب و آلوده ای استفاده می کند مانند اسبهای تروا یا Back door یا Root Kit.

برای آلوده کردن ماشین قربانی به این نرم افزارها راه های زیادی وجود دارد، مانند: محیط های Chat و ارسال ایمیل و ارسال نرم افزارهای رایگان و جذاب و یا فایل های دیگر مانند عکس،کلیپهای ویدویی و غیره.

اسب تروا:

اسب تروا یک برنامهٔ آلوده به کدهای اجرایی است که غالباً ظاهری فریبنده یا کاملاً معمولی دارد. این برنامه ها پس از اینکه توسط خود کاربر نصب شدند، می توانند هدایت ماشین قربانی را در اختیار نفوذگر قرار دهند.

در پشتی (Back door)

به هر معبر باز در نرم افزار، به طوری که کسی بتواند بدون اطلاع صاحب نرم افزار و کاربر نرم افزار، از آن عبور کرده و به داخل سیستم نفوذ کند، Back Door گفته می شود.

Back door نرم افزاری است که به نفوذگر اجازه می دهد بدون هیچ تشریفاتی (مثلا با وارد کردن User و Password) به راحتی به یک سیستم وارد شود. به عبارت دیگر می توان ورود به سیستم از طریق عادی وLogin را درب جلویی نامید.

اما با استفاده از در پشتی، هر چقدر هم که مسائل امنیتی را تقویت کنیم ، نفوذگر به راحتی از راهی دیگر به سیستم وارد می شود.

✧ **نکته**: مهاجمان حرفه ای وقتی وارد یک وارد یک سیستم شوند، ابتدا یک در پشتی ایجاد می کنند که فقط و فقط خود از آن خبر دارند، سپس همان شکافی را که خود از آن وارد شده اند را می بندند تا نفوذگر دیگری وارد آن نشود!!! حتی ممکن است برای جلوگیری از نفوذ دیگر مهاجمان، به تقویت امنیت سیستم قربانی بپردازد. از طرف دیگر مهاجمان گاهی برای ورود از در پشتی نیز اقدامات امنیتی قرار می دهند مانند (Password و ...).

نفوذگر می تواند برای نصب یک در پشتی از اسبهای تروا استفاده کند. چنین اسبهای ترواییی به2 دسته تقسیم می شوند:

- **اسبهای تروا در سطح برنامهٔ کاربردی.**
- **اسبهای تروا در سطح سیستم عامل (یا همان RootKit)**

اسب های تروا در سطح برنامه های کاربردی:

این اسب های تروا، برنامه های مستقل هستند که دقیقاً همانند یک برنامهٔ معمولی روی سیستم اجرا می شوند. اولین کاری که این برنامه ها انجام می دهند این است که سریعاً پیکر بنددی سیستم عامل (O.S.) را تغییر داده و کاری می کنند تا از آن به بعد به محض ورود کاربر به سیستم، این برنامه هم اجرا می شود. این گونه اسبهای تروا هیچ پنجره یا علامتی ندارند و حتی به صورت Invisible Proccess (پروسس مخفی) اجرا می شوند.

✧ **نکته**: در پشتی ای که به این صورت نصب می شود معمولاً یک برنامه ی سوکت است که بر روی ماشین قربانی اجرا می شود و منتظر یک ارتباط از طرف نفوذگر می ماند و پس از برقراری ارتباط فرامینی را بر روی ماشین قربانی اجرا می کند.

اسب تروا BO2K (Back Office 2000)

به عنوان یک نمونه از اسب های تروا، BO2K را بررسی می کنیم. سرویس دهنده ی این برنامه که بر روی ماشین قربانی نصب می شود فقط 100KB حجم دارد و یک برنامه ی کلاینت نیز برای نفوذگر وجود دارد که 500KB حجم دارد.

برنامه ی BO2K این امکانات را به نفوذگر می دهد:

- ایجاد و نمایش پنجره های محاوره ای(Dialog Box) بر روی ماشین قربانیکه نفوذگر با این روش می تواند اطلاعاتی مثلِID و Password هایی را از کاربر دریافت کند.
- ثبت و ارسال کلید های فشار داده شده توسط کاربر برای نفوذگر.(برخی Password ها را می توان بدست آورد)
- ارسال مشخصات سیستم قربانی برای نفوذگر. مانند: نوع و نسخه ی .O.S،حافظه، CPU ، هاردو...
- اعمال مدیریت فایل. مانند: کپی، حذف، تغییر نام، دیدن کلیه ی فضای دیسک و....
- ارسال پایگاه داده ی SAM برای نفوذگر. این فایل لیست کاربران را به همراه کلمه عبور hash شده آنها نگهداری می کند
- تغییر رجیستری ویندوز.
- عملیات بر روی پروسه ها. مانند: اجرا و یا حذف.
- ارسال تمام بسته های ورودی به ماشین قربانی برای نفوذگر.
- ارسال تصاویر صفحه نمایش ماشین قربانی برای نفوذگر.
- به کار انداختن دوربین قربانی به طور مخفی و ارسال تصاویر برای نفوذگر.

✧ **نکات**:

- اغلب اسبهای تروای این چنینی، هرگاه که ماشین قربانی روشن شد و کاربر Login کرد، یک ایمیل حاوی آدرس IP ماشین قربانی و دیگر مشخصات لازم برای نفوذگر ارسال می کند.

- اسبهای تروا معمولاً توسط برنامه هایی موسوم به Wrapper به برنامه ها و فایل های دیگر الحاق می شوند. هنگامی که این فایل برای کاربر ارسال شد ابتدا اسب تروا اجرا می شود و سپس خود برنامه بی هیچ تغییری اجرا می شود.

راههای مقابله با اسب های تروا و درهای پشتی:

1) استفاده از ویروس یابهای قوی و به‌روز.
2) تهیه نرم افزارها و فایل ها از منابع مطمئن .
3) آموزش کاربران.

Rootkit ها

1. Rootkit های معمولی

اسبهای تروا و درهای پشتی برنامه های مستقلی هستند که بر روی ماشین هدف اجرا می شوند. بدین ترتیب کشف و آشکار سازی آنها چندان سخت نیست.

Rootkit ها در عملکرد اصل مؤلفه های اصلی سیستم عامل، راه را برای ورود مهاجم باز می کنند. و به این ترتیب هیچ نیازی به اجرای برنامه ی اضافی نیست.

✧ **نکته**: از آنجا که کد Windows پنهان است، بیشتر برای سیستم های عامل خانوادهٔ Unix مثل Linux و سولاریس، Rootkit نوشته شده است. به عنوان مثال یک Rootkit در linux، ماژول احراز هویت linux را که همان login است را تغییر داده و به مهاجم اجازه می دهد با بالاترین سطح دسترسی (root) به سیستم وارد شود و هیچ نیازی به کلمه عبور مربوط به Admin ندارد. یا می توان سیستم عامل (O.S) را طوری دستکاری کرد که فضای دیسک سخت و فضای آزاد آن را اشتباه اعلام کند تا احیاناً فضای استفاده شده توسط برنامه های مخرب را اعلام نکند. (دستور du در linux فضاها را گزارش می دهد). یا می توان حتی O.S را طوری دستکاری کرد که فایل ها و شاخه ها و برنامه های اجرایی نفوذگر را نشان ندهد.(برنامه ی find در linux این کار را می کند و Ls). یا می توان کاری کرد که O.S. شماره پورتهای باز را گزارش نکند (برنامه ی Netstat). یا می توان کاری کرد که O.S. پروسه های در حال اجرا روی سیستم را نشان ندهد (برنامه ی Ps).

راه های مقابله با Rootkit ها

- مراقب مجوزهای root و Admin باشید. زیرا فقط با مجوز root می توان در مؤلفه های بنیادی O.S. تغییر داد Password مربوط به root را باید خیلی قوی انتخاب کرد.
- برخی از Rootkit های شناخته شده توسط ابزارهای جستجوی Rootkit مانند Trip wire قابل شناسایی اند.
- استفاده از ابزارهای بررسی صحت فایل مثلا از فایل های مهم Message Digest بگیریم و در نتیجه هر تغییری را می توان فهمید.
- در ضمن MD ها رو هم می شود روی CD ذخیره شوند که قابل تغییر نباشد و نباید MD ها رو هم روی همون سیستم ذخیره کرد.

✧ **نکته**: اگر احیاناً توسط نرم افزارهایی مانند Trip Wire یا به هر نحوه دیگر متوجه یک Rootkit شدید، ابتدا کامپیوتر را از شبکه جدا کنید و سپس کل سیستم را پاک کرده و O.S. را از ابتدا نصب کنید، چون احتمالاً نفوذگر آنقدر تروجان و در پستی بر روی سیستم نصب کرده است که هیچ کار دیگری مطمئن نیست.

2. Rootkit های سطح هسته ی سیستم عامل

برخی از Rootkit ها هسته ی مرکزی سیستم عامل (Kernel) را دستکاری و تحریف می کنند. به این ترتیب هیچ ابزاری قادر به کشف موضوع نخواهد بود.

✧ **نکته**: هسته ی O.S. اولین بخشی است که به حافظه load می شود و کنترل سیستم را در دست می گیرد. کلیه ی دسترسی ها به منابع از طریق هسته انجام می شود. Rootkit های معمولی در حقیقت هسته ی O.S. را فریب می دهند اما Rootkit های سطح هسته، بطور پنهانی عملیات هسته را در اختیار نفـوذگر قرار می دهند.

مهم ترین قابلیتی که Rootkit های سطح هسته ی سیستم عامل دارند تغییر مسیر اجرا (execution Redirection) می باشد. بعنوان مثال فرض کنید یک پروسه از هسته تقاضا می کند تا پروسه A را اجرا کند اما هسته به جای A، پروسه ی B را اجرا می کند که می تواند پروسه ای در خدمت نفـوذگر باشد.

به عنوان یکی دیگر از قابلیت های Rootkit های سطح هسته، می توان به مخفی نگه داشتن لیست فایل ها و پروسه های در حال اجرا و ارتباطات شبکه ای اشاره کرد که حتی root و Admin نیز از این قاعده مستثنی نیست و قادر به مشاهده و یا کنترل آنها نمی باشد. به این ترتیب هیچ علامتی از حضور نفـوذگر در سیستم وجود ندارد.

برای پیاده سازی Rootkitهای سطح هسته، در ویندوز یکسری Patch را اجرا می کنند. و در Linux معمولاً از قابلیت L.K.M. استفاده می کنند. Loadable Kernel Module اجازه می دهد هسته ی O.S. گسترش پیدا کند.

راه های مقابله با Rootkit های سطح هسته

- از آنجا که فقط با بالاترین سطح دسترسی می توان یک Rootkit را نصب کرد، باید از مجوز root و Admin به شدت محافظت کرد.
- استفاده از ابزارهای جستجوگر Rootkit:

ChkRootkit: این برنامه بدون اتکا به هسته ی O.S. سعی به کشف Rootkit ها دارد که البته کار بسیار مشکلی در پیش دارد.

یادآوری: یک نفـوذگر در فاز شناسایی و یورش سعی می کند از طریق ضعیف ترین سیستم وارد عمل شود و به عنوان مثال شروع به نصب Sniffer بر روی آن می کند.

حال می توان در شبکه از یک یا چند ماشین بعنوان طعمه (Honey pot) استفاده کرد تا بتوان نفوذگران را گیر انداخت.

طعمه یا Honey pot:

طعمه یا Honey pot، اصطلاحاً ماشینی است که به طور عمدی بر روی آن سرویس دهنده های ضعیف و برنامه های با اشکال نصب می شود. این سیستم عملاً هیچ فایده ای برای سرویس دهنده ندارد.

این کار چند حسن دارد:

1- می توان بر روی آن اطلاعات غلط و گمراه کننده گذاشت و به این ترتیب نفـوذگر را گمراه کرد.

2- از طرفی وجود چنین سیستمی نفـوذگر را برای مدتی به خود مشغول می کند وی را معطل می کند.

3- می توان یک IDS را کنار این ماشین قرار داد تا هشدار حمله دهد و شاید بتوان نفـوذگر را شناسایی کرد.

گام پنجم حمله

رد گم کردن و پوشش مسیرها و پنهان کردن ردپاها

از یک دیدگاه می توان حملات را به دو دسته ی Active و Passive تقسیم کرد. در حملات Active معمولاً نتایج ظاهری آنها سریعاً آشکار می شود. مانند D.O.S که معمولا هدف از این حمله کسب شهرت یا مسائل سیاسی یا ضربه زدن به قربانی است.

اما برخی حمله ها هیچ جنجال به پا نمی کنند و نفـوذگر در این حملات و نفـوذها، اهداف بلند مدتی دارد. به این حمله ها Passive گویند. در این نوع حمله ها نفـوذگر باید آثار حمله را مخفی نگه دارد و ردپای خود را پاک کند. یک نفـوذگر برای پنهان کرد رد خود راهکارهایی دارد که در ادامه آنها را بررسی می کنیم.

دستکاری فایلهای event logs (ثبت رویداد)

در همهٔ سیستم عامل ها فایل هایی وجود دارند که تمام رخدادهای سیستمی در آنها ثبت می شوند. یک نفـوذگر پس از ورود به یک سیستم باید سراغ این فایل ها رفته و رد پاهای خود را پاک کند. رخدادهای زیر نمونه ای از این رخدادها هستند:

- ثبت تلاش های ناقص برای ورود به سیستم (مثلاً برای بدست آوردن Passwordها)
- ثبت خاتمه ی اجرای یک سرویس یا شروع یک سرویس جدید در سیستم
- ثبت دسترسی به انواع فایل ها مانند حذف، تغییر و ...
- ثبت تغییر در مجوزهای دسترسی کاربران
- و ...

مثال: رخدادهای ویندوز 2000 و NT

در ویندوز 2000 و NT یک سرویس با عنوان event log وجود دارد که تمام اتفاقات سیستمی را در فایلهایی درج می کند. اتفاقاتی از قبیل:

- ورود و خروج کابران
- اشکالات به وجود آمده در هر یک از سرویس دهنده ها
- خطاهای برنامه ی کاربردی
- هرگونه نقض حریم (مثلاً تجاوز از سطح دسترسی یا تجاوز از محدوده ی حافظه)

به محض وقوع یک رخداد اطلاعات آن به طور موقت در فایلهای security.log، system.log و application.log ذخیره می شوند و ویندوز در فواصل زمان مشخص این فایلها را در فایلهای دائمی

secevent.evt (رخدادهای مربوط به مسائل امنیتی)

sysevent.evt (رخدادهای مربوط به عملکرد o.s)

appevent.evt (رخدادهای مربوط به برنامه های کاربردی) منتقل می کند.

نکته: این فایلها ساختار دودویی خاصی دارند وASCII نیستند لذا تغییر دادن آن ها با ویرایشگر عادی ممکن نیست.

در کل باید این فایلها را در حافظه لود کرد، سپس تغییرات مورد نظر را اعمال کرد بعد دوباره انها را باز نویسی کرد. برای این کار می توان از ابزار هایی مانند WINZAPPER استفاده کرد. (به آدرس HTTP://ntsecurity.nu/toolbox/winzapper)

نکته: دقت کنید در ویندوز هرگونه تغییری در فایلهای ثبت رخداد، بعد ازrestart شدن سیستم اعمال می شود .

دفاع از فایلهایevent logs

- مجوزadmin و root را محافظت کنید زیرا فقط admin می تواند این فایلها را دستکاری کند.
- می توان فایلهای ثبت رخداد را رمزنگاری کرد. از ابزار SecureSyslog برای این منظور می توان استفاده کرد. (www.core_sdi.com/english/frresoft.html)

کانال پنهان Covert Channel

هنگامی که یک نفوذگر به یک سیستم وارد شد و یک backdoor و یا یک اسب تروا بر روی سیستم نصب کرد باید به نحوی برای شروع حمله با ماشین قربانی تبادل داده کند، مثلا؛ فرمان حمله را صادر کند، نوع عملیات را تعیین کند ویا تغییر استراتژی دهد. اما این احتمال وجود دارد که برقراری ارتباط و تبادل داده با ماشین نفوذگر، کشف شود و یا توسط IDS گزارش شود.

یک نفوذگر مایل است در حین حمله، تبادل داده با قربانی را به طور مخفیانه انجام دهد. برای این کار ازکانالهای مخفی استفاده می کند.

یاد آوری: در اکثر روشهای مبتنی بر درهای پشتی و اسب تروا، نفوذگر یک برنامه برروی ماشین قربانی نصب کرده است که کنترل ماشین قربانی را بدست گرفته و یا منتظر صدور فرمان از جانب نفوذگر و اجرای این فرمانها است.

نکته: البته می توان این تبادل اطلاعات را به طور معمول و ازطریقTCP یا UDP نیز انجام داد و داده‌ها را ردوبدل کرد. اما این روشها توسط خودکاربر،firewall یا IDS قابل کشف وشناسایی هستند مثلا برنامه یک پورت باز کند و شنود کند و ..

1. ایجاد کانال پنهان از طریق ICMP

یادآوری: با ICMP کاربران قادرند یک ماشین درون یا بیرون شبکه را PING کنند و ترافیکICMP معمول است. در این روش نفوذگر اطلاعات خود را در قالب بسته هایICMP برای ماشین قربانی ارسال می کند و برنامه نصب شده در ماشین قربانی خود را به جایICMP جازده وبسته ها را دریافت می کند واطلاعات درون آنها را استخراج می کند.

نکته: دقت کنید ICMP یک پروتکل در سطح لایه 3 می باشد ومفهوم پورت ندارد. در حقیقت فقط یک پروسه می تواند بسته های ICMP را دریافت کند پس پورت نمی خواهد.

این روش یک کانال پنهان موثر است زیرا هیچ پورتی باز نمی شود و نمی توان باپویش پورتهای باز، متوجه این امر شد.

مقابله با این روش:

در نهایت می توان در firewall اجازه عبور و مرور بسته های ICMP را نداد.

2. ایجاد کانال پنهان از طریق پورت UDP 53

اگر بسته های ICMP توسط firewall حذف شوند می توان از پورت udp_53 استفاده کرد این پورت مخصوص DNS می باشد و عبور و مرور بسته های UDP با پورت 53 در شبکه مجاز و عادی است. در این حالت نفوذگری می تواند در قالب پرس وجو وپاسخ های DNS، داده های خود را ارسال کند.

منابع:

1. Fundamental of Computer Security Technology, Edward Amoroso.

2. Cryptography and network security, Principal and Practice, William Stalings.

3. شبکه های کامپیوتری، اندرو. اس. تننباوم، ترجمه دکتر حسین پدرام.

4. نفوذگری در شبکه و راه های مقابله با آن، ترجمه احسان ملکیان.