



## یک کارآگاه خصوصی استخدام کنید

آشنایی ساخت سیستم های آشکارساز نفوذ به شبکه

امین صفائی

### اشاره

چه خوب بود اگر تمامی سیستم های کامپیوتری از امنیت کامل برخوردار بودند. ولی متأسفانه امنیت کامل شبکه های کامپیوتری محال است و به این زودی ها نمی توان انتظار داشت سیستم های کامپیوتری از امنیت کامل بهره مند شوند. زیرا حتی اگر این سیستم ها کاملاً امن باشند و ورود هرگونه عامل خارجی تهدیدکننده امنیت، به این سیستم ها محال باشد، همچنان امکان سوءاستفاده عوامل داخلی در این سیستم ها وجود دارد. هدف سیستم های آشکارساز (Intrusion Detection Systems) IDS در واقع شناسایی ترافیک های مشکوک شبکه، هشدار در مورد دسترسی های غیرمجاز، و سوءاستفاده از سیستم های کامپیوتری است. این سیستم ها را می توان به دزدگیر منزل یا اتومبیل تشبیه نمود که در زمان وقوع رفتارهای غیرعادی، صدای آژیر را به صدا در می آورد.

در شبکه های کامپیوتری یا سرورهای بزرگ، IDSها اطلاعات فرستنده بسته اطلاعاتی به شبکه را جمع آوری می کنند، محتوای بسته را تجزیه می نمایند و دنبال نشانه هایی از مزاحمت یا سوءاستفاده در آن می گردند. دوش متداول در ساختن این سیستم ها وجود دارد: سیستم های شناسایی سوءاستفاده گر (Misuse Detection) و سیستم های شناسایی رفتارهای غیرعادی (Detection Anomaly). در سیستم های شناسایی سوءاستفاده گر، سیستم از شکل حملات اطلاع دارد و شکل بسته های ارسالی از منبع ارسال کننده را با این الگوها مقایسه می کند. ولی در سیستم های شناسایی رفتارهای غیرعادی، سیستم از بسته هایی که سورس ها ارسال می کنند، اطلاعات آماری تهیه می کند و رفتارهای غیرعادی هر سورس را اطلاع می دهد. این مقاله با معرفی این سیستم ها و انواع آن ها طرز کار این سیستم ها را به صورت خلاصه توضیح می دهد و با طرح یک مثال طریقه ساخت یک IDS ساده را به شما می آموزد.

در این شماره:

۱۴۸/ یک کارگاه خصوصی استخدام کنید

۱۵۴/ توصیه های امنیتی برای امروز و هر روز

۱۵۶/ محافظت از شبکه در مقابل

خرابکاران داخلی

۱۵۸/ ویروس ها زیرک شده اند

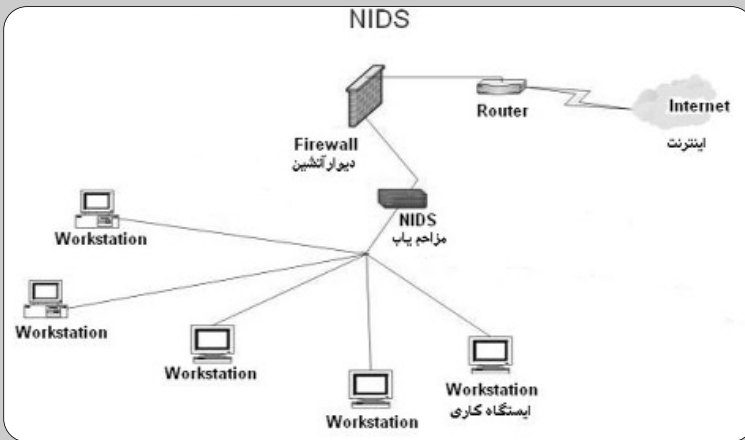
صفحاتی درباره امنیت شبکه ها و

کامپیوترهای شخصی

## IDSهای شبکه و انواع آن

سیستم آشکارساز شبکه، ترافیک شبکه را کنترل می‌کند، فعالیت‌های غیرعادی و مشکوک را زیر نظر می‌گیرد و در صورت مشاهده هرگونه مزاحمت یا نشانه مشکوک، مدیر شبکه را مطلع می‌سازند. در برخی موارد این سیستم‌ها می‌توانند با مشاهده رفتارهای غیرعادی و مشکوک یا نشانه‌های حمله از طرف یک آدرس IP، آن آدرس را در فهرست سیاه قرار دهند و مانع دسترسی آن به شبکه شوند. امروزه IDSهای مختلفی وجود دارند که از راه‌های گوناگونی ترافیک‌های مشکوک را شناسایی می‌کنند.

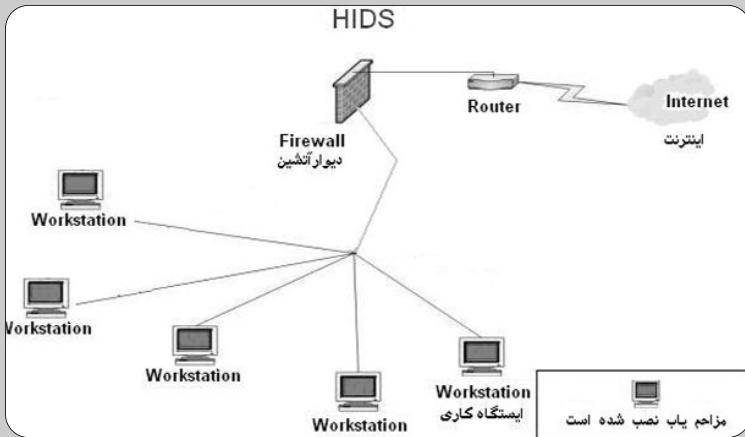
شکل ۱- نحوه عملکرد NIDS



## HIDS و NIDS

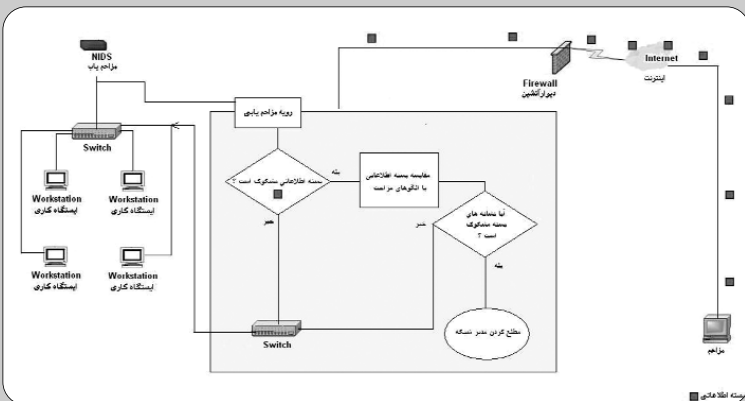
سیستم‌های آشکارساز نفوذ به شبکه یا NIDSها (Network Intrusion Detection System) یا در قسمت‌های مهم شبکه (مانند کانال‌های ارتباطی شبکه با محیط بیرون) قرار می‌گیرند و ترافیک شبکه را کنترل می‌کنند. البته از آن جایی که این سیستم‌ها کلیه ترافیک ورودی و خروجی و تمامی فعالیت‌های شبکه را کنترل می‌کنند، می‌توانند باعث کندی سرعت شبکه نیز بشوند. با این حال، وجود این سیستم‌ها نیاز اولیه اکثر شبکه‌های پیشرفته است.

شکل ۲- نحوه عملکرد HIDS



از طرف دیگر HIDSها (Host Intrusion Detection System) یا سیستم‌های آشکارساز میزبان، تنها روی یک دستگاه یا سیستم در شبکه نصب می‌شوند و صرفاً ترافیک‌های ورودی و خروجی به آن دستگاه را کنترل می‌کنند و در صورت مشاهده هر مورد مشکوک، مسئله را به مدیر سیستم هشدار می‌دهند. شکل ۱ و ۲ تفاوت این دو نوع سیستم را نشان می‌دهد و شکل ۳ ساختار سیستم مزاحم‌یاب شبکه را ارائه می‌نماید.

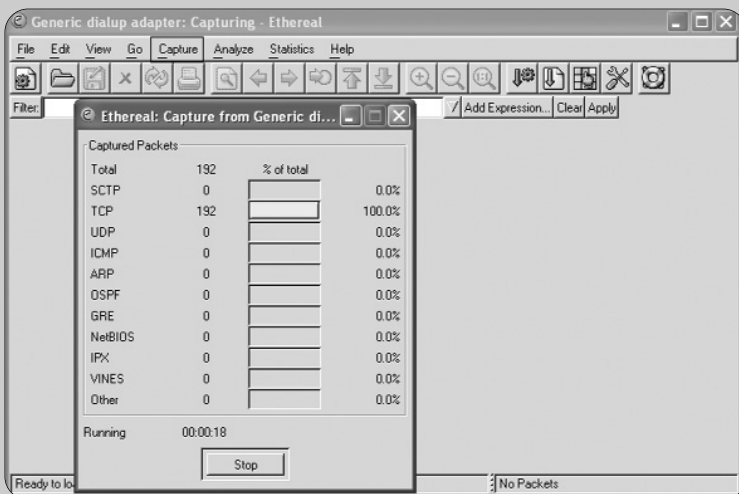
شکل ۳- نمای کلی سیستم آشکارساز نفوذ به شبکه



همانطور که در شکل ۳ می‌بینید، بسته اطلاعاتی فرد مزاحم از اینترنت و فایروال عبور می‌کند و به NIDS می‌رسد. NIDS بسته را کنترل می‌کند و اگر مشکوک باشد، نشانه‌های بسته اطلاعاتی یا پکت را با نشانه‌های بسته‌های مضر یا رفتارهای غیرعادی مقایسه می‌نماید. سپس اگر بسته دارای نشانه‌های حمله باشد و مشکوک به نظر برسد، به مدیر شبکه اطلاع می‌دهد و در برخی موارد نیز مانع ورود هر گونه بسته‌ای از این IP می‌گردد.

اگر بخواهیم IDSها را طبقه‌بندی کنیم، می‌توانیم آن‌ها را در دو گروه اصلی قرار دهیم: IDSهایی که محتوای پکت ارسالی را کنترل می‌نمایند و آن را با اطلاعات پایگاه اطلاعاتی شامل صفات و نشانه‌های بارز حملات مقایسه می‌کنند که اصطلاحاً به Signature Based معروفند. (درست شبیه کاری که ضدویروس‌ها انجام می‌دهند). البته این نوع IDSها یک مشکل اساسی دارند و آن ناتوانی در شناسایی حملات جدید است. این سیستم‌ها فقط

شکل ۴- جمع‌آوری ترافیک شبکه



قادر به شناسایی حملاتی هستند که قبلا برای آنها تعریف شده است.

نوع دیگری از IDSها سیستم‌هایی هستند که قادر به شناسایی عملکرد غیر عادی یک IP فرستنده بسته هستند. مثلا اگر یک منبع، دویست بار در دقیقه به پورت ۲۱ دسترسی دارد مشخص است که قصد هک کردن سیستم را دارد. به این نوع سیستم‌ها اصطلاحا Anomaly Based می‌گویند.

IDSها را می‌توان از نظر نوع واکنش به حملات نیز به دو گروه تقسیم نمود: گروه اول سیستم‌هایی هستند که فقط حملات را شناسایی می‌کنند و پیغام می‌دهند (Passive IDS) و گروه دیگر نه تنها حملات را شناسایی می‌کنند و به مدیر سیستم پیغام می‌دهند، بلکه راه ارتباطی فرستنده مورد نظر را نیز به شبکه مسدود می‌کنند و نمی‌گذارند هیچ بسته‌ای از طرف این IP به شبکه وارد شود (Reactive IDS). به عنوان نمونه یکی از IDSهای معروف، این سورس و رایگان

که می‌توانید از اینترنت دانلود کنید، Snort است. این نرم‌افزار در ویندوز و لینوکس قابل نصب می‌باشد. (نشانی [www.snort.org](http://www.snort.org))

### رابطه فایروال و IDS

در دیوارهای آتش امروزی IDS نقش مهمی را به عهده دارد. مثلا امروزه فناوری‌های جدیدی مثل IPS (Intrusion Prevention System) یا سیستم پیشگیری از نفوذ دیواره آتش شامل فیلترهای مختلف شبکه و یک IDS واکنشی (Reactive) است که شبکه را از آسیب‌های خارجی مصون نگاه می‌دارد. اصولا دیواره آتش اولین عامل امنیتی (دیوار حائل بین شبکه و محیط خارج) در هر شبکه است. بهترین روش استفاده از دیواره آتش این است که به صورت پیش فرض جلوی تمامی ترافیک‌های ورودی به شبکه را بگیرد و فقط به مدیر شبکه اجازه دهد پورت‌های مخصوصی را برای استفاده باز بگذارد. مثلا مدیر

شبکه، پورت‌های ۸۰ (میزبان وب سایت) یا پورت ۲۱ (FTP) سرور که به صورت مکرر مورد استفاده قرار می‌گیرد را بازنگه دارد. ولی در برخی از موارد همین پورت‌های باز مشکل سازند و نفوذگر می‌تواند از طریق آن‌ها به شبکه راه بیابد. نیاز به IDS در این قسمت محسوس به نظر می‌رسد. IDS می‌تواند ترافیک شبکه را در این پورت‌های باز زیر نظر بگیرد و در صورت مشاهده هرگونه رفتار غیرعادی، شما را مطلع کند.

اگرچه IDSها می‌توانند از شبکه محافظت کنند و ابزاری مناسب برای پیدا کردن عوامل خارجی مزاحم در شبکه شما هستند، اما درستی عملکرد و بروز بودن این سیستم‌ها الزامی به نظر می‌رسد. مثلا دزدگیر اتومبیلی را در نظر بگیرید که حساسیت زیادی دارد و حتی در صورت تماس دست با بدنه اتومبیل آژیر را به صدا در می‌آورد. اما دزدگیر یک اتومبیل دیگر که حساسیت کمی دارد، از امنیت کمتری برخوردار است و ممکن است به همین علت سرقت شود. این مشکل در IDSها نیز صادق است. در نتیجه قبل از استفاده از این سیستم‌ها باید از درستی و بروز بودن آن‌ها اطمینان حاصل کنیم. به علاوه، اگر یک IDS بروز نباشد و نشانه‌های حمله‌های جدید را نشناسد، هرکدام می‌توانند به راحتی شبکه را مورد حمله قرار دهند. یعنی این کار باعث خواهد شد شبکه‌ای آسیب پذیر داشته باشیم.

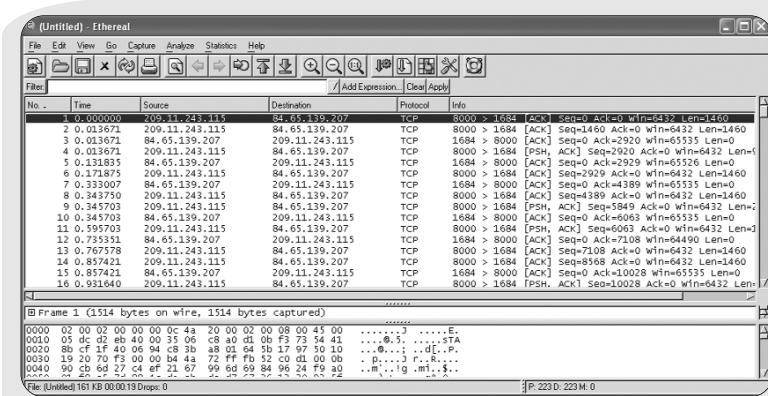
### طراحی و ایجاد یک NIDS ساده

تا اینجا با مدل کلی IDSها آشنا شدیم. در این قسمت یک NIDS ساده و برای سهولت کار Off-Line می‌سازیم. منظور از Off-Line این است که ترافیک شبکه را که در یک فایل ذخیره شده است بررسی می‌کنیم، سپس فایل را آنالیز می‌نماییم و در صورت مشاهده نشانه‌های مشکوک و رفتارهای غیرعادی هر سورس پیغام‌هایی را به کاربر می‌دهیم. برای طراحی این سیستم از UML و برای اجرای آن می‌توانید از جاوا یا هر زبان برنامه نویسی دیگری (مانند ++C یا C) استفاده کنید.

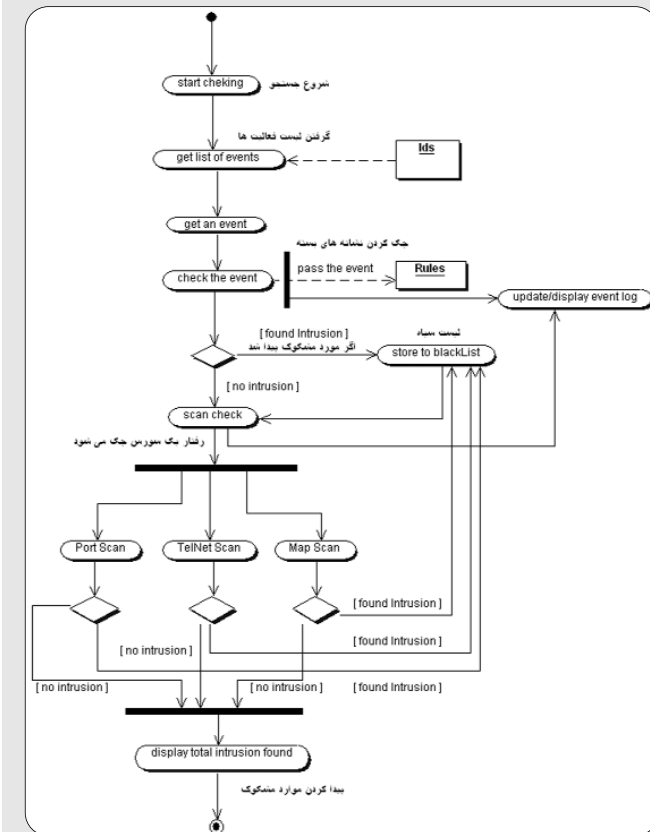
برای به دست آوردن اطلاعات ترافیک شبکه باید از سیستم‌های آنالیز ترافیک شبکه استفاده کنید. Ethernal یک نمونه از این سیستم‌هاست که به صورت رایگان قابل دانلود کردن و استفاده می‌باشد. (نشانی [www.ethereal.com/download.html](http://www.ethereal.com/download.html))

پس از دانلود و اجرای برنامه، به منوی Capture بروید و Start را انتخاب کنید. سپس در پنجره Capture Option، دکمه OK را کلیک کنید (شکل ۴). با این کار سیستم بسته‌های اطلاعاتی ورودی به شبکه و ترافیک شبکه را ضبط می‌کند.

چند ثانیه صبر کنید تا سیستم اطلاعات شبکه را ذخیره کند. سپس دکمه Stop را کلیک کنید. بعد از



شکل ۵- ترافیک شبکه (TcpDump)



شکل ۶- عملکرد کلی NIDS



## ایجاد پرونده برای هر سورس

همانطور که در شکل ۷ مشاهده می‌کنید، کلاس EventChecker به کلاس ScrProfile ارتباط دارد. کلاس EventChecker برای هر سورس IP یک پروفایل یا پرونده تشکیل می‌دهد تا بتواند عملیات آن منبع را بهتر زیر نظر داشته باشد.

کنترل کردن مشخصات فریم در مقایسه با نشانه‌های بسته‌های مشکوک (Signature) در مرحله بعدی انجام می‌شود. در برخی موارد، پیدا کردن نشانه‌های حمله به شبکه بسیار ساده است. مثلاً اگر فرستنده بسته، بسته را به آدرس مشابه خود ارسال کند (آدرس IP فرستنده و گیرنده یکی باشد)، به این نوع حمله اصطلاحاً Land Attack می‌گویند. پس اگر سیستم با فریمی مواجه شد که آدرس فرستنده و گیرنده بسته مشابه بود، احتمال حمله به شبکه وجود دارد. مثلاً شکل ۱۰، دو بسته (فریم ۱۴ و ۱۷) را نشان می‌دهد که احتمالاً Land Attack می‌باشند. جدول نمایش داده شده در شکل ۱۱ برخی از نشانه‌های سایر حملات به شبکه را که می‌توانید کنترل کنید، نمایش داده است.

کدهای زیر (با استفاده از جاوا) پروفایل هر سورس را کنترل می‌کند و اگر یک منبع تلاش دارد بیش از حد معمول از طریق یک پورت به شبکه متصل شود، این پورت در لیست سیاه قرار می‌گیرد. همانطور که در کدهای شکل ۱۲ می‌بینید اولین کار، جستجو در فهرست تمام سورس‌های منبع (پروفایل‌ها) است. سپس برنامه هر سورس را در نظر می‌گیرد و فعالیت‌های آن منبع را در شبکه پیدا می‌کند. در While بعدی برنامه پورت‌هایی که سورس مورد نظر به آن دسترسی دارد را یکی یکی ذخیره می‌کند و اگر یک سورس از مقدار تعریف شده (در اینجا پنجاه بار) بیشتر به یک پورت دسترسی داشته باشد، پیام اخطار داده می‌شود. (البته در این قسمت از برنامه مدت زمان دسترسی تعیین نشده است و شما می‌توانید با دستکاری این قسمت از برنامه، در مدت زمان مشخص، مثلاً پنج ثانیه این کار را انجام دهید).

## کنترل NIDS بر رفتارهای مشکوک

هکرها معمولاً با سه هدف عمده به شبکه شما نفوذ می‌کنند. از کار انداختن شبکه، خواندن و استفاده از اطلاعات سری، و تغییر یا حذف اطلاعات. در نتیجه منطقی‌ترین کار برای شناسایی رفتارهای مشکوک در یک شبکه، پیدا کردن نشانه‌هایی از مزاحمت‌هایی است که باعث مشکلات بالا می‌شوند. در مثال بالا که یک NIDS غیرفعال یا Offline است، موتور جستجوگر رفتارهای مشکوک را با جستجو در قسمت ابتدایی بسته‌های TCP/IP یعنی header پیدامی‌کند. به این صورت که محتوای هر بسته

نام حمله	نشانه حمله
Land Attack	آدرس فرستنده و گیرنده یکسان است
WinNuke Attack	این حمله که بسیار خطرناک است، باعث از کار افتادن کامل شبکه می‌گردد. در این حمله فرستنده (هکر) بسته آلوده را از پورت ۱۳۹ و از طریق پروتکل TCP به سرور می‌فرستد. IDS باید در صورت مشاهده این عمل از جانب هر سورسی، مدیر شبکه را مطلع سازد.
ACK scan	وقتی که آدرس منبع و مقصد یکی باشد و پارامتر Win بزرگ‌تر از ۱۰۲۸ باشد.
اتصال به پورت‌های مهم	وقتی فرستنده قصد اتصال به پورت‌های مهمی مثل ۳۱، ۴۵۶ یا ۵۵۵ را داشته باشد.
Port Scan و nmap	پورت اسکن در واقع عبارت است از اسکن کردن فراوان یک پورت در فاصله زمانی کم. مثلاً اگر یک IP آدرس پورت ۴۵۶ را در پنج ثانیه ۱۰۰ بار اسکن کند، رفتار این IP آدرس مشکوک است.

شکل ۱۱- نشانه حملات در بسته‌های ارسالی به شبکه

ارسالی از سوی هر منبع را تحلیل می‌نماید و پارامترهای آن را با نشانه‌های مشکوک از پیش تعیین شده مقایسه می‌کند و در صورت مشاهده هرگونه مورد مشکوک مدیر شبکه را مطلع می‌نماید. در این روش ترافیک‌های مشکوک شبکه بدون این‌که نیازی به خواندن محتوای بسته باشد، از طریق نشانه‌های مشکوک (Signatures) شناسایی می‌شوند. روش‌های مختلفی برای معرفی این نشانه‌ها وجود دارد که از آن جمله می‌توان از نشانه‌های برنامه‌ریزی شونده داخل سورس کد و سیستم‌های هوشمند نام برد.

سیستم موارد مشکوک (آدرس IP سورس) را شناسایی می‌کند و به شما اطلاع می‌دهد. کار کردن با این سیستم بسیار آسان است و می‌توانید با دستکاری فایل TCPdump و تغییر اطلاعات فریم‌ها، موارد مشکوک بیشتری پیدا کنید. بالین‌که نمی‌توان این سیستم را با IDSهای بزرگ مقایسه کرد، شاید بتوان گفت این سیستم یک نمونه بسیار کوچک از IDSهای بزرگ است و سازوکار اصلی این سیستم‌ها را نشان می‌دهد. استفاده از IDSها امروزه بسیار متداول شده است. در اکثر ضدویروس‌های امروزی IDSهای کوچکی تعبیه شده است که محتوای بسته‌های ارسالی به کامپیوتر شما را کنترل می‌کند و در صورت مشاهده مورد مشکوک، آدرس IP را مسدود می‌نماید و اجازه استفاده از شبکه (یا سیستم) را به آن منبع نمی‌دهد. شاید بتوان گفت با این کار امنیت بیشتری برای شبکه‌ها و حتی کامپیوترهای شخصی به وجود می‌آید. ولی برخی از این سیستم‌ها مثل سیستمی که در Norton Internet Security وجود دارد، در بعضی موارد اجازه استفاده از پورت‌های کامپیوتر را حتی به سیستم‌های داخلی نیز نمی‌دهد و پورت را به صورت خودکار می‌بندد. اگرچه شاید هنوز IDSها زیادی در شبکه‌های امروزی وجود نداشته باشند و جز معدود شرکت‌های بزرگ فناوری اطلاعات، بقیه شرکت‌ها هنوز از این فناوری استفاده نمی‌کنند و این سیستم‌ها را جدی نمی‌گیرند، باید این موضوع را در نظر داشت که همیشه کسانی هستند که حتی برای تفریح دوست دارند به شبکه شما راه پیدا کنند.


با استفاده از IDSها می‌توانیم یک کارآگاه خصوصی استخدام کنیم که هر گونه مورد

سیستم‌های آشنکارساز نفوذ شبکه می‌توانند برخی از مزاحمت‌های شبکه را اطلاع دهند و بسیار ساده هم تهیه می‌شوند. در برخی موارد ممکن است اشتباه کنند و به مدیر سیستم هشدار اشتباه بدهند. (به این حالت اصطلاحاً False Alarm می‌گویند).

برای آشنایی بیشتر با NIDSها و سازوکار آن‌ها می‌توانید از سایت ماهنامه شبکه (قسمت دریافت فایل) فایل IDS.jar که یک فایل اجرایی جاوا است را دانلود کنید و با استفاده از دو فایل TCPdump موجود (nmap.txt و nmapsP.txt) مشاهده کنید که چگونه IDSها قادر به شناسایی موارد مشکوک در شبکه هستند. برای این کار، مراحل زیر را دنبال کنید.

- برنامه را اجرا کنید
- با انتخاب منوی File، Import TCPdump را کلیک کنید.
- فایل TCPdump را انتخاب کنید. مثلاً فایل nmapsP.txt را انتخاب نمایید و منتظر شوید سیستم فایل را بخواند و فریم‌ها را آنالیز کند.
- از منوی Start، Detection را انتخاب کنید. سپس

مشکوکى را قبل از اين که زيان جدى به شبکه بزند، شناسايى و از ورود آن به شبکه ممانعت به عمل آورد.

اين مقاله مقدمه ساده‌اى بود در مورد NIDSها، مطالعه منابع زير در يادگيرى بيشتر در مورد اين سيستم‌ها به شما يارى مى‌رساند. 

### منابع:

- <http://sunsite.uakom.sk/sunworldonline/swol-09-1988/swol-09-security.html>
- Totsuka, A et al. 2000. Network-based intrusion detection-modeling for a larger picture. Proceedings of LISA 2000. 3- 8 November 2000. USENIX. pp. 227-232.
- [www.usenix.org/events/lisa02/tech/fullpapers/totsuka/totsuka.pdf](http://www.usenix.org/events/lisa02/tech/fullpapers/totsuka/totsuka.pdf)
- Paxson, V. 1999. Bro: a system for detecting network intruders in real-time. Computer Networks. 31(23-24). December 1999. pp. 2435-2463.
- [www.cs.unc.edu/jeffay/courses/nidsS05/signatures/paxson-bro-cn99.pdf](http://www.cs.unc.edu/jeffay/courses/nidsS05/signatures/paxson-bro-cn99.pdf)
- Richard, M(2001) Are there limitations of Intrusion Signatures
- [www.sans.org/resources/idfaq/limitations.php](http://www.sans.org/resources/idfaq/limitations.php)

```
public void ScanCheck(Map id) {
    حداکثر اسکن هر پورت
    int PortScanLimit= 50
    پروفایل‌ها را از لیست استخراج کن
    Set pro = profiles.entrySet();
    Iterator pro_ie = pro.iterator();
    Events profileIP = new Events();
    int portscan = 0;
    int eventnom = 0;
    String tempPort = "";
    جستجو در لیست همه پروفایل‌ها
    while (pro_ie.hasNext()) {
        Map.Entry ev3 = (Map.Entry) pro_ie.next();
        profileIP = (Events) ev3.getValue();
        یک ست جدید ایجاد کن
        Set evn4 = id.entrySet();
        Iterator evn_ie = evn4.iterator();
        Events IP = new Events();
        جستجو برای پیدا کردن پروفایلی که قصد حمله دارد
        while (evn_ie.hasNext()) {
            Map.Entry ev4 = (Map.Entry) evn_ie.next();
            IP = (Events) ev4.getValue();
            اگر منبع همان منبع ذخیره شده قبلی است
            if (IP.getSrcAddr().equals(profileIP.getSrcAddr())) {
                اگر منبع پورت را اسکن می‌کند
                if (IP.getDstAddr().equals(tempPort)) {
                    به مقدار پورت اسکن یک واحد اضافه کن
                    portscan++;
                    tempPort = profileIP.getDstAddr();
                } else {
                    اگر برای اولین بار این پورت را اسکن می‌کند
                    tempPort = profileIP.getDstAddr();
                }
            }
        }
        اگر مقدار اسکن از حد تعریف شده بیشتر است زنگ را فعال کن
        if (portscan >= PortScanLimit) {
            Toolkit.getDefaultToolkit().beep();
            try {
                Thread.sleep(500);
            } catch (InterruptedException e) {
            }
            اگر این سورس قبلا در لیست سیاه نبوده این سورس را اضافه کن
            if (!badIPprofiles.containsKey(s.getSrc())) {
                badIPprofiles.put(s.getSrc(), s);
            }
            در اینجا می‌توانید هشدار را نمایش دهید
            // System.out.println("مقدار")
        }
        مقدار اسکن را برای سورس بعدی مساوی صفر قرار بده
        portscan = 0;
    }
} //end while
```

شکل ۱۲- پیدا کردن پورت اسکن