Persian Magazine
Web Security Mgz .com

مهمترین نقاط آسیب پذیر

UNIX و Linux

تهیه کننده : امیر حسین شریفی info@websecuritymgz.com

تاریخ : 28 بهمن ماه 1382

منبع:

1- www.SRCO.ir

فهرست مطالب صفحه

مقدمه 3
مهمترین نقاط آسیب پذیر لینوکس 5
BIND Domain Name System 6
Remote Procedure Calls 10
Apache Web Server 14
ccountA هاي با رمز عبور ضعيف و يا فاقد رمز عبور 20
Clear Text Services 27
Sendmail 31
Simple Network Management Protocol 35
Secur Shel 39
عدم پيكربندي صحيح سرويسهاي NIS و FS 42
Open Secure Sockets Layer 47

مهمترین نقاط آسیب پذیر

یونیکس و لینوکس

مقدمه

سیستم عامل، یکی از عناصر چهار گانه در یک سیستم کامپیوتری است که دارای نقشی بسیار مهم و حیاتی در نحوه مدیریت منابع سخت افزاری و نرم افزاری می باشد . پرداختن به مقوله امنیت سیستم های عامل ، همواره از بحث های مهم در رابطه با ایمن سازی اطلاعات در یک سیستم کامپیوتری بوده که امروزه با گسترش اینترنت ، اهمیت آن مضاعف شده است. بررسی و آنالیز امنیت در سیستم های عامل می بایست با ظرافت و در چارچوبی کاملا" علمی و با در نظر گرفتن تمامی واقعیت های موجود ، انجام گیرد تا از یک طرف تصمیم گیرندگان مسائل استراتژیک در یک سازمان قادر به انتخاب مستند و منطقی یک سیستم عامل باشند و از طرف دیگر امکان نگهداری و پشتیبانی آن با در نظر گرفتن مجموعه تهدیدات موجود و آتی ، بسرعت و بسادگی میسر گردد .

اکثر کرم ها و سایر حملات موفقیت آمیز در اینترنت ، بدلیل وجود نقاط آسیب پذیر در تعدادی اندک از سرویس های سیستم های عامل متداول است . مهاجمان ، با فرصت طلبی خاص خود از روش های متعددی به منظور سوء استفاده از نقاط ضعف امنیتی شناخته شده ، استفاده نموده و در این راستا ابزارهای متنوع ، موثر و گسترده ای را بمنظور نیل به اهداف خود ، به خدمت می گیرند . مهاجمان ، در این رهگذر متمرکز بر سازمان ها و موسساتی می گردند که هنوز مسائل موجود امنیتی (حفره ها و نقاط آسیب پذیر) خود را برطرف نکرده و بدون هیچگونه تبعیضی آنان را بعنوان هدف ، انتخاب می نمایند . مهاجمان به سادگی و به صورت مخرب ، کرم هایی نظیر : بلستر ، اسلامر و Code Red را در شبکه منتشر می نمایند. آگاهی از مهمترین نقاط آسیب پذیر در سیستم های عامل ، امری ضروری است . با شناسائی و آنالیز اینگونه نقاط آسیب پذیر توسط کارشناسان امنیت اطلاعات ، سازمان ها و موسسات قادر به استفاده از مستندات علمی تدوین شده بمنظور برخورد منطقی با مشکلات موجود و ایجاد یک لایه حفاظتی مناسب می باشند.

در مجموعه مقالاتی که ارائه خواهد شد ، به بررسی مهمترین نقاط آسیب پذیر یونیکس و لینوکس خواهیم پرداخت . در این راستا ، پس از معرفی هر یک از نقاط آسیب پذیر ، علت وجود ضعف امنیتی ، سیستم های عامل در معرض تهدید ، روش های تشخیص آسیب پذیری سیستم و نحوه مقابله و یا پیشگیری در مقابل هر یک از نقاط آسیب پذیر ، بررسی می گردد

همانگونه که اشاره گردید ، اغلب تهدیدات و حملات ، متاثر از وجود نقاط آسیب پذیر در سیستم های عامل بوده که زمینه تهاجم را برای مهاجمان فراهم می آورد . شناسائی و آنالیز نقاط آسیب پذیر در هر یک از سیستم های عامل ، ماحصل تلاش و پردازش دهها کارشناس امنیتی ورزیده در سطح جهان است و می بایست مدیران سیستم و شبکه در یک سازمان بسرعت با آنان آشنا و اقدامات لازم را انجام دهند.

نقاط آسیب پذیر موجود در هر سیستم عامل که در ادامه به آنان اشاره می گردد ، سندی پویا و شامل دستورالعمل های لازم به منظور برخورد مناسب با هر یک از نقاط آسیب پذیر و لینک هائی به سایر اطلاعات مفید و تکمیلی مرتبط با ضعف امنیتی است .

مهمترین نقاط آسیب پذیر یونیکس

یونیکس ، یکی از سیستم های عامل رایج در جهان بوده که امروزه در سطح بسیار وسیعی استفاده می گردد . تا کنون حملات متعددی توسط مهاجمین متوجه سیستم هائی بوده است که از یونیکس (نسخه های متفاوت) بعنوان سیستم عامل استفاده می نمایند . با توجه به حملات متنوع و گسترده انجام شده ، می توان مهمترین نقاط آسیب پذیر یونیکس را به ده گروه عمده تقسیم نمود :

- BIND Domain Name System
- Remote Procedure Calls (RPC)
- Apache Web Server
- General UNIX Authentication Accounts with No Passwords or Weak Passwords
- Clear Text Services
- Sendmail
- Simple Network Management Protocol (SNMP)
- Secure Shell (SSH)
- Misconfiguration of Enterprise Services NIS/NFS
- Open Secure Sockets Layer (SSL)

BIND Domain Name System

نرم افزار **BIND** (Berkeley Internet Name Domain) ، در مقیاس گسترده ای و بمنظور پیاده سازی **DNS** (Domain Name Service) ، استفاده می گردد. BIND ، سیستمی حیاتی است که از آن بمنظور تبدیل اسامی میزبان (نظیر : http://www.srco.ir/) به آدرس IP ریجستر شده ،استفاده می گردد .با توجه به استفاده وسیع از BIND و جایگاه حیاتی آن در یک شبکه کامپیوتری ، مهاجمان آن را بعنوان یک هدف مناسب بمنظور انجام حملات ، خصوصا" از نوع Denila Of Service (DoS) انتخاب و حملات متنوعی را در ارتباط با آن انجام داده اند. حملات فوق،از کارافتادن سرویس DNS و عدم دستیابی به اینترنت برای سرویس های مربوطه و میزبانان را می تواند بدنبال داشته باشد. با اینکه پیاده کنندگان BIND ، تلاشی مستمر را از گذشته تا کنون بمنظور برطرف نمودن نقاط آسیب پذیر انجام داده اند ، ولی هنوز تعداد زیادی از نقاط آسیب پذیر قدیمی ، بدرستی پیکربندی نشده و سرویس دهندگان آسیب پذیر در آن باقی مانده است .

عوامل متعددی در بروز اینگونه حملات نقش دارد: عدم آگاهی لازم مدیران سیستم در خصوص ارتقاء امنیتی سیستم هائی که بر روی آنان Bind deamon بصورت غیر ضروری اجراء می گردد و پیکربندی نامناسب فایل ها ، نمونه هائی از عوامل فوق بوده و می تواند زمینه یک تهاجم از نوع DoS ، یک Buffer Overflow و یا بروز اشکال در DNS Cache را بدنبال داشته باشد.از جمله مواردیکه اخیرا" در رابطه با ضعف امنیتی BIND کشف شده است مربوط به یک تهاجم از نوع DoS است . مقاله CA- CERT Advisory 2002-15 جزئیات بیشتری را در این رابطه ارائه می نماید. از دیگر حملات اخیر ، تهاجمی از نوع Overflow Buffer است . مقاله CA-2002-19 CERT Advisory جزئیات بیشتری را در این رابطه در اختیار قرار می دهد. درتهاجم فوق ، یک مهاجم از نسخه آسیب پذیر پیاده سازی توابع Resolver مربوط به DNS استفاده و با ارسال پاسخ های مخرب به DNS و اجرای کد دلخواه ، امکان سوء استفاده از نقطه آسیب پذیر فوق را فراهم و حتی دربرخی موارد می تواند زمینه بروز یک تهاجم از نوع DoS را باعث گردد .

تهدیدی دیگر که می تواند در این رابطه وجود داشته باشد ، حضور یک سرویس دهنده BIND آسیب پذیر در شبکه است . در چنین مواردی ، مهاجمان از وضعیت فوق استفاده و از آن به منزله مکانی جهت استقرار داده های غیر معتبر خود و بدون آگاهی مدیرسیستم استفاده می نمایند. بدین ترتیب ، مهاجمان از سرویس دهنده بعنوان پلات فرمی بمنظور فعالیت های آتی مخرب خود بهره برداری خواهند کرد .

سیستم های عامل در معرض تهدید :

تقریبا" تمامی سیستم های عامل یونیکس و لینوکس بهمراه یک نسخه از BIND ارائه شده اند .در صورت پیکربندی میزبان بعنوان سرویس دهنده ، نسخه ای از BIND بر روی آن نصب خواهد شد.

نحوه تشخیص آسیب پذیری سیستم

در صورت دارا بودن نسخه خاصی از BIND که بهمراه سیستم عامل ارائه و بر روی سیستم نصب شده است ، می بایست عملیات بهنگام سازی آن را با استفاده از آخرین Patch های ارائه شده توسط تولید کننده (عرضه کننده) انجام داد. در صورت استفاده از نسخه BIND مربوط به **ISC**: Internet Software Consortium ، می بایست از نصب آخرین نسخه BIND ، اطمینان حاصل نمود . در صورتیکه BIND نصب شده بر روی سیستم ، نسخه ای قدیمی بوده و یا بطور کامل Patch نشده باشد ، احتمال آسیب پذیری سیستم وجود خواهد داشت . در اکثر سیستم ها ، دستور : " named -v " ، اطلاعات لازم در خصوص نسخه BIND نصب شده بر روی سیستم را بصورت X.Y.Z نمایش خواهد داد . X ، نشاندهنده نسخه اصلی ، Y جزئیات نسخه و

Y ، جزئیات نسخه و

- چندین نقطه آسیب پذیر DoS در رابطه با ISC BIND 8 از آدرس http://www.isc.org/products/BIND/bind-security.html استفاده گردد .

برای آگاهی و استفاده از پیشنهادات لازم بمنظور نصب ایمن تر BIND بر روی سیستم های سولاریس ، می توان از آدرس : Running the BIND9 DNS Server Securely و آرشیو مقالات ارائه شده در آدرس Afentis استفاده نمود.

Remote Procedure Calls

(RPC)

با استفاده از RPC برنامه های موجود بر روی یک کامپیوتر قادر به اجرای روتین هائی در کامپیوتر دوم از طریق ارسال داده و بازیابی نتایج می باشند . با توجه به جایگاه عملیاتی RPC ، استفاده از آن بسیار متداول بوده و درموارد متعددی از آن به منظور ارائه سرویس های توزیع شده شبکه نظیر مدیریت از راه دور ، اشتراک فایل NFS و NIS استفاده می گردد.وجود ضعف های امنیتی متعدد در RPC باعث بهره برداری مهاجمان به منظور انجام حملات مختلفی شده است .دراکثر موارد ، سرویس های RPC با مجوزهای بیش از حد معمول ، اجراء می گردند . بدین ترتیب یک مهاجم غیر مجاز قادر به استفاده از سیستم های آسیب پذیر در جهت اهداف خود خواهد بود.اکثر حملات از نوع DoS در سال 1999 و اوایل سال 2000 در ارتباط با سیستم هائی بود که دارای ضعف امنیتی و نقظه آسیب پذیر RPC بودند. مثلا" حملات گسترده و موفقیت آمیز در رابطه با سیستم های نظامی امریکا ، بدلیل نقطه آسیب پذیر RPC کشف شده در صدها دستگاه کامپیوتر مربوط به وزارت دفاع امریکا بوده است . اخیرا" نیز وجود یک ضعف امنیتی RPC DCOM در ویندوز ، باعث انتشار گسترده یک کرم در سطح اینترنت گردید .

سیستم های عامل در معرض تهدید :

تمامی نسخه های یونیکس و لینوکس که بر روی آنان سرویس های RPC نصب شده است در معرض این آسیب می باشند .

نحوه تشخیص آسیب پذیری سیستم

با استفاده از یک پویشگر نقاط آسیب پذیر و یا دستور " rpcinfo" ، می توان از اجراء یکی از سرویس های متداول RPC بر روی سیستم آگاه گرید :

RPC Program Number	RPC Service
100083	rpc.ttdbserverd
100068	rpc.cmsd
100024	rpc.statd
100005	rpc.mountd
100232	Sadmind
100235	Cachefsd
100249	SnmpXdmid

سرویس های RPC ، عموما" از طریق حملات buffer Overflow ، مورد سوء استفاده قرار می گیرند .علت این امر ، عدم انجام بررسی لازم و کافی در خصوص خطاها و یا اعتبار داده های ورودی توسط برنامه های RPC است . نقاط آسیب پذیر Buffer overflow ، این امکان را برای یک مهاجم فراهم می نماید که داده غیر قابل پیش بینی را (اغلب بصورت کد مخرب) به درون حافظه برنامه ، ارسال نماید . با توجه به ضعف موجود در رابطه با بررسی خطاء و صحت داده ، داده ارسالی مکان هایی حساس و کلیدی که مورد استفاده پردازنده می باشند را بازنویسی می نماید.در یک تهاجم موفقیت آمیز Overflow ، کد مخرب ارسالی ،در ادامه توسط سیستم عامل اجراء می گردد . با توجه به اینکه تعداد زیادی از سرویس های RPC ، با مجوزهای بیش از حد معمول ، اجراء می گردند ، استفاده موفقیت آمیز از نقاط آسیب پذیر فوق می تواند امکان دسـیابی غیر مجاز و از راه دور را به سیستم فراهم می نماید.

نحوه حفاظت در مقابل نقطه آسیب پذیر

بمنظور حفاظت سیستم در مقابل حملات مبتنی بر RPC ، موارد زیر پیشنهاد می گردد :

- غیر فعال نمودن و یا حذف هر یک از سرویس های RPC به استفا

Apache Web Server

آپاچی (Apache) یکی از متداولترین سرویس دهندگان وب بر روی اینترنت است . در مقایسه با سرویس دهنده های مایکروسافت (IIS) با سرویس دهنده های (IIS) با سرویس دهنده های (IIS) با سرویس دهنده های (IIS)

نحوه تشخیص آسیب پذیری سیستم

بمنظور آگاهی و کسب اطلاعات لازم در خصوص نحوه تشخیص آسیب پذیری سرویس دهنده وب آپاچی ، می توان از آدرس های زیر استفاده نمود :

- در رابطه با Apache 1.3.x را می توان از آدرس

http://www.apacheweek.com/features/security-13

- برای Apache 2.0.x می توان از آدرس

http://www.apacheweek.com/features/security-20

آدرس های اشاره شده ، دارای اطلاعات فنی لازم بمنظور نحوه تشخیص آسیب پذیری سیستم و پیشنهادات لازم در خصوص ارتقاء وضعیت امنیتی می باشند . استفاده از آدرس: http://httpd.apache.org/ نیز در این زمینه مفید است .

نحوه حفاظت در مقابل نقطه آسیب پذیر

بمنظور حفاظت یک سرویس دهنده وب آپاچی ، پیشنهادات زیر ارائه می گردد :

- اطمینان از نصب آخرین patch ارائه شده

- در این رابطه می توان از آدرس http://httpd.apache.org/ بمنظور آگاهی از آخرین وضعیت نسخه ها و levels Patch استفاده نمود.

- بمنظور دستیابی به code Source اکثر نسخه های آپاچی، می توان از آدرس http://httpd.apache.org/download.cgi استفاده نمود.
- بمنظور آگاهی و دریافت آخرین Patch های ارائه شده می توان از آدرس http://www.apache.org/dist/httpd/patches/ استفاده نمود.

- اطمینان از patching عناصر کلیدی سیستم عامل که آپاچی بعنوان مرجع از آنان استفاده می نماید .در این رابطه لازم است که صرفا" ماژول های ضروری به منظور صحت عملکرد سرویس دهنده ، در آپاچی کمپایل گردند .لازم است به این نکته اشاره

گردد که کرم **mod_ssl** (CA-2002-27) نمونه ای کامل در این زمینه بوده که از نقاط آسیب پذیر در **OpenSSL** (CA-2002-23) استفاده نموده است .

- از اجرای آپاچی بعنوان ریشه ، اجتناب کنید و می بایست بدین منظور ، کاربر و یا گروهی خاص با حداقل مجوز ایجاد گردد. سایر پردازه های سیستم ضرورتی به اجراء تحت کاربر و یا گروه فوق را نخواهند داشت .
- Chroot ، پتانسیلی است که باعث تعریف مجدد محدوده یک برنامه می گردد . در حقیقت chroot ، باعث تعریف مجدد دایرکتوری "ROOT و یا "/" برای یک برنامه و یا یک Login session می گردد .chroot می تواند بعنوان یک لایه تدافعی استفاده گردد . مثلا" در صورتیکه فردی به کامپیوتر شما دستیابی پیدا نماید ، قادر به مشاهده تمامی فایل های موجود بر روی سیستم نخواهد بود . علاوه بر محدودیت فوق ، محدودیت هائی در خصوص اجرای برخی از دستورات نیز بوجود می آید.در این رابطه یک دایرکتوری با نام /chroot ، ایجاد و تمامی سرویس های مورد نظر با یک انضباط خاص در آن مستقر می گردند . مثلا" سرویس دهنده آپاچی در chroot/httpd / قرار می گیرد. با توجه به موارد فوق ، می بایست آپاچی را در یک محیط chroot اجراء نمود . درصورتیکه آپاچی بصورت chrooted اجراء و فعالیت خود را آغاز نماید ، امکان دستیابی آن به سایر بخش های موجود در ساختار دایرکتوری سیستم عامل و خارج از chroot وجود نخواهد داشت . بدین ترتیب یک لایه تدافعی مناسب در خصوص سوء استفاده های احتمالی ایجاد می گردد. به عنوان نمونه ، ممکن است یک shell فراخوانده شده و با توجه به اینکه bin/sky / در chroot قرار ندارد ، می تواند زمینه سوء استفاده احتمالی را فراهم نماید. لازم است به این نکته مهم نیز اشاره گردد که Chrooting آپاچی می تواند اثرات جا نبی نامطلوبی را در ارتباط با CGI,PHP ، بانک های اطلاعاتی و سایر ماژول ها و یا ارتباطاتی که محیط سرویس دهنده وب بمنظور سرویس دهی به آنان نیازمند دستیابی به توابع کتابخانه ای خارجی است را بدنبال داشته باشد .روش های متعددی بمنظور chrooting وجود داشته و می بایست از مستندات نرم افزار مورد نظر ، بعنوان یک منبع اطلاعاتی مناسب در خصوص ارائه راهکارهای مربوطه ، استفاده گردد
- بمنظور مدیریت یک سرویس دهنده وب ، لازم است فیدبک های لازم در خصوص فعالیت و کارآیی سرویس دهنده و سایر مسایلی که ممکن است یک سرویس دهنده با آنان برخورد نماید را اخذ و در ادامه با آنالیز آنان تمهیدات لازم در خصوص مسائل موجود را بکار گرفت . سرویس دهنده آپاچی ، قابلیت ها و پتانسیل های انعطاف پذیری را در خصوص logging ارائه می نماید . بنابراین لازم است عملیات logging

با دقت نظر بالا بصورت موثر و موشکافانه انجام تا امکان ردیابی هر نوع فعالیت امنیتی غیر مجاز و یا رفتار غیر منطقی سرویس دهنده ، فراهم گردد .پیشنهاد می گردد که با یک نظم خاص از اطلاعات موجود در فایل های لاگ ، آرشیو تهیه شود . بدین ترتیب ، امکان مدیریت فایل های لاگ و بررسی آنان فراهم خواهد شد. بمنظور آشنائی با فرمت های متفاوت لاگ می تواند از منابع زیر استفاده نمود :

- برای Apache 1.3.x از آدرس http://httpd.apache.org/docs/logs.html استفاده شود .

- برای Apache 2.0.x از آدرس http://httpd.apache.org/docs-2.0/logs.html استفاده شود .

در موارد متفاوتی و با توجه به شرایط پیش آمده ممکن است محتوی فایل های لاگ به تنهایی کافی نباشد . وضعیت فوق در مواردیکه از PHP ، CGI و یا سایر تکنولوژی های مبتنی بر اسکریپت استفاده می گردد ، تشدید می شود و می توان بمنظور افزایش توان آنالیز یک تهاجم و سوءاستفاده از یک ضعف امنیتی ، اقدام به ثبت لاگ های مربوط به GET و POST نمود.لاگ نمودن عملیات مرتبط به GET و POST می تواند از طریق

- استفاده از ماژولهای اضافه بمنظوربهبود وضعیت امنیتی. مثلا"ماژول mod_Security می تواند باعث حفاظت در مقابل Cross Site Scripting: XSS ، شود . برای آشنائی و مشاهده اطلاعات تکمیلی در این خصوص می توان از آدرس http://www.modsecurity.org/ استفاده نمود.
- ممیزی و بررسی اسکریپت ها برای نقاط آسیب پذیر شامل SQL & XSS Injection نیز حائز اهمیت است . در این رابطه می توان از ابزارهای متعددی استفاده نمود. نرم افزار Nikto (قابل دسترس در آدرس http://www.cirt.net/code/nikto.shtml) یکی از مناسبترین ابزارهای پویش و بررسی CGI است .

account ها

با رمز عبور ضعیف و یا فاقد رمز عبور

استفاده از رمزعبور، روش های تائید کاربر و کدهای امنیتی در هر گونه تعامل ارتباطی بین کاربران وسیستم های اطلاعاتی ، امری متداول و رایج است . اکثر روش ها ی تائید کاربران ، نظیر حفاظت فایل و داده ، مستقیما" به رمزهای عبور ارائه شده توسط کاربران ، بستگی خواهد داشت . پس از تائید کاربران ، امکان دستیابی آنان به منابع مشخص شده فراهم و هر یک از آنان با توجه به امتیازات و مجوزهای نسبت داده شده ، قادر به استفاده از منابع موجودخواهند بود. در اغلب موارد ، فعالیت کاربرانی که مجاز بودن آنان برای دستیابی به منابع ، تائید شده است ، لاگ نشده و یا در صورتیکه فعالیت آنان ثبت گردد ، کمتر سوء ظنی به آنان می تواند وجود داشته باشد . (آنان پس از تائید وارد میدانی شده اند که بدون هیچگونه ردیابی ، قادر به انجام فعالیت های گسترده ای خواهند بود) . بنابراین ، رمز عبور دارای نقشی حیاتی و اساسی در ایجاد اولین سطح دفاع در یک سیستم اطلاعاتی بوده و از دست رفتن رمز عبور و یا ضعف آن می تواند سیستم را در معرض تهدیدات جدی قرار دهد . مهاجمان پس از دستیابی به رمز عبور کاربران تائید شده (استفاده از مکانیزم های متفاوت) قادر به دستیابی منابع سیستم و حتی تغییر در تنظیمات سایر account های تعریف شده و موجود بر روی سیستم خواهند بود،عملیاتی که می تواند پیامدهای بسیار منفی را بدنبال داشته باشد . پس می بایست بپذیریم که وجود یک account ضعیف و یا فاقد رمز عبور می تواند تهدیدی جدی در یک سازمان باشد . در این راستا علاوه بر اینکه می بایست از پتانسیل های ارائه شده توسط سیستم عامل با دقت استفاده نمود ، ضروری است ، تابع یک سیاست امنیتی تدوین شده در رابطه با رمز عبور در سازمان متبوع خود باشیم . تعریف و نگهداری یک account بهمراه رمز عبور مربوطه در سازمان ما تابع چه سیاست امنیتی است ؟ مهمترین و متداولترین نقاط آسیب پذیر در ارتباط با رمز عبور شامل موارد زیر است :

- Account تعریف شده دارای رمز عبور ضعیف و یا فاقد رمز عبور است .
- عدم حفاظت مناسب کاربران از رمزهای عبور ،صرفنظر از استحکام رمزهای عبور تعریف شده .
- سیستم عامل و یا سایر نرم افزارهای موجود ، امکان ایجاد account مدیریتی ضعیف و فاقد رمز عبور را فراهم می نمایند .
- الگوریتم های Hashing رمز عبور(رمزنگاری مبتنی بر کلید عمومی بر پایه یک مقدار hash ، استوار بوده و بر اساس یک مقدار ورودی که دراختیار الگوریتم hashing گذاشته می گردد ، ایجاد می گردد. در حقیقت مقدار hash ، فرم خلاصه شده و رمز شده ای از مقدار اولیه خود است) ، شناخته شده بوده و در اغلب موارد مقدار Hashe بدست آمده ، بگونه ای ذخیره می گردد که امکان مشاهده آن توسط سایرین وجود خواهد داشت. مناسبترین نوع حفاظت در این راستا ، تبعیت از یک سیاست رمز عبور قدرتمند بوده که در آن دستورالعمل ها ی لازم برای تعریف یک رمز عبورمناسب مشخص و در ادامه با استفاده از ابزارهای موجود، بررسی لازم در خصوص استحکام و بی نقص بودن رمز عبور صورت گیرد.

سیستم ها ی در معرض آسیب پذیر

هر سیستم عامل و یا برنامه ای که فرآیند تائید کاربران آن براساس یک User ID و رمز عبور باشد ، در معرض این تهدید خواهد بود.

نحوه تشخیص آسیب پذیری سیستم

در صورتیکه از account هائی استفاده می شود که بین کاربران متعدد و یا کارکنان موقت یک سازمان به اشتراک گذاشته شده و یا کاربران از رمزهای عبور بدرستی حفاظت ننمایند، پتانسیل نفوذ به شبکه توسط یک مهاجم فراهم می گردد.پیکربندی account های جدید کاربران با یک رمز عبور مشابه و یا رمز عبوری که بسادگی قابل حدس باشد نیز فرصتی مناسب را در اختیار مهاجمان بمنظور دستیابی به منابع اطلاعاتی موجود در یک سازمان قرار خواهد داد .

لازم است در خصوص ذخیره سازی رمز عبور hashes تصمیم گیری و مشخص شود که محل استقرار و ذخیره سازی آنان در etc/passwd / و یا etc/shadow / می باشد.قابلیت

خواندن فایل etc/passwd /، می بایست توسط تمامی کاربران شبکه وجود داشته تا زمینه و امکان تائید کاربران فراهم گردد. در صورتیکه فایل فوق ، شامل رمزعبور hashed نیز باشد ، در ادامه و پس از دستیابی کاربران به سیستم ، امکان خواندن مقادیر hash فراهم و مهاجمان می توانند با استفاده از یک برنامه cracker ، تلاش خود را جهت شکستن و تشخیص رمز عبور آغاز و به سرانجام برسانند . فایل etc/shadow/ ، صرفا" برای root قابل خواندن بوده و مکانی مناسب بمنظور ذخیره نمودن مقادیر hashes است . در صورتیکه account های محلی ، توسط etc/shadow/ حفاظت نشود ، ریسک رمزهای عبور افزایش خواهد یافت . اکثر سیستم های عامل جدید بصورت پیش فرض از etc/shadow / بمنظور ذخیره سازی رمز عبور hashes استفاده می نمایند (مگر اینکه شرایط فوق توسط نصب کننده تغییر یابد). در این رابطه می توان از الگوریتم MD5 بمنظور hash نمودن رمزهای عبور نیز استفاده نمود. الگوریتم فوق، بمراتب از الگوریتم قدیمی crypt ایمن تر است .

NIS (Network Information System) ، یک بانک اطلاعاتی توزیع شده بمنظور مدیریت یک شبکه است . در حقیقت NIS ، استانداردی برای اشتراک فایل ها بین سیستم های کامپیوتری متعدد را فراهم و شامل مجموعه ای از سرویس هائی است که بمنزله یک بانک اطلاعاتی از سرویس ها عمل نموده و اطلاعات مربوط به مکان سرویس (Mapping) را در اختیار سایر سرویس های شبکه نظیر(Network File System (NFS) ، قرار می دهد. با توجه به ماهیت طراحی بعمل آمده ، فایل های پیکربندی NIS ، شامل رمزهای عبور hash بوده و این امر می تواند امکان خواندن آنان را برای تمامی کاربران فراهم و عملا" رمزهای عبور در معرض تهدید قرار گیرند .نسخه های جدید پیاده سازی شده از NIS ، نظیر +NIS و یا LDAP عموما" دارای استحکام لازم در ارتباط با رمزهای عبور hashes می باشند(مگر اینکه شرایط فوق توسط نصب کننده تغییر یابد). تنظیم و پیکربندی نسخه های فوق (نسخه های جدید) ، مشکل تر بوده و همین امر می تواند استفاده از آنان را با تردید و مشکل مواجه نماید .

حتی اگر رمزهای عبور hashes توسط etc/shadow/ و یا امکانات پیاده سازی شده ، محافظت گردند ، امکان حدس و تشخیص رمزهای عبور توسط سایر افراد وجود خواهد داشت . در این رابطه می توان به موارد متعدد دیگری نظیر : ضعف رمز عبور ، وجود account های غیر استفاده مربوط به کارکنانی که سازمان خود را ترک نموده اند ، اشاره نمود .سازمان ها معمولا" در رابطه با غیر فعال نمودن account مربوط به کاربران قدیمی کوتاهی نموده و لازم است در این رابطه از روش های خاصی استفاده گرد. نصب های پیش فرض سیستم های عامل و یا شبکه توسط سازندگان و یا مدیران سیستم و یا شبکه ، می تواند نصب مجموعه ای از سرویس های غیرضروری را نیز بدنبال داشته

باشد. رویکرد فوق،با اینکه عملیات نصب سیستم عامل و سرویس ها ی مربوطه را تسهیل می نماید ولی مجموعه ای از سرویس های غیر ضروری و account هائی که بصورت پیش فرض ضعیف ویا فاقد رمز عبور می باشند را بهمراه بر روی سیستم مستقر و پیکربندی می نماید.

نحوه حفاظت در مقابل نقطه آسیب پذیر

بهترین و مناسبترین دفاع در مقابل ضعف رمزهای عبور ، تبعیت از یک سیاست امنیتی مستحکم بوده که دستورالعمل های لازم کهه موجب می شود رمزهای عبور مناسب و مستحکمی توسط کاربران تعریف و توسط مدیران سیستم بصورت مستمر پیوستگی و استحکام آنان بررسی می گردد با حمایت کامل سازمان . مراحل زیر توصیه های لازم برای ارائه یک سیاست امنیتی مناسب می باشد :

- اطمینان ازاستحکام و انسجام رمز های عبور . با استفاده از سخت افزار مناسب و اختصاص زمان کافی ، می توان هر رمز عبوری را crack نمود. در این راستا می توان با استفاده ازروش های ساده و در عین حال موفقیت آمیز، عملیات تشخیص رمز عبور را انجام داد . اغلب برنامه های تشخیص دهنده رمزعبوراز روشی موسوم به "حملات مبتنی بر سبک دیکشنری " ، استفاده می نمایند. با توجه به اینکه روش های رمز نگاری تا حدود زیادی شناخته شده می باشند ، برنامه های فوق ، قادر به مقایسه شکل رمز شده یک رمز عبور در مقابل شکل های رمز شده کلمات دیکشنری می باشند(در زبان های متعدد و استفاده از اسامی مناسب بهمراه جایگشت های مختلف آنان) . بنابراین ، رمز عبوری که ریشه آن در نهایت یک کلمه شناخته شده باشد ، دارای استعداد ذاتی در رابطه با این نوع از حملات خواهد بود . تعداد زیادی از سازمان ها ، آموزش های لازم در خصوص نحوه تعریف رمزهای عبور را به کارکنان خود داده و به آنان گفته شده است که رمزهای عبور مشتمل بر ترکیبی از حروف الفبائی و کاراکترهای ویژه را برای خود تعریف نمایند.متاسفانه اکثر کاربران این موضوع را رعایت ننموده و بمنظور تعریف یک رمز عبور با نام " password" ، صرفا" اقدام به تبدیل حروف به اعداد و یا حروف ویژه می نمایند (pa$$w0rd) . چنین جایگشت هائی نیز قادر به مقاومت در مقابل یک تهاجم مبتنی بر دیکشنری نبوده و pa$$w0rd"" به روش مشابهی که password"" تشخیص داده می شود ، crack خواهد شد .

یک رمز عبور خوب ، نمی بایست از ریشه یک کلمه و یا نام شناخته شده ای اقتباس شده باشد .در این راستا لازم است به کاربران آموزش لازم در خصوص انتخاب و ایجاد رمزهای عبور از موارد تصادفی نظیر یک عبارت ، عنوان یک کتاب ،نام یک آواز و یا نام یک فیلم داده شود. با انتخاب یک رشته طولانی که بر اساس رویکردهای خاصی می تواند انتخاب گردد(گرفتن اولین حرف هر کلمه ، جایگزینی یک کاراکتر خاص برای یک کلمه ، حذف تمامی حروف صدادارو سایر موارد) ، کاربران قادر به ایجاد رمزهای عبور مشتمل بر ترکیبی از حروف الفبائی و حروف ویژه بوده که در صورت مواجه شدن با حملات مبتنی بر دیکشنری ، تشخیص آنان بسختی انجام می شود. لازم است به این نکته نیز اشاره گردد که رمزعبور می بایست براحتی بخاطر سپرده شده و بازیابی (یادآوری)آن مشکل نباشد (هدف از ذخیره سازی ، بازیابی است اگر چیزی را ذخیره نمائیم ولی در زمان مورد نظر قادر به بازیابی آن نباشیم ، سیستم ذخیره و بازیابی ما با اشکال مواجه شده است !). پس از تدوین دستورالعمل لازم بمنظور تولید رمزهای عبور مناسب و آموزش کاربران بمنظور پایبندی به اصول امنیتی تعریف شده ، می بایست از روتین ها ی جانبی متعددی بمنظور اطمینان از پیروی کاربران از دستوراالعمل های اعلام شده ، استفاده گردد. بهترین گزینه در این راستا ، بررسی صحت رمزهای عبور پس از اعمال تغییرات توسط کاربران است .

پس از ارائه دستورالعمل ها ی لازم و مناسب برای ایجاد رمزهای عبور ، روتین های تکمیلی خاصی می بایست ایجاد تا این اطمینان حاصل گردد که کاربران پایبند به دستورالعمل های ارائه شده بوده اند. بهترین روش در این زمینه ، بررسی صحت اعتبار رمزهای عبور پس از اعمال تغییرات توسط کاربران است . اکثر نمونه های یونیکس و لینوکس می توانند از Npasswd بمنظور بررسی رمز عبور در مقابل سیاست امنیتی موجود استفاده نمایند. سیستم های PAM-Enabled نیز می توانند از Cracklib (کتابخانه لازم بمنظور هماهنگی با Crack) بمنظور بررسی رمزهای عبور ایجاد شده ، استفاده نمایند.اکثر سیستم های PAM-enabled را می توان بگونه ای پیکربندی نمود که رمزهای عبوری را که با سیاست های مشخص شده مطابقت ندارد ، رد نمایند . درمواردیکه امکان استفاده از ابزارهائی نظیر Npasswd و یا کتابخانه های PAM-Enabled ، وجود ندارد، مدیران سیستم و شبکه می توانند از برنامه های کاربردی Cracking در حالت stand-alone و بعنوان یک روتین کنشگرایانه مستمر، استفاده نمایند. l0phtcrack version (4 LC4) و John the Ripper ، نمونه هائی از برنامه های فوق ، می باشند. لازم است مجددا" به این موضوع اشاره گردد که بدون کسب مجوز لازم از مدیران ارشد سیستم در سازمان ، نمی بایست از برنامه های cracking استفاده

گردد.پس از کسب مجوزهای لازم ، می توان عملیات فبررسی رمزهای عبور را بر روی یک ماشین حفاظت شده انجام داد. به کاربرانی که رمزهای عبور آنان crack می گردد، بصورت محرمانه وضعیت فوق گزارش و دستورالعمل های لازم در خصوص نحوه انتخاب یک رمز عبور مناسب نیز به آنان ارائه گردد .اخیرا" و در پاسخ به رمزهای عبور ضعیف ، استفاده از روش هائی دیگر بمنظور تائید کاربران، نظیر بیومتریک (زیست سنجی) ، نیز مورد توجه واقع شده است .

- حفاظت رمزهای عبور مستحکم . در صورتیکه رمزهای عبور hashes در etc/passwd / ذخیره می گردند ، سیستم را بهنگام نموده تا از /etc/shadow استفاده گردد . در صورتیکه بر روی سیتستم NIS و یا LDAP اجراء که امکان حفاظت hashes وجود نداشته باشد ، هر کاربری قادر به خواندن رمزهای عبور hashes و تلاش بمنظور cracking آنان ، خواهد بود.در این رابطه می بایست بررسی لازم در خصوص استفاده از گزینه های ایمن تری از نسخه های NIS و LDAP را انجام داد. تا زمانیکه این نوع برنامه های غیر ایمن وجود داشته و با نمونه های ایمن جایگزین نشده اند، می بایست مجوزهای مربوطه را ایمن و از ابزارهای کنشکرایانه بصورت مستمر استفاده گردد.در این رابطه پیشنهاد می گردد که در مقابل استفاده از الگوریتم قدیمی Crypt بمنظورhash نمودن رمزهای عبور از الگوریتم MD5 استفاده گردد.
حتی اگر رمزهای عبور ، مستحکم و قدرتمند باشند ، در صورت عدم حفاظت آنان توسط کاربران ، سیستم های موجود در یک سازمان در معرض تهدید قرار خواهند گرفت . یک سیاست امنیتی مناسب ، می بایست شامل دستورالعمل های لازم بمنظور آموزش کاربران در رابطه با حفاظت رمزهای عبور می باشد.عدم ارائه رمز عبور به افراد دیگر، عدم نوشتن رمز عبور در محلی که امکان خواندن آن برای دیگران وجود داشته باشد و حفاظت اتوماتیک فایل هائی که رمزهای عبور در آن ذخیره شده اند ، از جمله مواردی می باشند که می بایست به کاربران آموزش داده شود. اغلب کاربران در مواجهه با پیامی مشابه "Your password has expired" که نشاندهنده اتمام عمر مفید یک رمز عبور است ، یک رمز عبور ضعیف را برای خود انتخاب می نمایند ، بنابراین لازم است در فرصت مناسب و قبل از برخورد با اینچنین پیام هائی ، به کاربران آموزش های لازم ارائه گردد.
- کنترل دائم و پیوسته accounts . هر account مدیریتی و یا مبتنی بر سرویس که از آن استفاده نمی گردد، می بایست غیر فعال و یا در صورت امکان از روی سیستم حذف گردد. هر account مدیریتی و یا مبتنی بر سرویس که از آن استفاده می گردد

، می بایست دارای رمزعبورجدید و مستحکمی باشد. پیکربندی account های جدید کاربران با رمزهای عبور اولیه (تولیده شده بصورت تصادفی) و ضرورت تغییر رمزهای عبور توسط کاربران و در اولین log in نیز می تواند در این زمینه مفید واقع شود. ممیزی account ها بر روی سیستم را انجام و لازم است در این رابطه یک لیستی اصلی ایجاد گردد .در این رابطه می بایست رمزهای عبور در ارتباط با سیستم هائی نظیر روترها ، چاپگرهای دیجیتالی متصل شده به اینترنت و سایر موارد دیگر نیز مورد بررسی قرار گرفته و روتین هائی خاص بمنظور افزودن account های تائید شده به لیست و یا حذف account هائی که ضرورتی به استفاده از آنان نمی باشد ، پیاده سازی و همواره خود را پایبند به آن بدانیم .اعتبار لیست را در فواصل زمانی خاصی بررسی تا از بهنگام بودن آن اطمینان حاصل گردد.از روتین های خاصی بمنظورحذف account متعلق به کارکنان و یا پیمانکارانی که سازمان را ترک نموده اند ، استفاده گردد .

Clear Text Services

تعداد زیادی از سرویس های شبکه استفاده شده توسط سیستم های مبتنی بر یونیکس ، بصورت text plain بوده و از رمزنگاری خاصی استفاده نمی نمایند. ضعف در رمزنگاری ، امکان شنود اطلاعاتی (مشاهده ترافیک شبکه) را فراهم و مهاجمان در ادامه امکان دستیابی به محتوی‌ات ارتباط ایجاد شده و یا اطلاعات حساس کاربران نظیر داده های مرتبط با رمز عبور را بدست خواهند آورد. مثلا" بمنظور تشخیص اطلاعات مرتبط با FTP و یا telnet (اطلاعات login) ، یک مهاجم ، می تواند یک sniffer را در شبکه و در محلی بین مسیر ارتباطی، مستقر نماید (بعنوان سرویس دهنده FTP و یا یا LAN Client) . تبادل اطلاعات بین دستورات ورودی توسط سرویس گیرندگان و پاسخ های ارائه شده توسط سرویس های موجود بر روی سرویس دهنده ، بصورت plain-text خواهد بود. بدین ترتیب ،امکان ردیابی (رهگیری) داده و یا کلیدهای فشرده شده توسط کاربر بسادگی فراهم می گردد. اغلب مهاجمان در حملات اخیر خود از برنامه های مختص شنود اطلاعاتی استفاده و عمدتا" اینگونه برنامه ها را بر روی دستگاهها ی آسیی‌ب پذیر نصب می نمایند. در چنین مواردی ، تشخیص نام و رمز عبور کاربر در داده های جمع آوری شده (شنود اطلاعاتی) ، بسادگی میسر خواهد شد.جدول زیر، لیست برخی از سرویس های شبکه یونیکس را که اطلاعات را بصورت Clear Text ، مبادله می نمایند ، نشان می دهد :

سرویس	پورت	Clear Content	Clear Auth	آیتم ارسالی
FTP	21 , 20	Yes	Yes	متن ، باینری
TFTP	69	Yes	N/A	متن ، باینری
telnet	23	Yes	Yes	متن
SMTP	25	Yes	N/A	متن ، باینری
POP3	110	Yes	Yes	متن ، باینری
IMAP	143	Yes	Yes	متن ، باینری
rlogin	513	Yes	Yes	متن
rsh	514	Yes	Yes	متن
HTTP	80	Yes	Yes	متن ، باینری

سرویس هائی نظیر Telnet و FTP که شامل اطلاعات مربوط به تائید هویت کاربران می باشند ، اطلاعات مورد نظر را بصورت متن ارسال که بالاترین ریسک را بدنبال خواهد داشت . در چنین مواردی مهاجمان می توانند با استفاده مجدد از داده های حساس کاربران نظیر نام و رمز عبور، با خیال راحت ! به سیستم دستیابی نمایند .علاوه براین ، اجرای دستورات بصورت clear text می تواند توسط مهاجمان استفاده تا با استفاده از آنان دستورات دلخواه خود را بدون الزامی برای تائید ، اجراء نمایند. سرویس های Clear text ، می توانند زمینه تهدیدات مختلفی نظیر : تسهیل در انجام حملات ، دستیابی از راه دور ، اجرای دستورات بر روی یک سیستم مقصد ، شنود و تشخیص اطلاعات را بدنبال داشته باشد .

سیستم های عامل در معرض تهدید

تمامی نمونه های یونیکس ارائه شده ، از سرویس ها ی Clear text استفاده می نمایند. (telnet و FTP دو نمونه متداول در این زمینه می باشند) . تمامی نمونه نسخه های ارائه شده یونیکس و لینوکس (تنها استثناء در این رابطه، مربوط به آخرین ویرایش

Free/OpenBSD می باشد) ، بصورت پیش فرض ، برخی از سرویس های فوق را نصب می نمایند .

نحوه تشخیص آسیب پذیری سیستم

موثرترین و مطمئن ترین روش بمنظور تشخیص نقطه آسیب پذیر فوق (سرویس های clear text) ، بکارگیری نرم افزاری مشابه با ابزارهای استفاده شده (sniffer) توسط مهاجمان است. متداولترین برنامه موجود در این زمینه ، tcpdump می باشد. برای دریافت برنامه " tcpdump " می توان از آدرس http://www.tcpdump.org/ ، استفاده نمود . نحوه فعال نمودن برنامه فوق،بمنظور تشخیص هر گونه ارتباط clear text ، بصورت زیر است :

```
# tcpdump -X -s 1600
```

دراین راستا می توان از برنامه های دیگر نظیر : "ngrep" نیز استفاده نمود. برنامه فوق،امکان جستجوی الگوئی خاص نظیر "sername " و یا " assword " را در شبکه فراهم می نماید. (اولین حروف، بمنظور سازگاری با حروف بزرگ احتمالی ، در نظر گرفته نشده است) . برای دریافت برنامه فوق ، می توان از آدرس http://www.packetfactory.net/projects/ngrep/ استفاده نمود . نحوه فعال نمودن برنامه فوق ، بصورت زیر است :

```
# ngrep assword
```

در این رابطه می توان از ابزارهای متنوع دیگری بمنظور تشخیص داده های حساس کاربران (نام و رمز عبور) استفاده نمود . Dsniff ، متداولترین ابزار در این زمینه است . برنامه فوق ، بررسی لازم در خصوص تمامی زوج نام و رمزعبور بر روی پروتکل های Plain text حجیم، نظیر FTP ، Telnet و POP3 را انجام و پس از تشخیص، آنان را نمایش خواهد داد. برای بدست آوردن برنامه فوق می توان از آدرس http://www.monkey.org/~dugsong/dsniff/ استفاده نمود . نحوه فعال نمودن برنامه فوق ، بصورت زیر است :

```
# /usr/sbin/dsniff
```

نحوه حفاظت در مقابل نقطه آسیب پذیر

استفاده از رمزنگاری End-To-End و یا حداقل رمزنگاری Link-level می تواند در این زمینه مفید واقع گردد. برخی پروتکل ها از لحاظ رمزنگاری معادل یکدیگر می باشند(نظیر : POP3S و HTTPS) . برای پروتکل هائی که دارای قابلیت ها و امکانات ، ذاتی رمزنگاری نمی باشند ، می توان آنان را از طریق SSH : Secure Shell ، و یا connection SSL انجام داد (tunneling).

OpenSSH یک نمونه پیاده سازی شده متداول و انعطاف پذیر ازSSH است.(قابل دسترس در آدرس http://www.openssh.org/) . برنامه فوق، در اکثر نسخه های یونیکس اجراء و می توان از آن بمنظور ارتباطات از راه دور (replaces telnet ,rlogin ,rsh) و tunneling (پروتکل هائی نظیر POP3,SMTP و X11) استفاده گردد .

دستور زیر نحوه tunnel نمودن POP3 را بر روی connection SSH نشان می دهد. بر روی سرویس دهنده POP3 ، می بایست سرویس دهنده SSH نیز اجراء گردد. در ابتدا آن را بر روی ماشین سرویس گیرنده اجراء می نمائیم :

```
# ssh -L 110:pop3.mail.server.com:110 username@pop3.mail.server.com
```

در ادامه ، برنامه سرویس گیرنده پست الکترونیکی را به localhost اشاره می دهیم ، پورت 110 TCP (برخلاف روال معمول که بصورت : pop3.mail.server.com ، پورت 110 است) . بدین ترتیب ، تمامی ارتباطات بین ماشین و سرویس دهنده پست الکترونیکی بصورت رمز شده انجام خواهد شد . (tunneled over SSH)

یکی دیگر از راه حل های متداول رمزنگاری مبتنی بر tunneling ، استفاده از stunnel است . روش فوق ، پروتکل SSL را پیاده سازی(با استفاده از OpenSSL Toolkit) و می توان آن را بمنظور tunel نمودن پروتکل های متفاوت plain text بخدمت گرفت . برای دریافت برنامه فوق ، می توان از آدرس http://www.stunnel.org/ استفاده نمود.

Sendmail

Sendmail ، برنامه ای است که از آن بمنظور ارسال ، دریافت و فوروارد نمودن نامه های الکترونیکی در اغلب سیستم های یونیکس و لینوکس استفاده می گردد. Sendmail ، یکی از متداولترین MTA : Mail Transfer Agent در اینترنت بوده که بطور گسترده ای از آن بعنوان "آژانش توزیع نامه های الکترونیکی" بهمراه سرویس دهندگان پست الکترونیکی ، استفاده می گردد . Sendmail ، یکی از اهداف اولیه مهاجمان در سالیان اخیر بوده و تاکنون حملات متعددی را در ارتباط با آن شاهد بوده ایم. اکثر حملات انجام شده بدلیل قدیمی بودن و یا عدم patch مناسب نسخه های نصب شده ، با موفقیت همراه بوده است .در این رابطه می توان به چندین نمونه از حملات اخیر اشاره نمود :

- CERT Advisory CA-2003-12 Buffer Overflow in Sendmail
- CERT Advisory CA-2003-07 Remote Buffer Overflow in Sendmail
- CERT Advisory CA-2003-25 Buffer Overflow in Sendmail

خطرات و تهدیدات مرتبط با Sendmail را می توان به دو گروه عمده تقسیم نمود : از دست رفتن امتیازات که علت آن buffer overflow درست سیستم که می buffer overflow

- در صورتیکه لازم است sendmail در حالت daemon اجراء گردد، می بایست از صحت پیکربندی انجام شده اطمینان حاصل گردد. در این رابطه می توان از منابع اطلاعاتی زیر استفاده نمود :

http://www.sendmail.org/tips/relaying.html
http://www.sendmail.org/m4/anti_spam.html

در نسخه Sendmail 8.9.0 ، امکان open relay بصورت پیش فرض غیرفعال می باشد.تعداد زیادی از عرضه کنندگان سیستم های عامل ، مجددا" آن رادر پیکربندی پیش فرض خود فعال می نمایند. در صورت استفاده از نسخه Sendmail ارائه شده بهمراه سیستم عامل ، می بایست دقت لازم در اینخصوص را انجام داد (عدم استفاده سرویس دهنده برای .(realying)

- در زمان استفاده از نسخه جدید sendmail (سوئیچ نمودن به یک نسخه دیگر) ، لازم است تدابیر لازم در خصوص تغییر فایل های پیکربندی ارائه شده توسط نسخه قدیمی ، اندیشیده گردد . برای آگاهی از جزئیات بیشتر در ارتباط با پیکربندی Sendmail ، می توان از آدرس http://www.sendmail.org/m4/readme.html استفاده نمود.
- در مواردیکه برنامه Sendmail از منابع موجود بر روی اینترنت Download می گردد ، می بایست بمنظور اطمینان از مجاز بودن نسخه تکثیری از PGP signature استفاده نمود. در این رابطه لازم است به این نکته دقت شود که بدون بررسی integrity مربوطه به source code برنامه Sendmail ، نمی بایست از آن استفاده شود. در گذشته ، نسخه هائی از Trojan ها در Sendmail مستقر تا در زمان مناسب حرکت مخرب خود را آغاز نمایند . بمنظور دریافت اطلاعات تکمیلی در این رابطه می توان از آدرس CA-2002-28 CERT Advisory استفاده نمود. کلیدهای استفاده شده بمنظور sign نمودن برنامه Sendmail دریافتی را می توان از آدرس http://www.sendmail.org/ftp/PGPKEYS بدست آورد . در صورت فقدان PGP ، می بایست از MD5 Checksum بمنظور بررسی integrity کد منبع Sendmail ، استفاده گردد.

برای کسب اطلاعات بیشتر می توان از منابع اطلاعاتی زیر استفاده نمود :

http://www.sendmail.org/secure-install.html
http://www.sendmail.org/m4/security_notes.html
http://www.sendmail.org/~gshapiro/security.pdf

Simple Network Management Protocol

(SNMP)

از پروتکل SNMP بمنظور کنترل ، مانیتورینگ از راه دور و پیکربندی تمامی دستگاه های پیشرفته مبتنی بر TCP/IP در ابعاد گسترده ای استفاده می شود.با اینکه استفاده از SNMP در بین پلات فرم های متفاوت شبکه استفاده می گردد، ولی در اغلب موارد از آن بمنظور پیکربندی و مدیریت دستگاههائی نظیر چاپگر ، روترها ، سوئیچ ها ، Access point ها و دریافت داده های مورد نیاز دستگاههای مانیتورینگ شبکه ، استفاده می شود . SNMP ، از روش های متفاوتی بمنظور مبادله پیام بین ایستگاههای مدیریت SNMP و دستگاههای شبکه ای استفاده می نماید . روش های استفاده شده بمنظور برخورد با پیام های مبادله شده و مکانیزم تائید و معتبر سازی پیا م ها، از جمله عوامل اصلی در رابطه با نقاط آسیب پذیر SNMP می باشند .

نقاط آسیب پذیر مرتبط با روش های استفاده شده در SNMP (نسخه یک) بهمراه جزئیات مربوطه را می توان در آدرس - CERT 2002 – 03 ، مشاهده نمود . نقاط آسیب پذیر متعددی در SNMP متاثر از روش برخورد با پیام ها توسط ایستگاه های مدیریتی است . نقاط آسیب پذیر فوق، به نسخه ای خاص از SNMP محدود نبوده و محصولات متعدد ارائه شده توسط تولید کنندگان را نیز شامل می گردد . مهاجمان با استفاده از نقاط آسیب پذیر فوق ، قادر به انجام حملات متفاوت از نوع DoS (از کار افتادن یک سرویس) تا پیکربندی و مدیریت ناخواسته ماشین آلات و تجهیزات مبتنی بر SNMP ، می باشند . برخی از نقاط آسیب پذیر در ارتباط با SNMP متاثر از روش های استفاده شده بمنظور تائید و معتبر سازی پیام ها در نسخه های قدیمی SNMP است (توارث مشکلات) . نسخه های یک و دو SNMP ، از یک " رشته مشترک " غیررمز شده بعنوان تنها گزینه موجود برای تائید پیام ها استفاده می نمایند . عدم استفاده از روش های مناسب رمزنگاری ، می تواند عاملی مهم در پیدایش نقاط آسیب پذیر باشد. نگرش پیش فرض نسبت به " رشته مشترک "

که توسط تعداد زیادی از دستگاههای SNMP استفاده می گردد ، از ذیگر عوامل مهم در ارتباط با عرضه نقاط آسیب پذیر است(برخی از تولید کنندگان بمنظور افزایش سطح ایمنی مربوط به داده های حساس ، رشته را بصورت "اختصاصی " تغییر و استفاده می نمایند) . شنود اطلاعاتی و ترافیک SNMP ، می تواند افشاء اطلاعات و ساختار شبکه (سیستم ها و دستگاههای متصل شده به آن) را بدنبال داشته باشد . مهاجمین با استفاده از اطلاعات فوق ، قادر به انتخاب مناسب و دقیق هدف خود بمنظور برنامه ریزی حملات خود می باشند . اکثر تولید کنندگان بصورت پیش فرض نسخه یک SNMP را فعال و تعدادی دیگر، محصولاتی را ارائه می نمایند که قادر به استفاده ازمدل های امنیتی نسخه شماره سه SNMP نمی باشند. (با استفاده از مدل های امنیـی ارائه شده در نسخه شماره سه SNMP ، می توان پیکربندی لازم در خصوص روش های تائید را بهبود بخشید) . SNMP ، مختص یونیکس نمی باشد و در ابعاد وسیعی در ویندوز ، در تجهیزات شبکه ای ، در چاپگرها ، point access ها و Bridges ، استفاده می گردد. با توجه به نتایج حاصل از آنالیز حملات مبتنی بر SNMP ، مشخص شده است که اکثر حملات در این رابطه بدلیل ضعف در پیکربندی SNMP در سیستم های یونیکس است .

سیستم های عامل در معرض تهدید

تقریبا" بر روی تمامی سیستم های یونیکس و لینوکس یک نسخه SNMP نصب و بهمراه آن عرضه می گردند. در اغلب موارد پروتکل فوق ، بصورت پیش فرض فعال می باشد. اکثر دستگاه ها و سیستم های عامل شبکه ای مبتنی بر SNMP فوق بوده

فوق را توسط یک TCP-Wrapper و یا پیکربندی Xined انجام داد . استفاده از یک فایروال فیلترینگ بسته های اطلاعاتی مبتنی بر agent بر روی یک میزبان نیز می تواند در بلاک نمودن درخواست های ناخواسته SNMP موثر واقع شود .

حفاظت در مقابل رشته های قابل حدس

- غیر فعال نمودن SNMP در صورت عدم ضرورت استفاده از آن
- استفاده از یک مدل امنیتی مبتنی بر کاربر SNMPv3 ، بمنظور تائید پیام ها و رمزنگاری داده ها (در صورت امکان)
- در صورت استفاده از SNMP نسخه یک و یا دو ، می بایست از یک سیاست خاص بمنظور اسامی community (استفاده شده بعنوان رمزهای عبور) استفاده گردد. در این راستا لازم است اسامی بگونه ای انتخاب گردند که غیر قابل حدس بوده و بصورت ادواری و در محدوده های خاص زمانی نیز تغییر داده شوند .
- با استفاده از امکانات موجود می بایست بررسی لازم در خصوص استحکام اسامی در نظر گرفته شده برای رمزهای عبور راانجام داد.در این رابطه می توان از خودآموز و ابزار ارائه شده در آدرس http://www.sans.org/resources/idfaq/snmp.php ، استفاده کرد.
- SNMP را در گلوگاه های ورودی شبکه فیلتر نمائید (پورت 161 مربوط به TCP/UDP و پورت 162 مربوطه به TCP/UDP) . عملیات فوق را در مواردیکه ضرورتی به مدیریت دستگاهها بصورت خارجی وجود ندارد ، می بایست انجام داد . پیکربندی فیلترینگ را صرفا" بمنظور ترافیک مجاز SNMP بین subnet های ممیزی شده ، انجام دهید.

SSH

(Secure Shell)

SSH ، یک سرویس عمومی برای ایمن سازی Login ، اجرای دستورات و ارسال فایل در یک شبکه است .اکثر سیستم های مبتنی بر یونیکس از بسته نرم افزاری OpenSSH (نسخه فوق بصورت open-source است) و یا نسخه تجاری SSH Communication Security ، استفاده می نمایند . با اینکه SSH دارای ایمنی مناسبتری نسبت به telnet,ftp و برنامه های R-Command می باشد ، ولی همچنان در هر دو نسخه اشاره شده ، ضعف های امنیتی متعددی وجود دارد . اکثر ضعف های موجود صرفا" اشکالات جزئی بوده و تعداد اندکی از آنان ، حائز اهمیت بوده و می بایست بلافاصله نسبت به برطرف نمودن آنان اقدام گردد . مهمترین تهدید مرتبط با ضعف های امنیتی SSH ، امکان دستیابی (سطح ریشه) به ماشین آسیب پذیر توسط مهاجمان است . با توجه به رشد چشمگیر استفاده از سرویس گیرندگان و سرویس دهندگان SSH در محیط های ویندوز، اکثر اطلاعات ارائه شده در رابطه با نقطه آسیب پذیر فوق ، به نسخه های پیاده سازی شده SSH در ویندوز و * nix (یونیکس ، لینوکس) بر می گردد .عدم مدیریت مناسب SSH ، خصوصا" در ارتباط با پیکربندی و بکارگیری patch ها و بهنگام سازی لازم ، می تواند مسائل و مشکلات خاص خود را بدنبال داشته باشد .

SSH2 ، ابزاری قدرتمند
تعداد زیادی از نقاط آسیب‌پذیر تشخیص داده شده در پروتکل هائی نظیر POP3 (جایگزین با SSH2 SFTP) ، برنامه Telnet ، سرویس , HTTP و ابزارهای مبتنی بر rhost (نظیر : روش های تائید ،, rsh rcp ,rlogin) باعث ارسال اطلاعات بصورت clear text و یا عدم پردازش مناسب session های سرویس گیرنده – سرویس دهنده می گردد. پروتکل SSH1 ، دارای پتانسیل آسیب پذیری بالائی خصوصا" در ارتباط با session موقتی رمزنشده می باشد . بدین دلیل مدیران سیستم و شبکه ، استفاده از پروتکل SSH2 را گزینه ای شایسته در اینخصوص می دانند(در مواردیکه امکان آن وجود دارد) . لازم است به این

نکته مهم اشاره گردد که SSH1 و SSH2 با یکدیگر سازگار نبوده و لازم است نسخه SSH بر روی سرویس گیرنده و سرویس دهنده یکسان باشند (در این رابطه موارد استثنا ء نیز وجود دارد) .

کاربران OpenSSH می بایست به این نکته توجه نمایند که کتابخانه های OpenSSH در مقابل پتانسیل های ایجاد شده توسط OpenSSH ، دارای نرم افزارهای آسیب پذیر مختص خود می باشند. بمنظور آگاهی از جزئیات مربوطه ، می توان از آدرس CERT Advisory 2002-23 استفاده نمود .در سال 2002 یک نسخه آلوده از OpenSSH (نسخه فوق دارای یک trojan-horse بود) در زمان کوتاهی گسترش و باعث آسیب های فراوانی گردید. بمنظور کسب اطلاعات بیشتر در این رابطه و اطمینان از عدم آسیب پذیری سیستم خود در مقابل نسخه آلوده فوق ، می توان از آدرس http://www.openssh.org/txt/trojan.adv استفاده نمود .

سیستم های عامل در معرض تهدید

هر نسخه یونیکس و یا لینوکس که بر روی آن OpenSSH 3.3 و یا بعد از آن (نسخه ارائه شده در سال 2003، version 3.6.1) و یا SSH Communication Security's SSH 3.0.0 و یا بعد از آن (نسخه ارائه شده در سال 2003 شماره version 3.5.2) نصب واجراء می گردد ، در معرض این آسیب قرار خواهد داشت .

نحوه تشخیص آسیب پذیری سیستم

با استفاده از یک پویشگر مناسب ، می توان بررسی لازم در خصوص آسیب پذیری یک نسخه را انجام داد . در این رابطه می توان با اجرای دستور " ssh -V " ، از شماره نسخه نصب شده بر روی سیستم آگاه گردید. ScanSSH ، ابزاری مفید بمنظور تشخیص از راه دور سرویس دهندگان SSH آسیب پذیر بدلیل عدم Patching ، می باشد. دستور خطی ScanSSH ، لیستی از آدرس های شبکه را برای سرویس دهندگان پویش و گزارشی در ارتباط با شماره نسخه های آنان را ارائه می نماید . آخرین نسخهScanSSH که در سال 2001 ارائه شده است را می توان از آدرس http://www.monkey.org/~provos/scanssh/ دریافت نمود .

نحوه حفاظت در مقابل نقطه آسیب پذیر

بمنظور حفاظت در مقابل نقطه آسیب پذیر فوق ، موارد زیر پیشنهاد می گردد :

- نسخه SSH و یا OpenSSH را به آخرین نسخه موجود ارتقاء دهید . درصورتیکه SSH و یا OpenSSH بهمراه سیستم عامل ، نصب شده باشد ، می بایست آخرین Patchمربوطه را از سایت ارائه دهنده سیستم عامل دریافت و آن را برروی سیستم نصب نمود. در صورت استفاده از OpenSSL ، از نصب آخرین نسخه آن مطمئن شوید .
- حتی المقدور سعی گردد، نسخه SSH1 به SSH2 ارتقاء یابد .در رابطه با توسعه SSH1 در آینده تصمیم خاصی وجود نداشته و توسعه SSH2 مورد نظر می باشد .
- دو نسخه پیاده سازی شده SSH ، دارای مجموعه ای از گزینه های انتخابی بوده که مدیران سیستم با استفاده از آنان و با توجه به سیاست های موجود می توانند پیکربندی مناسبی در اینخصوص را انجام دهند. امکان محدودیت در دستیابی به ماشین مورد نظر و اتصال به آن ، روش های تائید کاربران و ماهیت کاربران مجاز ، نمونه هائی از گزینه های انتخابی بوده که می توان از آنان بمنظور پیکربندی مطلوب استفاده گردد.
- پیکربندی مناسب سرویس گیرندگان SSH در زمان اتصال به سرویس دهنده ای که SSH را حمایت نمی نماید . در چنین مواردی سرویس گیرنده ممکن است به عقب برگشته و استفاده از rsh را در این رابطه مفید تشخیص دهد . بمنظور پیشگیری از مواردی اینچنین می بایست به کلید FallBackToRsh در فایل پیکربندی SSH ، مقدار NO را نسبت داد .
- از رمزنگاری blowfish در مقابل DES3 استفاده گردد (روش DES3 ، ممکن است بصورت پیش فرض در نسخه مربوطه در نظر گرفته شده باشد). بدین ترتیب علاوه بر افزایش سرعت در عملیات ، رمزنگاری انجام شده نیز از استحکام مناسبی برخوردار خواهد بود.

عدم پیکربندی مناسب سرویس های

NIS/NFS

Network File System (NFS) و Network Information Service (NIS) ، دو سرویس مهم استفاده شده در شبکه های یونیکس می باشند. NFS ، سرویسی است که توسط شرکت Sun Microsystems بمنظور اشتراک فایل ها بین سیستم های موجود در یک شبکه یونیکس ، طراحی و پیاده سازی گردیده است . با استفاده از سرویس فوق امکان اشتراک و دستیابی به فایل های موجود بر روی کامپیوترهای شخصی و کامپیوترهای بزرگ فراهم می گردد . NFS ، روش استاندارد اشتراک فایل بین سیستم های کامپیوتری متفاوت است . NIS ، یک سرویس بانک اطلاعاتی توزیع شده را برای مدیریت اکثر فایل های مدیریتی مهم نظیر فایل های passwd و hosts فراهم می نماید .مدیریت متمرکز NIS ، امکان اعمال تغییرات مورد نظر را از یک نقطه (بانک اطلاعاتی) فراهم می نماید.(عدم ضرورت اعمال تغییرات بر روی هر سیستم موجود در شبکه) . مهمترین هدف NIS ، ارائه اطلاعات مربوط به مکان یابی (Maps نامیده می شود) به سایر سرویس های شبکه نظیر NFS است .فایل های passwd و group ، نمونه هائی در این زمینه (Maps ها مورد نظر) بوده که از آنان بمنظور تمرکز در تائید کاربران استفاده می گردد. سرویس های NIS و NFS ، دارای ضعف های امنیتی متعددی (مثلا" ضعف در سیستم تائید کاربران) بوده و در سالیان اخیر مهاجمان با استفاده از آنان ، حملات متعددی را سازماندهی و به سرانجام رسانده اند .اکثر حملات انجام شده از نوع overflows buffer ، DoS می باشد . با توجه به آسیب پذیری سرویس های فوق ، مهاجمان همواره از آنان بعنوان هدفی مناسب در جهت حملات خود استفاده نموده اند .

درصورت عدم پیکربندی مناسب سرویس های NFS و NIS و Patching مناسب ، سیستم های کامپیوتری در معرض تهدید جدی بوده و مهاجمان با سوءاستفاده از حفره های امنیتی موجود قادر به دستیابی سیستم از راه دور و یا بصورت محلی خواهند بود. مکانیزم استفاده شده توسط NIS برای تائید کاربران دارای ضعف های خاص خود بوده و مهاجمان با

استفاده از نرم افزارهائی نظیر ypcat قادر به نمایش مقادیر و داده موجود در بانک اطلاعاتی NIS بمنظور بازیابی فایل رمزعبور می باشند.

سیستم های عامل در معرض تهدید

تقریبا" تمامی سیستم های یونیکس و لینوکس بهمراه یک نسخه از NFS و NIS ارائه می گردند. سرویس های فوق، بصورت پیش فرض فعال می باشند .

نحوه تشخیص آسیب پذیری سیستم

بمنظور تشخیص آسیب پدیری سیستم در رابطه با سرویس های NIS و NFS موارد زیر پیشنهاد می گردد :

- بررسی و اطمینان از نصب آخرین Patch ارائه شده توسط تولید کننده . تمامی نسخه هائی که بهنگام نشده و یا آخرین Patch موجود بر روی آنان نصب نشده باشد در معرض تهدید و آسیب قرار خواهند داشت . در اکثر نسخه ها با استفاده از دستور rpc.mountd -version می توان از شماره نسخه NFS نصب شده آگاهی یافت . بمنظور آگاهی از شماره نسخه NIS می توان از دستور ypserv -version استفاده نمود (دستور فوق شماره نسخه NFS را نیز نمایش خواهد داد) .
- بمنظور تشخیص آسیب پذیری نرم افزار، می توان از یک پویشگر نقاط آسیب پذیر بهنگام شده بصورت مستمر استفاده تا بررسی لازم در خصوص ضعف های جدید صورت پذیرد.

بمنظور برخورد با عدم پیکربندی مناسب NIS موارد زیر پیشنهاد می گردد :

- اطمینان از عدم نگهداری Password Root در یک NIS map
- سازگاری رمزهای عبور تعریف شده با سیاست های امنیتی موجود. در این راستا می توان از یک برنامه Cracker بمنظور بررسی استحکام رمزهای عبور تعریف شده،استفاده نمود.

بمنظور برخورد با عدم پیکربندی مناسب NFS موارد زیر پیشنهاد می گردد :

- بررس لازم در خصوص بهنگام بودن میزبانان ، netgroups و مجوزها در فایل \ etc \ exports .
- اجرای دستور Showmount e بمنظور مشاهده عناصر export شده و بررسی سازگاری آنان با سیاست های امنیتی .

نحوه حفاظت در مقابل نقطه آسیب پذیر

موارد زیر در ارتباط با پیکربندی NIS ، پیشنهاد می گردد:

- مشخص نمودن صریح سرویس دهندگان NIS بر روی سرویس گیرندگان (پیشگیری لازم در خصوص تظاهر سایر سیستم ها بعنوان یک سرویس دهنده NIS) .
- در زمان ایجاد فایل های DBM ، ویژگی YP_SECURE فعال گردد . بدین ترتیب ،سرویس دهنده صرفا" به درخواست های ارسالی توسط یک سرویس گیرنده و از طریق پورت های مجاز ، پاسخ خواهد بود . در این رابطه می توان از سوئیچ S بهمراه دستور makedbm استفاده نمود .
- درج میزبانان مورداعتماد و شبکه ها در var/yp/securenets / که توسط پردازه های ypserv و ypxfrd استفاده می گردد.
- درج ::: 0:0 : × :+ در password map بر روی سرویس گیرندگان NFS

موارد زیر در ارتباط با پیکربندی NFS پیشنهاد می گردد:

- استفاده از آدرس های عددی IP و یا FQDN (fully qualified domain names) در مقابل اسامی مستعار (زمانیکه به سرویس گیرندگان در فایل etc / exports اجازه داده می شود) .
- بمنظوربررسی پیکربندی سیستم می توان از برنامه ای با نام NFSBug ، استفاده نمود.برنامه فوق امکانات متنوعی را بمنظور تست پیکربندی سیستم ارائه می نماید . برای دریافت برنامه NFSBug ، می توان از آدرس ftp://coast.cs.purdue.edu/pub/tools/unix/nfsbug/ استفاده نمود .
- اسفاده از فایل etc\exports بمنظور اعمال محدودیت در رابطه با دستیابی به سیستم فایل NFS با افزودن پارامترهای زیر :

– ممانعت کاربران معمولی از mounting یک سیستم فایل NFS با افزودن یک

پارامتر ایمن پس از آدرس IP و یا نام Domain مربوط به سرویس گیرنده NFS .
مثلا" : (/ secure)10.20.1.25 home

- export نمودن سیستم فایل NFS با مجوزهای مناسب .عملیات فوق را می توان با افزودن مجوزهای لازم (ro برای فقط خواندنی و یا rw برای خواندن و نوشتن) پس از آدرس IP مربوط به نام domain سرویس گیرنده NFS در فایل etc\export انجام داد.
مثلا" : (/ ro)10.20.1.25 home .
- در صورت امکان ، از پارامتر root_squash بعد از آدرس IP و نام Domain مربوط به سرویس گیرنده NFS استفاده گردد . در صورتیکه پارامتر فوق فعال شده باشد ، superuser ID root بر روی سرویس گیرنده NFS با کاربر ID nobody در سرویس دهنده NFS جایگزین می گردد. بدین ترتیب user root بر روی سرویس گیرنده ، قادر به دستیابی و یا تغییر فایل ها بر روی root نخواهند بود.
مثلا" : (/ .)10.20.1.25 home root_squash

- بر روی سیستم عامل سولاریس ، مانیتورینگ پورت فعال گردد. عملیات فوق را می توان با افزودن 1 Line set nfssrv:nfs_portmon = در فایل etc/system / ، انجام داد .

موارد زیر در ارتباط با NFS و NIS پیشنهاد می گردد :

- بازنگری و بررسی سیاست های فایروال بمنظور اطمینان از بلاک شدن تمامی پورت های غیر ضروری(مثلا" پورت 111 (portmap) و پورت 2049 (Rpc.nfsd)) .امکان دستیابی به سرویس دهندگان NFS و NIS ، می بایست صرفا" از طریق سرویس گیرندگان مجاز انجام شود.
- بررسی لازم در خصوص استفاده از NFS بر روی یک پروتکل ایمن نظیر SSH . در این رابطه می توان از آدرس http://www.math.ualberta.ca/imaging/snfs/ استفاده نمود .
- نصب تمامی patch های ارائه شده توسط تولید کنندگان و یا ارتقاء سرویس دهندگان NIS و NFS به آخرین نسخه موجود. برای آگاهی از اطلاعات مربوط به نصب و پیکربندی یونیکس با لحاظ نمودن مسائل امنیتی مضاعف ، می توان از آدرس Checklist UNIX Security استفاده نمود.

- غیر فعال نمودن NFS و NIS مربوط به deamons بر روی سیستم هائی که مختص یک سرویس دهنده NFS و یا NIS طراحی و تائید نشده اند . بمنظور پیشگیری لازم در اینخصوص می توان سرویس های NIS و یا NFS (و یا هر دو آنان) را از روی سیستم حذف نمود.

Layer Open Secure Sockets

(SSL)

کتابخانه OpenSSL (بصورت Open source است) ، نرم افزاری رایج بمنظور افزدون امنیت رمزنگاری به برنامه ها ئی است که از طریق شبکه با یکدیگر ارتباط برقرار می نمایند .سرویس دهنده آپاچی ، مهمترین وشناخته ترین برنامه استفاده کننده از پکیج فوق می باشد (بمنظور حمایت از hhtps ، ارتباط بر روی پورت 443) . برنامه های متعدد دیگر با انجام برخی تغییرات ، استفاده از OpenSSL بمنظور افزایش امنیت اطلاعات مبادله شده در یک شبکه را در دستور کار خود قرار داده اند .

برنامه های متعددی بمنظور ارائه امنیت رمزنگاری در ارتباط با یک Connection از OpenSSL استفاده می نمایند . مهاجمان در اغلب موارد در مقابل هدف قرار دادن مستقیم OpenSSL ، برنامه هائی را برای حملات خود انتخاب می نمایند که از OpenSSL استفاده می نمایند . یکی از اهداف مهاجمین در این رابطه ، سرویس دهنده آپاچی است که از OpenSSL استفاده می نماید . برنامه هائی دیگر نظیر: mail ، openldap,CUPS که از OpenSSL استفاده می نمایند نیز در معرض تهاجم قرار خواهند داشت . OpenSSL دارای نقاط آسیب پذیر متعددی بوده که می توان به چهار نمونه مهم آنان اشاره نمود : مورد اول ، مورد دوم ، مورد سوم ، مورد چهارم . مهاجمان با استفاده از نقاط آسیب پذیر فوق، قادر به اجرای کد دلخواه بعنوان کاربر کتابخانه OpenSSL می باشند.

سیستم های عامل در معرض تهدید

هر یک از سیستم های یونیکس و یا لینوکس که بر روی آنان نخسه OpenSSL 0.9.7 و یا قبل از آن اجراء می گردد، در معرض این آسیب قرار خواهند داشت .

نحوه تشخیص آسیب پذیری سیستم

در صورتیکه نسخه version 0.9.7a و یا قبل از آن نصب شده باشد ، سیستم در معرض تهدید و آسیب قرار خواهد داشت . در این رابطه می توان از دستور OpenSSL version ، بمنظور آگاهی از شماره نسخه نرم افزار نصب شده ، استفاده نمود.

نحوه حفاظت در مقابل نقطه آسیب پذیر

بمنظور حفاظت در مقابل نقطه آسیب پذیر فوق ، موارد زیر پیشنهاد می گردد :

- ارتقاء به آخرین و جدیدترین نسخه OpenSSL . در صورتیکه OpenSSL بهمراه سیستم عامل نصب شده است ، از تولیدکننده سیستم عامل مربوطه می بایست آخرین نسخه مربوطه را دریافت و آن را بر روی سیستم نصب نمود.
- در صورت امکان ، از ipfilter و سایر ابزارهای موجود در این زمینه بمنظور اعمال محدودیت در رابطه با سیستم هائی که می بایست به سرویس دهنده OpenSSL متصل گردند ،استفاده شود .