

امنیت اطلاعات

محمدتقی روغنی

مدیریت امنیت اطلاعات/ رمزنگاری/ امنیت شبکه/ امنیت وب/ امنیت بسته الکترونیکی/ امنیت سیستم عامل و دیتابیس



قیمت نشر کاغذی کتاب "امنیت اطلاعات": ۲۵۰۰۰ تومان

قیمت نشر الکترونیک کتاب "امنیت اطلاعات": ۲۰۰۰۰ تومان

خواننده عزیز، در صورتی که فایل کتاب را از منبعی جز سایت نویسنده آن دریافت نموده‌اید؛ لطفا مبلغ نشر الکترونیک را توسط یکی از راه‌های زیر به حساب نویسنده واریز نمایید:

روش اول:

دریافت فایل کامل همزمان با پرداخت مبلغ از طریق وب‌سایت :

http://MTRoghani.ir

روش دوم:

وازیر مبلغ ۲۰۰۰۰ تومان به شماره کارت ۶۱۰۴-۳۳۷۹-۲۶۱۱-۸۲۷۲



حمد سزاوار خداوندیست
که نا بخشیدن، بر دارائی‌هایش نیافزاید؛
و اگر بخشش کند، بینوا نشود.

امام علی علیه‌السلام

امنیت اطلاعات

محمدتقی روغنی

مقدمه مؤلف:

خداوند حکیم را شاکرم که عنایت خویش را برای نگارش کتابی جدید شامل حال بنده نمود، تا بتوانم گام کوچک دیگری در راستای ارتقای علمی فناوری اطلاعات هموطنان عزیزم بردارم.

تعاملات بین مشتریان و سازمان‌ها از طریق سرویس‌های فناوری اطلاعات، روزبه‌روز بیشتر شده و سازمان‌ها را مجبور می‌نماید که برای رفع نیاز مشتریان، دیتای بیشتری را بر روی سرویس‌دهنده‌های خود قرار دهند. هرقدر فناوری اطلاعات در کسب‌وکار سازمان‌ها بیشتر وارد شده و باعث تسهیل امور می‌شود، در مقابل ریسک سازمان در حفاظت از دارایی‌های اطلاعاتی را بالا می‌برد. از طرف دیگر وابستگی ادامه حیات کسب‌وکار سازمان، بیشتر به سرویس‌های فناوری اطلاعات گره می‌خورد.

در این سیرِ رو به رشد استفاده از سرویس‌های فناوری اطلاعات، متأسفانه بحث امنیت بسیار مهجور و یا مغفول مانده است. مخصوصاً در بین شرکت‌ها و سازمان‌های ایرانی آن‌چنان‌که بایدوشاید به بحث امنیت توجه نمی‌شود.

بعضی از سازمان‌ها برای مباحث امنیتی ارزش خاصی قائل نشده و تا روز بروز مصیبت، هیچ قدمی در جهت محافظت از دارایی‌های اطلاعاتی سازمان خود برنمی‌دارند. بعضی دیگر، امنیت را به خرید چند محصول امنیتی مثل فایروال و آنتی‌ویروس، محدود کرده و تصور می‌کنند که باوجود این محصولات، امنیت را در سازمان خود برقرار نموده‌اند.

توجه داشته باشید که امنیت صرفاً با خرید چند محصول امنیتی، برقرار نمی‌شود. اگر امنیت را یک فرآیند در نظر بگیریم، محصولات امنیتی تنها بخشی از آن فرآیند خواهند بود. در این فرآیند امنیتی، طراحی و ایجاد سیاست‌های امنیتی در سطح مدیران ارشد سازمان، باید به‌عنوان یکی از مؤلفه‌های مهم در نظر گرفته شود.

پس از بحث مدیران ارشد، یکی دیگر از مشکلات در زمینه امنیت، ناآگاهی مدیران و کارشناسان IT نسبت به مفاهیم امنیت اطلاعات است. بسیاری از کارشناسان، پیکربندی تجهیزات را صرفاً به‌صورت تجربی و بدون اطلاع از نحوه عملکرد پروتکل‌ها انجام می‌دهند. لذا در اکثر مواقع پیکربندی درستی با توجه به نیازهای سازمان انجام نمی‌پذیرد.

در این کتاب سعی شده بجای یک محصول امنیتی خاص، به مفاهیم اساسی امنیت پرداخته شود. زمانی که یک مدیر یا کارشناس IT و امنیت، آگاهی درستی از مفاهیم امنیتی داشته باشد، توان تجزیه‌وتحلیل بهتری نسبت به فرآیندهای امنیتی پیداکرده و بالطبع در زمان طراحی سیاست‌های امنیتی و یا پیکربندی تجهیزات، عملکرد بهتری خواهد داشت.

مطالب این کتاب در چهار بخش تقسیم‌بندی شده است. در بخش اول به استاندارد سیستم مدیریت امنیت اطلاعات پرداخته‌شده است. قطعاً زمانی که وارد بحث امنیت اطلاعات می‌شویم، یکی از مهم‌ترین مواردی که باید در نظر گرفته شود، استانداردهای مربوطه است. اصلی‌ترین استاندارد درزمینه امنیت اطلاعات، استاندارد خانواده ISMS است. در فصل اول و دوم کتاب، استانداردهای ISO 27000، ISO 27001 و ISO 27002 به‌صورت اجمالی بررسی‌شده است.

در بخش دوم، به یکی از ارکان امنیت اطلاعات، یعنی رمزنگاری، پرداخته‌شده است. نحوه عملکرد الگوریتم‌های مهم رمزنگاری مثل DES، 3DES، AES، RSA و SHA شرح داده‌شده است. همچنین کاربرد این الگوریتم‌ها در برقراری سرویس‌های امنیتی مثل محرمانگی، احرازهویت، صحت، امضای دیجیتال و عدم انکار، تشریح شده است.

در بخش سوم این کتاب، مباحث امنیت شبکه مورد بررسی قرارگرفته است. با توجه به گسترش شبکه‌های ارتباطی، یکی از دغدغه‌های اصلی مدیران و کارشناسان، امنیت شبکه است. در فصول این بخش به مفاهیم امنیتی و تکنولوژی‌های مورداستفاده در امنیت شبکه ازجمله فایروال، IDS، IPS و هانی‌پات پرداخته شده است. در آخر این بخش نیز نحوه برقراری امنیت در ارتباطات شبکه‌ای با استفاده از IPsec و VPN، مورد بررسی قرار گرفته است.

در بخش چهارم، به امنیت سایر سرویس‌هایی که بیشتر مورداستفاده قرار می‌گیرند، پرداخته‌شده است. ازجمله تشریح پروتکل HTTPS برای امنیت وب، پروتکل PGP برای امنیت پست الکترونیک، بهترین شیوه‌های امنیت سیستم‌عامل و بررسی تهدیدات برتر دیتابیس و نحوه مقابله با آن‌ها، در فصول این بخش گنجانده شده است.

هرچند این کتاب توسط اساتید و کارشناسان امنیت مورد بازخوانی و رفع اشکال قرارگرفته، اما نمی‌توان آن را خالی از نقصان دانست. لذا خواهشمندم هرگونه انتقاد و پیشنهاد مدنظر خود را از طریق پست الکترونیک MTRoghani@Gmail.com، برای بنده ارسال نمایید.

در پایان امیدوارم که این کتاب بتواند نقشی هرچند کوچک، در افزایش آگاهی مدیران و کارشناسان IT و امنیت اطلاعات ایفا نماید. در ضمن از تمام اساتید و کارشناسان عزیزی که از راهنمایی‌های خود در نگارش این کتاب دریغ نکرده‌اند، صمیمانه سپاسگزاری می‌نمایم.

با احترام

محمدتقی روغنی

پاییز ۱۳۹۳

فهرست مطالب:

مقدمه مؤلف ۵

بخش اول استاندارد ISMS ۲۳

فصل اول مرور کلی استاندارد ISMS و الزامات آن ۲۵

مبحث اول؛ مرور استاندارد ISO/IEC 27000:2014 ۲۷

استانداردهای خانواده ISMS ۲۸

مروری بر استانداردهای خانواده ISMS ۲۹

۱- استاندارد ISO/IEC 27000 (همین استاندارد) ۲۹

۲- استاندارد ISO/IEC 27001 ۲۹

۳- استاندارد ISO/IEC 27006 ۲۹

٤- استاندارد ISO/IEC 27002 ۳۰

۵- استاندارد ISO/IEC 27003 ۳۰

٦- استاندارد ISO/IEC 27004 ۳۱

۷- استاندارد ISO/IEC 27005 ۳۱

۸- استاندارد ISO/IEC 27007 ۳۱

۹- استاندارد ISO/IEC TR 27008 ۳۱

۱۰- استاندارد ISO/IEC 27013 ۳۲

۱۱- استاندارد ISO/IEC 27014 ۳۲

۱۲- استاندارد ISO/IEC TR 27016 ۳۳

۱۳- استاندارد ISO/IEC 27010 ۳۳

۱٤- استاندارد ISO/IEC 27011 ۳۳

۱۵- استاندارد ISO/IEC TR 27015 ۳٤

۱٦- استاندارد ISO/IEC 27799 ۳٤

اصطلاحات و تعاریف ۳۵

مروری بر ISMS ۳۷

اصول PDCA ۳۹

برقراری، نظارت، نگهداری و بهبود عملکرد ISMS ٤۰

مبحث دوم؛ مرور استاندارد ISO/IEC 27001:2013 ٤۱

تعیین چارچوب سیستم مدیریت امنیت اطلاعات ٤۱

رهبری و تعهد ٤۲

سیاست‌گذاری و خط‌مشی ۴۲
نقش‌های سازمانی، مسئولیت‌ها و اختیارات ۴۳
برنامه‌ریزی ۴۳
ارزیابی ریسک امنیت اطلاعات ۴۳
مقابله با ریسک امنیت اطلاعات ۴۴
اهداف امنیت اطلاعات و برنامه‌ریزی برای دستیابی به آن‌ها ۴۴
صلاحیت و آگاهی ۴۵
ارتباط ۴۵
مستندسازی ۴۵
برنامه‌ریزی و کنترل عملیاتی ۴۶
ارزیابی کارایی ۴۶
بهبود ۴۷
نمودار کلی استاندارد ISO/IEC 27001:2013 ۴۷

فصل دوم دستورالعمل مدیریت امنیت اطلاعات ۴۹

مبحث اول؛ مرور استاندارد ISO/IEC 27002:2013 ۵۱
ساختار استاندارد ISO/IEC 27002 ۵۱
۵ – ماده ۱: سیاست‌های امنیت اطلاعات ۵۲
خط‌مشی امنیت اطلاعات ۵۲
۶ – ماده ۲: امنیت اطلاعات سازمان ۵۳
نقش‌ها و مسئولیت‌های امنیت اطلاعات ۵۳
خط‌مشی تجهیزات همراه ۵۴
دورکاری ۵۵
۷ – ماده ۳: امنیت منابع انسانی ۵۶
قوانین و شرایط استخدام ۵۶
۸ – ماده ٤: مدیریت دارایی ۵۷
فهرست دارایی‌ها ۵۷
طبقه‌بندی اطلاعات ۵۸
اداره دارایی ۵۹
مدیریت رسانه‌های قابل‌حمل ۶۰
دفع رسانه ۶۱
۹ – ماده ۵: کنترل دسترسی ۶۱
سیاست‌های کنترل دسترسی ۶۱
دسترسی به شبکه و سرویس‌های شبکه ۶۳

ثبت و لغو کاربر ٦٤
محدودیت دسترسی به اطلاعات ٦٥
سیستم مدیریت رمزعبور ٦٥
١٠ - ماده ٦: رمزنگاری ٦٦
سیاست استفاده از کنترل‌های رمزنگاری ٦٦
مدیریت کلید رمزنگاری ٦٨
١١ - ماده ٧: امنیت محیطی و فیزیکی ٦٨
امنیت فیزیکی محیط پیرامونی ٦٨
کنترل ورود فیزیکی ٦٩
حفاظت در مقابل تهدیدات بیرونی و زیست‌محیطی ٧٠
امنیت کابل‌کشی ٧٠
١٢ - ماده ٨: امنیت عملیات ٧١
مستندسازی دستورالعمل عملیاتی ٧١
مدیریت تغییر ٧٢
کنترل در برابر بدافزارها ٧٣
پشتیبان‌گیری ٧٤
ثبت رویداد ٧٥
نصب نرم‌افزار بر روی سیستم‌عامل ٧٦
محدودیت در نصب نرم‌افزار ٧٧
١٣ - ماده ٩: امنیت ارتباطات ٧٨
کنترل شبکه ٧٨
سیاست‌ها و دستورالعمل انتقال اطلاعات ٧٩
١٤ - ماده ١٠: کسب، توسعه و نگهداری سیستم ٨٠
محافظت از نرم‌افزار خدمات معاملات ٨٠
سیاست توسعه امن نرم‌افزار ٨١
دستورالعمل کنترل تغییرات سیستم ٨٢
١٥ - ماده ١١: ارتباط با تأمین‌کنندگان ٨٣
سیاست‌های امنیت اطلاعات برای ارتباط با تأمین‌کنندگان ٨٣
١٦ - ماده ١٢: مدیریت حوادث امنیت اطلاعات ٨٥
مسئولیت‌ها و دستورالعمل‌ها ٨٥
گزارش رویداد امنیت اطلاعات ٨٦
گزارش ضعف‌های امنیت اطلاعات ٨٧
واکنش به حوادث امنیت اطلاعات ٨٨
١٧ - ماده ١٣: جنبه‌های امنیت اطلاعات در مدیریت تداوم کسب‌وکار ٨٩
برنامه‌ریزی تداوم امنیت اطلاعات ٨٩

دسترس‌پذیر بودن امکانات پردازش اطلاعات ۸۹
۱۸ - ماده ۱٤: انطباق ۹۰
انطباق با استانداردها و سیاست‌های امنیتی ۹۰
بررسی انطباق فنی ۹۱
بخش دوم رمـزنـــگـاری ۹۳
فصل سوم مفاهیم رمزنگاری ۹۵
مبحث اول؛ مفاهیم و اصطلاحات رمزنگاری ۹۷
تعریف رمزنگاری ۹۷
اصطلاحات رمزنگاری ۹۸
انواع رمزنگاری ۱۰۰
انواع حملات به سیستم رمزنگاری ۱۰۱
انواع حملات کشف رمز ۱۰۱
اصول انتخاب سیستم رمزنگاری ۱۰۲
شروط شانون ۱۰۲
پنهان‌کاری (Steganography) ۱۰۳
رمزنگاری و فشرده‌سازی ۱۰۳
مبحث دوم؛ مفاهیم رمزنگاری متقارن ۱۰٤
اصول کلی رمزگذاری ۱۰۵
تقسیم‌بندی الگوریتم‌های جایگزینی ۱۰۵
رمزنگاری کلاسیک و مدرن ۱۰٦
انواع رمزنگاری کلاسیک ۱۰٦
۱. الگوریتم رمز سزار (Caesar Cipher) ۱۰۷
۲. الگوریتم رمز Playfair ۱۰۸
۳. الگوریتم رمز ویجینر (Vigen˛re Cipher) ۱۰۹
٤. الگوریتم رمز نرده راه‌آهن (Rail Fence) ۱۱۱
کلیدهای One-Time Pad ۱۱۲
مبحث سوم؛ مفاهیم رمزنگاری نامتقارن ۱۱۳
ویژگی‌های الگوریتم رمزنگاری نامتقارن ۱۱۵
مقایسه رمزنگاری متقارن با رمزنگاری نامتقارن ۱۱٦
محرمانگی و احرازهویت ۱۱٦
کاربردهای سیستم رمزنگاری کلیدعمومی ۱۱۷
الزامات رمزنگاری نامتقارن ۱۱۸
مبحث چهارم؛ مفاهیم توابع درهم‌سازی ۱۱۹

اصطلاحات توابع درهم‌سازی ۱۲۰
خواص توابع درهم‌سازی ۱۲۰
کاربرد Hash در بررسی صحت پیام ۱۲۱
کاربرد توابع درهم‌سازی ۱۲۲
امنیت توابع درهم‌سازی ۱۲٤
مقایسه توابع درهم‌سازی ۱۲۵
فصل چهارم الگوریتم‌های رمزنگاری متقارن ۱۲۷
مبحث اول؛ الگوریتم DES ۱۲۹
الگوریتم فایستل ۱۲۹
رمزگذاری و رمزگشایی در فایستل ۱۳۰
پارامترها و ویژگی‌های طراحی فایستل ۱۳۲
الگوریتـــم DES ۱۳۲
رمزگذاری در DES ۱۳۳
جایگشت اولیه (Initial Permutation) ۱۳۵
جزئیات یک دور (Round) از الگوریتم ۱۳٦
رمزگشایی در DES ۱۳۹
بررسی قدرت DES ۱۳۹
مبحث دوم؛ الگوریتم AES ۱٤۱
گالوا فیلد ۱٤۲
عملیات جمع در AES ۱٤۲
عملیات ضرب در AES ۱٤۲
ساختار کلی AES ۱٤۳
جزئیات ساختار AES ۱٤۵
جزئیات عملیات هر دور AES ۱٤۸
ویژگی‌های کلیدرمز در AES ۱۵۱
مبحث سوم؛ الگوریتم Triple DES ۱۵۲
3DES با دو کلید متفاوت ۱۵۲
3DES با سه کلید متفاوت ۱۵٤
مبحث چهارم؛ روش‌های عملیاتی رمزنگاری بلوکی ۱۵۵
روش Electronic Codebook (ECB) ۱۵٦
روش Cipher Block Chaining (CBC) ۱۵۷
روش Cipher Feedback (CFB) ۱۵۸
روش Output Feedback (OFB) ۱٦۰

روش Counter (CTR) ۱۶۲
مزایای روش CTR ۱۶۳

فصل پنجم الگوریتم‌های رمزنگاری نامتقارن ۱۶۵

مبحث اول؛ الگوریتم RSA ۱۶۷
تشریح الگوریتم RSA ۱۶۸
محاسبه کلید عمومی و خصوصی در RSA ۱۶۸
مثال محاسبه کلید خصوصی و عمومی ۱۶۹
تولید کلید ۱۷۲
امنیت RSA ۱۷۳
الف) حملات تهدیدکننده RSA ۱۷۳
ب) اقدامات متقابل با حملات RSA ۱۷۴
مبحث دوم؛ الگوریتم دیفی‌هلمن ۱۷۶
تشریح الگوریتم دیفی‌هلمن ۱۷۶
مثال محاسبه کلید دیفی‌هلمن ۱۷۸
حمله مردمیانی ۱۸۰
سیستم رمزنگاری الجمل ۱۸۱

فصل ششم توابع درهم‌سازی و کاربردها ۱۸۵

مبحث اول؛ تصدیق پیام ۱۸۷
روش اول؛ تصدیق پیام به‌علاوه محرمانگی ۱۸۷
روش دوم؛ تصدیق پیام بدون محرمانگی ۱۸۸
روش سوم؛ تصدیق پیام بدون استفاده از الگوریتم متقارن ۱۸۸
روش چهارم؛ تصدیق پیام با مقدار S به‌علاوه محرمانگی ۱۸۹
الگوریتم MAC ۱۹۰
توابع HMAC ۱۹۲
مبحث دوم؛ امضای دیجیتال ۱۹۳
روش اول: امضای دیجیتال ۱۹۴
روش دوم: امضای دیجیتال به‌علاوه محرمانگی با رمزنگاری متقارن ۱۹۵
روش سوم: امضای دیجیتال به‌علاوه محرمانگی با رمزنگاری نامتقارن ۱۹۶
الزامات امضای دیجیتال ۱۹۷
ویژگی عدم‌انکار ۱۹۷
استاندارد امضای دیجیتال (DSS) ۱۹۸

الگوریتم DSA ۲۰۰
تابع Signature ۲۰۱
تابع Verification ۲۰۲
مبحث سوم؛ الگوریتم SHA ۲۰۴
الگوریتم SHA-512 ۲۰۵
جزئیات یک دور (Round) از SHA ۲۰۹
جزئیات Message Schedule ۲۱۱
بخش سوم امنیت شبکه ۲۱۳
فصل هفتم مفاهیم امنیت شبکه ۲۱۵
مبحث اول؛ مفاهیم امنیتی ۲۱۷
مثلث امنیت اطلاعات ۲۱۹
محرمانگی (Confidentiality) ۲۱۹
صحت و تمامیت (Integrity) ۲۱۹
دسترس‌پذیری (Availability) ۲۲۰
طبقه‌بندی آسیب‌پذیری ۲۲۱
اصول اساسی امنیت ۲۲۱
مبحث دوم؛ انواع تهدیدات ۲۲۳
مالکیت معنوی ۲۲۴
حملات نرم‌افزاری ۲۲۵
ویروس ۲۲۵
کرم ۲۲۶
اسب تراوا ۲۲۶
بمب‌های منطقی ۲۲۶
درب پشتی ۲۲۶
تهدیدات چندشکلی ۲۲۶
مضرات آنتی‌ویروس‌ها ۲۲۷
تنزل در کیفیت خدمات ۲۲۷
جاسوسی یا تجاوز ۲۲۷
بلایای طبیعی ۲۲۸
خطاهای انسانی ۲۲۹
اخاذی اطلاعاتی ۲۲۹
سیاست یا برنامه‌ریزی سازمانی ناقص، ناکافی و یا گمشده ۲۲۹

کنترل‌های ناقص، ناکافی و یا گمشده ۲۳۰
کارشکنی یا خرابکاری ۲۳۰
سرقت ۲۳۰
خطاها و شکست سخت‌افزاری ۲۳۰
خطاها و شکست فنی نرم‌افزار ۲۳۱
منسوخ‌شدن تکنولوژی ۲۳۱
مبحث سوم؛ انواع حملات ۲۳۲
روش‌های حمله ۲۳۳
شناسایی ۲۳۳
مهندسی اجتماعی ۲۳۳
افزایش حق دسترسی ۲۳۳
درب پشتی(Back Door) ۲۳٤
کانال پنهان (Covert channel) ۲۳٤
حملات داخلی ۲۳٤
کشف کلمه عبور ۲۳٥
Botnet ۲۳٥
حملات امنیتی ۲۳٦
حملات غیرفعال (Passive Attack) ۲۳٦
حملات فعال (Active Attack) ۲۳۷
حقه‌بازی (Spoofing) ۲۳۹
حمله مردمیانی ۲۳۹
هرزنامه (Spam) ۲۳۹
استراق سمع (Sniff) ۲۳۹
مبحث چهارم؛ سرویس‌ها و مکانیسم‌های امنیتی ۲٤۰
الف) سرویس‌های امنیتی ۲٤۰
احراز هویت (Authentication) ۲٤۰
کنترل دسترسی (Access Control) ۲٤۱
محرمانگی اطلاعات (Data Confidentiality) ۲٤۱
صحت اطلاعات (Data Integrity) ۲٤۱
انکارناپذیری (Non-repudiation) ۲٤۲
ب) مکانیسم‌های امنیتی ۲٤۲
رمزگذاری (Encipherment) ۲٤۲
مکانیسم امضای دیجیتال (Digital Signature) ۲٤۳

مکانیسم کنترل دسترسی (Access Control) ۲۴۳
مکانیسم‌های صحت دیتا ۲۴۴
مکانیسم تبادل احراز هویت ۲۴۵
مکانیسم توسعه ترافیک (Traffic padding mechanism) ۲۴۶
مکانیسم کنترل مسیریابی ۲۴۶
مکانیسم گواهی رسمی (Notarization mechanism) ۲۴۶

فصل هشتم تکنولوژی‌ها و ابزارهای‌امنیتی ۲۴۷

مبحث اول؛ کنترل دسترسی ۲۴۹

۱. کنترل دسترسی اجباری (Mandatory Access Controls) ۲۴۹
۲. کنترل دسترسی غیراختیاری (Non-discretionary control) ۲۵۰
۳. کنترل دسترسی اختیاری (Discretionary Access Control) ۲۵۰
مکانیسم‌های کنترل دسترسی ۲۵۰
شناسایی ۲۵۰
احرازهویت ۲۵۱
مجوزدهی ۲۵۲
پاسخگویی ۲۵۳
پروتکل AAA ۲۵۳
پروتکل کربروس (Kerberos) ۲۵۴

مبحث دوم؛ فایروال ۲۵۵

گروه اول؛ روش‌های پردازش فایروال ۲۵۵
۱. فایروال فیلترینگ بسته‌ها ۲۵۶
۲. دروازه برنامه ۲۵۹
۳. دروازه مدار ۲۶۰
۴. فایروال لایه MAC ۲۶۰
۵. فایروال چندگانه ۲۶۰
گروه دوم؛ تقسیم‌بندی بر اساس دوران توسعه ۲۶۱
Next-Generation Firewall ۲۶۲
گروه سوم؛ تقسیم‌بندی بر اساس ساختار ۲۶۲
انواع مناطق در شبکه ۲۶۳
۱. منطقه قابل‌اطمینان (Trusted Zone) ۲۶۳
۲.منطقه غیرقابل‌اطمینان (Untrusted Zone) ۲۶۴
۳.منطقه کمتر حفاظت‌شده (Demilitarized Zone) ۲۶۴

٤.منطقه سرورها (Server Zone) ٢٦٤
بهترین شیوه پیکربندی فایروال ٢٦٦
مبحث سوم؛ IDS/IPS ٢٦٨
وظایف کلیدی تکنولوژی IDPS ٢٧٠
روش‌های تشخیص ٢٧٢
١. تشخیص مبتنی امضا (Signature-Based Detection) ٢٧٢
٢. تشخیص مبتنی بر رفتار خلاف قاعده (Anomaly-Based Detection) ٢٧٢
٣. تحلیل وضعیت پروتکل (Stateful Protocol Analysis) ٢٧٣
انواع تکنولوژی IDPS ٢٧٥
١. مبتنی بر شبکه (Network-Based) ٢٧٥
٢. بی‌سیم (Wireless) ٢٧٥
٣. تحلیل رفتار شبکه (Network Behavior Analysis) ٢٧٥
٤. مبتنی بر میزبان (Host-Based) ٢٧٦
اجزای IDPS ٢٧٦
١. سنسور یا عامل (Sensor or Agent) ٢٧٦
٢. سرور مدیریت (Management Server) ٢٧٦
٣. سرور دیتابیس (Database Server) ٢٧٧
٤. کنسول (Console) ٢٧٧
قابلیت‌های امنیتی IDPS ٢٧٧
١. قابلیت جمع‌آوری اطلاعات (Information Gathering Capabilities) ٢٧٧
٢. قابلیت واقعه‌نگاری (Logging Capabilities) ٢٧٧
٣. قابلیت تشخیص (Detection Capabilities) ٢٧٨
٤. قابلیت پیشگیری (Prevention Capabilities) ٢٨٠
معماری شبکه و محل قرارگیری سنسور ٢٨٠
١. برخط (Inline) ٢٨١
٢. منفعل (Passive) ٢٨٢
مبحث چهارم؛ هانی‌پات ٢٨٣
مزایا و معایب هانی‌پات ٢٨٤
١. مزایای استفاده از هانی‌پات ٢٨٤
٢. معایب استفاده از هانی‌پات ٢٨٤
انواع هانی‌پات ٢٨٥
١. کم‌تعامل (Low-Interaction) ٢٨٥
٢. پرتعامل (High-Interaction) ٢٨٥

فصل نهم ارتباطات امن شبکه ۲۸۷
مبحث اول؛ IPsec ۲۸۹
کاربردهای IPsec ۲۹۰
مزایای استفاده از IPsec ۲۹۲
پروتکل‌های مورداستفاده در IPsec ۲۹۳
۱. Authentication Header (AH) ۲۹۳
۲. Encapsulating Security Payload (ESP) ۲۹۳
حالت‌های استفاده از پروتکل‌های AH و ESP ۲۹۳
۱. Transport Mode ۲۹٤
۲. Tunnel Mode ۲۹٤
سیاست امنیت IP (IP Security Policy) ۲۹۵
Security Association (SA) ۲۹۵
Security Association Database (SAD) ۲۹٦
Security Policy Database (SPD) ۲۹۸
انتخاب نوع پردازش ۲۹۹
پروتکل Encapsulating Security Payload (ESP) ۳۰۰
حالت‌های ESP ۳۰۰
فرآیند رمزنگاری در ESP ۳۰۱
فرآیند حفاظت از صحت (تمامیت) در ESP ۳۰۱
ساختار بسته ESP ۳۰۲
پروتکل IKE ۳۰٤
فاز اول IKE ۳۰۵
۱. روش Main Mode ۳۰۵
۲. روش Aggressive Mode ۳۰۸
فاز دوم IKE ۳۰۹
مبحث دوم؛ شبکه خصوصی مجازی ۳۱۱
انواع VPN ۳۱۲
انواع VPN بر اساس تکنولوژی ۳۱۲
انواع VPN بر اساس پروتکل ۳۱۲
انواع VPN بر اساس لایه OSI ۳۱۲
انواع اتصال در VPN ۳۱۳
۱. Remote Access VPN ۳۱۳
۲. Site to Site VPN ۳۱۳
مزایای استفاده از VPN ۳۱٤

پروتکل PPTP ۳۱۵
پروتکل PAP ۳۱۶
پروتکل CHAP ۳۱۶
پروتکل MS-CHAP ۳۱۶
پروتکل MPPE ۳۱۷
پروتکل L2TP ۳۱۷
پروتکل IPsec ۳۱۸
پروتکل SSL VPN ۳۱۹
۱. SSL Portal VPN ۳۲۰
۲. SSL Tunnel VPN ۳۲۰

بخش چهارم امنیت سیستم‌ها و سرویس‌ها ۳۲۱

فصل دهم امنیت وب ۳۲۳

مبحث اول؛ پروتکل SSL/TLS ۳۲۵
معماری SSL ۳۲۵
مفاهیم Session و Connection در SSL ۳۲۶
پارامترهای تعیین وضعیت Session ۳۲۷
پارامترهای تعیین وضعیت Connection ۳۲۸
پروتکل SSL Record ۳۲۹
پروتکل Change Cipher Spec ۳۳۱
پروتکل Alert ۳۳۱
پروتکل Handshake ۳۳۳
فاز اول پروتکل Handshake؛ برقراری قابلیت‌های امنیتی ۳۳۵
فاز دوم پروتکل Handshake؛ احرازهویت سرور و تبادل کلید ۳۳۷
فاز سوم پروتکل Handshake؛ احرازهویت کلاینت و تبادل کلید ۳۳۸
فاز چهارم پروتکل Handshake؛ پایان ۳۳۹
مبحث دوم؛ پروتکل HTTPS ۳۴۱
ملاحظات امنیتی وب ۳۴۲
تهدیدات امنیتی وب ۳۴۳
روش‌های امنیت ترافیک وب ۳۴۴
پروتکل HTTPS ۳۴۵
شروع اتصال ۳۴۵
بستن اتصال ۳۴۷

فصل یازدهم امنیت پست الکترونیک ۳۴۹

مبحث اول؛ پروتکل PGP ۳۵۲

نشانه‌گذاری ۳۵۳

تشریح عملیات PGP ۳۵٤

احرازهویت در PGP ۳۵۵

محرمانگی در PGP ۳۵٦

احراز هویت به همراه محرمانگی در PGP ۳۵۷

فشرده‌سازی در PGP ۳۵۸

سازگاری با پست الکترونیک ۳۵۸

کلیدهای رمزنگاری و دسته‌کلیدها ۳۵۹

مولد کلید نشست (Session Key Generation) ۳٦۰

شناسه کلید ۳٦۱

دسته‌کلید ۳٦٤

تولید پیام در PGP ۳٦۷

دریافت پیام در PGP ۳٦۸

مبحث دوم؛ پروتکل S/MIME ۳۷۱

استاندارد RFC 5322 ۳۷۱

پروتکل MIME ۳۷۳

انواع محتوا در MIME ۳۷٤

کدگذاری انتقال در MIME ۳۷۵

پروتکل S/MIME ۳۷٦

الگوریتم‌های رمزنگاری در S/MIME ۳۷٦

پیام S/MIME ۳۷۹

Enveloped Data ۳۸۰

Signed Data ۳۸۰

پردازش گواهینامه در S/MIME ۳۸۱

نقش کاربر ۳۸۱

گواهینامه VeriSign ۳۸۱

فصل دوازدهم امنیت سیستم‌عامل ۳۸۳

مبحث اول؛ بهترین شیوه‌های مدیریت امن سیستم ۳۸٦

امنیت فیزیکی سیستم و کنسول ۳۸٦

نصب سیستم با حداقل برنامه‌ها ۳۸۸
رمزعبور مدیر سیستم ۳۸۸
محول کردن وظایف مدیریتی ۳۸۹
رمزعبور کاربران ۳۹۰
سیستم کاربر ۳۹۱
محدود کردن کاربر ۳۹۲
آموزش کاربران ۳۹۲
به‌روزرسانی سیستم‌ها ۳۹۳
آزمون آسیب‌پذیری ۳۹٤
مانیتورینگ سیستم ۳۹٥
مستندسازی پیکربندی ۳۹٥
پشتیبان‌گیری و بازیابی فاجعه ۳۹٦
آنتی‌ویروس ۳۹۷
مبحث دوم؛ سطوح امنیتی در سیستم‌عامل ۳۹۸
۱. Trusted Computing Base (TCB) ۳۹۹
۲. Automatic Data Processing (ADP) ۳۹۹
بخش D: محافظت حداقلی (Minimal Protection) ۳۹۹
بخش C: محافظت اختیاری (Discretionary Protection) ۳۹۹
• کلاس C1: Discretionary Security Protection ۳۹۹
• کلاس C2: Controlled Access Protection ٤۰۱
بخش B: محافظت اجباری (Mandatory Protection) ٤۰۳
• کلاس B1: Labeled Security Protection ٤۰۳
• کلاس B2: Structured Protection ٤۰٥
• کلاس B3: Security Domains ٤۰۷
بخش A: محافظت تأییدشده (Verified Protection) ٤۰۹
• کلاس A1: Verified Design ٤۰۹
انواع کنترل دسترسی ٤۱۱
• کنترل دسترسی اختیاری (DAC) ٤۱۱
• کنترل دسترسی اجباری (MAC) ٤۱۲
• کنترل دسترسی بر مبنای نقش (Role-Based Access Control) ٤۱۲
• کنترل دسترسی بر مبنای شبکه (Lattice-Based Access Control) ٤۱۲
• کنترل دسترسی بر مبنای خصوصیت(Attribute-Based Access Control)
٤۱۳
مبحث سوم؛ بهترین شیوه‌های امنیت سیستم‌عامل ٤۱٤

۱. نصب وصله‌ها و به‌روزرسانی سیستم‌عامل ٤١٥
۲. مقاوم‌سازی و ایمن‌سازی پیکربندی سیستم‌عامل ٤١٦
حذف یا غیرفعال کردن سرویس‌ها، برنامه‌ها و پروتکل‌های شبکه غیرضروری٤١٦
پیکربندی احرازهویت کاربر در سیستم‌عامل ٤١٧
پیکربندی کنترل‌های منابع ٤٢٠
۳. نصب و پیکربندی کنترل‌های امنیتی اضافی ٤٢٠
٤. آزمون امنیت سیستم‌عامل ٤٢٢

فصل سیزدهم امنیت دیتابیس ۴۲۳

مبحث اول؛ ده تهدید برتر دیتابیس ٤٢٦
۱. امتیازات بیش‌ازحد و استفاده‌نشده ٤٢٦
۲. سوءاستفاده از امتیازات ٤٢٧
۳. تزریق ورودی (SQL Injection سابق) ٤٢٧
٤. بدافزار ٤٢٨
٥. دنباله ممیزی ضعیف ٤٢٨
٦. در معرض خطر قرار گرفتن رسانه‌های ذخیره‌سازی ٤٢٩
۷. بهره‌برداری از آسیب‌پذیری‌ها و اشتباهات پیکربندی دیتابیس ٤٢٩
۸. دیتای حساس مدیریت نشده ٤٣٠
۹. منع خدمت ٤٣٠
۱۰. تخصص و آموزش امنیتی محدود ٤٣١
مبحث دوم؛ روش‌های مقابله با تهدیدات ٤٣٢
استراتژی دفاعی چندلایه برای امنیت دیتابیس ٤٣٢
۱. کشف و ارزیابی (Discovery and Assessment) ٤٣٤
۲. مدیریت حقوق کاربر (User Rights Management) ٤٣٦
۳. نظارت و مسدودسازی (Monitoring and Blocking) ٤٣٧
٤. حسابرسی (Auditing) ٤٣٩
٥. حفاظت از اطلاعات (Data Protection) ٤٣٩
٦. امنیت غیر فنی (Non-Technical Security) ٤٤٠

بخش پایانی ضمائم ۴۴۱

منابع ۴۴۳

معرفی کتاب ۴۴۵

امنیت یک محصول نیست؛

بلکه فرآیندی است شامل سیاست‌گذاری، طراحی، مدیریت و اجرای صحیح تمام سرویس‌ها و محصولات فناوری اطلاعات.

و البته

نظارت مستدام بر عملکرد، به‌روزرسانی و ارتقاء آن‌ها.

بخش اول

استاندارد ISMS

فصل اول

مرور کلی استاندارد ISMS و الزامات آن

فصل دوم

دستورالعمل مدیریت امنیت اطلاعات

فصل اول

مرور کلی استاندارد ISMS و الزامات آن

مبحث اول

مرور استاندارد ISO/IEC 27000:2014

مبحث دوم

مرور استاندارد ISO/IEC 27001:2013

☑ مبحث اول

مرور استاندارد ISO/IEC 27000:2014

سازمان‌ها در هراندازه و هر نوع فعالیت علمی، سیاسی و اقتصادی که قرار داشته باشند، باید مقداری اطلاعات را جمع‌آوری، پردازش، نگهداری و ارسال نمایند. سازمان‌ها دارای مخاطبانی نیز هستند که ممکن است در صورت اعمال نکردن سیاست‌ها و کنترل‌های امنیتی مناسب، خواسته یا ناخواسته تأثیر نامطلوب یا مخرب بر روی سیستم بگذارند. همچنین اطلاعات سازمان می‌تواند همواره برای اشخاص دیگر جالب و موردتوجه باشد و بخواهند با استفاده از ضعف‌های امنیتی موجود در سیستم، به اطلاعات حساس و حیاتی سازمان دسترسی پیدا کنند. اما تهدیدات سازمان تنها به کاربران ناشی، ضعف سیستم و اشخاص خرابکار ختم نمی‌شود؛ بلکه بلایای طبیعی مثل سیل و زلزله یا اتفاقات پیش‌بینی‌نشده مثل آتش‌سوزی و حتی مخاطرات نوظهور نیز تهدید به‌حساب می‌آیند.

مدیران سازمان‌ها درصورتی‌که بخواهند سطح امنیت خود را افزایش داده و از اطلاعات خود محافظت کنند باید استانداردهای امنیتی جامعی را مورداستفاده قرار دهند که تمام ابعاد امنیت اطلاعات را موردتوجه قرار داده باشد. سپس استاندارد موردنظر را توسط افراد متخصص و باتجربه در سازمان خود پیاده‌سازی نموده و همواره برای تداوم عملکرد مناسب، بر آن نظارت داشته باشند.

سازمان استاندارد جهانی (ISO) با همکاری کمیسیون بین‌المللی الکتروتکنیک (IEC) جهت امنیت اطلاعات، اقدام به معرفی استانداردهای سری ISO/IEC 27000 نموده است. در این استانداردها که سیستم مدیریت امنیت اطلاعات (Information Security Management System) یا به‌اختصار ISMS، نامیده می‌شوند؛ سعی شده تا تمام مؤلفه‌های موردنیاز امنیت اطلاعات، از امنیت فیزیکی تا نحوه مدیریت، مدنظر قرار داده شود.

در سال ۱۳۸۷ سازمان استاندارد ملی ایران نیز بر اساس استانداردهای 27001 و 27002 اقدام به معرفی استاندارد ملی امنیت نموده است. شما می‌توانید استاندارد ملی ایران را از طریق وب‌سایت این سازمان به آدرس www.isiri.org دانلود نمایید.

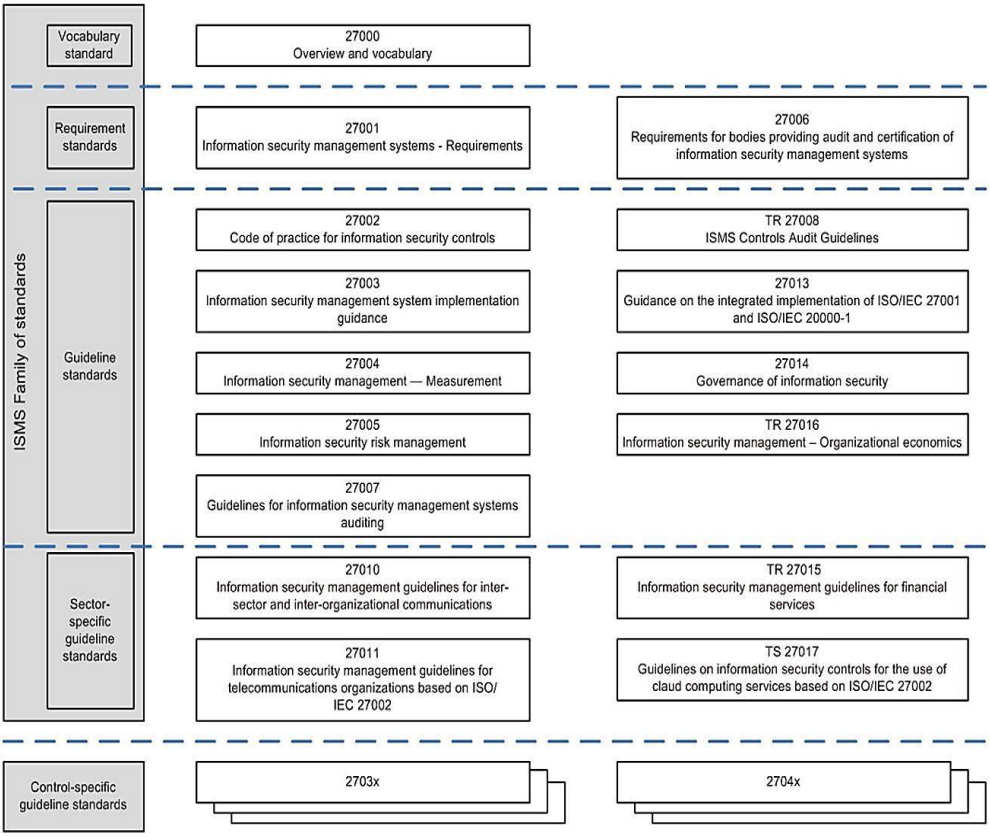
اما آخرین نسخه ارائه‌شده این استاندارد، ISO/IEC 27000:2014 است که در این مبحث به شرح مختصری از آن، که دربرگیرنده اصطلاحات و مرور کلی استانداردهای خانواده ISMS است، می‌پردازیم.

استانداردهای خانواده ISMS

خانواده ISMS متشکل از استانداردهای مرتبط باهم است که بعضی از آن‌ها در حال تدوین بوده و هنوز منتشرنشده و بعضی دیگر نیز منتشر گردیده است. این استانداردها دارای تعدادی مؤلفه مهم ساختاری می‌باشند که بر استانداردهای اصلی تمرکز دارند. استانداردهای اصلی شامل توصیف‌کننده الزامات ISMS (استاندارد ISO/IEC 27001) و الزامات مرکز صدور گواهینامه[1] (ISO/IEC 27006) جهت صادر کردن گواهی انطباق با ISO/IEC 27001 می‌باشند.

استانداردهای دیگر نیز فراهم‌کننده راهنما[2] برای جنبه‌های اجرایی مختلف ISMS، فرآیند عمومی، دستورالعمل‌های مربوط به کنترل و همچنین راهنمای یک بخش خاص می‌باشند.

تصویر زیر نشان‌دهنده ارتباطات درون خانواده استاندارد ISMS است.



[1] Certification
[2] Guidance

مروری بر استانداردهای خانواده ISMS

برای آشنایی با محتویات استانداردهای خانواده ISMS که در تصویر فوق نیز به آن‌ها اشاره گردیده است، به شرح مختصری از آن‌ها می‌پردازیم:

۱- **استاندارد ISO/IEC 27000 (همین استاندارد)**

طبقه‌بندی: فناوری اطلاعات- فنون امنیتی[1]- سیستم مدیریت امنیت اطلاعات- مرور کلی و واژگان

حوزه کاربرد: این استاندارد بین‌المللی به افراد و سازمان‌ها در زمینه‌های زیر کمک می‌کند:

i. بررسی اجمالی استانداردهای خانواده ISMS
ii. مقدمه‌ای بر سیستم مدیریت امنیت اطلاعات (ISMS)
iii. شرح مختصری بر فرآیند PDCA
iv. آشنایی با اصطلاحات و تعاریف استفاده‌شده در استانداردهای خانواده ISMS

هدف: استاندارد ISO/IEC 27000 به توصیف مبانی ISMS و اصطلاحات به‌کاررفته در این سری استاندارد می‌پردازد.

۲- **استاندارد ISO/IEC 27001**

طبقه‌بندی: فناوری اطلاعات- فنون امنیتی- سیستم مدیریت امنیت اطلاعات- الزامات[2]

حوزه کاربرد: این استاندارد بین‌المللی الزامات ایجاد، پیاده‌سازی، راه‌اندازی، نظارت، بررسی، نگهداری و بهبود رسمی سیستم مدیریت امنیت اطلاعات (ISMS) را درزمینه مخاطرات کلی کسب‌وکار سازمان تعیین می‌نماید. همچنین این استاندارد مشخص‌کننده الزامات پیاده‌سازی کنترل‌های امنیتی سفارشی‌شده موردنیاز سازمان‌های مختلف و یا بخش‌های وابسته به آن‌ها است. این استاندارد بین‌المللی تمام انواع سازمان‌ها (به‌عنوان‌مثال شرکت‌های تجاری، سازمان‌های دولتی و سازمان‌های غیرانتفاعی) را دربر می‌گیرد.

هدف: استاندارد ISO/IEC 27001 الزامات اصلی توسعه و راه‌اندازی ISMS را ارائه می‌نماید. این الزامات شامل مجموعه‌ای از کنترل‌ها برای مهار و کاهش خطرات مربوط به دارایی‌های اطلاعاتی بوده که سازمان می‌خواهد از طریق عملیاتی کردن ISMS از آن‌ها محافظت نماید.

۳- **استاندارد ISO/IEC 27006**

---

[1] Security Techniques
[2] Requirements

طبقه‌بندی: فناوری اطلاعات- فنون امنیتی- الزامات نهادهای ارائه‌دهنده خدمات ممیزی[1] و صدور گواهینامه ISMS

حوزه کاربرد: این استاندارد بین‌المللی علاوه بر الزامات موجود در ISO/IEC 17021، الزامات دیگری نیز مشخص نموده و راهنمایی‌هایی را برای نهادهای ارائه‌کننده ممیزی و صدور گواهینامه ISMS، طبق استاندارد 27001 فراهم می‌نماید. درواقع این استاندارد برای پشتیبانی از تائید صلاحیت نهادهایی است که بر اساس استاندارد 27001 اقدام به صدور گواهینامه ISMS می‌نمایند.

هدف: استاندارد ISO/IEC 27006 مکمل[2] استاندارد 17021 در جهت مشخص نمودن الزاماتی است که باید توسط نهادهایی که قصد ارائه خدمات ممیزی یا صدور گواهی‌نامه دارند، رعایت گردد؛ تا امکان ممیزی یا صدور گواهی انطباق با استاندارد 27001 توسط آن‌ها فراهم گردد.

۴- استاندارد ISO/IEC 27002

طبقه‌بندی: فناوری اطلاعات- فنون امنیتی- دستورالعمل[3] مدیریت امنیت اطلاعات

حوزه کاربرد: این استاندارد بین‌المللی لیستی از اهداف کنترلی پذیرفته‌شده و بهترین شیوه‌های[4] کنترلی، فراهم آورده تا جهت راهنمایی در زمان انتخاب و پیاده‌سازی کنترل‌ها برای دست‌یابی به امنیت اطلاعات، مورداستفاده قرار گیرند.

هدف: استاندارد ISO/IEC 27002 فراهم‌کننده راهنما جهت پیاده‌سازی کنترل‌های امنیت اطلاعات است. بندهای ۵ تا ۱۵ این استاندارد به‌طور خاص به مشاوره و راهنمایی پیاده‌سازی بر اساس بهترین شیوه‌ها در خصوص کنترل‌های موجود در بندهای A.5 تا A.15 استاندارد ISO/IEC 27001 است.

۵- استاندارد ISO/IEC 27003

طبقه‌بندی: فناوری اطلاعات- فنون امنیتی- راهنمای پیاده‌سازی ISMS

حوزه کاربرد: این استاندارد بین‌المللی ضمن فراهم‌سازی راهنمای پیاده‌سازی عملی، اطلاعات بیشتری را نیز برای برقراری، پیاده‌سازی، راه‌اندازی، نظارت، بررسی، نگهداری و بهبود ISMS طبق استاندارد 27001 ارائه می‌نماید.

هدف: استاندارد ISO/IEC 27003 رویکردی فرآیندگرا جهت اجرای موفقیت‌آمیز ISMS مطابق با استاندارد 27001 فراهم می‌آورد.

---

[1] Audit
[2] Supplement
[3] Code of practice
[4] Best practice

۶- **استاندارد ISO/IEC 27004**

طبقه‌بندی: فناوری اطلاعات- فنون امنیتی- مدیریت امنیت اطلاعات- سنجش[۱]

حوزه کاربرد: این استاندارد بین‌المللی راهنمایی و مشاوره برای توسعه و نحوه سنجش را به‌منظور ارزیابی اثربخشی ISMS، اهداف کنترلی و کنترل‌های استفاده‌شده برای پیاده‌سازی و مدیریت امنیت اطلاعات طبق استاندارد 27001، فراهم می‌نماید.

هدف: استاندارد ISO/IEC 27004 با ارائه چارچوب مشخصی برای سنجش، امکان ارزیابی تأثیرگذاری ISMS را مطابق با استاندارد 27001 فراهم می‌آورد.

۷- **استاندارد ISO/IEC 27005**

طبقه‌بندی: فناوری اطلاعات- فنون امنیتی- مدیریت ریسک امنیت اطلاعات

حوزه کاربرد: استاندارد بین‌المللی 27005 راهنمایی‌هایی را جهت مدیریت ریسک امنیت اطلاعات ارائه می‌نماید. روش‌های تشریح شده در این استاندارد بین‌المللی، از مفاهیم کلی مشخص‌شده در استاندارد 27001، پشتیبانی می‌کنند.

هدف: استاندارد ISO/IEC 27005 ارائه‌دهنده راهنمایی‌هایی جهت پیاده‌سازی رضایت‌بخش یک مدیریت ریسک فرآیندگرا و همچنین تحقق الزامات مدیریت ریسک امنیت اطلاعات بر اساس استاندارد 27001 است.

۸- **استاندارد ISO/IEC 27007**

طبقه‌بندی: فناوری اطلاعات- فنون امنیتی- راهنمای ممیزی ISMS

حوزه کاربرد: این استاندارد بین‌المللی علاوه بر شرایط موجود در استاندارد ISO 19011، شامل راهنمای طریقه انجام ممیزی ISMS و همچنین نحوه احراز صلاحیت بازرسان سیستم مدیریت امنیت اطلاعات، نیز است.

هدف: استاندارد ISO/IEC 27007 به ارائه راهنمایی برای سازمان‌هایی می‌پردازد که نیاز به انجام بازرسی داخلی یا خارجی ISMS داشته و یا برنامه ممیزی ISMS را جهت اجرای الزامات مشخص‌شده در استاندارد 27001 مدیریت می‌کنند.

۹- **استاندارد ISO/IEC TR 27008**[۲]

طبقه‌بندی: فناوری اطلاعات- فنون امنیتی- راهنمای بازرسان بر روی کنترل‌های امنیت اطلاعات

---

[1] Measurement

[۲] منظور از TR در این استاندارد، گزارش فنی یا Technical Report است.

حوزه کاربرد: این گزارش فنی، راهنمای نحوه بررسی پیاده‌سازی و بهره‌برداری از کنترل‌ها، شامل بررسی انطباق فنی کنترل‌های سیستم اطلاعات با استانداردهای امنیت اطلاعات سازمان را ارائه می‌دهد.

هدف: تمرکز این گزارش فنی بر روی بررسی کنترل‌های امنیت اطلاعات، ازجمله بررسی تطابق فنی با استاندارد امنیت اطلاعات که توسط سازمان پیاده‌سازی شده، می‌باشد. این استاندارد قصد ارائه راهنمایی خاصی جهت بررسی انطباق در مورد سنجش، ارزیابی ریسک یا ممیزی ISMS که در استانداردهای ISO/IEC 27004، ISO/IEC 27005 یا ISO/IEC 27007 مشخص گردیده را ندارد. این گزارش فنی برای سیستم‌های مدیریت ممیزی، نیز در نظر گرفته نشده است.

۱۰- استاندارد ISO/IEC 27013

طبقه‌بندی: فناوری اطلاعات- فنون امنیتی- راهنمای اجرای یکپارچه ISO/IEC 27001 و ISO/IEC 20000-1

حوزه کاربرد: این استاندارد بین‌المللی فراهم آورنده راهنمای اجرای یکپارچه ISO/IEC 27001 و ISO/IEC 20000-1، برای سازمان‌هایی است که قصد انجام یکی از کارهای زیر را دارند:

i. پیاده‌سازی ISO/IEC 27001 درزمانی که ISO/IEC 20000-1 قبلاً پذیرفته‌شده باشد؛ و بالعکس.
ii. پیاده‌سازی هر دو استاندارد ISO/IEC 27001 و ISO/IEC 20000-1 باهم.
iii. هم‌راستا نمودن مدیریت پیاده‌سازی سیستم‌های ISO/IEC 27001 و ISO/IEC 20000-1 موجود.

هدف: این استاندارد به‌منظور فراهم کردن درک بهتری برای سازمان، درباره ویژگی‌ها، شباهت‌ها و تفاوت‌های ISO/IEC 27001 و ISO/IEC 20000-1، ارائه شده است؛ تا کمکی در جهت برنامه‌ریزی یک سیستم مدیریت یکپارچه منطبق با هر دو استاندارد بین‌المللی، باشد.

۱۱- استاندارد ISO/IEC 27014

طبقه‌بندی: فناوری اطلاعات- فنون امنیتی- اداره امور امنیت اطلاعات

حوزه کاربرد: این استاندارد بین‌المللی فراهم آورنده راهنمای اصول و فرآیندهایی جهت اداره امور امنیت اطلاعات است؛ که از طریق آن، سازمان می‌تواند ارزیابی، هدایت و نظارت مدیریت امنیت اطلاعات را انجام دهد.

هدف: امنیت اطلاعات به یک موضوع کلیدی برای سازمان‌ها تبدیل‌شده و شکست اقدامات امنیت اطلاعات، می‌تواند به‌صورت مستقیم بر روی شهرت سازمان تأثیرگذار

باشد. بنابراین برای اطمینان از حصول اهداف سازمان، همواره باید بر امنیت اطلاعات نظارت گردد.

۱۲- استاندارد ISO/IEC TR 27016

طبقه‌بندی: فناوری اطلاعات- فنون امنیتی- مدیریت امنیت اطلاعات- اقتصاد سازمانی

حوزه کاربرد: این گزارش فنی روشی را ارائه می‌دهد که سازمان بتواند ازلحاظ اقتصادی درک بهتری از دارایی‌های اطلاعاتی خود داشته باشد. در این صورت سازمان با دقت بیشتری دارایی‌های اطلاعاتی خود را شناسایی نموده، خطرات بالقوه آن دارایی‌های اطلاعاتی را سنجیده، کنترل‌های حفاظت اطلاعات برای آن دارایی‌ها را تشخیص داده، و سطح بهینه منابعی که باید برای تأمین امنیت آن‌ها استفاده شود را تعیین می‌نماید.

هدف: این گزارش فنی مکمل استانداردهای خانواده ISMS است که با جمع‌آوری نقطه نظرات اقتصادی در حفاظت از دارایی‌های اطلاعاتی سازمان، از طریق مدل‌ها و نمونه‌ها، یک راهنمای اقتصاد سازمانی را ازلحاظ امنیت اطلاعات فراهم می‌آورد.

۱۳- استاندارد ISO/IEC 27010

طبقه‌بندی: فناوری اطلاعات- فنون امنیتی- مدیریت امنیت اطلاعات برای ارتباطات بین بخشی و درون‌سازمانی

حوزه کاربرد: این استاندارد بین‌المللی، علاوه بر راهنمایی‌های ارائه‌شده در خانواده استاندارد ISO/IEC 27000، دستورالعملی را برای اجرای مدیریت امنیت اطلاعات در محل‌های اشتراک‌گذاری اطلاعات، ارائه نموده و همچنین کنترل‌ها و راهنمایی‌های ویژه مربوط به شروع، اجرا، نگهداری و بهبود امنیت اطلاعات در ارتباطات بین بخشی و درون‌سازمانی را فراهم می‌کند.

هدف: این استاندارد بین‌المللی، قابل‌اجرا برای تمام اشکال تبادل و اشتراک‌گذاری اطلاعات حساس، در هر دو حالت عمومی و خصوصی، ملی و بین‌المللی، در همان بخش کاری و یا بین بخشی، است. به‌طور خاص این استاندارد می‌تواند قابل‌اجرا برای اشتراک‌گذاری یا تبادل اطلاعات مربوط به تأمین، نگهداری و محافظت از زیرساخت‌های حیاتی ملی و یا سازمانی، باشد.

۱۴- استاندارد ISO/IEC 27011

طبقه‌بندی: فناوری اطلاعات- فنون امنیتی- راهنمای مدیریت امنیت اطلاعات بر اساس استاندارد ISO/IEC 27002 برای سازمان‌های مخابراتی[1]

[1] Telecommunication

حوزه کاربرد: این استاندارد بین‌المللی شامل دستورالعمل نحوه پیاده‌سازی مدیریت امنیت اطلاعات (ISM) در سازمان‌های مخابراتی است.

هدف: استاندارد ISO/IEC 27011 علاوه بر الزامات موجود در ضمیمه A استاندارد 27001، دستورالعمل خاصی را با اقتباس از استاندارد 27002 برای سازمان‌های مخابراتی فراهم نموده است.

۱۵- استاندارد ISO/IEC TR 27015

طبقه‌بندی: فناوری اطلاعات- فنون امنیتی- دستورالعمل مدیریت امنیت اطلاعات برای خدمات مالی[1]

حوزه کاربرد: این گزارش فنی، علاوه بر دستورالعمل‌های ارائه‌شده در خانواده استاندارد ISO/IEC 27000، دستورالعملی را برای شروع، اجرا، نگهداری و بهبود امنیت اطلاعات در سازمان‌های ارائه‌دهنده خدمات مالی، ارائه می‌نماید.

هدف: این گزارش فنی یک مکمل تخصصی برای ISO/IEC 27001 و ISO/IEC 27002، جهت استفاده توسط سازمان‌های ارائه‌دهنده خدمات مالی است، تا آن‌ها را در موارد زیر حمایت نماید:

i. شروع، اجرا، نگهداری و بهبود یک سیستم مدیریت امنیت اطلاعات بر اساس استاندارد بین‌المللی ISO/IEC 27001:2005.

ii. طراحی و پیاده‌سازی کنترل‌های تعریف‌شده در ISO/IEC 27002:2005، یا در این استاندارد بین‌المللی.

۱۶- استاندارد ISO/IEC 27799

طبقه‌بندی: انفورماتیک بهداشت[2]- مدیریت امنیت اطلاعات در بهداشت بر اساس استاندارد 27002

حوزه عملکرد: این استاندارد بین‌المللی دستورالعمل پیاده‌سازی مدیریت امنیت اطلاعات (ISM) را در سازمان‌های بهداشت و درمان ارائه می‌نماید.

هدف: استاندارد ISO/IEC 27799 علاوه بر الزامات موجود در ضمیمه A استاندارد 27001، دستورالعمل خاصی را با اقتباس از استاندارد 27002 برای سازمان‌های بهداشت و درمان ارائه نموده است.

---

[1] Financial Services
[2] Health

اصطلاحات و تعاریف[1]

- **کنترل دسترسی (Access Control)**

اطمینان از دسترسی به دارایی‌ها به‌صورت مجاز و محدودشده بر اساس الزامات امنیتی و کسب‌وکار.

- **مسئولیت‌پذیری و پاسخ‌گویی (Accountability)**

مسئولیت‌پذیری یک نهاد در ازای تصمیمات و اقدامات خود.

- **دارایی (Asset)**

هر چیزی که برای سازمان ارزشمند باشد دارایی محسوب می‌گردد. دارایی می‌تواند شامل طیف گسترده‌ای از قبیل اطلاعات، برنامه‌های کاربردی، تجهیزات سخت‌افزاری، افراد و غیره باشد.

- **حمله (Attack)**

تلاش برای تخریب[2]، افشا[3]، تغییر[4]، غیرفعال کردن، سرقت[5] و دسترسی یا استفاده غیرمجاز[6] از یک دارایی را حمله می‌نامند.

- **احراز هویت (Authentication)**

ارائه تضمین، جهت درست بودن مشخصات ادعاشده.

- **اصالت (Authenticity)**

بررسی اینکه ارائه‌دهنده مشخصات، همان است که ادعا می‌کند.

- **دسترس‌پذیری (Availability)**

دارایی‌ها باید برای تقاضاهای مجاز همواره در دسترس و قابل‌استفاده باشد.

- **تداوم کسب‌وکار (Business Continuity)**

حصول اطمینان تداوم عملیات کسب‌وکار از طریق فرآیندها و رویه‌ها.

- **محرمانگی (Confidentiality)**

غیرقابل‌دسترس بودن یا آشکار نبودن اطلاعات برای افراد و فرآیندهای غیرمجاز را محرمانگی می‌گویند.

- **دارایی اطلاعاتی (Information Asset)**

---

[1] Terms and Definitions
[2] Destroy
[3] Expose
[4] Alter
[5] Steal
[6] Unauthorized

دانش یا دیتای دارای ارزش برای سازمان را دارایی‌های اطلاعاتی گویند.

- **امنیت اطلاعات (Information Security)**

منظور از امنیت اطلاعات حفظ محرمانگی، درستی و دسترس‌پذیری اطلاعات است.
البته امنیت اطلاعات علاوه بر موارد فوق می‌تواند شامل مواردی نظیر صحت، پاسخگویی[1]، انکارناپذیری و قابلیت اطمینان بودن اطلاعات نیز باشد.

- **رویداد امنیت اطلاعات (Information Security Event)**

یک رویداد امنیت اطلاعات می‌تواند شامل وقوع حالت شناخته‌شده‌ای در سیستم، شبکه یا سرویس باشد که نشان‌دهنده نقص احتمالی در امنیت اطلاعات، خط‌مشی یا مشکل در کنترل‌ها است. همچنین ممکن است رویداد شامل وضعیت ناشناخته‌ای باشد که به امنیت اطلاعات مربوط است.

- **حادثه امنیت اطلاعات (Information Security Incident)**

یک یا مجموعه‌ای از رویدادهای امنیت اطلاعات ناخواسته یا غیرمنتظره که به‌احتمال‌زیاد عملیات کسب‌وکار را به خطر انداخته و تهدیدی برای امنیت اطلاعات محسوب می‌گردد.

- **مدیریت حوادث امنیت اطلاعات (Information Security Incident Management)**

به مجموع فرآیندهایی که برای آشکارسازی[2]، گزارش دهی[3]، ارزیابی[4]، پاسخ‌دهی به[5]، رسیدگی به[6] و یادگیری حوادث امنیت اطلاعات مورداستفاده قرار می‌گیرد، مدیریت حوادث امنیت اطلاعات می‌گویند.

- **سیستم مدیریت امنیت اطلاعات (Information Security Management System)**

سیستم مدیریت امنیت اطلاعات یا ISMS، بخشی از سیستم کلان مدیریتی است که اساس آن بر رویکرد مخاطره کسب‌وکار بوده و شامل برقراری[7]، پیاده‌سازی، عملیات، نظارت، بررسی، نگهداری و بهبود امنیت اطلاعات است.

- **ریسک امنیت اطلاعات (Information Security Risk)**

توانایی بالقوه یک تهدید در بهره‌برداری از آسیب‌پذیری یک یا گروهی از دارایی‌ها که درنهایت منجر به آسیب رسیدن به سازمان می‌گردد.

---

[1] Accountability
[2] Detecting
[3] Reporting
[4] Assessing
[5] Responding to
[6] Dealing with
[7] Establish

- **درستی** [۱] **(Integrity)**

خاصیت حفاظت از صحت[۲] و تمامیت[۳] دارایی‌ها.

- **خط‌مشی (Policy)**

خط‌مشی یا سیاست، عبارت است از قصد و جهت‌دهی کلی سازمان که به‌صورت رسمی توسط مدیریت تبیین شده است.

- **فرآیند (Process)**

مجموعه فعالیت‌های مرتبط باهم که باعث تبدیل ورودی به خروجی می‌گردد.

- **رویه (Procedure)**

راه و روش مشخص‌شده برای انجام یک فعالیت یا فرآیند را رویه می‌گویند.

- **تهدید (Threat)**

عامل بالقوه حادثه‌ای ناخواسته که ممکن است منجر به آسیب یک سازمان یا سیستم گردد.

- **آسیب‌پذیری (Vulnerability)**

ضعف موجود در یک دارایی یا کنترل که می‌تواند توسط یک تهدید مورد سوءاستفاده قرار گیرد را آسیب‌پذیری می‌گویند.

مروری بر ISMS

سیستم مدیریت امنیت اطلاعات یا ISMS، مدلی را برای برقراری، پیاده‌سازی، راه‌اندازی[۴]، نظارت[۵]، بررسی[۶]، حفظ و نگهداری[۷]، و بهبود حفاظت از دارایی‌های اطلاعاتی به‌منظور تحقق اهداف کسب‌وکار فراهم می‌نماید. طراحی این مدل بر اساس ارزیابی ریسک و سطح پذیرش آن توسط سازمان انجام‌گرفته تا مدیریت و رفع ریسک به‌صورت مؤثر انجام پذیرد.

تجزیه‌وتحلیل الزامات حفاظت از دارایی‌های اطلاعاتی و اعمال کنترل‌های مناسب جهت حصول اطمینان از حفاظت آن‌ها، به پیاده‌سازی موفق ISMS کمک می‌نماید.

---

[۱] عبارت Integrity دارای معانی بسیاری ازجمله درستی، صحت، یکپارچگی و تمامیت است. اما به نظر بنده معنی قابل‌قبول در جملات این بخش، درستی است.

[2] Accuracy
[3] Completeness
[4] Operating
[5] Monitoring
[6] Reviewing
[7] Maintaining

اصول اساسی زیر می‌تواند کمک شایانی در جهت اجرای موفقیت‌آمیز ISMS باشد:

- آگاهی از ضرورت نیاز به امنیت اطلاعات.
- تخصیص مسئولیت برای امنیت اطلاعات.
- ترکیب تعهد مدیریتی و منافع ذینفعان.
- افزایش ارزش‌های اجتماعی.
- ارزیابی ریسک جهت تعیین کنترل‌های مناسب برای دستیابی به سطح قابل‌قبول ریسک.
- گنجاندن امنیت به‌عنوان یک عنصر ضروری در سیستم‌ها و شبکه‌های اطلاعاتی.
- پیشگیری و تشخیص فعال حوادث امنیت اطلاعات.
- حصول اطمینان از یک رویکرد جامع برای مدیریت امنیت اطلاعات.
- ارزیابی مستمر امنیت اطلاعات و ایجاد تغییرات مناسب.

دستیابی به امنیت اطلاعات نیازمند مدیریت مخاطرات ناشی از تهدیدات فیزیکی، انسانی و فناوری‌های مربوط به اطلاعات در تمام اشکال مورداستفاده سازمان است. لذا همواره بخشی از ISMS هر سازمان را مخاطرات مرتبط با دارایی‌های اطلاعاتی تشکیل می‌دهد.

انتظار می‌رود پیاده‌سازی ISMS در سازمان به‌عنوان یک تصمیم استراتژیک موردپذیرش قرار گیرد. این تصمیم لازم است مطابق با نیازهای سازمان کاملاً یکپارچه، متناسب و به‌روز باشد.

طراحی و پیاده‌سازی ISMS، تحت تأثیر نیازها، اهداف سازمانی، الزامات امنیتی، فرآیندهای کسب‌وکار، وسعت و ساختار آن سازمان خواهد بود. توجه داشته باشید که منافع و امنیت اطلاعات موردنیاز همه ذینفعان ازجمله مشتریان، تأمین‌کنندگان[1]، شرکای تجاری، سهامداران[2] و سایر اشخاص ثالث[3] باید در طراحی و راه‌اندازی ISMS منعکس گردد.

سیستم مدیریت امنیت اطلاعات(ISMS) برای کسب‌وکار هر دو بخش خصوصی و عمومی مهم است. این سیستم که برای فعالیت‌های مدیریت ریسک نیز ضروری است، هر صنعتی را قادر می‌سازد تا از کسب‌وکار الکترونیکی پشتیبانی نماید. زمانی که یک سازمان اقدام به اتخاذ استانداردهای خانواده ISMS می‌نماید، توانایی خود را در اعمال اصول امنیت اطلاعات متقابل و پایدار در برابر شرکای تجاری و سایر طرف‌های ذینفع نشان می‌دهد.

با همه این اوصاف در برخی موارد هنگام طراحی و توسعه سیستم‌های اطلاعات، ممکن است امنیت اطلاعات موردتوجه قرار نگیرد. علاوه بر این امنیت اطلاعات اغلب به‌عنوان یک راه‌حل فنی

[1] Suppliers
[2] Shareholders
[3] Third parties

در نظر گرفته می‌شود؛ **ولی امنیتی که از راه‌های فنی به دست می‌آید محدود بوده و حتی ممکن است بدون پشتیبانی توسط مدیریت و فقدان روش‌های اجرایی مناسب، بی‌تأثیر باشد.** همچنین گنجاندن امنیت داخل یک سیستم اطلاعاتی پس از پیاده‌سازی آن می‌تواند پرهزینه و پرزحمت باشد. در مقابل سیستم ISMS شامل تمام کنترل‌ها بوده و نیازمند برنامه‌ریزی دقیق و توجه به تمام جزئیات است. به‌عنوان‌مثال کنترل‌های دسترسی که ممکن است فنی (منطقی)، فیزیکی، اداری (مدیریتی) و یا ترکیبی از این‌ها باشد؛ ابزارهایی فراهم می‌آورد تا از دسترسی مجاز و محدودشده به دارایی‌های اطلاعاتی اطمینان حاصل شود.

امنیت اطلاعات با اجرای مجموعه‌ای از کنترل‌هایی به دست می‌آید که از طریق فرآیند مدیریت ریسک انتخاب و توسط ISMS مدیریت می‌شوند. این کنترل‌ها شامل سیاست‌ها (خط‌مشی‌ها)، فرآیندها، روش‌های اجرایی، ساختار سازمانی، نرم‌افزار و سخت‌افزارهایی است که در جهت حفاظت از دارایی‌های اطلاعاتی شناسایی‌شده است. برای حصول اطمینان از دستیابی به اهداف موردنظر امنیتی و کسب‌وکار سازمان، نیاز است که این کنترل‌ها مشخص، پیاده‌سازی، نظارت و بررسی گردیده و در صورت لزوم بهبود یابند. این انتظار وجود دارد که همواره کنترل‌های امنیت اطلاعات با فرآیندهای کسب‌وکار سازمان یکپارچه‌شده باشند.

اصول PDCA

استانداردهای سیستم مدیریتیISO دارای اصول بهره‌برداری مصوب بانامPDCA می‌باشند که این اصول در استانداردهای خانواده ISMS نیز مورداستفاده قرار می‌گیرد. منظور از PDCA، فرآیندهای Plan، Do، Check و Act است که در ادامه به تشریح آن‌ها می‌پردازیم:

۱- **برنامه (Plan)**

تعیین اهداف و ایجاد برنامه‌ها (تجزیه‌وتحلیل وضعیت سازمان، تعیین اهداف کلی و زیرمجموعه‌ها و درنهایت توسعه برنامه جهت دستیابی به آن‌ها).

۲- **اجرا (Do)**

اجرای برنامه‌ها (انجام کارهایی که در برنامه مشخص‌شده‌اند).

۳- **بررسی (Check)**

سنجش نتایج به‌دست‌آمده (سنجش و نظارت بر دستاوردها، تا مشخص شود چه حد به اهداف موردنظر نزدیک شده‌ایم).

۴- **اقدام (Act)**

اصلاح و بهبود فعالیت‌ها (به قول خودمون شکست پلی است برای پیروزی! درس گرفتن از اشتباهات گذشته به‌منظور بهبود فعالیت‌ها در دستیابی به نتایج بهتر).

برقراری، نظارت، نگهداری و بهبود عملکرد ISMS

یک سازمان جهت برقراری، نظارت، نگهداری و بهبود عملکرد ISMS، نیازمند تعهد به انجام مراحل زیر است:

۱- شناسایی دارایی‌های اطلاعاتی و الزامات امنیتی مربوط به آن‌ها.
۲- ارزیابی مخاطرات امنیت اطلاعات.
۳- انتخاب و پیاده‌سازی کنترل‌های مربوطه برای مدیریت مخاطرات غیرقابل‌قبول.
۴- نظارت، نگهداری و بهبود اثربخشی کنترل‌های امنیتی مرتبط با دارایی‌های اطلاعاتی سازمان.

برای اطمینان از عملکرد مؤثرISMS در محافظت از دارایی‌های اطلاعاتی سازمان، لازم است تا موارد فوق به‌صورت مداوم جهت شناسایی تغییرات ایجادشده در مخاطرات، راهبردهای سازمان و یا اهداف کسب‌وکار تکرار شوند.

☑ مبحث دوم

مرور استاندارد ISO/IEC 27001:2013

همان‌طور که در مبحث قبل گفته شد، استاندارد ISO/IEC 27001 شامل موارد زیر است:

طبقه‌بندی: فناوری اطلاعات- فنون امنیتی- سیستم مدیریت امنیت اطلاعات- الزامات

حوزه کاربرد: این استاندارد بین‌المللی الزامات ایجاد، پیاده‌سازی، راه‌اندازی، نظارت، بررسی، نگهداری و بهبود رسمی سیستم مدیریت امنیت اطلاعات (ISMS) را درزمینه مخاطرات کلی کسب‌وکار سازمان تنظیم می‌نماید. همچنین این استاندارد مشخص‌کننده الزامات پیاده‌سازی کنترل‌های امنیتی سفارشی‌شده موردنیاز سازمان‌های مختلف و یا بخش‌های وابسته به آن‌ها است. این استاندارد بین‌المللی تمام انواع سازمان‌ها (به‌عنوان‌مثال شرکت‌های تجاری، سازمان‌های دولتی و سازمان‌های غیرانتفاعی) را در برمی‌گیرد.

هدف: استاندارد ISO/IEC 27001 الزامات اصلی توسعه و راه‌اندازی ISMS را ارائه می‌نماید. این الزامات شامل مجموعه‌ای از کنترل‌ها برای مهار و کاهش خطرات مربوط به دارایی‌های اطلاعاتی بوده که سازمان می‌خواهد از طریق عملیاتی کردن ISMS از آن‌ها محافظت نماید.

سازمان ISO استاندارد ISMS را به‌روز کرده و نسخه ISO/IEC 27001:2013 را منتشر نموده است. لذا در این مبحث بخش‌های مهم استاندارد 27001 را بر اساس نسخه 2013 بررسی می‌کنیم. این استاندارد الزاماتی را که باید برای امنیت اطلاعات سازمانتان در نظر بگیرید مشخص می‌کند. این الزامات عمومی بوده و فارغ از اندازه سازمان و یا نوع کسب‌وکار آن است.

در گام اول سازمان باید تمام موارد داخلی و خارجی که در بحث امنیت اطلاعات برایش اهمیت دارد را شناسایی نموده و در زمان اجرای استاندارد مدنظر قرار دهد. سپس سایر موارد ذکرشده در این مبحث را برای رسیدن به اهداف خود دنبال نماید.

تعیین چارچوب سیستم مدیریت امنیت اطلاعات

سازمان باید مرز و دامنه اجرای سیستم مدیریت امنیت اطلاعات را مشخص نماید. برای تعیین این حوزه، باید به موارد زیر توجه گردد:

- مسائل و موارد داخلی و خارجی
- نیازمندی‌ها

- اینترفیس‌ها و وابستگی‌های بین فعالیت‌های انجام‌شده توسط این سازمان و فعالیت‌های اجراشده توسط سازمان‌های دیگر.

درنهایت باید چارچوب موردنظر به‌صورت مستندات سازمانی تهیه و آماده گردد.

رهبری و تعهد

مدیریت ارشد سازمان باید موارد زیر را برای رهبری و تعهد مربوط به امنیت اطلاعات در نظر بگیرد:

- باید از سازگار بودن سیاست‌های امنیت اطلاعات و اهداف امنیت اطلاعات که در حال برقراری می‌باشند، با مسیر استراتژیک سازمان اطمینان حاصل نماید.
- حصول اطمینان از یکپارچه‌سازی الزامات سیستم مدیریت امنیت اطلاعات با فرآیندهای سازمانی.
- حصول اطمینان از در دسترس بودن منابع موردنیاز برای اجرای ISMS.
- داشتن مدیریت امنیت اطلاعات مؤثر، به‌واسطه مطابقت آن با الزامات سیستم مدیریت امنیت اطلاعات.
- حصول اطمینان از اینکه سیستم مدیریت امنیت اطلاعات، باعث دستیابی به نتیجه موردنظر خواهد گردید.
- هدایت و حمایت از افراد جهت کمک و همکاری آنان در اثربخشی سیستم مدیریت امنیت اطلاعات.
- بهبود مستمر.
- حمایت از دیگر نقش‌های مدیریتی مرتبط، تا آن‌ها نیز بتوانند بر مناطق تحت کنترل خود مدیریت مؤثر داشته باشند.

سیاست‌گذاری و خط‌مشی

مدیریت ارشد سازمان در تدوین سیاست‌ها موظف به اجرای موارد زیر است:

- متناسب با اهداف سازمان باشد.
- شامل اهداف امنیت اطلاعات بوده و یا چارچوبی را برای تعیین اهداف امنیت اطلاعات فراهم آورد.
- متعهد به برآوردن نیازهای قابل‌اجرای مربوط به امنیت اطلاعات باشد.
- متعهد به بهبود مستمر سیستم مدیریت امنیت اطلاعات باشد.
- سیاست‌های تدوین‌شده در قالب مستندات سازمانی در دسترس باشد.

- شامل ارتباطات درون‌سازمانی باشد.
- به‌صورت مقتضی در دسترس افراد علاقه‌مند قرار گیرد.

نقش‌های سازمانی، مسئولیت‌ها و اختیارات

مدیریت ارشد سازمان موظف به مشخص کردن نقش‌های سازمانی در رابطه با سیستم مدیریت امنیت اطلاعات است. همچنین پس از مشخص نمودن نقش‌ها، مسئولیت‌ها و حدود اختیارات، باید آن‌ها را در سازمان مربوطه ابلاغ نماید.

تهیه گزارش در مورد عملکرد سیستم مدیریت امنیت اطلاعات برای ارائه به مدیر ارشد سازمان، ازجمله نقش‌های مهمی است که باید در نظر گرفته شود.

برنامه‌ریزی

در هنگام برنامه‌ریزی برای سیستم مدیریت امنیت اطلاعات، ضمن توجه به مسائل و الزامات ذکرشده، باید به تعیین ریسک‌ها و فرصت‌ها نیز اقدام نمایید.

ازجمله موارد موردنظر دیگر در زمان برنامه‌ریزی، می‌توان به جلوگیری یا کاهش اثرات نامطلوب و همچنین ارزیابی اثربخشی اقدامات انجام‌شده، اشاره نمود.

ارزیابی ریسک امنیت اطلاعات

سازمان باید اقدام به تعریف و به‌کارگیری فرآیندی جهت ارزیابی ریسک امنیت اطلاعات نماید. این فرآیند باید موارد زیر را لحاظ نموده باشد:

الف) ایجاد و حفظ اطلاعات مربوط به معیارهای ریسک امنیت اطلاعات که عبارتند از:

i. معیارهای پذیرش ریسک.
ii. معیارهای ارزیابی ریسک.

ب) حصول اطمینان از اینکه تکرار ارزیابی ریسک دارای نتایج سازگار، معتبر و قابل‌مقایسه باشد.

ج) شناسایی ریسک امنیت اطلاعات، شامل:

i. اعمال فرآیند ارزیابی ریسک برای شناسایی خطرات مرتبط با از دست دادن محرمانگی، درستی و در دسترس بودن اطلاعات.
ii. شناسایی صاحبان ریسک.

د) تجزیه‌وتحلیل خطرات امنیت اطلاعات شامل:

i. ارزیابی عواقب احتمالی، در صورت تحقق ریسک‌های شناسایی‌شده.
ii. ارزیابی واقع‌بینانه احتمال وقوع ریسک‌های شناسایی‌شده.
iii. تعیین سطح ریسک.

هـ) ارزیابی ریسک امنیت اطلاعات شامل:

i. مقایسه نتایج تجزیه‌وتحلیل ریسک با معیارهای ریسک تعیین‌شده در سازمان.
ii. اولویت‌دهی به تجزیه‌وتحلیل ریسک برای رفع آن.

مقابله با ریسک امنیت اطلاعات[1]

سازمان باید فرآیند مقابله با ریسک را برای موارد زیر تعریف و اعمال نماید:

- انتخاب گزینه‌های مناسب جهت مقابله با ریسک، بر اساس نتایج به‌دست‌آمده در ارزیابی ریسک.
- تعیین تمام کنترل‌های موردنیاز برای اجرای روش انتخاب‌شده جهت مقابله با ریسک.
- مقایسه کنترل‌های انتخاب‌شده فوق با موارد موجود در Annex A[2]، جهت بررسی اینکه هیچ کنترل ضروری حذف نشده باشد.

لازم به ذکر است که Annex A شامل فهرست جامعی از اهداف کنترل و کنترل‌ها است.

- تولید دستورالعمل‌های عملیات، شامل کنترل‌های لازم و توجیهی برای اجزاء، که آیا آن‌ها اجرا می‌شوند یا نه؛ و همچنین توجیهی برای استثنائات کنترل‌های Annex A.
- فرموله کردن و تدوین یک طرح مقابله با ریسک امنیت اطلاعات.
- مشخص کردن کسانی که در معرض ریسک هستند و پذیرش ریسک‌های باقی‌مانده.

اهداف امنیت اطلاعات و برنامه‌ریزی برای دستیابی به آن‌ها

سازمان باید اهداف امنیت اطلاعات را در توابع و سطوح مربوطه برقرار سازد. این اهداف باید مطابق با خط‌مشی و سیاست‌های سازمان، قابل‌سنجش و به‌روز باشند. سازمان باید در هنگام برنامه‌ریزی برای رسیدن به اهداف امنیتی، موارد زیر را مشخص نماید:

- چه چیز انجام خواهد گرفت؟
- چه منابعی موردنیاز خواهد بود؟
- چه شخصی پاسخگو خواهد بود؟

---

[1] Information security risk treatment

[2] برای دسترسی به پیوست A یا Annex A، به متن اصلی استاندارد مراجعه نمایید.

- چه زمانی به اتمام خواهد رسید؟
- چگونه نتایج به‌دست‌آمده ارزیابی شوند؟

صلاحیت و آگاهی

سازمان باید صلاحیت افرادی که کار آن‌ها، کنترل‌های امنیتی را تحت تأثیر قرار می‌دهد را بررسی نماید. باید اطمینان حاصل شود که این افراد آموزش‌های موردنیاز را گذرانده و صلاحیت لازم را به دست آورده‌اند.

همچنین افرادی که تحت کنترل‌های سازمان کار می‌کنند باید از سیاست‌های امنیت اطلاعات آگاهی داشته باشند. این افراد باید بدانند که سهم آن‌ها در اثربخشی سیستم مدیریت امنیت اطلاعات به چه میزان است.

ارتباط

سازمان باید ارتباط‌های موردنیاز داخلی و خارجی مرتبط با سیستم مدیریت امنیت اطلاعات را مشخص نماید. برای این کار باید به نکات زیر توجه شود:

- ارتباط در مورد چه؟
- چه زمانی ارتباط برقرار شود؟
- با چه کسانی ارتباط برقرار شود؟
- چه کسی باید ارتباط را برقرار نماید؟
- مشخص کردن فرآیندهای ارتباطات.

مستندسازی

سیستم مدیریت امنیت اطلاعات سازمان باید اطلاعات موردنیاز استاندارد بین‌المللی و همچنین اطلاعات ثبت‌شده سازمان در راستای اجرای استاندارد را به‌صورت مستندات سازمانی تهیه و نگهداری نماید. البته میزان اطلاعات ثبت‌شده در سازمان‌های مختلف بسته به‌اندازه سازمان، نوع فعالیت و شایستگی افراد می‌تواند متفاوت باشد.

این مستندات، پس از ایجاد باید به‌روزرسانی هم بشوند. اطلاعاتی از قبیل عنوان، تاریخ، نویسنده، قالب، زبان و نوع رسانه نیز باید در مستندات درج گردد.

سازمان باید همواره اطلاعات ثبت‌شده را کنترل نماید. نحوه توزیع، دسترسی، ذخیره‌سازی، حفظ و انتقال مستندات نیز ازجمله مواردی است که باید موردتوجه سازمان قرار داشته باشد.

برنامه‌ریزی و کنترل عملیاتی

سازمان باید فرآیندهای موردنیاز برای دستیابی به نیازهای امنیت اطلاعات و اقدامات تعیین‌شده در بخش‌های فوق را برنامه‌ریزی، پیاده‌سازی و کنترل نماید.

سازمان همچنین باید طرح‌هایی را برای دستیابی به اهداف امنیتی تعیین‌شده، اجرا نماید. سازمان باید تغییرات برنامه‌ریزی‌شده را کنترل و عواقب ناشی از تغییرات ناخواسته را بررسی نماید؛ تا در صورت لزوم، عوارض آن را کاهش دهد. سازمان باید اطمینان حاصل نماید که فرآیندهای برون‌سپاری مشخص و کنترل‌شده است.

در مورد فرآیندهای مربوط به ریسک نیز سازمان باید برنامه‌ریزی و کنترل داشته باشد. برای این کار سازمان باید در فواصل زمانی برنامه‌ریزی‌شده، یا زمانی که تغییرات قابل‌توجهی پیشنهاد یا انجام می‌شود، اقدام به ارزیابی ریسک‌های امنیتی نموده و مستندات مربوط به نتایج ارزیابی را حفظ و نگهداری نماید.

همچنین در مورد مقابله با ریسک نیز سازمان باید برنامه‌ای برای آن پیاده‌سازی نموده و همانند مراحل قبل مستندات مربوطه را نگهداری نماید.

ارزیابی کارایی[1]

اولین گام در ارزیابی کارایی، مانیتورینگ و تجزیه‌وتحلیل است. سازمان باید عملکرد و اثربخشی سیستم مدیریت امنیت اطلاعات را ارزیابی نماید. برای انجام این امر سازمان باید آنچه را که قرار است اندازه‌گیری شود، روش‌های نظارت، اندازه‌گیری و تجزیه‌وتحلیل را مشخص نماید. توجه داشته باشید که روش‌های اندازه‌گیری انتخاب‌شده باید قابل تکرار بوده و نتایج آن نیز قابل‌مقایسه باشند.

گام مهم بعدی، بازرسی داخلی[2] است. سازمان برای اینکه اطمینان حاصل نماید سیستم مدیریت امنیت اطلاعات مطابق با الزامات استاندارد بین‌المللی و الزامات مشخص‌شده سازمان، به‌طور مؤثر اجرا و نگهداری می‌شود، باید در فواصل زمانی مشخص اقدام به انجام بازرسی داخلی نماید. برای انجام بازرسی باید بازرس‌هایی انتخاب شوند که کاملاً بی‌طرف بوده و به آن‌ها اطمینان کامل وجود داشته باشد. همچنین باید نحوه بازرسی، زمان بازرسی، دامنه بازرسی و روش بازرسی مشخص‌شده باشد. البته باید اطمینان حاصل شود که نتایج به‌دست‌آمده از طریق بازرسی داخلی، ضمن مستندسازی و حفظ و نگهداری آن توسط سازمان، به اطلاع مدیران مربوطه هم رسیده باشد.

---

[1] Performance evaluation
[2] Internal audit

و اما گام سوم در ارزیابی کارایی، بازنگری مدیریت[1] است. منظور از بازنگری مدیریت، این است که مدیر ارشد سازمان باید برای اطمینان از تداوم مناسب و اثربخش بودن سیستم مدیریت امنیت اطلاعات، در فواصل زمانی مشخص اقدام به بررسی آن نماید.

ازجمله مواردی که مدیر ارشد در بحث بازنگری مدیریت باید به آن‌ها توجه نماید می‌توان به نظارت، بررسی نتایج بازرسی داخلی، تحقق اهداف، نتایج ارزیابی ریسک و بازخورد اشخاص علاقه‌مند، اشاره نمود.

خروجی بررسی مدیریت باید شامل تصمیم‌گیری‌های مربوط به بهبود مستمر و هرگونه نیاز به تغییرات در سیستم مدیریت امنیت اطلاعات باشد. البته همانند روش‌های قبلی باید این خروجی نیز به‌صورت مستندات سازمانی حفظ و نگهداری گردد.

بهبود[2]

گام اول در بهبود، بررسی عدم انطباق و اقدام اصلاحی است. زمانی که یک عدم انطباق روی می‌دهد سازمان باید به آن واکنش نشان داده و اقدام به کنترل، تصحیح و برخورد با عواقب آن نماید. سپس به ارزیابی نیاز، جهت اقدام برای از بین بردن آن عدم انطباق نماید. برای ارزیابی عدم انطباق باید ضمن بررسی آن، علل عدم انطباق و موارد مشابه بالقوه احتمالی آن را شناسایی نموده و در صورت لزوم اقدام به ایجاد تغییرات در سیستم مدیریت امنیت اطلاعات نمود. درنهایت باید نتایج هرگونه اقدام اصلاحی به‌صورت مستندات سازمانی نگهداری گردد.

گام دوم در بهبود، بهبود مستمر است. سازمان باید به‌طور مستمر، مناسب بودن، کفایت و اثربخشی سیستم مدیریت امنیت اطلاعات را بهبود بخشد.

نمودار کلی استاندارد ISO/IEC 27001:2013

برای درک بهتر استاندارد و همچنین مشخص شدن زیرشاخه‌های هر بخش، استاندارد فوق و بخش‌های مربوطه را در قالب نمودار و به‌صورت زیر برای شما آماده نموده‌ام.

[1] Management review
[2] Improvement

10
بهبود
عدم انطباق و اقدام اصلاحی
بهبود مستمر
9
ارزیابی
مانیتورینگ، اندازه گیری، تحلیل و ارزیابی
بازرسی داخلی
بازنگری مدیریت
8
عملیات
برنامه ریزی و کنترل عملیات
ارزیابی ریسک امنیت اطلاعات
مقابله با ریسک امنیت اطلاعات
7
پشتیبانی
منابع
صلاحیت
آگاهی رسانی
ارتباط
اطلاعات مستند
6
برنامه ریزی
اقدامات برای مدیریت ریسک و فرصت ها
اهداف امنیت اطلاعات و برنامه ریزی برای دستیابی به آنها
5
رهبری
رهبری و تعهد
خط مشی
نقش ها، وظایف و اختیارات سازمانی
4
ساختار سازمان
شناخت ساختار سازمانی
شناخت نیازها و انتظارات طرفهای ذینفع
تعیین محدوده سیستم مدیریت امنیت اطلاعات
سیستم مدیریت امنیت اطلاعات

فصل دوم

دستورالعمل
مدیریت امنیت اطلاعات

مبحث اول

مرور استاندارد ISO/IEC 27002:2013

☑ مبحث اول

مرور استاندارد ISO/IEC 27002:2013

همان‌طور که در مبحث اول گفته شد، استاندارد ISO/IEC 27002 شامل موارد زیر است:

طبقه‌بندی: فناوری اطلاعات- فنون امنیتی- دستورالعمل مدیریت امنیت اطلاعات

حوزه کاربرد: این استاندارد بین‌المللی لیستی از اهداف کنترلی پذیرفته‌شده و بهترین شیوه‌های کنترلی، فراهم آورده تا جهت راهنمایی در زمان انتخاب و پیاده‌سازی کنترل‌ها، برای دست‌یابی به امنیت اطلاعات، مورداستفاده قرار گیرند.

هدف: استاندارد ISO/IEC 27002 فراهم‌کننده راهنما جهت پیاده‌سازی کنترل‌های امنیت اطلاعات است. بندهای ۵ تا ۱۵ این استاندارد به‌طور خاص به مشاوره و راهنمایی پیاده‌سازی بر اساس بهترین شیوه‌ها در خصوص کنترل‌های موجود در بندهای A.5 تا A.15 استاندارد ISO/IEC 27001 است.

استاندارد 27002 برای آخرین بار در سال 2013 توسط سازمان ISO ویرایش و منتشر گردیده است. در این مبحث آخرین ویرایش این استاندارد مورد مرور کلی قرار می‌گیرد.

این استاندارد شامل دستورالعمل موارد مهم و موردنیاز در راستای پیاده‌سازی استاندارد 27001 در هر نوع سازمان و شیوه مدیریت امنیت اطلاعات شامل انتخاب، پیاده‌سازی و مدیریت کنترل‌ها با در نظر گرفتن شرایط ریسک امنیت اطلاعات در سازمان است. پیاده‌سازی موارد ذکرشده در این استاندارد بسته به نوع شرکت، کسب‌وکار، اندازه، کشور، سیاست‌ها و البته هزینه، می‌تواند در سازمان‌های مختلف، با برندها و محصولات متفاوتی صورت پذیرد.

ساختار استاندارد ISO/IEC 27002

این استاندارد شامل ۱۴ ماده کلی کنترل امنیت بوده که درمجموع حاوی ۳۵ گروه اصلی امنیت و ۱۱۴ کنترل است.

در هر ماده به تعریف کنترل‌های اصلی پرداخته می‌شود که شامل یک یا چند گروه اصلی امنیت خواهد بود. توجه داشته باشید که منظور از ماده در این استاندارد، اهمیت آن نیست! بلکه بسته به شرایط، کنترل‌های امنیتی می‌توانند از هر یا حتی همه ماده‌ها مهم‌تر باشند. بنابراین سازمان باید بر اساس فرآیندهای کسب‌وکار خود، کنترل‌های قابل‌اجرا را شناسایی نموده و تشخیص دهد که اجرای این کنترل‌ها چقدر می‌تواند دارای اهمیت باشد.

هر گروه اصلی امنیت شامل موارد زیر است:

- یک هدف کنترل که به بیان آن چیزی که می‌خواهیم به دست آوریم می‌پردازد.
- یک یا چند کنترل که می‌توانند برای رسیدن به هدف کنترل مورداستفاده قرار گیرند.

ساختار توصیف کنترل‌ها نیز به‌صورت زیر است:

- کنترل

معرفی کنترل خاص برای برآوردن هدف کنترل.

- راهنمای اجرایی

ارائه اطلاعات بیشتر برای اجرای کنترل و دست‌یابی به هدف موردنظر.

- سایر اطلاعات

درصورتی‌که نیاز به اطلاعات بیشتری مثل ملاحظات قانونی و ارجاع به دیگر استانداردها باشد، از این بخش استفاده می‌شود.



همان‌طور که گفته شد، این استاندارد دارای کنترل‌ها و گروه‌های متعددی است. در ادامه ذیل هر ماده اصلی، فقط به مواردی که به نظرم مهم‌تر بوده و در قالب مطالب این کتاب هستند پرداخته‌شده و از ذکر کنترل‌های شبیه به هم یا مواردی که کمتر مورداستفاده قرار می‌گیرند صرف‌نظر گردیده است. لذا توصیه می‌کنم در صورت نیاز، به متن اصلی استاندارد رجوع نمایید.

۵[1] – ماده ۱: سیاست‌های امنیت اطلاعات

خط‌مشی امنیت اطلاعات

- **کنترل**

مجموعه‌ای از سیاست‌های امنیت اطلاعات باید تدوین‌شده و پس از تصویب مدیریت، منتشر و در اختیار کارمندان سازمان و طرف‌های خارج از سازمان قرار گیرند.

- **راهنمای اجرایی**

در بالاترین سطح، سازمان‌ها باید سیاست‌های امنیت اطلاعات که توسط مدیریت تائید شده را تعریف نمایند. این سیاست‌ها مجموعه‌ای از رویکرد سازمان در مدیریت اهداف امنیت اطلاعات خود است. سیاست‌های امنیت اطلاعات باید بر اساس استراتژی کسب‌وکار، مقررات، قوانین و تهدیدات ایجادشده باشد.

---

[1] عدد ذکرشده قبل از هر ماده، بیانگر بخش موردنظر در متن اصلی استاندارد است که برای پیروی از ساختار استاندارد و در جهت سهولت دسترسی خوانندگان محترم، آورده شده است.

این سیاست‌ها باید شامل تعریف امنیت اطلاعات، اهداف و اصول فعالیت‌های مربوط به امنیت اطلاعات باشند. همچنین واگذاری مسئولیت‌های عمومی و اختصاصی برای مدیریت امنیت اطلاعات و فرآیند مواجهه با انحرافات و استثنائات نیز باید در این سیاست‌ها تعریف‌شده باشد.

اما در سطح پایین‌تر، سیاست امنیت اطلاعات باید به موضوعات خاص مثل کنترل دسترسی، امنیت فیزیکی، پشتیبان گیری و غیره بپردازد.

- **سایر اطلاعات**

در سازمان‌های بزرگ نیاز به سیاست‌های امنیت اطلاعات متفاوتی برای اجرا در سرتاسر سازمان است. سیاست‌ها و خط‌مشی‌های داخلی مخصوصاً برای سازمان‌های بزرگ بسیار مفید هستند. درصورتی‌که نیاز است این سیاست‌ها در اختیار طرف‌های بیرونی سازمان نیز قرار گیرند، باید به عدم افشای اطلاعات توجه گردد. در برخی از سازمان‌ها از این سیاست‌ها و خط‌مشی‌ها بانام‌های دیگری مثل استاندارد، بخشنامه و قوانین یاد می‌شود.

۶ – ماده ۲: امنیت اطلاعات سازمان

نقش‌ها و مسئولیت‌های امنیت اطلاعات

- **کنترل**

تمام مسئولیت‌های امنیت اطلاعات باید تعریف‌شده و اختصاص یابند.

- **راهنمای اجرایی**

تخصیص مسئولیت‌ها باید بر اساس سیاست‌های امنیت اطلاعات انجام پذیرد. این مسئولیت‌ها باید برای حفاظت هریک از دارایی‌ها و فرآیند جابجایی اطلاعات خاص تعریف گردد. همچنین مسئولیتی برای فعالیت‌های مدیریت ریسک و پذیرش خطرات باقی‌مانده، باید تعریف شود.

درصورتی‌که افراد مسئولیت‌های خود را به دیگران تنفیذ نمایند، بازهم شخص پاسخگو خود آن‌ها بوده و باید بر درستی انجام‌وظیفه مربوطه، نظارت داشته باشند.

حدود مسئولیت افراد باید مشخص شود، دارایی‌ها و فرآیندها تعریف‌شده و اختیارات شخص، متناسب با مسئولیتش تعیین گردد.

- **سایر اطلاعات**

در بسیاری از سازمان‌ها یک مدیر با مسئولیت کلی برای توسعه و پیاده‌سازی امنیت اطلاعات و همچنین پشتیبانی از شناسایی کنترل‌ها منصوب می‌شود؛ بااین‌حال در اغلب موارد، مسئولیت تأمین منابع و مسئولیت اجرای کنترل‌ها، بر عهده مدیریت‌های جداگانه باقی خواهند ماند.

خط‌مشی تجهیزات همراه

- **کنترل**

اتخاذ خط‌مشی و پشتیبانی از اقدامات امنیتی برای مدیریت ریسک‌هایی که زمان استفاده از تجهیزات همراه (قابل‌حمل)، ممکن است اتفاق افتد.

- **راهنمای اجرایی**

زمان استفاده از تجهیزات همراه، مخصوصاً در محیط‌های محافظت نشده، باید مراقبت‌های ویژه‌ای انجام پذیرد تا اطلاعات کسب‌وکار در معرض خطر قرار نگیرند.
در سیاست‌گذاری برای تجهیزات همراه، باید موارد زیر در نظر گرفته شود:

i. ثبت یا رجیستر تجهیزات همراه
ii. الزامات حفاظت فیزیکی
iii. محدود کردن نصب نرم‌افزار
iv. الزامات استفاده از نسخه به‌روز نرم‌افزار دستگاه و اعمال Patchها
v. محدودیت اتصال به سرویس‌های دیگر
vi. کنترل دسترسی
vii. استفاده از تکنیک‌های رمزنگاری
viii. محافظت در برابر نرم‌افزارهای مخرب
ix. غیرفعال سازی امکان پاک کردن و قفل‌کردن از راه دور (ریموت)
x. پشتیبان‌گیری
xi. استفاده از سرویس وب و برنامه‌های تحت وب
xii. جداسازی استفاده شخصی و سازمانی از تجهیزات همراه
xiii. دسترسی به اطلاعات، تنها پس از احراز هویت
xiv. آشنایی کاربر با وظایف امنیتی خود در زمان استفاده از تجهیزات همراه
xv. امکان پاک کردن اطلاعات روی تجهیزات از راه دور توسط سازمان؛ زمانی که دستگاه مفقودشده یا کاربر آن دیگر مجاز به‌حساب نمی‌آید.

- **سایر اطلاعات**

تجهیزات همراه که به‌صورت بی‌سیم به شبکه متصل می‌شوند، همانند سایر تجهیزات متصل به شبکه هستند، با این تفاوت که برخی پروتکل‌های امنیتی بی‌سیم هنوز به تکامل نرسیده و آسیب‌پذیری دارند. همچنین به دلیل پهنای باند کم ارتباطات بی‌سیم، ممکن است از اطلاعات موجود در تجهیزات همراه پشتیبان‌گیری به‌عمل‌نیامده باشد.

دورکـاری[1]

- **کنترل**

اتخاذ خط‌مشی و پشتیبانی از اقدامات امنیتی برای محافظت از اطلاعاتی که قابل‌دسترسی، پردازش و یا ذخیره‌شده بر روی سایت راه دور هستند.

- **راهنمای اجرایی**

سازمان‌هایی که دارای سایت راه دور هستند و قصد دارند به کارمندان خود خدمات دورکاری ارائه نمایند، باید اقدام به تعریف سیاست‌ها، شرایط و محدودیت‌های دورکاری نمایند.

برای تدوین خط‌مشی دورکاری باید نکات زیر در نظر گرفته شود:

i. امنیت فیزیکی سایت راه دور
ii. پیشنهاد محیط فیزیکی جهت دورکاری
iii. نیازمندی‌های امنیتی جهت ارتباط. این نکته بسیار مهم است، چراکه قرار است اطلاعات با استفاده از این خط ارتباطی انتقال یابند.
iv. ارائه دسکتاپ مجازی. این کار باعث می‌شود پردازش‌ها و ذخیره اطلاعات بر روی سیستم موجود در سایت انجام پذیرد و نه بر روی سیستم شخصی کاربر راه دور.
v. امکان دسترسی غیرمجاز افراد دیگر مثل اعضای خانواده، به اطلاعات و منابع.
vi. استفاده از شبکه‌های خانگی. الزامات یا محدودیت‌های تنظیمات اتصالات بی‌سیم.
vii. جلوگیری از اختلافات حقوق مالکیت معنوی در تجهیزات خصوصی.
viii. دسترسی سازمان به تجهیزات خصوصی افراد(جهت بررسی امنیت دستگاه) که ممکن است به دلیل حفظ حریم خصوصی افراد، منع قانونی داشته باشد.

[1] Teleworking

ix. توافق‌نامه لایسنس نرم‌افزار. البته این مورد در صورتی لازم است که سازمان، مسئول ارائه لایسنس نرم‌افزار مورداستفاده بر روی تجهیزات شخصی کارمندان یا طرف‌های خارجی باشد.
x. الزامات فایروال و حفاظت در مقابل بدافزارها.

همچنین راهنما و دستورالعمل دورکاری باید شامل موارد زیر گردد:
i. ارائه تجهیزات مناسب و دستگاه‌های ذخیره‌سازی برای فعالیت‌های دورکاری؛ درزمانی که استفاده از تجهیزات شخصی مجاز نیست.
ii. مشخص نمودن کار مجاز، ساعت کاری و طبقه‌بندی اطلاعاتی که کاربر ممکن است نگهداری کند و سرویس‌ها و سیستم‌هایی که مجاز به دسترسی هستند.
iii. ارائه تجهیزات ارتباطی مناسب، ازجمله روشی برای تأمین امنیت دسترسی از راه دور.
iv. امنیت فیزیکی.
v. قوانین و راهنمایی جهت دسترسی اعضای خانواده و افراد دیگر به تجهیزات و اطلاعات.
vi. ارائه پشتیبانی و نگهداری سخت‌افزار و نرم‌افزار.
vii. ارائه بیمه.
viii. روشی برای پشتیبان‌گیری و تداوم کسب‌وکار.
ix. بازرسی و نظارت امنیتی.
x. لغو اختیارات و حقوق دسترسی و برگشت تجهیزات درزمانی که فعالیت دورکاری خاتمه یافته است.

- **سایر اطلاعات**

عبارت دورکاری به تمام اشکال کار در خارج از سازمان، اشاره دارد.

۷ – ماده ۳: امنیت منابع انسانی

قوانین و شرایط استخدام

- **کنترل**

توافق یک قرارداد با کارمندان و پیمانکاران که حاکی از مسئولیت آنان و سازمان برای امنیت اطلاعات باشد.

- **راهنمای اجرایی**

تعهدات قرارداد کارمندان و پیمانکاران باید بر اساس سیاست‌های امنیت اطلاعات سازمان باشد. مضاف بر اینکه باید حاکی از موارد زیر هم باشد:

i. تمام کارمندان و پیمانکارانی که قرار است به اطلاعات محرمانه دسترسی داشته باشند، قبل از اجازه دسترسی به امکانات پردازش اطلاعات، باید اقدام به امضای تعهدنامه محرمانگی یا توافقنامه عدم افشا[1]، نمایند.

ii. حقوق و مسئولیت‌های قانونی کارمندان و پیمانکاران. به‌عنوان‌مثال راجع به قانون حق‌نشر[2] یا قوانین حفاظت از داده‌ها.

iii. مسئولیت‌های طبقه‌بندی اطلاعات، مدیریت دارایی‌های اطلاعاتی سازمان، امکانات پردازش اطلاعات و خدمات اطلاعاتی بکار گرفته‌شده توسط کارمندان یا پیمانکاران.

iv. مسئولیت‌های کارمندان یا پیمانکاران برای به‌کارگیری اطلاعات دریافت شده از طرف‌های خارجی یا سایر شرکت‌ها.

v. اقدامات در نظر گرفته‌شده، در صورت نادیده گرفتن الزامات امنیتی سازمان توسط کارمند یا پیمانکار.

قبل از استخدام باید مسئولیت‌ها و نقش‌های امنیت اطلاعات، به متقاضی شغل موردنظر ابلاغ گردد.

- **سایر اطلاعات**

می‌توان از یک کد رفتاری برای کارمندان یا پیمانکاران جهت مسئولیت‌های امنیت اطلاعات راجع به محرمانگی، حفاظت از داده‌ها، اصول اخلاقی، استفاده مناسب از تجهیزات و امکانات سازمان و همچنین شیوه‌های مورد انتظار سازمان، استفاده کرد.

۸ – ماده ۴: مدیریت دارایی

فهرست دارایی‌ها

- **کنترل**

دارایی‌های مرتبط با اطلاعات و امکانات پردازش اطلاعات، باید شناسایی‌شده و یک فهرست جامع از آن‌ها تهیه و نگهداری گردد.

---

[1] Non-Disclosure Agreements (NDA)

[2] Copyright

- **راهنمای اجرایی**

سازمان باید دارایی‌های مرتبط با چرخه اطلاعات را شناسایی کرده و آن‌ها را در قالب مستندات سازمانی نگهداری نماید.

چرخه اطلاعات باید شامل ایجاد، پردازش، ذخیره‌سازی، انتقال، حذف و تخریب باشد.

لیست دارایی‌ها باید دقیق، به‌روز، نامتناقض و همسو با دیگر فهرست‌ها باشد. برای هر دارایی شناسایی‌شده، باید مالک دارایی اختصاص داده شود و گروه‌بندی آن نیز باید مشخص گردد.

- **سایر اطلاعات**

فهرست دارایی‌ها کمک می‌کند تا از انجام حفاظت مؤثر اطمینان حاصل شود. همچنین این فهرست ممکن است برای مقاصد دیگر ازجمله دلایل مالی (مدیریت دارایی‌ها)، بهداشتی و ایمنی یا بیمه مورداستفاده قرار گیرد. توجه داشته باشید که داشتن فهرست دارایی، یکی از پیش‌شرط‌های مهم مدیریت ریسک است.

طبقه‌بندی اطلاعات

- **کنترل**

اطلاعات باید ازلحاظ الزامات قانونی، ارزش، حیاتی بودن و حساسیت نسبت به افشای غیرمجاز یا تغییر، طبقه‌بندی گردند.

- **راهنمای اجرایی**

طبقه‌بندی و کنترل‌های حفاظتی مربوطه، باید بر اساس نیازهای کسب‌وکار برای اشتراک‌گذاری یا اعمال محدودیت بر روی اطلاعات و همچنین الزامات قانونی، صورت پذیرد. دارایی‌های اطلاعاتی دیگر نیز می‌توانند بر اساس اطلاعاتی مثل: ذخیره‌شده در، پردازش‌شده توسط، به‌طور دیگر بکار گرفته‌شده یا محافظت‌شده توسط، طبقه‌بندی شوند. صاحبان دارایی‌های اطلاعاتی، باید پاسخگوی طبقه‌بندی آن‌ها باشند.

طرح طبقه‌بندی باید شامل قراردادها و معیارهایی برای بازنگری طبقه‌بندی در طول زمان باشد. سطح حفاظتی در این طرح باید با تجزیه‌وتحلیل محرمانگی، در دسترس بودن، درستی و هر نیاز دیگر که برای اطلاعات در نظر گرفته‌شده، ارزیابی گردد. این طرح باید هم‌راستا با سیاست‌های کنترل دسترسی باشد.

طرح باید شامل تمام فرآیندهای سازمان بوده و قابل‌اجرا در تمامی بخش‌های آن باشد. به‌طوری‌که هر کس قصد طبقه‌بندی اطلاعات و دارایی‌های مرتبط با آن را داشته باشد، از روش یکسانی استفاده نماید.

- **سایر اطلاعات**

ایجاد گروه‌هایی با سطح حفاظتی مشابه و تعیین روش‌های امنیت بر اساس هر گروه، باعث تسهیل مدیریت امنیت اطلاعات می‌گردد.

ممکن است پس از یک مدت‌زمان خاص، از اهمیت اطلاعات کاسته شده و از حالت حساس و بحرانی خارج شوند. در این صورت دیگری نیازی به‌صرف هزینه اضافی برای حفاظت از آن‌ها نیست. از طرف دیگر طبقه‌بندی اشتباه اطلاعات حساس می‌تواند تداوم کسب‌وکار سازمان را به خطر اندازد. لذا باید در زمان طبقه‌بندی محرمانگی اطلاعات، مشخص شود که افشای آن تا چه حد می‌تواند باعث آسیب به سازمان گردد. چهار سطح برای آسیب باید در نظر گرفته شود: ۱. افشای اطلاعات هیچ آسیبی نمی‌رساند ۲. باعث آسیب جزئی می‌شود ۳. اثر کوتاه‌مدت قابل‌توجهی بر روی عملیات یا اهداف تاکتیکی دارد. ۴. بقای سازمان را در معرض خطر قرار می‌دهد.

اداره دارایی

- **کنترل**

دستورالعمل اداره دارایی‌ها مطابق با طرح طبقه‌بندی اطلاعات سازمان، باید ایجاد و منتشر گردد.

- **راهنمای اجرایی**

دستورالعمل باید بر اساس به‌کارگیری، پردازش، ذخیره‌سازی و انتقال اطلاعات ایجادشده و همچنین مطابق با طبقه‌بندی آن‌ها باشد.

در زمان ایجاد دستورالعمل، موارد زیر باید در نظر گرفته شود:

i. اعمال محدودیت‌های دسترسی جهت پشتیبانی از الزامات حفاظتی در هر سطح از طبقه‌بندی.
ii. نگهداری یک رکورد رسمی از دسترسی افراد مجاز به دارایی‌ها.
iii. حفاظت از کپی موقتی یا دائم اطلاعات، در همان سطح حفاظت انجام‌شده از اطلاعات اصلی.
iv. ذخیره‌سازی دارایی‌های IT مطابق با مشخصات خواسته‌شده تولیدکنندگان.
v. علامت‌گذاری واضح برای تمام نسخه‌های رسانه، جهت مشخص کردن افرادی که مجاز به دریافت آن‌ها هستند.

مدیریت رسانه‌های قابل‌حمل

- **کنترل**

باید مطابق با طرح طبقه‌بندی سازمان، دستورالعملی برای مدیریت رسانه‌های قابل‌حمل ایجاد گردد.

- **راهنمای اجرایی**

موارد زیر برای مدیریت رسانه‌های قابل‌حمل باید در نظر گرفته شود:

i. درصورتی‌که سازمان، دیگر قصد استفاده از یک رسانه قابل‌استفاده مجدد[1] را نداشته باشد، آن رسانه باید غیرقابل‌بازیابی شود.
ii. در صورت لزوم و امکان، باید حذف رسانه از سازمان را منوط به صدور مجوز کرده و سپس آن مجوز برای بازرسی‌های آینده نگهداری شود.
iii. تمام رسانه‌ها باید در محیطی امن و ایمن[2]، مطابق با مشخصات خواسته‌شده تولیدکنندگان، نگهداری شوند.
iv. درصورتی‌که محرمانگی و صحت اطلاعات موجود بر روی رسانه‌ها مدنظر باشد، باید از تکنیک‌های رمزنگاری استفاده نمود.
v. برای کاهش ریسک از دست دادن اطلاعات موجود بر روی رسانه‌ها، باید قبل از خرابی رسانه، اطلاعات آن بر روی رسانه جدید انتقال یابد.
vi. برای کاهش خطر آسیب‌دیدگی و یا از دست دادن اطلاعات، باید نسخه‌های متعدد از داده‌های ارزشمند بر روی رسانه‌های جداگانه ذخیره شود.
vii. ثبت رسانه‌های قابل‌حمل برای جلوگیری از دست رفتن داده‌ها.
viii. درایورهای مربوط به رسانه‌ها تنها زمانی باید فعال باشند که توسط سازمان مورداستفاده قرار می‌گیرند.
ix. انتقال دیتا به رسانه‌های قابل‌حمل باید نظارت شود.

درنهایت دستورالعمل و مجوزهای دسترسی باید به‌صورت مستندات سازمانی تهیه و نگهداری گردد.

[1] re-usable
[2] Safe and Secure

دفع رسانه[1]

- **کنترل**

درصورتی‌که سازمان دیگر قصد استفاده از رسانه‌ای را ندارد، طبق یک دستورالعمل مشخص باید اقدام به دفع ایمن آن نماید.

- **راهنمای اجرایی**

برای به حداقل رساندن خطر نشت اطلاعات، نیاز به یک دستورالعمل جهت دفع امن رسانه‌های غیرقابل استفاده داریم. این روش‌ها باید متناسب با حساسیت اطلاعات مورداستفاده قرار گیرند.

برای ایجاد دستورالعمل دفع امن رسانه، موارد زیر باید در نظر گرفته شود:

i. رسانه حاوی اطلاعات محرمانه باید دفع ایمن شود. به‌عنوان‌مثال: سوزاندن، ریز کردن و یا پاک کردن اطلاعات برای استفاده توسط برنامه دیگری در داخل همان سازمان.
ii. روش دفع ایمن، باید همواره در دسترس باشد.
iii. وقایع‌نگاری از دفع اقلام حساس، به‌منظور استفاده جهت بازرسی.
iv. بعضی مواقع ممکن است دفع یکجای رسانه‌ها، روشی ساده‌تر باشد تا آنکه تلاش کنید رسانه‌ها را ازنظر حساسیت، تفکیک نمایید.
v. برخی از شرکت‌ها پیشنهاد انجام سرویس‌های مربوط به جمع‌آوری و دفع اطلاعات را به سازمان‌ها می‌دهند. در این صورت باید با دقت شرکتی باتجربه را انتخاب نموده و بر کار آن نظارت نمایید.

- **سایر اطلاعات**

تجهیزات آسیب‌دیده که حاوی اطلاعات حساسی بوده‌اند، به یک ارزیابی ریسک نیاز دارند تا بتوان تصمیم گرفت که تجهیزات را باید به‌صورت فیزیکی نابود کرد و یا برای تعمیر به شرکت مربوطه ارسال کرد و یا اینکه باید دور انداخته شود!

۹ – ماده ۵: کنترل دسترسی

سیاست‌های کنترل دسترسی

- **کنترل**

[1] Disposal of media

باید بر اساس الزامات کسب‌وکار و امنیت اطلاعات، یک سیاست کنترل دسترسی ایجاد، مستندسازی و بازنگری گردیده باشد.

- **راهنمای اجرایی**

صاحبان دارایی‌ها باید قوانین کنترل دسترسی مناسب، حقوق دسترسی و محدودیت برای کاربران نقش‌های خاص را برای دارایی‌های خود مشخص نمایند. میزان جزئیات و سخت‌گیری‌های این کنترل‌ها، باید نشأت‌گرفته از ریسک امنیت اطلاعات باشد.
کنترل دسترسی به دو صورت فیزیکی و منطقی بوده و این دو باید باهم مدنظر قرار گیرند. کاربران و ارائه‌دهندگان خدمات باید یک شرح واضح از نیازمندی‌های خود جهت کنترل دسترسی ارائه نمایند.
برای ایجاد سیاست کنترل دسترسی باید به موارد زیر توجه نمایید:

i. نیازمندی‌های امنیتی برنامه‌های کاربردی کسب‌وکار.
ii. خط‌مشی‌هایی برای انتشار اطلاعات و صدور مجوز. به‌عنوان‌مثال توجه به اصل Need to Know، سطوح امنیت اطلاعات و طبقه‌بندی اطلاعات.
iii. هماهنگی بین حقوق دسترسی و سیاست‌های طبقه‌بندی اطلاعات سیستم‌ها و شبکه‌ها.
iv. قوانین مربوطه و هرگونه تعهدات قراردادی در مورد محدودیت‌های دسترسی به داده‌ها یا سرویس‌ها.
v. مدیریت حقوق دسترسی در محیط شبکه و توزیع‌شده‌ای که تمام انواع اتصالات در دسترس را دارد.
vi. تفکیک نقش‌های کنترل دسترسی. مثل درخواست دسترسی، مجوز دسترسی و مدیریت دسترسی.
vii. الزامات برای مجوز رسمی درخواست‌های دسترسی.
viii. الزامات بازنگری دوره‌ای حقوق دسترسی.
ix. حذف حقوق دسترسی.
x. آرشیو سوابق مربوط به رویدادهای مهم؛ در رابطه با استفاده، مدیریت هویت کاربر و اطلاعات احراز هویت.
xi. نقش‌هایی با سطح دسترسی ممتاز.

- **سایر اطلاعات**

در زمان تهیه سیاست کنترل دسترسی باید بر اساس این فرض: "**همه‌چیز به‌طورکلی ممنوع است، مگر صریحاً اجازه داده‌شده باشد**"، اقدام نمایید.

قوانین کنترل دسترسی باید به‌صورت دستورالعمل‌های رسمی ارائه‌شده و برای آن مسئولیت تعریف گردد.

کنترل دسترسی مبتنی بر نقش[1]، توسط بسیاری از سازمان‌ها با موفقیت مورداستفاده قرارگرفته است. در این حالت اجازه دسترسی به کاربر بر اساس نقش سازمانی آن تعیین می‌شود.

دو اصل مکرر سیاست‌های کنترل دسترسی عبارتند از:

i. **Need-to-know**

شما تنها به اطلاعاتی دسترسی خواهید داشت که برای انجام وظایف سازمانی به آن‌ها نیاز دارید. نقش و وظیفه مختلف یعنی Need-to-know متفاوت؛ و درنهایت پروفایل دسترسی متفاوت.

ii. **Need-to-use**

شما تنها به امکانات پردازش اطلاعاتی دسترسی خواهید داشت[2] که برای وظیفه، شغل و یا نقش شما در سازمان لازم است.

منظور از امکانات پردازش اطلاعات مواردی مانند تجهیزات IT، برنامه‌های کاربردی، دستورالعمل‌ها و اتاق‌ها است.

دسترسی به شبکه و سرویس‌های شبکه

- **کنترل**

کاربران فقط باید به شبکه و سرویس‌هایی که مشخصاً به آن‌ها اجازه داده‌شده، دسترسی داشته باشند.

- **راهنمای اجرایی**

خط‌مشی باید موارد زیر را پوشش دهد:

i. شبکه و سرویس‌هایی که اجازه دسترسی به آن‌ها داده‌شده است.

ii. رویه‌های مجوزدهی؛ برای اینکه مشخص شود هرکسی اجازه دسترسی به کدام شبکه و سرویس‌های شبکه را دارد.

iii. کنترل‌های مدیریتی و دستورالعمل محافظت از اتصالات شبکه و سرویس‌ها.

iv. ابزار مورداستفاده برای دسترسی به شبکه و سرویس‌های شبکه. مثل استفاده از VPN یا شبکه‌های بی‌سیم.

---

[1] Role based access control

[2] granted

v. الزامات احراز هویت کاربر برای دسترسی به سرویس‌های مختلف شبکه.
vi. نظارت بر استفاده از سرویس‌های شبکه.

سیاست استفاده از خدمات شبکه، باید مطابق باسیاست‌های کنترل دسترسی سازمان باشد.

- **سایر اطلاعات**

اتصالات ناامن و غیرمجاز به سرویس‌های شبکه، می‌تواند سازمان را تحت تأثیر قرار دهد. این کنترل بخصوص برای اتصال به برنامه‌های کاربردی حساس و حیاتی سازمان، یا کاربرانی که در محیط‌های پرخطر هستند (اماکن عمومی و یا هر مکان دیگری که تحت مدیریت سازمان نباشد)، مهم است.

ثبت و لغو کاربر[1]

- **کنترل**

برای اینکه قادر به تخصیص حقوق دسترسی به کاربران باشید؛ باید ثبت‌نام و لغو ثبت‌نام کاربر دارای یک فرآیند مشخص و رسمی باشد.

- **راهنمای اجرایی**

با استفاده از شناسه کاربری[2] منحصربه‌فرد، کاربران مسئول پاسخگویی در قبال اعمال خود خواهند بود.

درصورتی‌که کاربر سازمان را ترک کرد، بلافاصله باید شناسه کاربری او را غیرفعال نموده و اطمینان حاصل شود که آن نام کاربری به شخص دیگری اعطا نشده باشد. همچنین باید در فواصل زمانی مشخص به بررسی شناسه‌های کاربری موجود پرداخته شود.

- **سایر اطلاعات**

ارائه یا لغو دسترسی به اطلاعات یا امکانات پردازش اطلاعات، معمولاً به دو روش زیر صورت می‌پذیرد:

i. تخصیص دادن، فعال کردن یا لغو شناسه کاربری. (یعنی خود شناسه کاربری تخصیص، فعال و یا لغو شود.)
ii. ارائه و یا لغو حقوق دسترسی اعطاشده به شناسه کاربری موردنظر.

[1] User registration and de-registration
[2] User ID

محدودیت دسترسی به اطلاعات

- **کنترل**

دسترسی به اطلاعات و برنامه‌های کاربردی باید مطابق باسیاست‌های کنترل دسترسی محدودشده باشند.

- **راهنمای اجرایی**

محدودیت دسترسی باید بر اساس الزامات هر یک از برنامه‌های کسب‌وکار و مطابق با سیاست‌های کنترل دسترسی باشد.

موارد زیر باید برای محدودیت دسترسی مدنظر قرار گیرند:

i. ارائه منوهایی برای کنترل دسترسی به توابع سیستمی برنامه کاربردی.
ii. کنترل اینکه هر کاربر چه دیتایی را می‌تواند ببیند.
iii. کنترل حقوق دسترسی کاربران، مثل: Read، Write، Delete و Execute.
iv. کنترل دسترسی به برنامه‌های کاربردی دیگر.
v. محدود کردن اطلاعات خروجی.
vi. فراهم‌سازی کنترل‌های دسترسی فیزیکی و منطقی، جهت جداسازی برنامه‌های کاربردی، داده‌های برنامه یا سیستم‌های حساس.

سیستم مدیریت رمزعبور

- **کنترل**

سیستم مدیریت رمزعبور باید به‌صورت تعاملی[1] بوده و باید کیفیت رمزهای عبور را تضمین نماید.

- **راهنمای اجرایی**

یک سیستم مدیریت رمزعبور باید دارای خصوصیات زیر باشد:

i. اجبار به استفاده از شناسه کاربری و رمزعبور جداگانه، برای حفظ مسئولیت‌پذیری.
ii. اجازه به کاربران برای انتخاب و تغییر رمزعبور خود؛ و شامل یک روش تائید برای بررسی خطاهای ورودی.
iii. اجبار به انتخاب کلمه عبور باکیفیت و قدرتمند.
iv. اجبار کاربران به تغییر کلمه عبور در اولین ورود به سیستم.

[1] interactive

v. اجبار به تغییر رمزعبور در فواصل زمانی مشخص و یا در مواقع موردنیاز.
vi. نگهداری سابقه کلمات عبوری که قبلاً استفاده‌شده، تا کاربر نتواند مجدداً از آن‌ها استفاده نماید.
vii. عدم نمایش رمزهای عبور بر روی صفحه نمایشگر در زمان تایپ آن‌ها.
viii. ذخیره کلمات عبور به‌صورت فایل جداگانه.
ix. ذخیره و انتقال کلمات عبور به‌صورت محافظت‌شده.

- **سایر اطلاعات**

بعضی از برنامه‌ها نیاز دارند که رمزعبور توسط سیستم دیگری اختصاص یابد. در این صورت برخی از موارد بالا برای آن سیستم صدق نخواهد کرد.

۱۰ - ماده ۶: رمزنگاری[1]

سیاست استفاده از کنترل‌های رمزنگاری

- **کنترل**

یک سیاست برای استفاده از کنترل‌های رمزنگاری جهت محافظت از اطلاعات، باید اجرا و توسعه داده شود.

- **راهنمای اجرایی**

هنگامی‌که در حال ایجاد یک سیاست رمزنگاری هستید، باید به موارد زیر توجه نمایید:

i. رویکرد مدیریت نسبت به استفاده از کنترل‌های رمزنگاری در سراسر سازمان، ازجمله اصول کلی است که بر اساس آن اطلاعات سازمان باید حفاظت شود.
ii. سطح حفاظت موردنیاز، بر اساس ارزیابی ریسک مشخص‌شده و با در نظر گرفتن نوع، قدرت و کیفیت الگوریتم‌های رمزنگاری انجام می‌پذیرد.
iii. استفاده از رمزنگاری جهت محافظت از اطلاعات منتقل‌شده توسط تجهیزات همراه، رسانه‌های قابل‌حمل و یا در خطوط ارتباطی.
iv. روش مدیریت کلید؛ ازجمله روش‌های محافظت از کلیدهای رمزنگاری و بازیابی اطلاعات رمز شده در صورت از دست دادن یا آسیب دیدن کلید.
v. نقش‌ها و مسئولیت‌ها. به‌عنوان‌مثال چه کسی مسئول اجرای سیاست و یا مدیریت کلید است.

[1] cryptography

vi. تصویب استاندارد برای اجرای مؤثر در سازمان.

vii. تأثیر استفاده از اطلاعات رمزشده بر کنترل‌هایی که توسط برنامه‌های بازرسی محتوا مثل Malware Detection[۱] انجام می‌پذیرد.

کنترل‌های رمزنگاری می‌توانند برای رسیدن به اهداف مختلفی مورداستفاده قرار گیرند:

i. محرمانگی[۲]: استفاده از رمزنگاری برای محافظت از اطلاعات حساس و حیاتی در زمان ذخیره‌سازی یا انتقال.

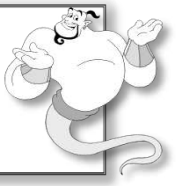
ii. درستی/اصالت[۳]: با استفاده از امضای دیجیتال و یا کد تائید هویت پیام، می‌توان از صحت اطلاعات حساس در زمان ذخیره‌سازی یا انتقال، اطمینان حاصل کرد.

iii. عدم انکار[۴]: می‌توان از تکنیک‌های رمزنگاری برای فراهم آوردن شواهد وقوع[۵] وقوع[۵] یا عدم وقوع یک رویداد یا عمل، استفاده نمود.

iv. احراز هویت: استفاده از تکنیک‌های رمزنگاری برای احراز هویت کاربران.

- **سایر اطلاعات**

تصمیم‌گیری اینکه آیا راه‌حل رمزنگاری مناسب است یا نه، باید به‌عنوان بخشی از ارزیابی ریسک و انتخاب کنترل‌ها انجام پذیرد. این ارزیابی سپس می‌تواند برای تعیین یک کنترل رمزنگاری مناسب مورداستفاده قرار گیرد.



در بخش رمزنگاری این کتاب (بخش دوم) به توضیح الگوریتم‌های رمزنگاری پرداخته‌شده و نحوه استفاده از این الگوریتم‌ها برای هر یک از اهداف: محرمانگی، صحت، امضای دیجیتال، عدم انکار و احرازهویت، تشریح گردیده است.

---

[۱] منظور از Malware Detection و برنامه‌های بازرسی محتوا، آنتی‌ویروس یا برنامه‌های امنیتی است که برای محافظت در مقابل کدهای مخرب و یا فیلترینگ مورداستفاده قرار می‌گیرند. به دلیل اینکه آنتی‌ویروس اقدام به مقایسه امضای فایل‌ها با امضاهای موجود در دیتابیس خود می‌نماید، لذا دیتای رمزشده به دلیل داشتن امضای متفاوت، ممکن است حاوی بدافزار بوده و به‌این‌ترتیب از دید آنتی‌ویروس یا دیگر برنامه‌های امنیتی پنهان بماند. همچنین در برنامه‌های بازرسی محتوا که معمولاً برای فیلترینگ کاربرد دارند، رمزگذاری باعث اختلال در عملکرد آن‌ها خواهد شد.

[2] confidentiality

[3] integrity/authenticity

[4] non-repudiation

[5] occurrence

مدیریت کلید رمزنگاری

- **کنترل**

سیاست و خط‌مشی برای استفاده، محافظت و طول عمر[1] کلیدهای رمزنگاری باید توسعه‌یافته و در سراسر چرخه حیات[2] کلیدها، اجرا گردد.

- **راهنمای اجرایی**

این خط‌مشی باید شامل الزامات مدیریت کلیدهای رمزنگاری در سراسر چرخه حیات آن‌ها باشد. چرخه حیات کلیدها شامل تولید[3]، ذخیره، آرشیو، بازیابی[4]، توزیع، بازنشستگی (کنار گذاشتن) و از بین بردن[5] کلید می‌گردد.

الگوریتم‌های رمزنگاری، طول کلید و شیوه استفاده، باید با توجه به بهترین شیوه انتخاب گردند. تمام کلیدهای رمزنگاری باید در برابر تغییر و از دست دادن محافظت شوند. علاوه بر این کلیدهای مخفی و کلیدهای خصوصی نیاز به محافظت در مقابل استفاده غیرمجاز و افشا نیز دارند.

برای تدوین سیاست‌های مربوط به مدیریت کلید، باید به مواردی مثل نحوه تولید کلید، چگونگی ذخیره‌سازی و تبادل کلید، نحوه برخورد با کلید به خطر افتاده، پشتیبان‌گیری از کلیدها، تغییر و به‌روزرسانی کلید، لغو و از بین بردن کلید، توجه گردد.

تجهیزات مورداستفاده برای تولید، ذخیره‌سازی و آرشیو کلید، باید به‌صورت فیزیکی تحت حفاظت قرار گیرند.

۱۱ - ماده ۷: امنیت محیطی و فیزیکی

امنیت فیزیکی محیط پیرامونی

- **کنترل**

برای حفاظت از مناطقی که حاوی اطلاعات حساس و حیاتی، و امکانات پردازش اطلاعات است، باید اقدام به تعریف امنیت محیط پیرامونی نمایید.

- **راهنمای اجرایی**

برای تعریف و اجرای امنیت‌فیزیکی محیط پیرامونی باید به‌رهنمودهای‌زیر توجه نمایید:

---

[1] lifetime
[2] lifecycle
[3] generating
[4] retrieving
[5] destroying

i. مکان، استحکام و قدرت هر محیط امنیتی باید بر اساس الزامات امنیتی دارایی‌ها و همچنین نتایج به‌دست‌آمده از ارزیابی ریسک، مشخص گردد.
ii. قسمتی از یک ساختمان یا سایت که حاوی امکانات پردازش اطلاعات است، باید ازنظر فیزیکی سالم و استوار باشد. سقف بیرونی، دیوارها و کف‌سازی باید کاملاً مستحکم ساخته‌شده باشند. تمام درهای بیرونی با استفاده از مکانیسم‌های کنترلی مثل آلارم و قفل، باید از دسترسی افراد غیرمجاز جلوگیری به عمل‌آورند. درب‌ها و پنجره‌ها باید قفل شوند. بهتر است پنجره‌ها از بیرون دارای حفاظ باشند.
iii. در ساختمان و یا سایت باید یک اتاق یا منطقه برای دسترسی فیزیکی افراد مجاز در نظر گرفته شود.
iv. استفاده از موانع فیزیکی در صورت امکان، برای جلوگیری از دسترسی غیرمجاز.
v. درب‌های ضد حریق محیط امنیتی باید دارای آلارم بوده و همچنین کنترل و آزمایش شوند. این درب‌ها باید به‌صورت امن و بر اساس استانداردهای محلی عمل نمایند.
vi. باید سیستم‌های تشخیص مزاحم، برای پوشش دادن تمام درب‌های خارجی و پنجره‌های قابل‌دسترس، مطابق با استانداردهای ملی، منطقه‌ای یا بین‌المللی نصب و همواره مورد آزمایش قرار گیرند. مناطق خالی و همچنین مناطق دیگری مثل اتاق کامپیوتر نیز باید توسط این سیستم پوشش داده شود.
vii. تجهیزات و امکانات پردازش اطلاعات که توسط سازمان مدیریت می‌شوند باید ازنظر فیزیکی به‌صورت جداگانه از تجهیزاتی نگهداری شوند که توسط اشخاص ثالث مدیریت می‌گردند.

کنترل ورود فیزیکی

- **کنترل**

برای حصول اطمینان از اینکه فقط افراد مجاز امکان دسترسی دارند، باید مناطق امنیتی توسط کنترل‌های ورودی مناسب محافظت شوند.

- **راهنمای اجرایی**

برای کنترل ورود فیزیکی به محیط امنیتی، موارد زیر باید رعایت شود:

i. تاریخ و ساعت ورود و خروج بازدیدکنندگان باید ثبت‌شده و هویت بازدیدکنندگان توسط ابزاری مناسب تصدیق شود. بازدیدکنندگان باید مشایعت یا نظارت شوند مگر آنکه قبلاً تائید شده باشند.

ii. دسترسی به مناطقی که حاوی اطلاعات محرمانه پردازش‌شده یا ذخیره‌شده می‌باشند، باید به افراد مجاز و تنها با اجرای کنترل دسترسی مناسب، محدود شود. به‌عنوان‌مثال استفاده از مکانیسم احراز هویت دو عامله[1] که هم از کارت دسترسی و هم از کد PIN، استفاده می‌نماید.

iii. وقایع دسترسی باید در قالب مستندات فیزیکی یا الکترونیکی نگهداری شود.

iv. تمام کارکنان، پیمانکاران و طرف‌های خارجی بازدیدکننده، باید ملزم به استفاده از نوعی پوشش قابل‌مشاهده باشند. به‌محض رؤیت بازدیدکننده‌ای که پوشش فوق را ندارد، باید سریعاً مأموران امنیتی سازمان را مطلع نمود.

v. دسترسی طرف‌های خارجی به مناطق امنیتی و امکانات پردازش اطلاعات باید محدودشده و فقط در موارد خاص، آن‌هم با نظارت انجام پذیرد.

vi. حقوق دسترسی به مناطق امنیتی باید همواره موردبررسی و به‌روزرسانی قرارگرفته تا در صورت لزوم لغو شوند.

حفاظت در مقابل تهدیدات بیرونی و زیست‌محیطی

- **کنترل**

حفاظت فیزیکی در برابر بلایای طبیعی، حمله‌های مخرب و حوادث، باید طراحی گردد.

- **راهنمای اجرایی**

باید مشاوره تخصصی در مورد چگونگی جلوگیری از آسیب‌های ناشی از سیل، زلزله، آتش‌سوزی، دیگر بلایای طبیعی، ناآرامی‌های مدنی و یا فجایع انسانی انجام پذیرد.

امنیت کابل‌کشی

- **کنترل**

کابل‌های برق و کابل‌های مخابراتی که جهت حمل دیتا یا سرویس‌های اطلاعات می‌باشند، باید در مقابل قطعی، تداخل یا آسیب‌دیدگی محافظت شوند.

- **راهنمای اجرایی**

[1] two-factor authentication mechanism

موارد زیر را جهت امنیت کابل‌کشی در نظر بگیرید:

i. مسیر کابل‌های برق و مخابراتی در صورت امکان باید در زیرِ زمین تعبیه گردیده، یا با جایگزین مناسبی تحت حفاظت قرار گیرند.
ii. برای جلوگیری از تداخل، باید کابل‌های برق و کابل‌های مخابراتی به‌صورت جداگانه از یکدیگر قرار گیرند.

برای سیستم‌های حساس و حیاتی، علاوه بر موارد فوق، باید به این موارد نیز توجه داشته باشید:

i. نصب لوله‌های آهنی و اتاق‌های قفل‌شده یا محفظه، در نقاط بازرسی و پایانی.
ii. استفاده از سپر الکترومغناطیس برای محافظت از کابل‌ها.
iii. بازرسی فنی و فیزیکی برای کشف تجهیزات غیرمجاز متصل شده بر روی کابل‌ها.
iv. کنترل دسترسی به Patch Panel و اتاق کابل.

۱۲ – ماده ۸: امنیت عملیات

مستندسازی دستورالعمل عملیاتی

- **کنترل**

دستورالعمل‌های عملیاتی باید مستند شده و در اختیار همه کاربرانی که به آن نیاز دارند، قرار گیرد.

- **راهنمای اجرایی**

مستندسازی برای دستورالعمل عملیاتی باید با توجه به موارد زیر صورت پذیرد:

i. نصب و پیکربندی[1] سیستم‌ها.
ii. پردازش و به‌کارگیری اطلاعات به هر دو روش دستی و خودکار.
iii. پشتیبان‌گیری.
iv. الزامات برنامه‌ریزی، شامل وابستگی به سیستم‌های دیگر، زمان شروع اولین کار[2] و زمان انجام آخرین کار.

[1] Configuration
[2] Job

v. دستورالعمل اداره اشتباهات و دیگر شرایط استثنایی که ممکن است در حین اجرای کار رخ دهد.
vi. پشتیبانی و تماس در صورت بروز مشکلات عملیاتی و فنی غیرمنتظره.
vii. دستورالعمل اداره برای خروجی خاص و رسانه؛ مانند مدیریت خروجی‌های محرمانه مثل خروجی مربوط به یک کار شکست‌خورده.
viii. دستورالعمل بازیابی و راه‌اندازی مجدد سیستم در صورت بروز خرابی[1].
ix. مدیریت پیگیری و اطلاعات وقایع‌نگاری سیستم.
x. دستورالعمل مانیتورینگ.

مدیریت تغییر

- **کنترل**

تغییرات در سازمان، فرآیند کسب‌وکار، امکانات پردازش اطلاعات و سیستم‌هایی که بر روی امنیت اطلاعات تأثیر می‌گذارند، باید کنترل شوند.

- **راهنمای اجرایی**

به‌طور خاص موارد زیر باید در نظر گرفته شود:

i. شناسایی و ثبت تغییرات قابل‌توجه.
ii. برنامه‌ریزی و آزمایش تغییرات.
iii. ارزیابی اثرات بالقوه تغییرات.
iv. دستورالعمل مصوب رسمی برای تغییرات پیشنهادشده.
v. بررسی برآورده شدن الزامات امنیت اطلاعات.
vi. ابلاغ جزئیات تغییر به‌تمامی افراد مربوطه.
vii. دستورالعمل اصلاح‌حالت[2](برگشت از خرابی). شامل مسئولیت‌ها و روش‌هایی برای صرف‌نظر و بازیابی از تغییرات ناموفق و حوادث پیش‌بینی‌نشده.
viii. تهیه فرآیند تغییر در شرایط اضطراری؛ برای اجرای سریع و کنترل‌شده تغییراتی که برای حل یک حادثه موردنیاز است.

- **سایر اطلاعات**

کنترل ناکافی تغییرات سیستم‌ها و امکانات پردازش اطلاعات، علت شایع خرابی سیستم‌ها و امنیت است. تغییرات در محیط‌های عملیاتی، خصوصاً زمان انتقال یک

[1] failure
[2] fall-back

سیستم از وضعیت توسعه به مرحله عملیاتی، می‌تواند قابلیت اطمینان سیستم را تحت تأثیر قرار دهد.

کنترل در برابر بدافزارها

- **کنترل**

کنترل‌های تشخیص، پیشگیری و بازیابی برای محافظت در مقابل بدافزارها و نرم‌افزارهای مخرب، باید اجرا شود. البته این کنترل‌ها باید با هوشیاری کاربر همراه باشد.

- **راهنمای اجرایی**

محافظت در مقابل بدافزارها باید بر اساس نرم‌افزار تشخیص و ترمیم آن‌ها، آگاهی امنیت اطلاعات و دسترسی به سیستم مناسب و کنترل‌های مدیریت تغییرات باشد.

برای کنترل در برابر نرم‌افزارهای مخرب موارد زیر را در نظر بگیرید:

i. ایجاد یک خط‌مشی رسمی جهت منع استفاده از نرم‌افزارهای غیرمجاز.
ii. پیاده‌سازی کنترل‌هایی برای تشخیص یا جلوگیری استفاده از نرم‌افزارهای غیرمجاز.
iii. پیاده‌سازی کنترل‌هایی جهت تشخیص یا جلوگیری استفاده از وب‌سایت‌های مخرب یا مشکوک.
iv. ایجاد یک خط‌مشی رسمی برای محافظت در برابر ریسک‌های مرتبط با دریافت فایل‌ها و نرم‌افزارها از طریق شبکه‌های خارجی یا هر رسانه دیگر.
v. کاهش آسیب‌پذیری‌هایی که می‌تواند توسط بدافزارها مورد سوءاستفاده قرار گیرد. به‌عنوان‌مثال از استفاده از مدیریت آسیب‌پذیری[1].
vi. انجام بررسی‌های منظم از نرم‌افزارها و دیتای حیاتی سازمان؛ تا در صورت وجود هرگونه فایل تائید نشده، سریعاً موردبررسی رسمی قرار گیرد.
vii. به‌عنوان یک کنترل پیشگیرانه باید اقدام به نصب، راه‌اندازی و بروز رسانی منظم نرم‌افزار تشخیص و ترمیم بدافزار نموده و از آن برای اسکن کامپیوترها و رسانه‌ها استفاده نمایید. اسکن می‌تواند شامل: اسکن قبل از استفاده از فایل‌ها، اسکن فایل‌های پیوست ایمیل و اسکن صفحات وب، باشد.
viii. تعریف دستورالعمل و مسئولیت‌ها جهت مقابله با بدافزارها؛ آموزش استفاده از دستورالعمل‌ها، گزارش‌گیری و ترمیم حملات بدافزارها.

[1] Vulnerability Management

ix. تهیه طرح مناسب برای تداوم کسب‌وکار در دوره بازیابی از حملات مخرب. ازجمله تمام اطلاعات لازم و نرم‌افزار پشتیبان‌گیری و بازیابی تنظیمات.
x. اجرای روشی به‌منظور جمع‌آوری منظم اطلاعات. ازجمله عضویت در لیست‌های پستی یا وب‌سایت‌هایی که اقدام به ارائه اطلاعات درباره نرم‌افزارهای مخرب جدید می‌نمایند.
xi. اجرای روشی به‌منظور بررسی اطلاعات مربوط به بدافزارها و اطمینان از اینکه آگهی‌های هشداردهنده دقیق و آموزنده هستند.
xii. جداسازی محیط‌هایی که ممکن است در آن‌ها اثرات فاجعه‌بار رخ دهد.

- **سایر اطلاعات**

استفاده از دو یا چند محصول مقابله با بدافزار با برندهای مختلف در محیط پردازش اطلاعات، می‌تواند باعث بهبود عملکرد مقابله با بدافزارها گردد.
معمولاً استفاده از نرم‌افزار تشخیص بدافزار به‌تنهایی نمی‌تواند کافی باشد، بلکه باید در کنار آن از روش‌هایی استفاده شود که از ورود بدافزار نیز جلوگیری نماید.

پشتیبان‌گیری[1]

- **کنترل**

مطابق با سیاست‌های پشتیبان‌گیری سازمان، باید به‌طور منظم یک نسخه پشتیبان از اطلاعات، نرم‌افزارها و وضعیت سیستم تهیه و آزمایش شود.

- **راهنمای اجرایی**

یک سیاست پشتیبان‌گیری از اطلاعات، نرم‌افزارها و سیستم‌ها بر اساس نیازهای سازمان باید تعریف شود. این سیاست باید شامل الزامات نگهداری و محافظت باشد.
وجود نسخه پشتیبان از اطلاعات، اطمینان می‌دهد که در صورت وقوع یک فاجعه یا خرابی رسانه، اطلاعات ضروری سازمان قابل بازیابی خواهند بود.
هنگام برنامه‌ریزی برای یک طرح پشتیبان‌گیری، موارد زیر باید در نظر گرفته شود:

i. تهیه سوابق دقیق و کامل از نسخه‌های پشتیبان و دستورالعمل بازیابی.
ii. مشخص کردن نوع (کامل[2]، تفاضلی[3] و افزایشی[4]) و تناوب پشتیبان‌گیری، بر اساس نیازهای کسب‌وکار سازمان و الزامات امنیتی.

[1] Backup
[2] Full
[3] Differential
[4] Incremental

iii. برای فرار از آسیب‌های مربوط به بروز فاجعه در محل اصلی، نسخه پشتیبان باید در یک مکان راه دور ذخیره شود.
iv. اطلاعات پشتیبان‌گیری باید در سطح مناسبی از حفاظت فیزیکی و محیطی، مطابق با استاندارد اعمال‌شده در سایت اصلی، قرار بگیرند.
v. رسانه‌های مورداستفاده جهت پشتیبان‌گیری، باید به‌صورت منظم مورد آزمایش قرارگرفته تا اطمینان حاصل شود که در موقع ضرورت می‌توان به آن‌ها تکیه کرد.
vi. در شرایطی که محرمانگی مهم است، اطلاعات پشتیبان گرفته‌شده باید توسط ابزارهای رمزنگاری محافظت شوند.



صرف گرفتن پشتیبان از اطلاعات، نمی‌تواند تضمین‌کننده بازیابی آن‌ها در زمان ضرورت باشد. بلکه باید به‌صورت منظم اقدام به بازیابی آزمایشی پشتیبان‌های گرفته‌شده نمود، تا هم از درستی عملیات پشتیبان‌گیری اطمینان حاصل شود و هم از درستی رسانه‌های مورداستفاده.

ثبت رویداد[1]

- **کنترل**

گزارش‌های[2] رویداد، اقدام به ثبت فعالیت‌های کاربر، استثنائات، نقص‌ها و رویدادهای امنیت اطلاعات می‌نماید. این گزارش‌ها باید تولید و نگهداری شده و به‌طور منظم موردبررسی قرار گیرند.

- **راهنمای اجرایی**

گزارش‌های رویداد، باید شامل موارد زیر باشد:

i. شناسه کاربر.
ii. فعالیت‌های سیستم.
iii. تاریخ، زمان و جزئیات مربوط به وقایع کلیدی مثل ورود و خروج.
iv. در صورت امکان، هویت، محل دستگاه و شناسه سیستم.
v. سوابق اقدامات موفق و ناموفق دسترسی به سیستم.
vi. سوابق اقدامات موفق و ناموفق دسترسی به دیتا و دیگر منابع.
vii. تغییرات اعمال‌شده در پیکربندی سیستم.

[1] Event logging
[2] Logs

viii. استفاده از حقوق دسترسی ویژه.
ix. استفاده از ابزارها و برنامه‌های سیستم.
x. فایل‌هایی که به آن‌ها دسترسی پیداکرده و همچنین نوع دسترسی.
xi. پروتکل‌ها و آدرس‌های شبکه.
xii. هشدارهای تولیدشده توسط سیستم کنترل دسترسی.
xiii. فعال‌سازی و غیرفعال‌سازی سیستم‌های حفاظتی ازجمله آنتی‌ویروس و سیستم‌های تشخیص نفوذ[1].
xiv. ثبت تراکنش‌هایی که کاربر در برنامه‌های کاربردی اجرا کرده است.

- **سایر اطلاعات**

با توجه به اینکه گزارش‌های رویداد، ممکن است شامل داده‌های حساس و اطلاعات قابل‌شناسایی شخصی باشد، لذا باید برای آن‌ها اقدامات مناسب امنیتی صورت پذیرد. در صورت امکان، مدیران سیستم نباید اجازه پاک کردن یا غیرفعال کردن گزارش‌های مربوط به فعالیت‌های خود را داشته باشند.

نصب نرم‌افزار بر روی سیستم‌عامل

- **کنترل**

باید دستورالعملی جهت کنترل نصب نرم‌افزار بر روی سیستم‌عامل، اجرا گردد.

- **راهنمای اجرایی**

برای کنترل تغییرات نرم‌افزاری بر روی سیستم‌عامل‌ها، باید موارد زیر در نظر گرفته شود:

i. بروز رسانی سیستم‌عامل و نرم‌افزارها باید توسط ادمین‌های آموزش‌دیده و بر اساس مجوز مدیریت انجام پذیرد.
ii. سیستم‌عامل‌ها تنها باید به‌صورت کدهای اجرایی نگهداری شوند؛ و نه به‌صورت کدهای کامپایل شده یا کدهای توسعه‌یافته.
iii. سیستم‌عامل و برنامه‌های کاربردی باید پس از آزمون‌های موفقیت‌آمیز مورداستفاده و اجرا قرار گیرند. این آزمون‌ها باید شامل قابلیت استفاده، امنیت، اثر بر روی سیستم‌های دیگر و کاربرپسندی باشد. همچنین توجه داشته باشید که این آزمون‌ها باید بر روی سیستم‌های جداگانه انجام پذیرد.

[1] Intrusion Detection Systems (IDS)

iv. برای نگهداری کنترل تمام نرم‌افزارهای اجراشده و همچنین مستندات سیستم، باید از یک سیستم کنترل پیکربندی، استفاده گردد.
v. قبل از اجرای تغییرات، باید یک راهکار عقب‌گرد[1] مشخص شده باشد.
vi. گزارش‌های بازرسی باید تمام به‌روزرسانی‌ها را نگهداری نمایند.
vii. نسخه‌های قبلی نرم‌افزارها به‌عنوان یک اقدام احتمالی باید حفظ گردند.
viii. نسخه‌های قدیمی نرم‌افزار به همراه اطلاعات، پارامترها، دستورالعمل‌ها، جزئیات پیکربندی و پشتیبانی از نرم‌افزار باید آرشیو شوند. این اطلاعات تا زمانی که دیتا در آرشیو نگهداری می‌شود، باید حفظ شوند.

ازآنجاکه فروشندگان نرم‌افزار پس از مدت‌زمان مشخصی دیگر از نسخه‌های قدیمی پشتیبانی نمی‌کنند، لذا باید سازمان ریسک ناشی از تکیه‌بر این نرم‌افزارها را در نظر بگیرد.
وصله‌های[2] نرم‌افزار که باعث رفع یا کاهش نقاط ضعف امنیتی می‌گردد، باید مورداستفاده قرار گیرند.
در صورت لزوم اجازه دسترسی فیزیکی یا منطقی به تأمین‌کنندگان برای مقاصد پشتیبانی، باید با دستور مدیریت صورت گرفته و بر کار آن‌ها نیز نظارت نمود.

محدودیت در نصب نرم‌افزار

- **کنترل**

باید قوانینی جهت نصب نرم‌افزار توسط کاربران، ایجاد و پیاده‌سازی شود.

- **راهنمای اجرایی**

سازمان باید یک سیاست سخت‌گیرانه برای نصب نرم‌افزار توسط کاربران، ایجاد و اعمال نماید. باید اصل "حداقل امتیاز" پیاده‌سازی شود. اگر به کاربران امتیازات خاص داده شود، ممکن است توانایی نصب نرم‌افزار را داشته باشد.

- **سایر اطلاعات**

نصب کنترل نشده نرم‌افزار، می‌تواند باعث ایجاد آسیب‌پذیری و درنهایت نشت[3] اطلاعات یا حوادث امنیتی دیگر گردد.

---
[1] Rollback
[2] Patches
[3] Leakage

۱۳ – ماده ۹: امنیت ارتباطات

کنترل شبکه

- **کنترل**

شبکه باید در راستای محافظت از اطلاعات سیستم‌ها و برنامه‌ها، مدیریت و کنترل گردد.

- **راهنمای اجرایی**

برای اطمینان از امنیت اطلاعات در شبکه و حفاظت سرویس‌ها از دسترسی غیرمجاز، کنترل‌های مربوطه باید اجرا گردند. این کنترل‌ها باید موارد زیر را در نظر بگیرند:

i. مسئولیت‌ها و دستورالعمل‌هایی برای مدیریت تجهیزات شبکه، تعریف شود.

ii. مسئولیت عملیاتی شبکه، باید از عملیات کامپیوتر، جدا باشد.

iii. باید کنترل‌های ویژه‌ای به‌منظور حفظ محرمانگی و صحت داده‌های تبادل شده بر روی شبکه‌های عمومی یا شبکه‌های بی‌سیم و محافظت از سیستم‌ها و برنامه‌های متصل شده، برقرار گردد.
ممکن است کنترل‌های ویژه‌ای نیز برای حفظ در دسترس بودن سرویس‌های شبکه و کامپیوترهای متصل شده، لازم باشد.

iv. با اعمال مانیتورینگ و ثبت‌رویداد مناسب، قادر به ثبت و تشخیص اقداماتی خواهید بود که مرتبط با امنیت اطلاعات بوده و یا ممکن است آن را تحت تأثیر قرار دهند.

v. فعالیت‌های مدیریتی باید به بهینه‌سازی خدمات به سازمان پرداخته و از اینکه کنترل‌ها به‌طور مداوم در سراسر زیرساخت پردازش اطلاعات، استفاده می‌شوند، اطمینان حاصل نماید.

vi. سیستم‌ها در شبکه باید احراز هویت شوند.

vii. سیستم‌های متصل به شبکه باید محدود شوند.

- **سایر اطلاعات**

برای اطلاعات بیشتر درباره امنیت شبکه می‌توانید به استاندارد ISO/IEC 27033 مراجعه نمایید.[۱]

---

[۱] استاندارد ISO/IEC 27033، یک استاندارد بین‌المللی جهت امنیت شبکه و دارای شش بخش مختلف است؛ تاکنون ۵ بخش آن به‌صورت رسمی منتشرشده و یک بخش آن نیز در مرحله پیش‌نویس است.

سیاست‌ها و دستورالعمل انتقال اطلاعات

- **کنترل**

کنترل‌ها، دستورالعمل‌ها و سیاست‌های انتقال، برای حفاظت انتقال اطلاعات از طریق انواع امکانات ارتباطی باید ایجاد گردد.

- **راهنمای اجرایی**

زمانی که از امکانات ارتباطی برای انتقال اطلاعات استفاده می‌شود، روش‌ها و کنترل‌های زیر باید دنبال شود:

i. دستورالعمل طراحی‌شده، باید از انتقال اطلاعات در برابر قطع، کپی، تغییر، مسیریابی اشتباه و تخریب، محافظت نماید.

ii. در نظر گرفتن روشی برای تشخیص و محافظت در برابر بدافزارهایی که ممکن است از طریق ارتباط الکترونیکی منتقل شوند.

iii. روش محافظت از اطلاعات الکترونیکی حساس که به‌صورت یک پیوست، دریافت می‌شوند.

iv. خط‌مشی یا دستورالعمل جهت استفاده درست از امکانات ارتباطی.

v. عملکرد کارکنان، طرف‌های خارجی و کاربران دیگر، نباید سازمان را در معرض خطر قرار دهد. ازجمله این عملکردها می‌توان از افترا، تعرض، جعل هویت، ارسال نامه‌های زنجیره‌ای، خرید غیرمجاز و غیره، نام برد.

vi. استفاده از تکنیک‌های رمزنگاری برای محرمانگی، درستی و صحت اطلاعات.

vii. نگهداری و در اختیار گذاشتن راهنما بر اساس قوانین مربوطه محلی و ملی، برای تمام مکاتبات سازمان ازجمله پیام‌ها.

viii. کنترل‌ها و محدودیت‌های مرتبط با استفاده از امکانات ارتباطی.

ix. آموزش کارکنان برای انجام اقدامات احتیاطی مناسب در جهت جلوگیری از افشای اطلاعات محرمانه.

x. پیام‌های حاوی اطلاعات محرمانه را بر روی دستگاه‌های پیام‌گیر[1] رها نکنید. تا در دسترس افراد غیرمجاز قرار نگیرد.

xi. آموزش کارکنان درباره مشکلات استفاده از سرویس‌ها یا دستگاه‌های فکس.

xii. کارکنان نباید مکالمات محرمانه را در مکان‌های عمومی یا از طریق کانال‌های ناامن انجام دهند.

- **سایر اطلاعات**

[1] Answering Machine

انتقال اطلاعات ممکن است از طریق انواع مختلفی از امکانات ارتباطی ازجمله پست الکترونیک، صدا، فکس و ویدئو، انجام پذیرد. انتقال نرم‌افزارها هم ممکن است از طریق انواع مختلفی از رسانه‌ها ازجمله دانلود از اینترنت و یا خرید آن‌ها، انجام پذیرد.
مفاهیم کسب‌وکار، حقوقی و امنیتی مرتبط با تبادل الکترونیکی داده‌ها، تجارت الکترونیک و ارتباطات الکترونیک و الزامات کنترلی، باید در نظر گرفته شود.

۱۴ – ماده ۱۰: کسب، توسعه و نگهداری سیستم

محافظت از نرم‌افزار خدمات معاملات[1]

- **کنترل**

اطلاعات مربوط به نرم‌افزار خدمات معاملات باید در مقابل انتقال ناقص، مسیریابی اشتباه، تغییر غیرمجاز پیام، افشا شدن، نسخه‌برداری غیرمجاز و بازپخش[2] محافظت شود.

- **راهنمای اجرایی**

ملاحظات راهنمای پیاده‌سازی محافظت از نرم‌افزار خدمات معاملات باید شامل موارد زیر باشد:

i. استفاده از امضای الکترونیک، توسط هریک از طرفین معامله.
ii. اطلاعات احراز هویت طرفین، معتبر و تائید شده باشد.
iii. معامله محرمانه باقی‌مانده و حریم خصوصی طرفین حفظ گردد.
iv. مسیرهای ارتباطی بین طرفین معامله، رمزگذاری شده باشد.
v. پروتکل مورداستفاده برای برقراری ارتباط طرفین، امن باشد.
vi. جزئیات معامله نباید در محیط عمومی و قابل‌دسترس، ذخیره شود.
vii. باید از مقام مورد اعتماد[3] استفاده کرد. مقام مورد اعتماد برای صدور و حفظ امضای دیجیتال و صدور گواهینامه دیجیتال کاربرد دارد.

- **سایر اطلاعات**

وسعت کنترل‌ها، متناسب با سطح ارزیابی ریسک در ارتباط با هر یک از اشکال نرم‌افزار خدمات معاملات، خواهد بود.

---

[1] Protecting application services transactions
[2] Replay
[3] Trusted authority

سیاست توسعه امن نرم‌افزار

- **کنترل**

قوانینی برای توسعه امن نرم‌افزار و سیستم‌ها باید ایجاد نموده و درون سازمان مورداستفاده قرار داد.

- **راهنمای اجرایی**

برای ایجاد یک سرویس امن، معماری امن، نرم‌افزار امن و سیستم امن، نیاز به یک توسعه[1] امن، است.

برای سیاست توسعه امن، موارد زیر باید در نظر گرفته شود:

I. امنیت محیط توسعه.
II. دستورالعمل امنیت در چرخه حیات توسعه نرم‌افزار شامل: امنیت درروش توسعه نرم‌افزار و ایمن‌سازی برنامه‌نویسی بر اساس زبان مورداستفاده.
III. الزامات امنیتی در مرحله طراحی.
IV. بازرسی امنیتی در نقاط عطف پروژه.
V. مخازن[2] امن.
VI. امنیت در کنترل نسخه[3].
VII. نیاز به دانش امنیت نرم‌افزار.
VIII. قابلیت توسعه‌دهندگان در اجتناب، پیدا کردن و رفع آسیب‌پذیری.

تکنیک‌های برنامه‌نویسی امن، هم باید برای توسعه‌های جدید و هم برای استفاده مجدد از کدهایی که استاندارد به آن‌ها اعمال نشده، مورداستفاده قرار گیرند.
استاندارد برنامه‌نویسی امن باید اجبار گردد. توسعه‌دهندگان نیز باید برای استفاده از آن آموزش‌دیده و رویه آزمایش و بررسی کد را مورداستفاده قرار دهند.
اگر توسعه نرم‌افزار برون‌سپاری گردیده است، سازمان باید از توسعه نرم‌افزار بر اساس قوانین توسعه امن، اطمینان حاصل نماید.

- **سایر اطلاعات**

همچنین ممکن است توسعه در داخل برنامه‌های کاربردی صورت گیرد. مثلاً نرم‌افزارهای اداری، اسکریپت نویسی، مرورگرها و دیتابیس.

---

[1] Development
[2] Repositories
[3] Version

دستورالعمل کنترل تغییرات سیستم

- **کنترل**

اعمال تغییرات در سیستم، در طول چرخه‌حیات توسعه باید با استفاده از روش‌های رسمی کنترل تغییرات، کنترل شوند.

- **راهنمای اجرایی**

دستورالعمل‌های رسمی کنترل تغییرات باید مستندسازی و اجبار گردند؛ تا از یکپارچگی سیستم، نرم‌افزارها و محصولات، از مراحل اولیه طراحی تا تمام تلاش‌های نگهداری پس‌ازآن، اطمینان حاصل شود.

معرفی سیستم‌های جدید و تغییرات بزرگ در سیستم‌های موجود، باید بر اساس یک فرآیند رسمی از مستندات، مشخصات، آزمایش، کنترل کیفیت و مدیریت پیاده‌سازی، دنبال گردد. این فرآیند باید شامل ارزیابی ریسک، تجزیه‌وتحلیل تأثیر تغییرات و مشخصات کنترل‌های امنیتی موردنیاز، باشد. این فرآیند باید تضمین کند که دستورالعمل امنیتی و کنترلی موجود به خطر نخواهد افتاد؛ لذا باید دسترسی برنامه‌نویسان، محدود به قسمت‌هایی باشد که مربوط به کار آن‌ها است.

روش‌های کنترل تغییرات باید شامل (و نه محدود به) موارد زیر باشد:

i. حفظ سابقه‌ای از سطح مجوز موافقت شده.
ii. حصول اطمینان از اینکه تغییرات توسط کاربران مجاز انجام‌شده.
iii. بررسی دستورالعمل‌های کنترل و صحت؛ برای حصول اطمینان از اینکه آن‌ها به‌واسطه تغییرات، به خطر نخواهند افتاد.
iv. شناسایی تمام نرم‌افزارها، اطلاعات، دیتابیس‌ها و سخت‌افزارها که نیاز به اصلاح دارند.
v. شناسایی و بررسی کدهای امنیتی حیاتی، برای به حداقل رساندن احتمال ضعف‌های امنیتی شناخته‌شده.
vi. دریافت تائید رسمی برای طرح پیشنهادی، قبل از شروع کار.
vii. حصول اطمینان از اینکه کاربران مجاز، تغییرات را قبل از پیاده‌سازی مورد موافقت قرار داده‌اند.
viii. حصول اطمینان از به‌روزرسانی مستندات سیستم پس از اتمام هریک از تغییرات. همچنین مستندات قدیمی باید آرشیو یا دفع شوند.
ix. نگهداری یک کنترل نسخه، برای تمام به‌روزرسانی‌های نرم‌افزار.
x. نگهداری سوابق ممیزی از تمام درخواست‌های تغییر.

xi. در صورت نیاز، ایجاد تغییر در مستندات عملیاتی و دستورالعمل‌های کاربران، برای حفظ عملکرد مناسب آن‌ها.

xii. حصول اطمینان از اجرای تغییرات در زمان مناسب؛ تا باعث اختلال در فرآیندهای کسب‌وکار نگردد.

- **سایر اطلاعات**

تغییر نرم‌افزار می‌تواند محیط عملیاتی را تحت تأثیر قرار دهد.

عمل مناسب، شامل تست نرم‌افزار در محیطی جدا از محیط تولید و محیط توسعه است؛ این به معنی داشتن کنترل بر نرم‌افزارهای جدید و محافظت بیشتر از اطلاعات موردنیاز در این آزمایش است. این روند باید برای وصله‌ها، Service Packها و سایر به‌روزرسانی‌ها نیز مورداستفاده قرار گیرد.

توجه داشته باشید که بعضی از به‌روزرسانی‌ها ممکن است باعث خرابی سیستم شوند، لذا در سیستم‌های حیاتی نباید به‌روزرسانی خودکار مورداستفاده قرار گیرد.

۱۵ – ماده ۱۱: ارتباط با تأمین‌کنندگان[1]

سیاست‌های امنیت اطلاعات برای ارتباط با تأمین‌کنندگان

- **کنترل**

الزامات امنیت اطلاعات برای کاهش خطرات مرتبط با دسترسی تأمین‌کنندگان به دارایی‌های سازمان، باید موردتوافق با تأمین‌کننده قرارگرفته و مستند شده باشد.

- **راهنمای اجرایی**

برای مشخص نمودن دسترسی تأمین‌کنندگان، سازمان باید در قالب سیاست‌های خود اقدام به شناسایی و اجرای کنترل‌های امنیتی نماید. این کنترل‌ها باید فرآیندها و روش‌هایی را با توجه به نکات زیر، برای اجرا توسط سازمان مشخص نمایند:

i. شناسایی و مستندسازی انواع تأمین‌کنندگان. مثل خدمات IT، تدارکات ابزارها، خدمات مالی، اجزای زیرساخت IT و کسانی که سازمان باید به آن‌ها اجازه دسترسی به اطلاعات را بدهد.

ii. استانداردسازی فرآیند و چرخه‌حیات برای مدیریت ارتباط با تأمین‌کنندگان.

iii. تعریف انواع دسترسی به اطلاعات برای انواع مختلف تأمین‌کنندگان؛ و کنترل و نظارت بر دسترسی‌ها.

[1] Supplier

iv. حداقل الزامات امنیت اطلاعات برای هر نوع از اطلاعات و هر نوع دسترسی، بر اساس نیازهای کسب‌وکار سازمان و ریسک‌های آن، جهت ارائه مبنایی برای توافق با تأمین‌کنندگان مختلف، مورد استفاده قرار گیرد.
v. فرآیندها و روش‌هایی برای نظارت بر پایبندی به الزامات امنیتی برقرارشده، جهت هر نوع تأمین‌کننده و هر نوع دسترسی، وجود داشته باشد. ازجمله اعتبار سنجی محصول.
vi. دقت و کنترل کامل برای اطمینان از یکپارچگی اطلاعات یا پردازش اطلاعاتی فراهم‌شده توسط هر دو طرف.
vii. انواع تعهدات قابل‌اجرا برای تأمین‌کنندگان، جهت محافظت از اطلاعات سازمان.
viii. ازجمله مسئولیت‌های سازمان و تأمین‌کننده، اداره حوادث و احتمالات مرتبط با دسترسی تأمین‌کنندگان است.
ix. برگشت‌پذیری[1] و در صورت لزوم بازیابی، برای حصول اطمینان از در دسترس بودن اطلاعات یا فرآیندهای اطلاعاتی فراهم‌شده توسط هر دو طرف.
x. آموزش پرسنل سازمان که درگیر جمع‌آوری سیاست‌ها، فرآیندها و دستورالعمل‌ها می‌باشند.
xi. آموزش پرسنل سازمان که در تعامل با پرسنل تأمین‌کنندگان هستند. این آموزش باید با توجه به قوانین تعامل و رفتار، بر اساس نوع تأمین‌کننده و سطح دسترسی آن‌ها به سیستم و اطلاعات سازمان، انجام پذیرد.
xii. ایجاد شرایطی که در آن، کنترل‌ها و الزامات امنیت اطلاعات، تحت مستنداتی مورد موافقت و امضای طرفین قرارگرفته باشند.
xiii. مدیریت انتقال اطلاعات، امکانات پردازش و هر چیز دیگری که نیاز به جابجایی دارد؛ و حصول اطمینان از اینکه امنیت اطلاعات در طول دوره انتقال حفظ خواهد شد.

- **سایر اطلاعات**

مدیریت نامناسب امنیت اطلاعات، می‌تواند اطلاعات را توسط تأمین‌کنندگان در معرض خطر قرار دهد. کنترل‌ها باید شناسایی و در مدیریت دسترسی تأمین‌کنندگان به امکانات پردازش اطلاعات، اعمال شوند.

[1] Resilience

برای مثال اگر نیاز ویژه‌ای به محرمانگی اطلاعات دارید، می‌توانید از توافقنامه عدم افشا استفاده نمایید.
سازمان باید آگاه باشد که مسئولیت قراردادی یا حقوقی حفاظت از اطلاعات، بر عهده سازمان باقی می‌ماند.

۱۶ – ماده ۱۲: مدیریت حوادث امنیت اطلاعات[1]

مسئولیت‌ها و دستورالعمل‌ها

- **کنترل**

مسئولیت‌های مدیریت و دستورالعمل‌ها جهت تضمین واکنش[2] سریع، مؤثر و منظم برای حوادث امنیت اطلاعات، باید برقرار گردد.

- **راهنمای اجرایی**

راهنمایی‌های زیر باید برای تبیین مسئولیت‌های مدیریت و دستورالعمل‌ها، با توجه به مدیریت حوادث امنیت اطلاعات، در نظر گرفته شود:

الف) مسئولیت مدیریت برای تضمین اینکه رویه‌های زیر اجرا و به‌اندازه کافی درون سازمان ابلاغ گردیده، باید اجرا گردد:

i. برنامه‌ریزی و آماده‌سازی دستورالعمل واکنش به حوادث.
ii. دستورالعملی جهت نظارت، شناسایی، تجزیه‌وتحلیل و گزارش‌دهی حوادث و رویدادهای امنیت اطلاعات.
iii. دستورالعملی جهت ثبت فعالیت‌های مربوط به مدیریت حوادث.
iv. دستورالعمل بررسی مدارک قانونی.
v. روش ارزیابی و تصمیم‌گیری برای رویدادهای امنیت اطلاعات و ارزیابی نقاط ضعف امنیتی.
vi. دستورالعمل واکنش به مواردی از قبیل تشدید، بازیابی کنترل‌شده از یک حادثه و برقراری تماس با اشخاص یا سازمان‌های داخلی و خارجی.

ب) دستورالعمل‌های اجراشده باید تضمین دهند:

i. اداره مسائل مربوط به حوادث امنیت اطلاعات درون سازمان، بر عهده پرسنل صلاحیت‌دار است.

---

[1] Information security incident management
[2] Response

ii. یک نقطه تماس برای تشخیص و گزارش حوادث امنیتی، مشخص‌شده باشد.
iii. ارائه اطلاعات تماس با مقامات، گروه‌های ذینفع خارجی یا گروهی که مسئول رسیدگی به مسائل مربوط به حوادث امنیت اطلاعات است.

ج) روش‌های گزارش‌گیری باید شامل موارد زیر باشد:

i. آماده‌سازی فرم‌های گزارش رویداد امنیت اطلاعات برای پشتیبانی از فعالیت‌های گزارش‌گیری. این فرم‌ها با یادآوری تمام فعالیت‌های لازم در مورد یک رویداد امنیت اطلاعات، به فرد گزارش‌گیر کمک می‌کند.
ii. دستورالعملی که در مورد یک رویداد امنیت اطلاعات باید انجام شود. به‌عنوان‌مثال: توجه به تمام جزئیات از قبیل نوع نقض یا عدم رعایت، نقص اتفاق افتاده، پیام‌های روی صفحه‌نمایش و گزارش فوری به نقطه تماس و دریافت اقدامات هماهنگ شده.
iii. ارجاع به یک فرآیند رسمی انضباطی، جهت نحوه برخورد با کارمندانی که مرتکب نقض امنیت اطلاعات شده‌اند.
iv. فرآیندهای بازخورد[1] مناسب برای حصول اطمینان از اینکه پس از برخورد با رویداد موردنظر و بسته شدن آن، نتیجه به اطلاع افراد مربوطه رسانده شده باشد.

اهداف مدیریت حوادث امنیت اطلاعات باید با مدیریت توافق شده باشند. همچنین باید مطمئن شد افرادی که عهده‌دار مدیریت حوادث امنیت اطلاعات هستند به درک درستی از اولویت‌های سازمان در این زمینه، دست پیداکرده‌اند.

- **سایر اطلاعات**

راهنمایی دقیق در این زمینه، طی استاندارد ISO/IEC 27035 ارائه گردیده است.

گزارش رویداد امنیت اطلاعات

- **کنترل**

رویداد امنیت اطلاعات باید از طریق کانال‌های مدیریتی مناسب، در اسرع وقت گزارش شوند.

- **راهنمای اجرایی**

[1] feedback

تمام کارمندان و پیمانکاران باید از مسئولیت خود درباره گزارش رویدادهای امنیت اطلاعات با بیشترین سرعت ممکن، آگاهی داشته باشند. آن‌ها همچنین باید از روش گزارش و اطلاعات تماسی که وقایع باید گزارش شود نیز آگاهی داشته باشند.
موقعیت‌هایی که برای گزارش امنیت اطلاعات باید در نظر گرفته شود شامل موارد زیر است:

i. کنترل امنیتی غیر مؤثر.
ii. نقض صحت، محرمانگی و دسترس‌پذیری اطلاعات.
iii. خطاهای انسانی.
iv. عدم رعایت سیاست‌ها و دستورالعمل‌ها.
v. نقض تمهیدات امنیت فیزیکی.
vi. تغییرات کنترل نشده سیستم.
vii. اختلال در عملکرد سخت‌افزار یا نرم‌افزار.
viii. نقض دسترسی.

- **سایر اطلاعات**

اختلال در عملکرد یا رفتار غیرعادی سیستم، ممکن است نشان از یک حمله امنیتی یا نقض امنیتی باشد، لذا رویدادهای امنیتی همیشه باید گزارش شوند.

گزارش ضعف‌های امنیت اطلاعات[1]

- **کنترل**

کارمندان و پیمانکارانی که از سرویس‌ها و سیستم‌های اطلاعاتی سازمان استفاده می‌کنند، باید ملزم به توجه و گزارش هرگونه نقاط ضعف امنیتی مشاهده‌شده یا مشکوک در سیستم‌ها و سرویس‌ها باشند.

- **راهنمای اجرایی**

تمام کارمندان و پیمانکاران موظف‌اند این مسائل را با بیشترین سرعت ممکن، جهت جلوگیری از حوادث امنیت اطلاعات گزارش دهند. مکانیسم‌های گزارش‌دهی باید تا جایی که امکان دارد آسان، قابل‌اجرا و در دسترس باشند.

- **سایر اطلاعات**

باید به کارکنان و پیمانکاران توصیه شود که هیچ تلاشی برای امتحان ضعف امنیتی مشکوک، انجام ندهند. آزمایش ضعف‌های امنیتی ممکن است به سوءاستفاده بالقوه در

[1] Reporting information security weaknesses

سیستم تفسیر شده و همچنین می‌تواند باعث آسیب به سیستم اطلاعات یا سرویس گردد؛ و درنتیجه به ایجاد مسئولیت قانونی برای فرد امتحان کننده منجر خواهد شد.

واکنش به حوادث امنیت اطلاعات

- **کنترل**

به حوادث امنیت اطلاعات باید طبق دستورالعمل‌های مستند، واکنش نشان داده شود.

- **راهنمای اجرایی**

به حوادث امنیت اطلاعات باید توسط نقطه تماس و یا سایر افراد مربوطه در داخل یا خارج سازمان، پاسخ داده شود.

واکنش باید شامل موارد زیر باشد:

i. جمع‌آوری مدارک در اسرع وقت، پس از وقوع حوادث.
ii. انجام تجزیه‌وتحلیل قانونی امنیت اطلاعات، در صورت نیاز.
iii. تشدید، در صورت نیاز.
iv. اطمینان از اینکه تمام فعالیت‌های واکنشی برای تجزیه‌وتحلیل بعدی، به‌درستی گزارش‌گیری و ثبت شده‌اند.
v. اطلاع دادن حادثه امنیت اطلاعات یا هرگونه جزئیات مرتبط، به دیگر افراد و سازمان‌های داخلی یا خارجی که باید از آن اطلاع داشته باشند. (بر اساس اصل Need-to-know)
vi. برخورد با ضعف‌های امنیتی کشف‌شده که علت حادثه بوده و یا به شکل‌گیری آن کمک کرده است.
vii. پس از برخورد موفقیت‌آمیز با یک حادثه، باید به‌طور رسمی آن را بسته و ثبت نمود.

در صورت لزوم جهت شناسایی منبع، باید پس از حادثه تجزیه‌وتحلیل انجام شود.

- **سایر اطلاعات**

اولین هدف در واکنش به حوادث، ازسرگیری "سطح امنیتی نرمال" و سپس شروع به بازیابی‌های لازم، است.

۱۷ - ماده ۱۳: جنبه‌های امنیت اطلاعات در مدیریت تداوم کسب‌وکار[1]

برنامه‌ریزی تداوم امنیت اطلاعات

- **کنترل**

سازمان باید نیازهای خود را جهت امنیت اطلاعات و تداوم مدیریت امنیت اطلاعات در شرایط نامطلوبی مثل بحران یا فاجعه، تعیین نماید.

- **راهنمای اجرایی**

یک سازمان باید مشخص کند که تداوم امنیت اطلاعات به‌دست‌آمده، طی فرآیند مدیریت تداوم کسب‌وکار بوده و یا طی فرآیند مدیریت بازیابی پس از فاجعه[2]. در زمان برنامه‌ریزی برای تداوم کسب‌وکار و بازیابی پس از فاجعه، باید الزامات امنیت اطلاعات مشخص گردند.

در صورت عدم برنامه‌ریزی رسمی جهت تداوم کسب‌وکار و بازیابی پس از فاجعه، مدیریت امنیت اطلاعات باید فرض کند که الزامات امنیتی در مقایسه با شرایط نرمال، در همان شرایط نامطلوب باقی خواهد ماند. روش دیگر آن است که سازمان می‌تواند با تجزیه‌وتحلیل تأثیرات کسب‌وکار مربوط به جنبه‌های امنیت اطلاعات، برای تعیین الزامات امنیت اطلاعات در شرایط نامطلوب اقدام نماید.

- **سایر اطلاعات**

به‌منظور کاهش زمان و تلاش برای تجزیه‌وتحلیل تأثیرات کسب‌وکار مربوط به امنیت اطلاعات، توصیه می‌شود جنبه‌های امنیت اطلاعات در مدیریت تداوم کسب‌وکار نرمال یا مدیریت بازیابی پس از حادثه، جمع‌آوری گردد. این به معنی آن است که الزامات تداوم امنیت اطلاعات، به‌صراحت در مدیریت تداوم کسب‌وکار یا فرآیند مدیریت بازیابی پس از فاجعه، تدوین گردیده است.

اطلاعات بیشتر در مورد مدیریت تداوم کسب‌وکار را می‌توانید در ISO/IEC 27031، ISO 22313، ISO 22301 به دست آورید.

دسترس‌پذیر بودن امکانات پردازش اطلاعات

- **کنترل**

---

[1] Information security aspects of business continuity management

[2] disaster recovery

برای دستیابی به الزامات دسترس‌پذیری امکانات پردازش اطلاعات، باید افزونگی[1] کافی اجرا گردد.

- **راهنمای اجرایی**

سازمان باید الزامات کسب‌وکار برای در دسترس بودن سیستم‌های اطلاعاتی را شناسایی نماید. جایی که با معماری سیستمی موجود امکان تضمین دسترس‌پذیری وجود ندارد، باید معماری یا مؤلفه‌های افزونگی در نظر گرفته شود.
در صورت امکان، سیستم‌های اطلاعاتی افزونه[2] باید مورد آزمایش قرار گیرند تا اطمینان حاصل شود که جایگزینی شکست[3] از یک بخش به بخش دیگر، در نظر گرفته‌شده و به‌درستی نیز کار می‌کند.

- **سایر اطلاعات**

انجام افزونگی می‌تواند باعث افزایش ریسک‌های مربوط به صحت و محرمانگی، در اطلاعات و سیستم‌های اطلاعاتی گردد؛ لذا این نکته باید به هنگام طراحی مدنظر قرار گیرد.

هر چه کوشیدم یک معادل فارسی مناسب برای عبارت **Failover** پیدا کنم، به نتیجه نرسیدم. لذا با توجه به توانایی‌های خودم! معادل "**جایگزینی شکست**"را برای عبارت فوق به کار بردم. برای اینکه حق مطلب ادا شده باشد، مفهوم این عبارت را به‌صورت زیر بیان می‌کنم:
توانایی انتقال عملیات از یک دستگاه به دستگاه دیگر، در زمان بروز شکست، بدون ایجاد وقفه در عملکرد را Failover می‌گویند.



۱۸ - ماده ۱۴: انطباق[4]

انطباق با استانداردها و سیاست‌های امنیتی

- **کنترل**

مدیران باید به‌طور منظم در محدوده مسئولیت خود اقدام به بررسی انطباق فرآیندها و دستورالعمل‌های اطلاعات با سیاست‌های امنیتی مناسب، استاندارد و هر نیازمندی امنیتی دیگر، نمایند.

---
[1] Redundancy
[2] Redundant
[3] Failover
[4] Compliance

- **راهنمای اجرایی**

مدیران باید بدانند که چگونه می‌توان بررسی نمود که الزامات امنیت اطلاعات تعریف‌شده در سیاست‌ها، استانداردها و دیگر مقررات قابل‌اجرا، به‌دست‌آمده باشد. برای بررسی منظم کارآمدی، باید ابزارهای گزارش‌گیری و سنجش خودکار در نظر گرفته شود.

اگر درنتیجه بررسی‌ها یک مورد عدم انطباق پیدا شود، مدیران باید:

i. شناسایی علل عدم انطباق.
ii. ارزیابی اقدامات موردنیاز برای رسیدن به انطباق.
iii. پیاده‌سازی اقدامات اصلاحی مناسب.
iv. بررسی عملکرد اصلاحی، به‌منظور بررسی اثربخشی آن و شناسایی هرگونه کمبود یا نقاط ضعف.

نتایج بررسی و اقدامات اصلاحی انجام‌شده توسط مدیران، باید ثبت و نگهداری گردد. مدیران باید نتایج را به افراد مربوطه گزارش نمایند.

- **سایر اطلاعات**

نظارت عملیاتی بر استفاده از سیستم، در ماده 12.4 استاندارد ISO/IEC 27002، بررسی شده است.

بررسی انطباق فنی

- **کنترل**

سیستم‌های اطلاعات باید به‌طور منظم برای تطابق با استانداردها و سیاست‌های امنیت اطلاعات سازمان، بررسی شوند.

- **راهنمای اجرایی**

انطباق فنی باید ترجیحاً با کمک ابزارهای خودکار بررسی شود. این ابزارها گزارش‌های فنی را جهت تفسیر توسط یک متخصص فنی، تولید می‌کنند. البته روش دیگر، بررسی دستی توسط یک مهندس سیستم باتجربه است.

اگر از تست نفوذ[1] یا ارزیابی آسیب‌پذیری استفاده می‌شود، باید احتیاط کرد که این فعالیت‌ها منجر به در معرض خطر قرار دادن امنیت سیستم نشود. این آزمایش‌ها باید برنامه‌ریزی‌شده، مستند و قابل تکرار باشند.

[1] Penetration Test

هرگونه بررسی انطباق فنی باید توسط افراد شایسته و مجاز و یا تحت نظارت چنین افرادی انجام پذیرد.

- **سایر اطلاعات**

بررسی انطباق فنی شامل آزمایش سیستم‌های عملیاتی است؛ تا از پیاده‌سازی درست کنترل‌های سخت‌افزار و نرم‌افزار اطمینان حاصل گردد. این نوع از بررسی تطابق به یک متخصص فنی خبره نیاز دارد.

توجه داشته باشید که تست نفوذ و ارزیابی آسیب‌پذیری، جایگزینی برای ارزیابی ریسک نیست.

استاندارد ISO/IEC TR 27008، به‌طور خاص به راهنمایی درباره بررسی انطباق فنی می‌پردازد.

بخش دوم

رمزنگاری

فصل سوم

مفاهیم رمزنگاری

فصل چهارم

الگوریتم‌های رمزنگاری متقارن

فصل پنجم

الگوریتم‌های رمزنگاری نامتقارن

فصل ششم

توابع درهم‌سازی و کاربردها

فصل سوم

مفاهیم رمزنگاری

مبحث اول

مفاهیم و اصطلاحات رمزنگاری

مبحث دوم

مفاهیم رمزنگاری متقارن

مبحث سوم

مفاهیم رمزنگاری نامتقارن

مبحث چهارم

مفاهیم توابع درهم سازی

☑ مبحث اول

مفاهیم و اصطلاحات رمزنگاری

رمزنگاری (Cryptography) یکی از پایه‌های اصلی امنیت اطلاعات است. بسیاری از فرآیندهای امنیتی بدون استفاده از رمزنگاری قابل‌دسترس نخواهند بود. رمزنگاری علاوه بر ایجاد محرمانگی، می‌تواند خصوصیات دیگری ازجمله امضای دیجیتال، عدم انکار و امنیت پروتکل‌های مورداستفاده در IT را نیز در اختیار شما قرار دهد.

همان‌طور که در بخش اول کتاب مطالعه کردید، رمزنگاری یکی از مؤلفه‌های پراهمیت در استاندارد سیستم مدیریت امنیت اطلاعات است. استاندارد ISMS فارغ از نام بردن یک الگوریتم رمزنگاری خاص، بسیاری از موارد امنیتی را بر عهده رمزنگاری گذاشته است. البته همان‌طور که در این بخش خواهید دید، الگوریتم‌های رمزنگاری بسیار متنوع بوده و هرکدام دارای نقاط قوت و ضعف نیز می‌باشند. لذا برای استفاده از رمزنگاری، مدیر امنیت اطلاعات باید تمامی جوانب مثل نیازهای سازمان و هزینه پیاده‌سازی رمزنگاری را در زمان طراحی مدنظر قرار دهد.

رمزنگاری دارای مفاهیم و اصطلاحات خاص خود می‌باشد که برای ورود به بحث رمزنگاری دانستن این اصطلاحات و مفاهیم ضروری است. معانی برخی از این اصطلاحات شبیه به هم ولی با کاربردهای متفاوت است، لذا توصیه می‌کنم حتماً به این مفاهیم دقت نمایید.

تعریف رمزنگاری

در گام اول بهتر است با چند جمله خوب به تعریف رمزنگاری بپردازیم.

- **تعریف اول:**

علم[1] یا مطالعه درباره تکنیک‌های نوشتن رازآلود؛ مخصوصاً سیستم‌های کد و رمزگذاری، روش‌ها و مواردی از این قبیل.

- **تعریف دوم:**

دستورالعمل، فرآیند یا روش ساخت و استفاده از نوشتن رازآلود.

- **تعریف سوم:**

علمی است شامل روش‌های رمزگذاری، رمزگشایی و ایجاد و تبادل کلید.

[1] science

اصطلاحات رمزنگاری

- **متن ساده (Plain Text)**

پیام اصلی که به‌صورت ساده و قابل‌فهم است.
متن ساده یا Plain Text در ورودی الگوریتم‌های رمزنگاری قرار می‌گیرد.

- **الگوریتم رمزگذاری (Encryption Algorithm)**

الگوریتمی که با انجام عملیات مشخصی بر روی متن اصلی یا داده‌های ورودی، آن را از حالت ساده و قابل‌فهم خارج می‌نماید.

- **کلید محرمانه (Secret Key)**

کلیدی است مستقل از متن اصلی و الگوریتم رمزگذاری که همراه با متن اصلی در ورودی الگوریتم رمزگذاری قرار می‌گیرد.
خروجی الگوریتم، بر اساس کلید رمز اعمال‌شده می‌باشد؛ بنابراین خروجی یک الگوریتم مشخص بر اساس کلیدهای رمز متفاوت، مختلف خواهد بود.

- **متن رمزشده (Cipher Text)**

متنی که به علت اعمال الگوریتم رمزگذاری، از حالت Plain Text خارج‌شده و غیرقابل فهم است؛ به‌عبارت‌دیگر یک متن قاتی‌پاتی درهم‌ریخته!
این متن وابسته به متن ساده ورودی و کلید رمز مورداستفاده می‌باشد. به‌طوری‌که اگر در یک الگوریتم رمزگذاری برای یک متن ساده ورودی از دو کلید مختلف استفاده کنیم، متن‌های رمزشده خروجی با یکدیگر تفاوت خواهند داشت.
متن رمزشده یا Cipher Text، خروجی الگوریتم رمزگذاری است.

- **الگوریتم رمزگشایی (Decryption Algorithm)**

الگوریتم رمزگشایی جهت تبدیل متن رمزشده به متن ساده اولیه مورداستفاده قرار می‌گیرد. در رمزنگاری متقارن، الگوریتم رمزگشایی اجرای معکوس الگوریتم رمزگذاری می‌باشد. در رمزنگاری متقارن، الگوریتم رمزگشایی متن رمزشده و کلید رمز را دریافت کرده و با انجام عملیات معکوس رمزگذاری، در خروجی متن ساده و قابل‌فهم اولیه را تحویل می‌دهد.
در رمزنگاری نامتقارن، بازهم وظیفه الگوریتم رمزگشایی تبدیل متن رمزشده به متن واضح اولیه است؛ با این تفاوت که الگوریتم رمزگشایی لزوماً معکوس الگوریتم رمزگذاری نیست.

- **رمزگذاری بلوکی(Block Cipher)**

در این حالت پردازش ورودی در یک‌زمان بر روی یک بلوک از عناصر انجام می‌گیرد و بالطبع خروجی الگوریتم نیز به‌صورت بلوکی خواهد بود.
درصورتی‌که رمزگذاری به‌صورت بلوکی انجام شود، رمزگشایی نیز باید به‌صورت بلوکی صورت پذیرد.

- **رمزگذاری جریانی[1] (Stream Cipher)**

در این حالت پردازش بر روی عناصر ورودی به‌صورت مداوم و پشت سر هم انجام می‌گیرد. به‌عبارت‌دیگر پردازش در یک‌زمان بر روی یک المان ورودی انجام‌شده و به‌صورت یک المان در خروجی ظاهر می‌گردد و این فرآیند به‌صورت متوالی برای المان‌های بعدی ادامه می‌یابد.

- **کانال امن (Secure Channel)**

در بحث رمزنگاری منظور از کانال امن، یک کانال قابل‌اطمینان است که بتوان از طریق آن کلیدهای رمز را تبادل نمود، بدون آنکه محرمانگی و درستی آن‌ها از بین برود. البته این کانال امن لزوماً یک کانال ارتباطی منطقی و مخابراتی نیست؛ بلکه می‌تواند یک پیک مورداطمینان یا یک وسیله ارتباطی دیگر نیز باشد.

- **کلید خصوصی (Private Key)**

کلید خصوصی نوعی کلید رمزنگاری است که باید به‌صورت محرمانه نزد صاحب کلید نگهداری شود؛ تا تضمین شود که تنها صاحب کلید است که امکان استفاده از آن را دارد. با توجه به شرایط، از این کلید می‌توان هم برای رمزگذاری و هم برای رمزگشایی استفاده نمود. کلیدعمومی در رمزنگاری نامتقارن کاربرد دارد.

- **کلید عمومی (Public Key)**

کلید عمومی نوعی کلید رمزنگاری است که می‌تواند توسط هرکسی مورداستفاده قرار گیرد. به‌عبارت‌دیگر، صاحب کلیدعمومی، آن را در اختیار تمام کسانی که قصد برقراری یک ارتباط امن دارند، قرار می‌دهد. کاربرد اصلی کلیدعمومی جهت استفاده در عملیات رمزگذاری است، ولی بسته به شرایط از آن می‌توان جهت رمزگشایی نیز استفاده نمود. کلیدعمومی در رمزنگاری نامتقارن کاربرد دارد.

در الگوریتم رمزنگاری نامتقارن، برای انجام عملیات رمزگذاری و رمزگشایی، کلیدهای عمومی و خصوصی مکمل یکدیگر هستند؛ همواره باید از یکی برای رمزگذاری و از دیگری برای رمزگشایی استفاده نمود.

---

[1] برخی از علمای فن از رمزگذاری جریانی بانام رمزگذاری نهری نیز یاد می‌کنند.

- **زیرساخت کلید عمومی (Public Key Infrastructure)**

زیرساخت کلیدعمومی که به‌اختصار PKI نیز نامیده می‌شود، مجموعه‌ای است شامل: سخت‌افزار، نرم‌افزار، افراد، فرآیندها، تکنیک‌های رمزنگاری و سیاست‌هایی که برای ایجاد، مدیریت، استفاده، ذخیره و لغو گواهی‌نامه دیجیتال مورداستفاده قرار می‌گیرند.
در رمزنگاری نامتقارن، زیرساخت کلید عمومی وظیفه تخصیص کلید را بر عهده دارد.

- **گواهی دیجیتال (Digital Certification)**

در رمزنگاری نامتقارن، گواهینامه دیجیتال برای تائید کلید عمومی کاربرد دارد.

- **مرکز صدور گواهی (Certificate Authority)**

مرکز صدور گواهی که به‌اختصار CA نیز گفته می‌شود، نهادی است که وظیفه صدور گواهینامه دیجیتال را بر عهده دارد.

انواع رمزنگاری

رمزنگاری را به‌صورت کلی می‌توان در سه گروه اصلی زیر طبقه‌بندی نمود:

۱- **رمزنگاری متقارن (Symmetric)**

درصورتی‌که ارسال‌کننده و دریافت‌کننده هر دو از یک کلید همسان استفاده نمایند، به آن رمزنگاری متقارن گفته می‌شود.
برای رمزنگاری متقارن از نام‌های دیگری مثل رمزنگاری متعارف[1]، کلید محرمانه و تک کلید[2]، نیز استفاده می‌شود.

۲- **رمزنگاری نامتقارن (Asymmetric)**

درصورتی‌که ارسال‌کننده و دریافت‌کننده از کلیدهای متفاوتی استفاده نمایند، به آن رمزنگاری نامتقارن گفته می‌شود.
برای رمزنگاری نامتقارن از نام‌های دیگری مثل کلید عمومی و دو کلید[3]، نیز استفاده می‌شود.

۳- **توابع درهم سازی (Hash Function)**

توابع درهم سازی، فارغ از حجم دیتای ورودی، همواره یک خروجی با طول ثابت ارائه می‌نمایند. به خروجی این توابع اثرانگشت نیز می‌گویند؛ چراکه به ازای هر دیتای ورودی یک‌رشته خروجی منحصربه‌فرد تولید می‌نمایند.

---
[1] Conventional
[2] Single-key
[3] Two-key

انواع حملات به سیستم رمزنگاری

به‌طورمعمول هدف از حملات به سیستم‌های رمزنگاری به دست آوردن کلید رمز مورداستفاده می‌باشد؛ چراکه در این صورت هر متنی را می‌توان رمزگشایی نمود.

حملات جهت به دست آوردن کلید رمزنگاری به دو گروه اصلی زیر تقسیم می‌شوند:

- **کشف رمز (Cryptanalysis)**

حملات کشف رمز، حملاتی مبتنی بر ماهیت الگوریتم و احتمالاً اطلاع از ویژگی‌های کلی متن اصلی، می‌باشند.

در این نوع حملات با سوءاستفاده از ویژگی‌های الگوریتم، سعی می‌شود که کلید مورداستفاده یا متن اصلی را استخراج نمایند.

- **جستجوی فراگیر (Brute-force)**

در این حالت، مهاجم بر روی قسمتی از متن رمزشده، تمام کلیدهای مفروض را امتحان می‌نماید تا بتواند متن را از حالت رمزشده خارج نموده و کلید را پیدا کند. به‌عبارت‌دیگر مهاجم تمام حالات ممکن را برای رسیدن به کلید رمز امتحان می‌کند.

در جستجوی فراگیر، به‌طور متوسط برای رسیدن به موفقیت، باید نیمی از تمام کلیدهای ممکن امتحان شود.

انواع حملات کشف رمز

حملات کشف رمز یا Cryptanalysis می‌تواند در حالت‌های مختلفی رخ دهد. در ادامه به ذکر چند نوع از آن‌ها می‌پردازیم.

- **Ciphertext Only**

در این حالت مهاجم ضمن دانستن الگوریتم رمزگذاری مورداستفاده، یک متن رمزشده نیز در اختیار دارد.

- **Known Plaintext**

مهاجم از الگوریتم مورداستفاده آگاهی دارد. همچنین یک یا چند متن رمزشده به همراه متن ساده مربوطه را نیز در اختیار دارد.

- **Chosen Plaintext**

مهاجم از الگوریتم آگاهی داشته و متن رمزشده هم در اختیار دارد. علاوه بر این موارد امکان آن را دارد که متن ساده دلخواه خود را به سیستم رمزنگاری داده و متن متناظر رمزشده بر اساس کلید محرمانه را به دست بیاورد.

- **Chosen Ciphertext**

مهاجم از الگوریتم آگاهی داشته و متن رمزشده هم در اختیار دارد. علاوه بر این موارد مهاجم امکان این را دارد که متن رمزشده موردنظر خود را به سیستم رمزنگاری داده و متن ساده متناظر را به دست آورد.

- **Chosen Text**

این حالت تلفیق دو حالت قبلی است؛ یعنی مهاجم ضمن آگاهی از الگوریتم و در اختیار داشتن متن رمز شده، امکان آن را دارد که از سیستم رمزنگاری برای متن ساده یا متن رمزشده موردنظر خود، متن رمزشده یا متن ساده متناظر بر اساس کلید محرمانه را به دست آورد.

اصول انتخاب سیستم رمزنگاری

هرقدر الگوریتم رمزنگاری قوی و هراندازه طول کلید زیاد باشد، بازهم نمی‌توان از شکسته نشدن آن اطمینان صددرصدی داشت. از طرف دیگر در بعضی مواقع شاید اصلاً نیازی به هزینه کردن برای استفاده از الگوریتم‌های پیشرفته نباشد. لذا در زمان انتخاب الگوریتم با توجه به دیتای موردنظرتان سعی کنید یک یا هر دو اصل مهم زیر را مدنظر قرار دهید:

۱. هزینه شکستن رمز بیش از ارزش اطلاعات رمزگذاری شده باشد.
۲. زمان لازم برای شکستن رمز بیش از طول عمر مفید اطلاعات رمزشده باشد.

شروط شانون

یکی از راه‌های مهم کشف رمز، تحلیل نشانه‌های آماری در متن رمزشده است. اگر بتوان راهکاری اتخاذ نمود که نشانه‌های آماری را در متن رمزشده از بین بُرد یا به حداقل رساند؛ شخص رمز یاب کار سختی را برای کشف رمز پیش رو خواهد داشت.

آقای Claude Shannon، برای رفع نگرانی فوق، اقدام به معرفی دو شرط اساسی ساخت بلوک برای هر سیستم رمزنگاری، نموده است:

- **پراکندگی (Diffusion)**

باید ساختار آماری متن ساده در سرتاسر متن رمزشده پراکنده شود. این ویژگی وقتی به دست می‌آید که تغییر یک کاراکتر در ورودی، چندین کاراکتر را در خروجی تحت تأثیر قرار دهد.

- **به‌هم‌ریختگی (Confusion)**

باید بین کلید و متن رمزشده یک رابطه غیرخطی و پیچیده برقرار گردد. این کار باعث می‌شود که رمز یاب در کشف کلید رمز ناموفق باشد.

پنهان‌کاری (steganography)

یکی از تکنیک‌هایی که می‌تواند جهت امنیت مورداستفاده قرار گیرد، تکنیک پنهان‌کاری است؛ البته توجه داشته باشید که پنهان‌کاری صرفاً یک تکنیک است و نه یک الگوریتم رمزنگاری. در الگوریتم‌های رمزنگاری سعی می‌شود متن از حالت خوانا به یک حالت نامفهوم درآید تا درصورتی‌که به دست افراد غیرمجاز رسید، امکان فهم متن وجود نداشته باشد؛ اما در پنهان‌کاری هدف مخفی کردن موجودیت پیام است.

مثلاً یک فرم پنهان‌کاری که البته وقت‌گیر است، نوشتن یک متن به‌ظاهر معمولی است که مثلاً با در کنار هم قرار دادن حرف اول هر کلمه، یا اولین کلمه هر جمله، متن موردنظر حاصل شود. همچنین استفاده از جوهر نامرئی، ایجاد علامت بر روی حروف یک متن تایپ‌شده مثل روزنامه، مخفی کردن متن در تصویر و کارهایی از این قبیل نیز جزو تکنیک‌های پنهان‌کاری محسوب می‌شود. البته امروزه برای پنهان‌کاری ابزار و روش‌های زیادی در دسترس هستند؛ یکی از سایت‌هایی که به‌صورت آنلاین متن شما را در تصویر موردنظرتان پنهان می‌کند، سایت زیر است:

http://manytools.org/hacker-tools/steganography-encode-text-into-image/

رمزنگاری و فشرده‌سازی[1]

فشرده‌سازی با فاکتور گرفتن از کاراکترها یا عبارات تکرار شده، حجم متن را کاهش می‌دهد. اما در رمزنگاری، قصد الگوریتم از بین بردن نشانه‌های آماری متن است. لذا درصورتی‌که قبل از فشرده‌سازی، اقدام به رمزگذاری متن شود، بهره‌وری فشرده‌سازی به‌شدت کاهش می‌یابد؛ چراکه عبارات تکرار شده متن به مقدار قابل‌توجهی کاهش‌یافته است.

لذا درصورتی‌که قصد دارید از رمزنگاری و فشرده‌سازی در کنار هم استفاده نمایید، بهتر است ابتدا عملیات فشرده‌سازی را انجام داده و سپس عملیات رمزنگاری را بر روی خروجی فشرده‌سازی اعمال نمایید.

در فصل ششم، مثال‌هایی جهت استفاده هم‌زمان از رمزنگاری و فشرده‌سازی و ترتیب اعمال آن‌ها، ذکرشده است.

[1] Compression

☑ مبحث دوم

مفاهیم رمزنگاری متقارن

رمزنگاری متقارن یا Symmetric، نوعی رمزنگاری است که در آن ارسال‌کننده و دریافت‌کننده از کلید یکسانی استفاده می‌نمایند. این کلیدها باید قبلاً بین هر دو طرف به توافق رسیده و یا اینکه از طریق کانال امنی بین طرفین انتقال یابد.

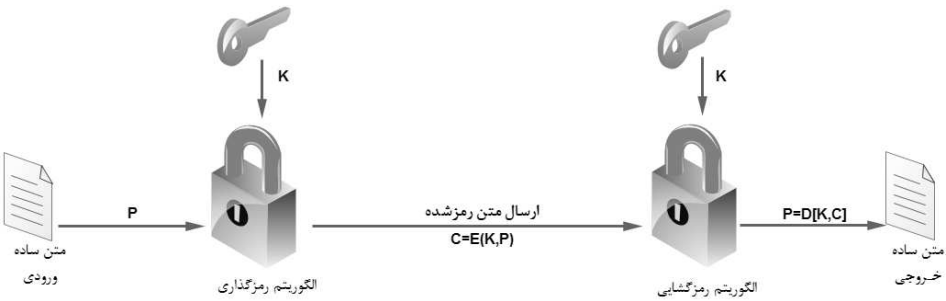
همان‌طور که قبلاً گفته شد برای رمزنگاری متقارن از نام‌های دیگری مثل رمزنگاری متعارف، کلید محرمانه و تک کلید، نیز استفاده می‌شود.

در رمزنگاری متقارن، در مرحله اول باید کلید رمز بین طرفین ارسال‌کننده و دریافت‌کننده به اشتراک گذاشته‌شده باشد. به دلیل اینکه رمزنگاری متقارن فاقد مکانیسمی برای توافق یا انتقال کلید بین طرفین است؛ لذا توافق باید قبل از رمزنگاری انجام‌شده باشد و یا اینکه کلید مورداستفاده، باید از طریق یک کانال ارتباطی امن به دست گیرنده برسد.

در مرحله بعد، ارسال‌کننده متن ساده ورودی را با توجه به الگوریتم رمزنگاری و بر اساس کلید مشخص‌شده، رمزگذاری کرده و نتیجه خروجی که یک متن رمزشده است را از طریق یک کانال ارتباطی معمولی و احتمالاً ناامن برای گیرنده ارسال می‌نماید.

گیرنده که در گام اول بر روی کلید مورداستفاده با فرستنده توافق کرده، پس از دریافت متن رمزشده، با استفاده از کلید و اجرای معکوس عملیات رمزگذاری، عملیات رمزگشایی را انجام داده و در خروجی همان متن ساده اولیه را تولید می‌نماید.

در شکل زیر یک مدل ساده از عملیات رمزنگاری متقارن نمایش داده شده است.



P=Plaintext , **E**=Encryption , **C**=Ciphertext , **D**=Decryption , **K**=Key



به‌طورمعمول در همه موارد (به‌جز مقاصد نظامی) از الگوریتم‌های رمزنگاری استاندارد که به‌صورت عمومی منتشرشده، استفاده می‌شود؛ لذا روش عملکرد الگوریتم برای همه واضح است. به همین دلیل تمام امنیت متن رمزشده وابسته به کلید مورداستفاده خواهد بود. درصورتی‌که مهاجم کلید را به دست آورد، ازآن‌پس به تمام اطلاعات به‌راحتی دسترسی خواهد داشت.

اصول کلی رمزگذاری

برای تبدیل یک متن ساده به متن رمزشده، دو اصل کلی وجود دارد که تمام الگوریتم‌های رمزگذاری متقارن بر اساس یک یا هر دو روش زیر، عملیات خود را انجام می‌دهند:

۱. **جایگزینی (Substitution)**

در این روش هر عنصر موجود در متن (بیت، حرف و یا گروهی از بیت یا حروف) با یک عنصر دیگر جایگزین می‌شوند.

در این روش برای اینکه مشخص باشد چه عناصری جایگزین چه عناصر دیگری می‌شوند، به یک جدول حاوی عناصر متناظر نیاز است.

۲. **جابجایی (Transposition)**

درروش جابجایی، عناصر موجود در متن دوباره چیده شده و یا به‌عبارت‌دیگر با یکدیگر جابجا می‌شوند. البته این جابجایی به شرطی است که هیچ اطلاعاتی ازدست‌داده نشود و همچنین عملیات برگشت‌پذیر نیز باشد.

تقسیم‌بندی الگوریتم‌های جایگزینی

الگوریتم‌های رمزنگاری که از اصل جایگزینی برای عملیات خود استفاده می‌کنند در دودسته کلی زیر تقسیم‌بندی می‌شوند:

i. **Monoalphabetic[1] Substitution Cipher**

در این حالت هر حرف از حروف الفبا با یک حرف دیگر جایگزین گردیده و این جایگزینی همواره تکرار خواهد شد. به‌عنوان‌مثال اگر حرف m با حرف p جایگزین شود، در تمام جایگزینی‌های دیگر همواره p جایگزین m می‌شود.

در حالت‌های چندحرفی نیز، خروجی هر زوج یکسان همواره تکرار خواهد شد. مثلاً اگر حروف ab با gh جایگزین شود، پس جایگزین ab همواره gh خواهد بود.

---

[1] تک الفبایی

ii. Polyalphabetic[1] Substitution Cipher

در این حالت امکان جابجایی هر حرف با چند حرف دیگر وجود دارد. به‌عنوان‌مثال حرف m ممکن است یک‌بار با حرف p و دفعه بعد با حرف s جایگزین شود.

تمام روش‌های Polyalphabetic، دارای ویژگی‌های مشترک زیر هستند:

i. مجموعه‌ای از قوانین مربوط به Monoalphabetic Substitution مورداستفاده قرارگرفته شده است.

ii. برای انجام دگرگونی[2] باید یک کلید دارای قوانین خاص و مشخص، مورداستفاده قرار گیرد.

رمزنگاری کلاسیک و مدرن

الگوریتم‌های رمزنگاری را می‌توان از جنبه‌های مختلف گروه‌بندی نمود. یکی از این جنبه‌ها، کلاسیک و مدرن بودن الگوریتم‌ها است.

الگوریتم‌های رمزنگاری متقارن را می‌توان در دو گروه کلاسیک و مدرن قرار داد. گروه کلاسیک شامل الگوریتم‌های سنتی و قدیمی است و گروه مدرن شامل الگوریتم‌های پیشرفته‌ای که امروزه می‌توان از آن‌ها استفاده کرد.

الگوریتم‌های رمزنگاری نامتقارن نیز عموماً به‌عنوان رمزنگاری مدرن شناخته‌شده و در گروه کلاسیک فاقد نماینده هستند.

هرچند گروه کلاسیک شامل الگوریتم‌های متقارن قدیمی است که امروزه دیگر قابل‌استفاده نیستند؛ اما برای فهم بهتر و یادگیری اصول رمزنگاری، باید به آن‌ها نیز پرداخته شود. لذا در این مبحث به‌طور اجمالی به بررسی بعضی از الگوریتم‌های کلاسیک می‌پردازیم. اما در فصل چهارم، مشهورترین و پرکاربردترین الگوریتم‌های رمزنگاری متقارن که در گروه مدرن قرار می‌گیرند شرح داده می‌شود.

انواع رمزنگاری کلاسیک

در این قسمت برای درک بهتر رمزنگاری به بررسی چند نمونه از الگوریتم‌های رمزنگاری متقارن که در گروه کلاسیک قرار می‌گیرند، خواهیم پرداخت. با توجه به قدیمی بودن رمزهای کلاسیک، عملیات آن‌ها به‌صورت دستی و یا توسط تجهیزات ابتدایی انجام می‌گرفته است.

---

[1] چند الفبایی

[2] Transformation

۱. الگوریتم رمز سزار (Caesar Cipher)

اولین و ساده‌ترین الگوریتمی که بر اساس روش جایگزینی ابداع شد، الگوریتم سزار بود که توسط آقای ژولیوس سزار ارائه گردید. در الگوریتم سزار هر حرف از حروف الفبا با حرفی که سه پله بالاتر از خودش (در جدول حروف الفبا) است، جایگزین می‌گردد.

برای مثال به خروجی رمزشده جمله زیر توجه نمایید[۱]:

```
Plain: meet me after the toga party
Cipher: PHHW PH DIWHU WKH WRJD SDUWB
```

همان‌طور که ملاحظه می‌کنید، هر حرف در متن اصلی با سه حرف بالاتر از خود در جدول حروف الفبای انگلیسی جایگزین شده است.

اگر به حروف الفبا طبق جدول زیر عدد اختصاص دهیم:

a	b	c	d	e	f	g	h	i	j	k	l	m
0	1	2	3	4	5	6	7	8	9	10	11	12
n	o	p	q	r	s	t	u	v	w	x	y	z
13	14	15	16	17	18	19	20	21	22	23	24	25

و آنگاه الگوریتم آقای سزار را به‌صورت فرمول بنویسیم:

فرمول رمزگذاری: $C = \mathrm{E}(k, p) = (p + k) \bmod 26$

فرمول رمزگشایی: $p = \mathrm{D}(k, C) = (C - k) \bmod 26$

توجه داشته باشید که منظور از حرف k موجود در فرمول‌های فوق، کلید رمزنگاری است؛ که در این الگوریتم عدد 3 در نظر گرفته‌شده است. سپس برای اولین حرف جمله فوق که حرف m است، از طریق فرمول حرف جایگزین را محاسبه می‌نماییم:

$$C = \mathrm{E}(3, p) = (12 + 3) \bmod 26 = 15$$

عدد مربوط به حرف m، 12 است که در صورت قرار دادن در فرمول عدد 15 حاصل می‌شود. عدد 15 هم نشانگر حرف p است. بقیه کاراکترها را هم به همین راحتی محاسبه و جایگزین می‌نماییم.



البته شما که مهندس هستید! ولی محض یادآوری عرض می‌کنم که منظور از $\bmod 26$، باقیمانده تقسیم انجام‌شده با مخرج 26 است. برای مثال:
$25 \bmod 5 = 0$ و یا $25 \bmod 4 = 1$.

[۱] به‌عنوان یک قرارداد، از حروف کوچک جهت نمایش متن‌ساده و از حروف بزرگ جهت متن‌رمزشده استفاده می‌کنیم.

۲. الگوریتم رمز Playfair

در الگوریتم سزار جایگزینی به‌صورت تک‌حرفی انجام می‌شد، ولی در الگوریتم Playfair هر دو حرف جایگزین دو حرف دیگر می‌شوند. به دلیل اینکه جفت‌های شبیه به‌هم، همواره جایگزین‌های یکسان دارند؛ الگوریتم Playfair نیز در گروه Monoalphabetic Cipher قرار می‌گیرد.

این الگوریتم بر اساس یک ماتریس 5×5 از حروف الفبا کار می‌کند. به دلیل اینکه حروف الفبای انگلیسی ۲۶ حرف دارد، باید یک حرف آن را حذف و یا همراه حرف دیگری در یک‌خانه از ماتریس قرار دهیم.

الگوریتم Playfair از کلید محرمانه نیز استفاده می‌نماید. به این صورت که حروف کلید محرمانه را در خانه‌های ابتدایی ماتریس قرار داده (بدون حروف تکراری) و در خانه‌های بعدی، اقدام به درج حروف دیگر (منهای حروف موجود در کلید) بر اساس ترتیب الفبا، می‌نماید.

برای مثال اگر کلید رمز ما monarchy باشد، جدول ماتریس به‌صورت زیر خواهد بود:

M	O	N	A	R
C	H	Y	B	D
E	F	G	I/J	K
L	P	Q	S	T
U	V	W	X	Z

پس از تکمیل جدول، باید حروف متن ساده را دوتادوتا جدا نمود و هر جفت حروف را با توجه به قوانین زیر در جدول قرار داده و حروف جایگزین آن‌ها را به دست آورد.

- اگر در هر جفت حرف متن اصلی تکرار وجود داشته باشد باید با یک حرف جداکننده مثل x آن‌ها را از هم جدا نمود. مثلاً کلمه Balloon را اگر دو حرف، دو حرف جدا کنیم، به‌صورت Ba ll oo n درمی‌آید که در دو جفت از حروف دارای تکرار هستیم. حال اگر از حرف جداکننده x استفاده نمائیم، جفت‌های تشکیل‌دهنده به این صورت تشکیل خواهد شد: Ba lx lo on.
- اگر یک جفت حروف متن اصلی در یک سطر ماتریس قرار گیرند، هر حرف با حرف سمت راستی خود در همان سطر به‌صورت مدور، جایگزین می‌شود. برای مثال اگر حروف متن اصلی ar باشد، حروف جایگزین RM خواهد بود.
- اگر یک جفت حروف متن اصلی در یک ستون ماتریس قرار گیرند، هر حرف با حرف پائینی خود در همان ستون به‌صورت مدور، جابجا می‌شود. برای مثال اگر حروف متن اصلی mu باشد، حروف جایگزین CM خواهد بود.
- درصورتی‌که جفت حروف در یک سطر یا یک ستون قرار نگیرند، آنگاه حرفی جایگزین حرف اصلی می‌شود که در همان سطر ولی در ستون حرف دیگر قرارگرفته باشد.

برای مثال اگر حروف متن اصلی hs باشد، حرف جایگزین برای h، حرف B است، چراکه تلاقی سطر h و ستون s، حرف B می‌شود. همچنین برای حرف s، حرف P که در همان سطر و در ستون حرف h قرار دارد، جایگزین می‌گردد.

الگوریتم Playfair که در زمان خود و در بین Monoalphabetic Cipherها پیشرفت بزرگی محسوب می‌گردید را می‌توان بهترین الگوریتم شناخته‌شده رمزنگاری چندحرفی، دانست. به علت اینکه الگوریتم Playfair برای مدت‌زمانی طولانی غیرقابل شکستن مانده بود، ارتش انگلیس از آن به‌عنوان استاندارد سیستم‌هایش در جنگ جهانی اول استفاده نمود. البته این الگوریتم همچنان در جنگ‌جهانی دوم نیز توسط ارتش امریکا و متفقین مورداستفاده قرار گرفت.

۳. الگوریتم رمز ویجینر (Vigenère Cipher)

الگوریتم رمز ویجینر یکی از ساده‌ترین و بهترین رمزهایی است که در گروه Polyalphabetic Substitution Cipher قرار می‌گیرد.

در این الگوریتم همانند رمز سزار به هریک از حروف الفبا یک عدد از 0 تا 25 تخصیص داده و سپس کلید موردنظر را انتخاب می‌نماییم.

پس از مشخص شدن طول کلید محرمانه، باید حروف متن ساده را بر اساس تعداد حروف کلید، دسته‌بندی نمائیم. سپس طبق فرمول باید جایگزین هر حرف را مشخص نمائیم.

برای فهم بهتر فرمول ویجینر:

اگر دنباله حروف متن ساده را به‌صورت زیر در نظر بگیریم:

$$P = p_0, p_1, p_2, \dots, p_{n-1}$$

و دنباله حروف کلید را به‌صورت زیر (معمولاً در آن $m<n$ است)

$$K = k_0, k_1, k_2, \dots, k_{m-1}$$

و دنباله متن رمز شده به‌صورت زیر باشد:

$$C = C_0, C_1, C_2, \dots, C_{n-1}$$

آنگاه محاسبه متن رمزشده به‌صورت زیر خواهد بود:

$$C = C_0, C_1, C_2, \dots, C_{n-1} = E(K, P) = E[(k_0, k_1, k_2, \dots, k_{m-1}), (p_0, p_1, p_2, \dots, p_{n-1})]$$
$$= (p_0 + k_0) mod\ 26, (p_1 + k_1) mod\ 26, \dots, (p_{m-1} + k_{m-1}) mod\ 26,$$
$$(p_m + k_0) mod\ 26, (p_{m+1} + k_1) mod\ 26, \dots, (p_{2m-1} + k_{m-1}) mod\ 26, \dots$$

در فرمول فوق توجه داشته باشید که:

به علت کوچک‌تر بودن طول کلید از طول متن ساده، باید متن اصلی به‌اندازه طول کلید تقسیم‌بندی شود؛ لذا از p_{m-1} و p_{2m-1} استفاده گردیده و این روند تا تمام شدن متن اصلی ادامه

خواهد یافت. برای مثال اگر کلید دارای 4 حرف و متن ساده دارای 20 حرف باشد، متن ساده باید به ۵ بخش ۴ حرفی تقسیم‌شده و در فرمول فوق قرار گیرد.

اگر بخواهیم فرمول فوق را ساده‌سازی کنیم، می‌توان به شکل زیر آن را نوشت:

$$C_i = (p_i + k_{i \bmod m}) mod\ 26$$

$$p_i = (C_i - k_{i \bmod m}) mod\ 26$$

برای فهم بهتر فرمول‌های فوق، به مثال زیر توجه فرمایید:

در ابتدا مانند سزار به حروف الفبا عدد تخصیص می‌دهیم:

a	b	c	d	e	f	g	h	i	j	k	l	m
0	1	2	3	4	5	6	7	8	9	10	11	12
n	o	p	q	r	s	t	u	v	w	x	y	z
13	14	15	16	17	18	19	20	21	22	23	24	25

بعد اقدام به تعیین کلید می‌نماییم: Key = mtr

سپس متن موردنظر را مشخص می‌کنیم: Plaintext = i love you

حالا باید با توجه به طول کلید، متن را به بخش‌های سه‌حرفی تقسیم و رمز نمائیم:

```
          key: mtrmtrmt
    plaintext: iloveyou
   ciphertext: UEFHXPAN
```

برای رمزشدن حروف به‌صورت زیر عمل می‌کنیم:

$$C_0 = (p_0 + k_{0 \bmod 3}) mod\ 26 = (8 + 12) mod\ 26 = 20$$
$$C_1 = (p_1 + k_{1 \bmod 3}) mod\ 26 = (11 + 19) mod\ 26 = 4$$
$$C_2 = (p_2 + k_{2 \bmod 3}) mod\ 26 = (14 + 17) mod\ 26 = 5$$
$$C_3 = (p_3 + k_{3 \bmod 3}) mod\ 26 = (21 + 12) mod\ 26 = 7$$
$$C_4 = (p_4 + k_{4 \bmod 3}) mod\ 26 = (4 + 19) mod\ 26 = 23$$
$$C_5 = (p_5 + k_{5 \bmod 3}) mod\ 26 = (24 + 17) mod\ 26 = 15$$
$$C_6 = (p_6 + k_{6 \bmod 3}) mod\ 26 = (14 + 12) mod\ 26 = 0$$
$$C_7 = (p_7 + k_{7 \bmod 3}) mod\ 26 = (20 + 19) mod\ 26 = 13$$

در محاسبه C_0، طبق جدول حروف، عدد مربوط به حرف اول متن ساده که حرف i می‌باشد معادل 8 است و همچنین عدد حرف اول کلید که حرف m است معادل 12 می‌باشد. درنتیجه خروجی فرمول فوق عدد 20 خواهد بود که طبق جدول به حرف U اشاره دارد. همین روال ادامه می‌یابد تا آخرین حرف متن ساده به متن رمزی تبدیل گردد.

در این الگوریتم هرچه طول کلید بزرگ‌تر و تعداد حروف تکراری آن کمتر باشد، متن رمزشده بهتری خواهیم داشت.

۴. الگوریتم رمز نرده راه‌آهن (Rail Fence)

تمام تکنیک‌هایی که تاکنون شرح داده شد، جزء گروه جایگزینی بودند. همان‌طور که گفته شد اصل دیگری که رمزنگاری بر اساس آن انجام می‌شود اصل جابجایی است. الگوریتم نرده راه‌آهن از اصل جابجایی برای رمزنگاری استفاده می‌نماید.

الگوریتم نرده‌راه‌آهن یک الگوریتم ساده است که متن ساده را در آن ستونی نوشته و سطری می‌خوانیم. برای مثال جمله meet me after the toga party، را روی دو نرده راه‌آهن به‌صورت زیر می‌نویسیم:

```
m e m a t r h t g p r y
 e t e f e t e o a a t
```

همان‌طور که مشاهده می‌کنید، یک حرف بالا و یک حرف پائین نوشته‌شده که اگر آن را به‌صورت سطری بخوانیم، درنهایت باعث به وجود آمدن جمله بدون معنی زیر می‌شود:

```
MEMATRHTGPRYETEFETEOAAT
```

به دلیل سادگی، روش فوق خیلی راحت قابل رمزیابی است. اما اگر به‌جای دو نرده از تعداد نرده‌های بیشتری استفاده کنیم، الگوریتم پیچیده‌تر خواهد شد.

برای پیچیده‌تر کردن این روش، پیام را در یک مستطیل به‌صورت سطر به سطر می‌نویسیم ولی به‌صورت ستون به ستون می‌خوانیم. البته ترتیب خواندن ستون‌ها را نیز بر عهده کلید مورداستفاده قرار می‌دهیم.

برای فهم بهتر مطلب به مثال زیر توجه نمایید:

```
Plaintext:  attack postponed until two am
      Key:  4312567
```

با توجه به اینکه کلید ما دارای هفت کاراکتر است، پس باید ابتدا متن ساده را در یک جدول هفت ستون به‌صورت سطر به سطر بنویسیم:

```
a t t a c k p
o s t p o n e
d u n t i l t
w o a m x y z
```

در جدول فوق حروف x,y,z جهت پر کردن جدول بوده و جزء پیام محسوب نمی‌شوند.

حالا کلید را بالای جدول درج می‌کنیم:

```
      Key: 4 3 1 2 5 6 7
Plaintext: a t t a c k p
           o s t p o n e
           d u n t i l t
           w o a m x y z
```

سپس به ترتیب اعداد کلید، حروف را به‌صورت ستونی می‌خوانیم. همان‌طور که از اعداد کلید مشخص است، ستون سوم به‌عنوان اولین ستون، ستون چهارم به‌عنوان دومین ستون و همین‌طور ادامه می‌دهیم تا درنهایت ستون هفتم نیز به‌عنوان آخرین ستون خوانده و به‌طور متوالی نوشته می‌شوند؛ و درنهایت متن رمزشده زیر حاصل می‌گردد.

Ciphertext: TTNAAPTMTSUOAODWCOIXKNLYPETZ

شاید روش فوق نیز به نظر شما ساده باشد؛ که البته هست. به همین خاطر شما می‌توانید عملیات فوق را چند بار تکرار نمایید. مثلاً متن رمزشده مرحله اول را مجدداً با همان کلید رمزکرده و این کار را به‌دفعات دلخواه انجام دهید. در این صورت پیچیدگی متن رمزشده بالا می‌رود. البته واضح است که در زمان رمزگشایی نیز باید مراحل را به همان تعداد انجام دهید.

کلیدهای One-Time Pad

یک افسر ارتش بنام Joseph Mauborgne، برای افزایش امنیت کلیدهای مورداستفاده در الگوریتم‌های رمزنگاری، پیشنهاد استفاده از کلیدهای تصادفی را مطرح کرد؛ به‌طوری‌که طول آن برابر طول پیام بوده و کلید فقط باید برای رمزگذاری و رمزگشایی همان پیام مورداستفاده قرارگرفته و سپس دور انداخته شود. هر پیام جدید نیاز به یک کلید جدید برابر با طول پیام دارد.

ازآنجاکه کلیدهای One-Time Pad به‌صورت تصادفی ایجادشده و هیچ رابطه آماری نیز با متن اصلی ندارند، باعث می‌شود متن رمزشده خروجی، حاوی هیچ اطلاعاتی از متن اصلی نباشد؛ لذا شکستن این رمز به‌راحتی امکان‌پذیر نیست.

دلیل امنیت On-Time Pad، در تولید و استفاده از کلیدهای تصادفی است. وقتی جریان کاراکترهای کلید به‌صورت تصادفی باشد، بالطبع جریان کاراکترهای متن رمزشده نیز تصادفی خواهد بود؛ پس هیچ نشانه آماری وجود نخواهد داشت که مهاجم بتواند بر اساس آن اقدام به شکستن رمز نماید.

هرچند که در تئوری، On-Time Pad امنیت کامل را به ما ارائه می‌دهد، اما در عمل اجرای آن دارای دو مشکل اساسی است:

i. مشکل عملی اول، ساخت مقادیر زیادی کلیدهای تصادفی است. سیستم‌های بزرگ دائماً به میلیون‌ها کلید تصادفی نیاز خواهند داشت؛ که این حجم زیاد کار می‌تواند قابل‌توجه باشد.
ii. مشکل اساسی و حادتر این روش، توزیع و محافظت از کلیدها است. برای هر پیامی که ارسال می‌شود باید یک کلید برابر با طول همان پیام نیز در اختیار فرستنده و گیرنده قرار گیرد.

☑ مبحث سوم

مفاهیم رمزنگاری نامتقارن

رمزنگاری نامتقارن (Asymmetric) یا کلیدعمومی (Public Key)، نوعی رمزنگاری است که در آن، رمزگذاری و رمزگشایی با دو کلید متفاوت انجام می‌پذیرد. این دو کلید بانام‌های کلیدخصوصی (Private Key) و کلیدعمومی (Public Key) شناخته می‌شوند.

یکی از مهم‌ترین دلایل پیدایش رمزنگاری نامتقارن یا کلید عمومی، مشکل تبادل کلید در الگوریتم‌های رمزنگاری متقارن بود. برخلاف رمزنگاری متقارن که رمزگذاری و رمزگشایی با یک کلید یکسان انجام می‌گرفت؛ در رمزنگاری نامتقارن، رمزگذاری با یک کلید و رمزگشایی با کلید دیگری انجام می‌پذیرد.

الگوریتم‌های کلیدعمومی به‌جای Permutation و Substitution، بر اساس توابع ریاضی ایجاد شده‌اند. همچنین استفاده از دو کلید مجزا برای رمزگذاری و رمزگشایی، باعث نتایج بهتری درزمینه محرمانگی، احرازهویت و تبادل کلید گردیده است.

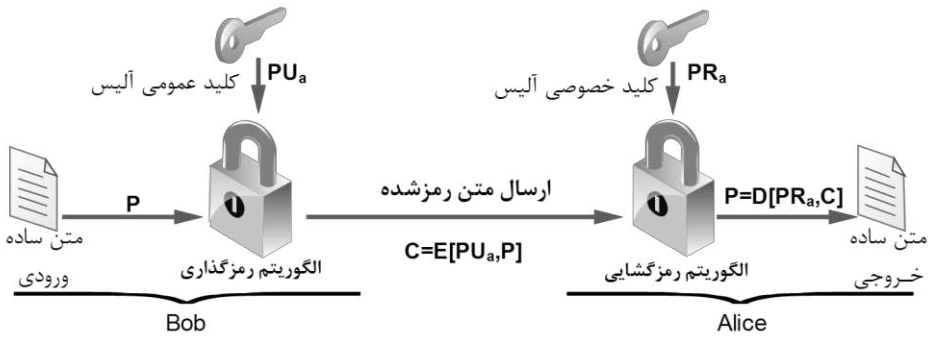
در مقایسه رمزنگاری نامتقارن یا کلیدعمومی با رمزنگاری متقارن، تصورات نادرستی وجود دارد که باید به آن‌ها پاسخ داده شود. اول آنکه تصور عمومی بر آن است که رمزنگاری کلیدعمومی از رمزنگاری متقارن امن‌تر است. ولی در اصل هیچ دلیلی وجود ندارد که بتوان یکی را از دیگری امن‌تر و در مقابل حملات مقاوم‌تر دانست؛ بلکه قدرت هر الگوریتم رمزنگاری به طول کلید و محاسبات انجام‌شده است.

تصور غلط دوم این است که باوجود رمزنگاری نامتقارن، رمزنگاری متقارن منسوخ‌شده یا به‌زودی منسوخ خواهد شد. اما به دلیل سربار محاسباتی کلیدعمومی، هیچ احتمال قابل پیش‌بینی برای کنار گذاشتن رمزنگاری متقارن وجود ندارد.

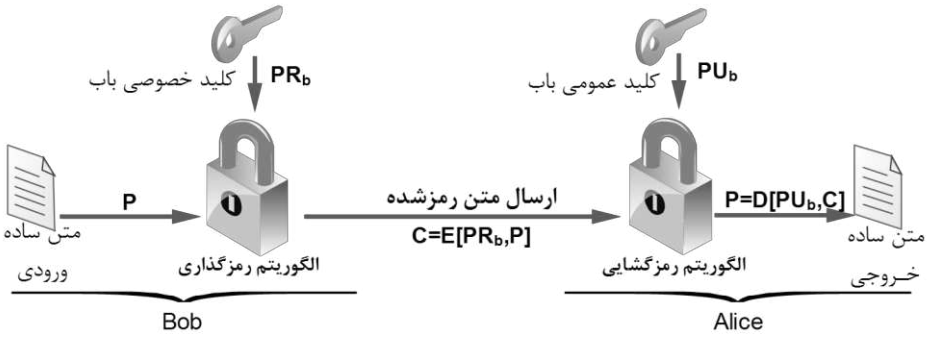
در رمزنگاری نامتقارن، رمزگذاری هم می‌تواند توسط کلید عمومی انجام شود و هم توسط کلید خصوصی. برای محرمانگی، رمزگذاری باید با کلیدعمومی گیرنده و برای احراز هویت، رمزگذاری باید با کلید خصوصی فرستنده انجام گیرد.

در الگوریتم رمز نامتقارن از هر دو کلیدخصوصی و عمومی، یکی برای رمزگذاری و دیگری برای رمزگشایی استفاده می‌شود؛ پس بالطبع هر دو کلید خصوصی و عمومی می‌توانند هم در رمزگذاری و هم در رمزگشایی مورداستفاده قرار گیرند. تصویر زیر نحوه استفاده از کلیدخصوصی و کلیدعمومی در رمزنگاری نامتقارن را به‌صورت ساده نمایش می‌دهد.

رمزگذاری با استفاده از کلید عمومی



رمزگذاری با استفاده از کلید خصوصی



رمزنگاری نامتقارن برای رمزگذاری و رمزگشایی از دو کلید متفاوت استفاده می‌نماید؛ به‌عبارت‌دیگر امکان رمزگشایی با همان کلید مورداستفاده در رمزگذاری، وجود ندارد. درصورتی‌که رمزگذاری با کلید عمومی گیرنده انجام‌شده باشد، برای رمزگشایی حتماً باید از کلید خصوصی گیرنده استفاده شود؛ و اگر برای رمزگذاری از کلید خصوصی فرستنده استفاده شده باشد، فقط با کلید عمومی فرستنده قابل رمزگشایی خواهد بود.

کلید خصوصی و کلیدعمومی همزمان برای یک شخص ایجاد می‌گردد؛ به‌طوری‌که این دو کلید مستقل ولی مرتبط با یکدیگر می‌باشند. کلید عمومی ایجادشده به‌صورت عمومی در اختیار همه طرف‌های ارتباط قرار می‌گیرد. اما کلید خصوصی فقط باید نزد صاحب کلید و به‌صورت کاملاً محرمانه نگهداری و محافظت گردد.

ویژگی‌های الگوریتم رمزنگاری نامتقارن

الگوریتم رمزنگاری نامتقارن دارای ویژگی‌های زیر است:

- حتی در صورت اطلاع از الگوریتم رمزنگاری و کلید رمزگذاری مورداستفاده، امکان محاسبه کلید رمزگشایی وجود ندارد.
- هریک از دو کلید عمومی و خصوصی می‌توانند برای رمزگذاری و کلید دیگر برای رمزگشایی مورداستفاده قرار گیرد.
- درصورتی‌که پیام با کلید عمومی گیرنده رمز شود، فراهم آورنده محرمانگی و اگر با کلید خصوصی فرستنده رمز شود، فراهم آورنده احرازهویت است.
- هر کاربر می‌تواند یک دسته‌کلید از کلیدهای عمومی کاربران مختلف را در اختیار داشته باشد؛ چراکه کلید عمومی برای هر درخواست‌کننده‌ای ارسال می‌شود.
- اگر پیامی با کلید عمومی یک کاربر (مثلاً آلیس) رمز شده باشد، رمزگشایی آن فقط برای آلیس امکان‌پذیر است؛ به دلیل اینکه شخص دیگری از کلید خصوصی آلیس اطلاع ندارد و البته نباید داشته باشد.
- اگر پیامی فقط با کلید خصوصی فرستنده رمز شده باشد، رمزگشایی آن برای هر کاربری که از کلید عمومی فرستنده اطلاع داشته باشد، امکان‌پذیر خواهد بود.
- هرزمانی که کاربر احساس نیاز کند، می‌تواند برای خود یک جفت کلید خصوصی و عمومی جدید ایجاد نماید. کلید خصوصی را به‌صورت امن نزد خود نگه‌داشته و کلید عمومی را نیز در یک محل عمومی برای دسترسی درخواست‌کنندگان قرار دهد.
- در هر فرآیند رمزگذاری و رمزگشایی، باید از یک جفت کلید مربوط به هم استفاده شود. مثلاً نمی‌توان پیام را با کلید خصوصی باب رمزگذاری کرد ولی با کلید عمومی آلیس رمزگشایی نمود!
- کلیدهای عمومی به‌طورمعمول در یک مکان عمومی قرار می‌گیرند تا هر کاربری که نیاز داشته باشد به‌راحتی بتواند به آن‌ها دسترسی پیدا کند.
- تبادل کلیدهای عمومی به‌راحتی می‌تواند بر روی بستر ارتباطی ناامن انجام پذیرد.
- پیام‌های رمزشده در رمزنگاری نامتقارن، بر روی بستر ارتباطی عمومی و ناامن انتقال می‌یابند.
- لغو[1] کلید خصوصی و عمومی توسط مالک کلید، امکان‌پذیر است.
- در صورت افشا شدن کلید خصوصی، باید سریعاً زوج کلید مربوطه لغو شوند.

[1] Revoke

مقایسه رمزنگاری متقارن با رمزنگاری نامتقارن

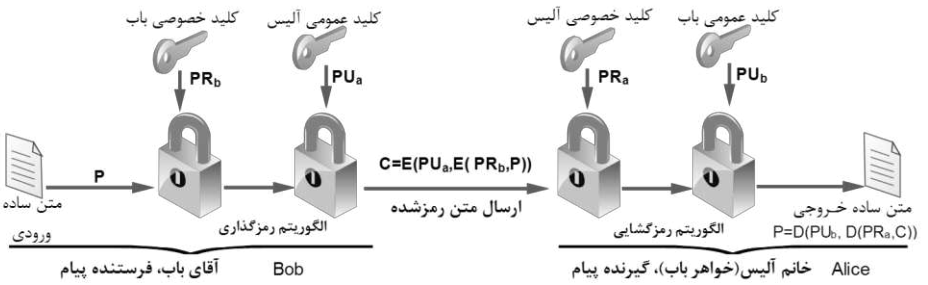
جدول زیر شامل مقایسه بین الگوریتم‌های رمزنگاری متقارن و الگوریتم‌های رمزنگاری نامتقارن است:

	رمزنگاری متقارن	رمزنگاری نامتقارن
نیازهای عملیاتی	۱. برای رمزگذاری و رمزگشایی از یک کلید و الگوریتم مشابه استفاده می‌شود. ۲. فرستنده و گیرنده باید الگوریتم و کلید مورداستفاده را باهم به اشتراک‌گذارند. ۳. برای اشتراک‌گذاری کلید، نیاز به یک کانال انتقال امن است.	۱. برای رمزگذاری و رمزگشایی از یک الگوریتم مشابه ولی دو کلید متفاوت استفاده می‌شود؛ یک کلید برای رمزگذاری و یک کلید برای رمزگشایی. ۲. فرستنده و گیرنده هرکدام باید یکی از کلیدها را در اختیار داشته باشند. (نیازی به یکسان بودن کلیدها نیست) ۳. کلید عمومی را می‌توان از طریق کانال ناامن انتقال داد.
نیازهای امنیتی	۱. کلید باید به‌صورت امن نگهداری شود. ۲. کشف پیام بدون داشتن سایر اطلاعات امکان‌پذیر نباشد. ۳. دانستن الگوریتم به‌علاوه داشتن چند نمونه متن رمزشده نباید منجر به کشف کلید شود.	۱. فقط یکی از دو کلید (کلید خصوصی) باید به‌صورت امن نگهداری شود. ۲. کشف پیام بدون داشتن سایر اطلاعات امکان‌پذیر نباشد. ۳. دانستن الگوریتم به‌علاوه یکی از کلیدها به همراه داشتن چند نمونه متن رمزشده، نباید منجر به کشف کلید شود.

محرمانگی و احرازهویت

همان‌طور که گفته شد، درصورتی‌که فرستنده، پیام را با کلید خصوصی خود رمز نماید، فراهم‌کننده احراز هویت و درصورتی‌که پیام را با کلید عمومی گیرنده رمز نماید، فراهم آورنده محرمانگی است. اما درصورتی‌که پیام صرفاً با کلید خصوصی فرستنده رمز شده باشد، هر کاربری که کلید عمومی فرستنده را در اختیار داشته باشد، قادر به رمزگشایی پیام خواهد بود.

اگر بخواهیم که احراز هویت و محرمانگی را همزمان باهم در اختیار داشته باشیم، باید فرستنده پس از رمزگذاری پیام با کلید خصوصی خود، مجدداً پیام را با کلید عمومی گیرنده نیز رمز نماید. در این صورت هیچ کاربری غیر از گیرنده موردنظر نمی‌تواند رمزگشایی مرحله اول که نیاز به کلید خصوصی گیرنده دارد را انجام دهد. اما گیرنده پس‌ازآنکه در مرحله اول پیام را با کلید خصوصی خود رمزگشایی نمود، در مرحله دوم رمزگشایی را با کلید عمومی فرستنده انجام می‌دهد. این عمل که در تصویر زیر نمایش داده‌شده است، باعث می‌گردد همزمان محرمانگی و احراز هویت را در اختیار داشته باشیم.



فرآیند احراز هویت در تصویر بالا به این صورت است که کلید خصوصی را هیچ‌کس جز مالک کلید در اختیار ندارد، از طرف دیگر وقتی رمزگذاری با کلید خصوصی انجام می‌شود، عملیات رمزگشایی فقط با کلید عمومی امکان‌پذیر است. پس وقتی آلیس می‌تواند با کلید عمومی باب، متن را رمزگشایی نماید، مطمئن می‌شود که فرستنده پیام نمی‌تواند کسی جز باب باشد.

کاربردهای سیستم رمزنگاری کلیدعمومی

بسته به کاربرد موردنظر، فرستنده می‌تواند از کلید خصوصی خود، کلید عمومی گیرنده و یا هر دو آن‌ها برای رمزگذاری استفاده نماید. اگر بخواهیم کاربردهای رمزنگاری نامتقارن را تقسیم‌بندی کنیم، سه گروه به‌صورت زیر خواهیم داشت:

- **رمزگذاری / رمزگشایی**

ارسال‌کننده، پیام را با کلید عمومی گیرنده رمزگذاری می‌نماید.

- **امضای دیجیتال**[1]

فرستنده، پیام را با کلیدخصوصی خود امضاء می‌نماید. معمولاً امضاء با اعمال الگوریتم رمزگذاری با کلیدخصوصی بر یک بلوک کوچک از دیتا انجام می‌پذیرد.

[1] Digital Signature

- تبادل کلید[1]

دو طرف باید کلید جلسه[2] را با یکدیگر تبادل نمایند.

بعضی از الگوریتم‌های رمزنگاری هر سه کاربرد فوق را می‌توانند فراهم آورند؛ و البته بعضی دیگر فقط می‌توانند یک یا دو کاربرد از موارد ذکرشده را پشتیبانی نمایند. جدول زیر نمایش‌دهنده این موضوع است:

الگوریتم	رمزگذاری/رمزگشایی	امضای دیجیتال	تبادل کلید
RSA	بلی	بلی	بلی
Elliptic Curve	بلی	بلی	بلی
Diffie-Hellman	خیر	خیر	بلی
DSS	خیر	بلی	خیر

الزامات رمزنگاری نامتقارن[3]

برای داشتن یک سیستم رمزنگاری نامتقارن، الزامات زیر باید فراهم باشد:

۱. محاسبات تولید یک جفت کلید (کلید عمومی و خصوصی) برای طرف B آسان باشد.
۲. برای فرستنده (A)، محاسبه متن رمزشده متناظر با کلید عمومی گیرنده و پیام (M)، آسان باشد. $C = \mathrm{E}(PUb, M)$
۳. برای دریافت‌کننده(B) محاسبه رمزگشایی(D) متن رمزشده(C) بر اساس کلید خصوصی خودش، آسان باشد.

$$M = \mathrm{D}(PRb, C) = \mathrm{D}[PRb, \mathrm{E}(PUb, M)]$$

۴. برای مهاجم، محاسبه کلیدخصوصی گیرنده (PR_b)، حتی با در اختیار داشتن کلید عمومی گیرنده(PU_b)، امکان‌پذیر نباشد.
۵. به دست آوردن پیام اصلی، حتی با داشتن متن رمزشده و کلیدعمومی گیرنده، برای مهاجم امکان‌پذیر نباشد.
۶. از هر دو کلید بتوان هم برای رمزگذاری و هم برای رمزگشایی استفاده نمود.

$$M = \mathrm{D}[PUb, \mathrm{E}(PRb, M)] = \mathrm{D}[PRb, \mathrm{E}(PUb, M)]$$

---

[1] Key Exchange
[2] Session Key
[3] همان‌طور که گفتم، برای رمزنگاری نامتقارن از عبارت کلیدعمومی نیز استفاده می‌شود. من در این کتاب نسبت به جمله جاری، به‌صورت متناوب از هر دو عبارت فوق استفاده کرده‌ام.

☑ مبحث چهارم

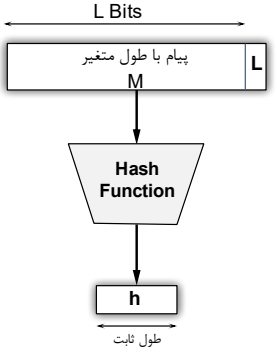
مفاهیم توابع درهم‌سازی

توابع درهم‌سازی یا Hash Functions، توابع ریاضی هستند که در ازای بلوک‌های مختلف ورودی، همواره یک خروجی با طول ثابت را تولید می‌نمایند. به‌عبارت‌دیگر، تابع درهم‌سازی یک بلوک دیتا با طول متغیر را به‌عنوان ورودی(M) قبول کرده و یک خروجی با اندازه ثابت را تولید می‌نماید: $h = H(M)$.

خروجی این توابع به ازای هر ورودی، یکتا و منحصربه‌فرد است؛ به‌طوری‌که یک تغییر کوچک در یک بیت ورودی، خروجی تابع را به‌طورکلی تحت تأثیر قرار می‌دهد. به همین دلیل این خروجی‌ها به‌عنوان اثرانگشت دیتای ورودی نیز شناخته می‌شود.

برخلاف رمزنگاری‌های متقارن و نامتقارن، که هم برای رمزگذاری و هم برای رمزگشایی مورداستفاده قرار می‌گیرند؛ از توابع درهم‌سازی فقط برای رمزگذاری استفاده‌شده و امکان رمزگشایی خروجی این توابع ممکن نیست. کاربرد اصلی توابع درهم‌ساز برای بررسی صحت و درستی (Integrity) داده است. از طریق مقدار Hash می‌توان تشخیص داد که پیام در طول مسیر دچار تغییر شده است یا نه! البته این تغییر هم ممکن است توسط هکر انجام‌شده باشد و هم ممکن است به دلایل طبیعی مثل نویز خط مخابراتی ایجادشده باشد. به‌هرحال با مقایسه Hash پیام دریافتی با Hash اعلام‌شده توسط فرستنده، می‌توان از صحت و تمامیت دیتای دریافت شده، اطمینان حاصل نمود. از توابع درهم‌ساز علاوه بر صحت و یکپارچگی، برای مواردی ازجمله امضای دیجیتال، نگهداری کلمات عبور و احراز هویت نیز استفاده می‌شود.

تصویر زیر عملیات توابع درهم‌سازی را به‌صورت ساده نمایش می‌دهد:



اصطلاحات توابع درهم‌سازی

در توابع درهم سازی دارای اصطلاحات خاصی هستیم که در ادامه شرح داده می‌شوند:

۱. **Preimage**

به متن ساده اولیه که قرار است تابع درهم‌سازی بر روی آن اعمال شود Preimage[۱] گفته می‌شود؛ و در دیاگرام معمولاً با حرف M از پیام یا همان Preimage یاد می‌شود. به‌عبارت‌دیگر اگر عملیات درهم‌سازی را طبق فرمول $h = \mathrm{H}(x)$ نمایش دهیم؛ در آن x به‌منزله Preimage مقدار Hash شده یا همان h است.

۲. **نام‌گذاری خروجی تابع درهم‌ساز**

برای خروجی تابع Hash، نسبت به مورداستفاده از عباراتی مثل Message Digest، اثرانگشت، خلاصه پیام و چکیده پیام استفاده می‌شود؛ که در دیاگرام آن را با حرف h نمایش می‌دهند.

۳. **Message Digest**

معمولاً زمانی به خروجی تابع Hash، عبارت Message Digest اطلاق می‌شود که از تابع برای تصدیق پیام استفاده شده باشد.

۴. **تصادم (Collision)**

اگر دارای دو پیام x و y باشیم، به‌طوری‌که $x \neq y$، ولی خروجی تابع درهم‌ساز برای هر دو مقدار یکسان باشد: $h_x = h_y$، در آن صورت می‌گوییم که تصادم یا Collision روی‌داده است.

۵. **تابع درهم‌سازی**

در فرمول‌ها و دیاگرام‌ها، توابع درهم‌سازی یا همان Hash Functions، با حرف H نمایش داده می‌شود.

خواص توابع درهم‌سازی

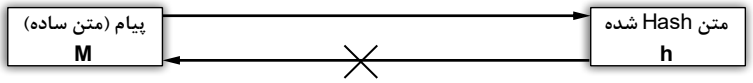
توابع درهم‌سازی باید دارای دو خاصیت اصلی زیر باشند. در صورت نقض هر یک از این خواص، می‌گوییم آن تابع درهم‌ساز شکسته شده است.

۱. **خاصیت یک‌طرفه بودن (One-Way)**

از طریق پیام Hash شده، نمی‌توان به پیام اصلی (متن‌ساده) دست پیدا کرد. به‌عبارت‌دیگر هیچ راه رمزگشایی برای متن Hash شده وجود ندارد و البته نباید داشته باشد.

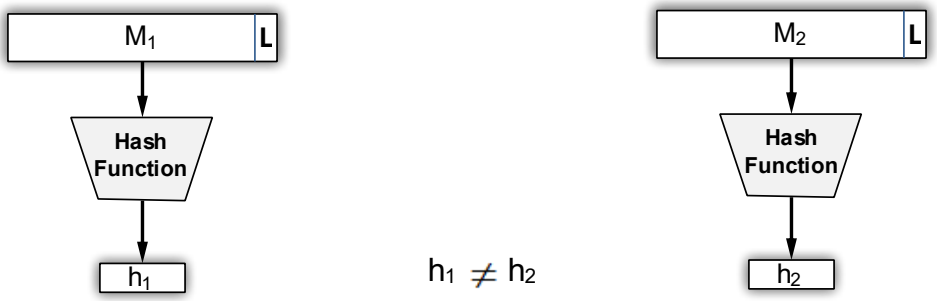
---

[۱] نمونه اولیه



۲. **بدون تصادم (Collision-Free)**

خروجیHash شده (چکیده پیام) دو پیام متفاوت توسط یک تابع درهم‌ساز، نباید شبیه یکدیگر باشند.

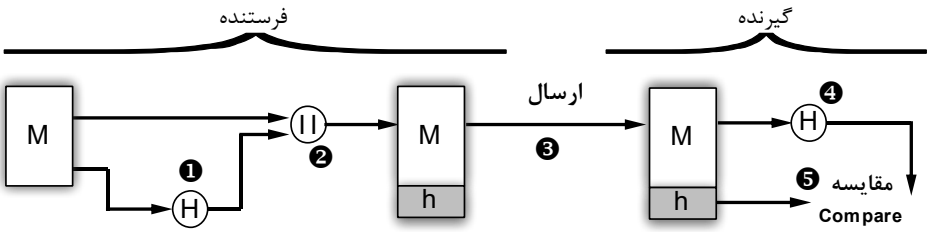


کاربرد Hash در بررسی صحت پیام

همان‌طور که گفته شد، کاربرد اصلی توابع درهم‌سازی برای بررسی صحت و تمامیت دیتای دریافت شده است. برای این منظور فرستنده چکیده پیام را برای گیرنده ارسال می‌نماید، گیرنده نیز از پیام دریافت شده Hash گرفته و آن را با مقدار Hash اعلام‌شده از طرف فرستنده مقایسه می‌نماید. درصورتی‌که مقادیر Hash باهم یکسان باشند، دریافت‌کننده اطمینان می‌یابد که دیتای دریافت شده بدون هیچ تغییر و یا نقصی تبادل گردیده است.

کاربرد Hash در بررسی صحت دیتا، به شکل ساده در تصویر زیر نمایش داده‌شده است:

فرستنده از پیام موردنظر Hash گرفته❶ و آن را به متن اصلی پیام الصاق می‌نماید.❷ سپس پیام را به همراه چکیده آن (اثرانگشت) برای گیرنده ارسال می‌کند.❸



گیرنده پس از دریافت پیام و اثرانگشت، باید اقدام به بررسی آن نماید. همان‌طور که گفته شد، نمی‌توان از مقدار Hash به متن اصلی پیام رسید؛ لذا گیرنده باید مجدداً از پیام، Hash گرفته ❹و سپس خروجی تابع درهم‌سازی را با مقدار Hash ارسال‌شده از طرف فرستنده مقایسه نماید. ❺ درصورتی‌که این دو مقدار با یکدیگر مساوی باشند، گیرنده می‌تواند از صحت و تمامیت پیام دریافتی اطمینان حاصل نماید.

کاربرد توابع درهم‌سازی

توابع درهم‌ساز به‌طور مستقل در صحت و تمامیت داده کاربرد دارند. اما این توابع به همراه الگوریتم‌های رمزنگاری دیگر نیز دارای کاربردهای خاصی دیگری می‌باشند.

۱. **صحت و تمامیت (Integrity)**

همان‌طور که تشریح شد، از توابع درهم‌سازی می‌توان برای تولید اثرانگشت استفاده کرد و از اثرانگشت نیز می‌توان برای بررسی صحت و تمامیت دیتای دریافت شده، بهره برد. در مثالی که برای صحت آورده شده، پیام و اثرانگشت از طریق کانال ناامن برای گیرنده ارسال‌شده است. شاید برای شما سؤال پیش‌آمده باشد که فرد هکر می‌تواند پیام و مقدار Hash را به‌راحتی با پیام و Hash موردنظر خود جابه‌جا نماید. البته این ایراد به طرح فوق وارد است؛ اما کاربرد آن طرح نیز متفاوت است. اگر دیتای ارسالی ما شامل اطلاعات مهمی نباشد و تنها ترس ما نویز و اشکالات خطوط مخابراتی است، می‌توانیم به‌راحتی از همان طرح استفاده نماییم. البته برخی از وب سایت‌ها نیز مقدار Hash فایل‌های قابل دانلود را در کنار آن‌ها بر روی سایت قرار داده‌اند. مثلاً وقتی‌که می‌خواهید یک IOS از وب‌سایت سیسکو دانلود نمایید، در کنار فایل مقدار Hash آن نیز قرار داده‌شده است. شما می‌توانید پس از دانلود IOS از آن Hash گرفته و با مقدار ذکرشده در سایت مقایسه نمایید. اما اگر پیام حاوی اطلاعات مهمی است که ممکن است توسط هکر در بین راه تغییر یابد، می‌توان از الگوریتم‌های رمزنگاری دیگر نیز برای محرمانگی استفاده کرد. استفاده از الگوریتم‌های رمزنگاری در کنار توابع درهم‌سازی (تصدیق پیام) در فصل ششم تشریح شده است.

۲. **نگهداری کلمات عبور**

برای نگهداری کلمات عبور، از توابع درهم‌سازی استفاده می‌شود. در تمام تجهیزات و برنامه‌های کاربردی که نیاز به احراز هویت دارند، نام‌های کاربری و کلمات عبور در قالب یک فایل نگهداری می‌شوند. حال تصور کنید که مهاجم بتواند از طریقی به این فایل دسترسی پیدا کند؛ مطمئناً می‌توانید فجایع بعدی را متصور شوید. اما اگر کلمات

عبور به‌صورت Hashشده در داخل آن فایل نگهداری شده باشند، حتی در صورت دسترسی مهاجم به فایل، امکان افشای آن‌ها وجود نداشته و یا حداقل برای مدت‌زمان قابل‌توجهی به تعویق خواهد افتاد.

البته لازم به ذکر است که طبق بهترین شیوه، حتی مدیر سیستم هم نباید از کلمات عبور سایر کاربران اطلاع داشته باشد؛ چراکه اتفاقات روی‌داده در سیستم بر اساس نام کاربر واقعه‌نگاری شده و بالطبع کاربر موردنظر باید پاسخگوی عمل انجام‌شده باشد.

۳. **تصدیق پیام (Message Authentication)**

برای حصول اطمینان از این‌که پیام در مسیر تبادل دچار هیچ‌گونه تغییری نشده است، می‌توان از اثرانگشت در کنار سایر الگوریتم‌های رمزنگاری بهره برد.

احرازهویت پیام به‌صورت کامل در فصل ششم بررسی‌شده است.

۴. **امضای دیجیتال (Digital Signature)**

یکی از کاربردهای مهم توابع درهم‌سازی، استفاده برای امضای دیجیتال است. درصورتی‌که از توابع درهم‌سازی در کنار رمزنگاری نامتقارن استفاده نماییم، قادر به بهره‌برداری از امضای دیجیتال خواهیم بود.

امضای دیجیتال در فصل ششم بررسی‌شده است.

۵. **کاربرد در نرم‌افزارهای امنیتی**

نرم‌افزارهای امنیتی مثل آنتی‌ویروس، برای مقابله با ویروس‌های شناخته‌شده دارای یک دیتابیس بانام Virus Signature هستند. آنتی‌ویروس با مقایسه اثرانگشت فایل‌های موجود بر روی سیستم با اثرانگشت‌های موجود در دیتابیس خود، قادر به شناسایی ویروس‌ها و کدهای مخرب است.[۱] در نرم‌افزارها و تجهیزات IDS/IPS[۲] نیز همانند آنتی‌ویروس، از اثرانگشت فایل‌ها بهره‌برداری می‌شود.

یکی دیگر از روش‌های بهره‌برداری از توابع درهم‌سازی توسط آنتی‌ویروس‌ها، جمع‌آوری یک دیتابیس از اثرانگشت مربوط به فایل‌های مهم موجود بر روی سیستم است. در این صورت اگر یک نرم‌افزار مخرب اقدام به تغییر این فایل‌ها نماید، اثرانگشت مربوط به فایل تغییر کرده و آنتی‌ویروس می‌تواند تشخیص دهد که حمله‌ای بر روی سیستم صورت پذیرفته است.

---

[۱] آنتی‌ویروس فقط برای پیدا کردن ویروس‌هایی که قبلاً شناخته‌شده‌اند و اثرانگشت آن‌ها در دیتابیس موجود است، از این مکانیسم استفاده می‌کند. اما برای شناسایی ویروس‌ها و حملات جدید دارای مکانیسم‌های دیگری ازجمله هوش مصنوعی است.

[2] Intrusion Detection System / Intrusion Prevention System

امنیت توابع درهم‌سازی

همان‌طور که گفته شد، یکی از موارد استفاده از توابع درهم‌سازی، نگهداری کلمات عبور و در برخی مواقع تبادل کلمه‌عبور است. در این صورت اگر مهاجم بتواند به فایل حاوی کلمات عبور دسترسی پیدا کند و یا کلمه عبور Hash شده را در هنگام تبادل به دست آورد، برای شکستن آن و کشف کلمه‌عبور اقدام خواهد کرد.

ازآنجاکه نباید هیچ الگوریتم و راهی برای تبدیل چکیده به متن اصلی وجود داشته باشد، لذا تنها حمله‌ای که می‌تواند این نوع کاربردهای توابع درهم‌سازی را تهدید نماید، حملات جستجوی فراگیر یا Brute Force است.

در این نوع حمله، هکر تمامی عبارات ممکن را ایجاد کرده و از آن‌ها Hash می‌گیرد، سپس مقدار به‌دست‌آمده را با Hash دریافتی مقایسه می‌نماید. هکر باید این کار را تا پیدا کردن مقدار Hash مشابه با Hash دریافتی انجام دهد.

بعضی از ابزارها و وب‌سایت‌ها، دیتابیس‌های بزرگی از Hash مربوط به کلمات و عبارات مختلف آماده‌سازی کرده‌اند که به‌صورت رایگان یا در ازای مبلغ اندکی، مقدار Hash موردنظر شما را با دیتابیس خود مقایسه کرده و در صورت موجود بودن، عبارت اصلی را به شما تحویل می‌دهند. البته این همان حمله جستجوی‌فراگیر است که با آماده‌سازی دیتابیس در وقت و هزینه صرفه‌جویی کرده است.
لازم به ذکر است که اکثر این دیتابیس‌ها برای کلمات و جملات انگلیسی تهیه‌شده است و اگر پیام شما به فارسی یا فینگلیش باشد، از کارایی آن‌ها به‌شدت کاسته خواهد شد.



یکی دیگر از خطرات پیش روی توابع درهم‌سازی این است که هکر بتواند پیامی را جایگزین پیام اصلی نماید؛ به‌طوری‌که چکیده آن با چکیده پیام اصلی برابر باشد. در این صورت گفته می‌شود که الگوریتم موردنظر شکسته شده است.

برای مقابله با تهدیدات فوق، بهتر است که توابع درهم‌سازی دارای خروجی‌های بزرگی باشند. هر چه طول خروجی تابع درهم‌ساز بزرگ‌تر باشد، آن تابع در برابر تهدیدات گفته‌شده مقاوم‌تر خواهد بود.

به‌عنوان‌مثال الگوریتم MD5 دارای خروجی ۱۲۸ بیت، الگوریتم SHA-1 دارای خروجی ۱۶۰ بیت و الگوریتم SHA-512 دارای خروجی ۵۱۲ بیت است. یادآوری می‌کنم که خروجی توابع درهم‌ساز همواره یکسان است؛ مثلاً الگوریتم MD5 چه ورودی ۱۰ بیتی دریافت نماید و چه ورودی ۱۰۰ بیتی، همواره خروجی ۱۲۸ بیتی ارائه خواهد داد.

البته توابع درهم‌سازی نیز همانند هر الگوریتم رمزنگاری دیگر مورد تجزیه‌وتحلیل هکرها قرار می‌گیرند تا نقاط ضعف آن را استخراج کرده و یا از ویژگی‌های آن سوءاستفاده نمایند.

مقایسه توابع درهم‌سازی

توابع درهم‌سازی دارای الگوریتم‌های متعددی است که برخی از آن‌ها منسوخ‌شده، برخی اصلاً به‌صورت استاندارد معرفی نشده و برخی دیگر هم‌اکنون نیز در حال استفاده هستند.

جهت آشنایی با مشهورترین این الگوریتم‌ها، جدول زیر را ملاحظه فرمایید.

نام تابع	طول خروجی	تعداد دور	سال ارائه	طراح	استاندارد
HAVAL	128 To 256	96 To 160	1992	Yuliang Zheng, Josef Pieprzyk, Jennifer Seberry	-
MD2	128	864	1989	Ronald Linn Rivest	RFC 1319
MD4		48	1990		RFC 1320
MD5		64	1992		RFC 1321
MD6	256	Variable	2008		-
SHA-1	160	80	1995	آژانس امنیت ملی امریکا NSA	FIPS 180-1
SHA-256	256	64	2002		FIPS 180-2
SHA-512	512	64			
SHA-384	384	80			
SHA-3	512	24	2014	Guido Bertoni, Joan Daemen, Michaël Peeters, Gilles Van Assch	Draft FIPS 202

فصل چهارم

الگوریتم‌های رمزنگاری متقارن

مبحث اول

الگوریتم DES

مبحث دوم

الگوریتم AES

مبحث سوم

الگوریتم Triple DES

مبحث چهارم

روش‌های عملیاتی رمزنگاری بلوکی

☑ مبحث اول

الگوریتم DES

الگوریتم Data Encryption Standard(DES)، یک الگوریتم رمزنگاری متقارن است و در گروه رمزنگاری مدرن قرار می‌گیرد. الگوریتم DES عملیات رمزنگاری خود را به‌صورت بلوکی و بر اساس کلید محرمانه انجام می‌دهد.

الگوریتم DES در دهه 1970 میلادی در امریکا مطرح شد و تا سال 1999 به‌عنوان استاندارد مورداستفاده قرار می‌گرفت. در سال 1999 الگوریتم رمزنگاری DES شکسته شد و به همین دلیل جای خود را به یک الگوریتم قدرتمندتر واگذار نمود.

هرچند که امروزه از این الگوریتم کمتر استفاده می‌شود، ولی به دلیل اینکه فرآیند عملکرد این الگوریتم مبنایی برای بعضی استانداردهای رمزنگاری دیگر است، لذا در این مبحث به شرح آن می‌پردازیم.

الگوریتم DES نام استاندارد برای الگوریتم Lucifer است که برمبنای الگوریتم فایستل توسعه داده‌شده است. این الگوریتم یک بلوک مشخصی از دیتا را گرفته و با استفاده از کلید، آن را رمزگذاری می‌نماید. خروجی الگوریتم نیز یک بلوک دیتا به‌اندازه بلوک ورودی خواهد بود.

در این مبحث ابتدا به تشریح الگوریتم فایستل و سپس به شرح الگوریتم DES می‌پردازیم.

الگوریتم فایستل

فایستل یک الگوریتم رمزنگاری بلوکی است که توسط آقای Horst Feistel ارائه گردیده است. این الگوریتم بعدها مبنای ایجاد بسیاری از الگوریتم‌های رمزنگاری متقارن قرار گرفت.

به‌طور خاص فایستل پیشنهاد رمز کردن بر اساس جایگزینی و جابجایی متناوب در متن اصلی را مطرح نموده است. درواقع فایستل با ارائه این دو روش خواسته که برای شروط شانون، یک راهکار عملی ارائه نماید.

۱. **جایگزینی (Substitution)**

هر عنصر یا گروهی از عناصر متن‌ساده، منحصراً با عنصر یا گروهی از عناصر متناظر در متن رمزشده، جایگزین شوند.

۲. **جایگشت (Permutation)**

طی فرآیند جایگشت، رشته‌ای[1] از عناصر متن ساده، با یکدیگر جابجا می‌شوند. در این جابجایی هیچ‌یک از عناصر داخلی موجود در آن رشته نباید اضافه، حذف و یا جابجا شوند؛ بلکه قصد این است که توالی خود رشته از بین برود و نه عناصر موجود در آن.

رمزگذاری و رمزگشایی در فایستل

فایستل یک الگوریتم Block Cipher است، لذا دیتای ورودی این الگوریتم نیز باید به‌صورت بلوکی باشد. پس باید در ورودی یک بلوک از دیتای مربوط به متن ساده همراه با کلید موردنظر قرار گیرد. در ابتدا فایستل بلوک دریافتی را به دو قسمت مساوی راست (R) و چپ (L) تقسیم می‌نماید. ❶ سپس هر دونیمه را تحویل دور (Round) اول می‌دهد. ❷ با توجه به تعدد دورها، خروجی هر دور به‌عنوان ورودی دور بعد خواهد بود. در هر دور نیاز به یک کلید مخصوص هم داریم که این کلید از کلید اصلی مشتق گردیده است. ❸

در الگوریتم فایستل تعیین تعداد دورها (Round) بر عهده استفاده‌کننده است. فقط باید توجه داشته باشید که کلیدهای محرمانه هر دور باید متفاوت و برگرفته از کلید اصلی باشند.

ساختار تمام‌دورها شبیه به هم است؛ یعنی عملیاتی که در هر دور اتفاق می‌افتد یکسان است. در هر دور، عمل جایگزینی (Substitution) بر روی نیمه چپ دیتا اتفاق می‌افتد؛ به همین دلیل یک کپی از نیمه سمت راست به‌عنوان نیمه سمت چپ دیتا تحویل دور بعدی می‌گردد. ❹ از طرف دیگر، پس‌ازآنکه تابع F بر اساس کلید بر روی نیمه راست دیتا اعمال شد، خروجی تابع با نیمه چپ دیتا XOR گردیده و به‌عنوان نیمه راست دیتا به دور بعد تحویل داده می‌شود. ❺

دور بعد دونیمه راست و چپ را تحویل گرفته و فارغ از عملیات انجام‌شده بر روی دیتا، همان مراحل دور قبل را البته با کلید جدید، اعمال می‌نماید. ❻

اما پس از انجام دور پایانی، نیمه راست و چپ به‌دست‌آمده بدون انجام هیچ کار خاصی بر روی دیتا، با یکدیگر جابجا می‌شوند. ❼ این جابجایی پس از آخرین مرحله جهت ویژگی Permutation یا جایگشت می‌باشد که بر اساس پیشنهاد شانون انجام‌شده است.

همان‌طور که قبلاً گفته شد، عملیات رمزگشایی دقیقاً عکس عملیات رمزگذاری اتفاق می‌افتد. در الگوریتم فایستل نیز رمزگشایی به تعداد دورهای انجام‌شده در عملیات رمزگذاری تکرار می‌شود؛ اما این بار به‌صورت معکوس. ❽ توجه داشته باشید که ترتیب کلیدها در رمزگشایی نیز باید به‌صورت عکس رمزگذاری (یعنی آخر به اول)، مورداستفاده قرار گیرد. ❾

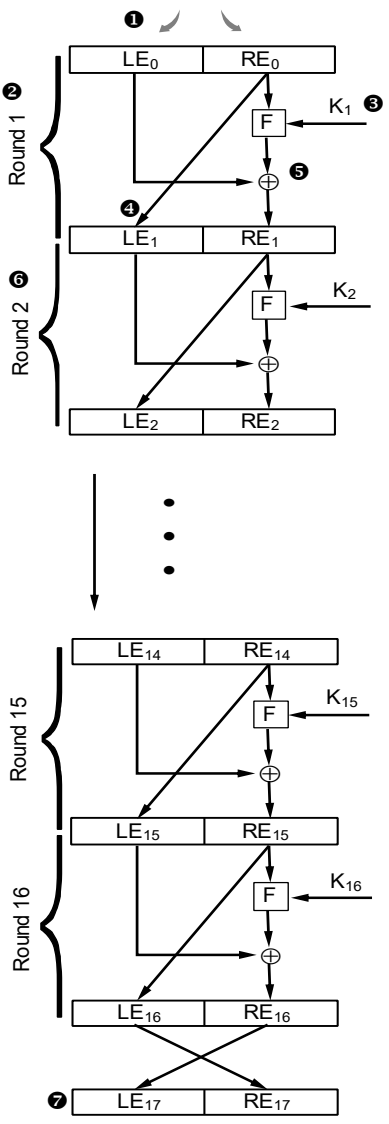
برای فهم بهتر الگوریتم فایستل، مراحل رمزگذاری و رمزگشایی که بر اساس ۱۶ دور ترسیم گردیده را در صفحه بعد ملاحظه بفرمائید.

[1] Sequence

مراحـل رمزگذاری فایستل
Encryption

ورودی (متن ساده)
Input

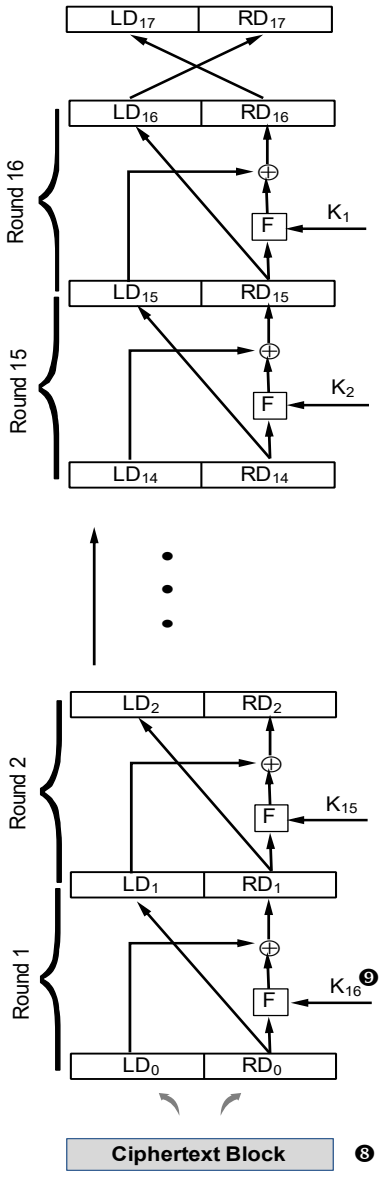
Plaintext Block



خـروجی (متن رمزشده)
Ciphertext

مراحـل رمزگشایی فایستل
Decryption

خروجی (متن ساده)
Output



Ciphertext Block ❽

ورودی (متن رمزشده)
Input

پارامترها و ویژگی‌های طراحی فایستل

تحقق دقیق الگوریتم فایستل به انتخاب پارامترها و ویژگی‌های طراحی زیر، بستگی دارد:

۱. **اندازه بلوک**

هرچه بلوک بزرگ‌تر باشد، امنیت بیشتر خواهد شد؛ اما بلوک بزرگ‌تر باعث کاهش سرعت رمزگذاری و رمزگشایی الگوریتم می‌شود. در فایستل اندازه بلوک ۶۴بیت در نظر گرفته‌شده است.

۲. **اندازه کلید**

کلید بزرگ‌تر به معنی امنیت بیشتر است؛ اما کلید بزرگ‌تر ممکن است باعث کاهش سرعت رمزگذاری و رمزگشایی گردد. امنیت بیشتر با مقاومت بیشتر در مقابل حملات جستجوی‌فراگیر و ایجاد به‌هم‌ریختگی (Confusion) بیشتر، حاصل می‌شود. در حال حاضر معمولاً کلیدهای ۶۴بیتی یا کمتر موردقبول نیست و از کلیدهای ۱۲۸بیتی استفاده می‌شود.

۳. **تعداد دورها**

ماهیت رمزنگاری فایستل به‌صورت تک دور می‌باشد که برای امنیت ناکافی است؛ اما می‌توان با افزایش تعداد دورها امنیت را افزایش داد.

۴. **الگوریتم تولید زیرکلید**[۱]

برای داشتن پیچیدگی بیشتر در این الگوریتم، باید از کلیدهای قدرتمند استفاده شود.

۵. **تابع F در هر دور**

بازهم پیچیدگی بیشتر به معنای مقاومت بیشتر در برابر رمزیابی است.

الگوریتـــم DES

در اواخر دهه 1960میلادی، شرکت IBM یک گروه تحقیقاتی رمزنگاری کامپیوتر به رهبری آقای فایستل ایجاد نمود. آقای فایستل و گروهش کار خود را جهت بهبود همان الگوریتم قبلی یعنی الگوریتم فایستل آغاز نمود. نهایتاً این گروه موفق شد در سال 1971 یک الگوریتم جدید به نام Lucifer را معرفی نماید. الگوریتم Lucifer که بر مبنای الگوریتم فایستل شکل گرفته بود؛ اندازه بلوک‌ها را ۶۴بیتی و اندازه کلید را ۱۲۸بیتی در نظر گرفت. سپس این الگوریتم به مرکز تجارت Lloyd's of London [۲] برای استفاده در یک سیستم توزیع پول، فروخته شد.

---

[1] Subkey generation algorithm

[۲] Lloyd یک مرکز تجارت جهانی بیمه تخصصی، واقع در پایتخت انگلیس است. www.lloyds.com

با توجه به نتایج امیدوارکننده پروژهLucifer، شرکتIBM تلاش خود را برای توسعه یک محصول رمزگذاری تجاری و قابل‌عرضه در بازار که امکان اجرا بر روی یک تراشه[1] را داشته باشد، آغاز نمود. اما این بار تحقیقات تنها به شرکت IBM محدود نماند؛ آقایان Walter Tuchman و Carl Meyer از طرف IBM در کنار دیگر مشاوران و همچنین مشاوره‌های فنی آژانس امنیت ملی[2] امریکا، توانستند یک نسخه اصلاح‌شده Lucifer که در برابر رمزگشایی مقاوم‌تر بود را ارائه نمایند. در این نسخه به دلیل اینکه الگوریتم امکان اجرا بر روی تراشه را داشته باشد، اندازه کلید به ۵۶ بیت کاهش یافت.

در سال 1973 در پی فراخوان اداره ملی استاندارد[3]، IBM نتایج تحقیقات Tuchman-Meyer را ارائه نمود. ازآنجاکه بهترین الگوریتم پیشنهادشده به اداره ملی استاندارد، الگوریتم Lucifer بود؛ لذا این الگوریتم در سال 1977 بانام استاندارد رمزگذاری داده‌ها یا همان Data Encryption Standard (DES)، به تصویب رسید.

هرچند که Lucifer به‌عنوان استاندارد توسط اداره ملی استاندارد معرفی گردید، اما منتقدان به دلایل زیر نسبت به این الگوریتم اعتراض داشتند:

۱. طول کلید در الگوریتم Lucifer اصلی IBM، ۱۲۸بیت بود؛ درصورتی‌که طول کلید سیستم پیشنهادی(DES)، با ۷۲ بیت کاهش به ۵۶ بیت رسیده بود. لذا منتقدان نگران بودند که این الگوریتم در مقابل حملات جستجوی فراگیر مقاومت لازم را نداشته باشد.
۲. نگرانی دوم نسبت به سری بودن معیارهای طراحی ساختار داخلی DES و S-Boxها بود. چراکه در این صورت کاربران نمی‌توانستند مطمئن شوند که DES عاری از هرگونه ضعف امنیتی است.

رمزگذاری در DES

همان‌طور که گفته شد، الگوریتم DES بر اساس Lucifer و الگوریتم Lucifer نیز بر اساس الگوریتم فایستل توسعه داده‌شده است؛ لذا DES ازنظر عملکرد بسیار شبیه فایستل می‌باشد. البته DES نسبت به فایستل دارای تغییراتی ازجمله اضافه شدن Initial Vector است. همچنین طول بلوک‌ها در DES، ۶۴ بیت و طول کلید ۵۶ بیت است. البته لازم به ذکر است که تابع، یک کلید ۶۴ بیتی را در ورودی به همراه متن‌ساده دریافت می‌کند؛ اما به دلیل اینکه۸ بیت آن به‌عنوان بیت‌توازن (Parity) مورداستفاده قرار می‌گیرد، کلید در اصل ۵۶ بیتی خواهد بود.

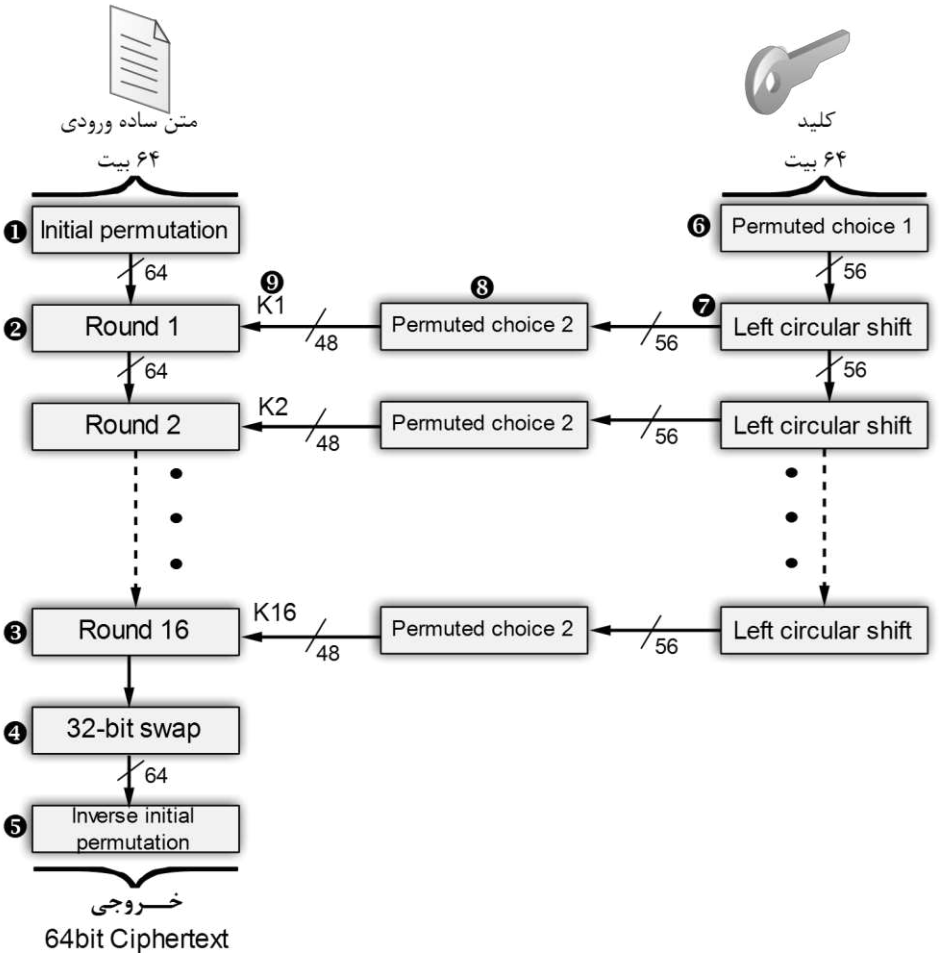
---

[1] Chip
[2] National Security Agency (NSA)
[3] National Bureau of Standards (NBS)

تصویر زیر نشان‌دهنده یک نمای کلی از نحوه عملکردDES است. در سمت چپ تصویر، فرآیند تبدیل متن‌ساده به متن‌رمزشده، در سه فاز نمایش داده می‌شود.

در فاز اول، الگوریتم طی یک جایگشت اولیه[1] که به‌اختصارIP نامیده می‌شود، اقدام به چینش دوباره ۶۴ بیت متن ساده ورودی می‌نماید. ❶

فاز دوم شامل۱۶ دور شبیه به هم است که در هریک از این دورها هردو عمل permutation وsubstitution، صورت می‌گیرد. ❷ در شانزدهمین دور، یک خروجی ۶۴ بیتی که تابعی از متن ورودی و کلیدها است، به‌دست می‌آید. ❸ این خروجی دارای دو قسمت مساوی راست و چپ است که در آخرین بخش از فاز دو، این دو قسمت نیز با یکدیگر جابجا می‌شود. ❹



[1] Initial Permutation (IP)

در فاز سوم یک عملیات جایگشت که دقیقاً معکوس جایگشت اولیه(در فاز اول) است، روی خروجی فاز دو انجام‌گرفته و درنهایت یک متن رمزشده تولید می‌شود.❺

اما در سمت راست تصویر فوق، یک کلید ۶۴ بیتی در ورودی داریم که پس از اعمال یک تابع جایگشت، به کلید ۵۶ بیتی تبدیل می‌شود.❻ سپس برای هر۱۶ دور یک زیرکلید که ترکیبی از جابجایی مدور به‌چپ[1]❼و جایگشت❽می‌باشد، ایجاد می‌گردد. هرچند که تابع جایگشت برای هر ۱۶ دور یکسان است، اما به دلیل جابجایی‌های مکرر بیت‌های کلید، زیرکلیدهای متفاوتی در هر دور ایجاد می‌گردد.❾

مراحل فوق یک بررسی کلی از نحوه عملکرد الگوریتم DES بود؛ در ادامه هرکدام از بخش‌های تصویر به‌طور اختصاصی تشریح می‌گردد.

جایگشت اولیه (Initial Permutation)

جایگشت اولیه و معکوس آن، از طریق جداول زیر تعریف گردیده است. همان‌طور که در جداول ملاحظه می‌نمایید، ورودی جدول ۶۴ بیت است که با شماره‌های ۱ تا ۶۴ محل قرارگیری هر بیت را نشان می‌دهد. بیت‌های ورودی بر اساس شماره بیت باید در خانه‌هایی از جدول که دارای شماره متناظر با آن‌ها است، جایگشت شده و به همان صورت در خروجی قرار گیرند.

توجه داشته باشید که ورودی و خروجی جداول ۶۴ بیت است و هیچ بیتی از متن‌اصلی اضافه یا حذف نگردیده و صرفاً عمل جایگشت است که بر روی بیت‌ها انجام می‌پذیرد.

Initial Permutation (IP)

58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

Inverse Initial Permutation

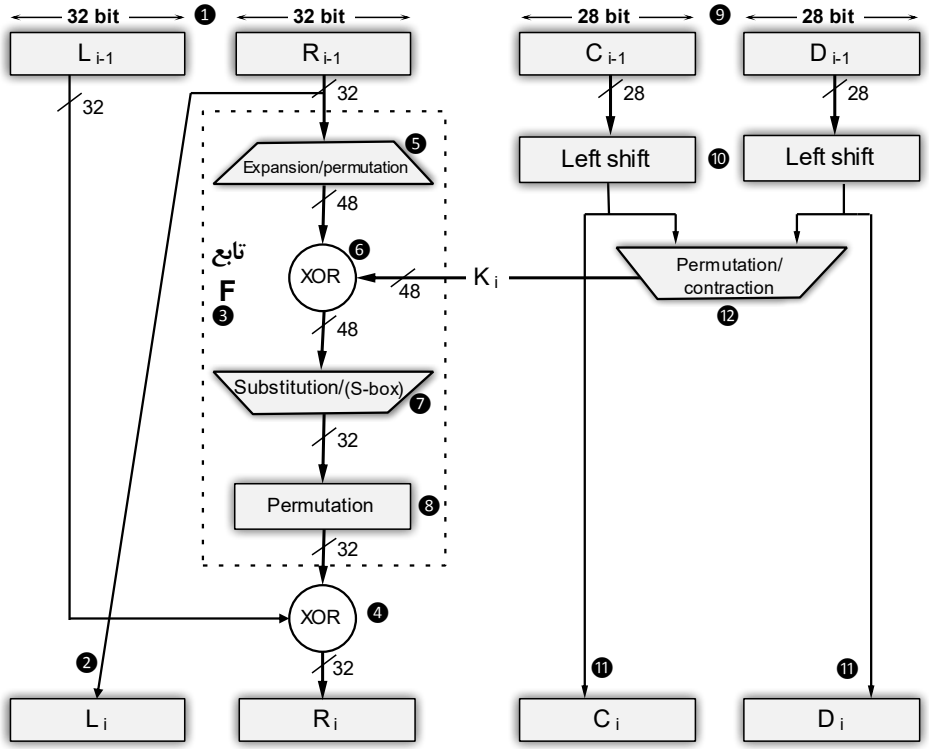
40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25

[1] left circular shift

جزئیات یک دور (Round) از الگوریتم

در تصویر زیر ساختار داخلی و جزئیاتی که در یک دور از الگوریتم DES انجام می‌گیرد، نمایش داده‌شده است. همان‌طور که در سمت چپ تصویر ملاحظه می‌کنید، بلوک ۶۴ بیتی ورودی به دو قسمت ۳۲ بیتی چپ و راست تقسیم‌شده است. ❶ قبل از اینکه هیچ عملی روی نیمه راست صورت پذیرد، یک کپی از آن به‌عنوان نیمه چپ راهی دور بعد می‌گردد. ❷ بر روی نیمه راست نیز تابع F بر اساس Subkey مربوطه اعمال گردیده ❸ و سپس خروجی تابع F با نیمه سمت چپ XOR می‌شوند. ❹ خروجی XOR به‌عنوان نیمه راست به دور بعد ارسال می‌گردد. عملیاتی که بر روی هر نیمه انجام می‌شود را می‌توان به‌صورت خلاصه و طبق فرمول زیر نمایش داد:

$$L_i = R_{i-1}$$
$$R_i = L_{i-1} \oplus F(R_{i-1}, K_i)$$



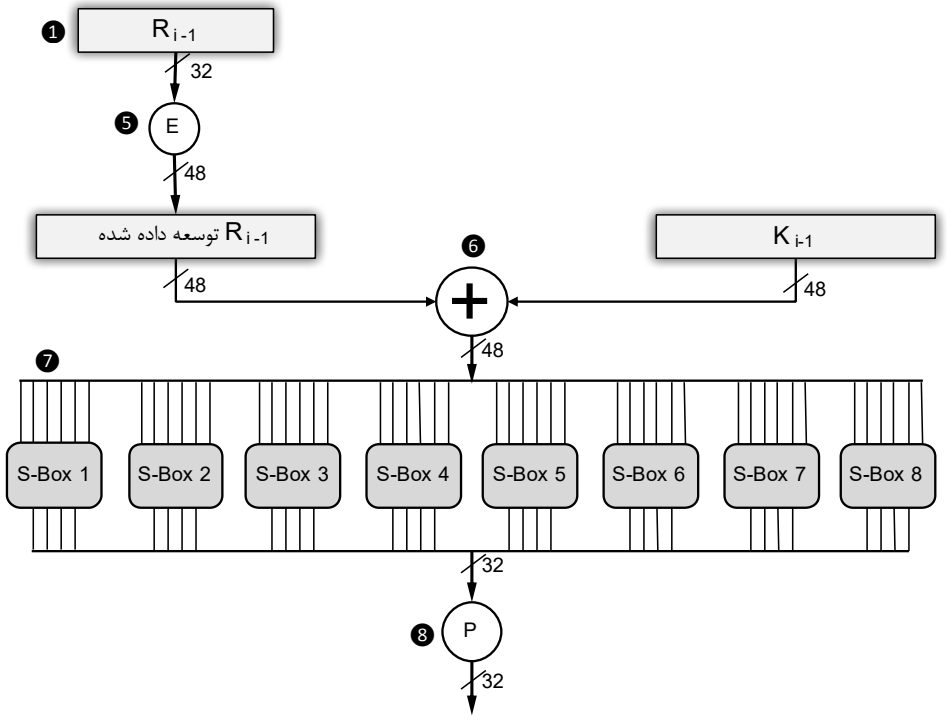
اولین کاری که داخل تابعF روی می‌دهد، عملیات جایگشت و توسعه ❺ بر روی نیمه راست دیتای ورودی است. در این عملیات که طبق جدول زیر انجام می‌پذیرد، یک ورودی ۳۲بیتی

دریافت و ضمن عمل جایگشت، توسعه نیز بر روی آن انجام‌شده و به‌صورت یک بلوک ۴۸ بیتی در خروجی ظاهر می‌شود. برای افزایش۳۲ بیت ورودی به ۴۸ بیت، طبق جدول جایگشت زیر، ۱۶عدد از بیت‌ها تکرار می‌شوند.

32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	1

خروجی ۴۸ بیتی فوق با زیرکلید مربوطه که آن‌هم دارای ۴۸ بیت است، XOR شده و به‌صورت یک خروجی ۴۸ بیتی به مرحله بعد تحویل داده می‌شود. ❻

در(S-box)/Substitution، ورودی ۴۸ بیتی توسط ۸ عدد S-Box (طبق تصویر زیر)، به ۳۲ بیت کاهش داده می‌شود. هرS-Box که دارای جدول مختص به خود است، ۶ بیت را به‌عنوان ورودی دریافت کرده و پس از جایگزینی، تنها ۴ بیت را در خروجی تحویل می‌دهد. ❼



پس از کاهش ۴۸ بیت به ۳۲ بیت، عمل جایگشت(Permutation) نیز طبق جدول زیر بر روی آن اعمال‌شده و یک خروجی ۳۲ بیتی تولید می‌نماید.❽

16	7	20	21	29	12	28	17
1	15	23	26	5	18	31	10
2	8	24	14	32	27	3	9
19	13	30	6	22	11	4	25

درنهایت خروجی تابع F که یک بلوک ۳۲ بیتی است، با نیمه سمت چپ XOR می‌شود؛ سپس بلوک ۳۲ بیتی ایجادشده به‌عنوان نیمه راست راهی دور بعد می‌گردد.❹

حالا برویم سراغ سمت راست تصویر که نحوه ایجاد زیرکلید را نشان می‌دهد. البته قبل از اینکه وارد تصویر شویم! یادآوری می‌کنم که کلید اصلی ۶۴ بیت است که با کم کردن ۸ بیت مربوط به Parity، یک کلید ۵۶ بیتی خواهیم داشت. بر روی این کلید ۵۶ بیتی طبق جدول زیر جایگشت اعمال می‌گردد.

57	49	41	33	25	17	9
1	58	50	42	34	26	18
10	2	59	51	43	35	27
19	11	3	60	52	44	36
63	55	47	39	31	23	15
7	62	54	46	38	30	22
14	6	61	53	45	37	29
21	13	5	28	20	12	4

حالا برگردیم به همون سمت راست تصویر خودمون!☺ کلید ۵۶ بیتی ورودی نیز همانند متن ورودی، به دو قسمت مساوی چپ و راست تقسیم می‌گردد.❾ سپس بر روی هر نیمه که یک بلوک۲۸ بیتی است، عمل جابجایی‌مدور به‌چپ انجام می‌پذیرد.❿ اما اینکه این جابجایی باید شامل چند بیت باشد، در دورهای مختلف، متفاوت و طبق جدول زیر است:

Round Number:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Bits Rotated:	1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

پس از جابجایی، یک کپی از هر دو نیمه راست و چپ برای استفاده در دور بعد ارسال می‌گردد.⓫ اما برای استفاده در همین دور، باید بر روی آن‌ها عمل Permutation /Contraction نیز انجام پذیرد.⓬ جایگشت این مرحله طبق جدول زیر و فقط برای ۴۸ بیت اتفاق افتاده و ۸ بیت دیگر (9, 18, 22, 25, 35, 38, 43, 54) نیز حذف گردیده تا خروجی ۴۸ بیتی به‌دست آید. به‌عبارت‌دیگر ضمن عمل جایگشت، انقباض نیز صورت گرفته است.

14	17	11	24	1	5	3	28
15	6	21	10	23	19	12	4
26	8	16	7	27	20	13	2
41	52	31	37	47	55	30	40
51	45	33	48	44	49	39	56
34	53	46	42	50	36	29	32

خروجی ۴۸ بیتی فوق به‌عنوان زیرکلید دور مربوطه، به تابع اعمال می‌گردد. ❻ در هر دور از یک زیرکلید برای رمزگذاری استفاده می‌شود که با زیرکلیدهای استفاده‌شده در سایر دورها متفاوت است. البته تمام زیرکلیدها از کلید اصلی ورودی مشتق گردیده است.

رمزگشایی در DES

مانند رمزگشایی در فایستل، در DES نیز الگوریتم رمزگشایی، معکوس الگوریتم رمزگذاری است. فقط توجه داشته باشید که کلیدها نیز از آخر به اول (از زیرکلید ۱۶ به زیرکلید ۱) باید مورداستفاده قرار گیرند.

بررسی قدرت DES

از زمانی که DES به‌عنوان استاندارد فدرال مورد تصویب قرار گرفت، همواره دو نگرانی درباره سطح امنیتی این الگوریتم وجود داشت: ۱. اندازه کلید و ۲. ماهیت الگوریتم.

۱. **اندازه کلید**

با کلیدی به طول ۵۶ بیت، امکان داشتن 2^{56} زیرکلید وجود دارد. بنابراین در مواجهه با حملات جستجوی فراگیر، با فرض اینکه به‌طور متوسط نیمی از کلید جستجو شود، برای یک ماشین تنها، که هر رمزگذاری DES را در یک میکروثانیه انجام می‌دهد، بیش از هزار سال زمان نیاز است که بتواند یک رمزDES را بشکند.

البته حضرات دیفی و هلمن در سال1977 ادعا کردند فنّاوری در اختیاردارند که آن‌ها را قادر به ساخت ماشینی می‌کند که به‌طور موازی دارای یک‌میلیون دستگاه رمزگذاری باشد؛ و هر دستگاه بتواند یک رمزگذاری را در یک میکروثانیه انجام دهد. با فرض قبول ادعای این دوستان، زمان شکستن یک رمز DES به حدود ۱۰ ساعت کاهش می‌یابد. هزینه ساخت ماشینی با این خصوصیت، در سال1977 چیزی حدود ۲۰ میلیون دلار تخمین زده می‌شود. نهایتاً در سال 1998، بنیاد مرزهای الکترونیکی[1] اعلام کرد

[1] Electronic Frontier Foundation (EFF)

ماشینی ۲۵۰ هزار دلاری ساخته که می‌تواند در کمتر از سه روز رمز DES را بشکند. بنیاد EFF، جزئیات این ماشین که آن را DES Cracker نامیده بود، منتشر نموده و در اختیار همه قرار داد. وجود چنین ماشینی با هزینه کم، عملاً استفاده از DES را بی‌ارزش نموده است.

۲. ماهیت الگوریتم

یکی دیگر از نگرانی‌ها، حملاتی است که ممکن بود با سوءاستفاده از ویژگی‌های الگوریتم انجام شود. اصل این نگرانی‌ها هم مربوط به S-Boxها بود؛ چراکه عمومی نشدن معیارهای ساخت S-Boxها، سوءظن علمای فن را برانگیخته بود که شاید قرار است از این S-Boxها برای شکستن رمز سوءاستفاده شود. و یا اینکه به دلیل روشن نبودن عملکرد آن‌ها، نمی‌توان S-Boxها را مورد ارزیابی قرار داده و نقاط آسیب‌پذیر آن را کشف کرد.

به‌هرحال پس از انتشار عملکرد S-Boxها نیز ظاهراً نقاط ضعف مهلکی در آن‌ها کشف یا منتشر نگردیده است.

گناهش گردن اونایی که میگن!

ولی شواهدی وجود دارد که دولت ایالات‌متحده با کمک شرکت IBM در سال‌هایی که DES به‌عنوان استاندارد رمزنگاری مورداستفاده قرار می‌گرفت، اقدام به استراق سمع پیام‌ها می‌نموده است.

البته این موضوع زیاد عجیب هم نیست! چراکه گزارش‌های تأییدشده و تأیید نشده فراوانی وجود دارد که آژانس امنیت ملی امریکا (NSA) به نحوی تمام شرکت‌های تولیدکننده محصولات IT که قصد فعالیت در بازار امریکا را دارند، مجبور به جاسوسی از کاربران خود در سراسر دنیا می‌نماید.



☑ مبحث دوم

الگوریتم AES

موسسه ملی استاندارد و فناوری[1](NIST) امریکا، در سال 2001 بر اساس الگوریتم رایندال[2] اقدام به معرفی الگوریتم رمزنگاری Advanced Encryption System (AES) نمود. الگوریتم رایندال توسط دو رمزنگار بلژیکی به‌نام‌های Joan Daemen و Vincent Rijmen ایجاد گردیده است. این رمزنگاری توانست در فراخوان موسسه ملی استاندارد و فناوری ایالات‌متحده، از رقبای خود عملکرد بهتری نشان داده و به‌عنوان استاندارد رمزگذاری تصویب و جایگزین DES گردد. به‌کارگیری این رمزنگاری حتی برای اطلاعات خیلی محرمانه نیز مورد تصویب قرارگرفته است.

AES یک الگوریتم رمزنگاری بلوکی متقارن است که برخلاف DES، از فایستل استفاده نکرده و بر اساس الگوریتم رایندال است. سایز بلوک‌های دیتا در این الگوریتم، ۱۲۸ بیت و سایز کلیدها می‌تواند ۱۲۸، ۱۹۲ یا ۲۵۶ بیت باشد. بسته به طول کلید مورداستفاده، AES دارای ۱۰، ۱۲ و یا ۱۴ دور خواهد بود. الگوریتمAES در هر دور دارای چهار تابع مختلف است که به شرح آن‌ها خواهیم پرداخت.

در مقایسه با رمزنگاری کلیدعمومی، ساختارAES همانند اغلب الگوریتم‌های رمزنگاری متقارن، از پیچیدگی‌هایی برخوردار است که تشریح کامل آن در یک مبحث نمی‌گنجد. درصورتی‌که به توضیحات بیشتری درباره این الگوریتم نیاز داشتید، به متن اصلی استاندارد مراجعه نمایید.[3]

در AES عملیات به‌جای بیت، بر روی بایت (۸ بیت) انجام می‌شود. به‌طور خاص، عملیات محاسباتی جمع، ضرب و تقسیم در AES، بر اساس گالوا فیلد انجام می‌پذیرد. ازآنجاکه دانستن گالوا فیلد برای فهم AES ضروری است، لذا در ابتدا به یادآوری آن پرداخته و سپس سراغ جزئیات الگوریتم می‌رویم.

نظریه گالوا توسط یک دانشمند جوان به نام اواریست گالوا ارائه گردیده است. این نظریه که بر پایه نظریه گروه‌ها شکل‌گرفته، باعث می‌شود معادلات ریاضی با توجه به ساختارهای مشابه، از روش‌های ساده‌تر و سریع‌تر حل شوند.

---

[1] National Institute of Standards and Technology (NIST)
[2] Rijndael
[3] http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf



اواریست گالوا
یک دانشمند و البته انقلابی فرانسوی بود؛ که علیرغم عمر کوتاهش به‌عنوان یکی از پیشگامان نظریه گروه‌ها مطرح شد و نظریه گالوا را بانام خود ارائه نمود. این دانشمند در سال 1811 در فرانسه متولد شد. وی پس از عدم موفقیت در ورود به دانشسرا، وارد عرصه مبارزات سیاسی شد و برای مدتی نیز به زندان افتاد.
پس از آزادی از زندان، دشمنان سیاسی گالوا وی را توسط دختری به دام انداخته و سپس به بهانه شرافت او را به یک دوئل کشاندند. گالوا که آن زمان کمتر از ۲۱ سال سن داشت، در دوئل کشته شد.

گالوا فیلد

گالوا فیلد یا GF بر اساس میدان محدود (Finite Field) عمل می‌کند. تفاوت اصلی میدان محدود با میدان نامحدود در آن است که عملگرهای میدان محدود یک عملگر MOD[1] در داخل خود دارند. به‌عنوان‌مثال در GF(2) فقط اعداد 0 و 1 دارای معنی هستند؛ لذا 1+1 برابر با 0 است، چراکه 2 در پیمانه 2 برابر 0 می‌شود. به عبارتی می‌توان گفت که عملیات در یک میدان محدود همیشه نتیجه‌ای در خود آن میدان تولید خواهند کرد.

استفاده از گالوا فیلد در AES نیز به این دلیل بود که بتوانند اعداد را به پیمانه موردنظر کاهش دهند. این کاهش باعث می‌شود که الگوریتم AES نسبت به فایستل سریع‌تر عمل نماید. در AES برای عملیات جمع از GF(2) و برای عملیات ضرب از $GF(2^8)$ استفاده می‌شود.

عملیات جمع در AES

توجه داشته باشید که در GF(2)، عمل جمع و تفریق یکسان است؛ چراکه 1+1=0 است و 1-1 نیز برابر 0 است. بنابراین تمامی عملیات را با XOR انجام می‌دهیم.

برای انجام عملیات جمع در AES بر اساس GF(2)، اعداد را به پیمانه دو کاهش می‌دهیم. به‌عبارت‌دیگر، تمام المان‌هایی که در جواب جمع، دارای ضریب دو هستند را حذف می‌نماییم.

عملیات ضرب در AES

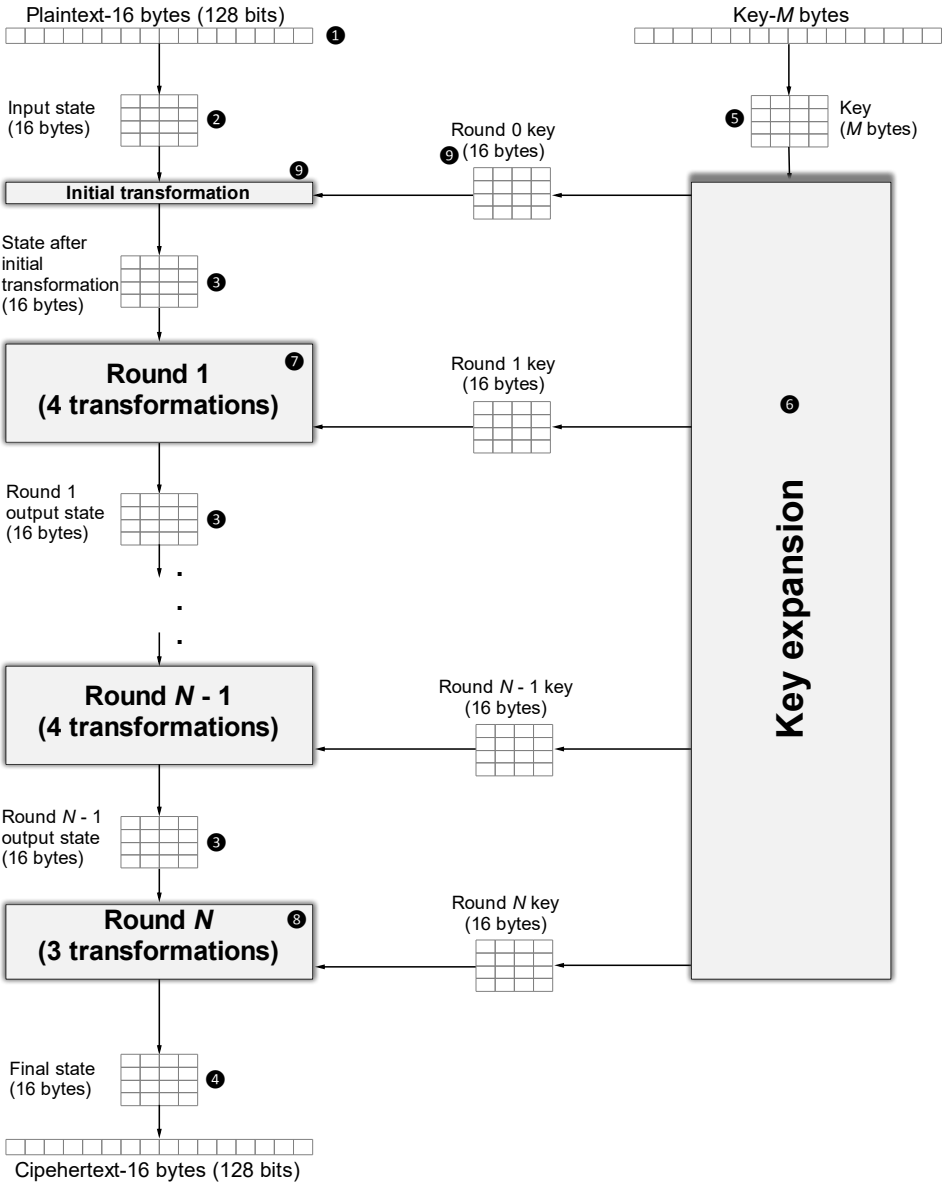
برای انجام عملیات ضرب در AES بر اساس $GF(2^8)$، اعداد را به پیمانه زیر کاهش می‌دهیم؛ این پیمانه در ضرب AES، همواره ثابت است.

$$X^8 + X^4 + X^3 + X + 1$$

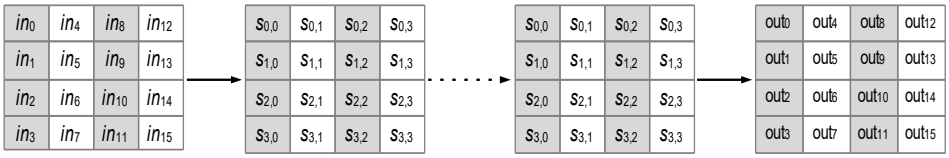
[1] باقیمانده. (قبلاً درباره Mod یادآوری کرده‌ام.)

ساختار کلی AES

تصویر زیر ساختار کلی فرآیند رمزگذاری AES را نمایش می‌دهد. درAES، اندازه بلوک متن ورودی ۱۶ بایت (۱۲۸بیت) است.❶ بسته به طول کلید که می‌تواند ۱۶، ۲۴ یا ۳۲ بایت (۱۲۸، ۱۹۲ یا ۲۵۶ بیت) باشد، الگوریتم را AES-128، AES-192 و یا AES-256 نام‌گذاری می‌نمایند.



درهرحال، متن اصلی ورودی جهت رمزگذاری و رمزگشایی در این الگوریتم، یک بلوک ۱۲۸ بیتی خواهد بود. ❶ این بلوک به‌صورت یک ماتریس مربع 4×4 در تصویر نمایش داده‌شده ❷ که هرکدام از خانه‌های ماتریس شامل یک بایت(۸ بیت) است. این ماتریس نمایش‌دهنده وضعیت[۱](State) قرارگیری بایت‌ها است؛ وضعیت (State) در هر مرحله از رمزگذاری یا رمزگشایی تغییر داده می‌شود. ❸ بعد از مرحله نهایی نیز، وضعیت در ماتریس خروجی کپی می‌گردد. ❹ این عملیات در تصویر زیر نمایش داده‌شده است.



به‌طور مشابه، کلید نیز به‌صورت یک ماتریس مربع از بایت‌ها نمایش داده‌شده است. ❺ این کلید توسط عملیات Key Expansion، ❻ به زیرکلیدهایی جهت استفاده در هر دور توسعه می‌یابد. تصویر زیر نحوه توسعه یک کلید ۱۲۸ بیتی را نمایش می‌دهد.

هر کلمه[۲] شامل ۴ بایت است، و جدول کلیدها درمجموع دارای ۴۴ کلمه برای ۱۲۸بیت است. به دلیل اینکه برای کلید ۱۲۸ بیتی، الگوریتم دارای ۱۰ دور است، پس ۱۰ ماتریس خواهیم داشت؛ البته یک ماتریس هم برای Initial داریم، که درنهایت تعداد ماتریس‌ها به ۱۱ می‌رسد. حالا ۱۱ ماتریس داریم که هرکدام دارای ۴ ستون می‌باشند: $11 \times 4 = 44$؛ پس‌نیاز به ۴۴ کلمه جهت کلید مربوط به هر ستون داریم.

توجه داشته باشید که ترتیب بایت‌ها در ماتریس بر اساس ستون است. به‌عنوان‌مثال اولین چهار بایت ورودی یک متن ساده، اولین ستون ماتریس را اشغال می‌کند و دومین چهار بایت، ستون دوم و به همین شکل تا ستون آخر ادامه می‌یابد. در کلید نیز به همین صورت است، اولین ۴ بایت کلید که به شکل یک کلمه (W) می‌باشد، ستون اول ماتریس را اشغال نموده و کلمات دیگر به همین صورت ستون‌های بعدی را اشغال می‌نمایند.



[۱] در اینجا منظور از وضعیت(State)، نحوه چینش بایت‌ها در آرایه‌های ماتریس است.

[2] Word

در الگوریتم AES، تعداد دورها به طول کلید مورداستفاده بستگی دارد. جدول زیر تعداد دور بر اساس نوع کلید را نمایش می‌دهد.

تعداد دور	طول کلید
10	128 bit
12	192 bit
14	256 bit

دور اول تا دور N-1، شامل چهار تابع تبدیل[1] مختلف : SubBytes، ShiftRows ، MixColumns و AddRoundKey است.❼ اما دور آخر تنها دارای سه تابع تبدیل است،❽ دلیل این موضوع Initial transformation❾ انجام‌شده قبل از اولین دور و با نام Round 0، است.

پس از انجام آخرین دور که نسبت به کلید مورداستفاده ممکن است دور ۱۰، ۱۲ و یا ۱۴ باشد، یک خروجی رمزشده ۱۲۸ بیتی، تولید خواهد شد.

پارامترهای AES به‌صورت خلاصه، در جدول زیر نمایش داده‌شده است:

Key Size (words/bytes/bits)	4/16/128	6/24/192	8/32/256
Plaintext Block Size (words/bytes/bits)	4/16/128	4/16/128	4/16/128
Number of Rounds	10	12	14
Round Key Size (words/bytes/bits)	4/16/128	4/16/128	4/16/128
Expanded Key Size (words/bytes)	44/176	52/208	60/240

جزئیات ساختار AES

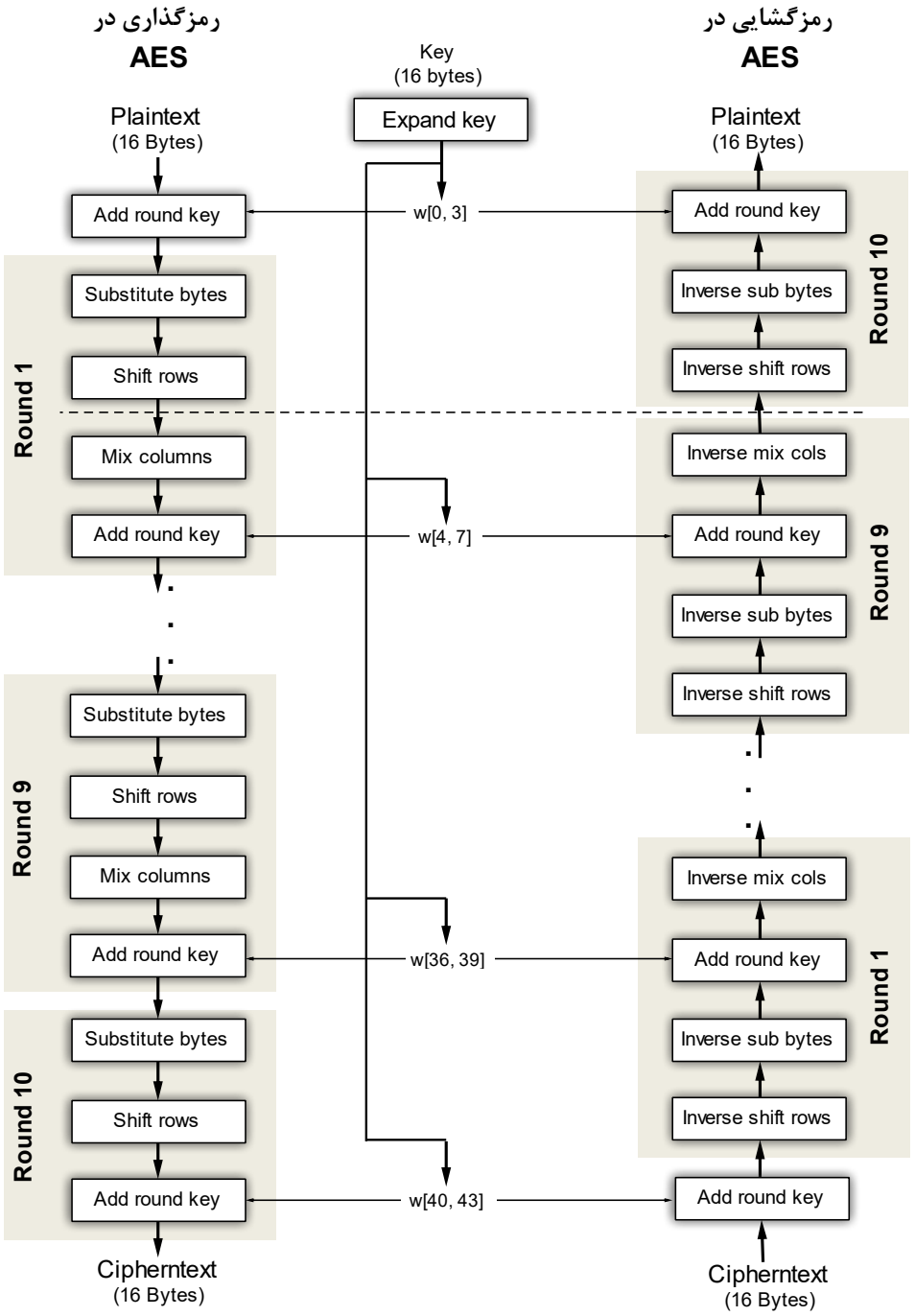
در تصویر زیر به جزئیات بیشتری از ساختار AES پرداخته‌شده است. همچنین عملیات رمزگشایی AES نیز در این تصویر نمایش داده‌شده است. در هر دور AES، توابع مختلفی وجود دارند، که نحوه تعامل مراحل با یکدیگر نیز در تصویر آمده است.

موارد زیر نکات مهمی است که باید راجع به ساختار AES در نظر داشته باشید:

۱. همان‌طور که به یاد دارید در فایستل بلوک دیتا به دونیمه تقسیم می‌گردید؛ نیمی از بلوک دیتا برای ایجاد تغییر در نیمه دیگر مورداستفاده قرار می‌گرفت و در مرحله بعد این دو نیمه با یکدیگر جابجا می‌شدند. اما در AES، در هر دور برای کل بلوک دیتا فرآیند جایگزینی و جایگشت انجام می‌شود.

۲. کلید ورودی به ۴۴ کلمه ۳۲ بیتی توسعه می‌یابد، تا برای هر دور ۴ کلمه مجزا در قالب یک کلید ۱۲۸ بیتی فراهم آورد.

[1] Transformation


رمزگذاری در
AES
Key
(16 bytes)
رمزگشایی در
AES
Plaintext
(16 Bytes)
Expand key
Plaintext
(16 Bytes)
Add round key
w[0, 3]
Add round key
Substitute bytes
Inverse sub bytes
Shift rows
Inverse shift rows
Round 1
Round 10
Mix columns
Inverse mix cols
Add round key
w[4, 7]
Add round key
Round 9
Substitute bytes
Inverse sub bytes
Shift rows
Inverse shift rows
Mix columns
Inverse mix cols
Add round key
w[36, 39]
Add round key
Substitute bytes
Inverse sub bytes
Shift rows
Inverse shift rows
Round 10
Add round key
w[40, 43]
Add round key
Cipherntext
(16 Bytes)
Cipherntext
(16 Bytes)

۳. هر دور شامل چهار مرحله مختلف است؛ که سه مرحله آن جایگزینی (Substitution)و یک مرحله آن جایگشت (Permutation) است:
   i. Substitution Bytes: با استفاده از S-Box اقدام به جایگزینی بایت با بایت بلوک دیتا می‌نماید.
   ii. ShiftRow: تنها مرحله‌ای که جایگشت (Permutation) در آن انجام می‌شود.
   iii. MixColumns: بر اساس $GF(2^8)$ اقدام به جایگزینی می‌نماید.
   iv. AddRoundKey: یک عمل ساده XOR بیتی، بین بلوک حاضر و کلید مربوطه.
۴. تعریف ساختار کلی آن ساده است. برای هر دو عمل رمزگذاری و رمزگشایی، الگوریتم با یک مرحله AddRoundKey شروع‌شده و سپس با ۹ دور که هر دور شامل ۴ مرحله است، ادامه می‌یابد. در دور دهم (آخر) هم فقط دارای ۳ مرحله است.
۵. از چهار مرحله‌ای که در هر دور انجام می‌شود، فقط مرحله AddRoundKey از کلید استفاده می‌نماید؛ به همین دلیل شروع و پایان رمزگذاری و رمزگشایی درAES، با مرحله AddRoundKey انجام می‌پذیرد. اما درصورتی‌که مراحل دیگر در ابتدا یا انتهای الگوریتم مورداستفاده قرار گیرد، بدون نیاز به دانستن کلید، برگشت‌پذیر بوده و هیچ امنیتی را اضافه نمی‌کنند.
۶. مرحله AddRoundKey بر اساس Vernam Cipher ایجاد گردیده، که البته رمز ورنام الگوریتم خفنی! محسوب نمی‌شود.[۱] سه مرحله دیگر هم که بدون استفاده از کلید انجام‌شده و به‌راحتی و بدون دانستن کلید برگشت‌پذیر می‌باشند.
پس توجه داشته باشید آن چیزی که باعث کارایی و قدرتمند شدن الگوریتم AES می‌شود، انجام هر چهار مرحله فوق در کنار هم است.

وقتی داشتم این مطلب را از کتاب رمزنگاری آقای استالینگز یادداشت می‌کردم، یاد پند وطنی خودمون افتادم! یادتونه؟
همون پیرمردی که به بچه‌هاش نفری یه شاخه‌ی درخت داد و اونا به‌راحتی شکوندنش!
بعد به هر کدوم چهارتا شاخه‌ی درخت داد، ولی بازم اونا شکوندنش!!!



۷. همان‌طور که گفته شد، سه مرحلهSubBytes، ShiftRows وMixColumns قابل بازیابی هستند، مرحلهAddRoundKey نیز با داشتن کلید، قابلیت برگشت‌پذیری دارد؛ لذا زمان رمزگشایی در صورت در اختیار داشتن کلید، کار سختی پیش رو نخواهید داشت.

[۱] به‌طور خلاصه، ورنام یک الگوریتم رمزنگاری ساده به‌صورت: Plaintext ⊕ Key = Ciphertext است.

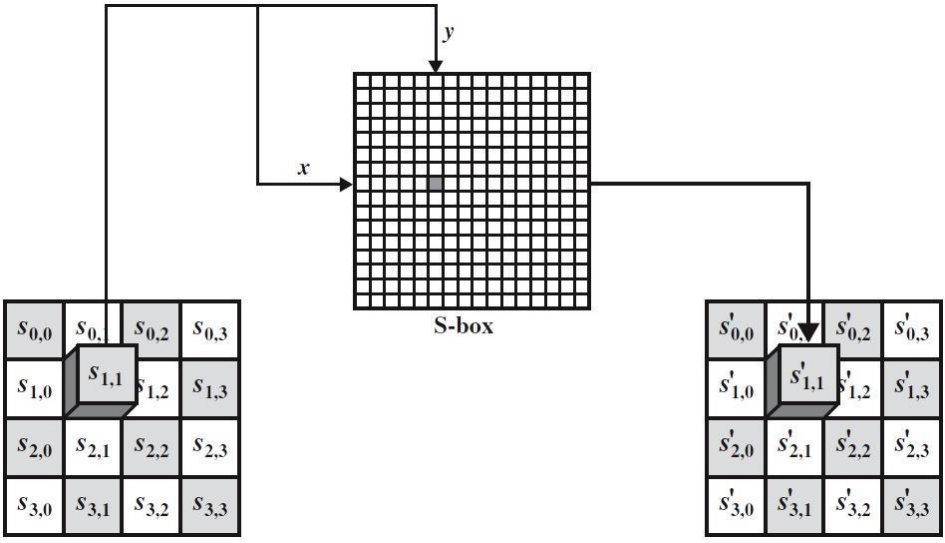
۸. برای رمزگشایی، مانند هر الگوریتم بلوکی دیگری، باید مراحل رمزگذاری به‌صورت معکوس انجام پذیرد. البته یادآوری می‌کنم که کلیدها باید از آخر به اول مورداستفاده قرار گیرند.
۹. پس‌ازآنکه برگشت‌پذیری هر چهار مرحله انجام شد، به‌آسانی می‌توان بررسی نمود که Plaintext به‌درستی در حال بازیابی باشد؛ چراکه در هر دور از رمزگذاری و رمزگشایی، باید Stateها شبیه یکدیگر باشند.
۱۰. دور نهایی در هر دو عملیات رمزگذاری و رمزگشایی، دارای سه مرحله است. به دلیل ساختار خاص AES، سه مرحله‌ای بودن دور آخر جهت برگشت‌پذیری رمزنگاری الزامی است.

جزئیات عملیات هر دور AES

AES در هر دور شامل چهار تابع تبدیل (Transformation) مختلف، به شرح زیر است:

۱. Substitute Bytes

این مرحله که به‌اختصار SubByte نیز نامیده می‌شود، طبق تصویر زیر اقدام به جایگزینی آرایه‌های ماتریس می‌نماید. AES یک ماتریس بایتی[۱] 16×16 به‌نام S-Box تعریف کرده که شامل ۲۵۶ حالت ۸ بیتی ممکن، برای جایگزینی است.



[۱] منظور از ماتریس بایتی، ماتریسی است که مقادیر خانه‌های آن بر اساس بایت باشد.

هر بایت از ماتریس وضعیت، به‌صورت جداگانه با یک بایت جدید از S-Box، به‌صورت زیر نگاشت[1] می‌شود:

چهار بیت سمت چپ بایت به‌عنوان شماره ردیف (x)، و چهار بیت راست بایت به‌عنوان شماره ستون (y)، مشخص‌کننده خانه‌ای از ماتریس S-Box است که باید با بایت موردنظر جایگزین شود.

مقادیر آرایه‌های ماتریس S-Box مورداستفاده در این تابع، برای تمام دورها یکسان است. برای مشاهده این مقادیر می‌توانید به متن استاندارد مراجعه نمایید.

۲. Shift Rows

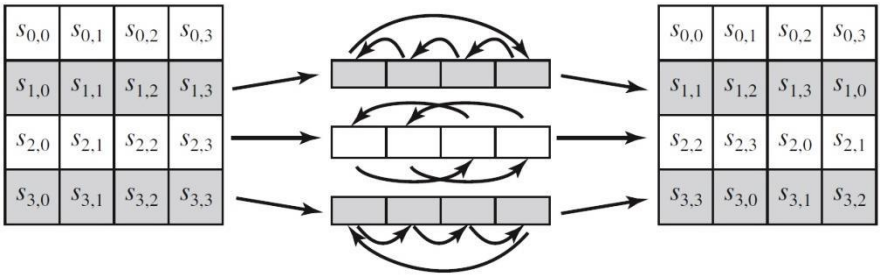
در این مرحله، جایگشت در هر سطر از ماتریس وضعیت، انجام می‌گیرد. البته مقدار جابجایی در هر سطر متفاوت و به‌صورت زیر است:

سطر اول = صفر بیت، به‌عبارت‌دیگر در سطر اول جابجایی انجام نمی‌شود.

سطر دوم = ۱ بیت به سمت چپ

سطر سوم = ۲ بیت به سمت چپ

سطر چهارم = ۳ بیت به سمت چپ



۳. Mix Columns

این مرحله به‌صورت جداگانه برای هر ستون از ماتریس وضعیت انجام می‌پذیرد. در MixColumns هم یک ماتریس ثابت به‌صورت زیر وجود دارد که مبنای عملیات انجام شده در این تابع است.

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix}$$

[1] Map

عناصر هر ستون از ماتریس وضعیت باید با عناصر هر سطر از ماتریس ثابت فوق، ضرب شده[1] و حاصل‌ضرب همه عناصر با یکدیگر XOR شوند. سپس جواب به‌دست‌آمده به‌عنوان آرایه ماتریس حاصل در این مرحله، ثبت می‌گردد.

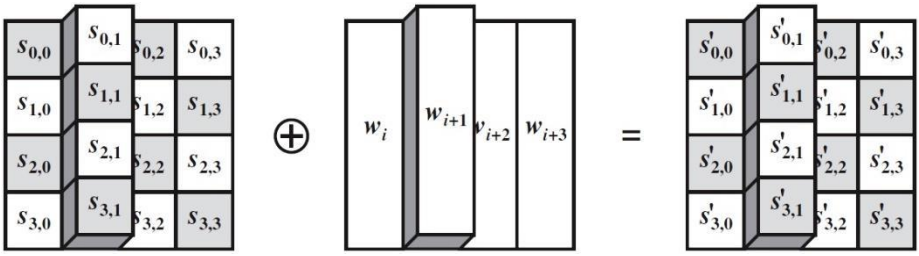
$$\begin{bmatrix}02 & 03 & 01 & 01\\ 01 & 02 & 03 & 01\\ 01 & 01 & 02 & 03\\ 03 & 01 & 01 & 02\end{bmatrix}\begin{bmatrix}s_{0,0} & s_{0,1} & s_{0,2} & s_{0,3}\\ s_{1,0} & s_{1,1} & s_{1,2} & s_{1,3}\\ s_{2,0} & s_{2,1} & s_{2,2} & s_{2,3}\\ s_{3,0} & s_{3,1} & s_{3,2} & s_{3,3}\end{bmatrix}=\begin{bmatrix}s'_{0,0} & s'_{0,1} & s'_{0,2} & s'_{0,3}\\ s'_{1,0} & s'_{1,1} & s'_{1,2} & s'_{1,3}\\ s'_{2,0} & s'_{2,1} & s'_{2,2} & s'_{2,3}\\ s'_{3,0} & s'_{3,1} & s'_{3,2} & s'_{3,3}\end{bmatrix}$$

فرمول محاسباتی ماتریس فوق به‌صورت زیر است:

$$s'_{0,j}=(2*s_{0,j})\oplus(3*s_{1,j})\oplus(1*s_{2,j})\oplus(1*s_{3,j})$$
$$s'_{1,j}=(1*s_{0,j})\oplus(2*s_{1,j})\oplus(3*s_{2,j})\oplus(1*s_{3,j})$$
$$s'_{2,j}=(1*s_{0,j})\oplus(1*s_{1,j})\oplus(2*s_{2,j})\oplus(3*s_{3,j})$$
$$s'_{3,j}=(3*s_{0,j})\oplus(1*s_{1,j})\oplus(1*s_{2,j})\oplus(2*s_{3,j})$$

۴. Add Round Key

فارغ از اینکه کلید اصلی چند بیت بوده است، به ازای هر دور یک کلید ۱۲۸ بیتی خواهیم داشت. ماتریس ورودی الگوریتم نیز همواره ۱۲۸ بیت است؛ لذا در این مرحله بیت‌های State با بیت‌های کلید، XOR شده و خروجی این مرحله را تشکیل می‌دهند.



کلید هر دور شامل چهار کلمه ۳۲ بیتی است. هر کلمه با یک ستون (۴ بایت یا ۳۲ بیت) از ماتریس وضعیت، XOR شده و در ماتریس خروجی قرار می‌گیرد.

[1] Multiplication over the finite field GF(2^8)

تنها تابعی که در هر دور بر اساس یک مقدار متغیر عمل می‌کند، تابع AddRoundKey است؛ به همین دلیل، اولین و آخرین مرحله‌ای است که در رمزگذاری و رمزگشایی باید انجام شود. به عبارتی می‌توان گفت این تابع تأمین‌کننده امنیت الگوریتم AES است.



مقادیر S-Box مرحله SubBytes، در همه دورها یکسان است.
مقدار جابجایی بیت در مرحله ShiftRow، در همه دورها یکسان است.
مقادیر ماتریس مورداستفاده در مرحله MixColumns، در همه دورها یکسان است.
تنها عامل متغیری که در هر دور وجود دارد و مختص به همان دور می‌باشد، کلید دور است.

ویژگی‌های کلیدرمز در AES

ایجادکنندگان رایندال، الگوریتم توسعه کلید را در برابر حملات شناخته‌شده، مقاوم طراحی کرده‌اند؛ به‌طوری‌که هر کلید مختص به یک دور ایجادشده و شباهتی با کلید دورهای دیگر ندارد. ازجمله ویژگی‌های خاص کلید در AES، می‌توان به موارد زیر اشاره کرد:

- دانستن بخشی از کلید رمز[1] یا کلید دور[2]، مهاجم را قادر به محاسبه بقیه کلید نمی‌کند.
- تبدیل معکوس‌پذیر.
- سرعت مناسب بر روی رنج وسیعی از پردازنده‌ها.
- استفاده از دورهای ثابت، برای از بین بردن تقارن.
- پراکندگی (Diffusion) کلیدرمز در کلید دورها؛ به صورتی که هر بیت کلید رمز، بسیاری از بیت‌های کلید دور را تحت تأثیر قرار می‌دهد.
- به‌قدر کفایت غیرخطی است.
- به‌سادگی می‌توان آن را توصیف کرد.

[1] Cipher key
[2] Round key

☑ مبحث سوم

الگوریتم Triple DES

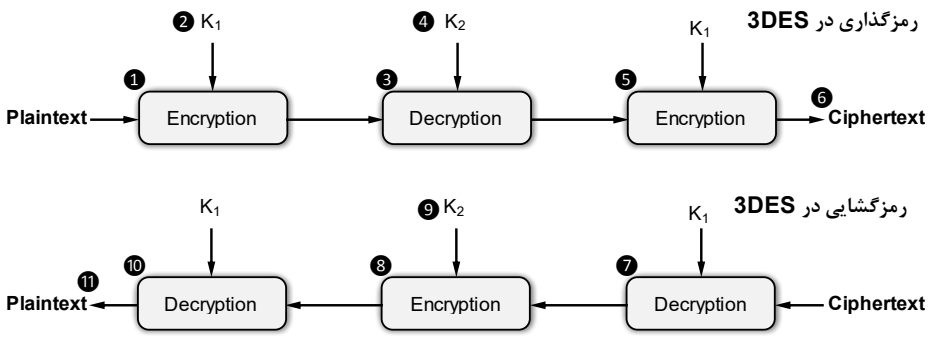
همان‌طور که گفتیم الگوریتمDES در سال 1999 شکسته شد و جای خود را به رمزAES داد. آسیب‌پذیری که درDES وجود داشت، اندازه ۵۶ بیتی کلید آن بود. کوچک بودن طول کلید باعث شد تا DES نتواند در مقابل حملات جستجوی فراگیر که با توجه به بهبود سخت‌افزارها، قدرت بیشتری گرفته بود، مقاومت نماید.

پس از شکسته شدنDES، باید جایگزین مناسبی برای رمزنگاری معرفی می‌گردید. یکی از این جایگزین‌ها معرفی یک استاندارد کاملاً جدید مثل AES بود. اما راه دیگر برای جایگزینی که می‌توانست سرمایه‌گذاری‌های سخت‌افزاری و نرم‌افزاری انجام‌شده روی DES را حفظ کند، انجام رمزنگاری متعدد با کلیدهای متعدد ولی بر اساس DES بود.

برای رمزنگاری متعدد با کلیدهای متعدد بر اساس DES، یک الگوریتم جدیدی به نام Triple DES ارائه گردید. در این الگوریتم عملیات DES سه مرتبه پشت سرهم انجام می‌گیرد، البته برای مقاومت در برابر حملات جستجوی فراگیر نیز، از دو یا سه کلید متفاوت در مراحل فوق استفاده می‌گردد. در این مبحث به بررسی Triple DES یا 3DES، در هر دو حالت، دو کلید و سه کلید پرداخته می‌شود.

3DES با دو کلید متفاوت

تصویر زیر طرح رمزگذاری و رمزگشاییTriple DES را با دو کلید متفاوت نمایش می‌دهد:



شاید دیدن کلمه Decryption در عملیات رمزگذاری و کلمه Encryption در عملیات رمزگشایی، تعجب شما را برانگیخته باشد. اما باید بگویم که در اینجا هر دو عملیات فوق در راستای وظایف رمزگذاری و رمزگشایی مورداستفاده قرار می‌گیرند.

برای رمزگذاری در 3DES، در مرحله اول بر روی متن ساده عملیات Encryption❶ با کلید $K_1$❷ انجام می‌شود. این مرحله دقیقاً همانند عملیات رمزگذاری در DES است.

اما در مرحله دوم، عملیات Decryption❸ بر اساس کلید $K_2$❹ روی خروجی مرحله اول انجام می‌پذیرد. به دلیل اینکه کلید مورداستفاده در این مرحله با کلید مورداستفاده در مرحله اول متفاوت است؛ لذا عملیات Decryptionای که انجام می‌شود نه‌تنها باعث رمزگشایی خروجی مرحله قبل نخواهد شد، بلکه خود موجب رمزگذاری اطلاعات در مرحله دوم البته با الگوریتمی متفاوت از مرحله قبل، می‌گردد.

در مرحله سوم، مجدداً عملیات Encryption❺ با همان کلید مورداستفاده در مرحله اول، بر روی خروجی مرحله دوم انجام می‌شود.

درنهایت خروجی سه مرحله فوق، یک متن رمزشده بر اساس دو الگوریتم متفاوت و دو کلید متفاوت، اما در سه مرحله خواهد بود.❻

همانند هر الگوریتم رمزنگاری بلوکی متقارن دیگر، باید عملیات رمزگشایی در 3DES نیز به‌صورت معکوس عملیات رمزگذاری صورت پذیرد.

در مرحله اول رمزگشایی، عملیات معکوسی که باید انجام شود، همان عملیات Decryption❼ است؛ چراکه در مرحله سوم رمزگذاری❺ عملیات Encryption انجام‌شده است. توجه داشته باشید که کلید مرحله اول رمزگشایی، کلید مورداستفاده در مرحله سوم رمزگذاری یا همان K_1 است.

چون در مرحله دوم عملیات رمزگذاری، Decryption❸ صورت گرفته، پس در مرحله دوم رمزگشایی باید الگوریتم معکوس آن، یعنی Encryption❽ انجام پذیرد. البته کلید هم باید همان K_2 ❾ باشد که در مرحله دوم رمزگذاری مورداستفاده قرارگرفته بود.

در مرحله سوم رمزگشایی، عملیات Decryption❿ که معکوس عملیات Encryption مرحله اول رمزگذاری❶ است، انجام می‌شود. کلید نیز همان K_1 مورداستفاده در رمزگذاری است.

درنهایت خروجی عملیات رمزگشایی، پس از سه مرحله با دو الگوریتم متفاوت و دو کلید متفاوت، متن ساده اولیه خواهد بود.⓫

فرمول رمزگذاری و رمزگشایی، به‌صورت خواهد بود:

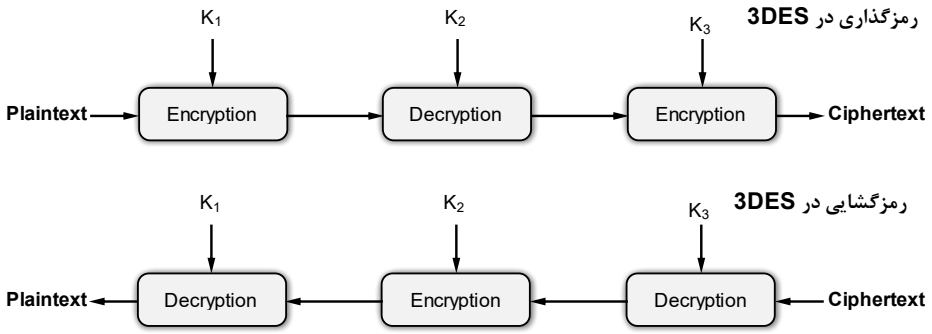
$$C = \mathrm{E}(K_1, \mathrm{D}(K_2, \mathrm{E}(K_1, P)))$$

$$P = \mathrm{D}(K_1, \mathrm{E}(K_2, \mathrm{D}(K_1, C)))$$

3DES با سه کلید متفاوت

درصورتی‌که بخواهیم بازهم باعث افزایش امنیت در پروتکل 3DES شویم، می‌توانیم به‌جای دو کلید متفاوت در سه مرحله، از سه کلید متفاوت در سه مرحله استفاده نمائیم.

تصویر زیر نمایش‌دهنده رمزگذاری و رمزگشایی در 3DES با سه کلید متفاوت، است:



همان‌طور که در تصویر فوق ملاحظه می‌نمایید، تنها تفاوت این روش با روش قبل، اضافه شدن کلید K_3 به مرحله سوم رمزگذاری و مرحله اول رمزگشایی است. در سایر مراحل و همچنین نحوه رمزگذاری و رمزگشایی نیز تفاوتی با توضیحات ارائه‌شده درروش قبلی ندارد.

فرمول رمزگذاری و رمزگشایی، به‌صورت خواهد بود:

$$C = \mathrm{E}(K_1, \mathrm{D}(K_2, \mathrm{E}(K_3, P)))$$

$$P = \mathrm{D}(K_3, \mathrm{E}(K_2, \mathrm{D}(K_1, C)))$$

3DES به دلیل استفاده از سه کلید متفاوت۵۶ بیتی، درمجموع دارای یک کلید ۱۶۸(56x3=168) بیتی است. به‌عبارت‌دیگر طول کلید در 3DES، ۱۶۸ بیت است.

افزایش طول کلید از ۵۶ بیت به ۱۶۸ بیت در کنار تکرار عملیات Encryption و Decryption، باعث گردیده تا 3DES از مقاومت خوبی در برابر حملات جستجوی فراگیر برخوردار باشد، به‌طوری‌که همچنان بسته به شرایط، مورداستفاده قرار می‌گیرد.

☑ مبحث چهارم

روش‌های عملیاتی رمزنگاری بلوکی

یک الگوریتم رمزنگاری بلوکی، یک مقدار ثابت از دیتا را به‌عنوان بلوک ورودی دریافت کرده و با یک مقدار ثابت از کلید اقدام به رمزگذاری می‌نماید. درصورتی‌که طول دیتای موردنظر بزرگ‌تر از بلوک ورودی الگوریتم باشد، آنگاه باید دیتا به بلوک‌هایی با اندازه موردنظر تقسیم‌شده و سپس بلوک به بلوک تحویل الگوریتم شوند. اما زمانی که چندین بلوک از متن‌ساده با یک کلید رمز می‌شوند، باعث بروز بعضی مسائل امنیتی می‌شوند. ازجمله این مسائل امنیتی، می‌توان به ظهور نشانه‌های آماری در بلوک‌های رمزشده اشاره کرد. با توجه به اینکه متن ساده اولیه به بلوک‌های هم‌اندازه تقسیم‌شده و عملیات مشابهی بر روی آن‌ها انجام می‌گیرد، قطعاً نشانه‌های آماری خوبی را در اختیار یک هکر تحلیل‌کننده قرار می‌دهد تا بتواند از آن برای شکستن رمز استفاده نماید.

موسسه ملی استاندارد و فناوری ایالات‌متحده (NIST) برای به کار بردن الگوریتم‌های رمزنگاری بلوکی متقارن ازجمله 3DES و AES، در انواع برنامه‌های کاربردی، پنج روش عملیاتی زیر را ارائه کرده است:

۱. Electronic Codebook (ECB)، یا دفترچه کد الکترونیکی.
۲. Cipher Block Chaining (CBC)، یا زنجیره‌ای شدن بلوک‌های رمزشده.
۳. Cipher Feedback (CFB)، یا بازخورد رمز.
۴. Output Feedback (OFB)، یا بازخورد خروجی.
۵. Counter (CTR)، یا شمارنده.

در اصل روش عملیاتی، تکنیکی برای افزایش اثر یک الگوریتم رمزنگاری یا تطبیق آن برای استفاده توسط برنامه‌هایی از قبیل بلوک‌های دنباله‌ای یا جریان داده، است. لذا پنج روش عملیاتی فوق برای پوشش طیف وسیعی از برنامه‌های کاربردی که می‌توانند از رمزنگاری بلوکی استفاده نمایند، ارائه گردیده است.

لازم به ذکر است که روش‌های فوق صرفاً نحوه به‌کارگیری الگوریتم رمزنگاری را تحت تأثیر قرار می‌دهند، و به‌عنوان یک الگوریتم رمزنگاری محسوب نمی‌شوند.

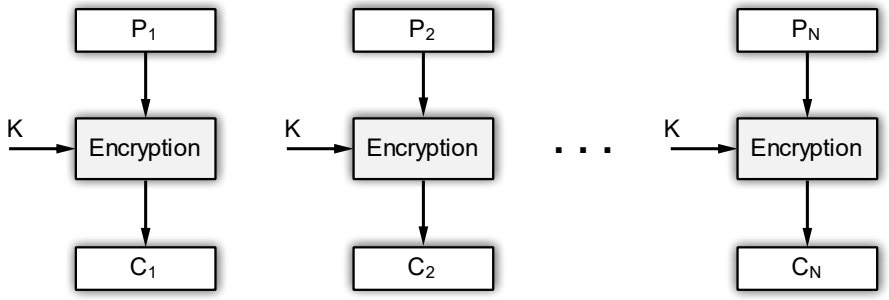
روش Electronic Codebook (ECB)

ساده‌ترین روش عملیاتی، روشECB است. ECB در هر زمان فقط از یک بلوک دیتا استفاده کرده و برای رمزگذاری بلوک‌های بعدی دیتا نیز از همان کلید قبلی استفاده می‌نماید. به دلیل اینکه برای هر بلوک متن‌ساده به ازای یک کلید مشخص، یک متن رمز شده منحصربه‌فرد وجود دارد؛ لذا از عبارت Codebook برای نام‌گذاری این روش استفاده‌شده است. بنابراین ما می‌توانیم یک Codebook غول‌پیکری را تصور کنیم که در آن برای هر بلوک احتمالی از متن ساده، متناظری برای متن رمزشده، وجود دارد.

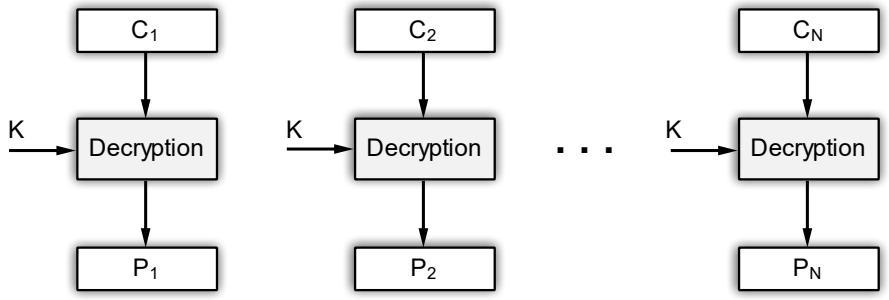
روشECB برای رمزنگاری یک مقدار کوتاه دیتا ایده‌آل است. مثلاً برای انتقال امن کلید DES یا AES می‌توان از آن استفاده نمود.

یک ویژگی قابل‌توجه ECB این است که اگر یک بلوک یکسان از دیتا، بیش از یک‌بار در متن ورودی تکرار شده باشد، متن رمزشده متناظر با آن، همواره یکسان خواهد بود. به همین دلیل روش ECB برای پیام‌های طولانی ممکن است امن نباشد؛ چراکه اگر پیام به‌شدت ساختاریافته باشد، احتمال دارد مهاجم بتواند نشانه‌های آماری خوبی به‌دست آورده و در تحلیل رمز از آن‌ها استفاده نماید.

رمزگذاری



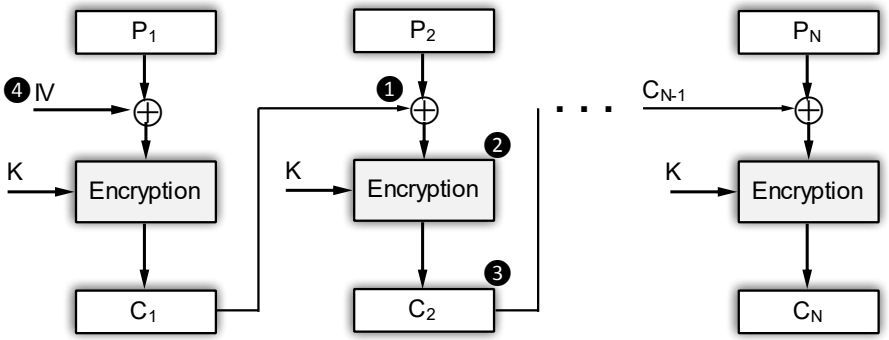
رمزگشایی



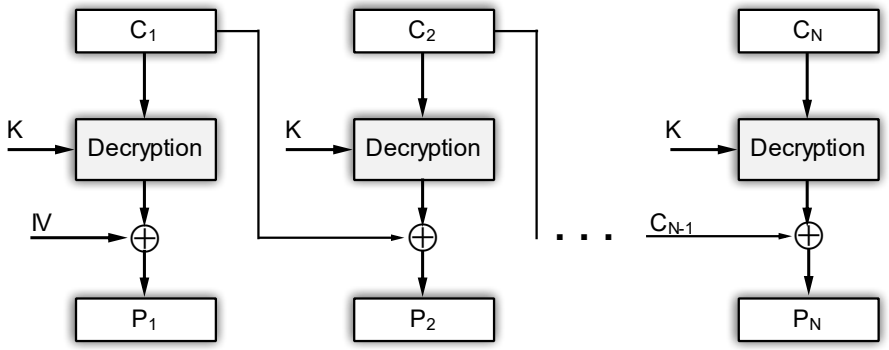
روش Cipher Block Chaining (CBC)

برای غلبه بر کاستی‌های امنیتیECB، به تکنیکی نیاز داریم که در صورت تکرار یک بلوک مشخص از دیتا در متن‌ساده، در متن رمزشده بلوک شبیه به هم نداشته باشیم. یک راه ساده برای برآورده کردن این نیاز، روش زنجیره‌ای شدن بلوک‌های رمزشده است.

رمزگذاری



رمزگشایی



همان‌طور که در تصویر فوق مشاهده می‌نمایید، در ورودی الگوریتم رمزگذاری، بلوک متن‌ساده با بلوک متن‌رمزشده مرحله قبل، XOR شده❶ و سپس عملیات Encryption❷ بر روی آن اعمال می‌شود. لذا درصورتی‌که یک بلوک متن‌ساده تکراری هم باشد، خروجی رمزگذاری❸ با قبل یکسان نخواهد بود. البته لازم به ذکر است که کلید مورداستفاده در همه زنجیره، یکسان است.

اما در مرحله اول که هیچ متن رمزشده‌ای از قبل وجود ندارد تا با متن‌ساده XOR شود، از مقدار Initialization Vector(IV)❹ استفاده می‌نماییم. IV یا بردار مقداردهی اولیه، یک بلوک دیتای

هم‌اندازه با بلوک متن‌رمزشده است که معمولاً به‌صورت تصادفی ایجاد می‌گردد. به دلیل اینکه IV برای رمزگشایی نیز موردنیاز است، باید هر دو طرف رمزگذار و رمزگشا، از مقدار آن اطلاع داشته باشند. اما برای حفظ امنیت نباید هیچ شخص ثالثی از آن اطلاع داشته باشد، لذا می‌توان برای جابجایی آن از روش ECB بهره برد.

برای ایجاد IV دو روش توصیه شده است:

۱. اول آنکه یک مقدارNonce[۱] به تابع رمزگذاری داده و بر اساس کلید مورداستفاده، یک خروجی از آن دریافت نمود؛ سپس این خروجی را به‌عنوان IV مورداستفاده قرار داد. توجه داشته باشید که Nonce باید یک بلوک منحصربه‌فرد باشد. به‌عنوان‌مثال از شمارنده، برچسب زمان[۲] یا شماره پیام، می‌توان جهت مقدارNonce استفاده نمود.
۲. روش دوم، استفاده از یک مولد عدد تصادفی[۳]، برای ایجاد بلوک‌های دیتای تصادفی است.

علاوه بر محرمانگی، از CBC برای احراز هویت نیز می‌توان بهره برد.



IV به معنی مقداردهی اولیه است؛ که غالباً به‌صورت تصادفی ایجاد می‌گردد. Nonce واژه‌ای است که برحسب نیاز در یک مقطعی از زمان ایجاد و استفاده‌شده و پس‌ازآن دیگر هیچ ارزشی ندارد. به عبارتی یک‌بارمصرف است.

روش Cipher Feedback (CFB)

روش CFB، در گروه روش‌های رمز جاری[۴] (نهری) قرار می‌گیرد. در این روش‌ها، دیگر نیازی به نگهداری شماره بلوک‌ها نیست؛ همچنین می‌توانند به‌صورت بلادرنگ[۵] کار نمایند. بنابراین برای انتقال جریانی از کاراکترها، می‌توان با استفاده از رمز جریانی کاراکترگرا[۶] (وابسته به کاراکتر)، هر کاراکتر را رمز نموده و بلافاصله انتقال داد. یکی از ویژگی‌های مطلوب رمز جاری این است که طول متن‌رمزشده با طول متن‌ساده برابر است.

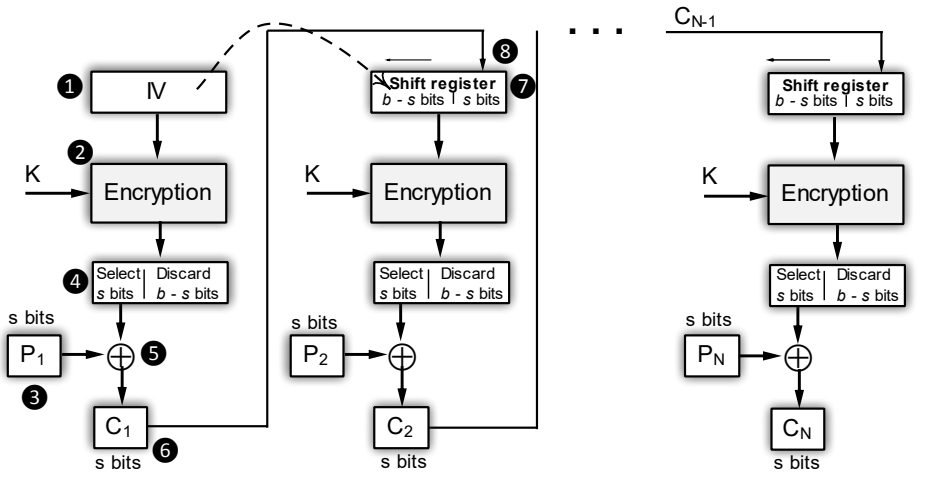
---

[۱] واژه یا عبارت موقتی / مقطعی / یک‌بارمصرف

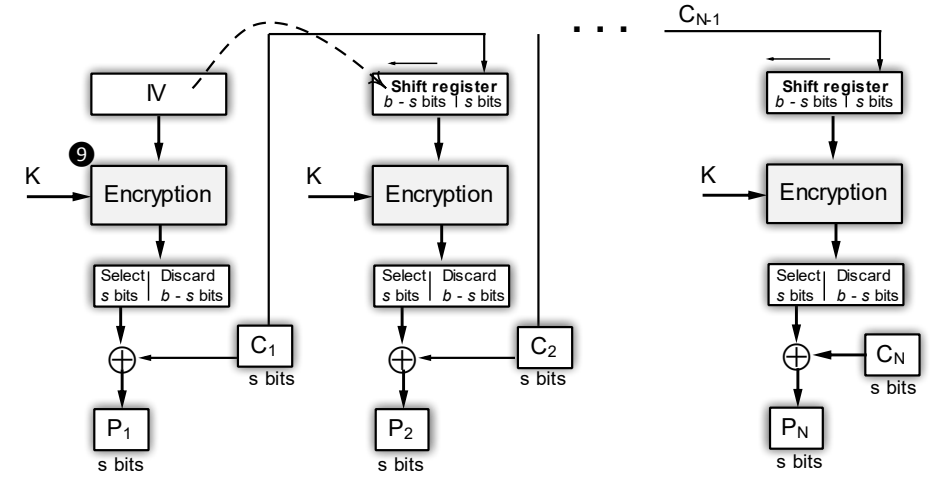
[2] Timestamp
[3] Random number generator
[4] Stream cipher
[5] Real-time
[6] Character-oriented

در CBC واحدهای متن‌ساده با یکدیگر زنجیر می‌شدند، به‌طوری‌که متن‌رمزشده هر واحد از متن‌ساده، تابعی از تمام متن‌های ساده قبلی بود. اما در CFB متن‌ساده به‌جای بلوک، به قطعات[1] بیتی تقسیم می‌شود.

رمزگذاری CFB



رمزگشایی CFB



[1] segments

در مرحله اول رمزگذاری، ابتدا یک IV❶ به الگوریتم رمزگذاری داده❷ و یک خروجی هم‌اندازه با IV می‌گیریم. به دلیل اینکه متن‌ساده ورودی دارای S-bit است❸، پس برای XOR نیاز به S-bit از خروجی Encryption داریم؛ لذا S-bit آن را از سمت چپ جدا کرده❹ و با S-bit از متن‌ساده XOR❺ می‌نماییم. خروجی به‌دست‌آمده یک متن رمزشده S-bitی است❻ که می‌توانیم آن را انتقال دهیم.

در ورودی مرحله بعدی رمزگذاری، یکShift Register❼ داریم؛ که در آنIV به‌اندازه S-bit به سمت چپ جابه‌جا شده تا b-s bit از آن در سمت چپ باقی بماند. سپس خروجی S-bitی رمزشده مرحله قبل را در سمت راست قرار داده❽ و بلوک به‌دست‌آمده را تحویل تابع Encryption می‌دهیم. این روال تا رمزشدن کامل متن موردنظر ادامه می‌یابد.

حالا برویم سراغ رمزگشایی. به دلیل اینکه متن‌ساده در مرحله رمزگذاری اصلاً داخل تابع Encryption قرار نگرفته و صرفاً با خروجی آن XOR شده؛ لذا در رمزگشایی نیز تابعی که باید مورداستفاده قرار گیرد، همان Encryption است.❾ چراکه در زمان رمزگشایی، ما مقادیر IV، کلید و متن رمزشده را در اختیار داریم؛ پس برای اینکه IV به همان شکلی که در رمزگذاری با متن ساده XOR شده بود❺ دربیاید، باید مجدد عملیات Encryption بر روی آن انجام پذیرد.

روش Output Feedback (OFB)

ساختار OFB بسیار شبیه CFB بوده و تنها دو تفاوت قابل‌توجه بین آن‌ها وجود دارد. اول اینکه در هر مرحله از OFB، یک بلوک (Block) از دیتای متن‌ساده در ورودی قرار می‌گیرد، درصورتی‌که در CFB دیتا به‌صورت قطعه (Segment) در ورودی پذیرفته می‌شود. دومین تفاوت نیز در نحوه استفاده از بازخورد (Feedback) است؛ در CFB متن رمزشده به‌عنوان بازخورد به مرحله بعد می‌رود، اما درOFB خروجی تابع (Encryption) مستقیماً به‌عنوان بازخورد در مرحله بعد مورداستفاده قرار می‌گیرد.

در مرحله اول رمزگذاری یک مقدار Nonce❶ به‌عنوان ورودی Encryption❷ قرار می‌گیرد. دلیل اینکه برای IV از مقدار Nonce استفاده می‌شود این است که مقدار Nonce برای هر پیام تغییر می‌نماید؛[۱] که در این صورت به ازای هر بار اجرای رمزگذاری، یک مقدار Nonce متفاوت خواهیم داشت.[۲]

---

[۱] توجه داشته باشید که پیام (Message) به Block یا Segment تقسیم می‌شود. مثلاً اگر بلوک یک الگوریتم ۶۴ بیت باشد، یک پیام ۱۲۸ بیتی به دو بلوک ۶۴ بیتی تقسیم می‌شود که برای این دو بلوک (P_1, P_2) یک مقدار Nonce در نظر گرفته و سپس برای پیام بعدی(که مثلاً ممکن است ۱۹۲بیت باشد) مقدار Nonce تغییر می‌کند.

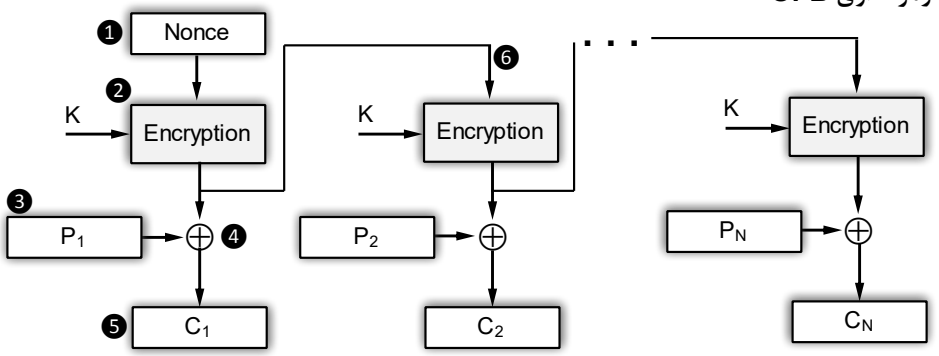
[۲] متغیر بودن مقدار Nonce برای هر پیام، باعث می‌شود تجزیه‌وتحلیل پیام‌های رمز شده برای مهاجم سخت شود.

طول بلوک Nonce باید با طول Plaintext برابر باشد. خروجی Encryption، با بلوک متن‌ساده❸ XOR شده❹ و متن رمزشده را تولید می‌نماید.❺

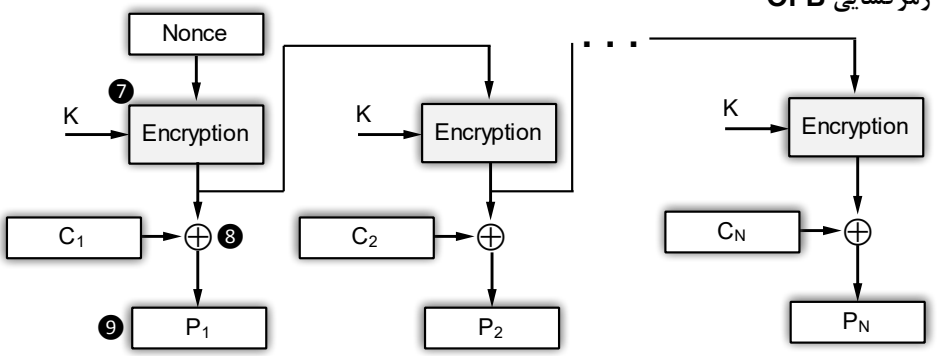
در ورودیEncryption مرحله دوم، به‌جای استفاده ازNonce، یک کپی از خروجی Encryption مرحله قبلی❻ قرار می‌گیرد. به همین دلیل است که به این روش، روش بازخورد خروجی می‌گویند. این فرآیند تا رمزشدن آخرین بلوک متن‌ساده ادامه می‌یابد.

اما در رمزگشایی نیز همانند CFB، بازهم از تابع Encryption استفاده می‌کنیم. چراکه در زمان رمزگشایی مقدار Nonce، کلید و متن رمزشده را در اختیار داریم و آن چیزی که نیاز است، خروجی Encryption بر اساس کلید و مقدار Nonce است. لذا مجدداً همان Encryption❼ بر روی Nonce انجام‌گرفته و با متن رمزشده XOR می‌شود؛❽ و نهایتاً در خروجی دارای متن ساده اولیه خواهیم بود.❾

رمزگذاری OFB



رمزگشایی OFB

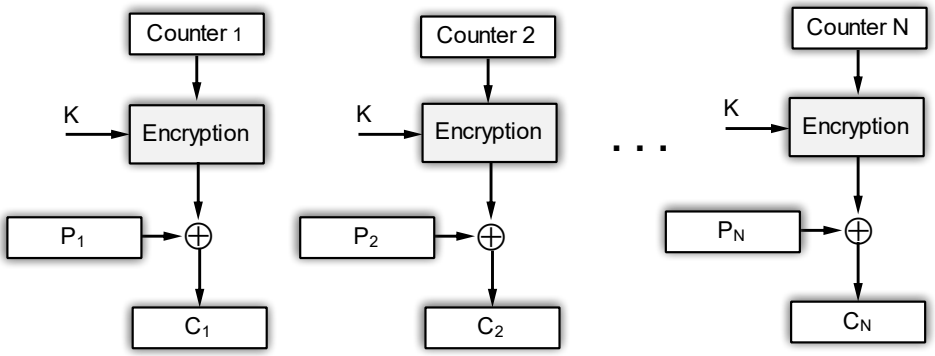


در CFB اگر حین انتقال، یک بیت دچار مشکل شود، ممکن است بلوک‌های بعدی را نیز تحت تأثیر قرار دهد. اما درروش OFB در صورت خرابی یک بیت در زمان انتقال، همان بلوک برای رمزگشایی دچار مشکل شده و بر سایر بلوک‌ها تأثیری ندارد. این مزیت به دلیل آن است که در زمان رمزگذاری، متن ساده و متن رمزشده هیچ دخالتی در بازخورد مورداستفاده ندارند.

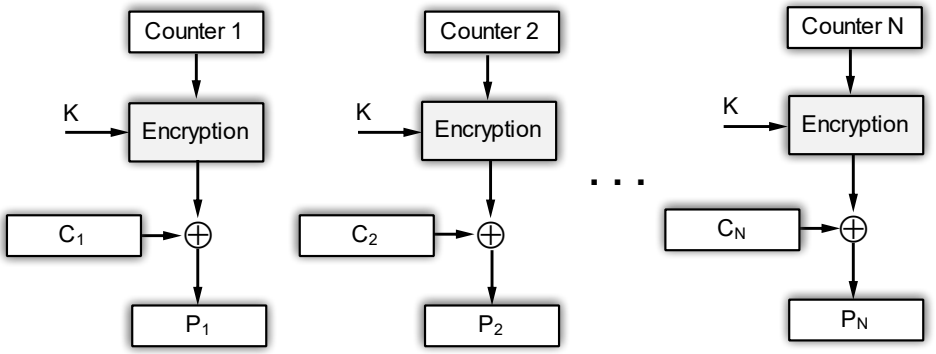
روش Counter (CTR)

روشCTR جهت امنیت شبکه‌های ATM[1] و IP Sec [2] توسعه داده شده است. نحوه عملکرد این روش، در تصویر زیر نمایش داده شده است.

رمزگذاری CTR



رمزگشایی CTR



[1] Asynchronous Transfer Mode
[2] IP Security

سایزCounter باید با سایز بلوک متن‌ساده برابر باشد. همچنین مقدارCounter به ازای هر بلوک از متن ساده باید متفاوت باشد. به همین دلیل مقدار Counter توسط یک مقدار اولیه مشخص‌شده و سپس برای هر بلوک بعدی، یکی به آن اضافه می‌شود. همان‌طور که ملاحظه می‌نمایید، در Counter عمل زنجیره‌ای شدن، برای هیچ‌یک از عناصر انجام نمی‌شود.

مزایای روش CTR

روش CTR نسبت به سایر روش‌ها، دارای مزایای زیر است:

- **بازده سخت‌افزاری**

در روش‌های زنجیره‌ای باید عملیات مربوط به هر بلوک تمام شود، تا شروع عملیات مربوط به بلوک بعدی امکان‌پذیر باشد. اما درروش CTR می‌توان به‌صورت موازی روی چند بلوک از متن‌ساده یا متن‌رمزشده کار کرد.

- **بازده نرم‌افزاری**

به‌طور مشابه با بازده سخت‌افزاری، امکان انجام موازی عملیات بر روی بلوک‌ها، بازده نرم‌افزاری را توسط پردازنده‌هایی که از ویژگی‌های موازی پشتیبانی می‌کنند، افزایش می‌دهد.

- **پیش‌پردازش**

الگوریتم رمزگذاری یا رمزگشایی برای شروع کار نیازی به ورود متن‌ساده یا متن رمزشده ندارد. لذا می‌توان از پردازشگر تا قبل از رسیدن متن‌ساده یا متن رمزشده، برای محاسبات اولیه بر روی Counter استفاده نمود؛ تا زمانی که متن‌ساده یا متن رمزشده رسید، فقط نیاز به یک عملیات XOR ساده باشد.

- **دسترسی تصادفی**

برای بعضی از برنامه‌های کاربردی، دسترسی تصادفی جذاب و کارآمد است. در CTR به دلیل وجود نداشتن رابطه زنجیره‌ای، امکان دسترسی تصادفی به بلوک‌های مختلف وجود دارد.

- **امنیت قابل‌اثبات**

روشCTR از سایر روش‌های گفته‌شده در این مبحث، امن‌تر است.

- **سادگی**

برخلاف روش‌های ECB و CBC، روش CTR فقط نیاز به الگوریتم Encryption داشته و از الگوریتمDecryption استفاده نمی‌نماید. نیاز نداشتن به الگوریتم رمزگشایی،

مخصوصاً زمانی ارزش خود را نشان می‌دهد که عملیات Decryption به‌طور قابل‌ملاحظه‌ای با عملیات Encryption تفاوت داشته باشد؛ مثل الگوریتم AES.
علاوه بر این نیازی به محاسبه کلید برای عملیات رمزگشایی، نیز نیست.

فصل پنجم

الگوریتم‌های رمزنگاری نامتقارن

مبحث اول

الگوریتم RSA

مبحث دوم

الگوریتم دیفی‌هلمن

☑ مبحث اول

الگوریتـم RSA

مشهورترین الگوریتم رمزنگاری نامتقارن، الگوریتم RSA است. تکنیک‌های اولیه آن در سال 1973 توسط آقای Clifford Cocks ایجاد گردید؛ ولی الگوریتم به‌صورت ساختاریافته و عملیاتی، در سال 1977 توسط سه دانش‌آموخته دانشگاه MIT، آقایان Ron Rivest، Adam Shamir و Len Adleman ارائه شد. نام الگوریتم RSA نیز برگرفته از حروف اول نام خانوادگی سه مخترع خود، یعنی Rivest-Shamir-Adleman (RSA) است.

الگوریتمRSA، یک الگوریتم رمزنگاری بلوکی نامتقارن (کلید عمومی) است. این الگوریتم از دو کلید با نام‌های کلیدخصوصی و کلیدعمومی، برای رمزگذاری و رمزگشایی استفاده می‌نماید. به دلیل خاصیت کلیدهای خصوصی و عمومی، این الگوریتم توانایی ارائه ویژگی‌هایی مثل احرازهویت و محرمانگی را نیز دارد. همچنین درصورتی‌که به همراه توابع درهم‌سازی مورد استفاده قرار گیرد، ویژگی‌هایی مثل امضای دیجیتال و سرویس عدم‌انکار را نیز فراهم می‌آورد.

الگوریتم RSA برای محاسبات کلید خود از ویژگی عدد اول استفاده نموده است. به‌طوری‌که بعداً در تشریح فرمول خواهید دید، برای قدرتمند کردن الگوریتم، نیاز به محاسبات اعداد اول بزرگ خواهیم داشت. الگوریتم RSA در طول سال‌هایی که از اختراع آن می‌گذرد، بدون تغییر و همواره به‌عنوان یک الگوریتم قابل‌اطمینان مورداستفاده قرارگرفته است. اما با توجه به تحولات سخت‌افزاری و پیشرفت قابل‌ملاحظه پردازشگرها، آن چیزی که این الگوریتم را در برابر شکست مصون نگه می‌دارد، مؤلفه‌های قدرتمندی است که باید بر اساس عدد اول، انتخاب‌شده و در محاسبات مورداستفاده قرار گیرند.

با توجه به اهمیت عدد اول ، مطالب زیر را جهت یادآوری عرض می‌کنم:

- **عدد اول یا Prime، عددی طبیعی و بزرگ‌تر از یک است که جز بر خودش و عدد ۱، بر هیچ عدد دیگری بخش‌پذیر نمی‌باشد.**
- **چونP عدد اول است، بنابراین تنها دو مقسوم‌علیه متمایز دارد.**
- **بینهایت عدد اول وجود دارد.**
- **هر عدد طبیعی بزرگ‌تر از یک را می‌توان به شکل حاصل‌ضربی از اعداد اول نوشت.**



تشریح الگوریتم RSA

الگوریتم RSA با استفاده از تابع‌نمایی بیان می‌گردد. در RSA متن ساده در قالب بلوک رمز می‌شود؛ به‌طوری‌که هر بلوک دارای یک **مقدار** باینری کوچک‌تر از n است. به این معنا که **سایز** بلوک باید کمتر یا مساوی با $log_2(n) + 1$ باشد؛ در عمل سایز بلوک i بیت است که در آن $2^i < n \leq 2^{i+1}$ است.

رمزگذاری و رمزگشایی در RSA طبق فرمول‌های زیر انجام می‌پذیرد: (در این فرمول، متن ساده با M و متن رمزشده با C نمایش داده‌شده است.)

$$C = M^e mod\ n$$

$$M = C^d\ mod\ n = (M^e)^d\ mod\ n = M^{ed}\ mod\ n$$

هر دو طرف فرستنده و گیرنده باید از مقدار n مطلع باشند. ارسال‌کننده مقدار e را می‌داند، اما فقط دریافت‌کننده باید مقدار d را بداند. بنابراین مقدار کلید خصوصی شامل $PR = \{d, n\}$ و کلیدعمومی شامل $PU = \{e, n\}$ خواهد بود.

محاسبه کلید عمومی و خصوصی در RSA

برای محاسبه کلید عمومی و خصوصی باید مراحل زیر انجام پذیرد:

۱. دو عدد اول به نام‌های p و q انتخاب می‌کنیم؛ به‌طوری‌که مساوی هم نباشند. هرچه این اعداد اول انتخابی بزرگ‌تر باشند، قدرت رمزنگاری بیشتر خواهد شد.

p = عدد اول و q = عدد اول

درصورتی‌که $p \neq q$

۲. عددی به نام n که حاصل‌ضرب p و q است را محاسبه می‌کنیم:

$$n = p \times q$$

۳. محاسبه m را به‌صورت زیر انجام می‌دهیم:

$$m = \varphi(n) = (p-1)(q-1)$$

۴. عدد e را طوری انتخاب می‌کنیم که از m کوچک‌تر و نسبت به آن، اول باشد. تأکید می‌کنم که e باید نسبت به m اول باشد؛ یا به‌عبارتی‌دیگر، باید بزرگ‌ترین مقسوم‌علیه مشترک m و e، برابر یک باشد.

$$gcd\ (m, e) = 1; \quad 1 < e < m$$

۵. عدد d را طوری انتخاب می‌کنیم که باقیمانده حاصلضرب $d \times e$ تقسیم‌بر m برابر با یک باشد، که به زبان ریاضی می‌شود:

$$(d \times e) mod\ m = 1$$

اگر e را داشته باشیم و بخواهیم d را محاسبه نماییم، می‌توانیم از فرمول زیر بهره ببریم:

$$d \equiv e^{-1}\ (mod\ \varphi(n))$$

پس از انجام مراحل فوق، کلیدخصوصی و کلیدعمومی مشخص می‌شوند. کلیدخصوصی نزد صاحب کلید نگهداری شده و کلیدعمومی نیز در اختیار هرکسی که می‌خواهد با صاحب کلید ارتباط برقرار کند، گذاشته می‌شود.

مثال محاسبه کلید خصوصی و عمومی

برای اینکه نحوه ایجاد و استفاده از کلیدهای خصوصی و عمومی را بهتر درک کنید، یک مثال ساده به‌صورت زیر حل می‌نماییم.

طرف گیرنده که در اینجا آلیس است، قصد ایجاد کلیدعمومی و خصوصی دارد. پس از ایجاد کلید قرار است کلیدعمومی خود را در اختیار برادر گرامی(آقای باب) قرار دهد تا تبادل اطلاعات بین آن‌ها به‌صورت رمزشده انجام پذیرد.

پس در ابتدا آلیس باید مراحل زیر را جهت ایجاد کلید عمومی و خصوصی خود انجام دهد:

۱. انتخاب دو عدد اول:

$$p = 17 \qquad q = 11$$

۲. برای محاسبه n باید دو عدد اول مشخص‌شده را با یکدیگر ضرب نمود:

$$n = p \times q = 17 \times 11 = 187$$

۳. محاسبه $\varphi(n)$ که ما آن را m می‌نامیم:

$$m = (p-1)(q-1) = 16 \times 10 = 160$$

۴. حالا باید آلیس عددی را به‌عنوان e انتخاب نماید که نسبت به m اول باشد.

آلیس عدد 7 را انتخاب می‌نماید. چراکه عدد 7 نسبت به 160 اول است؛ یعنی بزرگ‌ترین مقسوم‌علیه مشترک 7 و 160 برابر با 1 است. $e = 7$

۵. در این مرحله آلیس باید عددی را به‌عنوان d مشخص نماید که کوچک‌تر از m بوده و همچنین باقیمانده حاصل‌ضرب آن در e ، تقسیم‌بر m برابر 1 باشد.
آلیس عدد 23 را انتخاب نموده است: $d = 23$
عدد 23 شرط اول را دارد: یعنی $23 < 160$
همچنین حائز شرط دوم نیز هست، یعنی:

$$(d \times e) mod\ m = 1$$
$$(23 \times 7) mod\ 160 = 1$$

پس از انجام مراحل فوق، مؤلفه‌های موردنیاز جهت کلیدعمومی و کلیدخصوصی به‌دست می‌آید. با توجه به محاسبات انجام‌گرفته، کلیدخصوصی برابر با $PR = \{d, n\} = \{23,187\}$ و کلید عمومی برابر با $PU = \{e, n\} = \{7,187\}$ می‌باشد.

حالا آلیس باید کلید عمومی خود را برای باب ارسال نماید. ارسال کلید می‌تواند به‌راحتی از طریق یک کانال ناامن انجام پذیرد و اصلاً اشکالی ندارد که هرکسی از آن اطلاع پیدا کند. اما آلیس باید از کلیدخصوصی خود به‌دقت محافظت نماید.

پس از تبادل کلیدعمومی، فرستنده می‌تواند متن موردنظر خود را طبق فرمول، رمز کرده و برای گیرنده ارسال نماید. متنی که توسط کلید عمومی رمزشده، تنها با کلید خصوصی مربوطه قابل رمزگشایی است؛ به همین دلیل به‌راحتی می‌تواند از طریق یک کانال ناامن برای گیرنده ارسال شود.

در این مثال فرض می‌کنیم که باب می‌خواهد یک پیام برای آلیس ارسال نماید. توجه داشته باشید که پیام باید کوچک‌تر از n باشد. اگر متن‌ساده موردنظر را 88 در نظر بگیریم، طبق فرمول رمزگذاری و کلیدعمومی، محاسبات زیر باید توسط باب بر روی متن ساده انجام پذیرد:

i. فرمول رمزگذاری در RSA به‌صورت زیر است:

$$C = M^e mod\ n$$

ii. مقدار متن‌ساده نیز به‌صورت زیر است:

$$M = 88$$

iii. کلید عمومی که آلیس در اختیار باب قرار داده نیز به‌صورت زیر است:

$$PU = \{e, n\} = \{7,187\}$$

همان‌طور که ملاحظه می‌کنید، مقدار n نیز در کلیدعمومی مشخص شده است.

iv. با توجه به موارد فوق، باب تمام مؤلفه‌های موردنیاز برای رمزگذاری را در اختیار دارد. لذا طبق فرمول رمزگذاری محاسبات زیر باید انجام پذیرد:

$$C = M^e mod\ n$$

$$C = 88^7 mod\ 187 = 11$$

پس از اعمال فرمول رمزگذاری بر روی متن‌ساده، متن رمزشده به دست می‌آید؛ که در اینجا برابر با ۱۱ می‌باشد. پس از محاسبه متن رمزشده، باب می‌تواند آن را از طریق کانال ناامن برای آلیس ارسال نماید.

پس‌ازآنکه آلیس متن رمزشده را دریافت نمود، باید از طریق فرمول رمزگشایی و با استفاده از کلیدخصوصی خود، آن را رمزگشایی نماید.

مراحلی که سمت آلیس برای رمزگشایی انجام می‌شود به‌صورت زیر خواهد بود:

i. فرمول رمزگشایی الگوریتمRSA به‌صورت زیر است:

$$M = C^d\ mod\ n$$

ii. مقدار متن رمزشده که به دست آلیس رسیده است:

$$C = 11$$

iii. کلید خصوصی آلیس که خودش آن را محاسبه کرده:

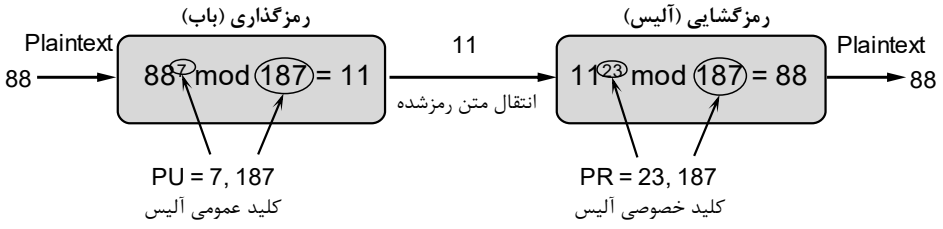
$$PR = \{d, n\} = \{23,187\}$$

iv. با توجه به موارد فوق، آلیس تمام مؤلفه‌های موردنیاز جهت جایگزینی در فرمول رمزگشایی را در اختیار دارد؛ لذا باید محاسبات زیر را انجام دهد:

$$M = C^d\ mod\ n$$

$$M = 11^{23}\ mod\ 187 = 88$$

همان‌طور که ملاحظه کردید، پس از جاگذاری مؤلفه‌ها در فرمول رمزگشایی، متن ساده اولیه به‌دست‌آمد. تصویر زیر فرآیند رمزگذاری و رمزگشایی در RSA را نمایش می‌دهد:



تولید کلید

قبل از استفاده از سیستم رمزنگاری کلیدعمومی، یک یا هر دو طرف باید اقدام به تولید جفت کلید خصوصی و عمومی نمایند. فرآیند تولید کلیدها شامل وظایف زیر است:

۱. مشخص کردن دو عدد اول p و q.
۲. انتخاب e و d و همچنین انجام محاسبات دیگر.

به دلیل اینکه مقدار n بر اساس حاصل‌ضرب دو عدد p و q به‌دست می‌آید، لذا ابتدا باید اقدام به مشخص کردن دو عدد اول نماییم. همان‌طور که در فرمول‌ها و مثال قبلی مشاهده نمودید، مقدار n طی کلیدعمومی در اختیار همه قرار می‌گیرد؛ پس باید دو عدد اول مربوطه، به‌قدری بزرگ انتخاب شده باشند که امکان تجزیه n به دو عدد اول تشکیل دهنده، برای فرد مهاجم وجود نداشته باشد.

در حال حاضر هیچ روش ریاضی کارامد و مفیدی برای محاسبه اعداد اول بزرگ وجود ندارد؛ لذا روش مرسوم این است که به‌صورت تصادفی یک عدد فرد را انتخاب نموده و سپس آن را آزمایش می‌کنند؛ که آیا عدد اول است یا نه. اگر عدد اول بود که مورد استفاده قرار می‌گیرد، در غیر اینصورت مجدداً اقدام به انتخاب تصادفی یک عدد فرد دیگر می‌نمایند.

به‌طور خلاصه فرآیند انتخاب یک عدد اول به‌صورت زیر است:

i. انتخاب تصادفی یک عدد صحیح فرد (n). به عنوان مثال می‌توان از یک مولد عدد شبه‌تصادفی[1] برای انتخاب این عدد استفاده نمود.
ii. انتخاب تصادفی یک عدد صحیح به عنوان a ، بطوریکه $a < n$ باشد.
iii. انجام آزمون بررسی عدد اول؛ مانند آزمون میلر–رابین[2].
اگر آزمون به شکست منجر شد، n را کنار گذاشته و دوباره به مرحله یک برمی‌گریدم.
iv. اگر به تعداد کافی در آزمون موفق بود، n را به عنوان عدد اول قبول می‌کنیم. در صورتی که تعداد آزمون کافی نبود به مرحله دوم برمی‌گردیم.

مراحل فوق جهت به دست آوردن عدد اول، شاید خسته کننده به نظر برسد؛ ولی توجه داشته باشید که شما به ندرت نیاز به انجام این مراحل دارید و آن‌هم زمانی است که می‌خواهید یک جفت کلید عمومی و خصوصی جدید ایجاد نمائید. پس از ایجاد جفت کلید جدید، برای مدتها می‌توانید از آنها استفاده نموده و خیرش را ببینید!

[1] pseudorandom number generator
[2] Miller-Rabin

پس از انتخاب اعداد اول، حالا باید مقادیر d و e را مشخص نمائید. شما می‌توانید برای d یک مقدار مشخص کرده و e را محاسبه نمائید و یا بالعکس؛ برای e یک مقدار مشخص نموده و d را محاسبه کنید.

عدد e را طوری انتخاب می‌کنیم که از m کوچکتر و نسبت به آن، اول باشد. به عبارتی دیگر، باید بزرگترین مقسوم علیه مشترک m و e ، برابر یک باشد.

$$gcd\,(m,e) = 1; \quad 1 < e < m$$

اگر بخواهیم d را بر اساس e محاسبه نماییم، می‌توانیم از فرمول زیر استفاده کنیم:

$$d \equiv e^{-1}\ (mod\ \varphi(n))$$

در نهایت با مشخص نمودن مقادیر e و d ، فرآیند تولید کلید کامل می‌شود.

امنیت RSA

برای حفظ امنیت الگوریتم RSA، باید ابتدا خطراتی که می‌تواند الگوریتم را تهدید کند، شناسایی نموده و سپس اقدامات متقابل[1] را برای جلوگیری یا کاهش آن‌ها انجام دهیم. متخصصین رمزنگاری، حملات احتمالی به RSA را در چهار گروه تقسیم‌بندی نموده‌اند. در ادامه ابتدا به تشریح حملات و سپس به اقدامات متقابل آن‌ها می‌پردازیم.

الف) حملات تهدید کننده RSA

حمله به الگوریتم RSA به چهار روش ممکن است انجام شود:

۱. **روش جستجوی فراگیر**

در این روش باید تمام کلیدهای ممکن جهت کلیدخصوصی مورد امتحان قرار گیرند.

۲. **حملات ریاضی**

سه روش مختلف در این حالت استفاده می‌شود:

i. پیدا کردن دو عدد اول بر اساس مقدار n. در اینصورت مهاجم قادر خواهد بود تا مقدار $\varphi(n)$ را محاسبه کرده و از طریق آن نیز به کلیدخصوصی(d) دست پیدا کند. $d \equiv e^{-1}\ (mod\ \varphi(n))$

ii. تعیین مستقیم $\varphi(n)$ بدون داشتن مقادیر p و q. در اینصورت بازهم می‌تواند کلیدخصوصی را محاسبه نماید.

iii. تعیین مستقیم d، بدون آنکه $\varphi(n)$ را به‌دست آورده باشد.

[1] Countermeasures

۳. **حملات زمانبندی**[1]

این نوع حملات به مقدار زمان اجرای الگوریتم رمزگشایی وابسته است. یک مشاور امنیتی به نام Paul Kocher، نشان داد که یک جاسوس با پیگیری اینکه یک سیستم چه مقدار زمان برای رمزگشایی صرف می‌کند، قادر به پیدا کردن کلیدخصوصی می‌باشد.[2] حملات زمانبندی به دو دلیل خطرناک می‌باشند: اول اینکه کاملاً غیرمنتظره روی می‌دهند؛ و دوم اینکه از نوع حملات Ciphertext-Only هستند.

۴. **حملات Chosen Ciphertext**

در این نوع حملات، از خواص الگوریتم RSA سوءاستفاده می‌شود. در حملات Chosen Ciphertext Attack که به اختصار CCA نیز نامیده می‌شود؛ فرد مهاجم می‌تواند تعدادی متن رمزشده دلخواه را در اختیار داشته باشد و با مقایسه آن با متن‌ساده، اقدام به تحلیل و کشف کلیدخصوصی نماید.

ب) اقدامات متقابل با حملات RSA

اقدامات متقابل با حملات احتمالی نیز در چهار گروه و به صورت زیر تقسیم بندی می‌شود:

۱. **مقابله با جستجوی فراگیر**

برای دفاع در برابر حملات جستجوی فراگیر، همانند سایر سیستم‌های رمزنگاری باید کلید قدرتمندتری انتخاب نمود. بنابراین هرچه تعداد بیت‌های d بیشتر باشد، بهتر است. اما باید توجه داشته باشید که هر چه کلید بزرگتر باشد، سرعت کم‌تر خواهد شد.

۲. **مقابله با حملات ریاضی**

جهت دفاع در برابر حملات ریاضی، علاوه بر بزرگ بودن مقدار n، در هنگام انتخاب مقادیر p و q باید به موارد زیر توجه شود:

i. طول p و q تنها با چند رقم باید باهم متفاوت باشند. بنابراین برای یک کلید ۱۰۲۴ بیتی (۳۰۹ رقمی)، مقدار p و q باید در محدوده 10^{75} تا 10^{100} قرار گیرد.

ii. هر دو مقدار $(p-1)$ و $(q-1)$، باید شامل یک عدد بزرگ اول باشند.

iii. $gcd(p-1, q-1)$ باید کوچک باشد.

۳. **مقابله با حملات زمانبندی**

اگر چه حمله زمانبندی، یک خطر جدی است؛ اما اقدامات متقابل ساده‌ای وجود دارد که می‌توان برای مقابله با آن انجام داد:

---

[1] Timing Attacks

[2] http://www.cryptography.com/public/pdf/TimingAttacks.pdf

i. زمان به‌توان‌رسانی ثابت

اطمینان حاصل نمائید که تمام عملیات به‌توان‌رسانی، همیشه یک زمان ثابتی را برای رسیدن به نتیجه صرف می‌کند.[۱] این کار ساده، خطر حملات زمانبندی را به‌شدت کاهش می‌دهد.

ii. تأخیر تصادفی

با ایجاد تأخیرات تصادفی در اجرای الگوریتم، می‌توانیم اطلاعات اشتباه زمانی برای فرد مهاجم ایجاد کنیم. البته آقای Kocher در این زمینه نیز اعتقاد دارد که اگر تعداد این تأخیرات تصادفی کافی نباشد، مهاجم می‌تواند آن‌ها را نیز اندازه‌گیری کند.

iii. اغفال‌کردن[۲]

اگر قبل از شروع عملیات به‌توان‌رسانی، متن رمزشده را در یک عدد تصادفی ضرب نماییم؛ مهاجم نمی‌تواند با دانستن طول متن رمزشده، تحلیل نماید که چه مقدار زمان برای توان‌رسانی آن صرف گردیده است.

۴. مقابله با حملات Chosen Ciphertext

برای غلبه بر حملات Chosen Ciphertext، باید از یک مقدار تصادفی قبل از رمزگذاری استفاده شود. به این معنا که متن ساده به همراه یک مقدار تصادفی، به متن رمزشده تبدیل شود.

---

[۱] به‌عبارت‌دیگر طول مدت‌زمان به توان‌رسانی ورودی‌ها، فارغ از اندازه آن‌ها، باید همواره یک مقدار ثابت باشد.

[2] Blinding

☑ مبحث دوم

الگوریتم دیفی‌هلمن

آقایان Whitfield Diffie و Martin Hellman یک زوج علمی مشهور و از بنیان‌گذاران رمزنگاری نامتقارن هستند. دیفی و هلمن، در رابطه با رمزنگاری نامتقارن دارای نظریه‌هایی هستند؛ اما در این مبحث به الگوریتم تبادل‌کلیدی[1] می‌پردازیم که آن‌ها بانام خودشان ارائه نموده‌اند.

الگوریتم دیفی‌هلمن یک الگوریتم ساده رمزنگاری نامتقارن می‌باشد که برای تبادل کلید بین فرستنده و گیرنده ارائه شده است. این پروتکل دو کاربر را قادر می‌سازد که یک کلید محرمانه را بر اساس الگوریتم کلیدعمومی، باهم به اشتراک گذارند.

در الگوریتم‌های رمزنگاری متقارن که هر دو طرف فرستنده و گیرنده باید از یک کلید برای رمزگذاری و رمزگشایی استفاده نمایند؛ بزرگ‌ترین چالش، تبادل و به اشتراک‌گذاری کلید بین طرفین ارتباط است. الگوریتم دیفی‌هلمن جهت تبادل کلید برای این نوع رمزنگاری می‌تواند مورداستفاده قرارگرفته و استفاده از رمزنگاری متقارن را تسهیل بخشد.

الگوریتم دیفی‌هلمن از لگاریتم گسسته برای سخت شدن محاسبات خود بهره می‌برد. ویژگی لگاریتم گسسته در آن است که محاسبه آن خیلی دشوار، اما بررسی جواب خیلی آسان است.

تشریح الگوریتم دیفی‌هلمن

در الگوریتم دیفی‌هلمن دارای دو مقدار q و α هستیم؛ به‌طوری‌که q عدد اول و α ریشه اولیه q است. هر دو طرف ارتباط باید از این دو مقدار اطلاع داشته و روی آن توافق کنند. نیازی نیست این اعداد محرمانه باشند، لذا به‌راحتی می‌توان مقادیر فوق را از طریق یک کانال ناامن به اطلاع طرفین رساند. هر چه مقدار q بزرگ‌تر باشد، امنیت الگوریتم بیشتر خواهد شد.

پس از توافق بر روی مقادیر فوق، طرفین اقدام به انجام محاسبات مربوط به کلید خصوصی و کلیدعمومی خود بر اساس q و α می‌نمایند. سپس کلیدخصوصی را به‌صورت محرمانه حفظ کرده و کلیدعمومی را در اختیار طرف مقابل قرار می‌دهند. در آخرین مرحله هر طرف ارتباط با استفاده از کلیدخصوصی خود و کلیدعمومی طرف مقابل، اقدام به محاسبه کلید اصلی می‌نمایند. درنهایت بدون اینکه نیازی به تبادل کلید باشد، هر دو طرف کلید را در اختیار خواهند داشت.

---

[1] Key Exchange

برای درک بهتر، عملیات انجام شده توسط دیفی‌هلمن را گام‌به‌گام شرح می‌دهیم. اگر طرفین ارتباط را آلیس و باب در نظر بگیریم، مراحل زیر به ترتیب توسط آن‌ها انجام می‌پذیرد:

۱. طرفین بر روی مقادیر q و α با یکدیگر توافق می‌نمایند.

۲. آلیس یک عدد صحیح تصادفی را بانام X_A، به صورتی که $X_A < q$ باشد، در نظر گرفته و سپس طبق فرمول زیر اقدام به محاسبه Y_A می‌نماید:

$$Y_A = \alpha^{X_A} \, mod \, q$$

همانند آلیس، باب نیز یک عدد صحیح تصادفی را بانام X_B در نظر می‌گیرد؛ به صورتی که $X_B < q$ باشد. سپس طبق فرمول زیر اقدام به محاسبه Y_B می‌نماید:

$$Y_B = \alpha^{X_B} \, mod \, q$$

۳. پس از انجام محاسبات فوق، طرفین مقدارX را به‌صورت خصوصی و محرمانه نزد خود نگه‌داشته و مقدارY را به‌صورت عمومی و از طریق کانال ناامن در اختیار طرف مقابل قرار می‌دهند.

۴. در این مرحله باید هر یک از طرفین طبق فرمول و با استفاده از کلیدعمومی طرف مقابل و کلیدخصوصی خود، اقدام به محاسبه کلید نمایند.

آلیس باید مقدار کلید را طبق فرمول زیر محاسبه نماید:

$$K_A = \ (Y_B)^{X_A} \, mod \, q$$

باب نیز باید مقدار کلید را طبق فرمول زیر محاسبه نماید:

$$K_B = \ (Y_A)^{X_B} \, mod \, q$$

در کمال شگفتی! خواهید دید که جواب هر دو محاسبه فوق برابر هم است:

$$K_A = \ K_B$$

درنهایت دو طرف کلید را دارند، بدون آنکه اقدام به تبادل آن کرده باشند. دلیل این شگفتی و برابر بودن نتیجه محاسبه کلیدها را می‌توانید در زیر مشاهده نمایید:

$$K_A = \ (Y_B)^{X_A} \, mod \, q$$

$$K_A = \ (\alpha^{X_B} \, mod \, q)^{X_A} \, mod \, q$$

$$K_A = \ (\alpha^{X_B} \ )^{X_A} \, mod \, q$$

$$K_A = \ \alpha^{X_B X_A} \, mod \, q$$

$$K_A = \underbrace{(\alpha^{X_A} \bmod q)}_{Y_A}^{X_B} \bmod q$$

$$K_A = (Y_A)^{X_B} \bmod q$$

$$K_A = \underbrace{(Y_A)^{X_B} \bmod q}_{K_B} \Rightarrow K_A = K_B$$

اما همان‌طور که در مراحل فوق ملاحظه کردید، برخی مقادیر مربوط به محاسبه کلید از طریق کانال ناامن تبادل شده و ممکن است که هرکسی به آن‌ها دسترسی پیداکرده باشد. اینجاست که سختی محاسبه لگاریتم گسسته به کمک الگوریتم آمده و کار را برای فرد رمزشکن دشوار می‌نماید. با فرض اینکه مهاجم مقادیر تبادل شده q ، α ، Y_A و Y_B را داشته باشد، برای رسیدن به کلید نیاز دارد که حداقل یکی از مقادیر خصوصی X_A یا X_B را نیز به دست آورد؛ که در این صورت مجبور است لگاریتم گسسته زیر را حل نماید:

$$X_A = \mathrm{dlog}_{\alpha,q}(Y_A)$$

$$X_B = \mathrm{dlog}_{\alpha,q}(Y_B)$$

مثال محاسبه کلید دیفی‌هلمن

برای درک بهتر الگوریتم دیفی‌هلمن، مثال عددی زیر را حل می‌کنیم:

۱. در گام اول طرفین ارتباط یعنی آلیس و باب برای ارقام q و α روی مقادیر زیر توافق می‌نمایند:

$$q = 353 \quad , \alpha = 3$$

۲. در گام دوم آلیس و باب باید برای خود یک مقدار تصادفی کوچک‌تر از q انتخاب نموده، سپس بر اساس آن اقدام به محاسبه مقدار عمومی Y نمایند.

آلیس عدد ۹۷ را به‌عنوان X_A انتخاب نموده و Y_A را محاسبه می نماید:

$$X_A = 97$$

$$Y_A = \alpha^{X_A} \bmod q = 3^{97} \bmod 353 = 40$$

باب عدد ۲۳۳ را به عنوان X_B انتخاب نموده و Y_B را محاسبه می نماید:

$$X_B = 233$$

$$Y_B = \alpha^{X_B} \bmod q = 3^{233} \bmod 353 = 248$$

۳. آلیس X_A و باب X_B را به‌صورت خصوصی نزد خود نگه‌داشته، و مقادیر Y_A و Y_B را با یکدیگر تبادل می‌نمایند.

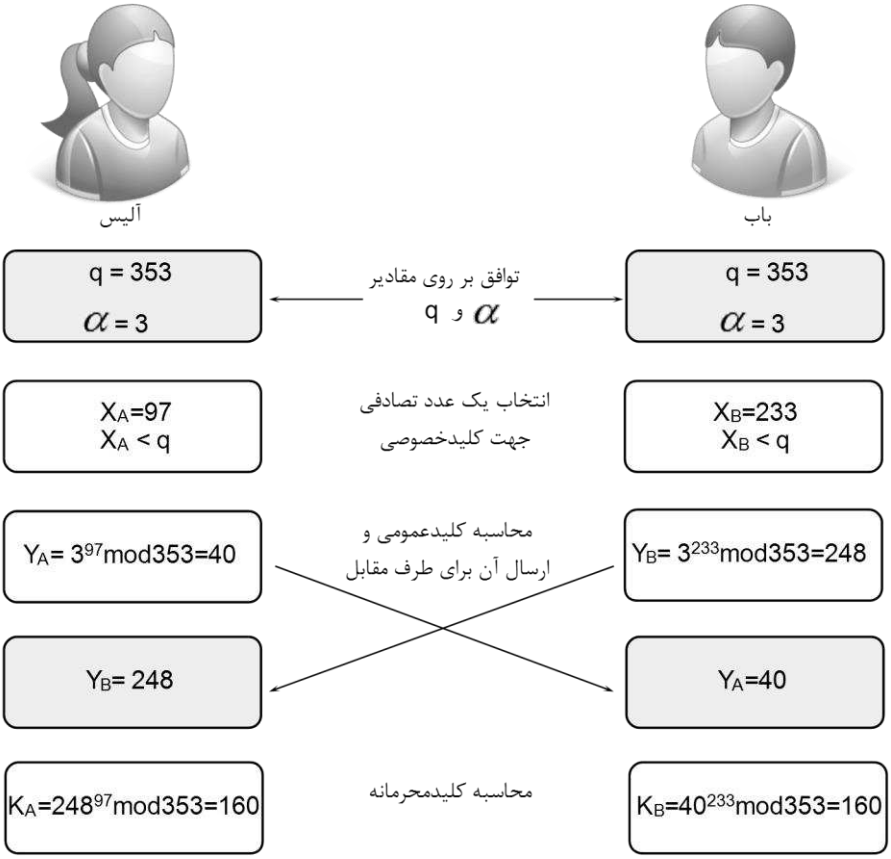
۴. حالا بر اساس داشته ها باید هریک از طرفین اقدام به محاسبه کلید نمایند.

$$K_A = (Y_B)^{X_A} \bmod q = 248^{97} \bmod 353 = 160$$

$$K_B = (Y_A)^{X_B} \bmod q = 40^{233} \bmod 353 = 160$$

همان‌طور که ملاحظه می‌نمایید، نتایج به‌دست‌آمده در هر دو طرف یکسان می‌باشد و آن‌ها ازاین‌پس می‌توانند از کلید محرمانه فوق برای رمزنگاری استفاده نمایند. البته لازم به تذکر است که اعداد انتخابی در مثال فوق خیلی کوچک در نظر گرفته‌شده تا مثال به‌راحتی حل شود، ولی در عمل عدد اول انتخابی جهت q باید خیلی بزرگ باشد تا قدرت الگوریتم دیفی‌هلمن حفظ شود.

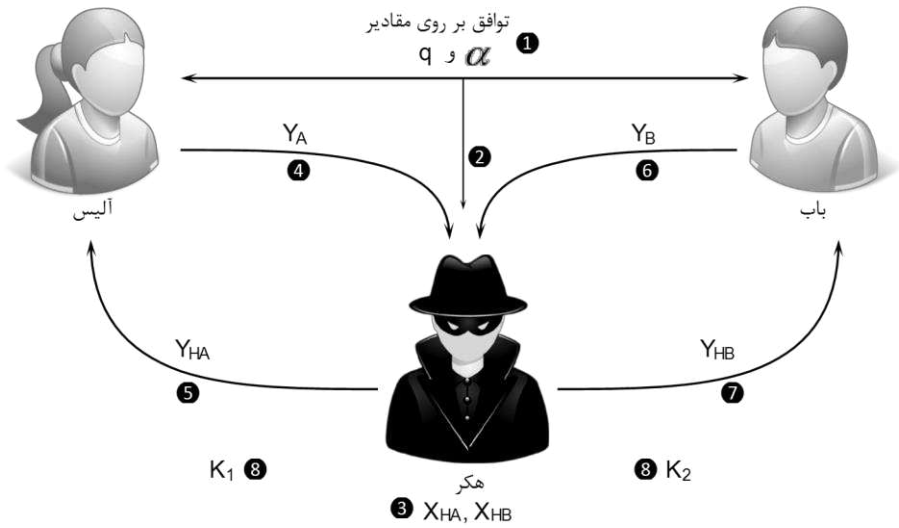
تصویر زیر مراحل الگوریتم دیفی‌هلمن را نمایش می‌دهد:



حمله مردمیانی

حمله مردمیانی یا Man-In-The-Middle که به‌اختصار MITM نامیده می‌شود؛ نوعی حمله استراق سمع فعال است. در این حمله مهاجم با هر یک از طرفین اتصال، ارتباط برقرار نموده و طوری عمل می‌نماید که طرفین تصور می‌کنند باهم در ارتباط هستند.

الگوریتم دیفی‌هلمن در مقابل حمله MITM آسیب‌پذیر نشان می‌دهد؛ چراکه هیچ احراز هویتی بین طرفین انجام نمی‌شود. مردمیانی می‌تواند با تولید کلیدخصوصی به‌ازای هر طرف ارتباط، اقدام به ایجاد کلیدمخفی برای هریک از آنان نماید. سپس پیام‌ها را از هر طرف گرفته و بعد از استراق‌سمع به‌طرف دیگر تحویل دهد. البته مهاجم می‌تواند علاوه بر استراق‌سمع، در پیام‌ها تغییر نیز ایجاد نماید. تصویر زیر حمله مردمیانی در الگوریتم دیفی‌هلمن را نمایش می‌دهد:



طبق روال آلیس و باب بر روی مقادیر q و α توافق می‌نمایند.❶ به دلیل اینکه توافق بر روی کانال ناامن انجام می‌شود، هکر به‌راحتی می‌تواند از آن مقادیر اطلاع پیدا کند.❷ فرد مهاجم به ازای هرکدام از طرفین یک مقدار تصادفی جهت کلیدخصوصی تعیین کرده (X_{HA},X_{HB})❸ و سپس بر اساس آن اقدام به محاسبه کلیدهای عمومی (Y_{HA},Y_{HB}) می‌نماید. آلیس نیز پس از محاسبه کلیدعمومی، آن را برای باب ارسال می‌نماید. ولی در بین راه مهاجم کلیدعمومی آلیس را دریافت کرده❹ و در جواب، کلیدعمومی خود را برای آلیس ارسال می‌نماید.❺ به دلیل اینکه احراز هویت انجام نمی‌شود، آلیس هم بی‌خبر از اتفاقات روی‌داده، کلیدعمومی دریافت شده را به‌عنوان کلیدعمومی باب در نظر می‌گیرد. از طرف دیگر، باب نیز کلید عمومی خود را محاسبه و

برای آلیس ارسال می‌نماید. اتفاق قبلی مجدداً روی می‌دهد؛ و هکر کلیدعمومی باب را دریافت نموده❻ و در جواب، کلیدعمومی خود را برای باب ارسال می‌نماید.❼ حالا هکر با اطلاعاتی که در اختیار دارد می‌تواند به ازای هر طرف یک کلیدمحرمانه تولید کرده($K_1 \neq K_2$) ❽ و ازاین‌پس تمام اطلاعاتی که بین آلیس و باب تبادل می‌شود را استراق سمع نموده و یا تغییر دهد.

آلیس با کلید K_1 اطلاعات خود را رمزگذاری نموده و به خیال خود! برای باب ارسال می‌نماید. هکر که کلید K_1 را در اختیار دارد، پیام را رمزگشایی کرده و پس از اطلاع یا تغییر پیام، مجدداً آن را با K_2 رمزگذاری نموده و برای باب ارسال می‌نماید. باب هم کلید K_2 را محاسبه کرده و در اختیار دارد، لذا می‌تواند پیام را رمزگشایی کرده و از آن استفاده نماید؛ و هر دو خوشحال که اطلاعات را به‌صورت رمزشده باهم تبادل نموده‌اند!

همان‌طور که گفته شد دلیل آسیب‌پذیر بودن الگوریتم دیفی‌هلمن در برابر حملات مردمیانی، نداشتن هیچ‌گونه مکانیسم احراز هویت است. برای رفع این مشکل می‌توان از امضای دیجیتال و یا گواهینامه کلیدعمومی استفاده نمود.

سیستم رمزنگاری الجمل[1]

در سال 1984 آقای طاهر الجمل اقدام به ارائه یک طرح رمزنگاری نامتقارن بر اساس الگوریتم دیفی‌هلمن نمود. رمزنگاری الجمل بانام ارائه‌دهنده آن و به‌صورت نگارشی ElGamal شناخته می‌شود. این الگوریتم ازآن‌جهت اهمیت دارد که در استانداردهای رمزنگاری دیگری ازجمله DSS[2]و S/MIME[3] مورداستفاده قرارگرفته است. هر چند طراحی این الگوریتم بر اساس دیفی‌هلمن است ولی نسبت به آن دارای تغییرات کوچکی می‌باشد که در ادامه به تشریح آن‌ها می‌پردازیم.

در گام اول، همانند الگوریتم دیفی‌هلمن، طرفین ارتباط باید بر روی مقادیرq و α با یکدیگر توافق نمایند؛ به‌طوری‌که q یک عدد اول بزرگ و α ریشه اولیه آن باشد. البته طرف شروع‌کننده ارتباط می‌تواند مقادیر فوق را انتخاب نموده و به اطلاع طرف مقابل برساند. به‌هرحال، توافق یا ارسال، بر روی کانال ناامن صورت می‌پذیرد.

در گام دوم، آلیس باید برای تولید کلیدخصوصی و کلیدعمومی خود مراحل زیر را انجام دهد:

۱. تولید یک عدد صحیح تصادفی به نام X_A، به‌طوری‌که $1 < X_A < q - 1$ باشد.
این عدد به‌عنوان کلیدخصوصی نزد آلیس نگهداری می‌شود.

---

[1] Elgamal
[2] Digital Signature Standard
[3] Secure/Multipurpose Internet Mail Extensions

۲. محاسبه مقدار Y_A بر اساس فرمول: $Y_A = \alpha^{X_A} mod\ q$.

مقدار Y_A به همراه مقادیر q و α به‌عنوان کلیدعمومی آلیس شناخته‌شده و باید در اختیار طرف مقابل قرار گیرد.

در گام سوم، پس‌ازآنکه کلیدعمومی آلیس در اختیار طرف مقابل (باب) قرار گرفت؛ باب باید یک پیام را به‌صورت زیر رمزنگاری نماید:

۱. پیام باید در قالب یک عدد صحیح که ما آن را M می‌نامیم ارائه شود؛ به‌طوری‌که $0 \leq M \leq q-1$ باشد. درصورتی‌که پیام بزرگ‌تر از مقدار q بود، باید آن را به بلوک‌هایی کوچک‌تر از q تقسیم کرد.

۲. تعیین یک مقدار صحیح تصادفی به نام k، به‌طوری‌که $1 \leq k \leq q-1$ باشد.

۳. محاسبه یک کلید یک‌بارمصرف توسط فرمول: $K = (Y_A)^k mod\ q$.

۴. رمزگذاری M به‌صورت یک جفت عدد صحیح، طبق فرمول‌های زیر:

$$C_1 = \alpha^k mod\ q$$
$$C_2 = KM\ mod\ q$$

خروجی رمزگذاری فوق شامل دو عدد صحیح C_1 و C_2 می‌باشد؛ که C_1 حاوی کلید یک‌بارمصرف و C_2 حاوی پیام موردنظر است.

در گام چهارم، پس‌ازآنکه آلیس مقادیر C_1 و C_2 را از باب دریافت نمود، باید مراحل زیر را جهت رمزگشایی انجام دهد:

۱. به‌دست آوردن کلید یک‌بارمصرف از طریق C_1 و توسط فرمول زیر:

$$K = (C_1)^{X_A}\ mod\ q$$

برای درک نحوه دستیابی به مقدار کلید یک‌بارمصرف بر اساس مقدار C_1، آنالیز فرمول را به‌صورت زیر مشاهده نمایید:

$$K = (Y_A)^k mod\ q$$
$$K = (\alpha^{X_A} mod\ q)^k mod\ q$$
$$K = \alpha^{kX_A}\ mod\ q$$
$$K = (C_1)^{X_A}\ mod\ q$$

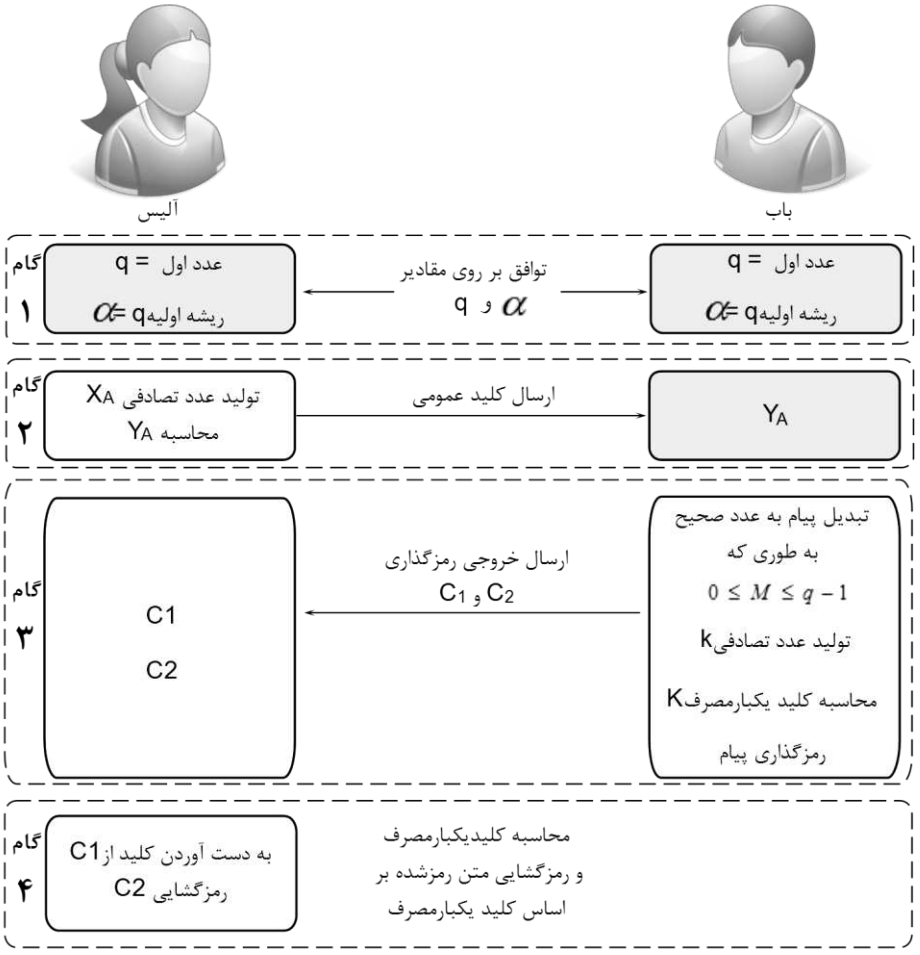
۲. محاسبه پیام توسط فرمول زیر از طریق C_2:

$$M = (C_2 K^{-1}) mod\ q$$

همان‌طور که ملاحظه می‌نمایید کلید یک‌بارمصرف مورداستفاده در رمزگذاری بر اساس کلیدعمومی آلیس ایجادشده، به همین دلیل دستیابی به آن فقط با کلیدخصوصی آلیس امکان‌پذیر است.

امنیت پروتکل الجمل نیز همانند دیفی‌هلمن، به دلیل سختی حل لگاریتم گسسته است. با توجه به امکانات سخت‌افزاری موجود، برای افزایش قدرت این پروتکل، باید عدد اول انتخابی، حداقل دارای ۳۰۰ رقم دسیمال باشد.

تصویر زیر چهار گام فرآیند پروتکل الجمل را نمایش می‌دهد:



فصل شش

توابع درهم‌سازی و کاربردها

مبحث اول

تصدیق پیام

مبحث دوم

امضای دیجیتال

مبحث سوم

الگوریتم SHA

☑ مبحث اول

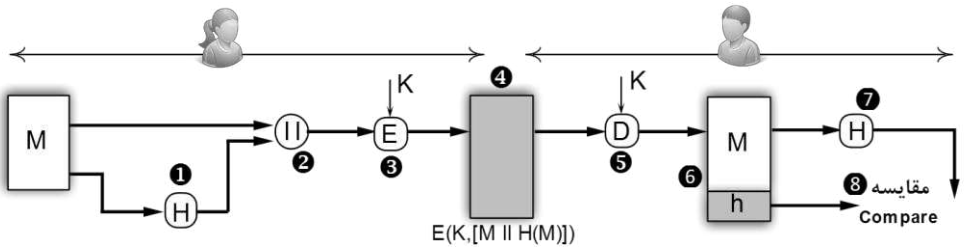
تصدیق پیام

توابع درهم‌سازی به‌تنهایی و همچنین در کنار الگوریتم‌های رمزنگاری متقارن می‌توانند امکان تصدیق پیام (Message Authentication) را فراهم آورند. با تصدیق پیام، گیرنده اطمینان حاصل می‌نماید که پیام دریافت شده دقیقاً همان چیزی است که ارسال‌شده؛ و در مسیر تبادل از خطراتی مثل تغییرات[1]، حذف[2]، درج[3] و بازپخش[4] مصون مانده است.

بسته به نوع و نحوه استفاده از الگوریتم رمزنگاری و نیازهای مدیر امنیت اطلاعات درزمینه محرمانگی و صحت، روش‌های اجرایی تصدیق پیام می‌تواند متفاوت باشد. در ادامه به تشریح چهار روش آن می‌پردازیم:

روش اول؛ تصدیق پیام به‌علاوه محرمانگی

درصورتی‌که بخواهیم ضمن تصدیق پیام، محرمانگی پیام نیز فراهم گردد، می‌توانیم از الگوریتم‌های رمزنگاری متقارن در کنار توابع درهم‌سازی به شکل زیر استفاده نماییم. اگر آلیس را ارسال‌کننده و باب را دریافت‌کننده در نظر بگیریم، مراحل به‌صورت زیر خواهد بود:



آلیس از پیام Hash گرفته❶ و آن را به متن اصلی الصاق می‌نماید.❷ سپس توسط یک الگوریتم رمزنگاری متقارن و بر اساس کلید توافق شده، متن ساده و Hash الصاقی را رمزگذاری نموده❸ و آن را برای باب ارسال می‌نماید.❹

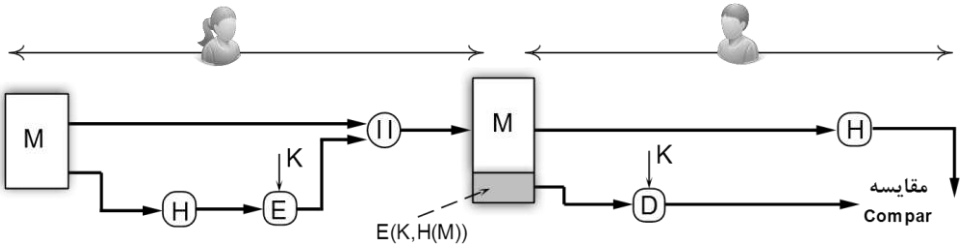
[1] Modification
[2] Deletion
[3] Insertion
[4] Replay

باب پس‌ازآنکه متن رمزشده ❹ را دریافت نمود، در اولین گام اقدام به رمزگشایی پیام طبق همان الگوریتم رمزنگاری متقارن مورداستفاده آلیس و کلید رمز توافق شده می‌نماید. ❺ پس از رمزگشایی، باب یک متن ساده به همراه مقدار Hash آن را در اختیار دارد. ❻ حالا باید برای بررسی صحت پیام، از متن ساده Hash گرفته ❼ و نتیجه را با Hash الصاقی، مقایسه نماید. ❽

همان‌طور که در تصویر فوق ملاحظه می‌کنید، پیام و Hash الصاقی به‌صورت کامل رمزگذاری شده‌اند، لذا در این طرح علاوه بر تصدیق پیام که توسط مقایسه Hashها به دست می‌آید؛ ویژگی محرمانگی نیز توسط الگوریتم رمزنگاری فراهم گردیده است.

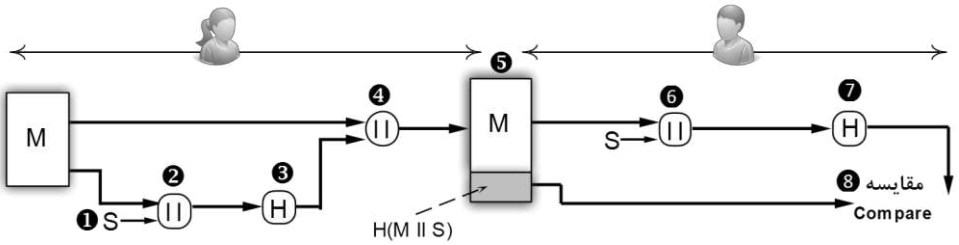
روش دوم؛ تصدیق پیام بدون محرمانگی

درصورتی‌که محرمانگی پیام مهم نباشد، می‌توان فقط مقدار Hash را رمزگذاری نمود تا تنها ویژگی تصدیق پیام فراهم گردد. اگر به یاد داشته باشید در مبحث چهارم از فصل سوم، مثالی جهت استفاده از توابع درهم‌سازی برای صحت و تمامیت ارائه گردید. در آن مثال گفتم که هکر می‌تواند هم پیام و هم مقدار Hash را تغییر دهد؛ چراکه گیرنده مکانیسمی جهت بررسی اصالت ارسال‌کننده ندارد. اما در این روش علیرغم اینکه محرمانگی برای ما اهمیتی ندارد، ولی چکیده پیام (مقدار Hash) توسط الگوریتم رمزنگاری متقارن به‌صورت رمزشده ارسال می‌شود. گیرنده پیام هم پس از رمزگشایی چکیده دریافتی و مقایسه آن با Hash پیام، می‌تواند از صحت پیام رسیده اطمینان حاصل نماید.



روش سوم؛ تصدیق پیام بدون استفاده از الگوریتم متقارن

اگر محرمانگی مهم نباشد می‌توان بدون استفاده از الگوریتم رمزنگاری متقارن که باعث افزایش سربار پردازشی می‌شود اقدام به تصدیق پیام نمود. در این روش یک مقدار محرمانه که بین فرستنده و گیرنده توافق گردیده، به پیام افزوده‌شده و سپس از پیام Hash گرفته می‌شود. چون مقدارمحرمانه بین فرستنده و گیرنده منتقل نمی‌شود، قاعدتاً هکر از آن اطلاع نخواهد داشت.



قبل از انجام هر کاری، باید یک مقدار محرمانه S ❶ بین فرستنده و گیرنده توافق شود. توافق بر روی این مقدار مثل توافق بر روی کلیدمحرمانه در رمزنگاری متقارن است؛ به‌عبارت‌دیگر مقدارS باید از طریق یک کانال امن به توافق طرفین رسیده باشد.

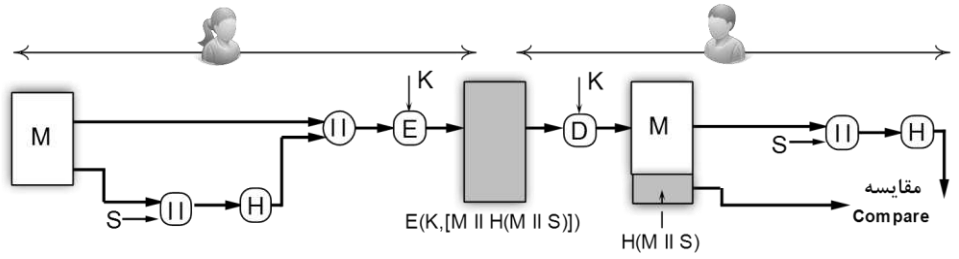
پس از توافق بر روی S، آلیس(فرستنده) باید S ❶ را به متن اصلی پیام الصاق نموده❷ و از پیامی که مقدارS به آن الصاق شده، Hash بگیرد.❸ سپس مقدار Hash را به پیام اصلی الصاق نموده❹ و برای باب(گیرنده) ارسال می‌نماید.❺

باب یک متن ساده به همراه یک مقدار Hash که حاصل از الصاقS به متن ساده است، دریافت می‌نماید.❺ باب S را به متن اصلی الصاق نموده❻ و از آن Hash می‌گیرد.❼ درنهایت Hash به‌دست‌آمده را با مقدار Hash دریافتی مقایسه می‌نماید. ❽

به دلیل اینکه هکر از مقدار محرمانه S اطلاع ندارد، لذا در صورت تغییر پیام نمی‌تواند Hash را طوری محاسبه نماید که مقدار S در آن دخیل باشد.

روش چهارم؛ تصدیق پیام با مقدار S به‌علاوه محرمانگی

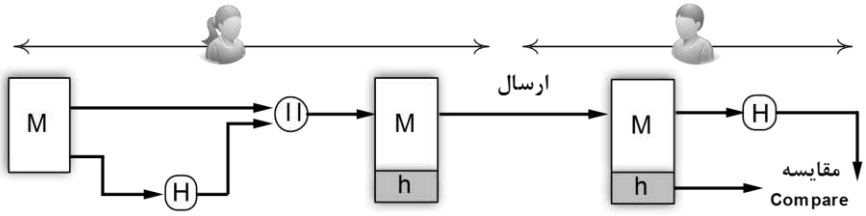
اگر بخواهیم تمام مزایای سه روش قبل را یکجا داشته باشیم، می‌توانیم به‌صورت زیر عمل نماییم. در این حالت همچنان از مقدار محرمانه S جهت تصدیق پیام استفاده می‌شود، اما به دلیل اهمیت محرمانگی از رمزنگاری متقارن نیز بهره می‌بریم.



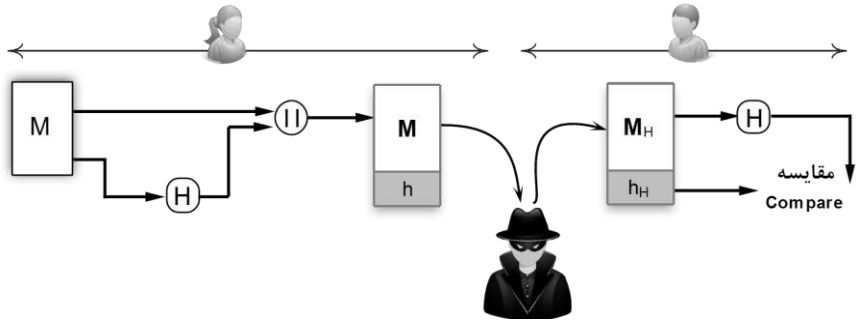
الگوریتم MAC

الگوریتم کدتصدیق‌پیام یا MAC (Message Authentication Code) یک کلیدمخفی به همراه یک پیام با طول متغیر را در ورودی دریافت کرده و یک کد تصدیق پیام در خروجی ارائه می‌نماید.

الگوریتم MAC بدون استفاده از الگوریتم‌های رمزنگاری، امکان احراز هویت را برقرار می‌سازد. همان‌طور که در این مبحث و مبحث امضای دیجیتال می‌بینید، با استفاده از الگوریتم‌های رمزنگاری متقارن و نامتقارن، می‌توان احراز هویت فرستنده را در سطوح مختلف فراهم آورد. اما اگر بخواهیم از توابع درهم‌سازی به‌تنهایی برای تصدیق پیام استفاده کنیم، فرآیندی به‌صورت زیر خواهیم داشت:



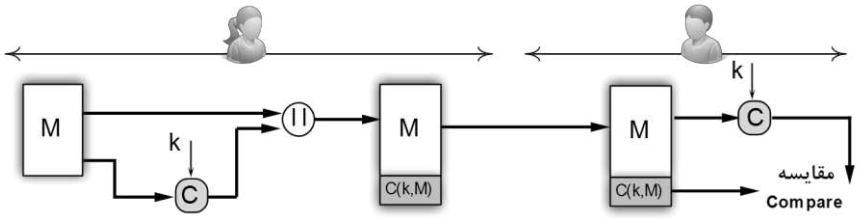
همان‌طور که مشاهده می‌کنید Hash به همراه متن اصلی بدون هیچ مکانیسم امنیتی بر روی کانال ناامن تبادل می‌گردد. در این صورت هکر به‌راحتی می‌تواند پیام و Hash الصاقی را دریافت نموده و با پیام موردنظر خود جابه‌جا نماید.



هکر همان‌طور که می‌تواند پیام را در مسیر تبادل تغییر دهد، مقدار Hash آن را نیز تغییر داده و جایگزین مقدار اصلی می‌نماید. یعنی هکر پیام و Hash الصاقی را دریافت نموده و پیام موردنظر خود را به همراه Hash آن برای گیرنده ارسال می‌نماید. گیرنده نیز Hash متن جعلی را باHash جعلی دریافتی، مقایسه کرده و تصور می‌کند پیام همانی است که فرستنده ارسال کرده است!

و اما MAC! شاید محرمانگی پیام برایمان اهمیتی نداشته باشد و نخواهیم از الگوریتم‌های رمزنگاری متقارن یا نامتقارن استفاده نماییم؛ ولی غالباً صحت و تمامیت پیام برایمان اهمیت داشته و می‌خواهیم پیام بدون نقص و به‌درستی به دست گیرنده برسد. در این صورت می‌توانیم تابع MAC را به همراه یک کلیدمحرمانه به متن اصلی اعمال نموده و یک کدتصدیق پیام تولید نماییم. سپس کدتصدیق‌پیام را به متن اصلی الصاق کرده و برای گیرنده ارسال کنیم.

عملیات انجام‌شده توسط MAC شبیه عملیات رمزنگاری متقارن است، با این تفاوت که در MAC نیازی به برگشت‌پذیری نداریم و می‌خواهیم شبیه توابع درهم‌ساز، یک خروجی یک‌طرفه (One-Way) را تولید نماییم. فرآیند MAC به‌صورت زیر انجام می‌پذیرد. در این تصویر تابع MAC با حرف C نمایش داده می‌شود:



همان‌طور که ملاحظه می‌نمایید، تابع MAC از یک کلیدمحرمانه بانام k استفاده نموده است. همانند الگوریتم رمزنگاری متقارن، مقدار کلیدمحرمانه باید از طریق یک کانال امن بین فرستنده و گیرنده توافق شده باشد.

در زمان استفاده از MAC هکر می‌تواند پیام و کدتصدیق‌پیام را دریافت کرده و حتی پیام را تغییر دهد؛ ولی به علت نداشتن کلیدمحرمانه، قادر به تولید کدتصدیق برای پیام جعلی خود نخواهد بود. لازم به ذکر است به دلیل اینکه کد تولیدشده ماهیتی یک‌طرفه داشته و برگشت‌پذیر نیست، لذا هکر با داشتن کدتصدیق، توانایی محاسبه کلید را نخواهد داشت. اما با توجه به اینکه هکر می‌تواند متن ساده و کدتصدیق‌پیام را به‌راحتی دریافت نماید، خطری که الگوریتم MAC را تهدید می‌کند، حملات جستجوی‌فراگیر است. یعنی مقادیر موردنظر را به‌عنوان کلید به متن اعمال نموده و برای آن کدتصدیق پیام تولید می‌کند؛ و این کار را تا جایی ادامه دهد که کدتصدیق اصلی با کدتصدیق جدید، مطابقت نماید؛ در این صورت مهاجم توانسته کلید را به دست آورد.

برای مقابله با حملات جستجوی‌فراگیر، باید از یک کلید بزرگ استفاده نمود. این کار باعث می‌شود خروجی تابع MAC قدرتمند شده و در برابر حملات آسیب‌پذیری کمتری داشته باشد.

درصورتی‌که محرمانگی پیام نیز برای شما اهمیت داشته باشد، می‌توانید از MAC در کنار الگوریتم‌های رمزنگاری متقارن و رمزنگاری نامتقارن بهره ببرید.

توابع HMAC[1]

استفاده از MAC برای تصدیق پیام، یک روش بهینه نسبت به سایر روش‌هایی محسوب می‌شود که از الگوریتم‌های رمزنگاری دیگر در کنار Hash استفاده می‌کنند. دلایل افزایش رغبت به استفاده از MAC نسبت به رمزنگاری‌های متقارن و نامتقارن را می‌توان در قالب دو مورد زیر بیان نمود:

۱. توابع رمزنگاری درهم‌سازی به‌طورکلی سریع‌تر از الگوریتم‌های رمزنگاری متقارن مثل DES هستند. (و بالطبع سریع‌تر از رمزنگاری نامتقارن)
۲. نسبت به سایر الگوریتم‌های رمزنگاری، کدهای موردنیاز برای توابع رمزنگاری درهم‌سازی به‌طور گسترده‌ای در دسترس هستند.

اما به دلیل ساختار توابع درهم‌سازی، یک مشکل در سر راه استفاده از MAC وجود داشت. به دلیل اینکه طراحی توابع درهم‌ساز مثل SHA بر اساس استفاده از کلیدمحرمانه انجام نگرفته بود، لذا نمی‌توانستند به‌صورت مستقیم جهت برقراری ویژگی MAC مورداستفاده قرار گیرند. برای رفع این مشکل، طرح‌هایی پیشنهاد گردید تا بتوان یک کلیدمحرمانه را به یک تابع درهم‌ساز الحاق نمود. در این میان طرحی که بیشتر مورد استقبال قرار گرفت، طرح HMAC بود. طرح HMAC توسط IETF طی RFC 2104[2] و همچنین توسط NIST طی FIPS 198[3] به‌صورت استاندارد معرفی گردید. پس‌ازآنکه IETF، HMAC را به‌صورت استاندارد منتشر کرد، استفاده از آن را در برقراری IPsec اجباری نمود؛ همچنین HMAC را در بعضی پروتکل‌های اینترنتی مثل SSL نیز مورداستفاده قرار داد.

امنیت HMAC به تابع درهم‌ساز و طول کلید مورداستفاده بستگی دارد. در HMAC شما می‌توانید تابع درهم‌ساز مورداستفاده را به‌راحتی با یک تابع قوی‌تر جایگزین کنید. به‌عنوان‌مثال اگر درHMAC از SHA-1 استفاده می‌کنید و احتیاج به امنیت بیشتری دارید، می‌توانید SHA-3 را جایگزین آن نموده و از مزایای تابع قدرتمندتر بهره‌مند شوید.

[1] Hash-based Message Authentication Code (HMAC)
[2] https://www.ietf.org/rfc/rfc2104.txt
[3] http://csrc.nist.gov/publications/fips/fips198/fips-198a.pdf

☑ مبحث دوم

امضـــای دیجیتـــال

آیا فرستنده پیام همان کسی است که ادعا می‌کند؟ در فضای واقعی برای اطمینان از اینکه پیام یا نامه موردنظر توسط همان کسی نوشته‌شده است که ادعا می‌کند، از امضا استفاده می‌شود. ولی در دنیای دیجیتال برای اطمینان از شخص ارسال‌کننده چه تمهیدی باید اندیشید؟

در فضای مجازی یا دیجیتال، با استفاده از امضای دیجیتال، گیرنده پیام می‌تواند اطمینان حاصل نماید که ارسال‌کننده پیام همان کسی است که ادعا می‌کند. اما امضاها در فضای واقعی و دیجیتال با یکدیگر متفاوت هستند. در فضای واقعی معمولاً امضای یک نفر همواره یکسان است. مثلاً شما برای فرم استخدام همان امضایی را انجام می‌دهید که بر روی دسته‌چک خود استفاده می‌کنید. یا یک مدیر برای تمام نامه‌ها و دستورالعمل‌هایی که ابلاغ می‌کند، همیشه یک امضای مشخص دارد. اما در امضای دیجیتال، به ازای هر سند یا پیام، امضایی متفاوت انجام می‌گیرد؛ به‌طوری‌که این امضا کاملاً وابسته به سند موردنظر بوده و آن را نمی‌توان برای پیام یا سند دیگری مورداستفاده قرار داد.

یکی از کاربردهای مهم توابع درهم‌سازی، همکاری در تولید امضای دیجیتال است. درصورتی‌که از توابع درهم‌ساز در کنار رمزنگاری نامتقارن استفاده نماییم، می‌توانیم از مزیت امضای دیجیتال بهره‌مند شویم. دلیل استفاده از توابع درهم‌ساز برای ایجاد امضای دیجیتال، ایجاد وابستگی امضا به سند مربوطه است.

نحوه تعامل توابع درهم‌ساز با رمزنگاری نامتقارن بسیار شبیه روش‌های گفته‌شده در مبحث اول است، با این تفاوت که در امضای دیجیتال به‌جای کلید محرمانه از کلیدهای خصوصی و عمومی استفاده می‌شود.

در رمزنگاری متقارن، رمزگذاری و رمزگشایی با استفاده از یک کلید همسان انجام می‌شود؛ ولی در رمزنگاری نامتقارن کلید رمزگذاری با کلید رمزگشایی متفاوت است. یعنی اگر رمزگذاری با کلیدعمومی گیرنده انجام شود، تنها کلیدی که می‌تواند برای رمزگشایی مورد استفاده قرار گیرد، کلیدخصوصی گیرنده است؛ و بالعکس اگر رمزگذاری با کلیدخصوصی فرستنده انجام پذیرد، تنها با کلیدعمومی فرستنده می‌توان اقدام به رمزگشایی پیام نمود. در امضای دیجیتال از این خاصیت رمزنگاری نامتقارن در کنار تابع درهم‌سازی استفاده می‌شود. برای اجرای امضای دیجیتال نیز همانند تصدیق پیام روش‌های متفاوتی وجود دارد؛ که با توجه به نیاز سازمان،

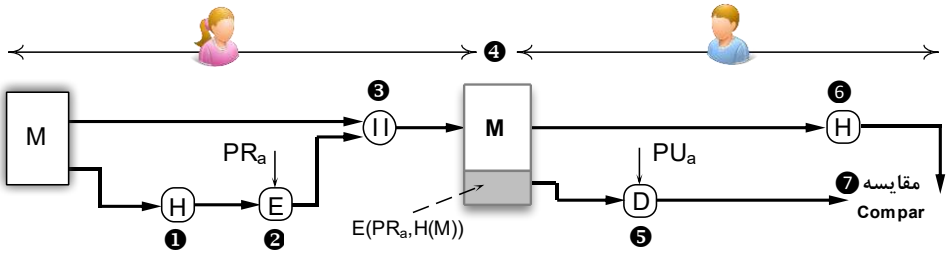
می‌توان از روش‌هایی که فقط کاربرد امضای دیجیتال دارند یا روش‌هایی که محرمانگی را نیز ارائه می‌دهند، استفاده نمود.

در ادامه به تشریح روش‌های اجرای امضای دیجیتال می‌پردازیم.

روش اول: امضای دیجیتال

این روش که ساده‌ترین روش ممکن برای ایجاد امضای دیجیتال است، از الگوریتم رمزنگاری نامتقارن در کنار تابع درهم‌ساز استفاده نموده است. این طرح فاقد محرمانگی است و فقط با توجه به استفاده از کلیدخصوصی فرستنده در رمزکردن Hash پیام، گیرنده می‌تواند از اصالت فرستنده اطمینان حاصل نماید.

آلیس توسط توابع درهم‌سازی از متن ساده Hash گرفته ❶ و از طریق الگوریتم رمزنگاری نامتقارن و با استفاده از کلیدخصوصی خود آن را رمزگذاری می‌نماید. ❷ سپس Hash رمزشده را به پیام اصلی الصاق کرده ❸ و آن را برای باب ارسال می‌کند. ❹



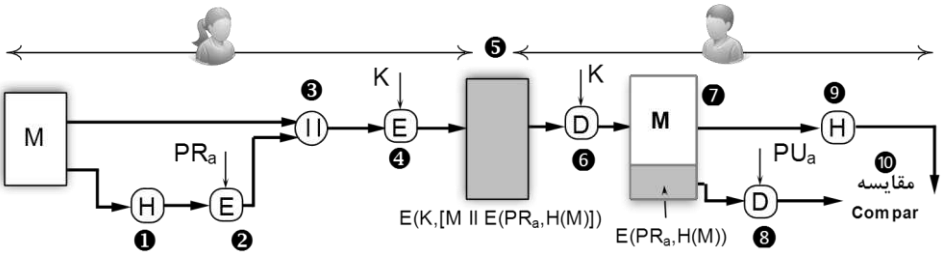
باب پیام را به همراه Hash رمزشده مربوطه دریافت می‌نماید. ❹ برای اینکه باب بتواند از مقدار Hash استفاده کند باید آن را رمزگشایی کند. با توجه به اینکه رمزگذاری با کلیدخصوصی آلیس انجام‌گرفته، تنها با استفاده از کلیدعمومی آلیس امکان رمزگشایی آن وجود دارد. این ویژگی باعث می‌شود باب اطمینان پیدا کند که مقدار Hash دریافتی، همان مقداری است که آلیس ارسال کرده؛ چراکه کس دیگری نمی‌تواند کلیدخصوصی آلیس را در اختیار داشته باشد. اما کلیدعمومی که برای رمزگشایی نیاز است، به‌صورت عمومی در اختیار همه قرارگرفته، پس باب به‌راحتی اقدام به رمزگشایی مقدار Hash الصاق شده نموده ❺ و سپس آن را با مقدار Hash پیام که خود محاسبه کرده ❻، مقایسه می‌کند. ❼

این طرح ویژگی تصدیق پیام و امضای دیجیتال را ارائه می‌دهد ولی هیچ مکانیسمی جهت محرمانگی اطلاعات تبادل شده، ندارد.

روش دوم: امضای دیجیتال به‌علاوه محرمانگی با رمزنگاری متقارن

طرح قبل فاقد محرمانگی بود، حتی اگر رمزگذاری با کلیدخصوصی را به‌جای مقدار Hash به‌کل پیام اعمال می‌کردیم، بازهم ویژگی محرمانگی را در اختیار نداشتیم؛ چراکه در صورت رمزگذاری با کلیدخصوصی، هرکسی کلیدعمومی را در اختیار داشته باشد می‌تواند پیام را رمزگشایی نماید. پس رمزگذاری با کلیدخصوصی جز اطمینان از ارسال‌کننده مزیت دیگری به دنبال ندارد.

در این طرح برای فراهم آوردن محرمانگی، از الگوریتم رمزنگاری متقارن استفاده می‌شود. به‌عبارت‌دیگر در این طرح هر سه حالت رمزنگاری متقارن، نامتقارن و توابع درهم‌سازی در کنار هم مورداستفاده قرار می‌گیرند تا تصدیق پیام، امضای دیجیتال و محرمانگی را فراهم آورند.



آلیس از پیام موردنظر Hash گرفته❶ و آن را با یک الگوریتم رمزنگاری نامتقارن توسط کلیدخصوصی خود رمزگذاری می‌نماید.❷ در مرحله بعد Hash رمزشده را به متن اصلی پیام الصاق می‌نماید.❸ سپس پیام و Hash رمزشده الصاقی را توسط الگوریتم رمزنگاری متقارن و کلیدی که قبلاً بین فرستنده و گیرنده توافق شده، رمزگذاری می‌نماید.❹ درنهایت یک پیام کاملاً رمزشده را برای گیرنده ارسال می‌نماید.❺

باب پس از دریافت پیام رمزشده❺، در مرحله اول توسط الگوریتم رمزنگاری متقارن و با استفاده از کلید توافق شده، اقدام به رمزگشایی پیام می‌نماید.❻ پس از رمزگشایی مرحله اول، باب یک متن ساده به همراه مقدار Hash پیام که به‌صورت رمزشده است را در اختیار دارد.❼ باب برای اینکه بتواند Hash دریافتی را رمزگشایی نماید، باید از کلیدعمومی آلیس استفاده کند.❽ نهایتاً باب از متن اصلی Hash گرفته ❾ و مقدار آن را با Hash الصاق شده به متن، مقایسه می‌نماید.❿

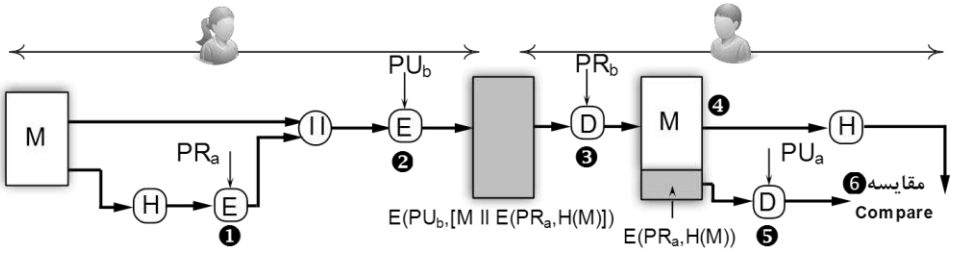
در این روش به دلیل رمزگذاری کامل پیام و چکیده توسط الگوریتم رمزنگاری متقارن و یک کلیدمخفی که فقط بین فرستنده و گیرنده توافق شده، ویژگی محرمانگی را در کنار تصدیق پیام و امضای دیجیتال خواهیم داشت.

روش سوم: امضای دیجیتال به‌علاوه محرمانگی با رمزنگاری نامتقارن

درروش قبلی برای فراهم آوردن محرمانگی، دست به دامان رمزنگاری متقارن شدیم. ولی در این روش علاوه بر امضای دیجیتال، برای محرمانگی نیز از رمزنگاری نامتقارن استفاده می‌کنیم.

برای فراهم آوردن ویژگی‌های امضای دیجیتال و احراز هویت، از کلیدخصوصی فرستنده استفاده می‌شود؛ ولی برای محرمانگی باید از کلیدعمومی گیرنده استفاده شود.

زمانی که پیام با کلیدعمومی گیرنده رمز می‌شود، رمزگشایی آن تنها با استفاده از کلیدخصوصی گیرنده امکان‌پذیر است. این ویژگی باعث می‌شود تا در زمان تبادل پیام، شخص ثالث نتواند پیام را رمزگشایی کرده و از محتوای آن مطلع شود.



در این حالت مراحل رمزگذاری و رمزگشایی مانند روش قبلی است. با این تفاوت که آلیس مقدار Hash پیام را با کلیدخصوصی خود رمز کرده❶ ولی پیام اصلی به همراه Hash الصاقی را با کلیدعمومی گیرنده (باب) رمزگذاری می‌کند.❷ در این روش هر دو الگوریتم رمزنگاری مورداستفاده از خانواده رمزنگاری نامتقارن هستند.

باب نیز تنها کسی است که می‌تواند پیام را رمزگشایی نماید؛ چراکه رمزگذاری نهایی در سمت فرستنده❷ با کلیدعمومی باب انجام‌گرفته است، پس قاعدتاً رمزگشایی اولیه فقط با کلیدخصوصی باب امکان‌پذیر خواهد بود.❸ پس‌ازآنکه باب پیام را رمزگشایی نمود، یک متن ساده به همراه یک چکیده رمزشده در اختیار خواهد داشت.❹ باب برای رمزگشایی چکیده باید از کلیدعمومی آلیس استفاده کند❺ و سپس مقدارHash دریافتی را با Hash پیام مقایسه نماید.❻

درصورتی‌که مقدار Hash محاسبه‌شده توسط باب با مقدار Hash ارسال‌شده توسط آلیس برابر باشد، باب مطمئن می‌شود که پیام دریافت شده دقیقاً همان چیزی است که توسط آلیس ارسال‌شده است. از طرف دیگر چون باب با کلیدعمومی آلیس موفق به رمزگشایی Hash گردیده، می‌تواند اطمینان داشته باشد که فرستنده قطعاً آلیس است.

الزامات امضای دیجیتال

برای ایجاد یک امضای دیجیتال قدرتمند و مقاوم در برابر حملات، باید الزامات زیر رعایت شوند:

- امضا باید یک الگوی بیتی وابسته به پیامی باشد که قرار است امضا شود.
- امضا باید از برخی مشخصات منحصربه‌فرد فرستنده، برای محافظت در برابر جعل و انکار، استفاده نماید.
- باید تولید امضای دیجیتال نسبتاً آسان باشد.
- فرآیند به رسمیت شناسی و بررسی امضای دیجیتال باید نسبتاً آسان باشد.
- باید محاسبات موردنیاز برای ایجاد یک امضای جعلی (چه ساخت پیام جعلی برای امضای موجود و یا ساخت امضای جعلی برای پیام موردنظر)، غیرممکن باشد.
- باید یک کپی از امضای دیجیتال نگهداری شود.

ویژگی عدم‌انکار

یکی از ویژگی‌های مهمی که می‌توان با استفاده از الگوریتم رمزنگاری نامتقارن در کنار توابع درهم‌سازی به دست آورد، ویژگی عدم‌انکار[1] است. این ویژگی در مواردی مثل محاکم قضایی و درزمانی که فرستنده پیام منکر ارسال آن می‌شود، کاربرد دارد.

ویژگی عدم انکار تنها در صورتی فراهم می‌شود که از رمزنگاری نامتقارن برای رمزگذاری مقدار Hash پیام استفاده کرده باشیم.

اگر از رمزنگاری متقارن یا MAC برای رمزکردن چکیده پیام استفاده کنیم، فرستنده به‌راحتی می‌تواند منکر ارسال پیام شود؛ چراکه هر دو طرف فرستنده و گیرنده از کلیدمحرمانه اطلاع دارند. لذا گیرنده به‌راحتی می‌تواند چکیده یک پیام جعلی را با کلیدمحرمانه رمزکرده و مدعی شود که طرف مقابل آن را ارسال کرده است. به همین دلیل برای اثبات هویت فرستنده، الگوریتم رمزنگاری متقارن و MAC در دادگاه وجاهت قانونی ندارند.

اما درصورتی‌که مقدارHash از طریق الگوریتم رمزنگاری نامتقارن و توسط کلیدخصوصی فرستنده رمزگذاری شده باشد، فرستنده پیام نمی‌تواند ارسال آن را منکر شود و ازلحاظ قضایی نیز مسئول پیام موردنظر است؛ چراکه هیچ کاربری جز صاحب کلید نمی‌تواند و نباید کلید خصوصی را در اختیار داشته باشد.

[1] Non-Repudiation

همان‌طور که قبلاً گفته شد، یکی از خصوصیات امضای دیجیتال نسبت به امضای سنتی، این است که امضای دیجیتال برای هر پیام و سندی به‌صورت منحصربه‌فرد و وابسته به همان متن تولید می‌شود. لذا به دلیل وجود Hash پیام و امضای آن توسط کلیدخصوصی، فرستنده حتی نمی‌تواند مدعی شود که امضای موردنظر را برای پیام دیگری انجام داده است.

ویژگی عدم انکار در مواردی مثل اتوماسیون اداری، پست الکترونیک و مخصوصاً تجارت الکترونیک و در زمان ارائه درخواست‌ها، صورتحساب، دریافت و پرداخت‌های مالی الکترونیکی دارای اهمیت فراوانی است.

استاندارد امضای دیجیتال (DSS)

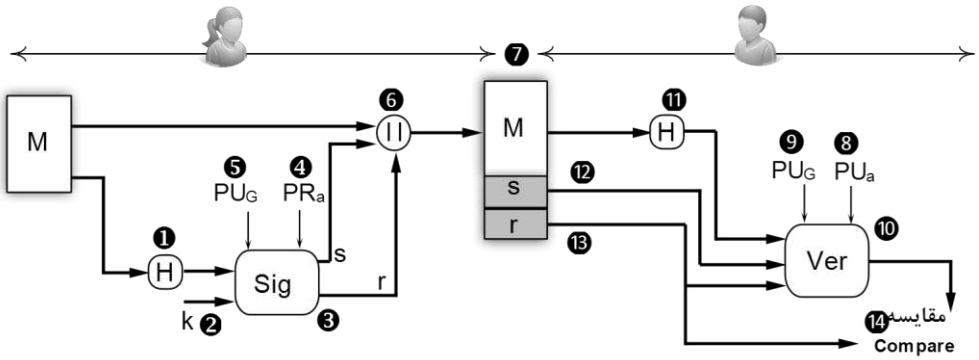
موسسه ملی استاندارد ایالات‌متحده (NIST)، برای اولین بار در سال 1991 استاندارد امضای دیجیتال (Digital Signature Standard) را طی سند FIPS 186[1] منتشر کرد. این سند در سال‌های بعد نیز با نگارش‌های جدید منتشر شد؛ آخرین نگارش آن در سال 2013 بوده که بانام FIPS 186-4 ارائه گردیده است.[2]

استاندارد DSS بر اساس الگوریتم درهم‌سازی SHA و الگوریتم رمزنگاری نامتقارن ElGamal اقدام به معرفی یک روش(تکنیک) جدید بانام الگوریتم DSA[3] نموده است. برخلاف RSA، الگوریتم DSA قادر به تبادل کلید و رمزگذاری نیست و تنها برای فراهم آوردن امضای دیجیتال ارائه گردیده است.

در روش‌هایی که تاکنون در این مبحث گفته‌شده، از الگوریتم رمزنگاری نامتقارن مثل RSA در کنار توابع درهم‌سازی برای تولید امضای دیجیتال استفاده‌شده است. به دلیل خاصیت کلیدخصوصی و کلیدعمومی، وقتی گیرنده می‌توانست Hash رمزشده را با کلیدعمومی فرستنده رمزگشایی کند، از اصالت فرستنده اطمینان حاصل کرده و اگر مقادیر Hash هم باهم برابر بود، مطمئن می‌گردید که امضای فرستنده دقیقاً برای همان پیام است.

در استاندارد امضای دیجیتال، به دلیل استفاده از الگوریتم DSA، فرآیند تولید امضای دیجیتال با روش‌های گفته‌شده متفاوت است. در DSA یک تابع امضا (Signature) در طرف ارسال‌کننده و یک تابع تائید (Verification) در سمت گیرنده داریم. تصویر زیر نحوه تولید امضای دیجیتال در استاندارد DSS را به‌صورت ساده نمایش می‌دهد:

[1] Federal Information Processing Standard
[2] http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.186-4.pdf
[3] Digital Signature Algorithm



ابتدا آلیس از متن پیام Hash گرفته ❶ و آن را همراه یک عدد تصادفی (k) ❷ که مختص همین امضا تولیدشده، به ورودی تابع Signature ❸ تحویل می‌دهد. تابع Signature به کلیدخصوصی فرستنده هم نیاز دارد، لذا باید از کلیدخصوصی آلیس استفاده نماید. ❹ همچنین تابع Signature به مجموعه‌ای از پارامترهای دیگر نیاز دارد که ما آن‌ها را کلیدعمومی‌سراسری[1] نامیده و با عبارت PU_G نمایش می‌دهیم. ❺

خروجی تابع Signature شامل دو جزء است که بانام‌های s و r شناخته می‌شوند. آلیس باید مقادیر s و r را به متن اصلی پیام الصاق نموده ❻ و نتیجه را برای گیرنده ارسال نماید. ❼

باب یک پیام دریافت می‌کند که مقادیر s و r به آن الصاق شده است. ❼ از طرفی به کلید عمومی آلیس ❽ و کلیدعمومی‌سراسری ❾ نیاز دارد که به‌راحتی می‌تواند آن را به دست آورد.

طرف گیرنده دارای تابعی به نام Verification است. ❿ وظیفه این تابع بررسی امضای دیجیتال دریافت شده، است. ورودی‌های این تابع شامل مقدار Hash شده پیام دریافتی ⓫، مقدار S ⓬ و مقدار r ⓭ الصاقی و همچنین کلیدعمومی فرستنده و کلیدعمومی‌سراسری است.

خروجی تابع Verification باید مقداری برابر با مقدار r باشد، لذا باب باید خروجی تابع را با مقدار r الصاق شده به پیام مقایسه کرده ⓮ و در صورت یکسان بودن مقادیر، می‌تواند اطمینان حاصل نماید که فرستنده پیام آلیس است.

همان‌طور که ملاحظه می‌کنید در این روش ویژگی محرمانگی فراهم نگردیده است؛ لذا درصورتی‌که می‌خواهید از ویژگی محرمانگی نیز بهره‌مند شوید، باید از یک الگوریتم رمزنگاری متقارن یا نامتقارن برای رمزنگاری کامل پیام استفاده نمایید.

[1] Global Public Key

الگوریتم DSA

الگوریتم DSA بر اساس طرح ElGamal و Schnorr طراحی گردیده است. سختی محاسبات DSA نیز همانند RSA و ElGamal بر پایه محاسبات لگاریتم گسسته است.

این الگوریتم دارای دو تابع Signature و Verification است که وظیفه تولید امضا در سمت فرستنده و تأیید امضای دیجیتال را در سمت گیرنده، بر عهده دارند.

در این الگوریتم از گروهی پارامترهای عمومی استفاده می‌گردد که بانام Global Public Key معرفی‌شده و با عبارت PU_G نمایش داده می‌شود. PU_G شامل سه پارامتر عمومی بانام‌های p ، q و g است که می‌تواند بین گروهی از کاربران به‌صورت مشترک وجود داشته باشد. محاسبه این سه پارامتر به‌صورت زیر انجام می‌پذیرد:

- **پارامتر q**

انتخاب یک عدد اول ۱۶۰ بیتی است؛ به‌طوری‌که باید مقسوم‌علیه $p-1$ بوده و در رنج $2^{159} < q < 2^{160}$ قرار گیرد.

- **پارامتر p**

انتخاب یک عدد اول با طول بین ۵۱۲ تا ۱۰۲۴ بیت؛ به‌طوری‌که q مقسوم‌علیه $p-1$ قرار گیرد:

اگر L را مضربی از ۶۴ و به‌صورت $512 \leq L \leq 1024$ در نظر بگیریم، p باید به‌صورت $2^{L-1} < p < 2^L$ ، قرار گیرد.

- **پارامتر g**

محاسبه g طبق فرمول زیر انجام می‌شود:

$$g = h^{(p-1)/q} \bmod p$$

که در آن h یک عدد صحیح به‌صورت $1 < h < (p-1)$ بوده و نتیجه فرمول فوق نیز باید به‌صورت $h^{(p-1)/q} \bmod p > 1$ ، یعنی بزرگ‌تر از یک باشد.

سه پارامتر فوق به‌عنوان کلیدعمومی‌سراسری در الگوریتم DSA مورداستفاده قرار می‌گیرند. الگوریتم DSA این سه پارامتر را از طرح امضای دیجیتال Schnorr اقتباس کرده است.

علاوه بر کلیدعمومی‌سراسری به سه پارامتر دیگر بانام‌های کلیدخصوصی کاربر، کلیدعمومی کاربر و مقدار تصادفی k برای توابع Signature و Verification نیازمندیم. طریقه محاسبه این پارامترها به‌صورت زیر است:

- **کلیدخصوصی**

انتخاب یک عدد صحیح تصادفی؛ به‌طوری‌که اگر کلیدخصوصی را x در نظر بگیریم، شرط: $0 < x < q$ ، برقرار باشد.

- **کلیدعمومی**

اگر کلید عمومی را y در نظر بگیریم، محاسبه کلیدعمومی بر اساس کلید خصوصی و طبق فرمول زیر انجام می‌پذیرد:

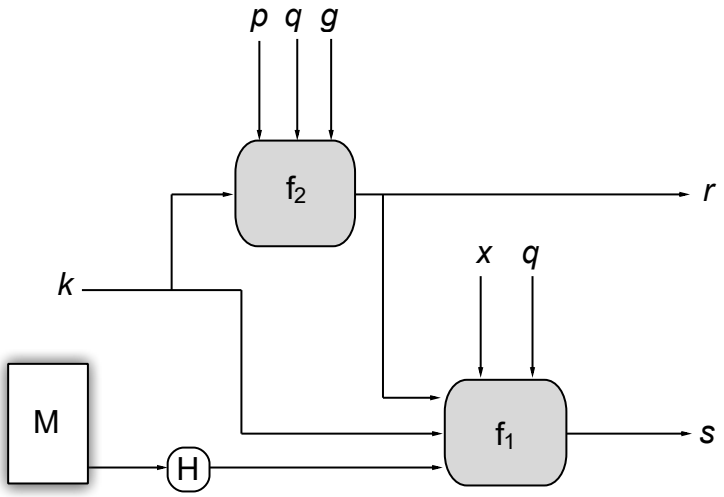
$$y = g^x \bmod q$$

- **مقدار تصادفی k**

k یک عدد صحیح تصادفی است که باید در رنج $0 < k < q$ قرار داشته باشد.

تابع Signature

برای ایجاد امضای دیجیتال، الگوریتم DSA دارای یک تابع سمت فرستنده است که آن را تابع امضا یا Signature می‌نامد. تصویر زیر عملیات انجام‌گرفته در این تابع را به‌صورت ساده نمایش می‌دهد:



همان‌طور که در شکل فوق مشاهده می‌نمایید، برای درک بهتر عملیات، تابع Signature به دو تابع f1 و f2 تقسیم‌شده است.

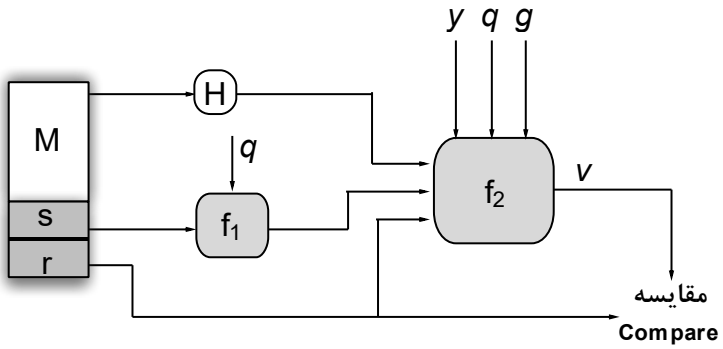
درنهایت خروجی تابع Signature دو مقدار بانام‌های s و r هستند که توسط فرمول‌های زیر محاسبه می‌شوند:

$$r = \left(g^k \bmod p\right) \bmod q$$

$$s = [k^{-1}\,(H(M) + xr)] \bmod q$$

تابع Verification

برای تأیید امضای دیجیتال، الگوریتم DSA دارای یک تابع سمت گیرنده است که آن را تابع تائید یا Verification می‌نامد. تصویر زیر عملیات انجام‌گرفته در این تابع را به‌صورت ساده نمایش می‌دهد؛ البته برای درک بهتر، این تابع نیز به دو قسمت تقسیم‌شده است:



محاسباتی که گیرنده بر اساس تابع Verification برای تأیید امضای دیجیتال انجام می‌دهد، به‌صورت زیر است. لازم به ذکر است که ما نام خروجی تابع Verification را v گذاشتیم، لطفاً شما هم v صداش کنید! ☺

$$w = (s)^{-1} \bmod q$$

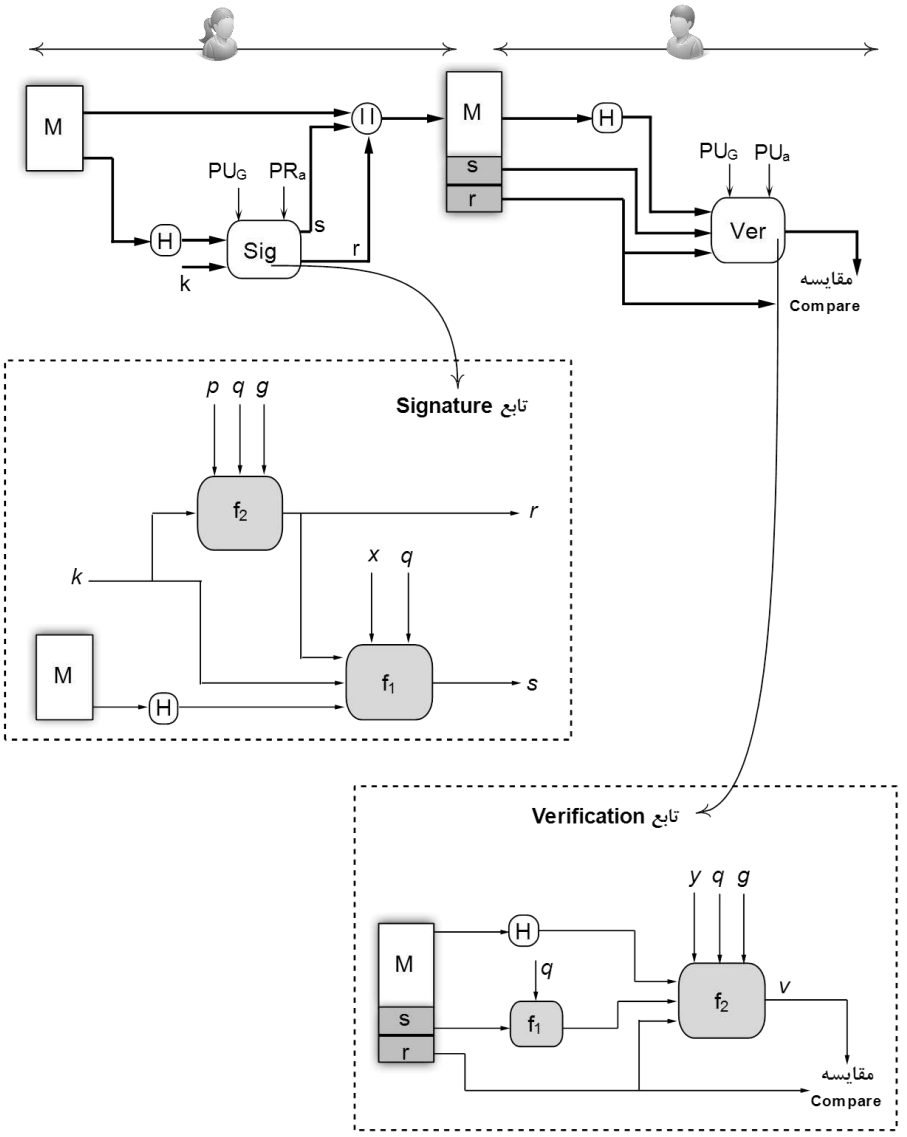
$$u_1 = [H(M)w] \bmod q$$

$$u_2 = rw \bmod q$$

$$v = [(g^{u_1} y^{u_2}) \bmod p] \bmod q$$

درنهایت گیرنده باید مقدار v را با مقدار r مقایسه نماید. درصورتی‌که این دو مقدار باهم برابر بود، گیرنده از اصالت فرستنده پیام اطمینان حاصل می‌نماید. در غیر این صورت پیام رسیده همان پیامی نیست که فرستنده ارسال و امضا کرده است.

در یک اقدام خلاقانه! جهت درک بهتر، تصویر کلی استاندارد DSS و توابع آن را به‌صورت زیر برای شما آماده کردم:



☑ مبحث سوم

الگـوریتم SHA

مشهورترین و پرکاربردترین الگوریتم توابع درهم‌سازی که در سال‌های اخیر مورداستفاده قرارگرفته، الگوریتم درهم‌سازی امن یا Secure Hash Algorithm است.

اولین بار در سال 1993 موسسه ملی استاندارد و فناوری ایالات‌متحده(NIST)، الگوریتم SHA را در قالب استاندارد FIPS 180 منتشر کرد. پس‌ازآنکه نقاط ضعف این نسخه از SHA که امروزه آن را بانام SHA-0 می‌شناسیم، کشف شد؛ موسسه NIST در سال 1995 اقدام به ارائه نسخه بهبودیافته آن بانام SHA-1 نمود. اسناد SHA-1 تحت عنوان Secure Hash Standard ارائه گردیده است. با توجه به اینکه طراحی SHA-1 بر اساس الگوریتم MD4 انجام‌گرفته بود، IETF نیز SHA-1 را با اعمال تغییرات جزئی، تحت استاندارد RFC 3174 منتشر ساخت. لازم به ذکر است که پنج سال قبل از ارائه SHA-1، یعنی در سال 1990 الگوریتم MD4[1] طی RFC 1320 توسط IETF به‌عنوان استاندارد معرفی‌شده بود.

در سال2002 موسسه NIST اقدام به انتشار سه نسخه جدید از SHA در قالب یک استاندارد بانام FIPS 180-2 نمود. نام‌گذاری این سه نسخه بر اساس طول خروجی آن‌ها و به‌صورت SHA-256، SHA-384 و SHA-512 انجام‌گرفته است. این سه نسخه که بانام SHA-2 نیز شناخته می‌شوند، ازنظر ساختار و انجام عملیات شبیه به SHA-1 می‌باشند.

در سال 2008 موسسه NIST یک نسخه اصلاح‌شده از SHA را طی استاندارد FIPS 180-3 منتشر کرد. در این نسخه ضمن اصلاح سه الگوریتم قبلی، یک الگوریتم جدید بانام SHA-224 نیز ارائه شد. این بارهم IETF بر اساس FIPS 180-3 اقدام به انتشار RFC 4634 نمود.

در سال‌های اخیر موسسه NIST در حال کار بر روی استاندارد جدیدی بانام SHA-3 است. این استاندارد تاکنون به تصویب نهایی نرسیده؛ اما NIST پیش‌نویس آن را طی FIPS 202 منتشر کرده است. ظاهراً آقای Ronald Rivest[2] به دنبال آن بود که الگوریتم MD6 را به‌عنوان SHA-3 موردپذیرش NIST قرار دهد. البته هنوز خود MD6 نیز به‌عنوان سند رسمی و استاندارد منتشر نشده است.

در جدول زیر الگوریتم‌های خانواده SHA به‌طور خلاصه با یکدیگر مقایسه شده است:

---

[1] Message Digest 4 (MD4)

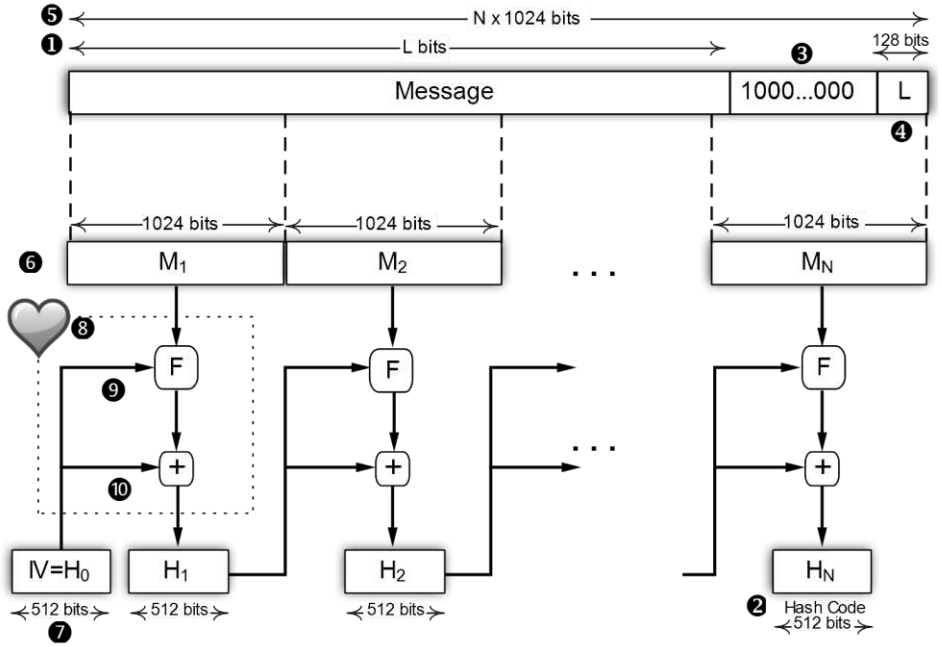
[2] آقای Ronald Rivest مبدع توابع درهم‌سازی MD2، MD4، MD5 و MD6 است.

مقادیر/الگوریتم	SHA-1	SHA-224	SHA-256	SHA-384	SHA-512
اندازه خروجی	160	224	256	384	512
اندازه پیام	$< 2^{64}$	$< 2^{64}$	$< 2^{64}$	$< 2^{128}$	$< 2^{128}$
اندازه بلوک	512	512	512	1024	1024
اندازه کلمه	32	32	32	64	64
تعداد مراحل(دور)	80	64	64	80	80

در ادامه این مبحث به تشریح الگوریتم SHA-512 می‌پردازیم؛ سایر الگوریتم‌های سری SHA نیز ازنظر عملیاتی شبیه SHA-512 هستند.

الگـوریتم SHA-512

الگوریتم SHA-512 یک پیام با طول متغیر بین 1 تا 2^{128} بیت را به‌عنوان ورودی دریافت کرده❶ و یک خروجی 512 بیتی تولید می‌نماید.❷ طول خروجی الگوریتم SHA-512 مثل دیگر توابع درهم‌سازی، همواره ثابت است.



پیام ورودی این الگوریتم برای پردازش باید به بلوک‌های ۱۰۲۴ بیتی تقسیم گردد. عملیاتی که توسط SHA-512 بر روی پیام انجام می‌شود، به‌صورت ساده در تصویر فوق نمایش داده‌شده است. این عملیات را می‌توان در پنج گام و به‌صورت زیر تشریح نمود:

- **گام اول**

اگر پیام ورودی را M در نظر بگیریم، ❶ مقدار آن می‌تواند در رنج $1 < M < 2^{128}$ بیت باشد. اما برای ارائه پیام به ورودی الگوریتم، باید طول پیام مضربی از ۱۰۲۴ باشد، به‌طوری‌که بعداً برای تقسیم آن به بلوک‌های ۱۰۲۴ بیتی مشکلی نداشته باشیم. لذا کمبود طول پیام تا رسیدن به مضربی از ۱۰۲۴ را با اضافه کردن بیت‌های توسعه انجام می‌دهیم؛ این کار در اصطلاح Padding یا توسعه گفته می‌شود. ❸

حتی اگر طول پیام ۱۰۲۴ بیت باشد، بازهم عمل Padding انجام می‌شود. به همین دلیل مقدار بیت‌های Padding می‌تواند در رنج ۱ تا ۱۰۲۴ قرار گیرد.

در Padding، بیت اول 1 و تمام بیت‌های باقیمانده 0 است. مثلاً برای اضافه کردن ۵ بیت، مقدار Pad به‌صورت 10000 خواهد بود.

- **گام دوم**

یک بلوک ۱۲۸ بیتی بانام L به انتهای پیام افزوده می‌شود. ❹ مقدار L بیانگر طول پیام، قبل از انجام فرآیند Padding است.

سپس پیام ورودی که حالا باید دارای طول $N \times 1024$ بیت باشد، ❺ به دنباله‌ای از بلوک‌های ۱۰۲۴ بیتی تقسیم می‌شود. $(M_1, M_2, M_3, \dots, M_N)$ ❻

- **گام سوم**

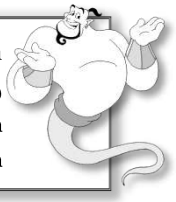
در این الگوریتم از یک بافر[1] ۵۱۲ بیتی برای نگهداری نتایج میانی و نهایی توابع درهم‌سازی، استفاده می‌شود. این بافر در قالب ۸ عدد ثبات[2] ۶۴ بیتی و بانام‌های (a, b, c, d, e, f, g, h) نشان داده می‌شود. به ثبات‌ها، کلمه یا Word نیز گفته می‌شود.

مقداردهی اولیه (IV) ❼ بافر با مقادیر هگزادسیمال و به‌صورت زیر است:

```
a = 6A09E667F3BCC908    e = 510E527FADE682D1
b = BB67AE8584CAA73B    f = 9B05688C2B3E6C1F
c = 3C6EF372FE94F82B    g = 1F83D9ABFB41BD6B
d = A54FF53A5F1D36F1    h = 5BE0CD19137E2179
```

این مقادیر در فرمت Big-endian ذخیره می‌شوند؛ به‌عبارت‌دیگر بایت باارزش یک کلمه، در آدرس پایین‌تر (سمت چپ) بایت قرار می‌گیرد.

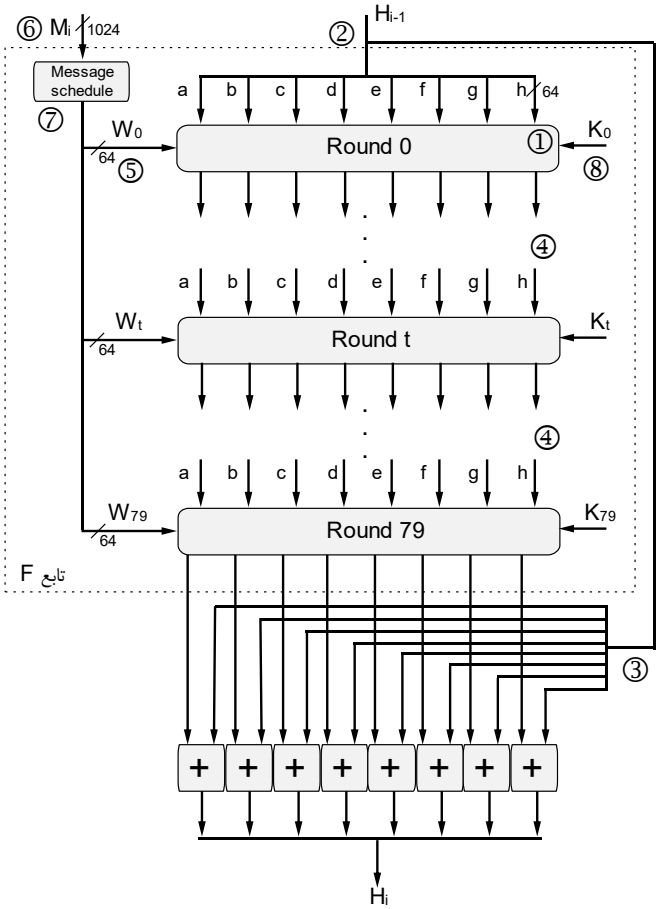
[1] Buffer
[2] Register



endian ترتیب قرارگیری بایت‌ها در حافظه را برای داده‌های چندبایتی معین می‌کند؛ و دارای فرمت‌های زیر است:
Big-endian: بایت باارزش بیشتر، در آدرس پایین‌تر حافظه قرار می‌گیرد.
Little-endian: بایت باارزش کمتر، در آدرس پایین‌تر حافظه قرار می‌گیرد.

- **گام چهارم**

ماژولی که با نماد ♥ در تصویر فوق مشخص‌شده❽ را می‌توان قلب الگوریتم SHA دانست. تابع F شامل ۸۰ دور است که در هر دور مقدار ۵۱۲ بیت موجود در بافر (a,b,c,d,e,f,g,h) را دریافت کرده و با انجام عملیاتی، مقادیر آن‌ها را به‌روز می‌نماید. تصویر زیر اجزای تابع F را نمایش می‌دهد:



در ورودی Round 0، ① مقدار Hash شده بلوک قبلی② ❾ که به‌صورت کلمات (a,b,c,d,e,f,g,h) در بافر ذخیره‌شده است، قرار می‌گیرد. البته یک نسخه از Hash بلوک قبلی نیز برای اعمال بر روی خروجی دور آخر(Round 79) ارسال می‌گردد.③ ❿ برای دورهای بعدی، خروجی دور قبل به‌عنوان ورودی قرار می‌گیرد.④ (البته توجه داشته باشید که مقدار ورودی اولین دور از اولین بلوک پیام، مقدار اولیه (IV) خواهد بود. ❼) همچنین در ورودی هر دور، یک مقدار ۶۴ بیتی⑤ مشتق شده از بلوک ۱۰۲۴ بیتی پیام جاری (M_i)⑥ نیز باید قرار گیرد. وظیفه ارائه مقادیر ۶۴ بیتی از بلوک ۱۰۲۴ بیتی بر عهده Message Schedule⑦ است.

آخرین مقداری که باید در ورودی هر دور قرار داشته باشد، یک مقدار ثابت بانام K است.⑧ طبق استاندارد مقدار K برای ۸۰ دور (K_0 تا K_{79}) به‌صورت زیر خواهد بود؛ این ارقام ۶۴ بیت نخستِ ریشه سوم(مکعب) اولین ۸۰ عدد اول(Prime) می‌باشند.

```
428a2f98d728ae22 7137449123ef65cd b5c0fbcfec4d3b2f e9b5dba58189dbbc
3956c25bf348b538 59f111f1b605d019 923f82a4af194f9b ab1c5ed5da6d8118
d807aa98a3030242 12835b0145706fbe 243185be4ee4b28c 550c7dc3d5ffb4e2
72be5d74f27b896f 80deb1fe3b1696b1 9bdc06a725c71235 c19bf174cf692694
e49b69c19ef14ad2 efbe4786384f25e3 0fc19dc68b8cd5b5 240ca1cc77ac9c65
2de92c6f592b0275 4a7484aa6ea6e483 5cb0a9dcbd41fbd4 76f988da831153b5
983e5152ee66dfab a831c66d2db43210 b00327c898fb213f bf597fc7beef0ee4
c6e00bf33da88fc2 d5a79147930aa725 06ca6351e003826f 142929670a0e6e70
27b70a8546d22ffc 2e1b21385c26c926 4d2c6dfc5ac42aed 53380d139d95b3df
650a73548baf63de 766a0abb3c77b2a8 81c2c92e47edaee6 92722c851482353b
a2bfe8a14cf10364 a81a664bbc423001 c24b8b70d0f89791 c76c51a30654be30
d192e819d6ef5218 d69906245565a910 f40e35855771202a 106aa07032bbd1b8
19a4c116b8d2d0c8 1e376c085141ab53 2748774cdf8eeb99 34b0bcb5e19b48a8
391c0cb3c5c95a63 4ed8aa4ae3418acb 5b9cca4f7763e373 682e6ff3d6b2b8a3
748f82ee5defb2fc 78a5636f43172f60 84c87814a1f0ab72 8cc702081a6439ec
90befffa23631e28 a4506cebde82bde9 bef9a3f7b2c67915 c67178f2e372532b
ca273eceea26619c d186b8c721c0c207 eada7dd6cde0eb1e f57d4f7fee6ed178
06f067aa72176fba 0a637dc5a2c898a6 113f9804bef90dae 1b710b35131c471b
28db77f523047d84 32caab7b40c72493 3c9ebe0a15c9bebc 431d67c49c100d4c
4cc5d4becb3e42b6 597f299cfc657e2a 5fcb6fab3ad6faec 6c44198c4a475817
```

- **گام پنجم**

پس‌ازاینکه توابع بر روی تمام بلوک‌های ۱۰۲۴ بیتی پیام اعمال شد، نهایتاً یک مقدار Hash شده ۵۱۲ بیتی در اختیار خواهیم داشت. ❷ می‌توان خروجی Hash یا همان Message Digest تولیدشده را به‌صورت زیر نمایش داد:

$$H_0 = IV$$

$$H_i = SUM_{64}(H_{i-1}, abcdefgh_i)$$

$$MD = H_N$$

که در عبارات فوق:

IV : مخفف Initial Value، و به معنی مقدار اولیه abcdefgh است، که در گام سوم انجام شد.

$abcdefgh_i$: خروجی مرحله قبلی. به‌عبارت‌دیگر خروجی مرحله i برای بلوک M_i.

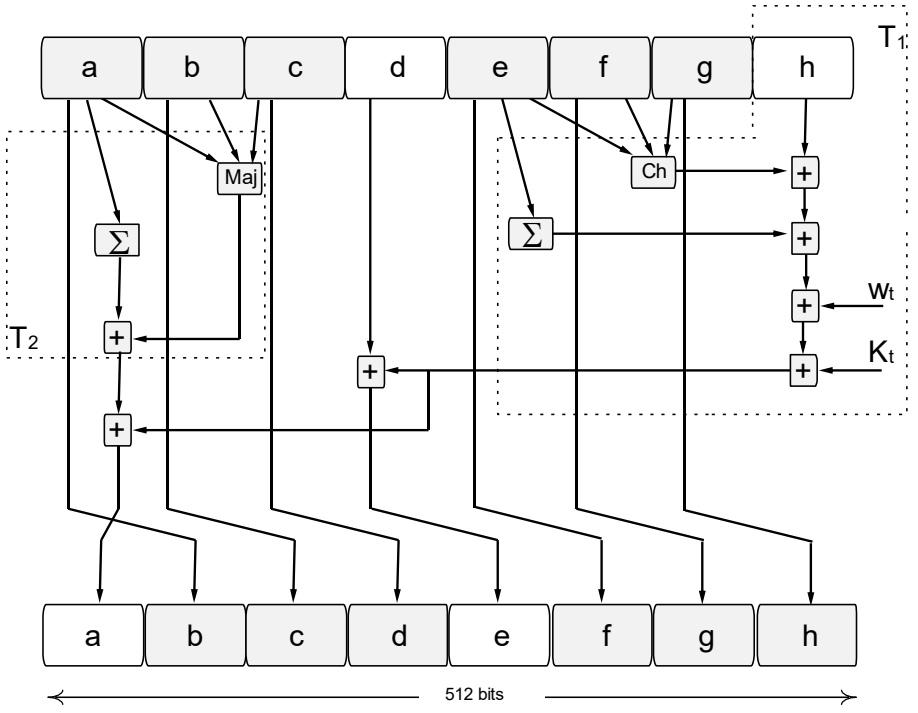
N : شماره آخرین بلوک پیام.

SUM_{64} : جمع به پیمانه 2^{64} . به‌عبارت‌دیگر حاصل جمع باید به $mod\ 2^{64}$ برسد.

MD : مقدار Hash یا Message Digest نهایی.

جزئیات یک دور (Round) از SHA

همان‌طور که گفته شد، تابع F شامل ۸۰ دور یا Round است. جزئیاتی که در هر دور از تابع F روی می‌دهد، در تصویر زیر نمایش داده‌شده است:



همان‌طور که در تصویر فوق ملاحظه می‌نمایید به‌جز کلمات d و h، کلمات دیگر بدون تغییر به سمت راست انتقال‌یافته و در خروجی ظاهر شده‌اند. دراین‌بین عملیاتی که روی کلمات ورودی d و h اتفاق افتاده، باعث گردیده تا کلمات a و e خروجی، متفاوت باشند.

اگر محاسبه مقادیر T_1 و T_2 به‌صورت زیر باشد:

$$T_1 = h + Ch(e,f,g) + \left(\sum_0^{512} e\right) + W_t + K_t$$

$$T_2 = \left(\sum_0^{512} a\right) + Maj(a,b,c)$$

آنگاه مقدار کلمات خروجی هر دور به‌صورت زیر خواهد بود:

$$a = T_1 + T_2$$
$$b = a$$
$$c = b$$
$$d = c$$
$$e = d + T_1$$
$$f = e$$
$$g = f$$
$$h = g$$

برای اینکه علائم استفاده‌شده در تصویر فوق را درک نمایید، به تشریح آن‌ها می‌پردازیم:

$$Maj\ (a,b,c) = (a \wedge b) \oplus (a \wedge c) \oplus (b \wedge c)$$

$$Ch\ (e,f,g) = (e \wedge f) \oplus (\neg e \wedge g)$$

$$\sum_0^{512} a\ = ROTR^{28}(a) \oplus ROTR^{34}(a) \oplus ROTR^{39}(a)$$

$$\sum_0^{512} e = ROTR^{14}(e) \oplus ROTR^{18}(e) \oplus ROTR^{41}(e)$$

و در آن‌ها:

$ROTR^n(x)$: [۱] جابه‌جایی مدور به سمت راست[۲]؛ به‌طوری‌که کلمه ۶۴ بیتی موردنظر (x) به مقدار n بیت به سمت راست جابه‌جا شود.

W_t : عبارت ۶۴ بیتی مشتق شده از بلوک ۵۱۲ بیتی ورودی.

K_t : مقدار ثابت کلید مربوطه.

$+$: جمع به پیمانه 2^{64}.

---

[۱] در متن اصلی استاندارد از عبارت R^n استفاده شده ولی آقای استالینگز در کتاب خود از عبارت $ROTR^n$ استفاده کرده است.

[2] Circular Right Shift

جزئیات Message Schedule

Message Schedule یک بلوک ۱۰۲۴ بیتی پیام را دریافت کرده و به ازای هر دور از ۸۰ دور تابع F، یک مقدار ۶۴ بیتی تولید می‌کند که از پیام ورودی مشتق شده است. سپس این مقدار ۶۴ بیتی را در ورودی هر دور قرار می‌دهد. با توجه به اینکه پیام ورودی ۱۰۲۴ بیت و مقدار خروجی برای هر دور ۶۴ بیت است، لذا ۱۰۲۴ بیت پیام پس از ۱۶ دور $(1024 = 64 \times 16)$ به اتمام رسیده و برای دورهای بعدی، روال تولید پیام متفاوت و به‌صورت زیر خواهد بود:

$$W_t = \sigma_1^{512}(W_{t-2}) + W_{t-7} + \sigma_0^{512}(W_{t-15}) + W_{t-16}$$

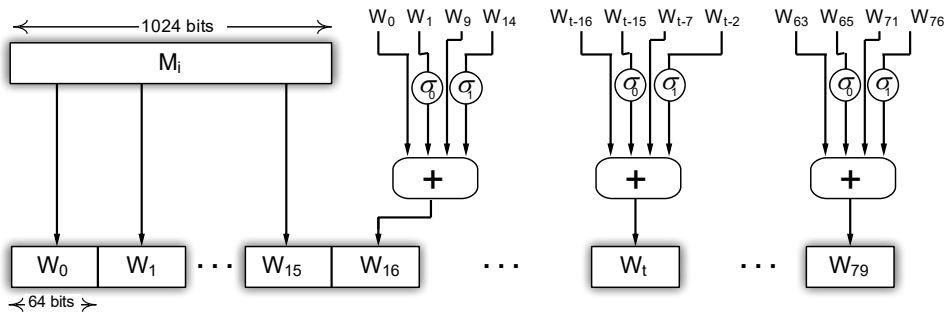
محاسبه اجزاء فرمول فوق به‌صورت زیر است:

$$\sigma_0^{512}(x) = ROTR^1(x) \oplus ROTR^8(x) \oplus SHR^7(x)$$

$$\sigma_1^{512}(x) = ROTR^{19}(x) \oplus ROTR^{61}(x) \oplus SHR^6(x)$$

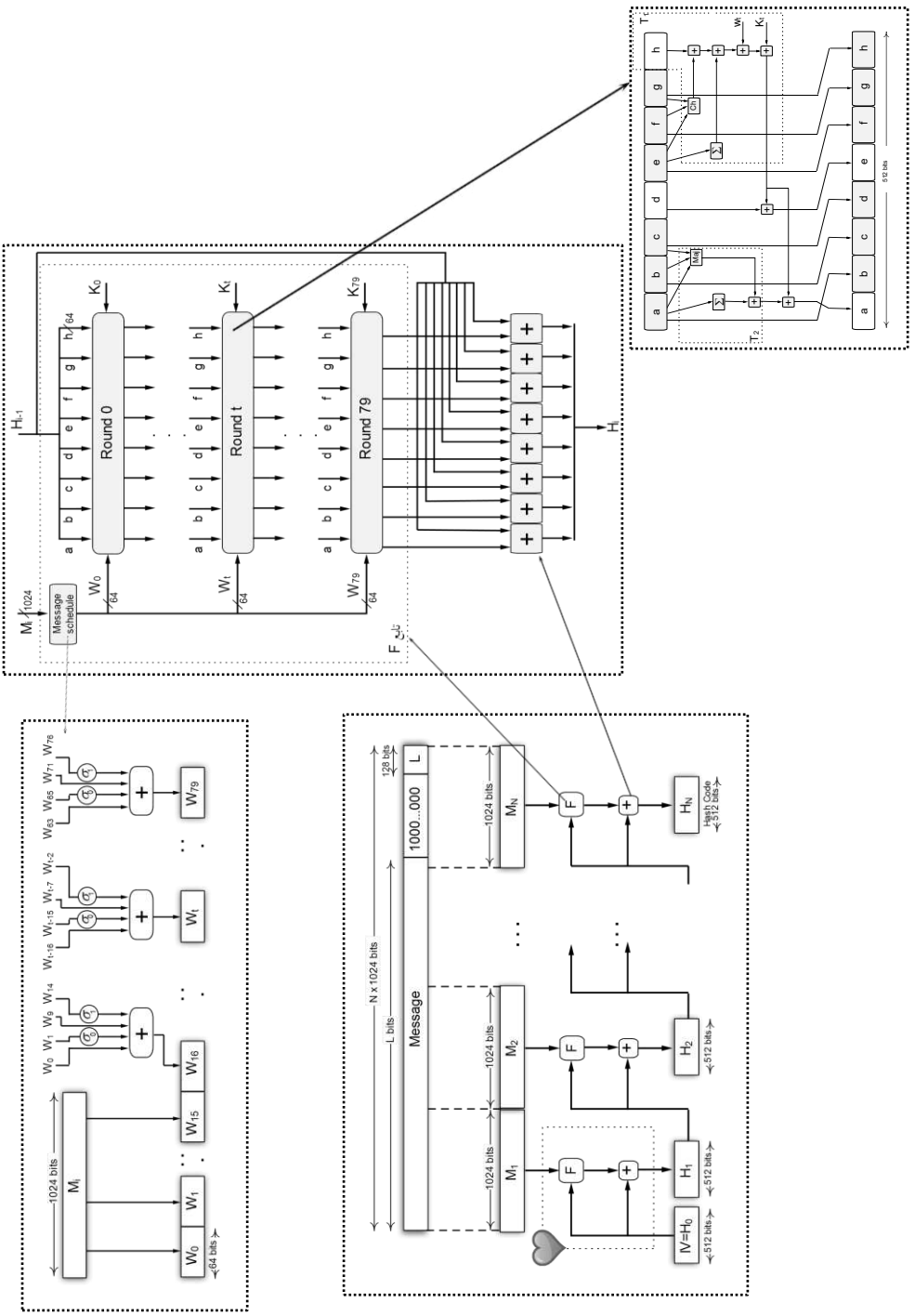
$SHR^n(x)$[۱]: انتقال به چپ[۲] عبارت ۶۴ بیتی (x) به‌اندازه n بیت با Padding توسط صفرهای سمت راست.

برای درک بهتر، در دیاگرام زیر جزئیات فرآیند Message Schedule نمایش داده شده است:



[۱] در متن اصلی استاندارد از عبارت S^n استفاده‌شده ولی آقای استالینگز در کتاب خود از عبارت SHR^n استفاده کرده است.

[2] Left Shift


Round 0
Round t
Round 79
Message schedule
Message
1024 bits
512 bits
64 bits
128 bits
1000...000
N x 1024 bits
L bits
IV=H0
Hash Code

بخش سوم

امنیت شبکه

فصل هفتم

مفاهیم امنیت شبکه

فصل هشتم

تکنولوژی‌ها و ابزارهای امنیتی

فصل نهم

ارتباطات امن شبکه

فصل هفتم

مفاهیم امنیت شبکه

مبحث اول

مفاهیم امنیتی

مبحث دوم

انواع تهدیدات

مبحث سوم

انواع حملات

مبحث چهارم

سرویس‌ها و مکانیسم‌های امنیتی

☑ مبحث اول

مفاهیم امنیتی

هر سازمانی در هر زمینه و ابعاد کاری، دارای اطلاعات مهمی است که ادامه حیات کسب‌وکار خود را وابسته به آن‌ها می‌داند. قطعاً دارایی‌های اطلاعاتی سازمان‌ها نسبت به نوع کار، اندازه، تعداد کارکنان، پراکندگی و گستردگی فیزیکی ساختمان‌ها، با یکدیگر متفاوت بوده و در درجه‌های مختلف امنیتی قرار می‌گیرند.

در ابعاد بزرگ، برخی از دارایی‌های اطلاعاتی ممکن حیات یک کشور و یا دولت را تحت تأثیر قرار دهد. در ابعاد میانی این اطلاعات می‌تواند باعث افشای یک تکنولوژی یا دستاورد جدید گردد. در ابعاد کوچک نیز ممکن است افشای اطلاعات یک شرکت تجاری مثل اطلاعات مشتری و یا بازرگانی، کسب‌وکار شرکت را دچار اختلال سازد.

به‌هرحال دارایی‌های اطلاعاتی هر سازمان کوچک و بزرگی، ازنظر خودش دارای اهمیت بالایی است و باید به‌دقت از آن‌ها محافظت شود. اهمیت محافظت از اطلاعات وقتی بیشتر می‌شود که سازمان دارای شبکه گسترده بوده و یا مجبور است شبکه خود را با شبکه دیگری که مدیریت آن مستقل است، مثل سازمان‌های دیگر و یا اینترنت، مرتبط سازد.

یک راه امنیتی این است که کامپیوتر خود را درون یک گاوصندوق (البته ضدآب) گذاشته و آن را به اعماق اقیانوس بفرستید! در این صورت کامپیوتری نخواهید داشت که نگرانش باشید. تعجب نکنید! متأسفانه قطع کامل ارتباط، به اولین راهکار مورداستفاده برخی از سازمان‌ها با هر سطحی از ارزش اطلاعات، تبدیل شده است. در دنیای امروز که تمام ابعاد زندگی انسان وابسته به ارتباطات دیتا گردیده، قطع ارتباط برای پاک کردن صورت‌مسئله، کاری غیرقابل‌قبول و تقریباً غیرممکن است.

البته در مقابل گروه حساس فوق، بسیارند کسانی که حداقل‌های امنیتی را هم رعایت نکرده و امیدوارند که هیچ حمله‌ای به آن‌ها صورت نگیرد.



و اما گروه اعجاب‌انگیز دیگری نیز هستند که علیرغم علم به ارزش بالای اطلاعات سازمان، به‌واسطه قدرت و توان بالای مدیریتی که در خود سراغ دارند! بدون انجام هیچ‌گونه اقدام پیشگیرانه‌ای، منتظر می‌مانند به سازمان حمله صورت پذیرد، تا بتوانند آن را مدیریت نمایند. :)

همان‌طور که در بخش اول کتاب گفته شد، امنیت باید در لایه‌های مختلف از امنیت فیزیکی تا دستورالعمل‌های مدیریتی، صورت پذیرد. اما آنچه در بخش سوم کتاب بر روی آن تمرکز داریم، امنیت شبکه و ارتباطات خواهد بود. نوع حملات به شبکه بسیار گسترده‌تر، پیچیده‌تر و خاص‌تر از حملات دیگر مثل دسترسی فیزیکی است؛ و بالطبع آسیب‌پذیری‌ها و اقدامات پیشگیرانه آن نیز وسیع‌تر از سایر ابعاد امنیتی خواهد بود.

امنیت شبکه نه‌تنها برای شبکه، بلکه برای سرویس‌های دیگر نیز دارای اهمیت فراوان است. حتی بسیاری از آسیب‌پذیری‌های دیگر که ممکن است بر روی سایر سرویس‌ها، مثل سیستم‌عامل و یا دیتابیس وجود داشته باشند را می‌توان با طراحی و پیکربندی مناسب شبکه، به حداقل رساند.

با توجه به وسعت مفاهیم امنیت شبکه، وقتی صحبت از امنیت شبکه می‌شود، بالطبع امنیت سرویس‌ها و سیستم‌های دیگر نیز در این عرصه وارد شده و مباحث پیرامونی آن نیز گسترده‌تر می‌شود و باید به‌صورت جامع مطرح شوند. لذا مباحثی که در این فصل مطرح‌شده غالباً جنبه عمومی داشته و علاوه بر امنیت شبکه، ممکن است شامل امنیت سایر سرویس‌ها و سیستم‌ها نیز گردد.

یک مدیر یا مشاور امنیت توانمند و باتجربه، باید یک طراحی مناسب و قابل‌قبول برای حفظ امنیت شبکه و سیستم‌ها ارائه نماید. طراحی درست و منطقی باید با توجه به میزان اهمیت اطلاعات سازمان، میزان ریسک قابل‌قبول و توان مالی سازمان انجام‌گرفته و بین آن‌ها تعادل برقرار نمود. به این معنی که نه آن‌قدر بی‌خیال که تا بروز یک اتفاق وحشتناک دارایی‌های اطلاعاتی را رها کند؛ و نه آن‌قدر سخت‌گیر و محتاط که هم روال کار را مختل کرده و هم باعث شود هزینه‌های سازمان برای حفاظت از اطلاعات، بیش از ارزش خود اطلاعات شود.

آگاهی از اصول امنیت، شناسایی راهکارها، تقسیم‌بندی دارایی‌های اطلاعاتی سازمان بر اساس درجه اهمیت و بررسی الزامات سازمانی، ازجمله مواردی است که در زمان طراحی امنیت اطلاعات، باید توسط طراح در نظر گرفته شوند.

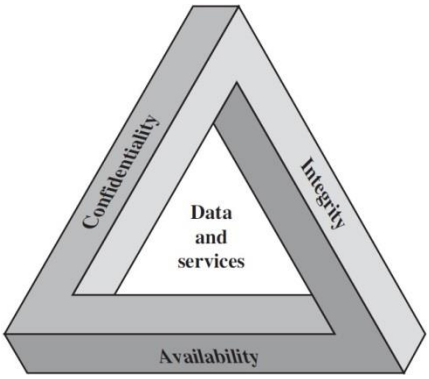
مدیر امنیت پس از فراهم کردن یک طرح امنیتی بر اساس بهترین شیوه‌ها (Best Practice)، باید آن را در سراسر سازمان اجرا نماید.

نکته مهمی که یک مدیر باید در نظر داشته باشد، نظارت همیشگی بر اجرای صحیح طرح امنیتی است. صرف اجرای یک طرح خوب نمی‌تواند برای مدت زیادی تضمین‌کننده امنیت باشد؛ بلکه نظارت بر اجرای صحیح طرح و همچنین به‌روزرسانی نرم‌افزارها و تجهیزات امنیتی، از الزامات ادامه روند وضعیت امن در فناوری اطلاعات است.

مثلث امنیت اطلاعات

امنیت صرفاً جلوگیری از افشای اطلاعات نیست. بلکه در دسترس نبودن اطلاعات در زمان موردنیاز و تغییر اطلاعات در مسیر تبادل نیز باعث از دست دادن امنیت می‌گردد.

امنیت اطلاعات دارای سه مفهوم اصلی Confidentiality، Integrity و Availability است که به دلیل حروف اول این سه کلمه، بانام مثلث CIA مشهور گردیده است.



- **محرمانگی (Confidentiality)**

محرمانه بودن یعنی فقط افراد یا سیستم‌های مجاز بتوانند به اطلاعات موردنظر دسترسی داشته باشند. به‌عبارت‌دیگر افراد غیرمجاز نباید امکان دسترسی به اطلاعات را داشته باشند.

دو نوع دیتا وجود دارد. اول، دیتای در حال حرکت بر روی شبکه. دوم، دیتای ذخیره‌شده بر روی تجهیزات مثل سیستم‌ها، سرورها، Storageها و Cloud.

برای امنیت دیتای در حال حرکت باید از روش‌های رمزنگاری استفاده نمود؛ یعنی ابتدا دیتا رمزشده، و سپس اقدام به ارسال آن نماییم. راه دیگر، ایجاد کانال‌های ارتباطی امن برای تبادل دیتا بین فرستنده و گیرنده است. راهکارهای امنیت دیتای در حال حرکت در این بخش از کتاب موردبحث قرار می‌گیرند.

برای امنیت دیتای ذخیره‌شده نیز باید مکانیسم‌های کنترل دسترسی فراهم گردد. البته در کنار مکانیسم کنترل دسترسی، می‌توان از رمزنگاری نیز بهره برد. درباره مکانیسم کنترل دسترسی بر روی سیستم‌عامل و دیتابیس، در بخش چهارم همین کتاب توضیحاتی ارائه شده است.

- **صحت و تمامیت (Integrity)**

صحت یعنی فقط افراد یا سیستم‌های مجاز، امکان تغییر و اصلاح دیتا را داشته باشد. در امنیت شبکه منظور از صحت این است که دیتای دریافتی دقیقاً همان چیزی باشد که فرستنده ارسال کرده است.

در بخش دوم کتاب، نحوه استفاده از توابع درهم‌ساز برای بررسی صحت پیام تشریح شد. البته از توابع درهم‌ساز می‌توان هم برای بررسی صحت اطلاعات ذخیره‌شده و هم برای بررسی صحت اطلاعات در حال حرکت بهره برد.

- **دسترس‌پذیری (Availability)**

افراد مجاز در زمان مجاز باید به دیتای موردنظر دسترسی داشته باشند.

اصلی‌ترین مباحث بر سر موضوع دسترس‌پذیری در بخش امنیت شبکه است. البته امنیت فیزیکی نیز در دسترس‌پذیری از اهمیت ویژه‌ای برخوردار است؛ اما اگر بحث حملات و هکرها را در نظر بگیریم، امنیت شبکه نسبت به امنیت فیزیکی و افزونگی، به‌صورت پویاتری با دسترس‌پذیری در ارتباط است.

در همین بخش از کتاب، حملاتی که باعث اختلال در دسترس‌پذیری می‌شوند، بررسی خواهند شد.

موسسه NIST، مثلث CIA را طی استاندارد FIPS 199 در جهت اجرای طرح دولت الکترونیک در ایالات‌متحده، برای سازمان‌های دولتی تعریف نموده است.[1] در این استاندارد تأثیرات بالقوه از دست دادن هریک از سه اصل فوق را در سه گروه و به‌صورت زیر تقسیم‌بندی کرده است:

- **تأثیرات بالقوه کم (Low)**

انتظار می‌رود از دست دادن محرمانگی، صحت و یا دسترس‌پذیر بودن، عوارض جانبی محدودی بر روی عملیات سازمانی، دارایی‌های سازمانی و افراد داشته باشد.

- **تأثیرات بالقوه متوسط (Moderate)**

انتظار می‌رود از دست دادن محرمانگی، صحت و یا دسترس‌پذیر بودن، باعث عوارض جانبی جدی بر روی عملیات سازمانی، دارایی‌های سازمانی و افراد داشته باشد. مثلاً ممکن است باعث تخریب قابل‌توجه مأموریت‌های سازمان گردیده و یا منجر به زیان مالی بزرگی شود.

- **تأثیرات بالقوه زیاد (High)**

انتظار می‌رود از دست دادن محرمانگی، صحت و یا دسترس‌پذیر بودن، باعث عوارض جانبی فاجعه‌باری بر روی عملیات سازمانی، دارایی‌های سازمانی و افراد داشته باشد. فاجعه‌بار به این معناست که ادامه حیات سازمان به خطر خواهد افتاد.

[1] http://csrc.nist.gov/publications/fips/fips199/FIPS-PUB-199-final.pdf

موسسه NIST با ارائه این تقسیم‌بندی، از سازمان‌ها می‌خواهد که امنیت مربوط به دارایی‌های اطلاعاتی خود را در قالب سه گروه فوق طبقه‌بندی نمایند.

این طبقه‌بندی می‌تواند جهت استفاده از راهکارهای امنیتی، سیاست‌گذاری امنیتی و تخصیص هزینه‌های مالی، کمک قابل‌توجهی به مدیر امنیت اطلاعات نماید. همچنین باعث می‌شود تا سازمان از هزینه‌های اضافی و بی‌مورد جهت اطلاعات کم‌ارزش، رهایی یابد.

طبقه‌بندی آسیب‌پذیری

شناخت آسیب‌پذیری‌ها و نقاط ضعف سیستم‌ها و شبکه، یک گام بزرگ برای اصلاح آن یا انجام اقدامات متقابل به‌حساب می‌آید.

بسیاری از آسیب‌پذیری‌ها ناشی از یک یا چند مورد زیر است:

- معایب سیاست‌گذاری (سیاست‌های اشتباه)
- اشتباهات طراحی
- نقاط ضعف پروتکل
- پیکربندی اشتباه (Misconfiguration)
- آسیب‌پذیری‌های نرم‌افزار
- عوامل انسانی
- بدافزارها (نرم‌افزارهای مخرب)
- آسیب‌پذیری‌های سخت‌افزاری
- دسترسی فیزیکی به منابع شبکه

شرکت‌های بزرگ اقدام به ایجاد یک دیتابیس از آسیب‌پذیری‌های موجود نموده و آن را بر روی اینترنت در دسترس عموم قرار داده‌اند. این دیتابیس بانام Common Vulnerabilities and Exposures (CVE) شناخته‌شده و توسط آدرس https://cve.mitre.org در دسترس است.

اصول اساسی امنیت

در زمان طراحی، پیاده‌سازی و نگهداری امنیت شبکه، باید اصول اساسی امنیت را مدنظر قرار دهید. اصول موردنیاز امنیت شبکه در زمان طراحی و پیاده‌سازی به‌صورت زیر است:

- **قانون حداقل امتیاز**[1]

این قانون بیان می‌کند که باید حداقل دسترسی به منابع موردنیاز فراهم شود و نه بیشتر. **به عبارتی می‌توان گفت همه غیرمجاز هستند مگر اینکه خلاف آن ثابت شود.**

- **دفاع در عمق**[2]

این اصل می‌گوید که شما امنیت را در هر نقطه‌ای از شبکه باید برقرار سازید. به‌عبارت‌دیگر برپایی امنیت باید در لایه‌های مختلف شبکه انجام پذیرد.

مفهوم دفاع در عمق این است که شکست یک سیستم واحد، به معنای حذف کامل امنیت در شبکه نباشد. به‌عنوان‌مثال اگر شبکه شما فقط دارای یک فایروال باشد، در صورت شکست آن، امنیت شبکه شما از بین می‌رود. ولی اگر شبکه بر اساس اصل دفاع در عمق دارای تجهیزات و نرم‌افزارهای امنیتی در لایه‌های مختلف باشد، شکست یکی از آن‌ها کل امنیت شبکه را زیر سؤال نخواهد برد.

- **تفکیک وظایف**[3]

قرار دادن افراد خاص در وظایف خاص، باعث می‌شود سیاست‌های امنیتی به‌درستی اجرا شوند. همچنین چرخش افراد به‌صورت دوره‌ای در نقش‌های مختلف، می‌تواند تا مقداری جلوی آسیب‌پذیری‌ها را بگیرد؛ چراکه افراد با قرارگیری در نقش جدید، اقدام به بررسی صحت اجرای سیاست‌ها در محل جدید می‌نمایند.

- **بازرسی**[4]

سوابق ممیزی و همچنین هر آنچه در شبکه اتفاق افتاده، جهت بازرسی و حسابرسی باید نگهداری شوند. هر رویدادی که در شبکه رخ می‌دهد باید واقعه‌نگاری شده و اطلاعات آن جهت رسیدگی و بازرسی‌های آینده نگهداری شود.

کسانی که در حال ایجاد تغییر بر روی شبکه هستند، نباید دسترسی مستقیم جهت ویرایش و یا حذف وقایع را داشته باشند.

---

[1] Rule of least privilege
[2] Defense in depth
[3] Separation of duties
[4] Auditing

☑ مبحث دوم

انواع تهدیدات

حدود ۵۰۰ سال قبل از میلاد، یک ژنرال چینی به نام Sun Tzu Wu، کتابی با عنوان هنر جنگ نوشت. در این رساله نظامی بر اهمیت شناخت خود و سپس تهدیداتی که با آن‌ها مواجه خواهید بود، پرداخته‌شده است.

برای محافظت از دارایی‌های اطلاعاتی سازمان، باید:

۱. خودتان را بشناسید؛ اطلاعاتی که باید از آن‌ها محافظت شود، سیستم‌های ذخیره سازی مورداستفاده، سیستم‌ها و سرویس‌های مورداستفاده جهت تبادل دیتا و تمام فرآیندهای دیگری که در سازمان مورداستفاده قرار می‌گیرند، باید شناسایی شوند.

۲. با چه تهدیداتی ممکن است مواجه شوید؟ برای تصمیم‌گیری در مورد امنیت اطلاعات، مدیریت باید از تهدیدات ممکن برای افراد سازمان، برنامه‌های کاربردی، دیتا و اطلاعات سازمان، آگاه باشد.

در امنیت اطلاعات هر شیء، شخص و هر چیز دیگری که دارایی‌های اطلاعاتی را در معرض خطر قرار دهد، یک تهدید محسوب می‌شود.

درصورتی‌که شبکه شما به اینترنت نیز متصل باشد، علاوه بر تهدیداتی که مستقیماً متوجه شما یا سازمانتان است، تهدیدات بی‌جهت دیگری نیز برای شما به وجود خواهد آمد. اتصال به اینترنت مثل رانندگی در ایران است.☺ شما هرقدر راننده خوبی باشید و مقررات راهنمایی و رانندگی را به‌درستی اجرا نمایید، بالاخره یک نفر پیدا می‌شود که به هر زحمتی، با شما تصادف کند. یعنی باید همواره آمادگی تصادف را داشته باشید. در اینترنت نیز داستان به همین منوال است. حتی اگر دارایی‌های اطلاعاتی شما ارزش فوق‌العاده‌ای هم نداشته باشد، باید همواره آمادگی مورد هدف قرار گرفتن را داشته باشید.

در ادامه این مبحث، تهدیدات در ۱۴ گروه زیر تقسیم‌بندی و تشریح شده است.

۱. مالکیت معنوی ۲. حملات نرم‌افزاری ۳. تنزل در کیفیت خدمات ۴. جاسوسی یا تجاوز ۵. بلایای طبیعی ۶. خطاهای انسانی ۷. اخاذی اطلاعاتی ۸. سیاست یا برنامه‌ریزی سازمانی ناقص، ناکافی و یا گمشده ۹. کنترل‌های ناقص، ناکافی و یا گمشده ۱۰. کارشکنی یا خرابکاری ۱۱. سرقت ۱۲. خطاها و شکست سخت‌افزاری ۱۳. خطاها و شکست فنی نرم‌افزار ۱۴. منسوخ شدن تکنولوژی.

مالکیت معنوی[1]

بسیاری از سازمان‌ها، ایجاد یا حمایت از توسعه مالکیت معنوی را به‌عنوان بخشی از عملیات کسب‌وکار خود، قرار داده‌اند. مالکیت معنوی به معنی مالکیت ایده‌ها و کنترل آن‌ها است. استفاده از مالکیت معنوی شخص دیگر، ممکن است با پرداخت مالی و یا بدون هزینه باشد؛ ولی به هر صورت باید اعتبار منبع آن حفظ شود. مالکیت معنوی می‌تواند شامل اسرار تجاری، حق نشر، علائم تجاری و اختراعات باشد.

کارکنان ممکن است به‌واسطه انجام وظایفشان به مالکیت معنوی دسترسی داشته باشند؛ لذا باید توجه داشت، اختصاص غیرمجاز مالکیت معنوی به‌منزله تهدیدی برای امنیت اطلاعات است.

شایع‌ترین نقض مالکیت معنوی، استفاده غیرقانونی یا تقلید از یک نرم‌افزار است. آمار نشان می‌دهد که یک‌سوم نرم‌افزارهای مورداستفاده در سطح جهان (و ۹۰درصد نرم‌افزارهای استفاده‌شده در ایران)[۲]، به‌صورت غیرقانونی مورداستفاده قرارگرفته شده است.

برای به اجرا درآوردن حق نشر، از مکانیسم‌هایی مثل علامت دیجیتال، کد داخلی، کد حق نشر و استفاده از Bad sector بر روی رسانه نرم‌افزار استفاده می‌گردد. یکی دیگر از ابزار رایج حق نشر، توافقنامه‌ای است که در زمان نصب نرم‌افزار نمایش داده می‌شود و از کاربر می‌خواهد که با آن موافقت کند تا بتواند فرآیند نصب را ادامه دهد.

البته روش‌های جدیدی مثل ثبت‌نام آنلاین، نیز برای حفاظت از نرم‌افزار رایج شده است. این روش بر اساس یک الگوریتم مخفی از سیستم شما یک کد ایجاد کرده و بر اساس آن اقدام به ثبت نرم‌افزار می‌نماید. به دلیل مخفی بودن نحوه محاسبه کد، این طرح مخالفانی دارد که ادعا می‌کنند حریم خصوصی افراد ممکن است در معرض خطر قرار گیرد.

تمام روش‌های فنی و الکترونیکی مورداستفاده برای حق نشر به طریقی قابل دور زدن هستند؛ بعضی از روش‌ها به‌راحتی و البته بعضی روش‌های دیگر نیز به‌سختی. لذا فارغ از اینکه از چه روشی برای حق نشر استفاده می‌شود، باید یک عامل بازدارنده قدرتمند نیز برای جلوگیری از استفاده غیرمجاز، وجود داشته باشد. به‌عنوان‌مثال وضع قوانین جزایی و البته اجرای صحیح و مقتدرانه آن، می‌تواند یک عامل بازدارندگی مناسب برای افراد سودجویی باشد که به‌راحتی یکی از صنایع استراتژیک، پرسود و اشتغال‌زای کشور را مورد هدف قرار داده‌اند.

---

[1] Intellectual Property

[۲] در ایران استفاده غیرقانونی از نرم‌افزار فقط به نرم‌افزارهای خارجی محدود نشده است. متأسفانه نرم‌افزارهای داخلی نیز به‌صورت گسترده‌ای مورداستفاده غیرقانونی قرار می‌گیرند. به همین دلیل است که علیرغم توانایی بالای مهندسین ایرانی، متأسفانه پیشرفت چشمگیری در عرصه نرم‌افزار نداریم؛ چراکه برای برنامه‌نویسان و شرکت‌های مهندسی، رغبتی جهت تولید نرم‌افزار باقی نمانده است.

حملات نرم‌افزاری

فرد یا گروهی اقدام به طراحی و گسترش نرم‌افزاری می‌نمایند که هدف آن حمله به سیستم یا شبکه و تخریب آن است. از این نرم‌افزارها بانام‌هایی مثل کدهای مخرب[1]، نرم‌افزارهای مخرب[2] و بدافزار[3] نام می‌برند. البته عبارت بدافزار (Malware) از ادغام کلمات Malicious و Software یا همان نرم‌افزارهای مخرب، ایجاد گردیده است. طراحی نرم‌افزارهای مخرب بر اساس صدمه زدن، از بین بردن[4] و یا انکار سرویس بر روی سیستم هدف است.

بدافزارها یا همان کدهای مخرب را می‌توان به گروه‌های زیر تقسیم‌بندی نمود:

i. ویروس (Virus)
ii. کرم (Worm)
iii. اسب تراوا (Trojan Horse)
iv. بمب‌های منطقی (Logical Bombs)
v. درب پشتی (Backdoor)

هرقدر برنامه‌نویسان بدافزارها پیشرفت می‌کنند، مفاهیم فوق نیز بیشتر با یکدیگر تلفیق می‌شوند. مثلاً ممکن است بدافزاری نوشته شود که به‌صورت همزمان در چند گروه فوق قرار گیرد. به‌هرحال گروه‌های فوق را به‌صورت زیر تشریح می‌کنیم.

i. ویروس

ویروس کامپیوتری، قطعه کدی است که باعث انجام اقدامات خرابکارانه بر روی سیستم کامپیوتری می‌گردد. اما چرا نام ویروس را برای این دسته از کدهای مخرب انتخاب کرده‌اند؟ همانند دنیای انسان‌ها، ویروس‌های دنیای دیجیتال نیز برای اجرا، خود را به برنامه‌ها و فایل‌های موجود بر روی سیستم متصل کرده و از حق دسترسی آن برای اهداف خود استفاده می‌کنند؛ به همین ترتیب نیز اقدام به تکثیر خود بر روی سایر سیستم‌ها می‌نمایند.

در بسیاری از موارد، کاربران ناخواسته ویروس را دریافت می‌کنند، مثل زمانی که بر روی یک فایل الصاقی ایمیل کلیک کرده و یا هر پیغامی که بر روی صفحه ظاهر می‌شود را بدون خواندن، OK می‌کنند.

---

[1] Malicious Code
[2] Malicious Software
[3] Malware
[4] Destroy

ii. **کرم**

کرم یک برنامه مخرب است که بدون نیاز به برنامه دیگر می‌تواند خود را تکرار نماید. کرم‌ها می‌توانند تکثیر خود را تا زمان اشغال کامل منابع، مثل حافظه، هارددیسک و پهنای باند، ادامه دهند.

کرم‌ها قادرند خود را به‌راحتی در شبکه منتشر نمایند. کرم موجود در یک کامپیوتر آلوده، می‌تواند یک کپی از خود را برای تمام آدرس‌های موجود در پست الکترونیک، ارسال نماید. کرم‌ها خود را در فضای به اشتراک گذاشته شبکه کپی می‌کنند تا هر سیستمی که از آن فضا استفاده می‌نماید را آلوده نمایند.

iii. **اسب تراوا**

اسب‌های تراوا، کدهای مخربی هستند که ماهیت خرابکارانه خود را پشت یک ابزار به‌ظاهر مفید پنهان می‌کنند. این نام برگرفته از افسانه یونان باستان است؛ که در آن تعدادی از سربازان داخل یک اسب چوبی زیبا پنهان‌شده و آن را به پادشاه تروی هدیه کردند. پس‌ازآنکه پادشاه اسب را به داخل قلعه آورد، سربازان از آن خارج‌شده و اقدام به تسخیر قلعه نمودند.

iv. **بمب‌های منطقی**

بمب منطقی، قطعه کد مخربی است که در زمان مشخص یا به هنگام وقوع شرایط مشخصی، عملیات مخرب خود را انجام می‌دهد.

برخی از برنامه نویسان نیز از این نوع کد، حتی در برنامه‌های تجاری استفاده می‌کنند.

v. **درب پشتی**

ویروس یا کرم می‌تواند با نصب یک نرم‌افزار بر روی سیستم قربانی، راه نفوذ آسان هکر را برای دفعات بعدی نیز فراهم آورد.

تهدیدات چندشکلی[1]

یکی از چالش‌های بزرگ مبارزه با ویروس‌ها و کرم‌ها، ظهور تهدیدات چندشکلی است. تهدید چندشکلی، یک قطعه کد مخرب (مثل ویروس یا کرم) است که می‌تواند در طول زمان خود را تغییر دهد تا توسط آنتی‌ویروس قابل‌شناسایی نباشد.

درواقع این نوع ویروس‌ها و کرم‌ها می‌توانند ضمن تکامل، اندازه و دیگر ویژگی‌های فایلی خود را که آنتی‌ویروس از آن‌ها برای شناخت ویروس استفاده می‌کند، تغییر دهند.

[1] Polymorphic Threats

مضرات آنتی‌ویروس‌ها

حتی اگر شما با داشتن یک آنتی‌ویروس قدرتمند از صدمات کدهای مخرب تا حد قابل‌توجهی در امان باشید، اما بدافزارها بازهم برای شما هزینه‌ساز خواهند بود.

علاوه بر هزینه‌ای که معمولاً سالانه برای خرید یک آنتی‌ویروس پرداخت می‌کنید، فراهم کردن منابعی که آنتی‌ویروس نیاز دارد تا بتواند به‌درستی وظایف خود را انجام دهد نیز برای شما هزینه‌ساز خواهد بود. ازجمله می‌توان به منابع سخت‌افزاری مثل CPU، حافظه و هارددیسک و همچنین پهنای باند موردنیاز جهت بروز رسانی، اشاره نمود.

تنزل در کیفیت خدمات

تداوم برقراری سیستم اطلاعات یک سازمان، وابسته به موفقیت‌آمیز بودن بسیاری از سیستم‌های پشتیبانی‌کننده دیگر، ازجمله شبکه برق، شبکه مخابرات، تأمین‌کنندگان قطعات، فروشندگان خدمات و کارکنان است؛ که البته هر یک از این خدمات ممکن است در اثر طوفان، بیماری کارمند و دیگر حوادث پیش‌بینی‌نشده قطع شوند. این قبیل قطع خدمات، نوعی از دست دادن دسترس‌پذیری تلقی می‌شود. همان‌طور که گفته شد دسترس‌پذیری یا Availability یکی از ارکان مهم امنیت محسوب می‌شود.

یکی از راه‌های جلوگیری یا ترمیم مشکلات مرتبط باکیفیت خدمات، امضای توافق‌نامه سطح خدمات[1](SLA) با فراهم آورنده سرویس است. اگر سازمان بابت هزینه‌ای که پرداخت می‌نماید سطح خدمات موردنظر خود را طی قرارداد SLA مشخص نموده باشد، در صورت بروز خرابی یا افت سطح سرویس، می‌تواند خسارت واردشده را از سرویس‌دهنده طلب نماید.[2]

جاسوسی یا تجاوز

جاسوسی و تجاوز یک دسته‌بندی کاملاً شناخته‌شده و البته گسترده از فعالیت‌های الکترونیکی و انسانی است که می‌تواند محرمانگی را نقض نماید.

زمانی که یک فرد غیرمجاز، به اطلاعات محافظت‌شده سازمان دسترسی پیدا کند، گفته می‌شود جاسوسی یا تجاوز صورت گرفته است. مهاجم می‌تواند از روش‌های مختلفی برای دسترسی به اطلاعات استفاده نماید.

---

[1] Service Level Agreement (SLA)

[2] خدا را شکر که ما از این قرتی‌بازی‌ها در ایران نداریم. مخصوصاً با اداراتی مثل برق و مخابرات! :)

برخی از روش‌های جمع‌آوری اطلاعات کاملاً قانونی هستند؛ به‌عنوان‌مثال، مرور وب برای انجام تحقیقات بازار. این روش‌های حقوقی را درمجموع، هوشمندی رقابتی[1] می‌نامند. در سال‌های اخیر، هوشمندی رقابتی به یکی از مفاهیم مهم مدیریتی تبدیل‌شده است.

اما زمانی که روش‌های جمع‌آوری اطلاعات از خط قرمز قانون و اخلاق عبور می‌کند، به آن جاسوسی صنعتی گفته می‌شود.

البته برخی گونه‌های جاسوسی هم وجود دارند که زیاد با تکنولوژی درگیر نیستند. یکی از این روش‌ها، دیدزنی شانه‌ای[2] است. به‌عنوان‌مثال در زمان کار با خودپرداز (ATM)، بانکداری اینترنتی، موبایل بانک و یا حتی پیکربندی تجهیزات، این روش جاسوسی ممکن است برای فرد مهاجم کارگشا باشد.



هنگامی‌که شخصی می‌تواند یک نفر را در حال واردکردن اطلاعات شخصی به یک سیستم، مشاهده نماید؛ روش دیدزنی شانه‌ای اتفاق افتاده است. شاید در این کار قوانین و مقررات نقض نشده باشند، اما حریم خصوصی افراد نادیده گرفته‌شده و می‌تواند به‌عنوان یک تهدید برای از بین رفتن اطلاعات محرمانه محسوب شود.

بلایای طبیعی

بلایای طبیعی یکی از خطرناک‌ترین تهدیدات به‌حساب می‌آید؛ چراکه با هشدار کمی رخداده و معمولاً از کنترل انسان خارج است. تهدید بلایای طبیعی شامل رویدادهایی مثل سیل، زلزله،

[1] Competitive intelligence
[2] Shoulder Surfing

آتش‌سوزی، رعدوبرق، آتش‌فشان و حتی هجوم حشرات است. این اتفاقات می‌توانند ذخیره‌سازی، انتقال و استفاده از اطلاعات را مختل نمایند.

خطاهای انسانی

عمل بدون قصد یا هدف مخرب که توسط کاربر مجاز انجام‌گرفته و باعث بروز اشکال در سیستم اطلاعاتی شده است را خطاهای انسانی می‌گویند.

عواملی مثل بی‌تجربگی، آموزش نادرست، مدیر نالایق و پیش‌فرض‌های نادرست، باعث بروز خطاهای انسانی می‌گردد. گاهی اوقات یک اشتباه کوچک می‌تواند یک فاجعه بزرگ برای سازمان به وجود آورد.

یکی از بزرگ‌ترین تهدیدات امنیت اطلاعات هر سازمان، کارمندان همان سازمان می‌باشند؛ چراکه کارمندان نزدیک‌ترین عوامل به داده‌های سازمانی هستند. کارمندان برای انجام وظایف شغلی خود روزانه با دارایی‌های اطلاعاتی سازمان در ارتباط هستند. لذا اشتباهات کارکنان، به‌راحتی می‌تواند منجر به مواردی از قبیل: افشای دیتای طبقه‌بندی‌شده، ورود اطلاعات اشتباه، حذف تصادفی، ذخیره اطلاعات در محل حفاظت نشده و حتی انداختن اطلاعات طبقه‌بندی‌شده در سطل زباله، باشد.

اخاذی اطلاعاتی[1]

اخاذی اطلاعاتی زمانی رخ می‌دهد که یک مهاجم یا شخص مورد اعتماد، اطلاعاتی را از سیستم سرقت کرده و در عوض بازگرداندن یا فاش نکردن آن‌ها، مطالبه مزد می‌نماید.

سیاست یا برنامه‌ریزی سازمانی ناقص، ناکافی و یا گمشده[2]

سازمانی که دارای یک سیاست یا برنامه‌ریزی مدون برای امنیت اطلاعات نباشد، قطعاً در برابر حملات آسیب‌پذیر خواهد بود.

مدیر اجرایی سازمان مسئول برنامه‌ریزی راهبردی امنیت است. حتی اگر مدیران IT طرح و سیاست‌های امنیتی را آماده کرده باشند، باید توسط مدیر ارشد سازمان تصویب گردد تا تمام کارکنان در هر سمت و جایگاهی ملزم به رعایت آن باشند.

---

[1] Information Extortion

[2] Missing, Inadequate, or Incomplete Organizational Policy or Planning

کنترل‌های ناقص، ناکافی و یا گمشده

برای حفاظت از سازمان، محافظت مداوم از دارایی‌های سازمان باید انجام پذیرد. دارایی‌های سازمان باید همواره کنترل‌شده تا در صورت نیاز، ارتقاءیافته و یا تعویض گردند.

برای مثال اگر نرم‌افزار تجهیزات برای مدت زیادی بروز نشوند، باعث می‌شود تا آسیب‌پذیری‌های شناخته‌شده آن‌ها همچنان باقی‌مانده و برای سازمان تهدید محسوب شوند.

کنترل بر روی پیکربندی‌های انجام‌شده نیز باید صورت پذیرد.

کارشکنی یا خرابکاری[۱]

این دسته از تهدیدات شامل کارشکنی عمدی بر روی سیستم کسب‌وکار و یا اعمال خرابکاری برای صدمه زدن به وجهه و شهرت سازمان است.

این نوع اعمال می‌تواند در سطح خرد و توسط کارمندان، و یا در سطح کلان به‌صورت سازمان در برابر سازمان اتفاق بیفتد.

این خرابکاری‌ها لزوماً به لحاظ مالی ویران‌کننده نیستند؛ هدف اصلی در این روش، خراب کردن شهرت و از بین بردن آبروی شغلی سازمان هدف است.

سرقت[۲]

سرقت به معنی برداشتن غیرقانونی اموال دیگران است. سرقت می‌تواند فیزیکی، الکترونیکی یا فکری (ایده) باشد.

سرقت فیزیکی را می‌توان به‌راحتی توسط طیف گسترده‌ای از اقدامات امنیت فیزیکی، مثل درب‌های قفل‌شده، پرسنل امنیتی آموزش‌دیده، نصب دوربین و سیستم‌های کنترلی، تا حد قابل قبولی، کنترل کرد. اما مدیریت و کنترل سرقت الکترونیکی پیچیده‌تر است.

وقتی یک شیء فیزیکی دزدیده می‌شود، به‌راحتی قابل‌تشخیص است. اما سرقت الکترونیکی به‌راحتی قابل‌کشف نیست، و ممکن است زمانی متوجه سرقت شویم که خیلی دیر شده باشد.

خطاها و شکست سخت‌افزاری[۳]

[1] Sabotage or Vandalism
[2] Theft
[3] Hardware Failures or Errors

شکست و خطای سخت‌افزاری ممکن بر اساس یک عیب شناخته‌شده یا ناشناخته، رخ دهد. به‌هرحال این نقص باعث می‌شود تا به دسترس‌پذیری و قابل‌اعتماد بودن سیستم، خدشه وارد شود.

خطاها و شکست فنی نرم‌افزار[1]

نرم‌افزارهایی که پس از نوشته شدن اشکال‌یابی نشده و بدون آنکه ایرادات آن‌ها مشخص شود به بازار ارائه می‌شوند؛ یکی از دلایل شکست و خطا در نرم‌افزار است. البته نرم‌افزارهای شرکت‌های بزرگ که در آزمایشگاه‌ها و بعدازآن به‌صورت نسخه آزمایشی هم مورد امتحان قرار می‌گیرند، بازهم از بروز خطا و شکست مصون نمی‌باشند. نوع دیگری از شکست نرم‌افزاری زمانی رخ می‌دهد که نرم‌افزار با سخت‌افزار خاصی دچار مشکل شود.

بررسی و آزمایش ناقص نرم‌افزار نه‌تنها می‌تواند منجر به بروز خطا و شکست در سیستم شود؛ بلکه ممکن است باعث ایجاد رخنه امنیتی نیز گردد.

وب‌سایت امنیتی www.securityfocus.com، دقیقه‌به‌دقیقه آخرین آسیب‌پذیری‌های امنیتی نرم‌افزارها را بر روی سایت خود قرار می‌دهد.

منسوخ‌شدن تکنولوژی[2]

زیرساخت‌های قدیمی و منسوخ‌شده می‌توانند به سیستم‌های غیرقابل‌اعتماد تبدیل شوند. مدیریت IT، باید متوجه باشد که وقتی یک تکنولوژی (نرم‌افزار و یا سخت‌افزار)، از رده خارج می‌شود، سیستم اطلاعات سازمان در برابر حملات آسیب‌پذیر خواهد شد.

مدیریت باید طی یک برنامه‌ریزی مناسب، همواره بر وضعیت تکنولوژی‌های مورداستفاده نظارت داشته باشد تا در صورت منسوخ شدن به‌سرعت با تکنولوژی جدید جایگزین شوند.

برای نمونه، علیرغم اینکه مدت زیادی از منسوخ‌شدن ویندوزXP توسط مایکروسافت می‌گذرد، اما همچنان بسیاری از سازمان‌ها در مقابل تعویض این ویندوز با نسخه جدیدتر، از خود مقاومتی جانانه! نشان می‌دهند. در سازمانی این مقاومت به شکلی صورت می‌گرفت که من فکر کردم آن سازمان برای تمام ویندوزهای کاربرانش لایسنس خریداری کرده است!!!

[1] Technical Software Failures or Errors
[2] Technological Obsolescence

☑ مبحث سوم

انواع حملات

شبکه شما هرروز با تهدیدات و حملات جدیدی روبرو است. هکرهای عزیز! روزبه‌روز به دانش خود افزوده و با استفاده از آسیب‌پذیری‌های موجود در نرم‌افزارها و تجهیزات، روش‌های جدیدی برای حمله و دسترسی به منابع شبکه، ایجاد می‌کنند. می‌توان گفت سرعت هکرها در ایجاد حملات و تهدیدات جدید از سرعت ما انسان‌های خوب برای پیشگیری و دفاع، به‌مراتب بیشتر است. لذا شناخت تهدیدات و خطراتی که ممکن است شبکه شما در معرض آن‌ها قرارگرفته باشد، می‌تواند یک گام روبه‌جلو در امنیت شبکه به‌حساب آید.

در گام اول، مهاجمین را طبقه‌بندی می‌کنیم. با توجه به سازمان مطبوع خود، باید دشمنان بالقوه را شناسایی نماییم. انواع دشمن را می‌توان به‌صورت زیر نام برد:

- تروریست‌ها
- تبهکاران
- سازمان‌های دولتی
- ملت‌ها، کشورها
- هکرها
- کارمندان ناراضی
- رقبا
- هرکسی که دارای یک دستگاه رایانه است! (غمناک ولی واقعی)

یک فرد مهاجم می‌تواند دارای انگیزه‌های متفاوتی باشد. یک دانشجو که می‌خواهد یک روش یا ابزار را آزمایش نماید تا اطلاعات خود را بسنجد، یک تبهکار باانگیزه مالی، یک کشور یا سازمان باانگیزه سیاسی، یک شرکت برای جاسوسی صنعتی، یک کارمند ناراضی برای بدنام کردن سازمان، یک رقیب برای اختلال درروند کسب‌وکار و یا یک شخص دارای کامپیوتر ولی بی‌اطلاع! که به‌عنوان زامبی در جریان یک حمله قرارگرفته است.

به‌هرحال شناسایی کسانی که می‌توانند برای سازمان یک خطر بالقوه محسوب شوند، در انتخاب راهکارهای امنیتی بسیار مفید خواهد بود.

روش‌های حمله[1]

طراحی حملات به صورتی انجام می‌شود که حتی‌الامکان قابل‌کشف و شناسایی نباشند، لذا هکرها از روش‌هایی استفاده می‌کنند که آن‌ها را در سایه و به‌صورت مخفی نگه دارد. مهم‌ترین این روش‌ها در ادامه توضیح داده‌شده است:

- **شناسایی[2]**

این فرآیند جهت کشف و جمع‌آوری اطلاعاتی در مورد شبکه هدف انجام می‌پذیرد. فرآیند شناسایی می‌تواند شامل اسکن شبکه برای پیدا کردن آدرس‌های IP پاسخگو و همچنین تشخیص پورت‌های باز بر روی آن‌ها، باشد.

اولین گام در حمله، کشف چیزهایی است که در شبکه موجود است. سپس بر اساس آن‌ها آسیب‌پذیری‌های بالقوه شبکه تعیین گردیده و طراحی حمله صورت می‌پذیرد.

- **مهندسی اجتماعی[3]**

در مهندسی اجتماعی نقطه‌ضعف و آسیب‌پذیری شبکه هدف، کاربر آن شبکه است. مهاجم سعی می‌کند اطلاعات موردنیاز خود جهت حمله را از طریق کاربر به دست آورد. مهندسی اجتماعی می‌تواند از طریق دوستی با کاربران سازمان و یا تماس‌های تلفنی با آن‌ها انجام پذیرد.

همچنین از Phishing و Pharming نیز در مهندسی اجتماعی کمک گرفته می‌شود. در Phishing یک لینک شبیه لینک مورد اعتماد به کاربر نمایش داده می‌شود. در Pharming، از طریق آلوده کردن DNS، کاربر را به یک سایت جعلی راهنمایی (Re-Direct) می‌کنند؛ به‌طوری‌که کاربر فکر می‌کند با وب‌سایت موردنظر در حال تعامل است. در هر دو حالت Phishing و Pharming، مهاجم قصد به دست آوردن اطلاعات محرمانه کاربر مثل نام کاربری، کلمه عبور و اطلاعات کارت‌بانکی را دارد.

- **افزایش حق دسترسی**

در این فرآیند مهاجم می‌خواهد حق دسترسی را کسب کند که مجاز به استفاده از آن نیست. این حق دسترسی که معمولاً در حد مدیر سیستم است می‌تواند به مهاجم کمک کند تا کنترل شبکه یا تجهیزات موردنظر را در دست گیرد. به‌عنوان‌مثال در صورت به دست آوردن امتیاز سطح ۱۵ در روترهای سیسکو، مهاجم قادر است تا روتر موردنظر خود را به هر صورتی که می‌خواهد، پیکربندی نماید.

---

[1] Attack Methods
[2] Reconnaissance
[3] Social engineering

- **درب پشتی[۱] (Back Door)**

وقتی مهاجم توانست دسترسی به یک سیستم را به دست آورد، قطعاً نمی‌خواهد که آن دسترسی را از دست بدهد و یا این‌که دفعه بعد هم همان مقدار تلاش را برای به دست آوردن دسترسی انجام دهد. لذا برنامه‌هایی بانام Backdoor را بر روی سیستم نصب می‌کنند که در دسترسی‌های بعدی به آن‌ها کمک کرده و یا اطلاعات موردنیاز جهت حمله را جمع‌آوری نماید.



بسیاری از تجهیزات و نرم‌افزارها حتی در برندهای مشهور! از Backdoor استفاده می‌کنند. البته معمولاً هدف اصلی برندهای مشهور، جاسوسی از دولت‌ها است و نه جاسوسی از اشخاص.

برنامه‌های Backdoor همیشه توسط هکر به‌طور مستقیم نصب نمی‌شوند؛ بلکه نصب آن‌ها ممکن است توسط یک کاربر ناآگاه، مثلاً با کلیک بر روی یک فایل الصاقی به ایمیل صورت پذیرد. همچنین Backdoorها می‌توانند توسط ویروس‌ها و بدافزارها بر روی سیستم نصب شوند.

- **کانال پنهان (Covert channel)**

در این روش از برنامه‌ها یا ارتباطات درراه‌های غیرمعمول استفاده می‌شود. به‌عنوان‌مثال اگر بر اساس سیاست‌های امنیتی، ترافیک HTTP مجاز ولی ترافیک Peer-to-peer غیرمجاز شناخته‌شده باشد؛ کاربر سعی کند با ایجاد تونلی در داخل HTTP، ترافیک Peer-to-peer را تبادل نماید.

مهاجم نیز ممکن است شبیه روش فوق، اقدام به ایجاد تونلی نماید تا ترافیک موردنظر خود را مخفی سازد. به‌عبارت‌دیگر یک کانال مخفی برای ترافیک موردنظر در داخل ترافیک یک پروتکل مجاز ایجاد و سوءاستفاده نماید.

- **حملات داخلی**

یک نکته بسیار مهم برای حفظ امنیت سیستم‌ها و شبکه، آگاهی از مسیرهای بالقوه جهت استفاده در حملات است. همیشه حملات از خارج به داخل شبکه صورت نمی‌گیرد؛ بلکه نفوذ فیزیکی هکر به داخل سازمان، کارمندان ناراضی، کارمندان بی‌اطلاع و Backdoorها، می‌توانند ازجمله عوامل حملات داخلی باشند.

---

[۱] به Backdoor درب بازگشت نیز گفته می‌شود.

برای جلوگیری از حملات داخلی، داشتن یک سیاست امنیتی مناسب می‌تواند کارگشا باشد. احرازهویت کاربران، احرازهویت سیستم‌ها با پروتکل IEEE 802.1x، آنتی‌ویروس، VLANبندی، لیست‌های کنترل دسترسی[1]، ایجاد Zoneهای مخصوص مثل Server Farm و DMZ، راه‌اندازی Active Directory و سرویس کنترل پذیرش شبکه[2] (NAC)، می‌توانند ازجمله موارد مهم سیاست‌گذاری امنیتی در شبکه باشند.

- **کشف کلمه عبور**

در این روش مهاجم سعی می‌کند کلمه عبور را کشف نماید. یکی از راه‌های کشف کلمه عبور، حملات جستجوی فراگیر است.

همچنین کشف کلمه عبور می‌تواند توسط بدافزار، حمله مردمیانی، استراق سمع و یا استفاده از Key logger صورت پذیرد.

برای مقابله با حمله جستجوی فراگیر، باید نام کاربری پس از چند بار تلاش ناموفق در واردکردن کلمه عبور، قفل شود.

- **Botnet**

به گروهی از کامپیوترهایی که توسط بدافزارها آلوده‌شده و ناخواسته کنترل آن‌ها به دست مهاجم افتاده است، Botnet می‌گویند. به‌عبارت‌دیگر Botnet را می‌توان لشگری از زامبی‌ها[3] دانست که منتظر دستور حمله هستند.

هکرها از Botnet برای حمله گسترده و توزیع‌شده به یک سیستم یا شبکه استفاده می‌نمایند. ازجمله حملاتی که از این روش استفاده می‌کنند، می‌توان از حملات توزیع‌شده منع خدمت یا DDoS[4] نام برد.

اصطلاح Botnet از دو کلمه انگلیسی Bot و Network گرفته‌شده است. Bot به معنی ربات و Network به معنی شبکه است. شاید بتوان معادل فارسی ربات شبکه را برای آن در نظر گرفت، ولی من ترجیح می‌دهم از همان عبارت Botnet در این کتاب استفاده نمایم.

ازجمله Botnetهای مشهور می‌توان از BredoLab یا Oficla در سال 2009، و همچنین Conficker یا Downadup در سال 2008، نام برد.

---

[1] Access List (ACL)

[2] Network Admission Control

[3] زامبی (Zombie) به سیستمی گفته می‌شود که توسط بدافزارها آلوده‌شده و بدون اطلاع کاربر، تحت کنترل هکر قرارگرفته است. هکر از زامبی برای حمله به سیستم‌های دیگر استفاده می‌نماید.

[4] Distributed Denial of Service (DDoS)

حملات امنیتی

در استانداردهای ITU X.800 و RFC 2828، حملات امنیتی به دو گروه اصلی بانام‌های حملات فعال و حملات غیرفعال تقسیم‌بندی گردیده است.

- **حملات غیرفعال (Passive Attack)**

در حملات غیرفعال هکر فقط اطلاعات را استراق سمع و جمع‌آوری می‌نماید و هیچ تأثیری بر روی آن‌ها نمی‌گذارد. در این نوع از حملات، هدف هکر به دست آوردن اطلاعات در حال انتقال است، و نه تغییر آن‌ها.

حملات غیرفعال دارای دو نوع می‌باشند: انتشار محتوای پیام و تجزیه‌وتحلیل ترافیک.

منظور از انتشار محتوای پیام، فهمیدن اطلاعات حساس و محرمانه تبادل شده بین فرستنده و گیرنده، توسط هکر است.

اگر اطلاعات تبادل شده بین فرستنده و گیرنده رمزنگاری‌شده باشد، مهاجم صرفاً با به دست آوردن پیام نمی‌تواند از متن آن باخبر شود. در این صورت باید بر روی پیام‌ها تجزیه‌وتحلیل انجام دهد که بتواند آن را رمزگشایی کرده و یا به‌واسطه فرستنده، تعداد پیام و طول پیام، ماهیت ارتباط در حال وقوع را حدس بزند.

تشخیص حملات غیرفعال بسیار سخت است. به دلیل اینکه این نوع حملات هیچ تغییری بر روی پیام و الگوی رفتاری انجام نمی‌دهند، لذا گیرنده و فرستنده نمی‌توانند از وجود یک شخص ثالث که در حال گرفتن پیام‌ها است، اطلاع یابند.

در برخورد با حملات غیرفعال، پیشگیری بهتر از تشخیص است. به همین دلیل استفاده از رمزنگاری قبل از تبادل دیتا، می‌تواند تا حد قابل قبولی شمارا در مقابل حملات غیرفعال، مصون نگه دارد.



- **حملات فعال (Active Attack)**

برخلاف حملات غیرفعال، حملات فعال بر روی جریان دیتا تأثیرگذار می‌باشند. هکر ضمن دریافت پیام، اقدام به تغییر آن نموده و یا یک جریان غلط دیتا بین گیرنده و فرستنده برقرار می‌نماید.

حملات فعال به چهار گروه تغییر چهره، بازپخش، تغییر و منع خدمت تقسیم می‌شوند.

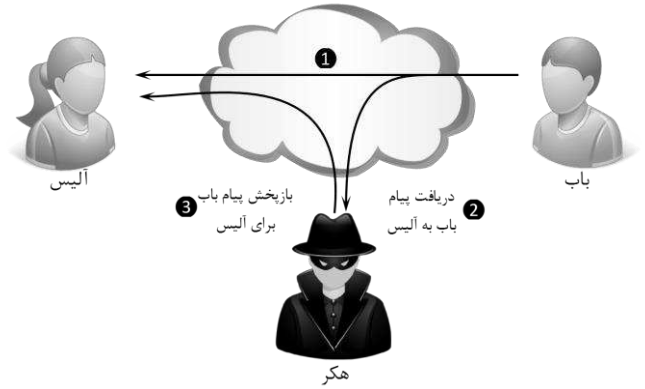
i. **تغییر چهره (Masquerade)**

هکر وانمود می‌کند شخص مورد اعتماد است. در این نوع حملات، هکر توالی احرازهویت فرستنده را به دست آورده و برای گیرنده بازپخش می‌نماید. به همین دلیل گیرنده تصور می‌کند که احراز هویت به‌درستی انجام‌گرفته است.



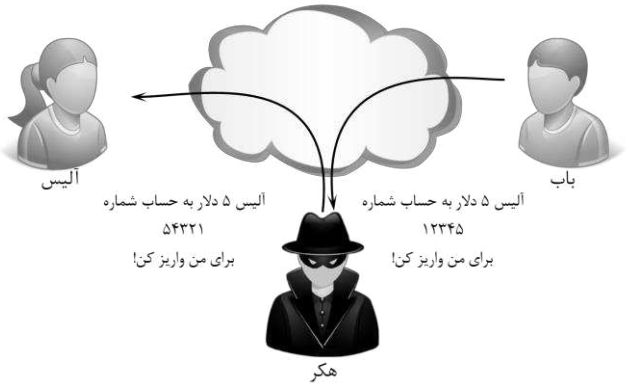
ii. **بازپخش (Replay)**

جذب یک واحد داده و ارسال مجدد آن برای گیرنده را حمله بازپخش می‌گویند. در این نوع حمله، پیام بدون هیچ تغییری، مجدداً ارسال می‌گردد.



iii. **تغییر (Modification)**

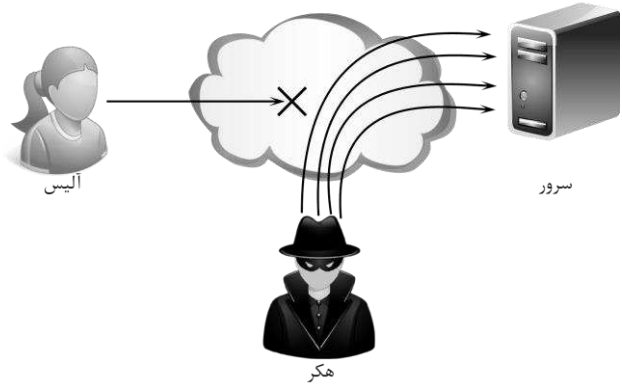
تغییر تمام یا قسمتی از پیام اصلی با پیام موردنظر هکر، Modification گفته می‌شود. در این نوع حملات هکر پیام را از فرستنده دریافت کرده، تغییرات موردنظر خود را اعمال نموده و سپس پیام تغییریافته را برای گیرنده ارسال می‌نماید.



iv. منع خدمت (DoS)

در حمله Denial of Service یا منع خدمت، هکر قصد دارد از سرویس‌دهی یک سرور یا شبکه به کاربر مجاز جلوگیری نماید.

در این نوع حملات، هکر با ارسال پیام‌های پی‌درپی باعث می‌شود تمام منابع سرور (مثل پهنای باند یا توان پاسخگویی) به هدررفته و از سرویس دادن به کاربر مجاز بازبماند. نوع پیشرفته‌تر حملاتDoS، حملات منع‌خدمت توزیع‌شده (DDoS) است. در این حالت هکر از لشگر زامبی‌های (Botnet) برای اشغال منابع شبکه یا سرور هدف، استفاده می‌نماید.



حقه‌بازی (Spoofing)

در این نوع حملات، هکر با استفاده از روش‌هایی، آدرس IP کامپیوتر مجاز را شناسایی کرده و با قرار دادن آن به‌جای آدرس IP خود، اقدام به دسترسی غیرمجاز به منابع می‌نماید. از این روش در مواردی استفاده می‌شود که دسترسی توسط Access List یا فایروال، فقط برای آدرس‌های خاصی مجاز شده باشد.

حمله مردمیانی

در این نوع حملات، هکر در بین فرستنده و گیرنده قرارگرفته و هویت خود را برای طرفین جعل می‌کند. هر دو طرف فرستنده و گیرنده تصور می‌کنند که با طرف مقابل در حال تبادل دیتا هستند، درصورتی‌که تمام اطلاعات توسط هکر دریافت شده و پس از تغییر برای آن‌ها ارسال می‌شود.

حملات مردمیانی در مبحث تبادل کلید دیفی‌هلمن کاملاً تشریح گردیده است.

هرزنامه (Spam)

به ایمیل‌های ناخواسته، هرزنامه گفته می‌شود. شاید هرزنامه برای کاربر یک مزاحمت بی‌اهمیت جلوه کند، اما سیلی از هرزنامه‌ها برای سازمان می‌تواند زیان‌آور بوده و حتی دسترس‌پذیری را مختل نماید. علاوه بر آن، هرزنامه‌ها می‌توانند حاوی کدهای مخرب بوده و باعث آلودگی سیستم‌ها و شبکه گردند.

استراق سمع[۱] (Sniff)

وظیفه دستگاه یا نرم‌افزار Sniff، نظارت و جمع‌آوری اطلاعات در حال تبادل بر روی شبکه است. Sniff هم دارای عملکرد مشروع و هم نامشروع می‌باشد. زمانی که مدیر شبکه برای تجزیه‌وتحلیل وضعیت شبکه از نرم‌افزار استراق سمع استفاده می‌نماید، کاربردی مشروع است. اما زمانی که هکر با استفاده از ابزار Sniff، اقدام به استراق سمع دیتای در حال تبادل بر روی شبکه می‌نماید، کاری نامشروع و نوعی حمله محسوب می‌گردد. خود Sniff جزء حملات غیرفعال است؛ اما اطلاعات به‌دست‌آمده از این طریق، در حملات فعال مورداستفاده قرار می‌گیرند.

---

[۱] معنی لغوی Sniff، بو کشیدن است. اما با توجه به نوع کاربرد آن در بحث امنیت، من از عبارت استراق سمع برای Sniff استفاده می‌کنم.

☑ مبحث چهارم

سرویس‌ها و مکانیسم‌های امنیتی

در مباحث قبل، تهدیدات و حملاتی که ممکن است برای سازمان وجود داشته باشند، مورد بررسی قرار گرفتند. در مستندات ITU X.800[1] یک سری از سرویس‌ها و مکانیسم‌های امنیتی توصیه‌شده که می‌توانند در برقراری امنیت تأثیرگذار باشند.

در ادامه این مبحث به سرویس‌ها و مکانیسم‌های امنیتی موجود در ITU X.800 پرداخته می‌شود.

الف) سرویس‌های امنیتی

سرویس‌های امنیتی شامل پنج سرویس: ۱. احرازهویت ۲. کنترل دسترسی ۳.محرمانگی اطلاعات ۴. صحت اطلاعات و ۵. انکارناپذیری، است.

احراز هویت (Authentication)

این سرویس در زمان برقراری ارتباط بین دو نهاد، احرازهویت را به شکل‌های زیر فراهم می‌کند:

i. **Peer entity authentication**
این سرویس فراهم آورنده احراز هویت، برای استفاده در زمان برقراری ارتباط و یا در طول زمان انتقال داده، بین یک یا چند نهاد متصل‌به‌هم، است.
به‌عبارت‌دیگر هویت طرفین در شروع ارتباط و در طول آن تضمین می‌شود.

ii. **Data origin authentication**
این سرویس فراهم آورنده احراز هویت مبدأ داده است.
البته سرویس Data Origin Authentication، در مقابل حملات تکرار (Replay) و تغییر (Modification)، محافظتی روی دیتا ندارد.

[1] https://www.itu.int/rec/T-REC-X.800-199103-I/en

کنترل دسترسی (Access Control)

این سرویس از دسترسی غیرمجاز به منابع، جلوگیری به عمل می‌آورد. کنترل دسترسی در این سرویس بر اساس مدل OSI صورت می‌گیرد. این سرویس ممکن است حفاظت را بر اساس انواع مختلفی از دسترسی مثل خواندن، نوشتن، حذف و غیره، ارائه نماید.

محرمانگی اطلاعات (Data Confidentiality)

این سرویس فراهم‌آورنده محافظت از داده در برابر افشای غیرمجاز بوده و به شرح زیر صورت می‌پذیرد:

i. **Connection confidentiality**
فراهم‌آورنده محرمانگی برای تمام اتصالات کاربر است. در این سرویس از سرویس‌های ارتباطی اتصال‌گرا[1] مثل TCP، استفاده می‌شود.

ii. **Connectionless confidentiality**
فراهم‌آورنده محرمانگی برای یک بلوک دیتای کاربر است. در این سرویس از سرویس‌های ارتباطی غیراتصال‌گرا[2] مثل UDP، استفاده می‌شود.

iii. **Selective field confidentiality**
این سرویس فراهم آورنده محرمانگی برای یک فیلد خاص است.

iv. **Traffic flow confidentiality**
این سرویس فراهم آورنده محافظت از اطلاعاتی است که ممکن است از طریق مشاهده جریان ترافیک به دست آید.

صحت اطلاعات (Data Integrity)

این سرویس جهت مقابله با حملات فعال مورداستفاده قرارگرفته و ممکن است به یکی از اشکال زیر صورت پذیرد:

i. **Connection integrity with recovery**
این سرویس فراهم‌کننده صحت برای تمام دیتای کاربر در تمام اتصالات است. همچنین قابلیت تشخیص و بازیابی هرگونه تغییر، درج، حذف یا تکرار نیز در این سرویس وجود دارد.

ii. **Connection integrity without recovery**

[1] Connection Oriented
[2] Connectionless

همانند سرویس قبل، ولی بدون قابلیت بازیابی است.

iii. **Selective field connection integrity**
فراهم آورنده صحت برای یک فیلد خاص بر روی تمام اتصالات کاربر است.

iv. **Connectionless integrity**
فراهم آورنده صحت بر روی یک بلوک دیتای کاربر.

v. **Selective field connectionless integrity**
فراهم‌کننده صحت برای یک فیلد خاص بر روی یک بلوک دیتا.

انکارناپذیری (Non-repudiation)

این سرویس ممکن به یک یا هر دو حالت زیر عمل نماید:

i. **Non-repudiation with proof of origin**
دریافت‌کننده ضمن دریافت دیتا، فرستنده آن را نیز احراز هویت می‌کند. در این صورت فرستنده نمی‌تواند پیام ارسالی را انکار نماید.

ii. **Non-repudiation with proof of delivery**
فرستنده، پس از ارسال پیام، از گیرنده پیامی مبتنی بر تحویل پیام دریافت می‌نماید. در این صورت گیرنده نمی‌تواند دریافت پیام را انکار کند.

ب) مکانیسم‌های امنیتی

مکانیسم‌های امنیتی شامل هشت مکانیسم: ۱. رمزگذاری ۲. امضای دیجیتال ۳. کنترل دسترسی ۴. صحت دیتا ۵. تبادل احراز هویت ۶. توسعه ترافیک ۷. کنترل مسیریابی ۸. گواهی رسمی، است.

رمزگذاری (Encipherment)

رمزگذاری می‌تواند محرمانگی دیتا یا جریان ترافیک اطلاعات را فراهم آورد. علاوه بر محرمانگی، رمزنگاری در کنار سایر مکانیسم‌های امنیتی، می‌تواند باعث ایجاد یا کامل شدن یک سرویس امنیتی دیگر شود. الگوریتم‌های رمزنگاری به‌صورت زیر می‌باشند[1]:

i. الگوریتم‌های رمزنگاری متقارن که برای رمزگذاری و رمزگشایی از یک کلید مشترک استفاده می‌کنند.

[1] در بخش دوم این کتاب، به سیستم رمزنگاری و کاربردهای آن به‌صورت مبسوط پرداخته‌شده است.

ii. الگوریتم‌های رمزنگاری نامتقارن که دارای دو کلید مختلف برای رمزگذاری و رمزگشایی هستند.
iii. نوع سوم رمزنگاری، توابع درهم‌سازی هستند که رمزگذاری انجام‌شده در آن‌ها برگشت‌پذیر نیست.
iv. در مواردی که برای رمزنگاری نیاز به کلید است، باید یک مکانیسم تبادل کلید نیز مورداستفاده قرار گیرد.

مکانیسم امضای دیجیتال (Digital Signature)

در امضای دیجیتال دو فرآیند انجام می‌پذیرد: امضا و بررسی امضا. فرستنده باید با اطلاعات خصوصی، محرمانه و منحصربه‌فرد خود امضا را انجام دهد. گیرنده نیز با استفاده از اطلاعاتی که در دسترس عموم است، باید بتواند درستی امضا را بررسی نماید.

i. فرستنده باید با اطلاعات خصوصی خود که کلیدخصوصی گفته می‌شود، واحد داده و یا چکیده آن را رمزگذاری نماید.
ii. فرآیند راستی آزمایی باید با استفاده از روش‌های عمومی به بررسی امضای فرستنده بپردازد.
iii. از ویژگی‌های مهم مکانیسم امضا این است که امضا فقط باید با استفاده از اطلاعات خصوصی فرستنده امکان‌پذیر باشد. در این صورت می‌توان جهت قضاوت و داوری، به آن استناد نمود.

مکانیسم کنترل دسترسی (Access Control)

این مکانیسم پس از احراز هویت، اجازه دسترسی به منابع مجاز را فراهم می‌کند. مکانیسم کنترل دسترسی ممکن است از یک نهاد دیگر جهت احراز هویت استفاده نماید.

اگر کاربری قصد دسترسی به منابع غیرمجاز را داشته باشد، مکانیسم کنترل دسترسی آن درخواست را رد کرده و پس از ثبت رخداد مربوطه، آن را به‌صورت یک آلارم (هشدار) به اطلاع مدیر شبکه می‌رساند.

مکانیسم کنترل دسترسی ممکن است به‌عنوان‌مثال از یک یا چند مورد زیر استفاده نماید:

i. پایگاه اطلاعات کنترل دسترسی؛ که در آن حقوق دسترسی مربوط به هر نهادی نگهداری می‌گردد. این اطلاعات ممکن است در مراکز مجوز یا توسط نهادی در

دسترس، و در قالب یک لیست کنترل دسترسی یا ماتریس با ساختار سلسله مراتبی یا توزیع‌شده، نگهداری گردد.

ii. اطلاعات احراز هویت مانند کلمات عبور، مالکیت و اقدامات پس‌ازآن؛ که مشخص می‌نماید نهاد موردنظر مجوز دسترسی دارد یا نه.

iii. صلاحیت‌ها، مالکیت و اقدامات پس‌ازآن؛ که مشخص‌کننده حقوق دسترسی به منابع تعریف‌شده، است.

iv. برچسب‌های امنیتی[1]؛ که با توجه به سیاست‌های امنیتی برای قبول[2] یا رد[3] دسترسی یک نهاد، مورداستفاده قرار می‌گیرند.

v. زمان اقدام به دسترسی.

vi. مسیر اقدام به دسترسی.

vii. مدت‌زمان دسترسی.

مکانیسم‌های کنترل دسترسی ممکن است در دو طرف ارتباط و همچنین در یک نقطه میانی مورداستفاده واقع شوند. کنترل‌های دسترسی در مبدأ و نقاط میانی برای تعیین اینکه آیا فرستنده، مجاز به برقراری ارتباط با گیرنده یا استفاده از منابع ارتباطی است، مورداستفاده قرار می‌گیرند.

مکانیسم‌های صحت دیتا

دو جنبه صحت دیتا عبارتند از : صحت یک واحد داده یا فیلد؛ و صحت جریانی از داده یا فیلد. به‌طورکلی از مکانیسم‌های مختلفی برای تأمین دو سرویس فوق استفاده می‌شود؛ اگرچه ارائه سرویس دومی بدون فراهم کردن سرویس اولی امکان‌پذیر نیست.

i. تعیین صحت یک واحد از اطلاعات، شامل یک فرآیند در ارسال‌کننده و یک فرآیند در دریافت‌کننده است. ارسال‌کننده به واحد دیتا یک مقدار که تابعی از خود دیتا است (مثل چکیده پیام) اضافه می‌نماید. البته ممکن است چکیده یا پیام را رمزگذاری نیز نماید. دریافت‌کننده نیز باید یک مقدار متناظر با آنچه ارسال‌کننده به‌عنوان چکیده پیام ارسال کرده، تولید نموده و با مقدار دریافت شده مقایسه کند. در صورت برابر بودن هر دو مقدار، گیرنده می‌تواند از صحت و تمامیت پیام دریافتی مطمئن باشد.

[1] Security labels
[2] Grant
[3] Deny

ii. در سرویس‌های اتصال‌گرا، جهت برقراری حفاظت از صحت دنباله‌ای از داده‌ها (یعنی حفاظت در برابر Misordering، بازپخش، از دست دادن یا تغییر) علاوه بر مورد قبلی، به مکانیسم دیگری مثل شماره توالی، مهر زمانی و یا زنجیره‌ای شدن دیتای رمزشده، نیاز داریم.

iii. در سرویس‌های غیراتصال‌گرا، می‌توان از مهر زمانی استفاده نمود.

مکانیسم تبادل احراز هویت[1]

مکانیسمی است که برای اطمینان از هویت یک نهاد، اقدام به تبادل اطلاعات مربوطه می‌نماید.

i. برخی از روش‌هایی که ممکن است برای تبادل احرازهویت مورداستفاده قرار گیرند، عبارتند از:
   ۱. استفاده از اطلاعات احراز هویت مثل کلمه عبور ارائه‌شده توسط فرستنده و بررسی انجام‌شده توسط گیرنده.
   ۲. روش‌های رمزنگاری.
   ۳. استفاده از ویژگی‌ها یا دارایی‌های یک نهاد.

ii. این مکانیسم ممکن است به‌منظور احرازهویت در هر لایه از OSI گنجانده شود. اگر مکانیسم موفق به احراز هویت یک نهاد نگردد، اتصال را قطع[2] یا رد[3] نموده و همچنین ممکن است رخداد را واقعه‌نگاری کرده و یا به مرکز مدیریت امنیت گزارش دهد.

iii. زمانی که از روش‌های رمزنگاری استفاده می‌شود، برای محافظت در برابر بازپخش، ممکن است با پروتکل Handshaking ترکیب شده باشند. (به‌عنوان‌مثال برای اطمینان از زنده‌بودن)

iv. انتخاب روش‌های تبادل احراز هویت، به شرایطی که باید از آن استفاده شود، بستگی دارد:
   ۱. مهر زمانی و همگام‌سازی ساعت.
   ۲. Two and Three way handshake، برای احراز هویت یک‌طرفه و متقابل.
   ۳. به دست آوردن سرویس عدم انکار توسط امضای دیجیتال و یا مکانیسم‌های گواهی رسمی.

---

[1] Authentication exchange mechanism
[2] Termination
[3] Rejection

مکانیسم توسعه ترافیک (Traffic padding mechanism)

مکانیسم توسعه ترافیک را می‌توان برای محافظت در مقابل تجزیه‌وتحلیل ترافیک مورداستفاده قرار داد. این مکانیسم تنها در صورتی می‌تواند مؤثر باشد که ترافیک توسعه داده‌شده توسط سرویس محرمانگی، محافظت‌شده باشد.

در مکانیسم توسعه ترافیک، داخل فاصله‌های جریان داده را با قرار دادن بیت‌هایی پر می‌کنند.

مکانیسم کنترل مسیریابی

i. مسیرها می‌توانند به‌صورت پویا یا از قبل تعریف‌شده، انتخاب شوند. بنابراین از آن‌ها تنها می‌توان برای امنیت زیر شبکه‌ها و لینک‌ها استفاده کرد.
ii. سیستم پایانی ممکن است به علت تشخیص حملات مداوم، از فراهم‌کننده خدمات شبکه بخواهد که ارتباط را از مسیری متفاوت برقرار سازد.
iii. ممکن است سیاست‌های امنیتی، حمل اطلاعات خاصی را از طریق یک زیر شبکه یا لینک ممنوع کرده باشد. همچنین آغازگر اتصال ممکن است برای استفاده از مسیرهای خاصی هشدار داده باشد.

مکانیسم گواهی رسمی (Notarization mechanism)

یک سری از ویژگی‌ها در مورد اطلاعات تبادل شده بین دو یا چند نهاد، مثل صحت، منبع، زمان و مقصد، می‌تواند با ارائه یک مکانیسم رسمی، تضمین شود. این تضمین باید توسط یک نهاد رسمی ثالث که توسط نهادهای ارتباطی قابل‌اعتماد ارائه‌شده، فراهم گردد. در ارتباطات ممکن است از مکانیسم‌های امضای دیجیتال، رمزنگاری و صحت استفاده شده باشد تا شخص ثالث بتواند سرویس مناسبی ارائه دهد. زمانی که چنین گواهی رسمی درخواست شده باشد، دیتای تبادل شده بین نهادها باید از طریق موارد ارائه‌شده، محافظت گردند.

فصل هشتم

تکنولوژی‌ها و ابزارهای امنیتی

مبحث اول

کنترل دسترسی

مبحث دوم

فایروال

مبحث سوم

IDS/IPS

مبحث چهارم

هانی‌پات

☑ مبحث اول

کنترل دسترسی

کنترل دسترسی جزء اولین اقدامات امنیتی است که باید جهت محافظت از منابع، مورد استفاده قرار گیرد. کنترل دسترسی مشخص می‌کند هر کاربر یا سیستم به کدام منابع و در چه سطحی می‌تواند دسترسی داشته باشد.

نحوه کنترل دسترسی بر اساس سیاست‌گذاری‌ها، برنامه‌ها و تکنولوژی‌ها مشخص می‌شود. کنترل دسترسی باید هم به‌صورت فیزیکی و هم به‌صورت منطقی مورداستفاده قرار گیرد. کنترل دسترسی‌ها می‌توانند به سه صورت اجباری، اختیاری و غیر اختیاری باشند.[۱]

۱. کنترل دسترسی اجباری (Mandatory Access Controls)

کنترل دسترسی اجباری که به‌اختصار MAC نامیده می‌شود، از طرح طبقه‌بندی داده‌ها[۲] برای اعطای دسترسی استفاده می‌کند؛ که در آن کاربران و مالک[۳]داده، کنترل محدودشده‌ای برای اعطای دسترسی به منابع اطلاعاتی را دارند. به‌عبارت‌دیگر نحوه و سطح دسترسی‌ها توسط سیستم و بر اساس سیاست‌های امنیتی مشخص می‌شود و حتی مالک داده نمی‌تواند به‌راحتی، اقدام به اعطای مجوز به سایر کاربران نماید.

در طرح طبقه‌بندی داده‌ها، مجموعه‌های اطلاعاتی را رتبه‌بندی می‌کنند؛ از طرفی به هر کاربر نیز امتیازی تعلق می‌گیرد که مشخص‌کننده سطح دسترسی او به اطلاعات رتبه‌بندی شده است. تعیین این امتیازات بر اساس میزان حساسیت، انجام‌گرفته و نشان‌دهنده سطح محرمانگی اطلاعات است.

MAC دارای یک ماتریس کنترل دسترسی است که در آن، نحوه ارتباط هر فاعل (Subject) را با هر مفعول (Object) مشخص می‌کند.

Subject\Object	File 1	File 2	Printer
MTR	Read\Write\Execute	Read\Write\Execute	Yes
Bob	Read	Read	No
Alice	-	Read\Write	Yes

[۱] در مبحث دوم فصل دوازدهم، سطوح امنیتی طی استاندارد DOD 85، به‌صورت مفصل تشریح شده است.

[2] Data Classification

[3] Owners

۲. **کنترل دسترسی غیراختیاری (Non-discretionary control)**

کنترل دسترسی غیراختیاری، یک نسخه "به‌شدت اجباری" از MAC است؛ که توسط یک قدرت مرکزی در سازمان، مدیریت‌شده و می‌تواند بر اساس نقش (Role-Based) یا بر اساس وظیفه (Task-Based)، وجود داشته باشد.

کنترل‌های مبتنی بر نقش (Role-Based)، به نقش یک کاربر در سازمان گره‌خورده و کنترل مبتنی بر وظیفه (Task-Based)، به وظیفه یا مسئولیت خاصی مرتبط است.

استفاده از کنترل‌های مبتنی بر نقش و وظیفه، مدیریت کنترل‌ها را آسان‌تر می‌نماید. در این صورت، مدیریت به‌جای اعطای جداگانه دسترسی به هر فرد، یک‌بار به هر نقشی دسترسی‌های موردنیاز را داده و ازآن‌پس افراد موردنظر را در آن نقش یا وظیفه قرار می‌دهد. با قرارگیری فرد در آن نقش یا وظیفه، تمام دسترسی‌های آن نقش به کاربر اعطا می‌شود و به‌محض خارج کردن کاربر از آن نقش یا وظیفه، تمام دسترسی‌های داده‌شده، باطل می‌شود. این کار مخصوصاً برای محیط‌هایی که جابجایی افراد زیاد است، می‌تواند تأثیرات خوبی داشته باشد.

۳. **کنترل دسترسی اختیاری (Discretionary Access Control)**

در کنترل دسترسی اختیاری که به‌اختصار DAC نامیده می‌شود؛ کاربر می‌تواند برای دیتای تحت مالکیت خود، اقدام به اعطای مجوز به سایر افراد نماید.

مکانیسم‌های کنترل دسترسی

به‌طورکلی، تمامی روش‌های کنترل دسترسی، بر اساس مکانیسم‌های زیر عمل می‌کنند:

۱. شناسایی (Identification)
۲. احرازهویت (Authentication)
۳. اختیارات یا مجوزدهی (Authorization)
۴. پاسخگویی (Accountability)

در ادامه هریک از چهار مکانیسم فوق تشریح می‌شوند:

شناسایی

مکانیسم شناسایی، به هر نهادی که قصد دسترسی به منابع را دارد، یک برچسب (Label) تخصیص می‌دهد تا از طرف سیستم قابل‌شناسایی باشد. برچسب داده‌شده به متقاضی را شناسه (Identifier) نامیده و به‌اختصار با ID نمایش می‌دهند. این ID باید به‌صورت منحصربه‌فرد و فقط به یک متقاضی اختصاص داده شود.

در سازمان‌های مختلف، تخصیص ID به روش‌های گوناگونی انجام می‌گیرد؛ ممکن است از شماره پرسنلی، کدملی و یا حروف نام و فامیل برای ایجاد ID استفاده شود. البته بعضی از سازمان‌ها برای افزایش امنیت، به‌جای استفاده از یک الگوی خاص، از کاراکترهای تصادفی در تخصیصID استفاده می‌نمایند.

احرازهویت

به فرآیند اعتبارسنجی هویت ادعا کننده، احرازهویت گفته می‌شود. سه مکانیسم تأیید هویت که به‌صورت گسترده‌ای مورداستفاده قرارگرفته یا به‌عنوان عوامل احراز هویت شناخته می‌شوند، به‌صورت زیر است:

- **چیزی که درخواست‌کننده می‌داند.**

این عامل از احرازهویت آن چیزی است که درخواست‌کننده می‌داند یا می‌تواند به یاد آورد؛ مثل کلمه عبور، عبارت عبور، کد تأیید منحصربه‌فرد یا یک شماره شناسایی شخصی[1] (PIN).

رمز عبور یک کلمه یا ترکیبی از کاراکترها است که تنها باید کاربر از آن اطلاع داشته باشد. یکی از مهم‌ترین مباحث در امنیت اطلاعات، پیچیدگی رمزعبور است؛ چراکه رمز عبور نباید یک کلمه راحت و مرتبط با شخص، مثل نام، شماره تلفن و تاریخ تولد باشد که به‌راحتی قابل حدس زدن است. و همچنین نباید به‌قدری سخت و پیچیده باشد که کاربر نتواند آن را به خاطر سپارد و مجبور به یادداشت آن باشد.

- **چیزی که درخواست‌کننده دارد.**

این عامل از احراز هویت، آن چیزی است که درخواست‌کننده در اختیار دارد و یا در صورت لزوم می‌تواند آن را تولید کند. به‌عنوان‌مثال کارت‌های غیرهوشمند[2]، مانند کارت خودپرداز (ATM) یا کارت شناسایی، که قسمت مغناطیسی آن‌ها، حاوی PIN رمزشده کاربر است. همچنین کارت هوشمند[3] که شامل یک تراشه کامپیوتری است، و می‌تواند بررسی و تأیید هویت را بر اساس اطلاعات مختلفی انجام دهد. دستگاه دیگری که می‌تواند برای احراز هویت مورداستفاده قرار گیرد، Token است. Token یک کارت یا کلید کوچکی است که دارای تراشه کامپیوتری و یک صفحه نمایشگر بوده و می‌تواند کلمات عبور مختلفی برای استفاده در احرازهویت، تولید نماید. Token به دو صورت

[1] Personal Identification Number
[2] Dumb Card
[3] Smart Card

همزمان و غیرهمزمان وجود دارد. در Token همزمان، باید سرور و Token باهم هماهنگ بوده تا در یک‌زمان خاص، یک عدد شبیه به‌هم جهت استفاده به‌عنوان کلمه عبور تولید کنند. اما در Token غیرهمزمان، نیازی نیست که تنظیمات ساعت آن‌ها مثل هم باشد؛ چراکه از سیستم پرسش و پاسخ برای تولید کلمه عبور استفاده می‌کنند. سیستم یک عبارت تولید می‌نماید، کاربر نیز عبارت را به Token داده و Token نیز بر اساس الگوریتم‌های توافق شده با سرور، یک رمزعبور تولید می‌نماید[۱]؛ کاربر نیز از آن رمز برای ورود به سیستم استفاده می‌کند.

- **چیزی که درخواست‌کننده هست.**

این فاکتور از احرازهویت، آن چیزی است که درخواست‌کننده هست یا می‌تواند آن را تولید کند. این فاکتور بر اساس خصوصیات شخصی موجود، مانند اثرانگشت، کف دست و عنبیه چشم، و یا خصوصیات شخصی، که درخواست‌کننده می‌تواند آن را تولید کند، مثل الگوی صدا و امضا، ایجاد می‌گردد.

مجوزدهی

به هر نهادی که بتواند احرازهویت شود، یک لیست از دارایی‌های اطلاعاتی به همراه سطح دسترسی مربوطه، جهت مشخص نمودن حدود اختیارات آن نهاد الصاق می‌شود. این لیست که مجوز نامیده می‌شود؛ معمولاً به‌صورت ACL یا ماتریس دسترسی وجود دارد.

به‌طورکلی مجوزدهی می‌تواند دریکی از سه روش زیر به کار گرفته شود:

۱. مجوز برای هر کاربر احرازهویت شده؛ در این روش سیستم پس از احرازهویت هر شخص، اقدام به اعطای مجوزدسترسی به همان شخص می‌نماید. یعنی به ازای هر شخص، این اعطای مجوز باید صورت پذیرد. به همین دلیل فرآیند مجوزدهی پیچیده‌تر شده و باعث استفاده بیشتر از منابع یک سیستم کامپیوتری می‌شود.
۲. مجوز برای اعضای یک گروه؛ در این روش اعطای مجوز به‌جای شخص، به گروه انجام می‌گیرد. هر شخص پس از احرازهویت، در گروه موردنظر قرارگرفته و از مجوزهای اعطاشده به آن گروه بهره‌مند می‌گردد. این روش رایج‌ترین روش اعطای مجوز دسترسی است.
۳. مجوزدهی سراسری برای سیستم‌های مختلف؛ در این روش احرازهویت و اعطای مجوز دسترسی توسط یک سیستم مرکزی انجام می‌پذیرد.

---

[۱] به این نوع رمز تولیدشده، کلمه عبور یک‌بارمصرف یا One Tim Password (OTP) نیز گفته می‌شود.

گاهی اوقات روش احرازهویت و اعطای مجوز دسترسی که به‌صورت مرکزی انجام می‌شود را SSO[1] می‌نامند. در این روش باید از یک ساختار دایرکتوری به اشتراک گذارده شده، برای عملیات احرازهویت و اعطای مجوز، استفاده شود. به‌عنوان یک نمونه از دایرکتوری به اشتراک گذاشته‌شده، می‌توان از LDAP[2] نام برد.

پاسخگویی

پاسخگویی، تضمین می‌کند که تمام اقدامات مجاز و غیرمجاز انجام‌شده در سیستم را می‌توان به یک شناسه (ID) احرازهویت شده نسبت داد. پاسخگویی، اغلب با استفاده از سیستم‌های واقعه‌نگاری[3] و سپس بازرسی[4] وقایع ثبت‌شده، انجام می‌پذیرد.

سیستم‌های واقعه‌نگاری، اقدام به ثبت اطلاعات خاصی از قبیل: تلاش ناموفق جهت دسترسی و تغییرات انجام‌شده در سیستم، می‌نمایند. این گزارش‌ها موارد استفاده دیگری ازجمله تشخیص نفوذ، علت بروز شکست در یک سیستم یا ردیابی استفاده از یک منبع خاص، را نیز دارند.

پروتکل AAA

پروتکل AAA که به‌صورت Triple A خوانده می‌شود، شامل سه سرویس Authentication، Authorization و Accounting است. تعریف هر سه سرویس مانند تعریف‌های ذکرشده فوق می‌باشد؛ با این تفاوت که در این پروتکل، سرویس Identification هم جزء سرویس Authentication در نظر گرفته شده است.

دلیل ایجاد پروتکل AAA، استفاده از سه سرویس (به عبارتی چهار سرویس) فوق در کنار هم و به‌صورت مرکزی است؛ چراکه برای برقراری امنیت به هر سه عامل فوق در کنار هم نیاز است. اگر هر سه عامل به‌صورت مرکزی توسط یک سیستم فراهم شود، ضمن کاهش سربار مدیریتی، باعث افزایش کارایی و امنیت نیز می‌گردد.

سازمان IETF طی RFC 4962، اقدام به ارائه راهنمایی‌های موردنیاز جهت راه‌اندازی و استفاده از پروتکل AAA نموده است. هرچند که استفاده از پروتکل AAA اجباری نیست؛ اما در صورت پیاده‌سازی آن، اجرای راهنمایی‌های ارائه‌شده در این سند الزامی است.

---

[1] Single Sign-on (SSO)
[2] Lightweight Directory Access Protocol (LDAP)
[3] Logging
[4] Auditing

پروتکل کربروس (Kerberos)

در افسانه‌های یونان باستان، سگی سه سر بر دروازه الماس(دنیای زیرین) ایستاده و ارواح را به درون دنیای مردگان هدایت می‌کند، اما از خروج آن‌ها جلوگیری می‌نماید. سپس ارواح را نزد سه داور برده تا در مورد آن‌ها قضاوت نموده و رأی صادر کنند. بر اساس رأی سه داور گناهکاران و نیکان مشخص می‌شوند؛ گناهکاران به عذاب جاودانی و نیکان به سعادت ابدی رهسپار می‌شوند.[۱]



دانشگاه MIT، یک پروتکل رایگان و تحت قانون حق مؤلف، جهت امنیت احرازهویت در شبکه ارائه نموده و نام آن را کربروس گذاشت. کربروس بر اساس سیستم رمزنگاری متقارن ایجاد گردیده و برای انجام عملیات خود نیاز به یک Third-Party تأییدشده در شبکه دارد. البته برحسب نیاز، امکان استفاده از رمزنگاری نامتقارن به‌جای رمزنگاری متقارن در کربروس وجود دارد.

اساس ایجاد کربروس، احرازهویت در شبکه‌های ناامن است. با توجه به اینکه در هنگام تبادل اطلاعات مربوط به احراز هویت، ممکن است هکر اطلاعات را Sniff کرده و آن‌ها را به دست آورد؛ لذا کربروس از مکانیسم‌های امنیتی برای تبادل اطلاعات مربوط به احرازهویت استفاده می‌نماید تا حتی در صورت شنود اطلاعات، محرمانگی آن حفظ شود.

---

[۱] این هم از نکته اخلاقی این پروتکل. :)

☑ مبحث دوم

فـــایروال (دیوار آتش)

در ساختمان‌سازی از یک دیوار سنگی یا بتنی به نام دیوار آتش، بین سقف طبقه پایین و کف طبقه بالا استفاده می‌شود تا در صورت بروز آتش‌سوزی، از گسترش آن به طبقات دیگر جلوگیری به عمل آید. در هواپیما و خودرو نیز از عایق‌هایی جهت دیوار آتش استفاده می‌شود. این عایق بین موتور و محل قرارگیری سرنشین که معمولاً اشتعال‌زا است، قرارگرفته و جان مسافرین را حفظ می‌نماید.

دیوار آتش در امنیت اطلاعات نیز برای محافظت قسمت قابل‌اطمینان شبکه، در مقابل جهان خارج یا قسمت‌های کمتر قابل‌اطمینان، مورداستفاده قرار می‌گیرد.

فایروال (دیوار آتش) ممکن است در قالب یک سیستم کامپیوتری مجزا، یک سرویس نرم‌افزاری در حال اجرا بر روی روتر یا سرور و یا به‌صورت تجهیزات مستقل سخت‌افزاری، در شبکه وجود داشته باشد.

فایروال‌ها را می‌توان بر اساس سه مؤلفه زیر تقسیم‌بندی نمود:

۱. روش پردازش (Processing mode)
۲. دوران توسعه (Development era)
۳. ساختار (Structure)

گروه اول؛ روش‌های پردازش فایروال

فایروال‌ها بر اساس روش‌های پردازشی به پنج گروه اصلی زیر تقسیم می‌شوند:

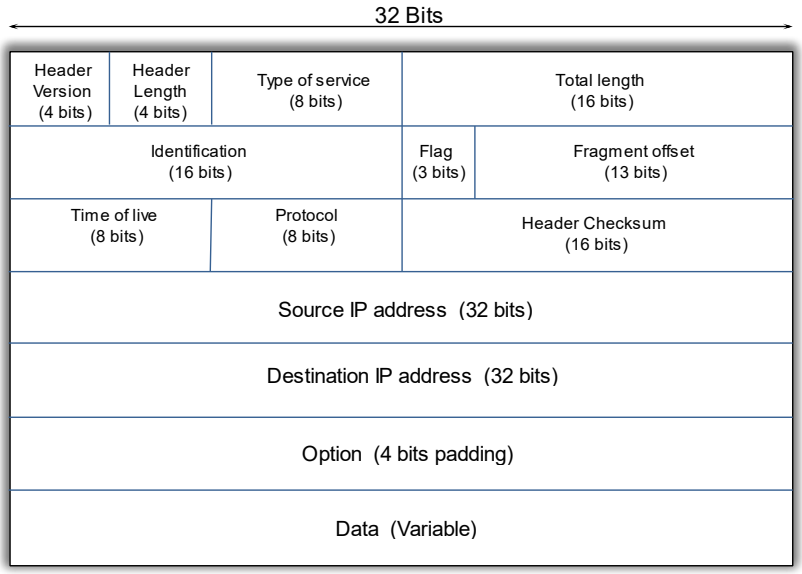
۱. فایروال فیلترینگ بسته‌ها (Packet Filtering Firewall)
۲. دروازه برنامه (Application Gateway)
۳. دروازه مدار (Circuit Gateway)
۴. فایروال لایه MAC (MAC Layer Firewall)
۵. چندگانه (Hybrid)

فایروال‌های چندگانه از ترکیب چهار حالت دیگر به وجود می‌آیند. در عمل اکثر فایروال‌های امروزی را می‌توان در گروه فایروال ترکیبی قرار داد.

۱. فایروال فیلترینگ بسته‌ها

این نوع فایروال‌ها به بررسی اطلاعات سرآیند[1] بسته‌های دیتایی که به شبکه وارد یا از آن خارج می‌شود، می‌پردازند. فایروال فیلترینگ بسته، اقدام به بررسی سرآیند بسته‌های TCP/IP نموده و بر اساس قوانین[2] برنامه‌ریزی‌شده بر روی خود، تشخیص می‌دهد که باید بسته را ممنوع[3] کرده و یا به آن اجازه عبور و رسیدن به شبکه بعدی را بدهد.

این نوع فایروال‌ها می‌توانند فیلترینگ را بر اساس تمام اطلاعات کلیدی موجود در سرآیند بسته، مثل آدرس مبدأ[4]، آدرس مقصد[5] و نوع بسته[6] انجام دهند. شکل زیر ساختار بسته‌های IPv4 IPv4 را نمایش می‌دهد.



فایروال فیلترینگ بسته‌ها، عملیات خود را بر اساس اطلاعات موجود در لایه سوم مدل OSI انجام می‌دهد. فایروال اقدام به اسکن بسته‌های شبکه می‌نماید تا انطباق یا عدم انطباق اطلاعات موجود در سرآیند را با قوانین خود، تشخیص دهد. درصورتی‌که فایروال بسته‌ای پیدا کند که طبق قوانین باید محدود شود، آن بسته را از ادامه مسیر به شبکه بعدی بازمی‌دارد.

---

[1] Header
[2] Rules
[3] Deny
[4] Source Address
[5] Destination Address
[6] Packet Type

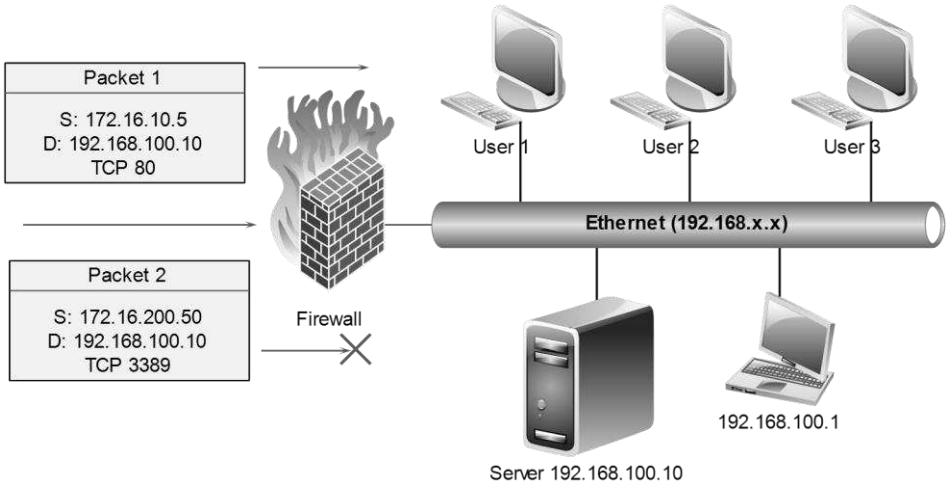
محدودیت‌هایی که فایروال اعمال می‌کند، می‌تواند بر اساس یک یا ترکیبی از موارد زیر باشد:

- آدرس‌های مبدأ و مقصد.
- مسیر ورودی[1] یا خروجی[2].
- پورت‌های مبدأ و مقصد مربوط به پروتکل‌های TCP و UDP.

جدول زیر یک نمونه از قوانین موجود بر روی فایروال Packet-filtering، را نمایش می‌دهد:

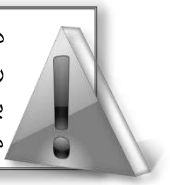
Source Address	Destination Address	TCP/UDP Port	Action
172.16.x.x	192.168.x.x	TCP 80 (HTTP)	Allow
172.16.x.x	192.168.x.x	TCP 3389 (RDP)	Deny

در قانون اول گفته‌شده که اگر بسته‌ای با آدرس مبدأ 172.16.x.x به مقصدی در شبکه 192.168.x.x بر اساس پورت 80 پروتکل TCP ارسال‌شده است، اجازه عبور را داشته باشد. اما در قانون دوم گفته‌شده که اگر بسته‌ای با آدرس مبدأ 172.16.x.x به مقصدی در شبکه 192.168.x.x بر اساس پورت 3389 پروتکل TCP ارسال‌شده است، اجازه عبور را نداشته باشد. به‌عبارت‌دیگر مدیر شبکه اجازه دسترسی HTTP از شبکه 172.16.x.x به 192.168.x.x را صادر کرده ولی دسترسی RDP[3] (ریموت دسکتاپ) را ممنوع کرده است.



[1] Inbound
[2] Outbound
[3] Remote Desktop Protocol (RDP)

این نوع فیلترینگ در بسیاری از روترها و از طریق نوشتن ACL در دسترس است. اما متأسفانه روترها و فایروال‌های Packet-Filtering، قادر به تشخیص اینکه آیا سرآیند پیام عوض‌شده است یا نه! نیستند؛ لذا در برابر روش‌هایی مثل Spoofing، آسیب‌پذیر می‌باشند.



درصورتی‌که اطلاعات سرآیند یک بسته با هیچ‌یک از قوانین (Allow, Deny) موجود بر روی فایروال یا ACL تطابق پیدا نکند، قانون Deny، برای آن بسته در نظر گرفته می‌شود. به‌عبارت‌دیگر می‌توان گفت که فایروال قانونAny Any Deny (یعنی هر مبدأ به هر مقصدی بر روی هر پورتی، ممنوع) را به‌صورت خودکار به انتهای قوانین، اضافه می‌نماید.

فایروال‌های Packet-filtering، دارای سه زیرمجموعه می‌باشند:

i. **Static Filtering**

برای فیلترینگ ایستا نیاز است که در زمان نصب فایروال قوانین فیلترینگ نیز بر روی آن پیکربندی گردد. نوشتن قوانین و ترتیب قرارگیری آن‌ها توسط یک شخص(مدیر شبکه) انجام‌شده و اعمال هرگونه تغییرات بعدی نیز با دخالت انسان باید انجام شود. این نوع فیلترینگ بر روی روتر و Gateway مشترک است.

ii. **Dynamic Filtering**

در فیلترینگ پویا، فایروال می‌تواند به یک رویداد اورژانسی، واکنش نشان داده، به‌روز شود و قانونی برای مقابله با آن رویداد ایجاد نماید.

این واکنش می‌تواند مثبت باشد؛ وقتی‌که به یک کاربر داخلی اجازه شرکت در یک فعالیت خاص داده می‌شود. یا واکنشی منفی باشد؛ زمانی که بسته‌های اطلاعاتی از یک آدرس خاص به‌صورت ناهنجار افزایش پیداکرده است.

iii. **Stateful Inspection**

فایروال Stateful، ارتباطاتی که بین سیستم‌های داخلی با خارج از شبکه برقرارشده را توسط جدول وضعیت[1]، پیگیری می‌نماید. جدول وضعیت، حاوی اطلاعاتی در مورد وضعیت بسته و اینکه چه کسی و چه زمانی آن را ارسال کرده، می‌باشد.

فایروال Stateful عملیات فیلترینگ را همانند فایروال‌های Stateless انجام می‌دهد، ولی با یک گام بیشتر.

فایروال‌های ساده یا همان Stateless، فیلترینگ را بر اساس اطلاعات هر بسته انجام می‌دهند. یعنی اگر ما یک قانون داشته باشیم که ارتباط پورت ۸۰ را از داخل به خارج مجاز دانسته باشد، ولی قانونی نداشته باشیم که ارتباط پورت ۸۰ را از خارج به داخل

[1] State Table

مجاز کرده باشد؛ در این صورت کاربر داخلی پیام درخواست ارتباط را ارسال کرده و از فایروال نیز عبور می‌کند، ولی فایروال جواب درخواست کاربر را که از طرف سرور ارسال‌شده، ممنوع می‌کند. درنتیجه ارتباط برقرار نمی‌شود. حال اگر مدیر شبکه یک قانون ایجاد کند که ارتباط پورت ۸۰ را از بیرون به داخل مجاز نماید، شبکه آسیب‌پذیر خواهد شد.

اما در فایروال‌های Stateful، اطلاعات مربوط درخواست‌هایی که اجازه عبور از فایروال را دارند، در جدولی به نام جدول وضعیت ثبت می‌شود. در این صورت زمانی که جواب یک درخواست مجاز به فایروال می‌رسد، بدون آنکه قانونی برای آن در فایروال نوشته‌شده باشد، بر اساس جدول وضعیت امکان عبور پیدا خواهد کرد. در این صورت امنیت نیز افزایش پیدا می‌کند؛ چراکه فقط بسته‌هایی که در جواب درخواست‌های مجاز ارسال گردیده، امکان عبور از فایروال را دارند؛ ولی زمانی که شروع ارتباط از بیرون شبکه باشد، به دلیل وجود نداشتن قانون موردنظر، ورود بسته ممنوع خواهد شد.

۲. دروازه برنامه

دروازه برنامه به‌عنوان یک فایروال سطح کاربرد یا فایروال برنامه نیز شناخته می‌شود. این فایروال به‌صورت جداگانه از فیلترینگ، بر روی یک کامپیوتر نصب می‌شود، ولی در ارتباط با فیلترینگ فعالیت می‌کند. دروازه برنامه به‌عنوان یک پروکسی سرور نیز شناخته می‌شود؛ چراکه برای درخواست سرویس‌ها به‌صورت پروکسی عمل می‌کند.

به‌عنوان‌مثال، دروازه برنامه بر روی یک سرور در شبکه داخلی نصب می‌گردد، ازآن‌پس تمام درخواست کاربران برای دیدن صفحات وب، به این سرور ارسال‌شده و سرور نیز به نمایندگی از کاربر، درخواست را برای سرور وب ارسال کرده و جواب آن را نیز برای کاربر داخلی ارسال می‌کند. البته پروکسی سرور می‌تواند صفحات تازه بازدید شده را در حافظه Cache خود نگهداری نموده و در جواب درخواست‌های شبیه به هم ارسال نماید.

برای حفاظت بیشتر می‌توان سرور پروکسی را در قسمت DMZ شبکه قرار داد. در این صورت امکان فیلترینگ ارتباطات بین سرور پراکسی و کاربر داخلی نیز فراهم می‌گردد.

همان‌طور که از نام فایروال سطح کاربرد مشخص است، این نوع فایروال برای کار در لایه Application مدل OSI کاربرد دارد. لذا عیب اصلی فایروال سطح کاربرد این است که معمولاً برای یک یا چند پروتکل خاص طراحی‌شده و به‌راحتی نمی‌توان برای محافظت در برابر حملات دیگر، آن را پیکربندی نمود. همچنین به دلیل اینکه این نوع فایروال باید اطلاعات بسته‌ها را در لایه‌های بالا(لایه ۵ و۶ و۷) پردازش کند، لذا به منابع بیشتری نیاز داشته و زمان پردازش نیز بالا می‌رود.

۳. دروازه مدار

فایروال دروازه مدار، عملیات خود را در لایه چهارم مدل OSI، یعنی لایه Transport، انجام می‌دهد. این نوع فایروال عملیات فیلترینگ انجام نمی‌دهد، بلکه از برقراری ارتباط مستقیم بین دو شبکه جلوگیری به عمل می‌آورد.

یک دروازه سطح مدار، اتصالات TCP را بازپخش می‌کند، اما هیچ پردازش اضافی یا فیلترینگی بر روی آن اعمال نمی‌کند.[۱]

دروازه مدار به نیابت از کاربر داخلی، اتصالات TCP و UDP را با شبکه‌های بیرونی برقرار می‌کند.[۲]

شبیه حالت پروکسی، وقتی ارتباطات توسط دروازه مدار با شبکه خارجی برقرار می‌گردد، اطلاعات درخواست‌کننده داخلی (مثل آدرسIP) از دید شبکه خارجی پنهان می‌ماند.

ترجمه آدرس شبکه[۳](NAT) نوعی از دروازه مدار است. در این روش، آدرس تمام درخواست‌های داخل به بیرون شبکه، به آدرس عمومی (Public) تغییریافته و سپس به دست گیرنده خارجی می‌رسد.

۴. فایروال لایه MAC

فایروال لایه MAC، در لایه دوم مدل OSI عملیات خود را انجام می‌دهد. این نوع فایروال فیلترینگ را بر اساس آدرس MAC[۴] مربوط به کارت شبکه، اعمال می‌کند.

فایروال لایه MAC، شناسایی کامپیوترها را بر اساس آدرس MAC انجام می‌دهد، به همین دلیل در جدول ACL، از آدرس MAC به‌عنوان شناسه کامپیوتر، استفاده می‌شود.

۵. فایروال چندگانه

فایروال‌های چندگانه یا Hybrid، از ترکیب انواع دیگر فایروال‌ها به وجود می‌آیند. هر سیستم فایروال، کار خود را مستقل انجام می‌دهد، اما با فایروال‌های دیگر نیز مرتبط بوده و درنهایت یک خروجی را تولید می‌نمایند.

فایروال‌هایی که امروزه مورداستفاده قرار می‌گیرند، غالباً در گروه فایروال‌های چندگانه تعریف می‌شوند.

---

[۱] توصیفی برای فایروال سطح مدار، از آقای John Wack در مستندات NIST SP 800-10.

[۲] توصیفی برای فایروال سطح مدار، از آقای محمدتقی روغنی در کتاب امنیت اطلاعات. :)

[3] Network Address Translation (NAT)

[4] Media Access Control (MAC)

گروه دوم؛ تقسیم‌بندی بر اساس دوران توسعه

منظور از دوران توسعه، نسل‌های مختلف فایروال است. فایروال‌ها نیز همانند هر تکنولوژی دیگر، در طول زمان و با پیشرفت علم IT، دچار تغییراتی گردیده و نسل به نسل پیشرفته‌تر شده است. اولین نسل فایروال‌ها بر روی روتر قرار داشته و خیلی ساده اقدام به فیلتر بسته‌ها می‌کردند. اما نسل‌های اخیر فایروال به‌طور فزاینده‌ای پیچیده‌تر گردیده و امنیت را نیز افزایش داده است.

در حال حاضر می‌توان فایروال‌ها را در پنج نسل مختلف، گروه‌بندی نمود:

- **فایروال نسل اول**

فایروال‌های Static Packet Filtering، یا فیلترینگ ایستای بسته‌ها، نسل اول فایروال‌ها را تشکیل می‌دهند. این فایروال‌ها، فیلترینگ ورودی و خروجی شبکه را بر اساس اطلاعات سرآیند بسته‌های IP انجام می‌دهند.

- **فایروال نسل دوم**

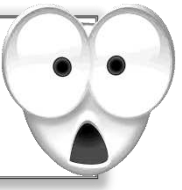
فایروال‌های سطح برنامه (Application Level Firewall) و پروکسی‌ها (Proxy)، در این نسل قرار می‌گیرند. این سیستم‌ها معمولاً به‌صورت جداگانه از فایروال نصب می‌شدند ولی هم‌راستا با فایروال عمل می‌کردند. (البته در آن زمان بیشتر از فیلترینگ بسته‌ها توسط روتر استفاده می‌گردید.)

- **فایروال نسل سوم**

فایروال‌های Stateful Inspection، نسل سوم فایروال‌ها هستند. همان‌طور که گفتیم، فایروال Stateful، اقدام به نظارت ارتباط بین شبکه‌ها، بر اساس جدول وضعیت می‌نماید.

- **فایروال نسل چهارم**

فایروال‌های نسل چهارم بانام فیلترینگ پویای بسته‌ها (Dynamic Packet Filtering) شناخته می‌شوند. در این نوع فایروال‌ها اجازه داده می‌شود که صرفاً یک بسته اطلاعاتی با یک آدرس مبدأ، مقصد و شماره پورت خاص، از طریق فایروال ارسال گردد.



نسل سوم و چهارم فایروال‌ها خیلی به یکدیگر نزدیک هستند، به‌طوری‌که در برخی منابع، هردوی آن‌ها به‌عنوان یک نسل فایروال معرفی‌شده‌اند.
فایروال‌های پویا اجازه برقراری ارتباط از بیرون شبکه را بر اساس جدول وضعیت صادر می‌نمایند.

- **فایروال نسل پنجم**

در فایروال‌های نسل پنجم، ارزیابی بسته در چندلایه مختلف از مدل OSI صورت می‌گیرد. به‌نوعی می‌توان گفت فایروال‌های نسل‌های مختلف در یکدیگر ادغام‌شده است.

Next-Generation Firewall

برخی شرکت‌های بزرگ مثل سیسکو، محصولات فایروال جدید خود را به‌عنوان نسل جدید فایروال (Next-Generation) معرفی می‌کنند. این فایروال‌ها از ادغام ماژول‌های مختلفی مثل فایروال، IDS، IPS ، فیلترینگ برنامه و فیلترینگ محتوا، ایجاد گردیده‌اند.

اما به نظر من این تجهیزات را نمی‌توان فایروال نامید؛ چراکه از ماژول‌های مختلفی که دارای تعاریف مخصوص به خود هستند تشکیل‌شده، که یکی از آن‌ها، فایروال است. البته بعضی از برندهای مشهور دیگر مثل جونیپر، نام این‌گونه تجهیزات را UTM[1] گذاشته‌اند، که به عقیده من، نام مناسب‌تری است.

جونیپر، اصطلاح UTM را به این صورت تعریف می‌کند:

"UTM اصطلاحی است برای توصیف ترکیبی از چندین ویژگی امنیتی در یک دستگاه، برای حفاظت از انواع تهدیدات. مزیت UTM در نصب، راه‌اندازی و مدیریت چندین قابلیت امنیتی، به صورتی ساده است."

گروه سوم؛ تقسیم‌بندی بر اساس ساختار

نوع دیگری از تقسیم‌بندی فایروال‌ها، می‌تواند بر اساس ساختار پیاده‌سازی آن‌ها، صورت پذیرد. فایروال‌های تجاری معمولاً به‌صورت یک سخت‌افزار جداگانه و مستقل ارائه می‌شود. این سخت‌افزارها دارای Firmware یا سیستم‌عامل مخصوص به خود بوده و تمام پردازش آن‌ها نیز بر روی سخت‌افزار فایروال انجام می‌پذیرد.

گروه دیگری از فایروال‌ها هستند که به‌صورت نرم‌افزاری ارائه می‌گردند؛ که خود این گروه به دودسته تقسیم می‌شوند. یک سری از فایروال‌های نرم‌افزاری نیاز به سیستم‌عامل ندارند و می‌توانند به‌صورت مستقیم بر روی سخت‌افزار یک سرور نصب‌شده و همانند فایروال سخت‌افزاری نقش خود را در شبکه ایفا نمایند. اما نوع دیگر فایروال‌های نرم‌افزاری، وابسته به سیستم‌عامل هستند؛ یعنی نسبت به برند و مدل، بر روی سیستم‌عامل‌های مختلفی قابلیت نصب دارند. در این صورت فایروال انحصاراً برای همان سیستم عمل می‌کند. البته برای پردازش‌های خود نیز از سخت‌افزار همان سیستم استفاده می‌نماید.

[1] Unified Threat Management (UTM)

دسته سوم فایروال‌هایی هستند که در شبکه‌های کوچک‌تر که پردازش کمتری دارند، مثل دفاتر کوچک، شعب سازمان‌های بزرگ و حتی منازل مسکونی، مورداستفاده قرار می‌گیرند. در اصطلاح به تجهیزات دفاتر کوچک SOHO[1] گفته می‌شود.

البته دفاتر کوچک یا منازل می‌توانند از فایروال‌های نرم‌افزاری بهره ببرند. در سال‌های اخیر اکثر تولیدکنندگان آنتی‌ویروس، ماژول فایروال را نیز به آنتی‌ویروس خود افزوده و در اختیار کاربران قرار داده‌اند.

انواع مناطق در شبکه

یکی از کاربردهای مهم فایروال، ایجاد مناطق (Zone) مختلف در شبکه است. گروه‌بندی سیستم‌ها و کاربران در مناطق مختلف می‌تواند سربار مدیریتی را کاهش و سطح امنیتی را افزایش دهد. فایروال‌ها به‌صورت پیش‌فرض سه منطقه را در شبکه به وجود آورده و معمولاً یک سری قانون (Rule) پیش‌فرض نیز برای روابط بین مناطق تعریف می‌کنند. این قوانین پیش‌فرض باعث می‌شود حتی اگر مدیر شبکه پیکربندی درستی روی فایروال انجام نداده باشد، وجود فایروال یک سطح حداقلی امنیت را برای شبکه فراهم آورد.

فایروال به ازای هر Zone با Zoneهای دیگر یک فایروال(مجازی) جداگانه ایجاد می‌نماید؛ به‌طوری‌که قوانین نوشته‌شده بین هر دو منطقه کاملاً مستقل از قوانین همان منطقه با مناطق دیگر خواهد بود.

تمام فایروال‌ها دارای سه Zone پیش‌فرض به‌صورت زیر می‌باشند. البته بنده پیشنهاد می‌کنم به دلیل اهمیت سرورها، حداقل یک Zone دیگر بانام Server Farm نیز به آن‌ها افزوده شود.

۱. **منطقه قابل‌اطمینان (Trusted Zone)**

منظور از منطقه قابل‌اطمینان، منطقه‌ای است که مدیریت آن بر عهده شما بوده و سیاست‌های امنیتی سازمان شما در آنجا اعمال می‌شود. معمولاً شبکه داخلی سازمان به‌عنوان منطقه قابل‌اطمینان معرفی می‌شود.

قوانین پیش‌فرض فایروال، تمام اتصالاتی که مبدأ آن، منطقه قابل‌اطمینان باشد را به هر منطقه دیگری مجاز می‌دانند. اما برخورد فایروال با اتصالاتی که مبدأ آن‌ها مناطق دیگر بوده و به مقصد منطقه قابل‌اطمینان ارسال‌شده، سخت‌گیرانه است.

در بعضی از فایروال‌ها، برای این نوع منطقه از نام LAN نیز استفاده می‌نمایند.

[1] Small Office/Home Office (SOHO)

۲. **منطقه غیرقابل‌اطمینان (Untrusted Zone)**

منطقه‌ای که دارای مدیریت مستقل بوده و از سیاست‌های امنیتی سازمان شما پیروی نمی‌کند را منطقه غیرقابل‌اطمینان می‌گویند. اتصالات اینترنت همواره به‌عنوان منطقه غیرقابل اطمینان معرفی می‌شوند. البته در ارتباط بین سازمانی نیز، بهتر است لینک سازمان همکار را در منطقه غیرقابل اطمینان قرار دهید.

برای برقراری اتصالاتی که مبدأ آن‌ها منطقه غیرقابل‌اطمینان است، فایروال قوانین پیش‌فرض سخت‌گیرانه‌ای دارد.

۳. **منطقه کمتر حفاظت‌شده (Demilitarized Zone)**

منطقه کمتر حفاظت‌شده یا DMZ، منطقه‌ای است که مدیریت آن بر عهده شما بوده و باید از سیاست‌های امنیتی شما پیروی کند؛ اما بنا به دلایلی مجبورید که سیاست‌های خیلی راحت یا خیلی سخت‌گیرانه به آن اعمال نکنید.

معمولاً سرورهایی که باید از طریق اینترنت یا شبکه‌های عمومی دیگر به مشتریان سرویس دهند، در این منطقه قرار می‌گیرند.

قوانین پیش‌فرض فایروال برای این منطقه، نه خیلی راحت و نه خیلی سخت‌گیرانه است.

۴. **منطقه سرورها (Server Zone)**

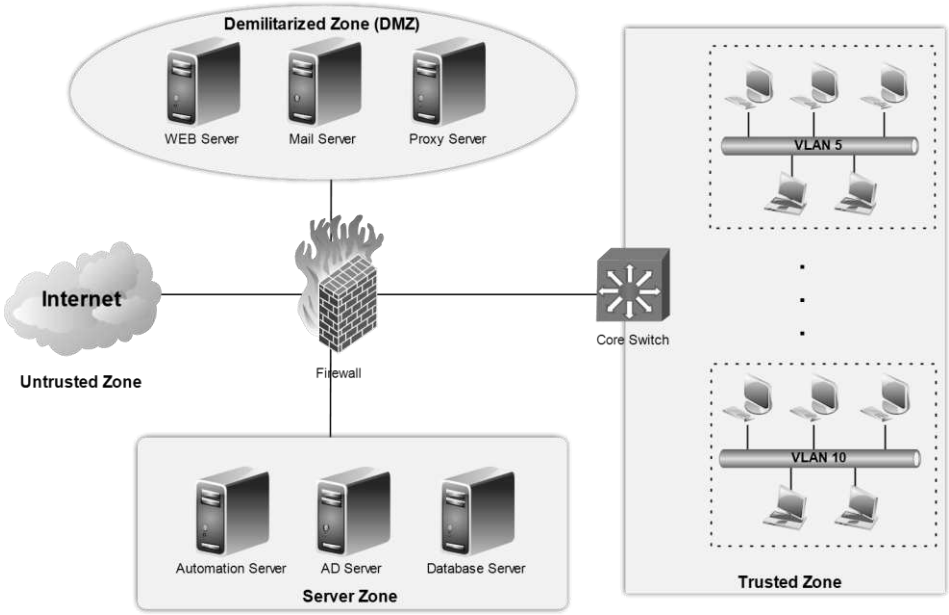
این منطقه به‌صورت پیش‌فرض در فایروال‌ها موجود نیست، لذا Rule پیش فرضی هم ندارد. اما به دلیل اهمیت سرورها، پیشنهاد می‌کنم حتماً یک منطقه مخصوص سرورها ایجاد نمایید.

البته بعضی از کارشناسان شبکه، برای سرورها یک VLAN جداگانه در نظر گرفته و سعی می‌کنند با ACL امنیت آن را فراهم کنند؛ البته این کار درستی است، اما در صورت داشتن یک فایروال قدرتمند که توانایی اضافه کردن یک Zone دیگر را دارد، می‌توان VLAN سرورها را به‌عنوان یک منطقه فایروال تعریف کرد. در این صورت امنیت سرورها به میزان قابل‌توجهی افزایش پیدا می‌کند. ضمن اینکه مدیریت قوانین فایروال خیلی راحت‌تر و انعطاف‌پذیرتر از مدیریت ACL روترها است.

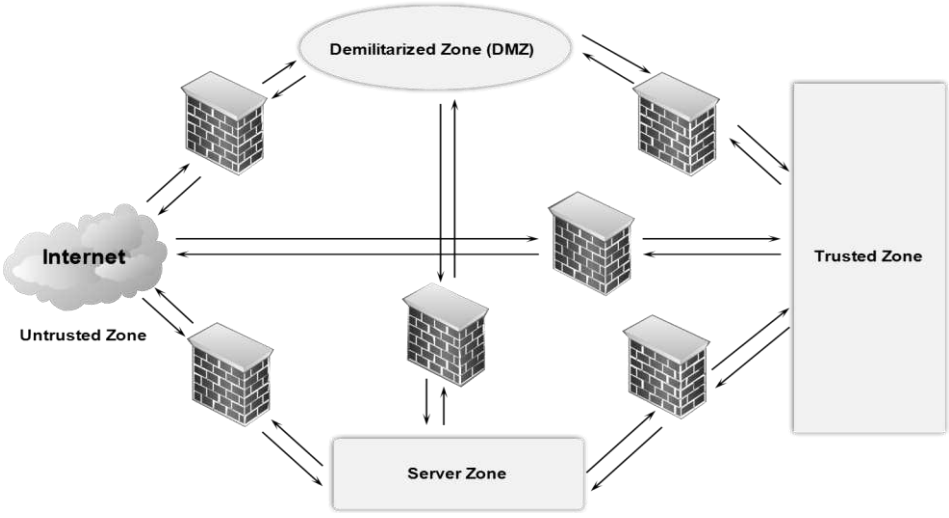
تصویر زیر نمایش‌دهنده ایجاد مناطق مختلف توسط فایروال است. البته شما می‌توانید به‌دلخواه مناطق دیگری را بر اساس سه منطقه پیش‌فرض، ایجاد نموده و نام‌گذاری نمایید؛ لذا من یک منطقه بر اساس Trusted Zone ایجاد کرده و نامش را Server Zone گذاشتم.

منطقه‌بندی به هر دو صورت فیزیکی و منطقی در تصویر زیر نمایش داده‌شده است:

این تصویر نشان‌دهنده نحوه اتصال **فیزیکی** فایروال به Zoneها می‌باشد؛ هر منطقه (Zone) به یک اینترفیس فایروال متصل گردیده است.



اما ازلحاظ **منطقی**، بین هر Zone ایجادشده با سایر مناطق، یک فایروال قرار می‌گیرد که ازنظر وضعیت ایجاد Rule از سایر فایروال‌های مستقل است.



بهترین شیوه پیکربندی فایروال

پیکربندی فایروال باید بر اساس الزامات و سیاست‌های امنیتی سازمان انجام شود. شناخت سرویس‌های موردنیاز و شبکه‌هایی که باید در دسترس باشند، می‌تواند در پیکربندی کارآمد فایروال نقش مهمی داشته باشد.

توجه داشته باشید که صرف وجود تجهیزات و نرم‌افزارهای امنیتی، باعث افزایش امنیت اطلاعات نمی‌شود. حتی در بسیاری از موارد عدم پیکربندی صحیح، باعث بروز ایرادات زیادی در شبکه نیز شده است.



در غالب موارد، تجهیزات ارزان‌قیمتی که توسط یک کارشناس خبره پیکربندی‌شده، کارایی به‌مراتب **بهتری** نسبت به تجهیزات گران‌قیمتی داشته، که توسط افراد کم‌اطلاع پیکربندی گردیده است.
با خرید محصولات امنیتی، تنها ۳۰ درصد از ارزش آن را کسب کرده‌اید؛ و ۷۰ درصد ارزش باقیمانده را با پیکربندی صحیح به دست خواهید آورد.

به‌هرحال در زمان پیکربندی تجهیزات فایروال، بهتر است موارد زیر را رعایت نمایید:

- تمام ترافیک خروجی از منطقه قابل‌اعتماد، مجاز است. این اجازه باعث می‌شود که کاربران سازمان به سرویس‌های موردنیاز، دسترسی داشته باشند.
البته درصورتی‌که سیاست‌های امنیتی سازمان ایجاب کند، می‌توان ترافیک خروجی را فیلتر و واقعه‌نگاری نمود.
- هیچ‌گاه نباید دستگاه فایروال برای مقاصد پیکربندی و مدیریتی، از طریق شبکه عمومی به‌صورت مستقیم، در دسترس باشد.
برای دسترسی مدیریتی به تجهیزات، بهتر است از مکانیسم‌های تأیید هویت امن که ترجیحاً از رمزنگاری استفاده نموده باشند، بهره برد.
- دیتای مربوط به پست الکترونیک (SMTP)[1] باید اجازه ورود به شبکه را داشته باشد. البته بهتر است قبل از تحویل به سرور یا کلاینت‌ها، برای انجام عملیات فیلترینگ از یک دروازه SMTP عبور نماید.
- تمام پیام‌های پروتکل ICMP[2]، برای شبکه‌های عمومی باید ممنوع شود. هکرها از سرویس‌های ICMP، جهت شناسایی شبکه استفاده می‌نمایند. همچنین امکان استفاده از ICMP برای حملات منع خدمات نیز وجود دارد.

[1] Simple Mail Transport Protocol (SMTP)
[2] Internet Control Message Protocol (ICMP)

- دسترسی Telnet از طریق شبکه عمومی به تمام سرویس‌دهنده‌های داخلی باید ممنوع باشد. حداقل اینکه، دسترسی Telnet به سرور DNS باید مسدود باشد.
اگر کاربران سازمان نیاز دارند از طریق شبکه عمومی به سرویس‌ها دسترسی داشته باشند، باید این کار را از طریق VPN یا دیگر سرویس‌های امن انجام دهند.
- زمانی که یک سرویس‌دهنده وب را برای ارائه خدمات به کاربران خارج از شبکه (کاربران شبکه عمومی یا اینترنت) ارائه می‌نمایید، دسترسی کاربران داخلی به آن سرویس‌دهنده باید از طریق پروکسی یا معماری DMZ باشد.
- تمام دیتایی که قابل‌تأیید نیستند، باید ممنوع شوند. یکی از راه‌های حمله این است که هکر یک آدرس داخلی را به‌عنوان آدرس مبدأ در سرآیند بسته قرار می‌دهد. برای جلوگیری از این مشکل، باید تمام ترافیک خارجی که آدرس مبدأ آن‌ها، جزء آدرس‌های داخلی است، ممنوع شوند.[۱]
- بسته‌های ترافیک داخل به خارج سازمان که آدرس IP مبدأ آن‌ها جزء آدرس‌های داخلی نیست، باید ممنوع شود.[۲]
- فیلتر بسته‌هایی که آدرس مبدأ یا مقصد آن‌ها جزء آدرس‌های موجود در RFC 3330[۳] است.

[1] RFC 2827
[2] RFC 2827
[۳] استاندارد RFC 3330 شامل رنج آدرس‌هایی است که توسط IANA به موارد خاص اختصاص داده شده است.

☑ مبحث سوم

IDS/IPS

زمانی که یک مهاجم سعی در گرفتن دسترسی غیرمجاز یا ایجاد اختلال درروند عملیات یک سیستم اطلاعات را دارد، یک نفوذ[1] رخ می‌دهد. تقریباً همیشه نفوذ به‌قصد آسیب رساندن انجام می‌شود. حتی اگر نفوذ به‌صورت خود انتشار، توسط ویروس یا حملات توزیع‌شده منع خدمت، انجام شود، بازهم هدف شخص مهاجم، آسیب رساندن به سازمان بوده است.

البته هر نوع دسترسی غیرمجاز، درنهایت منجر به آسیب‌رسانی به سازمان خواهد شد؛ ولی مهاجمین معمولاً رغبتی به شناخته شدن ندارند و فقط به فکر کسب منافع خود هستند. اما در طرف مقابل مزاحمین غالباً به دنبال شهرت هستند؛ لذا از خرابکاری نیز در این راه بهره می‌برند.

در سایت Intrusion.com، فرمول جالبی برای امنیت گفته‌شده است:

Security = Visibility + Control
کنترل + پدیداری = امنیت

برای مقابله با نفوذ و نفوذگر، دو تکنولوژی سیستم تشخیص نفوذ (Intrusion Detection System) و سیستم پیشگیری از نفوذ (Intrusion Prevention System) را می‌توان مورداستفاده قرار داد؛ در راستای فرمول فوق، IDS را برای Visibility و IPS را برای Control.

سیستم تشخیص نفوذ (IDS)، عبارت است از روش‌ها و سیستم‌هایی که نفوذ را شناسایی می‌کنند. واکنش نفوذ، شامل اقدامات سازمان است که پس از تشخیص نفوذ انجام می‌پذیرد. این اقدامات جهت از بین بردن یا محدود کردن نفوذ و بازگشت عملیات به وضعیت عادی است. اما پس‌ازآن اقدامات اصلی برای از بین بردن کامل نفوذ و شناسایی منبع نفوذ است. اگر منبع نفوذ شناسایی نشود امکان گزیده شدن مجدد از همان سوراخ، وجود خواهد داشت.

سیستم پیشگیری از نفوذ (IPS)، شامل فعالیت‌هایی است که جهت جلوگیری از نفوذ انجام می‌شود. استفاده از تجهیزات و تکنولوژی‌هایی مثل فایروال و IPS، تنها یکی از اقدامات مؤثر در پیشگیری از نفوذ است. برای داشتن یک فرآیند پیشگیری از نفوذ مناسب، باید به عوامل دیگری ازجمله: سیاست‌گذاری مناسب در امنیت اطلاعات و اجرای صحیح آن، راه‌اندازی و آزمایش اقدامات متقابل امنیت اطلاعات و آموزش کارکنان، نیز توجه کافی داشت.

[1] Intrusion

سیستم‌های IDS تجاری اواخر دهه 1990 وارد بازار شدند. سیستم IDS همانند یک زنگ خطر کارکرده و در صورت بروز یک نقض، زنگ را فعال می‌سازد تا مدیر شبکه از نقض روی‌داده مطلع شود. هشدار IDS می‌تواند همانند آنچه در فیلم‌های سینمایی سایبری نشان می‌دهند خیلی پرسروصدا بوده و ضمن به صدا درآوردن آژیر، چراغ‌های قرمزی را روشن و خاموش کند! و البته می‌تواند خیلی ساکت و بی‌سروصدا بوده و هشدار مربوطه را از طریق پست الکترونیک، SMS و پیجر، به اطلاع مدیر امنیت اطلاعات برساند. نوع هشدار و اتفاقی که منجر به تولید هشدار می‌گردد، توسط مدیر امنیت قابل پیکربندی بر روی تجهیزات IDS است.

پیشرفت تکنولوژی‌ها، باعث در هم آمیختن آن‌ها با یکدیگر می‌شود. برای IDS و IPS نیز همین اتفاق روی‌داده است. اکثر محصولات امنیتی موجود در بازار، IDS و IPS را در کنار هم ارائه می‌نمایند. اما واقعیت آن است که IPS فقط می‌تواند از نفوذهایی که قبلاً شناخته‌شده و IPS Signature آن‌ها در دیتابیس موجود است، جلوگیری به عمل آورد. حال‌آنکه وظیفه اصلی IDS تشخیص نفوذهایی است که تاکنون شناخته‌نشده‌اند. البته مکانیسم‌هایی وجود دارد که IDS در صورت تشخیص نفوذ، به فایروال یا IPS دستور جلوگیری از آن را بدهد؛ ولی به هر صورت مأموریت اصلی IDS، تشخیص نفوذهای جدید و ناشناخته است و نه جلوگیری از آن‌ها. از طرف دیگر، همان‌طور که گفتیم تشخیص نفوذ بر اساس تنظیماتی که مدیر شبکه بر روی IDS انجام داده، صورت می‌گیرد و ممکن است تشخیص نفوذی که صورت گرفته درست نباشد؛ لذا بهتر است نتایج IDS توسط کارشناس امنیت بررسی‌شده و سپس برای جلوگیری از آن اقدام نمایند.

نحوه قرارگیری تجهیزات IDS و IPS در شبکه متفاوت است. IPS باید جلوی ترافیک مربوط به نفوذ را مسدود نماید، لذا در مسیر ترافیک[1] قرار می‌گیرد. به‌عبارت‌دیگر، همانند فایروال، ترافیک باید از درون IPS عبور کند تا بتواند وظیفه خود را به‌درستی انجام دهد. اما IDS هیچ تأثیری روی جریان ترافیک نمی‌گذارد، پس لازم نیست ترافیک از درون آن عبور کند. IDS دارای حسگرهایی است که در محل‌های موردنظر قرارگرفته و یک کپی از ترافیک عبوری را جهت تجزیه‌وتحلیل برای IDS ارسال می‌نمایند.

بسیاری از قابلیت‌ها و فناوری‌های IDS و IPS، مشابه یکدیگر هستند. از طرفی اکثر برندهای امنیتی هر دو تکنولوژی را باهم ادغام نموده و در قالب یک محصول ارائه می‌نمایند. در راهنمای NIST SP800-94[2] نیز برای بررسی قابلیت‌های مشابه IDS و IPS، از عبارت IDPS، استفاده شده است. به دلیل اینکه برخی مطالب این مبحث از راهنمای NIST نقل‌شده، لذا ما هم از عبارت IDPS (البته برای قابلیت‌های مشترک) استفاده می‌کنیم.

---

[1] In-line

[2] http://csrc.nist.gov/publications/nistpubs/800-94/SP800-94.pdf

وظایف کلیدی تکنولوژی IDPS

روش‌های مورداستفاده توسط IDPSها برای تشخیص و شناسایی حوادث، متفاوت است. ولی همه آن‌ها، برای نظارت و تجزیه‌وتحلیل رویدادها در جهت شناسایی فعالیت‌های نامطلوب، به‌طورمعمول وظایف زیر را انجام می‌دهند:

- ثبت اطلاعات مربوط به وقایعی که اتفاق افتاده است. این اطلاعات معمولاً به‌صورت محلی ذخیره‌شده و یک نسخه از آن نیز ممکن است برای سیستم‌های جداگانه دیگری مثل سرورهای مرکزی ثبت رویداد، SIEM[1] و سیستم‌های مدیریتی ارسال گردد.
- اطلاع‌رسانی[2] رویدادهای مهم به مدیر امنیت اطلاعات. این اطلاع‌رسانی می‌تواند از روش‌های مختلفی مثل پست الکترونیک، پروتکل SNMP[3]، پیام‌های Syslog، رابط کاربری IPDS و اسکریپت، صورت پذیرد.

اطلاع‌رسانی معمولاً شامل اطلاعات اولیه در مورد یک رویداد است، و مدیر امنیت برای دریافت اطلاعات بیشتر باید به تجهیزات IDPS مراجعه نماید.

- ایجاد گزارش‌ها. گزارش‌ها شامل خلاصه‌ای از تمام رویدادها، به همراه جزئیاتی از رویدادهای خاص و قابل‌توجه، است.

بعضی از IDPSها قادرند در صورت تشخیص یک تهدید جدید، مشخصات امنیتی مربوطه را تغییر دهند. به‌عنوان‌مثال، IDPS ممکن است اطلاعات دقیقی از یک Session که در آن یک فعالیت مخرب شناسایی‌شده، جمع‌آوری نماید. همچنین پس‌ازآنکه یک تهدید خاص شناسایی شد، IDPS باید بتواند تنظیمات مربوط به هشدارها یا اولویتی که باید به آن‌ها اختصاص داده شود، را تغییر دهد.

اما همان‌طور که گفته شد، برخلاف IDS، تکنولوژی IPS پس از شناسایی یک تهدید، سعی می‌کند از آن جلوگیری نماید. IPS از روش‌های مختلفی جهت پاسخ به تهدیدات استفاده می‌کند که می‌توان آن‌ها را در گروه‌های زیر تقسیم‌بندی نمود:

- خود IPS اقدام به متوقف کردن حمله می‌نماید. به‌عنوان‌مثال:
  i. پایان دادن[4] به اتصال شبکه یا Session کاربر، که برای حمله مورداستفاده قرارگرفته است.

[1] Security Information and Event Management (SIEM)
[2] Notification
[3] Simple Network Management Protocol (SNMP)
[4] Terminate

ii. مسدود کردن[1] دسترسی به هدف یا اهداف، بر اساس ویژگی‌های مهاجم؛ مانند اکانت و آدرس IP کاربر متخلف. (مسدود کردن دسترسی مهاجم)

iii. مسدود کردن تمام دسترسی‌ها به میزبان، سرویس، برنامه و دیگر منابعی که مورد هدف قرارگرفته شده است. (مسدود کردن تمام دسترسی‌ها)

- تغییر محیط امنیتی، توسط IPS؛ به این معنی که برای مختل کردن یک حمله، IPS می‌تواند کنترل‌های امنیتی دیگر را تغییر دهد. به‌عنوان‌مثال، IPS می‌تواند برای مسدود کردن دسترسی مهاجم به هدف، اقدام به پیکربندی مجدد تجهیزات شبکه (سوئیچ، فایروال و روتر) نماید. حتی بعضی از IPSها می‌توانند در صورت تشخیص آسیب‌پذیر بودن یک سیستم، وصله‌های[2] امنیتی مربوطه را به آن اعمال نمایند.
- تغییر محتوای حملات، توسط IPS. برخی از IPSها قادر به حذف و یا جایگزینی بخش‌های مخرب یک حمله می‌باشند. به‌عنوان‌مثال، IPS می‌تواند فایل آلوده پیوست شده به یک Email را حذف نماید. به‌عنوان مثالی پیچیده‌تر، IPS می‌تواند به‌عنوان یک پروکسی عمل کرده و سرآیند بسته‌ها را تغییر دهد. در این صورت باعث ایجاد اختلال در بعضی از حملات می‌گردد.

یکی دیگر از ویژگی‌های مشترک IDS و IPS، امکان بروز اشتباه در تشخیص حملات است. هنگامی‌که IDPS یک فعالیت درست را، اشتباهاً به‌عنوان فعالیت مخرب شناسایی کند، می‌گوییم مثبت کاذب[3] رخداده و هنگامی‌که IDPS نتواند یک فعالیت مخرب را به‌درستی شناسایی کند، می‌گوییم منفی کاذب[4] رخداده است. به‌هرحال بروز مثبت کاذب و منفی کاذب اجتناب‌ناپذیر است؛ به همین دلیل برخی از سازمان‌ها برای جلوگیری از بروز اختلال، IDPS را طوری پیکربندی می‌کنند که در برابر وقایع مخرب واکنش نشان داده، ولی در برابر وقایعی که احتمال تخریب کمتری دارند، به ایجاد گزارش و ثبت وقایع بسنده نماید. پس‌ازآن مدیر امنیت، گزارش‌ها را مطالعه کرده و درباره نحوه برخورد، تصمیم‌گیری می‌کند.

مقابله با روش‌های گریز[5]، از دیگر مواردی است که اغلب IDPSها به آن می‌پردازند. منظور از روش‌های گریز، تغییرات ظاهری است که مهاجم در فرمت و زمان‌بندی فعالیت مخرب انجام می‌دهد تا توسط IDPS قابل‌شناسایی نباشد، ولی به همان میزان اثرات مخرب داشته باشد.

---

[1] Block
[2] Patch
[3] False Positive
[4] False Negative
[5] Evasion techniques

روش‌های تشخیص

روش‌های تشخیص مورداستفاده IDPSها را می‌توان در سه گروه اصلی تقسیم‌بندی نمود. البته برای فراهم‌آوری تشخیص بهتر و گسترده‌تر، IDPSها معمولاً چند روش را به‌صورت همزمان مورداستفاده قرار می‌دهند.

۱. **تشخیص مبتنی امضا (Signature-Based Detection)**

امضا، یک الگوی مربوط به یک تهدید شناخته شده است. در این روش برای شناسایی حوادث، امضای به‌دست‌آمده از رویدادها با امضای تهدیدات شناخته‌شده مقایسه می‌گردد. نمونه‌هایی از امضا به‌صورت زیر است:

i. تلاش شبکه راه دور برای Telnet بانام کاربری root؛ که این کار نقض سیاست امنیتی یک سازمان است.
ii. ایمیل با موضوع تصاویر رایگان به همراه فایل پیوست شده بانام pic.exe؛ که از ویژگی‌های یک شکل شناخته‌شده از بدافزارها است.
iii. ثبت Log با کد 645 توسط سیستم‌عامل؛ که نشانده این است که Auditing غیرفعال شده است.[۱]

متأسفانه روش تشخیص امضا در مقابله با روش‌های گریز ناتوان است. مثلاً در نمونه‌های فوق اگر نام فایل پیوست شده به‌جای pic.exe، نام pic1.exe داشته باشد، با امضا مطابقت نکرده و به‌عنوان تهدید شناخته نمی‌شود! تشخیص مبتنی بر امضا در مقابل تهدیدات ناشناخته هم که اصلاً اثری ندارد!

۲. **تشخیص مبتنی بر رفتار خلاف قاعده (Anomaly-Based Detection)**

تشخیص انحرافات ازآنچه به‌عنوان رفتار طبیعی شناخته می‌شود را تشخیص مبتنی بر رفتار خلاف قاعده یا ناهنجاری می‌گویند. IDPSهای مبتنی بر رفتار خلاف قاعده، دارای مشخصات (پروفایل) مربوطه به رفتارهای طبیعی کاربران، میزبان‌ها، اتصالات شبکه و برنامه‌های کاربردی می‌باشند. این مشخصات (پروفایل) با نظارت بر ویژگی‌های فعالیت‌های معمول در یک دوره زمانی، جمع‌آوری می‌گردند.
به‌عنوان‌مثال، ممکن است پروفایل یک شبکه حاکی از آن باشد که فعالیت وب‌سایت به‌طور متوسط ۱۳ درصد از پهنای باند اینترنت را در یک ساعت مورداستفاده قرار می‌دهد؛ لذا درصورتی‌که پهنای باند مورداستفاده توسط وب‌سایت به بیش‌ازحد انتظار برسد، یک هشدار برای مدیر شبکه ارسال می‌گردد.

---

[۱] غیرفعال کردن Auditing، برای جلوگیری از امکان ردیابی و پاک کردن ردپا توسط هکر صورت می‌پذیرد.

یک مزیت بزرگ تشخیص مبتنی بر رفتار خلاف قاعده، توانایی شناسایی تهدیدات ناشناخته است. به‌عنوان‌مثال، فرض کنید یک کامپیوتر با بدافزار جدیدی آلوده‌شده است. این بدافزار می‌تواند با ارسال نامه‌های حجیم و ایجاد تعداد زیادی اتصالات شبکه، از منابع شبکه و سیستم بیش‌ازحد ثبت‌شده در مشخصات (پروفایل) سیستم استفاده کند. در این صورت IDPS با مقایسه رفتار ناهنجار جدید با رفتار ثبت‌شده در پروفایل، می‌تواند وجود بدافزار را تشخیص دهد.

برای دور زدن تشخیص مبتنی بر رفتار خلاف قاعده، مهاجم استفاده از منابع شبکه را با یک شیب ملایم، افزایش می‌دهد. به دلیل اینکه مشخصات جمع‌آوری‌شده از یک کاربر یا میزبان، در طول یک دوره زمانی خاص مثلاً یک‌هفته‌ای بوده، لذا مهاجم با افزایش ملایم هفتگی، می‌تواند بدون اینکه حساسیت IDPS را برانگیزد، به فعالیت مخرب خود ادامه دهد.

از طرف دیگر، تشخیص مبتنی بر رفتار خلاف قاعده ممکن است دچار مثبت کاذب شود. مثلاً اگر یک فعالت خاص مثل انتقال فایل حجیم در شبکه، یک‌بار در ماه انجام شود، ممکن است IDPS آن را به‌عنوان فعالیت مخرب شناسایی کرده و از آن جلوگیری کند. تصور کنید که انتقال فایل حجیم در ماه به علت گرفتن پشتیبان از سرورهای سازمان بوده و IDPS آن را مسدود کرده است؛ این تشخیص مثبت کاذب می‌تواند منجر به از دست دادن اطلاعات مهم سازمان شود.

مشکل دیگر درروش تشخیص مبتنی بر رفتار خلاف قاعده، دشواری تحلیل هشدارهای اعلام‌شده توسط این روش برای تحلیل‌گران است. به دلیل پیچیدگی حوادث و تعداد اتفاقاتی که ممکن است باعث تولید هشدار شود، تحلیل‌گران به‌سختی می‌توانند مشخص کنند که هشدار تولیدشده درست بوده یا مثبت کاذب رخ‌داده است.

برای رفع مشکلاتی مثل تشخیص اشتباه، کارشناس امنیت باید پیکربندی IDPS را با شناسایی درست فرآیندهای شبکه و با دقت کامل انجام دهد.

۳. تحلیل وضعیت پروتکل (Stateful Protocol Analysis)

فرآیند مقایسه مشخصات از پیش تعیین‌شده فعالیت‌های یک پروتکل با هر وضعیت مشاهده‌شده از آن پروتکل، در جهت شناسایی انحرافات را تحلیل وضعیت پروتکل می‌گویند. استفاده از کلمه Stateful در عبارت Stateful Protocol Analysis، به این معنی است که IDPS قادر به درک و ردیابی وضعیت (State) پروتکل‌های لایه شبکه، لایه انتقال و لایه برنامه کاربردی است.

در تشخیص مبتنی بر رفتار خلاف قاعده، مبنای تشخیص، مشخصات (پروفایل) جمع‌آوری‌شده توسط خود IDPS بود. اما در تحلیل وضعیت پروتکل، مبنای تشخیص، مشخصات ارائه‌شده توسط ارائه‌دهنده پروتکل است.

به‌عنوان‌مثال هنگامی‌که کاربر یک نشست[1] پروتکل انتقال فایل (FTP) را شروع می‌کند، این نشست ابتدا در وضعیت احرازهویت نشده قرارگرفته و کاربر فقط باید بتواند چند فرمان ساده مربوط به اطلاعات راهنما یا ورود نام کاربری و کلمه عبور را اجرا نماید. قسمت مهم درک وضعیت، جفت‌وجور شدن درخواست با پاسخ است، بنابراین وقتی تلاش برای احرازهویت صورت می‌گیرد، IDPS می‌تواند با بررسی کد وضعیت پاسخ مربوطه، مشخص کند که احراز هویت موفقیت‌آمیز بوده یا نه. درصورتی‌که احراز هویت موفقیت‌آمیز باشد، نشست در وضعیت احرازهویت شده قرارگرفته و کاربر می‌تواند از فرمان‌های مختلف استفاده نماید. انجام بسیاری از فرمان‌هایی که در وضعیت احراز هویت نشده، مشکوک به‌حساب می‌آمد، در وضعیت احراز هویت شده، امری عادی تلقی می‌شود.

تحلیل وضعیت پروتکل می‌تواند توالی غیرمنتظره فرمان‌ها، مثل تکرار دستورات شبیه به‌هم و یا اجرای فرمان بدون رعایت فرمان اولیه، را شناسایی کند. همچنین در صورت احرازهویت، IDPS می‌تواند استفاده مجاز مربوط به هر نشست را ردیابی کرده و فعالیت‌های مشکوک را ثبت نماید. این اطلاعات می‌تواند کمک بزرگی برای تحقیق در مورد یک حادثه باشد.

همان‌طور که گفته شد روش تحلیل وضعیت پروتکل، بر اساس مدل پروتکل که معمولاً توسط استانداردها معرفی‌شده، می‌باشد. اما ممکن است بعضی از جزئیات ارائه‌شده در استانداردها در زمان پیاده‌سازی پروتکل، اجرا نشوند. از طرف دیگر برخی از فروشندگان و تولیدکنندگان تجهیزات و نرم‌افزارها، با کم‌وزیاد کردن ویژگی‌هایی درون پروتکل‌ها، باعث نقض استاندارد مربوطه می‌گردند. همچنین پروتکل‌های اختصاصی برخی از برندها که اطلاعات جامعی درباره آن‌ها در دسترس نیست، نیز باعث اختلال درروش تحلیل وضعیت پروتکل می‌گردند. برای رفع این مشکلات، باید اطلاعات مربوط به پروتکل‌ها، برای IDPS به‌روزرسانی شود.

ایراد اولیه روش تحلیل وضعیت پروتکل، استفاده زیاد از منابع جهت تجزیه‌وتحلیل‌های پیچیده است. ایراد دیگر این است که اگر مهاجم رفتار معمولی پروتکل را نقض نکند، امکان شناسایی فعالیت مخرب از طریق IDPS فراهم نمی‌گردد.

[1] Session

انواع تکنولوژی IDPS

تکنولوژی IDPS دارای انواع مختلفی است که بر اساس نوع رویداد و راه‌های نظارت و استقرار[1]، به‌صورت کلی می‌توان آن را در چهار گروه زیر تقسیم‌بندی نمود:

۱. **مبتنی بر شبکه (Network-Based)**

درروش مبتنی بر شبکه، IDPS بر روی ترافیک دستگاه یا بخش خاصی از شبکه نظارت کرده و برای شناسایی فعالیت‌های مخرب و مشکوک، اقدام به تجزیه‌وتحلیل فعالیت‌های پروتکل شبکه و نرم‌افزارهای کاربردی، می‌نماید. این روش می‌تواند انواع مختلفی از رویدادهای قابل‌توجه را شناسایی نماید.

درروش مبتنی بر شبکه، IDPS اغلب در مرز بین شبکه‌ها، ازجمله در نزدیکی مرز فایروال یا روتر، سرورهای شبکه خصوصی (VPN)، سرورهای دسترسی راه دور[2] و شبکه‌های بی‌سیم، مستقر می‌شود.

۲. **بی‌سیم (Wireless)**

درروش بی‌سیم، IDPS بر روی ترافیک شبکه بی‌سیم نظارت کرده و اقدام به تجزیه‌وتحلیل پروتکل‌های شبکه بی‌سیم می‌نماید؛ تا بتواند فعالیت‌های مشکوک مربوط به این پروتکل‌ها را شناسایی کند. این روش نمی‌تواند فعالیت‌های مشکوک در لایه‌های بالاتر از لایه شبکه، مثل لایه Application، را شناسایی نماید.

در این روش، IDPS در محدوده شبکه‌های بی‌سیم سازمان مستقر می‌شود تا بتواند بر روی ترافیک بی‌سیم نظارت نماید. البته می‌توان آن را در مکان‌هایی که ممکن است شبکه‌های بی‌سیم غیرمجاز رخ دهد، نیز مستقر نمود.

۳. **تحلیل رفتار شبکه (Network Behavior Analysis)**

تحلیل رفتار شبکه به‌اختصار NBA نیز نامیده می‌شود. در این روش، نظارت بر روی ترافیک شبکه، باهدف شناسایی تهدیداتی که با تولید جریان ترافیک غیرمعمول به وجود می‌آیند، صورت می‌پذیرد؛ مثل حملات توزیع‌شده منع خدمت (DDoS)، نرم‌افزارهای مخرب (کرم، تروجان و ...) و نقض سیاست‌های امنیتی.

سیستم‌های NBA اغلب برای نظارت بر جریان ترافیک شبکه‌های داخلی سازمان، مستقر می‌شوند. البته از آن‌ها می‌توان برای نظارت بر روی جریان ترافیک بین شبکه سازمان با شبکه‌های خارجی(مثل اینترنت یا سازمان‌های دیگر)، نیز استفاده نمود.

---

[1] Deploy

[2] Remote access servers

۴. **مبتنی بر میزبان (Host-Based)**

در این روش، نظارت بر روی ویژگی‌ها و رویدادهای یک میزبان خاص انجام‌شده، تا حوادث مشکوکی که برای آن میزبان ممکن است رخ دهد، شناسایی گردد. به‌عنوان‌مثال، IDPS بر مواردی مثل ترافیک مربوط به میزبان، سیستم واقعه‌نگاری، فرآیندهای در حال اجرا، فعالیت برنامه‌ها، دسترسی به فایل و تغییرات پیکربندی سیستم، نظارت می‌کند.

در این روش IDPS، اغلب بر روی میزبان‌های حساس مثل سرورهای در دسترس عموم و سرورهای حاوی اطلاعات حساس، مستقر می‌شوند.

اجزای IDPS

معمولاً IDPSها شامل اجزای زیر هستند:

۱. **سنسور یا عامل (Sensor or Agent)**

سنسورها و عوامل، بر روی فعالیت‌ها نظارت کرده و آن‌ها را تجزیه‌وتحلیل می‌کنند. اصطلاح سنسور معمولاً برای IDPSهایی استفاده می‌شود که وظیفه نظارت بر روی شبکه رادارند؛ مانند تکنولوژی مبتنی بر شبکه، بی‌سیم و تحلیل رفتار شبکه. اما اصطلاح عامل (Agent) برای IDPSهای مبتنی بر میزبان مورداستفاده قرار می‌گیرد.

۲. **سرور مدیریت (Management Server)**

سرور مدیریت، یک دستگاه مرکزی است که اطلاعات را از سنسورها و عوامل، دریافت کرده و آن‌ها را مدیریت می‌کند. سرور مدیریت، هم می‌تواند تجزیه‌وتحلیل اطلاعات مربوط به رویدادهایی را انجام دهد که سنسورها و عوامل توانسته‌اند آن‌ها را شناسایی کنند؛ و هم می‌تواند به تجزیه‌وتحلیل آن دسته از رویدادهایی بپردازد که سنسورها و عوامل نتوانسته‌اند آن‌ها را شناسایی کنند.

سرور مدیریت به تطابق اطلاعات مربوط به رویدادهایی که از چندین سنسور و عامل مختلف دریافت شده، می‌پردازد؛ مانند پیدا کردن رویدادهای اتفاق افتاده با یک آدرس IP خاص. به این عمل در اصطلاح Correlation گفته می‌شود.

سرورهای مدیریت به هر دو حالت سخت‌افزاری و نرم‌افزاری وجود دارند. در شبکه‌های کوچک ممکن است از سرور مدیریت استفاده نشود. اما در شبکه‌های بزرگ که اطلاعات از حساسیت بیشتری برخوردارند و همچنین تعداد سنسورها نیز بیشتر است، ممکن است از چند سرور مدیریت استفاده شود.

۳. **سرور دیتابیس (Database Server)**

سرورهای دیتابیس، مخازنی[1] هستند برای ثبت اطلاعات رویدادهایی که توسط سنسورها، عوامل و یا سرور مدیریت، جمع‌آوری گردیده شده است. بسیاری از IDPSها از سرور دیتابیس پشتیبانی می‌کنند.

۴. **کنسول (Console)**

کنسول برنامه‌ای است که یک رابط[2] کاربری را برای ارتباط با کاربران و مدیران IDPS فراهم می‌آورد. نرم‌افزارهای کنسول معمولاً بر روی یک سیستم دسکتاپ یا لپ‌تاپ نصب می‌گردند. بعضی از کنسول‌ها فقط توسط مدیران، برای وظایفی از قبیل پیکربندی سنسورها و عوامل، و اعمال به‌روزرسانی‌ها مورداستفاده قرار می‌گیرند. بعضی دیگر نیز فقط برای نظارت و تجزیه‌وتحلیل مورداستفاده قرار می‌گیرند. البته برخی از کنسول‌ها نیز هر دو قابلیت مدیریت و نظارت را فراهم می‌آورند.

قابلیت‌های امنیتی IDPS

تکنولوژی IDPS، می‌تواند طیف وسیعی از قابلیت‌های امنیتی را ارائه دهد. این قابلیت‌ها را می‌توان در چهار دسته عمومی و به‌صورت زیر تقسیم‌بندی نمود:

۱. **قابلیت جمع‌آوری اطلاعات (Information Gathering Capabilities)**

برخی از IDPSها قابلیت جمع‌آوری اطلاعاتی از قبیل فعالیت‌های مشاهده‌شده بر روی میزبان‌ها و شبکه را فراهم می‌آورند. به‌عنوان‌مثال شناسایی میزبان‌ها، سیستم‌عامل‌ها و برنامه‌هایی که بر روی آن‌ها استفاده می‌شود؛ و همچنین شناسایی ویژگی‌های کلی شبکه.

۲. **قابلیت واقعه‌نگاری (Logging Capabilities)**

به‌طورمعمول IDPSها فراهم آورنده امکان واقعه‌نگاری گسترده‌ای از اطلاعات مربوط به رویدادهای شناسایی‌شده، می‌باشند. این داده‌ها می‌توانند برای اعتبارسنجی[3] هشدارها، بررسی حوادث، و ارتباط رویدادها بین IDPSها و دیگر منابع واقعه‌نگاری، مورداستفاده قرار گیرند.

فیلدهای مربوط به دیتا که به‌صورت معمول توسط IDPSها استفاده می‌شود، شامل: تاریخ و زمان رویداد، نوع رویداد، درجه اهمیت (مثل اولویت، شدت، تأثیر) و اقدام

---

[1] Repository
[2] Interface
[3] Confirm the validity

پیشگیری انجام‌شده (در صورت وجود)، است. انواع خاصی از IDPSها دارای یک فیلد اضافه دیگر نیز می‌باشند؛ مانند IDPSهای مبتنی بر شبکه، جهت بسته به‌دست‌آورده شده[1] و IDPSهای مبتنی بر میزبان، جهت ثبت شناسه کاربر.

معمولاً IDPSها به مدیران شبکه اجازه می‌دهند که ذخیره رویدادها را به‌صورت محلی انجام داده و یک کپی از آن را برای یک سرور ثبت رویداد مرکزی (مثل Syslog یا نرم‌افزار مدیریت رویدادها و اطلاعات امنیتی) نیز ارسال نمایند.

به‌طورکلی برای پشتیبانی از صحت و در دسترس بودن دیتا، ذخیره وقایع‌نگاری باید به هر دو صورت محلی و مرکزی انجام پذیرد. همچنین IDPSها باید از طریق پروتکل NTP[2] یا به‌صورت دستی، هماهنگ‌سازی ساعت خود را انجام دهند؛ به‌طوری‌که اطلاعات ثبت‌شده دارای برچسب زمانی دقیق باشند.

۳. قابلیت تشخیص (Detection Capabilities)

تکنولوژی‌های IDPS به‌طورمعمول طیف وسیعی از قابلیت‌های تشخیص را ارائه می‌دهند. بسیاری از IDPSها از این قابلیت‌ها به‌صورت ترکیبی استفاده می‌نمایند تا امکان تشخیص دقیق‌تر، انعطاف‌پذیری بیشتر و سفارشی‌سازی را فراهم آورند.

نوع حوادث شناسایی‌شده و دقت تشخیص، تا حد زیادی به نوع تکنولوژی IDPS بستگی دارد. اغلب IDPSها برای بهبود دقت، صحت تشخیص، قابلیت استفاده و اثربخشی خود، نیاز به انجام یک حداقل پیکربندی و سفارشی‌سازی دارند. هرقدر تنظیم و سفارشی‌سازی قابلیت‌های یک محصول بهتر انجام شود، قدرت تشخیص و کارایی آن نیز افزایش پیدا می‌کند.

برخی از این قابلیت‌ها به شرح زیر است:

i. آستانه تحمل (Threshold)

آستانه تحمل مقداری است، بین رفتار طبیعی و رفتار غیرطبیعی. آستانه تحمل، معمولاً حداکثر سطح قابل‌قبول را مشخص می‌کند. مثلاً X کاراکتر برای طول نام یک فایل، یا X بار شکست اتصال در ۶۰ ثانیه.

آستانه تحمل، اغلب در تشخیص مبتنی بر رفتار خلاف قاعده و تحلیل وضعیت پروتکل، مورداستفاده قرار می‌گیرد.

ii. لیست سیاه و لیست سفید (Blacklists and Whitelists)

[1] Capture
[2] Network Time Protocol (NTP)

یک لیست سیاه، لیستی از نهادهای مجزا مثل میزبان، شماره پورت TCP یا UDP، نوع و کد ICMP، برنامه‌ها، نام‌های کاربری، URLها، نام فایل‌ها یا پسوند فایل‌ها، است که قبلاً مشخص گردیده آن‌ها با فعالیت مخربی همراه هستند. از لیست سیاه برای مسدود کردن فعالیت‌هایی که بسیار شبیه فعالیت‌های مخرب هستند، استفاده می‌شود.

یک لیست سفید، لیستی از نهادهای مجزا است که بی‌خطر شناخته‌شده‌اند. لیست سفید معمولاً به‌صورت جزءبه‌جزء[1]، مثل پروتکل به پروتکل مورداستفاده قرار می‌گیرد تا امکان بروز تشخیص مثبت کاذب را در رابطه با فعالیت‌های بی‌خطر میزبان‌های مورد اعتماد، کاهش دهد.

لیست‌های سیاه و سفید، در تشخیص مبتنی بر امضا و تحلیل وضعیت پروتکل، مورداستفاده قرار می‌گیرند.

iii. تنظیمات هشدار (Alert Settings)

معمولاً IDPSها امکان سفارشی‌سازی هشدارها را برای مدیران فراهم می‌نمایند. نمونه‌هایی از اقداماتی که می‌توان روی هشدارها انجام داد، به شرح زیر است:

- خاموش یا روشن کردن آن.
- تنظیم سطح دقت و اولویت.
- تعیین اینکه چه اطلاعاتی باید ثبت شود، و چه روش اطلاع‌رسانی (به‌عنوان‌مثال ایمیل، پیجر) باید استفاده شود.
- مشخص کردن قابلیت‌های پیشگیری که باید مورداستفاده قرار گیرد.

iv. مشاهده و ویرایش کد (Code Viewing and Editing)

برخی از IDPSها به مدیر شبکه اجازه دیدن تمام یا بخشی از کدهای مربوط به تشخیص را می‌دهند. این اجازه معمولاً محدود به امضا است؛ ولی بعضی از IDPSها به مدیر شبکه اجازه دیدن کدهای اضافی دیگر، ازجمله برنامه‌های مورداستفاده برای تحلیل وضعیت پروتکل، را نیز می‌دهند.

مشاهده کد می‌تواند به تحلیل‌گران برای تعیین دلیل تولید یک هشدار خاص، بررسی اعتبار هشدار و شناسایی مثبت کاذب، کمک قابل‌توجهی نماید.

جهت امکان ویرایش تمام کدهای مربوط به تشخیص و نوشتن کد جدید (مثل یک امضای جدید) لازم است انواع خاصی از قابلیت‌های تشخیص را

[1] Granular

به‌صورت کامل، سفارشی‌سازی نمایید. به‌عنوان‌مثال، یک هشدار خاص ممکن است بر اساس یک سری رویدادهای پیچیده مربوط به کدهای مختلف، تولیدشده باشد؛ لذا سفارشی کردن IDPS، بدون ویرایش مستقیم کد امکان‌پذیر نخواهد بود. ویرایش کد نیز به مهارت‌های برنامه‌نویسی و تشخیص نفوذ نیاز دارد. البته برخی از IDPSها از زبان‌های برنامه‌نویسی اختصاصی استفاده می‌نمایند.

باید توجه داشته باشید که اشکالات فرآیند سفارشی‌سازی، می‌تواند منجر به شکست کامل یا عملکرد نادرست IDPS گردد. بنابراین مدیران باید سفارشی‌سازی کدها را به‌دقت انجام دهند تا باعث بروز اختلال در سایر کدها نگردد.

۴. **قابلیت پیشگیری (Prevention Capabilities)**

IDPSها قابلیت‌های پیشگیری متعددی را ارائه می‌نمایند؛ که ممکن است بر اساس نوع IDPS، با یکدیگر متفاوت باشند. IDPSها معمولاً به مدیران شبکه اجازه پیکربندی قابلیت پیشگیری را برای هر نوع هشدار می‌دهند. این کار شامل فعال یا غیرفعال کردن پیشگیری، و همچنین مشخص کردن نوع قابلیت پیشگیری مورداستفاده، است.

برخی سنسورهای IDPS دارای حالت یادگیری[1] یا شبیه‌سازی[2] هستند؛ که در این حالت عمل پیشگیری توسط آن‌ها انجام‌نشده و به‌جای آن، صرفاً اقدام پیشگیری که باید انجام می‌گرفته را نمایش می‌دهند. این کار به مدیران شبکه اجازه می‌دهد قبل از فعال‌سازی اقدامات پیشگیرانه، نظارت و پیکربندی دقیق قابلیت‌های پیشگیری را انجام دهند، تا امکان مسدود شدن سهوی فعالیت‌های بی‌خطر، کاهش یابد.

معماری شبکه و محل قرارگیری سنسور

نحوه قرارگیری تجهیزات امنیتی در شبکه، یکی از مهم‌ترین مؤلفه‌های کارکرد صحیح آن‌ها است. تجهیزات IDPS نیز دارای سنسورهایی برای انجام عملیات مربوط به تشخیص و پیشگیری می‌باشند. نحوه صحیح قرارگیری این سنسورها در شبکه بر کارایی IDPS تأثیرات مهمی دارد.

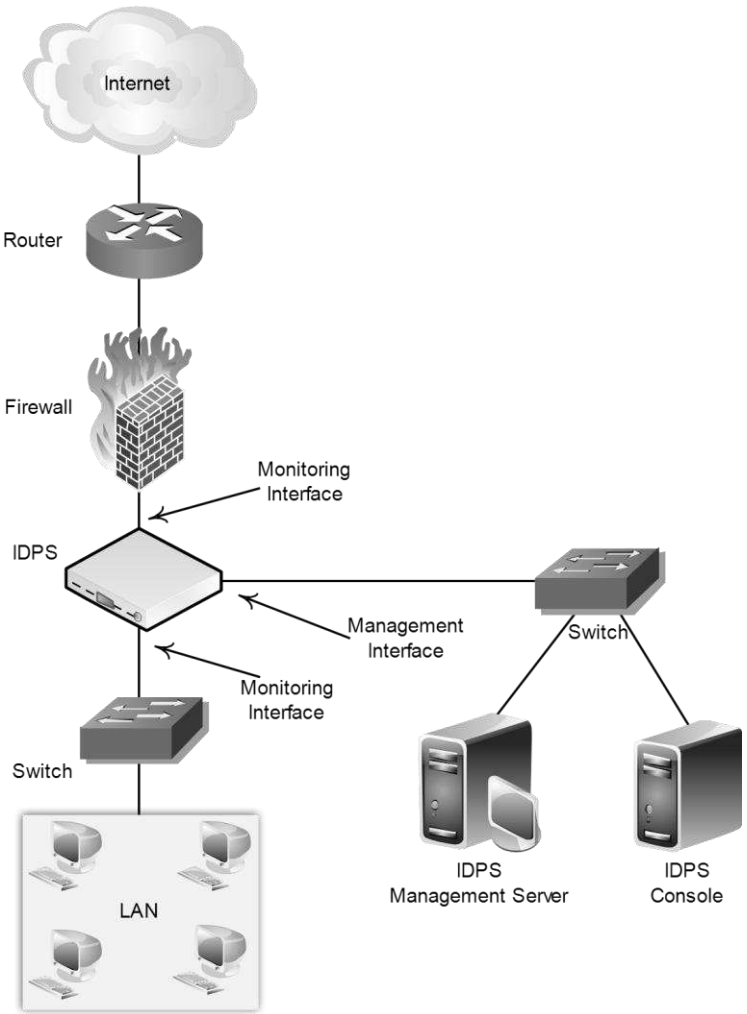
سنسورهای IDPS می‌توانند به یکی از دو حالت زیر در شبکه قرار گیرند:

[1] Learning
[2] Simulation

۱. **برخط (Inline)**

سنسورهای برخط، طوری در شبکه مستقر می‌شوند که ترافیک از آن‌ها عبور نماید. دلیل عبور ترافیک از این نوع سنسورها، مسدود کردن ترافیک خطرناک مربوط به فعالیت‌های مخرب است. اگر IDS و IPS را دو ماژول جدا از هم در نظر بگیریم؛ سنسورهای Inline مربوط به تجهیزات IPS می‌شوند.

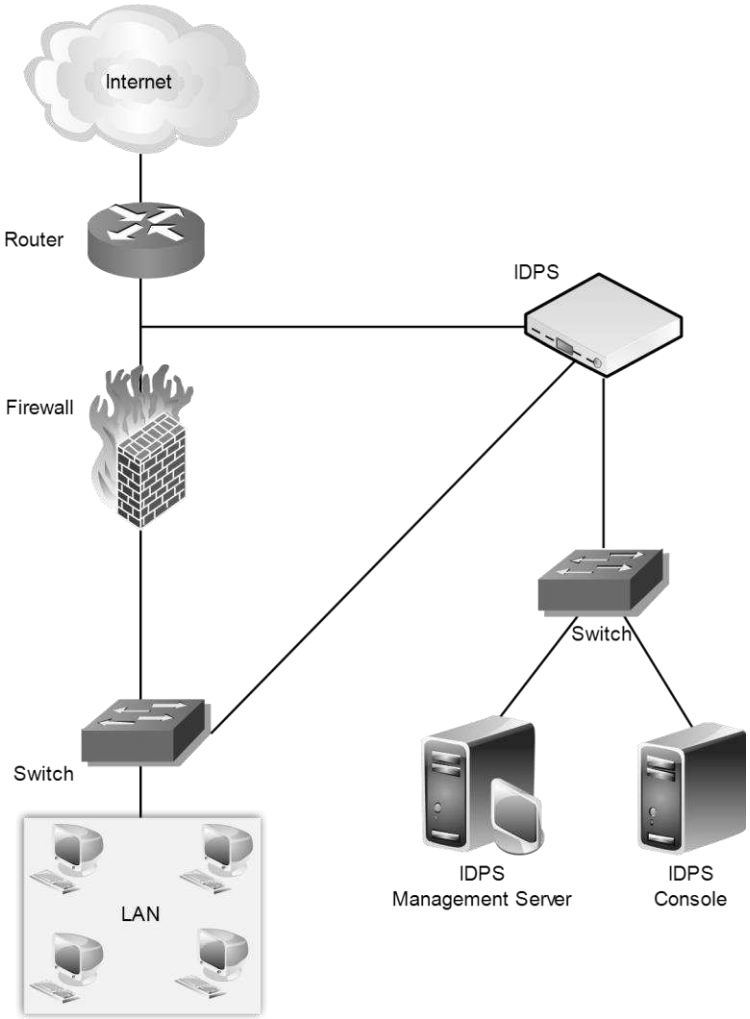
تصویر زیر نشان‌دهنده نحوه قرارگیری سنسورهای برخط (Inline) است.



۲. منفعل (Passive)

سنسور منفعل (Passive) به نحوی در شبکه قرار می‌گیرد که بر روی جریان ترافیک تأثیرگذار نباشد. وظیفه این نوع سنسور نظارت بر روی ترافیک یا ارسال یک کپی از آن برای IDPS است. سنسور منفعل در مکان‌های کلیدی شبکه که ترافیک مهمی از آن عبور می‌کند قرار داده می‌شود.

اگر IPS و IDS را دو ماژول جدا از هم در نظر بگیریم؛ سنسور Passive مربوط به تجهیزات IDS است.



☑ مبحث چهارم

هانی‌پات

هانی‌پات[1] یا ظرف عسل، یک کلاس قدرتمند از ابزار امنیتی است که عملیاتی فراتر از یک تشخیص نفوذ معمولی را انجام می‌دهد. هانی‌پات یک سیستم طعمه[2] است که برای فریب حملات بالقوه طراحی‌شده، تا آن‌ها را از سیستم‌های حیاتی دورنگه دارد. هنگامی‌که از چندین دستگاه هانی‌پات در یک شبکه استفاده می‌شود، آن را هانی‌نت می‌نامند.

سیستم هانی‌پات اقدام به شبیه‌سازی خدمات مهم شبکه نموده و خود را سیستمی آسیب‌پذیر نمایش می‌دهد. این کار برای فریب مهاجمین بوده تا حملات بالقوه را روی سیستم هانی‌پات به اجرا درآورند. پس‌ازآنکه مهاجمین حملات موردنظر خود را روی سیستم هانی‌پات انجام دادند، بدون اینکه به دارایی‌های اطلاعاتی سازمان آسیبی رسیده باشد، سازمان از نوع حملات آگاه شده و می‌تواند دارایی‌های واقعی خود را در برابر آن‌ها مقاوم نماید.



هانی‌پات یک سیستم امنیت اطلاعات است، که ارزش آن در دسترسی غیرمجاز و مورد سوءاستفاده قرار گرفتن آن است.

درمجموع هانی‌پات‌ها برای انجام موارد زیر طراحی‌شده‌اند:

- منحرف کردن مهاجم از سیستم‌های حیاتی.
- جمع‌آوری اطلاعات در مورد فعالیت‌های مهاجم.
- ترغیب مهاجم برای باقی ماندن طولانی‌مدت بر روی سیستم؛ تا مدیر امنیت وقت کافی برای مستندسازی رویدادها و در صورت لزوم پاسخ به حملات را داشته باشد.

به دلیل اینکه اطلاعات روی هانی‌پات‌ها باارزش به نظر می‌رسند، لذا هرگونه دسترسی غیرمجاز به آن‌ها، به‌منزله فعالیت‌های مشکوک است. هانی‌پات‌ها با نظارت حساس و ثبت رویدادها برای هر تلاش انجام‌گرفته برای دسترسی، اقدام به جمع‌آوری اطلاعات در مورد حملات بالقوه می‌نمایند.

---

[1] Honeypot
[2] Decoy

مزایا و معایب هانی‌پات

استفاده از هانی‌پات می‌تواند دارای یک سری مزایا و معایب برای امنیت اطلاعات گردد. در ادامه برخی از مزایا و معایب این سیستم ذکرشده است.

۱. مزایای استفاده از هانی‌پات

- مهاجمین به اهدافی منحرف می‌شوند که هیچ خطری برای دارایی‌های اطلاعاتی سازمان ندارد.
- مدیران امنیت اطلاعات وقت کافی برای تصمیم‌گیری نحوه برخورد با مهاجم را در اختیاردارند.
- اقدامات مهاجم می‌تواند به‌راحتی و به‌صورت گسترده مورد نظارت قرارگرفته و ثبت شود. وقایع ثبت‌شده نیز می‌تواند برای تصحیح مدل‌های تهدید و بهبود حفاظت سیستم، مورداستفاده قرار گیرد.
- هانی‌پات‌ها ممکن است در شناسایی کارمندان داخلی که قصد جاسوسی از شبکه رادارند نیز مؤثر باشند.
- هانی‌پات‌ها به نسبت تجهیزات امنیتی دیگر مثل IDPSها، هشدارهای کمتر ولی باارزش‌تری را برای مدیر شبکه فراهم می‌آورند.
- هانی‌پات‌ها بر روی حملاتی که بر اساس ارتباط رمزشده صورت می‌گیرد نیز مؤثر هستند؛ چراکه به‌عنوان نقطه نهایی ارتباط، قادر به رمزگشایی دیتای ارسال‌شده از طرف مهاجم، می‌باشند.
- احتمال بروز مثبت کاذب در هانی‌پات‌ها به نسبت سایر تکنولوژی‌های امنیتی، کمتر است.
- برای راه‌اندازی هانی‌پات به تجهیزات گران‌قیمت نیازی نیست.

۲. معایب استفاده از هانی‌پات

- مجازات قانونی سوءاستفاده از این تجهیزات به‌خوبی شناخته‌نشده است.
- هانی‌پات‌ها هنوز نتوانسته‌اند به‌صورت عمومی نشان دهند که تکنولوژی امنیتی مفیدی هستند.
- ممکن است درصورتی‌که یک مهاجم متخصص یک‌بار در این تورها گرفتار شود، به‌شدت عصبانی شده و یک حمله تهاجمی‌تر و مؤثرتر را برای سازمان شما، تدارک ببیند!
- برای استفاده ازاین‌گونه تجهیزات، مدیران شبکه و مدیران امنیت اطلاعات، باید در سطح بالایی از تجربه و تخصص قرار داشته باشند.

انواع هانی‌پات

هانی‌پات‌ها را می‌توان بر اساس میزان تعاملاتشان به دو گروه زیر تقسیم‌بندی نمود:

۱. **کم‌تعامل (Low-Interaction)**

هانی‌پات‌های کم تعامل معمولاً به تعامل با چند مهاجم یا بدافزار محدود هستند. همه خدمات ارائه‌شده توسط هانی‌پات کم تعامل به‌صورت شبیه‌سازی‌شده[1] است. به عبارتی سیستم‌ها و سرویس‌هایی که در اختیار مهاجم قرار می‌گیرند، به‌صورت شبیه‌سازی‌شده بوده و فعالیت‌های مهاجمان نیز محدود به همان سرویس‌ها خواهد بود.

پیاده‌سازی این نوع هانی‌پات‌ها ساده است؛ نرم‌افزارهایی که جهت راه‌اندازی این نوع هانی‌پات مورداستفاده قرار می‌گیرند، به‌راحتی سرویس‌های موردنظر را (البته محدود) شبیه‌سازی می‌نمایند.

هانی‌پات کم‌تعامل آسیب‌پذیر نبوده و معمولاً آلوده نمی‌شود، به همین دلیل شناسایی هانی‌پات کم‌تعامل برای هکر راحت‌تر است؛ چراکه این نوع هانی‌پات به دلیل شبیه‌سازی کردن سرویس‌ها، در مقابل حملات ناشناخته عکس‌العمل مناسبی را به هکر نشان نمی‌دهند و هکر می‌تواند به جعلی بودن سرور پی ببرد.

۲. **پرتعامل (High-Interaction)**

در هانی‌پات پرتعامل برخلاف کم‌تعامل، سیستم‌عامل‌ها و برنامه‌های کاربردی به‌صورت واقعی مورداستفاده قرار می‌گیرند. در این سیستم‌ها همه‌چیز واقعی بوده و هیچ سرویسی شبیه‌سازی نمی‌شود.

مزیت این روش در آن است که پس از وقوع حمله واقعی، کارشناس امنیت اطلاعات می‌تواند طیف وسیعی از اطلاعات مربوط به مهاجم و نوع حمله را به دست آورد. همچنین به علت واقعی بودن سیستم‌ها و سرورها، در برابر هر عملی که مهاجم انجام می‌دهد یک عکس‌العمل معمولی و مناسب مشاهده می‌کند؛ لذا شناسایی جعلی بودن این نوع هانی‌پات برای مهاجم تقریباً غیرممکن است.

به دلیل واقعی بودن سیستم‌عامل و سرویس‌ها، پیاده‌سازی هانی‌پات پرتعامل پیچیده‌تر و سخت‌تر است. یک نگرانی بزرگ در رابطه با این نوع هانی‌پات این است که به دلیل وجود آسیب‌پذیری‌های عمدی بر روی سیستم، امکان آن وجود دارد که مهاجم بتواند کنترل هانی‌پات را در دست گرفته و به علت واقعی بودن سیستم‌عامل آن، از همان طریق بتواند به سایر سیستم‌های شبکه حمله کند.

---

[1] Emulation

فصل نهم

ارتباطات امن شبکه

مبحث اول

IPsec

مبحث دوم

شبکه خصوصی مجازی

☑ مبحث اول

IPsec

برای برقراری امنیت، پروتکل‌ها و سرویس‌های مختلفی وجود دارد. برخی از این پروتکل‌ها تاکنون بررسی‌شده و به برخی دیگر نیز در فصول آینده پرداخته می‌شود. اما هر یک از این پروتکل‌ها، امنیت یک لایه یا سرویس را فراهم آورده و در یک لایه از مدل OSI کار می‌کنند. هرچه امنیت را در لایه‌های پایین‌تر OSI برقرار نماییم، بالطبع لایه‌های بالاتر نیز از امنیت بهره‌مند خواهند شد. حتی می‌توانند علاوه بر امنیت تأمین‌شده در لایه‌های پایین‌تر، در لایه مربوط به خود نیز از پروتکل‌ها و سرویس‌های امن استفاده نمایند.

یکی از مهم‌ترین پروتکل‌های امنیتی، IP Security است. این پروتکل که به‌اختصار IPsec نامیده می‌شود در لایه شبکه (لایه سوم) مدل OSI اقدام به برقراری امنیت می‌نماید. در این صورت لایه‌های بالاتر نیز از امنیت نسبی برخوردار خواهند شد.

اجرای پروتکل IPsec باعث فراهم شدن امنیت در سطح شبکه می‌شود. وقتی یک شبکه امن باشد، تبادل دیتا نیز بر روی بستر امن صورت گرفته و تا حد قابل قبولی از امنیت برخوردار خواهد بود. با اجرای IPsec، سازمان می‌تواند مطمئن باشد که نه‌تنها برای برنامه‌های کاربردی دارای مکانیسم امنیتی، بلکه برای برنامه‌های کاربردی فاقد هرگونه مکانیسم امنیتی نیز سطح خوبی از امنیت را فراهم آورده است.

امنیت در سطح IP شامل سه سطح کاربردی: احرازهویت، محرمانگی و مدیریت کلید است. مکانیسم احرازهویت، دریافت‌کننده را مطمئن می‌سازد که بسته دریافتی را فرستنده موردنظر ارسال کرده و در مسیر انتقال نیز دچار تغییر نشده است. محرمانگی نیز با استفاده از رمزنگاری باعث می‌شود در صورت استراق سمع بسته توسط شخص ثالث، از افشای اطلاعات جلوگیری شود. مکانیسم مدیریت کلید نیز تسهیل‌کننده تبادل امن کلیدهای مربوطه است.

سازمان IETF طی RFC 4301، اقدام به ارائه استاندارد معماری امنیت برای پروتکل اینترنت (IP) نموده است.[1] این استاندارد نحوه پیاده‌سازی امنیت برای پروتکل IP را تشریح کرده است. همان‌طور که در این RFC بیان‌شده است، امکان استفاده از IPsec در محیط هر دو پروتکل IPv4 و IPv6 وجود دارد. البته این سند ارائه‌دهنده یک معماری امنیتی کلی برای اینترنت نیست؛ بلکه امنیت ارائه‌شده در این سند، فقط برای فراهم کردن امنیت در لایه IP، از طریق ترکیب رمزنگاری

[1] https://www.ietf.org/rfc/rfc2401.txt

و مکانیسم‌های امنیتی دیگر است. استفاده از IPsec در پروتکل IPv4 به‌صورت اختیاری است؛ ولی IAB[1] استفاده از آن را در نسل بعدی IP، یعنی IPv6 به‌صورت اجباری اعلام کرده است.

کاربردهای IPsec

استفاده از پروتکل IPsec قابلیت برقراری امنیت در کانال‌های ارتباطی را فراهم می‌آورد. حفاظت از کانال‌های ارتباطی در سراسر شبکه LAN، سراسر شبکه‌های خصوصی و عمومی WAN، و سراسر اینترنت، را می‌توان با استفاده از IPsec برقرار نمود. نمونه‌هایی از استفاده IPsec به‌صورت زیر است:

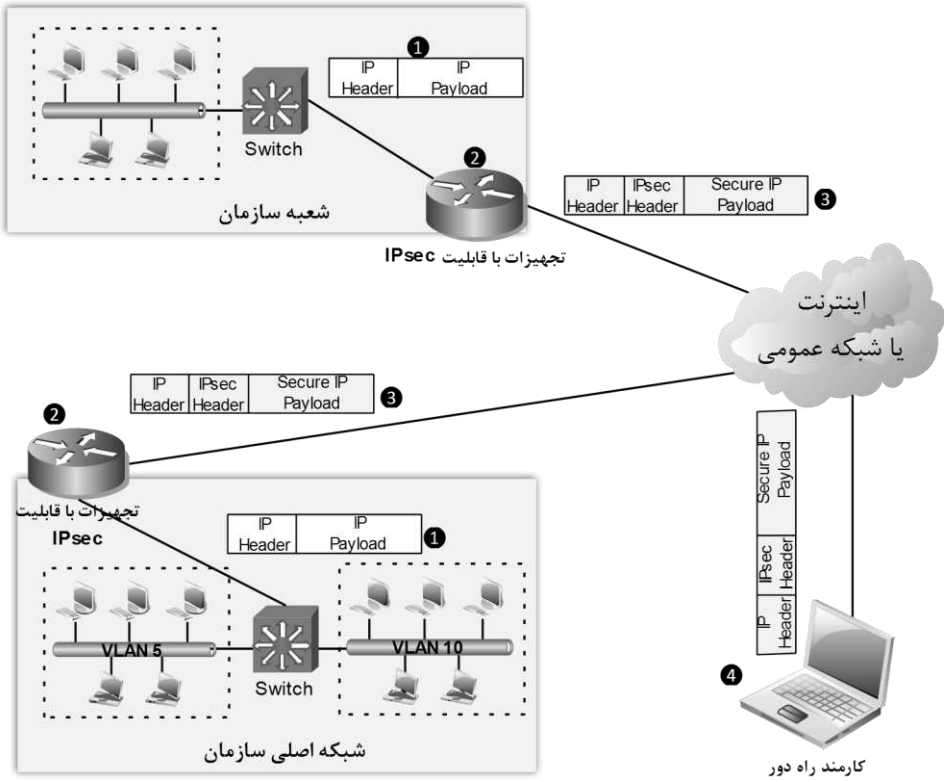
- برقراری اتصال امن بین شعبه و سازمان، بر روی اینترنت.
شرکت می‌تواند یک شبکه خصوصی مجازی امن، بر روی اینترنت یا شبکه عمومی WAN، ایجاد نماید. این کار شرکت را قادر می‌سازد برای تسهیل کسب‌وکار خود از ظرفیت عظیم اینترنت بهره برده و هزینه مربوط به برپایی شبکه‌های خصوصی و سربار مدیریتی را کاهش دهد.
- دسترسی از راه دور امن بر روی اینترنت.
کاربر نهایی با استفاده از IPsec می‌تواند از طریق اینترنت یک ارتباط امن را با شبکه سازمان خود برقرار نماید. این کار باعث صرفه‌جویی در هزینه جابجایی کارمندان شده و باعث تسهیل دورکاری نیز می‌گردد.
- برقراری شبکه‌های داخلی[2] و خارجی[3] با شرکا.
جهت برقراری یک ارتباط امن با سایر سازمان‌ها و شرکا، می‌توان از IPsec بهره برد. با استفاده از IPsec هر دو طرف می‌توانند از احرازهویت، محرمانگی و مکانیسم تبادل کلید مطمئن باشند.
- افزایش امنیت تجارت الکترونیک[4].
اگرچه برخی برنامه‌های کاربردی وب و تجارت الکترونیک به‌صورت توکار[5] از پروتکل‌های امنیتی استفاده می‌نمایند؛ اما به‌کارگیری IPsec تضمین می‌کند که تمام ترافیک تبادل شده، تحت رمزنگاری و تصدیق هویت بوده و یک‌لایه امنیتی به هر آنچه توسط لایه کاربرد ارائه‌شده، اضافه گردیده است.

---

[1] Internet Architecture Board (IAB)
[2] Intranet
[3] Extranet
[4] Electronic Commerce
[5] Built-in

یکی از ویژگی‌های مهم IPsec این است که می‌تواند از برنامه‌های کاربردی متنوعی پشتیبانی نماید. به دلیل اینکه IPsec امنیت را در لایه شبکه برقرار می‌سازد، حتی اگر پروتکل‌های مورداستفاده در لایه کاربرد، فاقد مکانیسم‌های امنیتی باشند، بازهم ترافیک آن‌ها در سطح لایه شبکه دارای محرمانگی و احراز هویت خواهد بود. بنابراین IPsec می‌تواند تمام برنامه‌های کاربردی توزیع‌شده (ازجمله ورود به سیستم از راه دور[1]، سرویس‌گیرنده/سرویس‌دهنده، پست الکترونیک، انتقال فایل، دسترسی وب، و غیره) را امن نماید.

تصویر زیر نمونه‌ای از استفاده IPsec برای سازمانی که دارای کارمندان و شعب پراکنده است را نمایش می‌دهد.



همان‌طور که در تصویر فوق ملاحظه می‌نمایید، تبادل ترافیک در داخل شبکه سازمان و شعبه به‌صورت ناامن انجام می‌گیرد. ❶ به‌عبارت‌دیگر از همان IPv4 به‌صورت پیش‌فرض استفاده‌شده

[1] Remote logon

است. اما هر شبکه دارای تجهیزاتی (مانند روتر و فایروال) باقابلیت IPsec است❷؛ که ترافیک داخل شبکه را به‌صورت ناامن دریافت کرده ولی به‌صورت امن و بر اساس IPsec به بیرون از شبکه ارسال می‌نماید.❸ این تجهیزات همچنین وظیفه رمزگشایی اطلاعات دریافتی را نیز بر عهده‌دارند. کاربران راه دور نیز باید امکان برقراری ارتباط IPsec را بر روی کامپیوتر خود داشته باشند.❹

مزایای استفاده از IPsec

برخی از مزایای استفاده از IPsec به‌صورت زیر است:

- زمانی که IPsec از طریق یک روتر یا فایروال در شبکه پیاده‌سازی می‌شود، یک امنیت قوی برای تمام ترافیک تبادل شده با محیط بیرونی، را فراهم می‌آورد. در این صورت عملیات رمزگذاری فقط بر روی دیتایی که قصد خروج از شبکه را دارد، اعمال‌شده؛ و دیتای درون شبکه به‌صورت عادی تبادل می‌گردند. لذا باعث ایجاد سربار اضافی برای منابع شبکه نخواهد شد.
- اگر تمام ترافیک خارجی فقط از طریق فایروال امکان ورود به شبکه را داشته باشند، IPsec راه‌اندازی شده بر روی فایروال، در مقابل دور زدن[1] مقاوم خواهد بود.
- با توجه به اینکه IPsec در زیر لایه انتقال (TCP/UDP) است، لذا از دید لایه کاربرد شفاف[2] خواهد بود[3]. بنابراین درصورتی‌که IPsec بر روی فایروال یا روتر راه‌اندازی شده باشد، نیازی به تغییر نرم‌افزار بر روی سرور یا سیستم کاربر نیست. حتی اگر IPsec در سیستم کاربر نهایی راه‌اندازی شده باشد، بازهم نرم‌افزارهای لایه بالاتر را تحت تأثیر قرار نخواهد داد.
- IPsec از دید کاربر نهایی نیز می‌تواند شفاف باشد. بدون نیاز به آموزش کاربران یا دخالت دادن آن‌ها در اجرای مکانیسم‌های امنیتی، می‌توان IPsec را راه‌اندازی نمود.
- از IPsec می‌توان در برپایی شبکه خصوصی مجازی (VPN) نیز استفاده کرد.
- IPsec را می‌توان برای کاربران به‌صورت جداگانه راه‌اندازی نمود. درصورتی‌که قرار است یک کاربر، از راه دور با سازمان در ارتباط باشد و یا حتی در داخل سازمان با اطلاعات حساسی کار می‌کند، می‌توان از IPsec برای آن کاربر خاص بهره برد.

---

[1] bypass
[2] Transparent
[3] منظور از شفافیت، این است که برنامه موجود در لایه کاربرد از اتفاقی که در لایه شبکه می‌گذرد بی‌اطلاع خواهد بود؛ و بدون اینکه به یکدیگر وابسته باشند، هرکدام وظیفه خود را انجام می‌دهند.

پروتکل‌های مورداستفاده در IPsec

IPsec برای فراهم آوردن امنیت از دو پروتکل AH و ESP استفاده می‌نماید. در پیاده‌سازی IPsec **باید** از ESP استفاده شود و **شاید** از AH نیز استفاده گردد. دلیل کاهش ارجحیت استفاده از AH، از "باید" به "شاید"، این است که تجربه نشان داده در موارد کمی ممکن است ESP نتواند به‌تنهایی سرویس‌های امنیتی موردنظر را فراهم کند.

طبق RFC 4301، هریک از پروتکل‌های AH و ESP، ارائه‌دهنده سرویس‌های امنیتی خاصی به‌صورت زیر هستند:

۱. **Authentication Header (AH)**
پروتکل AH، ارائه‌دهنده سرویس صحت[۱] و احراز هویت مبدأ[۲] است؛ همچنین می‌تواند ویژگی ضد بازپخش[۳] را نیز به‌صورت اختیاری[۴] فراهم آورد.

۲. **Encapsulating Security Payload (ESP)**
پروتکلESP، ارائه‌دهنده تمام سرویس‌های فوق بوده و علاوه بر آن، می‌تواند سرویس محرمانگی[۵] را نیز فراهم آورد. استفاده از ESP برای فراهم آوردن محرمانگی بدون ارائه سرویس صحت، پیشنهاد نمی‌گردد. اگر زمان استفاده از ESP، محرمانگی در آن فعال باشد، مقرراتی برای محدودیت جریان ترافیک محرمانه وجود خواهد داشت. به‌عنوان‌مثال مقرراتی برای پنهان کردن طول بسته و همچنین برای تسهیل تولید و دور انداختن بسته‌های ساختگی.

۳. هر دو پروتکل AH وESP، ارائه‌دهنده سرویس کنترل دسترسی (Access Control) نیز می‌باشند. کنترل دسترسی از طریق توزیع کلیدهای رمزنگاری و مدیریت جریان ترافیک منطبق بر دیتابیس سیاست امنیتی[۶]، فراهم می‌گردد.

این پروتکل‌ها ممکن است به‌صورت جداگانه یا ترکیبی، برای فراهم آوردن سرویس‌های امنیتی در IPv4 یا IPv6 مورداستفاده قرار گیرند.

حالت‌های استفاده از پروتکل‌های AH و ESP

هر دو پروتکل AH و ESP می‌توانند در دو حالت زیر مورداستفاده قرار گیرند:

---

[1] Integrity
[2] Data Origin Authentication
[3] Anti-Replay
[4] Optional
[5] Confidentiality
[6] Security Policy Database (SPD)

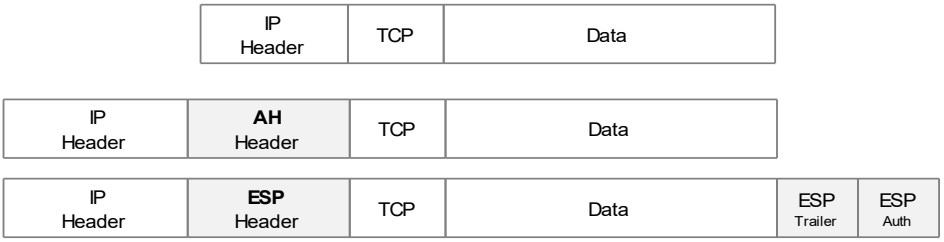
۱. Transport Mode

در حالت Transport، در درجه اول امنیت برای پروتکل‌های لایه‌های بالاتر فراهم می‌شود؛ چراکه در این حالت، حفاظت فقط از محموله[۱] بسته IP انجام می‌پذیرد؛ و نه از کل بسته IP. به‌طورمعمول از حالت Transport برای ارتباطات End-to-end بین دو میزبان استفاده می‌شود. به‌عنوان‌مثال، بین سیستم یک کاربر و سرور.

زمانی که یک میزبان اقدام به اجرای AH یا ESP بر روی IPv4 می‌نماید، محموله به‌طورمعمول به دنبال سرآیند بسته قرار می‌گیرد. برای IPv6، محموله به‌صورت دیتایی به دنبال هر دو سرآیند بسته (سرآیند IP و هر سرآیند توسعه داده‌شده IPv6) قرار می‌گیرد.

پروتکلESP در حالت Transport، اقدام به رمزنگاری (اجباری) و احرازهویت (اختیاری) محموله IP نموده، ولی بر روی سرآیند هیچ عملی انجام نمی‌دهد.

پروتکل AH در حالت Transport، اقدام به احرازهویت محموله به همراه بخش‌های انتخاب‌شده از سرآیند بسته IP می‌نماید.



۲. Tunnel Mode

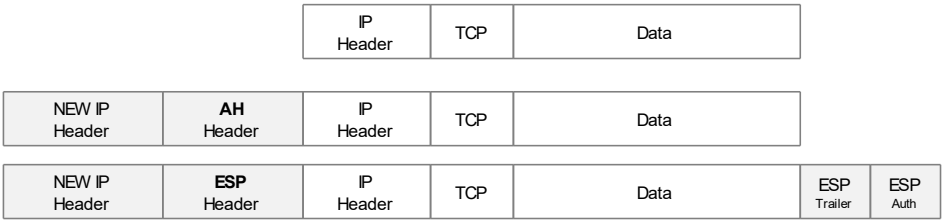
در حالت Tunnel، امنیت برای تمام بسته IP فراهم می‌گردد. برای این منظور پس‌ازآنکه فیلدهای AH یا ESP به بسته IP اضافه گردید، کل بسته به‌علاوه فیلدهای امنیتی به‌عنوان محموله بسته جدیدی که دارای سرآیند جدید است، قرار می‌گیرند. بسته اصلی درون بسته جدید قرارگرفته و از طریق Tunnel، از یک نقطه شبکه به نقطه دیگر می‌رود؛ بدون آنکه هیچ روتری در مسیر توانایی بررسی سرآیند داخلی بسته را داشته باشد. چون بسته اصلی درون بسته جدید محصورشده[۲]، معمولاً برای افزایش امنیت، بسته جدید دارای آدرس IP مبدأ و مقصد متفاوتی خواهد بود.

---

[۱] بهترین معادلی که توانستم برای کلمه Payload پیدا کنم، کلمه محموله است. منظور از Payload یا محموله در IP، دیتای موردنظر است که در بسته (Packet) قرارگرفته است.

[2] Encapsulate

این حالت که معمولاً توسط تجهیزاتی مثل فایروال یا روتر در Gateway شبکه انجام می‌شود، امکان استفاده چندین کاربر را از این Tunnel فراهم می‌آورد، بدون آنکه نیاز باشد هر کاربر به‌صورت مستقل اقدام به پیکربندی IPsec بر روی سیستم خود نماید.
پروتکل ESP در حالت Tunnel، اقدام به رمزنگاری(اجباری) و احرازهویت(اختیاری) تمام بسته IP داخلی به همراه سرآیند آن می‌نماید.[۱]
پروتکل AH در حالت Tunnel، اقدام به احرازهویت تمام بسته داخلی به همراه بخش‌های انتخاب‌شده سرآیند بسته خارجی، می‌نماید.



سیاست امنیت IP (IP Security Policy)

پایه و اساس اجرای IPsec، در مفهوم سیاست امنیتی اعمال‌شده به هر بسته‌ی در حال انتقال از یک مبدأ به یک مقصد است. سیاست IPsec، بر اساس تعامل با دو دیتابیس تعیین می‌شود. این دیتابیس‌ها عبارتند از:

۱. Security Association Database (SAD)
۲. Security Policy Database (SPD)

در ادامه، ابتدا به توضیح SA پرداخته و سپس دیتابیس‌های فوق را تشریح می‌کنیم.

Security Association (SA)

یک مفهوم کلیدی، که در هر دو مکانیسم احرازهویت و محرمانگی برای IP وجود دارد، Security Association است. Association یک ارتباط منطقی یک‌طرفه، بین فرستنده و گیرنده است که سرویس‌های امنیتی را جهت انتقال ترافیک ارائه می‌دهد. به‌عبارت‌دیگر، SA مشخص‌کننده سرویس‌های امنیتی مورداستفاده، جهت برقراری یک ارتباط امن است.

---

[۱] ESP رمزنگاری را همواره به‌صورت پیش‌فرض و اجباری انجام می‌دهد. اما احراز هویت در ESP، به‌صورت بوده و توسط کاربر می‌تواند فعال یا غیرفعال گردد.

اگر ارتباط به تبادل امن دوطرفه نیاز داشته باشد، باید از دو SA استفاده نمود. سرویس‌های امنیتی ارائه‌شده توسط SA، برای استفاده در AH یا ESP است و نمی‌توان از آن سرویس‌ها برای هر دو استفاده نمود.

یک SA منحصراً توسط سه پارامتر زیر شناسایی می‌شود:

۱. **Security Parameters Index (SPI)**
SPI یک رشته بیت اختصاص داده‌شده به همین SA بوده و فقط دارای مفهوم محلی است. SPI در سرآیند AH یا ESP قرار می‌گیرد، تا سیستم دریافت‌کننده بتواند تشخیص دهد کدام بسته حاوی اطلاعات مربوط SA است.

۲. **IP Destination Address**
این آدرس، مشخص‌کننده آدرس مقصد SA است؛ که می‌تواند یک سیستم کاربر نهایی یا تجهیزات شبکه مثل فایروال یا روتر باشد.

۳. **Security Protocol Identifier**
شناسه پروتکل امنیتی، یک فیلد موجود در سرآیند بسته خارجی است که نشان می‌دهد Association در قالب AH است یا در قالب ESP.

بنابراین در هر بسته IP، شناسه SA به‌صورت منحصربه‌فرد خواهد بود؛ این شناسه بر اساس آدرس مقصد موجود در سرآیند IPv4 یا IPv6، و مقدار SPI که در سرآیند ESP یا AH قرارگرفته، ایجاد می‌گردد.

Security Association Database (SAD)

در هر اجرای IPsec، یک دیتابیس بانام Security Association Database (SAD) وجود دارد که مشخص‌کننده پارامترهای مربوط به هر SA است.

هر SA، به‌صورت معمول توسط پارامترهای زیر، که در دیتابیس SAD ثبت‌شده، تعریف می‌گردد.

۱. **Security Parameter Index**
یک مقدار ۳۲ بیتی است که توسط گیرنده از مقدار منحصربه‌فرد شناسه SA، انتخاب‌شده است.

۲. **Sequence Number Counter**
یک مقدار۳۲ بیتی است که از آن برای تولید فیلد شماره توالی در سرآیند AH یا ESP استفاده می‌شود. (برای تمام پیاده‌سازی‌ها موردنیاز است.)

۳. **Sequence Counter Overflow**

این علامت[1] نشان‌دهنده آن است که سرریز[2] شمارنده توالی اتفاق افتاده؛ لذا باید یک رویداد قابل بازرسی تولید شود تا از انتقال بسته‌های بیشتر در این SA جلوگیری به‌عمل آید. (برای تمام پیاده‌سازی‌ها موردنیاز است.)

۴. **Anti-Replay Window**
یک شمارنده ۳۲ بیتی است که برای مشخص کردن تکراری بودن بسته‌های ورودی AH یا ESP، مورداستفاده قرار می‌گیرد. (برای تمام پیاده‌سازی‌ها موردنیاز است.)

۵. **AH Information**
اطلاعاتی است شامل: الگوریتم احرازهویت، کلیدها، طول عمر کلیدها و پارامترهای مربوط به AH استفاده‌شده. (برای پیاده‌سازی AH موردنیاز است.)

۶. **ESP Information**
اطلاعاتی است شامل: الگوریتم احرازهویت و رمزنگاری، کلیدها، مقدارهای اولیه، طول عمر کلیدها و پارامترهای مربوط به ESP استفاده‌شده. (برای پیاده‌سازی ESP موردنیاز است.)

۷. **Lifetime of this Security Association**
فاصله زمانی که باید پس‌ازآن، SA با یک SA جدید (و SPI جدید) جایگزین شده، یا خاتمه یابد؛ به‌علاوه نشانه‌ای که مشخص کند کدام اقدام فوق (خاتمه یا جایگزینی) باید رخ دهد. (برای تمام پیاده‌سازی‌ها موردنیاز است.)

۸. **IPsec Protocol Mode**
مشخص‌کننده Tunnel، Transport یا Wildcard بودن حالت پروتکل IPsec، است.

۹. **Path MTU**
مشخص‌کننده حداکثر اندازه‌ای است که یک بسته می‌تواند بدون تکه‌تکه شدن[3]، انتقال یابد؛ و همچنین شامل متغیرهای سالخوردگی[4] نیز می‌باشد. (برای تمام پیاده‌سازی‌ها موردنیاز است.)

مکانیسم مدیریت کلید که برای توزیع کلید مورداستفاده قرار می‌گیرد، با مکانیسم حریم خصوصی[5] و احرازهویت، باید تنها از طریق SPI به یکدیگر مرتبط شوند؛ در غیراینصورت احرازهویت و حریم خصوصی، کاملاً مستقل از هر مکانیسم مدیریت کلید می‌باشد.

---

[1] Flag
[2] overflow
[3] Fragmentation
[4] aging
[5] Privacy

Security Policy Database (SPD)

SPD مشخص می‌کند که این ترافیک IP مربوط به کدام SA خاص است. در ساده‌ترین شکل، یک SPD حاوی رکوردهایی است که هرکدام آن‌ها مشخص‌کننده زیرمجموعه ترافیک IP و SAهای مربوط به آنان است. در محیط‌های پیچیده‌تر ممکن است چندین رکورد به‌صورت بالقوه به یک SA، یا چندین SA به یک رکورد SPD، مربوط باشند.

هر رکورد SPD که توسط مجموعه‌ای از IP و مقادیر فیلد پروتکل لایه بالاتر، مشخص گردیده را Selector (انتخاب‌کننده) می‌نامند. درواقع این انتخاب‌کننده‌ها برای فیلتر ترافیک خروجی منطبق با یک SA خاص، استفاده می‌شوند.

در پردازش خروجی برای هر بسته IP، موارد زیر به ترتیب اجرا می‌شوند:

۱. مقایسه مقادیر فیلدهای درون بسته (فیلدهای انتخاب‌کننده) با دیتابیس SPD، جهت پیدا کردن تطابق با یک رکورد SPD، که به صفر SA یا بیشتر اشاره داشته باشد.
۲. مشخص کردن SA در صورت وجود، برای این بسته و SPI مربوط به آن.
۳. آیا نیاز به پردازش IPsec است؟ (مثلاً پردازش AH یا ESP).

انتخاب‌کننده (Selector)های زیر یک رکورد SPD را مشخص می‌کنند:

- **Remote IP Address**
آدرس IP راه دور، ممکن است یک آدرس IP یا یک رنج آدرس IP باشد.
- **Local IP Address**
آدرس IP محلی، ممکن است یک آدرس IP یا یک رنج آدرس IP باشد.
- **Next Layer Protocol**
سرآیند پروتکل IP، شامل فیلدی است که مشخص‌کننده پروتکل بعدی (بالاتر از لایه IP) است.
- **Name**
Name یک شناسه کاربری از سیستم‌عامل است. Name یک فیلد موجود در سرآیند IP یا لایه‌های بالاتر نیست؛ اما درصورتی‌که IPsec به‌عنوان کاربر بر روی همان سیستم‌عامل راه‌اندازی شده باشد، در دسترس خواهد بود.
- **Local and Remote Ports**
ممکن است به‌صورت پورت‌های جداگانه TCP یا UDP بوده و یا اینکه شامل یک لیستی از پورت‌ها باشد.

انتخاب نوع پردازش

SPD باید میان ترافیکی که IPsec از آن‌ها محافظت کند با سایر ترافیک‌ها، تمایز قائل شود. این امر که باید توسط فرستنده و گیرنده بر روی ترافیک ورودی یا ترافیک خروجی انجام پذیرد؛ دارای سه گزینه پردازشی زیر است:

۱. **Discard**
در این حالت به ترافیک اجازه عبور از مرز IPsec داده نمی‌شود.

۲. **Bypass**
در این حالت بدون اعمال محافظت IPsec، به ترافیک اجازه عبور از مرز IPsec داده می‌شود.

۳. **Protect**
در این حالت باید برای ترافیک، حفاظت IPsec فراهم گردد. برای برقراری حفاظت، SPD باید پروتکل امنیتی، حالت آن، گزینه‌های سرویس امنیتی و الگوریتم‌های رمزنگاری مورداستفاده را مشخص نماید.

جدول زیر یک نمونه از دیتابیس SPD را بر روی تجهیزاتی مثل روتر یا فایروال نمایش می‌دهد:

Protocol	Local IP	Port	Remote IP	Port	Action	Comment
UDP	1.2.3.101	500	*	500	Bypass	ترافیک مربوط به IKE، بدون IPsec اجازه عبور دارد.
ICMP	1.2.3.101	*	*	*	Bypass	ترافیک مربوط به ICMP بدون IPsec اجازه عبور دارد.
*	1.2.3.101	*	1.2.3.0/24	*	Protect ESP/Transport mode	ترافیک داخلی باید رمزنگاری شود.
TCP	1.2.3.101	*	1.2.4.10	80	Protect ESP/Transport mode	ترافیک ارسالی TCP:80 برای سرور 1.2.4.10 باید رمزنگاری شود.
*	1.2.3.101	*	1.2.4.0/24	*	Discard	به‌جز ترافیک فوق از مابقی ترافیک به سمت شبکه 1.2.4.0 باید جلوگیری شود.

پروتکل Encapsulating Security Payload (ESP)

همان‌طور که قبلاً گفته شد، برای اجرایی کردن IPsec، می‌توان از دو پروتکل AH و ESP بهره برد. اگر به یاد داشته باشید، طبق مقایسه‌ای که انجام دادیم، پروتکل ESP علاوه بر تمام ویژگی‌های AH، دارای ویژگی محرمانگی نیز بود. لذا با توجه به قابلیت‌های بیشتر و کاربرد وسیع‌تر پروتکل ESP، در ادامه به شرح این پروتکل می‌پردازیم.

در نسخه‌های اولیه IPsec، پروتکل ESP تنها فراهم آورنده محرمانگی بود؛ و در صورت نیاز به صحت، باید از پروتکل AH استفاده می‌گردید. اما در نسخه دوم IPsec، پروتکل ESP انعطاف‌پذیرتر شد و توانست، سرویس صحت را نیز فراهم آورد. همچنین در ESP، امکان غیرفعال کردن رمزنگاری نیز وجود دارد.

پروتکل ESP فراهم آورنده محرمانگی، احرازهویت مبدأ داده، صحت غیر اتصال گرا، سرویس ضدتکرار و محرمانگی جریان داده، است. مجموعه سرویس‌های ارائه‌شده در ESP، به انتخاب گزینه‌ها در زمان برقراری SA و به محل اجرا در توپولوژی شبکه، بستگی دارد.

پروتکل ESP توانایی کار با انواع الگوریتم‌های رمزنگاری و احرازهویت را دارد. پروتکل ESP با IP protocol number 50، شناخته می‌شود. آخرین نسخه ESP ارائه‌شده، ESP v3 است؛ که البته فعلاً IETF آن را به‌صورت پیش‌نویس ارائه کرده است.[1] با توجه به پیش‌نویس بودن نسخه سوم، ویژگی‌های تشریح شده در این کتاب بر اساس نسخه دوم پروتکل ESP است.

حالت‌های ESP

در مورد حالت‌های اجرای ESP قبلاً توضیح کامل داده‌شده است. ESP می‌تواند در دو حالت زیر مورداستفاده قرار گیرد:

۱. Transport Mode
۲. Tunnel Mode

در حالت Transport، از سرآیند اصلی بسته استفاده‌شده و فقط عمل رمزنگاری و صحت، بر روی محموله (Payload) بسته انجام می‌پذیرد. اما در حالت Tunnel، بسته اصلی مجدداً بسته‌بندی (Encapsulate) می‌شود؛ به این معنا که پروتکلESP، یک سرآیند جدید برای بسته ایجاد می‌نماید. در این صورت، بسته قبلی که حالا به‌عنوان محموله بسته جدید قرارگرفته، به‌صورت کامل (سرآیند بسته قبلی به‌علاوه محموله) رمزنگاری می‌گردد.

[1] https://tools.ietf.org/html/draft-ietf-ipsec-esp-v3-10

فرآیند رمزنگاری در ESP

پروتکل ESP برای رمزگذاری بسته‌های IPsec، از رمزنگاری متقارن استفاده می‌نماید. اگر به یاد داشته باشید، برای رمزنگاری متقارن، هر دو طرف فرستنده و گیرنده باید از یک کلید مشابه برای رمزگذاری و رمزگشایی استفاده نمایند. زمانی که فرستنده اقدام به رمزگذاری دیتا می‌کند، بر اساس الگوریتم مورداستفاده باید آن را به قطعات کوچک‌تر تقسیم نماید؛ به‌عنوان‌مثال اگر از الگوریتم رمزنگاری AES، استفاده کرده باشد، دیتا به بلوک‌های ۱۲۸ بیتی تقسیم خواهند شد. در طرف دیگر، گیرنده نیز پس از دریافت بسته رمزگذاری شده، باید بر اساس همان الگوریتم و کلید مورداستفاده توسط فرستنده، اقدام به رمزگشایی بسته نماید. در رمزنگاری متقارن، عملیات رمزگشایی، معکوس عملیات رمزگذاری خواهد بود.

به‌عنوان‌مثال، پروتکل ESP می‌تواند از الگوریتم‌های رمزنگاری AES-CBC[1]، AES-CTR[2] و 3DES، استفاده نماید.

فرآیند حفاظت از صحت (تمامیت) در ESP

یکی از ویژگی‌هایی که هر دو پروتکل AH و ESP فراهم می‌کنند، ویژگی صحت (Integrity) است. اولین گام در محافظت از صحت، ساخت چکیده پیام[3] توسط الگوریتم درهم سازی کلیددار، که آن را بانام الگوریتم کد احرازهویت پیام (MAC) نیز می‌شناسیم، می‌باشد.

اگر بخش رمزنگاری را به‌دقت مطالعه کرده باشید، به یاد دارید که در بحث MAC، ابتدا یک مقدار مخفی به پیام اضافه کرده و از آن Hash می‌گرفتیم. سپس مقدار Hash تولیدشده را به پیام الصاق کرده و برای گیرنده ارسال می‌کردیم. گیرنده پس از دریافت پیام با اضافه کردن کلید مخفی به پیام، از آن Hash گرفته و نتیجه را با مقدار Hash دریافتی مقایسه می‌کرد. در صورت برابر بودن نتایج، گیرنده از صحت پیام اطمینان حاصل می‌کند. همچنین به دلیل استفاده از یک کلید مخفی، که بین فرستنده و گیرنده توافق شده، گیرنده از تغییر نکردن پیام در طول مسیر و همچنین اصالت فرستنده نیز مطمئن می‌شود.

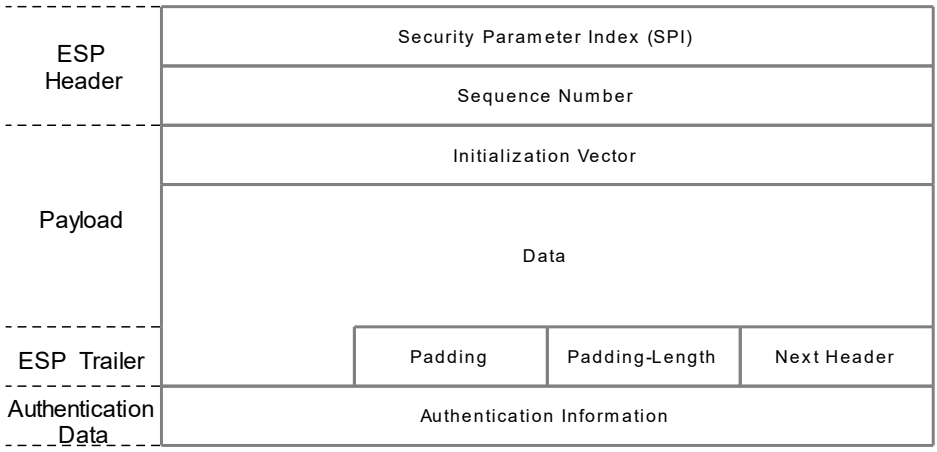
پروتکل IPsec برای تولید چکیده کلیددار، از الگوریتم‌های HMAC[4] استفاده می‌کند. الگوریتم‌های HMAC، برای استفاده از توابع درهم سازی در کنار یک کلید مخفی، کاربرد دارد. به‌عبارت‌دیگر، HMAC فراهم آورنده امکان استفاده از کلید مخفی در توابع درهم‌ساز است. درباره HMAC نیز در بخش دوم کتاب مطالبی ارائه گردیده است.

---

[1] AES-Cipher Block Chaining (AES-CBC)
[2] AES Counter Mode (AES-CTR)
[3] Hash (Message Digest)
[4] Hash Message Authentication Code (HMAC)

ساختار بسته ESP

تصویر زیر نمایش‌دهنده ساختار یک بسته ESP است. پروتکل ESP به هر بسته یک سرآیند و پشت‌بند[1] اضافه می‌نماید. بسته را می‌توان به چهار بخش ESP Header، Payload، ESP Trailer و Authentication Data، تقسیم کرد. در ادامه به تشریح فیلدهای ESP می‌پردازیم.



همان‌طور که در تصویر فوق ملاحظه می‌نمایید، سرآیند ESP از دو فیلد زیر تشکیل‌شده است:

- **SPI**

هر نقطه پایانی در هر اتصال IPsec، به‌دلخواه یک مقدار به‌عنوان SPI انتخاب می‌کند. این مقدار یک شناسه منحصربه‌فرد برای اتصال خواهد بود. دریافت‌کننده با استفاده از مقدار SPI، آدرس IP مقصد و نوع پروتکل IPsec (در اینجا ESP)، مشخص می‌کند که باید از کدام SA استفاده نماید.

- **Sequence Number**

به هر بسته یک شماره توالی پی‌درپی[2] تخصیص داده می‌شود و بسته‌ها تنها بر اساس داشتن دنباله قابل‌قبول، موردپذیرش قرار می‌گیرند. این عمل، مکانیسمی است که برای جلوگیری از تکرار بسته‌ها مورداستفاده قرار می‌گیرد. همچنین در خنثی‌سازی حملات منع خدمت (DoS)[3] نیز می‌توان از این خصوصیت بهره برد.

---

[1] Trailer
[2] Sequential sequence number
[3] Denial of Service

بخش بعدی بسته، Payload نام دارد. این بخش دارای دو فیلد Initialization Vector (IV) و Data است؛ که بر روی فیلد دیتا عملیات رمزگذاری انجام‌گرفته است، ولی مقدار IV بدون رمزگذاری می‌باشد.

از مقدار IV در عملیات رمزنگاری استفاده می‌شود. مقدار IV در هر بسته متفاوت است؛ لذا حتی اگر دو بسته دارای محتوای یکسان باشند، به دلیل IV متفاوت، خروجی رمزگذاری آن‌ها متفاوت خواهد بود. این ویژگی باعث می‌شود تا IPsec در برابر حملات تجزیه‌وتحلیل مقاوم‌تر باشد.

سومین بخش بسته، ESP Trailer است. این بخش از سه فیلد (دو فیلد ثابت و یک فیلد اختیاری) تشکیل‌شده است:

- **Padding**

یک بسته ESP، ممکن است به‌صورت اختیاری دارای فیلدی بانام Padding باشد. Padding یک سری بایت اضافه است که برای بزرگ کردن بسته، به آن اضافه‌شده و پس از دریافت بسته، توسط گیرنده حذف می‌شود. به دلیل اینکه ESP از رمزنگاری بلوکی استفاده می‌کند، لذا ممکن است برای تکمیل بلوک‌ها در اندازه موردنظر، مجبور به استفاده از Padding باشد. همچنین برای اینکه پشت‌بند (Trailer) باید مضربی از ۴ بایت باشد، می‌توان از Padding بهره برد. از جهت دیگر می‌توان با استفاده از Padding سایز دیتا را تغییر داده تا در برابر حملات تجزیه‌وتحلیل مقاوم‌تر باشند.

- **Padding Length**

این فیلد نشان‌دهنده مقدار بایت مورداستفاده برای Padding است. این فیلد به‌صورت اجباری در بسته وجود دارد.

- **Next Header**

در حالت Tunnel، Payload حاوی یک بسته کامل است که خود دارای سرآیند و محموله است. لذا فیلد Next Header معرفی کننده نوع داده موجود در Payload است. در حالت Transport، فیلد Next Header، معمولاً معرفی کننده پروتکل لایه انتقال است. این فیلد به‌صورت اجباری در بسته وجود دارد.

اگر خصوصیت محافظت از صحت، در ESP فعال باشد؛ یک فیلد بانام Authentication Information به انتهای بسته اضافه می‌گردد.[۱] شبیه AH، این فیلد شامل مقدار MAC است؛ اما برخلاف AH، در ESP مقدار هدر بسته خارجی در محاسبه MAC نقشی ندارد.

---

[۱] در بعضی از منابع برای این فیلد از نام Integrity check value (ICV)، نیز استفاده شده است.

پروتکل IKE

هدف پروتکل Internet Key Exchange (IKE)، مذاکره[1]، ایجاد و مدیریت SA است. همان‌طور که گفته شد، SA مشخص‌کننده ویژگی‌های IPsec و حفاظت‌های اعمال‌شده به یک اتصال است. البته SAها می‌توانند با استفاده از روش‌های از پیش تعریف‌شده، به‌صورت دستی ایجاد شوند؛ اما در این صورت امکان به‌روزرسانی نخواهند داشت. به همین دلیل روش دستی برای مقیاس بزرگ‌تر و واقعی، روش قابل قبولی نیست.

برای حل مشکل فوق در IPsec، از پروتکل IKE استفاده گردیده است. در IPsec از پروتکل IKE برای ارائه یک مکانیسم امن جهت برقراری ارتباطات محافظت‌شده، استفاده می‌شود. پروتکل IKE از پنج روش تبادل، برای ساخت SA، انتقال وضعیت و اطلاعات خطا، و تعریف گروه‌های جدید دیفی‌هلمن، استفاده می‌نماید. این پنج روش عبارتند از:

۱. Main Mode
۲. Aggressive Mode
۳. Quick Mode
۴. Informational
۵. Group

عملیات تبادل در پروتکل IKE در دو فاز صورت می‌گیرد. که در هر فاز نسبت به وظایفی که باید انجام شود از یک یا چند روش فوق استفاده می‌شود.

۱. **فاز اول IKE**

در فاز اول، یک کانال امن بانام IKE SA ایجاد می‌گردد. هدف از ایجاد این کانال، فراهم آوردن یک ارتباط امن دوطرفه برای تبادلات دیگر IKE است.
در این فاز بر چگونگی برقرار کردن یک ارتباط IPsec مذاکره می‌شود.

۲. **فاز دوم IKE**

هدف فاز دو، برقراری یک SA، برای اتصال واقعی IPsec است. این SA بانام IPsec SA شناخته می‌شود. برخلاف IKE SAها که دوطرفه هستند، IPsec SAها یک‌طرفه می‌باشند؛ به این معنی که یک اتصال بین دو سیستم، نیاز به دو SA دارد.
در این فاز یک کانال امن IPsec، جهت تبادل دیتا، شکل می‌گیرد.

در ادامه به تشریح پنج روش تبادل و نحوه تعامل آن‌ها با یکدیگر در جهت برقراری IPsec، در قالب دو فاز اجرایی IKE، می‌پردازیم.

[1] Negotiate

فاز اول IKE

هدف از فاز اول IKE، این است که یک کانال امن ایجاد نماید تا دو طرف ارتباط IPsec، امکان یک مذاکره موفق را برای انتخاب IPsec SA های موردنظر داشته باشند.

کانال امنی که در فاز اول ایجاد می‌گردد، بانام IKE SA شناخته می‌شود. هدف از ایجاد IKE SA، فراهم آوردن یک کانال ارتباطی امن دوطرفه برای سایر تبادلات IKE است که شامل: مذاکرات تشکیل فاز دو، انتقال وضعیت و اطلاعات خطا، و ایجاد یک گروه دیفی‌هلمن جدید، است.

در حقیقت فاز اول IKE باید با موفقیت انجام شود تا امکان ادامه تبادلات IKE فراهم گردد. برقراری یک IKE SA می‌تواند طی دو روش Main Mode و Aggressive Mode صورت پذیرد.

۱. روش Main Mode

درروش Main Mode، مذاکره برقراری IKE SA، از طریق سه جفت پیام انجام می‌گیرد.

در اولین جفت پیام، هر طرف پارامترهایی را برای برقراری SA پیشنهاد می‌دهند. چهار عدد از این پارامترها که تبادل آن‌ها ضروری است، را Protection Suite می‌نامند که شامل موارد زیر است:

i. **الگوریتم رمزنگاری**

مشخص‌کننده این است که از چه الگوریتمی جهت رمزنگاری دیتا باید استفاده شود. ازجمله این الگوریتم‌ها می‌توان از AES، DES، 3DES و Bluefish نام برد.

ii. **الگوریتم صحت (تمامیت)**

مشخص می‌کند از کدام الگوریتم HMAC باید جهت صحت استفاده شود. ازجمله این الگوریتم‌ها می‌توان HMAC-MD5 و HMAC-SHA1 را نام برد.

iii. **روش احرازهویت**

برای احرازهویت دو نقطه پایانی (End point) ممکن است چندین روش به‌صورت زیر وجود داشته باشد:

- Pre-shared Keys

باید از قبل یک کلید مخفی بین طرفین ارتباط توافق شده باشد. هر طرف اقدام به ایجاد یک مقدار محرمانه بر اساس کلید مخفی نموده و از آن برای برقراری کانال امن فاز اول و همچنین IPsec SA استفاده می‌نمایند. اتمام فاز اول IKE، به‌منزله آن است که هر دو طرف با داشتن کلید مخفی توانسته‌اند یکدیگر را احرازهویت نمایند.

- Digital Signatures

هر نقطه پایانی دارای یک Digital Certificate است که شامل کلیدعمومی می‌باشد. هر طرف ارتباط قبل از ارسال دیتا، آن را با کلیدخصوصی خود امضا کرده و

برای طرف مقابل ارسال می‌کند. گیرنده نیز با استفاده از کلیدعمومی فرستنده قادر به رمزگشایی آن خواهد بود. به‌عنوان‌مثال برای الگوریتم‌های امضای دیجیتال می‌توان از RSA و استاندارد DSS نام برد.

- External Authentication

در برخی از پیاده‌سازی‌های IPsec می‌توان از یک سرور خارجی مثل کربروس برای انجام عملیات احرازهویت بهره برد.

iv. گروه دیفی‌هلمن (DH)

اگر به یاد داشته باشید الگوریتم دیفی‌هلمن برای تبادل امن کلید، مورداستفاده قرار می‌گرفت. در برقراری IPsec نیز می‌توان از دیفی‌هلمن برای تولید یک کلید مشترک مخفیانه بین فرستنده و گیرنده، بهره برد.[۱] بر اساس این کلید یک مقدار تولید می‌شود که بعداً در محاسبه کلیدمخفی فاز اول و دوم، مورداستفاده قرار می‌گیرد.

در IPsec گروه‌هایی بر اساس طول کلید و نوع تولیدکننده رمز، برای دیفی‌هلمن ایجاد گردیده است.

Group Number	Generator	Modulo or Field size
1	MODP[۲]	768–bit modulus
2	MODP	1024-bit modulus
3	EC2N[۳]	155-bit field size
4	EC2N	185-bit field size
5	MODP	1536-bit modulus
14	MODP	2048-bit modulus
15	MODP	3072-bit modulus
16	MODP	4096-bit modulus
17	MODP	6144-bit modulus
18	MODP	8192-bit modulus

دومین جفت پیام‌های Main mode، جهت تبادل کلید از طریق دیفی‌هلمن مورداستفاده قرار می‌گیرد. تعیین نوع دیفی‌هلمن با استفاده از پارامترهای مذاکره شده در گام قبل، انجام می‌شود.

---

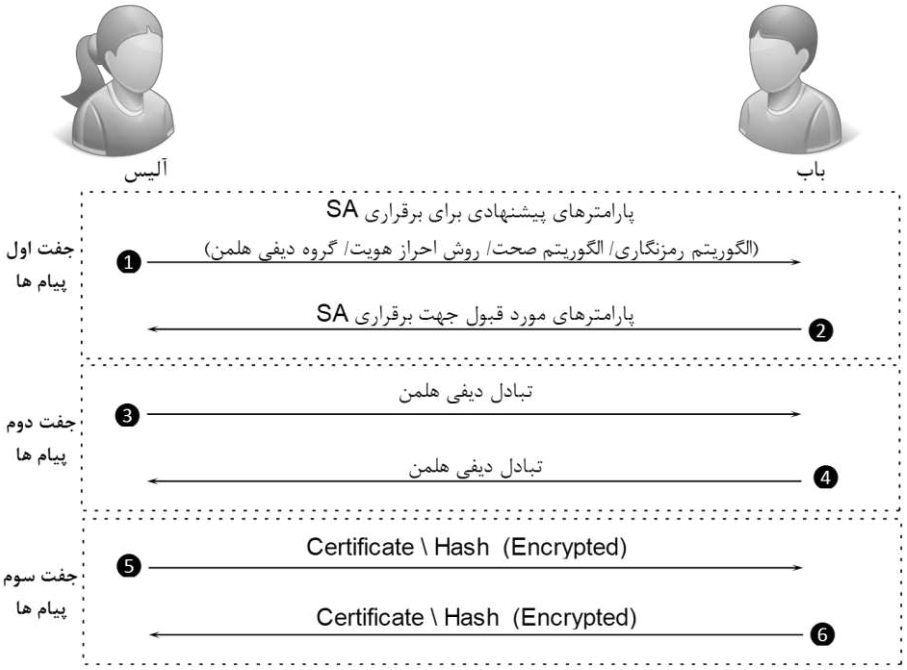
[۱] این کلید با Pre-shared key گفته‌شده در احرازهویت، متفاوت است.

[2] Exponentiation over a prime modulus [MODP]

[3] Elliptic curve over $G[2^N]$

زمانی که برای احرازهویت از Pre-shared Key یا امضای دیجیتال استفاده می‌شود، شناسه‌ها تبادل نمی‌گردند؛ تا جفت سوم پیام‌ها از طریق دیفی‌هلمن کلید را مشخص نموده و امنیت تبادل IDها را تضمین نمایند.

در جفت سوم پیام‌های Main mode، هر طرف اقدام به ارائه اطلاعات احرازهویت خود برای طرف مقابل می‌نماید؛ که این هم به روش احرازهویت توافق شده بستگی دارد. اگر بر روی Pre-shared key توافق شده باشد، Message digest تبادل می‌شود؛ و اگر بر روی امضای دیجیتال توافق شده باشد، از certificateها استفاده می‌شود. صرف‌نظر از روش مورداستفاده، جفت سوم پیام‌ها، بر اساس کلید توافق شده در مرحله قبل (دیفی‌هلمن)، رمزنگاری می‌شوند.



تصویر فوق را به‌صورت خلاصه می‌توان این‌گونه بیان کرد: آلیس پارامترهای پیشنهادی خود را جهت برقراری یک ارتباط امن برای باب ارسال می‌کند.❶ باب پارامترهای موردقبول را انتخاب کرده و به اطلاع آلیس می‌رساند.❷ سپس آلیس❸ و باب❹ توسط دیفی‌هلمن بر روی یک کلید مخفی توافق می‌نمایند. توسط سومین جفت پیام‌ها نیز، آلیس اقدام به احرازهویت خود برای باب کرده؛❺ و باب نیز اطلاعات احرازهویت خود را برای آلیس ارسال می‌کند.❻

۲. روش Aggressive Mode

روش Aggressive یک جایگزین سریع‌تر برای روش Main است. در این روش به‌جای سه جفت پیام (۶ پیام)، از سه پیام برای برقراری IKE SA استفاده می‌شود.

پیام اول و پیام دوم، برای مشخص کردن پارامترهای IKE SA و تبادل کلید مورداستفاده قرار می‌گیرند. مجدداً از پیام دوم به همراه پیام سوم جهت تبادل اطلاعات احرازهویت طرفین استفاده می‌گردد. وظیفه سه پیام مورداستفاده در Aggressive mode به‌صورت زیر است:

- پیام اول

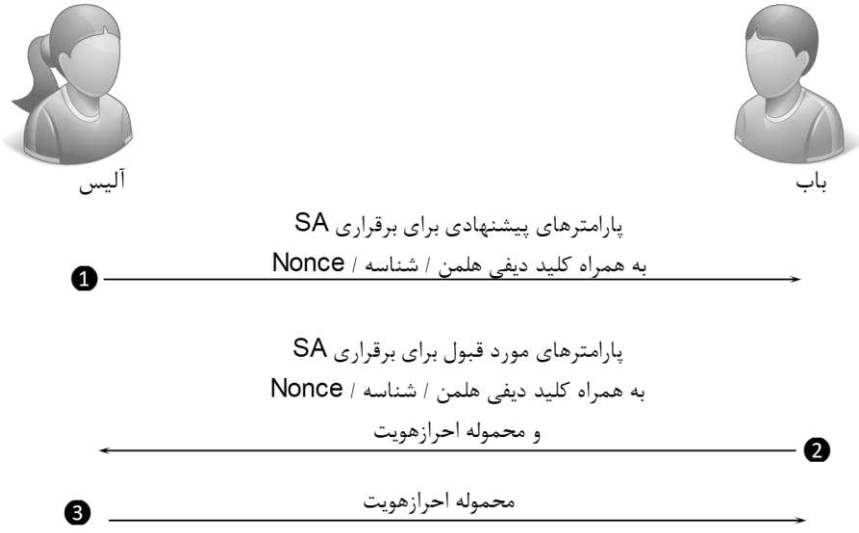
یک‌طرف ارتباط (آلیس)، اقدام به ارسال پیامی حاوی تمام پارامترهای حفاظتی، و همچنین تبادل کلید دیفی‌هلمن، مقدار Nonce و شناسه خود، می‌نماید.

- پیام دوم

طرف مقابل(باب)، اقدام به ارسال پیامی حاوی پارامترهای حفاظتی موردقبول، و همچنین تبادل کلید دیفی‌هلمن، مقدار Nonce، شناسه خود و محموله احرازهویت خود (از طریق Hash یا امضای دیجیتال)، می‌نماید.

- پیام سوم

طرف اول (آلیس)، اقدام به ارسال محموله احرازهویت خود برای طرف مقابل می‌نماید.



همان‌طور که ملاحظه می‌نمایید، تمام اطلاعات تبادل شده در Main mode در Aggressive mode نیز تبادل گردیده است؛ اما با تعداد پیام کمتر و بالطبع سرعت بیشتر.

در حالت Aggressive mode، همان‌طور که سرعت افزایش پیدا می‌کند، در مقابل امنیت کاهش می‌یابد. به دلیل اینکه تبادل کلید دیفی‌هلمن از همان پیام اول شروع می‌شود، لذا دو طرف فرصت مذاکره جهت تعیین پارامترهای دیفی‌هلمن ندارند. همچنین اطلاعات مربوط به شناسه، در حال Aggressive، همیشه به‌صورت پنهانی و رمزشده نیست(مگر در صورت استفاده از الگوریتم‌های کلیدعمومی). توجه داشته باشید که روش Aggressive، مستعد حملات مردمیانی (MITM) است.

با توجه به موارد فوق، جهت افزایش امنیت، پیشنهاد می‌گردد در صورت امکان از Main Mode در فاز اول IKE استفاده نمایید.

فاز دوم IKE

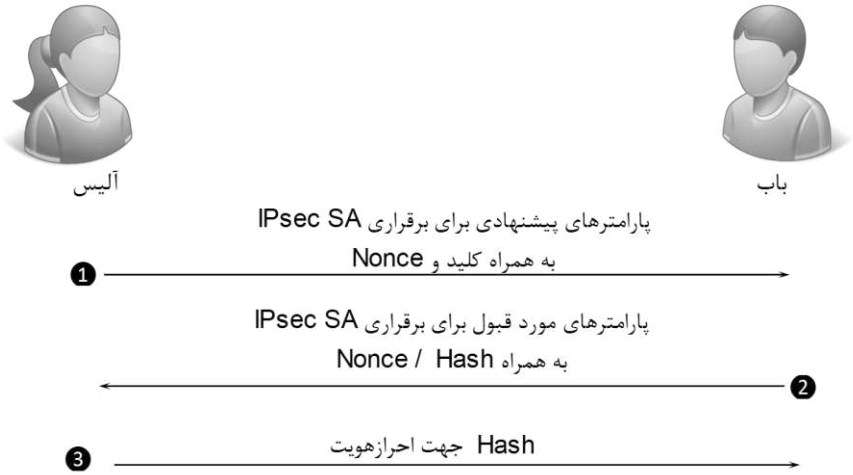
هدف فاز دوم IKE، ایجاد یک SA برای اتصال IPsec واقعی است. منظور از واقعی، اتصالی است که قرار است از آن برای تبادل اطلاعات به‌صورت محرمانه، استفاده گردد. این SA، را IPsec SA می‌نامند. برخلاف IKE SA که اتصالی دوطرفه بود، IPsec SA اتصالی یک‌طرفه است. به همین دلیل برای برقراری ارتباط بین دو سیستم، نیاز به برقراری دو IPsec SA است.

در فاز اول دو حالت Main و Aggressive برای برقراری IKE SA داشتیم؛ اما برای برقراری IPsec SA، فقط یک حالت به نام Quick Mode وجود دارد.

برای برقراری SA، حالت Quick mode از سه پیام استفاده می‌نماید. البته تمام پیام‌های تبادل شده در فاز دوم، بر اساس روش‌های مشخص‌شده در فاز اول، به‌صورت رمزشده، انجام می‌پذیرد.

مهم‌ترین اطلاعات سه پیام تبادل شده در حالت Quick mode در فاز دوم IKE به شرح زیر است:

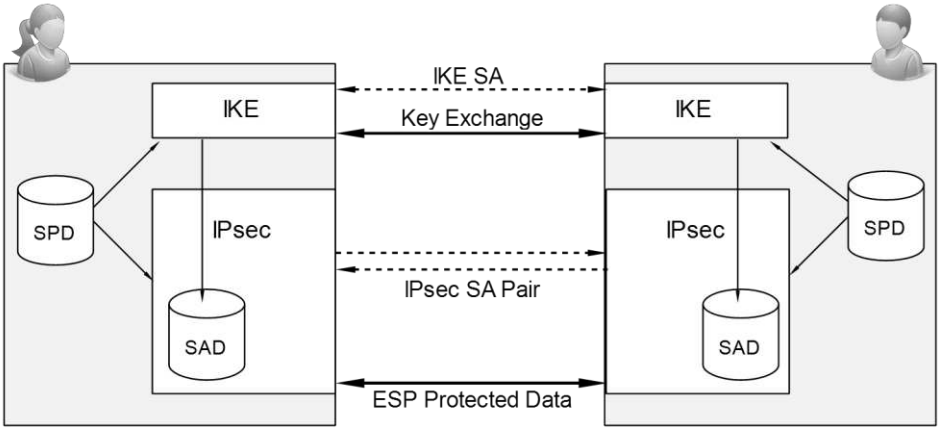
- پیام اول
  یک طرف ارتباط (آلیس)، اقدام به ارسال پارامترهای پیشنهادی جهت IPsec SA، کلید و Nonce، می‌نماید. (مقدار Nonce یک اقدام ضد بازپخش است.)
- پیام دوم
  طرف مقابل (باب)، پارامترهای موردقبول جهت IPsec SA، کلید و Nonce را به همراه یک مقدار Hash جهت احراز هویت، ارسال می‌نماید.
- پیام سوم
  طرف اول (آلیس) نیز در جواب، اقدام به ارسال یک مقدار Hash جهت احرازهویت می‌نماید.



پس‌ازاینکه طرف دوم (باب)، پیام سوم را قبول کرد، اتصال IPsec برقرارشده و تمام SAهای فعال در دیتابیس SAD ذخیره می‌شوند.

هر دو SA مربوط به IKE و IPsec، دارای طول عمر محدود هستند، که پس از ایجاد SA، امکان افزایش آن مقدور نیست. طول عمر SA می‌تواند بر اساس زمان و یا بر اساس مقدار ترافیک شبکه مشخص گردد. اگر یک SA به پایان عمر خود نزدیک شود، نقطه پایانی (آلیس یا باب) باید اقدام به ایجاد یک SA جدید نموده و آن را جایگزین SA موردنظر نماید؛ به این عمل Rekeying گفته می‌شود.

ساختار IPsec و نحوه تعامل دیتابیس‌ها و IKE در تصویر زیر نمایش داده‌شده است:



☑ مبحث دوم

شبکه خصوصی مجازی

یکی از راه‌های برقراری امنیت شبکه، داشتن یک شبکه کاملاً خصوصی است. البته داشتن یک شبکه خصوصی مثل LAN برای سازمان‌ها امکان‌پذیر است، اما درصورتی‌که سازمان ازنظر جغرافیایی پراکنده باشد و یا با کاربران راه دور در ارتباط باشد، برپایی شبکه با لینک‌های خصوصی، سازمان را متحمل هزینه‌های سنگینی خواهد کرد. حتی در برخی موارد داشتن لینک خصوصی امکان‌پذیر نیست.

برای رفع مشکل ارتباطات برای سازمان‌های گسترده، استفاده از شبکه‌های عمومی مثل اینترنت، اجتناب‌ناپذیر است. اما استفاده از شبکه‌های عمومی، امنیت را به طرز چشمگیری کاهش می‌دهد. لذا درجایی که استفاده از لینک‌های خصوصی امکان‌پذیر یا به‌صرفه نیست، باید از شبکه‌خصوصی، اما به‌صورت مجازی استفاده نمود.

شبکه خصوصی مجازی (Virtual Private Network)، که به‌اختصار VPN نامیده می‌شود، با برقراری تونل[1] بر روی بستر شبکه عمومی، یک شبکه خصوصی را فراهم می‌آورد. البته یک تونل در درجه اول باعث برقراری ارتباط مستقیم بین دو نقطه جغرافیایی شده و امنیت را برقرار نمی‌کند؛ لذا برای برقراری امنیت بر روی تونل، باید از پروتکل‌های دیگری نیز بهره گرفت.

به‌عنوان‌مثال، اگر سازمانی داری شعب مختلفی در ایران باشد؛ مثلاً یک ساختمان مرکزی در تهران و شعب دیگری در مشهد مقدس، شیراز و اصفهان داشته باشد؛ در این صورت احتمالاً با همکاری مخابرات و یا سایر شرکت‌های خصوصی، می‌تواند به‌راحتی برای هرکدام از شعب با دفتر تهران یک لینک مستقل برقرار نماید[2]، که ازنظر اقتصادی نیز به‌صرفه است. اما اگر سازمان دارای شعبی در خارج از کشور باشد، آیا بازهم خرید لینک‌های خصوصی به‌صرفه و امکان‌پذیر است؟ آیا لینک‌های خریداری‌شده داخلی و خارجی دارای امنیت قابل‌قبولی هستند؟ در اینجاست که حتی با داشتن لینک خصوصی، بازهم پای امنیت به میان کشیده می‌شود و بهترین راهکار برای رفع این تهدید امنیتی، استفاده از شبکه‌های خصوصی رمزنگاری‌شده است.

---

[1] Tunnel

[2] البته بازهم ازنظر امنیتی به لینک‌هایی که توسط مخابرات و سایر شرکت‌های فراهم‌کننده، برقرار می‌شود نباید اطمینان کرد. حتی در صورت استفاده از لینک‌های شخصی ماهواره‌ای، وایمکس و لینک‌های مخابراتی، مؤکدا توصیه می‌شود، تبادل اطلاعات بر روی این لینک‌ها به‌صورت رمزشده انجام پذیرد.

انواع VPN

تقسیم‌بندی VPN، می‌تواند بر اساس مؤلفه‌های متفاوتی صورت پذیرد. در برندها و مستندات فنی مختلف، از زوایای متفاوتی به VPN نگاه شده و تقسیم‌بندی‌های متفاوتی نیز انجام گرفته است. برخی تونل و VPN را جدا از هم در نظر می‌گیرند؛ چون تونل باعث افزایش امنیت نمی‌شود، آن را جزو خانواده VPN محسوب نمی‌کنند. بعضی دیگر نیز بر اساس الگوریتم‌های رمزنگاری مورداستفاده، VPN را تقسیم‌بندی می‌نمایند. نوع دیگر تقسیم‌بندی نیز بر اساس لایه OSI صورت می‌گیرد. به‌هرحال به قول حضرت مولوی، هرکسی از ظن خود شد یار من![1]

در اینجا، مرسوم‌ترین طبقه‌بندی‌ها را برای شما ذکر می‌کنم.

انواع VPN بر اساس تکنولوژی[2]

ازنظر تکنولوژی، می‌توان VPN را به سه گروه زیر تقسیم‌بندی نمود:

- IPsec
- SSL
- MPLS

انواع VPN بر اساس پروتکل

با توجه به پروتکل مورداستفاده، می‌توان تقسیم‌بندی زیر را انجام داد:

- PPTP
- L2TP
- GRE
- IPsec
- SSL

انواع VPN بر اساس لایه OSI

با توجه به لایه مدل OSI که VPN در آن شکل می‌گیرد، می‌توان تقسیم‌بندی زیر را انجام داد:

- Data link
- Network
- Application

در این مبحث، فارغ از تقسیم‌بندی‌های انجام‌شده، به شرح اجمالی پروتکل‌های مشهوری که به‌صورت گسترده مورداستفاده قرار می‌گیرند، می‌پردازیم.

---

[1] مثنوی معنوی / نی‌نامه

[2] CCNA Security

انواع اتصال در VPN

از هر نوع پروتکلی برای برقراری VPN استفاده شود، نحوه برقراری ارتباطات VPN برای سازمان‌ها و افراد، نهایتاً به یکی از روش‌های زیر صورت می‌پذیرد:

۱. Remote Access VPN

درصورتی‌که نیاز باشد کارمندان از خارج سازمان به منابع اطلاعاتی شبکه داخلی دسترسی داشته باشند، باید از سیستم کاربر به شبکه یا سرور موردنظر یک اتصال VPN برقرار نمود؛ به این نوع اتصال، دسترسی راه دور یا Remote Access گفته می‌شود. در این صورت وظیفه برقراری VPN بر عهده کاربر بوده، و با توجه به سیاست سازمان، ممکن است نقطه انتهایی در سازمان، یک فایروال یا یک سرور باشد.



۲. Site to Site VPN

درصورتی‌که از VPN برای اتصال دو شبکه به یکدیگر استفاده شود؛ آن را Site to Site می‌گویند. در صورت استفاده از Site to Site VPN، بدون آنکه نیازی باشد هر کلاینت به‌صورت مستقل اقدام به برقراری VPN نماید؛ تمام کلاینت‌های هر دو شبکه می‌توانند از طریق کانال امن ایجادشده با یکدیگر ارتباط برقرار نمایند.[۱] در این روش ارتباط بین دو سایت از دید کاربر شفاف بوده و کاربر با آدرس‌های IP داخلی، اقدام به برقراری ارتباط با طرف مقابل می‌نماید. در این نوع ارتباط نقطه انتهایی در هر دو طرف باید روتر یا فایروال باشد.



[۱] البته توسط ACL یا قوانین فایروال، امکان کنترل دسترسی‌ها وجود خواهد داشت.

مزایای استفاده از VPN

با توجه به نوع برقراری VPN و پروتکل‌های مورداستفاده، اتصال VPN ممکن یک یا چند مورد از مزایای زیر را در اختیار شما قرار دهد:

- **محرمانگی**

درصورتی‌که از الگوریتم‌های رمزنگاری یا پروتکل‌های امن مثل IPsec در برقراری VPN استفاده نماییم، ویژگی محرمانگی را خواهیم داشت. در این صورت حتی اگر اطلاعات تبادل شده بر روی شبکه عمومی، توسط افراد ثالث مورد استراق سمع قرار بگیرد، بازهم محرمانگی آن حفظ‌شده و از افشای اطلاعات جلوگیری می‌گردد.

- **صحت و تمامیت داده**

گیرنده می‌تواند از اینکه اطلاعات را بدون تغییر دریافت نموده، اطمینان حاصل نماید. درصورتی‌که در برقراری VPN از پروتکلی استفاده شود که توانایی استفاده از توابع درهم‌ساز را داشته باشد، امکان برقراری مکانیسم حفظ صحت و تمامیت دیتای تبادل شده فراهم می‌گردد.

- **احرازهویت**

قطعاً ویژگی VPN جذاب است، مخصوصاً وقتی‌که مکانیسم‌های محرمانگی و صحت نیز در آن فراهم گردیده باشند. اما آیا شما مطمئن هستید که آن‌طرف تونل همان شخص موردنظر شماست؟ احتمالاً شما با هکر عزیز یک ارتباط تونل برقرار نکرده‌اید؟ مکانیسم احرازهویت باعث می‌گردد تا دو طرف تونل، توانایی احرازهویت طرف مقابل را داشته باشند. البته مکانیسم احراز هویت نیز همانند دیگر ویژگی‌ها، زمانی در دسترس خواهد بود که پروتکل مورداستفاده در برقراری تونل، امکان آن را فراهم نماید. برای ارتباط Site to Site، احرازهویت می‌تواند با Pre-shared key یا رمزنگاری کلیدعمومی صورت پذیرد. برای ارتباط Remote Access نیز می‌توان احرازهویت را بر اساس نام کاربری و کلمه عبور انجام داد؛ هرچند که محدودیتی برای استفاده از روش‌های Pre shared key و کلیدعمومی نیز جهت احراز هویت کاربر وجود ندارد.

- **ضد بازپخش**

هکر می‌تواند با استراق سمع و دریافت بسته‌های تبادل شده در تونل، آن‌ها را مجدداً برای گیرنده ارسال نموده و باعث پر شدن ترافیک یک SA گردد. در این صورت طرفین مجبورند دوباره اقدام به برقراری SA جدید نمایند؛ هکر هم می‌تواند در زمان برقراری اتصال جدید، بخت خود را برای حمله مردمیانی آزمایش نماید. برای رفع این مشکل،

پروتکل‌های مورداستفاده در برقراری تونل، مکانیسم مقابله با بازپخش را ارائه می‌نمایند.

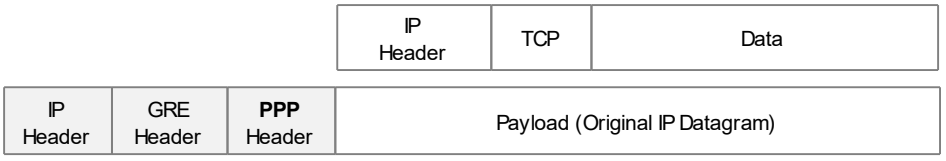
پروتکل PPTP

یکی از ساده‌ترین روش‌های برقراری ارتباط VPN، استفاده از پروتکل PPTP است. کلمه PPTP مخفف عبارت Point to Point Tunneling Protocol است. این پروتکل در سال 1999 توسط شرکت مایکروسافت ارائه گردید و بعداً طی RFC 2637 منتشر شد.

پروتکل PPTP، فقط اقدام به برقراری یک تونل بین فرستنده و گیرنده نموده؛ و هیچ‌گونه محرمانگی برای دیتای تبادل شده فراهم نمی‌آورد. اساس به وجود آمدن این پروتکل، فراهم آوردن دسترسی کلاینت/سروری بوده است. عملیات این پروتکل در لایه دوم مدل OSI صورت می‌گیرد، لذا امکان استفاده از آن به همراه پروتکل‌های لایه سوم دیگری غیر از IP نیز وجود دارد. پروتکل PPTP، یک پروتکل اتصال‌گرا است و در صورت استفاده با پروتکل IP، از پورت TCP 1723 استفاده می‌نماید.

همیشه به خاطر داشته باشید که وقتی اسم تونل به میان می‌آید، حتماً باید بسته دیتا مجدداً بسته‌بندی یا Encapsulate شود. **اساس کار تونل، بر بسته‌بندی مجدد و اضافه کردن آدرس مبدأ و مقصد جدید است**. آدرس مبدأ نقطه شروع تونل و آدرس مقصد نقطه انتهایی تونل است. در نقطه انتهایی نیز مجدداً بسته به حالت اول خودش برگشته؛ و از این به بعد، بر اساس سرآیند اصلی هدایت می‌شود.

در PPTP نیز همانند هر تونل دیگری، عملیات بسته‌بندی مجدد صورت می‌گیرد. اگر پروتکل لایه سه، IP باشد؛ بسته‌بندی جدید به‌صورت زیر خواهد بود:



همان‌طور که ملاحظه می‌کنید، یک فیلد بانام GRE[1] نیز در بسته جدید قرارگرفته است. این فیلد شامل اطلاعاتی راجع به مسیریابی، توالی و تشخیص خطا می‌باشد.

[1] Generic Routing Encapsulation (GRE)

همان‌طور که گفته شد، PPTP به‌صورت پیش‌فرض فقط یک تونل برقرار می‌کند؛ اما مایکروسافت برای اینکه سرویس‌های محرمانگی و احرازهویت را در PPTP فراهم آورد، امکان استفاده از پروتکل‌هایی نظیر MS-CHAP v2[1]، MPPE[2] و PAP[3] را در کنار PPTP فراهم آورده است.

ازآنجاکه احساس می‌کنم ذهن مبارکتان را بانام بردن از پروتکل‌های جدید، مشغول کرده‌ام، لذا به‌صورت اجمالی به توضیح آن‌ها می‌پردازم.

پروتکل PAP

پروتکل PAP، یک پروتکل ساده جهت احراز هویت است. نام کاربری و کلمه عبور توسط این پروتکل به‌صورت متن ساده به سرور انتقال می‌یابد. معمولاً جهت احراز هویت یک کاربر برای سرور از این پروتکل استفاده می‌شود.

حتماً به یاد داشته باشید که ارسال نام کاربری و کلمه عبور توسط این پروتکل بدون هیچ مکانیسم امنیتی و به‌صورت متن ساده انجام می‌گیرد.

پروتکل CHAP

پروتکل CHAP، یک پروتکل احرازهویت است که بر اساس درخواست و پاسخ[4] کار می‌کند. این پروتکل در جواب درخواست اطلاعات کاربری، از الگوریتم درهم‌سازی MD5، برای رمزکردن اطلاعات کاربری استفاده نموده و سپس آن‌ها را ارسال می‌نماید. برخلاف PAP که اطلاعات کاربری به‌صورت متن واضح ارسال می‌شد؛ در CHAP به‌صورت Hash شده ارسال می‌گردد.

پروتکل MS-CHAP

پروتکل MS-CHAP نسخه مایکروسافتی، پروتکل CHAP است. مایکروسافت این پروتکل را در دو نسخه ارائه داده است. در نسخه دوم، پروتکل را قدرتمندتر کرده و ایرادات نسخه اول را نیز مرتفع نموده است.

---

[1] Microsoft Challenge-Handshake Authentication Protocol (MS-CHAP)
[2] Microsoft Point-to-Point Encryption (MPPE)
[3] Password Authentication Protocol (PAP)
[4] Challenge-response

پروتکل MPPE

مایکروسافت در اتصالات نقطه‌به‌نقطه و PPTP، پروتکل MPPE را جهت افزودن مکانیسم محرمانگی معرفی کرده است. پروتکل MPPE، جهت رمزنگاری اطلاعات تبادل شده توسط یک لینک ارتباطی کاربرد دارد. طرح MPPE امکان استفاده از کلیدهایی با طول ۴۰ بیت، ۵۶ بیت و ۱۲۸ بیت را دارد.

پروتکل MPPE از پروتکل‌های رمزنگاری متقارن، جهت محرمانگی استفاده می‌نماید. الگوریتم رمزنگاری مورداستفاده در این پروتکل، RC4 است.

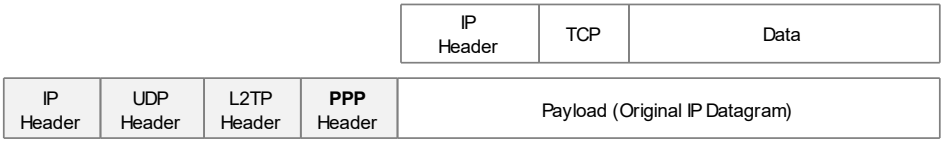
پروتکل L2TP

کلمه L2TP مخفف عبارت Layer 2 Tunneling Protocol است. این پروتکل در لایه دوم مدل OSI، اقدام به برقراری تونل می‌نماید. این پروتکل ابتدا توسط شرکت سیسکو ارائه گردید و سپس IETF آن را طی RFC 2661[1] منتشر نمود. سیسکو L2TP را از ترکیب پروتکل‌های PPTP و L2F[2]، ایجاد نموده است.

پروتکل L2TP، فارغ از پروتکل مورداستفاده در لایه شبکه، اقدام به ایجاد تونل می‌نماید؛ لذا هم امکان استفاده در شبکه‌های مبتنی برIP و هم در شبکه‌های دیگری مثل ATM[3] و Frame Relay را دارد. درصورتی‌که L2TP در شبکه مبتنی برIP مورداستفاده قرار گیرد، تبادل اطلاعات خود را روی پورت UDP 1701 انجام می‌دهد.

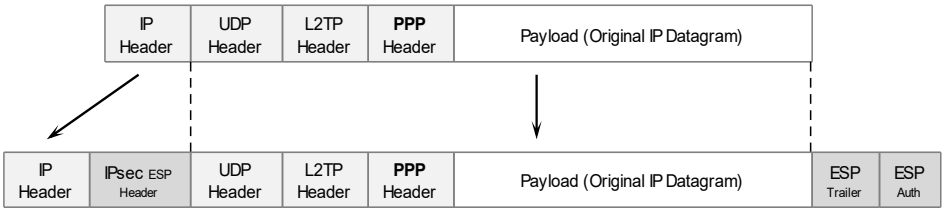
همانند PPTP، پروتکل L2TP به‌تنهایی، فقط برقرارکننده تونل بوده و برای فراهم آوردن احرازهویت و محرمانگی از سایر پروتکل‌ها بهره می‌برد. برای احرازهویت، همانند پروتکل PPTP از پروتکل‌هایی مثل CHAP و PAP استفاده نموده ولی برای محرمانگی برخلاف PPTP، می‌تواند از IPsec، Pre-shared key و Certificate Authority استفاده نماید.

تصویر زیر نحوه بسته‌بندی مجدد پروتکل L2TP نمایش می‌دهد.



[1] https://tools.ietf.org/html/rfc2661
[2] Layer 2 Forwarding (L2F)
[3] Asynchronous Transfer Mode (ATM)

درصورتی‌که از پروتکل IPsec برای رمزنگاری درL2TP استفاده شود؛ این پروتکل را به‌صورت L2TP/IPsec نام‌گذاری می‌نمایند[1]. استفاده همزمان از IPsec و L2TP باعث می‌شود تا بسته دیتای اصلی، دو بار به‌صورت متوالی بسته‌بندی گردد، دفعه اول توسط L2TP و دفعه دوم توسط IPsec. تصویر زیر نمایش‌دهنده نحوه بسته‌بندی دوم توسط IPsec است.



پروتکل IPsec

در مورد پروتکل IPsec در مبحث اول، به‌طور مفصل صحبت شد. پروتکل IPsec شامل چند پروتکل دیگر، جهت برقراری مکانیسم‌های احرازهویت و محرمانگی است. نحوه به‌کارگیری IPsec به دو صورت Transport و Tunnel است.

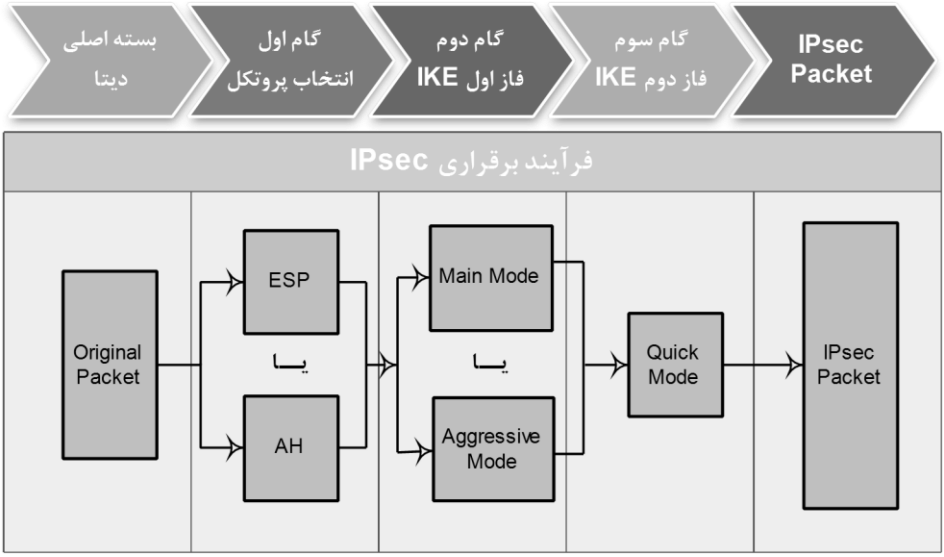
در حال Transport، هیچ تونلی ایجاد نشده و بسته از همان سرآیند اصلی استفاده می‌کند. در این حالت، IPsec سرویس‌های محرمانگی، صحت، ضدبازپخش و احرازهویت را برای محموله بسته دیتا فراهم می‌آورد. پروتکل‌های دیگری مثل L2TP که صرفاً توانایی ایجاد تونل را دارند، برای فراهم کردن سرویس‌های امنیتی برای تونل خود، از پروتکل IPsec در حالت Transport بهره می‌برند.

پروتکل IPsec می‌تواند با بسته‌بندی مجدد، ویژگی تونل را نیز فراهم آورد. در این روش IPsec، ضمن برقراری سرویس‌های محرمانگی، صحت، ضدبازپخش و احرازهویت، فرآیند بسته‌بندی مجدد را با اضافه کردن سرآیند جدید، انجام می‌دهد. این روش نسبت به روش Transport از امنیت بالاتری برخوردار است؛ چراکه این بار کل بسته اصلی دیتا به‌عنوان Payload بسته جدید قرارگرفته و تمام سرویس‌های امنیتی بر روی آن اعمال می‌شوند.

پروتکل IPsec بیشترین کاربرد را در زمان برقراری اتصالات Site to site VPN دارد. در این نوع اتصال، کل ارتباط بین دو شبکه از طریق یک کانال امن صورت می‌گیرد؛ بدون آنکه نیاز باشد کاربر تنظیم خاصی را بر روی دستگاه خود انجام دهد.

تصویر زیر فرآیند عملیات IPsec را به‌صورت ساده نمایش می‌دهد:

[1] RFC 3193



پروتکل SSL VPN

پروتکل‌هایی که تاکنون بررسی شد در لایه‌های دوم و سوم مدل OSI، عملیات مربوط به ایجاد تونل و اجرای مکانیسم‌های امنیتی را انجام می‌دادند. این نوع پروتکل‌ها از دید لایه‌های بالاتر مثل لایه کاربرد، شفاف بوده و هیچ وابستگی به پروتکل مورداستفاده در لایه‌های بالاتر ندارند. این امر می‌تواند برای مدیران شبکه، مخصوصاً در سازمان‌های بزرگ که بخش شبکه و بخش برنامه‌های کاربردی از یکدیگر مستقل هستند، مطلوب باشد.

اما اگر مدیر شبکه مجبور باشد امنیت را برای یک برنامه خاص نیز فراهم آورد، چه راه‌کاری خواهد داشت؟ در بعضی مواقع سازمان به دلیل عدم احساس نیاز به یک پروتکل امن در لایه‌های بالا، سفارش نرم‌افزار خود را بدون در نظر گرفتن راه‌کارهای امنیتی مطرح می‌نماید. مثلاً سازمان به دلیل اطمینان به شبکه داخلی خود، نرم‌افزار اتوماسیون را بدون نیاز به پروتکل امنیتی مثل SSL سفارش داده، اما حالا به دلایلی نیاز دارد بر روی اینترنت نیز اتوماسیون را در اختیار برخی از کاربران خود قرار دهد. مثل زمانی که مدیرکل محترم سازمان به سفر خارج از کشور تشریف برده و می‌خواهد ازآنجا کارتابل خود را مشاهده نماید. در اینجاست که مدیر IT مجبور است امکان دسترسی به اتوماسیون را از طریق یک بستر عمومی خطرناک مثل اینترنت برای مدیریت فراهم آورد. اما از طرفی امکان تغییر اتوماسیون و گنجاندن پروتکلی مثل SSL از عهده مدیر شبکه خارج و البته زمان‌گیر است؛ و از طرف دیگر آموزش برقراری VPNهایی مثل IPsec و L2TP به مدیریت معزز، بسیار دشوار خواهد بود. در این شرایط بهترین راه‌کار، می‌تواند

برقراری یک ارتباط SSL VPN باشد. چراکه یکی از مزایای SSL VPN این است که برخلاف سایر VPNها، کاربر بدون انجام تنظیمات پیچیده، به‌راحتی با یک مرورگر وب می‌تواند اتصال امن را برقرار نماید. وقتی‌که تونل SSL ایجاد گردد، غالب برنامه‌ها می‌توانند از طریق آن به‌صورت امن اقدام به تبادل دیتا نمایند، چراکه این تونل در لایه Transport برقرار گردیده و از دید برنامه کاربردی شفاف خواهد بود.

پروتکل SSL VPN ماهیتاً برای دسترسی کاربران راه دور به منابعی مثل وب سرورها ارائه گردیده است؛ ولی اخیراً بعضی از تولیدکنندگان تجهیزات امنیتی بر روی محصولات خود امکان Site to Site SSL VPN را ارائه نموده‌اند که البته هنوز به بلوغ نرسیده و کارایی IPsec Site to Site VPN را ندارد.

برقراری SSL VPN به دو صورت زیر امکان‌پذیر است:[1]

۱. **SSL Portal VPN**

در این حالت امکان دسترسی امن کاربر به وب، از طریق SSL فراهم می‌گردد. نام‌گذاری این نوع VPN بانام پورتال به این دلیل است که معمولاً یک صفحه وب، منجر به دسترسی به سایر منابع شبکه می‌گردد. درروش SSL Portal VPN از پورت TCP 443، برای تبادل دیتا استفاده می‌گردد.

در این روش کاربر با استفاده از یک مرورگر وب[2]، اقدام به شناساندن خود به دروازه SSL می‌نماید؛ پس از احرازهویت، سرویس‌دهنده یک صفحه وب در اختیار کاربر قرار می‌دهد تا از طریق آن بتواند به منابع مجاز دسترسی داشته باشد.

۲. **SSL Tunnel VPN**

در این حالت نیز کاربر از طریق یک مرورگر به دروازه SSL متصل می‌شود، اما کارایی این روش بیشتر بوده و اتصال امن فقط محدود به یک برنامه خاص نیست.

مرورگر باید توانایی پشتیبانی از انواع خاصی از محتوای فعال (از قبیل جاوا، جاوا اسکریپت، فلش و ActiveX) را داشته باشد؛ و کاربر نیز باید قادر به اجرا کردن آن‌ها باشد.

تونل SSL VPN، شبیه سایر تونل‌ها ایجاد می‌گردد. با استفاده از تونل، به‌جای پشتیبانی از یک برنامه خاص، می‌توان برای طیف وسیعی از برنامه‌ها مکانیسم‌های امنیتی را فراهم نمود.

---

[1] http://csrc.nist.gov/publications/nistpubs/800-113/SP800-113.pdf
[2] Web browser

بخش چهارم

امنیت سیستم‌ها و سرویس‌ها

فصل دهم

امنیت وب

فصل یازدم

امنیت پست الکترونیک

فصل دوازدهم

امنیت سیستم‌عامل

فصل سیزدهم

امنیت دیتابیس

فصل دهم

امنیت وب

مبحث اول

پروتکل SSL/TLS

مبحث دوم

پروتکل HTTPS

☑ مبحث اول

پروتکل SSL/TLS

شرکت Netscape برای برقراری امنیت در ارتباطات اینترنتی، اقدام به انتشار یک پروتکل امنیتی بانام Secure Socket Layer (SSL) نموده و از آن، در مرورگرهای خود پشتیبانی کرد. سازمان IETF به دلیل کاربرد وسیع و مناسب SSL، اقدام به انتشار این پروتکل در قالب استاندارد Transport Layer Security(TLS) و طی RFC 5246 نمود[۱]. پروتکل TLS بر اساس SSL v3.1 ارائه گردیده و از بسیاری جهات شبیه به یکدیگر می‌باشند. لذا در این مبحث پروتکل SSL v3 (ارائه‌شده در RFC 6101) تشریح خواهد گردید.

لازم به ذکر است، IETF در سال 2011 طی RFC 6176 استفاده از SSL v2 را به دلایلی مثل استفاده ازMD5 در احرازهویت، عملیات Handshake محافظت نشده، استفاده از کلید یکسان برای هر دو عملیات رمزنگاری و صحت، و همچنین آسیب‌پذیری در مقابل حملات مردمیانی، ممنوع اعلام کرده است.

معماری SSL

پروتکل SSL طوری طراحی‌شده که از TCP برای فراهم آوردن یک سرویس End-to-end امن و قابل‌اطمینان استفاده می‌نماید. همان‌طور که در تصویر زیر نشان داده‌شده، SSL یک پروتکل تنها نیست، بلکه شامل دو لایه از پروتکل‌ها است. به‌عبارت‌دیگر SSL، یک پروتکل لایه‌بندی شده است.

<table>
<tr><td>SSL Handshake Protocol</td><td>SSL Change Cipher Spec Protocol</td><td>SSL Alert Protocol</td><td>Application (HTTP)</td></tr>
<tr><td colspan="4">SSL Record Protocol</td></tr>
<tr><td colspan="4">TCP</td></tr>
<tr><td colspan="4">IP</td></tr>
</table>

[۱] نسخه‌های اولیه این استاندارد عبارتند از RFC 3268، RFC 4346 و RFC 4366.

توجه داشته باشید که لایه‌بندی پروتکل SSL، برای تعریف بهتر و نحوه تعامل پروتکل‌ها با یکدیگر بوده و تأثیری در لایه‌های مدل OSI ندارد.

پروتکل SSL Record فراهم آورنده سرویس‌های امنیتی اساسی برای پروتکل‌های مختلف لایه‌های بالاتر است. به‌طور خاص، پروتکل HTTP که یک سرویس تعاملی کلاینت/سروری است، می‌تواند در بالای SSL مورداستفاده قرار گیرد.

پروتکل SSL، علاوه بر SSL Record Protocol، دارای سه پروتکل دیگر در لایه بالاتر است که عبارتند از: SSL Handshake Protocol، SSL Change Cipher Spec Protocol و SSL Alert Protocol؛ این پروتکل‌ها برای مدیریت تبادلات SSL مورداستفاده قرار می‌گیرند.

مفاهیم Session و Connection در SSL

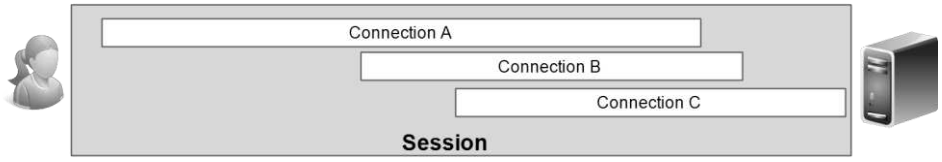
دو مفهوم مهم در SSL، عبارتند از Connection و Session، که تشریح آن‌ها به‌صورت زیر است:

- **Connection**

یک اتصال (Connection)، یک انتقال (لایه چهارم مدل OSI یا همان Transport) بوده، که نوع مناسبی از سرویس را فراهم می‌کند. برای SSL، این قبیل اتصال‌ها، ارتباط Peer-to-peer هستند.[۱] اتصال‌ها، زودگذر و موقتی هستند. هر Connection با یک Session در ارتباط است.

- **Session**

یک SSL Session، یک ارتباط بین کلاینت و سرور است؛ که توسط پروتکل Handshake ایجاد می‌گردد. نشست‌ها (Session) مجموعه‌ای از پارامترهای امنیتی رمزنگاری را تعریف می‌کنند که می‌توانند در میان اتصالات متعدد به اشتراک گذاشته شوند. نشست‌ها از انجام مذاکرات متعدد و پرهزینه که برای مشخص کردن پارامترهای امنیتی جدید در هر اتصال باید انجام شود، جلوگیری می‌کنند.



[۱] در ارتباطات Peer-to-peer، دو نقطه بدون نیاز به برقراری یک ساختار متمرکز و پرهزینه، اقدام به برقراری ارتباط با یکدیگر می‌نمایند.

بین دو طرف (مثل ارتباط HTTP یک کلاینت با سرور)، ممکن است چندین Connection امن وجود داشته باشد. البته در تئوری، ممکن است چندین Session هم‌زمان نیز بین یک کلاینت و سرور، وجود داشته باشد؛ اما این ویژگی در عمل استفاده نمی‌شود.

برای هر Session، تعدادی وضعیت (State) وجود دارد که بعد از برقراری یک Session، به‌عنوان وضعیت عملیات جاری، برای هر دو عمل خواندن و نوشتن (یا ارسال و دریافت) قرار می‌گیرند. همچنین در حین پروتکل Handshake[۱]، وضعیت‌های خواندن و نوشتن معلق نیز به وجود می‌آیند که پس از اتمام موفقیت‌آمیز Handshake، به‌عنوان وضعیت جاری قرار می‌گیرند.

پارامترهای تعیین وضعیت Session

وضعیت یک Session از طریق پارامترهای زیر تعیین می‌شود:

- **Session identifier**
یک توالی بایت دلخواه، که توسط سرور برای شناسایی وضعیت یک Session فعال یا قابل ادامه[۲]، انتخاب می‌شود.
- **Peer certificate**
یک گواهینامه X.509 v3[۳] از طرف مقابل. البته این فیلد، ممکن است تهی[۴] باشد.
- **Compression method**
الگوریتم مورداستفاده جهت فشرده‌سازی. (البته فشرده‌سازی قبل از رمزگذاری.)
- **Cipher spec**
تعیین الگوریتم رمزنگاری (مثل AES، Null و غیره)، و یک الگوریتم درهم‌ساز (مثل MD5 یا SHA-1) برای محاسبه MAC. این پارامتر همچنین ویژگی‌های رمزنگاری، مثل Hash-size را تعیین می‌کند.
- **Master secret**
یک مقدار ۴۸ بایتی مخفی، که بین کلاینت و سرور به اشتراک گذاشته‌شده است.
- **Is resumable**
یک علامت[۵] برای نشان دادن اینکه آیا این Session می‌تواند برای شروع یک Connection جدید مورداستفاده قرار گیرد؟

---

[۱] به معنی دست دادن / تأیید

[2] Resumable

[۳] استاندارد X.509، برای تعریف گواهینامه دیجیتال بر اساس زیرساخت کلیدعمومی (PKI) ارائه گردیده است. این استاندارد توسط IETF و طی RFC 5280 ارائه گردیده است.

[4] Null

[5] Flag

پارامترهای تعیین وضعیت Connection

وضعیت یک Connection از طریق پارامترهای زیر تعیین می‌شود:

- **Server and client random**
یک توالی بایتی که توسط سرور و کلاینت برای هر Connection انتخاب می‌شود.
- **Server write MAC secret**
کلید مخفی استفاده‌شده برای محاسبه MAC دیتای ارسال‌شده توسط سرور.
- **Client write MAC secret**
کلید مخفی استفاده‌شده برای محاسبه MAC دیتای ارسال‌شده توسط کلاینت.
- **Server write key**
کلید مخفی رمزنگاری که توسط سرور برای رمزگذاری استفاده‌شده و توسط کلاینت باید برای رمزگشایی استفاده شود.
- **Client write key**
کلید مخفی رمزنگاری متقارن که توسط کلاینت برای رمزگذاری استفاده‌شده و توسط سرور باید برای رمزگشایی استفاده شود.
- **Initialization vectors**
وقتی‌که از روش CBC[۱] برای یک بلوک رمزنگاری استفاده می‌شود، برای هر کلید به یک IV نیاز است.[۲]
مقداردهی این فیلد برای اولین بار توسط پروتکل Handshake انجام می‌شود؛ اما پس‌ازآن، از بلوک رمزشده قبلی، برای IV بعدی استفاده می‌شود.
- **Sequence numbers**
هر طرف به‌صورت جداگانه شماره توالی پیام‌های ارسال و دریافت شده برای هر Connection را نزد خود نگه می‌دارند.
زمانی که یک‌طرف، پیام Change Cipher Spec[۳] را ارسال یا دریافت می‌کند، این مقدار را صفر می‌نماید.
شماره توالی نباید از مقدار $2^{64} - 1$ فراتر رود.

در ادامه مبحث به تشریح پروتکل‌های تشکیل‌دهنده SSL می‌پردازیم:

---

[1] Cipher Block Chaining

[۲] در بخش رمزنگاری همین کتاب، روش CBC و دیگر روش‌های رمزنگاری بلوکی تشریح شده است. درروش CBC بلوک‌های رمزشده به‌صورت زنجیروار به یکدیگر مرتبط می‌شوند.
درروش CBC، در اولین بلوک از مقدار IV برای ورودی الگوریتم استفاده می‌شود، ولی برای بلوک‌های بعدی، مقدار رمزشده بلوک قبلی مورداستفاده قرار می‌گیرد.

[۳] تغییر مشخصات رمزنگاری (Change Cipher Spec).

پروتکل SSL Record

پروتکل SSL Record، دو سرویس را برای SSL Connection فراهم می‌آورد:

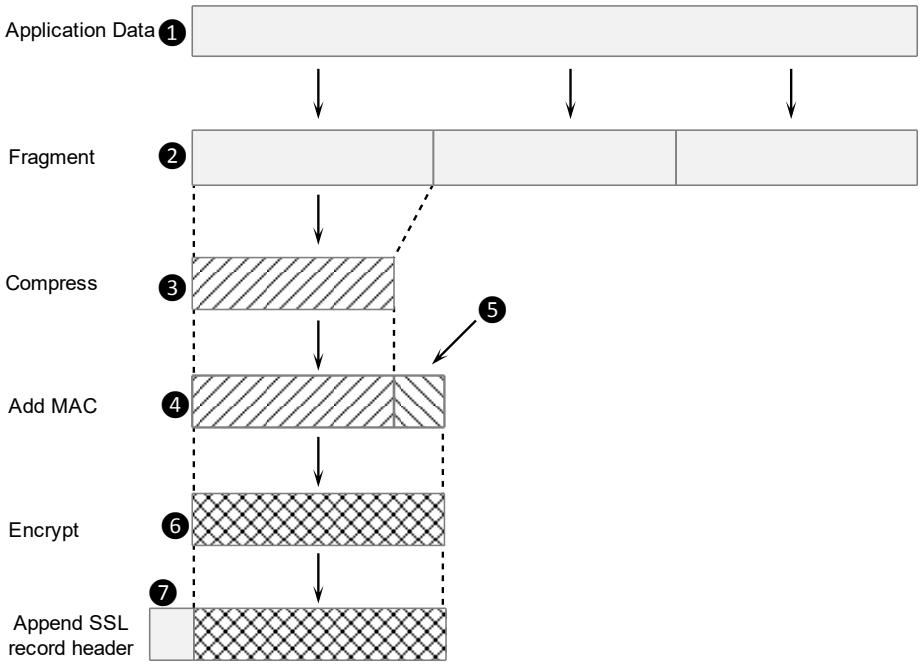
- **محرمانگی**

پروتکل Handshake، یک کلید مخفی مشترک تعیین می‌کند که از آن برای رمزگذاری محموله SSL استفاده می‌گردد.

- **صحت پیام**

پروتکل Handshake، همچنین یک کلید مخفی مشترک، تعیین می‌کند که از آن برای تولید MAC استفاده می‌شود.

تصویر زیر مراحل عملیات پروتکل SSL Record را نمایش می‌دهد:



پروتکل SSL Record یک بلوک دیتا را تحویل می‌گیرد. ❶ این بلوک دیتا باید به قطعات 2^{14} بایتی یا کوچک‌تر تقسیم‌بندی شود. ❷ در گام بعدی، قطعات ایجادشده فشرده‌سازی می‌شوند. ❸ البته فشرده‌سازی می‌تواند به‌صورت اختیاری انجام پذیرد؛ در SSL v3 و TLS، به‌صورت پیش‌فرض الگوریتمی برای فشرده‌سازی انتخاب‌نشده است.

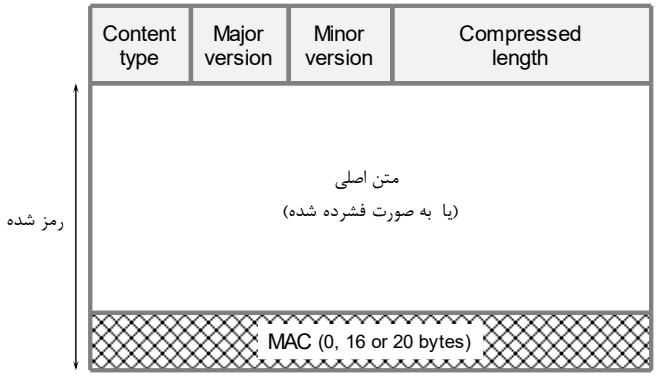
پس از فشرده‌سازی باید مقدار MAC محاسبه‌شده❹ و به انتهای پیام الصاق شود.❺ درصورتی‌که فشرده‌سازی انجام‌شده باشد، محاسبه MAC بر اساس دیتای فشرده‌شده انجام می‌پذیرد. برای محاسبه MAC به یک کلید مخفی نیاز است. نحوه محاسبه MAC بسیار شبیه توضیحات ارائه‌شده در بخش رمزنگاری است[۱]؛ البته با تفاوت‌های کوچک.

در گام بعد، پیام فشرده‌شده به‌علاوه مقدار MAC، با استفاده از یک الگوریتم رمزنگاری متقارن، رمزگذاری می‌شوند.❻ ازجمله الگوریتم‌های رمزنگاری متقارن مورداستفاده، می‌توان از الگوریتم‌های AES، DES، 3DES، IDEA و RC4، نام برد. همان‌طور که گفته‌شده مقدار MAC نیز همراه پیام (یا فشرده پیام) رمزگذاری می‌شود.

در گام آخر، پروتکل SSL Record، اقدام به‌اضافه کردن یک سرآیند به قطعه رمزشده می‌نماید.❼ این سرآیند شامل فیلدهای زیر است:

- **Content Type (8 bits)**
مشخص‌کننده پروتکل لایه بالاتر که این قطعه را پردازش نموده است. این پروتکل ممکن است Change Cipher Spec، Alert، Handshake یا Application Data باشد.
- **Major Version (8 bits)**
نشان‌دهنده نسخه اصلی SSL مورداستفاده است. برای SSL v3، این مقدار 3 است.
- **Minor Version (8 bits)**
نشان‌دهنده نسخه جزئی SSL مورداستفاده است. برای SSL v3، این مقدار 0 است.
- **Compressed Length (16 bits)**
طول قطعه متن اصلی (یا قطعه فشرده‌شده) در قالب بایت. حداکثر این مقدار می‌تواند $2^{14} + 2048$، باشد.

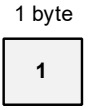
تصویر زیر نشان‌دهنده فرمت خروجی SSL Record است:



[۱] طبق RFC 2104

پروتکل Change Cipher Spec

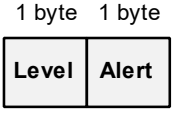
پروتکل Change Cipher Spec یکی از سه پروتکلی است که از SSL Record استفاده می‌نماید. این پروتکل ساده، تنها دارای یک پیام یک بایتی، با مقدار 1 است. تنها هدف این پیام این است که باعث کپی شدن وضعیت معلق در وضعیت جاری گردد؛ این به‌روزرسانی باعث می‌شود مجموعه رمزنگاری که باید برای این Connection مورداستفاده قرار گیرد، مشخص شود.



پروتکل Alert

از این پروتکل جهت انتقال هشدارهای مربوط به SSL، به‌طرف مقابل استفاده می‌شود. همان‌طور که برنامه‌های دیگر از SSL استفاده می‌کنند؛ پیام‌های هشدار نیز بر اساس ویژگی‌های اعلام‌شده توسط وضعیت جاری (Current State)، فشرده‌سازی و رمزنگاری می‌شوند.

هر پیام این پروتکل از دو بایت تشکیل‌شده است. تصویر زیر نمایش‌دهنده بایت‌های تشکیل‌دهنده Alert است:



بایت اول مشخص‌کننده سطح (Level) پیام است؛ که با مقادیر زیر مشخص می‌شود:

- برای پیام هشدار یا Warning، مقدار 1.
- برای پیام فاجعه‌آمیز یا Fatal، مقدار 2.

اگر بایت اول مشخص‌کننده پیام فاجعه‌آمیز یا Fatal باشد، SSL سریعاً به آن Connection خاتمه می‌دهد. ممکن است دیگر Connectionهای موجود بر روی همان Session ادامه یابند، ولی هیچ Connection جدیدی نمی‌تواند بر روی این Session برقرار شود.

بایت دوم شامل کدی است که مشخص‌کننده هشدار می‌باشد. این بایت دقیقاً به معرفی پیام هشداری که اتفاق افتاده می‌پردازد.

هشدارهای زیر همیشه مربوط به سطح Fatal است:

- **unexpected_message**
  یک پیام نامناسب دریافت شده است.
- **bad_record_mac**

یک مقدار MAC نادرست دریافت شده است.

- **decompression_failure**
تابع باز کردن فشرده‌سازی[1]، یک ورودی نامناسب دریافت کرده است.
- **handshake_failure**
با توجه به گزینه‌های قابل‌دسترس، فرستنده قادر به مذاکره برای قبول یک مجموعه از پارامترهای امنیتی نیست.
- **illegal_parameter**
یک فیلد در پیام Handshake، خارج از محدوده و یا در تضاد با سایر فیلدها است.

سایر هشدارها نیز به‌صورت زیر است:

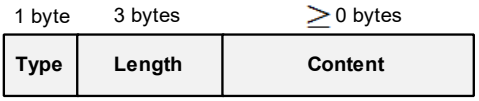
- **close_notify**
فرستنده به اطلاع گیرنده می‌رساند که دیگر از طریق این Connection پیام ارسال نخواهد کرد.
هر طرف که قصد بستن سَمت Write یک Connection را داشته باشد، باید قبل از آن، اقدام به ارسال پیام close_notify نماید.
- **no_certificate**
درصورتی‌که گواهینامه‌ای وجود نداشته باشد، این پیام در جواب درخواست گواهی‌نامه ارسال می‌گردد.
- **bad_certificate**
گواهی دریافت شده، خراب است. (به‌عنوان‌مثال، شامل یک امضای تأیید نشده است.)
- **unsupported_certificate**
نوع گواهی‌نامه دریافت شده، پشتیبانی نمی‌شود.
- **certificate_revoked**
گواهی‌نامه توسط امضاکننده آن (صاحب گواهی‌نامه)، ابطال گردیده است.
- **certificate_expired**
تاریخ اعتبار گواهی‌نامه منقضی شده است.
- **certificate_unknown**
اتفاقات نامشخصی برای پردازش گواهینامه به وجود آمده، که تفسیر آن را غیرقابل‌قبول نموده است.

[1] Decompression

پروتکل Handshake

پیچیده‌ترین بخش SSL، پروتکل Handshake است. این پروتکل اجازه می‌دهد تا سرور و کلاینت به احرازهویت یکدیگر پرداخته و مذاکرات تعیین الگوریتم رمزنگاری، الگوریتم MAC و کلیدهای مورداستفاده برای حفاظت دیتای ارسال‌شده توسط SSL Record، را انجام دهند.

قبل از انتقال هرگونه داده‌های برنامه، از پروتکل Handshake استفاده می‌شود. این پروتکل شامل یک سری پیام بوده که بین کلاینت و سرور، تبادل می‌شود. تصویر زیر نمایش‌دهنده قالب پیام‌های پروتکل Handshake است.

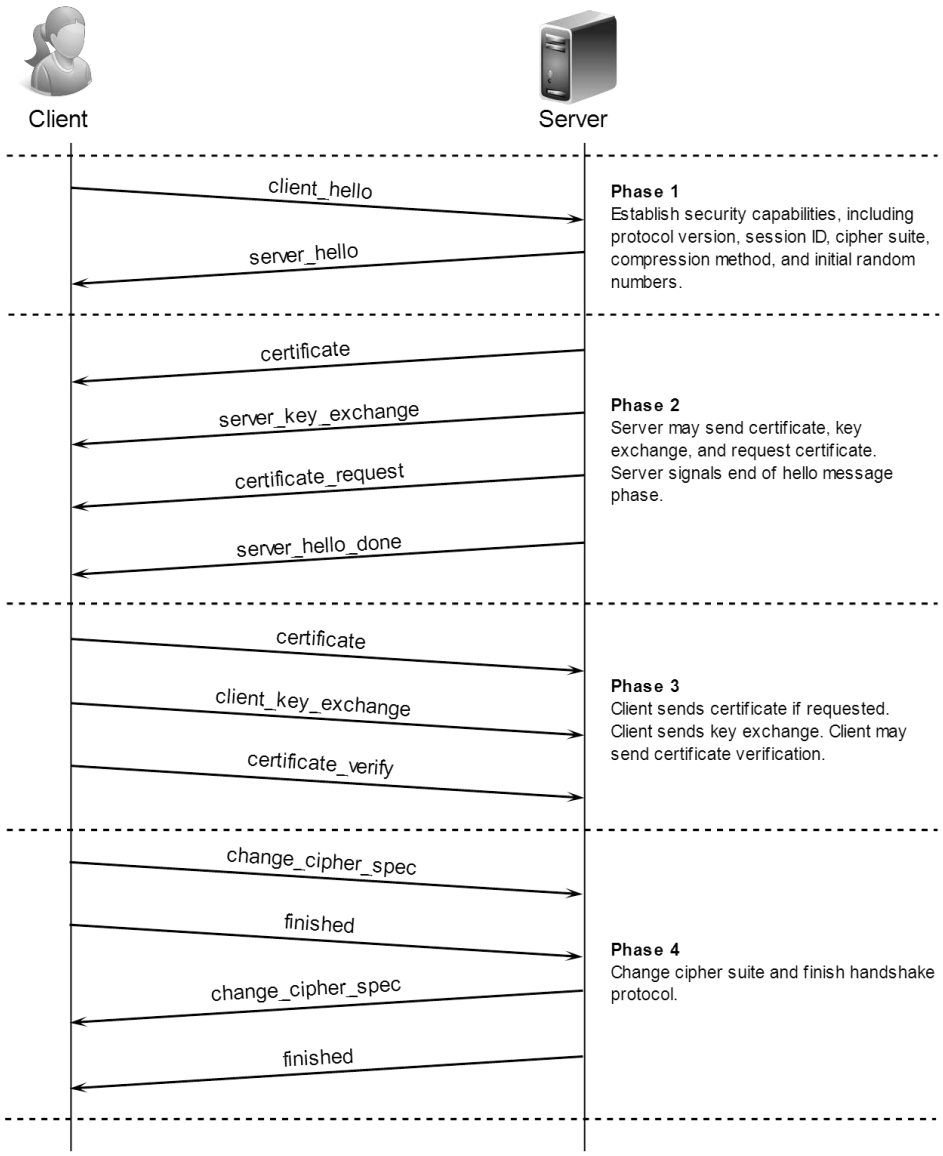


همان‌طور که در تصویر فوق ملاحظه می‌نمایید، هر پیام دارای سه فیلد، به شرح زیر است.

- **Type(1 byte)**
  مشخص‌کننده یکی از ۱۰ پیامی است که در جدول زیر نام‌برده شده است.
- **Length(3 bytes)**
  طول پیام به بایت.
- **Content(≥ 0 bytes)**
  پارامترهای مربوط به پیام. این پارامترها نیز در جدول زیر نام برده شده است.

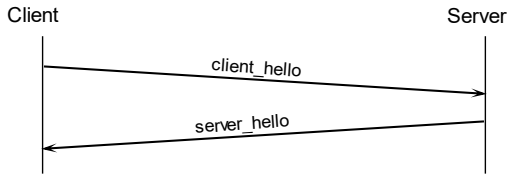
Message Type	Parameters
hello_request	null
client_hello	version, random, session id, cipher suite, compression method
server_hello	version, random, session id, cipher suite, compression method
certificate	chain of X.509v3 certificates
server_key_exchange	parameters, signature
certificate_request	type, authorities
server_done	null
certificate_verify	signature
client_key_exchange	parameters, signature
finished	hash value

تصویر زیر نمایش‌دهنده تبادل اولیه موردنیاز برای ایجاد یک ارتباط منطقی بین کلاینت و سرور است. این مراحل را می‌توان در قالب چهار فاز تقسیم‌بندی نمود. این چهار فاز در ادامه تشریح خواهند شد.



فاز اول پروتکل Handshake؛ برقراری قابلیت‌های امنیتی

در این فاز یک ارتباط منطقی شروع شده و قابلیت‌های امنیتی مربوط به آن، برقرار می‌شود. این تبادل، با ارسال پیام client_hello، توسط کلاینت شروع می‌شود.



پیام client_hello شامل پارامترهای زیر است:

- **Version**
بالاترین نسخه SSL که کلاینت توانایی استفاده از آن را دارد.
- **Random**
کلاینت اقدام به تولید یک مقدار تصادفی می‌نماید. این مقدار شامل یک مُهر زمانی ۳۲ بیتی و یک مقدار ۲۸ بایتی تصادفی امن، است.
این مقدار تصادفی به‌عنوان Nonce، جهت جلوگیری از حملات بازپخش در تبادل کلید مورداستفاده قرار می‌گیرد.
- **Session ID**
یک مقدار با طول متغیر است که جهت شناسه نشست استفاده می‌گردد.
مقدار 0، نشان‌دهنده این است که کلاینت مایل به ایجاد یک Connection جدید در یک Session جدید است.
مقدار غیر0، نشان‌دهنده این است که کلاینت قصد بروز رسانی پارامترهای همین Connection، و یا قصد ایجاد یک Connection جدید بر روی همین Session را دارد.
- **CipherSuite**
لیستی از الگوریتم‌های رمزنگاری است که کلاینت از آن‌ها پشتیبانی می‌کند؛ این لیست به‌صورت اولویت‌بندی شده ارائه می‌شود.
هر المان این لیست (هر مجموعه رمز) هم الگوریتم تبادل کلید و هم Cipherspec[1] را تعیین می‌کند.
- **Compression Method**
لیستی است شامل الگوریتم‌های فشرده‌سازی که توسط کلاینت پشتیبانی می‌گردد.

---

[1] Cipherspec به معنی مشخصات رمز است که در ادامه همین مبحث تشریح خواهد شد.

پس از ارسال پیام client_hello، کلاینت منتظر پیام server_hello از سمت سرویس‌دهنده می‌ماند. پیام server_hello نیز شامل همان پارامترهای client_hello است.

برای پارامترهای پیام server_hello، قرارداد زیر اعمال می‌شود:

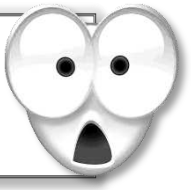
- فیلد Version شامل پایین‌ترین نسخه پیشنهادشده توسط کلاینت و بالاترین نسخه پشتیبانی شده توسط سرور است.
- مقدارRandom، توسط سرور و مستقل از مقدار Random کلاینت، ایجاد گردیده است.
- اگر مقدار Session ID پیام کلاینت غیر صفر باشد؛ همان مقدار توسط سرور نیز انتخاب می‌شود. و اگر مقدار Session ID پیام کلاینت صفر باشد؛ Session ID سرور حاوی مقداری جهت ایجاد یک Session جدید خواهد بود.
- فیلد CipherSuite، فقط شامل یک مجموعه رمز می‌باشد که از بین پیشنهادهای کلاینت، انتخاب گردیده است.

اولین المان در پارامترهای CipherSuite، روش تبادل کلید است. روش‌های تبادل کلید پشتیبانی شده عبارتند از:

i. RSA
ii. Fixed Diffie-Hellman
iii. Ephemeral Diffie-Hellman
iv. Anonymous Diffie-Hellman
v. Fortezza

روش تبادل کلید شامل فیلدهایی مثل الگوریتم رمزنگاری (DES، 3DES، IDEA و غیره)، نوع رمزنگاری (بلوکی یا نهری[1])، الگوریتم MAC (MD5 یا SHA-1)، سایز Hash و سایز IV است.

- فیلد Compression، شامل روش فشرده‌سازی است که از بین پیشنهادهای کلاینت انتخاب‌شده است.



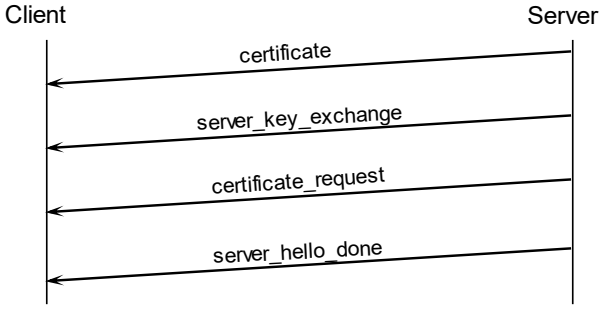
ازآنجاکه همیشه حق با مشتری است! ابتدا کلاینت توانایی‌های خود را در پشتیبانی از پروتکل‌های مختلف به اطلاع سرویس‌دهنده می‌رساند. سرور نیز باید با توجه به سیاست‌های امنیتی خود، از بین توانایی‌های کلاینت، موارد موردنظر را انتخاب کرده، تا بر اساس همان قابلیت‌ها، برقراری ارتباط بین کلاینت و سرور مقدور شود.

[1] Stream

فاز دوم پروتکل Handshake؛ احرازهویت سرور و تبادل کلید

این فاز را سرور با ارسال گواهینامه خود (درصورتی‌که برای احراز هویت نیاز باشد) آغاز می‌نماید. این پیام حاوی یک یا دسته‌ای گواهینامه X.509 است. پیام certificate برای تمام روش‌ها به‌استثنای روش anonymous Diffie-Hellman الزامی است.

توجه داشته باشید اگر روش fixed Diffie-Hellman مورداستفاده قرارگرفته باشد، پیام certificate، عملکردی به‌عنوان پیام server_key_exchange خواهد داشت؛ چراکه حاوی پارامترهای عمومی دیفی‌هلمن سرور است.



در گام بعدی ممکن است در صورت نیاز پیام server_key_exchange ارسال گردد. ارسال این پیام در دو مورد لازم نیست:

۱. سرور ارسال گواهینامه را از طریق پارامترهای fixed Diffie-Hellman انجام داده باشد.
۲. تبادل کلید RSA مورداستفاده قرارگرفته باشد.

البته ارسال پیام server_key_exchage در موارد زیر موردنیاز است:

- Anonymous Diffie-Hellman
- Ephemeral Diffie-Hellman
- server is using RSA but has a signature-only RSA key
- Fortezza

سومین پیام در فاز دوم، پیام certificate_request است. یک سرور غیرناشناس[1] (سروری که از Anonymous Diffie-Hellman استفاده نمی‌کند)، می‌تواند از کلاینت درخواست گواهینامه نماید.

پیام certificate_request شامل دو پارامتر زیر است:

- certificate_type
- certificate_authorities

[1] nonanonymous

پارامتر certificate_type نشان‌دهنده الگوریتم کلیدعمومی بوده و برای موارد زیر استفاده می‌شود:

- RSA, signature only
- DSS, signature only
- RSA for fixed Diffie-Hellman

(در این حالت از Signature فقط برای احرازهویت استفاده می‌شود.)

- DSS for fixed Diffie-Hellman

(در این حالت نیز از Signature فقط برای احرازهویت استفاده می‌شود.)

- RSA for ephemeral Diffie-Hellman
- DSS for ephemeral Diffie-Hellman
- Fortezza

دومین پارامتر certificate_authorities است. این پارامتر شامل لیستی از نام‌های برجسته از Certification Authority (CA)های قابل‌قبول است.

منظور از(Certificate Authority (CA ، مرجع صدور گواهینامه است. درصورتی‌که بخواهیم از گواهینامه در شبکه‌هایی مثل اینترنت استفاده نماییم، نیاز است تا یک شخص ثالث که برای طرفین ارتباط، مورد اعتماد و اطمینان است، اقدام به تأیید گواهینامه‌های دیجیتال نماید .
CA نیز به صدور گواهینامه و تأیید هویت دارنده آن می‌پردازد. در اینجا برای اینکه مراکز مورداطمینان مشخص شوند، پارامتر certificate_authorities لیستی از CAهای قابل‌اطمینان را ارائه می‌نماید.

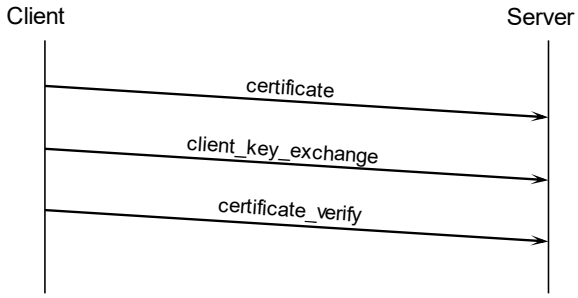


آخرین پیام فاز دو، که همیشه موردنیاز است، پیام server_done می‌باشد. سرور با ارسال این پیام، اعلام می‌کند که فرآیند server hello و پیام‌های مربوط به آن، به پایان رسیده است. این پیام پارامتری ندارد.

سرور پس از ارسال پیام server_done، منتظر جواب کلاینت می‌ماند.

فاز سوم پروتکل Handshake؛ احرازهویت کلاینت و تبادل کلید

پس از دریافت پیام server_done، کلاینت باید در صورت نیاز اقدام به اعتبارسنجی گواهینامه ارسالی از طرف سرور نموده؛ و قابل‌قبول پارامترهای server_hello را بررسی کند. اگر تمام نتایج بررسی‌های فوق رضایت‌بخش بود، کلاینت یک یا چند پیام را برای سرور ارسال می‌نماید.



اگر در فاز دوم، سرور از کلاینت درخواست گواهینامه کرده باشد؛ کلاینت فاز سوم را با ارسال پیام certificate، آغاز می‌کند. البته درصورتی‌که کلاینت گواهینامه مناسبی در اختیار نداشته باشد، به‌جای پیام certificate، هشدار no_certificate را ارسال خواهد نمود.

پیام بعدی که باید در این فاز فرستاده شود، پیام client_key_exchange است. محتوای این پیام با توجه به نوع تبادل کلید، به شرح زیر است:

- RSA
- Ephemeral or Anonymous Diffie-Hellman
- Fixed Diffie-Hellman
- Fortezza

نهایتاً در فاز سوم، کلاینت ممکن است اقدام به ارسال پیام certificate_verify نماید. این پیام اقدام به ارائه یک تأییدیه واضح از گواهینامه کلاینت می‌نماید.

پیام certificate_verify تنها به دنبال گواهینامه‌های کلاینت که قابلیت امضا داشته باشند، ارسال می‌گردد. به زبان ساده‌تر، منظور از تأییدیه واضح برای گواهینامه، امضایی است که کلاینت با استفاده از کلیدخصوصی خود بر روی مقدار Hash شده پیام، انجام داده باشد.

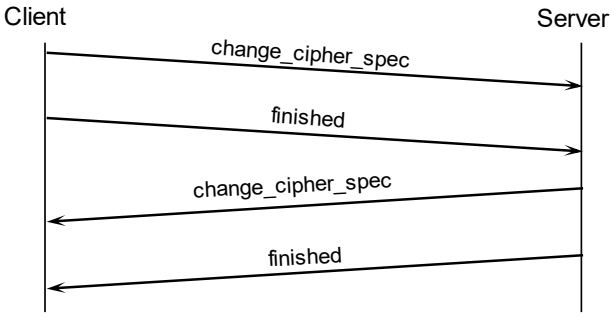
فاز چهارم پروتکل Handshake؛ پایان

این فاز، تکمیل‌کننده راه‌اندازی یک Connection امن است.

کلاینت یک پیام change_cipher_spec ارسال نموده و تنظیمات رمز معلق را در تنظیمات رمز جاری[1] کپی می‌نماید. البته توجه داشته باشید که این پیام به‌عنوان بخشی از پروتکل Handshake در نظر گرفته نمی‌شود، بلکه با استفاده از پروتکل Change Cipher Spec ارسال می‌گردد.

[1] Current CipherSpec

پس از پیام فوق، کلاینت باید سریعاً پیام finished را تحت الگوریتم‌ها و کلیدهای جدید، ارسال نماید. پیام finished تأیید می‌کند که تبادل کلید و فرآیند احرازهویت با موفقیت به انجام رسیده است.



در پاسخ پیام‌های فوق، سرور نیز عملیاتی شبیه به کلاینت، انجام می‌دهد. سرور یک پیام change_cipher_spec ارسال نموده و تنظیمات رمز معلق را در رمز جاری کپی می‌نماید. درنهایت نیز یک پیام finished برای کلاینت ارسال می‌کند.

پس از انجام موفقیت‌آمیز هر چهار فاز فوق، فرآیند Handshake کامل شده، و کلاینت و سرور می‌توانند شروع به تبادل دیتا در لایه کاربرد (Application) نمایند.

☑ مبحث دوم

پروتکل HTTPS

امروزه استفاده از وب، به بخش جدایی‌ناپذیر سرویس‌های فناوری اطلاعات مبدل گردیده است. نقشه راه اکثر شرکت‌های بزرگ و پیشرو در صنعت نرم‌افزار و سخت‌افزار، بر روی ارائه خدمات و کنسول مدیریتی از طریق وب، ترسیم گردیده است. فروشگاه‌های اینترنتی، خدمات بانکداری الکترونیک، اتوماسیون‌های اداری و بسیاری از سرویس‌های دیگر، مبتنی بر وب می‌باشند. دلیل استفاده گسترده و روزافزون از وب، کاهش وابستگی کلاینت به نصب و راه‌اندازی نرم‌افزارها و ابزارهای موردنیاز جهت استفاده از سرویس موردنظر است.

با افزایش کاربرد و استفاده از وب بر روی شبکه‌های داخلی و عمومی، امنیت وب نیز از اهمیت بیشتری برخوردار گردید. در بسیاری از موارد، ادامه حیات یک کسب‌وکار وابسته به سرویس‌های مبتنی بر وب گردیده است. اما متأسفانه سرویس وب به‌شدت آسیب‌پذیر نشان داده، لذا باید برای برقراری امنیت در وب، چاره‌ای اندیشیده شود.

امنیت می‌تواند در لایه‌های مختلف و با استفاده از ابزارهای امنیتی متفاوتی برقرار شود. اما بسیاری از ارائه‌دهندگان خدمات بر روی وب، مثل فروشگاه‌های اینترنتی یا بانکداری الکترونیکی، نمی‌توانند هیچ‌گونه مدیریت و سیاست امنیتی بر روی شبکه و سیستم کاربر اعمال نمایند؛ چراکه اساساً شبکه و سیستم کاربر به‌صورت مستقل اداره می‌گردد. از طرف دیگر، اگر کاربر را ملزم به استفاده از تجهیزات یا ابزارهای امنیتی مثل فایروال نمایند، ممکن است کاربر عطای استفاده از سرویس را به لقای هزینه آن بخشیده[۱] و از استفاده سرویس وب، صرف‌نظر کند.

با توجه به دلایل فوق، ارائه‌دهندگان سرویس‌های مبتنی بر وب، نیاز به برقراری امنیت به‌صورت مستقل از کاربر و شبکه مورداستفاده، دارند. لذا باید این امنیت را در لایه‌های بالاتر برقرار نمایند. همچنین برقراری امنیت وب، باید تا حد امکان نیاز به داشتن نرم‌افزار یا ابزار خاصی بر روی سیستم کاربر نداشته، و کاربر را درگیر اجرا و پیاده‌سازی فرآیندهای امنیتی (که به دلیل سختی، معمولاً کاربران عادی از آن اجتناب می‌ورزند) نکند.

برای حل این مشکل، پروتکل HTTPS که تلفیقی از پروتکل SSL\TLS و HTTP است، مورداستفاده قرار می‌گیرد. سازمان IETF پروتکل HTTPS را طی RFC 2818 ارائه نموده است.

---

[۱] عطایش را به لقایش بخشیدم؛ ضرب‌المثلی است در حکایت یازدهم از باب سوم گلستان سعدی.

در این مبحث، با توجه به گستردگی حملات و آسیب‌پذیری‌های وب، ابتدا به بررسی الزامات عمومی امنیت وب پرداخته و سپس پروتکل HTTPS را موردبحث قرار خواهیم داد.[1]

ملاحظات امنیتی وب

شبکه جهانی وب[2]، اساساً یک برنامه کلاینت/سروری است که بر اساس پروتکل TCP/IP بر روی شبکه اینترنت و اینترانت اجرا می‌گردد. از اغلب ابزارها و پروتکل‌های امنیتی که در این کتاب بحث شده، می‌توان در جهت برقراری امنیت وب استفاده نمود. اما وب دارای چالش‌های جدید و مختص به خود نیز است، که به‌صورت عمومی درزمینه کامپیوتر و شبکه‌های امنیتی وجود ندارد. برخی از این موارد، در زیر بیان‌شده است:

- اینترنت یک ارتباط دوطرفه است. برخلاف محیط‌های انتشار سنتی و حتی سیستم‌های نشر الکترونیک مثل پیام‌نما[3]، وب در مقابل حملات اینترنتی بر روی سرور، آسیب‌پذیر است.
- ادامه حیات کسب‌وکار سازمان‌ها به‌صورت قابل‌ملاحظه‌ای به وب وابسته گردیده است؛ لذا در صورت بروز خراب‌کاری بر روی سرور وب، سازمان متحمل ضررهای مالی و آسیب‌دیدگی حُسن شهرت، خواهد گردید.
- هرچند استفاده از مرورگرهای وب خیلی آسان است، سرورهای وب به‌راحتی مدیریت و پیکربندی می‌شوند، و محتوای وب به‌راحتی منتشر می‌شود؛ اما تمام این کارها تحت نرم‌افزارهای فوق‌العاده پیچیده انجام می‌پذیرد.
  این نرم‌افزارهای پیچیده ممکن است دارای نقص‌های امنیتی بالقوه فراوانی باشند. مثال‌های متعددی وجود دارد که حتی سیستم‌هایی که به‌درستی نصب‌شده‌اند، در مقابل طیف وسیعی از حملات امنیتی، آسیب‌پذیر هستند.
- وب سرور می‌تواند به‌عنوان مسیری جهت دسترسی به کل شبکه سازمان، مورد سوءاستفاده قرار گیرد. درصورتی‌که هکر بتواند به سرور وب نفوذ نماید، ممکن است قادر باشد از طریق آن به کل سیستم‌های شبکه داخلی که به سرور وب متصل می‌شوند، دسترسی پیدا کند.
- عموماً کلاینت‌های سرویس‌های مبتنی بر وب، کاربران موردی و آموزش ندیده در مسائل امنیتی، هستند. این کاربران لزوماً از خطرات و اقدامات متقابل آگاهی ندارند.

---

[1] Cryptography and Network Security Principles and Practice, William Stallings.
[2] World Wide Web (WWW)
[3] Teletext

تهدیدات امنیتی وب

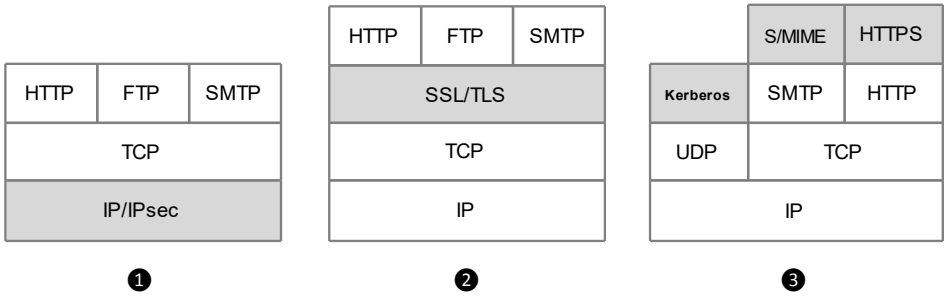
جدول زیر نمایش‌دهنده خلاصه‌ای از انواع تهدیداتی است که در هنگام استفاده از وب، ممکن است با آن‌ها مواجه شوید. یکی از روش‌های گروه‌بندی این حملات، بر اساس حملات فعال و غیرفعال است. روش دیگر گروه‌بندی حملات امنیتی وب، بر اساس محل وقوع حملات (سرور وب، مرورگر وب و ترافیک بین مرورگر و سرور) است.

	تهدید	عواقب	اقدام متقابل
صحت	* تغییر اطلاعات کاربر * تروجان مرورگر * تغییر حافظه * تغییر پیام‌های در حال تبادل	* از دست دادن اطلاعات * آسیب‌پذیری در مقابل تمام حملات دیگر * در معرض خطر قرار گرفتن ماشین	Checksum یا چکیده پیام رمزشده
محرمانگی	* استراق سمع در شبکه * سرقت اطلاعات از سرور * سرقت اطلاعات از کلاینت * جمع‌آوری اطلاعات درباره پیکربندی شبکه * جمع‌آوری اطلاعات درباره اینکه کدام کلاینت با سرور در حال مذاکره است	* از دست دادن اطلاعات * از دست دادن حریم خصوصی	* رمزنگاری * پروکسی وب
منع خدمت	* از بین بردن مسیر ارتباطی کاربر * سیل‌زده کردن ماشین با ارسال درخواست‌های جعلی * پر کردن دیسک و حافظه * جداسازی ماشین با حملات DNS	* ایجاد اختلال * آزاردهنده و اذیت کننده * جلوگیری از انجام شدن خواسته کاربر	جلوگیری از آن سخت است.
احرازهویت	* جعل هویت * جعل اطلاعات	* ارائه اطلاعات نادرست از کاربر * باور اطلاعات نامعتبر	روش‌های رمزنگاری

روش‌های امنیت ترافیک وب

روش‌های مختلفی برای تأمین امنیت وب وجود دارد؛ این روش‌ها از مکانیسم‌های تقریباً شبیه به هم استفاده نموده و درنهایت نیز سرویس‌های امنیتی مشابهی را فراهم می‌نمایند. اما آن‌ها در دامنه کاربرد و محل قرارگیری در لایه‌های TCP/IP، با یکدیگر متفاوت هستند.

تصویر زیر نمایش‌دهنده، سه روش برقراری امنیت برای ترافیک وب است. درروش اول از پروتکل IPsec استفاده‌شده است. مزیت استفاده از IPsec، شفاف بودن آن از دید کاربر نهایی و برنامه کاربردی است. همچنین IPsec دامنه کاربرد وسیعی دارد و یک‌راه حل امنیتی عمومی محسوب می‌شود. علاوه بر این، IPsec دارای قابلیت فیلترینگ است؛ در این صورت فقط ترافیک موردنظر متحمل سربار پردازشی IPsec خواهد شد.



راه‌حل دوم نیز، تقریباً همه‌منظوره است؛ چراکه امنیت را در بالای TCP، پیاده‌سازی می‌نماید. به‌عنوان‌مثال برای برقراری این روش، می‌توان از پروتکل SSL/TLS بهره‌برداری نمود.

در این روش دو انتخاب برای نحوه پیاده‌سازی وجود دارد:

۱. برای عمومی بودن، می‌توان SSL/TLS را به‌عنوان بخشی از پروتکل لایه پایین‌تر، پیاده‌سازی نمود. در این صورت از دید برنامه کاربردی شفاف خواهد بود.
۲. درروش دیگر، SSL می‌تواند دربسته‌های خاص تعبیه‌شده باشد. به‌عنوان‌مثال، مرورگرهایی مثل Netscape و IE، مجهز به SSL شده و در اکثر سرورها نیز، این پروتکل پیاده‌سازی شده است.

درراه حل سوم، سرویس‌های امنیتی خاص، درون همان برنامه خاص، تعبیه می‌شوند. مزیت این روش در آن است که سرویس می‌تواند مناسب با نیازهای خاص یک برنامه کاربردی، ارائه گردد. همچنین برقراری امنیت مستقل از سایر بخش‌ها و در اختیار برنامه‌نویس خواهد بود.

پروتکل HTTPS

پروتکل HTTPS یا HTTP over SSL، پروتکلی است که از ترکیب پروتکل‌های HTTP با SSL، برای برقراری امنیت ارتباط بین مرورگر وب و سرویس‌دهنده وب، به وجود آمده است. هرچند قابلیت استفاده از HTTPS، در تمامی مرورگرهای مدرن امروزی، وجود دارد؛ اما استفاده از آن بستگی به پشتیبانی سرور از این پروتکل نیز دارد. به‌عنوان‌مثال اکثر موتورهای جستجو، از HTTPS پشتیبانی نمی‌کنند.

زمان استفاده از HTTPS، تنها تفاوتی که کاربر در مرورگر ملاحظه می‌کند، استفاده از https:// به‌جای http:// در اول آدرس URL موردنظر است. اتصال HTTP بر روی پورت TCP 80 و اتصال HTTPS بر روی پورت TCP 443 برقرار می‌گردد.

زمانی که ازHTTPS استفاده می‌شود، المان‌های ارتباطی زیر به‌صورت رمزشده خواهند بود:

- URL سند[1] درخواستی
- محتوای سند
- محتوای فرم‌های مرورگر (که توسط کاربر پر شده است.)
- کوکی‌های[2] تبادل شده بین کلاینت و سرور
- محتوای سرآیند HTTP

سازمان IETF، استاندارد HTTP over TLS را طی RFC 2818 ارائه نموده است. توجه داشته باشید که هیچ تغییر اساسی در استفاده HTTP از پروتکل‌های TLS یا SSL وجود نداشته و در صورت استفاده از هرکدام از پروتکل‌های فوق برای برقراری امنیت، درنهایت پروتکل امن شده را HTTPS می‌نامند.

شروع اتصال

برای HTTPS، کلاینت هم باید به‌عنوان کلاینت HTTP و هم به‌عنوان کلاینت TLS عمل نماید. کلاینت، اتصال به سرور را بر روی پورت مناسب شروع کرده و سپس پیام TLS Client_hello را برای شروع فرآیند Handshake، ارسال می‌نماید. پس از پایان Handshake، کلاینت می‌تواند شروع به ارسال اولین درخواست HTTP خود نماید. تمام دیتای HTTP نیز به‌عنوان دیتای برنامه TLS ارسال می‌شود. البته برای حفظ اتصال‌ها، باید رفتار معمولی HTTP دنبال شود.

---

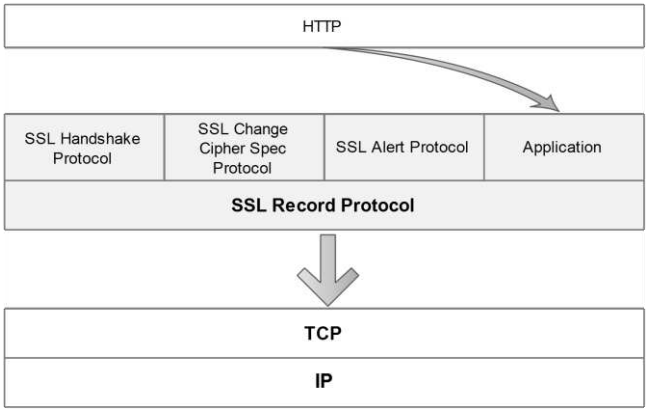
[1] Document

[2] کوکی‌ها اقدام به ذخیره‌سازی اطلاعات خاصی از کاربر یا سرور نموده، و در مراجعات بعدی مورداستفاده قرار می‌دهند.

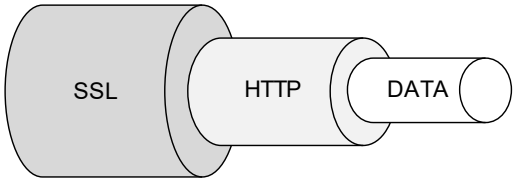
اتصال HTTPS در سه سطح صورت می‌گیرد:

۱. در سطح HTTP، کلاینت با ارسال یک Connection request به لایه پایین‌تر، از سرور HTTP درخواست یک اتصال (Connection) می‌نماید. به‌طورمعمول، لایه پایین‌تر، TCP است؛ البته ممکن است SSL یا TLS نیز باشد.[۱]
۲. درصورتی‌که لایه پایین‌تر، پروتکل SSL/TLS باشد، در سطح TLS یک نشست (Session) بین سرور و کلاینت برقرار می‌شود. این نشست می‌تواند یک یا چند اتصال را در هر زمان، پشتیبانی نماید.
۳. همان‌طور که قبلاً توضیح داده شد، شروع یک درخواست اتصال در TLS، با برقراری یک ارتباط TCP بین کلاینت و سرور خواهد بود.

تصویر زیر نمایش‌دهنده فرآیند SSL/TLS است:



این تصویر نیز می‌تواند در فهم بهتر HTTPS، مؤثر باشد.



[۱] همان‌طور که قبلاً گفته شد، درصورتی‌که از پروتکل SSL/TLS استفاده نماییم، این پروتکل در بالای لایه Transport که حاوی پروتکل TCP است، قرار می‌گیرد.

بستن اتصال

سرور یا کلاینت HTTP می‌توانند با ارسال connection : close در رکورد HTTP، نشان دهند که قصد بستن یک Connection رادارند. به این معنا که پس از تحویل این رکورد، اتصال بسته خواهد شد.

بسته شدن HTTPS، مستلزم بسته شدن اتصال TLS با طرف مقابل است؛ و برای بستن اتصال TLS، باید اتصال TCP نیز بسته شود. در سطح TLS، برای بستن یک اتصال، از پروتکل Alert جهت ارسال هشدار close_notify استفاده می‌شود. قبل از بستن یک اتصال در TLS، حتماً هشدارهای مربوطه باید تبادل گردیده باشند.

ممکن است یک TLS، پس از ارسال هشدار بستن، منتظر جواب طرف مقابل نمانده و اقدام به بستن اتصال نماید. در این صورت یک "بستن ناقص"[1] اتفاق افتاده است؛ لذا ممکن است برای استفاده مجدد جهت یک اتصال انتخاب شود.

کلاینت HTTP باید توانایی اداره وضعیتی که در آن اتصال‌های TCP، بدون هشدار close_notify و بدون ارسال connection : close، بسته‌شده‌اند، را داشته باشد. بروز چنین وضعیتی می‌تواند بر اساس خطاهای برنامه‌نویسی روی سرور و یا خطا در ارتباطات، باشد. البته بسته شدن اعلام‌نشده یک اتصال TCP، می‌تواند شواهد بروز نوعی حملات نیز باشد. بنابراین کلاینت HTTPS در صورت بروز این وضعیت، باید هشدارهای امنیتی صادر نماید.

[1] Incomplete Close

فصل یازدهم

امنیت پست الکترونیک

مبحث اول

پروتکل PGP

مبحث دوم

پروتکل S/MIME

پست‌الکترونیک یکی از پرکاربردترین سرویس‌هایی است که امروزه توسط کاربران مورداستفاده قرار می‌گیرد. پست‌الکترونیک نه‌تنها بر روی اینترنت، بلکه در شبکه‌های داخلی نیز کاربرد فراوانی دارد. حتی برخی از مؤسسات از پست‌الکترونیک خدماتی شبیه اتوماسیون اداری دریافت می‌کنند! و یا اتوماسیون‌ها و روال‌های کاری در سازمان‌ها به پست الکترونیک وابسته شده است.

اهمیت و کاربرد پست الکترونیک، در حدی است که بسیاری از مکاتبات محرمانه سازمان‌ها نیز از این طریق انجام می‌پذیرد. فاکتورهای فروش، رسیدهای پرداخت، مکاتبات و دستورات محرمانه اداری، بخشنامه‌های اداری، درخواست‌ها و بسیاری از مضامین پراهمیت دیگر، از طریق پست الکترونیک ارسال و دریافت می‌گردد.

پست‌الکترونیک از پروتکل Simple Mail Transfer Protocol(SMTP) برای تبادل نامه‌های الکترونیکی استفاده می‌کند. متأسفانه SMTP هیچ‌گونه امنیتی را برای محتوای نامه الکترونیک فراهم نکرده و همچنان که از اسم آن نیز مشخص است، صرفاً فقط وظیفه تبادل ساده پیام‌ها را بر عهده دارد.

به دلیل کمبودها و ضعف‌های موجود در SMTP، سازمان IETF اقدام به ارائه یک پروتکل جدید بانام MIME برای پست الکترونیک نمود. هرچند که این پروتکل ازنظر ویژگی‌های پستی بهتر از SMTP است، اما بازهم امنیت قابل‌توجهی بر روی آن وجود ندارد.

با توجه به گسترش روزافزون استفاده از پست‌الکترونیک در تمام ابعاد کسب‌وکار سازمان‌ها، باید برای احرازهویت و محرمانگی نامه‌های الکترونیک، تمهیداتی اندیشه شود. برای این منظور دو پروتکل PGP و S/MIME برای برقراری امنیت در پست الکترونیک ارائه شدند.

پروتکل PGP برای برقراری امنیت در پست الکترونیک مبتنی بر SMTP و پروتکل S/MIME برای برقراری امنیت پست الکترونیک مبتنی بر MIME توسعه داده شده‌اند.

در مبحث اول این فصل به بررسی پروتکل PGP و نحوه عملکرد آن می‌پردازیم. با توجه به اینکه پروتکل S/MIME، نسخه استانداردشده پروتکل PGP بوده و ازنظر عملکرد، شبیه آن است؛ لذا در مبحث دوم نیز به بررسی پروتکل S/MIME می‌پردازیم.

☑ مبحث اول

پروتکل PGP

پروتکل Pretty Good Privacy (PGP)، یک پروتکل متن‌باز[۱] است که توسط آقای Phil Zimmermann، جهت امنیت پست الکترونیک ارائه گردیده است. البته پروتکل PGP در برقراری امنیت موارد دیگری مثل فایل‌ها و دیتای ذخیره‌شده نیز می‌تواند مورداستفاده قرار گیرد؛ اما شهرت آن به دلیل کاربرد این پروتکل در پست‌الکترونیک است.

پروتکلPGP فراهم آورنده سرویس‌های امنیتی احرازهویت و محرمانگی برای پست الکترونیک و ذخیره‌سازی فایل است. این پروتکل توسط IETF و طی RFC 4880 نیز منتشر گردیده است.

در اصل می‌توان گفت آقای زیمرمن برای به وجود آوردن پروتکل PGP، کارهای زیر را انجام داده است:

۱. انتخاب بهترین الگوریتم رمزنگاری موجود، برای ساخت بلوک‌ها دیتا.
۲. یکپارچه‌سازی الگوریتم انتخاب‌شده در یک برنامه کاربردی همه‌منظوره[۲]، که مستقل از سیستم‌عامل و پردازشگر است.
۳. ایجاد یک Package و مستندسازی آن شامل : کد منبع، در دسترس عموم قرار دادن آن از طریق اینترنت، درج آن در اعلانات و شبکه‌های تجاری مثل AOL[۳].
۴. دستیابی به توافق با یک شرکت، جهت فراهم آوردن یک نسخه تجاری کم‌هزینه و کاملاً سازگار از PGP.

رشد PGP به‌صورت انفجارگونه بوده و در حال حاضر به‌طور گسترده‌ای مورداستفاده قرار می‌گیرد. موارد زیر برخی از دلایلی است که می‌توان برای این رشد سریع عنوان کرد:

۱. به‌صورت رایگان در سراسر جهان در دسترس بوده و امکان اجرا بر روی انواع سیستم‌عامل‌ها ازجمله ویندوز، یونیکس و مکینتاش را دارد. علاوه بر این، نسخه تجاری برای کاربرانی که نیاز به پشتیبانی فروشنده دارند، نیز فراهم گردیده است.

---

[۱] منظور از Open Source یا متن‌باز، برنامه‌ای است که کد آن به‌صورت کامل در اختیار همگان قرارگرفته و استفاده از آن بدون پرداخت هزینه امکان‌پذیر است.

[2] General-purpose
[3] America On Line

۲. بر اساس الگوریتم‌هایی ایجادشده، که امنیت آن‌ها به‌صورت گسترده مورد تائید قرارگرفته است. به‌صورت خاص، این Package شامل RSA، DSS و دیفی‌هلمن برای رمزنگاری کلیدعمومی؛ و الگوریتم‌های CAST-128، IDEA و 3DES برای رمزنگاری متقارن؛ و الگوریتم SHA برای عملیات درهم‌سازی، است.
۳. کاربرد وسیعی برای شرکت‌هایی که مایل به انتخاب و اجرای یک طرح استاندارد رمزنگاری جهت فایل‌ها و پیام‌ها برای افرادی که می‌خواهند با آن‌ها از طریق اینترنت یا شبکه‌های دیگر ارتباط امن داشته باشند، دارد.
۴. این پروتکل توسط یک دولت یا سازمان استاندارد، توسعه‌نیافته و کنترل نمی‌شود؛ لذا برای کسانی که نسبت به دولت‌ها و سازمان‌ها بی‌اعتماد هستند، استفاده از PGP جذاب خواهد بود.
۵. در حال حاضر PGP از طریق استانداردهای اینترنت دنبال می‌شود[1]؛ اما بااین‌وجود، PGP همچنان غیر وابسته به هر سازمان یا دولتی می‌تواند مورداستفاده قرار گیرد.

نشانه‌گذاری

در کتاب آقای استالینگز[2]، برای تشریح PGP از نمادها و نشانه‌هایی استفاده‌شده است. البته ممکن است بعضی از این نشانه‌ها و نمادها در استانداردها یا مراجع دیگر، به‌صورت متفاوتی مورداستفاده قرارگرفته باشد. لذا قبل از شروع بررسی PGP، به تشریح نشانه‌های مورداستفاده می‌پردازیم:

K_S = کلید نشست که توسط رمزنگاری متقارن مورداستفاده قرار می‌گیرد.
PR_a = کلیدخصوصی طرف A، که توسط رمزنگاری کلیدعمومی (نامتقارن) مورداستفاده قرار می‌گیرد.
PU_a = کلیدعمومی طرف A، که توسط رمزنگاری کلیدعمومی مورداستفاده قرار می‌گیرد.
EP = عملیات رمزگذاری(Encryption) در الگوریتم رمزنگاری کلیدعمومی.
DP = عملیات رمزگشایی(Dycryption) در الگوریتم رمزنگاری کلیدعمومی.
EC = رمزگذاری در الگوریتم رمزنگاری متقارن.
DC = رمزگشایی در الگوریتم رمزنگاری متقارن.
H = تابع درهم‌سازی

[1] RFC 3156; MIME Security with OpenPGP
[2] Cryptography and Network Security Principles and Practice

|| = الصاق یا Concatenation
Z = فشرده‌سازی با استفاده از الگوریتم ZIP.
R64 = تبدیل به فرمت ASCII 64.

تشریح عملیات PGP

پروتکل PGP، چهار سرویس زیر را فراهم می‌آورد:

- احرازهویت (Authentication)
- محرمانگی (Confidentiality)
- فشرده‌سازی (Compression)
- سازگاری با پست الکترونیک (E-mail compatibility)

در جدول زیر، توضیح مختصری درباره هر چهار مورد فوق آورده شده است. البته در ادامه این مبحث، این موارد به‌صورت کامل نیز تشریح خواهند شد.

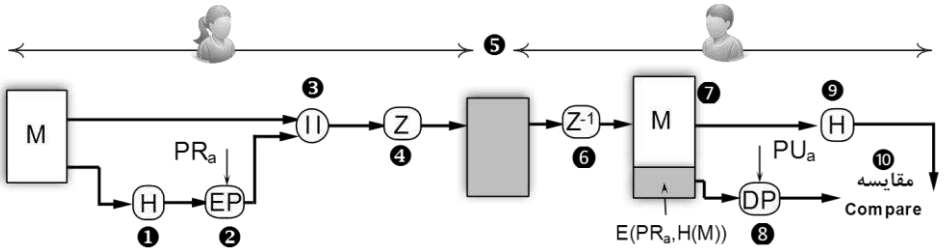
عملیات	الگوریتم مورداستفاده	توضیحات
امضای دیجیتال (احرازهویت)	DSS/SHA RSA/SHA	الگوریتم SHA چکیده پیام را تولید نموده و سپس چکیده توسط DSS یا RSA با کلیدخصوصی فرستنده امضا می‌شود.
رمزنگاری پیام (محرمانگی)	CAST or IDEA or 3DES 3DES با دیفی‌هلمن یا RSA	پیام توسط الگوریتم‌های CAST-128، IDEA یا 3DES رمزگذاری می‌شود. این رمزگذاری با استفاده از کلید نشست (Session key) یک‌بارمصرف تولیدشده توسط فرستنده انجام می‌شود. کلید نشست تولیدشده با استفاده از RSA یا دیفی‌هلمن و بر اساس کلیدعمومی گیرنده، تبادل می‌گردد.
فشرده‌سازی	ZIP	می‌توان پیام را قبل از ذخیره یا تبادل، توسط الگوریتم ZIP، فشرده‌سازی نمود.
سازگاری با پست الکترونیک	Radix-64 conversion	جهت فراهم آوردن شفافیت برای برنامه‌های پست الکترونیک، می‌توان پیام رمزشده را به کد ASCII تبدیل کرد.



پروتکل SMTP قبل از تبادل نامه‌های الکترونیک، متن آن را به کد اسکی تبدیل می‌نماید. پروتکل PGP نیز برای اینکه ازنظر SMTP شفاف باشد، اقدام به تبدیل متن رمزشده به کد اسکی نموده و پیام را در قالب کد اسکی به پروتکل SMTP تحویل می‌نماید.

احرازهویت در PGP

تصویر زیر نمایش‌دهنده مراحل انجام امضای دیجیتال برای فراهم کردن سرویس احرازهویت در PGP است. این مراحل خیلی شبیه به امضای دیجیتال است که قبلاً در بخش رمزنگاری توضیح داده شد. این حالت فقط فراهم آورنده سرویس احرازهویت است.



فرستنده از پیام چکیده می‌گیرد.❶ الگوریتم مورداستفاده برای تابع درهم‌ساز، SHA-1 بوده و یک خروجی ۱۶۰ بیتی تولید می‌نماید. سپس چکیده پیام توسط الگوریتم رمزنگاری نامتقارن که در اینجا الگوریتم RSA است، بر اساس کلیدخصوصی فرستنده رمزگذاری می‌شود.❷ چکیده رمزشده، به متن اصلی پیام الصاق می‌شود.❸ متن اصلی و چکیده رمزشده توسط الگوریتم فشرده‌سازی، فشرده می‌گردند.❹ الگوریتم فشرده‌سازی مورداستفاده، الگوریتم ZIP است. نهایتاً، فرستنده متن فشرده به‌دست‌آمده❺را برای گیرنده ارسال می‌نماید.

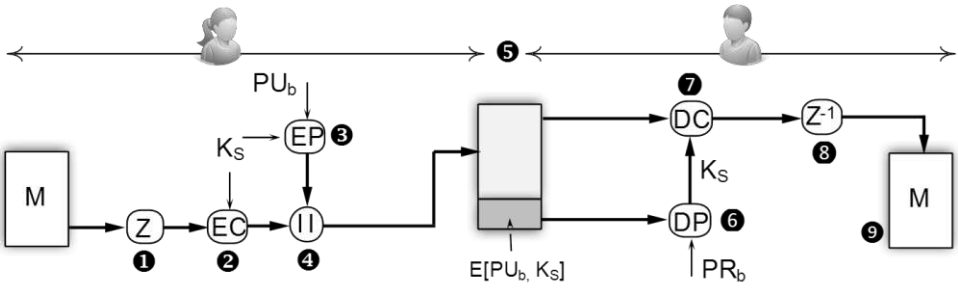
گیرنده باید در گام اول اقدام به غیر فشرده‌سازی پیام دریافتی نماید.❻ پس‌ازآن که پیام دریافتی از حالت فشرده خارج گردید، گیرنده یک پیام به همراه چکیده رمزشده در اختیار خواهد داشت.❼ با توجه به اینکه چکیده با کلیدخصوصی فرستنده رمز شده است، تنها کلیدی که می‌توان برای رمزگشایی آن استفاده کرد، کلیدعمومی فرستنده خواهد بود. لذا گیرنده با کلیدعمومی فرستنده چکیده را رمزگشایی می‌کند.❽ سپس از پیام Hash گرفته❾و آن را با Hash دریافتی مقایسه می‌نماید.❿ درصورتی‌که مقادیر Hash با یکدیگر یکسان باشند، گیرنده می‌تواند از صحت پیام و فرستنده اطمینان حاصل نماید.

محرمانگی در PGP

یکی دیگر از خدمات ارائه‌شده توسط PGP، محرمانگی است. پروتکل PGP محرمانگی را با استفاده از الگوریتم‌های رمزنگاری متقارن مانند CAST-128، IDEA و 3DES فراهم می‌آورد. فارغ از الگوریتم رمزنگاری مورداستفاده، PGP با استفاده از روش CFB[1] اقدام به رمزنگاری می‌نماید.

اما هر زمان که بحث رمزنگاری پیش می‌آید، مشکل تبادل کلید خود را نمایش می‌دهد. در پروتکل PGP نیز برای تبادل کلید مورداستفاده، نیاز به یک مکانیسم امن داریم. در PGP هر کلید رمز، فقط یک‌بار مورداستفاده قرار می‌گیرد. برای هر پیام یک کلید ۱۲۸ بیتی تصادفی تولید می‌شود؛ که آن را کلید نشست یا Session Key می‌نامند. به دلیل یک‌بارمصرف بودن کلید، فقط محدود به همان پیام بوده و همراه پیام منتقل می‌شود. برای حفاظت از کلید، آن را با کلیدعمومی گیرنده رمز نموده و ارسال می‌کنند.

تصویر زیر نحوه برقراری محرمانگی را نمایش می‌دهد. همان‌طور که ملاحظه می‌کنید، این روش نیز بسیار شبیه روش‌های توضیح داده‌شده در بخش رمزنگاری است.



فرستنده (آلیس) ابتدا پیام را فشرده‌سازی می‌نماید.❶ همان‌طور که در فصل‌های قبلی کتاب گفته شد، اگر بخواهیم از رمزنگاری و فشرده‌سازی در کنار هم استفاده نماییم، ابتدا باید فشرده‌سازی را انجام داده و سپس رمزنگاری اعمال شود. پیام فشرده‌شده بر اساس کلید نشست یا Session key و با استفاده از الگوریتم رمزنگاری متقارن مثل CAST-128، رمزگذاری می‌شود.❷ کلید نشست یک مقدار ۱۲۸ بیتی است که به‌صورت تصادفی توسط فرستنده تولید می‌گردد. این کلید یک‌بارمصرف بوده و فقط برای همان پیام استفاده می‌شود. برای اینکه بتوان کلید تولیدشده را به‌صورت امن به دست گیرنده (باب) رساند، آن را با کلید عمومی گیرنده رمزگذاری نموده❸و به انتهای پیام رمزشده، الصاق❹ و برای گیرنده ارسال می‌شود.❺

[1] روش Cipher Feedback(CFB) در بخش رمزنگاری توضیح داده‌شده است.

گیرنده یک پیام رمزشده با الگوریتم متقارن و یک کلید رمزشده با کلیدعمومی خود، دریافت می‌نماید. ❺ گیرنده باید در گام اول اقدام به رمزگشایی کلید بر اساس کلیدخصوصی خود نماید. ❻ به دلیل اینکه کلید نشست با کلیدعمومی گیرنده رمزشده، تنها کسی که قادر به رمزگشایی آن است، همان گیرنده یا باب خواهد بود. پس‌ازآن که گیرنده کلید نشست را به دست آورد، با استفاده از الگوریتم رمزنگاری متقارن و کلید نشست، اقدام به رمزگشایی پیام می‌نماید. ❼ متن حاصل از این رمزگشایی یک متن فشرده است که باید توسط الگوریتم ZIP، از حالت فشرده خارج شود. ❽ درنهایت پیام به‌صورت متن معمولی در اختیار گیرنده قرار خواهد گرفت. ❾

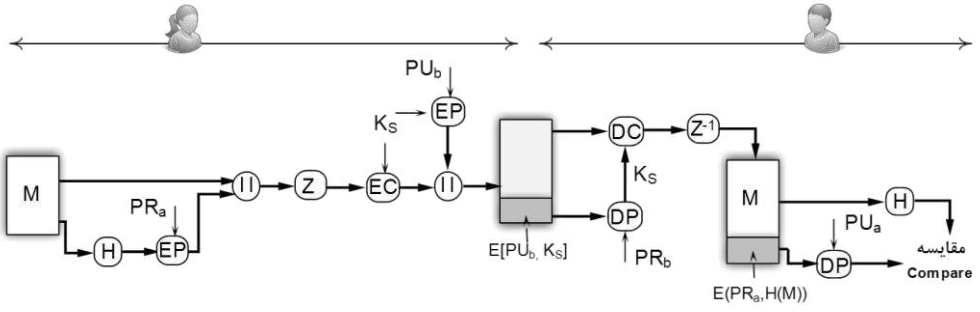
این روش فقط محرمانگی را فراهم آورده و در آن از احرازهویت خبری نیست؛ چراکه فقط از کلیدعمومی گیرنده استفاده‌شده است. به همین دلیل سرویس محرمانگی فراهم گردیده؛ اما با توجه به اینکه هرکسی می‌تواند از کلیدعمومی گیرنده برای رمزگذاری استفاده نماید، لذا احرازهویت فراهم نخواهد شد.

البته PGP امکان استفاده از RSA یا دیفی‌هلمن را جهت تبادل امن کلید، دارد. به‌جای روش فوق می‌توان از RSA یا دیفی‌هلمن برای توافق بر روی کلید موردنظر جهت کلید نشست، بهره برد. نحوه عملکرد RSA و دیفی‌هلمن، قبلاً در بخش رمزنگاری تشریح شده است.

احراز هویت به همراه محرمانگی در PGP

همان‌طور که ملاحظه کردید، در هر یک از حالات فوق، احرازهویت و محرمانگی به‌تنهایی فراهم گردیده است. اما می‌توان با ترکیب هر دو حالت، محرمانگی و احرازهویت را هم‌زمان در اختیار داشت. برای این کار باید از کلیدخصوصی فرستنده برای تولید امضای دیجیتال، و از کلیدعمومی گیرنده برای رمزگذاری متن و کلید نشست بهره برد.

تصویر زیر نمایش‌دهنده احرازهویت به همراه محرمانگی است.



فشرده‌سازی در PGP

به‌صورت پیش‌فرض، PGP پس از امضا و قبل از رمزگذاری، اقدام به فشرده‌سازی پیام می‌نماید. این فشرده‌سازی باعث صرفه‌جویی در استفاده از فضا، چه در ذخیره و چه در ارسال پیام، خواهد شد.

به دو دلیل زیر، امضا قبل از فشرده‌سازی انجام می‌شود:

۱. برای بررسی‌های بعدی، بهتر است امضا قبل از فشرده‌سازی انجام‌گرفته و همراه متن ساده ذخیره شود. اگر برای یک پیام فشرده، امضا تولید شود؛ هر بار که نیاز به بررسی امضا باشد، باید پیام مجدد فشرده‌شده و از آن چکیده گرفته شود، تا امکان مقایسه آن با امضای انجام‌شده بر روی پیام فشرده، وجود داشته باشد.

۲. درPGP امکان استفاده از الگوریتم‌های مختلف فشرده‌سازی وجود دارد. این الگوریتم‌ها دارای سرعت‌های مختلفی در اجرا و فشرده‌سازی بوده و خروجی‌های متفاوتی را تولید می‌نمایند. به‌هرحال اکثر این الگوریتم‌ها با یکدیگر سازگار بوده و می‌توانند خروجی یک نسخه دیگر را از حالت فشرده خارج نمایند. اما اگر چکیده و امضا پس از فشرده‌سازی انجام شود، آن‌وقت PGP به استفاده از همان الگوریتم فشرده‌سازی محدودشده و در هر دو طرف گیرنده و فرستنده، باید از یک الگوریتم جهت فشرده‌سازی و غیر فشرده‌سازی استفاده نماید.

همچنین به دلیل زیر فشرده‌سازی قبل از رمزگذاری، انجام می‌شود:

- رمزگذاری پس از فشرده‌سازی، باعث تقویت الگوریتم رمزنگاری می‌شود. به دلیل اینکه پیام فشرده‌شده دارای افزونگی‌های کمتری نسبت به متن اصلی است، رمزیابی مشکل‌تر می‌شود.

سازگاری با پست الکترونیک

در زمان استفاده از PGP، حداقل یک قسمتی از بلوک دیتای منتقل‌شده، رمزگذاری شده خواهد بود. شاید برای سرویس امضای دیجیتال، فقط چکیده پیام با کلید خصوصی فرستنده رمزشده باشد؛ یا شاید برای فراهم آوردن سرویس محرمانگی، پیام به همراه امضا (در صورت وجود) با استفاده از کلید متقارن یک‌بارمصرف، رمزگذاری شده باشد. بنابراین تمام یا قسمتی از بلوک دیتا، شامل جریانی از Octetهای ۸ بیتی خواهد بود. بااین‌حال بسیاری از سیستم‌های پست الکترونیک فقط اجازه استفاده از بلوک‌های متشکل از متن اسکی را می‌دهند. لذا برای تطبیق با این محدودیت، PGP سرویسی را برای تبدیل جریان باینری ۸ بیتی به جریان کاراکترهای اسکی،

فراهم آورده است. در این طرح هر گروه متشکل از سه Octet دیتای باینری، به چهار کاراکتر اسکی تبدیل می‌شوند. همچنین در این حالت، CRC نیز جهت تشخیص اشتباه در انتقال، اضافه می‌شود.

استفاده از Radix-64 باعث افزایش ۳۳ درصدی پیام می‌شود. اما خوشبختانه بخش کلیدنشست و امضای پیام نسبتاً جمع‌وجور بوده و متن اصلی پیام نیز فشرده‌شده است. درواقع فشرده‌سازی باید به‌اندازه‌ای باشد که این افزایش ۳۳ درصدی را جبران نماید.

یکی از جنبه‌های قابل‌توجه Radix-64، تبدیل کورکورانه جریان ورودی به اسکی است؛ یعنی حتی اگر ورودی، متن اسکی هم باشد، بازهم تبدیل انجام می‌شود. بنابراین اگر یک پیام امضاشده باشد ولی رمزگذاری نشده باشد، بازهم تبدیل بر کل بلوک اعمال شده و یک خروجی اسکی حاصل خواهد شد. لذا خروجی برای افراد دیگر، قابل‌خواندن نخواهد بود؛ به همین دلیل سطحی از محرمانگی (هرچند کم)، فراهم می‌گردد.

به‌عنوان یک گزینه، PGP می‌تواند طوری پیکربندی شود که تبدیل Radix-64 را فقط بر روی قسمت امضا، اعمال نماید. این کار، دریافت‌کننده را قادر می‌سازد تا بتواند بدون استفاده از PGP پیام را بخواند. اما اگر گیرنده بخواهد بررسی امضا را انجام دهد، باید از PGP استفاده کند.

کلیدهای رمزنگاری و دسته‌کلیدها

PGP از چهار نوع کلید استفاده می‌نماید:

۱. کلیدهای متقارن نشست یک‌بارمصرف (One-time session symmetric keys)
۲. کلیدهای عمومی (Public keys)
۳. کلیدهای خصوصی (Private keys)
۴. کلیدهای متقارن مبتنی بر عبارت عبور (Passphrase-based symmetric keys)

برای داشتن کلیدهای فوق، فراهم آوردن سه مورد زیر الزامی است:

- به ابزار یا راهکاری جهت تولید کلیدهای نشست غیرقابل‌پیش‌بینی، نیاز است.
- باید به کاربر اجازه داده شود که چندین جفت کلید عمومی و خصوصی داشته باشد. یک دلیل این اجازه آن است که شاید کاربر مایل به تغییر کلیدهای خود در زمان‌های مختلف باشد. اگر فرستنده دسته‌کلید نداشته باشد، در زمان تغییر کلید، پیام موجود در خط انتقال، با یک کلید منسوخ ساخته‌شده است. همچنین، دریافت‌کننده فقط کلیدعمومی قدیمی را در اختیار داشته و باید تا زمانی که به‌روزرسانی انجام می‌شود، صبر کند. دلیل دیگر آن است که ممکن است یک کاربر به‌منظور افزایش امنیت در تعامل با

گروه‌های مختلف، بخواهد از چند جفت کلید استفاده نماید. درنتیجه باید یک تناظر بین کاربران و کلیدهای مورداستفاده وجود داشته باشد. بنابراین به مکانیسمی جهت شناسایی کلیدهای خاص، نیاز خواهیم داشت.

- هر نهاد PGP باید جفت کلیدهای خصوصی و عمومی خود را در قالب یک فایل نگهداری نماید. همچنین باید کلیدهای عمومی طرف‌های مقابل را نیز به‌صورت فایل، نگهداری کند.

در ادامه این مبحث، شرایط فوق را موردبررسی قرار خواهیم داد.

مولد کلید نشست (Session Key Generation)

هر کلید نشست فقط برای یک پیام تولید شده، و تنها برای رمزگذاری و رمزگشایی همان پیام مورداستفاده قرار می‌گیرد. الگوریتم‌های رمزنگاری متقارن CAST-128 و IDEA از کلیدهای ۱۲۸ بیتی، و الگوریتم 3DES از کلید ۱۶۸ بیتی استفاده می‌نماید. برای ادامه بحث فرض می‌کنیم که از الگوریتم CAST-128 استفاده می‌شود.

تولید عدد ۱۲۸ بیتی تصادفی، با استفاده از خود الگوریتم CAST-128 انجام می‌شود. به این صورت که ورودی مولد عدد تصادفی، شامل یک کلید ۱۲۸ بیتی و دو بلوک ۶۴ بیتی (که به‌عنوان متن اصلی تلقی می‌شود که قرار است به متن رمزشده تبدیل شود)، خواهد بود. الگوریتم CAST-128 با استفاده از روش CFB، دو بلوک رمزشده ۶۴ بیتی به وجود آورده و با چسباندن این دو بلوک به هم، یک کلید نشست ۱۲۸ بیتی تولید می‌نماید.

متن ساده‌ای که در ورودی مولد عدد تصادفی قرار می‌گیرد، متشکل از دو بلوک ۶۴ بیتی است، که خود آن از یک جریان ۱۲۸ بیتی تصادفی مشتق شده است. این اعداد مبتنی بر ضربه‌های واردشده به کلیدها[1]، توسط کاربر است. توجه داشته باشید که هم از مقدار زمان ضربه‌زدن به کلیدها و هم از کلیدهای واقعی فشرده‌شده، برای تولید یک جریان تصادفی استفاده می‌شود.[2] بنابراین اگر کاربر، ضربه زدن به کلیدها را با سرعت معمول خود انجام دهد، یک ورودی تصادفی تولید خواهد شد. این مقدار تصادفی ورودی همچنین با کلید نشستی که قبلاً توسط CAST-128 تولیدشده است، ترکیب می‌شود. درنتیجه یک کلیدی که به‌طور مؤثر غیرقابل‌پیش‌بینی است، به دست خواهد آمد.

---

[1] Keystroke

[2] مثلاً کاربر کلیدهای M، T و R را در زمان حدود ۳ ثانیه وارد می‌کند. مولد برای تولید عدد تصادفی، هم از زمان ۳ ثانیه و هم از حروف MTR استفاده می‌نماید. به‌عبارت‌دیگر می‌توان گفت عدد تصادفی که از واردکردن عبارت MTR در ۳ ثانیه تولید می‌شود با عدد تصادفی که از واردکردن عبارت MTR در ۵ ثانیه تولید می‌گردد، متفاوت خواهد بود.

شناسه کلید

همان‌طور که گفته شد، پیام رمزگذاری شده باید با همان کلید نشست که برای رمزگذاری پیام استفاده گردیده، همراه باشد. البته کلید نیز به‌صورت رمزشده، ضمیمه پیام می‌گردد. کلید نشست با کلیدعمومی گیرنده رمز می‌شود؛ لذا تنها با کلیدخصوصی گیرنده امکان رمزگشایی کلید و درنتیجه رمزگشایی متن وجود خواهد داشت.

اگر هر کاربر فقط یک جفت کلیدعمومی و خصوصی داشته باشد، گیرنده به‌صورت خودکار کلید موردنظر برای رمزگشایی را تشخیص می‌دهد؛ چراکه یک کلید خصوصی و عمومی بیشتر وجود ندارد. اما اگر یک کاربر چندین جفت کلید عمومی و خصوصی داشته باشد، چگونه باید تشخیص بدهد که پیام با کدام کلیدعمومی رمزشده تا خودش بتواند با کلیدخصوصی مربوطه آن را رمزگشایی نماید!

یک راه‌حل ساده برای رفع مشکل فوق، ارسال کلیدعمومی استفاده‌شده، همراه پیام است. در این صورت دریافت‌کننده با بررسی کلیدعمومی، تشخیص می‌دهد که باید از کدام کلیدخصوصی خود برای رمزگشایی استفاده نماید. این روش درست کار خواهد کرد؛ اما با توجه به اینکه هر کلیدعمومی RSA، ممکن است دارای طولی به بزرگی صدها رقم دسیمال باشد، لذا با استفاده از این روش فضا را به‌راحتی هدر می‌دهیم.

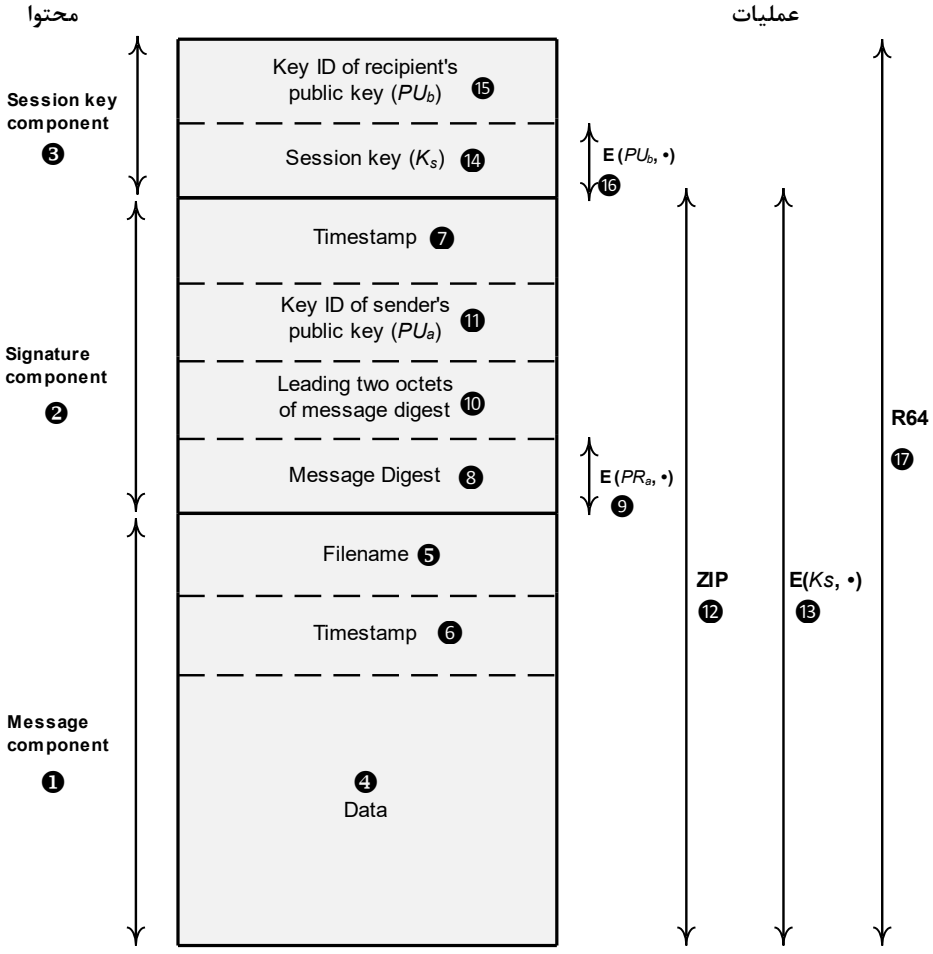
راه‌حل دوم، استفاده از شناسه برای هر کلیدعمومی است. این شناسه باید حداقل برای یک کاربر به‌صورت منحصربه‌فرد باشد، لذا می‌تواند ترکیبی از ID کاربر و ID کلید مورداستفاده باشد. در این صورت فقط نیاز است شناسه که یک مقدار کوچک است، همراه پیام ارسال شود. هرچند که راه‌حل فوق، یک راهکار مناسب‌تری است؛ اما به‌هرحال دارای سربار مدیریتی خواهد بود. باید شناسه کلید اختصاص داده شود و در هر دو طرف فرستنده و گیرنده نیز باید ارتباط کلیدعمومی وID مربوطه، ذخیره شود.

راه‌حل اتخاذشده در PGP، شبیه راه‌حل دوم است. یعنی اختصاص یک ID به هر کلیدعمومی؛ به‌طوری‌که با احتمال بسیار بالا، برای یک User ID، منحصربه‌فرد باشد. شناسه هر کلیدعمومی، حداقل دارای ۶۴ بیت باارزش است؛ که در آن شناسه کلید برای کلیدعمومی PUa، به‌صورت PUa $\bmod 2^{64}$ خواهد بود. این طول کلید کافیست، تا احتمال تکراری بودن شناسه را به حداقل برساند.

برای امضای دیجیتال در PGP، استفاده از Key ID لازم است. ازآنجاکه فرستنده ممکن است دارای چندین جفت کلیدخصوصی بوده و یکی از آن‌ها را برای امضای دیجیتال مورداستفاده قرار داده باشد؛ لذا گیرنده نیز باید بداند که از کدام کلیدعمومی متناظر برای رمزگشایی می‌تواند استفاده کند. بر همین اساس، یکی از اجزای امضای دیجیتال باید شامل یک Key ID ۶۴ بیتی برای کلیدعمومی باشد، تا گیرنده بتواند از کلیدعمومی مناسب برای بررسی امضا استفاده نماید.

تصویر زیر جزئیات بیشتری از فرمت پیام‌های تبادل شده در PGP را نمایش می‌دهد. یک پیام دارای سه بخش زیر است:

۱. بخش پیام ❶
۲. بخش امضا (به‌صورت اختیاری) ❷
۳. بخش کلید نشست (به‌صورت اختیاری) ❸



E(*PUb*, •) = encryption with user b's public key
E(*PRa*, •) = encryption with user a's private key
E(*Ks*, •) = encryption with session key
ZIP = Zip compression function
R64 = Radix-64 conversion function

بخش پیام❶ شامل دیتای اصلی است❹ که باید ذخیره یا ارسال گردد، نام فایل❺و مُهرزمانی❻ برای مشخص کردن زمان ایجاد آن است.

بخش امضا❷ نیز شامل موارد زیر است:

- **مُهر زمانی ❼**

زمان ساخت امضا را بیان می‌کند.

- **چکیده پیام ❽**

یک چکیده ایجادشده توسط الگوریتم SHA-1، که دارای طول ۱۶۰ بیت بوده و توسط کلیدخصوصی فرستنده رمزگذاری شده است.❾ محاسبه امضا بر روی مُهرزمانی الصاق شده با دیتای اصلی (که در بخش پیام قرارگرفته)، انجام می‌گیرد. گنجاندن مهرزمانی در امضا، جهت جلوگیری از حملات بازپخش (Replay) است. پس از حذف نام فایل و مهرزمانی از قسمت پیام، می‌توان از اینکه امضای منفصل، دقیقاً همان امضای متصل شده به پیام است، اطمینان حاصل نمود. منظور از امضای منفصل، امضایی است که محاسبه آن بر روی یک فایل جداگانه انجام‌شده که دارای هیچ‌کدام از فیلدهای سرآیند بخش پیام نیست.

- **دو بایت پیشرو چکیده پیام (Leading two octets of message digest) ❿**

برای اینکه گیرنده مطمئن شود که از کلیدعمومی درستی برای رمزگشایی استفاده کرده؛ دو بایت اول متن‌ساده را با دو بایت اول امضای رمزگشایی‌شده مقایسه می‌نماید. این بایت‌ها همچنین به‌عنوان یک قاب ۱۶ بیتی بررسی توالی پیام، به کار می‌روند.

- **شناسه کلید مربوط به کلیدعمومی ارسال‌کننده ⓫**

شناسه مربوط به کلیدعمومی است که باید برای رمزگشایی چکیده پیام مورداستفاده قرار گیرد.

بخش پیام و بخش امضا، ممکن است با استفاده از الگوریتم ZIP فشرده‌شده⓬ و همچنین ممکن است بر اساس کلید نشست نیز رمزنگاری شوند.⓭

بخش کلیدنشست❸، شامل کلید نشست⓮، و شناسه کلیدعمومی دریافت‌کننده است⓯ که فرستنده با استفاده از آن اقدام به رمزگذاری کلید نشست کرده است.⓰ گیرنده نیز باید با استفاده از کلیدخصوصی متناظر با کلیدعمومی مورداستفاده، اقدام به رمزگشایی کلیدنشست نماید؛ تا بتواند آن را برای رمزگشایی پیام مورد استفاده قرار دهد.

کل پیام معمولاً بر اساس Radix-64، به کد اسکی تبدیل می‌شود.⓱

دسته‌کلید

درباره شناسایی کلیدها و اهمیت آن، صحبت شد. همان‌طور که گفتیم، شناسایی کلید از اهمیت بالایی برخوردار است؛ چراکه گیرنده و فرستنده باید بدانند کدام کلید را باید برای رمزگذاری و رمزگشایی مورداستفاده قرار دهند. در PGP، از کلیدها هم برای محرمانگی و هم برای احرازهویت استفاده می‌شود.

برای حفظ کارآمدی و مؤثر بودن کلیدها، نیاز است که کلیدها به‌صورت اصولی و سازمان‌دهی شده بر روی سیستم تمام طرف‌های ارتباط، ذخیره شوند. PGP از طرحی برای سازمان‌دهی و ذخیره اصولی کلیدها استفاده می‌نماید. در این طرح، دو ساختار جهت نحوه ذخیره‌سازی و به‌کارگیری کلید، مشخص‌شده است. هر طرف باید دارای یک جدول جهت ذخیره دسته‌کلیدهای خصوصی و عمومی مربوط به خود بوده؛ و یک جدول نیز برای ذخیره دسته‌کلیدهای عمومی مربوط به طرف‌های دیگر ارتباط، داشته باشد. این ساختمان داده‌ها را دسته کلیدعمومی (Public-key ring) و دسته کلیدخصوصی (Private-key ring) می‌نامند.

ساختار ساختمان داده دسته کلیدخصوصی (Private-key ring) به‌صورت زیر است:

Timestamp	Key ID	Public Key	Encrypted Private Key	User ID
. . .	. . .	. . .	. . .	. . .
T_i	$PU_i \bmod 2^{64}$	PU_i	$E(H(P_i), PR_i)$	User i
. . .	. . .	. . .	. . .	. . .

همان‌طور که ملاحظه می‌نمایید، هر سطر جدول Private-key ring، شامل موارد زیر است:

- **Timestamp**
  تاریخ و زمانی که این جفت کلیدعمومی و خصوصی ایجادشده است.
- **Key ID**
  حداقل ۶۴ بیت باارزش از کلیدعمومی طبق فرمول: $PU_i \bmod 2^{64}$.
- **Public Key**
  کلیدعمومی مربوط به این جفت کلید.
- **Encrypted Private Key**
  کلیدخصوصی مربوط به این جفت کلید. برای حفظ امنیت، کلیدخصوصی حتی بر روی سیستم صاحب کلید نیز به‌صورت رمزشده نگهداری می‌گردد.

- **User ID**
معمولاً، این فیلد شامل پست الکترونیک کاربر است؛ مثل MTRoghani@Gmail.com. بااین‌حال، کاربر می‌تواند از نام‌های مختلفی برای هر جفت کلید استفاده نماید؛ مثل MTR، MohammadTaghiRoghani و غیره. البته می‌توان از یک User ID چندین بار نیز استفاده نمود.

دسته کلیدخصوصی را می‌توان بر اساس User ID یا Key ID، مرتب نمود. توجه داشته باشید که دسته کلیدخصوصی فقط بر روی سیستم صاحب کلید ذخیره می‌شود. برای افزایش امنیت کلیدخصوصی، قبل از ذخیره، آن را با استفاده از یک الگوریتم رمزنگاری متقارن مثل CAST-128، 3DES یا IDEA، رمزگذاری می‌نمایند. فرآیند رمزگذاری کلیدخصوصی به‌صورت زیر است:

۱. کاربر یک عبارت عبور[1] را انتخاب می‌نماید تا از آن برای رمزگذاری کلیدخصوصی استفاده شود.
۲. زمانی که سیستم با استفاده از الگوریتم RSA، یک جفت کلیدعمومی و خصوصی جدید ایجاد می‌نماید، از کاربر درخواست عبارت عبور کرده و با استفاده از الگوریتم SHA-1 یک چکیده ۱۶۰ بیتی از آن تولید کرده و سپس عبارت عبور را دور می‌اندازد.
۳. در گام سوم، سیستم کلیدخصوصی را با استفاده از الگوریتم CAST-128 و بر اساس کلید ۱۲۸ بیتی (که از ۱۶۰ بیت Hash مرحله قبل گرفته‌شده،) رمزگذاری می‌کند. پس از رمزگذاری، کلید مورداستفاده را دور انداخته و کلیدخصوصی رمزشده را در جدول مربوط به دسته کلیدخصوصی، ذخیره می‌کند.

ازاین‌پس هرگاه کاربر بخواهد از کلیدخصوصی خود استفاده نماید، باید عبارت عبور را به سیستم بدهد تا سیستم بتواند کلیدخصوصی را از حالت رمز خارج نموده و در اختیار کاربر قرار دهد. به عبارتی می‌توان گفت که امنیت کلیدخصوصی درگرو عبارت عبور مورداستفاده توسط کاربر خواهد بود. لذا کاربر باید یک عبارت عبور قدرتمند را مورداستفاده قرار دهد، تا به‌راحتی قابل حدس زدن نبوده و در برابر جستجوی فراگیر نیز مقاوم باشد.

دلیل رمزکردن کلیدخصوصی بر روی سیستم کاربر آن است که در صورت دسترسی غیرمجاز به سیستم یا جدول دسته کلیدخصوصی، فرد غیرمجاز نتواند از کلیدخصوصی اطلاع یافته و از آن استفاده نماید. با توجه به اینکه کلیدخصوصی جهت احرازهویت و عدم انکارپذیری مورداستفاده قرار می‌گیرد، لذا نگهداری امن کلیدخصوصی و حصول اطمینان از اینکه در دسترس افراد غیرمجاز قرار نگیرد، از اهمیت ویژه‌ای برای دارنده کلید برخوردار خواهد بود.

[1] Passphrase

همان‌طور که گفته شد، دومین دسته کلیدی که باید بر روی سیستم طرفین ارتباط ذخیره و نگهداری شود، دسته کلیدعمومی است. جدول زیر ساختار ساختمان داده دسته کلیدعمومی (Public-key ring) را نمایش می‌دهد:

Time stamp	Key ID	Public Key	Owner Trust	User ID	Key Legitimacy	Signature	Signature Trust
. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .
Ti	$PU_i \bmod 2^{64}$	PU_i	trust_flag $_i$	User i	trust_flag $_i$		
. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .

دسته کلیدعمومی، شامل کلیدهای عمومی مربوط به کاربرانی است که قرار است با آن‌ها ارتباط برقرار شود. کاربر باید کلیدهای عمومی اشخاصی که قصد برقراری ارتباط امن با آن‌ها را دارد، در اختیار داشته باشد.

این جدول دارای هشت فیلد است، که فیلدهای مهم آن به شرح زیر است:

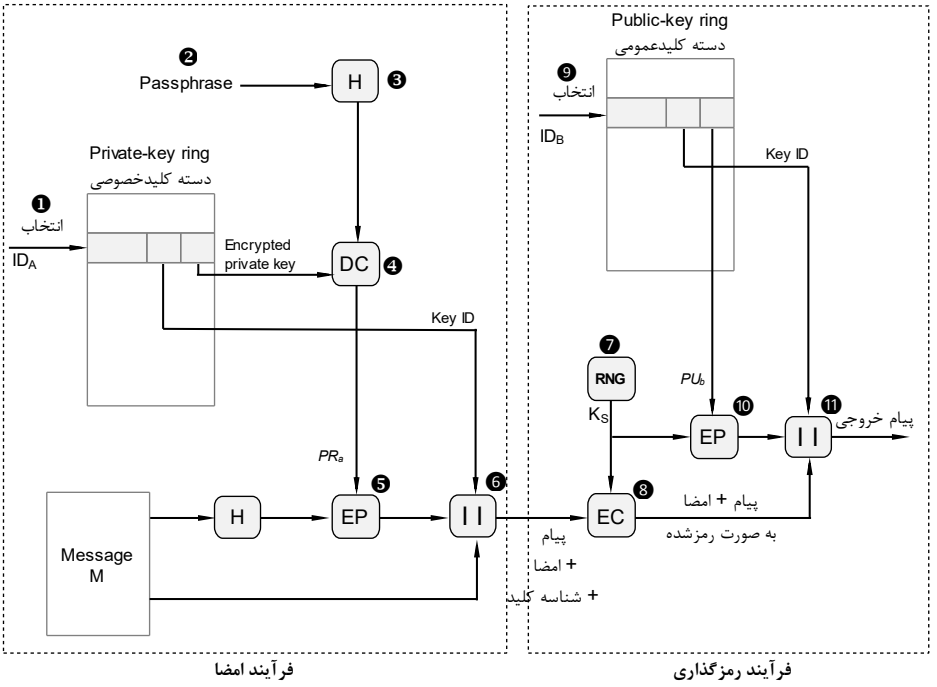
- **Timestamp**
  تاریخ و زمان تولید این نهاد.
- **Key ID**
  حداقل ۶۴ بیت باارزش از کلیدعمومی برای این نهاد.
- **Public Key**
  کلیدعمومی مربوط به این نهاد.
- **User ID**
  شناسایی صاحب این کلیدعمومی. ممکن است چندین شناسه کاربری مربوط به یک کلیدعمومی باشد.

همانند دسته کلیدخصوصی، دسته کلیدعمومی را نیز می‌توان بر اساس User ID یا Key ID، مرتب نمود.

نحوه استفاده از کلیدها برای تولید پیام، را در ادامه بررسی خواهیم نمود.

تولید پیام در PGP

در تصویر زیر نحوه استفاده از دسته‌کلیدهای خصوصی و عمومی برای تولید پیام در PGP نمایش داده‌شده است. ساختار پیام PGP که در این مرحله تولید می‌شود نیز قبلاً تشریح شده است. البته توجه داشته باشید که آقای استالینگز برای ساده‌تر و قابل‌فهم‌تر شدن تصویر، از فشرده‌سازی و تبدیل به اسکی، چشم‌پوشی کرده است.



همان‌طور که ملاحظه می‌نمایید، فرآیندی که فرستنده باید برای تولید پیام PGP انجام دهد را می‌توان به دو بخش امضا و رمزگذاری تقسیم نمود. در ادامه به تشریح مراحل فوق می‌پردازیم.

۱. **امضای پیام**

ابتدا فرستنده باید کلیدخصوصی موردنظر را بر اساس User ID انتخاب نماید.❶ اگر کلیدخصوصی خاصی مدنظر نباشد، اولین کلیدخصوصی از دسته کلیدخصوصی کاربر انتخاب می‌شود.

همان‌طور که قبلاً گفته شد، کلیدخصوصی به‌صورت رمزشده در جدول دسته کلیدخصوصی نگهداری می‌شود. به دلیل اینکه رمزگذاری کلیدخصوصی بر اساس

عبارت عبور انجام‌شده است، لذا سیستم از کاربر می‌خواهد که عبارت عبور یا Passphrase خود را وارد نماید.❷ پس از دریافت عبارت عبور، از آن Hash گرفته❸ و ۱۲۸ بیت از آن، به‌عنوان کلید تحویل الگوریتم رمزنگاری متقارن می‌شود. الگوریتم رمزنگاری بر اساس کلید دریافت شده، اقدام به رمزگشایی کلیدخصوصی نموده❹ و آن را جهت تولید امضای دیجیتال، تحویل الگوریتم رمزنگاری نامتقارن می‌دهد.
الگوریتم رمزنگاری نامتقارن نیز چکیده پیام را با استفاده از کلیدخصوصی فرستنده رمزگذاری می‌نماید.❺ درنهایت، متن اصلی پیام، امضای تولیدشده و شناسه کلیدخصوصی مورداستفاده، به یکدیگر الصاق شده❻ و تحویل قسمت رمزگذاری می‌شوند.

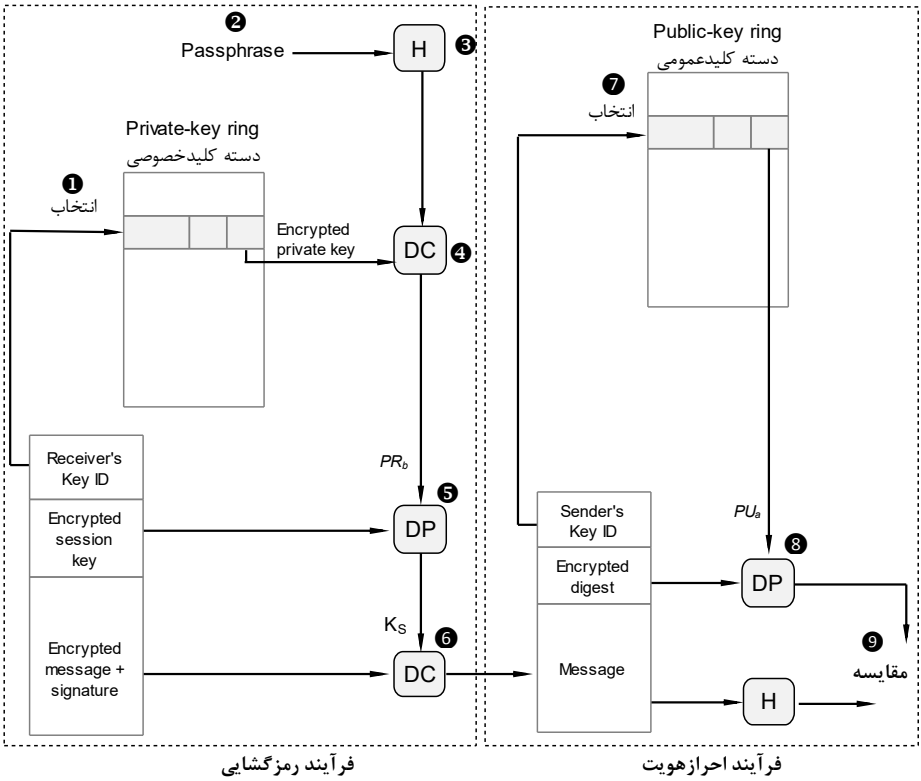
۲. **رمزگذاری پیام**

در اولین گام برای رمزگذاری، PGP باید به‌صورت تصادفی، اقدام به تولید کلیدنشست یک‌بارمصرف نمایند. این کار توسط مولد کلیدنشست انجام می‌شود که قبلاً نحوه عملکرد آن به‌صورت کامل تشریح شده است.❼
از کلیدنشست برای رمزگذاری پیام استفاده می‌شود.❽ چون الگوریتم رمزگذاری مورداستفاده، متقارن است؛ لذا باید کلیدنشست (K_S) به اطلاع گیرنده نیز برسد. گیرنده بدون اطلاع از کلیدنشست نمی‌تواند عملیات رمزگشایی را انجام دهد.
برای تبادل امن کلیدنشست، باید آن را با کلیدعمومی فرستنده رمزگذاری نمود. پس سیستم با توجه به شناسه دریافت‌کننده، اقدام به انتخاب کلیدعمومی مربوطه از جدول دسته کلیدعمومی می‌نماید.❾ سپس کلیدنشست توسط الگوریتم رمزنگاری نامتقارن و بر اساس کلیدعمومی گیرنده، رمزگذاری می‌شود.❿
در آخرین گام، متن رمزشده (شامل متن اصلی، امضا و کلیدخصوصی)، کلیدنشست رمزشده و شناسه کلیدعمومی گیرنده که برای رمزگذاری کلیدنشست استفاده شد، به یکدیگر الصاق شده⓫ و پیام PGP خروجی را ایجاد می‌نمایند. فرستنده نیز پیام را برای گیرنده ارسال می‌کند.

دریافت پیام در PGP

پس‌ازآنکه گیرنده، پیام PGP را دریافت نمود، باید مراحل رمزگشایی پیام را انجام دهد. تصویر زیر نمایش‌دهنده مراحل رمزگشایی پیام توسط دریافت‌کننده، است. البته همانند تصویر قبلی، برای درک بهتر، فرآیند فشرده‌سازی و تبدیل به کد اسکی، در این تصویر نیز نادیده گرفته‌شده است.

عملیات دریافت را می‌توان به دو بخش رمزگشایی و احرازهویت تقسیم‌بندی نمود.



۱. رمزگشایی پیام

گیرنده بر اساس Key ID تشخیص می‌دهد که فرستنده از کدام کلیدعمومی برای رمزگذاری کلیدنشست استفاده نموده، و بر اساس آن کلیدخصوصی متناظر را از جدول دسته کلیدخصوصی خود انتخاب می‌نماید.❶

سیستم برای رمزگشایی کلیدخصوصی، از کاربر درخواست عبارت عبور می‌نماید.❷ سپس از عبارت عبور Hash گرفته❸ و ۱۲۸ بیت از آن را به‌عنوان کلید رمزگشایی تحویل الگوریتم رمزنگاری متقارن می‌دهد.❹

الگوریتم رمزنگاری نامتقارن، از کلیدخصوصی جهت رمزگشایی کلیدنشست استفاده می‌نماید.❺ پس از به‌دست آمدن کلید نشست، گیرنده با استفاده از الگوریتم رمزنگاری متقارن و کلیدنشست، اقدام به رمزگشایی پیام و امضا می‌نماید.❻ پس از رمزگشایی،

متن ساده پیام، شناسه کلیدعمومی فرستنده و چکیده امضاشده، به بخش احرازهویت تحویل داده می‌شود.

۲. **احرازهویت پیام**

برای احرازهویت پیام، باید گیرنده اقدام به رمزگشایی چکیده پیام، که توسط کلیدخصوصی فرستنده امضا شده، نماید. برای این کار نیاز به کلیدعمومی فرستنده است؛ لذا بر اساس شناسه موجود در پیام، کلیدعمومی فرستنده را از جدول دسته کلیدعمومی خود انتخاب می‌نماید.❼

با استفاده از کلیدعمومی فرستنده و الگوریتم رمزنگاری نامتقارن، چکیده پیام رمزگشایی‌شده❽ و درنهایت برای بررسی صحت و احرازهویت، چکیده به‌دست‌آمده، با چکیده محاسبه‌شده از پیام، مقایسه می‌شود.❾

☑ مبحث دوم

پروتکل S/MIME

پروتکل Secure/Multipurpose Internet Mail Extension (S/MIME)، توسط پروتکل RSA، امنیت پست الکترونیکی که بر پایه تکنولوژی MIME عمل می‌نماید، را برقرار می‌کند. هر دو پروتکل PGP و S/MIME، توسط استانداردهای IETF ارائه گردیده است، اما ظاهراً پروتکل S/MIME به‌عنوان استاندارد صنعتی برای استفاده‌های تجاری انتخاب شده، درحالی‌که پروتکل PGP همچنان مورداستفاده کاربران، برای برقراری امنیت در ایمیل‌های شخصی است.

پروتکل S/MIME در تعدادی از اسناد IETF ارائه گردیده است؛ ازجمله مهم‌ترین آن‌ها می‌توان به RFCهای 3370، 3850، 3851، 3852 و 5751، اشاره نمود.

برای درک بهتر S/MIME، باید ابتدا یک آشنایی کلی با فرمت ایمیل‌هایی که بر اساس پروتکل MIME کار می‌کنند، داشته باشیم. به همین دلیل، در این مبحث یک مرور کلی بر روی استانداردهای مربوطه انجام داده و سپس به بررسی پروتکل S/MIME پرداخته می‌شود.

استاندارد RFC 5322

در RFC 5322 به تعریف یک فرمت برای پیام‌های متنی، که قرار است از طریق پست الکترونیک ارسال شوند، پرداخته شده است. این RFC همچنان به‌عنوان استانداردی برای پیام‌های متنی الکترونیکی مبتنی بر اینترنت شناخته‌شده و به‌طورمعمول مورداستفاده قرار می‌گیرد.

ازنظر RFC 5322، پیام دارای پاکت‌نامه[1] و محتوا[2] است. پاکت نامه حاوی هرگونه اطلاعاتی است که برای رسیدن پیام به مقصد لازم است. محتوا نیز شامل پیام (نامه) موردنظر ارسال‌کننده است که می‌خواهد به دست دریافت‌کننده برسد. هرچند که در RFC 5322، فقط به محتوای پیام پرداخته‌شده است؛ اما محتوای پیام نیز شامل مجموعه‌ای از فیلدهای سرآیند است که سیستم پست الکترونیک ممکن است از آن‌ها در ایجاد پاکت‌نامه استفاده نماید. به همین دلیل، استاندارد تمهیداتی را جهت تسهیل دستیابی به این اطلاعات در نظر گرفته است.

---

[1] Envelope
[2] Contents

ساختار کلی پیام در RFC 5322، بسیار ساده است. پیام متشکل از تعدادی خط در سرآیند و سپس متن نامه (بدون محدودیت) به‌عنوان بدنه پیام، است. سرآیند توسط یک خط خالی از بدنه (Body) جدا می‌شود. به‌بیان‌دیگر، با توجه به اینکه یک پیام در قالب کدهای اسکی است، لذا تمام خطوطی که در بالای اولین خط خالی قرارگرفته‌اند، به‌عنوان سرآیند در نظر گرفته‌شده و توسط سیستم پست الکترونیک از آن‌ها برای تولید پاکت نامه، استفاده می‌گردد.

خط سرآیند معمولاً شامل یک واژه کلیدی[1]، سپس علامت دونقطه[2] و در ادامه موارد مربوط به واژه کلیدی،[3] است. البته امکان شکسته شدن جملات طولانی در چند خط نیز فراهم گردیده است. اغلب واژه‌های کلیدی مورداستفاده، شامل From، To، Subject و Date است. تصویر زیر یک نمونه پیام را نشان داده است.

سرآیند (Header)

Date: July 11, 2014 2:15:49 PM EDT
From: "Mohammad Taghi Roghani" <mtroghani@gmail.com>
Subject: The Syntax in RFC 5322
To: "William Stallings" <ws@shore.net>
Cc: senatoar@yahoo.com

خط خالی

بدنه (Body)

Hello. This section begins the actual message body, which is delimited from the message heading by a blank line.

فیلد دیگری که RFC 5322 به‌طورمعمول از آن استفاده می‌کند، فیلد شناسه پیام یا Message-ID است. این فیلد شامل یک شناسه منحصربه‌فرد مربوط به پیام است.

البته پروتکل SMTP و همچنین RFC 5322 دارای محدودیت‌ها و بعضاً مشکلاتی هستند که باعث شده در طول زمان، کاربر در استفاده از آن‌ها دچار مشکل شود.

لیست زیر شامل محدودیت‌های طرح SMTP/5233 است:

۱. پروتکل SMTP نمی‌تواند فایل‌های اجرایی یا دیگر اشیاء باینری را انتقال دهد. البته برنامه‌هایی وجود دارد که فایل‌های باینری را به متن تبدیل کرده تا SMTP بتواند از آن‌ها استفاده نماید. اما هیچ‌یک از آن‌ها استاندارد نشده است.
۲. پروتکل SMTP نمی‌تواند متن دیتایی را که شامل کاراکترهای زبان‌های ملی است، انتقال دهد؛ چراکه این کاراکترها توسط کدهای ۸ بیتی با مقدار ۱۲۸ رقم دسیمال یا بیشتر، بیان می‌شوند، درصورتی‌که SMTP محدود به کدهای اسکی ۷ بیتی است.

[1] Keyword
[2] Colon
[3] Keyword's arguments

۳. سرورهای SMTP ممکن است پیام‌های بزرگ‌تر از یک اندازه خاص را رد کنند.
۴. دروازه[1] SMTP ممکن است در ترجمه بین کدهای اسکی و کدهای EBCDIC[2] با مشکل مواجه شود.[3]
۵. دروازه SMTP با شبکه‌های پست الکترونیک X.400، نمی‌تواند دیتای غیرمتنی را که در پیام‌های X.400 است، اداره نماید.[4]
۶. در برخی موارد، زمان پیاده‌سازی SMTP موارد گفته‌شده در استاندارد RFC 821 به‌طور کامل رعایت نشده و باعث بروز برخی مشکلات می‌گردد.

پروتکل MIME

برای رفع برخی محدودیت‌ها و مشکلات موجود در پروتکل SMTP[5]، بعضی پروتکل‌های انتقال و همچنین RFC 5322، سازمان IETF اقدام به توسعه RFC 5322 در قالب طرح Multipurpose Internet Mail Extension (MIME) برای پست الکترونیک نموده است. ایجاد MIME به شیوه‌ای صورت گرفته که با RFC 5322 سازگار باشد.

پروتکل MIME هم دارای چندین RFC است که از مهم‌ترین آن‌ها می‌توان به RFCهای 2045 تا 2049 اشاره نمود.

پروتکل MIME نسبت به پروتکل‌های دیگر پست الکترونیک، دارای عناصر ویژه زیر است:

۱. پنج فیلد جدید در سرآیند پیام اضافه‌شده، که شامل اطلاعاتی درباره بدنه پیام است. این پنج فیلد جدید شامل موارد زیر است:
   i. MIME-Version
   ii. Content-Type
   iii. Content-Transfer-Encoding
   iv. Content-ID
   v. Content-Description

---

[1] Gateway
[2] Extended Binary Coded Decimal Interchange Code (EBCDIC)
[3] EBCDIC یک کدگذار کاراکتر ۸ بیتی است که توسط IBM ایجاد گردیده است. برای اطلاعات بیشتر در این مورد می‌توانید به لینک زیر مراجعه نمایید:
http://nemesis.lonestar.org/reference/telecom/codes/ebcdic.html
[4] X.400 یک مجموعه از توصیه‌های ITU-T است که به تعریف استاندارد شبکه‌های ارتباطی دیتا برای سیستم‌های ایمیل، می‌پردازد. برای اطلاعات بیشتر می‌توانید به آدرس زیر در سایت ITU-T مراجعه نمایید:
http://www.itu.int/rec/T-REC-X.400/en
[5] RFC 821

۲. تعدادی فرمت محتوا برای MIME تعریف‌شده که آن را قادر می‌سازد از پست الکترونیک چندرسانه‌ای[1] پشتیبانی نماید.
۳. با تعریف کدگذاری انتقال، پروتکل می‌تواند هر نوع فرمت محتوا را به یک فرم تبدیل نموده تا در مقابل تغییرات سیستم پست الکترونیک، محافظت گردد.

انواع محتوا در MIME

بخش عمده‌ای از مشخصات MIME در رابطه با انواع مختلف محتوا است. این موضوع بیانگر نیاز به ارائه یک استاندارد برای تعامل با طیف وسیعی از بازنمایی اطلاعات در یک محیط چندرسانه‌ای است. در RFC 2046، انواع مختلف محتوا در هفت گروه اصلی و پانزده زیرگروه، به‌صورت زیر تقسیم‌بندی شده است.

۱. **متن (Text)**
   i. ساده (Plain)
   ii. غنی (Enriched)؛ انعطاف‌پذیری بیشتری را نسبت به ساده، فراهم می‌آورد.
۲. **چندبخشی (Multipart)**
   i. مخلوط (Mixed)؛ بخش‌های مستقل مختلف باهم ارسال می‌شوند.
   ii. موازی (Parallel)؛ شبیه مخلوط، با این تفاوت که روشی برای تحویل یک بخش به گیرنده، در نظر گرفته نشده است.
   iii. چکیده (Digest)؛ شبیه مخلوط؛ اما در RFC 822 به‌صورت پیش‌فرض در نظر گرفته‌شده است.
   iv. جایگزین (Alternative)؛ نسخه‌های جایگزین برای بخش‌های مختلف.
۳. **پیام (Message)**
   i. RFC 822؛ بسته‌بندی بدنه (Body) بر اساس RFC 822 انجام می‌پذیرد.
   ii. جزئی (Partial)؛ امکان شکستن موارد بزرگ به قسمت‌های کوچک‌تر.
   iii. بدنه خارجی (External-body)؛ شامل یک اشاره‌گر به یک شیء در جای دیگر.
۴. **تصویر (Image)**
   i. jpeg
   ii. gif
۵. **ویدئو (Video)**
   i. MPEG

[1] Multimedia

۶. صدا (Audio)

i. عمومی (Basic)؛ تنها یک کانال ۸ بیتی با نرخ 8KHz پشتیبانی می‌شود.

۷. نرم‌افزار (Application)

i. PostScript
ii. Octet-stream

کدگذاری انتقال در MIME

بخش مهم دیگر MIME، کدگذاری انتقال است؛ که در آن به تعریف کدگذاری بدنه پیام برای انتقال، پرداخته می‌شود. هدف از این کدگذاری، فراهم آوردن یک تحویل مطمئن در محیط‌های مختلف است.

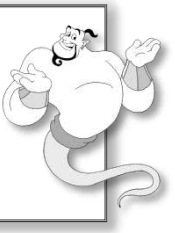
استاندارد MIME، دو روش را برای کدگذاری داده‌ها، تعریف می‌نماید. درصورتی‌که فیلد Content-Transfer-Encoding، شش مقدار زیر را جهت کدگذاری داده‌ها نشان می‌دهد:

۱. 7bit
۲. 8bit
۳. binary
۴. quoted-printable
۵. based64
۶. x-token

به‌هرحال، سه مقدار اول (7bit، 8bit و binary)، نشان می‌دهد که هیچ‌گونه کدگذاری برای دیتا فراهم نگردیده است.

مقدار دیگری که ممکن است در فیلد Content-Transfer-Encoding قرار گیرد، x-token است. دلیل استفاده از کلمه x این است که، کدگذاری می‌تواند طبق استاندارد نبوده و به‌صورت خاص طراحی‌شده باشد.

مقادیر دیگری که ممکن است در این فیلد قرار گیرند، based64 و quoted-printable هستند. در این طرح‌ها دو انتخاب فراهم می‌گردد. یکی ارائه روش‌های انتقالی است که برای انسان قابل‌خواندن است؛ و دیگری یک‌راه منطقی و جمع‌وجور برای امن نگه‌داشتن انواع دیتا می‌باشد.



Based-64 بانام Radix-64 نیز شناخته می‌شود. از Based-64 معمولاً زمانی استفاده می‌شود که بخواهیم دیتای باینری را بر روی رسانه‌ای که برای داده‌های متنی طراحی شده، ذخیره یا ارسال نماییم. با استفاده از Based-64 اطمینان حاصل می‌شود که دیتا در طول مسیر انتقال، دست‌نخورده باقی‌مانده است. به‌عبارت‌دیگر می‌توان گفت که Based-64 در اینجا، کاری شبیه به چکیده پیام را انجام می‌دهد؛ ولی نباید از آن، انتظار فراهم آوردن محرمانگی را داشته باشیم؛ چراکه به‌راحتی قابل Encoding است.

پروتکل S/MIME

ازلحاظ عملکرد کلی، S/MIME بسیار شبیه PGP است. هر دو پروتکل قابلیت ارائه امضا و رمزگذاری پیام را فراهم می‌آورند. معمولاً از PGP برای امنیت پست الکترونیک مبتنی بر SMTP و از S/MIME برای امنیت پست الکترونیک مبتنی بر MIME، استفاده می‌شود.

کارکردهای S/MIME شامل موارد زیر است:

- **ایجاد پاکت برای داده‌ها**[1]

این پاکت‌ها شامل هر نوع محتوای رمزشده به همراه کلیدهای رمزنگاری مورداستفاده، برای یک یا چند دریافت‌کننده است.

- **امضای داده‌ها**

یک امضای دیجیتالی است؛ که در آن، چکیده پیام موردنظر با استفاده از کلیدخصوصی فرستنده، رمزگذاری می‌شود. سپس امضا به همراه محتوا، توسط Base-64 encoding، کدگذاری می‌گردد. مشاهده داده‌های امضاشده، تنها برای گیرندگانی امکان‌پذیر خواهد بود که قابلیت استفاده از S/MIME را دارند.

- **امضای جدا از داده**[2]

در این حالت، به‌جای کل پیام، فقط امضای دیجیتال توسط Base-64 encoding، کدگذاری می‌گردد. درنتیجه حتی اگر دریافت‌کننده قابلیت پشتیبانی از S/MIME را نداشته باشد، می‌تواند پیام را ملاحظه نماید؛ اما امکان تائید امضا توسط گیرنده وجود نخواهد داشت.

- **امضا و پاکت کردن داده**

امضای تنها و رمزگذاری تنها، ممکن است باعث تودرتو شدن آن‌ها شود؛ به‌طوری‌که ممکن است داده‌های رمزگذاری شده، امضا شوند و امضاها نیز رمزگذاری شوند.

الگوریتم‌های رمزنگاری در S/MIME

جدول زیر به‌طور خلاصه الگوریتم‌های رمزنگاری مورداستفاده در S/MIME را تعریف می‌نماید. قبل از بررسی جدول باید به دو اصطلاح کلیدی Should و Must توجه نمایید. طبق RFC 2119 تعریف این اصطلاحات به شرح زیر است:

- **MUST**

برای Must من از معادل فارسی "**باید**" استفاده می‌کنم. این اصطلاح به تعریف نیاز مطلق به ویژگی موردنظر اشاره دارد.

---

[1] Enveloped data
[2] Clear-signed data

- **SHOULD**

برای Should، من از معادل فارسی "**بجاست**" استفاده می‌کنم. این اصطلاح زمانی استفاده می‌شود که بتوان از ویژگی موردنظر در شرایط خاص صرف‌نظر کرد. اما توصیه مؤکد می‌شود که آن ویژگی مورداستفاده قرار گیرد.

در جدول زیر به‌صورت خلاصه به الگوریتم‌های رمزنگاری مورداستفاده در S/MIME پرداخته‌شده است:

عملکرد	الزامات
ایجاد چکیده پیام برای استفاده جهت امضای دیجیتال	* باید از SHA-1 پشتیبانی شود. * بجاست دریافت‌کننده از MD5 برای سازگاری با روش‌های قدیمی‌تر، پشتیبانی نماید.
رمزگذاری چکیده پیام جهت ایجاد یک امضای دیجیتال	* باید فرستنده و گیرنده از DSS پشتیبانی نمایند. * بجاست فرستنده از رمزنگاری RSA پشتیبانی نماید. * بجاست دریافت‌کننده قابلیت بررسی امضای RSA را با طول کلید ۵۱۲ بیت تا ۱۰۲۴ بیت داشته باشد.
رمزگذاری کلیدنشست برای تبادل به‌همراه پیام	* بجاست فرستنده و گیرنده از دیفی‌هلمن پشتیبانی نمایند. * باید دریافت‌کننده و ارسال‌کننده از رمزنگاری RSA با طول کلید ۵۱۲ بیت تا ۱۰۲۴ بیت، پشتیبانی نماید.
رمزگذاری پیام با کلیدنشست یک‌بارمصرف، برای انتقال	* باید فرستنده و گیرنده از الگوریتم رمزنگاری 3DES پشتیبانی نمایند. * بجاست فرستنده از رمزنگاری AES پشتیبانی کند. * بجاست فرستنده از رمزگذاری RC2/40 پشتیبانی کند.
ایجاد کد احرازهویت پیام (MAC)	* باید دریافت‌کننده از HMAC با SHA-1 پشتیبانی نماید. * بجاست فرستنده از HMAC با SHA-1 پشتیبانی کند.

اگر به یاد داشته باشید، در مبحث SSL گفتم که حق با مشتری است! یعنی سرور باید بر اساس توانمندی‌های کلاینت اقدام به انتخاب الگوریتم‌های مورداستفاده نماید. در اینجا نیز فرستنده باید بر اساس توانایی‌های رمزگشایی گیرنده اقدام به انتخاب الگوریتم رمزگذاری نماید؛ البته فرستنده تا آنجایی می‌تواند کوتاه بیاید که سیاست‌های امنیتی تعریف‌شده، زیر پا گذاشته نشود!

برای اجرای صحیح فرآیند تصمیم‌گیری، بهتر است ارسال‌کننده قابلیت‌ها را به ترتیب اولویت برای گیرنده بیان کند. در S/MIME، نیز برای تصمیم‌گیری بهتر، قوانین زیر به ترتیب توسط ارسال‌کننده، دنبال می‌شوند:

۱. اگر فرستنده لیستی از قابلیت‌های رمزگشایی گیرنده در اختیار دارد، باید اولین قابلیت آن لیست، که قادر به استفاده از آن است را انتخاب نماید.
۲. اگر فرستنده لیستی از قابلیت‌های گیرنده در دست ندارد، ولی یک یا چند پیام از گیرنده دریافت نموده؛ باید از الگوریتم‌های مشابه که در آن پیام‌ها به کار گرفته‌شده، استفاده نماید.
۳. اگر فرستنده هیچ اطلاعی از قابلیت‌های رمزگشایی گیرنده نداشته و مایل هم نیست در مورد انتخاب الگوریتم ریسک نماید[۱]، پس **بجاست** از الگوریتم 3DES استفاده کند.
۴. اگر فرستنده هیچ اطلاعی از قابلیت‌های رمزگشایی گیرنده نداشته و مایل هم نیست در مورد انتخاب الگوریتم ریسک نماید، پس **باید** حداقل از الگوریتم RC2/40 استفاده کند.

اگر قرار باشد یک پیام به چندین دریافت‌کننده ارسال شود، و امکان استفاده از یک الگوریتم رمزنگاری مشترک برای همه آن‌ها فراهم نباشد؛ نیاز به ایجاد دو پیام با الگوریتم رمزگذاری متفاوت خواهد بود. این نکته بسیار مهم است که بدانید: امنیت کل پیام، به‌اندازه پایین‌ترین نسخه رمزگذاری شده آن، در معرض آسیب‌پذیری و خطر خواهد بود.



[۱] منظور از ریسک، انتخاب یک الگوریتم رمزگذاری قوی توسط فرستنده است؛ درصورتی‌که این احتمال وجود دارد که گیرنده قادر به استفاده از آن الگوریتم و درنتیجه رمزگشایی پیام نباشد.

پیام S/MIME

پروتکل S/MIME از چند نوع جدید محتوای MIME استفاده می‌نماید که در جدول زیر نمایش داده‌شده است. در این جدول، تمام کاربردهای جدید با عبارت PKCS[1] نمایش داده شده است. PKCS به مجموعه‌ای از مشخصات رمزنگاری کلیدعمومی اشاره دارد که توسط آزمایشگاه‌های RSA صادر گردیده و توسط S/MIME مورداستفاده قرارگرفته است.[2]

نوع (Type)	زیرگروه	پارامتر S/MIME	تشریح
Multipart	Signed		امضای جدا از داده؛ یک قسمت پیام و قسمت دیگر امضا.
Application	pkcs7-mime	signedData	تمام S/MIME امضا می‌شود.
	pkcs7-mime	envelopedData	تمام S/MIME رمزگذاری می‌شود.
	pkcs7-mime	degenerate signedData	فقط شامل گواهی‌نامه کلیدعمومی
	pkcs7-mime	CompressedData	تمام S/MIME فشرده‌سازی می‌شود.
	pkcs7-Signature	signedData	نوع محتوای امضا زیرمجموعه پیام Multipart/Signed.

پروتکل S/MIME، پیام‌های MIME را توسط امضا، رمزگذاری، و یا هر دو، امن می‌نمایند. یک MIME ممکن است تنها یک پیام (بدون سرآیند گفته‌شده در RFC 5322) بوده و یا از نوع چندقسمتی (Multipart) باشد؛ که در این صورت MIME دارای یک یا چند بخش خواهد بود.

یک پیام MIME بر اساس قوانین عادی مربوط به آماده‌سازی پیام ایجاد می‌گردد؛ لذا فرآیندهای امنیتی باید توسط S/MIME و بر اساس PKCSها بر روی آن‌ها اعمال گردد.

در ادامه به شرح بیشتر برخی از انواع محتوای S/MIME از جدول فوق می‌پردازیم. زیرمجموعه Application/pkcs7-mime، برای یکی از چهار گروه پردازش S/MIME مورداستفاده قرار می‌گیرد؛ البته با پارامتر منحصربه‌فرد خود. در همه موارد، نتیجه (که Object نامیده می‌شود) به‌صورت BER[3] و بر اساس توصیه‌های X.209 ارائه‌شده توسط ITU-T، بیان می‌شوند.

---

[1] Public-Key Cryptography Standard (PKCS)

[2] آزمایشگاه RSA، PKCSها را در گروه‌های مختلف تقسیم‌بندی کرده و با شماره‌های مختلف نمایش می‌دهد. هر PKCS نیز دارای یک RFC مخصوص به خود در IETF است. به‌عنوان‌مثال PKCS#7 که در مبحث ما مورداستفاده قرارگرفته دارای RFC 2315 و نام Cryptographic Message Syntax است.

[3] Basic Encoding Rules (BER)

- **Enveloped Data**

اگر پارامتر PKCS7، عبارت envelopedData باشد، S/MIME از PKCS7 جهت رمزگذاری تمام پیام استفاده می‌نماید. مراحل ایجاد Enveloped Data به‌صورت زیر است:

۱. برای الگوریتم رمزنگاری متقارن موردنظر(RC2/40 یا 3DES)، اقدام به تولید یک مقدار تصادفی به‌عنوان کلیدنشست، می‌نماید.
۲. کلیدنشست را بر اساس کلیدعمومی RSA هر گیرنده، رمزگذاری می‌نماید.
۳. برای هر گیرنده اقدام به آماده‌سازی یک بلوک دیتا با عنوان RecipientInfo می‌نماید؛ که حاوی شناسه گواهینامه کلیدعمومی گیرنده، شناسه الگوریتم مورداستفاده جهت رمزگذاری کلیدنشست، و کلیدنشست رمزگذاری شده، است.
۴. محتوای پیام را بر اساس کلیدنشست رمزگذاری می‌نماید.

بلوک‌های RecipientInfo و محتوای رمزشده، Enveloped Data را تشکیل می‌دهند. سپس این اطلاعات توسط Base64 کدگذاری می‌شوند.

- **Signed Data**

اگر پارامتر PKCS7، عبارت signedData باشد، S/MIME از PKCS7 جهت امضای تمام پیام استفاده می‌نماید. signedData می‌تواند از یک یا چند امضا استفاده نماید. اما برای درک بهتر، توضیحات به‌صورت تک امضا بیان می‌شود. مراحل ایجاد signedData به‌صورت زیر است:

۱. انتخاب یک الگوریتم درهم‌ساز (SHA یا MD5) برای گرفتن چکیده پیام.
۲. محاسبه چکیده پیام از محتوایی که قرار است امضا شود.
۳. رمزگذاری چکیده پیام توسط کلیدخصوصی امضاکننده.
۴. ایجاد یک بلوک به نام SignerInfo، که حاوی گواهینامه کلیدعمومی امضاکننده، شناسه الگوریتم درهم‌ساز مورداستفاده، شناسه الگوریتم استفاده‌شده جهت رمزگذاری چکیده پیام، و چکیده پیام رمزشده، است.

بلوک‌های تشکیل‌دهنده signedData شامل شناسه الگوریتم درهم‌ساز، اصل پیامی که امضاشده، و بلوک SignerInfo است. درنهایت این اطلاعات توسط Base64 کدگذاری می‌شوند.

پردازش گواهینامه در S/MIME

S/MIME از گواهینامه کلیدعمومی مطابق با X.509 v3 استفاده می‌نماید. طرح مدیریت کلید مورداستفاده توسط S/MIME، ترکیبی از گواهی سلسله مراتبی X.509 و PGP Trust است. همانند PGP در S/MIME نیز مدیران و در صورت نیاز، کاربران، باید کلاینت‌ها را برای استفاده از لیست کلیدهای مورد اعتماد[1] و لیست ابطال گواهینامه[2](CRL)، پیکربندی نمایند. در این‌صورت مسئولیت محلی برای حفظ گواهینامه‌ها به وجود می‌آید که نیاز است امضاهای ورودی را بررسی، و پیام‌های خروجی را رمزگذاری نماید. از طرف دیگر گواهینامه‌ها باید توسط مرجع صدور گواهینامه[3](CA)، امضا شده باشد.

نقش کاربر

کاربر S/MIME، باید کارهای زیر را برای مدیریت کلید انجام دهد:

- **تولید کلید (Key Generation)**

کاربر **باید** قادر به تولید جداگانه کلیدهای دیفی‌هلمن و DSS باشد؛ همچنین **بجاست** قادر به تولید کلید RSA نیز باشد. برای تولید هر جفت کلید **باید** از یک منبع امن که به‌خوبی مقدار تصادفی ایجاد می‌نماید، استفاده کند. **بجاست** کاربر جفت کلید RSA را با طول بین ۷۶۸ تا ۱۰۲۴ بیت تولید کند؛ و **نباید** طول آن‌ها کوچک‌تر از ۵۱۲ بیت باشد.

- **ثبت (Registration)**

برای دریافت گواهینامه کلیدعمومی X.509، کلیدعمومی کاربر **باید** در CA ثبت گردد.

- **ذخیره‌سازی و بازیابی گواهینامه (Certificate storage and retrieval)**

جهت بررسی امضاهای ورودی و رمزگذاری پیام‌های خروجی، کاربر باید به‌صورت محلی به لیست گواهینامه‌ها دسترسی داشته باشد. این لیست می‌تواند توسط کاربر به‌تنهایی و یا نهادهای مدیریت محلی به نمایندگی از تعدادی کاربر، نگهداری شود.

گواهینامه VeriSign

در حال حاضر، چندین شرکت وجود دارند که خدمات مرجع صدور گواهینامه یا CA، را فراهم می‌آورند. به‌عنوان‌مثال، شرکت نورتل یک راه‌حل CA طراحی نموده که می‌تواند S/MIME را در یک سازمان پشتیبانی نماید. ازجمله CAهای دیگر اینترنت، می‌توان به VeriSign، GTE و

[1] Trusted keys
[2] Certificate Revocation List (CRL)
[3] Certification Authority (CA)

U.S. Postal Service اشاره نمود. از مواردی که نام برده شد، VeriSign به‌صورت گسترده‌تری مورداستفاده قرار می‌گیرد. به همین دلیل به شرح مختصری از VeriSign می‌پردازیم.

VeriSign اقدام به ارائه خدمات CA سازگار با S/MIME و طیف وسیعی از کاربردهای دیگر، نموده است. VeriSign گواهینامه X.509 را در قالب محصولی بانام VeriSign Digital ID صادر می‌نماید. اطلاعاتی که در Digital ID است، بستگی به نوع Digital ID و استفاده از آن دارد.

حداقل هر Digital ID شامل موارد زیر است:

- کلیدعمومی مالک
- نام اصلی یا مستعار مالک
- تاریخ انقضای Digital ID
- شماره‌سریال Digital ID
- نام مرجع صدور گواهینامه (CA) که این Digital ID را صادر نموده.
- امضای دیجیتال مرجع صدور گواهینامه (CA) که این Digital ID را صادر نموده.

همچنین Digital ID می‌تواند شامل اطلاعات دیگر کاربر ازجمله موارد زیر باشد:

- آدرس
- آدرس پست الکترونیک
- اطلاعات عمومی ثبت‌شده (کشور، کد پستی، سن و جنسیت)

VeriSign سه سطح یا کلاس را برای امنیت گواهینامه‌های کلیدعمومی فراهم می‌آورد. یک کاربر می‌تواند درخواست گواهینامه خود را به‌صورت آنلاین برای VeriSign ارسال نماید. درخواست گواهینامه‌های کلاس ۱ و ۲ به‌صورت آنلاین ارسال گردیده و ظرف مدت چند ثانیه نیز صادر می‌گردد. اما صدور گواهینامه کلاس ۳ نیاز به درجه بالاتری از حصول اطمینان برای هویت دارد و درخواست‌کننده نیز باید اطلاعات بیشتر و کامل‌تری را برای اثبات هویت خود ارائه نماید.

فصل دوازدهم

امنیت سیستم‌عامل

مبحث اول

بهترین شیوه‌های مدیریت امن سیستم

مبحث دوم

سطوح امنیتی در سیستم‌عامل

مبحث سوم

بهترین شیوه‌های امنیت سیستم‌عامل

مدیریت صحیح سیستم‌ها و سرورها، و همچنین امنیت سیستم‌عامل، قصه هزارویک‌شب است؛ که برای هرکدام از موضوعات فوق به‌تنهایی، می‌توان در قالب یک یا چند کتاب مستقل به ارائه شیوه‌های امنیتی پرداخت.

مدیریت امن سیستم دارای کتاب‌ها، مقالات و ابزارهای متعددی است. همچنین در بخشی از سیستم مدیریت امنیت اطلاعات (ISMS) نیز به آن پرداخته‌شده است. ما نیز در این فصل یک مبحث را به ذکر بهترین شیوه‌های مدیریت سیستم اختصاص داده‌ایم. در این مبحث بر اساس مقالات معتبر منتشرشده، مواردی به مدیر سیستم گوشزد می‌شود که همواره باید مدنظر ایشان باشد.

امنیت سیستم‌عامل جزء مباحث بسیار گسترده است که دائماً در حال تغییر و به‌روزرسانی است. امنیت سیستم‌عامل را می‌توان بر اساس برند به گروه‌هایی نظیر مایکروسافت، مک، یونیکس و لینوکس تقسیم‌بندی نمود. هرکدام از این برندها نیز دارای نسخه‌های مختلفی از سیستم‌عامل در دو گروه اصلی دسکتاپ و سرور هستند. برای هر نسخه از هر برند، کتاب‌ها و مقالات امنیتی فراوانی چاپ و منتشرشده است. لذا برای فعالیت تخصصی درزمینه امنیت سیستم‌عامل، بهتر است که شما بر اساس سیستم‌عامل‌هایی که وظیفه مدیریت آن‌ها را بر عهده‌دارید، سراغ منابع مختص به آن‌ها بروید. البته یادتان باشد که آن منابع باید به‌روز باشند.

اما در این فصل با توجه به هدف و نام این کتاب، که امنیت اطلاعات است و بنده نیز تلاش کردم مطالب کتاب را به‌صورت عمومی و فارغ از برند یا نسخه خاصی جمع‌آوری و نگارش نمایم؛ لذا در رابطه با موضوعات مدیریت سیستم و امنیت سیستم‌عامل نیز سعی شده به مطالب عمومی که در تمام برندها و نسخه‌ها قابل‌اجرا بوده و باید مدنظر قرار گیرد، پرداخته شود.

☑ مبحث اول

بهترین شیوه‌های مدیریت امن سیستم

مدیر سیستم (System Administrator)، به افرادی اطلاق می‌شود که مسئول پاسخگویی اتفاقات مربوط به سیستم‌ها هستند. آن‌ها همچنین مسئول عملیات بدون وقفه سیستم‌هایی هستند که برای تداوم کسب‌وکار سازمان ضروری می‌باشند. این مدیران باید دانش خود را نسبت به آسیب‌پذیری‌ها و تهدیداتی که ممکن است برای سیستم‌ها وجود داشته باشد، ارتقاء دهند. یکی از راهکارهایی که معمولاً توسط مدیران باتجربه مورداستفاده قرار می‌گیرد، عمل به بهترین شیوه‌های ارائه‌شده جهت پیکربندی و امنیت سیستم‌ها، است.

علاوه بر انجام بهترین شیوه‌ها که در این مبحث به آن‌ها پرداخته‌شده؛ مدیر سیستم باید به مستندات فنی و امنیتی تجهیزات و نرم‌افزارها، که توسط تولیدکنندگان آن‌ها ارائه می‌شود، کاملاً توجه داشته باشد. یکی از کارهای دیگری که معمولاً توسط مدیران انجام می‌شود، عضویت در بولتن‌ها و خبرنامه‌های امنیتی است که آسیب‌پذیری‌ها و نحوه مقابله با آن‌ها را برای محصولات موردنظر، ارائه می‌دهند.

در ادامه به موارد عمومی که طبق بهترین شیوه‌ها، باید توسط مدیران سیستم، در نظر گرفته شود، می‌پردازیم.

امنیت فیزیکی سیستم و کنسول

امنیت فیزیکی همیشه جزو اولین مواردی است که در هر نوعی از امنیت باید اجرا گردد. در صورت فقدان امنیت فیزیکی، ممکن است خساراتی به سازمان وارد گردد که قابل‌مقایسه با سایر تهدیدات نباشد. اگر برای فرد مهاجم دسترسی فیزیکی به کنسول یا سیستم امکان‌پذیر باشد، به‌راحتی می‌تواند اقدام به خرابکاری یا سوءاستفاده از آن نماید. همان‌طور که می‌دانید در تجهیزات شبکه، سرورها و سیستم‌عامل‌ها، روش‌هایی توسط تولیدکننده در نظر گرفته‌شده تا در صورت اتفاقاتی مثل از یاد بردن کلمه‌عبور، امکان بازیابی کلمه‌عبور را داشته باشید. اغلب روش‌هایی که بر روی تجهیزات و سیستم‌عامل‌ها جهت بازیابی کلمه‌عبور در نظر گرفته‌شده، نیاز به دسترسی فیزیکی دارند. علاوه بر این، در صورت دسترسی فیزیکی به سرور، مهاجم می‌تواند از طریق یک سیستم‌عامل دیگر، اقدام به راه‌اندازی مجدد سرور نموده و خیلی راحت به اطلاعات

ذخیره‌شده بر روی آن، دسترسی پیدا نماید. به تمام این موارد، خراب‌کاری فیزیکی که توسط یک مهاجم عصبی! نیز ممکن است انجام شود، را اضافه نمایید.

البته امنیت فیزیکی فقط مقابله با افراد خرابکار نیست؛ بلکه امنیت در مقابل خطاهای انسانی، خرابی تجهیزات و بلایای طبیعی مانند سیل و زلزله، نیز در بخش امنیت فیزیکی قرار می‌گیرند.

برقراری امنیت فیزیکی دارای راهکارها و ابزارهای گوناگون و متفاوتی است که روزبه‌روز نیز تغییر می‌یابند. اینکه از چه ابزار یا تجهیزاتی برای امنیت فیزیکی استفاده شود، وابسته به نوع سازمان، کسب‌وکار، درجه اهمیت اطلاعات و البته هزینه آن بوده و در هر سازمان متفاوت است. اما آن چیزی که باید توسط مدیر سیستم انجام شود، رعایت مسائل حیاتی مربوط به امنیت فیزیکی است؛ حال این عمل با استفاده از چه ابزار یا امکاناتی انجام شود، وابسته به همان مسائلی است که گفته شد.

به‌هرحال، مدیر سیستم باید حداقل‌های زیر را جهت امنیت فیزیکی رعایت نماید:

- کنسول سیستم باید به‌صورت فیزیکی محافظت شود. سیستم را در یک محل امن نصب نموده و از اینکه تنها افراد مجاز امکان دسترسی فیزیکی دارند، اطمینان حاصل نمایید.
- در صورت دسترسی فیزیکی، مهاجم می‌تواند به‌راحتی کلید Power را فشرده و سیستم را خاموش نماید. همچنین می‌تواند با استفاده از درایورCD یا پورت USB، اقدام به Boot سیستم بر اساس سیستم‌عامل موردنظر خود نماید. لذا علاوه بر لایه‌های امنیت فیزیکی که تاکنون انجام‌شده، باید تجهیزات را درون رک قرار داده و درب رک نیز قفل باشد. همچنین حتی‌الامکان باید راه‌اندازی سیستم از طریق درایورها و پورت‌ها نیز غیرفعال شود. البته به این نکته توجه داشته باشید که در صورت غیرفعال کردن درایورها و پورت‌های خارجی، در زمان نیاز خود شما نیز غیرفعال خواهند بود؛ لذا هنگام برنامه‌ریزی، به این موضوع دقت کافی داشته باشید.
- کنسول سیستم را به حال خود رها نکنید. بعضی از کاربران یا مدیران، پس از Login به سیستم و در بین کار، کنسول را رها کرده و مثلاً برای صرف چای یا انجام کار دیگری، میز کار خود را ترک می‌کنند. در این صورت مهاجم به‌راحتی و بدون هیچ تلاشی می‌تواند دسترسی غیرمجاز موردنظر خود را به‌دست آورد. لذا کاربران و مخصوصاً مدیران، در زمان ترک میز کار باید حتماً از کنسول خارج شوند.
- اگر سیستم شما دارای قابلیت Timeout است (که البته تمام سیستم‌های امروزی این قابلیت را دارند)، حتماً زمان آن را تنظیم نموده و این قابلیت را در حالت فعال قرار دهید. در این صورت اگر کنسول برای مدتی در حالت آماده‌به‌کار باقی بماند، سیستم به‌صورت خودکار اقدام به بستن آن می‌نماید.

- سیستم‌هایی که مدیران از آن‌ها جهت ارتباط مدیریتی با سایر سیستم‌ها استفاده می‌نمایند نیز باید امن نگه‌داشته شود؛ چراکه دسترسی به چنین سیستمی ممکن است دسترسی به چند سیستم مهم دیگر را امکان‌پذیر نماید. لذا مدیران حتماً باید سیستم‌های خود را در مکان‌های امن قرار داده و از Lock بودن سیستم در زمان ترک آن، اطمینان حاصل نمایند.

نصب سیستم با حداقل برنامه‌ها

سیستم را لاغر و معتدل نگه‌دارید؛ به این معنی که نرم‌افزارها و سرویس‌ها را به‌صورت حداقلی و متناسب با نیازها و انتظاراتی که از آن سیستم دارید، نصب نمایید. هر چه تعداد سرویس‌ها و برنامه‌های کاربردی نصب‌شده بر روی سیستم بیشتر باشد، آسیب‌پذیری‌ها و تهدیدات بالقوه بیشتری روی سیستم وجود داشته و بالطبع بیشتر در معرض خطر خواهد بود. برای نصب نرم‌افزارها و سرویس‌ها به موارد زیر دقت نمایید.

- هنگام نصب سیستم‌عامل، به نصب حداقل Componentها و Packageها بسنده کنید. تنها اجزای ضروری که برای پیاده‌سازی و اجرای سرویس‌های موردنظر، نیاز است نصب گردد. نگران نباشید! شما همیشه می‌توانید اجزای دیگری را نیز نصب کنید. لذا با خیال راحت حداقل‌ها را در نظر گرفته و در صورت نیاز، به‌راحتی برای نصب اجزا و قسمت‌های جدید اقدام نمایید.
- هرگونه سرویس اضافه در حال اجرا بر روی سیستم، که در حال حاضر از آن استفاده نمی‌شود را حذف نمایید.
- تمام پورت‌های TCP و UDP که از آن‌ها استفاده نمی‌شود را بر روی سیستم ببندید. مهاجمین برای نفوذ به سیستم از پورت‌های TCP/UDP نهایت استفاده را می‌برند.

رمزعبور مدیر سیستم

مدیران سیستم باید در انتخاب رمزعبور بسیار محتاط باشند. بهتر است مدیران موارد زیر را برای حفاظت از رمزعبور در نظر بگیرند:

- استفاده از رمزعبور طولانی. هر چه تعداد کاراکترهای رمزعبور بیشتر باشد، بهتر است؛ لذا تا جایی که ذهن مبارکتان اجازه می‌دهد و سیستم نیز پشتیبانی می‌کند، رمزعبور را طولانی نمایید. برای رمزعبور می‌توانید از یک جمله استفاده کنید تا هم طولانی باشد و هم به یادآوردن آن کار دشواری نباشد.

- به خاطر اینکه کلمات عبور را به یادآورید، آن را ساده انتخاب نکنید.
- رمزعبور باید طوری انتخاب شود که قابل حدس زدن نباشد. مثلاً از نام، فامیل، تاریخ تولد، شماره موبایل و اطلاعاتی از این قبیل برای رمزعبور استفاده نکنید.
- کلمات عبور نباید قابل پیدا کردن در دیکشنری باشند.
- هرگز رمزعبور را به‌صورت متن ساده بر روی کاغذ یا درون فایل کامپیوتری ذخیره نکنید. اگر به هر دلیلی نیاز دارید که آن را درون فایل ذخیره نمایید، حتماً از روش‌های رمزنگاری بهره ببرید.
- رمزعبور باید مخلوطی از حروف بزرگ و کوچک باشد.
- توصیه می‌شود از سیمبل و کاراکترهایی مثل @، $، # و غیره، در رمزعبور استفاده شود.
- اگر سیستم شما قابلیت تنظیم ویژگی‌های رمزعبور را دارد، حتماً آن را پیکربندی نمایید. برای مثال، این ویژگی‌ها می‌توانند حداقل و حداکثر عمر، طول رمز، زمان تغییر رمز، نوع کاراکترهای مورداستفاده در رمز و غیره، باشند.
- توصیه می‌شود مدیر سیستم برای کارهای روزمره خود از اکانت مدیریتی (مثل Administrator یا root) استفاده نکند. لزومی ندارد مدیر سیستم حتی زمانی که کارهای روزمره و غیر مدیریتی را انجام می‌دهد از طریق بالاترین اکانت به سیستمLogin نماید.

محول کردن وظایف مدیریتی

در برخی موارد، نیاز است که دستیاران (مثلاً اعضای تیم Helpdesk) دارای بعضی دسترسی‌های مدیریتی باشند. این نیاز مخصوصاً در سازمان‌های بزرگ که دارای تعداد زیادی سرور، سرویس و کلاینت هستند، بیشتر احساس می‌شود.

اعطای دسترسی مدیریتی باید محدود و بر اساس وظایف سازمانی کاربر، صورت پذیرد. البته توجه داشته باشید که در برخی تجهیزات یا سیستم‌عامل‌ها مثل یونیکس، انعطاف‌پذیری کمتری برای دادن اختیارات به کاربران وجود دارد. در این صورت باید برای محول کردن وظایف مدیریتی به افراد دیگر، دقت بیشتری به خرج دهید.

باید اختصاص وظایف مدیریتی بر اساس User ID همان کاربر انجام گیرد. به‌عبارت‌دیگر نباید یک نام کاربری مثل Superuser تعریف نموده و آن را در اختیار چند نفر قرار دهیم. زمانی که چند کاربر از یک نام کاربری استفاده نمایند، امکان پیگیری و بررسی کارهای انجام‌شده توسط افراد مختلف را نخواهید داشت. این امر زمانی منجر به فاجعه می‌شود که یک خرابکاری یا اقدام

نادرست توسط آن نام کاربری در سیستم روی داده باشد، و شما نتوانید فرد مقصر را شناسایی نمایید.

یکی از وظایف مهمی که مدیر سیستم پس از اعطای وظایف مدیریتی بر عهده دارد؛ نظارت مستمر بر کارهای انجام‌شده توسط دستیاران است. همواره باید وقایع ثبت‌شده بر اساس فعالیت‌های دستیاران، موردبررسی و ارزیابی مدیر محترم قرار بگیرد.

رمزعبور کاربران

یکی از اصلی‌ترین اقدامات امنیتی برای جلوگیری از دسترسی غیرمجاز، طراحی و اجرای یک سیاست خوب امنیتی برای رمزعبور است. اگر صرفاً این سیاست امنیتی به‌صورت بخشنامه یا دستور مدیریتی باشد که کاربر آن را در کارتابل اتوماسیون خود دریافت می‌کند، قطعاً موردتوجه بخش عظیمی از افراد قرار نخواهد گرفت. لذا علاوه بر دستور مدیریت ارشد سازمان، مدیر سیستم نیز باید این سیاست را بر روی سیستم‌ها به‌صورت اجباری برای کاربران قرار دهد.

طراحی سیاست رمزعبور باید بر اساس سیاست‌های امنیتی سازمان، درجه اهمیت دیتای مورد دسترس کاربران، و مواردی از این قبیل باشد. البته نباید سیاست انتخاب رمزعبور به‌قدری سخت باشد که اشک کاربران را درآورد. این کار علاوه بر نارضایتی آن‌ها موجب افزایش سربار مدیریتی شما هم خواهد شد؛ چراکه کاربران دائماً از شما درخواست Reset کردن رمزعبور خواهند نمود. پس طوری سیاست را طراحی و اجبار نمایید که ضمن حصول نتیجه دلخواه، باعث نارضایتی کاربران نیز نشود.

موارد زیر ازجمله پارامترهای مهمی است که باید در سیاست‌گذاری رمزعبور کاربران، مدنظر قرار داده شود:

- **طول عمر رمزعبور**

این پارامتر طول عمر یک رمزعبور را مشخص کرده و پس‌ازآن کاربر را مجبور به تغییر رمزعبور می‌نماید. تغییر دوره‌ای رمزعبور، برای حفظ امنیت بسیار مهم است.

- **حداقل طول**

اجبار به تعیین یک رمزعبور با حداقل تعداد کاراکتر مشخص‌شده. پیشنهاد می‌کنم تعداد کاراکتر را روی ۸ عدد تنظیم نمایید.

- **کلمات غیر فرهنگ لغت**

درصورتی‌که سیستم‌عامل از این ویژگی پشتیبانی کند، کاربر را مجبور به انتخاب کلمه عبوری می‌نماید که در فرهنگ لغت وجود نداشته باشد.

- **استفاده مجدد از رمزعبور قبلی**

مشخص‌کننده تعداد دفعات تعیین رمزعبور است که کاربر پس‌ازآن می‌تواند مجدداً کلمات عبور قبلی خود را به‌عنوان رمز جدید انتخاب کند.

- **رمزعبور جدید**

در برخی از سیستم‌عامل‌ها این امکان به شما داده می‌شود که مشخص نمایید حداقل چه تعداد کاراکتر رمزعبور جدید باید با رمزعبور قبلی متفاوت باشد.

سیستم کاربر

اگر کاربر سیستم خود را بدون خاموش یا قفل (Lock) نمودن، ترک نماید، ممکن است مورد سوءاستفاده افراد خرابکار قرار گیرد. برای امنیت سیستم کاربر، موارد مهم زیر باید در نظر گرفته شود.

- از ویژگی قفل خودکار استفاده شود. در این صورت اگر هیچ فعالیتی از طریق تجهیزات ورودی (مثل موس و صفحه‌کلید) انجام نشود، پس از مدت‌زمانی که کاربر یا مدیر مشخص نموده، سیستم به‌صورت خودکار قفل خواهد شد.[۱]
- قفل سیستم حتماً باید با رمزعبور باز شود. به صورتی که در زمان بازگشت، کاربر مجدداً برای ورود به سیستم، باید نام کاربری و رمزعبور را وارد نماید.
- درصورتی‌که چندبار نام کاربری یا رمزعبور اشتباه وارد شود، باید سیستم قفل‌شده و فقط پس از گذشت مدت‌زمان مشخص یا توسط مدیر سیستم، امکان باز شدن قفل وجود داشته باشد. این ویژگی یکی از حداقل‌هایی است که برای جلوگیری از حملات جستجوی فراگیر (Brute force) یا حدس زدن رمزعبور، باید انجام پذیرد.
- بهتر است کاربر اجازه نصب هر نرم‌افزاری را بر روی سیستم خود نداشته باشد. درصورتی‌که کاربر امکان نصب برنامه داشته باشد، ممکن است از روی ناآگاهی، اجازه نصب برنامه‌ای را بر روی سیستم خود بدهد که شامل کدمخرب (مثل Backdoor) است.

البته توجه داشته باشید برخی از بدافزارها بدون نیاز به اجازه کلاینت و تنها با سوءاستفاده از سطح دسترسی کاربر، امکان نصب بر روی سیستم را دارند.

- بهتر است کاربر اجازه اتصال هرگونه تجهیزاتی (مثل گوشی تلفن هوشمند و هارد اکسترنال) را به پورت‌های ورودی دستگاه نداشته باشد.

---

[۱] به مدت‌زمانی که هیچ فعالیتی روی سیستم انجام نمی‌پذیرد، Idle time می‌گویند.

محدود کردن کاربر

ممکن است کاربر به دلیل عدم آگاهی یا اشتباه انسانی، باعث خرابکاری در سیستم یا شبکه گردد. لذا هر چه کاربر با توجه به وظایف سازمانی خود محدودتر باشد، خرابکاری‌های اتفاق افتاده، وسعت کمتری را تحت‌الشعاع خود قرار خواهد داد. کاربر باید به همان منابعی دسترسی داشته باشد که در راستای انجام وظایف سازمانی او نیاز است.

برای محدود کردن کاربر، بهتر است موارد زیر در نظر گرفته شود.

- اگر کاربر شما، یک کاربر راه دور است (یعنی به‌طور مستقیم به سیستم یا کنسول وصل نمی‌شود)، باید نسبت به آن محتاط‌تر عمل نمایید. در این صورت باید سیستم را طوری پیکربندی کنید که فقط برای آدرس‌های IP شناخته‌شده امکان اتصال را فراهم آورد. همچنین برای افزایش سطح امنیت، راه‌های دیگری مثل استفاده از سرور RADIUS، اتصالات VPN و فایروال، نیز وجود دارد.
- اگر اتصال راه‌دور از طریق خطوط تلفن برقرار می‌شود، می‌توان اتصال کاربر را محدود به شماره‌های خاص نمود.
- دسترسی کاربران به منابع سیستم و شبکه را به‌اندازه موردنیاز آن‌ها جهت انجام امورات محوله، محدود نمایید.

آموزش کاربران

مدیران سیستم اولین سطحی هستند که باید به کاربران سازمان خدمات IT ارائه دهند. البته در سازمان‌های بزرگ معمولاً افراد Helpdesk به‌طور مستقیم با کاربران در ارتباط هستند، ولی بازهم این خدمات تحت نظر مدیر سیستم باید انجام پذیرد. با توجه به این موضوع، هر چه سواد و اطلاعات کامپیوتری کاربران سازمان ارتقاء یابد، به همان میزان از سربار مدیریتی و زحمات Helpdesk کمتر خواهد شد. این آموزش می‌تواند نحوه درست استفاده از سیستم تا موارد امنیتی که باید توسط کاربر انجام پذیرد را شامل شود. ازجمله موارد امنیتی که می‌توان به آگاهی کاربران رساند، می‌توان به موارد زیر اشاره کرد:

- رمزعبور را قدرتمند انتخاب نمایید.
- رمزعبور را با دیگران به اشتراک نگذارید.
- رمزعبور را بر روی کاغذ و به‌صورت متن ساده یادداشت نکنید.
- رمزعبور خود را به‌صورت متناوب تغییر دهید.
- برای رمزعبور از کلمات غیر دیکشنری استفاده نمایید.
- زمان ترک میز کار خود، حتماً سیستم خود را Lock نمایید.

- اطلاعات مهم را از طریق کانال‌های ناامن، تبادل نکنید.
- به افراد بیرون از سازمان اعتماد نکنید. (برای جلوگیری از حملات مهندسی اجتماعی)
- از نصب نرم‌افزارهای ناشناخته خودداری نمایید.
- نامه‌های پست الکترونیک که دارای ارسال‌کننده ناشناخته است را باز نکنید.
- از اتصال تجهیزات ذخیره‌ساز همراه به سیستم خودداری نمایید.
- اطلاعات مهم را بر روی مکانی که برای همه به اشتراک گذاشته شد، قرار ندهید.

به‌روزرسانی سیستم‌ها

تمام نرم‌افزارها و تجهیزاتی کامپیوتری دارای حفره‌ها و ایرادات برنامه‌نویسی هستند، که پس از ارائه نسخه تجاری آن‌ها، تازه خودنمایی می‌کنند. حتی شرکت‌های بسیار بزرگ که دارای لابراتوارهای قدرتمند بوده و قبل از نسخه نهایی، نسخه‌های آزمایشی را بین متخصصین پخش می‌کنند، بازهم دارای ایراداتی هستند که پس از ارائه نهایی به بازار، مشخص می‌شوند. بعضی از این ایرادات دارای اهمیت خاصی نیستند، ولی برخی دیگر ممکن است منجر به شکست سیستم و از دست رفتن اطلاعات شوند. یکی از تفریحات سالم دوستان هکر نیز استفاده از همین حفره‌ها و ایرادات برنامه‌نویسی در سیستم‌ها است.

تولیدکنندگان محصولات، معمولاً ایرادات سیستم را با ارائه Patchها و Service Packها، رفع می‌نمایند. برای ارتقاء کلی محصول نیز، اقدام به ارائه یک نسخه جدید می‌کنند.

یکی از وظایف مهم مدیر سیستم، به‌روزرسانی بر اساس آخرین Patchها و Service Packها است. درصورتی‌که تولیدکننده توصیه نماید که کلاً نسخه محصول موردنظر را ارتقاء دهید، حتماً باید به آن توصیه عمل نمایید.

برخی از محصولات برای به‌روزرسانی دارای ابزارها و سرویس‌هایی هستند که این وظیفه را به‌صورت خودکار انجام می‌دهند. لذا توصیه می‌شود اگر از این امکانات استفاده می‌کنید، حتماً بر روی عملکرد صحیح آن نظارت داشته باشید.

یک نکته مهم را حتماً مدنظر قرار دهید؛ همان‌طور که خود محصولات دارای ایراداتی هستند، لذا اصلاً بعید نیست که Packageهای به‌روزرسانی نیز دارای حفره‌ها و اشکالاتی باشند. همچنین ممکن است به‌روزرسانی جدید باعث بروز اختلال در سیستم شود. لذا توصیه می‌کنم حتماً Patchها و Service Packها و البته نسخه جدید محصولات را ابتدا در یک محیط کوچک آزمایشگاهی، امتحان نموده و سپس به‌صورت عملیاتی در کل سازمان به اجرا درآورید.

تعلل در به‌روزرسانی[۱] به بهانه آزمایش آن‌ها، نباید زیاد به طول انجامد. چراکه برخی از حفره‌های شناخته‌شده به‌قدری خطرناک هستند که ممکن است تا زمان به‌روزرسانی، مورداستفاده مهاجمین قرارگرفته و سیستم‌های سازمان را نابود کرده باشند!



به‌روزرسانی یک تیغ دو لبه است؛
اعمال بدون آزمایش آن، همان‌قدر ممکن است به سیستم شما آسیب برساند؛
که تعلل در اعمال آن.

آزمون آسیب‌پذیری

پیشگیری بهتر از درمان است. اگر شما به‌عنوان مدیر سیستم، از آسیب‌پذیری‌های سیستم خود مطلع باشید، می‌توانید قبل از هرگونه اتفاق ناگواری، ضعف‌های موجود را برطرف کرده و سیستم خود را امن نگه‌دارید.

برای آزمون آسیب‌پذیری، روش‌ها و ابزارهای متنوعی وجود دارد. یکی از راه‌های آزمون آسیب‌پذیری، برون‌سپاری تست نفوذ یا Penetration Test به گروه‌ها و شرکت‌هایی است که در این زمینه دارای تخصص بالایی هستند. به همان میزان که این برون‌سپاری می‌تواند یک آزمون جامع را فراهم آورد، هزینه انجام آن نیز ممکن است چشمگیر باشد. لذا اگر شما دارای سازمان کوچکی هستید که هزینه برون‌سپاری به نسبت ارزش دیتای سازمان قابل توجیه نیست، می‌توانید از ابزار رایگانی که در دسترس است، برای آزمون آسیب‌پذیری استفاده نمایید.

یک توصیه نیز به مدیران سازمان‌های بزرگی که قصد برون‌سپاری آزمون آسیب‌پذیری را دارند، می‌کنم؛ با توجه به هزینه مالی و زمانی اجرای آزمون آسیب‌پذیری یا تست نفوذ، حتماً قبل از شروع آن، از مشاوره یک کارشناس خبره امنیت اطلاعات بهره ببرید.

زمانی تست آسیب‌پذیری ارزش واقعی خود را نمایش می‌دهد که شما تمام موارد امنیتی را در سازمان خود اجرا کرده باشید. در غیر این صورت پس از انجام آزمون، یک لیست بلندی از ایرادات به شما ارائه می‌شود که به دلیل تعدد مشکلات و نواقص مشاهده‌شده، ممکن است حفره‌ها و آسیب‌پذیری‌های مهم‌تر و حیاتی‌تر، ازنظر گروه آزمایش‌کننده و یا مدیران سیستم، مغفول بماند. حتی بعضی از ایرادات پیکربندی باعث می‌شود که گروه مجری آزمون، اصلاً متوجه آسیب‌پذیری اصلی که بر روی سیستم وجود دارد، نشود.

---

[۱] به‌روزرسانی‌های از قبیل آنتی‌ویروس، فایروال، IPS ، منظور این بخش نیست. بلکه به‌روزرسانی مدنظر این بخش، سیستم‌عامل‌ها ، Firmware ها و نسخه‌های اصلی محصولات است.

اما در مقابل هر چه شما موارد امنیتی موردنظر یک کارشناس امنیت اطلاعات را بهتر و کامل‌تر انجام داده باشید، انجام آزمون آسیب‌پذیری سخت‌تر شده و پولی که گروه آزمایش‌کننده دریافت می‌کنند، حلال‌تر خواهد بود. درنهایت نیز لیستی در اختیار شما قرار می‌گیرد که تمام موارد آن ارزشمند و قابل‌توجه هستند.

مانیتورینگ سیستم

اکثر سیستم‌ها دارای امکانات وقایع‌نگاری هستند. وقایع‌نگاری اتفاقات انجام‌شده بر روی سیستم، می‌تواند کمک بزرگی در زمان اشکال‌یابی و رفع ایرادات باشد. بسته به سیستم‌عامل محصول مورداستفاده، سطح متفاوتی از وقایع‌نگاری برای شما فراهم می‌گردد. درصورتی‌که احساس می‌کنید به چیزی بیشتر از امکانات موجود نیاز دارید، می‌توانید از سرویس‌هایی که مختص واقعه‌نگاری هستند، استفاده نمایید.

هر چه تعداد سرویس‌ها و سیستم‌ها بیشتر باشد، مقدار Log تولیدشده نیز بیشتر خواهد بود. بررسی این‌همه Log می‌تواند بسیار وقت‌گیر و کسل‌کننده باشد. لذا برای بررسی Logهای حجیم، می‌توان از نرم‌افزارهای تجزیه‌وتحلیل (Analyzer) استفاده نمود. این نرم‌افزارها می‌توانند در تقسیم‌بندی Logها، ارزشمند و مفید واقع شوند.

در برخی از سیستم‌ها و نرم‌افزارهای واقعه‌نگاری، امکان اطلاع‌رسانی وقایع از طریق آژیر، پیامک، ایمیل و غیره، وجود دارد؛ لذا می‌توان سیستم را طوری پیکربندی نمود که در صورت بروز اتفاقات ناگوار! همان لحظه مدیران سیستم را باخبر سازد.

پیشنهاد می‌کنم، حتی اگر اتفاق خاصی که منجر به آلارم شود، رخ نداده باشد، بازهم Logها را به‌صورت دوره‌ای مورد نظارت و بررسی قرار دهید. چراکه بعضی مواقع ممکن است اتفاقاتی رخ‌داده باشند، که تا قبل از آن برای شما اهمیت نداشته و یا به فکر مبارکتان خطور نکرده که بخواهید برای آن‌ها آلارم تعیین نمایید.

مستندسازی پیکربندی

باید تمام پیکربندی‌ها، تغییرات و تنظیمات انجام‌شده بر روی نرم‌افزارها و سخت‌افزارهای موجود در سازمان را در قالب اسناد فنی، ثبت و نگهداری نمایید. این اسناد در زمان بازیابی فاجعه[1] بسیار مفید خواهد بود. همچنین برای ارائه آمار و اطلاعات به مدیران ارشد سازمان، کمک خوبی برای مدیر سیستم است. در ضمن زمانی که می‌خواهید یک محصول نرم‌افزاری یا

[1] Disaster Recovery

سخت‌افزاری جدید را به سازمان اضافه نمایید که بر روی سایر تجهیزات تأثیرگذار است، این مستندات می‌تواند باعث تسریع کار شما در تصمیم‌گیری و پیاده‌سازی گردد.

و اما مهم‌ترین استفاده از مستندات، زمانی است که مدیر سیستم می‌خواهد سازمان یا احتمالاً دنیا را ترک نماید. در این صورت، هم تحویل وضع موجود به مدیر جدید بهتر انجام می‌شود، و هم مدیر جدید برای پیشرفت کارها می‌تواند از آن مستندات نهایت استفاده را ببرد.

پشتیبان‌گیری و بازیابی فاجعه

حتی باوجود داشتن سخت‌افزار و نرم‌افزار قابل‌اطمینان، و داشتن یک مدیر سیستم خوب، بازهم نمی‌توانید از عدم بروز شکست در سیستم اطمینان داشته باشید. شکست ممکن است بر اساس حمله، خطای انسانی، خرابی فیزیکی و بلایای طبیعی، پدید آید. لذا یک مدیر سیستم خوب و البته عاقل، مدیری است که همواره از دارایی‌های اطلاعاتی سازمان یک نسخه پشتیبان تهیه نماید. انتخاب نوع، نحوه و ابزار پشتیبان‌گیری وابسته به گستردگی سازمان، اطلاعات باارزش، نوع کسب‌وکار، و هزینه مالی آن خواهد بود. البته توجه داشته باشید که محل نگهداری نسخه‌های پشتیبان نیز به‌اندازه عمل پشتیبان گیری دارای اهمیت است.

اگر سازمان شما بزرگ و دارای اطلاعات بسیار مهم بوده و حیات سازمان کاملاً وابسته به دارایی‌های اطلاعاتی است؛ باید علاوه بر گرفتن نسخه پشتیبان، آن را در یک محل فیزیکی متفاوت که ازنظر جغرافیایی برخی از مشکلات احتمالی مثل جنگ، سیل و زلزله را نداشته باشد، نگهداری نمایید. این سایت که معمولاً offsite یا Disaster Recovery Site نامیده می‌شود، می‌تواند به‌صورت آنلاین و یا آفلاین از اطلاعات پشتیبان گرفته و ذخیره نماید.

باید موارد زیر را برای طراحی سیاست پشتیبان‌گیری، مدنظر قرار دهید.

- بهترین زمان برای عملیات پشتیبان‌گیری چه وقت است؟
- از چه مقدار اطلاعات باید پشتیبان گرفته شود؟
- ذخیره‌سازی داخل سایت انجام می‌شود؟ یا خارج سایت؟ یا هر دو؟
- بر روی چه رسانه‌ای (مدیا) باید پشتیبان گرفته شود؟
- چه مدت‌زمانی باید از آن‌ها نگهداری شود؟
- چه امنیتی باید برای آن‌ها در نظر گرفته شود؟
- به چه نحو باید از درستی انجام عملیات پشتیبان‌گیری، اطمینان حاصل نمود؟

نحوه گرفتن پشتیبان، آرشیو، نگهداری و بازیابی آن‌ها باید در قالب مستندات فنی آماده و در اختیار متصدی مربوطه قرار گیرد.

آنتی‌ویروس

معمولاً در سازمان‌های بزرگ، مدیر سیستم و مدیر شبکه دو مسئولیت جداگانه است. در این‌صورت برقراری امنیت شبکه بر عهده مدیر سیستم نیست؛ اما با توجه به مسئولیت امنیت سیستم‌ها، آنتی‌ویروس نیز جزء وظایف مدیر سیستم محسوب خواهد شد. البته ممکن است با توجه به تحت تأثیر قرار گرفتن شبکه توسط برخی از ویروس‌ها، بهتر باشد که مدیران سیستم و شبکه، با همکاری یکدیگر اقدام به برقراری آنتی‌ویروس نمایند.

انتخاب یک آنتی‌ویروس که ترجیحاً تحت شبکه بوده و دارای یک کنسول مدیریت مرکزی باشد، برای یک سازمان امری حیاتی محسوب می‌شود. مدیر سیستم پس از انتخاب، نصب، اجرا و پیکربندی آنتی‌ویروس، باید به‌صورت مرتب بر به‌روزرسانی آن نظارت داشته باشد؛ چراکه درصد بالایی از ارزش و کارایی آنتی‌ویروس به اطلاعات به‌روز آن خواهد بود.

همچنین مدیر سیستم باید یک سیاست امنیتی را طراحی و اجبار نماید که هیچ سیستمی بدون آنتی‌ویروس در سازمان وجود نداشته باشد.

☑ مبحث دوم

سطوح امنیتی در سیستم‌عامل

امروزه هر سیستم کامپیوتری مدرن، اعم از سرورهای شبکه، دسکتاپ‌های کاربران، لپ‌تاپ‌ها و تجهیزات همراه (مثل تلفن همراه هوشمند، تبلت و غیره)، دارای یک نرم‌افزار اصلی است که آن را کرنل یا سیستم‌عامل (Operating System) می‌نامند. سیستم‌عامل در بالای سخت‌افزار ماشین قرارگرفته و وظایف زیر را انجام می‌دهد:

- فراهم آوردن یک رابط کاربری برای تعامل بین کاربر و سخت‌افزار.
- تخصیص منابع سیستم؛ مثل CPU، حافظه، درایور و پورت‌های ارتباطی.
- نظارت بر اجرای برنامه‌های کاربردی در سیستم.
- تخصیص و نظارت بر روی تجهیزات ورودی و خروجی.

سیستم‌عامل بر روی سخت‌افزار سیستم و همچنین نرم‌افزارهایی که بر روی آن نصب شده‌اند، دسترسی ویژه دارد. درصورتی‌که یک هکر بتواند دسترسی به سیستم‌عامل را به دست بیاورد، تمام نرم‌افزارها و سخت‌افزار سیستم مورد تهدید قرار می‌گیرند. حتی ادامه کار نرم‌افزارهایی که دارای مکانیسم احرازهویت مستقل هستند، وابسته به سیستم‌عاملی است که بر روی آن نصب شده‌اند.

برخی از سیستم‌عامل‌ها مثل مایکروسافت ویندوز، یونیکس، لینوکس و مک، به‌صورت تجاری عرضه‌شده و از محبوبیت خاصی نیز در بین کاربران بهره‌مند هستند؛ که البته هرکدام از آن‌ها مستندات خاص خود را برای مقاوم‌سازی[1] و امنیت دارند. اما در این مبحث به ذکر سطوح امنیتی که بین تمام سیستم‌عامل‌ها مشترک است، می‌پردازیم.

بر اساس استاندارد معیارهای ارزیابی سیستم کامپیوتر مورد اعتماد[2]، که توسط دولت ایالات‌متحده ارائه‌شده است؛[3] سطوح امنیتی در سیستم‌عامل‌های تجاری، به چهار بخش[4] اصلی (A, B, C, D) تقسیم گردیده است. هر یک از این بخش‌ها نیز ممکن است در زیرمجموعه خود شامل یک یا چند کلاس باشند.

---

[1] Hardening
[2] Trusted Computer System Evaluation Criteria (TCSEC)
[3] http://csrc.nist.gov/publications/history/dod85.pdf
[4] Division

در تشریح بخش‌ها و کلاس‌ها، دو اصطلاح مهم مورداستفاده قرار می‌گیرد:

۱. **Trusted Computing Base (TCB)**
Trusted Computing Base که به‌اختصار آن را TCB می‌نامیم؛ مکانیسم کلی حفاظت از یک سیستم کامپیوتری و مسئول اجرای سیاست امنیتی است. اجزای TCB که ممکن است ترکیبی از یک یا چند جزء سخت‌افزاری، Firmware و نرم‌افزاری باشد؛ برای امنیت سیستم حیاتی بوده و در صورت بروز اختلال در هر یک از آن‌ها، ممکن است امنیت سیستم در معرض خطر قرار گیرد.

۲. **Automatic Data Processing (ADP)**
Automatic Data Processing که به‌اختصار آن را ADP می‌نامیم؛ شامل پیکربندی سخت‌افزار، Firmware و نرم‌افزار، به‌منظور طبقه‌بندی، مرتب‌سازی، محاسبه، خلاصه‌سازی، ارسال، دریافت، ذخیره‌سازی و بازیابی اطلاعات، با حداقل دخالت انسانی، است.

در ادامه به شرح مختصر هر یک از بخش‌ها و کلاس‌های زیرمجموعه TCSEC می‌پردازیم. البته توجه داشته باشید که در این مبحث، بخش‌ها و کلاس‌ها به‌صورت مختصر و مفید بیان‌شده، اما ساختار سند TCSEC حفظ گردیده است.

بخش D: محافظت حداقلی (Minimal Protection)

این بخش تنها شامل یک کلاس است؛ برای کسانی که سیستم‌های خود را مورد ارزیابی قرار داده‌اند اما موفق به دستیابی به الزامات موردنیاز برای یک کلاس ارزیابی بالاتر نشده‌اند.

بخش C: محافظت اختیاری (Discretionary Protection)

کلاس‌های این بخش محافظت اختیاری (Need-to-know)[۱] را ارائه نموده و با گنجاندن قابلیت‌های حسابرسی (بازرسی)، امکان پاسخگویی افراد و اقدامات انجام‌شده توسط آن‌ها را فراهم آورده است.

- **کلاس C1: Discretionary Security Protection**

اساس کار کلاس C1، در جدایی کاربران از داده‌ها است. در این صورت برای افراد در دسترسی به دیتا محدودیت به وجود آمده و هر کاربری نمی‌تواند از روی عمد و یا

[۱] هر کس به همان مقدار از اطلاعات که نیاز دارد، دسترسی داشته باشد. در مورد Need-to-Know در بخش اول کتاب توضیح داده‌شده است.

تصادفی، به هر دیتایی دسترسی داشته باشد. موارد زیر، حداقل الزامات برای اختصاص رتبه C1 به یک سیستم است:

۱. **سیاست امنیتی (Security Policy)**

i. Discretionary Access Control (DAC)

TCB باید کنترل دسترسی بین نام کاربران و نام اشیاء (مثل فایل و برنامه) را در یک سیستم ADP، تعریف نماید. مکانیسم‌های اجرایی باید به کاربران اجازه دهند که بر روی اشیایی که مالک آن هستند، اقدام به اعطای مجوز دسترسی به افراد خاص، گروه‌ها و یا هر دو، نمایند.

۲. **پاسخگویی (Accountability)**

i. Identification and Authentication

قبل از هرگونه اقدامی، TCB باید کاربران را شناسایی کند. بنابراین TCB باید برای تأیید هویت کاربر، از یک مکانیسم محافظتی مثل رمزعبور، استفاده نماید. TCB باید از داده‌ها در برابر دسترسی کاربر غیرمجاز محافظت کند.

۳. **تضمین (Assurance)**

i. Operational Assurance

a. System Architecture

TCB باید از دامنه تحت کنترل خود، در برابر دخالت یا دست‌کاری خارجی، محافظت به‌عمل‌آورد.

b. System Integrity

برای اطمینان از عملکرد صحیح اجزای TCB، باید سخت‌افزارها و نرم‌افزارها به‌صورت دوره‌ای، مورد بررسی قرار گیرند.

ii. Life-Cycle Assurance

a. Security Testing

مکانیسم‌های امنیتی در یک سیستم ADP، باید آزمایش شوند، تا اطمینان حاصل شود که هیچ راهی برای دسترسی غیرمجاز وجود ندارد.

۴. **مستندسازی (Documentation)**

i. Security Features User's Guide

یک راهنمای خلاصه برای توصیف مکانیسم‌های امنیتی فراهم‌شده توسط TCB؛ و نحوه استفاده از آن‌ها.

ii. Trusted Facility Manual
یک راهنما برای مدیر سیستم ADP، که ارائه‌دهنده هشدارهایی در مورد توابع و امتیازاتی است که باید در زمان اجرای تسهیلات امنیتی، کنترل شوند.

iii. Test Documentation
توصیف مراحل آزمون، مکانیسم‌های امنیتی آزمایش‌شده و نتایج حاصل از این آزمایش، است.

iv. Design Documentation
فراهم آورنده توصیفی از چرایی و استفاده از یک حفاظت است.

- **کلاس C2: Controlled Access Protection**

در این کلاس، سیستم‌ها کنترل دسترسی اختیاری را به‌صورت جزئی‌تر نسبت به کلاس C1، اعمال می‌نمایند. به همین دلیل امکان پاسخگویی هریک از کاربران در قبال فعالیت‌هایشان از طریق روش‌های ورود به سیستم، بازرسی رخدادهای امنیتی مربوطه و جداسازی منابع، فراهم می‌گردد. موارد زیر، حداقل الزامات برای اختصاص رتبه C2 به یک سیستم است:

۱. **سیاست امنیتی (Security Policy)**

i. Discretionary Access Control (DAC)
TCB باید کنترل دسترسی بین نام کاربران و نام اشیاء (مثل فایل و برنامه) را در یک سیستم ADP، تعریف نماید. مکانیسم‌هایی اجرایی باید به کاربران اجازه دهند که بر روی اشیایی که مالک آن هستند، اقدام به اعطای مجوز دسترسی به افراد خاص، گروه‌ها و یا هر دو، نمایند. این کنترل دسترسی باید بتواند برای یک کاربر خاص اجازه یا منع دسترسی صادر نماید. صدور اجازه دسترسی برای یک کاربری که به یک شیء دسترسی ندارد، فقط باید توسط کاربر مجاز صورت پذیرد.

ii. Object Reuse
برای استفاده مجدد از اشیاء استفاده‌نشده، باید تمام مجوزهای قبلی آن‌ها لغو گردد.

۲. **پاسخگویی (Accountability)**

i. Identification and Authentication

قبل از هرگونه اقدامی، TCB باید کاربران را شناسایی کند. بنابراین TCB باید برای تأیید هویت کاربر، از یک مکانیسم محافظتی مثل رمزعبور، استفاده نماید. TCB باید از داده‌ها در برابر دسترسی کاربر غیرمجاز محافظت کند.

ii. Audit

TCB باید قادر به ایجاد و نگهداری اطلاعات مربوط به بازرسی دسترسی به اشیائی که از آن‌ها محافظت می‌شود، بوده؛ و همچنین باید از آن اطلاعات بازرسی در مقابل تغییر، دسترسی غیرمجاز و یا تخریب، حفاظت نماید. داده‌های مربوط به Auditing باید توسط TCB در برابر دسترسی غیرمجاز محافظت شوند. همچنین TCB باید توانایی ضبط انواع حوادث موردنیاز جهت بازرسی (ازجمله استفاده از مکانیسم‌های احرازهویت و شناسایی، استفاده و حذف اشیاء و دیگر حوادث امنیتی)، را داشته باشد.

۳. تضمین (Assurance)

i. Operational Assurance

a. System Architecture

TCB باید از دامنه تحت کنترل خود، در برابر دخالت یا دست‌کاری خارجی، محافظت به عمل آورد.

b. System Integrity

برای اطمینان از عملکرد صحیح اجزای TCB، باید سخت‌افزارها و نرم‌افزارها به‌صورت دوره‌ای، مورد بررسی قرار گیرند.

ii. Life-Cycle Assurance

a. Security Testing

مکانیسم‌های امنیتی در یک سیستم ADP، باید آزمایش شوند، تا اطمینان حاصل شود که هیچ راهی برای دسترسی غیرمجاز وجود ندارد.

۴. مستندسازی (Documentation)

i. Security Features User's Guide
ii. Trusted Facility Manual
iii. Test Documentation
iv. Design Documentation

توضیح موارد فوق، همانند توضیحات ارائه‌شده در بند مستندسازی کلاس C1 است.

بخش B: محافظت اجباری (Mandatory Protection)

محافظت اجباری، مفهومی از TCB است که صحت برچسب‌های حساسیت[1] را حفظ کرده، و از آن‌ها برای اعمال مجموعه‌ای از قوانین اجباری کنترل دسترسی موردنیاز در این بخش، استفاده می‌نماید. سیستم‌ها در این بخش باید حامل برچسب‌های حساسیت به‌وسیله ساختار داده اصلی، در سیستم باشند. همچنین در این بخش توسعه‌دهنده سیستم، یک مدل سیاست امنیتی[2] بر اساس TCB فراهم می‌آورد.

- **کلاس B1: Labeled Security Protection**

نیازهای سیستم‌های کلاس B1، شبیه به نیازهای کلاس C2 است؛ علاوه بر این، باید یک شرح رسمی از مدل سیاست امنیتی، برچسب داده، و کنترل دسترسی اجباری بر روی نام افراد و اشیاء، وجود داشته باشد. موارد زیر، حداقل الزامات برای اختصاص رتبه B1 به یک سیستم است:[3]

۱. **سیاست امنیتی (Security Policy)**

i. Discretionary Access Control (DAC)

ii. Object Reuse

iii. Labels

برچسب‌های حساسیت مربوط به هر فرد و شیء ذخیره‌شده تحت این کنترل (به‌عنوان‌مثال، فرآیند، فایل، بخش و دستگاه)، باید توسط TCB نگهداری شوند. از این برچسب‌ها به‌عنوان پایه‌ای برای تصمیم‌گیری کنترل دسترسی اجباری استفاده می‌شود. برای داده‌های فاقد برچسب، TCB باید توسط یک درخواست، از کاربر مجاز بخواهد که سطح امنیتی آن را مشخص نماید. این‌گونه فعالیت‌ها باید برای TCB قابل بازرسی (ممیزی) باشد.

a. Label Integrity

برچسب حساسیت باید عیناً سطح امنیت یک Subject خاص یا اشیایی (Object) که با آن در ارتباط هستند را ارائه نماید.

---

[1] Sensitivity Labels

[2] Security Policy Model

[3] از تشریح موارد مشابه با کلاس C2، صرف‌نظر شده است.

b. Exportation of Labeled Information

TCB باید هر کانال ارتباطی و دستگاه I/O را به‌عنوان Single-Level، Multi-Level[1] یا هر دو تعیین نماید. هرگونه تغییر در تعیین صورت گرفته، باید به‌صورت دستی انجام‌شده و توسط TCB نیز قابل ممیزی باشد. TCB باید قادر به حفظ و ممیزی هرگونه تغییر در سطح امنیتی یا سطوح مرتبط با کانال ارتباطی و دستگاه I/O باشد.

iv. Mandatory Access Control (MAC)

TCB باید یک سیاست کنترل دسترسی اجباری برای تمام Subjectها و Objectهای ذخیره‌شده تحت کنترل خود (مانند پردازش‌ها، فایل‌ها و تجهیزات)، اجرا نماید. باید به این Subjectها و Objectها برچسب حساسیت که ترکیبی از سطوح طبقه‌بندی سلسله مراتبی و غیر سلسله مراتبی است، الصاق شود. از این برچسب‌ها به‌عنوان پایه‌ای جهت تصمیم‌گیری کنترل دسترسی اجباری استفاده می‌شود. TCB باید قادر به پشتیبانی از دو یا چند سطح امنیتی از این قبیل باشد.

۲. **پاسخگویی (Accountability)**

i. Identification and Authentication
ii. Audit

۳. **تضمین (Assurance)**

i. Operational Assurance
   a. System Architecture
   b. System Integrity
ii. Life-Cycle Assurance
   a. Security Testing
   b. Design Specification and Verification

برای اثبات عدم تناقض، باید یک مدل رسمی یا غیررسمی از سیاست امنیتی پشتیبانی شده توسط TCB، در طول حیاط سیستم ADP نگهداری شود.

۴. **مستندسازی (Documentation)**

i. Security Features User's Guide
ii. Trusted Facility Manual
iii. Test Documentation
iv. Design Documentation

[1] در Multi-Level از ورود جریان اطلاعات از یک سطح بالای امنیتی به سطح پایین امنیتی جلوگیری به عمل می‌آید.

- **کلاس B2: Structured Protection**

در کلاس B2، کنترل‌های اختیاری و اجباری موجود در کلاس B1، بر روی تمام Subjectها و Objectهایی که در سیستم ADP وجود دارند، گسترش داده‌شده است. علاوه بر این، در کلاس B2 به کانال‌های پنهان (Covert Channel) نیز پرداخته‌شده است.

کانال پنهان یا Covert Channel، راهی است که مهاجم از طریق آن می‌تواند دیتای موردنظر خود را به‌صورت غیرمجاز به دست آورد. کانال پنهان دارای دو نوع زیر است:

- **Covert storage channels**
  در این حالت به‌واسطه فضای ذخیره‌سازی مشترک، با مهاجم ارتباط برقرار می‌شود.
- **Covert timing channels**
  در این حالت با تغییر مدت‌زمان پاسخ، با مهاجم ارتباط برقرار می‌شود.



TCB باید به‌دقت در داخل عناصر protection-critical و non-protection-critical بنا شود. رابط TCB به‌خوبی تعریف‌شده و طراحی و اجرای TCB، آن را قادر به انجام آزمون دقیق‌تر و بررسی کامل‌تر می‌سازد. در این کلاس، مکانیسم‌های احرازهویت تقویت شده، مدیریت امکانات قابل‌اعتماد فراهم آمده، و کنترل‌های مدیریت پیکربندی قوی‌تری اعمال شده است. این سیستم نسبتاً به نفوذ مقاوم است.

موارد زیر، حداقل الزامات برای اختصاص رتبه B2 به یک سیستم است:[1]

۱. **سیاست امنیتی (Security Policy)**

i. Discretionary Access Control
ii. Object Reuse
iii. Labels
   a. Label Integrity
   b. Exportation of Labeled Information
   c. Subject Sensitivity Labels

در یک نشست تعاملی[2]، TCB باید سریعاً کاربر ترمینال را از بروز هر تغییری در سطح امنیتی مرتبط با آن کاربر مطلع سازد. کاربر ترمینال نیز باید قادر به پرسش از TCB، درباره برچسب حساسیت باشد.

---

[1] از تشریح موارد مشابه با کلاس B1، صرف‌نظر شده است.

[2] Interactive Session

d. Device Labels

TCB باید امکان تخصیص سطوح امنیتی حداقلی و حداکثری به تمام تجهیزات فیزیکی متصل شده به سیستم را داشته باشد. TCB از این سطوح امنیتی برای اعمال محدودیت‌های موردنظر بر تجهیزات استفاده می‌نماید.

iv. Mandatory Access Control (MAC)

TCB باید یک سیاست کنترل دسترسی اجباری را برای تمام منابع (به‌عنوان‌مثال Subjectها، Objectهای ذخیره‌شده و تجهیزات I/O که به‌طور مستقیم و غیرمستقیم برای افراد خارج از TCB، قابل‌دسترس هستند)، را اجرا نماید. باید به این Subjectها و Objectها برچسب حساسیت که ترکیبی از سطوح طبقه‌بندی سلسله مراتبی و غیر سلسله مراتبی است، الصاق شود. از این برچسب‌ها به‌عنوان پایه‌ای جهت تصمیم‌گیری کنترل دسترسی اجباری استفاده می‌شود. TCB باید قادر به پشتیبانی از دو یا چند سطح امنیتی از این قبیل باشد.

۲. **پاسخگویی (Accountability)**

i. Identification and Authentication

a. Trusted Path

TCB باید یک مسیر ارتباطی قابل‌اعتماد با کاربری که قصد ورود (Login) و احرازهویت دارد، برقرار نماید. ارتباطات از طریق این مسیر، انحصاراً باید توسط کاربر شروع شود.

ii. Audit

۳. **تضمین (Assurance)**

i. Operational Assurance

a. System Architecture

b. System Integrity

c. Covert Channel Analysis

توسعه‌دهنده سیستم باید یک جستجوی کامل برای کانال‌های ذخیره‌سازی پنهان انجام داده، و حداکثر پهنای باند (توسط اندازه‌گیری واقعی و یا محاسبه مهندسی) از هر کانال شناسایی‌شده را تعیین نماید.

d. Trusted Facility Management

TCB باید از عملکرد جداگانه اپراتور و مدیر (Administrator)، پشتیبانی نماید.

ii. Life-Cycle Assurance
   a. Security Testing
   b. Design Specification and Verification
   c. Configuration Management

در طول توسعه و نگهداری TCB، باید یک سیستم مدیریت پیکربندی اقدام به نگهداری تغییرات انجام گرفته در توصیف خصوصیات Top-Level، داده‌های دیگر طراحی، مستندات پیاده‌سازی، کد منبع، نسخه در حال اجرای کد شیء، و اسناد و لوازم آزمایش، نماید.

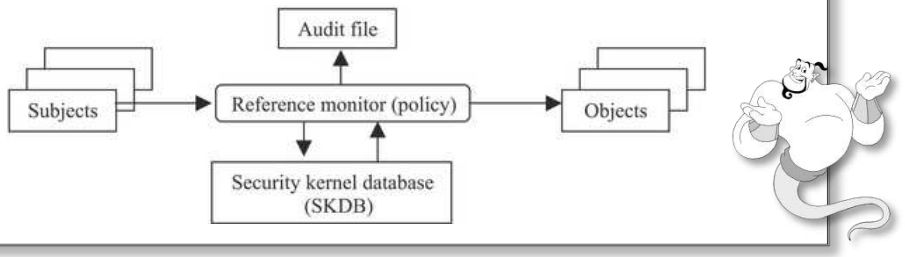
۴. **مستندسازی (Documentation)**

i. Security Features User's Guide
ii. Trusted Facility Manual
iii. Test Documentation
iv. Design Documentation

- **کلاس B3: Security Domains**

کلاس B3، باید الزامات مانیتور مرجع[1]، که واسط تمام دسترسی‌های Subjectها به Objectها است، را برآورده سازد. این واسط باید در برابر جعل مقاوم بوده، و به‌اندازه کافی کوچک باشد تا مورد تحلیل و آزمون قرار گیرد. برای این منظور ساختار TCB با حذف کدهایی که در اجرای سیاست امنیتی غیرضروری است، برقرار گردیده و پیچیدگی آن را به حداقل رسانده است. این سیستم در برابر نفوذ بسیار مقاوم است.

مانیتور مرجع (Reference Monitor)، مجموعه‌ای از الزامات طراحی در یک مکانیسم اعتبار مرجع است؛ که وظیفه اجرای سیاست کنترل دسترسی را بر عهده دارد.
تصویر زیر نمایش‌دهنده روابط مانیتور مرجع در یک سیستم است.



[1] Reference Monitor

موارد زیر، حداقل الزامات برای اختصاص رتبه B3 به یک سیستم است:[۱]

۱. **سیاست امنیتی (Security Policy)**

i. Discretionary Access Control
ii. Object Reuse
iii. Labels
a. Label Integrity
b. Exportation of Labeled Information
c. Subject Sensitivity Labels
d. Device Labels
iv. Mandatory Access Control (MAC)

۲. **پاسخگویی (Accountability)**

i. Identification and Authentication
a. Trusted Path
ii. Audit

۳. **تضمین (Assurance)**

i. Operational Assurance
a. System Architecture
b. System Integrity
c. Covert Channel Analysis
d. Trusted Facility Management
e. Trusted Recovery

روش‌ها و مکانیسم‌هایی باید ارائه گردد تا اطمینان حاصل شود که پس از شکست[۲] سیستم ADP و یا سایر گسست‌ها[۳] بدون اینکه حفاظت در معرض خطر قرار گیرد، بازیابی امکان‌پذیر خواهد بود.

ii. Life-Cycle Assurance
a. Security Testing
b. Design Specification and Verification
c. Configuration Management

۴. **مستندسازی (Documentation)**

i. Security Features User's Guide
ii. Trusted Facility Manual
iii. Test Documentation
iv. Design Documentation

---

[۱] از تشریح موارد مشابه با کلاس‌های قبلی، صرف‌نظر شده است.

[2] Failure
[3] Discontinuity

بخش A: محافظت تأییدشده (Verified Protection)

تمایز این بخش، به‌دلیل استفاده از روش‌های رسمی تأیید امنیت، برای حصول اطمینان از به‌کارگیری کنترل‌های امنیتی اجباری و اختیاری در سیستم است؛ که می‌تواند به‌طور مؤثر به محافظت از اطلاعات حساس طبقه‌بندی‌شده یا دیگر اطلاعات حساس ذخیره‌شده و یا پردازش‌شده توسط سیستم بپردازد. برای اثبات اینکه TCB، الزامات امنیتی را در تمام جنبه‌های طراحی، توسعه و پیاده‌سازی رعایت کرده، به مستندات مبسوطی نیاز خواهد بود.

- **کلاس A1: Verified Design**

عملکرد سیستم در کلاس A1، همانند کلاس B3 بوده و در آن هیچ ویژگی معماری اضافه نشده است. اما خصوصیت متمایز سیستم در این کلاس، تجزیه‌وتحلیل به‌دست‌آمده از مشخصات طراحی رسمی و روش‌های تأیید است؛ که درنتیجه، با اطمینان بالایی می‌توان گفت که TCB به‌درستی اجراشده است. این اطمینان با یک مدل رسمی از سیاست امنیتی و مشخصات رسمی Top-Level (FTLS)[1] از طراحی، شروع می‌شود.

فارغ از مشخصات خاص زبان یا سیستم تأیید مورداستفاده، پنج معیار مهم برای کلاس A1 وجود دارد:

* یک مدل رسمی از سیاست امنیتی، باید به‌وضوح مشخص و مستند شده باشد. این مدل رسمی شامل یک اثبات ریاضی است که یک مدل سازگار با موضوع اصلی بوده و برای پشتیبانی از سیاست امنیتی کافی می‌باشد.
* ساخت یک FTLS، که شامل تعاریف انتزاعی[2] از عملکرد TCB بوده و از مکانیسم‌های Firmware یا سخت‌افزاری برای پشتیبانی از حوزه‌های اجرایی جداگانه، استفاده کرده باشد.
* FTLS باید در صورت امکان، مطابق با مدل روش‌های رسمی (به‌عنوان‌مثال که در آن ابزار بررسی وجود داشته باشد)، و در غیر این صورت روش‌های غیررسمی، باشد.
* پیاده‌سازی TCB (به‌عنوان‌مثال سخت‌افزار، Firmware و نرم‌افزار)، باید به‌صورت غیررسمی نشان دهد که با FTLS سازگار است. با استفاده از روش‌های غیررسمی باید نشان داده شود که عناصر FTLS مطابق با عناصر TCB است. FTLS باید مکانیسم

[1] Formal Top-Level Specification (FTLS)
[2] Abstract

حفاظتی یکپارچه موردنیاز برای تأمین سیاست امنیتی را بیان کند؛ که در آن عناصر مکانیسم حفاظتی به عناصر TCB نگاشت شده باشد.

* روش‌های تحلیل رسمی باید برای شناسایی و تجزیه‌وتحلیل کانال‌های مخفی مورداستفاده قرار گیرند. روش‌های غیررسمی باید برای شناسایی کانال‌های زمانی پنهان، مورداستفاده قرار گیرند. ادامه روند شناسایی کانال‌های پنهان در سیستم، باید توجیه‌پذیر باشد.

موارد زیر، حداقل الزامات برای اختصاص رتبه A1 به یک سیستم است:[۱]

۱. **سیاست امنیتی (Security Policy)**

i. Discretionary Access Control
ii. Object Reuse
iii. Labels
   a. Label Integrity
   b. Exportation of Labeled Information
   c. Subject Sensitivity Labels
   d. Device Labels
iv. Mandatory Access Control (MAC)

۲. **پاسخگویی (Accountability)**

i. Identification and Authentication
   a. Trusted Path
ii. Audit

۳. **تضمین (Assurance)**

i. Operational Assurance
   a. System Architecture
   b. System Integrity
   c. Covert Channel Analysis
   d. Trusted Facility Management
   e. Trusted Recovery
ii. Life-Cycle Assurance
   a. Security Testing
   b. Design Specification and Verification
   c. Configuration Management
   d. Trusted Distribution

باید یک سیستم کنترل ADP و امکانات توزیع مورد اعتماد، برای حفظ یکپارچگی نگاشت‌های بین توصیف داده اصلی نسخه جاری از TCB، با کپی اصلی از کدهای نسخه جاری،

[۱] از تشریح موارد مشابه با کلاس‌های قبلی، صرف‌نظر شده است.

فراهم آورده شود. باید برای حصول اطمینان از اینکه توزیع به‌روزرسانی نرم‌افزار، Firmware و سخت‌افزار TCB، به مشتری، دقیقاً منطبق با کپی اصلی بوده است، دستورالعمل‌هایی (به‌عنوان‌مثال آزمون پذیرش امنیت سایت)، وجود داشته باشد.

۴. **مستندسازی (Documentation)**

i. Security Features User's Guide
ii. Trusted Facility Manual
iii. Test Documentation
iv. Design Documentation

انواع کنترل دسترسی

استاندارد TCSEC که در این مبحث موردبررسی قرار گرفت طی مستندات Orange Book وزارت دفاع[1] ایالات‌متحده در سال 1985 منتشر گردیده است. استانداردهایی که بانام Orange Book منتشر می‌شد، از سال 2005 تغییر نام یافته و تحت نام Common Criteria ارائه گردید. هرچند Orange Book در سال‌های بعدی به‌روزرسانی گردید و برخی از استانداردهای قبلی خود را منسوخ کرد؛ اما به نظر من سطوح دسترسی گفته‌شده در TCSEC برای فهم نحوه دسترسی در سیستم‌عامل‌ها مفید بوده و همچنان نیز مورداستفاده قرار می‌گیرد.

تعریف سطوح دسترسی در سال‌های بعد علاوه بر موارد گفته‌شده در TCSEC، شامل موارد دیگری نیز شده، که در ادامه این مبحث به تشریح مختصری از آن‌ها می‌پردازیم.

توجه داشته باشید که نحوه تعریف کنترل دسترسی ممکن است در محصولات و مستندات شرکت‌های تولیدکننده مختلف، متفاوت باشد؛ اما معمولاً شامل همه یا چند مورد از موارد زیر خواهد بود.

- **کنترل دسترسی اختیاری (DAC)**

از کنترل دسترسی اختیاری بانام کنترل دسترسی احتیاطی نیز نام برده می‌شود. در DAC مالک اشیاء تعریف‌کننده نحوه دسترسی به آن‌ها هستند.

در این حالت مالک هر شیء مشخص می‌کند که چه Subjectای و در چه سطحی می‌تواند به Object موردنظر دسترسی داشته باشد.

کنترل دسترسی اختیاری به‌صورت مبسوط در TCSEC تشریح شده است.

[1] Department of Defense (DoD)

- **کنترل دسترسی اجباری (MAC)**

در کنترل دسترسی اجباری، سیستم مشخص‌کننده نحوه دسترسی Subjectها به Objectها است. در این نوع سیستم شاید انعطاف‌پذیری کمتر باشد ولی در مقابل امنیت بالاتر خواهد بود. انجام کنترل دسترسی در MAC بر اساس برچسب‌های امنیتی (Security Label) یا همان برچسب‌های حساسیت (Sensitivity Label) انجام می‌گیرد.
کنترل دسترسی اجباری به‌صورت مبسوط در TCSEC تشریح شده است.

- **کنترل دسترسی بر مبنای نقش (Role-Based Access Control)**

کنترل دسترسی بر مبنای نقش که به‌اختصار RBAC نیز نامیده می‌شود؛ اعطای دسترسی را بر اساس نقش تعلق‌گرفته به Subject انجام می‌دهد. در این حالت نقش‌های موردنیاز در سیستم تعریف‌شده و بر اساس نیاز آن‌ها در انجام وظایف سازمانی، سطوح دسترسی تعلق می‌گیرد. پس از تعیین نقش‌ها و اعطای مجوز به آن‌ها، کاربران در نقش‌های مختلف قرارگرفته و به‌واسطه نقش موردنظر، سطوح دسترسی به کاربر اختصاص می‌یابد.
در این حالت با تغییر کاربران بخش‌های مختلف نیازی به تعریف مجدد دسترسی‌ها نبوده و فقط کاربر جدید را جایگزین کاربر قبلی می‌نمایند. به‌عبارت‌دیگر زمانی که یک نقش از کاربر گرفته می‌شود، بالطبع تمام دسترسی‌های مربوط به آن نقش نیز از کاربر گرفته می‌شود؛ و زمانی که یک نقش به یک کاربر تعلق می‌گیرد، تمام دسترسی‌های مربوطه نیز به آن کاربر اعطا می‌گردد.
استفاده از RBAC مخصوصاً در سازمان‌های بزرگی که دارای تعداد کاربران زیاد است، باعث می‌گردد اشتباهات مدیر سیستم در اعطای مجوزها به حداقل کاهش یابد. همچنین بازرسی‌های بعدی مدیر سیستم برای بررسی سطوح تعلق‌گرفته به کاربران نیز آسان‌تر خواهد بود.
روش RBAC می‌تواند به‌صورت اختیاری (DAC) یا اجباری (MAC) انجام پذیرد.

- **کنترل دسترسی بر مبنای شبکه (Lattice-Based Access Control)**

کنترل دسترسی بر مبنای شبکه که به‌اختصار LBAC نامیده می‌شود، اعطای دسترسی را به‌صورت سلسله مراتبی انجام می‌دهد.
در برخی از مستندات فنی LBAC را در همان مدل MAC توصیف می‌کنند؛ چراکه ازنظر عملکرد بسیار شبیه MAC است.
در LBAC دارای یک سلسله‌مراتب به‌صورت زیر هستیم:



در این حالت کنترل دسترسی بر اساس سطحی از سلسله‌مراتب فوق، که کاربر در آن قرار می‌گیرد، انجام می‌شود. ابتدا تمام Objectها بر اساس طبقه‌بندی فوق، سازمان‌دهی می‌شوند، سپس کاربر دریکی از سطوح سلسله‌مراتب قرار می‌گیرد. کاربری که در سطح بالاتر قرار گیرد می‌تواند به تمام اشیائی که در سطوح پایین‌تر هستند نیز دسترسی داشته باشد. اما کاربری که در سطوح پایین‌تر قرار می‌گیرد، بالطبع نمی‌تواند به اشیاء سطوح بالاتر دسترسی داشته باشد.

- **کنترل دسترسی بر مبنای خصوصیت (Attribute-Based Access Control)**

کنترل دسترسی بر مبنای خصوصیت که به‌اختصار ABAC نیز نامیده می‌شود، یک روش انعطاف‌پذیر برای ارائه دسترسی بر اساس ارزیابی ویژگی‌ها است. البته ABAC را می‌توان یک مدل دسترسی کنترل دانست تا یک روش کنترل دسترسی مثل MAC.

ABAC یک مدل کنترل دسترسی منطقی است که تشخیص دسترسی به اشیاء را با ارزیابی ویژگی نهاد (Subject یا Object)، اقدامات و محیط مربوط به درخواست انجام می‌دهد. در ساده‌ترین شکل، ABAC مبتنی بر بررسی خصوصیات Subject، خصوصیات Object، شرایط محیط، و یک ارتباط رسمی یا قوانین کنترل دسترسی مشخص‌شده بر اساس عملیات مجاز Subject-Object و ترکیب شرایط محیطی است.

سیستم ABAC قادر به اجرای کنترل دسترسی اختیاری (DAC) و همچنین کنترل دسترسی اجباری (MAC) است.

مدل ABAC طی مستندات NIST SP 800-162 ارائه‌شده است.[1]

[1] http://csrc.nist.gov/publications/nistpubs/800-162/sp800_162_pre-publication.pdf

☑ مبحث سوم

بهترین شیوه‌های امنیت سیستم‌عامل

غالباً سرورها از سیستم‌عامل‌های همه‌منظوره استفاده می‌نمایند که دارای سرویس‌های مختلفی هستند. به‌دلیل اینکه تولیدکنندگان سیستم‌عامل از نیازها و سیاست‌های امنیتی سازمان‌ها مطلع نیستند، لذا تنها می‌توانند یک امنیت حداقلی را در نظر گرفته و سیستم‌عامل را به همراه یک سری تنظیمات پیش‌فرض ارائه نمایند. به‌همین دلیل مدیران سیستم باید بر اساس طراحی‌های انجام‌شده، نیازها، سیاست‌ها و تهدیدات بالقوه‌ای که ممکن است در کمین سازمان آن‌ها باشد، اقدام به پیکربندی و امن سازی سیستم‌عامل و سرور نمایند.

روش‌های توصیه‌شده در این بخش به مدیران سیستم کمک می‌کند تا برای تنظیمات امنیتی سرور، یک چک‌لیست فراهم آمده از بهترین روش‌ها را در اختیار داشته باشند. همان‌طور که قبلاً نیز گفته شد، روش‌ها و دستورالعمل‌های بیان‌شده در این مبحث، به‌صورت عمومی بوده و قابل پیاده‌سازی در اکثر سیستم‌عامل‌ها است. بازهم توصیه می‌کنم علاوه بر موارد گفته‌شده در این مبحث، مقاوم‌سازی سیستم‌عامل‌های خود را با توجه به برند و نسخه آن، بر اساس سایر مستنداتی که توسط تولیدکننده سیستم‌عامل ارائه گردیده نیز، انجام دهید.

در NIST SP800-123[1] موارد عمومی که لازم است برای امن سازی سیستم‌عامل در نظر گرفته شود، در چهار گام اصلی، و به‌صورت زیر تقسیم‌بندی شده است:

۱. نصب وصله‌ها[2] و به‌روزرسانی سیستم‌عامل.
۲. مقاوم‌سازی[3] و پیکربندی سیستم‌عامل، برای حل مسائل امنیتی.
۳. نصب و پیکربندی کنترل‌های امنیتی اضافی؛ البته در صورت نیاز.
۴. آزمون امنیت سیستم‌عامل؛ برای حصول اطمینان از اینکه کارهای انجام‌شده در مراحل قبلی، توانسته مسائل امنیتی را پوشش داده و امنیت را به‌اندازه کافی فراهم آورده باشد.

درنتیجه ترکیب مراحل فوق، باید یک سطح معقول برای حفاظت از سیستم‌عامل، فراهم آید.

---

[1] http://csrc.nist.gov/publications/nistpubs/800-123/SP800-123.pdf
[2] Patch
[3] Hardening

۱. نصب وصله‌ها و به‌روزرسانی سیستم‌عامل

برای سیستم‌عامل‌هایی که قبلاً نصب‌شده‌اند، مهم‌ترین کار نصب وصله‌ها و در صورت وجود، ارتقاء نسخه سیستم‌عامل است. نصب وصله‌ها و به‌روزرسانی سیستم‌عامل یکی از ضروری‌ترین کارها برای اصلاح آسیب‌پذیری‌های شناخته‌شده است. قبل از اینکه یک دستگاه را به‌عنوان سرویس‌دهنده در دسترس افراد غیرقابل‌اعتماد قرار دهیم، باید تمام آسیب‌پذیری‌های شناخته‌شده سیستم‌عامل، رفع و اصلاح‌شده باشند.

مدیران سیستم برای شناسایی و اصلاح آسیب‌پذیری‌ها، باید کارهای زیر را انجام دهند:

- ایجاد، مستندسازی و برپایی یک فرآیند جهت نصب وصله‌ها.[۱]
- شناسایی آسیب‌پذیری‌ها و وصله‌های مناسب.
- کاهش موقتی آسیب‌پذیری، در صورت نیاز و در صورت امکان؛ تا زمانی که وصله‌های موجود مورد آزمایش قرارگرفته و نصب شوند.
- انجام دائمی تعمیرات؛ مثل نصب وصله‌ها یا ارتقاء محصول.

مدیران باید اطمینان حاصل نمایند که سرویس‌دهندگان، مخصوصاً آن‌هایی که جدید هستند، به‌اندازه کافی در طول فرآیند نصب وصله‌ها، محافظت می‌شوند. مدیران باید برای آماده‌سازی یک سرور جدید، موارد زیر را انجام دهند:

- تا زمانی که وصله‌ها بر روی سرور نصب نشده و همچنین دیگر موارد گفته‌شده جهت پیکربندی، بر روی سیستم‌عامل اعمال نگردیده است، باید سرور را از شبکه قطع نگه‌داشته و یا به یک شبکه ایزوله مورد اعتماد متصل نمود.
- درصورتی‌که برای نصب وصله‌ها یا به‌روزرسانی و ارتقاء، مجبور به اتصال سرور به شبکه هستید، سرور را در یک VLAN یا بخشی از شبکه که کاملاً حفاظت می‌شود، قرار دهید. تنها زمانی مجاز به اتصال سرور به شبکه عمومی هستید که تمام وصله‌ها و به‌روزرسانی‌ها بر روی سیستم‌عامل آن اعمال‌شده باشد.

با توجه به اینکه نصب وصله‌ها و به‌روزرسانی‌ها ممکن است بر روی سرور تأثیرات غیرمنتظره داشته باشد؛ لذا مدیران قبل از نصب وصله‌ها و به‌روزرسانی سرورهای عملیاتی که در حال دادن سرویس هستند، باید آن‌ها را بر روی یک سرور مشابه با همان سرویس‌ها و مشخصات، نصب و آزمایش نمایند. همچنین مدیران می‌توانند سرورها را برای دریافت خودکار وصله‌ها پیکربندی نمایند؛ اما نباید اجازه نصب خودکار وصله‌های دانلود شده را به آن‌ها بدهند.

---

[۱] برای کسب اطلاعات بیشتر در این مورد می‌توانید به NIST SP800-40 مراجعه نمایید.

۲. مقاوم‌سازی و ایمن‌سازی پیکربندی سیستم‌عامل

مدیران باید مراحل زیر را برای مقاوم‌سازی و ایمن‌سازی پیکربندی سیستم‌عامل انجام دهند:

- حذف سرویس‌ها، برنامه‌های کاربردی و پروتکل‌های شبکه، که غیرضروری هستند.
- پیکربندی احراز هویت کاربر در سیستم‌عامل.
- پیکربندی کنترل‌های منابع.

چند مرتبه در این کتاب توضیح دادم که نباید اقدام به نصب سرویس‌ها و برنامه‌های غیرضروری نماییم، اما اعتراف می‌کنم! همه ما اغلب به دو دلیل زیر، این مورد را نادیده می‌گیریم:
*** اگر برای نصب سیستم‌عامل یا برنامه کاربردی، لایسنس خریداری‌شده باشد، تمام سرویس‌ها و اجزای آن را نصب می‌کنیم تا مبادا یک ریال از پولی که بابت محصول پرداخت‌شده، حرام شود.**
*** اگر هم لایسنس خریداری نشده باشد، فکر می‌کنیم که با نصب کامل سرویس‌ها و اجزای محصول، انتقام تمام تحریم‌ها و پول‌های بلوکه‌شده ایران را از امریکا می‌گیریم!**



موارد فوق نیاز به شرح بیشتری دارند، که در ادامه به آن‌ها می‌پردازیم.

حذف یا غیرفعال کردن سرویس‌ها، برنامه‌ها و پروتکل‌های شبکه غیرضروری

به‌صورت ایده‌آل، سرور باید اختصاصاً برای یک منظور خاص مورداستفاده قرار گیرد. در صورت امکان باید سیستم‌عامل را با حداقل پیکربندی نصب کرده، و در صورت نیاز بعداً اقدام به حذف و اضافه سرویس‌ها و اجزای موردنظر نمود. در زمان پیکربندی سیستم‌عامل، تمام سرویس‌ها، برنامه‌ها و پروتکل‌های شبکه که موردنیاز نیست باید حذف شوند؛ و درصورتی‌که قابل‌حذف نباشد، باید آن‌ها را غیرفعال نمود. با توجه به اینکه معمولاً اسکریپت‌ها و فرآیندهای حذف، به‌طور کامل همه اجزای یک سرویس را از روی سیستم حذف نمی‌کند، لذا بهتر است که سرویس‌های غیرضروری اصلاً نصب نشوند. همچنین حذف سرویس‌ها و برنامه‌های غیرضروری بهتر از غیرفعال کردن آن‌ها است؛ چراکه در صورت غیرفعال بودن، امکان فعال‌سازی آن‌ها توسط فرد نفوذکننده و یا حتی خطای انسانی، وجود دارد. موارد زیر، انواع متداول سرویس‌ها و برنامه‌هایی است که در صورت عدم نیاز، باید حذف یا غیرفعال شوند:

- سرویس‌های اشتراک‌گذاری فایل و چاپگر (مانند NetBIOS، NFS[1] و FTP[2]).
- سرویس‌های شبکه‌های بی‌سیم (مثل Wi-Fi و Bluetooth).
- سرورها و سرویس‌های وب.
- سرویس‌های پست الکترونیک (مثل SMTP).

[1] Network File System(NFS)
[2] File Transfer Protocol(FTP)

- دسترسی راه‌دور و برنامه‌های دسترسی که از کانال‌های ارتباطی امن و رمزگذاری شده، استفاده نمی‌کنند(مثل Telnet).
- سرویس‌های دایرکتوری (مثل LDAP و NIS[1]).
- کامپایلر زبان و کتابخانه.
- ابزار توسعه سیستم.
- ابزارهای مدیریت سیستم و شبکه، که شامل پروتکل SNMP هستند.

حذف یا غیرفعال کردن سرویس‌های غیرضروری، از راه‌های زیر باعث افزایش امنیت سیستم می‌گردد:

- مهاجم نمی‌تواند سایر سرویس‌ها را در اختیار گرفته و از آن‌ها برای حمله به سرور یا ایجاد اختلال استفاده نماید. اضافه شدن هر سرویس، سرور را بیشتر در معرض خطر قرار می‌دهد؛ چراکه هر سرویس یک‌راه ممکن برای هکر محسوب می‌شود.
- سرویس‌های دیگر ممکن است دارای نقص بوده و یا با سرور ناسازگار باشند؛ که با حذف و غیرفعال سازی می‌توانید از تأثیرگذاری آن‌ها بر روی سرور جلوگیری نمایید.
- برای یک سرویس خاص، می‌توان میزبان (Host) را با توجه به موارد موردنیاز بهتر پیکربندی نمود. سرویس‌های مختلف ممکن است به پیکربندی سخت‌افزاری و نرم‌افزاری مختلفی نیاز داشته باشند؛ که این امر می‌تواند منجر به آسیب‌پذیری یا کاهش کارایی در سیستم شود.
- با کاهش سرویس‌ها، بالطبع Logها و وقایع ثبت‌شده نیز کاهش می‌یابند. در این صورت نیاز به بررسی حجم کمتری از وقایع ثبت‌شده است و تشخیص رفتار غیرمنتظره را تسهیل می‌نماید.

پیکربندی احرازهویت کاربر در سیستم‌عامل

برای سرورها، کاربرانی که مجاز به پیکربندی سیستم‌عامل هستند به گروه کوچکی از مدیران سرور محدود می‌شوند. اما برای اجرای محدودیت‌ها، مدیر سرور باید سیستم‌عامل را جهت تأیید هویت کاربران پیکربندی نماید. حتی اگر یک سرور بر روی برخی از سرویس‌های خود به کاربران احرازهویت نشده، اجازه دسترسی می‌دهد، بازهم باید دسترسی‌های مدیریتی و خاص سیستم را به افراد یا گروه‌های خاص محدود نماید. فعال کردن احرازهویت توسط قسمتی از سیستم‌عامل، Firmware و برنامه‌های کاربردی روی سرور مثل نرم‌افزارهایی که سرویس‌های

[1] Network Information System(NIS)

شبکه را پیاده‌سازی می‌نمایند، صورت می‌گیرد. اما در شرایط خاص، مثل سرورهایی که دارای ارزش بالا بوده و یا به‌شدت در معرض خطر هستند، سازمان ممکن است از احرازهویت سخت‌افزاری مثل Token یا تجهیزات OTP نیز استفاده نماید.

استفاده از مکانیسم‌هایی که همواره از یک رمزعبور استفاده کرده و آن را به‌صورت متن ساده آن‌هم از طریق کانال‌های ناامن انتقال می‌دهند، به‌شدت سیستم را در معرض خطر قرار می‌دهند؛ چراکه هکر می‌تواند با به دست آوردن رمزعبور اقدام به حملات Masquerade نماید.

برای اطمینان از احرازهویت صحیح کاربر، موارد زیر را انجام دهید:

- **حذف یا غیرفعال کردن اکانت‌های پیش‌فرض غیرضروری**

سیستم‌عامل‌ها به‌صورت پیش‌فرض دارای اکانت‌هایی مثل Guest، Administrator، root و اکانت‌هایی از این قبیل هستند. بعضی از این اکانت‌ها دارای رمزعبور پیش‌فرض نیز هستند؛ لذا نام کاربری و رمزعبور پیش فرضی که در سیستم‌عامل‌ها وجود دارد، برای همه به‌خوبی شناخته‌شده است. برای جلوگیری از سوءاستفاده هکرها از اکانت‌های پیش‌فرض، باید اکانت‌های غیرضروری را حذف یا غیرفعال نمایید. برای اکانت‌هایی که مجبور به نگه‌داشتن آن‌ها هستید محدودیت تعریف کرده و در صورت امکان نام کاربری آن را نیز تغییر دهید.

- **غیرفعال کردن اکانت‌های غیرتعاملی**

اکانت‌هایی که باید وجود داشته باشند ولی برای Login تعاملی مورداستفاده قرار نمی‌گیرند، باید غیرفعال شوند. برای سیستم‌های یونیکس، Login Shell باقابلیت Null، باید غیرفعال گردد.

- **ایجاد گروه‌های کاربری**

کاربران را به گروه‌های مناسب اختصاص داده، و سپس طبق مستندات طراحی، حقوق دسترسی را به گروه‌ها اختصاص دهید. این روش مخصوصاً زمانی مفید واقع می‌شود که یک سازمان دارای کاربران زیادی است.

- **ایجاد اکانت کاربری**

در طرح استقرار مشخص می‌شود که چه کسی مجاز به دسترسی بر روی چه سیستم و چه سرویس‌هایی است. لذا باید فقط اکانت‌های موردنیاز ساخته‌شده و در گروه مناسب قرار گیرند. توصیه می‌شود که از اکانت‌های اشتراکی استفاده نکنید.

- **پیکربندی همگام‌سازی خودکار زمان**

برخی از پروتکل‌های احرازهویت مثل کربروس، برای انجام عملیات خود، وابستگی زیادی به همگام بودن زمان بین سرور و کلاینت دارند؛ لذا باید با پروتکلی مانند NTP، همگام‌سازی زمان را با یک سرور قابل‌اعتماد در شبکه انجام دهید.

- **بررسی سیاست‌های سازمان در مورد رمزعبور**

قبلاً درباره انتخاب یک رمزعبور قدرتمند، توضیحاتی ارائه‌شده است. شما باید بر روی سیستم اقدام به اعمال یک سیاست مناسب جهت انتخاب رمزعبور نمایید. برای انتخاب رمزعبور باید به موارد زیر توجه کنید:

i. طول رمزعبور
ii. طول عمر یا سالخوردگی
iii. استفاده مجدد
iv. اختیار

چه کسی اجازه دارد کلمه عبور را تغییر داده یا تنظیم مجدد (Reset) نماید؟ همچنین چه نوع مدارکی قبل از شروع هرگونه تغییر نیاز است.

v. امنیت کلمه عبور

چگونه از کلمات عبور ذخیره‌شده بر روی سیستم محافظت به عمل آید؟

- **پیکربندی کامپیوترها برای جلوگیری از حدس زدن رمزعبور**

یکی از راه‌های ورود به سیستم، حدس زدن رمزعبور است. هکرها معمولاً با ابزارهایی شروع به آزمایش انواع کلمات مختلف برای تشخیص رمزعبور می‌نمایند. لذا باید سیستم احرازهویت پس از چند بار ورود اشتباه رمزعبور، برای مدت‌زمان مشخصی قفل گردد. این کار باعث می‌شود که فرآیند حدس رمزعبور، نیاز به یک مدت‌زمان خیلی طولانی پیدا کند. البته این کار به‌صورت کامل جلوی حمله را نمی‌گیرد، لذا توصیه می‌شود درصورتی‌که نیاز به دسترسی از راه دور مدیریتی بر روی سرور ندارید، دسترسی مدیریتی به سیستم از طریق شبکه‌های ناامن و عمومی مثل اینترنت، را غیرفعال نمایید.

- **نصب و پیکربندی مکانیسم‌های دیگر امنیتی برای احرازهویت قوی‌تر**

می‌توانید از مکانیسم‌های دیگری مثل بیومتریک، کارت هوشمند، گواهی سرویس دهنده/سرویس‌گیرنده، و یا رمز عبور یک‌بارمصرف(OTP)، نیز برای احرازهویت استفاده نمایید. هرچند که اجرای این مکانیسم‌ها جهت احرازهویت دارای بار مالی و پیچیدگی اجرایی است؛ ولی ممکن است این سختی‌ها با توجه به اطلاعاتی که روی سرور دارید، توجیه‌پذیر باشند.

برای اینکه نیاز به استفاده از مکانیسم‌های اضافی احرازهویت را کاهش دهید، می‌توانید در موارد عادی از همان مکانیسم‌های معمول احرازهویت ولی در بستری امن استفاده نمایید. به

عنوان مثال استفاده از SSL/TLS، IPsec و SSH، که انتقال اطلاعات را در کانالی امن انجام می‌دهند، می‌تواند در این مورد راهگشا باشد.

پیکربندی کنترل‌های منابع

تمامی سیستم‌عامل‌های مورداستفاده در سرورها، قابلیت تعیین امتیازات دسترسی جداگانه برای فایل‌ها، دایرکتوری‌ها، تجهیزات، و سایر منابع محاسباتی، را فراهم می‌آورند. مدیر سرور می‌تواند با تنظیم دقیق کنترل دسترسی و ممنوع کردن دسترسی کاربران غیرمجاز، امکان بروز نقض امنیتی عمدی و غیرعمدی را کاهش دهد.

یکی از مواردی که نباید مغفول بماند، فعال کردن Auditing یا همان قابلیت بازرسی در سیستم است. این قابلیت به مدیر کمک می‌کند تا بر روی تلاش‌هایی که برای ورود به سیستم صورت گرفته، نظارت نماید.

۳. نصب و پیکربندی کنترل‌های امنیتی اضافی

غالباً سیستم‌عامل‌ها شامل تمام کنترل‌های امنیتی لازم برای حفظ امنیت خود نیستند؛ لذا نمی‌توان سیستم‌عامل‌ها را بی‌نیاز از کنترل‌های امنیتی اضافی دانست. توصیه می‌شود برای حفظ امنیت سیستم‌عامل، اقدام به نصب کنترل‌های زیر نمایید:

- **Anti-malware**
از نرم‌افزارهای ضد بدافزار، همانند آنتی‌ویروس، ضد جاسوس‌افزار، و تشخیص دهنده‌های Rootkit، برای محافظت از سیستم‌عامل استفاده نمایید.
- **Host-based IDPS**
از نرم‌افزارهای تشخیص و جلوگیری از نفوذ مبتنی بر میزبان، استفاده نمایید. این نرم‌افزارها فقط مسئولیت محافظت از همان سروری را دارند که بر روی آن نصب شده‌اند؛ لذا با دقت بیشتری اقدام به تشخیص نفوذ و جلوگیری از آن می‌پردازند.
- **Host-based Firewall**
برای جلوگیری از دسترسی غیرمجاز به سرور، می‌توان از فایروال‌های مبتنی بر میزبان استفاده نمود. این نوع فایروال‌ها فقط مسئول حفاظت از همان سروری هستند که بر روی آن نصب‌شده‌اند؛ لذا با انعطاف بیشتری می‌توان نسبت به تعریف دسترسی‌های مجاز و غیرمجاز بر روی آن، اقدام نمود.
- **Patch Management and Vulnerability Management**
با استفاده از نرم‌افزارهای مدیریت وصله و مدیریت آسیب‌پذیری، می‌توان اطمینان داشت که آسیب‌پذیری‌ها به‌سرعت برطرف شده است. از این ابزارها می‌توان برای

اعمال وصله‌ها و همچنین شناسایی آسیب‌پذیری‌های جدید بر روی سیستم‌عامل، سرویس‌ها و برنامه‌های کاربردی، استفاده نمود.

- **Cryptography**

می‌توان از تکنولوژی رمزنگاری دیسک برای محافظت از دیتای ذخیره‌شده در برابر دسترسی‌های فیزیکی غیرمجاز، استفاده نمود. در برخی از سیستم‌عامل‌ها این تکنولوژی به‌صورت Built-in[1] در اختیار کاربر قرار می‌گیرد؛ البته در سیستم‌عامل‌های دیگر نیز می‌توان از نرم‌افزارهای خارجی برای این منظور بهره برد. درصورتی‌که اطلاعات ذخیره‌شده به‌صورت رمزگذاری شده باشد، حتی اگر مهاجم بتواند به سرور دسترسی فیزیکی پیدا کند، بازهم محرمانگی اطلاعات حفظ‌شده و از افشای آن جلوگیری به عمل خواهد آمد.

استفاده از نرم‌افزارهای Third-Party فوق، برای افزایش امنیت سرور مهم و در برخی موارد واجب و ضروری است. اما استفاده از این نرم‌افزارها ممکن است بخش قابل‌توجهی از منابع سیستم را اشغال نموده و باعث کاهش کارایی سیستم در انجام مأموریت اصلی خود گردد. لذا مدیران سیستم باید در انتخاب نرم‌افزارها و سرویس‌های امنیتی مبتنی بر میزبان، نهایت دقت را داشته باشند. درصورتی‌که محدودیت منابع سرور اجازه استفاده از تمام یا برخی از نرم‌افزارهای امنیتی مبتنی بر میزبان را نمی‌دهد، باید سعی کنید که این کمبودها را با نرم‌افزارها و سرویس‌های امنیتی مبتنی بر شبکه، تا حد قابل قبولی جبران نمایید. لازم به ذکر است که استفاده از کنترل‌های امنیتی مبتنی بر شبکه قطعاً نمی‌توانند جای کنترل‌های امنیتی مبتنی بر میزبان را بگیرند، اما در زمان اضطرار، می‌توان از آن‌ها برای پوشش یکدیگر استفاده نمود؛ به قول ضرب‌المثل: کاچی به از هیچی!

یک نکته مهم را همواره به یاد داشته باشید؛ که شاید بتوان از NIDS، NIPS و فایروال مبتنی بر شبکه، برای پوشش و جبران کاستی‌های HIDS، HIPS و فایروال مبتنی بر میزبان استفاده نمود، اما هرگز آنتی‌ویروس مبتنی بر شبکه (مثل آنتی‌ویروس موجود بر روی UTMها) نمی‌تواند جایگزین و یا حتی پوشش‌دهنده آنتی‌ویروس مبتنی بر میزبان باشد؛ چراکه اساساً تشخیص بسیاری از کدهای مخرب فقط زمانی امکان‌پذیر است که بر روی سیستم وجود داشته و شروع به فعالیت نمایند. آنتی‌ویروس مبتنی بر شبکه تنها می‌تواند بر اساس دیتابیس خود که حاوی امضای ویروس‌های شناخته‌شده است، عمل نماید و قادر به تشخیص ویروس‌هایی که بر اساس رفتار باید شناسایی شوند، نخواهد بود.

---

[1] توکار

بهترین روش، استفاده از ابزارها و نرم‌افزارهای امنیتی مبتنی بر شبکه و مبتنی بر میزبان در کنار یکدیگر است؛ این دو مکمل یکدیگر بوده و هیچ‌یک نمی‌توانند به‌درستی و به‌صورت کامل، جای همتای خود را پر نمایند.

۴. آزمون امنیت سیستم‌عامل

برای بررسی وضعیت سرور، باید بر اساس یک دوره زمانی، متناوباً اقدام به انجام آزمون‌های امنیتی بر روی سرور نمود. این آزمون‌ها باعث می‌شود از درستی اقدامات احتیاطی و امنیتی انجام‌شده، اطمینان حاصل شود. روش‌های معمول آزمون سیستم‌عامل، شامل بررسی آسیب‌پذیری‌ها و آزمون نفوذ است.

عواملی که در زمان آزمون امنیت سیستم‌عامل باید مدنظر داشت، به‌صورت زیر است:

- ممکن است آزمون امنیت بر روی عملکرد سیستم تأثیر منفی داشته باشد. اگر یک آزمون خاصی بر روی سرور انجام شود، ممکن است باعث محرومیت از سرویس (DoS) شده و کاربران دیگر نتوانند از خدمات آن سرویس‌دهنده بهره‌مند شوند. یک راه‌حل، انجام آزمون در زمان‌هایی است که احتمالاً سرور با کمترین مراجعه کاربران درگیر است. یک راه توصیه‌شده دیگر این است که بر روی سرور دیگری که دارای مشخصاتی دقیقاً منطبق با مشخصات سرور است، ولی به‌صورت عملیاتی از آن استفاده نمی‌شود، آزمون را انجام داد. (البته راه‌حل دوم را خودم نمی‌پسندم؛ چراکه آزمون امنیت کاملی محسوب نمی‌شود.)
- اگر سرور دارای اطلاعات حساس شخصی مثل شماره کارت‌های اعتباری و اطلاعاتی از این قبیل است، بهتر است برای آزمون از یک سرور مشابه که دارای اطلاعات غیرواقعی است، استفاده شود.
- اگر از متخصصین داخلی برای آزمون امنیت استفاده می‌کنید، اشخاص مورد تأیید را انتخاب نمایید. اگر از شرکت‌های متخصصی بیرون از سازمان برای آزمون استفاده می‌نمایید، حتماً قرارداد منع افشای اطلاعات (NDA) با آن‌ها امضا کنید.
- هرقدر سروهای عملیاتی و غیرعملیاتی (برای آزمون) را شبیه به هم پیکربندی نموده باشید، بازهم آزمونی که در محیط آزمایشی انجام می‌گیرد، با آزمونی که در محیط واقعی انجام می‌شود، متفاوت خواهد بود. لذا ممکن است در محیط‌های آزمایشی برخی از آسیب‌پذیری‌ها همچنان ناشناخته باقی بمانند.

امنیت دیتابیس

فصل سیزدهم

مبحث اول

ده تهدید برتر دیتابیس

مبحث دوم

روش‌های مقابله با تهدیدات

دیتابیس یکی از مهم‌ترین دارایی‌های اطلاعاتی است که طبق گزارش Verizon Data Breach[1] در سال 2014، بیش از سایر دارایی‌های اطلاعاتی در معرض خطر قرار داشته است. دلیل اینکه چرا دیتابیس هدف هکرها قرار می‌گیرد، بسیار روشن است. دیتابیس برای افراد مهاجم مثل کندوی عسل است که می‌توانند به‌راحتی به تمام داده‌های یک سازمان دسترسی پیدا کنند.

علیرغم ارزش بالای دیتایی که سازمان‌ها در دیتابیس‌های خود ذخیره می‌کنند، معمولاً محافظت انجام‌شده از آن‌ها مناسب نیست. طبق گفته IDC[2]، از ۲۷ میلیارد دلاری که هزینه محصولات امنیتی شده، کمتر از ۵ درصد آن مستقیماً به امنیت مرکز داده تعلق گرفته است.

هنگامی‌که هکر یا کارمند داخلی خرابکار، به اطلاعات حساس دسترسی پیدا کند، می‌تواند به‌سرعت داده‌ها را استخراج کرده و باعث خسارت به سازمان شود. علاوه بر خسارت‌های مالی و آسیب رسیدن به شهرت، ممکن است سازمان متحمل نقض قانون، جریمه و هزینه‌های قانونی دیگر نیز گردد.

اما خبر خوش آنکه، بنا به گزارشی که OTA[3] در سال 2013 ارائه نموده، با اجرای مراحل ساده، کنترل‌های داخلی و پیروی از بهترین شیوه‌ها، از ۹۷ درصد حوادث جلوگیری به‌عمل‌آمده است.

همان‌طور که در فصل مربوط به امنیت سیستم‌عامل گفته شد، بررسی‌های امنیتی که در این کتاب انجام می‌شود، فارغ از برند یا نسخه خاصی از دیتابیس است. مطالبی که در این فصل برای امنیت دیتابیس تشریح می‌شود، اغلب مواردی است که به‌عنوان بهترین شیوه‌های امنیتی شناخته‌شده و در بین انواع دیتابیس‌ها نیز مشترک است. لذا مجدد توصیه می‌کنم، برای افزایش امنیت دیتابیس، حتماً مستندات امنیتی مربوط به دیتابیس مورداستفاده خود را، مطالعه کرده و دستورات آن را نیز پیاده‌سازی نمایید.

در این فصل ابتدا به ده تهدید برتری که غالباً انواع دیتابیس‌ها را در معرض خطر یا افشای اطلاعات قرار می‌دهند، پرداخته می‌شود. سپس در مبحث بعد، مختصراً نحوه مقابله با تهدیدات برتر تشریح خواهند شد.

[1] http://www.verizonenterprise.com
[2] http://www.idc.com
[3] https://www.otalliance.org

☑ مبحث اول

ده تهدید برتر دیتابیس

ده تهدید توصیف‌شده در این مبحث ممکن است برای تمام دیتابیس‌های کوچک و بزرگ روی دهد. این تهدیدات فارغ از برند و نسخه محصول، بررسی‌شده و در ادامه نیز به‌صورت اختصار، هریک از آن‌ها تشریح شده است.

جدول زیر ده تهدید برتر دیتابیس را به ترتیب اولویت نمایش داده است:

رتبه‌بندی	تهدید
۱	امتیازات بیش‌ازحد و استفاده‌نشده
۲	سوءاستفاده از امتیازات (Privilege Abuse)
۳	تزریق ورودی (Input Injection) یا همان SQL Injection سابق
۴	بدافزار (Malware)
۵	دنباله ممیزی ضعیف
۶	در معرض خطر قرار گرفتن رسانه‌های ذخیره‌سازی
۷	بهره‌برداری از آسیب‌پذیری‌ها و اشتباهات پیکربندی دیتابیس
۸	دیتای حساس مدیریت نشده
۹	منع خدمت (DoS)
۱۰	تخصص و آموزش امنیتی محدود

در ادامه به تشریح هر یک از موارد فوق می‌پردازیم:

۱. امتیازات بیش‌ازحد و استفاده‌نشده

وقتی اعطای امتیازات دسترسی افراد به دیتابیس، بیش‌از اندازه موردنیاز آن‌ها باشد، آماده باشید که این امتیازات مورد سوءاستفاده قرار گیرند. به‌عنوان‌مثال یک کارمند بانک که برای انجام وظایفش فقط نیاز به دسترسی اطلاعات تماس صاحب حساب دارد، ممکن است در صورت داشتن دسترسی بیش‌ازحد، خواسته یا ناخواسته باعث تغییر در موجودی حساب مشتریان گردد. علاوه بر این زمانی که یک کارمند سازمان را ترک می‌کند، متأسفانه اغلب دسترسی او به

دیتای حساس سازمان، تغییر داده نمی‌شود؛ که این امتیازات دسترسی بلااستفاده، می‌تواند به‌راحتی مورد سوءاستفاده قرار گیرد. از طرف دیگر، اگر این کارمند، سازمان را در شرایط نامطلوبی ترک کرده باشد، احتمال دارد که با استفاده از دسترسی‌های قدیمی خود (که همچنان غیرفعال نشده است)، اقدام به سرقت اطلاعات و بروز خسارت در سازمان نماید.

اما سؤال اینجاست که چرا دسترسی بیش‌ازاندازه به کاربر تعلق می‌گیرد؟ معمولاً دلیل آن، تعریف نادرست سطوح دسترسی برای نقش‌های شغلی مختلف است. با توجه به اینکه برای تمام نقش‌ها، دسترسی به‌طور کامل و منطبق با سیاست‌های امنیتی صورت نگرفته، لذا دسترسی‌های عمومی و یا پیش فرضی که به کاربران اعطا می‌شود، معمولاً فراتر از نیاز او بوده و یک ریسک غیرضروری را به خود تحمیل کرده‌اید.



دو نوع ریسک وجود دارد: ریسک ضروری و ریسک غیرضروری.
زمانی که شما قصد عبور از عرض خیابان را دارید و در آن محدوده هیچ پل عابری وجود ندارد، پس مجبورید ریسک عبور از خیابان را بپذیرید؛ این ریسک ضروری است.
اما در صورت وجود پل عابر، یک انسان عاقل بدون اینکه هیچ ریسکی را به خود تحمیل کند، با آرامش خاطر از روی پل عابر عبور می‌کند. در صورت وجود پل عابر، ریسک عبور از خیابان، یک ریسک غیرضروری است.

۲. سوءاستفاده از امتیازات

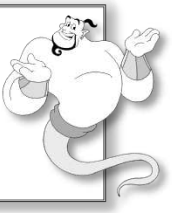
درصورتی‌که کاربر بخواهد از امتیازات قانونی دسترسی خود، برای اهداف غیرمجاز استفاده نماید، می‌گوییم که از امتیازات سوءاستفاده شده است. به‌عنوان‌مثال یک برنامه مراقبت پزشکی را در نظر بگیرید که از طریق رابط وب، می‌توان اقدام به مشاهده سوابق بیمار نمود. معمولاً رابط وب اطلاعات محدودی از بیمار را به کاربر نمایش داده و امکان دیدن سوابق چند بیمار به‌صورت همزمان و یا کپی‌برداری از اطلاعات را نیز فراهم نمی‌کند. اما یک کاربر رِند ممکن است قادر به دور زدن محدودیت‌ها بوده و با ابزاری مثل MS-Excel به دیتابیس متصل شود. در این صورت با استفاده از MS-Excel و اعتبار ورود مشروع خود، می‌تواند به تمام سوابق بیماران و اطلاعات موجود بر روی دیتابیس، دسترسی پیداکرده و از آن‌ها کپی‌برداری نماید.

۳. تزریق ورودی (SQL Injection سابق)

حملات تزریق به دیتابیس به دو گروه عمده تقسیم می‌شوند:

i. SQL Injection؛ که هدف آن‌ها سیستم‌های دیتابیس مرسوم و سنتی است.

ii. NoSQL Injection؛ که هدف آن‌ها پلتفرم‌های بزرگ دیتا است.



در مقابل دیتابیس‌های مرسوم که به‌صورت رابطه‌ای (RDBMS) عمل می‌کنند، دیتابیس NoSQL یک دیتابیس غیر رابطه‌ای و توزیع‌شده است که برای داده‌های بزرگ طراحی گردیده است. سیستم‌های NoSQL دیتا را به‌صورت توزیع‌شده بر روی چندین سرور نگهداری می‌کنند؛ درحالی‌که در دیتابیس‌های مرسوم، اگر یک دیتابیس بزرگ‌تر از یک سرور شود، با مشکلات و پیچیدگی‌هایی مواجه خواهد شد.

حملات SQL Injection معمولاً شامل قرار دادن یا تزریق گزاره غیرمجاز یا مخرب در داخل فیلدهای ورودی برنامه‌های تحت وب است. درحالی‌که حملات NoSQL Injection شامل قرار دادن گزاره‌های مخرب داخل قطعات بزرگ داده است. یک حمله موفق تزریق ورودی، می‌تواند یک دسترسی بدون محدودیت را بر روی دیتابیس، برای مهاجم فراهم آورد.

متأسفانه این تصور غلط وجود دارد که داده‌های بزرگ در برابر حملات SQL Injection آسیب‌پذیر نیستند. دلیل این تصور غلط، این واقعیت است که دیتای بزرگ از فناوری‌های مبتنی بر SQL، استفاده نمی‌کند؛ اما آگاه باشید که خود قطعات بزرگ دیتا همچنان مستعد ابتلا به حملات تزریق ورودی هستند.

۴. بدافزار

مجرمان سایبری، هکرهای دولتی و جاسوس‌ها، طی حملات پیشرفته‌ای که ترکیبی از چندین تاکتیک ازجمله فیشینگ و بدافزار است، جهت نفوذ به سازمان‌ها و سرقت اطلاعات اقدام می‌نمایند. سپس کاربران قانونی، بی‌خبر از آنکه بر روی سیستم خود بدافزار دارند، به مجرایی جهت دسترسی غیرقانونی هکرها و مجرمان سایبری تبدیل می‌شوند.

۵. دنباله ممیزی ضعیف

ضبط خودکار تراکنش‌های مربوط به دیتای حساس در دیتابیس، باید بخشی از هرگونه استقرار دیتابیس باشد. بروز شکست در جمع‌آوری دقیق سوابق ممیزی (بازرسی) از فعالیت‌های دیتابیس، نشان‌دهنده یک خطر جدی است که سازمان را در تمام سطوح تهدید می‌کند.

تمام دیتابیس‌ها دارای یک سری ابزار ممیزی محلی برای ثبت رخدادها هستند؛ اما این ابزارها علاوه بر استفاده از منابعی مثل CPU و دیسک، جزئیات لازم را جهت بازرسی و تشخیص حمله ثبت نمی‌کنند. از طرفی هر برند دیتابیس، دارای مکانیسم‌های ممیزی منحصر به خود است، که در

سازمان‌های بزرگ با دیتابیس‌های متفاوت، یک مانع قابل‌توجه برای اجرای یکنواخت فرآیند بازرسی خواهد بود.

از طرف دیگر، متأسفانه بیشتر مکانیسم‌های ممیزی، هیچ آگاهی نسبت به فعالیت‌هایی که توسط کاربر نهایی بر روی دیتابیس انجام می‌شود، ندارند؛ لذا کاربر دارای دسترسی مشروع یا غیرمشروع مدیریتی به دیتابیس، می‌تواند ممیزی محلی دیتابیس را غیرفعال نماید.



هرگاه در هر سیستمی، سرویس ثبت Log و ممیزی را در حالت غیرفعال مشاهده کردید، تردید نکنید که یک تهدید بزرگ سازمان شما را تهدید می‌کند؛ چراکه یکی از اولین اقدامات نفوذگر، غیرفعال کردن سیستم‌های ممیزی جهت جلوگیری از ردیابی و شناسایی نفوذ است.

به‌صورت ایده‌آل، برای اطمینان از داشتن یک ممیزی قدرتمند، بهتر است وظایف ممیزی به‌صورت جداگانه از مدیران دیتابیس و سرور دیتابیس، قرار داشته باشد.

۶. در معرض خطر قرار گرفتن رسانه‌های ذخیره‌سازی

رسانه‌های ذخیره‌سازی پشتیبان‌گیری، غالباً به‌صورت کامل در مقابل حملات محافظت نمی‌شوند؛ درنتیجه نقض امنیتی متعددی نسخه‌های پشتیبان موجود بر روی دیسک‌ها و نوارها را تهدید می‌کند. علاوه بر این، عدم نظارت و ممیزی بر روی فعالیت‌های مدیران که دسترسی سطح پایین به دیتای حساس دارند، می‌تواند اطلاعات را در معرض خطر قرار دهد.

حفاظت از نسخه پشتیبان دیتای حساس، و نظارت بر روی کاربران با سطح دسترسی بالا، نه‌تنها بهترین شیوه امنیتی است؛ بلکه توسط بسیاری از مقررات نیز، لازم‌الاجرا گردیده است.

۷. بهره‌برداری از آسیب‌پذیری‌ها و اشتباهات پیکربندی دیتابیس

پیدا کردن دیتابیس‌های بدون وصله و آسیب‌پذیر، و همچنین کشف دیتابیس‌هایی که بر اساس پارامترهای پیکربندی پیش‌فرض فعالیت می‌کنند، یکی از راه‌های متداول مورداستفاده توسط هکرها برای تدارک حملات است. مهاجمین به‌خوبی می‌دانند که چگونه از این آسیب‌پذیری‌ها برای یک حمله موفق علیه سازمان شما، بهره‌برداری نماید.

متأسفانه بسیاری از سازمان‌ها نسبت به اعمال وصله‌های امنیتی ارائه‌شده برای دیتابیس تعلل کرده و حتی ممکن است چند ماه طول بکشد تا وصله امنیتی مربوطه را نصب نمایند. اما

باید این موضوع مهم را در نظر داشته باشند، تا مادامی‌که وصله بر روی دیتابیس اعمال نگردیده باشد، در مقابل حملات آسیب‌پذیر خواهند بود.

۸. دیتای حساس مدیریت نشده

بسیاری از شرکت‌ها تلاش می‌کنند که یک موجودی دقیق از دیتابیس خود و اشیاء دیتای مهم آن را حفظ و نگهداری نمایند. به فراموشی سپردن این دیتابیس‌ها، که ممکن است شامل اطلاعات حساس باشند، برای سازمان خطرناک است؛ چراکه ممکن است مثلاً در محیط آزمون نرم‌افزار، به‌دوراز چشم تیم امنیتی، اطلاعات در معرض خطر قرار گیرد.

متأسفانه در برخی از سازمان‌ها، مدیران IT ضعیف و یا بی‌تفاوت، توانایی مجتمع‌سازی و یکپارچه‌کردن دیتابیس‌ها را ندارند؛ به همین دلیل هر برنامه‌ای که نیاز به دیتابیس دارد، به‌صورت مستقل و بدون اطلاع مدیرIT، برای خود یک دیتابیس نصب‌کرده و از آن استفاده می‌کند. لذا دیتابیس‌ها به‌صورت پراکنده در سازمان پخش‌شده و مدیریت متمرکزی ندارند. وجود دیتابیس‌های متعدد و پراکنده، نه‌تنها باعث می‌شود که از منابع سرورها (مثل CPU و RAM) استفاده نادرست گردد؛ بلکه نگهداری، حفاظت و پشتیبان‌گیری از آن‌ها را نیز در معرض خطر قرار می‌دهد.

اطلاع نداشتن مدیر IT از تعداد دیتابیس‌ها و دارایی‌های اطلاعاتی سازمان، باعث می‌گردد که یک طراحی امنیتی ضعیف صورت گرفته و مدیریت ریسک قابل قبولی نیز پیش‌بینی نشده باشد.[۱]

۹. منع خدمت

منع خدمت، یک دسته‌بندی کلی از حملات است که از دسترسی کاربران به خدمت موردنیاز بر روی سرور یا شبکه جلوگیری می‌نماید. حملات DoS از روش‌های گوناگونی اجرا می‌شود، اما روش متداول در محیط‌های دیتابیس این است که توسط سیلی از درخواست‌های دیتابیسی، یک سربار قابل‌توجه به منابع سرور مثل CPU و حافظه اعمال کرده و باعث شکست آن می‌شوند.

معمولاً هدف از حملات منع خدمت باج‌گیری از سازمان موردنظر است. افرادی که اقدام به انجام حملات DoS می‌نمایند، سازمان را تهدید می‌کنند که تا زمان اجرای خواسته آن‌ها، حملات خود را ادامه داده و از سرویس‌دهی به کاربران مجاز، جلوگیری به عمل می‌آورند.

---

[۱] البته مدیری که توانایی داشتن لیستی از دارایی‌های اطلاعاتی سازمان خود را ندارد، قطعاً در سایر کارها و امور مدیریتی نیز بی‌تفاوت و بی‌کفایت است.

۱۰. تخصص و آموزش امنیتی محدود

کنترل‌های امنیتی داخلی با سرعت رشد دیتا پیش نمی‌روند، و بسیاری از سازمان‌ها برای مقابله با نقض امنیتی به‌درستی مجهز نشده‌اند. این اتفاق اغلب به علت عدم تخصص موردنیاز جهت پیاده‌سازی کنترل‌ها، سیاست‌ها و آموزش‌های امنیتی است.

بر اساس گزارش PWC[1] در سال 2012، ۷۵ درصد از سازمان‌های موردبررسی، تجربه نقض مربوط به کارکنان را در هنگامی‌که درک ضعیفی از سیاست امنیتی شده بود، داشته‌اند. همچنین ۵۴ درصد از کسب‌وکارهای کوچک، هیچ برنامه‌ای برای آموزش کارکنان خود در مورد خطرات امنیتی نداشته‌اند.

[1] http://www.pwc.com

☑ مبحث دوم

روش‌های مقابله با تهدیدات

امنیت دیتابیس فقط به پیکربندی دیتابیس ختم نمی‌شود؛ بلکه باید یک سیاست امنیتی متشکل از مجموعه کنترل‌های فیزیکی، شبکه و سیستم را برای برقراری امنیت، پیاده‌سازی نمود. در سازمان‌های بزرگ، معمولاً بخش‌های مختلف دارای مدیران مستقل می‌باشند؛ به‌عنوان‌مثال Network Admin، Server Admin و Software Admin، با مدیر دیتابیس یا همان DB Admin متفاوت خواهد بود. واضح است که برای کارایی مطلوب، وظایف امنیتی باید توسط تمام این افراد و بر اساس سیاست‌های امنیتی سازمان و تحت نظر یک مشاور امنیتی، به‌صورت یکپارچه انجام پذیرد. اما فارغ از اینکه مدیریت سازمان این بستر را به چه نحوی آماده‌سازی می‌کند، وظیفه DB Admin است که حداقل وظایف خود را در ارتباط با امنیت دیتابیس انجام داده و سایر مواردی که خارج از حیطه مسئولیت مدیر دیتابیس است و یا نیاز به فراهم آوردن محصولات دیگری است را به اطلاع مدیر خود برساند.

در این مبحث نیز به بیان روش‌های مقابله با تهدیدات دیتابیس می‌پردازیم. مدیر دیتابیس می‌تواند برخی از این روش‌ها را به‌صورت مستقل انجام داده و تا حدی دیتابیس خود را امن نماید، و البته برای برخی دیگر از این روش‌ها، نیاز به محصولات جانبی بوده و یا خارج از حوزه مدیریت دیتابیس است.

استراتژی دفاعی چندلایه برای امنیت دیتابیس

در مبحث اول ده تهدید برتری که دیتابیس‌ها را در معرض خطر قرار می‌دهند، به‌صورت خلاصه تشریح گردید. همان‌طور که قبلاً گفته شد، با راهکارهای ساده و اجرای بهترین شیوه‌ها می‌توان از این تهدیدات جلوگیری به‌عمل آورد.

اما ازآنجاکه روش‌های حمله مختلفی بر اساس هر یک از تهدیدات فوق وجود دارد، لذا یک استراتژی دفاعی چندلایه برای نگهداری امن دیتابیس، موردنیاز است.

جدول زیر راه‌های دفاعی را بر اساس ده تهدید برتر نمایش داده است. البته راه‌های ذکرشده در این جدول نیز در ادامه به‌صورت خلاصه تشریح می‌شوند.[1]

[1] http://www.imperva.com

	Solution \ Treat	1	2	3	4	5	6	7	8	9	10
Discovery and Assessment	Scan for Vulnerabilities			*	*					*	
	Calculate Risk Scores			*				*			
	Mitigate Vulnerabilities			*				*		*	
	Identify Compromised Endpoints				*			*			
	Analyze Risk and Prioritize Remediation Efforts							*			
	Discover Database Servers								*		
	Analyze Discovery Results								*		
	Identify and Classify Sensitive Data	*							*		
User Rights Management	Aggregate Access Rights	*	*								
	Enrich Access Rights Information	*	*								
	Identify and Remove Excessive Rights	*	*		*						
	Review and Approve/Reject Individual User Rights	*									
	Extract “Real” User Identity					*					
Monitoring and Blocking	Real-Time Alerting and Blocking	*	*	*						*	
	Detect Unusual Access Activity	*	*	*	*					*	
	Block Malicious Web Requests			*							
	Monitor Local Database Activity					*					
	Impose Connection Controls									*	
	Validate Database Protocols									*	
	Response Timing									*	
Auditing	Automate Auditing					*					
	Capture Detailed Transactions					*					
	Generate Reports for Compliance and Forensics					*					
Data Protection	Archive External Data					*					
	Encrypt Databases						*				
Non-Tech. Security	Cultivate Experienced Security Professionals										*
	Educate Your Workforce										*

همان‌طور که در جدول فوق مشاهده می‌نمایید، راه‌حل‌های امنیتی پایگاه داده در شش گروه مختلف و به‌صورت زیر تقسیم‌بندی شده است:

۱. **کشف و ارزیابی (Discovery and Assessment)**
کشف و ارزیابی محل آسیب‌پذیری‌های دیتابیس و دیتای حیاتی.
۲. **مدیریت حقوق کاربر (User Rights Management)**
شناسایی حقوق بیش‌ازحد تعلق‌گرفته، بر روی دیتای حساس.
۳. **نظارت و مسدودسازی (Monitoring and Blocking)**
محافظت از دیتابیس در برابر حمله، سرقت و از دست رفتن داده‌ها.
۴. **حسابرسی (Auditing)**
در بررسی تطابق با مقررات، کمک می‌کند.
۵. **حفاظت از اطلاعات (Data Protection)**
اطمینان از صحت و محرمانگی دیتا.
۶. **امنیت غیرفنی (Non-Technical Security)**
فهماندن و تقویت فرهنگ آگاهی‌ها و آمادگی‌های امنیتی.

هر یک از شش گروه فوق، دارای چند زیرگروه نیز است. در ادامه به بررسی زیرمجموعه مربوط به هر گروه می‌پردازیم.

۱. کشف و ارزیابی (Discovery and Assessment)

- **اسکن آسیب‌پذیری (Scan for Vulnerability)**

اطلاع از آسیب‌پذیری‌هایی که دیتابیس را در معرض Input Injection قرار می‌دهند، ضروری است. بدافزارها ممکن است به دنبال بهره‌برداری از آسیب‌پذیری‌های شناخته شده باشند؛ لذا یک دیتابیسی که وصله‌ها بر روی آن نصب نشده، می‌تواند یک هدف آسان برای هکر باشد. همچنین قوانین احرازهویت ضعیف ممکن است منجر به حمله DoS شود.

می‌توان از ابزارهای ارزیابی آسیب‌پذیری برای شناسایی آسیب‌پذیری‌های امنیتی، اشتباه در پیکربندی، و وصله‌های ازدست‌رفته، استفاده نمود. ارزیابی باید بر اساس بهترین شیوه‌های امنیتی دیتابیس صورت پذیرد. (مثل معیارهای ارائه‌شده توسط DISA STIG[1])

[1] http://iase.disa.mil/stigs

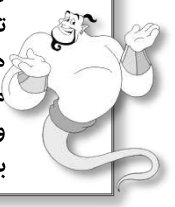
- **محاسبه امتیاز ریسک (Calculate Risk Scores)**

امتیازدهی به ریسک‌ها بر اساس شدت آسیب‌پذیری و حساسیت دیتا صورت می‌پذیرد. مقدار شدت، باید بر اساس یک سیستم شناخته‌شده مثل CVSS[1] تعیین شود. امتیازدهی به ریسک می‌تواند کمک خوبی در اولویت‌بندی ریسک، مدیریت، و تحقیق درباره آسیب‌پذیری‌ها باشد. هر چه مقدار امتیاز به‌دست‌آمده در این مورد بیشتر باشد، خطر ابتلا به Input Injection نیز بیشتر خواهد بود.

- **کاهش آسیب‌پذیری (Mitigate Vulnerabilities)**

اگر یک آسیب‌پذیری کشف‌شده باشد ولی تولیدکننده دیتابیس هنوز وصله‌ای برای آن ارائه نکرده باشد؛ باید از یک وصله مجازی برای کاهش آسیب‌پذیری استفاده نمود. وصله‌های مجازی، تلاش‌های انجام‌گرفته برای بهره‌برداری از آسیب‌پذیری را بدون نیاز به وصله واقعی و یا اعمال تغییر در تنظیمات فعلی، ناکام می‌گذارد. به‌عبارت‌دیگر، وصله مجازی تا زمان ارائه وصله اصلی، از دیتابیس مراقبت می‌کند.

البته توجه داشته باشید که وصله مجازی، صرفاً یک راه‌حل کوتاه‌مدت است؛ و باید وصله اصلی هر چه سریع‌تر آماده و نصب گردد.



وصله مجازی یا Virtual Patch، یک اصطلاح است که در چند سال اخیر توسط برخی تولیدکنندگان محصولات امنیتی مخصوصاً IPS و WAF، مورداستفاده قرارگرفته است. هرچند که وصله مجازی مختص به برنامه‌های تحت‌وب نیست و درباره سایر پروتکل‌ها نیز مصداق دارد؛ اما بیشتر در تجهیزات WAF مورداستفاده قرار می‌گیرد.
وصله‌های مجازی از ترافیک احتمالی که قصد بهره‌برداری از آسیب‌پذیری را دارد، جلوگیری به عمل می‌آورد.

- **شناسایی نقاط پایانی به خطر افتاده (Identify Compromised Endpoints)**

با شناسایی میزبان‌هایی که توسط کدهای مخرب آلوده‌شده، می‌توانید از دسترسی آن‌ها به اطلاعات حساس در دیتابیس جلوگیری به عمل‌آورید. پس‌ازآنکه شما موفق به شناسایی دستگاه آلوده شدید، باید سریعاً کنترل‌هایی را جهت محدود کردن دسترسی به داده‌ها، اجرا نمایید.

- **تحلیل ریسک و اولویت‌بندی بازسازی**
(Analyze Risk and Prioritize Remediation Efforts)

استفاده از ابزارهای گزارشی و تحلیلی برای درک خطرات و اولویت‌بندی بازسازی.

[1] Common Vulnerability Scoring System (CVSS)

- **پیداکردن سرورهای دیتابیس (Discover Database Servers)**

به‌منظور ایجاد و حفظ یک فهرست از دیتابیس‌ها، و همچنین جداسازی اطلاعات حساس موجود در آن‌ها، باید ابتدا تمام دیتابیس‌های موجود در سازمان و مراکز داده، شناسایی گردند. برای این کار می‌توان از ابزار جستجوگر دیتابیس استفاده نمود.

- **تجزیه‌وتحلیل نتایج کشف دیتابیس (Analyze Discovery Results)**

نتایج به‌دست‌آمده از کشف دیتابیس و طبقه‌بندی آن‌ها، باید مورد تجزیه‌وتحلیل قرار گیرد؛ تا مشخص شود که کدام دیتای حساس، نیاز به نظارت و مراقبت بیشتر دارد.

- **شناسایی و طبقه‌بندی دیتای حساس (Identify and Classify Sensitive Data)**

پس‌ازآنکه یک فهرست از دیتابیس‌ها به‌وجود آمد، بسیار مهم است که بفهمید کدام دیتابیس حاوی دیتای حساس است؛ لذا باید اشیاء، ردیف و ستون دیتابیس را برای شناسایی دیتای حساس، اسکن نمایید. باید از راه‌حل طبقه‌بندی دیتا برای مشخص کردن انواع داده‌ها مثل کارت اعتباری، آدرس پست الکترونیک و شماره هویت ملی، و اینکه کدام کاربر قادر به اضافه کردن انواع دیتای سفارشی است، استفاده نمایید.

۲. مدیریت حقوق کاربر (User Rights Management)

- **تجمیع حقوق دسترسی (Aggregate Access Rights)**

باید برای مشخص شدن هر دو حقوق اعطاشده و ممتاز، و همچنین جزئیات حقوق دسترسی (مانند Connect، Delete، Select و غیره)، و اینکه چه کسی آن حقوق را اعطا و چه کسی آن را دریافت کرده و بر روی چه Objectهایی اعمال‌شده، دیتابیس‌ها را اسکن نمود. تجمیع حقوق کاربر در یک منبع، کمک می‌کند که گزارش‌گیری و تحلیل دسترسی کاربر به دیتای حساس، ساده گردد.

- **غنی‌سازی اطلاعات حقوق دسترسی با استفاده از جزئیات اطلاعات کاربر و حساسیت دیتا**

(Enrich Access Rights Information with User Details and Data Sensitivity)

اضافه کردن اطلاعات مربوط به نقش کاربر و رفتار دیتابیس، ارزش قابل‌توجهی در تحلیل حقوق کاربر داشته و سوءاستفاده از امتیازات را در حد صفر کاهش می‌دهد. جزئیات متنی شامل نام کاربری، گروه، حساسیت Object دیتابیس، و زمان آخرین دسترسی، باید جمع‌آوری و به اطلاعات حقوق کاربر اضافه گردد. این کار به شما اجازه می‌دهد تا بر روی تجزیه‌وتحلیل حقوق دسترسی که بالاترین ریسک کسب‌وکار محسوب می‌شود، تمرکز نمایید.

- **شناسایی و حذف حقوق بیش‌ازحد و کاربران خفته**
  (Identify and Remove Excessive Rights and Dormant Users)

کاربرانی که دارای حقوق بیش‌ازاندازه بوده و همچنین کاربرانی که از امتیازات خود استفاده نمی‌کنند (کاربران خفته)، باید شناسایی گردند. این کار کمک می‌کند تا وظایف کاربران و حقوق تعلق‌گرفته به آنان مجدد موردبررسی قرار گیرد.

یکی از راه‌های نفوذ هکرها، استفاده از حقوق بیش‌ازاندازه و یا استفاده‌نشده کاربران است. لذا با محدود کردن کاربران به حقوق دسترسی موردنیاز، می‌توان از بسیاری از تهدیدات جلوگیری به‌عمل آورد.

- **بررسی و تصویب/رد حقوق کاربران به‌صورت جداگانه**
  (Review and Approve/Reject Individual User Rights)

باید حقوق تعلق‌گرفته به افراد موردبررسی قرار گیرد و تعیین شود آیا این حقوق مناسب شخص موردنظر است یا خیر. پس از بررسی، باید حقوق تعلق‌گرفته به کاربر تصویب و یا رد شده و یا اینکه به شخص دیگری واگذار گردد.

- **استخراج واقعی هویت کاربر (Extract "Real" User Identity)**

توسط راهکاری، ارتباط اطلاعات کاربر با تراکنش‌های دیتابیس را مشخص و ثبت نمایید. باید مشخص شود که هر تراکنش انجام شده، دقیقاً توسط چه کاربری انجام گرفته است. درنتیجه واقعه‌نگاری نیز دقیق و بر اساس مشخصات کاربر انجام گرفته و باعث تسهیل بازرسی‌های بعدی خواهد شد.

۳. نظارت و مسدودسازی (Monitoring and Blocking)

- **هشدار و مسدودسازی بلادرنگ (Real-Time Alerting and Blocking)**

برای تشخیص نشت اطلاعات، تراکنش‌های دیتای بزرگ و غیرمجاز SQL، و حملات، باید یک نظارت بلادرنگ بر تمام فعالیت‌های دسترسی به دیتابیس و الگوهای استفاده، وجود داشته باشد. وقتی‌که تلاشی برای دسترسی غیرمجاز رخ می‌دهد، باید یک هشدار تولیدشده و یا به آن نشست (Session) کاربر خاتمه داده شود. همچنین ترافیک دیتابیس باید موردبررسی قرارگرفته و برای تشخیص حملات، با الگوهای حملات شناخته‌شده مثل DoS و فعالیت‌های غیرمجاز، مطابقت داده شود.

سیاست‌های امنیتی نه‌تنها برای تشخیص سوءاستفاده از امتیازات بیش‌ازحد کاربران، بلکه برای محافظت دیتابیس در برابر اغلب ده حمله برتر، نیز مفید است.

- **تشخیص فعالیت دسترسی غیرمعمول (Detect Unusual Access Activity)**

یک پروفایل جامع از فعالیت‌های طبیعی هر کاربر دیتابیس، ایجاد گردیده و از آن به‌عنوان پایه‌ای برای تشخیص DoS، بدافزار، Input Injection، و فعالیت‌های غیرعادی استفاده گردد. اگر کاربری که می‌خواهد یک ارتباط با دیتابیس را آغاز کند، دارای پروفایل مناسب نباشد، باید رویداد مربوطه ثبت‌شده و سپس یک پیام هشدار، تولید و یا کاربر مسدود گردد. ایجاد پروفایل مبتنی بر فعالیت برای هر کاربر، احتمال تشخیص دسترسی نامناسب به دیتای حساس را افزایش می‌دهد.

- **مسدود کردن درخواست‌های مخرب وب (Block Malicious Web Requests)**

ازآنجاکه برنامه‌های تحت وب، شایع‌ترین مسیر شروع یک حمله تزریق ورودی است؛ لذا باید به‌عنوان اولین سنگر دفاعی، از یک Web Application Firewall (WAF) استفاده نمود. تجهیزات WAF، حملات تزریق ورودی که از برنامه‌های تحت وب نشأت‌گرفته را شناسایی و مسدود می‌نمایند.

- **نظارت بر فعالیت‌های محلی دیتابیس (Monitor Local Database Activity)**

باید راهکاری برای بازرسی و نظارت بر روی فعالیت کاربرانی که دارای امتیازات بالا و ویژه هستند، فراهم گردد. این امتیازات ویژه معمولاً به Administratorها و مدیران سیستم اعطا می‌شود تا بتوانند وظایف سازمانی خود را انجام دهند. اما اگر این حد ویژه از اختیارات در معرض خطر سوءاستفاده توسط کدهای مخرب قرار گیرد، باعث افزایش چشمگیر ریسک سازمان می‌گردد.

- **اعمال کنترل‌های اتصال (Impose Connection Controls)**

باید نرخ اتصال[1]، نرخ درخواست[2]، و دیگر متغیرها، به ازای هر کاربر دیتابیس محدود گردد تا از ایجاد سربار بر روی منابع سرور جلوگیری به عمل آید.

- **اعتبارسنجی پروتکل‌های دیتابیس (Validate Database Protocols)**

راهکاری در نظر گرفته شود که بر روی فعالیت دیتابیس نظارت کرده و بتواند پروتکل را تحلیل و ارتباطات غیرعادی را ایزوله نماید. زمانی که رویدادهای ارتباطی غیرمعمول شناسایی می‌شود، این راهکار باید توانایی تولید هشدار و یا مسدود کردن تراکنش را داشته باشد.

پروتکل‌هایی مورداستفاده در دیتابیس، باید تحت نظارت و اعتبار سنجی قرار گیرند؛ تا در صورت بروز هرگونه تخطی، سریعاً مدیر سیستم را آگاه سازند.

---

[1] Connection Rates
[2] Query Rates

- **زمان پاسخ (Response Timing)**

حملات DoS دیتابیس به‌گونه‌ای طراحی می‌شوند که با اعمال اضافه‌بار به منابع سرور، باعث ایجاد تأخیر در پاسخگویی سرور به کاربران مجاز گردد. لذا باید از راهکاری برای نظارت بر زمان پاسخ استفاده نمود؛ تا زمانی که در سیستم تأخیر یا کندی در جواب درخواست‌ها مشاهده می‌شود، پیام هشدار تولید نماید.

۴. حسابرسی (Auditing)

- **حسابرسی خودکار (Automate Auditing)**

استفاده از راهکاری که بتواند حسابرسی را به‌صورت مستقل از پایگاه داده فراهم آورد. این راهکار باید بتواند در صورت وجود دیتابیس‌های مختلف، یک حسابرسی یکپارچه را در سازمان ارائه نماید. همچنین به دلیل جدا بودن این حسابرسی از سرور دیتابیس، باعث می‌گردد که منابع دیتابیس صرف وظایف حسابرسی نشود.

- **ضبط جزئیات تراکنش‌ها (Capture Detailed Transactions)**

برای رعایت الزامات قانونی و تشخیص تقلب، باید راهکاری پیاده‌سازی شود که توانایی ضبط وقایع حسابرسی، شامل جزئیاتی مانند نام منبع، متن کامل query، نام میزبان و اطلاعاتی از این قبیل، را داشته باشد.

- **ایجاد گزارش برای پذیرش و دادگاه**

(Generate Reports for Compliance and Forensics)

گزارش‌هایی با قالب قانونی باید جهت اثبات ادعا و استفاده در دادگاه، ایجاد گردد.

۵. حفاظت از اطلاعات (Data Protection)

- **آرشیو خارجی دیتا (Archive External Data)**

راهکاری اتخاذ گردد که به‌صورت دوره‌ای اقدام به آرشیو اطلاعات بر روی تجهیزات ذخیره‌سازی خارجی (بیرون از سرور دیتابیس) نماید. این داده‌ها می‌توانند بر روی تجهیزات ذخیره‌سازی، به‌صورت فشرده، رمز شده و یا امضاشده نگهداری شوند.

- **رمزگذاری دیتابیس (Encrypt Databases)**

مکانیسمی اتخاذ نمایید که تمام اطلاعات حساس در دیتابیس‌ها رمزگذاری شود. بالطبع در این صورت هم نسخه‌های اصلی و هم نسخه‌های پشتیبان از دیتای حساس، به‌صورت محرمانه نگهداری خواهند شد.

۶. امنیت غیر فنی (Non-Technical Security)

- **پرورش کارشناس حرفه‌ای امنیت**
  (Cultivate Experienced Security Professionals)

برای دفاع در مقابل صف رو به رشد تهدیدات داخلی و خارجی، استخدام پرسنل امنیت اطلاعات که به‌خوبی امنیت را بشناسند و دارای تجربه‌های ارزشمندی در رابطه با پیاده‌سازی، مدیریت و نظارت بر روی راه‌حل‌های امنیتی باشند، امری ضروری است. آموزش مداوم پرسنل و ارتقای دانش و مهارت‌های امنیتی نیز مهم است.

همچنین می‌توانید از مشاوران و کارشناسان خارج از سازمان نیز برای پیاده‌سازی، ارزیابی امنیتی، تست نفوذ و آموزش مدیران و پرسنل خود بهره‌مند شوید.

- **آموزش کارمندان (Educate Your Workforce)**

آموزش کارمندان درباره تکنیک‌های کاهش ریسک ازجمله چگونگی تشخیص تهدیدات سایبری (مثل فیشینگ)، یکی از بهترین شیوه‌های امنیتی مورداستفاده است. عدم آموزش و ایجاد آگاهی امنیتی، احتمال بروز نقض امنیتی را افزایش می‌دهد.

آموزش کارمندان هم می‌تواند از حملاتی مثل مهندسی اجتماعی جلوگیری نماید، و هم می‌تواند باعث جلوگیری از بروز اشتباهات انسانی توسط کارمندان سازمان گردد.



منابع و معرفی کتاب

منابع:

[۱] روغنی، محمدتقی. شبکه؛ صفر تا صد. انتشارات ناقوس، ۱۳۹۲
[2] Stalling, William. *CRYPTOGRAPHY AND NETWORK SECURITY PRINCIPLES AND PRACTICE*. Fifth Edition.
[3] ISO/IEC 27000:2014
[4] ISO/IEC 27001:2013
[5] ISO/IEC 27002:2013
[6] NIST FIPS-197 (AES)
[7] NIST FIPS-198 (HMAC)
[8] IETF RFC 2104. *Keyed-Hashing for Message Authentication*.
[9] NIST FIPS-186 (DSS)
[10] NIST FIPS-180 (SHA)
[11] IETF RFC 3174. *Secure Hash Algorithm.*
[12] Barker, Keith. *CCNA Security 640-554 Official Cert Guide*. Cisco Press.
[13] Wilkins, Sean. *CCNP Security SECURE 642-637 Official Cert Guide*.
[14] NIST FIPS-199. *Standards for Security Categorization of Federal Information and Information Systems*.
[15] Whitman, Michael E. *Principles of Information Security*. Fourth Edition
[16] ITU X.800. *Security architecture for Open Systems Interconnection for CCITT applications.*
[17] IETF RFC 2828. *Internet Security Glossary*.
[18] IETF RFC 4962. *Guidance for Authentication, Authorization, and Accounting (AAA).*
[19] NIST SP 800-10.
[20] IETF RFC 2827. *Network Ingress Filtering.*
[21] NIST SP 800-94. *Guide to Intrusion Detection and Prevention Systems (IDPS).*
[22] IETF RFC 4301. *Security Architecture for the Internet Protocol.*
[23] IETF RFC 2637. *Point-to-Point Tunneling Protocol (PPTP).*
[24] IETF RFC 2661. *Layer Two Tunneling Protocol (L2TP).*

[25] NIST SP 800-113. *Guide to SSL VPNs*.
[26] IETF RFC 6101. *The Secure Sockets Layer (SSL) Protocol Version 3.0*.
[27] IETF RFC 2818. *HTTP Over TLS*.
[28] IETF RFC 4880. *OpenPGP Message Format*.
[29] IETF RFC 5322. *Internet Message Format*.
[30] IETF RFC 2045 – 2049. *Multipurpose Internet Mail Extensions (MIME)*.
[31] IETF RFC 2119. *Key words for use in RFCs to Indicate Requirement Levels*.
[32] IETF RFC 2315. *Cryptographic Message Syntax*.
[33] Setty, Harish. *System Administrator - Security Best Practices*. SANS Institute.
[34] NIST SP 800-123. *Guide to General Server Security*.
[35] Cole, Eric. *Security Best Practices*. Secure Anchor.
[36] Cui-Qing Yang. *Operating System Security and Secure Operating Systems*.
[37] DoD 5200.28-STD.*Trusted Computer System Evaluation Criteria*. TCSEC.
[38] NIST SP 800-40. *Guide to Enterprise Patch Management Technologies*.
[39] NIST SP 800-162. *Guide to Attribute Based Access Control (ABAC) Definition and Considerations*.
[40] Basta, Alfred. *Database Security*. Course Technology.
[41] *Top Ten Database Threats*. imperva.com.
[42] http://www.ietf.org
[43] https://cve.mitre.org
[44] http://www.securityfocus.com
[45] https://www. Intrusion.com
[46] http://www.verizonenterprise.com
[47] http://www.idc.com
[48] https://www.otalliance.org
[49] http://www.pwc.com
[50] http://www.imperva.com
[51] http://iase.disa.mil/stigs


NAGHOOS
PUBLICATION
0-100
شبکه
صفر تا صد
* سازمانهای توسعه دهنده شبکه و اینترنت
* مدلها، استانداردها، پروتکل‌ها و اصطلاحات شبکه
* شبکه های LAN و WAN بر اساس پروتکل های IPv4 و IPv6
* اجرای ۱۹ سناریوی عملیاتی بر اساس IPv4 و IPv6 در موضوعات سوئیچینگ و مسیریابی
* امنیت شبکه ضمن بررسی مدل امنیتی سیسکو و استاندارد سیستم مدیریت امنیت اطلاعات (ISMS)
* توصیه شده برای مدیران و مشاوران IT ، کارشناسان و علاقه‌مندان به شبکه
محمدتقی روغنی
MFP



کتاب: شبکه؛ صفر تا صد
مؤلف: محمدتقی روغنی

چاپ اول؛ تابستان ۱۳۹۲
۵۴۶ صفحه، وزیری
انتشارات ناقوس

در این کتاب سعی شده مطالب مربوط به شبکه از صفر تا صد به‌صورت علمی، عملیاتی و با نگارشی روان و قابل‌فهم ارائه گردد. همچنین با اجرای ۱۹ سناریو عملیاتی، ضمن ارائه مطالب علمی و فنی، پیکربندی تجهیزات را نیز آموزش داده تا خواننده پس از پایان این کتاب بتواند به‌عنوان یک کارشناس شبکه با توانایی انجام کار عملیاتی معرفی گردد.

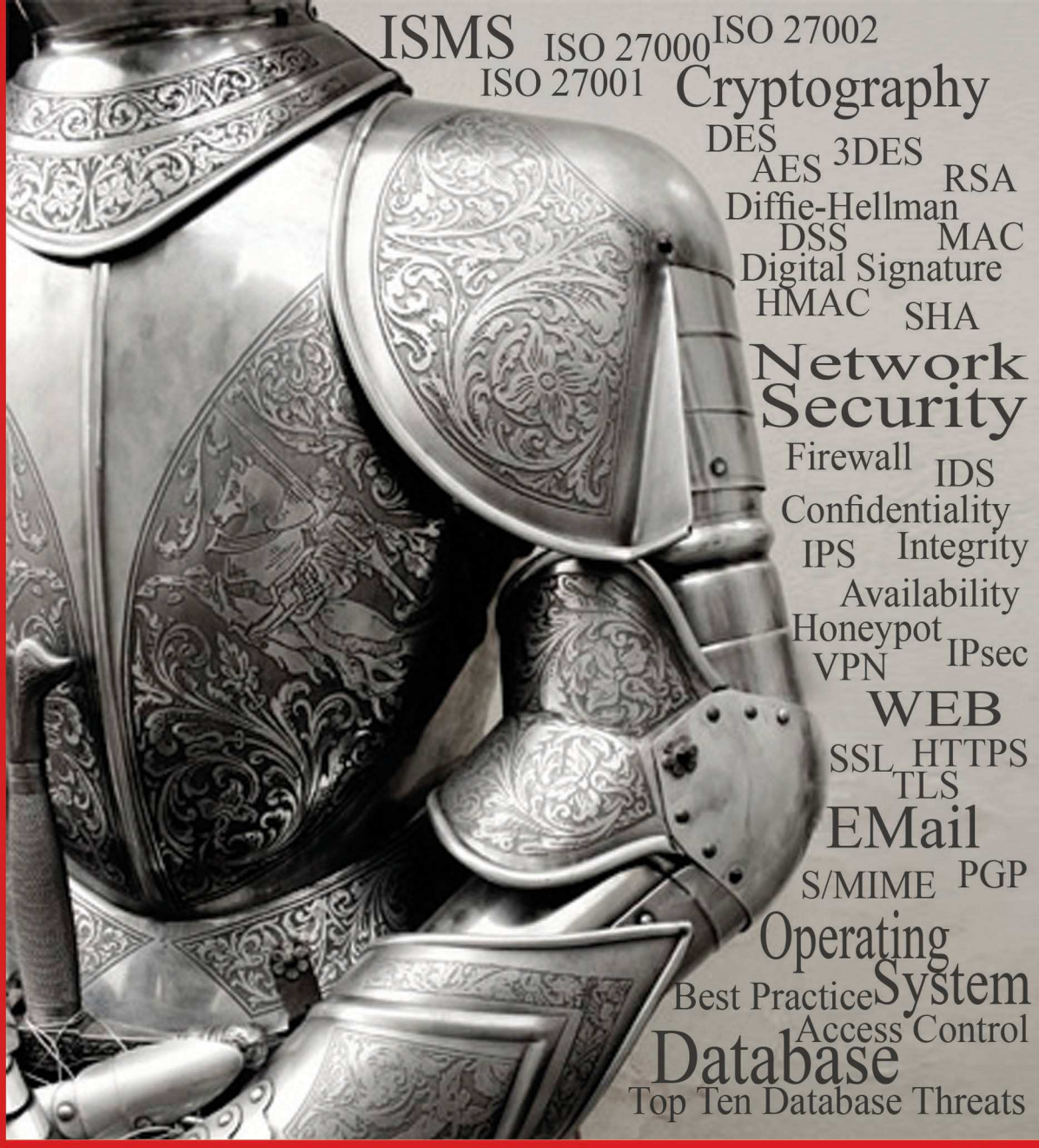
ازآنجاکه خواست نگارنده بر آن بوده که این کتاب بتواند نیاز طیف وسیعی از علاقه‌مندان به شبکه را برآورده سازد؛ لذا مطالب در بخش‌ها، فصل‌ها و مباحث مختلفی طبقه‌بندی گردیده تا خواننده بتواند به‌راحتی به مطلب موردنظر دسترسی پیدا نماید.

شروع کتاب با معرفی سازمان‌های بین‌المللی و شرکت‌های پیشرو درزمینه شبکه شروع شده، سپس به پروتکل‌های اولیه و پایه موردنیاز در شبکه‌های کوچک پرداخته می‌شود. این روند به‌صورت تکاملی ادامه می‌یابد تا به شبکه‌های محلی بزرگ‌تر و پروتکل‌های مسیریابی در شبکه‌های گسترده می‌رسیم. بررسی پروتکل‌های شبکه‌های محلی و گسترده، هم بر اساس IPv4 و هم IPv6 انجام شده است. در بخش پایانی نیز به امنیت و استانداردهای مربوط به شبکه پرداخته شده است.

با توجه به مطالب ذکرشده، این کتاب برای مدیران و مشاوران IT، مدیران شبکه، کارشناسان و علاقه‌مندان به شبکه، با هر سطحی از دانش، پیشنهاد می‌گردد.

INFORMATION SECURITY

Mohammad Taghi Roghani



INFORMATION SECURITY

Mohammad Taghi Roghani