۱) مقدمه

تا یکی دو دهه قبل شبکه های کامپیوتری معمولاً در دو محیط وجود خارجی داشت:

- محیطهای نظامی که طبق آئین نامه های حفاظتیِ ویژه بصورت فیزیکی حراست می‌شد و چون سایتهای ارتباطی خودشان هم در محیط حفاظت شدهٔ نظامی مستقر بود و هیچ ارتباط مستقیم با دنیای خارج نداشتند ، لذا دغدغه کمتری برای حفظ اسرار و اطلاعات وجود داشت. (نمونه بارز این شبکه ARPANET در وزارت دفاع آمریکا بود)
- محیطهای علمی و دانشگاهی که برای مبادله دستاوردهای تحقیقی و دسترسی به اطلاعات علمی از شبکه استفاده می‌کردند و معمولاً بر روی چنین شبکه هائی اطلاعاتی مبادله می‌شد که آشکار شدن آنها لطمه چندانی به کسی وارد نمی‌کرد. (اداراتی هم که اطلاعات محرمانه و سری داشتند معمولاً از کامپیوترهای Mainframe استفاده می‌کردند که هم مدیریت و حراست ساده تری نیاز دارد و هم کنترل کاربران آن بصورت فیزیکی ساده است)

با گسترش روز افزون شبکه های بهم پیوسته و ازدیاد حجم اطلاعات مورد مبادله و متکی شدن قسمت زیادی از امور روزمره به شبکه های کامپیوتری و ایجاد شبکه های جهانی چالش بزرگی برای صاحبان اطلاعات پدید آمده است . امروزه سرقت دانشی که برای آن هزینه و وقت ، صرف شده یکی از خطرات بالقوه شبکه‌های کامپیوتری به شمار می‌آید .

در جهان امروز با محوّل شدن امور اداری و مالی به شبکه های کامپیوتری زنگ خطر برای تمام مردم به صدا در آمده است و بر خلاف گذشته که خطراتی نظیر دزدی و راهزنی معمولاً توسط افراد کم سواد و ولگرد متوجه مردم بود امروزه این خطر توسط افرادی تحمیل می‌شود که با هوش و باسوادند (حتی باهوش تر از افراد معمولی) و قدرت نفوذ و ضربه به شبکه را دارند . معمولاً هدف افرادی که به شبکه های کامپیوتری نفوذ یا حمله می‌کنند یکی از موارد زیر است :

- تفریح یا اندازه گیری ضریب توانائی فردی یا کنجکاوی (معمولاً دانشجویان!)
- دزدیدن دانشی که برای تهیه آن بایستی صرف هزینه کرد. (راهزنان دانش)
- انتقام جوئی و ضربه زدن به رقیب
- آزار رسانی و کسب شهرت از طریق مردم آزاری (بیماران روانی)

- جاسوسی و کسب اطلاع از وضعیت نظامی و سیاسی یک کشور یا منطقه
- رقابت ناسالم در عرصه تجارت و اقتصاد
- جابجا کردن مستقیم پول و اعتبار از حسابهای بانکی و دزدیدن شماره کارتهای اعتبار
- کسب اخبار جهت اعمال خرابکاری و موذیانه (توسط تروریستها)

بهر حال امروزه امنیت ملی و اقتدار سیاسی و اقتصادی به طرز پیچیده ای به امنیت اطلاعات گره خورده و نه تنها دولتها بلکه تک تک افراد را نیز تهدید می‌کند. برای ختم مقدمه از شما سوال می‌کنیم که چه حالی به شما دست می‌دهد وقتی متوجه شوید که شماره حساب بانکی یا کارت اعتباریتان توسط فرد ناشناسی فاش شده و انبوهی هزینه روی دست شما گذاشته است؟ پس بعنوان یک فرد مطلع از خطراتی که یک شبکه کامپیوتری را تهدید می‌کند این فصل را دنبال کنید.

۱-۱) سرویسهای امنیتی در شبکه ها

تهدیدهای بالقوه برای امنیت شبکه های کامپیوتری بصورت عمده عبارتند از:

- فاش شدن غیر مجاز اطلاعات در نتیجهٔ استراق سمع داده‌ها یا پیامهای در حال مبادله روی شبکه
- قطع ارتباط و اختلال در شبکه به واسطه یک اقدام خرابکارانه
- تغییر و دستکاری غیر مجاز اطلاعات یا یک پیغام ارسال شده

بایستی با مفاهیم اصطلاحات زیر بعنوان سرویسهای امنیتی آشنا باشید:

الف) محرمانه ماندن اطلاعات[۱]: دلایل متعددی برای یک سازمان یا حتی یک فرد عادی وجود دارد که بخواهد اطلاعات خود را محرمانه نگه دارد.

ب) احراز هویت[۲]: پیش از آنکه محتوای یک پیام یا اطلاعات اهمیت داشته باشد باید مطمئن شوید که پیام حقیقتاً از کسی که تصور می‌کنید رسیده است و کسی قصد فریب و گمراه کردن (یا آزار) شما را ندارد .

---

[۱] Confidentality
[۲] Authentication

ج) سلامت داده‌ها[۱]: یعنی دست نخوردگی و عدم تغییر پیام و اطمینان از آنکه داده‌ها با اطلاعات مخرب مثل یک ویروس کامپیوتری آلوده نشده اند.

د) کنترل دسترسی[۲]: یعنی مایلید دسترسی افرادی را که مجاز نیستند ، کنترل کنید و قدرت منع افرادی را که از دیدگاه شما قابل اعتماد به شمار نمی‌آیند از دسترسی به شبکه داشته باشید.

ه) در دسترس بودن[۳]: با این تفاصیل ، باید تمام امکانات شبکه بدون دردسر و زحمت در اختیار آنهائی که مجاز به استفاده از شبکه هستند ، باشد و در ضمن هیچکس نتواند در دسترسی به شبکه اختلال ایجاد کند .

زمانی که یکی از سرویسهای امنیتی پنج گانه فوق نقض شود می‌گوئیم به سیستم حمله شده است . معمولاً یک شبکه کامپیوتری درمعرض چهار نوع حمله قرار دارد:

- **حمله از نوع وقفه**[۴]: بدین معنا که حمله کننده باعث شود شبکه مختل شده و مبادله اطلاعات امکان پذیر نباشد.
- **حمله از نوع استراق سمع** [۵]: بدین معنا که حمله کننده به نحوی توانسته اطلاعات در حال تبادل روی شبکه را گوش داده و بهره برداری نماید .
- **حمله از نوع دستکاری داده‌ها**[۶]: یعنی حمله کننده توانسته به نحوی اطلاعاتی که روی شبکه مبادله می‌شود را تغییر دهد یعنی داده‌هایی که در مقصد دریافت میشود متفاوت با آنچیزی باشد که از مبداء آن ارسال شده است .
- **حمله از نوع افزودن اطلاعات**[۷]: یعنی حمله کننده اطلاعاتی را که در حال تبادل روی شبکه است تغییر نمی‌دهد بلکه اطلاعات دیگری را که می‌تواند مخرب یا بنیانگذار حملات بعدی باشد ، به اطلاعات اضافه می‌نماید (مثل ویروسها)

به حمله‌ای که هنگام شروع با بروز اختلال در شبکه علنی می‌شود و در کار ارسال یا دریافت مشکل ایجاد می‌کند "حمله فعال" می‌گویند. بر عکس حمله‌ای که شبکه را

---

۱ Integrity
۲ Access Control
۳ Availability
۴ Interruption
۵ Interception
۶ Modification
۷ Fabrication

با اختلال مواجه نمی‌کند و ظاهراً مشکلی در کار ارسال و دریافت بوجود نمی‌آورد "حمله غیر فعال"[۱] نامیده می‌شود و از خطرناکترین انواع حمله به شبکه به شمار می‌رود.

در ادامه این فصل دو راه کلی برای حراست و حفظ امنیت اطلاعات در یک شبکه کامپیوتری معرفی می‌شود:

- حراست و حفاظت داده‌ها و شبکه از طریق نظارت بر اطلاعات و دسترسیها به کمک سیستمی که "دیوار آتش"[۲] نامیده می‌شود.
- رمزگذاری اطلاعات به گونه ای که حتی اگر کسی آنها را دریافت کرد نتواند محتوای آنرا بفهمد و و از آن بهره برداری کند .

برای تمایز دو مورد فوق مثال عامیانه زیر بد نیست:

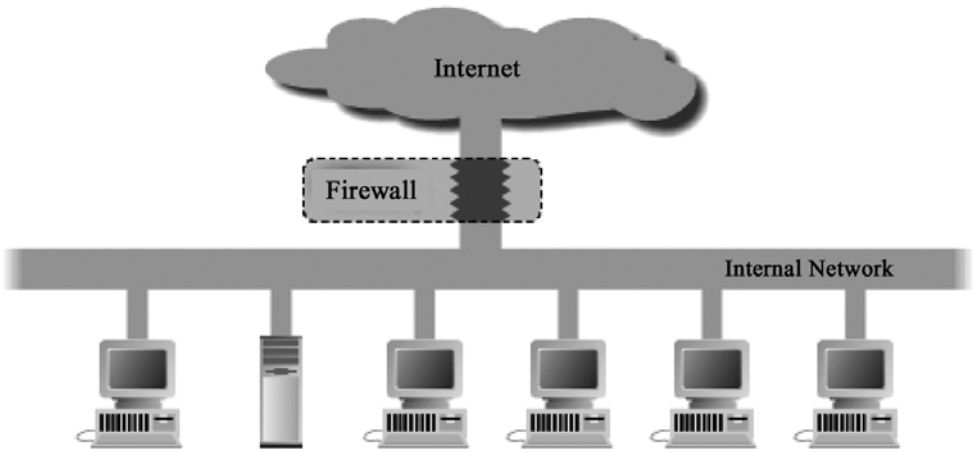
چون احتمال سرقت همیشه وجود دارد اولاً شما قفلهای مطمئن و دزدگیر برای منزل خود نصب می‌کنید و احتمالاً نگهبانی می‌گمارید تا ورود و خروج افراد را نظارت کند (کاری که دیوار آتش انجام می‌دهد) ثانیاً چون باز هم احتمال نفوذ می‌دهید لوازم قیمتی و وجوه نقد را در گوشه ای مخفی می‌کنید تا حتی در صورت ورود سارق موفق به پیدا کردن و بهره برداری از آن نشود . با تمام این کارها باز هم اطمینان صددرصد وجود ندارد چرا که هر کاری از یک انسان باهوش بر می‌آید.

۲) دیوار آتش

دیوار آتش سیستمی است که در بین کاربران یک شبکه محلی و شبکه بیرونی (مثلاً اینترنت) قرار می‌گیرد و ضمن نظارت بر دسترسیها ، در تمام سطوح ورود و خروج اطلاعات را تحت نظر دارد. مدلی ساده برای یک سیستم دیوار آتش درشکل (۱-۱۱) ارائه شده است . در این ساختار هر سازمان یا نهادی که بخواهد ورود و خروج اطلاعات شبکه را کنترل کند موظف است تمام ارتباطات مستقیم شبکه داخلی خود را با دنیای خارج قطع کرده و هرگونه ارتباط خارجی از طریق یک دروازه که در شکل (۱-۱۱) نشان داده شده ، انجام شود.

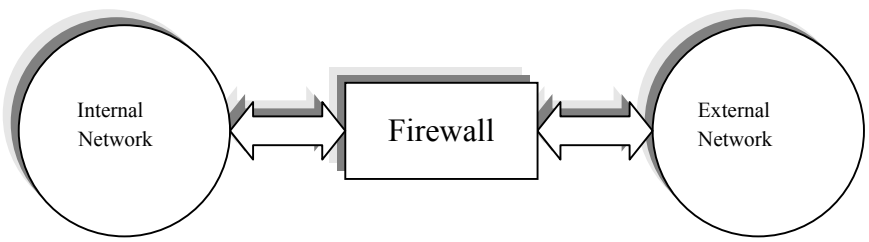
---

[۱] Passive
[۲] Firewall



شکل (۱-۱۱) نمودار کلی بکارگیری یک دیوار آتش

قبل از آنکه اجزای یک دیوار آتش را تحلیل کنیم باید عملکرد کلی و مشکلات استفاده از یک دیوار آتش را بررسی کنیم.
در شکل (۲-۱۱) مدل ساده تر شده یک دیوار آتش را در نظر بگیرید:



شکل (۲-۱۱) نمایی ساده از یک دیوار آتش

بسته‌های IP قبل از مسیریابی روی شبکه اینترنت ابتدا وارد دیوار آتش می‌شوند و منتظر می‌مانند تا طبق معیارهای حفاظتی و امنیتی پردازش شوند . پس از پردازش و تحلیل بسته سه حالت ممکن است اتفاق بیفتد:
الف) اجازه عبور بسته صادر شود. (Accept Mode)

ب) بسته حذف گردد. (Blocking Mode)
ج) بسته حذف شده و پاسخ مناسب به مبداء آن بسته داده شود. (Response Mode)
(به غیر از پیغام حذف بسته می‌توان عملیاتی نظیر اخطار ، رد گیری ، جلوگیری از ادامه استفاده از شبکه و توبیخ هم در نظر گرفت)
در حقیقت دیوار آتش محلی است برای ایست و بازرسی بسته‌های اطلاعاتی به گونه ای که بسته‌ها بر اساس تابعی از قواعد امنیتی و حفاظتی ، پردازش شده و برای آنها مجوز عبور یا عدم عبور صادر شود .
اگر P مجموعه ای از بسته‌های ورودی به سیستم دیوار آتش در نظر گرفته شود و S مجموعه ای متناهی از قواعد امنیتی باشد داریم:

$$X=F(P,S)$$

F تابع عملکرد دیوار آتش و X نتیجه تحلیل بسته (شامل سه حالت Accept,Blocking,Response) خواهد بود.

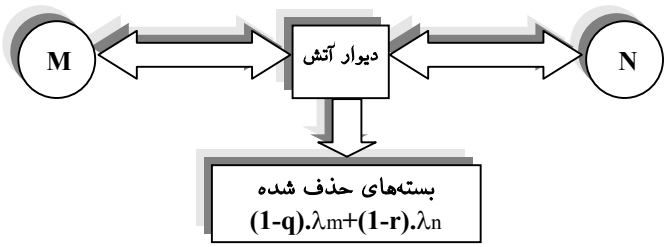
همانطوریکه همه جا عملیات ایست و بازرسی وقتگیر و اعصاب خرد کن است دیوار آتش هم بعنوان یک گلوگاه[۱] می‌تواند منجر به بالارفتن ترافیک ، تاخیر، ازدحام[۲] و نهایتاً بن بست در شبکه شود . (بن بست زمانی است که بسته‌ها آنقدر در حافظه دیوار آتش معطل می‌شوند تا طول عمرشان تمام شده و فرستنده اقدام به ارسال مجدّد آنها کرده و این کار بطور متناوب تکرار شود) به همین دلیل دیوار آتش نیاز به طراحی صحیح و دقیق دارد تا از حالت گلوگاهی خارج شود. (تاخیر در دیوار آتش مجموعاً اجتناب ناپذیر است فقط بایستی بگونه ای باشد که بحران ایجاد نکند.)
اگر از دیدگاه نظریه صف[۳] به یک دیوار آتش نگاه کنیم می‌توان تخمینی از تاخیر تحمیل شده به هر بسته را بدست آورد. معمولاً تابع توزیعِ تولید بسته‌ها را در شبکه های اطلاعاتی پوآسون در نظر می‌گیرند.
در شکل زیر فرض کنید λ_n متوسط انتقال بستهٔ IP در واحد زمان از شبکه N به دیوار آتش و λ_m متوسط انتقال بسته در واحد زمان از شبکه M باشد . q را احتمال عبور بسته P_M و r را احتمال عبور بسته P_N فرض کنید؛ طبق شکل (۳-۱۱) داریم:

متوسط بسته‌های حذف شده $= (1-q).\lambda_m+(1-r).\lambda_n$

---

۱ Bottleneck
۲ Congestion
۳ Queuing Theory

متوسط انتقال بسته از دیوار آتش به M = $r.\lambda_n$
متوسط انتقال بسته از دیوار اتش به N = $q.\lambda_m$



شکل (۳-۱۱) دیوار آتش از دیدگاه نظریه صف

طبق نظریه صف اگر دیوار آتش بخواهد از نقش گلوگاهی خود بکاهد بایستی بگونه ای طراحی شود که نسبت متوسط خروجی بسته‌ها از دیوار آتش (μ) به ورودی بسته‌ها (یعنی نسبت μ/λ) تا حدّ امکان زیاد باشد که این کار منوط به افزایش سرعت پردازش ، داشتن حافظه کافی برای ذخیره بسته‌های پردازش نشده و هر چه سریعتر کردن تابع تصمیم گیری می‌باشد. مشکل زمانی حاد می‌شود که دیوار آتش مجبور باشد برای تصمیم گیری و اجازه عبور تعدادی از بسته‌ها را نگه دارد تا تصمیم گیری بر اساس مجموعه ای از بسته‌ها انجام شود . این موضوع در ادامه آشکار خواهد شد.

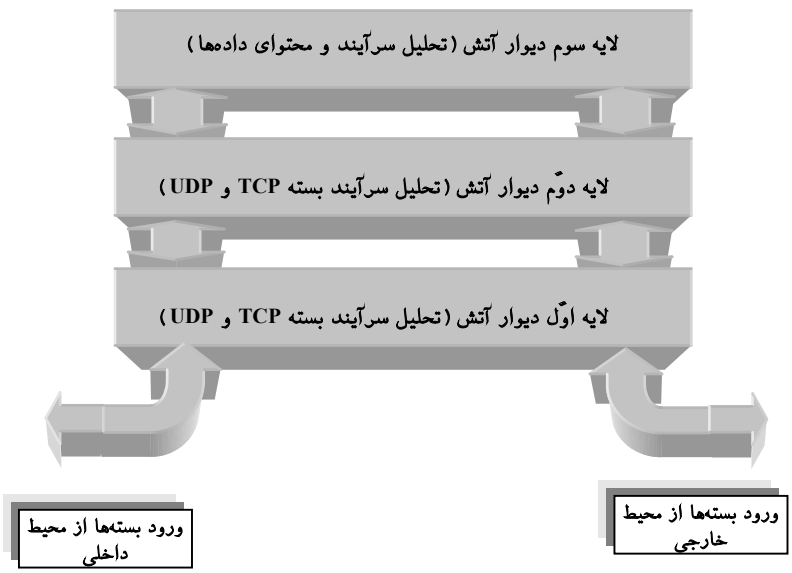
۳) مبانی طراحی دیوار آتش

از آنجائی که معماری شبکه بصورت لایه به لایه است ، در مدل TCP/IP برای انتقال یک واحد اطلاعات از لایه چهارم بر روی شبکه باید تمام لایه ها را بگذراند و و هر لایه برای انجام وظیفهٔ خود تعدادی فیلد مشخص به ابتدای بستهٔ اطلاعاتی اضافه کرده و آنرا تحویل لایه زیرین می‌دهد. قسمت اعظم کار یک دیوار آتش تحلیل فیلدهای اضافه شده در هر لایه و سرآیند هر بسته می‌باشد . در بسته ای که وارد دیوار آتش می‌شود به تعداد لایه ها (٤ لایه) سرآیند متفاوت وجود خواهد داشت . معمولاً سرآیند لایه اوّل (لایه فیزیکی یا Network Interface در شبکه اینترنت) اهمیت چندانی ندارد چرا که محتوای این فیلدها فقط روی کانال فیزیکی از شبکه محلی معنا دارند و در گذر از هر شبکه یا مسیر یاب این فیلدها عوض

خواهند شد. بیشترین اهمیت در سرآیندی است که در لایه های دوم ، سوم و چهارم به یک واحد از اطلاعات اضافه خواهد شد:

- در لایه شبکه دیوار آتش فیلدهای بسته IP را پردازش و تحلیل می‌کند.
- در لایه انتقال دیوار آتش فیلدهای بسته‌های TCP یا UDP را پردازش و تحلیل می‌کند .
- در لایه کاربرد دیوار آتش فیلدهای سرآیند و همچنین محتوای خود داده‌ها را بررسی می‌کند . (مثلا سرآیند و محتوای یک نامه الکترونیکی یا یک صفحه وب می‌تواند مورد بررسی قرار گیرد.)

با توجه به لایه لایه بودن معماری شبکه لاجرم یک دیوار آتش نیز لایه به لایه خواهد بود به شکل (٤-١١) دقت کنید:



شکل (٤-١١) لایه بندی ساختار یک دیوار آتش

اگر یک بسته در یکی از لایه های دیواره آتش شرایط عبور را احراز نکند همانجا حذف شده و به لایه های بالاتر ارجاع داده نمی‌شود بلکه این امکان وجود دارد که

آن بسته جهت پیگیریهای امنیتی نظیر ثبت عمل و ردگیری به سیستمی جانبی تحویل داده شود.

سیاست امنیتی یک شبکه مجموعه ای متناهی از قواعد امنیتی است که بنابر ماهیتشان در یکی از سه لایه دیوار آتش تعریف می‌شوند ، بعنوان مثال:

- قواعد تعیین آدرسهای ممنوع در اولین لایه از دیوار آتش
- قواعد بستن برخی از سرویسها مثل Telnet یا FTP در لایه دوم
- قواعد تحلیل سرآیند متن یک نامه الکترونیکی یا صفحه وب در لایه سوم

۱-۳) لایه اول دیوار آتش

لایه اول در دیوار آتش بر اساس تحلیل بسته IP و فیلدهای سرآیند این بسته کار می‌کند و در این بسته فیلدهای زیر قابل نظارت و بررسی هستند :

- **آدرس مبداء**: برخی از ماشینهای داخل یا خارج شبکه با آدرس IP خاص "حق ارسال" بسته نداشته باشند و بسته‌های آنها به محض ورود به دیوار آتش حذف شود.
- **آدرس مقصد**: برخی از ماشینهای داخل یا خارج شبکه با آدرس IP خاص "حق دریافت" بسته نداشته باشند و بسته‌های آنها به محض ورود به دیوار آتش حذف شود .
- (آدرسهای IP غیر مجاز توسط مسئول دیوار آتش تعریف می‌شود)
- **شماره شناسایی یک دیتاگرام**[1] : بسته‌هائی که متعلق به یک دیتاگرام خاص هستند حذف شوند.
- **شماره پروتکل**: بسته‌هایی که متعلق به پروتکل خاصی در لایه بالاتر هستند می‌توانند حذف شود. یعنی بررسی اینکه بسته متعلق به چه پروتکلی در لایه بالاتر است و آیا برای تحویل به آن پروتکل مجاز است یا نه .
- **زمان حیات بسته**: بسته‌هائی که بیش از تعداد مشخصی مسیریاب را طی کرده اند مشکوک هستند و باید حذف شوند.

[1] Identifier & Fragment offset

- بقیه فیلدها بنابر صلاحدید و قواعد امنیتی مسئول دیوار آتش قابل بررسی هستند.

مهمترین خصوصیت لایهٔ اول از دیوار آتش آنست که در این لایه بسته‌ها بطور مجزا و مستقل از هم بررسی می‌شوند و هیچ نیازی به نگه داشتن بسته‌های قبلی یا بعدی یک بسته نیست. بهمین دلیل ساده ترین و سریعترین تصمیم گیری در این لایه انجام می‌شود. امروزه برخی از مسیریابها با امکان لایه اوّل دیوار آتش به بازار عرضه می‌شوند یعنی به غیر از مسیریابی ، وظیفهٔ لایه اوّل یکَ دیوار آتش را هم انجام می‌دهند که به آنها "مسیریابهای فیلتر کنندهٔ بسته"[1] گفته می‌شود. بنابراین مسیریاب قبل از اقدام به مسیریابی ، بر اساس جدولی بسته‌های IP را غربال می‌کند و تنظیم این جدول بر اساس نظر مسئول شبکه و برخی از قواعد امنیتی انجام می‌گیرد.

با توجه به سریع بودن این لایه هر چه درصد قواعد امنیتی در این لایه دقیقتر و سختگیرانه تر باشد حجم پردازش در لایه های بالاتر کمتر و در عین حال احتمال نفوذ پایینتر خواهد بود ولی در مجموع بخاطر تنوع میلیاردی آدرسهای IP نفوذ از این لایه با آدرسهای جعلی یا قرضی امکان پذیر خواهد بود و این ضعف در لایه های بالاتر بایستی جبران شود.

۲-۳) لایه دوم دیوار آتش

در این لایه از فیلدهای سرآیند لایه انتقال برای تحلیل بسته استفاده می‌شود. عمومی‌ترین فیلدهای این بسته عبارتند از :

- **شماره پورت پروسه مبداء و شماره پورت پروسه مقصد**: با توجه به آنکه پورتهای استاندارد شناخته شده هستند ممکن است مسئول یک دیوار آتش بخواهد سرویس ftp (انتقال فایل) فقط در محیط شبکه محلی امکان پذیر باشد و برای تمام ماشینهای خارجی این سرویس وجود نداشته باشد بنابراین دیوار آتش می‌تواند بسته‌های TCP با شماره پورت ۲۰ و ۲۱ (مربوط به ftp) که قصد ورود یا خروج از شبکه را دارند ،حذف کند. یکی دیگر از سرویسهای خطرناک که ممکن است مورد

[1] Pocket Filtering Router

سوء استفاده قرار گیرد Telnet است که می‌توان براحتی پورت ۲۳ را مسدود کرد یعنی بسته‌هائی که شماره پورت مقصدشان ۲۳ است حذف شوند.

- **فیلد شماره ترتیب[1] و فیلد Acknowledgment:** این دو فیلد نیز بنا بر قواعد تعریف شده توسط مسئول شبکه قابل استفاده هستند.

از مهمترین خصوصیات این لایه آنست که تمام تقاضاهای برقراری ارتباط TCP بایستی از این لایه بگذرد و چون در ارتباط TCP ، تا مراحل **"دست تکانی سه گانه اش"** به اتمام نرسد انتقال داده امکان پذیر نیست لذا قبل از هر گونه مبادله داده دیوار آتش می‌تواند مانع برقراری هر ارتباط غیر مجاز شود . یعنی دیوار آتش می‌تواند تقاضاهای برقراری ارتباط TCP را قبل از ارائه به ماشین مقصد بررسی نماید و در صورت غیر قابل اعتماد بودن ، مانع از برقراری ارتباط شود. دیوار آتش در این لایه نیاز به جدولی از شماره پورتهای غیر مجاز دارد.

۳-۳) لایه سوم دیوار آتش

در این لایه حفاظت بر اساس نوع سرویس و برنامه کاربردی انجام می‌شود. یعنی با در نظر گرفتن پروتکل در لایه چهارم به تحلیل داده‌ها می‌پردازد. تعداد سرآیند ها در این لایه بسته به نوع سرویس بسیار متنوع و فراوان است. بنابراین در لایه سومِ دیوار آتش برای هر سرویس مجزا (مثل سرویس پست الکترونیکی، سرویس ftp، سرویس وب و ...) باید یک سلسله پردازش و قواعد امنیتی مجزا تعریف شود و به همین دلیل حجم و پیچیدگی پردازش در لایه سوم زیاد است . توصیه موکّد آنست که تمام سرویسهای غیرضروری و شماره پورتهایی که مورد استفاده نیستند در لایه دوم مسدود شوند تا کار در لایه سوم کمتر باشد.

بعنوان مثال فرض کنید موسسه ای نظامی سرویس پست الکترونیکی خود را دائر کرده ولی نگران فاش شدن برخی اطلاعات محرمانه است . در این حالت دیوار آتش در لایه سوّم می‌تواند کمک کند تا برخی از آدرسهای پست الکترونیکی مسدود شود ، در عین حال می‌تواند در متون نامه های رمز نشده دنبال برخی از کلمات کلیدی

---

[1] Sequence Number

حسّاس بگردد و متون رمزگـذاری شـده را در صورتـی کـه موفـق بـه رمـزگشـائی آن نشـود حذف نمـاید.

بعنوان مثالی دیگـر یـک مرکـز فرهنگـی علاقمنـد اسـت قبـل از تحویـل صفحـه وب بـه یـک کـاربر ، درون آنـرا از لحـاظ وجـود برخـی از کلمـات کلیـدی بررســـی کنـــد و اگـــر کلماتی که با معیارهـــای فرهنگـی مطـابقت نـدارد درون متـن صفحـه یـافت شـد آن صفحـه را حذف نمـاید.

۴) اجزای جانبی یک دیوار آتش

دیوار آتـش یـک سیسـتم امنیتـی اسـت کـه سیاسـتهای مسـئول شـبکه را پیـاده و اعمـال می‌کنـد. بنـابراین دیـوار آتـش بایسـتی از طریـق یـک ورودی سـهل و راحـت قواعـــد را از مسـئول شـبکه دریـافت نمـاید و همـواره فعالیتـهای موجـــود روی شـــبکه را بـــه مســـئول شـبکه گـزارش بدهـد . بـهمین دلیـل معمـولاً یـک سیسـتم دیـوار آتـش دارای اجـزاء ذیـل است:

۴-۱) واسط محاوره ای و ساده ورودی/خروجی

برای تبادل اطلاعات و ســهولت در تنظیـم قواعـد امنیتـی و ارائـه گـزارش، نیـاز بـه یـک واسط کـاربر[۱] کـه سـاده و در عیـن حـال کـارآمد باشـد وجـود دارد . معمـولاً واسـط کـاربر مستقل از سیستم دیوار آتــش اسـت تـا حجـم پـردازش اضـافی روی سیسـتم تحمیـل نکنـد یعنی معمولاً دیوار آتـش دارای دسـتگاهی بـه عنـوان صفحـه نمـایش نیسـت بلکـه از طریـق وصل یک ابزار جانبی مثــل یـک ترمینـال سـاده یـا یـک کـامپیوتر شـخصی معمولـی فرمـان می‌گیرد یا گــزارش می‌دهـد.

۴-۲) سیستم ثبت[۲]

برای بـالاتر بـردن ضریـب امنیـت و اطمینـان در شـبکه ، دیـوار آتـش بـاید بتوانـد حتـی در مواقعیکـه اجـازه خـروج یـا ورود یـک بسـته بـه شـبکه صـــادر می‌شـــود اطلاعـــاتی در

---

[۱] User Interface
[۲] Logger

مورد آن بسته ذخیره کند تا در صورت هر گونه حمله یا نفوذ بتوان مسئله را پیگیری کرد. در یک دیوار آتش عملی که ثبت کننده می‌تواند انجام بدهد آنست که مبداء و مقصد بسته‌های خروجی و ورودی ، شماره پورتهای مبداء و مقصد ، سرآیند یا حتی محتوای پیام در لایه کاربرد را (برای تمام مبادلات خارج از شبکهٔ محلی) ذخیره کند و لحظه به لحظه مبادلهٔ اطلاعات تمام کاربران و حتی مسئول شبکه را در فایلی درج نماید . این اطلاعات می‌تواند بعنوان سندی بر علیه فرد خاطی استفاده شود یا به یافتن کسی که در خارج از شبکه مشغول اخلال گری است کمک نماید .

۳-۴) سیستم هشدار دهنده

در صورت بروز هر گونه مشکل یا انتقالی مشکوک ، دیوار آتش می‌تواند مسئول شبکه را مطلع نماید و در در صورت لزوم کسب تکلیف کند . اعمال مشکوک در هر سه لایه تعریف میشود : مثل تقاضای ارتباط با آدرسهای IP مشکوک ، آدرسهای پورت مشکوک ، اطلاعات مشکوک در لایه کاربرد (صفحات وب یا نامه های مشکوک).

ارتباط مشکوک را می‌توان مصداق ارتباطاتی دانست که بی هدف یا مکرر در طی روز برقرار می‌شود یا آنکه اطلاعات ارسالی مفهوم یا مضمون خاصّی نداشته باشد یا آنکه رمز شده باشد. در این حالت سیستم دیوار آتش ضمن کسب تکلیف می‌تواند یک آدرس مشکوک را به عنوان آدرس غیر مجاز مسدود کند.

۵) راه حل نهائی

با تمام نظارتی که بر تردد بسته‌های اطلاعاتی حین ورود یا خروج از شبکه می‌شود باز هم می‌توان زیرکانه از مرز دیوار آتش عبور کرد و بهترین حفاظت برای جلوگیری از فاش شدن اطلاعات محرمانه به دنیای خارج، نابود کردن خط ارتباطی شبکه به دنیای خارج است! چرا که می‌توان اطلاعات سری را رمز و فشرده کرد و آنرا بعنوان بیت آخر از نقاط تصویر یک گل رز بعنوان کارت پستال تبریک سال نو ارسال نمود. در حقیقت سیستم دیوار آتش فقط یک ابزار محدود کننده است و اطمینان صد در صد ندارد.

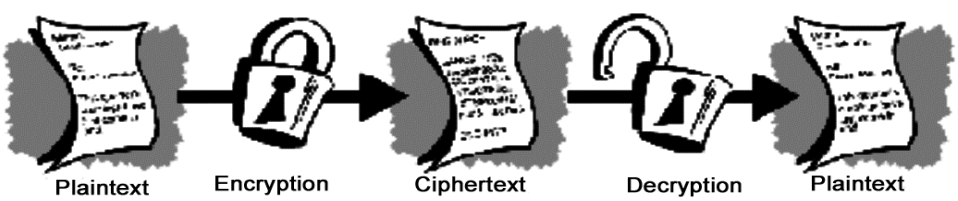
۶) رمزنگاری

زمانیکه ژولیوس سزار پیامهائی را برای فرمانده ارتش خود در جنگ می‌فرستاد از بیم کشته شدن یا خیانت پیک ، در تمام متنِ نامهٔ خود هرحرف را با حرفی که سه تا بعد از آن قرار گرفته بود عوض می‌کرد (مثلاً بجای A حرف D و بجای B حرف E را قرار می‌داد) تا متن حالت معنی دار خود را از دست بدهد. تنها کسی می‌توانست از مفهوم متن چیزی بفهمد که به رمز آن (یعنی Shift by 3) آگاهی داشت.

داده‌هائی که به راحتی قابل فهم هستند و هیچ نکته و ابهام خاصی در درک آنها وجود ندارد، متن ساده یا متن آشکار نامیده می‌شوند. روشی که باعث می‌شود متن ساده حالت قابل درک و فهم خود را از دست بدهد "رمزنگاری"[1] نامیده می‌شود. معمولاً در دنیای شبکه های کامپیوتری رمزنگاری سلسله ای از عملیات ریاضی است که مجموعه ای از اطلاعات خالص و قابل فهم را به مجموعه ای از اطلاعات غیرقابل فهم ، بی معنا و بلا استفاده تبدیل می‌کند. به گونه ای که فقط گیرنده اصلی آن قادر باشد آنرا از حالت رمز خارج و از آن بهره برداری نماید. (یعنی کلید رمز را در اختیار داشته باشد)

علم رمزنگاری [2] با اصول ریاضی به رمز درآوردن اطلاعات و خارج کردن آنها از حالت رمز سر و کار دارد. در مقابلِ علم رمزنگاری ، علم تحلیل رمز[3] قرار دارد که روشهای تجزیه و شکستن رمز اطلاعات (بدون نیاز به کلید) و کشف کلید رمز را مورد بحث قرار می‌دهد.

به شکل (۵-۱۱) دقت کنید. در این شکل سیمای کلی یک سیستم رمزنگاری و رمزگشائی به تصویر کشیده شده است .



شکل (۵-۱۱) سیمای کلّی سیستم رمزنگاری و رمزگشائی

---

[1] Encryptioin
[2] Cryptography
[3] Cryptoanalysis

الگوریتم یا روشی که بر اساس آن متن رمز می‌شود باید بگونه ای قابل برگشت (وارون پذیر) باشد تا بتوان به متن اصلی دست پیدا کرد.

در ادامه چند روش رمزنگاری را معرفی می‌نمائیم:

۱-۶) روشهای جانشینی[1]

روش جانشینی قدیمی‌ترین نوع رمزنگاری است که اولین بار سزار آن را بکار برده است. در این روش هر حرف از جدول حروف الفبا به حرفی دیگر تبدیل می‌شود . بعنوان مثال در رمز سزار هر حرف به حرف سوم بعد از خودش تبدیل می‌شد که با این روش کلمه "حمله"[2] بصورت زیر در می‌آمد:

Attack متن اصلی

Dwwdfn متن رمز شده

این روش بعداً بهبود داده شد و بجای آنکه تمام حروف بطور منظم و با قاعده به یکدیگر تبدیل شوند جدول حروف الفبا طبق یک قاعده نامشخص که "جدول رمز" نامیده می‌شد به هم تبدیل می‌شدند. بعنوان مثال اگر نامه یا متن تماماً حروف کوچک در نظر بگیریم جدول رمز می‌تواند بصورت زیر باشد:

a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	Y	z
Q	W	E	R	T	Y	U	I	O	P	A	S	D	F	G	H	J	K	L	Z	X	C	V	B	N	M

طبق این جدول که گیرنده پیام بایستی از آن آگاهی داشته باشد کلمه attack به کلمه QZZQEA تبدیل می‌شود. شاید یک مبتدی احساس کند که این روش امروزه مفید خواهد بود چرا که جدول رمز دارای !26 (معادل $٤*١٠^{٢٦}$) حالت متفاوت خواهد بود و امتحان تمام این حالات مختلف برای یافتن جدول رمز کاری مشکل است ، در حالی که چنین نیست و این نوع رمزنگاری برای متون معمولی در کسری از ثانیه و بدون کلید رمز شکسته خواهد شد! نقطه ضعف این روش در مشخصات

[1] Substitution
[2] Attack

آماری هر حرف در یک زبان می‌باشد. بعنوان مثال درزبان انگلیسی حرف e در متن بیش از همه حروف تکرار می‌شود . ترتیب فراوانیِ نسبی برای شش حرف پرتکرار در متون انگلیسی بصورت زیر است:

e > t > o > a > n > i

اولین اقدام در رمزشکنی (رمزشکنی همان رمزگشائی است بدون در اختیار داشتن کلید یا جدول رمز) آنست که متن رمز شده تحلیل آماری شود و هر کاراکتری که بیش از همه در آن تکرار شده باشد معادل e ، حرف پرتکرار بعدی معادل t قرار بگیرد و روند به همین ترتیب ادامه یابد. البته ممکن است برخی از حروف اشتباه سنجیده شوند ولی می‌توانند در مراحل بعدی اصلاح شود.

دومین نکته آنست که در زبانی مثل انگلیسی حروف کنار هم وابستگی آماری بهم دارند مثلاً در ۹۹/۹ درصد مواقع در سمت راست حرف q حرف u قرار گرفته (qu) یا در کنار حرف t معمولاً (البته با احتمال پائین تر) h قرار گرفته است. یعنی بمحض کشف حرف q رمز u هم کشف می‌شود و اگر t کشف شد کشف h ساده تر خواهد شد. ترتیب فراوانیِ نسبی برای پنج "دوحرفی"[1] پرتکرار در متون انگلیسی بصورت زیر است:

th > in > er > re > an

سومین نکته نیز به سه حرفی ها بر می‌گردد. مثلاً در زبان انگلیسی سه حرفی های ion, and, the, ing به کرّات استفاده می‌شوند و می‌توانند ملاک رمزشکنی قرار بگیرند.

چهارمین نکته برای رمزشکنی مراجعه به فرهنگ لغات یک زبان است که بر اساس پیدا شدن چند حرف از یک کلمه رمز بقیه حروف آن نیز آشکار میشود.

به دلائل فوق روش رمزگذاری جانشینی کارآئی مناسبی ندارد و براحتی رمز آن بدست خواهد آمد.

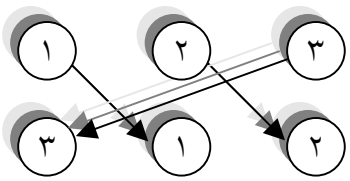
۲-۶) رمزنگاری جایگشتی[2]

در رمزگذاری جانشینی محل قرار گرفتن و ترتیب حروف کلمات در یک متن بهم نمی‌خورد بلکه طبق یک جدول رمز جایگزین می‌شد. در روش رمزنگاری جایگشتی آرایش و ترتیب کلمات به هم می‌خورد. بعنوان یک مثالِ بسیار ساده فرض

---

[1] Digram
[2] Permutation

کنید تمام حروف یک متن اصلی را سه تا سه تا جدا کرده و طبق قاعده زیر ترتیب آن را بهم بریزیم:



کلمه اصلی: **the**
کلمه رمز: **eth**

برای رمزگشائی، گیرندهٔ پیام باید کلید جایگشت را که در مثال ما (۲و۱و۳) است بداند.

معمولا برای راحتی در به خاطر سپردن کلید رمز ، یک کلید متنی انتخاب میشود و سپس جایگشت بر اساس ترتیب حروف کلمهٔ رمز انجام می‌شود. برای وضوح این روش به مثال زیر دقت کنید:

متن اصلی: please-trasnfer-one-million-dollors-to-my-swiss-bank-account-six-two-two

کلمه رمز: **MEGABUCK**

تمام کلمات متن اصلی بصورت دسته های هشت تائی جدا شده و تماماً زیر هم نوشته می‌شود: (علامتِ – را فاصلهٔ خالی در نظر بگیرید)

کلمه رمز	*M*	*E*	*G*	*A*	*B*	*U*	*C*	*K*
ترتیب حروف کلمه رمز	۷	٤	٥	۱	۲	۸	۳	٦
۱	p	l	e	a	s	e	-	t
۲	r	a	n	s	f	e	r	-
۳	o	n	e	-	m	i	l	l
٤	i	o	n	-	d	o	l	l
٥	a	r	s	-	t	o	-	m
٦	y	-	s	w	i	s	s	-
۷	b	a	n	k	-	a	c	c
۸	o	u	n	t	-	s	i	x
۹	t	w	o	-	t	w	o	-

حال بر اساس ترتیب الفبایی هر حرف در کلمه رمز، ستونها بصورت پشت سر هم نوشته می‌شوند . یعنی ابتدا ستون مربوط به حرف A ، سپس B ، E و ...

پس رمز بصورت زیر در می‌آید:

"as---wkt-sfmdti---rll-sciolanor-auwenenssnnot-llm-cx-proiayboteeioosasw"

بنابراین برای بازیابیِ اصل پیام درمقصد ، گیرنده باید کلید رمز (یا حداقل ترتیب جایگشت) را بداند.

این روش رمز هم قابل شکستن است چرا که اگر چه ترتیب حروف بهم ریخته است ولی در متن رمز شده تمام حروف هر یک از کلمات وجود دارند. بعنوان مثال تک تک حروف dollars یا swiss bank را می‌توان در متن رمز شده پیدا کرد لذا با بررسی تمام حالات ممکن که کلمه dollars را به صورت پراکنده در متن در می‌آورد می‌توان رمز را بدست آورد. البته حجم پردازش مورد نیاز بیشتر خواهد بود ولی بهر حال این نوع رمزگذاری براحتی قابل شکستن می‌باشد و در دنیای امروز چندان قابل اعتماد نیست.

۷) استانداردهای نوین رمزگذاری

در اوائل دهه هفتاد دولت فدرال آمریکا و شرکت آی.بی.ام (IBM) مشترکاً روشی را برای رمزنگاری داده‌ها ایجاد کردند که بعنوان استانداردی برای نگهداری اسناد محرمانه دولتی مورد استفاده قرار گرفت. این استاندارد که DES[1] نام گرفت امروزه محبوبیت خود را از دست داده است .

الگوریتم روش رمزنگاری DES در شکل (٦-١١) نشان داده شده است که در ادامه کلیّت آنرا توضیح می‌دهیم:

ورودی رمزنگار یک رشته ٦٤ بیتی است، بنابراین متنی که باید رمز شود بایستی در گروههای هشت کاراکتری دسته بندی شوند . اولین عملی که بر روی رشته ورودی ٦٤ بیتی انجام می‌شود جابجا کردن محل بیتهای رشتهٔ ٦٤ بیتی طبق جدول صفحهٔ بعد است . به این عمل جایگشت مقدماتی[2] گفته می‌شود :

[1] Data Encryption Standard
[2] Initial permutation

جدول جایگشت مقدماتی							
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

در جدول بالا پس از عمل جایگشت, بیت اول به موقعیت بیت پنجاه وهشتم و بیت دوم به بیت پنجاهم از رشته جدید منتقل شده و بهمین ترتیب ادامه می‌یابد تا رشته ٦٤ بیتی جدید بدست آید.

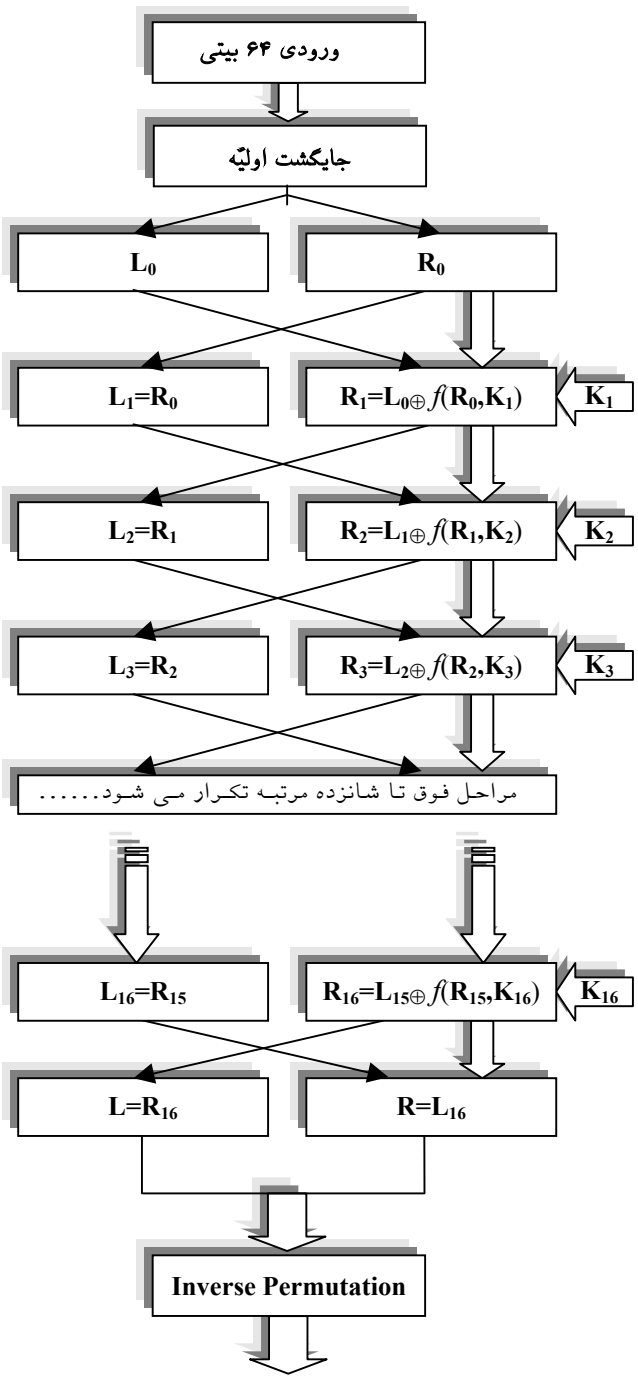
در مرحله بعدی رشته ٦٤ بیتی جدید از وسط به دو قسمت ٣٢ بیتی چپ و راست تقسیم می‌شود. ٣٢ بیت سمت چپ را L_0 و ٣٢ بیت سمت راست را R_0 می‌نامیم. (به شکل (٦-١١) نگاه کنید)

سپس در ١٦ مرحلهٔ پیاپی اعمال زیر انجام می‌شود:

در هر مرحله ٣٢ بیت سمت راست مستقیماً به سمت چپ منتقل شده و ٣٢ بیت سمت چپ طبق رابطه زیر به یک رشته بیت جدید تبدیل و به سمت راست منتقل خواهد شد.

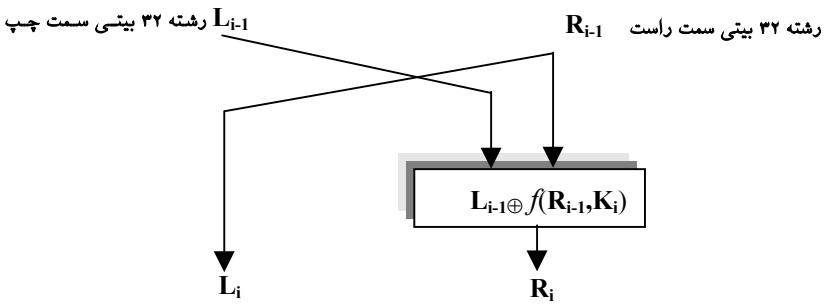
$$\mathbf{L_{i-1} \oplus f(R_{i-1}, K_i)}$$

L_{i-1} رشته سی و دو بیتی سمت چپ از مرحله قبل می‌باشد . علامت $\oplus$ بمعنای XOR و f تابع خاصی است که آنرا به صورت مجزا توضیح خواهیم داد $\mathbf{R_{i-1}}$ رشته سی و دو بیتی سمت راست از مرحله قبل و **Ki** کلید رمز در هر مرحله است. (پس مجموعاً ١٦ کلید مختلف وجود دارد.)



شکل (٦-١١) الگوریتم روش رمزنگاری DES

نمودار زیر یک مرحلـه از ١٦ مرحلـه عملیـات را نشـان می‌دهـد:



ایـن عملیـات ١٦ مرحلـه اجـرا می‌شـود و پـس از مرحلـه آخـر جـای L_i و R_i عـوض خواهد شـد.

حال عکـس عمـل جایگشـتی کـه در ابتـدا انجـام شـده بـود صـورت مـی‌گـیرد تـا بیتـها سرجایشان برگردند این کار طبـق جـدول زیـر انجـام می‌شـود:

جدول جایگشت معکوس							
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

پس از ایـن عمـل ، ٦٤ بیـت جدیـد معـادل هشـت کـاراکتر رمـز شـده خواهـد بـود کـه می‌توان آنها را بجای متـن اصلـی ارسـال کـرد.

حال بایستی جزئیات تابع f را کـه اصـل عمـل رمزنگـاری اسـت تعییـن کنیـم:

در تابع f کـه بـه صـورت یـک بلـوک پیـاده سـازی می‌شـود ابـتدا رشـته ۳۲ بیتـی Ri که از مرحله قبل بدست آمده بر طبـق جـدول زیـر بـه یـک رشـته ٤٨ بیتـی تبدیـل می‌شـود. بنابراین بعضی از بیتـها در رشـته جدیـد تکـراری هسـتند .

جدول بسط ۳۲ به ٤٨					
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

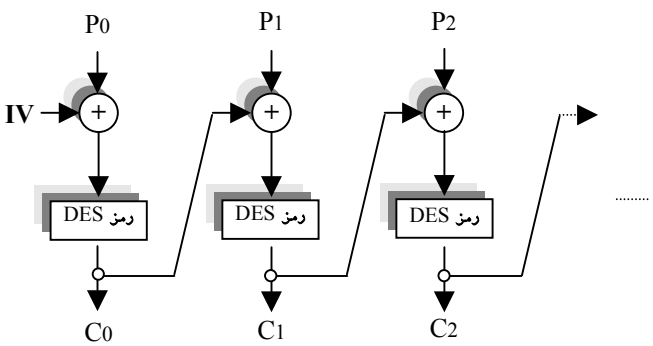
پـس از تبدیـل R_i بـه رشـته ٤٨ بیتـی عمـل XOR روی آن بـا کلیـد ٤٨ بیتـیK_i انجـام می‌شود. نتیجهٔ عمل یـک رشـتهٔ ٤٨ بیتـی اسـت و بایسـتی بـه ۳۲ بیتـی تبدیـل شـود . بـرای اینکار ٤٨ بیت به هشت مجموعـه ٦ بیتـی تبدیـل شـده و هـر کـدام از شـش بیتـی هـا طبـق جداولی به یک چـهار بیتـی جدیـد نگاشـته می‌شـود (در مجمـوع ۸ جـدول) . بـرای کـامل شـدن عمـل تـابع f رشـته ۳۲ بیتـی جدیـد طبـق جـدول زیـر جایگشـت مجـددی خواهـد داشـت.

جایگشت ۳۲ بیتی در تابع f			
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

در سیستم DES فقط یک کلید ۵۶ بیتی وجود دارد که تمام ۱۶ کلید مورد نیاز در هر مرحله با جایگشتهای متفاوت از همان کلید ۵۶ بیتی استخراج خواهد شد . بنابراین کاربر برای رمزگشایی فقط باید یک کلید در اختیار داشته باشد و آنهم همان کلیدی است که برای رمزنگاری به کار رفته است .

برای رمزگشائی از سیستم DES دقیقاً مراحل قبلی تکرار می‌شود با این تفاوت که کلید K_1 برای رمزگشائی ، کلید K_{16} در مرحله رمزنگاری خواهد بود، کلید K_2 همان K_{15} است و به همین ترتیب . در حقیقت برای رمزگشائی کافی است ۱۶ کلید بصورت معکوس به سیستم اعمال شوند.

نکته دیگری که در مورد سیستم DES قابل توجه است آنست که چون رشتهٔ رمز شده بصورت هشت کاراکتری رمز می‌شود ، تکرار بلوکهای رمز می‌تواند به رمزشکنها برای حمله به سیستم DES کمک نماید . بهمین دلیل در سیستم DES قبل از آنکه یک بلوک رمز شود ابتدا با بلوک رمز شدهٔ قبلی خود XOR می‌شود و سپس این ۸ کاراکتر مجدداً رمز خواهد شد. به شکل (۷-۱۱) دقت کنید:



شکل (۷-۱۱) عملیات بین بلوکهای داده در سیستم رمزنگاری DES

بلوک اول با یک رشته ٦٤ بیتی اولیه بنام IV[1] (بردار اولیّه) XOR می‌شود . نتیجه این بلوک کد رمز ٦٤ بیتی است . همین کد رمز برای رمز کردن بلوک بعدی بکار می‌آید ، بدینصورت که بلوک رمز نشدهٔ Pi با بلوک رمز شدهٔ قبلی Ci-1 ابتدا XOR شده و متن جدید مجدّداً رمز خواهد شد.

[1] Initialization Vector

این الگوی رمزنگاری بعنوان استانداردی برای اسناد حسّاس فدرال آمریکا پذیرفته شد تا آنکه در سال ۱۹۷۷ یکی از محققین دانشگاه استانفورد با هزینه ای معادل ۲۰ میلیون دلار ماشینی طراحی کرد که در عرض ۲۴ ساعت می‌توانست رمز DES را بشکند. بعد از آن ایده های جدیدی برای رمزنگاری مطرح شد که DES را در سیستمهای عملی کنار زد.

نکته دیگر آنست که چون کلید رمزنگاری و رمزگشائی هر دو یکی است لذا باید از کلید شدیداً حفاظت شود . در مدلهای جدید کلید رمزنگاری را همه می‌دانند ولی کلید رمزگشائی سرّی است .

۸) رمزگذاری کلید عمومی[1]

در هر یک از الگوهای رمزنگاری که مورد بحث قرار گرفتند لازم است که فرستنده پیام و گیرنده پیام کلید رمز را بدانند . وقتی فرستندهٔ پیام از کلیدی برای رمزنگاری استفاده می‌کند و گیرندگان هم از همان کلید برای رمزگشایی بهره می‌برند ، افشا شدن کلید رمز توسط یکی از گیرندگان پیام ، امنیت را به خطر می‌اندازد. در الگوهای جدیدِ رمزگذاری، برای حل مشکل از دو کلید متفاوت استفاده می‌شود . یک کلید برای رمز کردن پیام و کلید دیگر برای رمزگشائی آن . با کلیدِ مخصوص رمزنگاری نمی‌توان رمزگشائی پیام را انجام داد. بنابراین رمزکنندهٔ پیام خودش کلیدی دارد که حتی معتمدین و گیرندگان پیام هم آنرا لازم ندارند چرا که فقط برای رمزنگاری بکار می‌آید و افشا شدن آن هم لطمه ای به کسی نمی‌زند چرا که با آن کلید نمی‌توان متون رمز شده را برگرداند و پیدا کردن کلید رمزگشائی از روی کلید رمزنگاری کار ساده ای نیست. (هنوز امکان پذیر نشده است)

در سال ۱۹۷۸ سه نفر بنامهای ری وست ، شامیر و اَدلمن روشی را برای پیاده سازی "رمزنگاری کلید عمومی" با یک جفت کلید ابداع کردند . این روش که چگونگی آن در زیر تشریح شده است بنام روش RSA (مخفف اسامی آنها) مشهور است و بطرز فزاینده ای از آن استفاده می‌شود:

[1] Public Key Cryptography

روش کار فوق العاده ساده است: دو عدد صحیح (e,n) برای رمزگذاری انتخاب می‌شوند؛ متنی که باید رمز شود به بلوکهایی تقسیم می‌شود . مثلاً کل متن پیام به K تا بلوک تقسیم شده و هر بلوک به نحوی به یک عدد صحیح تبدیل می‌شود . مثلاً کدهای اَسکی هر حرف پشت سر هم قرار می‌گیرند. برای آنکه همین ابتدا بحث را پیچیده نکنیم فرض کنید بخواهیم رشته "M="IDESOFMARCH را رمز کنیم . برای سادگی این رشته را به بلوکهای ۲ کاراکتری تقسیم کرده و سپس هر بلوک را به یک عدد صحیح تبدیل می‌نماییم:

رشته اصلی بلوکهای ۲ کاراکتری →	ID	ES	OF	MA	RC	HX
تبدیل رشته به شش بلوک →	M_1	M_2	M_3	M_4	M_5	M_6
تبدیل بلوک به عدد صحیح →	0803	0418	1405	1200	1702	0723
بلوکهای جدید عددی →	P_1	P_2	P_3	P_4	P_5	P_6

روش تبدیل در مثال بالا این بوده که برای A عدد ۰۰ ، B عدد ۰۱ ، ... و Z عدد 25 در نظر گرفته شده و در هر بلوک عدد متناظر با هر کاراکتر پشت سر هم قرار می‌گیرد تا کد بلوک را بسازد . شما می‌توانستید کد اَسکی آن یا هر قاعدهٔ دیگری را به کار ببرید.

در مرحله بعدی جفت عدد صحیح (17,2773) معادل (e,n) برای رمزگذاری بلوکها با استفاده از روش زیر انتخاب می‌شوند:

$$Ci=(P_i)^e \bmod n$$

بلوکهای عددی پس از آنکه به توان e رسید ، باقیماندهٔ تقسیم آن بر n محاسبه می‌شود و بلوکهای Ci بدست می‌آید. بلوکهای Ci کدهای رمز هستند و بجای متن اصلی ارسال می‌شوند . پس در مثال فوق داریم:

Pi	0803	0418	1405	1200	1702	0723
$Ci=(P_i)^e \bmod n$	0779	1983	2641	1444	0052	0802

قبل از آنکه روش رمزگشائی را تشریح کنیم الگوی رمزنگاری RSA را بصورت جمع بندی شده ارائه می‌دهیم:

الف) رشته ای که باید رمز شود ، به بلوکهای K کاراکتری تبدیل می‌شود.

ب) هر بلوک طبق قاعدهٔ دلخواه بـه یـک عـدد صحیـح تبدیـل می‌شـود. (P_i)
ج) با جفت عدد صحیح (e,n) بـرای تمـام بلوکـها اعـداد جدیـدی طبـق رابطـه زیـر بدسـت می‌آیـد:

$$C_i=(P_i)^e \bmod n$$

د) کدهای C_i ، بجای کـد اصلـی ارسـال می‌شـود.

نکته اساسی در این الگو آنست کـه بـرای رمـزگشـائی کدهـا بـاید عـددی مثـل d پیـدا شـود که در رابطه زیر صـدق کنـد:

$$(x^{e.d}) \bmod n=x$$

با چنین عددی خواهیـم داشـت:

$$P_i =(C_i^d) \bmod n$$

یعنی مشـابه عمـل رمزنگـاری مجـدداً کدهـای رمـز بـه تـوان d رسـیده ، باقیمـاندهٔ آن بـر n محاسبه خواهد شد. کدهای حـاصل دقیقـاً همـان کدهـای اولیـه هسـتند.
به کلید (e,n) کـه بـا آن متـن رمـز می‌شـود "کلیـد عمومـی"[1] و بـه کلیـد (d,n) کـه بـا آن متن از رمز خارج می‌شـود "کلیـد خصوصـی" [2] اطـلاق می‌شـود.

قبل از آنکـه مثـالی دیگـر ارائـه بدهیـم اجـازه بدهیـد روش انتخـاب و معیارهـای e , d را که توسط ابداع کنندگان ایـن روش پیشـنهاد شـده اسـت ، معرفـی کنیـم:
الف) دو عـدد اوّل دلخـواه (ولـی بـزرگ) p , q انتخـاب کنیـد. (بـرای کاربردهـای عملـی اگـر ایـن اعـداد صـد رقمـی باشـند اطمینـان بخـش خواهـد بـود - یعنـی از مرتبـه 10^{100} باشـد-)
ب) عدد n , z را طبق دو رابطـه زیـر محاسـبه نمـاید:

$$n = p \times q$$
$$z = (p-1)(q-1)$$

ج) عـدد d را بگونـه ای انتخـاب کنیـد کـه نسـبت بـه z اول باشـد یعنـی هیـچ عـامل مشترکی که هر دو بر آن بخـش پذیـر باشـند نداشـته باشـد.
د) براساس d عدد e را بگونه ای انتخاب کنیـد تـا رابطـه زیـر برقـرار باشـد:

$$(e \times d) \bmod z = 1$$

---

[1]- Public key
[2]- Private key

نکاتی که در رمزنگاری باید رعایت شود آنست که کدهای P_i که به هر بلوک نسبت می‌دهیم باید $P_i < n$? 0 باشد بنابراین اگر بلوکها را بصورت رشته های k بیتی مدل می‌کنید باید شرط $2^k < n$ برقرار باشد.

برای یک مثال آموزشی فرض کنید بخواهیم رشته "SUZANNE" را رمز نمائیم. برای راحتی کار مجبوریم کلیدها را بسیار کوچک بگیریم ولی دقت داشته باشید در عمل اینطور نیست:

الف) دو عدد اول p=3 و q=11 را انتخاب می‌کنیم.

ب) عدد n=33 و z=20 بدست می‌آیند.

ج) عدد 7 که نسبت به z اول است را برای d انتخاب می‌نمائیم.

د) باید عدد e بگونه ای پیدا شود که رابطهٔ $(7 \times e) \bmod 20 = 1$ برقرار باشد این عدد را 3 انتخاب کرده ایم. (عدد ۲۳ هم قابل قبول است)

پس داریم :

(3,33)=(e,n) کلید عمومی

(7,33)=(d,n) کلید خصوصی

برای آشنایی با مراحل کار به شکل (۸-۱۱) دقت نمائید. بدلیل آنکه n عدد کوچکی است و باید $P_i<33$ باشد، مجبوریم بلوکها را یک کاراکتری فرض کرده و به A عدد ۱ ، به B عدد ۲ نسبت داده و بهمین ترتیب کاراکترها را به عدد صحیح تبدیل نمائیم.

سمبولهای متن	عدد P_i	محاسبهٔ P^3	$P^3 \bmod 33$	محاسبهٔ C^7	$C^7 \bmod 33$
S	19	6859	28	13492928512	**19**
U	21	9261	21	1801088541	**21**
Z	26	17576	20	1280000000	**26**
A	01	1	1	1	**1**
N	14	2744	5	78125	**14**
N	14	2744	5	78125	**14**
E	05	125	26	8031810176	**5**

رمزنگاری **رمزگشایی**

شکل (۸-۱۱) مثالی از رمزنگاری و رمزگشائی RSA

همانگونه که اشاره شد در عمل p و q صد رقمی انتخاب می‌شوند. (یعنی $(q \approx 10^{100}, P \approx 10^{100})$ بنابراین مقدار n از مرتبه 10^{200} (دویست رقمی) خواهد بود. سؤال آنست که عدد صحیح مربوط به بلوک های P_i که باید از n کوچکتر باشند چند بیتی خواهند بود؟

$$n < 10^{200} \text{ و } (10^{200} \approx 2^{664}) \Rightarrow n < 2^{664}$$

پس هر بلوک متن بایستی حداکثر 664 بیت یا معادل 83 کاراکتر هشت بیتی باشد.

ممکن است تاکنون ذهن شما مشغول این نکته شده باشد که چگونه می‌توان اعداد با این عظمت را به توان رساند. نکته ظریفی که وجود دارد آنست که برای محاسبهٔ $P^e \bmod n$ لازم نیست که اول P به تعداد e بار در خودش ضرب شود و بعد باقیمانده آن بر n بدست آید بلکه می‌توان پس از انجام یکبار عمل ضرب ، پیمانهٔ n (mod n) آن هم محاسبه شود تا نتیجهٔ محاسبه کاهش مقدار داشته باشد. برای روشن شدن قضیه به الگوی زیر دقت کنید:

$$7^3 \bmod 5 = ((7 \bmod 5) * 7^2) \bmod 5 = (2 * 7^2) \bmod 5 = ((2*7 \bmod 5) * 7) \bmod 5 = ((4*7) \bmod 5 = 3$$

فرض کنید بخواهیم A را به توان E برسانیم و بسط E در مبنای دودوئی بصورت زیر باشد:

$$E = (e_{k-1}, ..., e_0)_2 = \sum_{i=0}^{k-1} e_i 2^i$$

پس داریم :

$$A^E = A^{\sum_{i=0}^{k-1} e_i 2^i} = A^{2^{k-1}.ek-1} \times ..... \times A^{2^{e1}} \times A^{e0}$$

بنابراین برای رساندن A به توان E می‌توان براساس بسط باینری عدد E عمل کرد. این بسط دودویی مشکل رشد بی نهایت حاصل را حل نخواهد کرد. مشکل زمانی حل خواهد شد که ما بخواهیم $A^E \bmod n$ را محاسبه کنیم که دراین حالت باید پس از هر بار به توان رساندن ، باقیماندهٔ حاصل را بر n محاسبه کنیم تا نتیجه به زیر n کاهش یابد سپس ادامه می‌دهیم تا اعداد رشد بی نهایت نکند.

الگوریتم زیر حاصل $P^E \bmod n$ را محاسبه می‌کند و در Y بر می‌گرداند:

الف) نمایش دودوئی E بصورت $E = e_{k-1}e_{k-2}.....e_0$ مشخص می‌شود.

ب) $y \leftarrow 1$

ج) از شمارنده k-1 تا صفر بصورت شمارش معکوس دو عمل زیر تکرار می‌شود (i شمارنده است)

- y=(y*y) mod n

- اگر e_i مساوی 1 است آنگاه $y=(y*P) \bmod n$

د) نتیجه در y قرار دارد.

اگر دقت کافی داشته باشید الگوریتم فوق با مثال قبلی ($7^3 \bmod 5$) معادل خواهد بود. بنابراین مشکل حادّی در عملیات محاسبه کدهای رمز RSA و همچنین رمزگشائی آن وجود ندارد

به یاد داشته باشید که کلید رمزگذاری (e,n) یک کلید عمومی است و دلایلی بر سرّی و محرمانه ماندن آن وجود ندارد در حالی که کلید رمزگشائی (d,n) کلید اختصاصی است و باید سرّی باشد. برای شکستن رمز RSA باید مقدار d را از (e,n) به دست آورد و برای بدست آوردن d ابتدا باید n را به عوامل اول [1] تجزیه کرد تا بتوان p ، q و z و نهایتاً d را بدست آورد. با توجه به آنکه n معمولا دویست رقمی است با کامپیوترهای معمولی برای تجزیه چنین عددی چهار میلیون سال طول خواهد کشید!

به جدول (۹-۱۱) نگاه کنید فرض کنید کامپیوتری هر عمل را در یک میکروثانیه انجام بدهد این جدول زمان تجزیه یک عدد را به عوامل اوّل بر حسب تعداد ارقامِ عدد مشخص کرده است.

زمان محاسبه	تعداد ارقام
٤ ساعت	٥٠
١٠٤ روز	٧٥
٧٤ سال	١٠٠
چهار میلیون سال	٢٠٠
$٥*١٠^{١٥}$ سال	٣٠٠
$٤*١٠^{٢٥}$ سال	٥٠٠

جدول (۹-۱۱) زمان لازم برای تجزیه یک عدد به عوامل اوّل

[1] Prime factors

گرچه تحقیق بر روی تجزیهٔ اعداد به عوامل اول ادامه دارد ولی هیچ الگوریتم کارآمدتری که بتواند زمانهای جدول فوق را کاهش بدهد پیدا نشده است و بهمین دلیل بطور فراگیر از آن استفاده می‌شود.

۹) احراز هوّیت[1]

"احراز هوّیت" در شبکه های کامپیوتری بدین معناست که یک سرویس دهنده بتواند تشخیص بدهد کسی که تقاضائی را روی آن سیستم دارد شخص مجازی است یا یک شیّاد است . بعنوان مثال فرض کنید A می‌خواهد سعی کند از حسابش در بانکی مقداری پول را به حسابی دیگر منتقل نماید؛ سرویس دهندهٔ بانک باید مطمئن شود کسی که ادعا می‌کند شخص A است حقیقی است یا یک شیّاد است که خود را بجای A جا زده است چرا که با استفاده از تکنیکهائی می‌توان اطلاعاتی را که در قالب اسم رمز و کلمه عبور روی شبکه منتقل می‌شود دزدید و از آن استفاده کرد. تکنیکهای بهتری برای احراز هویت مبتنی بر اصول رمزنگاری با کلید عمومی وخصوصی قابل پیاده سازی است.

در مدل اوّل برای احراز هوّیت روشی را معرفی می‌کنیم که مبتنی بر اصول رمزنگاری و کلید رمز است . در این روش کلید رمزنگاری و رمزگشایی مشترک است:

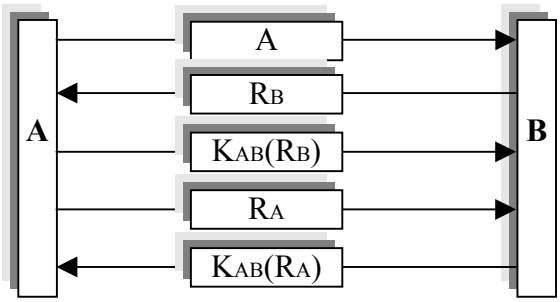
فرض کنید دو برنامه A , B می‌خواهند قبل از تبادل هر گونه اطلاعاتی هوّیت یکدیگر را تصدیق کرده و بعد انتقال داده‌ها را انجام بدهند. (A برنامهٔ سرویس دهنده و B برنامهٔ مشتری است) A مشخصه ای مثل شماره شناسائی یا کلمه عبور برای معرفی خود به B دارد که احتمال فاش شدن آن هم وجود دارد چرا که روی شبکه مبادله می‌شود و ممکن است استراق سمع شود.

حال به روند زیر دقت کنید:

فرض کنید A بعنوان مشتری ، شروع کنندهٔ ارتباط و B بعنوان سرویس دهنده پذیرنده ارتباط است . بعد از برقراریِ ارتباط و قبل از مبادله هر گونه داده ، عملیات

[1] Authentication

زیر مطابق شکل (۱۰-۱۱) صورت میگیرد. (در این روش A و B بر روی کلید رمز مشترکی توافق کرده اند که نام آنرا K_{AB} فرض کرده ایم و کاملاً سرّی است)



شکل (۱۰-۱۱) احراز هوّیت با کلید مشترک

الف) A با ارسال مشخصه شناسائی (ID) ، خود را به B معرفی می‌کند.

ب) B در پاسخ ، یک عدد تصادفی بزرگ مثلاً (صدرقمی) تولید کرده و برای A می‌فرستد. این عدد تصادفی در شکل با R_B مشخص شده است.

ج) A با کلید مشترک عدد دریافتیِ R_B را رمز کرده و برای B پس می‌فرستد. عدد رمز شده $K_{AB}(R_B)$ نامیده شده است.

د) B پس از دریافتِ عدد رمز شده آنرا با کلید مشترک رمزگشائی کرده و پس از مقایسه با عدد اصلی یعنی R_B می‌تواند مطمئن شود که هوّیت A محرز است و کسی قصد فریب ندارد.

ه) چون A هم تمایل دارد هویت طرف مقابل خود را تشخیص بدهد بنابراین او هم یک عدد تصادفی بزرگ تولید کرده و برای B می‌فرستد. B با کلید مشترک آنرا رمز کرده پس می‌فرستد؛ A هم با رمزگشائی و مقایسهٔ آن هوّیت B را تشخیص می‌دهد.

نکته مهم در روش های احراز هویت ، آنست که بر خلاف شماره شناسایی و کلمه عبور ، کلید رمز روی کانالهای ارتباطی ارسال نمی‌شود و توسط خود کاربر حفظ می‌شود تا مسئله افشای کلید از طریق استراق سمع ممکن نباشد؛ به روش فوق "احراز هویت دو مرحله ای"[1] گفته می‌شود.

[1] 2-Way Authentication

روش مشابه دیگری برای احراز هوّیت با استفاده از کلید عمومی [۱] وجود دارد که مبتنی بر روش رمزنگاری RSA است. برای تشریح این روش مجدداً فرض کنید A بعنوان برنامهٔ مشتری و B بعنوان سرویس دهنده تمایل دارند قبل از هر گونه مبادله اطلاعات ، هویت همدیگر را تصدیق نمایند. در این روش هریک از طرفین یک کلید سرّی و یک کلید عمومی دارند. A و B کلید عمومی همدیگر را می‌دانند ولی هرگز از کلید سرّی یکدیگر مطلع نیستند. ثابت شده که این روش صد تا هزار بار از روش قبلی سریعتر است. مراحل احراز هوّیت بصورت زیر است:

الف) A بعنوان شروع کننده ، شمارهٔ شناسائی خود و همچنین یک عدد تصادفی بزرگ R_A را با استفاده از کلید عمومیِ B به روش RSA رمز کرده و برای B می‌فرستد.

ب) سرویس دهندهٔ B که دارنده کلید سرّی خودش است آنرا رمزگشائی کرده و ضمن استخراج مشخصهٔ طرف مقابل و عدد R_A ، خودش عدد تصادفی R_B را تولید کرده و بهمراه یک کلید سرّی دیگر (یعنی جمعاً سه آیتم K_S , R_B , R_A) با استفاده از کلید عمومیِ A رمز کرده و برای A پس می‌فرستد. به کلید Ks که بعنوان کلید رمزنگاری جدید برای ادامه ارتباط مورد استفاده قرار می‌گیرد "کلید جلسه" [۲] گفته می‌شود.

ج) وقتی A اطلاعات رمز شدهٔ قبلی را از B دریافت و با استفاده از کلید سرّی خود رمزگشایی کرد ، اولا R_A را دارد که با مقایسهٔ آن متوجه می‌شود طرف مقابل همانی است که باید باشد . ثانیا کلید جدید Ks را دارد که رمزنگاری اطلاعات در مراحل بعد باید با آن کلید باشد ، ثانیا R_B را استخراج کرده است که باید با استفاده از کلید Ks مجدّداً آنرا رمز کرده برای A پس بفرستد. سرویس دهندهٔ B با دریافت آن و رمزگشائی هوّیت A را احراز می‌کند.

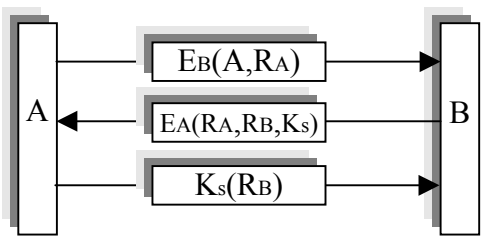
احراز هوّیت بر این مبناست که اگر B دروغگو باشد پس کلید سرّی لازم برای رمزگشایی اطلاعات ارسالی در مرحلهٔ الف را ندارد و اصلاً نمی‌تواند بفهمد چه کسی تقاضا داده و مراحل احراز هویّت ادامه نخواهد یافت. اگر A دروغگو باشد پس کلید سرّی لازم برای رمزگشایی اطلاعات در مرحلهٔ ب را نداشته و قادر به استخراج

---

[۱] Public key
[۲] Session key

"کلید جلسه" نبوده و بنابراین در اجرای مرحلهٔ سوم از احراز هویّت موفّق نخواهد شد.

در مرحلهٔ ۱ و ۲ از شکل (۱۱-۱۱) عملیات رمزنگاری با استفاده از کلید عمومی که علنی است انجام می‌شود ولی در مرحلهٔ ۳ رمزنگاری با کلید جدیدی انجام می‌شود که طرفین از آن اطلاع دارند. (یعنی کلید Ks یا همان کلید جلسه)
مراحل سه گانه این روش در شکل (۱۱-۱۱) مشخص شده است.



شکل (۱۱-۱۱) احراز هویت با استفاده از کلید عمومی و روش RSA

۱۰) امضاهای دیجیتالی [1]

یکی دیگر از منافع روش رمزگذاری RSA امضای دیجیتالی است . امضای دیجیتالی همانند امضای معمولی ابزاری است که برای رسمیت بخشیدن به یک پیام یا نامه استفاده می‌شود. با استفاده از امضاهای دیجیتالی :

- گیرندهٔ یک پیام بوسیله آن می‌تواند هوّیت فرستندهٔ آنرا تصدیق نماید.
- فرستندهٔ پیام نمی‌تواند محتوای پیام ارسالی اش را انکار کند.
- گیرندهٔ پیام نمی‌تواند پیامهای جعلی بسازد و همچنین دیگران قادر به جعل پیام نیستند.

[1] Digital Signature

امروزه با محوّل شدن امور مالی همانند حسابهای بانکی و کارتهای اعتباری به شبکه ها و ناامنی شبکه در مبادلهٔ اسناد و همچنین ناکارآمدی امضاهای دست نوشته روشی برای رسمیت بخشیدن به پیامها (یا فرامین) در شبکه وضع شده که حتی می‌تواند در دادگاه مورد استناد قرار گیرد. (شاید بتوان امضاهای دست نوشته را جعل کرد ولی در مورد امضاهای دیجیتالی هنوز امکان پذیر نشده است)

فرض کنید به یک سرویس دهندهٔ بانکی متصل شده و فرمان می‌دهید تا مقداری پول از حسابتان به حساب دیگری واریز شود. در اینجا فقط تشخیص و احراز هوّیت کافی نیست بلکه باید همانند امضاء روشی وجود داشته باشد که شما نتوانید در آینده اقامه دعوا کرده و عنوان کنید هیچگاه چنین تقاضایی نکرده اید.

روشهای متفاوتی برای پیاده سازی امضاهای دیجیتالی وجود دارد که دو مورد از آنها را معرفی می‌کنیم:

۱-۱۰) امضا با کلید سرّی [1]

در این روش که باید با کمک دولت یا انجمن حقوقدانان یا بانکها پیاده سازی شود ، مرکزی وجود دارد که معتمد همه است و قانون از آن حمایت می‌نماید. هر شخص با مراجعه حضوری به آن مرکز و تنظیم تعهّدنامه های لازم برروی یک کلید سرّی توافق می‌کند. (این مرکز را در ذهن خود محضر رسمی گواهی امضاء فرض کنید) بنابراین فقط شخص و مرکز مورد نظر آن کلید رمز را می‌دانند. حال فرض کنید که شخص A بخواهد سندی را در قالب یک پیام متنی امضا کرده برای B بفرستد (مثلاً سند تقاضای جابجایی پول از حساب بانکی) A دارای کلید رمز K_A است و بنابراین آیتمهای زیر را با کلید خودش رمز کرده و به همراه شماره شناسایی خود به سرویس دهندهٔ مرکز گواهی امضاء که فعلا آنرا BB می‌نامیم ارسال می‌نماید. آیتمهائی که رمز می‌شوند عبارتند از :

B : مشخصه شناسائی گیرنده نهائی پیام

R_A : یک عدد تصادفی بزرگ

[1] Secret key signature

t : زمان دقیق صدور پیام (تاریخ+زمان) معمولا زمان بر حسب گرینویچ –GMT– است .

P: متن پیام

حال مرکز گواهی امضای دیجیتالی که کلید سری A را در اختیار دارد متن و آیتم ها را رمزگشائی کرده و در صورتی که اینکار موفقیت آمیز انجام شد عملا هویت A تائید شده است چرا که هیچکس غیر از این دو کلید رمز را نمی‌داند.

در ادامهٔ این روند ، مرکز گواهی امضاء ضمن ثبت این تقاضا ، آیتمهای زیر را با کلید مشترک توافق شده بین خودش و B ، که کاملاً سرّی است ، برای B ارسال می‌نماید؛ این کلید را K_B فرض نمایید.

A : مشخصه فرستنده پیام

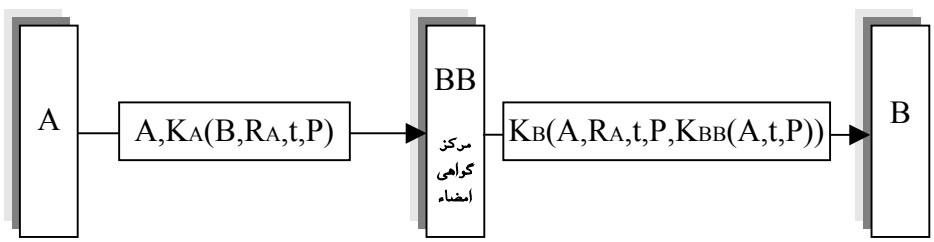
R_A : عدد ارسالی از A

t : زمان دقیق صدور پیام (تاریخ + زمان)

P : متن اصلی پیام ارسالی از A

$K_{BB}(A,t,p)$: آیتمهای رمز شدهٔ P,t,A که با کلیدی کاملاً سرّی رمز شده اند. این کلید را که فقط و فقط مرکز گواهی امضاء در اختیار دارد K_{BB} فرض نمایید.

پس از رمزگشایی پیام در B تمام آیتمها برای استناد قانونی ذخیره شده و می‌توان به محتوای پیام یا تقاضا عمل کرد. شمای کلّی روش امضای دیجیتالی با کلید سرّی در شکل (۱۲–۱۱) نشان داده شده است.



شکل (۱۲–۱۱) امضای دیجیتالی با کلید سرّی

اگر زمانی A منکر ارسال پیام P شود و ادعا کند پیام ساختگی است ، B می‌تواند متن رمز شده $K_{BB}(A,t,P)$ را به همراه متن اصلی پیام و R_A به دادگاه ارائه کند. کلید K_{BB} در اختیار مرکز گواهی امضاء است که مورد اعتماد دادگاه می‌باشد. مرکز گواهی امضاء متن رمز شدهٔ $K_{BB}(A,t,P)$ را رمزگشائی کرده و با اصل پیام مورد دعوا مطابقت می‌دهد و اگر مطابق بود B تبرئه می‌شود!

۲-۱۰) امضای دیجیتالی با کلید عمومی[1]

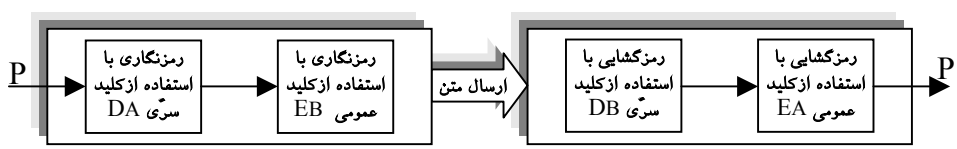
در این بخش روش ساده تری را که نیاز به مرکز واسطه ندارد و از رمزنگاری RSA استفاده میکند ، معرفی میکنیم. ساختار کلّیِ این روش در شکل (۱۳-۱۱) نشان داده شده است.

فرض کنید A و B می‌خواهند با هم ارتباط رسمی داشته باشند. A در رمزنگاری RSA دو کلید برای خود تعریف می‌کند: کلید D_A که سرّی و خصوصی است و کلید E_A که عمومی است. E_A را B می‌داند ولی D_A را فقط خودش خبر دارد.

B هم برای خود دو کلید تعریف می‌کند کلید D_B بعنوان کلید سرّی و کلید E_B که عمومی است. E_B را A هم می‌داند چون کلیدی عمومی است ولی D_B را فقط خودش خبر دارد.

وقتی A می‌خواهد متنی را برای B بفرستد اول آن را با کلید خصوصی اش یعنی D_A رمز می‌کند تا متن رمز شده $D_A(P)$ بدست آید. متن رمز شده جدید را مجدداً با کلید عمومی E_B رمز کرده و نتیجه را برای B می‌فرستد.

در B ابتدا متن دریافتی با کلید خصوصی یعنی D_B از رمز درآمده و مجدداً با کلید عمومی E_A رمزگشائی می‌شود تا متن اصلی بدست آید.



شکل (۱۳-۱۱) روش امضای دیجیتالی با کلید عمومی

[1] Public Key Signature

اصول کار این روش بر دو مورد زیر استوار است:

- اگر B جعلی و دروغین باشد هر چند می‌تواند کلید عمومی EA را داشته باشد ولی بهیچوجه کلید خصوصی B را نداشته و قادر به رمزگشایی متن نخواهد بود.
- اگر A جعلی و دروغین باشد چون کلید خصوصی A را ندارد بنابراین نخواهد توانست متن را رمز کند و اگر از کلید جعلی استفاده کند قابل بازیابی و رمزگشائی نخواهد بود.

در مجموع استفاده از امضاهای دیجیتالی به طرز فزاینده ای در حال رواج یافتن است ولی هنوز بسیاری از کشورها قوانینی در حمایت از آن وضع نکرده اند و استناد قانونی به چنین امضاهایی هنوز با مشکلاتی روبرو است.

۱۱) مراجع این فصل

مجموعهٔ مراجع زیر می‌توانند برای دست آوردن جزییات دقیق و تحقیق جامع در مورد مفاهیم معرفی شده در این فصل مفید واقع شوند.

"Computer Networks" , Andrew S.Tanenbaum, Third Edition, Prentice-Hall, 1996.	
RFC1244	"Site Security Handbook"
RFC1115	"Privacy Enhancement for Internet Electronic Mail: Part III—Algorithms, Modes, and Identifiers [Draft]," Linn, J.; 1989
RFC1114	"Privacy Enhancement for Internet Electronic Mail: Part II—Certificate-Based Key Management [Draft]," Kent, S.T.; Linn, J.; 1989
RFC1113	"Privacy Enhancement for Internet Electronic Mail: Part I—Message Encipherment and Authentication Procedures [Draft]," Linn, J.; 1989
RFC1108	"Security Options for the Internet Protocol," 1991