امنیت شبکه

جلسه دوم : رمزنگاری متقارن

تهیه و تنظیم: دکتر آرش حبیبی لشکری

منبع: کتاب اصول و مبانی امنیت شبکه (استانداردها و کاربردها)

اولین نسخه: دی 1393
بروزرسانی: دی 1393

فهرست:

- تعاریف
- رمزنگاری جریانی
  - RC4
- رمزنگاری بلوکی
  - DES
  - 3DES
  - AES
- اعداد تصادفی
- توزیع کلید متقارن

تعاریف اصلی

واژه های پایه در رمزگذاری متقارن

متن ساده: این متن در واقع پیام اصلی و یا اطلاعاتی است که به عنوان ورودی به الگوریتم داده می‌شود.

الگوریتم رمزگذاری: الگوریتم رمزگذاری تعویض و تبدیلهای بسیاری را روی متن اصلی اعمال خواهد نمود.

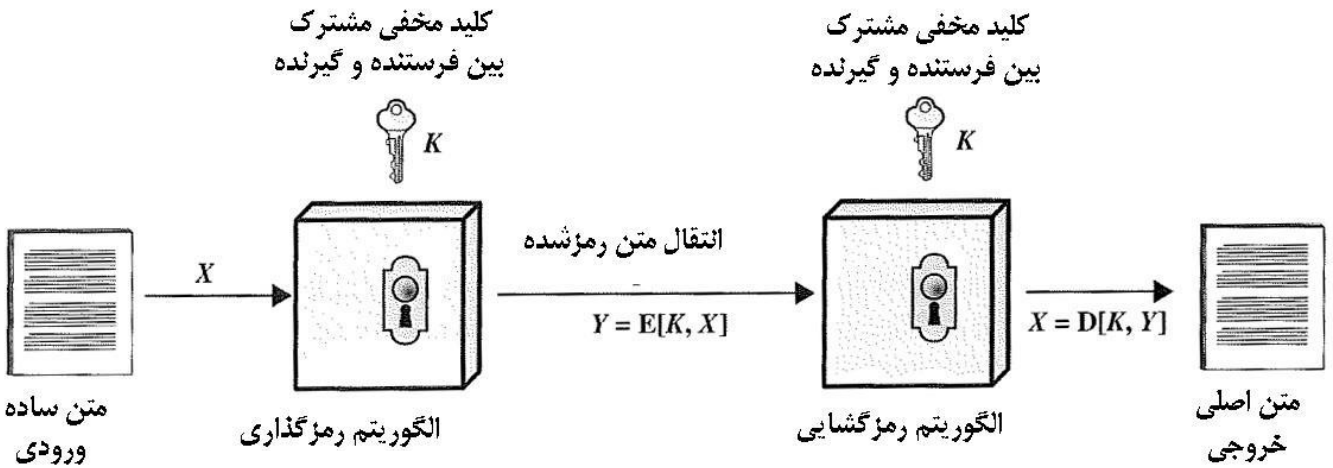
کلید مخفی: کلید مخفی نیز یکی از ورودی‌های الگوریتم است. درستی تعویض‌ها و تبدیلهای انجام شده توسط الگوریتم به کلید مخفی بستگی دارد.

متن‌رمزشده: یک پیام بهم ریخته است که به متن اصلی و کلید مخفی وابسته است. برای هر پیام ورودی به یک الگوریتم، دو کلید متفاوت باعث تولید دو متن رمزشده متفاوت خواهند شد.

الگوریتم رمزگشایی: در واقع همان الگوریتم رمزگذاری است که به صورت معکوس عمل می-نماید. این الگوریتم متن‌رمزشده و کلید مخفی را گرفته و متن ساده اصلی را تولید می‌نماید.

نامهای دیگر رمزنگاری متقارن :رمز نگاری مرسوم یا کلیدهای مخفی ویا رمزنگاری تک کلیدی

مدل ساده شده رمزنگاری متقارن



پیشنیازها

با توجه به تصویر صفحه قبل دو پیشنیاز برای استفاده امن از رمزنگاری متقارن وجود دارد:

1. به یک الگوریتم رمز گشایی قدرتمند احتیاج داریم. یعنی دوست داریم که حداقل الگوریتم طوری باشد که یک مهاجم که الگوریتم را می‌شناسد و دسترسی به یک یا چند نمونه از متن‌رمزشده دارد، نتواند متن‌رمزشده را رمزگشایی نموده یا کلید را کشف نماید. این نیاز در یک قالب قویتر چنین اظهار شده است: مهاجم نباید قادر به رمزگشایی متن‌رمز شده یا کشف کلید باشد، حتی اگر تعدادی از متون‌رمزشده را بهمراه متن اصلی آنها داشته باشد.

2. فرستنده و گیرنده باید یک کپی از کلید مخفی را به روشی امن دریافت کنند و باید کلید را ایمن نگهدارند. اگر کسی بتواند کلید را کشف نماید و الگوریتم را نیز بداند آنگاه همه ارتباطاتی که از این کلید استفاده نمایند را خواهد خواند.

نکته: امنیت رمزنگاری متقارن به رازداری کلید آن بستگی دارد نه رازداری الگوریتم آن. به زبان دیگر نیازی به مخفی نگه داشتن الگوریتم نیست بلکه باید کلید را مخفی نگه داریم. این همان ویژگی الگوریتم رمزنگاری متقارن است که باعث شده تا استفاده از آن بصورت گسترده امکان پذیر شود. نتیجه این امر این است که تولید کنندگان می‌توانند با پیاده سازی تراشه‌های کم هزینه الگوریتم‌های رمز نگاری داده‌ها را توسعه دهند.

دسته بندی سیستمهای رمزنگاری

سیستم‌های رمزنگاری به طور عمومی از سه بعد دسته بندی می‌شوند:

1. نوع عملیات بکار رفته برای تبدیل متن اصلی به متن‌رمز شده. همه الگوریتم های رمزنگاری بر اساس دو اصل عمومی عمل می کنند. یکی **جایگزینی** است که در آن هر عنصر در متن اصلی (بیت، نامه، گروهی از بیتها یا حروف) به یک عنصر دیگر نگاشت می‌شوند و دومی **جابجایی** است که در آن عناصر متن اصلی بازآرایی می‌شوند. پیش نیاز اساسی این است که هیچ اطلاعاتی گم نشود (به عنوان مثال همه عملیات بازگشت پذیر باشند). اغلب سیستم‌ها که به عنوان سیستمهای محصول شناخته می‌شوند، شامل مراحل متعددی از جایگزینی وترانهادن یا جابجایی هستند.

2. تعداد کلیدهایی که استفاده می‌شوند. اگر هر دو گیرنده و فرستنده از کلید مشابه استفاده نمایند آنگاه از سیستم با عنوان متقارن، تک کلید، کلید مخفی، یا رمزگشایی مرسوم یاد می‌شود. اگر فرستنده و گیرنده هر کدام از کلیدهای متفاوتی استفاده نمایند آنگاه از سیستم با عنوان نامتقارن، دو کلید، رمزنگاری با کلید عمومی یاد می‌شود.

3. روشی که بوسیله آن متن پردازش می‌شود. یک **رمز بلوکی** در هر زمان تنها یک بلوک از وروردی را پردازش نموده و به ازای هر بلوک ورودی یک بلوک خروجی تولید می‌نماید. یک **رمز جریانی** بطور مداوم اجزا ورودی را پردازش کرده و در هر زمان یک خروجی تولید نموده و همینطور به پیش می‌رود.

عملیات جایگزینی یا جابجایی

جابجایی:

متن اصلی: There are twenty students in this class.

متن رمز شده: ereht era ytnewt stenduts ni siht ssalc.

جایگزینی:

حروف اصلی: ABCDEFGHIJKLMNOPQRSTUVWXYZ

حروف جایگزین: CDEFGHIJKLMNOPQRSTUVWXYZAB

متن اصلی : There are twenty students in this class.

متن رمز شده: vjgtg ctg vygpva uvwfgpvu kp vjku encuu

انواع حملات مرتبط با پیامهای رمزگذاری شده

نوع حمله	شناخته‌شده برای رمزگشا
فقط - متن رمزشده	• الگوریتم رمزگذاری • متن رمزشده رمزگشایی می‌شود
متن اصلی شناخته شده	• الگوریتم رمزگذاری • متن رمزشده رمزگشایی می‌شود • یک یا چند جفت متن اصلی - متن رمزشده با کلید مخفی ترکیب شده‌اند
متن اصلی انتخابی	• الگوریتم رمزگذاری • متن رمزشده رمزگشایی می‌شود • متن اصلی انتخاب شده توسط رمزگشا، همراه با متن رمزشده مرتبط با کلید مخفی تولید خواهد شد
متن رمزشده انتخابی	• الگوریتم رمزگذاری • متن رمزشده رمزگشایی می‌شود • متن اصلی انتخاب شده توسط رمزگشا، همراه با متن رمزشده مرتبط با کلید مخفی تولید خواهد شد
متن انتخابی	• الگوریتم رمزگذاری • متن رمزشده رمزگشایی می‌شود • متن اصلی انتخاب شده توسط رمزگشا، همراه با متن رمزشده مرتبط با کلید مخفی تولید خواهد شد • متن رمزشده انتخابی توسط رمزگشا، بهمراه متن اصلی رمزگشایی متناظر با همراه کلیدهای مخفی تولید خواهد شد.

حملات (ادامه)

سخت ترین مشکل برای حمله زمانی وجود خواهد داشت که تنها متن‌رمزشده قابل دسترسی باشد. البته در برخی موارد ممکن است الگوریتم رمزنگاری نیز شناخته شده باشد اما به طور کلی، در حملات رمزنگاری چنین فرض میشود که مهاجم الگوریتم مورد استفاده برای رمزگذاری را می‌شناسد.

یکی از حملات احتمالی در این شرایط روش حمله جستجوی فراگیر برای امتحان کل کلیدهای ممکن است.

اگر فضای کلید بسیار بزرگ باشد، اینکار غیر ممکن خواهد بود.

بنابراین، مهاجم باید در تجزیه و تحلیل متن‌رمزشده به خود تکیه نموده و به طور کلی از آزمونهای آماری مختلف برای آن استفاده نماید.

کشف رمز

یک طرح رمزگذاری بصورت محاسباتی امن خواهد بود اگر متن‌رمزشده در این طرح یک یا هر دو شرط زیر را دارا باشد:

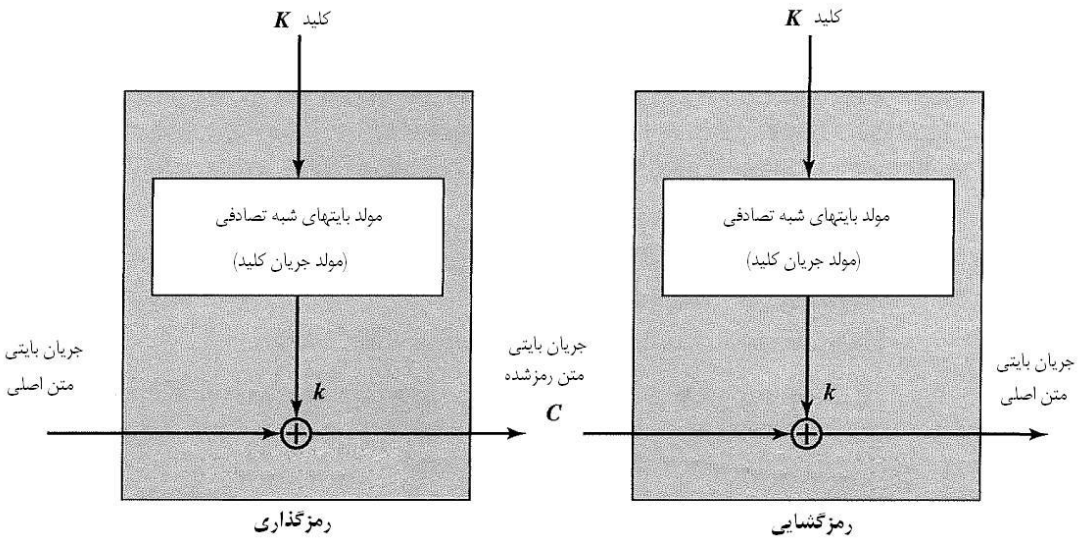
- هزینه‌های شکستن رمز فراتر از ارزش اطلاعات رمزگذاری شده باشد.

- زمان مورد نیاز برای شکستن رمز بیش از طول عمر مفید اطلاعات باشد.

رمز جریانی

دیاگرام رمز جریانی

یک رمز جریانی عناصر ورودی (بیت یا بایت) را به طور مداوم پردازش می‌کند و همین طور که جریان رو به جلو می‌رود یک عنصرخروجی نیز در هر زمان تولید می‌شود.



دیاگرام رمز جریانی

در این ساختار، یک کلید بعنوان ورودی به یک مولد بیت شبه تصادفی ارسال می‌گردد که یک جریان از اعداد 8 بیتی که ظاهرا تصادفی هستند را تولید می‌نماید.

خروجی مولد، جریان کلیدی نامیده شده و در هر زمان جریان متن ورودی از یک عملیات XOR بیتی برای ترکیب شدن استفاده می‌نماید.

برای مثال:

متن اصلی : 11001100

رشته کلید : ⊕01101100

متن رمز شده: 10100000

رشته کلید : ⊕01101100

متن اصلی : 11001100

ملاحظات مهم برای طراحی رمز جریانی

1. توالی رمزگذاری باید یک دوره تکرار طولانی داشته باشد. یک مولد عدد شبه تصادفی با استفاده از یک تابع یک جریان قطعی بیتی را تولید می‌کند که در نهایت تکرار می‌شود. هر چه دوره تکرار طولانی‌تر باشد رمزگشایی آن نیز دشواتر خواهد بود.

2. جریان کلید باید تا حد ممکن خواص تقریبی از یک جریان عدد تصادفی را داشته باشد. برای مثال، باید تقریبا تعداد برابری از 0ها و 1ها وجود داشته باشد. هر چه رشته کلید بیشتر تصادفی باشد، متن‌رمزشده تولید شده تصادفی‌تر بوده و لذا رمزگشایی آن مشکل‌تر خواهد بود.

3. نکته مورد توجه بنا بر تصویر پیشین این است که خروجی مولد عدد شبه تصادفی وابسته به مقدار کلید ورودی است. برای مقابله علیه حملات جستجوی فراگیر طول کلید باید به اندازه کافی بزرگ باشد. بنابراین، با تکنولوژی فعلی، طول کلید با اندازه حداقل 128 بیت مناسب خواهد بود.

الگوریتم RC4

این الگوریتم یک رمز جریانی با طول کلید متغیر بهمراه عملیاتی بر پایه بایت است.

همچنین این الگوریتم بر مبنای استفاده از جایگشت تصادفی پیاده‌سازی شده است.

الگوریتم RC4 در استانداردهای لایه سوکت‌های امن (SSL/TLS)که در اصل برای ارتباط بین مرورگر وب و سرویس‌دهنده‌ها تعریف شده است، بکار می‌رود.

*همچنین در پروتکلهای امنیتی WEP و WPA استاندارد WiFi که بخشی از استاندارد شبکه‌های بی‌سیم یا 802.11 هستند، نیز بکار رفته‌است.

*A survey on wireless security protocols (WEP, WPA and WPA2/802.11 i), 2nd IEEE International Conference on Computer Science and Information Technology (ICCSIT), 2009

ساختار RC4

گام اول – مقدار دهی اولیه S : در آغاز، درایه‌های آرایه S برابر مقادیر از 0 تا 255 به صورت صعودی به طوری که S[0]=0، S[1]=1، ... و S[255]=255 قرار می‌گیرند. بردار موقت T، نیز ایجاد شده است. اگر طول کلید k برابر 256 کیلوبایت باشد، آنگاه K به T انتقال داده می‌شود. در غیر این صورت، برای یک کلید به طول keylen بایت، اولین عناصر keylen ازT ازمقادیر K کپی شده و سپس K هر چند بار که لازم باشد برای پر کردن T تکرار خواهد شد. تکرار این عملیات اولیه می‌تواند به صورت زیر شرح داده شود:

```
/* Initialization*/
    For i= 0 to 255 do
    S[i] = i;
    T[i] = K [i mod keylen] ;
```

الگوریتم RC4

گام دوم – تولید جایگشتها: سپس از T برای تولید جایگشت اولیه S استفاده می‌کنیم. این کار از S[0] شروع شده و تا S[255] ادامه دارد و برای هر S[i] مقدار S[i] را با بایت دیگر در S با توجه به طرح دیکته شده توسط T[i] تعویض می‌نماید:

```
Initial Permutation of S */
j = 0;
for i = 0 to 255 do
j = ( j + S[i] + T[i]) mod 256;
Swap (S [i ], S [j]);
```

از آنجا که تنها عملیات در S، تعویض است لذا تنها اثر همان جایگشت خواهد بود. البته توجه داشته باشید که همچنان S شامل تمام اعداد از 0 تا 255 خواهد بود.

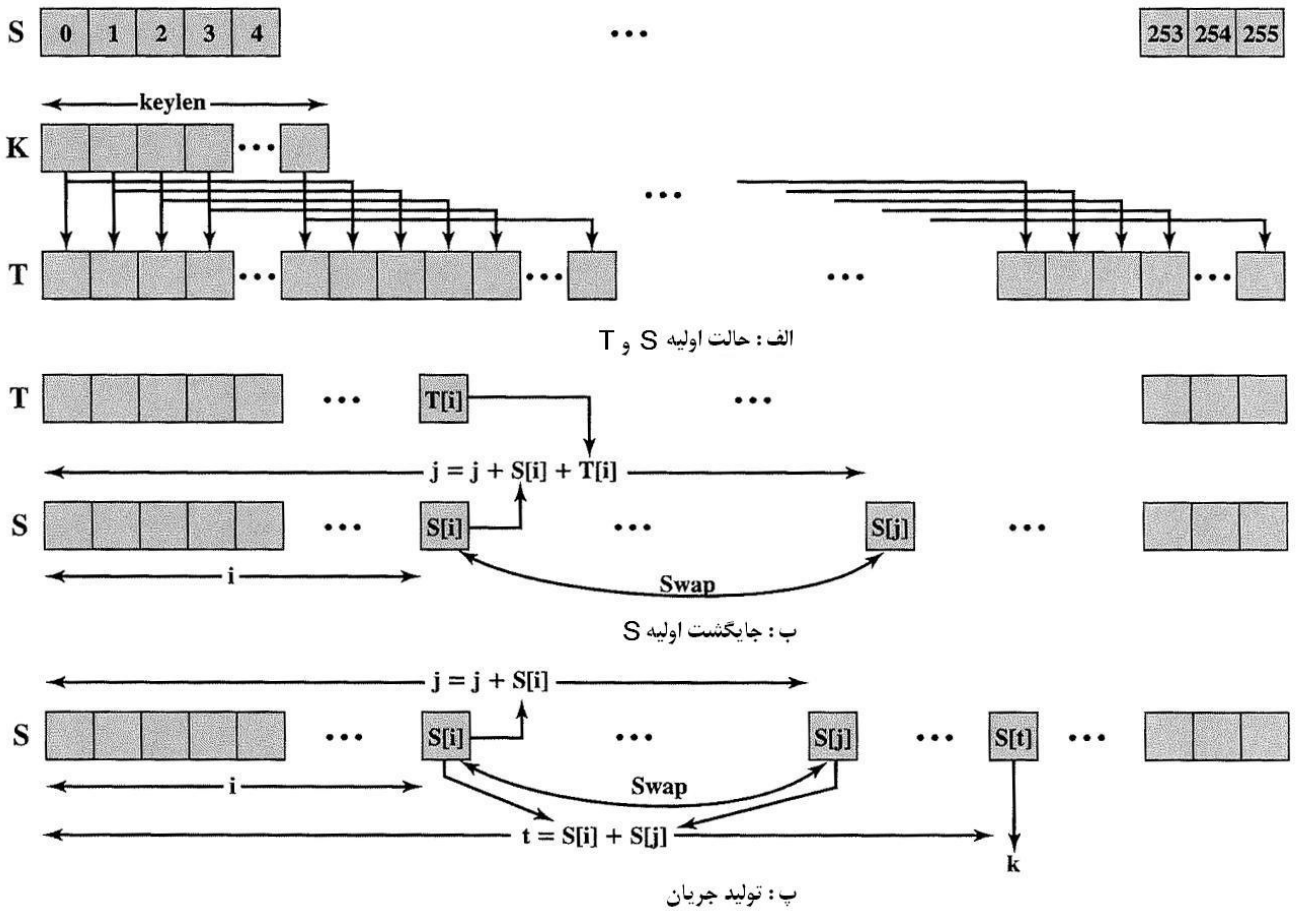
الگوریتم RC4

گام سوم - تولید جریان: هنگامی که بردار S مقداردهی اولیه می‌شود، کلید ورودی دیگر استفاده نمی‌شود. تولید جریان شامل چرخش تمام عناصرS[i] است، برای هر S[i]، مبادله S[i] با بایت دیگر درS مطابق طرح دیکته شده توسط پیکربندی فعلی S است.
پس از اینکه به S[255] رسید، این روند دوباره باS[0] شروع شده و ادامه می یابد:

```
/* Stream Generation
    i, j :: 0;
    while ( true)
    i = (i + 1) mod 256;
    j =(j + S[i]) mod 256;
    Swap (S [ i],S[ J]);
    t= (S [i] + S[j]) mod 256;
    k =S [t];
```

برای رمزگذاری، مقدار K با بایت بعدی از متن XOR می‌شود. برای رمزگشایی نیز، مقدار K با بایت بعدی از متن‌رمزشده XOR می‌شود.

دیاگرام الگوریتم RC4



قدرت RC4

تعدادی از مقالات تجزیه و تحلیل روشهای حمله به RC4 را بیان نموده‌اند، اما هیچ یک از این روش‌ها در برابرRC4 با طول معقول، کلیدی مانند 128 بیت، کاربرد نداشته و قابل استفاده نخواهند بود.

البته یک مشکل جدی‌تری در اینجا وجود دارد که محققین نشان داده اند که پروتکل WEP، که برای محرمانه ماندن اطلاعات در شبکه‌های بی‌سیم 802.11 در نظر گرفته شده است، در برابر یک روش حمله خاص آسیب‌پذیر است. در اصل، مشکل با خود RC4 نیست بلکه با روشی است که در آن کلیدها برای استفاده به عنوان عبور به RC4 در این الگوریتم تولید میشوند. این مشکل خاص به نظر مرتبط با برنامه‌های دیگری که از RC4 استفاده می‌کنند نبوده و می‌تواند درWEP با تغییر روشی که در آن کلید تولید میشود، رفع گردد.

توزیع کلید بین دو طرف در رمزگذاری متقارن چگونه است؟

رمز بلوکی

ساختار رمزی Feistel

بسیاری از الگوریتم‌های رمزگذاری بلوکی متقارن، از جمله DES، دارای ساختاری هستند که برای اولین بار توسط Horst Feistel از آی - بی - ام در 1973 شرح داده شده (تصویر بعدی).

ورودی الگوریتم رمزگذاری یک بلوک متن اصلی بطول 2w بیت و یک کلید بنام k است. بلوک متن به دو نیمه LE0 و RE0 تقسیم می‌شود. در هر دو نیمه داده‌ها از n دور پردازش عبور کرده و سپس با هم ترکیب شده و بلوک رمزشده را بوجود می‌آورند.

هر دور i دارای دو بخش است:

ورودی LE_{i-1} و RE_{i-1} که از دور پیشین بدست آمده‌اند

یک زیر کلید k_i ، که از کلید کلی k مشتق شده است

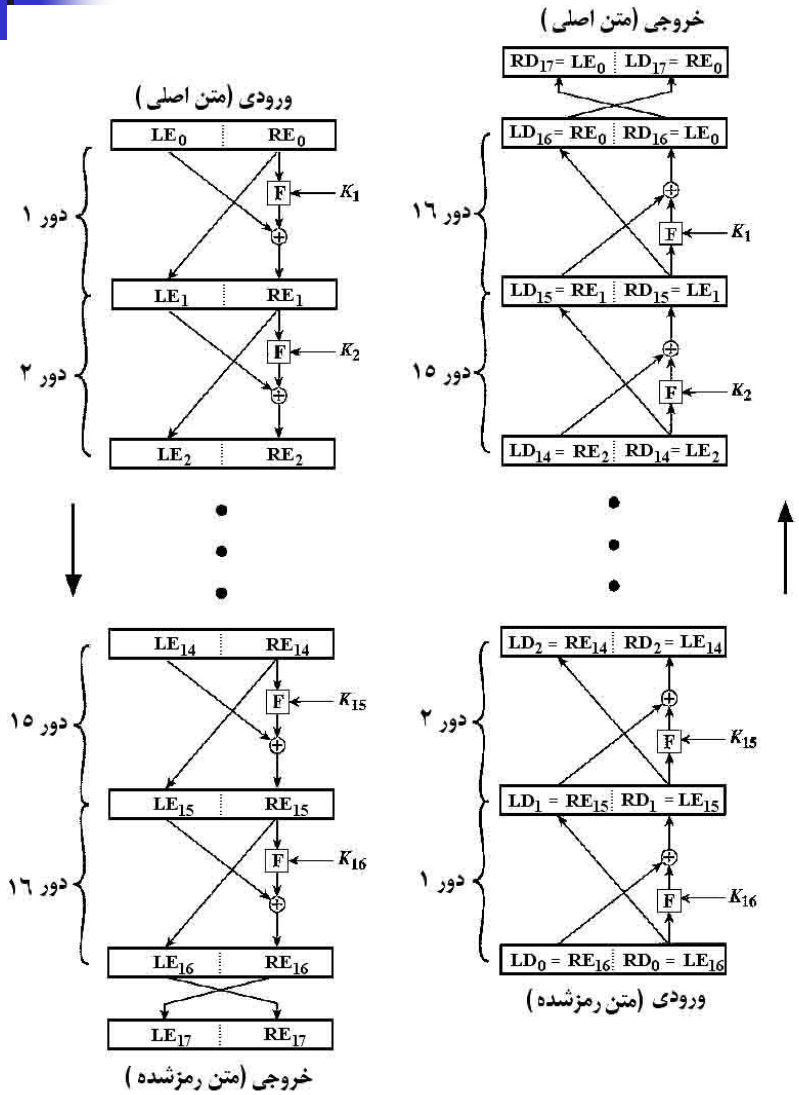
در تصویر بعدی نمونه ای با 16 دور از این الگوریتم نمایش داده شده است که البته می‌تواند هر تعداد دور مورد استفاده قرار گیرد. سمت راست تصویر فرایند رمزگشایی را نشان میدهد که دقیقا بلعکس مراحل رمزگذاری است.

ساختار رمزی Feistel

یک جایگزینی در نیمه سمت چپ داده‌ها انجام می‌شود. این کار با اعمال تابع گردکردن F به نیمه سمت راست داده و گرفتن XOR از خروجی آن تابع و نیمه چپ داده-ها انجام می‌شود.

تابع گردکردن دارای ساختار کلی یکسان برای هر دور بوده اما هر دور بوسیله زیر کلید k_i پارامتری می‌شود.

پس از این جایگزینی، یک جایگشت انجام شده که شامل تبادل دو نیمه از داده‌ها خواهد بود.



رمزگذاری بلوکی متقارن و ملاحظات مهم

رمزگذاری بلوکی متقارن در اصل **دنباله ای از دورهاست که در هر دور جایگزین‌ها و جایگشتهای مشروطی با استفاده از مقدار یک کلید مخفی انجام می‌گیرد.**

تحقق دقیق رمزنگاری بلوکی متقارن به انتخاب پارامترهای زیر و ویژگی‌های طراحی آن بستگی دارد:

اندازه بلوک: اندازه بلوک بزرگتر به معنای امنیت بیشتر است (اگر همه موارد دیگر یکسان باشند) ولی سرعت رمزگذاری/ رمزگشایی کاهش خواهد یافت. (بلوک ۱۲۸ بیتی یک استاندارد جهانی است)

اندازه کلید: کلید با اندازه بزرگتر به معنای امنیت بیشتر است اما ممکن است که باعث کاهش سرعت رمزگذاری/ رمزگشایی نیز بشود. (رایجترین طول کلید 128 بیت است)

تعداد دورها: ماهیت رمزنگاری بلوکی متقارن این است که یک دور آن امنیت کافی را ارائه خواهد داد اما چندین دور باعث افزایش امنیت آن خواهند شد. (تعداد دور معمول 10 تا 16)

الگوریتم تولید زیر کلید: پیچیدگی بیشتر در این الگوریتم منجر به سختی بیشتر عملیات رمزگشایی می‌گردد.

تابع گردکردن: بازهم، پیچیدگی بیشتر در این تابع به طور کلی به معنای مقاومت بیشتر در برابر عملیات رمزگشایی است.

سایر ملاحظات مهم

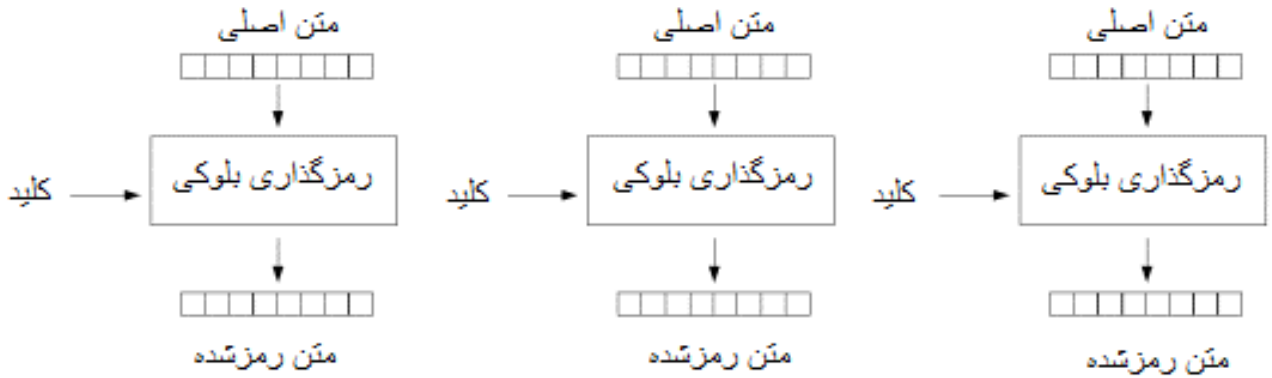
دو ملاحظه دیگر نیز در طراحی یک رمزنگاری بلوکی متقارن وجود دارند:

نرم افزار رمزگذاری/ رمزگشایی سریع : در بسیاری از موارد، اگر پیاده سازی سخت افزاری مشکل باشد آنگاه رمزگذاری در برنامه های نرم افزاری یا توابع سودمندی تعبیه شده است. بنابراین، سرعت اجرای الگوریتم یکی از نگرانی‌ها خواهد بود.

سهولت تجزیه و تحلیل: اگر چه ترجیح می‌دهیم الگویتم را بخاطر عملیات کشف رمز تا حد ممکن دشوار سازیم اما منفعت بزرگی در ساخت الگوریتمی با ساختار ساده بخاطر تجزیه و تحلیل آسان آن وجود دارد. چراکه اگر الگوریتم ساده باشد آنگاه تجزیه و تحلیل آن برای تعیین میزان آسیب پذیری در رمزگذاری آسان‌تر بوده و در نتیجه می‌توان یک الگوریتم با درجه بالاتری از اطمینان را که قویتر هم خواهد بود، پیاده‌سازی نمود.

رمزگذاری بلوکی متقارن

ساختار عمومی رمزهای بلوکی متقارن بصورت زیر است:



رمزهای بلوکی شایع‌ترین الگوریتم‌های رمزنگاری متقارنی هستند که تاکنون استفاده شده‌اند. همانطور که در تصویر مشاهده میشود، رمزهای بلوکی، متن ورودی را بصورت بلوک‌هایی با طول ثابت پردازش می‌کنند و یک بلوک رمزشده با طول مساوی را به ازای هر بلوک متن اصلی تولید می‌نمایند. سه مورد از مهم‌ترین رمزهای بلوکی متقارن عبارتند از استاندارد رمزگذاری داده‌ها (DES)، DES سه‌گانه (3DES)، و استاندارد رمزگذاری پیشرفته (AES).

استاندارد رمزگذاری داده‌ها DES

متن ورودی 64 بیت و طول کلید 56 بیت و تعداد دور 16

متنهای ورودی طولانی‌تر در بلوک‌های 64- بیتی شکسته شده و پردازش خواهند شد.

ساختار DES یک تغییر جزئی روی شبکه Feistel که قبلا گفته شد دارد. از کلید اصلی 56 بیتی تعداد 16 زیر کلید ایجاد شده که برای هر دور یکی از این زیر کلیدها استفاده خواهند شد.

فرایند رمزگشایی DES بطور کلی مشابه رمزگذاری است:

متن‌رمزشده به عنوان ورودی الگوریتم DES استفاده می‌شود ولی از زیر کلید k در جهت معکوس استفاده می‌شود. به این معنا که از k_{16} در تکرار اول استفاده شده و از k_{15} در تکرار دوم و به همین ترتیب تا زمانی که k_1 در دور 16ام و آخرین تکرار مورد استفاده قرار گیرد.

توانایی ها و نگرانیهای DES

نگرانیها در مورد توانایی DES به دو دسته تقسیم بندی شده‌اند:

- یکی نگرانی‌ها در مورد خود الگوریتم
- دیگری نگرانی‌ها در مورد استفاده از یک کلید 56 بیتی

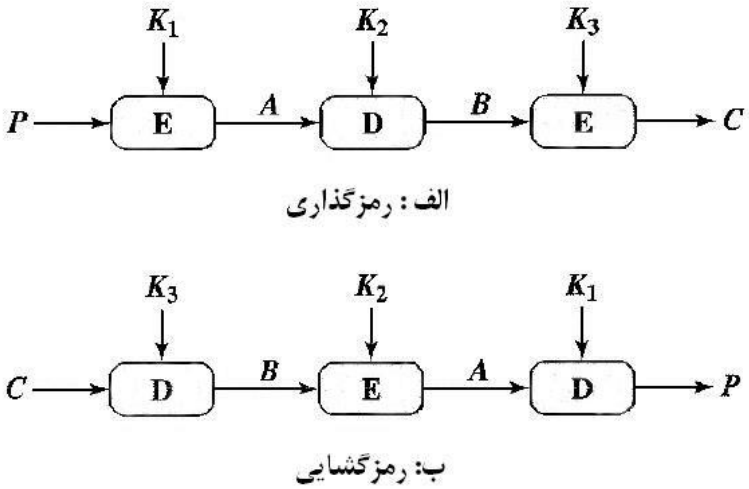
اولین نگرانی اشاره به این احتمال دارد که آنالیز رمز با بهره‌گیری از ویژگی‌های الگوریتم DES امکان پذیر باشد. در طول سالها، از آنجایی که تلاشهای زیادی برای پیدا کردن و بهره برداری از نقاط ضعف الگوریتم صورت گرفته است، لذا الگوریتم DES به الگوریتم رمزگشایی تبدیل شده که بیشترین مطالعات بر روی آن وجود دارد. با وجود روش‌های متعدد، هیچکس تاکنون موفق به کشف یک ضعف مهلک در DES نشده است.

نگرانی بعدی در این الگوریتم طول کلید است. با کلیدی به طول 56 بیت تعداد 2^{56} حالت ممکن وجود دارد که تقریبا برابر $7.2*10^{16}$ تعداد کلید خواهد بود. بنابراین، چنین بنظر می‌آید که یک حمله جستجوی فراگیر در اینجا غیر ممکن خواهد بود. **آیا واقعا چنین است؟**

الگوریتم DES سه‌گانه یا 3DES

در واقع این الگوریتم از سه بار اجرای الگوریتم DES با سه کلید تشکیل شده است (تصویر زیر).
تابع زیر توالی رمزگذاری، رمزگشایی و رمزگذاری را نمایش می‌دهد:

$$C = E(K_3, D(K_2, E(K_1, P)))$$



که در آن:

C= متن‌رمزشده

P= متن اصلی رمز نشده

E(K,X)= رمزگذاری X با کلید K

D(K,Y)= رمز گشایی Y با کلید K

رمزگشایی نیز به سادگی همین عمل را با کلیدهای معکوس انجام می‌دهد :

$$P = D(K_1, E(K_2, D(K_3, C)))$$

با سه کلید مجزا، الگوریتم 3DES طول کلید موثری برابر 168 بیت خواهد داشت و لذا حملات جستجوی فراگیر به شکل موثری ناموفق خواهند بود.

توانایی ها و نگرانیهای 3DES

اشکالات اصلی 3DES را بر اساس آنکه از DES منتج شده است میتوان به دو گروه دسته بندی نمود:

اشکال اصلی 3DES این است که الگوریتم در نرم افزارهای طراحی شده نسبتا کند است. چرا که الگوریتم DES اصلی برای پیاده‌سازی در سخت‌افزارهای اواسط 1970 طراحی شده بود و کد نرم افزاری کارآمدی را تولید نمی‌نمود. لذا الگوریتم 3DES که تعداد دورهایش سه برابر DES است به نسبت کندتر خواهد بود.

اشکال دوم این است که همانند DES الگوریتم 3DES نیز از اندازه بلوک 64 بیتی استفاده می‌کند. و میدانیم که به دلایل کارایی و امنیت، اندازه بزرگتر برای بلوک مطلوبتر است.

استاندارد رمزگذاری پیشرفته یا AES

یا همان الگوریتم Rijndael که با نام AES مشهور است

بلوک به طول 128 بیت و طول کلید می‌تواند 128، 192 و 256 بیت باشد

یکی از ویژگی‌های قابل توجه این ساختار آن است که یک Fiestel نیست (تصویر بعدی):

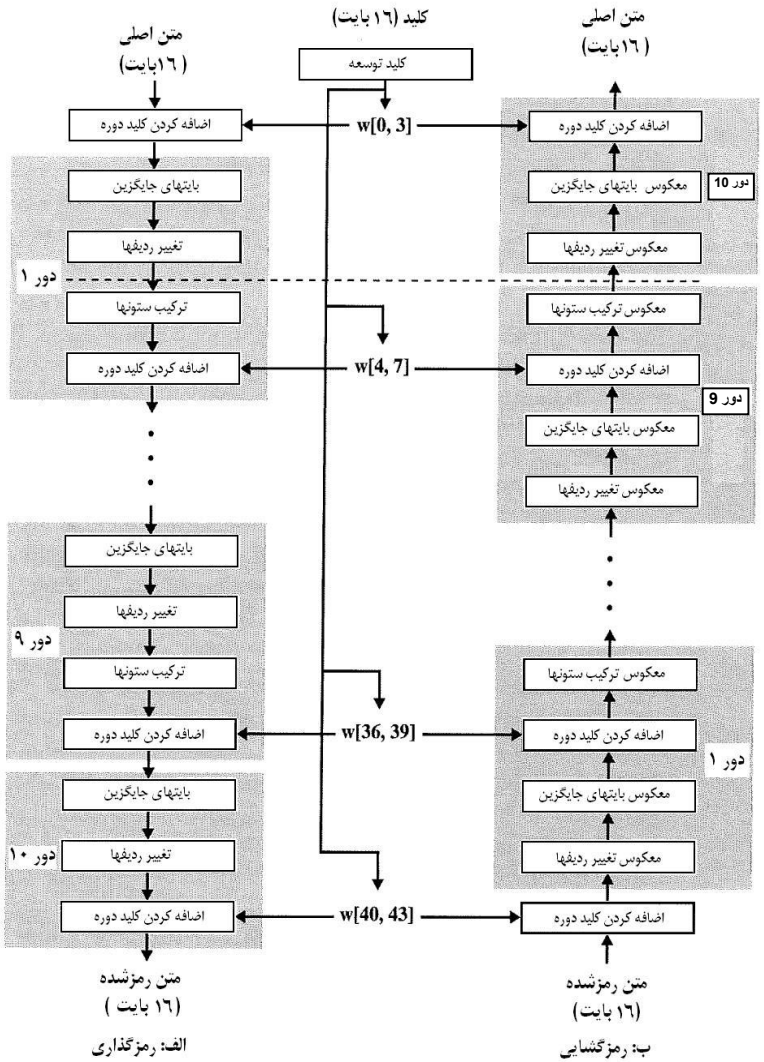
کلیدی که به عنوان یک ورودی درنظر گرفته شده است به یک آرایه چهل و چهار عنصری با کلمات 32 بیتی بنام W[i] گسترش خواهد یافت. چهار کلمه مجزا (128 بیت) به عنوان یک کلید گردشی برای هر دور بکار خواهد رفت.

چهار مرحله مختلف که شامل یک عمل جایگشت و سه عمل جایگزینی است، اجرا خواهند شد (بایت‌های جایگزین، شیف ردیف، ترکیب ستونها، اضافه کردن کلید دوره).

رمزنگاری با مرحله اضافه کردن کلید دوره شروع شده و با 9 دور ادامه می‌یابد که هر کدام شامل 4 مرحله است و دور 10 که دارای 3 مرحله است نیز دور پایانی خواهد بود.

امنیت AES چگونه است؟

دیاگرام AES


متن اصلی
(۱۶بایت)
کلید (۱۶ بایت)
متن اصلی
(۱۶بایت)
کلید توسعه
اضافه کردن کلید دوره
w[0, 3]
اضافه کردن کلید دوره
بایتهای جایگزین
معکوس بایتهای جایگزین
دور 10
تغییر ردیفها
معکوس تغییر ردیفها
دور ۱
ترکیب ستونها
معکوس ترکیب ستونها
اضافه کردن کلید دوره
w[4, 7]
اضافه کردن کلید دوره
دور 9
بایتهای جایگزین
معکوس بایتهای جایگزین
تغییر ردیفها
معکوس تغییر ردیفها
دور ۹
ترکیب ستونها
معکوس ترکیب ستونها
اضافه کردن کلید دوره
w[36, 39]
اضافه کردن کلید دوره
دور ۱
بایتهای جایگزین
معکوس بایتهای جایگزین
تغییر ردیفها
معکوس تغییر ردیفها
دور ۱۰
اضافه کردن کلید دوره
w[40, 43]
اضافه کردن کلید دوره
متن رمزشده
(۱۶ بایت)
متن رمزشده
(۱۶ بایت)
الف: رمزگذاری
ب: رمزگشایی

سوالات مرتبط

1. عناصر اصلی یک رمز متقارن را نام ببرید؟
2. دو تابع اصلی که در الگوریتم های رمزگذاری بکار میروند، کدامند؟
3. چه تعداد کلید برای ارتباط بین دو نفر ازمیان یک رمز متقارن مورد نیاز خواهد بود؟
4. فرق بین یک رمز قطعه‌ای (بلوکی) و یک رمز دنباله‌ای (جریانی) در چیست؟
5. دو خط مشی عمومی برای حمله به یک رمز کدامند؟
6. چرا تعدادی از روشهای عملیاتی در رمز قطعه‌ای تنها از رمزگذاری استفاده می‌کنند در حالی که سایرین از رمزگذاری و رمزگشایی با هم استفاده می‌نمایند؟
7. چرا بخش میانی یک 3DES بیشتر از آنکه رمزگذاری باشد یک رمزگشایی است؟

خلاصه:

رمزنگاری متقارن، رمزهای جریانی متقارن، رمزهای بلوکی متقارن،
AES ،3DES ،DES ،RC4

جلسه بعدی: رمزنگاری نامتقارن

منبع: کتاب اصول و مبانی امنیت شبکه (استانداردها و کاربردها)

ترجمه: دکتر آرش حبیبی لشکری، مهندس نسرین بدیع، مهندس فرنار توحیدی

هیچ راهی برای به دست آوردن تجربه به جز از
طریق تجربه وجود ندارد.