مجله علمی- پژوهشی "پدافند الکترونیکی و سایبری"
ویژه‌نامه کنفرانس بین‌المللی ترکیبیات، رمزنگاری و محاسبات، ص ۶۷ -۶۳

اصول جدید برای الگوریتم‌های رمزنگاری

نوید عبودی[۱]، ناصر هاشمی[۲*]

۱- کارشناس ارشد، دانشکده ریاضی و علوم کامپیوتر دانشگاه امیر کبیر، ۲- استادیار، دانشکده ریاضی و علوم کامپیوتر دانشگاه امیر کبیر

(دریافت: ۹۵/۰۷/۰۵، پذیرش: ۹۶/۰۳/۰۶)

چکیده

با توجه به اهمیت اصول رمزنگاری، هدف ما در این مقاله این است که یک سری اصول جدید برای طراحی الگوریتم‌های رمزنگاری با حفظ اصول قبلی ارائه دهیم که با استفاده از این اصول، زمان لازم برای پیداکردن کلید بالاتر رفته و هم‌چنین منجر به افزایش سختی تحلیل الگوریتم‌ها شود و در نتیجه بتوانیم از امنیت حفظ اطلاعات بالاتری برخوردار باشیم.

واژه‌های کلیدی: رمزنگاری متقارن، اصول کرشهف، انتقال امن اطلاعات

۱- مقدمه[۱]

کرشهف در سال ۱۸۸۳، اصولی را برای رمزنگاری اطلاعات اعلام کرد که این اصول در شش بند ارائه گردید و تاکنون از آن‌ها در طراحی الگوریتم‌های رمزنگاری استفاده شده است. نکته قابل توجه این است که این اصول بیش از ۱۳۰ سال قبل نوشته شده است و بنا به پیشرفت سریع در علم و فناوری و افزایش سرعت رایانه ها و سریع‌ترشدن تجزیه و تحلیل الگوریتم‌های پیچیده در رایانه، نیازمند به افزودن اصول جدید در جهت تکمیل اصول قبلی است. برای به‌روزماندن الگوریتم‌ها با توجه به سرعت پیشرفت فناوری، روز به روز این اصول باید ارتقاء یابند و بر اساس آن‌ها، الگوریتم‌های جدیدی نوشته شوند به‌طوری‌که درجه سختی تحلیل الگوریتم‌ها پیچیده‌تر شود تا هم‌چنان از امنیت حفظ اطلاعات بالاتری برخوردار باشند.

در ادامه ابتدا به بیان اصول کرشهف پرداخته و سپس در بخش سه، اصول بهبودیافته رمزنگاری بیان خواهد شد و در بخش چهار، اصول جدید را تجزیه و تحلیل خواهیم کرد و با اصول کرشهف مقایسه کرده و دلایل اهمیت اصول جدید به صورت مجزا گفته خواهد شد و در انتها نتیجه‌گیری کلی صورت خواهد گرفت.

۲- اصول کرشهف

اصول کرشهف در شش بند ارائه شده است که این شش بند عبارتند از [۱]:

۲-۱) سیستم رمزنگاری از لحاظ تئوری و عملی غیرقابل شکست باشد.

۲-۲) سیستم رمزنگار باید هیچ نکته پنهان و محرمانه‌ای نداشته باشد بلکه تنها چیزی که باید سری نگه داشته شود کلید رمز است. طبق اصل اساسی کرکهافس، طراح سیستم رمزنگار نباید جزئیات سیستم خود را حتی از دشمنان مخفی نگاه دارد.

۲-۳) کلید رمز باید به گونه‌ای قابل انتخاب باشد که اولاً بتوان به راحتی آن را عوض کرد و ثانیاً بتوان آن را به خاطر سپرد و نیازی به یادداشت‌کردن کلید رمز نباشد.

۲-۴) متون رمزنگاری‌شده باید از طریق خطوط تلگراف قابل مخابره باشند.

۲-۵) دستگاه رمزنگاری یا اسناد رمزشده باید توسط یک نفر قابل حمل باشد.

۲-۶) سیستم رمزنگاری باید به سهولت قابل راه‌اندازی و کاربری باشد. چنین سیستمی نباید به آموزش‌های مفصل و رعایت فهرست بزرگی از قواعد و دستورالعمل‌ها نیاز داشته باشد.

۳- اصول بهبودیافته رمزنگاری

بعضی از اصول کرشهف مثل بندهای ۲-۴ الی ۲-۶ به‌خاطر پیشرفت فناوری و وسایل ارتباطی امروزه امری بدیهی است و اگر این اصول رعایت نشود الگوریتم نوشته‌شده کارایی خاصی نخواهد داشت و قابل اجرا نخواهد بود و هر الگوریتم کارایی این سه اصل را داراست. سه اصل مابقی یعنی ۲-۱ الی ۲-۳ در هر زمانی لازم-الاجرا است و باید در تمام الگوریتم‌های رمزنگاری رعایت شود تا الگوریتم نوشته‌شده حافظ امنیت داده‌ها باشد. امروزه به‌دلیل

* رایانامه نویسنده مسئول: nhashemi@aut.ac.ir

وجود رایانه و سرعت پیشرفت روزافزون فن‌آوری، این اصول و الگوریتم‌ها بیش از پیش نیازمند تغییر و بهبود هستند و باید متناسب با نیاز زمان نوشته شوند و به آن‌ها رفته رفته باید اصول جدیدی اضافه شده و بهبود یابند تا هم‌چنان امنیت حفظ اطلاعات بالاتری داشته باشیم و اطلاعات شخصی ما و دیگران به راحتی در اختیار فرد ثالث قرار نگیرد و متحمل ضررهای جانی و مالی نشویم.

در زیر مهم‌ترین اصول کرشهف همراه با اصول جدید آمده است که در آن، اصول ۳-۱ و ۳-۲ همان اصول قبلی ۲-۱ و ۲-۲ می‌باشند و اصل ۳-۳ تکمیل یافته اصل قبلی یعنی ۲-۳ است و بعد از آن اصول پیشنهادی جدید در سه بند ۳-۴ ، ۳-۵ و ۳-۶ ارائه شده است:

۳-۱) یک سیستم رمزنگاری از نظر تئوری و عملی باید غیرقابل شکست باشد.

۳-۲) سیستم رمزنگار باید هیچ نکته پنهان و محرمانه‌ای نداشته باشد بلکه تنها چیزی که باید سری نگه داشته شود کلید رمز است. طبق اصل اساسی کرکهافس، طراح سیستم رمزنگار نباید جزئیات سیستم خود را حتی از دشمنان مخفی نگاه دارد.

۳-۳) باید کلید رمز تصادفی باشد به‌طوری‌که با احتمال بالا قابل حدس‌زدن نباشد.

۳-۴) باید طول کلید ثابت و مشخص و محدود به الگوریتم رمزنگاری نبوده و به صورت تصادفی انتخاب شود و یا در اختیار رمزکننده باشد.

۳-۵) باید خود الگوریتم رمزنگاری تصادفی باشد به‌طوری‌که برای هر متن و کلیدی به‌صورت یکسان عمل نکند.

۳-۶) حتی‌الامکان از قطعه‌بندی متن اصلی به اندازه متن کلید پرهیز شده و از رمزنگاری گسسته و قطعه‌ای استفاده نشود بدین مفهوم که کل متن به‌صورت یک‌جا رمزنگاری شود و یا تعداد قطعات متن اصلی در کم‌ترین حد ممکن باشد (که تعداد این قطعات به سرعت و اندازه حافظه رایانه‌ها بستگی دارد و با پیشرفت فناوری و افزایش سرعت رایانه‌ها اندازه قطعه‌های انتخاب‌شده بیش‌تر می‌شود به‌طوری‌که طول قطعه انتخاب‌شده برابر طول متن اصلی می شود).

ملاحظه۱:

رعایت اصول بالا یک الگوریتم خوب برای رمزنگاری برای متونی با طول بیش‌تر از یک حد معین می‌دهد ولی برای متونی با طول کلید کم‌تر از آن دارای امنیت بالایی نیست و برای به‌دست‌آوردن این حد کران باید به مسئله زیر توجه کرد. با توجه به سرعت رایانه‌ها و تعداد حالات الگوریتم و طول کلید می‌توان به یک حد پایین مطلوب برای طول کلید رسید به‌طوری‌که با احتمال پایین در زمان طولانی الگوریتم شکسته شود و برای شکستن الگوریتم با احتمال بالای $\frac{1}{2}$ زمان زیادی لازم باشد. اگر p را برابر احتمال موفقیت و k برابر تعداد بیت‌ها و ۲^{k} تعداد حالات باشد و حالات طوری انتخاب شوند که حالت تکراری در انتخاب‌هایمان نداشته باشیم بنا به مسئله گوی و سکه و یا مسئله روز تولد داریم:

$$p > 1 - \frac{2^k}{2^k} * \frac{2^k - 1}{2^k} * \frac{2^k - 2}{2^k} * \dots * \frac{2^k - i}{2^k}$$

عبارت بالا برای iهـای بیش‌تر از $1.17 * 2^{k/2}$ احتمال p بالای ۵۰٪ نتیجه می‌دهد.

$$\text{مدت زمان جستجو} = \frac{\text{تعداد کل حالات جستجو}}{\text{سرعت هر رایانه} \times \text{تعداد رایانه}}$$

برای مثال، اگر از ۲۰۰۰ رایانه برای رمزگشایی استفاده شود و هر رایانه بتواند در هر ثانیه، ۱۰۰۰ حالت از کل حالات را برای پیداکردن کلید مسئله بررسی کند و حالت‌هایی که برای رایانه‌ها انتخاب می‌شوند حالت تکراری نداشته باشند برای شکست الگوریتم با احتمال ۵۰٪ در مدت زمان بیش از یک سال باید طول کلید حداقل برابر k بیت باشد.

اگر متن با کلیدی به‌طول k رمزنگاری شده باشد آن‌گاه در کل ۲^{k} حالت خواهیم داشت و برای پیداکردن کلید رمز با احتمال بالای ۵۰٪ باید حداقل $1.17 * 2^{\frac{k}{2}}$ حالت مختلف را جستجو کنیم. در مثال بالا اندازه حد پایین برای k به صورت زیر به‌دست می‌آید:

$$1\ year = 365 * 86400 = \frac{1.17 * 2^{k/2}}{2000 * 1000}$$

$$k = 2 * \log_2\left(\frac{365 * 86400 * 1000 * 2000}{1.17}\right) \approx 91.23 \approx 92$$

با توجه به مفروضات بالا، برای این که بخواهیم برای شکست الگوریتم با احتمال بالای ۵۰٪ مدت زمانی بیش‌تر از یک سال طول بکشد، باید طول کلید بیش از ۹۲ بیت باشد. که اگر تعداد رایانه‌ها و یا سرعت رایانه‌ها زیادتر شود باید بر این مقدار اضافه گردد تا احتمال شکست الگوریتم ما هم‌چنان کم باشد و با احتمال بالای ۵۰٪ به راحتی نشکند. در الگوریتم اولیه DES که از طول کلیدی برابر ۵۶ بیت استفاده می‌شد اگر بخواهیم با احتمال بالای ۵۰٪ متن رمز را پیدا کنیم، نیاز داریم که $۲^{۲۸}$ حالت را جستجو کرده و اگر از یک رایانه استفاده کنیم که در هر ثانیه ۱۰۰۰ حالت را بتواند جستجو کند، در مدت زمانی کم‌تر از ۲۵۶۰۰۰ ثانیه برابر کم‌تر از ۳ روز می‌توان کلید رمز را پیدا کرد و یکی از علل شکست الگوریتمی DES همین مسئله می‌تواند باشد.

۴- تجزیه و تحلیل اصول جدید و مقایسه با اصول کرشهف

اصولی که کرشهف اعلام کرد، اصول اولیه و کلی برای رمزنگاری است و رعایت این اصول برای هر الگوریتم رمزنگاری ضروری است و پایه و اساس الگوریتم‌های رمزنگاری است و شرط لازم برای ایجاد امنیت بالا است ولی شرط کامل و کافی نیست. به عبارت دیگر، اگر الگوریتم رمزنگاری این اصول را رعایت نکند قطعاً امنیت بالایی ندارد و برای حفظ اطلاعات مناسب نیست.

الگوریتم‌های موجود که در حال حاضر استفاده می‌شوند همگی بر اساس این اصول نوشته شده‌اند. الگوریتمی مثل الگوریتم DES که با استفاده از این اصول طراحی شده بود، از سال ۱۹۷۷ تا ۱۹۹۸ استفاده می‌شد ولی در سال ۱۹۹۸ Hellman و Diffie یک الگوریتم و سخت‌افزاری معرفی کردند که با این سخت‌افزار، الگوریتم DES شکسته شد و در عرض کم‌تر از هفت ساعت، کلیدرمز متن رمزشده پیدا گردید و رابطه‌ای بین متن رمزشده و متن اصلی با کلید پیدا شد. بعد از آن، این الگوریتم را تغییر دادند و به‌جای یک بار رمزنگاری، از سه بار با سه کلید متفاوت رمزنگاری کردند: در رمزنگاری DES جدید، ابتدا با کلید اول رمزنگاری می‌شود و بعد با کلید دوم رمزگشایی می‌شود و سپس با کلید سوم دوباره رمزنگاری می‌شود تا با این کار درجه سختی الگوریتم بالاتر رفته و به راحتی رابطه‌ای بین متن رمزشده و متن کلید پیدا نشود. این درحالی است که روش کلی این الگوریتم تغییر چندانی نکرده است و فقط تعداد تکرار آن زیاد شده است. الگوریتم‌های دیگر رمزنگاری که مهم‌ترین و مشهورترین آن‌ها الگوریتم AES است همانند الگوریتم DES، به صورت قطعه‌ای و گسسته رمزنگاری می‌شوند و طول متن رمز این الگوریتم‌ها ثابت است [۳-۲].

برای برطرف‌کردن مشکل‌های گفته‌شده و بهبود الگوریتم‌های رمزنگاری و بالابردن درجه سختی آن‌ها، اصول جدید را در بخش سه ارائه کردیم که در ادامه به بررسی و تحلیل این اصول جدید خواهیم پرداخت.

الف) اثبات درستی‌اصول ۳-۱ و ۳-۲

این دو اصل، همان اصول اصلی ۲-۱ و ۲-۲ کرشهف هستند که قبلاً اهمیت استفاده از این دو اصول برای برقراری امنیت بالا توسط خود کرشهف اثبات شده است.

ب) اثبات درستی‌اصل ۳-۳

اصل ۳-۳ اصلاح‌شده اصل ۲-۳ کرشهف است. در اصل ۲-۳ گفته شده بود که کلید رمز باید طوری نوشته شود که قابل حفظ‌کردن باشد که در این صورت متن کلید به متون زبان و اعداد وابستگی پیدا کرده و تعداد حالات کلید کم‌تر می‌شود و با احتمال زیاد، قابل حدس‌زدن خواهد بود. درحالی‌که امروزه از رایانه و وسایل الکتریکی در رمزنگاری استفاده می‌شود و خود این سیستم‌ها توانایی تولید کلید مجزا برای هر متن را دارند. امروزه چون از الگوریتم‌های شبه‌تصادفی در تولید کلید رمز استفاده می‌شود باید این الگوریتم‌ها طوری طراحی شوند که برای کلید، کل حالات موجود را با احتمال یکسان تولید کنند. برای هر متنی سعی شود کلید مجزایی تولید گردد، کلید دو متن یکسان انتخاب نشوند، متن آن‌ها ساده و وابسته به هم نباشند. در نتیجه حدث‌زدن کلید متون سخت‌تر می‌شود و تعداد حالات انتخاب‌شده برای کلید در رمزنگاری بیش‌تر خواهد شد و رمزنگاری امن‌تری خواهیم داشت.

ج) اثبات درستی اصل ۳-۴

اصل ۳-۴ گویای این است که باید طول متن کلید یکسان نباشد و یکی از دلایل شکست الگوریتم‌هایی مثل DES همین طول پایین و ثابت متن کلید است. برای مثال اگر طول متن اصلی ۱۰۲۴ بیت باشد و طول متن کلید الگوریتم‌های موجود ۱۲۸ بیت باشد چون این متن به قطعاتی به طول ۱۲۸ بیت تقسیم می‌شوند و از بین $۲^{۱۰۲۴}$ حالت موجود برای متن اصلی ما کافی است بین $۲^{۱۲۸}$ حالت مختلف به دنبال جملات با معنی بگردیم که تعداد جملات با معنی به این نسبت خیلی کم‌تر می‌شود و اگر طول کلید یکسان نباشد برای یک متن ۱۰۲۴ بیتی ما به تعداد زیر:

$$P = 2^1 + 2^1 + 2^3 + \cdots + 2^{1024} = 2^{2048} - 1$$

حالت خواهیم داشت که این عدد نسبت به $۲^{۱۲۸}$ یک عدد خیلی بزرگی است.

در الگوریتم‌های موجود برای هر طولی از متن اصلی به اندازه m کافی است ما بین $۲^{۱۲۸}$ حالت به دنبال جواب بگردیم درحالی که اگر طول متن رمز ثابت نباشد تعداد این حالات برابر $۲^{۲m}$ - ۱ حالت می‌شود یعنی کل بازه را می‌توانند در برگیرد و پیداکردن متن اصلی از روی متن رمزشده سخت‌تر می‌شود.

ه) اثبات درستی اصل ۳-۵

بنا به اصل ۳-۵، علاوه بر این که متن و طول کلید تصادفی است باید جزئیات الگوریتم نیز تصادفی باشد و در طراحی الگوریتم از الگوریتم‌های تولید شبه اعداد تصادفی استفاده کرد و الگوریتم را به‌صورت شبه تصادفی اجرا کرد. به‌عنوان مثال اگر الگوریتم AES دارای چهار مرحله a، b،c وd است و این مراحل شش بار به‌صورت زیر و با نظم و ترتیب عمل می‌کنند:

a1,b1,c1,d1,a2,b2,c2,d2,a3,b3,c3,d3,a4,b4,c4,d4,a5,b5,c5,d5,a6,b6,c6,d6

تصادفی‌شدن الگوریتم به این معنی است که این چهار مرحله با این نظم و ترتیب اجرا نشوند و به صورت تصادفی با هم جابه‌جا شوند و به‌جای یک راه و روش ثابت بتواند یکی از روش‌های موجود در زیر را طی کند:

1)a1,b1,c1,d1,a2,b2,c2,d2,a3,b3,c3,d3,a4,b4,c4,d4,a5,b5,c5,d5,a6,b6,c6,d6

2)b1,c1,d1a1,b2,c2,d2,a2,b3,c3,d3,a3,b4,c4,d4,a4,b5,c5,d5,a5,b6,c6,d6,a6

3) ,b1,c1,d1,a2, ,c2,d2,a3,b3, ,d3,a4,b4,c4, , ,b5,c5,d5,a6, ,c6,d

4)a1,b1, ,d1,a2,b2, ,d2,a3,b3,c3, ,a4,b4,c4,d4,a5, ,c5,d5,a6,b6,c6,d6

5)a1,b1,d1,c1,a2,b2,d2,c2,a3,c3,b3,d3,a4,c4,b4,d4,a5,b5,c5,d5,a6,b6,c6,d6

6)……

در این‌جا چون مراحل اجرای الگوریتم و خود این مراحل به صورت تصادفی، ثابت و منظم برای همه متن‌ها اجرا نمی‌شوند، درجه سختی الگوریتم بالاتر می‌رود و فردی که سعی در شکستن الگوریتم دارد دقیقاً نمی‌داند که متن داده‌شده با کدام روش از روش‌های موجود اجرا شده است و این کار مدت زمان شکسته‌شدن الگوریتم را بالاتر می‌برد.

الگوریتم برای متن و کلیدهای یکسان باید به یک صورت رمز شود تا فرد گیرنده بتواند متن رمزشده را به متن اصلی برگرداند و فرد گیرنده اعداد تصادفی تولیدشده توسط رمزکننده را بتواند به همان ترتیب تولید کند. مگر این‌که روش حل الگوریتم به طرقی بین فرستنده و گیرنده ردوبدل شده باشد و در این الگوریتم‌ها از الگوریتم‌های برگشت‌پذیر استفاده شود تا فرد گیرنده به متن اصلی دست پیدا کند.

و) اثبات درستی اصل ۳-۶

اصل ۳-۶ که مهم‌ترین اصل از اصول بالاست حائز این نکته است که باید در الگوریتم از قطعه‌بندی متن اصلی و رمزنگاری گسسته پرهیز شود. این کار دو نتیجه مهم دارد یکی این‌که معلوم نمی‌شود طول متن کلید چند بیت است و برای به‌دست‌آوردن یک حالت از حالات موجود باید به‌جای تحلیل یک قطعه باید کل متن رمزشده را تحلیل کرد تا به جواب رسید و نتیجه دوم این است که زمان لازم برای به‌دست‌آوردن کلید رمز برای فرد ثالث که قصد شکست الگوریتم را دارد، بالاتر می‌رود. برای مثال، اگر طول متن اصلی برابر ۱۰۰۰۰۰۰ بیت باشد و طول متن کلید برابر ۱۰۰ بیت باشد به‌جای تحلیل ۱۰۰ بیت باید ۱۰۰۰۰۰۰ بیت را تحلیل کرد و زمان لازم برای این‌کار ۱۰۰۰۰۰ برابر می‌شود.

ملاحظه ۲:

یکی از اصولی که در اصول کرشهف مطرح نشده است و با رعایت این اصل احتمال شکسته‌شدن الگوریتم و متن اصلی برابر صفر می‌شود و نمی‌توان از روی متن رمزشده به متن اصلی رسید، اصل ۳-۴ است. یعنی الگوریتمی که در آن بتوان کل حالات موجود را تولید کرد. وقتی که کل حالات موجود را بتوان تولید کرد دیگر نمی‌توان فهمید که متن اصلی با کدام یک از این متن‌ها رمز شده است، مگر با دانستن متن کلید. اصل جدید ۳-۴ این امکان را به فرستنده و گیرنده می‌دهد که طول متن کلید را هم‌اندازه طول متن کلید اصلی انتخاب کنند که این یک اصل مهم در رمزنگاری است. درحالی‌که الگوریتم‌های قبلی چنین امکانی را برای فرستنده فراهم نکرده بودند و این یکی از نواقص الگوریتم‌های موجود و اصول کرشهف است که به این نکته توجه نکرده است و الگوریتم‌هایی که ساخته شده‌اند، این اصل اساسی رمزنگاری را نادیده گرفته‌اند.

ملاحظه ۳:

سوالی که در این‌جا مطرح می‌شود این است که آیا متنی که با اصول جدید با طول کلید ۴۰ بیت رمز می‌شود دارای امنیت بالاتری نسبت به الگوریتم‌های ساخته‌شده با اصول قبلی با طول کلید ۶۴ بیت دارد یا نه؟

در جواب این سوال باید گفت که در الگوریتم‌های قبلی برای هر متن اصلی، به طول m بیت و متن رمز به طول k بیت نیاز داریم که به دنبال متن کلیدهایی بگردیم که طول آن‌ها به اندازه k بیت بوده و نیازی به جستجوی حالاتی به طول بیش‌تر و کم‌تر از k بیت نیست، چون k عددی معلوم و ثابت است ولی در اصول جدید این عدد k عدد ثابتی نیست و هر عددی بین ۱ تا m را می‌تواند به خود بگیرد. برای مثال، اگر طول کلید ۴۰ بیت باشد فرد ثالث که قصد شکستن الگوریتم را دارد دقیقاً نمی‌داند که طول کلید، چند بیت است و باید به‌دنبال کل حالات بگردد. اگر طول کلید ۴۰ بیت باشد باید کلیدهایی به طول ۳۰ بیت و ۳۸ بیت را هم بررسی کنیم. در الگوریتم‌های قبلی تعداد حالات کلید برابر ۲^{k} حالت بوده و در اصول جدید این تعداد حالات به تعداد $۲^{۲m}-۱$ حالت افزایش می‌یابد و در نتیجه، درجه سختی الگوریتم برای بعضی از حالات پیچیده و برای بعضی از حالات آسان است. ولی در کل چون تعداد حالات بیش‌تر است و ما دقیقاً این حالات سخت و آسان را نمی‌دانیم برای به‌دست‌آوردن جواب باید همه این‌ها را بگردیم که در بین حالات موجود، حالات سخت هم

انتخاب می‌شوند و در کل درجه سختی الگوریتم بالاتر از الگوریتم‌های دیگر می‌شود.

۵- نتیجه‌گیری

بنا به اصول اولیه رمزنگاری کرشهف، الگوریتم‌های نوشته‌شده با این اصول، به‌راحتی شکسته می‌شوند زیرا بعضی از نکات مهم رمزنگاری در آن نادیده گرفته شده است. یکی از این نکات این است که همه این الگوریتم‌ها به صورت قطعه‌ای و گسسته متن‌های اصلی را رمزنگاری می‌کنند و امروزه با احتمال بالا بعضی از این الگوریتم‌ها به‌راحتی شکسته شده‌اند. بنابرین، احتمال شکسته‌شدن الگوریتم‌های مشابه زیاد است. با پیشرفت فناوری و افزایش سرعت رایانه‌ها، لازم است که اصول رمزنگاری داده‌ها اصلاح شده و بهبود یابند و الگوریتم‌های جدید براساس اصول معرفی‌شده در این مقاله برای رمزنگاری طراحی شوند. امروزه باید سعی شود در تولید الگوریتم‌های جدید رمزنگاری، از الگوریتم‌های پیچیده ریاضی مثل الگوریتم‌های تصادفی و الگوریتم‌های کوانتومی در رمزنگاری استفاده شود تا درجه امنیت و حفظ اطلاعات هم‌چنان بالا باشد واحتمال شکسته‌شدن الگوریتم‌ها کم‌تر شود [۴].

۶- مراجع

[1] "Kerckhoffs' Law," WordNet 3.0, Farlex clipart collection, 2003-2008. Princeton University, Clipart.com, Farlex Inc. 23 May2016. http://www.thefreedictionary.com/Kerckhoffs%27+Law

[2] A. K. Mandal, C. Parakash, and A.Tiwari, Performance evaluation of cryptographic algorithms: DES and AES, In Electrical, Electronics and Computer Science (SCEECS), 2012 IEEE Students' Conference on, IEEE, pp. 1-5, March 2012.

[3] L. B. Kish, D. Abbott, and C. G. Granqvist, "Critical analysis of the Bennett–Riedel attack on secure cryptographic key distributions via the Kirchhoff-law–Johnson-noise scheme," PloS one, vol. 8, no. 12, p. e81810, 2013.

[4] T. Adamski, "Introduction to optical quantum cryptography," In Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments 2007, pp. 693723-693723, International Society for Optics and Photonics, October 2007.