**به نام خدا**

دانشگاه علم و صنعت ایران

دانشکده مهندسی کامپیوتر

گزارش کارآموزی دوره کارشناسی مهندسی کامپیوتر – سخت افزار

امنیت شبکه

Network Security

محل کارآموزی:

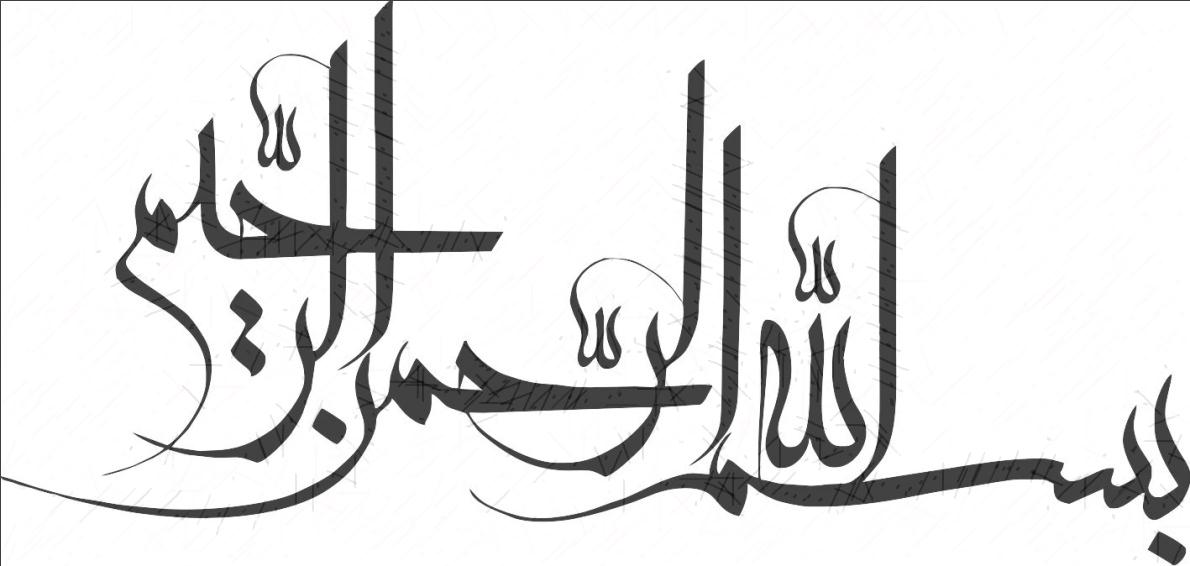
موسسه تحقیقات ارتباطات و فناوری اطلاعات

استاد راهنما:

نویسنده گزارش:

زمستان 1391

[این صفحه به عمد سفید گذاشته شده است]



تقدیم به

پیشگفتار

فهرست مطالب

موضوع صفحه

چکیده .............................................................................................................................................................................................................................................................. XI

معرفی دانشگاه، محل کارآموزی، استاد راهنما، استاد سرپرست کارآموزی و دانشجو .................................................................................................................. XII

**فصل اول: مفاهیم امنیت .................................................................................................................................................................................... 1**

* 1. تعریف واژگان کلیدی ............................................................................................................................................................................................................................ 2
  2. رمزنگاری .................................................................................................................................................................................................................................................. 6

1-2-1 رمزنگاری متقارن ..................................................................................................................................................................................................................... 7

1-2-1-1 رمزگذاری بلاکی ................................................................................................................................................................................................................. 9

1-2-1-2 رمزگذاری جریانی ............................................................................................................................................................................................................... 9

1-2-2 رمزنگاری نامتقارن ................................................................................................................................................................................................................ 11

1-2-2-1 کاربردهای سیستم های رمزگذاری کلید عمومی ..................................................................................................................................................... 13

1-2-2-2 سرویس های رمزنگاری کلید عمومی .......................................................................................................................................................................... 14

1-2-2-3 الگوریتم کلید عمومی RSA ........................................................................................................................................................................................ 14

* 1. اصالت پیام ............................................................................................................................................................................................................................................. 15

1-3-1 کد اصالت پیام ....................................................................................................................................................................................................................... 16

1-3-2 تابع درهم آمیزی یک طرفه ............................................................................................................................................................................................... 17

1-3-2-1 نیازمندی های تابع درهم آمیزی ................................................................................................................................................................................. 18

1-3-2-2 توابع درهم آمیزی ساده ................................................................................................................................................................................................. 18

1-3-3 امضا دیجیتال ......................................................................................................................................................................................................................... 19

* 1. توزیع کلید ............................................................................................................................................................................................................................................ 19

1-4-1 توزیع کلید با استفاده از رمزنگاری متقارن ..................................................................................................................................................................... 20

1-4-2 توزیع کلید با استفاده از رمزنگاری نامتقارن ................................................................................................................................................................... 21

1-4-2-1 گواهی نامه های کلید عمومی ....................................................................................................................................................................................... 22

1-4-2-2 توزیع کلید مخفی با روش کلید عمومی ..................................................................................................................................................................... 23

1-4-3 گواهی نامه های X.509 .................................................................................................................................................................................................... 23

1-4-3-1 ساختار گواهی نامه ها ..................................................................................................................................................................................................... 24

1-4-3-2 به دست آوردن گواهی نامه یک کاربر ......................................................................................................................................................................... 25

1-4-3-3 فسخ گواهی نامه ها ......................................................................................................................................................................................................... 26

1-4-4 زیرساخت کلید عمومی ........................................................................................................................................................................................................ 26

نتیجه گیری ..................................................................................................................................................................................................................................................... 56

پیشنهادات و انتقادات .................................................................................................................................................................................................................................... 56

پیوست 1: واژه نامه ........................................................................................................................................................................................................................................ 58

مراجع ................................................................................................................................................................................................................................................................ 62

فهرست شکل ها

موضوع صفحه

**فصل اول: مفاهیم امنیت**

* 1. مثلث نیازمندی های امنیت ................................................................................................................................................................................................................. 3
  2. مدل ساده شده رمزنگاری متقارن ....................................................................................................................................................................................................... 8
  3. زمان شکستن یک کد با توجه به طول کلید .................................................................................................................................................................................... 9
  4. نمودار رمزگذاری جریانی ................................................................................................................................................................................................................... 10
  5. رمزنویسی کلید عمومی ...................................................................................................................................................................................................................... 12
  6. الگوریتم RSA .................................................................................................................................................................................................................................... 15
  7. اصالت پیام به وسیله کد اصالت پیام ............................................................................................................................................................................................... 16
  8. اصالت پیام به وسیله تابع درهم آمیزی یک طرفه ....................................................................................................................................................................... 17
  9. تابع درهم آمیزی ساده به وسیله عملگر XOR بیتی ................................................................................................................................................................ 19
  10. استفاده از گواهی نامه کلید عمومی ................................................................................................................................................................................................ 22
  11. قالب گواهی نامه X.509 .................................................................................................................................................................................................................. 25

فهرست جدول ها

موضوع صفحه

**فصل اول: مفاهیم امنیت**

* 1. انواع حملات به پیام رمزشده بر اساس میزان اطلاعات تحلیل گر رمز ....................................................................................................................................... 5
  2. زمان میانگین برای جستجو فراگیر کلید .......................................................................................................................................................................................... 5
  3. کاربردهای الگوریتم های مختلف در سیستم های رمزگذاری کلید عمومی ........................................................................................................................... 13

چکیده

ماژول امنیت سخت افزاری کامپیوتری اختصاصی و امن است که برای کاربردهای امنیتی مورد استفاده قرار می گیرد. امروزه نقش این ماژول ها در امنیت شبکه، مدیریت گواهی نامه ها و زیرساخت کلید عمومی انکار ناپذیر است. در این گزارش نخست در فصل اول به معرفی امنیت و مفاهیم آن پرداخته می شود. سپس در فصل دوم، ماژول امنیت سخت افزاری تعریف شده و ویژگی ها، کاربردها و نکات مهم در ارزیابی و خرید آن ها بیان می شود. در پایان این فصل هم چند شرکت فروشنده معرفی شده و محصولاتی از آن ها مورد بررسی قرار می گیرد. همچنین تلاش شده است تا رهنمودهایی برای طراحی و ساخت این دستگاه ها نیز، آورده شود.

فصل اول: مفاهیم امنیت

1-1 تعریف واژگان کلیدی

طبق تعریف موسسه ملی استانداردها و فناوری (NIST)[[1]](#footnote-1) محافظت یک سیستم اطلاعاتی اتوماتیک به منظور حفاظت هرچه بیشتر از یکپارچگی[[2]](#footnote-2)، فراهمی[[3]](#footnote-3) و محرمانگی[[4]](#footnote-4) منابع اطلاعاتی سیستم (شامل سخت افزار، نرم افزار، ثابت افزار[[5]](#footnote-5)، اطلاعات/داده و ارتباط دوربرد) امنیت کامپیوتر نامیده می شود.

این تعریف شامل سه بخش است که قلب امنیت کامپیوتر می باشند:

1- **محرمانگی**: این مورد دو مفهوم مرتبط را پوشش می دهد:

* محرمانگی داده: اطمینان می دهد که اطلاعات شخصی و محرمانه در اختیار اشخاص غیر مجاز قرار نمی گیرند.
* حریم[[6]](#footnote-6): به اشخاص اطمینان می دهد که کنترل کنند که چه اطلاعاتی مربوط به آن ها است که احتمالا جمع آوری و ذخیره می شوند و به وسیله چه کسی و برای چه کسانی این اطلاعات قابل مشاهده اند.

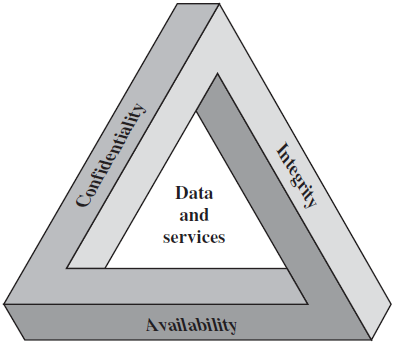
2- **یکپارچگی**: این مورد دو مفهوم مرتبط را پوشش می دهد:

* یکپارچگی داده: اطمینان می دهد که اطلاعات و برنامه ها تنها در یک حالت مشخص شده و مجاز تغییر می کنند.
* یکپارچگی سیستم: اطمینان می دهد که سیستم عملکرد مورد انتظار خودش را در یک حالت صحیح، بدون دستکاری غیر مجاز عمدی یا سهوی، انجام می دهد.

3- **فراهمی**: اطمینان می دهد که سیستم بدون وقفه به کار خود ادامه داده و سرویس دهی به کاربران مجاز به تعویق نمی افتد.

اگرچه مثلث CIA[[7]](#footnote-7) شامل سه مورد امنیتی بالا می باشد، برخی از فعالین در زمینه امنیت، مفاهیم اضافه دیگری را نیز برای امینت متصورند. برای مثال:

**اصلیت**[[8]](#footnote-8): خاصیت اصلی بودن و شناخته شده و مورد اطمینان بودن. اطمینان در صحت یک انتقال، یک پیام و یا صادر کننده یک پیام. به این معنی که بررسی شود که کاربران به همان صورتی که خود را معرفی می کنند باشند و پذیرش هر ورودی دریافت شده به سیستم از یک منبع مورد اطمینان صورت گیرد.



شکل 1-1 مثلث نیازمندی های امنیت [1]

همچنین واژه نامه امنیت اینترنت RFC 2828[[9]](#footnote-9) دو واژه تهدید[[10]](#footnote-10) و حمله[[11]](#footnote-11) را اینگونه تعریف می کند:

**تهدید**: یک پتانسیل برای تجاوز به امنیت، زمانی وجود دارد که یک رویداد، قابلیت، عامل یا حادثه که می تواند به امنیت نفوذ کرده و باعث خسارت شود، وجود داشته باشد. در واقع یک تهدید یک خطر محتمل است که احتمالا از نقطه ضعف بهره برداری می کند.

**حمله**: یک تهاجم به امنیت سیستم که از یک تهدید هوشمند ناشی می شود. این یک رفتار هوشمند و عملی عمدی (مخصوصا با توجه به روش و تکنیک آن) است که به منظور دور زدن سرویس های امنیتی و تجاوز به سیاست امنیتی یک سیستم صورت می گیرد [1].

در طول تاریخ، رمزنگاری با استفاده از روش های تغییر، جابجایی یا اضافه کردن حروف کلمات، برای ارسال پیغام های امن از میان سرزمین های دشمن، مورد استفاده قرار می گرفت. به طور طبیعی زمانی که پیغام به مقصد می رسید، برای خوانا شدن نیاز به رمزگشایی داشت و از همین جا داستان جالب رمزنگاری آغاز می شود. بسیاری از شیوه های استفاده شده در زمان های دور، پایه های امنیت کامپیوتر و شبکه در عصر جدید را تشکیل می دهند.

صدها و شاید هزاران سال پیش پیغام های مهم که از میدان جنگ به پشت جبهه ارسال می شدند به صورت رمزی در می آمدند تا در صورتی که سرباز حامل نامه ها اسیر شود، اطلاعات حساس لو نروند. امروزه رمزنگاری برای مثال در مورد نامه های الکترونیکی مهم انجام می شود تا در صورتی که یک مهاجم به شبکه نفوذ کند، نتواند از محتوای ایمیل ها آگاهی پیدا کند.

به دانش رمزنگاری رمزنویسی[[12]](#footnote-12) گفته می شود که از واژه های یونانی "Kryptos" به معنی پنهان و "Graphia" به معنی نوشتن تشکیل شده است. به فرایند بازکردن (شکستن) یک پیام رمز شده بدون کلید تحلیل رمز[[13]](#footnote-13)، به علم ایجاد کدهای رمزنگاری و شکستن آن ها به طور هم زمان رمزشناسی[[14]](#footnote-14) و به فرایند نوشتن مطلبی به صورت رمز شده به طوری که تنها افراد مجاز قادر به رمزگشایی و خواندن آن باشند، رمزنگاری[[15]](#footnote-15) گفته می شود.

در واقع رمزنگاری هنر نوشتن رمز است به طوری که هیچ کس به جز دریافت کننده مورد نظر نتواند محتوای پیام را بخواند. رمزنگاری همواره جزئی از جنگ، سیاست و حکومت داری بوده است [3].

همچنین پیام ساده اولیه متن ساده[[16]](#footnote-16) و پیام رمز شده متن رمزشده[[17]](#footnote-17) و فرایند عکس رمزنگاری، رمزگشایی[[18]](#footnote-18) نامیده می شوند.

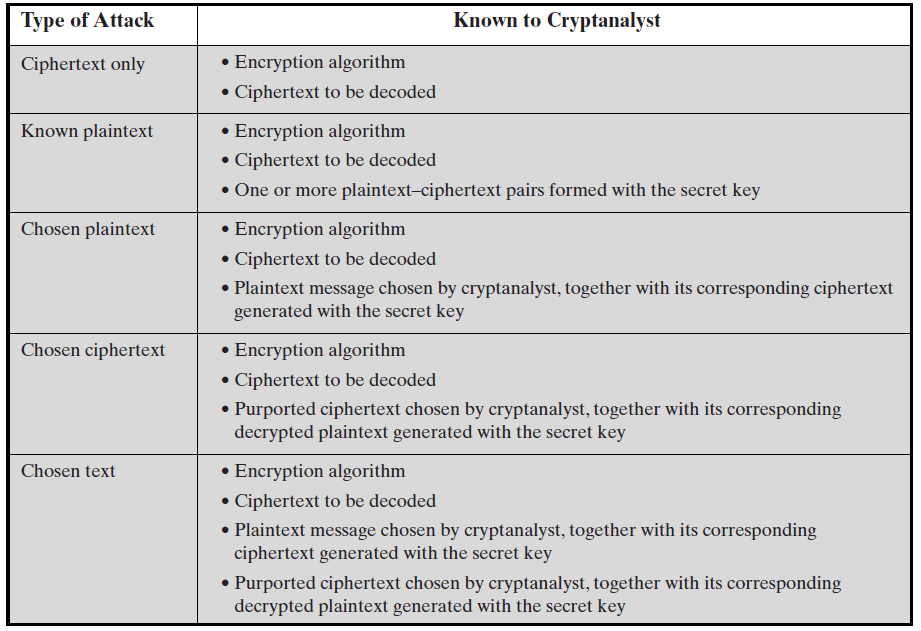
همان طور که گفته شد فرآیند تلاش برای کشف متن ساده یا کلید به عنوان تحلیل رمز شناخته می شود. راهبرد مورد استفاده تحلیل گر رمز[[19]](#footnote-19) به ماهیت طرح رمزنگاری و اطلاعاتی که در اختیار وی می باشد، بستگی دارد [1]. در حالی که رمزنویسی هنر مخفی نگه داشتن پیام است، تحلیل رمز هنر شکستن کدهای رمز[[20]](#footnote-20) و بازیابی متن ساده از متن رمزشده، بدون دانستن کلید مناسب است [2].

تنها الگوریتم های نسبتا ضعیف در برابر حمله تنها متن رمزشده[[21]](#footnote-21) شکست می خورند. در کل یک الگوریتم رمزنگاری برای مقاومت در برابر یک حمله متن ساده آشکار شده[[22]](#footnote-22) طراحی می شود.

یک طرح رمزنگاری[[23]](#footnote-23) به صورت محاسبه شده امن است اگر متن رمزشده تولید شده به وسیله این طرح، یک یا دو شرط زیر را داشته باشد:

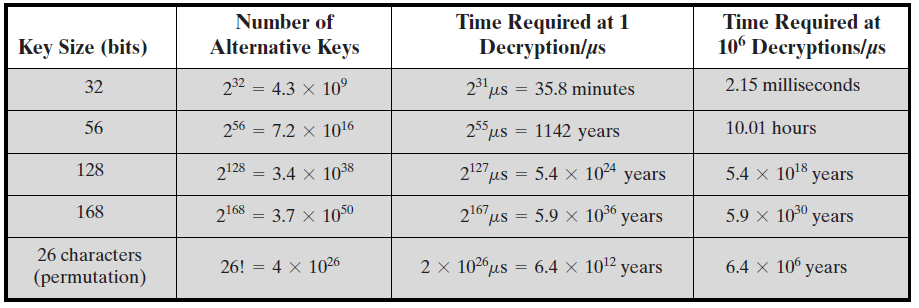
* هزینه شکستن رمز[[24]](#footnote-24) فراتر از ارزش اطلاعات رمزنگاری شده باشد.
* زمان مورد نیاز برای شکستن رمز فراتر از طول عمر مفید اطلاعات باشد.

جدول 1-1 انواع حملات به پیام رمزشده بر اساس میزان اطلاعات تحلیل گر رمز [1]



یک راهبرد وحشی[[25]](#footnote-25) به عملی می گویند که در آن تمام کلید های ممکن مورد آزمایش قرار می گیرد تا یک ترجمه قابل فهم از متن رمزشده به متن ساده، به دست آید. به طور میانگین، برای دستیابی به موفقیت، نیمی از تمام کلیدهای ممکن باید مورد آزمایش قرار بگیرد.

جدول 1-2 زمان میانگین برای جستجو فراگیر کلید [1]



اتحادیه بین المللی ارتباطات (ITU-T)[[26]](#footnote-26) در توصیه نامه X.800، معماری امنیتی برای OSI[[27]](#footnote-27)، یک رهیافت سیستماتیک را معرفی می کند. معماری امنیتی OSI برای مدیران به عنوان یک راه سازماندهی روند فراهم سازی امنیت، مفید است. معماری امنیتی OSI بر روی حملات امنیتی[[28]](#footnote-28)، مکانیزم های امنیتی[[29]](#footnote-29) و سرویس های امنیتی[[30]](#footnote-30) تمرکز دارد. به طور خلاصه می تواند این سه بخش را به صورت زیر تعریف کرد:

* **حمله امنیتی**: هر عملی که امنیت اطلاعات یک سازمان را به خطر بیاندازد. این حملات به دو دسته حملات غیر فعال[[31]](#footnote-31) (هدف دشمن[[32]](#footnote-32) تنها به دست آوردن اطلاعات ارسال شده است) و حمله فعال[[33]](#footnote-33) (هدف دشمن تغیر اطلاعات فرستاده شده است) تقسیم می شوند.
* **مکانیزم امنیتی**: یک فرآیند (یا یک وسیله که شامل یک فرآیند است) که برای تشخیص، جلوگیری یا بازیابی یک حملات امنیتی طراحی شده است.
* **سرویس امنیتی**: یک فرآیند یا سرویس ارتباطی است که امنیت سیستم های پردازش داده و انتقال اطلاعات یک سازمان را بهبود ببخشد. سرویس ها با هدف مقابله با حملات امنیتی ایجاد می شوند و از یک یا چند مکانیزم امنیتی برای فراهم سازی سرویس، بهره می برند [1].

1-2 رمزنگاری

از سه زاویه مستقل می توان رمزنویسی را دسته بندی کرد:

1. نوع عملیات صورت گرفته برای تبدیل متن ساده به متن رمزشده

تمام الگوریتم های رمزنگاری بر دو اصل کلی بنا نهاده شده اند:

* جانشینی، که در آن یک عنصر در متن ساده (بیت، کاراکتر، گروهی از بیت ها یا کارکترها) به یه عنصر دیگر نگاشت داده می شود.
* تبدیل، که در آن یک عنصر در متن ساده بازآراسته می شود.

نیاز اساسی این است که هیچ اطلاعاتی از بین نرود. درواقع تمام علمیات باید معکوس پذیر باشند.

1. تعداد کلیدهای مورد استفاده:

* اگر فرستنده و گیرنده از یک کلید مشابه استفاده کنند، به این سیستم رمزنگاری متقارن[[34]](#footnote-34)، رمزنگاری تک کلیدی[[35]](#footnote-35)، رمزنگاری کلید مخفی[[36]](#footnote-36) و یا رمزنگاری مرسوم[[37]](#footnote-37) می گویند.
* اگر فرستنده و گیرنده هر کدام از یک کلید مختلف استفاده کنند، به این سیستم رمزنگاری نامتقارن[[38]](#footnote-38)، رمزنگاری دو کلیدی[[39]](#footnote-39) یا رمزنگاری کلید عمومی[[40]](#footnote-40) می گویند.

1. روشی که متن ساده پردازش می شود:

* یک پردازش رمزگذاری بلاکی[[41]](#footnote-41) در هر زمان یک بلاک از عناصر را به عنوان ورودی دریافت کرده و به ازای آن یک بلاک خروجی تولید می کند.
* یک پردازش رمزگذاری جریانی[[42]](#footnote-42) عناصر را به صورت پیوسته دریافت می کند و تا هر زمان که به طول انجامد در هر لحظه یک خروجی تولید می کند [1].

1-2-1 رمزنگاری متقارن

رمزنگاری متقارن که از آن به عنوان رمزنگاری مرسوم، رمزنگاری کلید مخفی و یا رمزنگاری تک کلیدی نیز یاد می شود، تنها نوع رمزنگاری بود که پیش از توسعه رمزنگاری کلید عمومی در اواخر دهه 1970، مورد استفاده قرار می گرفت.

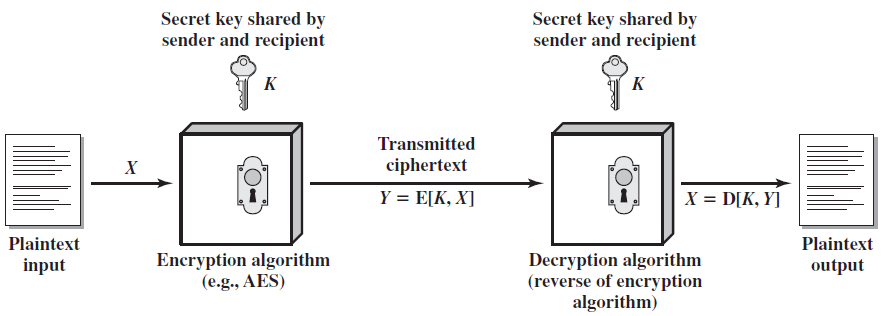
یک رمزنگاری متقارن دارای پنج جز است:

* **متن ساده**: پیام یا داده اصلی می باشد که به عنوان ورودی به الگوریتم داده می شود.
* **الگوریتم رمزنگاری**[[43]](#footnote-43): الگوریتم رمزنگاری جانشینی ها و تبدیلات مختلفی را روی متن انجام می دهد.
* **کلید مخفی**[[44]](#footnote-44): کلید مخفی نیز به عنوان ورودی به الگوریتم داده می شود. جانشینی ها و تبدیلات که به وسیله الگوریتم صورت می گیرد به کلید وابسته است.
* **متن رمزشده**: پیام درهم آمیخته ای است که به عنوان خروجی تولید شده است. محتوای این پیام به متن ساده و کلید بستگی دارد. برای یک متن ساده داده شده، دو کلید مختلف، دو متن رمزشده مختلف تولید می کنند.
* **الگوریتم رمزگشایی**[[45]](#footnote-45): درواقع معکوس الگوریتم رمزنگاری است. این الگوریتم متن رمزشده و کلید را دریافت می کند و متن ساده اصلی را تولید می کند.

برای استفاده امن از رمزنگاری متقارن دو نکته قابل توجه می باشد:

1. ما به یک الگوریتم رمزنگاری قوی نیازمندیم. در واقع ما به الگوریتمی نیاز داریم که اگر دشمن الگوریتم را کشف کرد و به یک یا چند متن رمزشده دست پیدا کرد، همچنان از دستیابی به متن ساده و کلید ناتوان باشد. همچنین اگر او به تعدادی متن رمزشده و متن ساده نظیر آن دست پیدا کرد، باید در دستیابی به کلید ناتوان باشد.
2. فرستنده و گیرنده باید کلید را در حالتی امن به دست آورده و از آن به خوبی محافظت کنند. چرا که اگر کسی کلید را کشف کند و الگوریتم رمزنگاری را بداند، به تمام اطلاعاتی که از این کلید استفاده می کنند، دست یافته است.

این نکته مهمی است که امنیت رمزنگاری متقارن به امنیت کلید وابسته است، نه امنیت الگوریتم. در واقع ما احتیاجی به مخفی نگه داشتن الگوریتم نداریم، بلکه تنها نیازمند مخفی ماندن کلید هستیم. با استفاده از رمزنگاری متقارن، مسئله امنیتی اصلی نگه داری و حفظ امنیت کلید است [1].



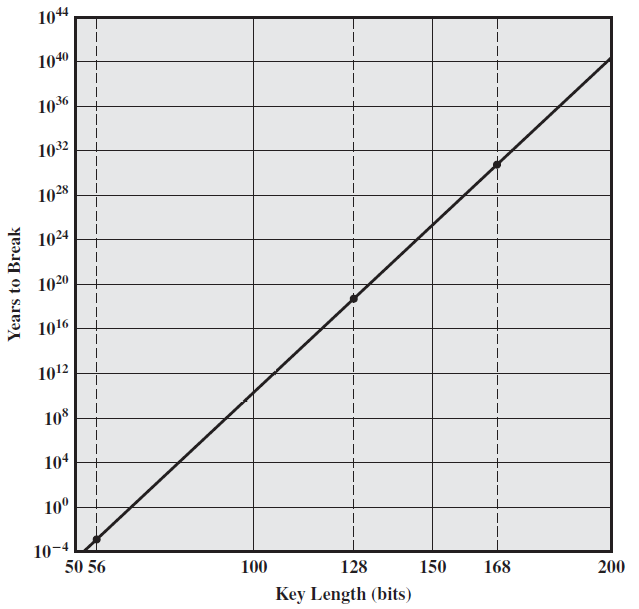
شکل 1-2 مدل ساده شده رمزنگاری متقارن [1]

اگر چه رمزنگاری های متقارن سریع هستند ولی از چند مشکل در محیط ارتباط دیجیتال رنج می برند. این مشکلات نتیجه مستقیم خصوصیات ذاتی رمزنگاری متقارن است. شاید بزرگترین مشکل آن ها اشتراک گذاری یک تک کلید[[46]](#footnote-46) بین فرستنده و گیرنده باشد. علاوه بر این مشکلات دیگری نیز وجود دارد:

* یکپارچگی داده ها می تواند به خطر بیفتد، چرا که گیرنده نمی تواند بررسی کند که پیام قبل از دریافت تغییر کرده است یا نه.
* ممکن است که فرستنده، پیام را انکار کند[[47]](#footnote-47)، زیرا مکانیزمی برای گیرنده وجود ندارد تا اطمینان پیدا کند که پیام به وسیله فرستنده مدعی، فرستاده شده است.
* این روش هیچ راهی برای اطمینان از رازداری[[48]](#footnote-48)، حتی در صورت به خطر اکتشاف افتادن فرایند رمزنگاری، ارائه نمی دهد.
* نمی توان کلید مخفی را به منظور اطمینان از محرمانگی به اندازه کافی، به طور مرتب تغییر داد [2].

1-2-1-1 رمزگذاری بلاکی

معمول ترین استفاده الگوریتم های رمزنگاری متقارن[[49]](#footnote-49)، رمزگذاری بلاکی می باشد. یک پردازش رمزگذاری بلاکی، یک بلاک متن ساده به اندازه ثابت را دریافت کرده و یک بلاک متن رمزشده به همان سایز تولید می کند. DES[[50]](#footnote-50)، 3DES[[51]](#footnote-51)، AES[[52]](#footnote-52) سه الگوریتم مهم رمزگذاری بلاکی متقارن می باشند [1].



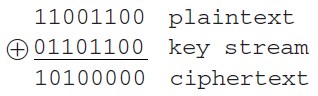
شکل 1-3 زمان شکستن یک کد با توجه به طول کلید (با فرض 106 رمزگشایی در 1 میکرو ثانیه) [1]

1-2-1-2 رمزگذاری جریانی

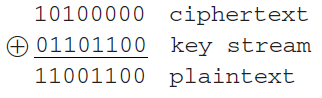
همان طور که گفته شد، یک پردازش رمزگذاری بلاکی در هر زمان یک بلاک را به عنوان ورودی دریافت می کند و یک بلاک خروجی را به ازای آن تولید می نماید. در یک پردازش رمزگذاری جریانی، ورودی پیوسته می باشد و تا زمانی که به طول انجامد، در هر لحظه یک عنصر در خروجی تولید می شود. اگرچه رمزگذاری بلاکی بسیار رایج تر هستند، کاربردهای خاصی وجود دارد که رمزگذاری جریانی برای آن ها مناسب تر است. یکی از محبوب ترین الگوریتم های رمزگذاری جریانی متقارن RC4 است.

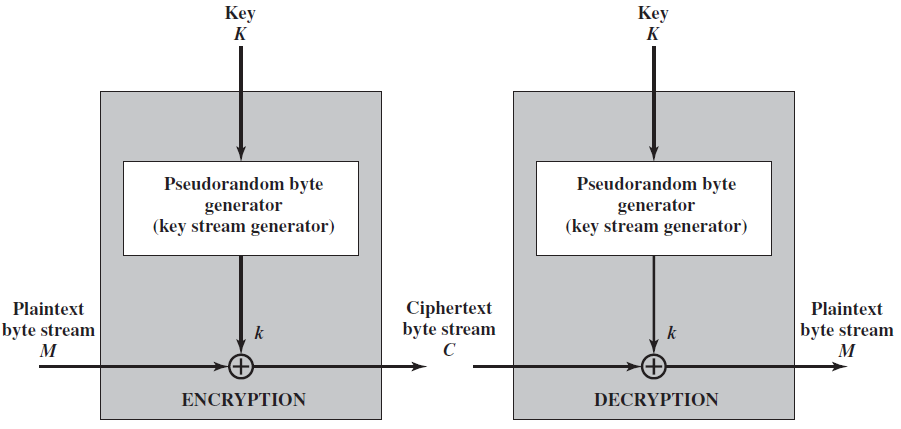
معمولا یک رمزگذاری جریانی در هر لحظه یک بایت از متن ساده را رمزنگاری می کند، گرچه می تواند برای رمزنگاری یک بیت یا واحدهای بزرگ تر از یک بایت هم طراحی شود. در این ساختار یک کلید به عنوان ورودی به یک مولد بیت شبه تصادفی[[53]](#footnote-53) وارد می شود که یک جریان 8 بیتی از اعداد ظاهرا تصادفی تولید می کند. یک جریان شبه تصادفی[[54]](#footnote-54) عددی است که بدون آگاهی از کلید ورودی غیر قابل پیش بینی و ظاهرا تصادفی است. خروجی مولد، جریان کلید[[55]](#footnote-55) نامیده می شود که در هر لحظه ترکیب یک بایت با جریان متن ساده، به وسیله عملگر XOR بیتی است.

برای مثال اگر خروجی مولد 01101100 و بایت بعدی متن ساده برابر 11001100 باشد بایت متن رمزشده برابر است با:



و در رمزگشایی نیازمند استفاده از همان دنباله شبه تصادفی هستیم تا متن ساده به دست آید [1]:





شکل 1-4 نمودار رمزگذاری جریانی [1]

با وجود اینکه رمزگذاری جریانی سریع تر هستند و پیاده سازی ساده تری دارند، ولی دارای شکاف امنیتی مهمی هستند. اگر کلید جریان مشابهی مورد استفاده قرار بگیرد، انواع خاصی از حملات ممکن است سبب افشا اطلاعات بشود. از طرف دیگر رمزگذاری بلاکی یک پیام را به قطعه هایی می شکنند و یک کلید را با هر قطعه ترکیب می کنند. برای مثال قطعه های 64 بیتی یا 128 بیتی. نوین ترین رمزگذاری ها، رمزگذاری بلاکی هستند [2].

1-2-2 رمزنگاری نامتقارن

اهمیت رمزنگاری کلید عمومی به همان اهمیت رمزنگاری مرسوم است که در اصالت پیام[[56]](#footnote-56) و توزیع کلید[[57]](#footnote-57) مورد استفاده قرار می گیرد.

رمزنگاری کلید عمومی که اولین بار توسط Diffie و Hellman در 1976 ارائه شد، اولین پیشرفت انقلاب حقیقی در زمینه رمزنگاری در طول هزاران سال بود. الگوریتم های کلید عمومی به جای عملیات ساده بر روی الگوهای بیتی مورد استفاده در الگوریتم های رمزنگاری متقارن، بر پایه توابع ریاضی استوار است. از آن مهم تر رمزنویسی کلید عمومی، نامتقارن است، یعنی برخلاف رمزنگاری مرسوم متقارن از دو کلید جداگانه استفاده می کند. استفاده از دو کلید تاثیر عمیقی در محرمانگی، توزیع کلید و اصالت[[58]](#footnote-58) دارد.

معمولا چند تصور غلط در رابطه با رمزنگاری کلید عمومی وجود دارد. اول این که امنیت رمزنگاری کلید عمومی بیشتر از رمزنگاری مرسوم است، در حالی که امنیت هر طرح رمزنگاری[[59]](#footnote-59) به (1) طول کلید و (2) کار قابل محاسبه برای شکستن یک رمز بستگی دارد. در واقع هیچ اصلی درباره برتری هیچ یک از رمزنگاری کلید عمومی و رمزنگاری مرسوم بر یکدیگر از دید مقاومت در برابر تحلیل رمز وجود ندارد.

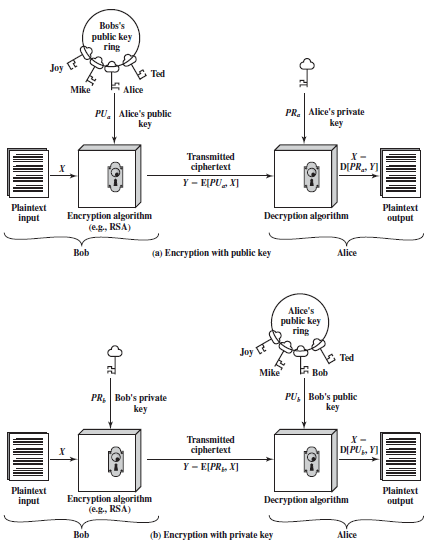
دومین تصور غلط این است که رمزنگاری کلید عمومی یک تکنیک همه منظوره می باشد که باعث از رده خارج شدن رمزنگاری مرسوم شده است. برعکس، به دلیل سربار محاسباتی طرح های رمزنگاری کلید عمومی کنونی، نمی توان آینده قابل پیش بینی ای برای از رده خارج شدن رمزنگاری مرسوم در نظر گرفت [1]. برای مثال سریع ترین الگوریتم رمزنویسی کلید عمومی مثل RSA هنوز بسیار کندتر از الگوریتم های معمولی متقارن است. این مسئله، الگوریتم های کلید عمومی را برای استفاده در مورد پیام های طولانی نامناسب می کند [2].

سوم این که احساسی وجود دارد که توزیع کلید به وسیله رمزنگاری کلید عمومی در مقایسه با مکانیزم دست دهی[[60]](#footnote-60) سنگین درگیر با مراکز توزیع کلید که با رمزنگاری مرسوم پیاده سازی شده است، کاری بی اهمیت است. در واقع فرم هایی از پروتکل مورد نیاز است که اغلب با یک عامل مرکزی درگیرند و رویه های درگیر ساده تر نیستند و در مقایسه با رمزنگاری مرسوم کارایی بیشتری دارند.

یک طرح رمزنگاری کلید عمومی دارای شش بخش است:

* **متن ساده**: یک پیام یا داده قابل خواندن است که به عنوان ورودی به الگوریتم داده می شود.
* **الگوریتم رمزنگاری**: الگوریتم رمزنگاری تبدیلات مختلفی را روی متن ساده صورت می دهد.
* **کلید عمومی و کلید خصوصی**[[61]](#footnote-61): یکی از این دو کلید برای رمزنگاری و دیگری برای آشکارسازی مورد استفاده قرار می گیرند. تبدیل دقیق صورت گرفته به وسیله الگوریتم رمزنگاری به کلید عمومی یا خصوصی مورد استفاده در ورودی بستگی دارد.
* **متن رمزشده**: پیامی مرکب است که به عنوان خروجی ایجاد می شود و به متن ساده و کلید بستگی دارد. برای یک پیام داده شده، دو کلید مختلف، دو متن رمزشده مختلف ایجاد می کنند.
* **الگوریتم رمزگشایی**: این الگوریتم متن رمزشده و کلید مناسب را دریافت می کند و متن ساده اصلی را تولید می نماید.

همان طور که از اسم ها مشخص است کلید عمومی در اختیار همه قرار می گیرد ولی کلید خصوصی تنها برای صاحب آن شناخته شده است. یک الگوریتم رمزنویسی کلید عمومی همه منظوره به یک کلید برای رمزنگاری و کلیدی متفاوت اما وابسته برای رمزگشایی وابسته است [1].



شکل 1-5 رمزنویسی کلید عمومی [1]

تا زمانی که کاربر از کلید خصوصی خود محافظت می کند، ارتباطات امن خواهد بود. در هر زمان کاربر می تواند کلید خصوصی خود را تغییر دهد و کلید عمومی نظیر آن را برای جایگزینی به جای کلید عمومی قدیمی، منتشر کند.

کلید مورد استفاده در رمزنگاری مرسوم، کلید مخفی نامیده می شود. دو کلید به کار رفته در رمزنگاری کلید عمومی به نام کلید عمومی و کلید خصوصی شناخته می شوند. کلید خصوصی نیز مخفی نگه داشته می شود ولی برای برای جلوگیری از تشابه اسمی و اشتباه به جای کلید مخفی، کلید خصوصی نامیده می شود [1].

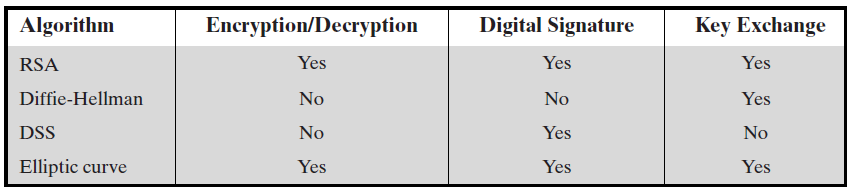
ویژگی رمزنگاری کلید عمومی این است که هیچ کلید مخفی بین دو طرف ارتباط عبور داده نمی شود. مزیت دیگر آن، حل کردن مسئله دیرینه انکار[[62]](#footnote-62) است که در رمزنگاری متقارن وجود داشت. این مسئله به وسیله استفاده از امضاهای دیجیتال[[63]](#footnote-63) و گواهی نامه ها[[64]](#footnote-64) در بسیاری از موارد حل شده است [2].

1-2-2-1 کاربردهای سیستم های رمزگذاری کلید عمومی

سیستم کلید عمومی به وسیله دو کلید که یکی خصوصی نگه داری می شود و دیگری برای همگان آشکار است، کار می کند. بسته به کاربرد، فرستنده ممکن است از کلید خصوصی خودش، کلید عمومی گیرنده و یا هر دو برای رمزنگاری پیام استفاده کند. سیستم های کلید عمومی را می توان به سه دسته تقسیم کرد:

* **رمزنگاری/رمزگشایی**: فرستنده یک پیام را به وسیله کلید عمومی گیرنده رمزنگاری می کند.
* **امضا دیجیتال**: فرستنده یک پیام را به وسیله کلید خصوصی خودش "امضا"[[65]](#footnote-65) می کند. امضا به وسیله اعمال یک الگوریتم رمزنگاری بر روی پیام یا بلاک کوچکی از داده که بیانگر نقش پیام است صورت می گیرد.
* **تبادل کلید**[[66]](#footnote-66): دو طرف به منظور جابه جایی کلید جلسه[[67]](#footnote-67) همکاری می کنند [1].

جدول 1-3 کاربردهای الگوریتم های مختلف در سیستم های رمزگذاری کلید عمومی[[68]](#footnote-68) [1]



1-2-2-2 سرویس های رمزنگاری کلید عمومی

همان طور که گفته شد رمزنگاری کلید عمومی برای حل مشکلات موجود دیگر طرح های رمزنگاری، دست به کار شده و سرویس های زیر را فراهم آورده است:

* **رازداری**: دستیابی به متن رمزشده و استخراج متن ساده نظیر آن را برای متجاوزی که قادر به انجام این کار است، بسیار دشوار می کند.
* **اصلیت**: به گیرنده اجازه می دهد تا منبع پیام را تایید کند.
* **یکپارچگی**: اطمینان می دهد که پیام فرستاده شده به هیچ وجه در طول ارسال نمی تواند تغییر کند.
* **عدم انکار**[[69]](#footnote-69): اطمینان می دهد که فرستنده پیام نمی تواند بعدا رو بگرداند و ارسال پیام را انکار کند [2].

1-2-2-3 الگوریتم کلید عمومی RSA

RSA یکی از اولین طرح های کلید عمومی است که در سال 1977 توسط Ron Rivest، Adi Shamir و Len Adleman در MIT توسعه پیدا کرد و در 1978 به انتشار رسید. RSA از آن زمان تاکنون استفاده گسترده ای داشته است. RSA یک رمزگذار بلاکی است که در آن متن ساده و متن رمزشده به ازای هر n، اعداد صحیحی بین 0 تا n – 1 هستند.

به ازای بلاک متن ساده M و بلاک متن رمزشده C این الگوریتم به صورت زیر عمل می کند:

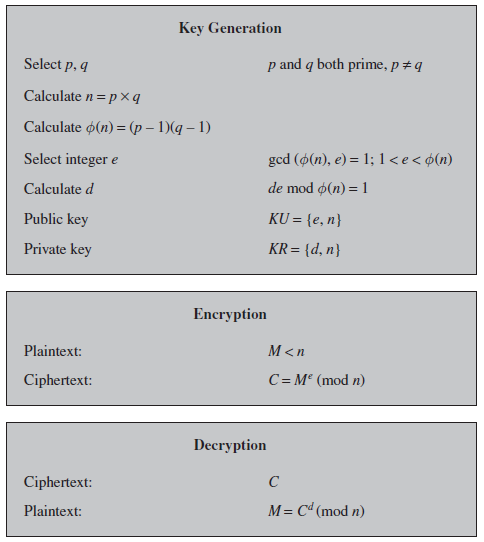
رمزنگاری C = Me mod n

رمزگشایی M = Cd mod n = Med mod n

هم فرستنده و هم گیرنده باید مقادیر n و e را داشته باشند و تنها گیرنده باید مقدار d را بداند. این یک الگوریتم رمزنگاری کلید عمومی با کلید عمومی KU = {e, n} و کلید خصوصی KR = {d, n} می باشد. برای دستیابی به یک نتیجه رضایت بخش به عنوان یک الگوریتم رمزنگاری کلید عمومی، باید شرایط زیر را در نظر گرفت:

1. به ازای هر M < n یافتن e، d و n با شرط Med mod n = M ممکن باشد.
2. به ازای هر M < n محاسبه Me و Cd نسبتا ساده باشد.
3. تشخیص d با داشتن e و n غیر ممکن باشد.

دو شرط اول به آسانی به دست می آیند. شرط سوم نیز با در نظر گرفتن مقادیر بزرگ n و e ممکن می شود [1].



شکل 1-6 الگوریتم RSA [1]

1-3 اصالت پیام

علاوه بر محرمانگی پیام، اصالت پیام[[70]](#footnote-70) نیز یکی از حوزه های مهم امنیت شبکه می باشد. رمزنگاری، اطلاعات را تنها در برابر حمله غیر فعال حفاظت می کند. به همین سبب راهبرد دیگری برای محافظت در برابر حمله فعال مورد نیاز است. حفاظت در برابر این حملات به عنوان اصالت پیام شناخته می شود.

یک پیام، فایل، سند یا هر مجموعه ای از داده معتبر[[71]](#footnote-71) نامیده می شود، اگر اصلی[[72]](#footnote-72) باشد و از یک منبع مورد انتظار[[73]](#footnote-73) به دست آمده باشد. اصالت پیام به فرآیندی گفته می شود که به طرفین ارتباط اجازه می دهد تا پیام دریافت شده را معتبر تلقی کنند. بررسی عدم تغییر پیام رسیده شده و معتبر بودن منبع، دو بعد مهم اصالت پیام می باشد. همه این مسائل در زیرگروه یکپارچگی داده قرار می گیرد.

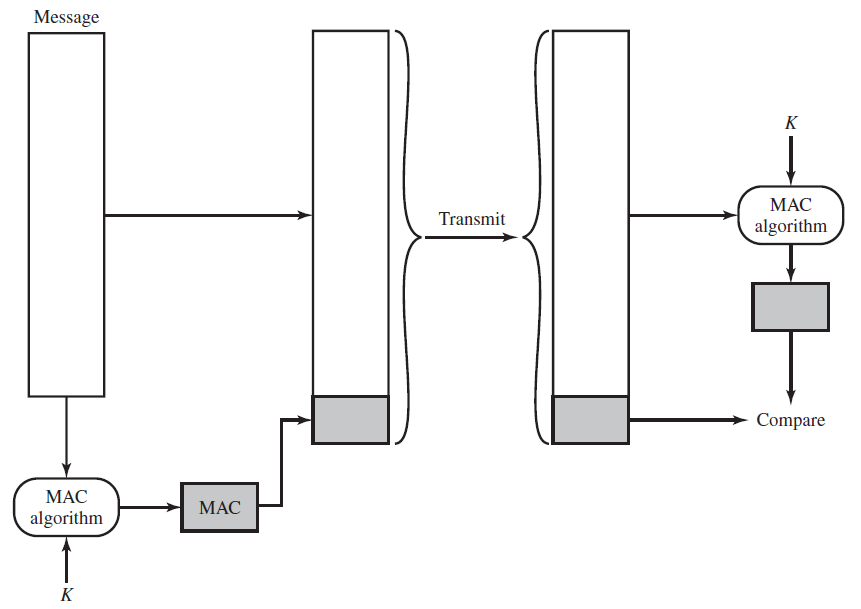
درواقع رمزنگاری متقارن به تنهایی ابزار مناسبی برای اصالت داده نیست. راهبردهایی برای دستیابی به اصالت پیام وجود دارند که به رمزنگاری وابسته نیستند. در همه آن ها، یک برچسب اصالت[[74]](#footnote-74) تولید می شود و به هر پیام پیوست می شود. پیام رمزنگاری نمی شود و مستقل از عملکرد اصالت در مقصد قابل خواندن است.

پیام رمزنگاری شده، به خودی خود دارای اصالت نیست. با این حال ترکیب اصالت و محرمانگی در یک الگوریتم به وسیله رمزنگاری یک پیام و پیوست یک برچسب اصالت به آن امکان پذیر است [1].

1-3-1 کد اصالت پیام

یکی از تکنیک های اصالت که از یک کلید مخفی برای تولید یک بلاک داده کوچک استفاده می کند کد اصالت پیام (MAC)[[75]](#footnote-75) می باشد، که به پیام پیوست می شود. در این تکنیک دو طرف ارتباط با نام های A و B یک کلید مخفی به نام KAB را با هم به اشتراک می گذارند. زمانی که A قصد ارسال یک پیام برای B را دارد، کد اصالت پیام را به وسیله تابعی با ورودی های پیام و کلید محاسبه می کند: MACM = F(KAB,M).

سپس پیام به همراه کد به مقصد مورد نظر فرستاه می شود. در مقصد همان محاسبات به وسیله همان کلید مخفی بر روی پیام دریافتی صورت می گیرد، تا کد اصالت پیام جدید به وجود بیاید. کد دریافت شده با کد محاسبه شده مقایسه می شود. اگر ما مطمئن باشیم که تنها گیرنده و فرستنده از کلید مخفی آگاه اند و اگر کد محاسبه شده و کد دریافت شده برابر باشند، گیرنده می تواند اطمینان حاصل کند که پیام تغییر پیدا نکرده است و پیام از فرستنده مورد انتظار فرستاده شده است [1].



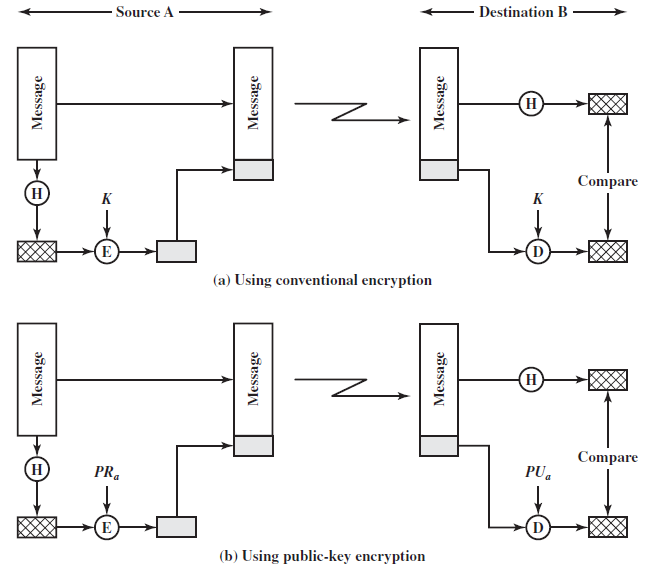
شکل 1-7 اصالت پیام به وسیله کد اصالت پیام [1]

1-3-2 تابع درهم آمیزی یک طرفه

یک راه جایگزین برای کد اصالت پیام، تابع درهم آمیزی یک طرفه[[76]](#footnote-76) است. مشابه کد اصالت پیام یک تابع درهم آمیزی یک پیام با اندازه دلخواه M را را به عنوان ورودی دریافت می کند و یک پیام خلاصه شده با اندازه ثابت H(M) را در خروجی تولید می کند. این پیام خلاصه شده به همراه پیام اصلی فرستاده می شود. برخلاف MAC یک کلید مخفی را به عنوان ورودی دریافت نمی کند. به منظور اصالت بخشی به یک پیام، خلاصه پیام به همراه پیام فرستاده می شود.

این پیام خلاصه شده می تواند به وسیله رمزنگاری مرسوم (قسمت a) رمزنگاری بشود. اگراطمینان حاصل بشود که تنها فرستنده و گیرنده به کلید دسترسی دارند، اصلیت تضمین می شود. پیام خلاصه شده را می توان با استفاده از راهبرد کلید عمومی (قسمت b) رمزنگاری کرد. این راهبرد دو مزیت دارد: (1) به همراه اصالت پیام امضا دیجیتال را هم فراهم می آورد. (2) دیگر نیازی به توزیع کلیدها بین طرفین ارتباط نیست.

تابع درهم آمیزی یک طرفه یا تابع درهم آمیزی امن[[77]](#footnote-77) نه تنها در اصالت پیام بلکه در امضا دیجیتال نقش مهمی دارد. یکی از مهمترین الگوریتم تابع درهم آمیزی، الگوریتم SHA است [1].



شکل 1-8 اصالت پیام به وسیله تابع درهم آمیزی یک طرفه [1]

1-3-2-1 نیازمندی های تابع درهم آمیزی

هدف تابع درهم آمیزی ایجاد یک "اثر انگشت"[[78]](#footnote-78) برای یک فایل، پیام یا هر بلاک داده دیگری است. برای مفید بودن آن در تامین اصالت پیام، یک تابع درهم آمیزی H باید خصوصیات زیر را داشته باشد:

1. H باید توانایی دریافت یک بلاک داده به هر اندازه ای را داشته باشد.
2. H باید خروجی با اندازه ثابتی داشته باشد.
3. H(x) باید نسبتا محاسبه آسانی برای هر x ورودی داشته باشد.
4. برای هر کد h داده شده، یافتن یک x با شرط H(x) = h، غیر ممکن باشد. (خاصیت یک طرفه یا پایداری پریمیج[[79]](#footnote-79))
5. برای هر بلاک x داده شده، یافتن یک y با شرطy ≠ x و H(y) = H(x) غیر ممکن باشد. (خاصیت پایداری پریمیج دوم[[80]](#footnote-80) یا پایداری برخورد ضعیف[[81]](#footnote-81))
6. از نظر محاسباتی یافتن یک زوج (x, y) با شرط H(x) = H(y) غیر ممکن باشد. (خاصیت پایداری برخورد[[82]](#footnote-82) یا پایداری برخورد قوی[[83]](#footnote-83))

خلاصه پیام علاوه بر اصالت، یکپارچگی داده را نیز فراهم می سازد. در واقع نقش دنباله وارسی قاب[[84]](#footnote-84) را نیز ایفا می کند، به این صورت که اگر بیتی در پیام تصادفا در انتقال تغییر پیدا کرد، خلاصه پیام بیانگر خطا خواهد بود [1].

1-3-2-2 توابع درهم آمیزی ساده

تمامی توابع درهم آمیزی یک روند کلی دارند. ورودی (پیام، فایل و ...) به صورت دنباله ای از بلاک های n بیتی دریافت می شود. ورودی به صورت تکراری به پردازش یک بلاک در هر زمان می پردازد تا یک بلاک n بیتی را که خروجی تابع درهم آمیزی می باشد را، تولید نماید. یکی از ساده ترین توابع درهم آمیزی، XOR بیت به بیت تمام بلاک ها با هم است [1]:

Ci = bi1  bi2  ...  bim

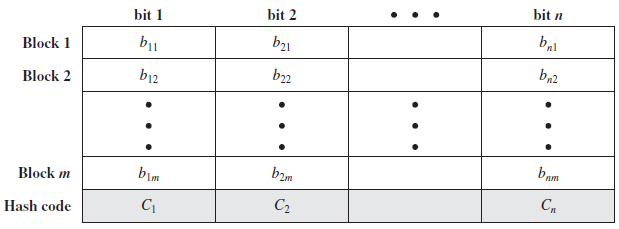
که در آن

Ci = i امین بیت کد درهم آمیزی، 

m = تعداد بلاک های n بیتی در ورودی

bij = i امین بیت در j امین بلاک

 = عملگر XOR



شکل 1-9 یک تابع درهم آمیزی ساده به وسیله عملگر XOR بیتی [1]

1-3-3 امضا دیجیتال

می توان از رمزنگاری کلید عمومی به شکلی دیگر استفاده کرد. فرض کنید فرستنده می خواهد پیامی برای گیرنده بفرسند، در حالی که اهمیتی ندارد که پیام مخفی بماند. بلکه می خواهد گیرنده اطمینان پیدا کند که پیام به وسیله فرستنده مورد نظر فرستاده شده است. در این مورد، فرستنده از کلید خصوصی خودش برای رمزنگاری پیام استفاده می کند. زمانی که گیرنده متن رمزشده را دریافت می کند، در می یابد که می تواند با کلید عمومی فرستنده، آن را بازیابی کند. پس اطمینان پیدا می کند که پیام به وسیله فرستنده مورد نظر رمزنگاری شده است. هیچ کس کلید خصوصی فرستنده را ندارد، پس هیچ کسی نمی تواند یک متن رمزشده ایجاد کند که با کلید عمومی فرستنده بازیابی شود. این پیام رمزنگاری شده از امضا دیجیتال بهره می برد. همچنین تغییر این پیام بدون داشتن کلید خصوصی فرستنده غیر ممکن است. پس این پیام با اصالت[[85]](#footnote-85) است.

این نکته قابل توجه است که این فرآیند رمزنگاری تنها پیام را از جهت تغیر در امان نگه می دارد و محرمانگی را فراهم نمی کند، چرا که هر بیننده ای به سادگی می تواند به وسیله کلید عمومی فرستنده به محتوا پیام دست یابد [1].

1-4 توزیع کلید

مقاومت یک الگوریتم رمزنگاری به تکنیک های توزیع کلید آن بستگی دارد. تکنیک های توزیع توزیع کلید ضعیف، یک محیط ایده آل را برای برخی از حملات ایجاد می کنند. بنابراین، مسئله توزیع کلید نیاز به تکنیک های توزیع کلید قوی را برجسته می کند. حتی با این که مسئله تبادل کلید در روش های رمزنگاری متقارن مشخص تر است و به طور اساسی با روش های رمزنویسی کلید عمومی بر طرف شده است، بعضی از مسائل تبادل کلید همچنان در روش های رمزنویسی کلید عمومی نیز وجود دارند. برای مثال، رمزنگاری متقارن کلید نیازمند این است که دو طرف ارتباطی بر روی کلید مخفی شان پیش از زمان برقراری ارتباط توافق داشته باشند و رمزنگاری کلید عمومی از دشواری دستیابی امن به کلید عمومی گیرنده رنج می برد. با این حال، هر دو این مسائل با استفاده از یک شخص سوم مورد اعتماد یا یک میانجی قابل حل است. برای رمزنگاری متقارن، میانجی مورد اعتماد مرکز توزیع کلید (KDC) نامیده می شود. در رمزنگاری کلید عمومی، میانجی مورد اعتماد و مقیاس پذیر، مرجع صلاحیت دار گواهی نامه (CA) نامیده می شود [2].

1-4-1 توزیع کلید با استفاده از رمزنگاری متقارن

یک مرکز توزیع کلید (KDC)[[86]](#footnote-86) تنها موجودیت مورد اعتماد شبکه است که باید با همه عناصر ارتباطی شبکه یک کلید مخفی را به اشتراک بگذارد. این نیازمند این است که همه عناصر ارتباطی، یک کلید مخفی را به اشتراک گذاشته و با آن به طور محرمانه با KDC ارتباط برقرار کنند. با این حال همچنان مشکل توزیع این کلید اشتراکی وجود دارد. KDC کلیدها را برای عناصر ارتباطی ایجاد نمی کند، بلکه تنها کلیدهای توزیع شده را ذخیره می کند. تولید کلیدها باید در جای دیگری صورت بگیرد [2].

برای استفاده از یک رمزنگاری متقارن در توزیع کلید، طرفین ارتباط باید یک کلید را با هم به اشتراک بگذارند و آن کلید باید از دسترس دیگران محافظت شود. همچنین معمولا تغییر مکرر کلید برای کاهش میزان به خطر افتادن داده ها در صورت افشا کلید، مناسب می باشد. بنابراین مقاومت هر سیستم رمزنویسی به تکنیک توزیع کلید وابسته است. توزیع کلید واژه ای است که به روش تحویل کلید به دو طرفی که قصد تبادل داده را با هم دارند گفته می شود، بی آنکه دیگران از کلید آگاهی یابند.

توزیع کلید توسط روش های زیادی قابل اجراست. برای دو طرف A و B گزینه های زیر موجود است:

1. یک کلید می تواند توسط A انتخاب شود و به صورت فیزیکی به B فرستاده شود.
2. یک طرف سوم می تواند کلید را انتخاب کند و به صورت فیزیکی آن را به A و B بفرستند.
3. اگر A و B قبلا با هم ارتباط داشته و از یک کلید استفاده می کردند، یک طرف می تواند به وسیله این کلید قدیمی کلید جدید را رمزنگاری کرده و به طرف دیگر بفرستند.
4. اگر A و B یک ارتباط رمزنگاری شده با یک طرف سومی به نام C داشته باشند، C می تواند کلید را از طریق لینک های رمزنگاری برای A و B بفرستد.

برای گزینه 4 دو نوع کلید مورد استفاده قرار می گیرد:

* کلید جلسه: زمانی که دو سیستم انتهایی[[87]](#footnote-87) (هاست، ترمینال و ...) قصد ارتباط با هم را دارند، یک ارتباط منطقی (برای مثال مدار مجازی) با هم برقرار می کنند. در طول این ارتباط منطقی که یک جلسه نامیده می شود، همه داده ها به وسیله یک کلید جلسه یک بار مصرف[[88]](#footnote-88) رمزنگاری می شوند. در پایان جلسه کلید جلسه نابود می شود.
* کلید دائمی[[89]](#footnote-89): یک کلید دائمی کلیدی است که بین دو موجودی، با هدف توزیع کلیدهای جلسه، مورد استفاده قرار می گیرد.

عنصر لازم برای گزینه 4 یک مرکز توزیع کلید (KDC) می باشد. KDC مشخص می کند که کدام سیستم ها اجازه دارند با دیگر سیستم ها ارتباط برقرار کنند. زمانی که اجازه ایجاد یک ارتباط بین دو سیستم داده شد، مرکز توزیع کلید یک کلید جلسه یک بار مصرف را برای آن ارتباط فراهم می آورد.

در کل، عملکرد KDC به شکل زیر است:

1. زمانی که هاست A قصد ارتباط با هاست B را دارد، یک بسته درخواست ارتباط را به KDC ارسال می کند. ارتباط بین A و KDC به وسیله یک شاه کلید[[90]](#footnote-90) که تنها بین A و KDC به اشتراک گذاشته شده است، رمزنگاری می شود.
2. اگر KDC با درخواست ارتباط موافقت کند، یک کلید جلسه یک بار مصرف یگانه را ایجاد می کند. این کلید جلسه به وسیله کلید دائمی که با A به اشتراک گذاشته شده است، رمزنگاری شده و به A ارسال می شود. به طور مشابه این کلید جلسه به وسیله یک کلید دائمی که با B به اشتراک گذاشته شده است، رمزنگاری شده و به B ارسال می شود.
3. حال A و B می توانند یک ارتباط منطقی با هم برقرار کرده و به تبادل پیام و داده بپردازند. تمام این ارتباط با این کلید جلسه موقت رمزنگاری می شود [1].

معایب KDC عبارت اند از:

* دو عنصر ارتباطی شبکه باید به یک KDC تعلق داشته باشند.
* امنیت به یک مسئله تبدیل می شود، زیرا از آن جا که یک مرجع صلاحیت دار مرکزی[[91]](#footnote-91) به کلیدها دسترسی دارد در معرض نفوذ قرار می گیرد. به دلیل اهمیت زیاد آن، یک نفوذ امنیتی به KDC می تواند کل سیستم را به خطر بیندازد.
* در شبکه های بزرگ که با همه توپولوژی های ارتباطی سر و کار دارد، KDC تبدیل به یک تنگنا[[92]](#footnote-92) می شود که هر دو کاربر نیازمند یک کلید باید حداقل یک بار به گره مرکزی دسترسی پیدا کنند. همچنین شکست مرجع صلاحیت دار می تواند سبب از هم گسیختن سیستم توزیع کلید بشود [2].

1-4-2 توزیع کلید با استفاده از رمزنگاری نامتقارن

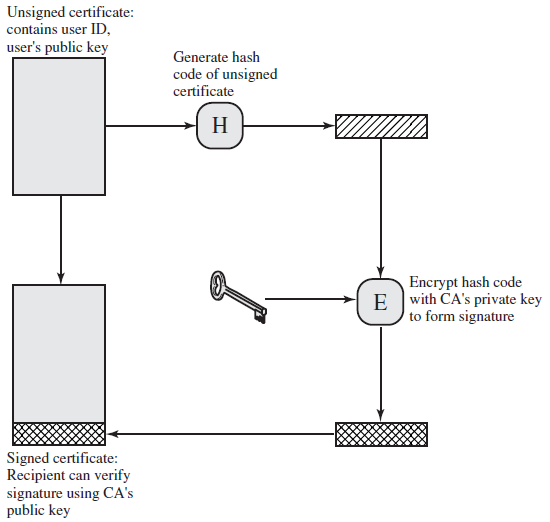
یکی از نقش های اساسی رمزنگاری کلید عمومی حل مسئله توزیع کلید است. درواقع دو رویکرد مجزا از رمزنگاری کلید عمومی در این خصوص وجود دارد [1]:

* توزیع کلیدهای عمومی
* استفاده از رمزنگاری کلید عمومی برای توزیع کلیدهای مخفی

1-4-2-1 گواهی نامه های کلید عمومی

همان طور که گفته شد، نکته اساسی در رمزنگاری کلید عمومی این است که کلید عمومی، عمومی است. بنابراین اگر یک الگوریتم به طور وسیع پذیرفته شده مثل RSA وجود داشته باشد، هر شرکت کننده می تواند کلید عمومی خودش را به دیگر شرکت کنندگان بفرستد و یا کلید را به طور وسیع در اجتماع منتشر کند. گرچه این راهبرد سودمند است، یک ضعف اساسی دارد. هر کسی قادر به جعل این آگهی است. هر کسی می تواند ادعا کند که کاربر A است و یک کلید عمومی به دیگر شرکت کنندگان بفرستند و یا آن را منتشر کند. تا آن زمان که کاربر A متوجه شود و به دیگر شرکت کنندگان هشدار بدهد، جعل کننده قادر به خواندن همه پیام های رمزنگاری شده برای A می باشد و می تواند از کلیدهای جعلی برای اصالت دادن استفاده کند.

راه حل این مشکل استفاده از گواهی نامه کلید عمومی است. ضرورتا هر گواهی نامه از یک کلید عمومی به اضافه شناسه[[93]](#footnote-93) کاربر صاحب کلید تشکیل شده است و کل آن دارای امضای یک شخص سوم مورد اعتبار می باشد. شخص سوم یک مرجع صلاحیت دار گواهی نامه (CA)[[94]](#footnote-94) می باشد که مثل یک آژانس دولتی یا یک موسسه مالی مورد اعتماد انجمن کاربران است. یک کاربر می تواند کلید عمومی خود را در یک حالت امن به مرجع صلاحیت دار ارائه کند تا یک گواهی نامه دریافت کند. سپس کاربر می تواند گواهی نامه را منتشر نماید. هر کسی که به کلید عمومی این کاربر نیاز دارد، می تواند گواهی نامه او را به دست آورد و به وسیله امضا مورد اعتماد پیوست شده به آن، اعتبار آن را بررسی کند [1].



شکل 1-10 استفاده از گواهی نامه کلید عمومی [1]

درواقع مقاومت الگوریتم به تکنیک های توزیع کلید وابسته است. مسئله تبادل کلید، نیاز به تکنیک های توزیع کلید قوی را به خوبی نشان می دهد. حتی با وجود این که مسئله تبادل کلید توسط روش های رمزنویسی کلید عمومی بسیار بهبود یافته است و به طور اساسی حل شده است، بعضی از مشکلات تبادل کلید هنوز در روش های رمزنویسی کلید عمومی نیز همچنان وجود دارد. برای مثال رمزنگاری کلید متقارن نیازمند این است که دو طرف ارتباط بر روی کلید مخفی شان قبل از شروع ارتباط توافق داشته باشند، و رمزنگاری کلید عمومی از دشواری دستیابی به کلید عمومی گیرنده به صورت امن رنج می برد. با این حال هر دو این مشکلات با استفاده از یک شخص سوم مورد اعتماد یا یک واسطه به نام مرجع صلاحیت دار گواهی نامه (CA) قابل حل است [2].

طرحی که عموما برای قالب بندی گواهی نامه کلید عمومی مورد پذیرش قرار گرفته است استاندارد X.509 می باشد [1].

1-4-2-2 توزیع کلید مخفی با روش کلید عمومی

برای استفاده از رمزنگاری مرسوم یک نیاز اساسی برای طرفین ارتباط، به اشتراک گذاشتن امن یک کلید مخفی است. یک راه قدرتمند برای این کار استفاده از گواهی نامه های کلید عمومی[[95]](#footnote-95) است. زمانی که A می خواهد با B ارتباط برقرار کند، A باید این مراحل را انجام دهد:

1. یک پیام را آماده کند.
2. آن پیام را با استفاده از رمزنگاری مرسوم توسط یک کلید جلسه مرسوم یک بار مصرف رمزنگاری کند.
3. کلید جلسه را با استفاده از رمزنگاری کلید عمومی، توسط کلید عمومی B رمزنگاری کند.
4. کلید جلسه رمزنگاری شده را به پیام پیوست و آن را به B ارسال کند.

تنها B می تواند کلید جلسه را رمزگشایی و بنابراین پیام اصلی را بازیابی کند. اگر A کلید عمومی B را به وسیله گواهی نامه کلید عمومی B به دست آورد، پس A اطمینان حاصل می کند که این کلید معتبر است [1].

1-4-3 گواهی نامه هایX.509

توصیه نامه X.509 اتحادیه بین المللی ارتباطات که بخشی از سری توصیه نامه های X.500 است یک سرویس پوشه[[96]](#footnote-96) را تعریف می کند. پوشه تحت یک سرور یا مجموعه سرورهای توزیع شده است که یک دیتابیس از اطلاعات مربوط به کاربران را نگه داری می کند. این اطلاعات شامل یک نگاشت از نام کاربری[[97]](#footnote-97) به آدرس شبکه، همچنین ویژگی های دیگر و اطلاعات مربوط به کاربران است.

X.509 یک چارچوب[[98]](#footnote-98) را برای فراهم سازی سرویس های اصالت به وسیله پوشهX.500 برای کاربرانش را معرفی می کند. دایکتوری همچنین ممکن است به عنوان یک انبار گواهی نامه های کلید عمومی به کار برود. هر گواهی نامه شامل کلید عمومی یک کاربر است که به وسیله کلید خصوصی یک مرجع صلاحیت دار گواهی[[99]](#footnote-99) مورد اعتماد به امضا رسیده است. به اضافه، X.509 پروتکل های اصالت پیشنهادی دیگری را بر پایه استفاده از گواهی نامه های کلید عمومی تعریف می کند.

X.509 بر پایه استفاده از رمزنویسی کلید عمومی و امضاهای دیجیتال می باشد. این استاندارد استفاده از الگوریتم خاصی را تحمیل نمی کند ولی RSA را پیشنهاد می کند. طرح امضا دیجیتال نیز استفاده از یک تابع درهم آمیزی را ملزم می نماید. بار دیگر استاندارد استفاده از الگوریتم درهم آمیزی خاصی را تحمیل نمی کند [1].

1-4-3-1 ساختارگواهی نامه ها

قلب طرحX.509 ، گواهی نامه های کلید عمومی است که با هر کاربر مرتبط می باشند. این گواهی نامه های کاربری به وسیله بعضی از مراجع صلاحیت دار گواهی (CA) به وجود می آیند و توسط CA یا کاربر در پوشه جای داده می شوند. سرور پوشه به خودی خود مسئول ایجاد کلیدهای عمومی یا عمل گواهی نیست. بلکه تنها یک مکان دستیابی آسان را برای دستیابی کاربران به گواهی نامه ها فراهم می آورد.

استفاده استاندارد نشان گذاری زیر، یک گواهی نامه را تعریف می کند:

CA<<A>> = CA {V, SN, AI, CA, UCA, A,UA,Ap,TA}

که در آن

Y<<X>> = گواهی نامه کاربر X توسط مرجع صلاحیت دار گواهی Yصادر شده است

Y {I} = امضا I توسط Y، شامل I به اضافه یک کد درهم آمیزی رمزنگاری شده که به آن پیوست شده است

V = نگارش[[100]](#footnote-100) گواهی نامه

SN = شماره سریال گواهی نامه

AI = شناسه الگوریتم استفاده شده برای امضا گواهی نامه

CA = نام مرجع صلاحیت دار گواهی نامه

UCA = شناسه یکتا اختیاری CA

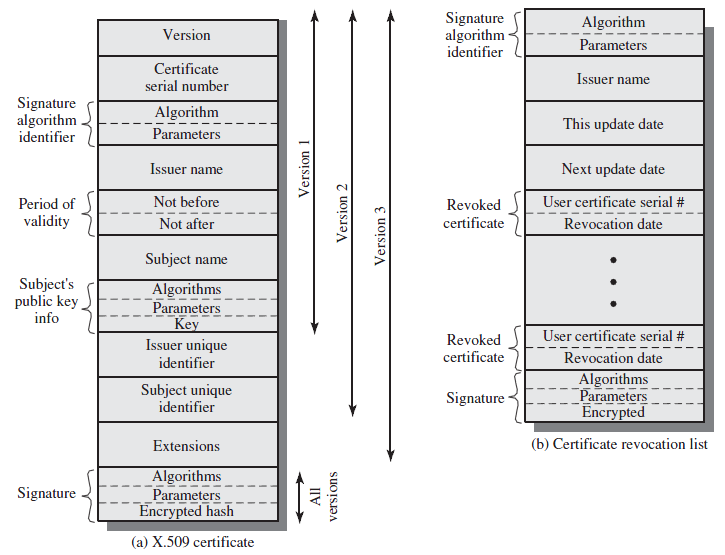
A = نام کاربر A

UA = شناسه یکتا اختیاری کاربر A

Ap = کلید عمومی کاربر A

TA = دوره اعتبار گواهی نامه

CA گواهی نامه را به وسیله کلید خصوصی خودش امضا می کند. اگر کلید عمومی متناظر آن برای یک کاربر شناخته شده باشد، آن کاربر می تواند بررسی کند که آن گواهی نامه توسط CA امضا شده و معتبر است [1].



شکل 1-11 قالب X.509 [1]

1-4-3-2 به دست آوردن گواهی نامه یک کاربر

گواهی نامه کاربر که توسط CA ایجاد می شود دارای خصوصیات زیر است:

* هر کاربر با دسترسی به کلید عمومی CA می تواند کلید عمومی کاربر را بررسی کند که تایید شده باشد.
* هیچ کسی به جز مرجع صلاحیت دار گواهی نمی تواند گواهی نامه را بدون این که مشخص شود، اصلاح کند.

از آن جا که گواهی نامه ها غیر قابل جعل است، می توان آن ها را در پوشه قرار داد بی آن که نیاز به اقدامات خاصی برای محافظت از آن ها وجود داشته باشد.

اگر همه کاربران عضو یک CA باشند، یک اعتماد همگانی به آن CA وجود دارد. اما در جمع بزرگی از کاربران، عضویت همه کاربران در یک CA غیر عملی است. از آن جا که CA گواهی نامه ها را امضا می کند، هر کاربر عضو باید یک کپی از کلید عمومی، CA خود داشته باشد تا بتواند امضاها را بررسی کند.

حال فرض کنید A یک گواهی نامه از مرجع صلاحیت دار گواهی X1 و B یک گواهی نامه از مرجع صلاحیت دار گواهی X2 دریافت کرده باشند. اگر A به طور امن از کلید عمومی X2 آگاهی نیابد، گواهی نامه B که توسط X2 صادر شده است، برای A بی استفاده می شود. A می تواند گواهی نامه B را بخواند، اما نمی تواند امضا آن را بررسی کند. با این حال اگر دو CA به صورت امن کلید عمومی شان را با هم تبادل کنند، روند زیر به A اجازه می دهد تا کلید عمومی B را به دست آورد:

1. A از پوشه به گواهی نامه X2 که توسط X1 امضا شده است دست پیدا می کند. از آن جا که A به طور امن از کلید عمومی X1 آگاهی دارد، A می تواند کلید عمومی X2 را از گواهی نامه اش به دست آورده و با استفاده از امضای X1 در گواهی نامه آن را بررسی کند.
2. سپس A به پوشه بر می گردد و گواهی نامه B را که توسط X2 امضا شده است را به دست می آورد. از آن جا که A اینک یک کپی مورد اطمینان از کلید عمومی X2 دارد، A می تواند امضا را بررسی کند و به طور امن به کلید عمومی B دست یابد.

A از یک زنجیره از گواهی نامه ها برای دستیابی به کلید عمومی B استفاده می کند. در نماد گذاری X.509 این زنجیره به شکل زیر نشان داده می شود:

X1<<X2>> X2<<B>>

به شکل مشابه B می تواند در جهت برعکس به کلید عمومی A دست پیدا کند:

X2<<X1>> X1<<A>>

این طرح به یک زنجیر با دو گواهی نامه محدود نیست. مسیری به طول دلخواه از CA ها می تواند در یک زنجیر قرار بگیرد. یک زنجیر با N عنصر به شکل زیر نمایش داده می شود [1]:

X1<<X2>> X2<<X3>> . . . XN<<B>>

1-4-3-3 فسخ گواهی نامه ها

هر گواهی نامه مثل یک کارت اعتباری دارای یک دوره اعتبار است. معمولا یک گواهی نامه جدید پیش از انقضا گواهی نامه قبلی، صادر می شود. همچنین گاهی مناسب است یک گواهی نامه به دلایل زیر پیش از اتمام دوره اعتبار آن، فسخ شود [1]:

1. کلید خصوصی یک کاربر به خطر افتاده باشد.
2. کاربر دیگر توسط CA تایید نشود.
3. گواهی نامه CA به خطر افتاده باشد.

1-4-4 زیر ساخت کلید عمومی

همان طور که گفته شد در شبکه های بزرگ با توپولوژی های ارتباطی متفاوت که عناصر ارتباطی شبکه نمی توانند همگی به یک KDC تعلق داشته باشند، توزیع کلید به یک مسئله واقعی تبدیل می شود. این مشکلات زمانی برطرف می شوند که از یک زیرساخت کلید عمومی (PKI)[[101]](#footnote-101) به جای KDC به منظور فراهم سازی و مدیریت گواهی نامه ها مورد اعتبار و کارا استفاده شود [2].

واژه نامه امنیت اینترنت RFC 2822 زیرساخت کلید عمومی را به عنوان مجموعه سخت افزار، نرم افزار، مردم، سیاست ها و روش هایی است که برای ایجاد، مدیریت، ذخیره، توزیع و فسخ گواهی نامه های دیجیتال بر پایه رمزنویسی نامتقارن مورد نیاز است، تعریف می کند. هدف اصلی توسعه یک PKI فراهم سازی امنیت، راحتی و دستیابی کارا به کلیدهای عمومی است [1].

PKI همه فعالیت ها را اتوماتیک می کند. PKI زمانی که حجم عظیمی از کاربران وجود داشته باشد به بهترین شکل کار می کند. تحت این شرایط، PKI گواهی نامه ها را ایجاد و به طور گسترده بین کاربران بسیاری توزیع می کند [2].

نتیجه گیری

ماژول امنیت سخت افزاری کامپیوتری اختصاصی است که برای شرکت در کاربردهای امنیتی طراحی شده است. این ماژول ها الگوریتم های رمزنگاری را با استفاده از سخت افزار اختصاصی طراحی شده در داخل آن ها، با سرعت بیشتری انجام می دهند. همچنین کلیدهای خصوصی گواهی نامه های زیرساخت کلید عمومی را نگه داری و بسیاری از نقش های امنیتی مثل امضا دیجیتال، صدور گواهی نامه و ... را ایفا می کنند. از سوی دیگر این امکانات را در قالب یک بسته بندی با پایداری دست اندازی فراهم می آورند، به این معنی که در صورت دسترسی و نفوذ فیزیکی به دستگاه، همه اطلاعات داخل آن پاک می شود.

امروزه استفاده از این ماژول ها در مراجع صلاحیت دار گواهی نامه الزامی است. از سوی دیگر، شرکت های گوناگون و محصولات بسیاری در بازارهای جهانی وجود دارند. خود به خود این سوال مهم پیش می آید که "کدام دستگاه برای نیازهای من مناسب تر است"؟ به همین منظور این گزارش ضمن معرفی کامل ماژول امنیت سخت افزاری، بیان ویژگی ها و کاربردها آن، به پاسخ به این سوال مهم پرداخته و برخی از شرکت ها و محصولات مهم کنونی را معرفی نمود.

همچنین تلاش شد تا این گزارش در کنار بیان مسائل بالا، در طراحی و ساخت این ماژول ها نیز راهگشا باشد. چرا که پس از فاز شناخت و آشنایی با این دستگاه ها، شایسته است که وارد فاز طراحی و ساخت آن ها بشویم.

پیشنهادات و انتقادات

مهم ترین پیشنهادی که من برای هر کارشناس و دانش پژوه علاقه مند در پایان این گزارش دارم، شروع تلاش برای ساخت این ماژول ها است. این سخن نه به معنی بسیار ساده بودن این کار و نه شعاری احساسی است. بلکه پیشنهاد دانشجویی است که 3 ماه به صورت مداوم بر روی موضوع ماژول امنیت سخت افزاری و گرداوری این گزارش تلاش داشته است. امروز بر این باورم که دسترسی به این مهم و ساخت این ماژول ها به هیچ وجه دور از دسترس نیست. گرچه شاید کمی زمان بر باشد، ولی باورم این است که با تشکیل تیمی قوی از دانشجویان و استادان علاقه مند به این موضوع، این فرآیند بسیار سریع تر از آن چه تصور می شود صورت می گیرد.

از نظر تکنیکی هم پیشنهاد من این است که ساخت این دستگاه در چندین فاز انجام شود. به این معنی که برای مثال در یک فاز تنها مدار لازم برای اجرای الگوریتم RSA طراحی و پیاده سازی شود. سپس به تست این دستگاه فرعی پرداخته شود و در صورتی که آزمایش های صورت گرفته موفق باشند طراحی ماژول RSA به پایان می رسد. با طراحی دیگر الگوریتم ها و تکمیل ماژول های دیگر مثل بخش رمزنگاری متقارن، رمزنگاری نامتقارن، درهم سازی و ... و جمع آوری این ماژول های مستقل از هم در غالب بسته ای با امنیت فیزیکی، می توان به مرور و در طی فازهایی به هدف نهایی رسید.

در پایان این گزارش لازم می بینم که علاوه بر تشکر از خانواده، اساتید، دانشجویان، دوستان و همه افرادی که در گرداوری این اثر مرا یاری دادند نکاتی را هم یادآور شود.

از مسئولین و اساتید دانشگاه علم و صنعت ایران گلایه کوچکی دارم. برخلاف نام این دانشگاه که انتظار می رود حداقل این دانشگاه به اندازه بقیه دانشگاه ها بزرگ کشور با صنعت ارتباط داشته باشد، متاسفانه باید بیان کنم که مسئولین ما هیچ تلاشی در جهت ارتباط با صنعت ندارند. نمونه این مسئله برای مثال زمانی مشخص می شود که دانشجویی قصد دارد که دوره کارآموزی خود را به صورت کامل و واقعی در یک سازمان یا موسسه و یا شرکت کامپیوتری بزرگ بگذراند. متاسفانه هیچ تلاشی از سوی دانشگاه و دانشکده در معرفی دانشجو به شرکت های بزرگ و معتبر کشور صورت نمی گیرد و این شرکت ها نیز از پذیرش دانشجویان به عنوان کارآموز سر باز می زنند.

گلایه دیگرم از مسئولین و استانید موسسه تحقیقات ارتباطات و فناوری اطلاعات است. بسیار خرسندم که دوره کارآموزی ام را در خدمت این موسسه بوده ام و در بخش امنیت به خدمت پرداختم و امیدوارم این همکاری ادامه داشته باشد. اما به نظرم نحوه کار دانشجویان و گذراندن درس کارآموزی در این موسسه بیشتر به درس پروژه شباهت دارد تا کارآموزی. چرا که در این موسسه علی رغم وجود امکانات سخت افزاری و ساختمانی بزرگ، هیچ آزمایش گاهی در اختیار کارآموزان برای انجام هرچه بهتر مسئولیت های سپرده شده به آن ها قرار نمی دهد و در نتیجه شخصی مثل من مجبور می شود درباره قطعه ای پژوهش کند که نه تنها از نزدیک به آزمایش بر روی آن نپرداخته است، بلکه حتی از نزدیک این دستگاه را ندیده است! و بدیهی است که این موضوع در کیفیت گزارش ها تاثیرگذار است.

شخصا علاقه داشتم تا این گزارش جنبه ترجمه و گرداوری چند مرجع نباشد. برای مثال ایده ها و نظرات شخصی بسیاری برایم به وجود آمد که قصد داشتم به عنوان آغازی مناسب در جای جای این گزارش قرار بدهم. اما به دلیل همین مشکل، یعنی عدم آزمایش و کار با این ماژول ها، اطلاعات عملی کافی در اختیار نداشتم و به همین سبب، اعتماد به نفس لازم برای بیان ایده های شخصی از من گرفته شد. به همین دلیل شاید کمی بعد نوآوری این گزارش کم رنگ شده باشد.

در پایان جا دارد که بار دیگر از همه اشخاصی که در فراهم سازی این گزارش مرا یاری دادند تشکر کنم و اظهار امیدواری و آمادگی نمایم که در صورتی که دانشگاه علم و صنعت ایران و یا موسسه تحقیقات ارتباطات و فناوری اطلاعات تصمیم به ساخت این دستگاه ها نمایند، من نیز به عنوان عضو کوچکی در کنار پژوهشگران این عرصه، به خدمت به کشور عزیزمان بپردازم.

پیوست 1: واژه نامه

حمله فعال Active Attack

مورد انتظار Alleged

واسط کاربردی برنامه نویسی Application Programming Interface

رمزنگاری نامتقارن Asymmetri Encryption

امنیت نامتقارن Asymmetri Security

تحلیل حمله Attac Cryptanalysis

حمله Attack

معتبر Authentic

برچسب اصالت Authentication Tag

اصالت Authentication

اصلیت Authenticity

با اصالت Authenticated

مرجع صلاحیت دار Authority

اجازه دهی Authorization

فراهمی Availabilit

رمزگذاری بلاکی Block Cipher

تنگنا Bottleneck

حمله وحشی Brute-Force Attack

کارت خوان Card Reader

مرجع صلاحیت دار مرکزی Central Authority

گواهی نامه Certificate

مرجع صلاحیت دار گواهی نامه Certificate Authority

مرجع صلاحیت دار گواهی Certification Authority

زنجیره اعتماد Chain of Trust

رمز Cipher

متن رمزشده Ciphertext

حمله تنها متن رمزشده Ciphertext-Only Attack

مشتری Client

دسته بندی Clustering

پایداری برخورد Collision Resistant

امنیت کامپیوتر Computer Security

محرمانگی Confidentiality

پیکربندی Configuration

رمزنگاری مرسوم Conventional Encryption

تحلیل رمز Cryptanalysis

تحلیل گر رمز Cryptanalyst

سیستم رمزنویسی Cryptographic System

رمز نویسی Cryptography

رمزشناسی Cryptology

متمرکز بر روی داده Data-Centric

رمزگشایی Decryption

الگوریتم رمزگشایی Decryption Algorithm

خلاصه Digest

امضا دیجیتال Digital Signature

پوشه Directory

سرویس پوشه Directory Service

پویا Dynamic

رمزنگاری Encryption

لینک رمزنگاری شده Encrypted Link

الگوریتم رمزنگاری Encryption Algorithm

طرح رمزنگاری Encryption Scheme

سیستم انتهایی End System

انتها به انتها End-to-End

اثر انگشت Fingerprint

مدل حالت نهایی Finite State Model

ثابت افزار Firmware

دنباله وارسی قاب Frame Cheack Sequence

چارچوب Framework

همه منظوره General Purpose

اصلی Genuine

دست دهی Handshaking

ماژول امنیت سخت افزاری Hardware Security Module

ساکن سخت افزار Hardware-Resident

تابع درهم آمیزی Hash Function

شناسه ID

یکپارچگی Integrity

مصادره کردن Intercept

واسط Interface

پایداری در برابر نفوذ Intrusion-Resistant

توزیع کلید Key Distribution

مرکز توزیع کلید Key Distibution Center

تبادل کلید Key Exchange

تولید کلید Key Generation

سیدکلید Key Seed

جریان کلید Keystream

واقعه نگار ضربه کلید Keystrok Logger

حمله متن ساده آشکار شده Known-Plaintex Attack

تراز بار Load Balancing

شاه کلید Master Key

اصلالت پیام Message Authentication

کد اصالت پیام Message Authentication Code

روش شناسی Methodlogy

حمله غیرقابل تست سبک شده Mitigated Non-Testable Attack

عدم انکار Nonrepudiation

از رده خارج شده Obsolete

کلید جلسه یک بار مصرف One-Time Session Key

تابع درهم آمیزی یک طرفه One-Way Hash Function

دشمن Opponent

حمله غیر فعال Passive Attack

کلید دائمی Permanent Key

دست اندازی فیزیکی Physical Tamper

جریان متن ساده Plaintext Stream

متن ساده Plaintext

پایداری پریمیج Preimage Resistant

حریم Privacy

کلید خصوصی Private Key

واسط برنامه نویسی Programmable Interface

بیت شبه تصادفی Pseudorandom Bit

تولید عدد شبه تصادفی Pseudo-Random Number Generation

جریان شبه تصادفی Pseudorandom Stream

کلید عمومی Public Key

الگوریتم کلید عمومی Public-Key Algorithm

گواهی نامه کلید عمومی Public-Key Certificate

رمزنویسی کلید عمومی Public-Key Cryptography

سیستم رمزگذاری کلید عمومی Public-Key Cryptosystem

رمزنگاری کلید عمومی Public-Key Encryption

زیرساخت کلید عمومی Public-Key Infrastructure

طرح کلید عمومی Public-Key Scheme

تولید عدد تصادفی Random Number Generation

تصادفی بودن Randomness

قابلیت اطمینان Reliability

انکار کردن Repudiate

انکار Repudiation

فسخ گواهی نامه Revocation of Certificate

کلید اصلی Root Key

مقیاس پذیری Scalability

پایداری پریمیج دوم Second Preimage Resistant

رازداری Secrecy

کلید مخفی Secret Key

رمزنگاری کلید مخفی Secret-Key Encryption

تابع درهم آمیزی امن Secure Hash Function

حمله امنیتی Security Attack

مکانیزم امنیتی Security Mechanism

سرویس امنیتی Security Service

کلید جلسه Session Key

امضا کردن Sign

تک کلید Single Key

رمزنگاری تک کلیدی Single-Ke Encryption

کارت هوشمند Smart Card

ماژول امنیت نرم افزاری Software Security Module

خاص منظوره Special Purpose

رمزگذاری جریانی Stream Cipher

پایداری برخورد قوی Strong Collision Resistant

رمزگذاری بلاکی متقارن Symmetric Block Cipher

رمزنگاری مرسوم متقارن Symmetric Conventional Encryption

رمزنگاری متقارن Symmetric Encryption

تشخیص دست اندازی Tamper Detection

اثبات دست اندازی Tamper Evidence

مقاوم در برابر دست اندازی Tamper Proof

حفاظت دست اندازی Tamper Protection

پایداری دست اندازی Tamper Resistance

پاسخ دست اندازی Tamper Response

دست اندازی Tampering

فناوری Technology

ارتباط دوربرد Telecommunication

تهدید Threat

نشانه Token

اصالت دو فاکتوری Two-Factor Authentication

رمزنگاری دو کلیدی Two-Key Encryption

امنیت دوسطحی Two-Level Security

بهبود Upgrade

کاربرپسند User Friendly

واسط کاربری User Interface

نام کاربری User Name

نگارش Version

پایداری برخورد ضعیف Weak Collision Resistant

صفرکردن Zeroize

مراجع

[1] STALLINGS W., Network Security Essentials: Applications and Standards, Fourth Edition, Prentice Hall, Boston, 2011.

[2] KIZZA J.M., A Guide to Computer Network Security, Springer, New York, 2009.

[3] رسولی میثم، الگوریتم های رمزنگاری و امضا دیجیتال، www.parsbook.org.

[4] ATTRIDGE J., An Overview of Hardware Security Modules, SANS Institute InfoSec Reading Room, 2002.

[5] IVARSSON J. and NILSSON A., A Review of Hardware Security Modules Fall 2010, Certezza, Stockholm, 2010.

[6] SMITH S.W., Hardware Security Modules, Chapman and Hall, 2010.

[7] Haw to Choose a Hardware Security Module (HSM), www.arx.com, Visited 2012.

[8] CREN Strategic and Practical FAQ, Hardware Security Modules, http://www.cren.net/crenca/onepagers/hsm.html, 2001.

[9] Beyond FIPS 140-1: Essential Best Practices for Hardware Security Modules, Chrysalis-ITS, 2001.

[10] Hardware Security Modules: Where Businesses Puts Its Trust, AEP Systems, 2003.

[11] [www.safenet-inc.com](http://www.safenet-inc.com), Visited 2012.

[12] [www.thales-esecurity.com](http://www.thales-esecurity.com/), Visited 2012.

[13] www.utimaco.com, Visited 2012.

Abstract

Hardware Security Module is a dedicated and secured computer, used for security purposes. Today, these modules have a undeniable role in network security, certificates managment and public key infrastructure. At the beginning, this report explains security concept and its categories, covered in chapter one. Next in chapter two, the definition of Hardware Security Module, its properties and applications, along with important points for evaluating and purchasing are provided. At the end of the chapter, some vendors and new products are studied. Also there are some efforts to explain about designing and producing these devices.

[This Page Intentionally Left Blank]

In The Name Of God



Iran University of Scinece and Technology

School of Computer Engineering

Computer Engineering (Hardware) Bachelor Internship Report

Hardware Security Module

Institute:

Research Institute for Information and Communication Technology

(ex ITRC)

Adviser: Dr. Mohsen Soriyani

Supervisor: Dr. Mohammad Hessam Tadayon

Author: Mohammad Mahdavi Lahijani

Summer 2012

1. 1 National Institute of Standards and Technology [↑](#footnote-ref-1)
2. 2 Integrity [↑](#footnote-ref-2)
3. Availability [↑](#footnote-ref-3)
4. Confidentiality [↑](#footnote-ref-4)
5. Firmware [↑](#footnote-ref-5)
6. Privacy [↑](#footnote-ref-6)
7. Central Intelligence Agency [↑](#footnote-ref-7)
8. Authenticity [↑](#footnote-ref-8)
9. Request For Comments [↑](#footnote-ref-9)
10. Threat [↑](#footnote-ref-10)
11. Attack [↑](#footnote-ref-11)
12. Cryptography [↑](#footnote-ref-12)
13. Cryptanalysis [↑](#footnote-ref-13)
14. Cryptology [↑](#footnote-ref-14)
15. Encryption [↑](#footnote-ref-15)
16. Plaintext [↑](#footnote-ref-16)
17. Ciphertext [↑](#footnote-ref-17)
18. Decryption [↑](#footnote-ref-18)
19. Cryptanalyst [↑](#footnote-ref-19)
20. Cipher Code [↑](#footnote-ref-20)
21. Ciphertext-Only Attack [↑](#footnote-ref-21)
22. Known-Plaintext Attack [↑](#footnote-ref-22)
23. Encryption Scheme [↑](#footnote-ref-23)
24. Cipher [↑](#footnote-ref-24)
25. Brute-Force [↑](#footnote-ref-25)
26. The International Telecommunication Union (ITU) Telecommunication Standardization Sector (ITU-T) [↑](#footnote-ref-26)
27. Open System Interconnection [↑](#footnote-ref-27)
28. Security Attack [↑](#footnote-ref-28)
29. Security Mechanism [↑](#footnote-ref-29)
30. Security Service [↑](#footnote-ref-30)
31. Passive Attack [↑](#footnote-ref-31)
32. Opponent [↑](#footnote-ref-32)
33. Active Attack [↑](#footnote-ref-33)
34. Symmetric Encryption [↑](#footnote-ref-34)
35. Single-Key Encryption [↑](#footnote-ref-35)
36. Secret-Key Encryption [↑](#footnote-ref-36)
37. Conventional Encryption [↑](#footnote-ref-37)
38. Asymmetric Encryption [↑](#footnote-ref-38)
39. Two-Key Encryption [↑](#footnote-ref-39)
40. Public-Key Encryption [↑](#footnote-ref-40)
41. Block Cipher [↑](#footnote-ref-41)
42. Stream Cipher [↑](#footnote-ref-42)
43. Encryption Algorithm [↑](#footnote-ref-43)
44. Secret Key [↑](#footnote-ref-44)
45. Decryption Algorithm [↑](#footnote-ref-45)
46. Single Key [↑](#footnote-ref-46)
47. Repudiate [↑](#footnote-ref-47)
48. Secrecy [↑](#footnote-ref-48)
49. Symmetric Encryption Algorithm [↑](#footnote-ref-49)
50. Data Encryption Standard [↑](#footnote-ref-50)
51. Triple DES [↑](#footnote-ref-51)
52. Advanced Encryption Standard [↑](#footnote-ref-52)
53. Pseudorandom Bit [↑](#footnote-ref-53)
54. Pseudorandom Stream [↑](#footnote-ref-54)
55. Keystream [↑](#footnote-ref-55)
56. Message Authentication [↑](#footnote-ref-56)
57. Key Distribution [↑](#footnote-ref-57)
58. Authentication [↑](#footnote-ref-58)
59. Encryption Scheme [↑](#footnote-ref-59)
60. Handshaking [↑](#footnote-ref-60)
61. Public Key and Private Key [↑](#footnote-ref-61)
62. Repudiation [↑](#footnote-ref-62)
63. Digital Signature [↑](#footnote-ref-63)
64. Certificate [↑](#footnote-ref-64)
65. Sign [↑](#footnote-ref-65)
66. Key Exchange [↑](#footnote-ref-66)
67. Session Key [↑](#footnote-ref-67)
68. Public-Key Cryptosystem [↑](#footnote-ref-68)
69. Nonrepudiation [↑](#footnote-ref-69)
70. Message Authentication [↑](#footnote-ref-70)
71. Authentic [↑](#footnote-ref-71)
72. Genuine [↑](#footnote-ref-72)
73. Alleged [↑](#footnote-ref-73)
74. Authentication Tag [↑](#footnote-ref-74)
75. Message Authentication Code [↑](#footnote-ref-75)
76. One-Way Hash Function [↑](#footnote-ref-76)
77. Secure Hash Function [↑](#footnote-ref-77)
78. Fingerprint [↑](#footnote-ref-78)
79. Preimage Resistant [↑](#footnote-ref-79)
80. Second Preimage Resistant [↑](#footnote-ref-80)
81. Weak Collision Resistant [↑](#footnote-ref-81)
82. Collision Resistant [↑](#footnote-ref-82)
83. Strong Collision Resistant [↑](#footnote-ref-83)
84. Frame Check Sequence [↑](#footnote-ref-84)
85. Authenticated [↑](#footnote-ref-85)
86. Key Distribution Center [↑](#footnote-ref-86)
87. End System [↑](#footnote-ref-87)
88. One-Time Session Key [↑](#footnote-ref-88)
89. Permanent Key [↑](#footnote-ref-89)
90. Master Key [↑](#footnote-ref-90)
91. Central Authority [↑](#footnote-ref-91)
92. Bottleneck [↑](#footnote-ref-92)
93. ID [↑](#footnote-ref-93)
94. Certificate Authority [↑](#footnote-ref-94)
95. Public-Key Certificate [↑](#footnote-ref-95)
96. Directory Service [↑](#footnote-ref-96)
97. User Name [↑](#footnote-ref-97)
98. Framework [↑](#footnote-ref-98)
99. Certification Authority [↑](#footnote-ref-99)
100. Version [↑](#footnote-ref-100)
101. Public-Key Infrastructure [↑](#footnote-ref-101)