امنیت اطلاعات

از ابتدا تا امروز

محسن کرمی

امنیت اطلاعات: از ابتدا تا امروز

محسن کرمی

ویراست اول

تابستان ۱۳۹۶

(CC BY-SA 4.0)

نام کتاب : امنیت اطلاعات: از ابتدا تا امروز

نویسنده : محسن کرمی

گواهی : اشتراک همسان (کریییتیو کامنز) – نسخه ۴.۰

ویراست : اول ـ تابستان ۱۳۹۶

آدرس الکترونیکی کتاب : Https://mrgeo22.github.io/per/publications/information-security

Https://mrgeo22.github.io

چنین گفته می‌شود که جنگ جهانی اول، جنگ شیمی‌دان‌ها بود چون گاز خردل و کلر برای اولین بار بود که به‌کار گرفته می‌شدند، و جنگ جهانی دوم جنگ فیزیک‌دان‌ها بود، زیرا بمب اتم منفجر شد.

با همین استدلال بحث می‌شود که جنگ جهانی سوم جنگ ریاضی‌دان‌ها خواهد بود، زیرا که ریاضی‌دان‌ها روی سلاح فوق‌العاده بعدی کنترل دارند- اطلاعات.

ریاضی‌دان‌ها مسئول توسعه کدهایی می‌شوند که در حال حاضر برای محافظت از اطلاعات نظامی مورد استفاده قرار می‌گیرند. جای شگفتی نیست که ریاضی‌دان‌ها هم‌چنین در صف اول نبرد برای شکستن این کدها هستند.

سایمون سینگ

کتاب رمز: دانش محرمانگی از مصر باستان تا رمزنگاری کوانتومی

این کتاب به شکل آزاد منتشر شده است و از سویی برآن شدم که این محتوا را هم‌چنین به شکل رایگان در دسترس قرار دهم(توجه داشته باشید که محتوای آزاد با رایگان متفاوت است). اگر پس از مطالعه کتاب آن را مفید یافتید و مورد پسندتان بود خوشحال خواهم شد که در مرحله اول آن را به دیگران نیز پیشنهاد دهید تا اگر فرد دیگری نیز می‌باشد که ممکن است این کتاب برایش مفید باشد، از آن بهره‌مند شود. اگر می‌خواهید این کتاب را در وب‌گاه یا وب‌نوشته خود بازنشر دهید خوشحال می‌شوم که پیوند به نشانی الکترونیکی کتاب در برگه پیش را فراموش نکنید تا دوستان از همان برگه کتاب را دانلود نمایند.

هدف از نگارش و انتشار این کتاب گشترش آگاهی در این زمینه به شکل عامه‌فهم بوده است، متاسفانه بیشتر کتاب‌هایی که در این زمینه نگاشته شده‌اند یا تخصصی بوده و یا بیشتر جنبه کاربردی (بدون ایجاد نگرشی ژرف از سازوکار آن) داشته‌اند، از این روی برآن شدم تا خود کتابی را به زبانی غیرتخصصی برای دوستداران این مبحث به نگارش درآورم، امیدوارم که توانسته باشم این جای خالی را تا حدی پر کرده باشم. با این وجود، اگر از نتیجه کار خشنود بودید می‌توانید با حمایت مالی از آن به فرهنگ محتوای آزاد یاری دهید و جزئی از چرخه انتشار آزاد محتوا باشید.

در صورت تمایل می‌توانید مبلغ مورد نظر را به شماره حساب 1-1054330-800-9261 یا شماره شبای IR240560926180001054330001 به نام محسن کرمی در بانک سامان واریز نمایید، برای آگاهی از دیگر راه‌های موجود می‌توانید به نشانی زیر مراجعه نمایید:

Https://mrgeo22.github.io/per/donate

فهرست مطالب

پیش‌گفتار ۷

فصل ۱: پنهان‌نگاری ۱۱

۱- راه‌های پنهان‌نگاری ۱۱

۲- پنهان‌نگاری در دنیای دیجیتال ۱۲

۱- پنهان‌نگاری در صوت

۲- پنهان‌نگاری در تصویر

۳- تحلیل پنهان‌نگاری ۱۴

فصل ۲: کدگذاری ۱۷

۱- روش‌های کدگذاری ۱۹

۱- کد یکبار مصرف

۲- کد احمق

۲- تحلیل کدگذاری ۲۱

فصل ۳: رمزنگاری ۲۳

۱- الگوریتم‌های رمزنگاری ۲۵

۱- رمزنگاری جانشینی

۲- رمزنگاری جابه‌جایی

۳- پد یکبار مصرف

۲- رمزنگاری در دنیای دیجیتال ۴۰

۱- رمزنگاری کلید متقارن

۲- رمزنگاری کلید نامتقارن

۳- رمزنگاری کوانتومی

۳- روش‌های رمزگشایی ۴۶

۱- حمله جستجوی فراگیر

۲- تحلیل فراوانی

۳- سخن پایانی

پیوست A: نمونه‌های پنهان‌نگاری ۵۵

پیوست B: اصول کرکهف ۵۹

پیوست C: فراوانی حروف پارسی در متن‌های استاندارد ۶۱

پیوست D: فراوانی حروف پارسی در متن‌های استاندارد (کد منبع برنامه) ۶۳

کتاب‌نامه ۶۹

نمایه ۷۱

پیش‌گفتار

اگر گمان می‌کنید که فناوری می‌تواند مشکلات امنیتی شما را برطرف سازد پس شما نه مشکلات را درک کرده‌اید و نه فناوری را فهمیده‌اید.

بروس شنیر[1]

من بیشتر از اینکه یک نویسنده باشم، روزنامه‌نگار علم هستم. تا کنون نوشته‌های من در قالب مقاله در نشریه‌ها چاپ شده‌اند و راستش را بخواهید شروع این کار نیز به هدف نگارش یک مقاله بود، با این وجود چون خودم بیشتر از اینکه نشریه‌خوان باشم کتاب‌خوان هستم همیشه این دید در کارهایم نیز خودنمایی می‌کرد و به نوعی روح مقاله‌ها بیشتر از اینکه خبری و رویدادی باشد تحلیلی و جامع‌نگرانه بود.

البته این نوع از مقاله‌نویسی در دنیای روزنامه‌نگاری کاملا معمول بوده و بسیاری از کارها به این شکل نوشته می‌شوند اما نکته‌ای که تفاوت ایجاد می‌کند در حجم کار است. در این مورد به دلیل علاقه شخصی و دانش نسبی که در این زمینه داشتم کار را با دقت بالا و به شکل گسترده‌ای انجام دادم که در پی آن علاوه بر زمان‌بر بودن، نتیجه بسیار مفصل‌تر از آنچه شد که انتظار داشتم و از همین روی تصمیم بر چاپ نوشته‌ام در قالب کتاب و ورود به دنیای نوینی از نویسندگی گرفتم.

1. Bruce Schneier: رمزنگار، متخصص امنیت رایانه و نویسنده آمریکایی

از دیرباز انسان‌ها در پی یافتن راه‌هایی برای حفظ امنیت اطلاعات مهم بوده‌اند. پادشاهانی که پیام‌هایی برای کشورهای متحد ارسال می‌کردند و فرماندهانی که دستوراتشان را برای سپاهیان خود در نقاط مختلف می‌فرستادند. گاهی ارسال درست و پنهانی یک پیام کلیدی می‌توانست نتیجه یک نبرد تاریخی را مشخص کند و می‌توان انسان‌هایی را دید که برای این هدف جان خود را فدا می‌کردند. درواقع هر چه زمان بیشتری می‌گذرد لزوم اهمیت به امنیت اطلاعات نیز بیشتر احساس می‌شود، چه بسا اگر متفقین نمی‌توانستند رمزهای نازی‌ها را بشکنند امروز دنیای دیگری را شاهد می‌بودیم!

جدای از اهمیت بسیار این موضوع، روش‌های به کار گرفته شده بسیار زیبا هستند و آموختن این روش‌ها حتی بدون کاربرد مستقیم در زندگی لذت‌بخش هستند. در این کتاب تاریخچه‌ای از روش‌های گوناگون حفظ امنیت اطلاعات آورده شده‌است، ترفندهایی از گذشته‌های دور در یونان باستان تا نوین‌ترین روش‌هایی که امروزه مورد استفاده قرار می‌گیرند. در این مسیر در کنار تصویری که از چگونگی تکامل و بلوغ جداگانه‌ی هرکدام از این رویکردها بدست می‌آوریم با روش انجام بسیاری از آن‌ها نیز آشنا می‌شویم، به گونه‌ای که خواننده پس از مطالعه کتاب می‌تواند متن خود را به روش‌های بسیار متنوع و کاملاً متفاوت پنهان و یا رمزنگاری کند و حتی می‌تواند مطالعه این کتاب را به دوستان خود نیز پیشنهاد کند تا بتوانند با یکدیگر از راه‌های گوناگون به شکلی ایمن تبادل پیام داشته باشند.

به یاد دارم در دوران دبیرستان به روش رمزنگاری جانشینی برای خود الفبایی ساختم و آن را با یکی از دوستانم به اشتراک گذاشتم، برای این کار از نمادهایی نوین استفاده کردم که پایه طراحی آن‌ها تا حدودی به نمادهای نجومی گرد آسمان[1] نزدیک بود.

---

۱. گرد آسمان یا منطقه البروج شامل بخشی از آسمان است که صورت‌های فلکی و خورشید و چندی از سیاره‌ها را دربر دارد و از همان دوران کهن برای هرکدام نمادی ساخته‌اند که تا به امروز تقریباً به همان شکل باقی مانده‌اند.

با این کار سیستم پیام‌رسانی را در اختیار داشتم که بدون نگرانی از نشت اطلاعات میتوانستم با آن ارتباط برقرار کنم و این برای من لذت‌بخش بود. این کتاب ادامه همان کنجکاوی‌ها است که به آرامی در من رشد کرد و امروز می‌خواهم بخشی از آن را با شما نیز به اشتراک بگذارم.

نوشته‌های این کتاب شامل دو بخش است، شیوه‌هایی که انسان‌ها بدون نیاز به رایانه‌ها استفاده می‌کرده‌اند و در نتیجه تقریباً کامل آموزش داده شده‌اند و می‌توان پس از مطالعه کتاب آن‌ها را با همان کیفیت انجام داد و روش‌هایی که پس از ظهور رایانه‌ها و قدرت محاسباتی فوق‌العاده آن‌ها به دست ریاضی‌دان‌ها به وجود آمده‌اند.

به احتمال زیاد تاکنون از روش‌های مبتنی بر رایانه استفاده کرده‌اید ولی احتمالاً خود هم متوجه این کار نشده‌اید!

در این موارد به دلیل پیچیدگی بسیار و تخصصی بودن کار امکان انجام آن برای غیرمتخصص‌ها فراهم نمی‌باشد و تنها به واسطه نرم‌افزارها از آن‌ها بهره می‌بریم، اما در این کتاب آن‌ها را نیز کنار نگذاشته‌ام.

برای این دسته از روش‌ها علاوه بر توضیح چگونگی ایجاد و تکامل آن‌ها، نقاط قوت و ضعف هرکدام را بیان کرده‌ام و درکنار این‌ها چگونگی کارکردشان را نیز توضیح داده‌ام، به‌گونه‌ای که خواننده پس از مطالعه کتاب اساس کار این شیوه‌ها را خواهد دانست و دیگر نسبت به آنچه در حال روی دادن است ناآگاه نیست.

آیا تاکنون دقت کرده‌اید، زمانی که به درگاه پرداخت الکترونیکی متصل می‌شوید آدرس اینترنتی از Http به Https تغییر می‌کند؟ آیا توجه کرده‌اید که چرا پس از اینکه گذرواژه‌تان را در وب‌گاهی وارد می‌کنید و سپس دستور ارسال می‌دهید به یکباره تعداد کاراکترهای گذرواژه‌تان چند برابر می‌شود؟

این‌ها نمونه‌هایی از رمزنگاری در دنیای امروز هستند که ما هر روزه با آن‌ها سروکار داریم ولی شاید تا کنون هیچ‌گاه به این نیندیشیده باشیم که این کار چگونه انجام می‌گیرد، یا اگر هم برایمان پرسشی جالب بوده هیچ‌گاه به دنبال یافتن پاسخ برنیامده‌ایم.

اینکه بدانیم چه نرم‌افزارهایی برای امنیت اطلاعات مناسب هستند و یا چگونه از هک شدن جلوگیری کنیم یک روی سکه است و اینکه بدانیم پشت‌صحنه این نرم‌افزارها و روش‌ها چه خبر است روی دیگر آن. این کتاب کوششی است در جهت آگاهی از بنیان‌های امنیت اطلاعات و بررسی رویکردهای مختلف در این زمینه که از گذشته‌های دور مورد توجه بشر بوده است.

شایسته است که از همراهی دوستانی سپاسگذاری کنم که در نگارش این کتاب به راه‌های گوناگون به من یاری داده‌اند که اگر همیاری این دوستان نبود چه بسا که بسیاری کاستی‌ها در کارم خودنمایی می‌کردند، هم «جادی» عزیز که کار برنامه‌نویسی و تهیه نمودارها را به خوبی انجام داد و هم دوستان بسیاری که در خواندن متن اولیه و ویراستاری آن به من یاری بسیار رساندند. امیدوارم که دست‌آورد این کوشش‌ها موردپسند خوانندگان قرار گیرد.

محسن کرمی

تابستان ۱۳۹۶

۱

پنهان نگاری

اگر قصد ارسال پیامی سری داریم بهترین حالت این است که کسی از وجود این پیام آگاه نشود، اینگونه ما بیشترین میزان امنیت را داریم. از این رو اولین گزینه پیش روی بشر همواره پنهان نگه داشتن پیام‌هایی بوده که ارسال می‌کرده است، هرچه جزئیات بیشتری پنهان می‌ماند امنیت پیام نیز بیشتر میشد. در گام اول بایستی کسی از قصد فرستنده [برای ارسال پیام] آگاه نمیشد و در گام بعدی بایستی این کار به گونه‌ای انجام میگرفت که کسی متوجه آن نشود، به شکلی که یا از مسیرهای نامعمول که کسی در آن مسیرها حرکت نمیکرد پیام را منتقل میکردند و یا اینکه پیام را به گونه‌ای پنهان می‌کردند که کسی به پیام‌رسان شک نکند.

۱-۱ راه‌های پنهان‌نگاری

برای پنهان کردن متن در زمان‌های گذشته گاهی پیام را با جوهرهای نامرئی می‌نوشتند که درواقع موادی همچون شیر و آبلیمو بودند، این مواد پس از خشک شدن به شکل

عادی روی کاغذ دیده نمی‌شوند ولی زمانی که در برابر حرارت قرار می‌گیرند تیره شده و قابل خواندن می‌شوند. جوهر نامرئی حتی امروزه نیز کاربرد دارد و همواره یکی از وسیله‌های رایج مورد استفاده جاسوسان در کشورهای مختلف بوده است (البته کم‌کم دارای ترکیباتی نوین و پیچیده شده که به راحتی قابل شناسایی نیست) و از آن‌جایی که این جاسوس‌ها در شغل‌های کاملاً متفاوت و متنوعی در کشورهای مورد نظر مشغول به کار می‌شدند امکان پیگیری نامه‌های آنان وجود نداشت.

پیگیری نمودن همه این نوع نامه‌ها گران است، روشی یقیناً ساده انگارانه در همه نمونه‌های یک سامانه تجسسی. **پنهان‌نگاری ساده نیست**، اما یافتن آن کاری دشوارتر است. متن پنهان شده هرجایی می‌تواند باشد، برای نمونه ابتدای همین پاراگراف انتخاب خوبی است! [اولین حرف از هر واژه را بخوانید.]

اما راه دیگری که گاهی کشورهای مختلف در بازه‌های زمانی ویژه‌ای برای مقابله با این مشکل در پیش می‌گرفتند، مسدود کردن راه ارتباطی بود. به این شکل که ارسال تمام کالاهایی که امکان پنهان‌نگاری در آن‌ها وجود داشت همچون نقشه، نامه، کارت‌پستال و امثال آن را ممنوع می‌کردند. با این وجود همیشه راه‌های هوشمندانه‌ای برای دور زدن این محدودیت‌ها وجود دارد، برای نمونه در گذشته گاهی سر غلامان را می‌تراشیدند و متن پیام را روی سر آن‌ها خالکوبی می‌کردند و پس از رویش دوباره موهای سرشان آن‌ها را به مقصد مورد نظر می‌فرستادند و در این راه هم اگر نگهبانی به او شک می‌کرد غلام با آسودگی خاطر تمام هرآنچه همراه خود داشت را تسلیم وی می‌کرد.

۱-۲ پنهان‌نگاری در دنیای دیجیتال

پنهان‌نگاری به مرور زمان پیشرفته‌تر و پیچیده‌تر شد، به ویژه پس از ورود رایانه‌ها و توان محاسباتی فوق‌العاده بالایشان پنهان‌نگاری نیز رنگی تازه به خود گرفت و از سوی دیگر

پس از ورود اینترنت، حجم بسیار بالای اطلاعات رد و بدل شده کار را برای پنهان شدن بسیار راحت‌تر کرد. داده‌ها را دیگر به جای کاغذ در بیت‌ها پنهان می‌کردند و علاوه بر متن دیجیتالی گزینه‌های بیشتری چون صدا، تصویر و فیلم نیز قابل استفاده بودند.

۱-۲-۱ پنهان‌نگاری در صوت

هیچ موجود زنده‌ای توانایی شنیدن تمام فرکانس‌ها را نداشته و محدوده شنوایی خاص خود را دارد، انسان نیز از این قائده مستثنی نیست و دارای محدوده شنوایی ۲۰ تا ۲۰ هزار هرتز است. البته هری فردیناند اولسون[1] که فردی پیشرو در زمینه مهندسی صوت محسوب می‌شود، در کتاب موزیک، فیزیک و مهندسی[2] ادعا کرده که در شرایط مناسب آزمایشگاهی گوش انسان توانایی شنیدن صداهایی تا فرکانس‌های در حد ۱۲ هرتز را نیز دارا است.

با این توضیحات بایستی حدس زده باشید که با استفاده از محدودیت‌های سیستم شنوایی انسان می‌توان پیام‌هایی را در فایل‌های صوتی پنهان کرد، پیام‌هایی که در فرکانس‌های بسیار بلندتر و یا پایین‌تر از این محدوده قرار دارند و گوش ما نسبت به آن‌ها ناشنوا به حساب می‌آید. راه دیگر در حساسیت سیستم شنوایی ما نهفته است، در حالت کلی گوش ما بسیار حساس است و بنا به گفته استنلی گلفند[3] در کتاب اصول شنوایی شناسی[4] در فرکانس‌های ۲۰۰۰ تا ۵۰۰۰ هرتز گوش ما حتی نسبت به ضعیف‌ترین صداها نیزحساس است و آن‌ها را تشخیص می‌دهد. اما نکته اصلی این است که می‌توان این صداهای ضعیف

1. Harry Ferdinand Olson
2. Music, Physics and Engineering
3. Stanley Gelfand
4. Essentials of Audiology

را با صداهای با شدت بالا ترکیب کرد تا سیستم شنوایی انسان گمراه شده و آن‌ها را تشخیص ندهد.

۱-۲-۲ پنهان‌نگاری در تصویر

شاید بتوان ادعا کرد که یکی از بهترین گزینه‌ها برای پنهان‌نگاری استفاده از تصاویر است. یکی از ویژگی‌های سیستم بینایی ما عدم توانایی تشخیص تفاوت‌های جزیی در الگوهای تصویری پیچیده است، به شکلی که نمی‌تواند در تصویری از یک منطقه وسیع که به ظاهر یکنواخت است جزییات و تفاوت‌های بخش‌های مختلف را تشخیص دهد که در این شرایط ما می‌توانیم از تصاویر به شکلی کاملاً مؤثر برای پنهان‌نگاری داده‌هایمان استفاده کنیم. نکته دیگر در مورد پنهان‌نگاری در تصاویر امکان جاسازی داده‌های زیاد است، به گونه‌ای که در هر تصویر می‌توان تا نزدیک به ۵۰ درصد داده‌های آن را بدون اینکه شاهد فروپاشی در تصویر باشیم با داده‌های مورد نظرمان جایگزین کنیم.

۱-۳ تحلیل پنهان‌نگاری

با وجود تمام مطالب گفته شده، تحلیل پنهان‌نگاری ـ که به معنای پی بردن به وجود پیام جاسازی شده و تخریب یا در صورت امکان استخراج آن می‌باشد ـ نیز هم‌ارز با پنهان‌نگاری رشد کرده و به پیشرفت‌های خوبی رسیده است. همان‌گونه که گفته شد سیستم‌های بینایی و شنوایی انسان محدودیت‌هایی دارد که باعث می‌شود به راحتی فریب خورده و توانایی تشخیص این جاسازی‌ها را نداشته باشند، راه حل این مشکل **تحلیل آماری** است. (می‌توانید نمونه‌هایی از پنهان‌نگاری را در پیوست A بیابید.)

زمانی که روی یک فایل جاسازی انجام می‌شود، هرچقدر هم که تغییرات به چشم نیایند باز تأثیر خود را در سیگنال‌های آن نشان خواهند داد، از این رو با بررسی آماری فایل مشکوک و مقایسه آن با فایل اصلی (تغییر نیافته) می‌توان به راحتی به وجود داده‌های جاسازی شده در آن پی برد. اما پس از استفاده روزافزون از روش‌های **فشرده‌سازی اتلاف داده** که نمونه‌های آن فایل‌های تصویری JPEG و فایل‌های صوتی MP3 هستند فرصتی دیگر برای پنهان‌نگاری بهتر داده‌ها ایجاد شد. این‌گونه فشرده‌سازی برای اینکه بتواند تا حد امکان از حجم فایل‌ها بکاهد مقداری از اطلاعات را که آسیب جدی به کلیت فایل وارد نمی‌کنند را پاک می‌کند، از این رو فایل به دست آمده کیفیتی پایین‌تر داشته و با فایل اصلی مقداری متفاوت خواهد بود.

امروزه تحلیل پنهان‌نگاری این نوع از فایل‌ها نیز به راحتی انجام می‌گیرد، زیرا شکل تغییراتی که این نوع فشرده‌سازی در سیگنال داده ایجاد می‌کند شناخته شده است و در صورتی که نوسانات مشاهده شده در سیگنال فایل به شکل متفاوتی باشد به راحتی تشخیص داده می‌شود و نمی‌تواند درترکیب با نوسانات ناشی از فشرده‌سازی فایل پنهان شود. این نوع از تحلیل‌های آماری پایه‌ای بوده و دقت آن تاحد زیادی به میزان دسترسی ما به فایل اصلی بستگی دارد. اما در برخی موارد تنها یک نمونه از فایل موجود است، در این حالت امکان مقایسه نیست و به تکنیک‌های تحلیلی پیشرفته‌تری نیاز داریم.

زمانی که سیگنال‌های مربوط به هر نوع فایلی را بررسی کنیم متوجه می‌شویم که همه دارای مقداری اطلاعات اضافه و ناخواسته هستند که منشاء آن‌ها به محیط برمی‌گردد و می‌توانند از عوامل مختلفی ایجاد شده باشند، به مجموعه این سیگنال‌های ناخواسته در اصطلاح **نویز** گفته می‌شود. به طور کلی پنهان‌نگاری می‌کوشد تا داده‌ها را به شکلی در فایل جاسازی کند که نوسان‌های ایجاد شده توسط آن از این نویزها قابل تشخیص نباشند، اما در عمل می‌بینیم که در بیشتر موارد به جای اینکه فایل‌ها را تحلیل و مدل سازی کرده و سپس به شکلی بی نقص ویژگی‌های نویز واقعی سیگنال را تقلید کنند، کار

را ساده‌تر کرده و تغییرات را تا حد امکان شبیه به نویز سفید ایجاد می‌کنند. به طور کلی بیشتر سیستم‌های پنهان‌نگاری به سادگی کم ارزش‌ترین بیت (آخرین بیت در کد باینری) را تغییر می‌دهند، در این حالت شکل نویز فایل تغییر داده شده تفاوت عمده‌ای با فایل اصلی ندارد ولی می‌توان انتظار داشت که با تحلیل بیت‌های باارزش‌تر بتوان این تفاوت نویز را آشکار کرد، در سال ۴۲۰۰/ ۱۳۸۳ الگوریتمی توسط جسیکا فردریش[1] و میروسلاو گلجن[2] در سازمان ثبت اختراع آمریکا تأیید شد که می‌تواند تغییراتی با چگالی حتی ۱ درصد را نیز با درجه اطمینان قابل قبولی آشکارسازی کند.

1. Jessica Fridrich

2. Miroslav Goljan

۲

کدگذاری

گاهی پنهان کردن ارسال یک متن کار راحتی نیست، حتی گاهی بدیهی است که پیامی ارسال خواهد شد. برای نمونه اگر کشوری وارد جنگ شود هر لحظه ممکن است که فرماندهان حاضر در میدان‌های جنگ در مرزهای کشور با پایتخت ارسال گزارش و دریافت مشورت داشته باشند و یا کاملاً روشن است که اگر از وب‌گاهی خرید اینترنتی انجام دهید اطلاعات بانکی شما در این بین منتقل خواهد شد. می‌توان نمونه‌های بسیاری را نام برد که امکان پنهان نگه‌داشتن اطلاعات منتقل شده کاری بسیار سخت و یا ناممکن است، از این رو خیلی زود شاهد ایجاد روش‌هایی برای تغییر متن پیام‌های ارسالی بودیم که باعث میشد اگر در این انتقال کسی هم پیام را به دست می‌آورد نتواند از محتوای آن سر در بیاورد. **کدگذاری** و **رمزنگاری** دو گزینه ما برای این کار هستند، اساس هر دو تغییر متن اصلی به متنی نامفهوم است به گونه‌ای که بتوان با در دست داشتن اطلاعات لازم دوباره متن اصلی را از آن بازتولید نمود.

روش کار در کدگذاری به این شکل است که برای هر واژه برابری را در نظر می‌گیرند که به آن کلمه کد[1] گفته می‌شود و سپس این برابرها را در کتابی با نام کتاب کد[2] گردآوری می‌کنند. بدین گونه با یاری گرفتن از این ابزار می‌توانند هر متنی را به شکلی درآورند که برای کسانی که کتاب کد را در اختیار ندارند قابل درک و خواندن نباشد. این روش شباهت زیادی به کار ترجمه دارد و کتاب‌کد نیز نقش یک واژه‌نامه را ایفا می‌کند، واژه‌نامه‌ای دوزبانه که می‌توان هم برای کدگذاری و هم بازگردانی متن اصلی از آن استفاده نمود. کتاب کد معمولاً دارای دو بخش است : یک بخش برای تبدیل واژه‌های زبان اصلی به کلمه‌های کد و بخش دیگر برای حالت برعکس.

تفاوت پایه‌ای کدگذاری و رمزنگاری در سطوح کاری آن‌ها است. در حالی که کدگذاری روی واژه‌ها تغییرات را اعمال می‌کند و با آن‌ها سروکار دارد، در رمزنگاری ابزار کار ما حروف (و در دنیای دیجیتال بیت‌ها) هستند. یکی از نکات منفی کدگذاری این است که به دلیل ساختار کاری این روش ما حتما نیازمند یک کتاب کد برای انجام کار هستیم که نه تنها کار با آن راحت و سریع نیست که به دلیل حجم کتاب، سری و دور از دسترس نگه داشتن آن کار دشواری است. در بیشتر این کتاب‌ها یک فهرست تهیه شده است که هرچند سرعت کار با آن را به شکل چشم گیری بهتر می‌کند ولی امنیت آن را شدیداً تحت تأثیر قرار می‌دهد و نیز اگر قصد انتقال کتاب را داشته باشیم نمی‌توان به راحتی امنیت آن را تأمین کرد.

اما از سویی دیگرکار بر روی واژه‌ها امتیاز بزرگی را نیز با خود به همراه دارد و آن سطح امنیتی بسیار بالایی است که برای پیام کد شده به ارمغان می‌آورد. با توجه به اینکه می‌توان از هر گونه نماد یا شکلی برای ساخت کلمه‌های کد استفاده کرد و نیز با توجه به

1. CodeWord

2. CodeBook

اینکه الگوی خاصی برای ایجاد این کلمه‌های کد وجود ندارد و می‌توان برای هر واژه از الگوی منحصر به فردی پیروی کرد، احتمال پیش‌بینی و به دست آوردن متن اصلی بسیار ضعیف است. درواقع اگر بتوان امنیت کتاب کد را تأمین کرد این روش بسیار ایمن‌تر از رمزنگاری است که به دلیل استفاده از یک الگوی ثابت برای رمزنگاری کل متن پتانسیل بالاتری برای درهم شکسته شدن دارد. (البته به جز روش رمزنگاری «پد یکبار مصرف» که در ادامه توضیح خواهم داد.)

۲-۱ روش‌های کدگذاری

۲-۱-۱ کد یکبار مصرف

گاهی از کدهای یکبار مصرف برای کدگذاری استفاده می‌شود (با پد یکبار مصرف که یکی از روش‌های رمزنگاری است اشتباه نشود)، این‌گونه کدها معمولاً برای ارسال پیام‌های کوتاه مورد استفاده قرار می‌گیرند. در زمانی که نیاز به تأیید یا تکذیب یک خبر مهم داریم، خواهان ارسال دستوری برای اجرا کردن یک نقشه یا تاکتیک نظامی هستیم، می‌خواهیم شکست یا پیروزی ماموریتی را گزارش دهیم و موارد این‌چنینی دیگر، یکی از بهترین گزینه‌های ممکن استفاده از کدهای یکبار مصرف است. روش کار آن‌ها بدین شکل است که واژه‌نامه‌ای قراردادی و محدود که دارای واژگان معمول مورد استفاده در اینگونه پیام‌ها است در دسترس طرفین قرار می‌گیرد و تنها یکبار در زمان مورد نیاز از آن استفاده می‌شود، اینگونه هم به دلیل محدود و کوچک بودن واژه‌نامه امکان لو رفتن آن کم است و هم به دلیل اینکه تنها یک بار از آن استفاده می‌شود امکان پی بردن به متن پیام‌ها و دستیابی به واژه‌نامه از روی جمع‌آوری و بررسی مجموع پیام‌ها (در صورتی که پیام‌های ارسالی لو رفته و به دست دشمن بیفتند) وجود ندارد، درواقع حتی اگر دشمن هم متن

پیام اصلی و هم متن رمزنگاری شده ارسالی را نیز به دست آورد باز خطری پیام‌های بعدی را تهدید نمی‌کند. از سویی دیگر به دلیل ماهیت یکبار مصرف بودن این واژه‌نامه می‌توان هر متن را به متن دلخواه دیگری تبدیل کرد که حتی اگر همه انسان‌ها پیام را دریافت کنند باز تنها فرد مورد نظر بتواند آن را تشخیص دهد. نمونه‌های بسیاری از استفاده از اینگونه کدگذاری‌ها مخصوصاً در زمان جنگ وجود دارد،یکی از نمونه‌های مشهور آن ارسال پیام **در سراسر اسپانیا، آسمان صاف است**[1] بود که از رادیو پخش شد! این پیام برای دیگران پیامی معمولی اما درواقع دستور آغاز جنگ داخلی اسپانیا بود.

۲-۱-۲ کد احمق

یکی از روش‌های کدگذاری استفاده از کد احمق[2] است. روش کار این کد شباهت زیادی به حرکت‌هایی دارد که سربازها در میدان جنگ و یا بازیکن‌های یک تیم در زمین بازی انجام می‌دهند، اشاره‌هایی که برای دیگران بی‌مفهوم اما برای این افراد دارای معنای خاصی هستند. شباهت این روش کدگذاری در این است که قانون و قاعده خاصی در آن وجود ندارد و همه چیز به شکل قراردادی در بین طرف‌های استفاده کننده تعیین می‌شود. در این روش قسمت عمده متن بی‌ارزش بوده و تنها کلیدواژه هایی که تعیین شده‌اند دارای معنی هستند. برای نمونه شما می‌خواهید با یکی از دوستانتان قرار ملاقات‌هایی داشته باشید که تمایل ندارید دیگران از آن آگاه شوند. می‌توانید با او چنین تعیین کنید که هرگاه نامی از کتاب یا امثال آن آوردید، در آن جمله مکان قرار را نیز خواهید گفت. حال می‌توانید با خیال راحت در جمع دوستان خود بپرسید: «بچه‌ها کسی میدونه تو میدون

1. Over all of Spain, the sky is clear

2. Idiot Code

انقلاب کتاب‌های کدگذاری به زبان پارسی هم پیدا میشن یا نه؟» و روزی دیگر می‌توانید در یک جمع تعریف کنید که خواهرتان از شما خواسته که با او به یک طلافروشی در میدان ولی عصر بروید ولی شما چون مشغول مطالعه یک کتاب در مورد روش‌های کدگذاری بوده‌اید و غرق آن شده‌اید درخواست او را رد کرده‌اید. نمونه‌های بسیاری وجود دارند که می‌توانید با آسودگی خاطر در هر جمعی بیان کنید و مطمئن باشید که تنها دوست مورد نظرتان مفهوم گفته‌های شما را درک می‌کند. یکی از نمونه‌های استفاده از کد احمق در حمله‌های ۱۱ سپتامبر در ایالات متحده بود که افراد درگیر در این حمله در پیام‌های خود از این روش استفاده کرده بودند.

۲-۲ تحلیل کدگذاری

کشف رمز پیام‌های کدگذاری شده بسیار دشوار است، برای درک آن بپندارید که متنی از یک زبان بیگانه را مقابل شما قرار می‌دهند و سپس از شما می‌خواهند با وجود نداشتن هیچگونه آشنایی با آن زبان و بدون داشتن واژه‌نامه آن را ترجمه کنید، با این وجود انجام آن ناممکن نیست و گزینه‌هایی برای این کار وجود دارند. یکی از راه‌های تحلیل کدگذاری توجه به ساختارهای زبانی متن است، در هر زبانی تعدادی از واژه‌ها بیش از دیگران مورد استفاده قرار می‌گیرند، برای نمونه در زبان انگلیسی واژه‌های a و The از واژه‌های پرتکرار محسوب می‌شوند یا مثلاً در تلگراف آخر پیام‌ها معمولاً با واژه‌های مشخصی پایان می‌یابند. یکی دیگر از گزینه‌های پیش رو به دست آوردن اطلاعات است، اطلاعاتی که احتمال حضور در متن پیام را دارند، اطلاعاتی همچون زمان و تاریخ ارسال پیام، مبدأ و مقصد آن، رویدادهایی که انجام گرفته و یا قرار است در آینده‌ای نزدیک انجام بگیرند و دیگر موارد این‌چنینی. برای نمونه اگر در پیام‌هایی که از یک پایگاه نظامی ارسال می‌شود موردی یافت شد که در هیچ یک از پیام‌های دیگر این ارتش در مکان‌های دیگر دیده نشد، به احتمال زیاد آن علامت به معنای فرمانده آن پایگاه خواهد بود.

۳

رمزنگاری

امروزه از رمزنگاری بیش از هر روش دیگری برای حفاظت از اطلاعات استفاده می‌شود و می‌توان دلیل آن را ساختار نوین ارتباطی و انتقال اطلاعات دانست، به شکلی که دیگر تنها جاسوس‌ها و سازمان‌های اطلاعاتی نیستند که به حفظ امنیت داده‌هایشان بها می‌دهند، بلکه تقریباً همه انسان‌ها اطلاعاتی دارند که نیازمند حفظ امنیت آن‌ها هستند. اطلاعاتی همچون گذرواژه‌های کارت‌های اعتباری برای انجام کارهای بانکی و نیز انجام خریدهای اینترنتی، گذرواژه رایانشانی و بسیاری وب‌گاه‌های خدماتی که در آن‌ها عضویت داریم و ...

در این میان بسیاری از این داده‌ها را نمی‌توان پنهان‌نگاری نمود، برای نمونه احتمال اینکه هر شخص بالای ۲۰ سال دارای یک کارت اعتباری بانکی و یا یک سیم‌کارت باشد بسیار بالاست، پس کاملاً روشن است که این افراد برای انتقال پول و شارژ سیم کارت و استفاده از شبکه‌های ارتباطی روی گوشی همراه خود اطلاعات منحصر به فردی که بازگوی هویت آن‌هاست را منتقل می‌کنند. از سویی دیگر استفاده از کدگذاری هم اصلاً کار

مناسبی به نظر نمی‌رسد، بایستی برای هر فرد یک کتاب کد جداگانه تهیه شود زیرا کتاب کدی که در اختیار همه باشد دیگر امنیتی ندارد!

از همین روی تنها گزینه مناسب که امکان استفاده گسترده را در شرایط کنونی دارد، رمزنگاری داده‌ها است. همان‌گونه که در بخش پیشین گفته شد، در رمزنگاری بر خلاف کدگذاری کار بر روی حروف (و پس از ورود رایانه‌ها بر روی بیت‌ها) انجام می‌گیرد. تفاوت دیگر آن‌ها این است که روش کار برای همه حروف یکسان است، یعنی برخلاف کدگذاری که برای هر واژه به شکل متفاوتی کلمه کد ساخته میشد، در رمزنگاری از یک الگوریتم ثابت برای تبدیل تمام حروف استفاده می‌شود. این کار هرچند امکان شکستن رمز را افزایش می‌دهد اما نقطه ضعف کدگذاری را برطرف می‌کند، دیگر با یک کتاب بزرگ سروکار نداریم و **کلید** رمزنگاری ما داده بسیار کوچکی است که به راحتی می‌تواند پنهان یا منتقل شود. کلید رمزنگاری درواقع همان اطلاعاتی است که به وسیله آن می‌توان پیام را رمزگذاری یا رمزگشایی کرد. نوع کلید می‌تواند در امنیت سیستم رمزنگاری بسیار مؤثر باشد، میزان بلندی یا کوتاهی و نیز میزان پیچیدگی یا سادگی کلید نقشی مهم در افزایش یا کاهش امنیت پیام دارد.

هر کسی که در زمینه رمزنگاری مطالعه کرده باشد به احتمال زیاد نام اصول‌کرکهف[1] را شنیده است، به ویژه معروف‌ترین آن‌ها (اصل دوم) که امروزه در بسیاری از سیستم‌های امنیتی درنظر گرفته شده است. رعایت کردن این اصول می‌تواند ضریب امنیتی سامانه را بسیار بهبود بخشد و نیاز است که در زمان ساخت یک سامانه رمزنگاری توجه ویژه‌ای به آن‌ها داشته باشیم. (در پیوست B لیست کامل این اصول را مشاهده نمایید.)

یکی از روش‌هایی که می‌تواند در افزایش امنیت سامانه‌ها تأثیر چشمگیری داشته باشد

1. Kerckhoffs's Principle

استفاده همزمان از چند روش متفاوت است، بدین گونه که یا از رمزنگاری و پنهان‌نگاری به شکل ترکیبی استفاده کنیم و یا از رمزنگاری چندگانه بهره ببریم، در این روش از چند لایه رمزنگاری استفاده می‌شود که کلیدها ممکن است یکسان و یا متفاوت باشند.

۳-۱ الگوریتم‌های رمزنگاری

۳-۱-۱ رمزنگاری جانشینی

یکی از نخستین رمزنگاری‌هایی که مورد استفاده بشر بوده است با نام رمزنگاری جانشینی[1] شناخته می‌شود، در این‌گونه از رمزنگاری به جای هرکدام از حروف یک نماد قرار می‌گیرد. این نماد می‌تواند هرچیزی باشد، از یک حرف دیگر از همان الفبا گرفته تا نمادی ساختگی توسط خودتان. البته امروزه دیگر تمام انواع رمزنگاری جانشینی به سادگی شکسته می‌شوند و برای استفاده درمورد پیام‌های مهم پیشنهاد نمی‌شوند.

حالت ساده این نوع رمزنگاری به شکل تک الفبایی است، یعنی از یک الفبا برای این جانشانی‌ها بهره گرفته می‌شود و بدون توجه به مکان قرارگیری حروف یک شکل مشخص برای هرکدام درنظر گرفته می‌شود. در زیر نمونه‌هایی چند از این‌گونه را بررسی می‌کنیم:

رمز سزار: شاید بتوان رمز سزار را ساده‌ترین گونه رمزنگاری جانشینی دانست. در این روش هر حرف الفبا با حرف سوم پس از خودش جایگزین می‌شود، برای نمونه به جای حرف «ک» حرف «م» قرار می‌گیرد. برای اینکه بتوانید از این روش استفاده کنید ابتدا حروف الفبا را در یک خط بنویسید، سپس در زیر آن از چهارمین حرف شروع کنید به

1. Substitution Cipher

نوشتن دوباره حروف الفبا و پس از رسیدن به آخرین حرف، سه حرف ابتدایی را وارد کنید:

ا	ب	پ	ت	ث	ج	چ	ح	خ	د	ذ	ر	ز	ژ	س	ش
ت	ث	ج	چ	ح	خ	د	ذ	ر	ز	ژ	س	ش	ص	ض	ط

ص	ض	ط	ظ	ع	غ	ف	ق	ک	گ	ل	م	ن	و	ه	ی
ظ	ع	غ	ف	ق	ک	گ	ل	م	ن	و	ه	ی	ا	ب	پ

حال حروف واژه‌ای که می‌خواهید رمزنگاری کنید را در ردیف بالا پیدا کنید و برابر آن در ردیف پایین را به جایش قرار دهید، برای نمونه نام من پس از رمزنگاری به شکل «هذضی مسهپ» درخواهد آمد. برای رمزگشایی نیز به سادگی عکس این کار انجام دهید، یعنی حروف پیام رمزنگاری شده را در ردیف پایینی پیدا کرده و حروف برابر آن در ردیف بالایی را جایگزین نمایید.

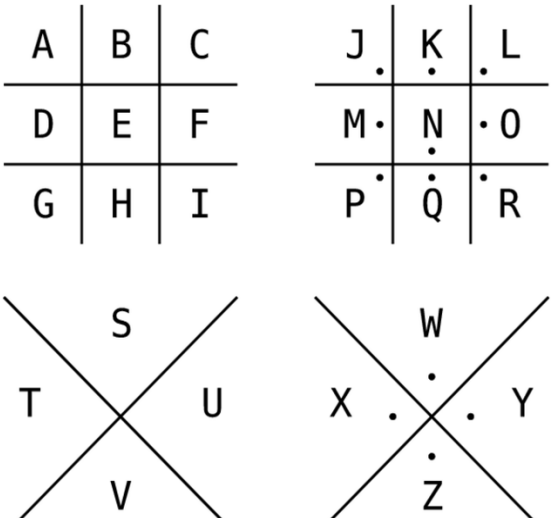
رمز سزار را می‌توان به شکل‌های مختلف، با میزان انتقال‌های متفاوت و در جهت‌های گوناگون استفاده کرد. از روش‌هایی که از رمز سزار گرفته شده‌اند می‌توان به ROT13 ROT5 و ROT47 اشاره کرد که هرکدام از آن‌ها کاربرد خاص خود را دارند.

رمز خوکدانی[1]: این رمز که الفبایی هندسی دارد و از نمادهای هندسی برای رمزنگاری استفاده می‌کند، همچون نامش روشی عجیب دارد و برای استفاده از آن باید از شکل‌های ۱ و ۲ استفاده کنیم. از آن‌جایی که نمادهای ایجاد شده توسط این روش برابر تعداد

1. Pigpen Cipher

حروف الفبای انگلیسی است نمی‌توان از آن در هر زبانی (همچون پارسی) استفاده نمود. نمونه‌ای از رمزنگاری به این روش را می‌توانید در پایین سمت راست شکل ۲ مشاهده کنید. رمز نیوآرک[1] از این روش توسعه پیدا کرده ولی به جای نقطه‌ها از یک تا سه خط کوتاه استفاده می‌کند، از این رو می‌توان با ایجاد تفاوت در میزان کشیدن این خطوط در زمان نوشتن این توهم را در بیننده ایجاد کرد که با الفبای گسترده‌تری روبرو است. از دیگر رمزنگاری‌هایی که همچون این روش از نمادهای هندسی مشابه استفاده می‌کنند می‌توان به رمز معبد[2] و رمز الیان[3] اشاره کرد.

شکل ۱- روش بدست آوردن الفبای رمز خوکدانی

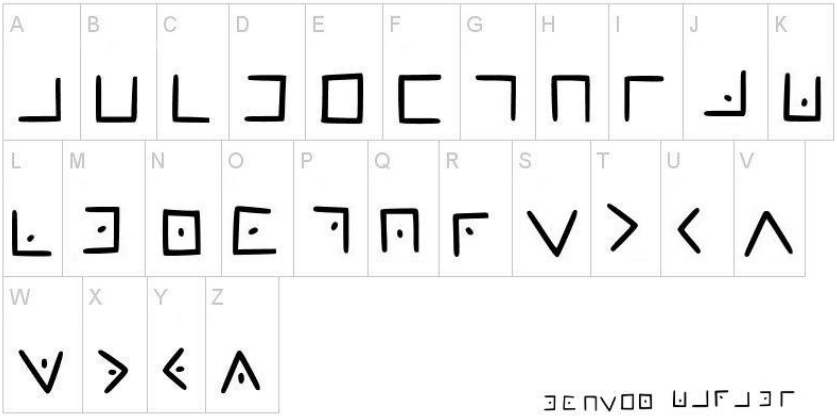


1. Newark Cipher

2. Templar Cipher

3. The Elian Script

شکل ۲- الفبای رمز خوکدانی پس از مرتب کردن و یک نمونه رمز ایجاد شده با این روش



جانشینی هم‌آهنگ[1]: از راه‌های مقابله با رمزنگاری‌های جانشینی می‌توان استفاده از روش **تحلیل فراوانی** را نام برد، در این روش که در بخش تحلیل رمزنگاری به آن خواهم پرداخت می‌بینیم که می‌توان با توجه به میزان پراکندگی حروف، متن را رمزگشایی کرد. یکی از روش‌هایی که باعث افزایش امنیت رمزنگاری جانشینی می‌شود جانشینی هم‌آهنگ است، در این حالت میزان توزیع فراوانی تا حد زیادی متعادل خواهد شد که تحلیل رمزنگاری را بسیار مشکل‌تر خواهد کرد. روش کار به این شکل است که برای هر حرف بیش از یک برابر درنظر گرفته می‌شود، به گونه‌ای که هرچه فراوانی یک حرف بیشتر باشد تعداد برابر بیشتری برایش تعریف می‌شود. این‌گونه متنی متعادل و یک‌دست خواهیم داشت که نمی‌توان به راحتی آن را رمزگشایی کرد. از آن‌جایی که در این روش به تعداد حرف بیشتری از الفبای زبان اصلی نیازمندیم، یا از اعداد استفاده می‌کنند و یا با تغییر

1. Homophonic Substitution

حروف الفبا همچون وارونه کردن و امثال آن به تعداد حروف مورد نیاز دست پیدا می‌کنند.

جانشینی چند الفبایی[1]: همان‌گونه که پیشتر گفته شد در حالت تک الفبایی برای هر حرف برابری مشخص تعریف می‌شود، اما در این حالت هرحرف با توجه به جایگاهی که در متن دارد می‌تواند برابرهای گوناگونی داشته باشد. دیوید کان[2] در کتابش با عنوان رمزشکنان: داستان محرمانه نویسی[3] که در سال ۱۹۶۷/ ۱۳۴۵ منتشر شد، ادعا می‌کند که لئون باتیستا آلبرتی[4] این روش را ابداع کرده و او را پدر رمزنگاری غربی می‌خواند. هرچند که به گفته این تاریخ‌دان، آلبرتی نخستین فرد غربی بوده که تحلیل رمز را توصیف کرده است اما بدون شک ابویوسف کندی[5] چندین قرن پیش از او با چنین روشی آشنا بوده است، این دانشمند عراقی در زمینه‌های بسیاری همچون ریاضیات، فیزیک و فلسفه صاحب نظر بود. نویسنده بریتانیایی سایمون سینگ[6] در کتاب خود با نام کتاب رمز: دانش محرمانگی از مصر باستان تا رمزنگاری کوانتومی[7] با ارایه و بررسی صفحه نخست کتاب دست نویس کندی (شکل ۳) با عنوان درباب رمزگشایی پیام‌های رمزنگاری شده[8] آن را کهن‌ترین توصیف شناخته شده تحلیل رمز به روش تحلیل فراوانی عنوان می‌کند. هنوز

---

1. Polyalphabetic Substitution

2. David Kahn

3. The Codebreakers – The Story of Secret Writing

4. Leon Battista Alberti

۵. أبو یوسف یعقوب بن إسحاق الکندی

6. Simon Lehna Singh

7. The Code Book: The Science of Secrecy from Ancient Egypt to Quantum Cryptography

۸. فی فک رسائل التشفیر

شکل ۳- صفحه نخست کتاب دست نویس ابویوسف کندی

۵۰ سال از مرگ آلبرتی نگذشته بود که اولین کتاب چاپ شده در زمینه رمزنگاری نوشته دانشمند آلمانی یوهانس تریتمیوس[1] منتشر شد، این کتاب که شش کتاب چندنگاری[2] نام داشت روشی نوین در رمزنگاری چند الفبایی را توصیف می‌کند که برای هر حرف یک کلید جداگانه در نظر می‌گیرد.

1. Johannes Trithemius

2. Polygraphiae Libri Sex

ایده پشت این روش یک مربع است که هر ضلع آن به تعداد حروف الفبای زبان مورد استفاده تقسیم شده است، سپس حروف را به گونه‌ای می‌نویسیم که در هر ردیف نسبت به ردیف بالایی یک خانه به چپ حرکت کرده باشند و ردیف اول با همان ترتیب اصلی حروف الفبا و بدون جابه‌جایی نوشته می‌شود. می‌توانید نمونه‌ای از این مربع که براساس زبان لاتین نوشته شده است را در صفحه ۴۶۳ کتاب یوهانس مشاهده نمایید. (شکل ۴)

برای رمزنگاری به روش رمز تریتمیوس[1] حرف اول پیام را در ردیف اول پیدا کرده و سپس خانه زیرین آن در ردیف دوم (که یک خانه به چپ انتقال دارد) را جایگزین می‌کنیم و نیز برای حرف دوم پیام جایگزین آن را در ردیف سوم قرار می‌دهیم.

به عبارت دیگر با توجه به شماره هر حرف در متن به همان اندازه شمرده و حرف پس از آن را در الفبا قرار می‌دهیم، برای نمونه در این روش «محسن» به شکل «ندضا» نوشته می‌شود. یکی از ایرادهای اصلی این روش ثابت بودن کلید آن است، درواقع کلید و الگوریتم رمزنگاری در هم آمیخته هستند که به آن **کلید خودکار** گفته می‌شود، پس اگر شخصی پی ببرد یا حتی گمان برد که نامه‌ای به این روش رمزنگاری شده است دیگر به راحتی می‌تواند آن را رمزگشایی نماید، این مشکل از رعایت نکردن اصل دوم کرکهف ناشی می‌شود. برای حل این مشکل رمز ویژنر[2] ایجاد شد که تنها تفاوتی که با رمز تریتمیوس دارد در کلید آن است، در اینجا دیگر کلید ثابت نبوده و واژه‌ای قراردادی بین طرفین بود. روش کار هم به این صورت بود که از تطبیق دادن حروف پیام در سطر بالایی و حروف کلید در ستون سمت چپ به حروف رمزشده می‌رسیدند، برای نمونه واژه «محسن» با کلید «کرمی» به شکل رمزشده «ظطدم» درمی‌آید. به نوعی می‌توان رمز تریتمیوس را یک رمز ویژنر با کلید حروف الفبا (ابپتثجچحخدذر....منوهی) دانست.

1. Trithemius cipher
2. Vigenère cipher

۳-۱-۲ رمزنگاری جابه‌جایی

یکی دیگر از راه‌های رمزنگاری با نام رمزنگاری جابه‌جایی[1] شناخته می‌شود. برخلاف رمزنگاری جانشینی که در آن ساختار حروف دگرگون می‌شود، در اینجا حروف دست نخورده باقی می‌مانند اما دچار جابه‌جایی می‌شوند. به روش‌های گوناگونی می‌توان حروف را در یک متن جابه‌جا نمود به گونه‌ای که دیگر متن اصلی قابل شناسایی و خواندن نباشد، در زیر چند نمونه از این روش‌ها را توضیح خواهم داد:

رمزنگاری ریلی[2]: برای رمزنگاری به این شیوه ابتدا تعدادی خط (که در اصطلاح به آن‌ها ریل گفته می‌شود) را به صورت افقی در زیر یکدیگر رسم می‌کنیم و سپس حروف پیام را از بالاترین ریل به سمت پایین می‌نویسیم، به گونه‌ای که در هر ریل حرف موردنظر را جلوتر از حرف پیشین (در ردیف بالایی آن) بنویسیم.

زمانی که به آخرین ریل رسیدیم همین حرکت را به سمت بالا تکرار می‌کنیم و این روند را ادامه می‌دهیم تا زمانی که پیام کامل نوشته شود، حال برای به دست آوردن پیام رمزنگاری شده بایستی متن را از ابتدای ریل بالایی به شکل افقی سرهم کنیم و به نوبت همه ریل‌ها را به همین شیوه تا آخرین حرف پایین‌ترین ریل ادامه دهیم. برای نمونه «محسن کرمی» به شیوه زیر رمزنگاری می‌شود:

م . . . ک . . .

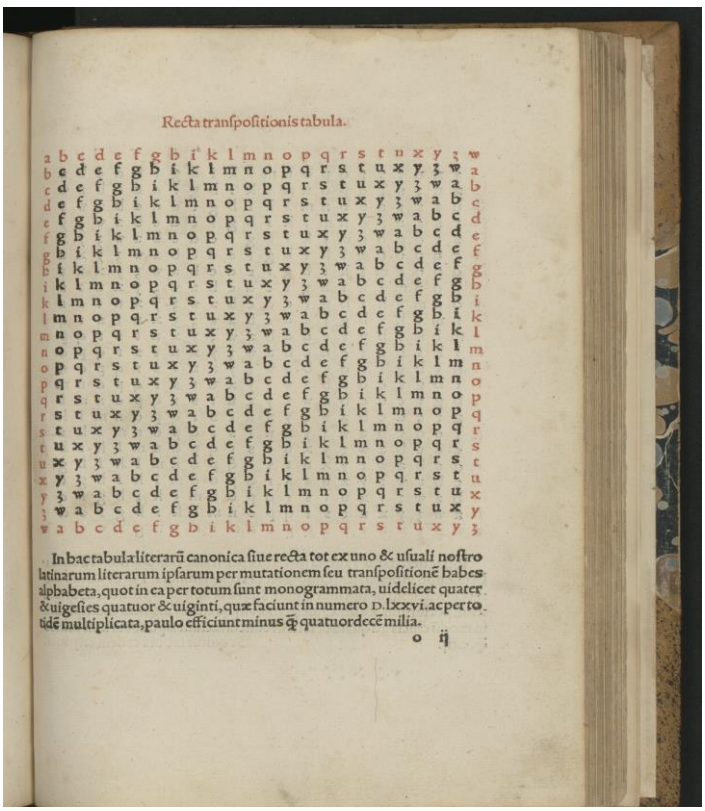
. ح . ن . ر . ی

. . س . . . م .

1. Transposition Cipher

2. Rail Fence Cipher

شکل ۴- صفحه ۴۶۳ کتاب یوهانس تریتمیوس



Recta transpositionis tabula.

a	b	c	d	e	f	g	h	i	k	l	m	n	o	p	q	r	s	t	u	x	y	z	w	
b	c	d	e	f	g	h	i	k	l	m	n	o	p	q	r	s	t	u	x	y	z	w	a	
c	d	e	f	g	h	i	k	l	m	n	o	p	q	r	s	t	u	x	y	z	w	a	b	
d	e	f	g	h	i	k	l	m	n	o	p	q	r	s	t	u	x	y	z	w	a	b	c	
e	f	g	h	i	k	l	m	n	o	p	q	r	s	t	u	x	y	z	w	a	b	c	d	
f	g	h	i	k	l	m	n	o	p	q	r	s	t	u	x	y	z	w	a	b	c	d	e	
g	h	i	k	l	m	n	o	p	q	r	s	t	u	x	y	z	w	a	b	c	d	e	f	
h	i	k	l	m	n	o	p	q	r	s	t	u	x	y	z	w	a	b	c	d	e	f	g	
i	k	l	m	n	o	p	q	r	s	t	u	x	y	z	w	a	b	c	d	e	f	g	h	
k	l	m	n	o	p	q	r	s	t	u	x	y	z	w	a	b	c	d	e	f	g	h	i	
l	m	n	o	p	q	r	s	t	u	x	y	z	w	a	b	c	d	e	f	g	h	i	k	
m	n	o	p	q	r	s	t	u	x	y	z	w	a	b	c	d	e	f	g	h	i	k	l	
n	o	p	q	r	s	t	u	x	y	z	w	a	b	c	d	e	f	g	h	i	k	l	m	
o	p	q	r	s	t	u	x	y	z	w	a	b	c	d	e	f	g	h	i	k	l	m	n	
p	q	r	s	t	u	x	y	z	w	a	b	c	d	e	f	g	h	i	k	l	m	n	o	
q	r	s	t	u	x	y	z	w	a	b	c	d	e	f	g	h	i	k	l	m	n	o	p	
r	s	t	u	x	y	z	w	a	b	c	d	e	f	g	h	i	k	l	m	n	o	p	q	
s	t	u	x	y	z	w	a	b	c	d	e	f	g	h	i	k	l	m	n	o	p	q	r	
t	u	x	y	z	w	a	b	c	d	e	f	g	h	i	k	l	m	n	o	p	q	r	s	
u	x	y	z	w	a	b	c	d	e	f	g	h	i	k	l	m	n	o	p	q	r	s	t	
x	y	z	w	a	b	c	d	e	f	g	h	i	k	l	m	n	o	p	q	r	s	t	u	
y	z	w	a	b	c	d	e	f	g	h	i	k	l	m	n	o	p	q	r	s	t	u	x	
z	w	a	b	c	d	e	f	g	h	i	k	l	m	n	o	p	q	r	s	t	u	x	y	
w	a	b	c	d	e	f	g	h	i	k	l	m	n	o	p	q	r	s	t	u	x	y	z	

In hac tabula literarũ canonica siue recta tot ex uno & usuali nostro latinarum literarum ipsarum per mutationem seu transpositionẽ habes alphabeta, quot in ea per totum sunt monogrammata, uidelicet quater & uigesies quatuor & uiginti, quæ faciunt in numero D.lxxvi. ac per totidẽ multiplicata, paulo efficiunt minus q̃ quatuordecẽ milia.

o ij

حال اگر این متن را به شکل افقی بخوانیم به واژه «مکحنریسم» می‌رسیم. با توجه به الگوریتم توضیح داده شده می‌توان با تعریف کلیدهای متفاوت، هربار به روش نوینی از این شیوه استفاده نمود. کلید رمزنگاری در اینجا تعداد ریل‌های تعیین شده است، برای نمونه اگر به جای کلید ۳ از ۲ استفاده می‌کردیم به متن «مسکمحنری» می‌رسیدیم.

در متن‌های بلند اگر بخواهیم تمام متن را پشت سرهم بنویسیم قابل خواندن نخواهد بود و از سویی نیز اگر همانند متن اصلی عمل کنیم راهنمایی بزرگی برای حدس زدن کلید خواهد بود، از همین روی به شکل قراردادی رمزنگاران زمان نوشتن این متن‌های رمز شده آن‌ها را به گروه‌های ۵ حرفی تقسیم می‌کنند تا هم قابل خواندن بوده و هم از محتوای متن اصلی سرنخی بدست ندهند.

روشی بسیار مشابه با رمزنگاری ریلی در یونان باستان مورد استفاده بوده است که اسکتلی[1] نامیده میشد. این وسیله شامل یک استوانه بود که روبانی به دور آن پیچیده شده بود، سپس متن پیام روی آن روبان پیچیده شده نوشته میشد که پس از باز شدن قابل خواندن نبود.(شکل ۵) برای اینکه دوباره به پیام دسترسی پیدا می‌کردیم بایستی روبان را دور یک استوانه با همان قطر استوانه اولیه می‌پیچاندیم، درواقع قطر استوانه نقش کلید رمزنگاری را داشت.

شکل ۵- روش رمزنگاری اسکتلی

1. Scytale

رمزنگاری مسیری[1]: در این شیوه متن پیام را در شبکه‌ای چهارگوش نوشته و سپس در یک مسیر دلخواه بازنویسی می‌کنند. مسیرهای ممکن برای بازنویسی این متن در فصل نهم از مدارک انجمن متون رمزی آمریکایی[2] که سازمانی غیرانتفاعی در زمینه گسترش آگاهی در مورد رمزها و راه‌های شکستن آن‌ها می‌باشد، چنین بیان شده است: افقی، عمودی، افقی متناوب، عمودی متناوب، مورب، مورب متناوب، چرخش درونی در جهت عقربه‌های ساعت، چرخش درونی در خلاف جهت عقربه‌های ساعت، چرخش بیرونی در جهت عقربه‌های ساعت و چرخش بیرونی در خلاف جهت عقربه‌های ساعت، همچنین این سند انتخاب‌های گوناگونی که برای نقطه شروع حرکت در اختیار داریم را یادآوری می‌کند.

برای نمونه اگر بخواهیم «الگوریتم‌های مختلف رمزنگاری» را به این شیوه رمزنگاری کنیم و کلید آن را به شکل «حرکت مورب متناوب و آغاز با چرخش هم‌جهت با حرکت عقربه‌های ساعت» تعریف کنیم، به شکل (۶) می‌رسیم. من در این‌جا یک شبکه ۴ در ۶ را در نظر گرفته‌ام و نقطه شروع را بالا سمت چپ قرار داده‌ام، این را نیز در نظر داشته باشید که مسیر بایستی کامل گردد، یعنی تمام حروف در طول مسیر پیموده شده خوانده شوند.

* راهنمایی: از حرف الف در سمت چپ آغاز کرده و ابتدا یک حرکت افقی و سپس مورب انجام دهید، آن‌گاه یک حرکت به سمت پایین و دوباره مورب و ...

حال اگر متن را به شکل افقی از سمت راست بازنویسی کنیم به عبارت «تیلاتمرگلخهوگفماانرییرزم» می‌رسیم.

1. Route Cipher

2. American Cryptogram Association

شکل ۶- یک نمونه شبکه ایجاد شده برای رمزنگاری پیام به روش مسیری

ت	ی	ل	ا
ت	م	ر	گ
ل	خ	ه	و
گ	ف	م	ا
ا	ن	ر	ی
ی	ر	ز	م

۳-۱-۳ پد یکبار مصرف

یکی از روش‌های بسیار ایمن رمزنگاری اطلاعات پد یکبار مصرف[1] است، درواقع چنین گفته می‌شود که اگر این روش به درستی انجام گیرد شکستن رمز غیرممکن خواهد بود. در این شیوه رمزنگاری دو نسخه از پد تهیه خواهد شد که در اختیار فرستنده و گیرنده پیام خواهند بود، وجود هر کپی اضافه‌ای از این پد امنیت سامانه را به شدت تحت تأثیر قرار خواهد داد، زیرا با کشف آن توسط هر فرد غیرمجاز شکستن رمز قطعی خواهد بود. هر پد درواقع قطعه‌ای از داده‌های تصادفی است که طول آن با طول پیامی که رمزنگاری می‌شود برابر است. در اینجا مهم‌ترین نکته تصادفی بودن داده‌های پد است ـ تصادفی بودن به معنای واقعی کلمه ـ زیرا در این سامانه معمولاً گفته می‌شود که امنیت یا ۱۰۰

1. One-Time Pad

درصد است یا صفر درصد و این تا حد زیادی به میزان تصادفی بودن پد برمی‌گردد.

یک سند ثبت اختراع در ایالات متحده که توسط گیلبرت ورنام[1] در سال ۱۹۱۹/ ۱۲۹۸ به ثبت رسیده است نشان می‌دهد که روش پد یکبار مصرف اختراع ورنام است ولی با استناد به مقاله استیون بلووین[2] استاد علوم رایانه دانشگاه کلمبیا که در سال ۲۰۱۱/ ۱۳۹۰ منتشر شده است می‌توان پی برد که فرانک میلر[3] ۳۵ سال پیش از ورنام ایده پد یکبار مصرف را توضیح داده است و درواقع خالق اولیه این روش میلر است.

نکته دیگر نحوه انتقال پد است. بسیار مهم است که پد را از راه ایمنی منتقل کنیم، درواقع اگر از هر روش رمزنگاری دیگری برای تبادل پد استفاده کنید امنیت سامانه شما به میزان امنیت آن روش کاهیده خواهد شد. از این رو معمولاً از روش‌های بسیار ایمن برای انتقال پد استفاده می‌شود، برای نمونه پد را در یک فلاپی دیسک به شکل حضوری منتقل می‌کنند.

می‌توان پد یکبار مصرف را به یک منبع پارازیت تشبیه کرد که متن پیام را در خود پنهان کرده است، در این حالت تنها کسانی می‌توانند به پیام دسترسی پیدا کنند که کپی پارازیت را در اختیار داشته باشند و بتوانند به وسیله آن، منبع پارازیت حاوی پیام را فیلتر کنند و به متن پیام برسند.

از مهم‌ترین هشدارهایی که باید به آن توجه جدی داشت این است که به هیچ‌عنوان نباید از یک پد بیش از یکبار استفاده کرد. در این مورد بسیار به یکبار مصرف بودن پد اهمیت دهید و همچون بسیاری محصولات یکبار مصرف دیگر به سادگی از کنار آن نگذرید!

---

1. Gilbert Sandford Vernam

2. Steven M. Bellovin

3. Frank Miller

تا زمانی که تنها یکبار از پد استفاده کنید امنیت فوق‌العاده بالایی را خواهید داشت و با هیچ یک از روش‌های کشف رمز نمی‌توانند آن را رمزگشایی کنند، اما اگر تنها برای بار دوم از آن استفاده کنید امنیت آن را تا نزدیک به صفر درصد پایین خواهید آورد. نمونه‌ی بسیار خوبی که در این زمینه می‌توان بیان کرد به جنگ جهانی دوم برمی‌گردد، دانشمند انگلیسی و یکی از ماموران سابق آژانس امنیتی انگلستان[1]، پیتر رایت[2] در کتابی با عنوان شکارچی جاسوس: خودزندگی‌نامه‌ای صادقانه از یک افسر ارشد اطلاعاتی[3]، داستانی را به این شکل بیان می‌کند که سازمان اطلاعاتی اتحاد جماهیر شوروی در زمان جنگ جهانی دوم پس از گذشت سال‌ها از پخش اولیه پدهای یکبار مصرف در بین ماموران خود در بریتانیا، دوباره آن‌ها را مورد استفاده قرار داد.

ماموران بریتانیا متوجه وجود الگویی در پیام‌های رمزی شدند و سپس در بایگانی پیام‌های رمزی که در طول سال‌ها کشف شده بودند به جستجو پرداختند و توانستند تعدادی از پیام‌ها را رمزگشایی کنند. جزییات بیشتر ماجرا را می‌توان در سندی مشاهده کرده که آژانس امنیت ملی ایالات متحده آمریکا[4] پس از سال‌ها محرمانه ماندن منتشر کرد، در آن سند آمده است که سازمان اطلاعاتی شوروی در سال‌های ۱۹۴۲/ ۱۳۲۱ تا ۱۹۴۴/ ۱۳۲۳ از کتاب کد یکسانی برای رمزنگاری استفاده نمود و از این رو برخی برگه‌های پدهای یکبار مصرف دوباره استفاده می‌شدند.

این فرصت بسیار خوبی بود که نباید از دست می‌رفت، اما به دلیل آنکه از صفحه‌های کپی بسیار کمی در پدهای سال ۱۹۴۲ استفاده شده بود تعداد خیلی کمی از آن‌ها را توانستند

1. MI5: Military Intelligence, Section 5

2. Peter Maurice Wright

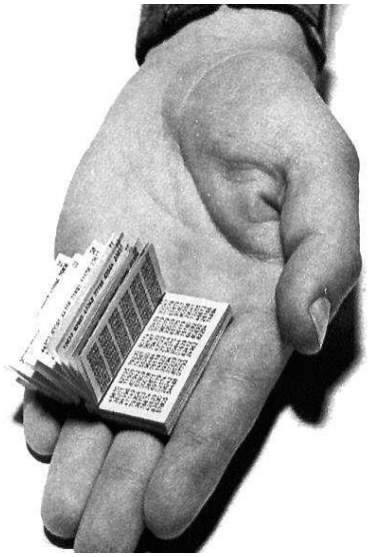
3. Spycatcher: The Candid Autobiography of a Senior Intelligence Officer

4. NSA: National Security Agency

رمزگشایی کنند. در این سند چنین آمده است که: «وضعیت در سال ۱۹۴۳/ ۱۳۲۲ بسیار مطلوب‌تر بود و در ۱۹۴۴/ ۱۳۲۳ حتی بهتر از آن و در نتیجه نرخ موفقیت بهبود پیدا کرد.»

همان‌گونه که گفته شد سخت‌ترین و کلیدی‌ترین قسمت استفاده از پدهای یکبار مصرف ایجاد داده‌های تصادفی است، اما از چه منابعی می‌توان چنین داده‌هایی را ایجاد کرد؟ یکی از گزینه‌های مورد پسند، استفاده از تابش پس زمینه کیهانی و تبدیل آن توسط یک تابع یک‌طرفه همچون MD5 می‌باشد. توجه کنید که در این روش استفاده از روش‌های رمزنگاری همچونMD5 برای متعادل کردن ویژگی‌های آماری داده‌ها است و نه رمزنگاری آن‌ها، از این روی استفاده از روش‌هایی که امنیت کمتری نسبت به پد دارند مشکلی ایجاد نمی‌کند. گزینه دیگر استفاده از واپاشی هسته‌ای است که یک ویژگی کاملاً تصادفی ماده است.

یک پد یکبار مصرف روسی که توسط
آژانس امنیتی انگلستان کشف شده است



۳-۲ رمزنگاری در دنیای دیجیتال

امروزه در بیش‌تر مواردی که از امنیت اطلاعات سخن به میان می‌آید منظور رمزنگاری‌های مدرن و پیشرفته است، رمزنگاری‌هایی که به وسیله رایانه‌ها و با سرعت و دقت فوق‌العاده بالا انجام می‌گیرند. بنا به دلایلی که در بخش‌های پیشین گفته شد پنهان‌نگاری را می‌توان تنها در موارد خاصی استفاده نمود و کدگذاری نیز به دلیل ساختار خود نمی‌تواند به شکل گسترده مورد استفاده قرار گیرد، پس بیشترین استفاده از راه رمزنگاری خواهد بود که با توجه به اصل دوم کرکهف بسیاری از رمزنگاری‌ها نیز کنار گذاشته می‌شوند. اما در این بین تعدادی که باقی مانده‌اند نیز بایستی به راحتی و با تحلیل‌های آماری یا روش‌های موجود دیگر شکسته نشده و از مقاومت بالایی برخوردار باشند، زیرا با وجود رایانه‌ها می‌توان تحلیل‌های سنگین و گسترده را در زمانی کوتاه روی رمزها انجام داد و نیاز به رمزنگاری‌هایی پیشرفته است که بتوانند در مقابل این شیوه‌ها مقاومت کنند. در ادامه برجسته‌ترین شیوه‌های رمزنگاری امروزی را بررسی خواهم کرد:

۳-۲-۱ رمزنگاری کلید متقارن

یکی از روش‌های رمزنگاری که ویژگی‌های موردنیاز دنیای مدرن را دارا بوده و توانسته به استانداردی برای رمزنگاری تبدیل شود رمزنگاری کلید خصوصی[1] می‌باشد که با نام رمزنگاری کلید متقارن[2] نیز شناخته می‌شود. همان‌گونه که انتظار می‌رود این شیوه از اصل دوم کرکهف پیروی کرده و امنیت آن به جای پنهان ماندن الگوریتم به محرمانه بودن کلید وابسته می‌باشد. درواقع الگوریتم روش‌هایی که توسط سازمان‌های امنیتی به

1. Private-key Encryption

2. Symmetric-key Cryptography

عنوان استاندارد تأیید می‌شوند، به شکل عمومی منتشر شده و در طول سال‌ها توسط کارشناسان بسیاری بررسی و رفع اشکال می‌شوند، از این روی تقریباً دارای الگوریتمی بدون اشکال خواهند بود و بایستی روی حفظ امنیت کلید تمرکز داشت. الگوریتم‌هایی که در روش‌های امروزی و توسط رایانه‌ها انجام می‌گیرد پیچیده و گسترده هستند و به سادگی قابل فهم و اجرا توسط انسان نیستند و به شکل تخصصی به مهندسین امنیت اطلاعات آموزش داده می‌شوند.

در رمزنگاری کلید متقارن از یک کلید برای رمزگذاری و رمزگشایی استفاده می‌شود، از این رو فرستنده و گیرنده هردو بایستی دارای کلید یکسانی باشند. از آن‌جایی که الگوریتم‌های رمزگذاری و رمزگشایی کاملاً عمومی و شناخته شده هستند پس در صورتی که کلید محرمانگی کافی را نداشته و لو برود، امنیت سامانه به کلی ازهم فروپاشیده و پیام‌ها به راحتی بازیابی می‌شوند. مهم‌ترین نقطه ضعف رمزنگاری کلید خصوصی فراهم نکردن راهی ایمن برای انتقال کلید است که مانعی بزرگ محسوب می‌شود، این مشکل تا زمانی‌که روش رمزنگاری‌کلیدنامتقارن[1] ایجاد شد حل نشده ماند.

در میان تمامی الگوریتم‌های رمزنگاری، الگوریتم کلید متقارن سریع‌ترین اجرا و پیاده‌سازی را چه در سخت‌افزار و چه نرم‌افزار دارا می‌باشد، از این رو گزینه‌ای بسیار مناسب برای رمزنگاری داده‌های با حجم بالا است. از سویی دیگر رمزنگاری کلید نامتقارن امنیتی بالاتر را ارایه می‌دهد که می‌توان با استفاده از آن، کلید خصوصی را با اطمینان خاطر منتقل کرد و به دلیل حجم پایین کلید نگرانی بابت سرعت عمل الگوریتم نیز وجود ندارد، پس می‌توان رمزنگاری‌های کلید متقارن و نامتقارن را مکمل یکدیگر دانست. چندی از الگوریتم‌های استاندارد کلید خصوصی را در زیر معرفی خواهم کرد:

1. Asymmetric key cryptography

DES که به معنای استاندارد رمزنگاری داده‌ها[1] می‌باشد، زمانی که در سال ۱۹۷۷/ ۱۳۵۶ به عنوان استانداردی برای رمزنگاری تأیید شد به شکل گسترده‌ای توسط دولت‌ها، بانک‌ها و نیز در تجارت مورد استفاده قرار گرفت و درواقع به عنوان مبنایی برای ارتباطات امن و مورد اعتماد به محبوب‌ترین الگوریتم کلید متقارن تبدیل شد. در این الگوریتم، کلید ۵۶ بیتی بوده (البته ۸ بیت توازن نیز به کلید اضافه خواهد شد) و توسط آن متن ۶۴ بیتی دریافت شده و متن رمزشده‌ای ۶۴ بیتی به دست خواهد آمد.

3DES: مدت کوتاهی پس از انتشار DES دو رمزنگار آمریکایی با نام‌های وایتفیلد دیفی[2] و مارتین هلمن[3] اندازه کوچک کلید ۵۶ بیتی این الگوریتم را مورد انتقاد قرار داده و پیشنهاد کردند که از این الگوریتم در حالت چندگانه استفاده شود. در حالت رمزنگاری سه گانه که ۳ کلید ۵۶ بیتی متفاوت دارد، استحکام رمزنگاری بسیار بیشتر می‌باشد، این الگوریتم به شکلی است که نتیجه چند رمزنگاری با کلیدهای متفاوت یک الگوریتم DES با کلیدی نوین نخواهد بود. (برای نمونه در رمز سزار اگر شما از رمزنگاری چندلایه با کلیدهای متفاوت استفاده کنید نتیجه نهایی خود یک رمز سزار با کلیدی نوین است، پس در نتیجه رمزنگاری چندگانه در آن حالت تأثیر زیادی در استحکام متن رمزشده نخواهد داشت.) بنا به گفته هانس دلفس[4] و هلموت کنیبل[5] در کتاب مقدمه‌ای بر رمزنگاری: اصول و کاربردها[6] بهترین حمله کاربردی شناخته‌شده درمقابله با الگوریتم DES حمله‌ی

1. DES: Data Encryption Standard

2. Bailey Whitfield Diffie

3. Martin Edward Hellman

4. Hans Delfs

5. Helmut Knebl

6. Introduction to Cryptography: Principles and Applications

جستجوی فراگیر[1] است، در این روش برای پیدا کردن کلید تمام حالت‌های ممکن آزموده خواهند شد. ضریب اطمینان این روش بسیار بالاست، به ویژه اگر توسط رایانه‌هایی انجام گیرد که به همین منظور طراحی و ساخته شده‌اند. برای نمونه در این کتاب می‌خوانیم که یک ابررایانه که طراحی ویژه‌ای داشته است، توانسته به کمک اتصال به شبکه‌ای از ۱۰۰ هزار رایانه (به واسطه درگاه اینترنت) پس از ۲۲ ساعت و ۱۵ دقیقه پردازش، کلید را بیابد.

AES: در ژانویه ۱۹۹۷/ ۱۳۷۵ مؤسسه ملی فناوری و استانداردها[2] پروسه‌ای را برای برگزیدن یک الگوریتم به عنوان استاندارد رمزنگاری پیشرفته[3] یاAES آغاز کرد. این مؤسسه از سازمان‌ها و افراد فعال در این زمینه در سراسر جهان درخواست کرد تا طرح‌های پیشنهادی خود را ارایه کنند، این الگوریتم‌ها بایستی کلیدهایی با اندازه‌های ۱۲۸، ۱۹۲ و ۲۵۶ بیتی را پشتیبانی می‌کردند. بنا به آنچه در کتاب هانس و هلموت گفته شده داوری این الگوریتم‌ها در دو مرحله انجام گرفته است، در مرحله اول از ۲۱ طرح پیشنهادی ۱۵ طرح برای نامزدی پذیرفته شدند و سپس این نامزدها در یک بحث همگانی مورد ارزیابی قرار گرفتند که از میان آن‌ها ۵ نامزد برای مرحله دوم برگزیده شدند. با نگاهی به بایگانی سایت رسمی مؤسسه ملی فناوری و استانداردها می‌توان دید که در مرحله اول دو کنفرانس و در مرحله دوم نیز یک کنفرانس برگزار شده است که بنا به آماری که از سوی این موسسه اعلام شده تنها در کنفرانس سوم بیش از ۲۵۰ نفر از ۲۵ کشور دنیا حضور داشته‌اند، در نهایت در اکتبر سال ۲۰۰۰/ ۱۳۷۹ الگوریتم **ریندال**[4] از میان آن‌ها به عنوان

1. Brute-Force Attack

2. National Institute of Standards and Technology

3. AES: Advanced Encryption Standard

4. Rijndael

استاندارد رمزنگاری پیشرفته برگزیده شد.

۳-۲-۲ رمزنگاری کلید نامتقارن

روش دیگری که امروزه به شکل گسترده‌ای همراه با رمزنگاری متقارن مورد استفاده قرار می‌گیرد با نام رمزنگاری کلید نامتقارن شناخته می‌شود که به آن رمزنگاری کلید عمومی[1] نیز گفته می‌شود.

این الگوریتم برخلاف کلید متقارن دارای کلیدهای متفاوتی برای رمزگذاری و رمزگشایی است، کلیدی که برای رمزگذاری استفاده می‌شود معمولاً در دسترس همگان قرار می‌گیرد (کلید عمومی) و توانایی رمزگشایی متن را ندارد و کلید دیگر که تنها در اختیار گیرنده مورد نظر می‌باشد برای رمزگشایی مورد استفاده قرار می‌گیرد و امنیت سامانه نیز کاملاً به پنهان ماندن آن وابسته است (کلید خصوصی).

همان‌گونه که گفته شد این شیوه با وجود امنیت بالایی که دارد به دلیل سرعت کمتری که نسبت به الگوریتم کلید متقارن داراست معمولاً برای رمزنگاری داده‌های با حجم بالا مناسب نیست اما داده‌های زیادی نیز همچون گذرواژه‌ها و امضاهای دیجیتالی هستند که دارای حساسیتی زیاد و حجمی کم هستند، از سوی دیگر برای رمزنگاری داده‌های پر حجم توسط رمزنگاری کلید متقارن نیز در نهایت برای انتقال کلید آن به این شیوه نیازمندیم، از همین روی این الگوریتم بسیار پرکاربرد است. کلیدهای عمومی و خصوصی مورد استفاده در رمزنگاری کلید نامتقارن با هم بی ارتباط نبوده و توسط رابطه‌ای ریاضی به هم پیوند خورده‌اند، اما با وجود این به دلیل ساختار پیچیده آن‌ها به دست آوردن کلید خصوصی از راه دانستن کلید عمومی تقریباً غیرممکن است.

1. Public-key cryptography

۳-۲-۳ رمزنگاری کوانتومی

بدون شک آینده دنیای رمزنگاری در دستان رمزنگاری کوانتومی[1] خواهد بود، روشی که بسیار پیشرفته‌تر و پیچیده‌تر از هر روش دیگری است و ضریب اطمینان آن به شکلی باورنکردنی بالا است. در مارس ۲۰۱۴/ ۱۳۹۳ مقاله‌ای توسط آرتور ایکرت[2] و رناتو رنر[3] در نشریه نیچر[4] به چاپ رسید که به خوبی پیشرفت‌های رمزنگاری کوانتومی را توصیف می‌کند، در این مقاله می‌خوانیم که بر خلاف روش‌های دیگر رمزنگاری در شکل کوانتومی آن نیازی نیست که حتماً به دستگاه رمزنگاری و فرد گیرنده پیام اطمینان داشته باشیم، یعنی تأیید هویت گیرنده و سالم بودن الگوریتم که دو فاکتور اصلی در امنیت انتقال پیام هستند در اینجا نمی‌توانند مشکلی ایجاد کنند، حتی اگر دستگاه رمزنگاری را از دشمن‌مان خریده باشیم و تکنولوژی ساختش نیز برای ما کاملاً ناشناخته باشد می‌توانیم همچنان ارتباط‌هایی ایمن داشته و از خود در مقابل نفوذ دشمنان محافظت کنیم!

اکرت و بسیاری دیگر در طول ۲۰ سال گذشته در حال کار بر روی استفاده از ویژگی‌های کوانتومی ذرات نور برای به اشتراک گذاری یک کلید امنیتی هستند، کلیدی که در صورت انتقال ایمن می‌تواند برای ایجاد ارتباط‌های امن و محرمانه مورد استفاده قرار بگیرد. این کلید داده‌ای تصادفی از صفر و یک هاست که تصادفی بودن آن از انتخاب‌های تصادفی ما برای نحوه اندازه‌گیری ذره و دیگر موارد این چنینی ناشی می‌شود. رنر و اکرت رابطه‌ای ریاضی را درباره افزایش تصادفی بودن داده‌ها کشف کرده‌اند که از یک حقه کوانتومی استفاده می‌کند تا داده‌های تقریباً تصادفی را به داده‌های کاملاً تصادفی تبدیل کند، ابزاری

1. Quantum cryptography

2. Artur Konrad Ekert

3. Renato Renner

4. Nature

که اگر در رمزنگاری مورد استفاده قرار بگیرد می‌تواند توانایی ما را برای انتخاب کلیدهای کاملاً تصادفی تثبیت کند و امنیت ارتباط‌های ما را تضمین کند.

۳-۳ روش‌های رمزگشایی

به همان میزانی که رمزنگاران زمان صرف کرده و با کوشش و پشتکار رمزنگاری‌های پیشرفته‌تری را ایجاد می‌کنند که غیر قابل نفوذ باشند، رمزشکنان نیز با همان جدیت و سخت‌کوشی تحلیل‌های پیشرفته‌تری را به دست می‌آورند که می‌توانند به ساختار آن رمزها نفوذ کرده و آن‌ها را بشکنند. می‌بینیم که در طول تاریخ همیشه این دو گروه در حال پیشی گرفتن از یکدیگر بوده‌اند و همان‌گونه که رمزنگاری بسیار کهن‌سال است تحلیل رمز نیز چیز نوینی نیست و از همان روزها به آن می‌اندیشیدند. به دلیل پیچیدگی‌های بیشتر تحلیل رمز اولین نشانه‌های موفقیت در آن را دیرتر مشاهده می‌کنیم، به گونه‌ای که رمز سزار ده‌ها سال پیش از میلاد استفاده میشد اما اولین گواه توانایی تحلیل آن را نزدیک به نهصد سال پس از آن در کارهای کندی می‌بینیم.

توانایی تحلیل رمز به میزان اطلاعاتی بستگی دارد که می‌توانیم از متن رمز شده بدست آوریم، اگر این متن هیچ اطلاعاتی در مورد متن اصلی به ما ندهد یعنی غیرقابل شکستن است. یکی از روش‌هایی که از چنین ویژگی برخوردار است پد یکبار مصرف است که اگر به درستی اجرا شود، بدون داشتن کلید نمی‌توانیم هیچ اطلاعاتی در مورد متن اولیه از آن به دست بیاوریم. حال به شیوه‌های مختلف تحلیل رمز و دامنه کاربرد هرکدام می‌پردازم.

۳-۳-۱ حمله جستجوی فراگیر

اگر شما با یک متن رمز شده برخورد کنید و با روش‌های رمزگشایی آشنایی نداشته باشید چه راهی را پیشنهاد می‌دهید؟ درست است، آزمودن تمام حالت‌های گوناگون!

این راه درواقع بدیهی‌ترین گزینه و در زمانی که بشر هنوز دانشی از راه‌های کشف رمز نداشت تنها گزینه موجود بود، با این حال از این راه‌حل به عنوان اولین شیوه تحلیل رمز نام برده نمی‌شود، زیرا که در عین بدیهی بودن بسیار ناکارآمد است. این روش تنها در حالتی که متن رمز شده بسیار کوتاه باشد و یا در شرایطی خاص قابل استفاده است، برای نمونه اگر بدانیم که متن نامه به شیوه سزار رمزنگاری شده است آن‌گاه می‌توانیم با آزمودن تعداد حالت‌های جابه‌جایی حروف که برابر با تعداد الفبای زبان مورد استفاده می‌باشد نامه را رمزگشایی کنیم. ناکامی این روش ادامه داشت تا اینکه ما توانستیم از قدرت محاسباتی ماشین‌ها بهره بگیریم، هرچند که در این روش حالت‌های مورد آزمون بسیار زیاد هستند اما ماشین‌ها با توان محاسباتی فوق‌العاده خود می‌توانستند در زمان کوتاهی تعداد زیادی از حالت‌ها را بیازمایند. در ابتدا ماشین‌ها نسبت به امروز بسیار ضعیف و کند بودند اما توان محاسباتی آن‌ها بسیار سریع پیشرفت کرد و میشد در حالت‌های بسیار بیشتری از این شیوه برای تحلیل رمز استفاده نمود، البته با وجود اینکه امروزه رایانه‌های بسیار قدرتمندی در اختیار داریم اما شیوه‌های نوین رمزنگاری به اندازه‌ای پیچیده هستند و تعداد حالت‌های مورد آزمون در آن‌ها به حدی زیاد است که نمی‌توان همیشه به عنوان گزینه‌ای مناسب سراغ این روش تحلیل رفت.

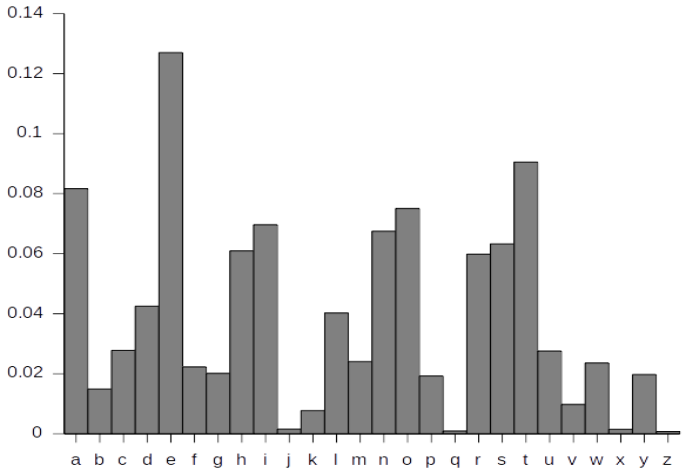
۳-۳-۲ تحلیل فراوانی[1]

اگر شیوه جستجوی فراگیر را در نظر نگیریم می‌توان گفت اولین روش تحلیل رمزی که شناخته شده است روش تحلیل فراوانی است، این روش اولین بار توسط کندی در کتاب درباب رمزگشایی پیام‌های رمزنگاری شده شرح داده شد. این روش تحلیل برپایه این واقعیت ایجاد شده است که در هر زبان میزان پراکندگی حروف مختلف در متن‌های گوناگون تا حدودی یکسان است و با در اختیار داشتن نمودارهای مربوط به این فراوانی‌ها و مقایسه آن با نمودار فراوانی حروف متن رمز شده به راحتی می‌توان متن را رمزگشایی کرد. البته محدودیت‌هایی نیز در این روش وجود دارد و استفاده از آن همیشه راحت نیست، برای نمونه اگر متن‌رمز کوتاه باشد کار بسیارسخت بوده و احتمال اشتباه زیاد است زیرا این درصدهای (تقریبا) ثابت در متن‌های کوتاه خود را نشان نمی‌دهند. ایراد دیگری که ممکن است ما را با مشکل مواجه کند این است که درصد فراوانی حروف در حالت استاندارد بهترین نتیجه را خواهد داد و در صورتی که موضوع پیام تخصصی باشد احتمال تغییر این درصدها بالاست که البته امکان رمزگشایی را از بین نخواهد برد و امکان بررسی فراوانی در متن‌های تخصصی نیز وجود دارد، اما کار را با سختی‌هایی مواجه خواهد کرد. مشکل دیگر این است که زبان مورد استفاده در پیام اصلی را ندانیم که البته این مورد خیلی جدی نیست زیرا در بیشتر موارد این موضوع آشکار است و از سوی دیگر این نمودارها برای بسیاری از زبان‌ها وجود دارند و بررسی چند زبان مختلف چالش بزرگی نخواهد بود. این روش در مورد رمزنگاری‌هایی پاسخ می‌دهد که ساختار زبانی متن تغییر نکرده باشد و تنها حروف دچار دگرگونی شده باشند، برای نمونه جامعه هدف این روش انواع حالت‌های رمزنگاری جانشینی را شامل می‌شود.

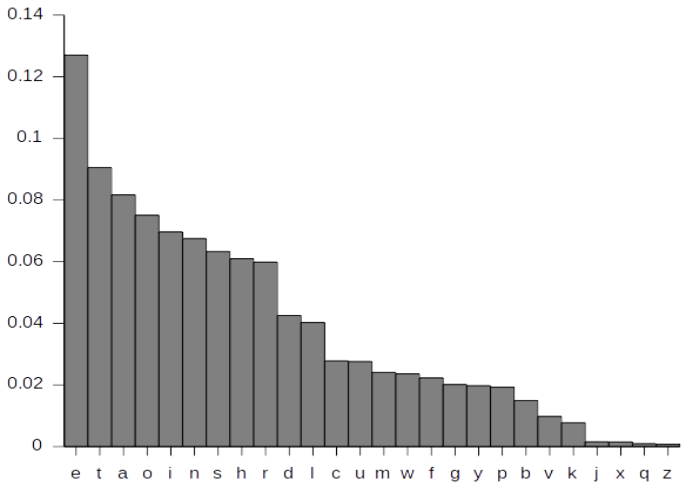
1. Frequency Analysis

در شکل (۷) نمودار درصد فراوانی را برای حروف انگلیسی می‌بینید که برای متن‌های استاندارد به چه شکل است و در نمودار دیگری (شکل۸) این درصدها به ترتیب قرار داده شده‌اند تا دید بهتری داشته باشیم، می‌توان دید که حروف t،e و a پرتکرارترین حروف و q،x و z نیز کم‌کاربردترین آن‌ها هستند. این داده‌ها را برای بسیاری از زبان‌ها می‌توان به راحتی به دست آورد، اما متأسفانه در مورد زبان پارسی هرچه جستجو کردم نتوانستم چنین اطلاعاتی را به دست آورم!

شکل ۷- نمودار درصد فراوانی حروف انگلیسی برای متن‌های استاندارد

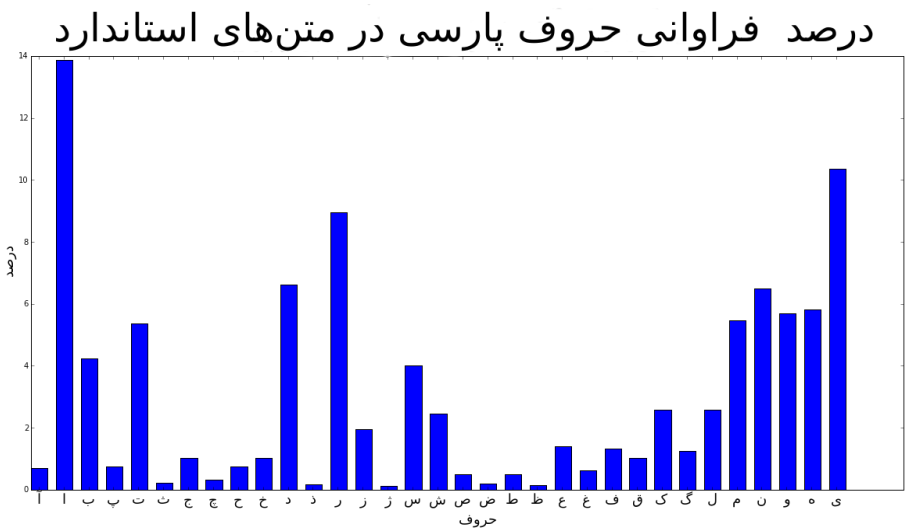


شکل ۸- نمودار درصد فراوانی حروف انگلیسی برای متن‌های استاندارد

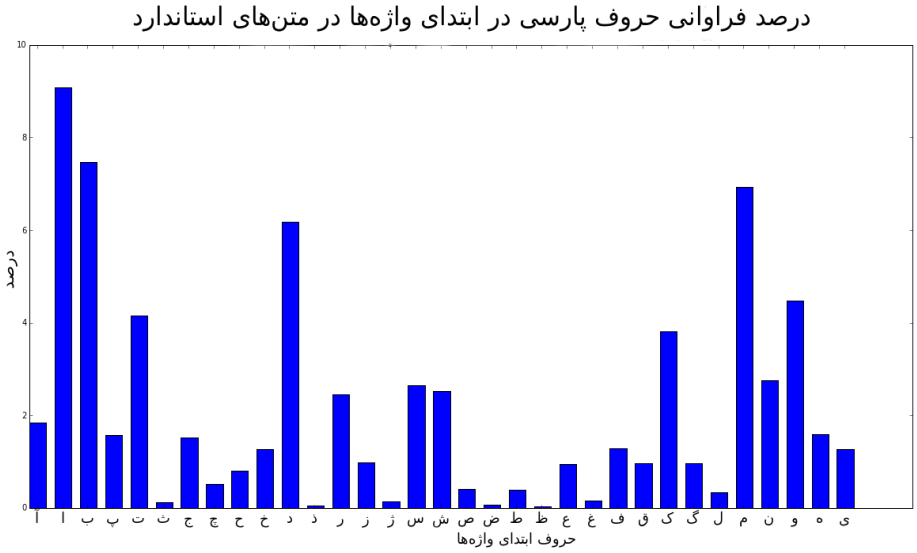


برایم کاملاً شگفت‌آور و البته ناراحت‌کننده بود که می‌دیدم تاکنون کسی این نمودارهای باارزش را برای زبان پارسی تهیه نکرده است و یا اگر هم این کار انجام گرفته چنان پنهان مانده و مورد بازنشر قرار نگرفته است که من در هیچ‌جایی نتوانستم آن را بیابم. از این روی برآن شدم تا با یاری یکی از دوستان گرانقدرم آقای **جادی میرمیرانی** این کار را به انجام برسانم. راه رسیدن به نمودارهای (شکل‌های ۹ و ۱۰ و ۱۱) موردنیاز را در پیوست C به شکل کوتاهی توضیح داده‌ام، این کار تا حد امکان به شکلی دقیق صورت گرفته و می‌تواند برای کاربردهای گوناگون مورد استفاده قرار گیرد. برای علاقمندانی که قصد بررسی تخصصی‌تر موضوع را دارند کد برنامه را که به زبان پایتون و با نسخه سوم پروانه عمومی همگانی گنو منتشر شده است در پیوست D به شکل کامل آورده‌ام.

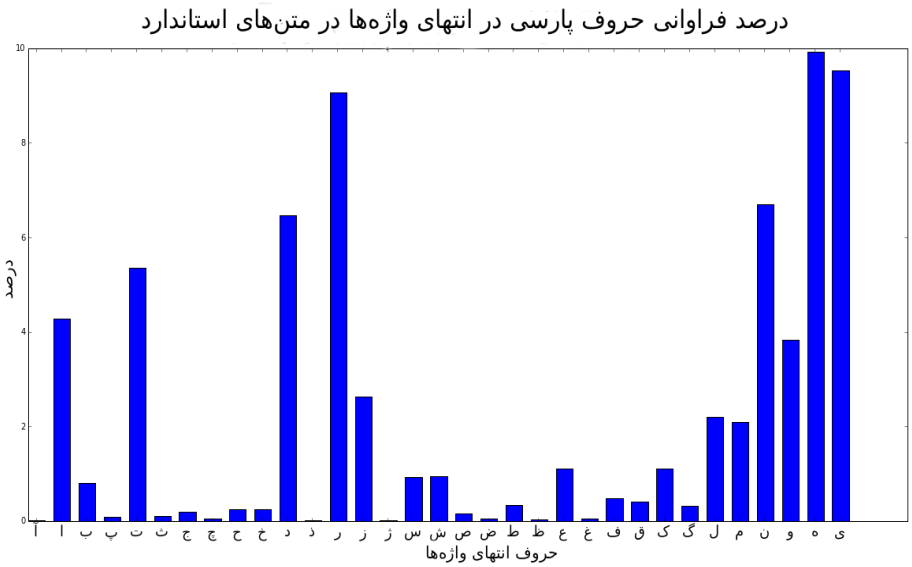
شکل ۹- درصد فراوانی حروف پارسی در متن‌های استاندارد



شکل ۱۰- درصد فراوانی حروف پارسی در ابتدای واژه‌ها در متن‌های استاندارد



شکل ۱۱- درصد فراوانی حروف پارسی در انتهای واژه‌ها در متن‌های استاندارد



نمودار شکل (۹) درصد فراوانی حروف را در حالت کلی نشان می‌دهد و می‌توان انتظار داشت که در یک متن عمومی چنین نسبت‌هایی از حروف را شاهد باشیم. حالت‌های خاصی را نیز می‌توان بدست آورد که برخی بسیار کاربردی هستند، ما در این‌جا دو مورد را بررسی کرده‌ایم که یکی (شکل ۱۰) مربوط به درصد فراوانی حروف در ابتدای واژه‌ها است، به این معنی که واژه‌ها بیشتر با چه حروفی آغاز می‌شوند و دیگری نیز (شکل ۱۱) مربوط به این است که واژه‌ها معمولاً چه حرفی را در پایان خود دارند.

در برخی زبان‌ها مانند انگلیسی که کارهای بیشتری صورت گرفته است بجز نمودارهای ارایه شده (شکل‌های ۷ و ۸) موارد دیگری نیز بررسی شده‌اند که در استفاده دقیق‌تر از تحلیل فراوانی بسیار راهگشا خواهند بود، برای نمونه می‌توان مواردی را در وب‌گاه شخصی

نویسنده بریتانیایی سایمون سینگ مشاهده کرد، در این برگه موارد مختلفی که در تحلیل رمز مفید هستند بررسی شده که گزیده‌ای از آن‌ها را در شکل (۱۲) آورده‌ام.

شکل ۱۲- مواردی که براساس تحلیل فراوانی متن‌های استاندارد برای زبان انگلیسی بدست آمده‌اند

- **پرتکرارترین حروف**

E, T, A, O, I, N, S, H, R, D, L, U

- **دو حرفی‌های رایج**

ss, ee, tt, ff, ll, mm, oo

- **کوچک‌ترین واژه‌ها**

یک حرفی: a, I

دو حرفی: of, to, in, it, is, be, as, at, so, we, he, by, or, on, do

سه حرفی: the, and, for, are, but, not, you, all, any, can

- **حروفی که در ابتدای واژه‌ها رایج‌ترند**

T, O, A, W, B, C, D, S, F, M, R, H, I, Y, E, G, L, N, P, U

- **حروفی که در انتهای واژه‌ها رایج‌ترند**

E, S, T, D, N, R, Y, F, L, O, G, H, A, K, M, P, U, W

* راهنما: تعداد واژه‌های دوحرفی و سه‌حرفی بیشتر بودند که گزیده‌ای از آن‌ها آورده شده است.

۳-۳-۳ سخن پایانی

روش‌های نوین رمزگشایی متناسب با افزایش پیچیدگی روش‌های رمزنگاری پیشرفت کرده و به شکل چشم گیری تخصصی شده‌اند، همیشه از ابتدای تاریخ شاهد این مبارزه بوده‌ایم و احتمالاً پایانی برای آن نخواهد بود. در دنیای امروز سرعت افزایش این پیچیدگی‌ها بسیار بیشتر شده و پول‌های هنگفتی برای ارتقای امنیت از سوی شرکت‌ها و کشورها هزینه می‌شود، از سوی دیگر همین مؤسسات و سازمان‌های خصوصی و دولتی سرمایه‌های کلانی را در جهت فروریختن این لایه‌های امنیتی هزینه می‌کنند. در نگاه اول ممکن است به نظر هدر دادن انرژی و سرمایه بیاید اما با توجه به پیچیدگی‌های روزافزون روش‌ها و نیز دسترسی بیشتر به اطلاعات و نیاز بیشتر به ارتباطات امن نیاز است که چنین روندی پیاده‌سازی شود، زیرا که جهان به سویی رفته که تمام نقاط آن با هم در ارتباط هستند و تقریباً همه از روش‌های مشخص و یکسانی برای ایجاد امنیت استفاده می‌کنند، از این روی اگر توانایی نفوذ در لایه‌های امنیتی روش‌های رمزنگاری استاندارد را کسب کنیم هم توانایی دسترسی به اطلاعات دیگران را خواهیم داشت و هم اینکه می‌توانیم با ارتقای سطح امنیتی آن‌ها روش‌های نوین و ایمن‌تری را برای ارسال پیام‌های خود ایجاد کنیم.

A

پیوست

نمونه‌های پنهان‌نگاری

یکی از نمونه‌های مشهور پنهان‌نگاری در متن مربوط به یکی از نامه‌های ارسالی توسط یک جاسوس آلمانی در زمان جنگ جهانی اول است. متن ارسالی توسط او به شکل زیر بود:

Apparently neutral's protest is thoroughly discounted and ignored. Isman hard hit. Blockade issue affects pretext for embargo on byproducts, ejecting suets and vegetable oils

که پس از کنار هم قرار دادن دومین حرف از هر واژه به متن اصلی می‌رسیم:

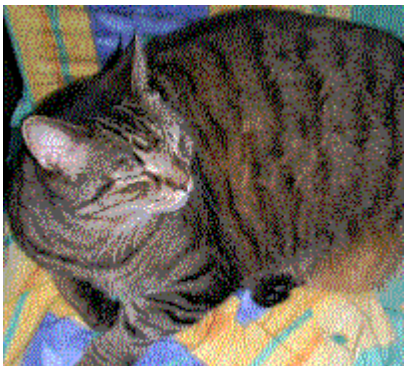
Pershing sails from NY June I

در مورد پنهان‌نگاری در تصویر نیز نمونه‌های بسیاری می‌توان ارائه کرد.

در تصویر(۱۳) چیزی غیرعادی توجه شما را به خود جلب نمی‌کند؟ به نظر تصویری عادی می‌آید. در این تصویر پنهان‌نگاری انجام گرفته است و برای به دست آوردن تصویر اصلی باید دو بیت آخر هر فضای رنگ را جدا کنیم. پس از بهنجارش و نرمال‌سازی داده‌ها به وسیله پردازش تصویر به شکل (۱۴) می‌رسیم. فکرش را هم نمی‌کردید که یک گربه روی درخت لانه کرده باشد، درست است؟!

شکل ۱۳- تصویری که در آن پنهان‌نگاری انجام گرفته شده است

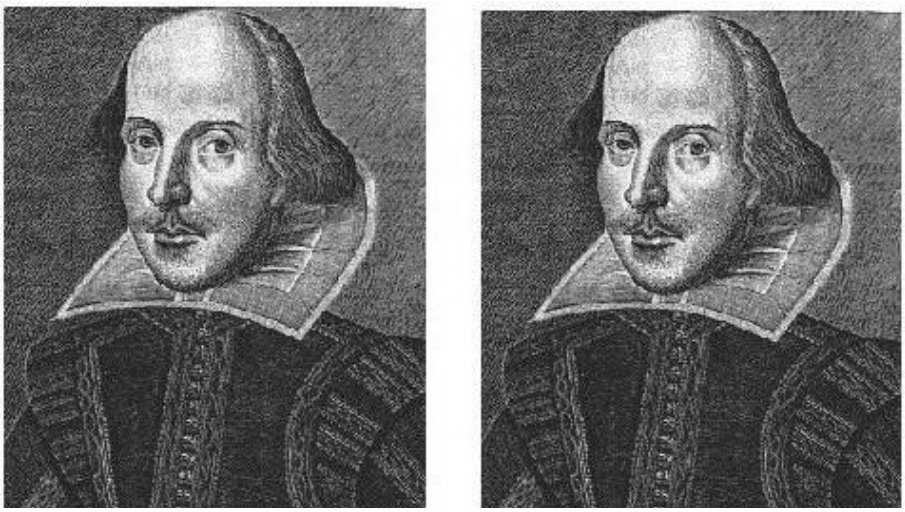
شکل ۱۴- تصویر پنهان‌نگاری شده در شکل ۱۳



ممکن است گاهی متن مورد نظرمان را در یک تصویر جاسازی کنیم، در اینکه انجام این کار در تصاویر پیچیده‌تر از پنهان‌نگاری در متن است شکی نیست، اما روش کار هنوز هم

به همان شکل است. دو تصویر زیر را در نظر بگیرید: کاملاً همانند یکدیگر به نظر می‌رسند، اما در تصویر سمت راست پنهان‌نگاری انجام گرفته است.

شکل ۱۵- دو تصویری که کاملاً همانند یکدیگر به نظر می‌رسند، اما در تصویر سمت راست پنهان‌نگاری انجام گرفته است



برای رسیدن به متن اصلی بایستی در جاهای خاصی از تصویر که موردنظر است کدهای باینری را با تصویر اصلی مقایسه کنیم.

روش کار به این شکل است که نقاطی از تصویر انتخاب می‌شوند و سپس کدهای باینری آن‌ها را به شکلی تغییر می‌دهند که قابل توجه نباشد، برای نمونه اگر واژه HELLO در تصویر جاسازی شده باشد برای انجام آن در پنج نقطه تصویر تغییراتی ایجاد شده است. روش کار را برای حرف اول توضیح می‌دهم:

برای نمونه کد باینری مربوط به نقطه‌ای خاص در تصویر به شکل زیر است:

۰۱۰۰۰۰۰۰ ۱۱۰۱۰۱۰۱ ۱۱۰۰۱۰۰۱ ۱۰۰۰۰۰۰۰ ۱۱۰۰۱۰۱۰
۰۱۰۰۰۰۰۱ ۱۰۰۱۰۰۱۰ ۱۱۰۰۰۰۱۱ ۱۱۱۰۰۱۱۱

حال این کد را در تصویر به شکل زیر تغییر می‌دهیم:

۱۰۰۰۰۰۰۱ ۱۰۱۰۱۰۱۰ ۱۰۰۱۰۰۱۰ ۱۰۰۰۰۰۰۰ ۱۰۰۱۰۱۰۰
۱۰۰۰۰۰۱۱ ۱۰۰۱۰۰۱۰ ۰۱۱۰۰۰۰۱ ۰۱۱۱۰۰۱۱

با قراردادن بیت‌های تغییر داده شده در کنار هم به مقدار ۱۰۰۱۰۰۰ می‌رسیم که در جدول اسکی* برابر با حرف H انگلیسی است. اسکی یک روش کدبندی بر اساس الفبای انگلیسی است که در رایانه‌ها و وسایل الکترونیکی دیگر از آن استفاده می‌شود.

* ASCII

B

پیوست

اصول کرکهف

در شماره‌های ژانویه و فوریه سال ۱۸۸۳/ ۱۲۶۱ ژورنال علوم نظامی* که به زبان فرانسه منتشر می‌شود، دو مقاله از آگوست کرکهف** منتشر شد که حاوی شش اصل اساسی برای رمزنگاری بودند:

۱- سامانه باید نه از لحاظ ریاضیاتی که در عمل غیرقابل نفوذ باشد.

۲- نبایستی سامانه نیاز به پنهان‌سازی داشته باشد و باید به گونه‌ای طراحی شود که اگر توسط دشمن دزدیده شد، دردسری ایجاد نشود.

- این اصل می‌گوید، بنا را بر این بگذارید که الگوریتم رمزنگاری شما برای دشمن شناخته شده است و امنیت سامانه در پنهان ماندن کلید رمزنگاری نهفته است. البته این الزاماً به معنای آن نیست که باید روش و الگوریتم رمزنگاری در اختیار همگان باشد، اما بایستی در زمان ساخت هر الگوریتم چنین پنداشته شود که همه به آن دسترسی دارند و به عبارت دیگر امنیت سامانه به پنهان

ماندن الگوریتم وابسته نباشد. یکی دیگر از ویژگی‌های این اصل در این است که اگر الگوریتم را در دسترس همه قرار دهیم درواقع باعث افزایش امنیت آن می‌شویم زیرا پژوهشگران امنیتی بسیار بیشتری به بررسی آن خواهند پرداخت و با کشف حفره‌های امنیتی آن از سوءاستفاده از آن‌ها توسط دشمنان و جاسوس‌ها جلوگیری می‌کنند.

۳- برقراری ارتباط راحت بوده و بتوان بدون نیاز به نوشتن کلید به راحتی آن را به خاطر سپرد، همچنین کلید به شکلی باشد که در صورت خواست طرفین به راحتی قابل تغییر و اصلاح باشد.

- یکی از ویژگی‌های مثبت استفاده از کلید در این است که اگر به دلیل لو رفتن آن و یا به خطر افتادن امنیت سامانه (به هر شکلی) نیاز به تغییر روش رمزنگاری داشته باشیم، بدون دستکاری الگوریتم که کاری دشوار و زمان‌بر است با انتخاب کلیدی نوین که به مراتب راحت‌تر و سریع‌تر است می‌توانیم دوباره سیستمی کاملا ایمن داشته باشیم و بدون هیچ نگرانی از کشف کلید قبلی توسط دشمن از آن استفاده کنیم.

۴- سامانه با ارتباطات تلگراف سازگاری داشته باشد.

۵- سامانه قابل حمل بوده و استفاده از آن به بیش از یک نفر نیاز نداشته باشد.

۶- سامانه به راحتی قابل راه اندازی و استفاده باشد، باعث فشار فکری نشده و نیازمند دانستن مجموعه زیادی از قوانین نباشد.

❄ La cryptographie militaire

❄❄ Auguste Kerckhoffs

C

پیوست

فراوانی حروف پارسی در متن‌های استاندارد

برای به دست آوردن درصد فراوانی حروف پارسی در متن‌های گوناگون بایستی نکته‌های مختلفی را در نظر می‌گرفتیم که تا حد امکان نتیجه نهایی دقیق و قابل استناد باشد. اولین چیزی که نیاز داشتیم نرم‌افزاری بود که بتواند این کار را به انجام برساند، تا حد زیادی کارهای تکنیکی را جادی* برعهده داشت و برنامه‌ای را نوشت که بتواند این پردازش را روی بانک داده مورد نظر انجام دهد.

این نرم‌افزار با نسخه سوم پروانه عمومی همگانی گنو جی پی ال** منتشر شده است و افرادی که قصد بررسی کد منبع آن را دارند می‌توانند در پی‌نوشت مقاله پیوند مربوط به آن*** را مشاهده کنند.

یکی دیگر از موارد مهمی که باید به دقت برگزیده میشد منبع مورد استفاده برای بررسی و پردازش بود، بهترین گزینه ممکن وب‌گاه ویکی‌پدیای پارسی است که (در زمان نوشتن این کتاب)

با دارا بودن بیش از ۴۶۰ هزار مقاله در موضوعات گوناگون منبعی سرشار از واژه‌های مختلف در بسیاری از زمینه‌ها است. با توجه به گستردگی بسیار زیاد ویکی پدیا ۱۰ درصد مقاله‌های آن را به شکلی تصادفی مورد پردازش قرار دادیم که نمودارهای حاصل از آن را در شکل‌های ۹، ۱۰ و ۱۱ مشاهده می‌کنید.

در این نرم‌افزار نوشته‌های حاشیه وب‌گاه که در تمامی برگه‌ها تکرار می‌شدند در نظر گرفته نشده است و به جای حروفی که ممکن بود به شکل عربی نگاشته شوند از برابر پارسی آن‌ها در درصدگیری‌ها استفاده شده است، برای نمونه ممکن بود که گاهی به جای حرف «ی» از شکل عربی آن «ي» استفاده شود.

❄ Https://Jadi.net

❄❄ GPLv3

❄❄❄ http://j.mp/persianletters

D

پیوست

فراوانی حروف پارسی در متن‌های استاندارد

(کد منبع برنامه)

همان‌گونه که پیش‌تر توضیح دادم، این برنامه توسط دوست گرانقدرم جادی و به زبان پایتون نوشته شده است. دوستانی که علاقمند و آشنا به برنامه‌نویسی هستند می‌توانند از روش کار آگاه شده و حتی آن را توسعه دهند.

بهترین منبعی که برای این کار پیدا کردیم ویکی‌پدیای پارسی بود که به دلیل حجم زیاد و گستردگی موضوعات نتیجه‌ای تا حد امکان دقیق برای متن‌های استاندارد به دست می‌دهد، اما می‌توان با انجام این تحلیل روی متن‌های تخصصی نمودارهای سفارشی برای موضوعات مختلف را نیز به سادگی به دست آورد.

به دلیل حجم بسیار زیاد نوشته‌ها ما این پردازش را روی ۱۰ درصد مقاله‌های ویکی‌پدیای پارسی انجام دادیم که نزدیک به ۵۰ هزار مقاله را شامل می‌شود، برای دریافت نتایج دقیق این انتخاب به صورت کاملاً تصادفی صورت می‌گیرد.

کد زیر تنها متن مقاله‌ها را از ویکی‌پدیا با شرایط گفته شده بیرون کشیده و مطالب حاشیه‌ای را وارد پردازش نمی‌کند:

```
In[5]: %matplotlib inline

In[1]: import xml.etree.ElementTree as etree

       import re
       import random

       inFile = '/home/jadi/w/wikipedia/fawiki-20150807-pages-articles.xml'

       random.seed()
       counter = 0

       for event, elem in etree.iterparse(inFile, events=('start', 'end', 'start-ns', 'end-ns')):
          if random.random() < 0.9: #only work on 10% of articles
             try:
                elem.clear()
             except:
                pass
             continue

          thisTxt = None
          try:
             if elem.tag.endswith('/}text'):
                thisTxt = elem.text
                elem.clear()
          except:
             continue

          if not thisTxt:
             elem.clear()
             continue

       print thisTxt[1:10000]
```

متنی که به‌دست می‌آید هنوز آماده پردازش نیست و در نتیجه با استفاده از کد زیر هم حروف عربی را با هم‌ارز پارسی‌شان جایگزین می‌کنیم و هم کاراکترهایی را که جزو زیر-ساخت دستوری ویکی‌پدیا هستند و در متن بارها تکرار شده‌اند را پاک می‌کنیم:

```
In[2]: import re

    inputText = '/home/jadi/w/wikipedia/wiki_fa.txt'
    f = open(inputText, 'r')
    alltext = f.read()
    text = alltext
    text=re.sub("\n", " ", text)
    text=re.sub("\[+", "[", text)
    text=re.sub("\]+", "]", text)
    text=re.sub("\{+", "{", text)
    text=re.sub("\}+", "}", text)
    text=re.sub("{.*?}", " ", text)
    text=re.sub("\<.*?\>", " ", text)
    text=re.sub("\[.*?\]", " ", text)
    text=re.sub("\s+", " ", text)

    # changing some arabic chars to correct persian ones
    text=re.sub(u"ي", u"ی", text)
    text=re.sub(u"ك", u"ک", text)

    print text[1:10000]
```

حال برای اطمینان الفبای مورد نظرمان را با استفاده از کد زیر از متن بدست آمده بیرون می‌کشیم و برای پردازش آماده می‌کنیم:

```
g' wiki_fa_only_text.txt > wiki_only_farsi_chars.txt//*[ ]/sed 's $[^آئابپتثجچحخدذرزژسشصضطظعغفقکگلمنوهی
```

متن نهایی آماده است و پردازش را با کد زیر شروع می‌کنیم:

```
In[3]: f = open('/home/jadi/w/wikipedia/wiki_only_farsi_chars.txt', 'r')

    alltext = f.read()
    alltext = alltext.decode("utf-8")
```

ابتدا هرکدام از حروف را در متن نهایی شمرده و تعداد آن‌ها را بدست می‌آوریم و سپس برای هر کدام درصدگیری می‌کنیم و با توجه به آن‌ها نمودار را رسم می‌کنیم:

```
In[14]: allchars = {}

        for i in range(0, len(alltext)):
           allchars[alltext[i]] = allchars.get(alltext[i], 0) + 1

        allwordsnum = allchars[' ']
        totalcharsnum = len(alltext) - allchars[' ']
        del allchars[' ']
        del allchars['\n']

        import numpy as np
        from matplotlib import pyplot as plt

        letters = u'آابپتثجچحخددذرزژسشصضطظعغفقکگلمنوهی'
        lettervals =  []
        letterlist = []
        for letter in list(letters):
           print letter, allchars[letter]
           lettervals.append(allchars[letter]*1.0/totalcharsnum*100)

        width = 1/1.5
        plt.figure(figsize=(20,10))
        plt.bar( range(len(lettervals)), lettervals, width)
        plt.xticks([x+0.3 for x in range(len(lettervals))], list(letters), fontsize=18 )
        plt.title(u'Percentage of persian letters\n10% of wikipedia articles are tested', fontsize=34)
        plt.ylabel('Percent', fontsize=20)
        plt.xlabel('Letter', fontsize=20)
        plt.show()
```

نمودار بدست آمده را در شکل (۹) می‌بینید. برای بدست آوردن نمودار شکل‌های (۱۰) و (۱۱) نیز به همین ترتیب عمل می‌کنیم، با این تفاوت که تمرکز خود را به ترتیب روی ابتدا و انتهای واژه‌ها می‌گذاریم.

درصد فروانی حروف در ابتدای واژه‌ها:

```
In[15]: allwords = alltext.split()

        initialchars = {}
        for word in allwords:
            initialchars[word[0]] = initialchars.get(word[0], 0) + 1

        import numpy as np
        from matplotlib import pyplot as plt

        letters = u'آابپتثجچحخددذرزژسشصضططظعغفقککگلمنوهی'
        lettervals =  []
        letterlist = []
        print 'Initial letters'
        for letter in list(letters):
            print letter, allchars[letter]
            lettervals.append(initialchars[letter]*1.0/allwordsnum*100)

        width = 1/1.5
        plt.figure(figsize=(20,10))
        plt.bar( range(len(lettervals)), lettervals, width)
        plt.xticks([x+0.3 for x in range(len(lettervals))], list(letters) , fontsize=18)
        plt.title(u'Percentage of persian initial letters\n10% of wikipedia articles are
tested', fontsize=34)
        plt.ylabel('Percent', fontsize=20)
        plt.xlabel('Initial Letter', fontsize=20)
        plt.show()
```

درصد فروانی حروف در انتهای واژه‌ها:

```
In[16]: allwords = alltext.split()


        finalchars = {}
        for word in allwords:
            finalchars[word[-1]] = finalchars.get(word[-1], 0) + 1

        # lets repeat for final letters
        import numpy as np
        from matplotlib import pyplot as plt

        letters = u'آابپتثجچحخدذرزژسشصضطظعغفقکگلمنوهی'
        lettervals =  []
        letterlist = []
        print 'Final letters'
        for letter in list(letters):
            print letter, allchars[letter]
            lettervals.append(finalchars[letter]*1.0/allwordsnum*100)

        width = 1/1.5
        plt.figure(figsize=(20,10))
        plt.bar( range(len(lettervals)), lettervals, width)
        plt.xticks([x+0.3 for x in range(len(lettervals))], list(letters), fontsize=18 )
        plt.title(u'Percentage of persian final letters\n10% of wikipedia articles are
tested', fontsize=34)
        plt.ylabel('Percent', fontsize=20)
        plt.xlabel('Final Letter', fontsize=20)
()plt.show
```

کتاب‌نامه

Article (ISI)

Bellovin S.M. Frank Miller: Inventor of the One-Time Pad. Cryptologia 35(3): 203-222

Ekert A., Renner R. The ultimate physical limits of privacy. Nature 507: 443–447

Book

Benson R.L. The Venona Story. Maryland: National Security Agency (NSA). 63 Pages. 2001

Delfs H., Knebl H. Introduction to Cryptography: Principles and Applications. New York: Springer Publishing; second edition. 367 pages. 2007

Kahn D. The Codebreakers – The Story of Secret Writing. New York: Charles Scribner's Sons. 1200 pages. 1996

Schneier B. Secrets & Lies: Digital Security in a Networked World. New Jersey: John Wiley & Sons. 432 pages. 2000

Singh S. The Code Book: The Science of Secrecy from Ancient Egypt to Quantum Cryptography. New York: Anchor; Reprint edition. 432 pages. 2000

Trithemius J. Polygraphiae Libri Sex. 550 Pages. 1508

Wright P. Spycatcher: The Candid Autobiography of a Senior Intelligence Officer. New York: Viking Press. 392 Pages. 1987

Patent

Vernam G.S. Secret signaling system. US1310719 A. 1919

Fridrich J., Goljan M. Reliable detection of LSB steganography in color and grayscale images. US6831991 B2. 2004

Article

ELIAN C.C. EVOLUTION OF THE ELIAN SCRIPT: From Code to Calligraphic Writing System. 29 Pages. 2006

The ACA and You: A Handbook For The Members Of The American Cryptogram Association. American Cryptogram Association. 105 Pages. 2016

Website

http://www.cryptogram.org/resources

http://www.simonsingh.net/The_Black_Chamber

http://csrc.nist.gov/archive/aes

نمایه

ابویوسف کندی ۲۹، ۳۰، ۴۶، ۴۸
اسکتلی ۳۴
اصل دوم کرکهوف ۳۱، ۴۰
ریندال ۴۳
MD5 ۳۹
انجمن متون رمزی آمریکایی ۳۵
AES ۴۳
آژانس امنیت ملی آمریکا ۳۸
آژانس امنیتی انگلستان ۳۸، ۳۹
پدر رمزنگاری غربی ۲۹
تابش پس زمینه کیهانی ۳۹
جدول اسکی ۵۸
جوهر نامرئی ۱۲
حمله ۱۱ سپتامبر ۲۱
در سراسر اسپانیا، آسمان صاف است ۲۰
DES ۴۲
دیوید کان ۲۹
سازمان اطلاعاتی شوروی ۳۸
سازمان ثبت اختراع آمریکا ۱۶
سایمون سینگ ۳، ۲۹، ۵۳
سند ثبت اختراع ۳۷
نشریه نیچر ۴۵
نویز سفید ۱۶
واپاشی هسته‌ای ۳۹

از دیرباز انسان‌ها در پی یافتن راه‌هایی برای حفظ امنیت اطلاعات مهم بوده‌اند. پادشاهانی که پیام‌هایی برای کشورهای متحد ارسال می‌کردند و فرماندهانی که دستوراتشان را برای سپاهیان خود در نقاط مختلف می‌فرستادند. گاهی ارسال درست و پنهانی یک پیام کلیدی می‌توانست نتیجه یک نبرد تاریخی را مشخص کند و می‌توان انسان‌هایی را دید که برای این هدف جان خود را فدا می‌کردند. درواقع هر چه زمان بیشتری می‌گذرد لزوم اهمیت به امنیت اطلاعات نیز بیشتر احساس می‌شود، چه بسا اگر متفقین نمی‌توانستند رمزهای نازی‌ها را بشکنند امروز دنیای دیگری را شاهد می‌بودیم!

(CC BY-SA 4.0)