بسم الله الرحمن الرحیم

# رمزنگاری هیل

# Hill Cipher

روش رمزنگاری هیل که در سال ۱۹۲۹ توسط لستر هیل بوجود آمد. این رمزنگاری جایگزینی چاپی، برپایه‌ی عملیات‌های جبر خطی بر روی ماتریس تولید شده از پیام کار می‌کند.

- مراحل و عملیات‌های مورد نیاز برای رمزنگاری
- مراحل و عملیات‌های مورد نیاز برای رمزگشایی
- علت کارا بودن رمزنگاری
- امنیت رمزنگاری و توضیحاتی پیرامون کلید
- پیاده‌سازی توسط زبان برنامه‌نویسی Python

**عرفان مهربان ۹۵۱۳۰۳۲**

# مراحل و عملیات‌های مورد نیاز برای رمزنگاری

در اولین قدم ما هر کارکتر در الفبای زبان در حال رمز نگاری را به یک عدد نظیر می‌کنیم.

برای رمزگشایی غیرمبهم برای باید این اعداد به پیمانه‌ی تعداد تمامی حروف الفبا باشد.

همچنین تعداد حروف الفبا می‌بایست اول باشد.

برای مثال حروف الفبای زبان زیر دارای ۷ کارکتر است.

| | | | | | | |
|---|---|---|---|---|---|---|
| a | b | c | d | e | f | g |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 |

پس کد متناظر به fgdbacce می‌شود : (5,6,3,1,0,2,2,4)

حال نیاز به یک **ماتریس کدگذاری (کلید)** داریم که بین دریافت‌کننده و فرستنده مشترک باشد. این ماتریس می‌بایست مربع باشد. همچنین دقت شود که این ماتریس معکوس‌پذیر باشد. یعنی دترمینان آن مخالف ۰ باشد.

برای مثال ماتریس کدگذاری زیر را در نظر بگیرید:

$$K = \begin{bmatrix} 5 & 0 \\ 1 & 4 \end{bmatrix}$$

در مرحله‌ی بعدی رمزنگاری پیام باید پیام کد شده توسط جدول بالا را به یک **ماتریس پیام** تبدیل کنیم. برای اینکار ابتدا سطر اول را از چپ به راست پر کرده و سپس به سطر بعدی می‌رویم.

این ماتریس باید تعداد سطرهای مساوی با ماتریس کدگذاری شده‌ی مشترک را داشته باشد.

در صورتی که درایه‌ای از این ماتریس پر نشد می‌توان آن را به صورت قراردادی بین فرستنده و گیرنده با 0 پر کرد.

بنابراین ماتریس پیام مثال گفته شده این‌گونه خواهد بود:

$$M = \begin{bmatrix} 5 & 6 & 3 & 1 \\ 0 & 2 & 2 & 4 \end{bmatrix}$$

برای رمزنگاری پیام کافی است این ماتریس پیام از سمت راست در ماتریس کلید ضرب شود. سپس از درایه‌های ماتریس حاصل پیمانه‌ی تعداد حروف الفبا گرفته شود.

$$KM \bmod 7 = \begin{bmatrix} 5 & 0 \\ 1 & 4 \end{bmatrix} \begin{bmatrix} 5 & 6 & 3 & 1 \\ 0 & 2 & 2 & 4 \end{bmatrix} \bmod 7 = \begin{bmatrix} 25 & 44 & 29 & 33 \\ 5 & 14 & 11 & 17 \end{bmatrix} \bmod 7 = \begin{bmatrix} 4 & 2 & 1 & 5 \\ 5 & 0 & 4 & 3 \end{bmatrix}$$

در مرحله‌ی آخر ماتریس حاصل را می‌توان با همان جدول کدگذاری به پیام رمزنگاری شده تبدیل کرد. برای انجام این کار می‌بایست اعداد را به ترتیب از بالا و سمت راست شروع به تبدیل کرد و با حرکت به سمت چپ و سطرهای پایین کدگذاری را ادامه داد.

برای مثال ماتریس حاصل از فرآیند مرحله‌ی قبل اینگونه تبدیل به پیام رمزنگاری شده می‌شود:

$$\begin{bmatrix} 4 & 2 & 1 & 5 \\ 5 & 0 & 4 & 3 \end{bmatrix} \rightarrow (4,2,1,5,5,0,4,3) \rightarrow \text{ebaffaedd}$$

# مراحل و عملیات‌های مورد نیاز برای رمزگشایی

برای رمزگشایی ما تنها به **معکوس ماتریس کدگذاری** نیاز داریم. این معکوس نیاز است که در پیمانه‌ی تعداد حروف الفبا معکوس ماتریس کلید باشد.

$$K^{-1} = \begin{bmatrix} 5 & 0 \\ 1 & 4 \end{bmatrix}^{-1} \equiv 6\begin{bmatrix} 4 & 0 \\ -1 & 5 \end{bmatrix} \equiv \begin{bmatrix} 3 & 0 \\ 1 & 2 \end{bmatrix}$$

حال برای رمزگشایی پیام رمزنگاری شده می‌بایست آن را با جدول کدگذاری تبدیل به یک رشته از اعداد کرده و در یک ماتریس با تعداد سطرهای مساوی ماتریس کلید قرار داده و معکوس ماتریس کدگذاری را در آن ضرب کنیم. (یعنی دقیقا همان مراحل قبل را به صورت معکوس انجام دهیم.)

$$\text{ebaffaedd} \rightarrow (4,2,1,5,5,0,4,3) \rightarrow \begin{bmatrix} 4 & 2 & 1 & 5 \\ 5 & 0 & 4 & 3 \end{bmatrix} = S$$

$$K^{-1}S = M = \begin{bmatrix} 3 & 0 \\ 1 & 2 \end{bmatrix}\begin{bmatrix} 4 & 2 & 1 & 5 \\ 5 & 0 & 4 & 3 \end{bmatrix} \equiv \begin{bmatrix} 5 & 6 & 3 & 1 \\ 7 & 2 & 2 & 4 \end{bmatrix}$$

در این مرحله ماتریس را به صورت سطری به یک رشته از اعداد تبدیل کرده و با جدول کدگذاری می‌توان اعداد را به حروف تبدیل کرد و بدین صورت پیام رمز شده به دست می‌آید.

$$\begin{bmatrix} 5 & 6 & 3 & 1 \\ 7 & 2 & 2 & 4 \end{bmatrix} \rightarrow (5,6,3,1,0,2,2,4) \rightarrow \text{fgdbacce}$$

# علت کارا بودن رمزنگاری

رمزنگاری هیل از قاعده‌ی ساده‌ی ضرب ماتریس‌های معکوس استفاده می‌کند. فرض کنید $K$ کلید رمزنگاری و $M$ پیام مورد نظر باشد. همچنین تعداد کارکترهای الفبا را $n$ در نظر بگیرید. با توجه به این‌که ضرب دو ماتریس معکوس برابر ماتریس همانی می‌شود و ماتریس پیام همواره به پیمانه‌ی تعداد کارکترهای الفبا می‌باشد داریم:

$$encryption: S := KM \ mode \ n$$

$$decryption: K^{-1}S = K^{-1}(KM \ mode \ n) \equiv (K^{-1}K)M \ mode \ n \equiv IM \ mode \ n = M$$

# امنیت رمزنگاری و توضیحاتی پیرامون کلید

متاسفانه رمزنگاری هیل به علت ساختار کاملا خطی خود نسبت به حمله‌ی متن آشکار[1] آسیب پذیر است. یعنی حمله‌کننده تنها با داشتن یک پیام و متن رمزشده‌ی آن می‌تواند ماتریس کلید را به دست آورد. حتی با دانستن بخشی از پیام و متن رمز شده می‌توان کلید را پیدا کرد. اگر طریقه‌ی پر کردن ماتریس پیام سطری باشد حمله‌کننده تنها $n$کارکتر از نیمه‌ی اول متن و $n$ کارکتر از نیمه‌ی دوم را بداند و در صورتی که ماتریس پیام به صورت ستونی پر شده باشد کافی است دو $n$ تایی کارکتر پشت‌سر هم را بداند تا بتواند ماتریس کلید را به دست آورد. البته دانستن مکان دقیق کارکترهای مشخص شده برای حمله‌کننده در متن اصلی حائز اهمیت است. درصورتی که این اطلاعات در دسترس حمله‌کننده نباشد می‌تواند با بررسی تمام حالت‌های موجود و بررسی نتیجه‌ی حدس، به کلید مورد نظر دست پیدا کند. همچنین می‌توان با دانستن چندتایی‌های پرتکرار در الفبای مورد نظر حدس‌های بهتری زد و سریع‌تر به کلید دسترسی پیدا کرد.

انتخاب یک ماتریس کلید مناسب باعث می‌شود که تغییر کوچکی در ماتریس متن باعث تغییر بزرگی در رمز شده بشود. درحالی که ضرب ماتریسی به تنهایی یک رمزنگاری امن را توجیه نمی‌کند ولی به علت همین ویژگی وقتی با عملیات‌های غیرخطی ترکیب می‌شود می‌تواند امنیت بهتری را تضمین کند. برای مثال رمزنگاری‌های دارای استاندارد AES ویا Twofish. با توجه به روش تولید رمز کمتر بودن مقادیر 0 در ماتریس کلید باعث می‌شود که خاصیت ذکر شده افزایش پیدا کند.

برای مثال در مثال زیر می‌بینید که ۲ کارکتر یکسان در کنار هم در پایان به ۲ کارکتر ناهمسان می‌رسد:

$$baae \rightarrow 1004 \rightarrow \begin{bmatrix} 1 & 0 \\ 0 & 4 \end{bmatrix}$$

$$S = \begin{bmatrix} 5 & 7 \\ 1 & 4 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 4 \end{bmatrix} = \begin{bmatrix} 5 & 14 \\ 1 & 16 \end{bmatrix} \equiv \begin{bmatrix} 5 & 0 \\ 1 & 4 \end{bmatrix} \rightarrow 5014$$

برای الفبایی با $m$ حرف $m^{n^2}$ ماتریس $n \times n$ وجود دارد. بنابراین اندازه کلید دودویی[2] حداکثر $\log_2(m^{n^2})$ خواهد بود. البته این باند دقیق نیست زیرا تمامی ماتریس‌های شمرده شده معکوس پذیر نیستند. توسط قضیه باقی‌مانده چینی[3] می‌توان کران بهتری برای این عدد پیدا کرد.

برای بهبود این شیوه از رمزنگاری می‌توان ماتریس کلید را با استفاده از روش‌های زیر بهبود داد:

- استفاده از حداقل ۰
- استفاده‌ی تصادفی از تمامی اعداد
- بزرگ بودن اندازه‌ی ماتریس

به علت محاسبات زیاد این نوع رمزنگاری برای $n$ های بالاتر از۲، لستر هیل و همکارش ماشینی درست کردند[4] که برای $n = 6$ و ۲۶ کارکتر عملیات رمزنگاری و رمزگشایی را انجام دهد. متاسفانه برای هر ماشین ماتریس کلید ثابت بود و به علاوه به علت شکستن راحت رمزها در صورتی که از یک عملیات غیرخطی استفاده نمی‌شد، ماشین او به اندازه‌ی کافی نفروخت.

---

1 حملهٔ متن آشکار (Known-plaintext (KPA)) یک مدل حمله برای تحلیل رمز است، جایی که مهاجم دارای نمونه‌هایی از متن آشکار و نسخهٔ رمز شدهٔ آن است.

2 اندازه کلید دودویی یا طول کلید، تعداد بیت‌هایی است که بیانگیر کلید یک الگوریتم رمزنگاری است. این طول عموما رابطه‌ی مستقیمی با پیچیدگی و سختی رمزگشایی یک الگوریتم رمزنگاری دارد. عموما برای حدس کلید از الگوریتم‌های brute force استفاده می‌شود که با افزایش طول کلید کارایی این الگوریتم‌ها کاهش شدیدی پیدا می‌کند.

3 For more information, visit: www.en.wikipedia.org/wiki/Chinese_remainder_theorem

4 U.S. Patent 1,845,947

# پیاده‌سازی توسط زبان برنامه‌نویسی Python

برای مشاهده و استفاده از پیاده‌سازی به لینک زیر مراجعه کنید:

https://github.com/erfan-mehraban/hill_cipher

# منابع


http://practicalcryptography.com/cryptanalysis/stochastic-searching/cryptanalysis-hill-cipher/

http://practicalcryptography.com/ciphers/hill-cipher/

http://practicalcryptography.com/cryptanalysis/text-characterisation/quadgrams/

https://robalaban.com/journal/hill-cypher-python/

https://en.wikipedia.org/wiki/Hill_cipher

https://en.wikipedia.org/wiki/Chinese_Remainder_Theorem