

暗号の仕組み

情報科学の世界 2

2024 年度前期

佐賀大学理工学部 只木進一

- ① 近代以前の暗号
- ② 暗号の要素
- ③ 鍵と暗号
- ④ RSA (Riverst-Shamir-Adleman) 暗号
- ⑤ 課題

atbash 暗号: 旧約聖書

- 人類が秘密を持つようになって以来、暗号が出現
- 宗教団体が弾圧を逃れるために、重要情報を暗号化
- 旧約聖書: 紀元前 5 世紀
 - 重要な都市名のアルファベットを置き換え

スパルタの暗号: scytale 暗号

- 戦争の際にも暗号が必要
 - 前線に作戦を指令
 - 前線の状況を司令部に報告
 - 文書を持った兵士が走る・乗馬
- scytale 暗号: 紀元前 5 世紀
- 皮に書いた文字を円筒に巻き付ける
- 数文字毎に読み解く

<https://ja.wikipedia.org/wiki/%E3%82%B9%E3%82%AD%E3%83%A5%E3%82%BF%E3%83%AC%E3%83%BC>

Caesar の暗号

- 紀元前 1 世紀
- アルファベットの先頭から鍵の文字列に置き換える
- 残りは、鍵の終端の後ろに残ったアルファベットを順番に対応させる
- 例: 鍵 JULISCAER

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

上杉暗号: 16世紀

- いろはを数字にコード化

	七	六	五	四	三	二	一
一	ゑ	あ	や	ら	よ	ち	い
二	ひ	さ	ま	む	た	り	ろ
三	も	き	け	う	れ	ぬ	は
四	せ	ゆ	ふ	ゐ	そ	る	に
五	す	め	こ	の	つ	を	ほ
六	ん	み	え	お	ね	わ	へ
七		し	て	く	な	か	と

近代以前の暗号の弱点

- 文字の置き換えが固定
- 文字の出現頻度から推測できる
 - 英語で一文字の単語: "a"と"I"
 - 二文字単語が推測できる: "an"、"in"、"if"
 - いまでは、単語の出現頻度も知られている

- 現代英語出現頻度

`https:`

`//user.keio.ac.jp/~rhotta/hellog/2010-03-01-1.html`

第二次世界大戦中の暗号

- Enigma (1918)
<https://www.cryptomuseum.com/crypto/enigma/>
- Lorenz SZ-40/42(1941)
<https://www.cryptomuseum.com/crypto/lorenz/sz40/>
- Bombe (1939)
<https://www.cryptomuseum.com/crypto/bombe/>
- COLOSSUS (1943)
<http://www.cryptomuseum.com/crypto/colossus/>
- Alan Turing and Bletchley Park
<https://www.nationalww2museum.org/war/articles/alan-turing-betchley-park>

暗号方式と暗号鍵

- 暗号の方式
 - どういう方法で文字を置き換えるのか
 - 暗号の鍵
 - 何文字ずらす
 - 何文字置きに読む

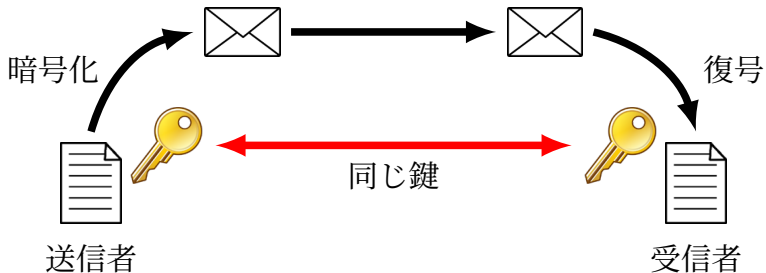
基本的用語

- 符号化、暗号化: Encode, Encipher, Encrypt
 - 平文テキスト (plain text) を暗号テキスト (cipher text) にする
- 復号化: Decode, Decipher, Decrypt
 - 暗号テキストを平文テキストに戻す

cipher /'saɪfə(r)/ a secret way of writing, especially one in which a set of letters or symbols is used to represent others.

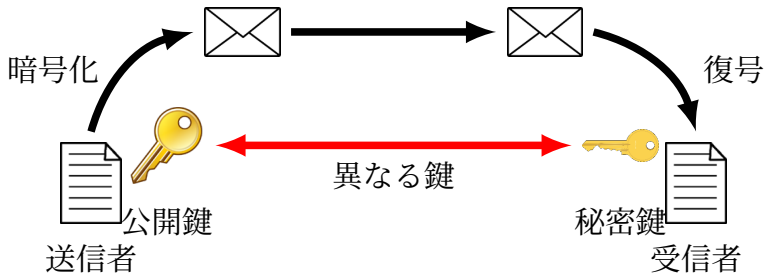
共通鍵暗号

- 鍵を送信者と受信者が共有する方法
 - 符号化と復号化で同じ鍵
 - どうやって鍵を送る？



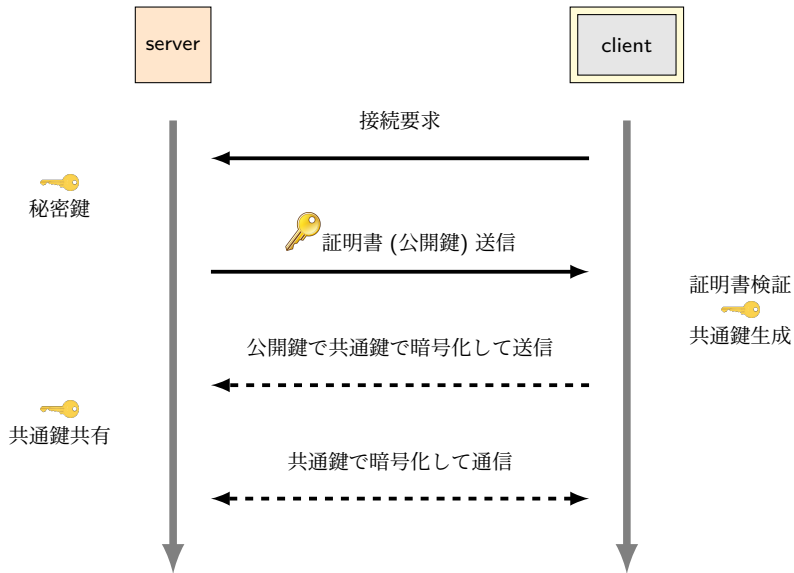
公開鍵暗号

- 鍵が送信者と受信者で異なる方法
 - 符号化と復号化が異なる鍵
 - 一方向にしか送れない



SSL: Secure Socket Layer

- HTTPS で利用している暗号化方式
 - 現在は TLS (Transport Layer Security) を使用
- 公開鍵と共通鍵を併用
- Web の証明書提示



復号できない暗号: パスワード

- ユーザが入力したものを符号化し、保存しているものと比較
- 攻撃手法
 - ユーザ名、名前、生年月日、英単語などをヒントに
 - 総当たり

RSA (Riverst-Shamir-Adleman) 暗号

- 整数論という数学の応用
- 因数分解が困難であることに基づく
- 公開鍵暗号に利用される
- James H. Ellis (1969) 及び Clifford Cocks (1973) が理論的基礎を発見したが、長く秘密にされていた
- 1977 年に RSA が公表。

整数の合同: Congruence

- 二つの整数 a と b が合同: ある整数 m で除した余りが等しい
 - a と b は法 m について合同: $a \equiv b \pmod{m}$
- 例: $m = 5$

$$7 \equiv 2 \pmod{5}$$

$$8 \equiv 3 \pmod{5}$$

$$11 \equiv 1 \pmod{5}$$

$$3 \equiv 3 \pmod{5}$$

整数の和

- $a \equiv a' \pmod{m}$ かつ $b \equiv b' \pmod{m}$ ならば
 $a + b \equiv a' + b' \pmod{m}$

$$a = n_a m + a'$$

$$b = n_b m + b'$$

$$\begin{aligned} a + b &= (n_a m + a') + (n_b m + b') \\ &= (n_a + n_b) m + (a' + b') \end{aligned}$$

- 例: $m = 11$

$$13 \equiv 2 \pmod{11}$$

$$15 \equiv 4 \pmod{11}$$

$$\begin{aligned} 13 + 15 &= 28 = 2 \times 11 + 6 \\ &\equiv 6 \pmod{11} \end{aligned}$$

整数の積

- $a \equiv a' \pmod{m}$ かつ $b \equiv b' \pmod{m}$ ならば
 $ab \equiv a'b' \pmod{m}$

$$a = n_a m + a'$$

$$b = n_b m + b'$$

$$\begin{aligned} ab &= (n_a m + a')(n_b m + b') \\ &= (n_a n_b m + n_a b' + n_b a') m + a'b' \end{aligned}$$

- 例: $m = 7$

$$8 \equiv 1 \pmod{7}$$

$$10 \equiv 3 \pmod{7}$$

$$\begin{aligned} 8 \times 10 &= 80 = 7 \times 11 + 3 \\ &\equiv 3 \pmod{7} \end{aligned}$$

Fermat の小定理

- p を素数、整数 $1 < a < p$ とする: $a \not\equiv 0 \pmod{p}$
- このとき: $a^{p-1} \equiv 1 \pmod{p}$
- 例を示す: $p = 11$ 、 $a = 3$

$$3^2 \equiv 9 \pmod{p}$$

$$3^4 \equiv 81 \pmod{p} \equiv 4 \pmod{p}$$

$$3^8 \equiv 16 \pmod{p} \equiv 5 \pmod{p}$$

$$\begin{aligned} 3^{10} &\equiv (3^2 \times 3^8) \pmod{p} \equiv 45 \pmod{p} \\ &\equiv 1 \pmod{p} \end{aligned}$$

- \pmod{p} のみに注目し、演算を簡素化

Fermat の小定理: 応用

- p と q を素数、 a を pq と互いに素とする
- このとき: $a^{(p-1)(q-1)} \equiv 1 \pmod{pq}$
- 例: $p = 5$ 、 $q = 7$ 、 $a = 11$

$$11^2 \equiv 121 \pmod{35} \equiv 16 \pmod{35}$$

$$11^4 \equiv 256 \pmod{35} \equiv 11 \pmod{35}$$

$$11^8 \equiv 16 \pmod{35}$$

$$11^{16} \equiv 256 \pmod{35} \equiv 11 \pmod{35}$$

$$\begin{aligned} 11^{4 \times 6} &= 11^{16+8} \equiv (11 \times 16) \pmod{15} \\ &\equiv 1 \pmod{35} \end{aligned}$$

秘密鍵と公開鍵

- 受信者
 - 二つの大きな素数 p と q を生成し、秘密鍵とする。
 - $m = pq$
 - $\phi(m) = (p - 1)(q - 1)$
 - k : $\phi(m)$ と互いに素である適当な自然数
- m と k を公開鍵とする
 - m を因数分解して p と q を得ることが難しい

送信者によるメッセージ暗号化

- m は L ビットであるとする
- メッセージ M を $L-1$ ビット毎の語に区切る

$$M = a_0a_1 \cdots a_n$$

- 各語を変換

$$b_i \equiv a_i^k \pmod{m}$$

$$M' = b_0b_1 \cdots b_n$$

- M' を送信

受信者による復号

- $kv - \phi(m)u = 1$ の適当な解 (u, v) を得る

$$\begin{aligned} b_i^v &\equiv a_i^{kv} \pmod{m} \equiv a_i^{1+\phi(m)u} \pmod{m} \\ &\equiv (a_i \pmod{m}) \left(a_i^{\phi(m)} \pmod{m} \right)^u \\ &\equiv (a_i \pmod{m}) (1 \pmod{m})^u \\ &\equiv a_i \pmod{m} \end{aligned}$$

- 復号完了

例

- 秘密鍵: $p = 13, q = 11, \phi(m) = 120$
- 公開鍵: $m = 143, k = 7$
- m は 8 ビット
 - 7 ビット毎の語に分離
- $kv - \phi(m)u = 1$ の解 $(u, v) = (6, 103)$

逆方向の符号化

- 秘密鍵を使って符号化

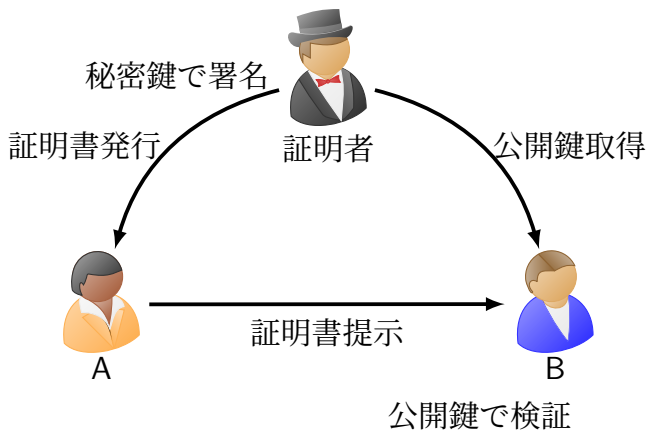
$$b_i \equiv a_i^v \pmod{m}$$

- 公開鍵を使って復号

$$\begin{aligned} b_i^k &\equiv a_i^{kv} \pmod{m} \equiv a_i^{1+\phi(m)u} \pmod{m} \\ &\equiv (a_i \pmod{m}) \left(a_i^{\phi(m)} \pmod{m} \right)^u \\ &\equiv (a_i \pmod{m}) (1 \pmod{m})^u \\ &\equiv a_i \pmod{m} \end{aligned}$$

- 公開鍵で復号できるため、暗号にはならない!

デジタル証明書



数学的裏付けのある暗号

- 確実に符号化・復号化ができる
 - 数学的に保証されている
- 方式は公開／鍵は非公開
- 素数への因数分解が困難
 - 今のところ高速なアルゴリズムなし
- コンピュータの高速化によって、長い鍵が必要になっている

課題

https では Web サーバの証明書をブラウザが受信します。その証明書の真正性はどのように担保されているのでしょうか。