公開鍵暗号方式について

形式化数学研究室
宮島啓一

- 公開鍵暗号方式とは
- RSA公開鍵暗号方式について

共通鍵暗号と公開鍵暗号

共通鍵暗号方式（Common Key Cryptosystem）

送信者と受信者が同じ秘密鍵を共有する暗号方式

暗号化と複合化で同じ鍵を用いる方式

例）ブロック暗号、ストリーム暗号等

公開鍵暗号方式（Public Key Cryptosystem）

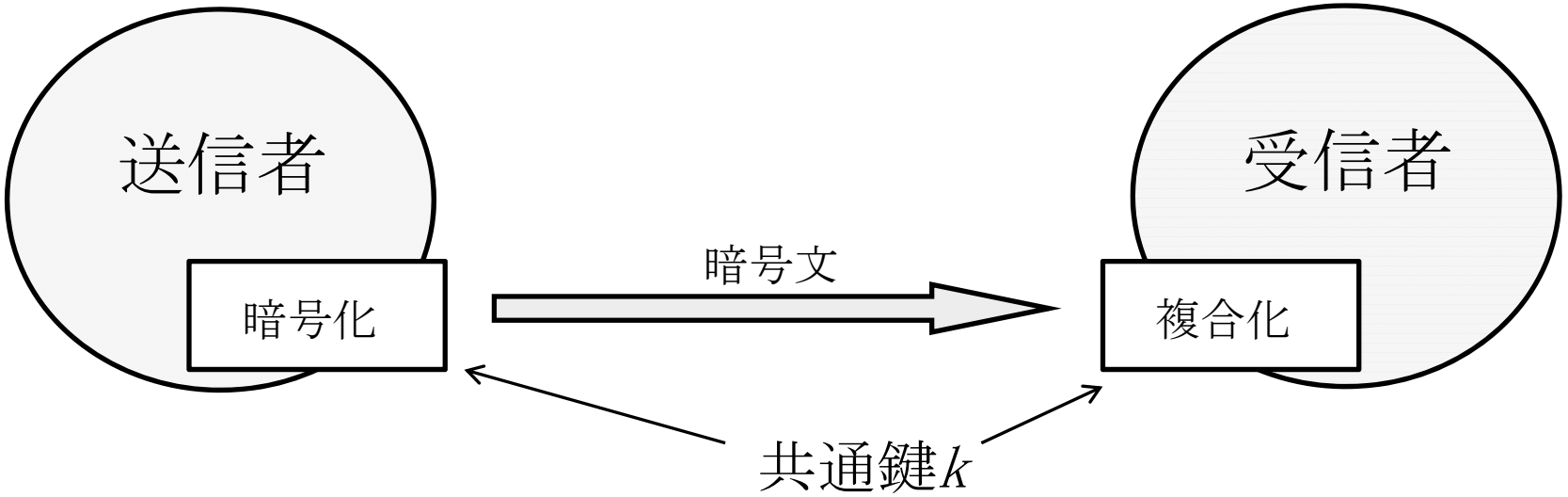
送信者は公開されている鍵で暗号化し、受信者は秘密にされている鍵で複合化する暗号方式

例）RSA暗号等

共通鍵暗号方式
(Common Key Cryptosystem)

送信者と受信者が同じ秘密鍵kを共有する暗号方式を共通鍵暗号方式という。

公開鍵暗号方式の登場までの暗号方式は全て共通鍵暗号方式であった。



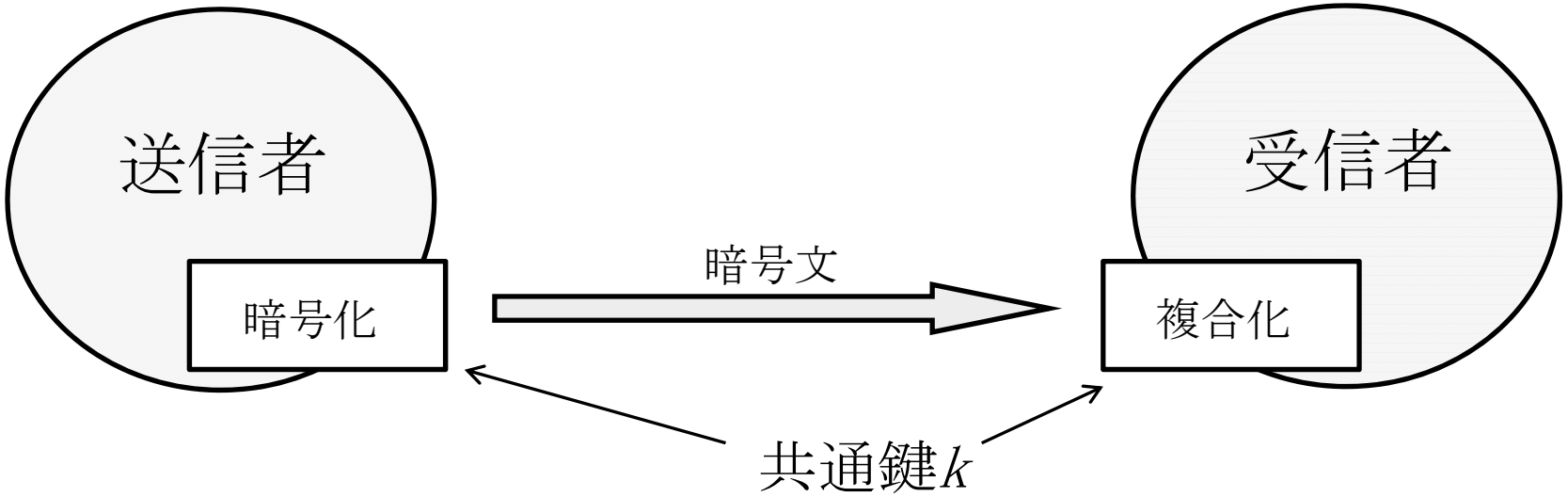
共通鍵暗号方式
(Common Key Cryptosystem)

鍵配送の問題(デメリット)

送信者と受信者が第三者に知られる事なく同じ秘密鍵を共有しなくてはならない。

↓

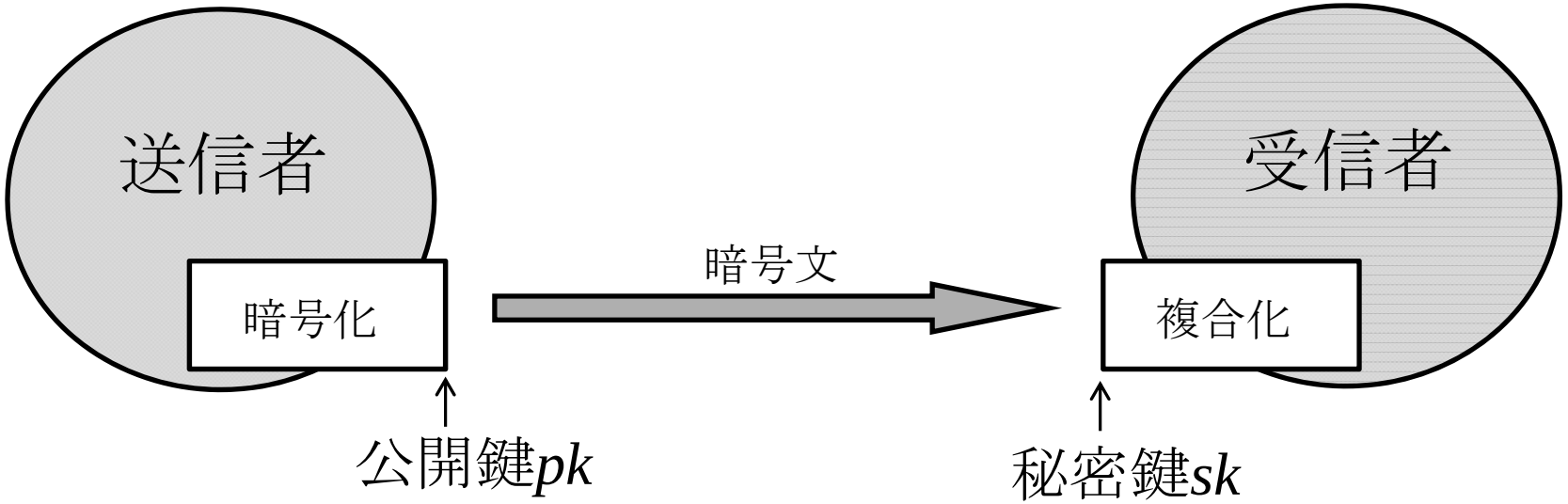
1976年、暗号史上最大のブレイクスルーである公開鍵暗号方式の考案



公開鍵暗号方式
(Public Key Cryptosystem)

公開鍵暗号方式についての最初の論文は1976年に、ホイットフィールド・ディフィーによって発表された。

暗号化に使用する鍵*pk*は公開してしまい複合化に使用する鍵*sk*のみ秘密にしておくというもの

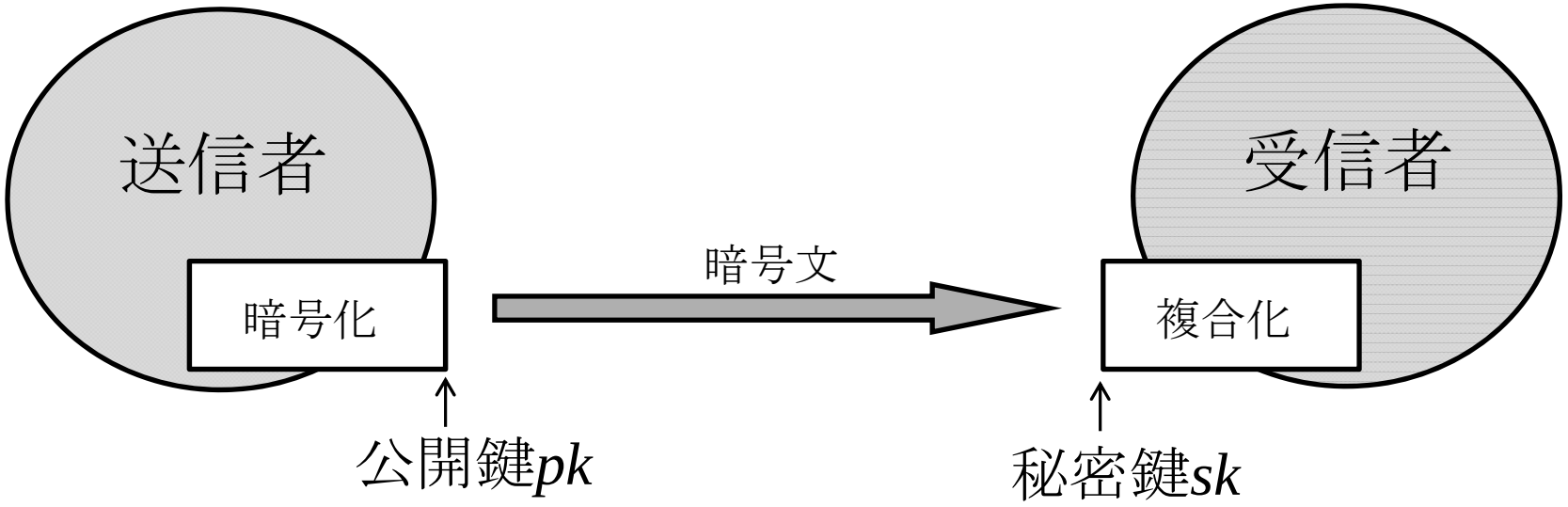


公開鍵暗号方式
(Public Key Cryptosystem)

暗号化は誰でもできるが複合化できるのは鍵の作成者である受信者のみ

↓

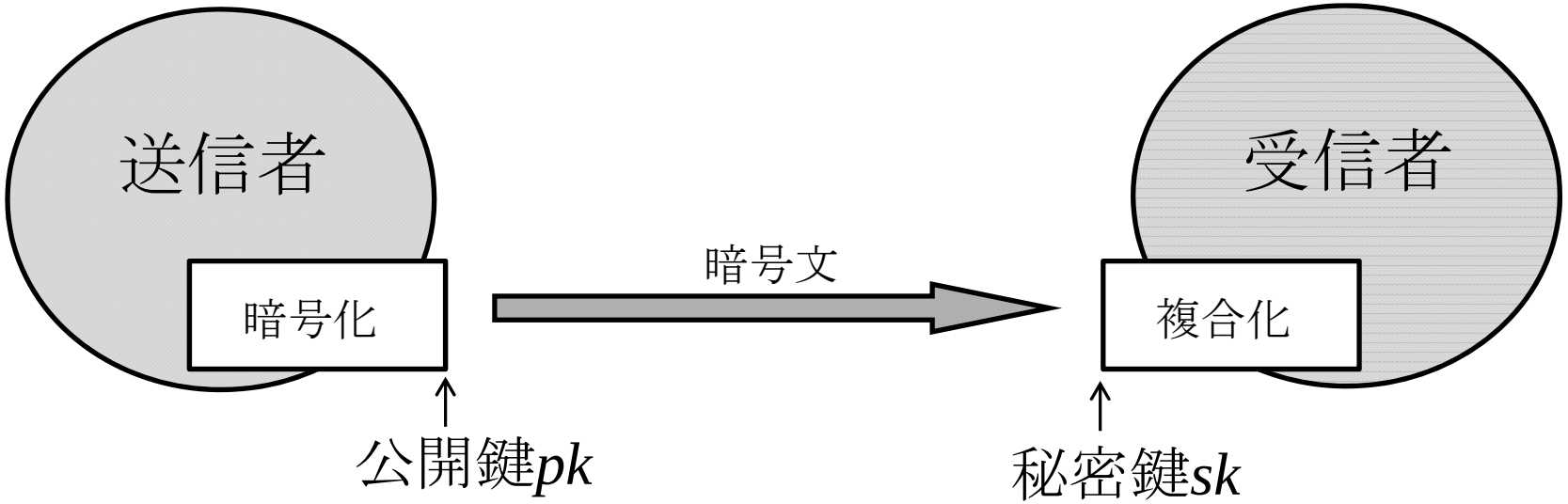
公開鍵暗号方式により、鍵配送の問題を解決できる。



公開鍵暗号方式
(Public Key Cryptosystem)

今までの共通鍵暗号方式と比較して、

・相手の数に関係なく公開鍵は一つで良い
・鍵の共有が容易でありかつ安全性が高い
といった利点がある。



- 公開鍵暗号方式とは
- RSA公開鍵暗号方式について
- ペピンの判定法の形式化

RSA暗号方式

最も典型的な公開鍵暗号方式がRSA暗号方式である

RSA暗号の場合、
「大きな素数同士の掛け算は簡単であってもその逆の素因数分解は難しい」
という性質を利用している

（一方向性関数の性質を利用している）

素因数分解仮定

2つの素数p、qを例えばp=257、q=251としたとき、合成数を求めるのは容易である。

$$p \times q = 257 \times 251 = 64507$$

この計算は容易

しかし、64507を素因数分解し、257と251という2つの素数を求めることは困難である。

この計算は困難

$$64507 = (\text{素因数分解}) = 257 \times 251$$

このような2つの素数の合成数を素因数分解する効率的なアルゴリズムは存在しないと予想されている。

RSA公開鍵暗号方式のアルゴリズム

[1]RSA暗号の公開鍵と秘密鍵の鍵生成アルゴリズム

① 二つの大きな素数p、qを生成し、$N = pq$を計算してNを導出する。

② $(p-1)(q-1)$とeの最大公約数が1となるeをランダムに選ぶ。

③ 拡張ユークリッドの互除法を適用し、
$ed \equiv 1(\mathrm{mod}(p-1)(q-1))$となる$d$を求める。

ここで導出した(N, e)を公開鍵として公開し、dは秘密鍵として保持しておく。

＊拡張ユークリッドの互除法
$a_0x + a_1y =$（a_0、a_1の最大公約数）となる整数x、yを求める多項式時間アルゴリズム

RSA暗号 (1)

鍵生成アルゴリズム

(1) 二つの素数 p、qをランダムに選び、$N=pq$とする

(2) $\gcd(e,(p-1)(q-1))=1$となるeをランダムに選ぶ。

(3) $ed=1\bmod(p-1)(q-1)$を満たす $d\ (>0)$を求める。

(4) 公開鍵を(N,e), dを秘密鍵とする。

(2)のgcdとは、ユークリッドの互換方といい、二つの正整数の最大公約数を求めるアルゴリズムである

RSA暗号 (2)

暗号化アルゴリズム

平文の集合をZ_Nとする。公開鍵(N,e),および平分$m \epsilon Z_N$に対し,
暗号文 c を

$$c = m^e \bmod N$$

と計算する。

RSA暗号(3)

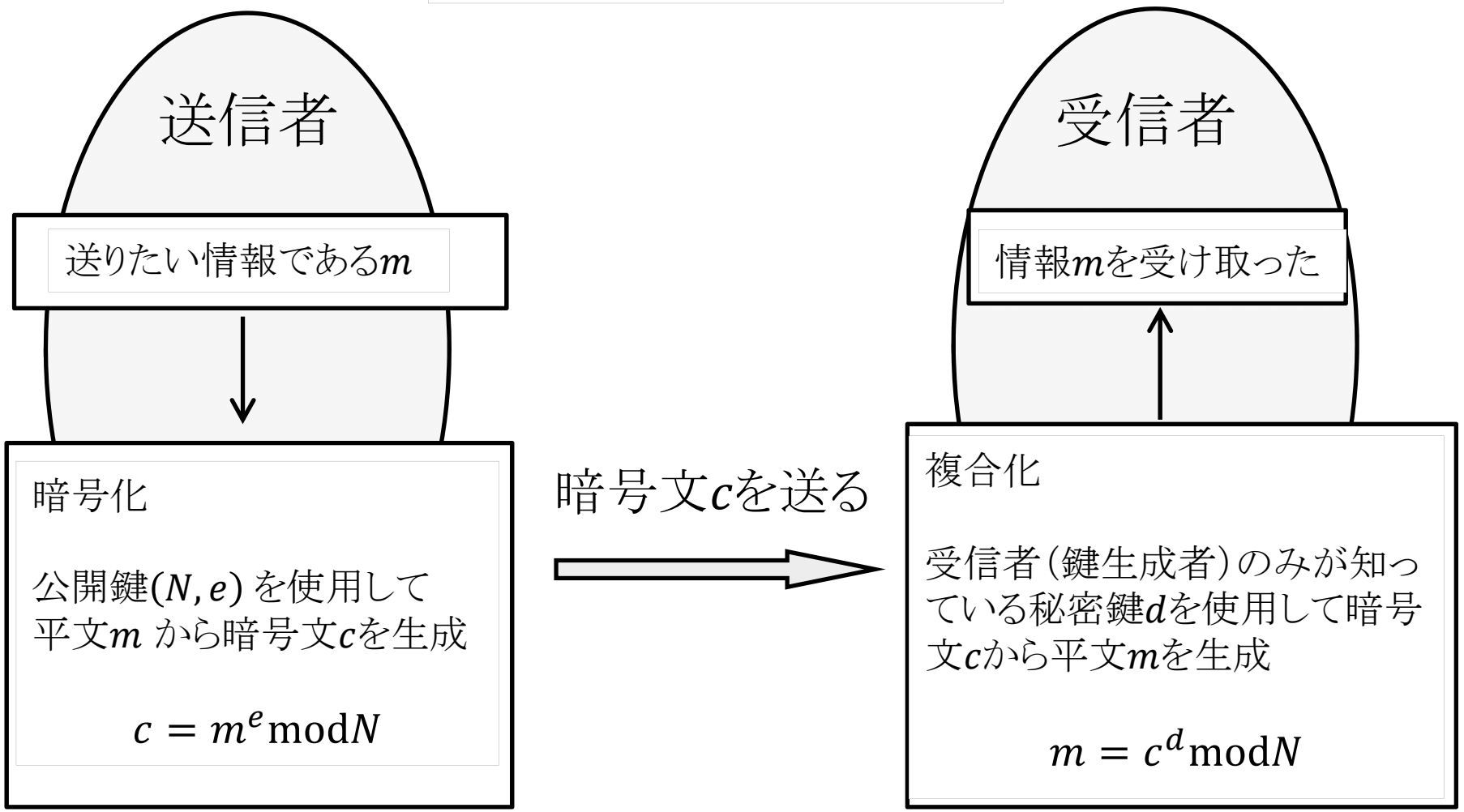
復号アルゴリズム

秘密鍵d、*および暗号文*$c \in Z_N$*から、平分*m*を*

$$m = c^d \bmod N$$

と計算する。

RSA公開鍵暗号方式のアルゴリズム



RSA公開鍵暗号方式の安全性

敵が暗号文を解読するためには秘密鍵dを求めなくてはならない。

しかし、大きな素数からなる素因数分解の困難さにより敵は公開されているNからp、qを求めることはできない。

p、qが求められないのなら

$$ed = 1(\mathrm{mod}(p-1)(q-1))$$

を計算することはできないため秘密鍵dを求めることはできない。

RSA方式の実例

RSA方式の実例を紹介する。前項までで紹介したとおりの条件で実際に値を代入する。

p=11,q=37とするとN=p×q=407となる。また、公開鍵を7、秘密鍵を103とし、通信文を234とする。

$$234^{7} \bmod 407 \equiv (234^{5} \times 234^{2}) \bmod 407$$
$$\equiv (155 \times 218) \bmod 407 \equiv 9$$

となり、暗号文9を導き出せた。また、復号化は

$$9^{103} \bmod 407 \equiv 234$$

より、導ける。

RSA公開鍵暗号方式の安全性

素因数分解は大きい数であればあるほど困難になる

↓

つまり、RSA暗号の強度は使用される2つの素数の大きさに依存する

RSA公開鍵暗号方式の安全性

NTTは2009年12月、海外の研究機関と共同で素因数分解問題で世界記録を更新

一般数体ふるい法で768ビット、10進232ケタの素因数分解に成功

768bit合成数の素因数分解に成功、NTTらが世界記録更新 -INTERNET Watch:
http://internet.watch.impress.co.jp/docs/news/20100108_341353.html

RSA公開鍵暗号方式の安全性

- 内閣官房情報セキュリティセンター「情報セキュリティ政策会議」にて1024ビットのRSA暗号の安全性低下を指摘

平成24年10月26日改定　情報セキュリティ対策推進会議決定
政府機関の情報システムにおいて使用されている暗号アルゴリズムSHA-1及びRSA1024に係る移行指針
http://www.nisc.go.jp/active/general/pdf/angou_ikoushishin.pdf

- Googleでは2013年に2048ビットRSA暗号化キーへのアップグレードを完了

グーグル、2048ビットRSA暗号化キーへのアップグレードを完了 - CNET Japan:
http://japan.cnet.com/news/service/35040219/

- LINEは2014年6月に暗号化に2048ビットRSAを採用していることを明かした

LINEは2048ビットRSA採用 「暗号化が弱いためデータ流出の可能性」は「誤解」と技術ブログで説明 - ITmedia ニュース:
http://www.itmedia.co.jp/news/articles/1406/26/news113.html

RSA公開鍵暗号方式の安全性

RSA公開鍵暗号方式の安全性を維持するには大きい素数を見つけていく必要がある。

（2016年1月 現在では2,233万8,618桁）

(2進数では74,207,281bit)

参考文献

・現代暗号への招待

サイエンス社　著者　黒澤　馨

・Aiichi Yamasaki’s Homepage

http://www.math.h.kyoto-u.ac.jp/~yamasaki/index.php?a3f

ご清聴ありがとうございました。