暗号のための基礎理論

田中　圭介

暗号理論を理解するための基礎について解説する．はじめに，暗号理論における計算モデルや，暗号システムの安全性を含む，暗号理論でよく用いられる概念について解説する．つぎに，現代の暗号理論の重要な理論的土台である一方向性関数について説明する．さらに，暗号理論を構成する4つの重要な要素について解説する．おわりに，暗号理論をさらに知るための参考文献について紹介する．

キーワード：暗号理論，公開鍵暗号，共通鍵暗号，安全性，効率，帰着

1. はじめに

暗号という用語は広義で用いられる場合と狭義で用いられる場合がある．狭義の暗号とは，送信者と受信者が第三者に内容がわからないように秘密に通信するための手段を指す．この狭義の暗号は秘匿通信とも呼ばれる．広義の暗号とは，狭義の暗号や署名，認証などを含む研究分野のことを指す．本特集や本稿のタイトル中の暗号という用語は広義で用いている．

このような二通りの用いられ方は，公開鍵暗号や共通鍵暗号という用語中の暗号についても当てはまる．分野としての公開鍵暗号（public-key cryptosystem, asymmetric-key cryptosystem）が関わる最も重要な要素としては，公開鍵暗号（公開鍵秘匿通信，public-key encryption scheme）と署名（signature scheme）がある．分野としての共通鍵暗号（symmetric-key cryptosystem, private-key cryptosystem）が関連する最も重要な要素としては，共通鍵暗号（共通鍵秘匿通信，symmetric-key encryption scheme）とメッセージ認証コード（message authentication code）がある．

この4つ以外にも，様々な重要な基本技術があるが，本稿ではこれらを網羅的に取り上げることはせずに，この4つの重要な基本要素をわかりやすく説明することを目指すとともに，現代の暗号理論の重要な理論的土台である一方向性関数および落し戸関数についても説明する．

たなか　けいすけ
東京工業大学 大学院情報理工学研究科
〒152-8552 目黒区大岡山 2-12-1-W8-55

2. 暗号で用いられる概念

2.1 参加者，敵，計算モデル

ここで挙げた4つの基本要素はいずれも2人で通信を行うことにより目的が実現される．この通信に参加する人は，暗号理論においては確率的多項式時間アルゴリズム，より正確には，確率的多項式時間 Turing 機械として定式化される．これは，最適化などの計算問題のためのアルゴリズムと同様の計算モデルである．例えば，Alice と Bob という2人が安全に文書をやり取りしたい状況を考える．まず，Alice は文書を，Bob と共有している秘密情報を用いて暗号化する．次に，その暗号化したデータを通信路を用いてBに送信する．最後に，データを受け取ったBはそのデータを，Alice と共有している秘密情報を用いて復号する．

この Alice と Bob のやり取りに要求されることは主に2つある．ひとつは安全性であり，もうひとつは効率である．

効率については，最適化アルゴリズムなどと同様にアルゴリズムの解析が行われる．暗号理論におけるアルゴリズム解析の特徴としては，確率的なアルゴリズムが用いられることが多いため出力の分布の解析がよく行われたり，最悪ケースの解析に加えて平均ケースの理論的および実験的解析がよく行われたりすることが挙げられる．

安全性については，最適化アルゴリズムなどでは対象とされない性質である．この安全性では，Alice と Bob 以外に敵，あるいは攻撃者と呼ばれる第三者が存在し，Alice と Bob の通信路を盗聴や改ざんする状況を考える．なお，暗号理論における通信路のモデルにおいては，自然発生的エラーなどは起こらないと

通常は考える．この敵はやはり確率的多項式時間アルゴリズム，より正確には，通常，多項式サイズの論理回路として定式化される．Turing機械でなく論理回路として定式化されるのは，AliceとBobの通信ビット数や通信を行う状況から暗号システムが扱うデータのサイズがわかってしまうからである．すなわち，1024ビットの暗号システムがよく用いられる状況では，敵はこの1024ビットに攻撃を最適化するはずだからである．

2.2 アルゴリズム公開，鍵，全数探索，ワンタイム・パッド

このようなAliceとBobのやり取りは共通鍵暗号と呼ばれる．また，AliceとBobが共有している秘密情報は鍵（共通鍵，秘密鍵）と呼ばれる．よって鍵とは暗号化アルゴリズムおよび復号アルゴリズムで使用される情報の一部である．暗号理論においては，この鍵以外の情報，すなわち，暗号化アルゴリズムおよび復号アルゴリズムは敵に公開されていると考える．このような設定は暗号を商業的に用いるためなどからきている．

鍵をもっている人は誰でも文書の暗号化や復号ができるので，鍵の機密性を維持することは暗号化された文書の安全にとってきわめて重要である．鍵が短すぎると，鍵のとり得る範囲を全数探索することで発見されてしまう．

鍵を単に長くするだけで安全性が得られるとは一般的には言えないが，一定の長さ以上にすれば安全性が得られることも知られている．メッセージ全体と同じ長さの鍵はワンタイム・パッド（one-time pad）と呼ばれる．本質的に，ワンタイム・パッドの鍵の各ビットは文書の1ビットを暗号化するために一度だけ使用され，その後，鍵のそのビットは廃棄される．ワンタイム・パッドの主な問題点は，大量の通信が予想されるときに鍵が極端に長くなることである．ほとんどの用途でワンタイム・パッドは実用的とは考えにくい．よって暗号理論では，全数探索を回避できる適度な長さ，例えば，128ビットの鍵で安全な共通鍵暗号をつくることを考察する．

2.3 安全性証明，仮定，帰着

適度な長さの鍵による暗号が，仮定なしで安全であることを数学的に証明する方法は今のところ知られていない．これは，重要な未解決問題であるP vs. NP問題が解決されない限りは難しいと考えられている．鍵を見つける問題は全数探索で解けるので，非決定性Turung機械を用いれば多項式時間で解ける．すなわち，鍵を多項式時間で発見できないことが証明できればP≠NPを証明したことになる．

計算複雑さの理論においては，解くのが難しい問題に対して多項式時間帰着によって，NP完全性などの解くことが難しいことの状況証拠を与えてきた．暗号理論でも計算複雑さの理論と同様なアプローチをとりたいが，計算複雑さの理論でよく用いられるNP完全性は最悪ケースの解析である．NP完全性は最悪ケースにおける複雑さを対象としているため，一般的には，ある問題がNP完全であっても平均ケースではその問題が効率的に解ける場合がある．

暗号理論では最悪ケースでの計算複雑さよりも平均ケースでの計算複雑さを評価したい．公開鍵暗号でよく使われる計算問題のひとつに素因数分解問題がある．この問題は平均ケースで解くのが難しいと信じられている問題である．暗号理論では，ある公開鍵暗号を破るような敵を利用すれば，素因数分解問題が効率的に計算できることが証明されている．

3. 一方向性関数

本節では，一方向性関数（one-way function）および落し戸関数（trapdoor function）という現代の暗号理論の重要な理論的土台について述べる．暗号理論の基礎として計算複雑さの理論を利用することの利点の一つとして，安全性証明の際に用いる仮定を明確にすることがある．例えば，一方向性関数の存在を仮定すると，一定レベルの安全な秘密鍵暗号をつくることができたり，落し戸関数の存在を仮定すると，一定レベルの安全な公開鍵暗号をつくることができる．

関数fに対して，すべてのwに対して，wと$f(w)$の長さが等しいとき，fは長さ不変（length-preserving）であるという．長さ不変の関数は置換である場合がある．

確率的Turing機械Mが入力wで動作を始め，xを出力して停止するときの確率を

$$Pr[M(w)=x]$$

で表す．

一方向性関数を定義する．直観的には，ある関数の計算は容易だが，その逆関数の計算がほとんどすべての場合に困難であるとき，その関数は一方向性関数であると定義される．以下の定義で，fは効率的に計算可能であり，Mはfの逆関数を計算しようとする確率的多項式時間アルゴリズムである．一方向性置換の

定義の方が簡単なのでまず一方向性置換から定義する．

一方向性置換（one-way permutation）は以下の二つの性質をもつ長さ不変の置換 g である．

1．多項式時間で計算可能である．

2．すべての確率的多項式時間 Turing 機械 M，すべての k，十分大きな n に対して，長さ n のランダムな入力 w に対して M を動作させたとき，

$$Pr_{M,w}[M(g(w))=w]\leq n^{-k}$$

である．ここで，$Pr_{M,w}$ は，M のコイン投げおよび w のランダムな選択の上で確率がとられることを意味する．

一方向性関数（one-way function）は以下の二つの性質をもつ長さ不変の関数 f である．

1．多項式時間で計算可能である．

2．すべての確率的多項式時間 Turing 機械 M，すべての k，十分大きな n に対して，長さ n のランダムな入力 w に対して M を動作させたとき，

$$Pr[M(f(w))=y,\ \text{ただし}\ f(y)=f(w)]\leq n^{-k}$$

である．

一方向性置換では，g の逆関数の計算を，どんな確率的多項式時間アルゴリズムを用いても小さな確率でしかできない．すなわち，$g(w)$ から w は計算できそうにない．一方向性関数でも同様に，$f(w)$ に写像されるような y を見つけることは，どんな確率的多項式時間アルゴリズムを用いても計算できそうにない．

例えば，乗算関数 Mult は一方向性関数の有力な候補である．任意の $w\in\{0,1\}^*$ に対して $\mathsf{Mult}(w)$ を w の前半部分と後半部分の積を表す文字列とする．形式的には

$$\mathsf{Mult}(w)=w_1\cdot w_2$$

である．ここで，$w=w_1|w_2$（w_1 と w_2 を単に連結したもの）であり，$|w_1|=|w_2|$，もしくは $|w|$ が奇数の場合には $|w_1|=|w_2|+1$ とする．ここでは，文字列 w_1 および w_2 を 2 進数として扱うものとする．Mult の逆関数を計算する確率的多項式時間アルゴリズムは現在まで知られていない．

落とし戸関数は，落とし戸情報を用いなければ一方向性関数と同じであるが，落とし戸情報を用いれば逆関数が確率的多項式時間アルゴリズムで計算できるというような関数である．落し戸関数は，公開鍵暗号の構成に直接的に利用される．

4. 暗号における基本要素

4.1 公開鍵暗号

Alice と Bob による共通鍵暗号では，Alice と Bob は秘密に共有する鍵を利用して通信を行った．公開鍵暗号では Alice と Bob は異なる鍵を用いる（ただし，これらの鍵は同時につくる必要がある）．さらに，送信者が用いる鍵は公開してもかまわないという，とても重要で役に立つ性質をもつ．

以下の定義には，暗号でよく用いられる無視できる（negligible）という用語が現れる．

すべての多項式 $p(\cdot)$ に対して，次のような N が存在するならば，関数 negl は無視できるという：

すべての数 $n>N$ に対して $\mathsf{negl}(n)<1/p(n)$ である．

この無視できる関数は，敵がとても低い確率でしか成功できないことを表すのに主には用いられる．

公開鍵暗号（公開鍵秘匿通信）方式は以下のような確率的多項式時間アルゴリズムの 3 つ組（Gen，Enc，Dec）である．

1．鍵生成アルゴリズム Gen は，セキュリティーパラメーター 1^n を入力としてとり，鍵の組 (pk, sk) を出力する．pk は公開鍵と呼び，sk は秘密鍵と呼ぶ．

2．暗号化アルゴリズム Enc は，公開鍵 pk と文書 m を入力としてとり，暗号文 c を出力する．文書 m は pk に依存して平文空間からとられる．

3．復号アルゴリズム Dec は，秘密鍵 sk と暗号文 c を入力としてとり，文書 m あるいは，エラーを表す特殊な記号 $\perp$ を出力する．通常，Dec は確率的ではなく決定的なアルゴリズムが用いられる．

公開鍵暗号方式の通常の利用イメージは図 1 のとおりである．まず，受信者が鍵生成アルゴリズム Gen を実行し公開鍵と秘密鍵を生成する．つぎに，公開鍵を受信者に送る．その後，送信者が暗号化アルゴリズム Enc を用いて暗号文を生成し受信者に送る．最後に，受信者は復号アルゴリズム Dec を用いて文書を得る．

Enc_{pk} は落とし戸関数であることが要求される．この公開鍵暗号方式は基本的性質として，暗号化された文書が正しく復号できるような性質をもたなくてはならない．これは，すべての n，$\mathsf{Gen}(1^n)$ によって出力されるすべての (pk, sk)，適切な平文空間からの文書 m に対して，

$$Pr[\mathsf{Dec}_{sk}(\mathsf{Enc}_{pk}(m))\rightarrow m]\geq 1-\mathsf{negl}(n)$$

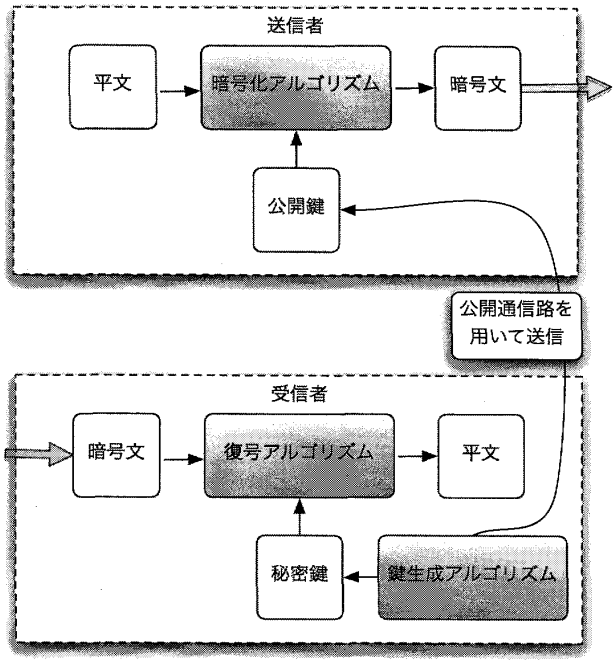

図1　公開鍵暗号方式

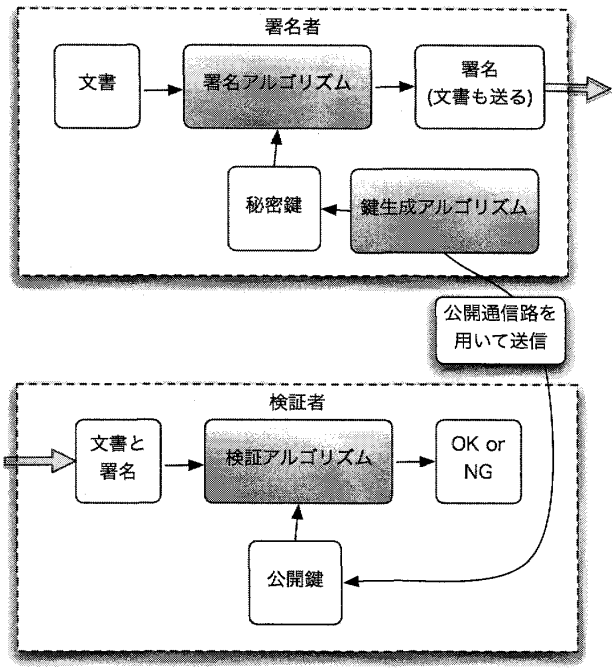

図2　署名方式

が成り立つような無視できる関数 negl が存在するという条件で定式化できる．

この基本的性質に加え，公開鍵暗号方式は文書の秘匿性をはじめとする安全性が要求される．これについては高木氏による公開鍵暗号に関する本特集の記事を参照してほしい．

4.2　署名

ある種の公開鍵暗号方式は，署名方式としても利用できる．公開鍵暗号が，落とし戸置換である場合を考える．秘密鍵の所有者が復号アルゴリズムを文書に対して適用した後に，誰かに送ると，公開されている公開鍵を用いて暗号化アルゴリズムを適用することにより，文書が秘密鍵の所有者から送られてきたということを検証できる．

署名方式は以下のような確率的多項式時間アルゴリズムの3つ組（Gen, Sign, Verify）である．

1. 鍵生成アルゴリズム Gen は，セキュリティーパラメーター 1^n を入力としてとり，鍵の組 (pk, sk) を出力する．pk は公開鍵と呼び，sk は秘密鍵と呼ぶ．
2. 署名アルゴリズム Sign は，秘密鍵 sk と文書 m を入力としてとり，署名 σ を出力する．文書 m は任意のビット列 $\{0,1\}^*$ でよい．
3. 検証アルゴリズム Verify は，公開鍵 pk，文書 m，署名 σ を入力としてとり，1ビット b を出力する．$b=1$ で署名が有効であることを表し，$b=0$ で無効であることを表す．通常，Verify は確率的ではなく決定的なアルゴリズムが用いられる．

署名方式の通常の利用イメージは図2のとおりである．まず，署名者が鍵生成アルゴリズム Gen を実行し鍵を生成する．つぎに，公開鍵を検証者に送る．その後，署名者が署名アルゴリズム Sign を用いて署名を生成し，文書とともに受信者に送る．最後に，検証者は検証アルゴリズム Verify を用いて文書と署名の組の正しさを確かめる．

一方向性関数があれば安全な署名方式が構成できることが示されている．

この署名方式は基本的性質として，署名がつけられた文書が正しく検証できるような性質をもたなくてはならない．これは，すべての n，Gen(1^n) によって出力されるすべての (pk, sk)，すべての文書 $m \in \{0,1\}^*$ に対して，

$$\mathrm{Verify}_{pk}(m, \mathrm{Sign}_{sk}(m)) \to 1$$

が常に成り立つという条件で定式化できる．

この基本的性質に加え，署名方式には，署名の偽造不可能性をはじめとする安全性が要求される．要求される最も高い安全性の一つとして，適応的選択文書攻撃のもとでの適用的存在的偽造不可能性がある．これは，秘密鍵をもたない敵に，好きな文書を選ばせてその署名を渡したとしても，（ここで選んだ文書とそのときに渡した署名の組以外で）正しいと検証されるよ

うな文書と署名の組をつくることはできないという性質である．安全性についての詳細は本稿のおわりに挙げる文献を参照してほしい．

また，この署名方式は耐衝突ハッシュ関数（collision-resistant hash function）と組み合わせられて用いられることも多い．この性質は直観的には以下のとおりである．

ハッシュ関数 h が耐衝突であるとは，どんな確率的多項式時間アルゴリズムも，$h(x)=h(y)$ かつ $x \neq y$ であるような x, y を無視できる確率でしか見つけられないときをいう．

近年，SHA-1 や MD 5 などの秘密鍵暗号に関する要素を用いているハッシュ関数に対する攻撃の成功例が数多く報告されている．文書を署名アルゴリズムに入力する前に，まずこの耐衝突ハッシュ関数を通すことにより，文書の短いダイジェストをつくることができ，それによって，任意の長さの文書の入力が可能になる．

4.3 共通鍵暗号

2 節で紹介した Alice と Bob による共通鍵暗号を定式化する．

共通鍵暗号（共通鍵秘匿通信，秘密鍵暗号（秘密鍵秘匿通信））方式は以下のような確率的多項式時間アルゴリズムの 3 つ組（Gen，Enc，Dec）である．

1. 鍵生成アルゴリズム Gen は，セキュリティーパラメーター 1^n を入力としてとり，鍵 k を出力する（k は共通鍵，秘密鍵とも呼ばれる）．通常，$\mathrm{Gen}(1^n)$ は単に，$\{0,1\}^n$ から一様ランダムに k を選んで出力する．
2. 暗号化アルゴリズム Enc は，鍵 k と文書 m を入力としてとり，暗号文 c を出力する．文書 m は n に依存して決まる任意のビット列 $\{0,1\}^{\ell(n)}$ からなる平文空間からとられる．通常，Enc は確率的ではなく決定的なアルゴリズムが用いられる．
3. 復号アルゴリズム Dec は，鍵 k と暗号文 c を入力としてとり，文書 m を出力する．通常，Dec は確率的ではなく決定的なアルゴリズムが用いられる．

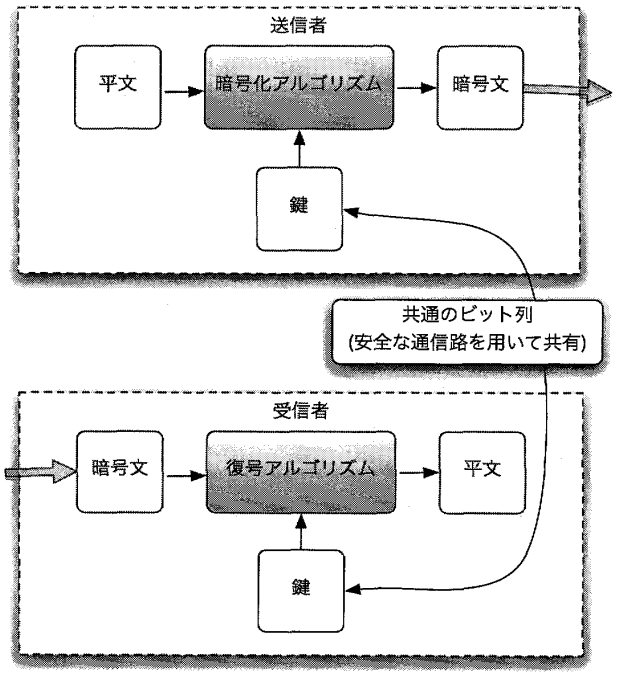

図3　共通鍵暗号方式

共通鍵暗号方式の通常の利用イメージは図 3 のとおりである．まず，送信者と受信者によって，一様ランダムに選んだ一定の長さのビット列を安全に共有する．次に，送信者が暗号化アルゴリズム Enc を用いて暗号文を生成し受信者に送る．最後に，受信者は復号アルゴリズム Dec を用いて文書を得る．

Enc_k は一方向性関数であることが要求される．

この共通鍵暗号方式は基本的性質として，公開鍵暗号方式と同様に，暗号化された文書が正しく復号できるような性質をもたなくてはならない．これは，すべての n，$\mathrm{Gen}(1^n)$ によって出力されるすべての k，適切な平文空間からの文書 m に対して，

$$\mathrm{Dec}_k(\mathrm{Enc}_k(m)) \rightarrow m$$

が常に成り立つという条件で定式化できる．

この基本的性質に加え，共通鍵暗号方式は公開鍵暗号と同様に，文書の秘匿性をはじめとする安全性が要求される．この共通鍵暗号方式に要求される安全性は，公開鍵暗号の安全性とは大きく異なる．これについては松井氏による共通鍵暗号に関する本特集の記事を参照してほしい．

4.4 メッセージ認証コード

メッセージ認証コード（MAC）は，秘密鍵暗号と関連した技術であり，署名者と検証者の鍵が共通であるような署名方式である．

メッセージ認証コード（MAC）は以下のような確率的多項式時間アルゴリズムの 3 つ組（Gen，Mac，Verify）である．

1. 鍵生成アルゴリズム Gen は，セキュリティーパラメーター 1^n を入力としてとり，鍵 k を出力する（k は共通鍵，秘密鍵とも呼ばれる）．通常，$\mathrm{Gen}(1^n)$ は $\{0,1\}^n$ から一様ランダムに n を選ぶ．
2. タグ生成アルゴリズム Mac は，鍵 k と文書 m

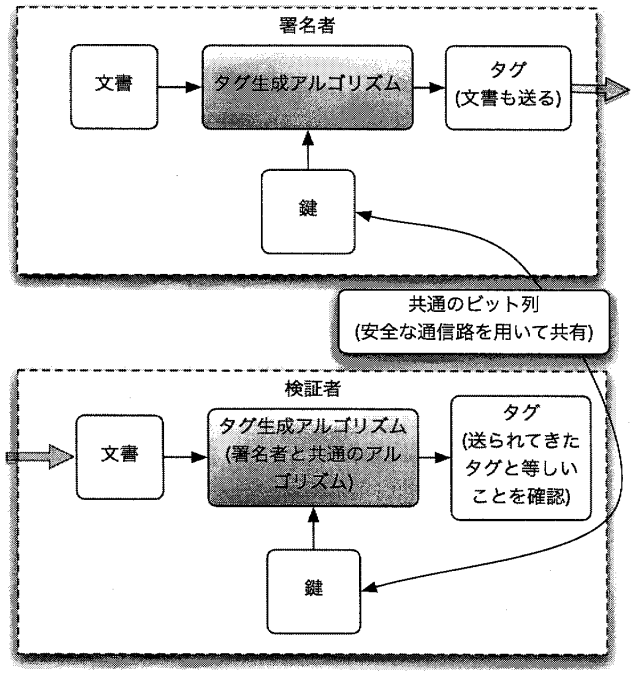

図4　メッセージ認証コード

を入力としてとり，タグ t を出力する．文書 m は任意のビット列 $\{0,1\}^*$ からとられる．通常，Mac は確率的ではなく決定的なアルゴリズムが用いられる．

3. 検証アルゴリズム Verify は，鍵 k，文書 m，タグ t を入力としてとり，1ビット b を出力する．$b=1$ で署名が有効であることを表し，$b=0$ で無効であることを表す．通常，Verify(k, m, t) のアルゴリズムは次のような単純なアルゴリズムを用いる．入力 k と m から Mac(k, m) を計算する．その結果を入力の t と比較し，一致したら $b=1$，一致しなかったら $b=0$ を出力する．これは決定的なアルゴリズムである．

メッセージ認証コードの通常の利用イメージは図4のとおりである．まず，署名者と検証者によって，一様ランダムに選んだ一定の長さのビット列を安全に共有する．次に，署名者がタグ生成アルゴリズム Mac を用いてタグを生成し，文書とともに受信者に送る．最後に，検証者は送られてきた k と m を入力として Mac を用いてタグを生成し，その結果を送られてきた t と比較することで文書とタグの組の正しさを確かめる．

Mac$_k$ は一方向性関数であることが要求される．

このメッセージ認証コードは基本的性質として，タグがつけられた文書が正しく検証できるような性質をもたなくてはならない．これは，すべての n，Gen (1^n) によって出力されるすべての k，すべての文書 $m \in \{0,1\}^*$ に対して，

$$\text{Verify}_k(m, \text{Mac}_k(m)) \to 1$$

が常に成り立つという条件で定式化できる．

この基本的性質に加え，メッセージ認証コードには，署名方式と同様に，タグの偽造不可能性をはじめとする安全性が要求される．安全性についての詳細は本稿のおわりに挙げる文献を参照してほしい．

メッセージ認証コードは秘密鍵暗号に関わる基本要素であるが，近年，このメッセージ認証コードは効率的な公開鍵暗号の構成にしばしば用いられ，公開鍵暗号にとっても重要な要素となった．

5. おわりに

本稿では，暗号理論でよく用いられる概念，重要な理論的土台である一方向性関数，特に重要な4つの基本技術を取り上げた．おわりに，暗号理論をさらに知るための参考文献について紹介する．

日本語で読むことのできるものとして，文献[5]は本特集の尾形氏も関わっており，学部授業の教科書にも用いることができる大変コンパクトにまとまったものである．文献[4]は本特集の神田氏も関わっており，利用者の観点から網羅的に分野をカバーするとともに暗号の標準化動向についても詳しく述べられている．文献[6]は暗号理論に関するそれぞれのトピックがバランスよく含まれている教科書である．

英語のものとして，文献[1][2]は特に理論的な定式化について細かい扱いがなされている．文献[3]は暗号理論の現代的な定式化についての記述が充実している．

参考文献

[1] Goldreich, O., *Foundations of Cryptography : Volume I-Basic Tools*, Cambridge University Press, 2001.

[2] Goldreich, O., *Foundations of Cryptography : Volume II-Basic Applications*, Cambridge University Press, 2004.

[3] Katz, J. and Lindell, Y., *Introduction to Modern Cryptography*, Chapman & Hall/CRC, 2007.

[4] NTT 情報流通プラットフォーム研究所，最新暗号技術，ASCII, 2006.

[5] 黒澤馨，尾形わかは，現代暗号の基礎数理，コロナ社，2004.

[6] 岡本龍明，山本博資，現代暗号，産業図書，1997.