第15章

セキュリティ

＜略＞

15.4　セキュリティツールとしての暗号

コンピュータ攻撃に対しては方法論から技術にわたって多くの防御法がある．システム設計者およびユーザに利用可能な最も適用範囲の広いツールは，暗号である．本節では，暗号の詳細と，コンピュータセキュリティにおける利用法について述べる．

隔離されているコンピュータでは，オペレーティングシステムは，コンピュータのすべての通信チャネルを制御しているので，すべてのプロセス間通信の送り手と受け手を確実に特定できる．コンピュータネットワークでは，状況は極めて異なる．ネットワークに接続されたコンピュータは「回線から」ビットを，それらのビットがどのマシンあるいはアプリケーションから送られたものかを特定する，直接的でしかも信頼できる方法がないまま，受信する．同様に，コンピュータはビットを，それらのビットが結局誰に受信されることになるのかを知る方法がないまま，ネットワーク上に送信する．

一般に，ネットワークアドレスは，メッセージの潜在的な送り手と受け手を推定するのに使われているのである．ネットワークパケットは，IPアドレスといった送り元（ソース）アドレスが付いてやってくる．そして，コンピュータがメッセージを送信するときには，送付先アドレスを指定することによって，意図した受け手を指定する．しかしながら，セュリティが重要なアプリケーションに対して，パケットの送り元アドレスあるいは送付先アドレスから，そのパケットの送り手あるいは受け手を確実に特定できるのは当然のことと決めてかかっているなら，自ら災難を招いていることになる．不正なコンピュータは偽造された送り元アドレスをかたってメッセージを送信することができるし，送付先アドレスで指定されたコンピュータ以外の多数のコンピュータがパケットを受信することもできる（ような仕組みになっている）．たとえば，送付先までの途中にあるルータも，すべてそのパケットを受信する．それならば，オペレーティングシステムは，要求に明記された送り元が信用できないとき，その要求を認めるかどうかをどのようにして決めればよいのか？ また，ネットワークを経由して送られる返答またはメッセージの受け手が誰か特定できないとき，要求またはデータに対する保護がなされるかどうかをどのように考えるか？

一般に，どのような規模のネットワークでも，パケットの送り元アドレスおよび送付先アドレスが，この意味で「信用できる」ように構築することは不可能である．したがって，唯一の代替案は，ネットワークを信頼する必要性を何とかして取り除くことである．これが暗号の仕事である．抽象的には，暗号（cryptography）は，メッセージの潜在的な送り手と受け手とを束縛するために使われる．現代の暗号は，ネットワーク内のコンピュータに選択的に配送され，メッセー

ジを処理するのに使われる，鍵（key）と呼ばれる秘密に基づいている．暗号により，メッセージの受け手は，そのメッセージがある鍵を保持しているあるコンピュータで作り出されたものであることを検証することができる．その鍵がそのメッセージの「送り元」なのである．同様に，送り手は，ある鍵を保持しているコンピュータだけがそのメッセージを復号できるようにメッセージを暗号化できるので，鍵が「送付先」になる．しかしながら，ネットワークアドレスとは違い，鍵は，その鍵を使って作り出されたメッセージやその他の周知の情報から計算的に推論不可能なように設計される．したがって，鍵はメッセージの送り手と受け手とを束縛する，より信頼できる手段なのである．暗号は，多かれ少なかれ複雑性と緻密性を持つ特殊な研究分野であることに注意しよう．ここでは，オペレーティングシステムに関係する暗号の最も重要な局面について考察する．

15.4.1　暗号化

暗号は広範にわたる通信セキュリティの問題を解決するので，現代の計算の様々な局面で用いられる．暗号はメッセージの可能な受け手を束縛する手段である．暗号化アルゴリズムは，メッセージの送り手にあらかじめ決まっている鍵を持っているコンピュータしかそのメッセージを読むことができないことを保証できる．メッセージの暗号化は，もちろん古来から用いられていて，暗号化アルゴリズムも数多くあり，シーザー以前に遡ることができる．本節では，重要な現代暗号の原理とアルゴリズムについて述べる．

安全でない通信路を使って安全に通信する二人のユーザを図15.7に示す．

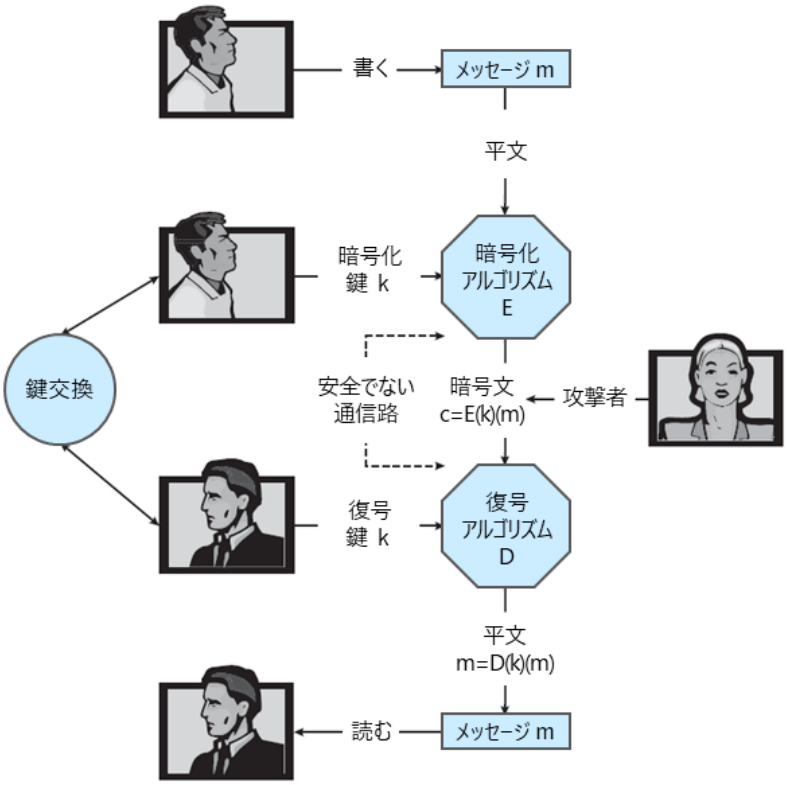

図15.7　安全でない手段を使った安全な通信

＜略＞

15.5 ユーザ認証

上述の認証の議論には，メッセージおよびセッションが含まれている．しかし，ユーザについてはどうであろう？ システムがユーザを認証できなければ，そのユーザから来たメッセージの認証は無意味である．したがって，オペレーティングシステムに対する主なセキュリティ問題は，**ユーザ認証**（user authentication）である．保護システムは現在実行中のプログラムおよびプロセスを同定する能力に依存する．この能力は，システムのユーザを同定する能力によっている．ユーザは，通常自分自身を同定する．ユーザの同定が真正かどうかはどのように判定すればよいか？ 一般に認証は三つの項目，すなわちユーザが持っている何か（キーまたはカード），ユーザが知っている何か（ユーザの識別子およびパスワード），およびユーザ自身の何か（指紋，網膜パターンあるいは署名）の，一つ以上に基づく．

15.5.1 パスワード

ユーザの身元を認証する最も一般的な方法は，**パスワード**（password）を用いることである．ユーザが自分自身をユーザIDあるいはアカウント名で同定するとき，パスワードを求められる．ユーザが与えたパスワードがシステムに蓄えられているパスワードと合致するなら，システムはアカウントがそのアカウントの所有者によってアクセスされていると考える．

パスワードは，もっと完全な保護方式がない限り，コンピュータシステム内のオブジェクトを保護するのに用いられることが多い．パスワードは，キーあるいはケーパビリティの特殊な場合と考えることができる．たとえば，パスワードを各々の資源（たとえばファイル）と関連付けることができる．その資源を使う要求をするときには，そのパスワードを与えなければならない．パスワードが正しければ，アクセスが許される．パスワードはアクセス権ごとに異なったものにすることができる．たとえば，読出し，追加，更新といったファイル操作の各々に対して，別々のパスワードを用いることができる．

実際には，ほとんどのシステムでは，一人のユーザがすべての権限を取得するのに一つのパスワードしか必要としない．理論的には，パスワードが多ければ多いほど安全であるが，セキュリティと利便性との標準的なトレードオフの結果，そのようなシステムは実装されない傾向にある．セキュリティによって何か不便になるのなら，往々にして，セキュリティは無視されるか，そうでなければ回避される．

15.5.2 パスワードの脆弱性

パスワードは理解しやすく使いやすいので，極めて一般的である．残念ながら，パスワードは，以下に述べるように，推測できたり，偶然わかってしまったり，嗅ぎ付けられたり，あるいは許可された使用者から許可されていない使用者へ不正に渡されたりすることがよくある．

パスワードを推測するための一般的な方法は二つある．一つは侵入者（人あるいはプログラム）がユーザを知るか，あるいはユーザに関する情報を得るかすることである．パスワードとして，明らかな情報（たとえば，飼っている猫とか配偶者の名前）を用いることがあまりにも多い．も

う一つの方法は，力ずくでやる方法である．つまり，列挙すること，すなわち文字，数字，システムによっては区切り記号の可能な組み合わせを，パスワードが見つかるまですべて試みるのである．短いパスワードは，特段この方法に対して脆弱である．たとえば，4 桁の数字のパスワードは 10,000 通りしかない．平均して 5,000 回の試行で正しいものに行き当たる．プログラムで一つのパスワードの試行が 1 ミリ秒でできるとすると，4 桁の数字のパスワードの推測には，約 5 秒しかかからない．大文字と小文字を区別し，数字や区切り記号を使うことができたりする，もっと長いパスワードが使えるシステムでは，パスワードを見つけるのに列挙する方法はそれほどうまくいかない．もちろん，ユーザは巨大なパスワード空間を利用すべきであり，たとえば小文字だけを用いるようなことをしてはならない．

推測されることに加えて，パスワードは見られたり電子的に監視されたりすることによって暴かれてしまうことがある．侵人者はユーザがログインするときに，ユーザの肩越しに見ることができるので（覗き見（shoulder surfing）という），キーボードをじっと見て容易にパスワードを知ることができる．あるいは，コンピュータが接続されているネットワークにアクセスし，ネットワークモニタを継ぎ目なくつなぐことができれば，ネットワーク上を流れるユーザ ID やパスワードを含むすべてのデータを見ることができる（傍受（sniffing）という）．この問題は，パスワードを含むデータストリームを暗号化することで解決できる．しかしながら，そのようなシステムであっても，パスワードは盗まれる．たとえば，パスワードを取っておくのにファイルを使うと，オフラインでの解析のためにコピーされてしまう．あるいは，アプリケーションに送られる前にキーストロークをすべて捕捉してしまうトロイの木馬プログラムが，システムにインストールされているような場合を考えてみるとよい．

パスワードを，人に読まれたり，なくしたりしてしまうようなところに書くようなことをすれば，露見はとりわけ重大な問題である．後述するが，覚えておくのが難しいパスワードや長いパスワードをユーザに強制するシステムもある．そうなると，ユーザはパスワードを書き留めたり，何度もそれを使うことになる．その結果，そのようなシステムは，ユーザが簡単なパスワードを選べるシステムよりも，セキュリティが悪くなる！

パスワードをどうするかとか，不法な状況をどうするかとかいった問題は，最終的には人間の本質に帰着する．ほとんどのコンピュータ施設では，ユーザがアカウントを共有することを許さない．この規則は，アカウンティング上の理由で実装されている場合もあるが，セキュリティを促進する理由による場合がほとんどである．たとえば，一つのユーザ ID を何人ものユーザで共有し，そのユーザ ID でセキュリティ違反が生じたとする．すると，そのとき誰がそのユーザ ID を使ったのかを知ることができないだけでなく，使ったユーザが許可されているユーザかどうかを知ることすらできない．一つのユーザ ID を一人のユーザに限定すると，どのユーザにもそのアカウントの使用について直接問いただすことができる．さらに，ユーザがアカウントについて何か変だと気づき，侵入を検出するかもしれない．友達を助けたり，アカウントを回避したりするためにアカウントの共有規則を破ることもよくあるが，このようなことをすると，許可されていないユーザ，おそらく有害なユーザに，システムにアクセスされてしまうことになりかねない．

パスワードはシステムで生成することも，ユーザに選択させることもできる．システムが生成

したパスワードは覚えにくく，その結果ユーザはどこかに書き留めることになるだろう．しかしながら，ユーザが選択したパスワードは，前述したように（ユーザの名前とか好きな車とかだったりして）推測しやすいことが多い．システムによっては，提案されたパスワードを受理する前に，それが推測しやすいか，つまりクラッキングしやすいかをチェックする．サイトによっては，管理者がときどきユーザのパスワードをチェックし，推測しやすかったときにはユーザに通知する．また，システムによっては，長く使われているパスワードを調べ，ユーザに一定期間ごと（たとえば 3 か月ごと）にパスワードを変更させるものもある．しかし，これらの方法はどちらも確実ではない．というのは，ユーザが二つのパスワードを交互に用いることが容易にできるからである．解決策は，システムによって実装されていることもあるが，各ユーザのパスワードの履歴を記録することである．たとえば，システムで，最後に使われたN 個のパスワードを記録しておき，これらの再使用を許さないようにするのである．

これらの単純なパスワード方式の変種を用いることもできる．たとえば，パスワードを極めて頻繁に変更することができる．極端には，パスワードをセッションごとに変えるのである．新しいパスワードを各セッションの終わりに（システムまたはユーザが）選び，そのパスワードを次のセッションで使うのである．この場合には，たとえパスワードが悪用されても，一度しか使うことはできない．正当なユーザが次のセッションで本来のパスワードを用いようとしたときには，セキュリティ侵犯を犯されたことを知ることになる．違反したセキュリティを回復するステップがとられることになる．

15.5.3　暗号化パスワード

＜略＞

15.5.4　使い捨てパスワード

＜略＞

15.5.5　バイオメトリクス認証（生体認証）

＜略＞