第 2 回 システム環境設定ゼミ

ゼミ担当者　：　吉井健吾，千野晋平，高畑泰祐
指導院生　　：　米澤基，市川親司
開催日　　：　2004 年 4 月 23 日

ゼミ内容: 本ゼミでは，TCP/IP プロトコルで通信する際に必要となる IP アドレスやプロキシサーバ，ルータの仕組みや DNS の仕組みといったようなネットワークの基礎知識，そして SSH を用いたセキュリティの向上への取り組みとして，ポートフォワーディングや暗号・認証の必要性について学ぶ．

1　IP アドレス

1.1　グローバル・アドレスとプライベート・アドレス

IP アドレスとは，ネットワークに接続した機器一つひとつを識別する番号である．今使われている IPv4 の IP アドレスは，32 ビットで構成されおり，8 ビットずつ区切って「172.20.11.2」のように 10 進数で表現されている．IP アドレスはそれで個々の機器を識別するため，他と重複しない固有なアドレスを割り当てる必要がある．しかし，近年，インターネットの急成長により，割り当てるアドレスが不足してしまう問題がある．そこで，違う LAN 内であれば，ある条件の下で割り当てるアドレスが重複してもよいというルールを設けることが考えられた．このルールに基づき付けられた LAN 内用のアドレスを**プライベート・アドレス**と呼び，従来の固有なアドレスを**グローバル・アドレス**と呼ぶ．その条件とは，プライベートとして社内で使うものは下記の範囲 (Table 1) に限定するというものである．

Table 1　各クラスのプライベートアドレス領域

クラス名	領域
クラスA	10.0.0.0〜10.255.255.255
クラスB	172.16.0.0〜172.31.255.255
クラスC	192.168.0.0〜192.168.255.255

1.2　サブネット番号とホスト番号を分けるサブネットマスク

プライベート IP アドレスは，サブネットワーク番号とホスト番号のビットを合計すると 32 ビットになるが，その内訳は固定されていない．32 ビット中の左何ビットがサブネットワーク番号を表し，何ビットがホスト番号を表すかといった両者のビット配分は，**サブネットマスク**という値で決めることになっている．そのため，32 ビットのプライベート IP アドレスだけでは，IP アドレスの内容を正しく表現できない．つまり，IP アドレスとネットマスクの二つがワンセットとなってはじめて，ネットワーク番号とホスト番号の値を正しく表現できるのである．

例えば，「172.20.11.2」という IP アドレスを「255.255.255.0」というサブネットマスク値を使って分割する．するとこの IP アドレスが示すのは，172.20.11.0 というサブネットワーク上にある，ホストアドレス 2 の端末ということが分かる (Fig. 1).

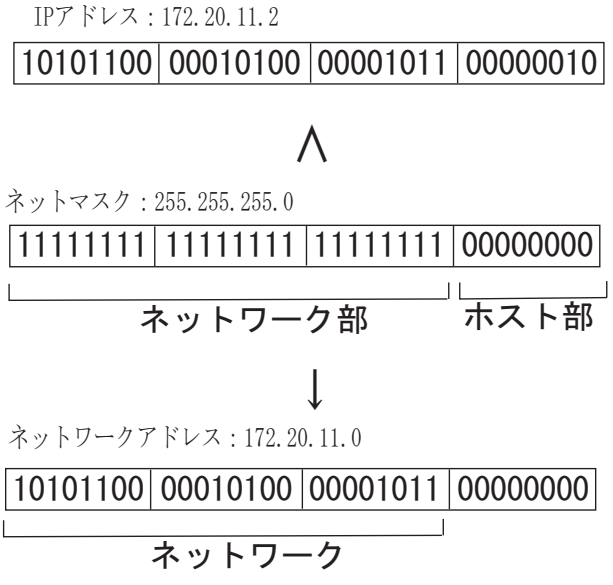

Fig. 1　サブネットマスク

ところで，IP ホストは正確には「0」と「255」は使用できないことになっている.「0」はそのネットワーク自身のことを表すために使用され，また「255」は**ブロードキャストアドレス**として使用される．ブロードキャストアドレスとは，ネットワーク上の全てのコンピュータに対して通信を行うために使用されるアドレスである．例えば「172.20.11.255」とすると,「172.20.11.0」のネットワーク全ての端末に通信が行われる．

1.3 ルータとアドレス変換

プライベートアドレスはあくまでも LAN 内用のものにすぎないため，これを利用する端末は直接インターネットに接続することができない．そこで、プライベートアドレスとグローバルアドレスを相互に変換する仕組みが必要になる．この機能を担うのが**ルータ**である．ルータのアドレス変換機能の仕組みは，パケット中継するときにパケットに記載されている IP アドレスとポート番号を書き換えるものである．例えば，プライベートアドレス「172.20.11.2」のマシンから，グローバルアドレス「192.0.2.79」の web サーバにアクセスするとき，ルータにてプライベートアドレス「172.20.11.2」からグローバルアドレス「95.3.8.31」に書き換える．ここで使うグローバルアドレスは，ルータに割り当てられたアドレスである．それと同時にポート番号もルータが未使用の番号を適当に選んで書き換える．そして，書き換える前のプライベートアドレスとポート番号，書き換えた後のグローバル・アドレスとポート番号をワンセットにしてルータ内部にある対応表に記録しておく．よって相手の web サーバに届くパケットは送り先アドレスにルータのグローバルアドレスとポート番号が記載されており，web サーバもルータのグローバルアドレスとポート宛てにパケットを送ることになる．

LAN 側		インターネット側	
プライベートアドレス	ポート番号	グローバルアドレス	ポート番号
172.20.11.2	1025	95.3.8.31	5436
172.20.11.3	1025	95.3.8.31	5437
172.20.11.4	1025	95.3.8.31	5438

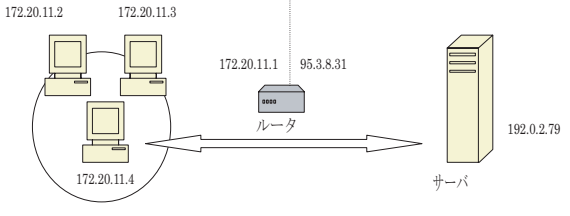

Fig. 2　ルータの仕組み

2 URL

URL(Uniform Resource Locator) とは，インターネット上に存在する情報資源 (文書や画像など) の場所を指し示す記述方式である．インターネットにおける情報の「住所」にあたる．ブラウザでは，この URL を「アドレス」として指定することで，インターネット上の目的の情報にアクセスできる．例えば URL は以下のようなものである．

```
http://www.doshisha.ac.jp/index.html
http://mikilab.doshisha.ac.jp/index.html
```

「http」は，Web サーバにアクセスするときのプロトコルで，他にも ftp，mailto，file 等がある．次に「://」から「/」までの間をドメイン名という（ホスト名と呼ばれることもある）．ドメイン名は「ホスト部＋ドメイン部」で構成されており，「www」や「mikilab」はホスト部,「doshisha.ac.jp」がドメイン部にあたる．ドメイン名は，インターネット上の IP アドレスは人間には覚えにくいため，わかりやすくつけた名前のことである．ドメイン名は各ドメインごとに意味づけをしてあり，右から順にわかりやすいようになっている．「www.doshisha.ac.jp」は日本の大学の同志社の Web サーバということになる．「index.html」にあたる部分をパス名という．パス名ではサーバ内での情報の場所を示す．よって「http://www.doshisha.ac.jp/index.html」は日本の大学の同志社の Web サーバにある，index.html というファイルにアクセスすることになる．サーバ内で階層的に情報が管理されている場合には,「/」を用いて階層ごとにパス名を指定する．

3 DNS サーバ

クライアントが Web サーバ等にアクセスするためには，Web サーバの IP アドレスを知る必要がある．その IP アドレスを教えることが DNS サーバの役割である．具体的には，クライアントが mikilab のサーバにアクセスしようとするとき,「mikilab.doshisha.ac.jp」の IP アドレスを，クライアントが設定した優先 DNS サーバ「172.20.11.16」に問い合わせると，DNS サーバは登録してある「mikilab.doshisha.ac.jp」の IP アドレスをクライアントに返すのである．しかし，必ずしもクライアントが設定した DNS サーバがクライアントが要求するサーバの IP アドレスを知っているとは限らない．そのときは，他の DNS サーバに問い合わせることになる．この時の動作を解説する．まず，クライアントがブラウザで「http://www.doshisha.ac.jp」にアクセスしようとして，優先 DNS サーバに問い合わせるとする．しかし，優先 DNS サーバには「www.doshisha.ac.jp」の IP アドレスが登録されていなかったため，他の DNS サーバに問い合わせることになる．DNS サーバというのは階層構造になっており，ドメインの右に行くほど上位の階層になる (Fig. 3)．つまり,「www.doshisha.ac.jp」では jp が一番上位の階層になるが，そのもう一つ上にルートドメインサーバがある．このように階層化したドメインの一つひとつが DNS サーバの担当範囲を表す単位となる．具体的には，下位のドメインを担当する DNS サーバを，すぐ上に位置する DNS サーバに登録する．そして，その上位の DNS サーバをさらに上位の DNS サーバに登録する，というように順に登録していく．つまり，www.doshisha.ac.jp というドメインを担当する DNS サーバを doshisha.ac.jp の DNS サーバに登録し，

doshisha.ac.jp の DNS サーバを ac.jp ドメインの DNS サーバに，といった順に登録していくのである．よってルートドメイン DNS サーバは jp ドメインの DNS サーバを登録しているのである．このルートドメインの DNS サーバを DNS サーバ全てに登録することによって，全ての DNS サーバをつなぐことができるのである．つまり，クライアントが設定した優先 DNS サーバは，jp ドメインの DNS サーバの IP アドレスをルートドメインに問い合わせ，次に ac.jp ドメインの DNS サーバを jp ドメインの DNS サーバに問い合わせる．また doshisha.ac.jp の DNS サーバの IP アドレスを ac.jp の DNS サーバに問い合わせ，www.doshisha.ac.jp のサーバの IP アドレスを，doshisha.ac.jp に問い合わせることで，クライアントに www.doshisha.ac.jp の IP アドレスを返せるようになるのである．

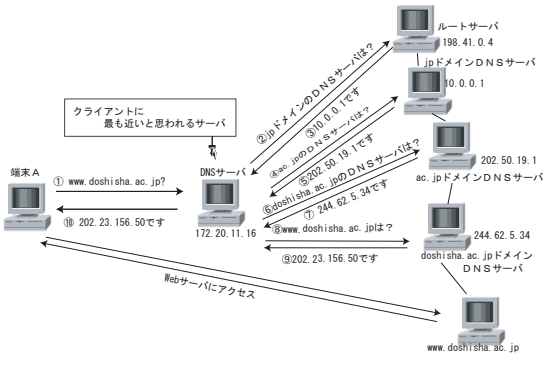

Fig. 3　DNS サーバの仕組み

4　プロキシサーバ

プロキシサーバは，プロキシ (代理) という言葉が示す通り，インターネット上の目的のコンテンツ (ファイルや Web ページ等) をユーザのコンピュータの代わりに取って来てくれる機能を提供するサーバである．プロキシサーバを利用すると，主に以下のような効果が期待できる．

4.1　キャッシング

プロキシサーバはキャッシュ機能を実装しており，過去にサーバから受信したコンテンツを一時的にプロキシサーバ自身のディスク内に蓄積しておくことができる．そのため，プロキシサーバがユーザからすでにキャッシュされているコンテンツのリクエストを受けたとき，わざわざ遠くにある本来のサーバに同じコンテンツを取りに行くことなく自分自身にキャッシュされているデータをユーザに送り返すので，ユーザはコンテンツを素早く取得することが可能となる．また，その場合プロキシサーバは本来のサーバにアクセスしなくて済むため，そのサーバに掛かる負荷を軽減することができる．このようにプロキシをキャッシュサーバとして利用することによって，遠くのサーバに何度もアクセスしなくて済むため，結果的にネットワークのトラフィック（回線使用量）を軽減できるのである．

4.2　フィルタリング

プロキシは Web ブラウザ等のクライアントと WWW サーバ等の外部のサーバ間に位置しているため，これらの間を通過する全てのデータをフィルタリングすることができる．つまり，プロキシはクライアントからサーバに送信されるデータ，あるいはサーバからクライアントに送信されるデータをモニタリングして，それらのデータを自由に改変したり，通信自体を妨害することができる．このフィルタリングは主に内部ネットワーク（LAN）の効率的運用を目的として行われる．

4.3　匿名性確保

プロキシを利用しないで Web ページを閲覧する場合，ユーザのコンピュータが Web サーバに直接アクセスするために，ユーザの様々な個人情報 (IP アドレスやホスト名など) が要求先の Web サーバと Web ページ管理者に伝わってしまう．しかし，プロキシを介して Web ページを閲覧する場合，プロキシがユーザの代わりに WWW サーバにアクセスして目的の Web ページを取って来てくれるため，ユーザが WWW サーバに直接アクセスすることなく目的のコンテンツを閲覧することができのである．

5　パーミッションについて

ファイルのアップロードやファイルの更新で最も注意を要するのがファイルに対するパーミッションである．ファイルに対するアクセス権限のことをファイルのパーミッションもしくはモードという．勝手に自分や他のユーザのファイルを読んだり書き換えたりできないようにすることができる．

UNIX コマンドで”ls -l”コマンドを入力すると，ファイルの所有者や作成日時など多くの情報が表示されるが，一番左に”-rwxr--r--”のように表示される部分がパーミッションである．”r”や”-”は，それぞれが 1 ビットに対応している．先頭のファイルの型をあらわすビットを除いた 9 ビットを 3 ビットずつ区切った各部分は，左から「所有者」「グループ」「その他のユーザ」を表している．その三つに対して,「読み込み」「書き込み」「実行」の 3 つのアクセス権限を表している．パーミッションは各部分ごとに，許可する権限の値を足し算して計算する．たとえば,「読み取り」と「実行」のみを許可する場合は，4+1 で 5 になり，すべてを許可する場合は，4+2+1 で 7 となる．つまり，”-rwxr--r--”という文字列は，”744”という数値に置き換えることができる．またディレクトリの場合，Fig. 5 のようになる．

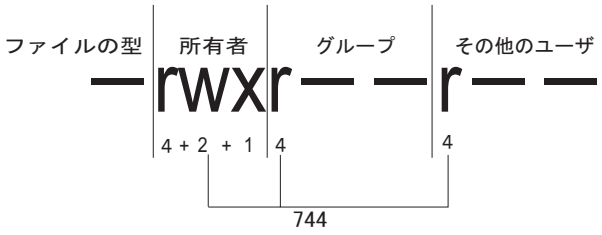

Fig. 4　パーミッションの表記

Fig. 5　ディレクトリのパーミッションの表記

Table 2　ファイルのアクセス権限

権限	表示	8 進数表記
読み取り権限	r（read）	4
書き込み権限	w(write)	2
実行権限	x(execute)	1

Table 3　ディレクトリのアクセス権限

r	ディレクトリ情報読み取り権限
w	ファイル作成権限
x	カレントディレクトリ変換権限

6　SSH を用いたセキュリティ向上のための取り組み

UNIX 系 OS では従来，ネットワークを介して別のマシンにログインしたり，コマンドを実行したり，ファイルを転送するのに，Telnet や r 系コマンド (rlogin, rsh, rcp) と呼ばれる方法を用いる．しかし，この方法ではネットワークを流れる情報が暗号化されないので，パスワードややり取りされる情報が盗まれてしまう危険性がある．

そこで，三木研究室では前述のことを行う際に，セキュリティ向上のために SSH(Secure SHell) という方法を用いている．

SSH を用いた通信ではパスワードだけではなく，やり取りされるデータも暗号化されるので，通信の途中でデータを盗聴されたり，改ざんされたりする心配がなくなる．また，ユーザ認証も通常のパスワードによる認証に加え，RSA 公開鍵暗号による認証も利用することができ，より通信の安全性を向上させることが可能である．

6.1　ポートフォワーディング

ポートフォワーディングとは，SSH の機能を用いて暗号化機能を持たないプロトコルでも通信路を暗号化させ，セキュリティを向上させるものである．

電子メールを送信するときのプロトコルである SMTP や受信するときのプロトコルである POP では，メール本文やパスワードの内容が暗号化されない．これでは，途中の通信路でデータを盗み見られる危険性がある．この様子を Fig. 6 に示す．

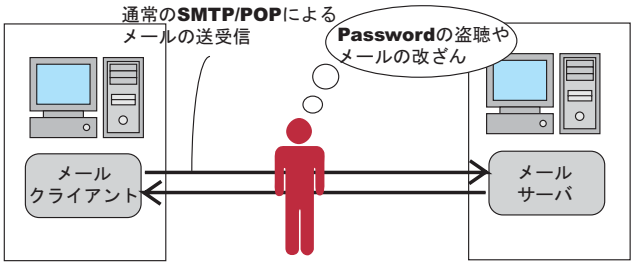

Fig. 6　通常の通信

そこで，ポートフォワーディングの登場となる．ポートフォワーディングとは，SSH に対応したプログラムを起動しておいて，ローカルホスト上にポートを開き，そこに送られてきたデータを SSH を用いてあらかじめ設定しておいたリモートホスト上の SSH サーバに送信し，SSH サーバがそのデータをさらにマシン内や別ホスト上で稼働しているサーバに送るというものである．このことを図で表すと，Fig. 7 のようになる．こうすることで，データの暗号化がサポートされていないプロトコルでも途中の通信路では SSH を用いたデータ転送が行われるために安全にデータを送ることができる．

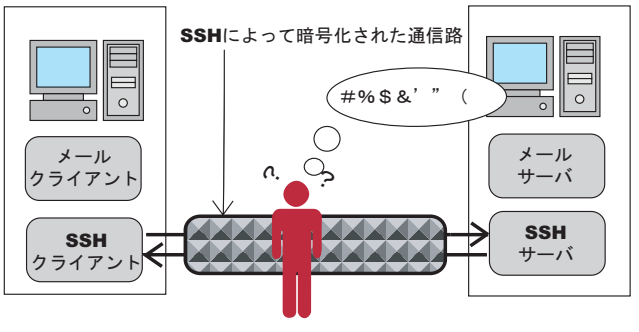

Fig. 7　ポートフォワーディングを用いた通信

ポートフォワーディングを行うためには,「自分のどのポートに送られたデータを」,「どの相手の」,「どのポートに」フォワーディングするかを設定する．たとえば,

L8025:mikilab.doshisha.ac.jp:25

のような設定をすることで，ローカルホスト上の 8025 に送られたデータは SSH を経由し，最終的に mikilab の 25 番ポートに送り届けられることになる．

6.2 SCP を用いたファイル転送

SCP とは，SSH を用いたファイル転送コマンドである．従来，UNIX にはリモートホストにファイルを転送する方法として rcp というコマンドが使われていた．しかし，rcp はデータが暗号化されないために途中の通信路でデータを盗聴される危険性がある．そこで，rcp に変わるものとして，SCP が用いられるようになった．SCP を用いることでデータが暗号化されるため，途中で盗聴される心配がなくなる．以下に，その利用方法について説明する．

scp [オプション][転送したいファイル名][転送先のパス]

ローカルホストからリモートホストへファイルを転送するためには，転送したいファイル名にローカルホストにあるファイルを指定し，転送先のパスにリモートホストのパスを指定する．リモートホストを転送先のパスとして指定する場合，以下のように記述する．

[ユーザ名]@[ホスト名]:[パス]

例えば，ユーザ名に chino，ホスト名 mikilab.doshisha.ac.jp でディレクトリを”public”と指定する場合，”chino@mikilab.doshisha.ac.jp:public”となる．また，scp コマンドに”-r”オプションをつけることによって，ディレクトリを転送することが可能である．

scp -r [ユーザ名]@[ホスト名]:[ディレクトリ] chino-dir

上記のように入力することで”chino-dir”というディレクトリに指定したディレクトリの情報がコピーされる．

7 SSH で用いられる暗号技術

暗号化とは，ネットワークを通じて文字や画像などのデータをやり取りする際，通信途中で第三者に盗み見られたり改ざんされたりされないよう，決まった規則に従ってデータを変換することである．

SSH の主な機能である「認証」および「通信」には暗号技術が用いられており，暗号化および復号化を行っている．SSH での暗号化，復号化には暗号表に当たる「鍵」を用いるが，次の 2 通りの方式が使われている．

7.1 共通鍵暗号方式

暗号化と復号化に同じ「鍵」を用いる暗号方式．暗号文の送信者と受信者で同じ「鍵」を共有する必要があるため，暗号文を送受信する前にあらかじめ安全な経路を使って「鍵」を共有する必要がある．

- 利点
  同じ鍵を使うため高速処理が可能で，大きなデータの暗号化にも適している．

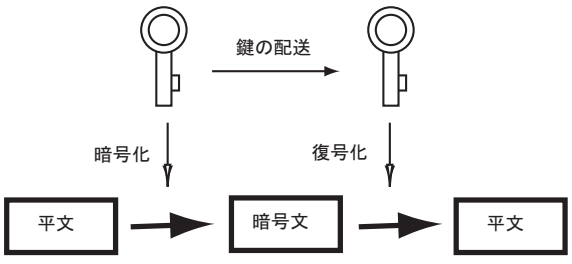

Fig. 8　共通鍵暗号方式

- 欠点
  事前に相手に対して秘密裏に鍵を手渡しておく必要がある．通信相手に対してそれぞれ違う「鍵」が必要なため，数が多くなると管理が大変である．

SSH が使用している主な共通鍵暗号の方式を紹介する．

- DES (Data Encryption Standard)
  1970 年代に IBM が開発し，1997 年に米国連邦政府の暗号標準に定められ，広く利用されている．平文に対してデータの置換操作と，位置を入れ替える転置操作といった比較的に単純な処理を組み合わせて，複雑な変換を施す．56bit 長の鍵を利用し，データを 64bit のブロック毎に処理する．

- 3DES
  異なった鍵で DES 暗号化を 3 段階行うことにより，DES を強化したもの．

- IDEA (International Data Encryption Algorithm)
  1991 年にスイスで開発された．128bit 長の鍵を用いて，データを 64bit のブロック毎に処理する．

- Blowfish
  Bruce Schneier によって開発された高速の暗号．32bit から 448bit 長の鍵をサポートしている．SSH1 では 128bit 長の鍵を使用し，データを 64bit のブロック毎に処理する．

7.2 公開鍵暗号方式

共通鍵暗号方式を利用する場合の，鍵の管理の問題を解決するために開発された．この方式では 2 つの対になった「鍵」を用いる．通常，一方を秘密鍵と呼び，もう一方を公開鍵と呼ぶ．公開鍵を使って暗号化したものは，秘密鍵でしか復号化できず，逆に秘密鍵を使って暗号化したものは，公開鍵でしが復号化できない．

このシステムの概要は Fig. 9 の通りである．

- 利点
  安全に保持しなければならないのは秘密鍵だけであり，送信する必要もないので管理しやすい．

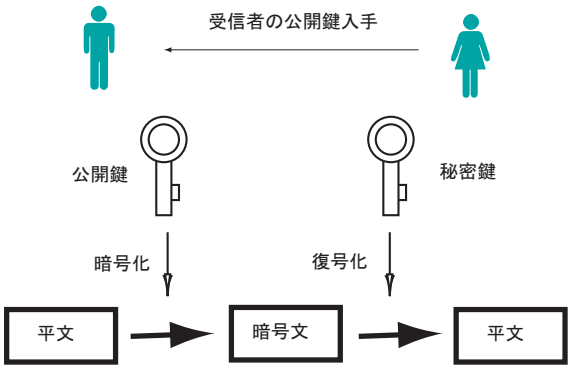

Fig. 9　公開鍵暗号方式

- 欠点
  暗号化，復号化に時間がかかる．暗号化された文が長くなってしまうため，大きなデータには向いていない．

公開鍵暗号方式は，通信の暗号化以外に電子署名の用途にも利用される．秘密鍵でメッセージを暗号化することができるのは本人だけであり，暗号化されたメッセージを受け取ったものは公開鍵で復号化することによって，発信者の本人認証を行なうことができる．

SSH2 では公開鍵暗号システムとして RSA と DSA を使っている．

- RSA
  1977 年に R.L.Rivest, A.Shamir, L.Adelman の 3 人によって考案された．名称は 3 人の頭文字を取って付けられている．大きな整数を素因数分解することは非常に困難であるという性質を利用している．暗号化および電子署名に利用される．

- DSA(Digital Signature Algorithm)
  1994 年にアメリカ政府の標準に定められた，DSS(Digital Signature Standard) の中で定められており，電子署名のみに利用できる．SSH2 では標準となっている．

公開鍵暗号システムは,「鍵」の管理については強力である一方，処理が遅く大きなデータを暗号化するのには向いていない．そのため，共通鍵暗号方式の「鍵」の転送に利用される．

8　SSH での認証

通常の認証方式としては，ユーザー ID とパスワードを入力して認証を受けるのが一般的である．しかし，リモートアクセスにおいては，パスワードによる認証は非常に弱く，IP アドレスの偽装やパスワードの盗聴など，不正アクセスに弱い．

SSH はこのような不正アクセスを防ぐために，暗号方式により以下の 3 つの仕組みにより通信のセキュリティを確保している．

1. ホスト認証
2. 通信内容の暗号化
3. ユーザー認証

SSH プロトコルには SSH1 と SSH2 の 2 種類があるが，これらの間には互換性がない．

SSH1 では，公開鍵暗号化方式に RSA を使い，暗号通信には 3DES と Blowfish を用いている．他にも IDEA を含んでいるものもあったが，特許問題などにより使用されていないのが現状である． また SSH1 は，データ改ざんが行われていないか検査するために，単純な CRC を用いている．

一方 SSH2 は，RSA に関する特許問題を避けることを意図して作られた (現在すでに特許は切れているため，この問題はもうなくなっている)．また，SSH1 の問題であった CRC によるデータ改ざん検査も，強力な HMAC アルゴリズムによって改善され，公開鍵暗号化方式に DSA も使うことによってすべての特許問題もクリアしている．

この節では SSH による認証がどのように行なわれるかを説明する．

8.1　ホスト認証

SSH を用いた通信を行う際，クライアントがサーバーへ接続要求を出し，お互いのバージョン番号などを交換した後，クライアントが共通鍵をランダムに生成し，サーバーに対して送信する．この共通鍵は，クライアントとサーバーとの間の通信を暗号化するために使われる．共通鍵をサーバーに送信する際に，盗聴される心配があるため，暗号化して送ることになるが，この時のプロトコルを Fig. 10 に示す．

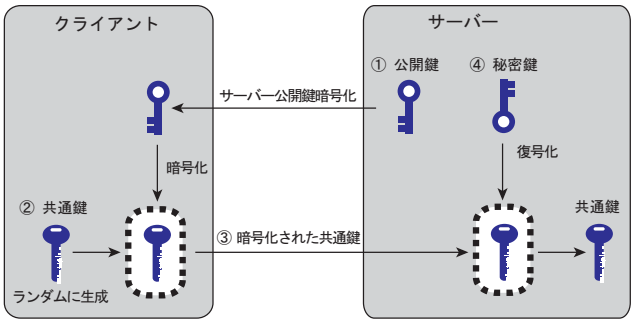

Fig. 10　ホスト認証

1. サーバーは自身の公開鍵　をクライアントに対して送信．

2. クライアントは接続相手サーバーごとの公開鍵を保存しているので，もしサーバーの公開鍵　が，以前同じサーバーから送られてきた公開鍵と一致しなかった場合，不正侵入の可能性がある，という警告が出る．

3. 公開鍵　が正しいことが確認できると，クライアントはサーバーの公開鍵　を使って共通鍵　を暗号化．

4. 暗号化された共通鍵　をサーバーへ送信．

5. サーバーは自身の秘密鍵　を使って，暗号化された共通鍵　を復号化

以上で，クライアントとサーバーは同じ共通鍵を，安全な方法を用いて持つことができる．この後のクライアントとサーバー間の通信は，この共通鍵によって暗号化される．この暗号通信が行えるということは，サーバーがクライアントと同じ共通鍵を持っていることの証明であり，サーバーが正しい秘密鍵　を持っていることの証明である．つまり，ホスト認証が成立したことになる．

8.2　通信内容の暗号化

SSH はホスト認証以後の通信は，上で述べたように共通鍵暗号方式によって守られている．これは，公開鍵による暗号化処理が共通鍵によるものより時間がかかるためである．SSH ではこのようにお互いの認証には公開鍵暗号を，そして実際の通信には共通鍵暗号を用いることで，安全かつ処理の速い通信を行うように設計されている．

8.3　ユーザー認証

サーバーによるユーザー認証は，次の４つの方法が提供されている．

- rhosts による認証
- rhosts と公開鍵暗号方式を組み合わせた方式による認証
- 公開鍵暗号方式による認証
- パスワード認証

実際に使用される認証方式は，SSH サーバーの運用方針と，ユーザー側の設定によって決まる．SSH サーバーは，受け入れを許す設定になっている認証を順番に試みる．SSH1 では，rhosts による認証，rhosts 公開鍵認証，公開鍵認証，パスワード認証の順に試みる．SSH2 では，公開鍵認証，パスワード認証の順に試みる．通常、rhosts による認証，rhosts 公開鍵認証は安全上の問題からサーバーによって禁止されており、ここでは，SSH2 で用いられるパスワード認証と公開鍵認証について解説する．

8.3.1　パスワード認証

通常のパスワード認証を暗号化したもの．クライアントはユーザーが入力したパスワードをサーバーに送り、サーバーは通常のパスワード認証を行う．しかしパスワードは共通鍵によって暗号化されているので，通常通信より格段に安全である．

8.3.2　公開鍵暗号方式よる認証

接続に先立って，サーバー上にログインを許可するクライアントの公開鍵を登録しておく．ユーザー認証とは，登録された公開鍵に対応する秘密鍵をクライアントが持っているか確認することである．もちろん秘密鍵自体をサーバーに送ってしまうと,「秘密」でなくなってしまうので，秘密鍵を送らずに秘密鍵を持っていることをサーバーに対して証明しなければならない．Fig. 11 にクライアント認証のプロトコルを示す．

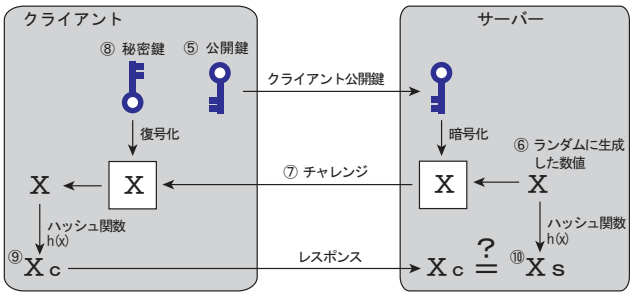

Fig. 11　ユーザー認証

1. クライアントは，サーバーにログインするときのユーザー ID と，自身の公開鍵　をサーバーに対して送信．

2. サーバーは，ログインを要求されたユーザーのログインが許可されているか調べる．

3. クライアントの公開鍵　が登録されていれば，サーバーはランダムな数値 X　を生成し，公開鍵　で暗号化．

4. 暗号化された X をクライアントに送信（これをチャレンジ　と呼ぶ）．つまり，クライアントが本当に公開鍵　に対応する秘密鍵を持っているのか試すための「チャレンジ」である．

5. クライアントは，チャレンジ　を秘密鍵　と passphrase で復号化し，元の X　を得る．

6. X をハッシュ関数 MD5 で変換した値 Xc　をレスポンスとしてサーバーへ送信．

7. サーバーはクライアントと同様に，X　をハッシュ関数 MD5 で変換した値 Xs　を計算し，レスポンスと比較．

これで一致していれば，クライアントが正しい秘密鍵を持っており、かつ passphrase を知っている本人の証明にもなる．つまり，クライアント認証が成立したことになる．

補足資料

- アカウント

  コンピュータやネットワーク上の資源を利用できる権利のこと，または利用する際に必要な ID のこと．

- TCP/IP【Transmission Control Protocol/Internet Protocol】

  インターネットやイントラネットで標準的に使われるプロトコル．米国防総省が，核攻撃で部分的に破壊されても全体が停止することのないコンピュータネットワークを開発する過程で生まれた．UNIX に標準で実装されたため急速に普及し，現在世界で最も普及している．OSI 参照モデルでは IP が第 3 層 (ネットワーク層)，TCP が第 4 層 (トランスポート層) にあたり，HTTP や FTP などの基盤となるプロトコルである．

- プロトコル

  複数のデバイスやコンピュータシステムが互いに通信するための規約．たとえばコンピュータシステムに SCSI デバイスを接続すると，両者は SCSI インターフェイスで定義された手順に従ってデバイスの初期化やデータ転送などを行なう．この場合の手順の取り決めをプロトコルと呼ぶ．また 2 つのコンピュータをネットワークで接続するとき，両者が通信するために使用する手順もプロトコルである．

- IP アドレス【Internet Protocol Address】

  インターネットやイントラネットなどの IP ネットワークに接続されたコンピュータ 1 台 1 台に割り振られた識別番号．現在広く普及している IPv4 では，8 ビットずつ 4 つに区切られた 32 ビットの数値が使われており,「210.145.108.18」などのように，0 から 255 までの 10 進数の数字を 4 つ並べて表現する．インターネット上ではこの数値に重複があってはならないため，割り当てなどの管理は各国の NIC が行っている．単なる数値の羅列である IP アドレスはこのままでは人間にとっては覚えにくいため，コンピュータに名前 (ドメイン名) がつけられている場合もあり，DNS というシステムによって IP アドレスとの相互変換が可能となっている．現在の IPv4 では約 42 億台までしかインターネットに接続することができず，アドレスが足りなくなることが懸念されており，IPv4 に代わる次世代の IPv6 の標準化が進行している．IPv6 では 128 ビットのアドレスが使われるため，当分アドレスが足りなくなる心配はない．

- サブネットマスク

  インターネットのような巨大な TCP/IP ネットワークは，複数の小さなネットワーク (サブネット) に分割されて管理されるが，ネットワーク内の住所にあたる IP アドレスのうち，何ビットをネットワークを識別するためのネットワークアドレスに使用するかを定義する 32 ビットの数値．

- デフォルトゲートウェイ (default gateway)

  所属するネットワークの外のコンピュータへアクセスする際に使用する「出入り口」の代表となるコンピュータやルータなどの機器．アクセス先の IP アドレスについて特定のゲートウェイを指定していない場合に，デフォルトゲートウェイに指定されているホストにデータが送信される．設定元のコンピュータからデフォルトゲートウェイまでは直接アクセスできなければならない．

- DNS 【Domain Name System】

  インターネット上のホスト名と IP アドレスを対応させるシステム．全世界の DNS サーバが協調して動作する分散型データベースである．IP アドレスをもとにホスト名を求めたり，その逆を求めたりすることができる．各 DNS サーバは自分の管理するドメインについての情報を持っており，世界で約 10 台運用されているルートサーバにドメイン名と自分のアドレスを登録しておく．

- DNS サフィックス

  FQDN(完全修飾ドメイン名) ではない部分的なホスト名などを指定した場合に補われるドメイン名のことで，OS のネットワーク設定の項目の一つである．

- FQDN

  インターネットやイントラネットなどの TCP/IP ネットワーク上で，ドメイン名・サブドメイン名・ホスト名を省略せずにすべて指定した記述形式のこと．例えば,「www.e-words.ne.jp」は FQDN だが,「e-words.ne.jp」はホスト名が省略されているので FQDN ではない．

- NetBIOS

  1984 年に IBM 社によって開発された通信インターフェイス．MS-DOS 環境から通信を行なうために必要なプログラムが BIOS という基本システムの形式で提供されたことから，MS-DOS の世界では急速に普及し，パソコン LAN の基本モデルになっている．

- プロキシ (proxy)

企業などの内部ネットワークとインターネットの境にあって，直接インターネットに接続できない内部ネットワークのコンピュータに代わって,「代理」としてインターネットとの接続を行なうコンピュータのこと．また，そのための機能を実現するソフトウェア.

- プロキシサーバ

  社内ネットワークとインターネットとの接続地点で，社内クライアントからのリクエストを代行して，両者の通信を中継するアプリケーション．Proxy とは「代理」の意味で，通常はファイアウォール上で稼働させる．インターネットから社内ネットワークへの通信は遮断する一方，社内ネットワークからインターネットに関しては，ユーザーやアプリケーションを指定してアクセスの制御を行なう.

- LAN

  同一フロア，同一のビルないしは近隣のビル内などにあるコンピュータ同士を，Ethernet などの比較的高速なデータ転送能力を持つ方法で接続したネットワーク.

- ポート (port)

  インターネット上の通信において，複数の相手と同時に接続を行なうために IP アドレスの下に設けられたサブ (補助) アドレス．TCP/IP で通信を行なうコンピュータはネットワーク内での住所にあたる IP アドレスを持っているが，複数のコンピュータと同時に通信するために，補助アドレスとして複数のポートを持っている．ポートの指定には 0 から 65535 までの数字が使われるため,「ポート番号」とも呼ばれる．IP アドレスとポートを組み合わせたネットワークアドレスを「ソケット」と呼び，実際にはデータの送受信はソケット単位で行われる.

- ゲートウェイ (gateway)

  ネットワーク上で，媒体やプロトコルが異なるデータを相互に変換して通信を可能にする機器．OSI 参照モデルの全階層を認識し，通信媒体や伝送方式の違いを吸収して異機種間の接続を可能とする.

- ブロードキャスト (broadcast)

  ネットワーク内で，不特定多数の相手に向かってデータを送信すること．ネットワーク全体を意味する特殊なアドレスを指定することによって行なう.

- ドメイン

  インターネット上に存在するコンピュータやネットワークにつけられる識別子．インターネット上の住所のようなもの.

- DHCP

  Dynamic Host Configuration Protocol の略．LAN 上のコンピュータに動的に IP アドレスを割り当てる方法．コンピュータがネットワークにログインすると，DHCP サーバーが，あらかじめ用意された IP アドレスの 1 つをそのコンピュータに割り当てる.

- アドミニストレータ (administrator)

  コンピュータやネットワークの管理者のこと．管理下のマシンなどを設定し，良好な環境を維持するのがアドミニストレータの仕事である.

- ドライバ

  ＯＳやアプリケーションに新たな機能を追加したり，機能を拡張する際に，その橋渡しをするソフト.

- デバイスドライバ (device driver)

  周辺機器を動作させるためのソフトウェア．OS が周辺機器を制御するための橋渡しを行なう．単に「ドライバ」と呼ばれることもある.

- Telnet (テルネット)

  インターネットやイントラネットなどの TCP/IP ネットワークにおいて，ネットワークにつながれたコンピュータを遠隔操作するための標準方式．また，そのために使用されるプロトコル．Telnet サーバを立ち上げてあるコンピュータにネットワークにつながれたほかのコンピュータから Telnet クライアントを使ってログオンし，そのコンピュータの目の前にいるのと同じように操作することができる.

- ssh【Secure SHell】

  主に UNIX コンピュータで利用される，ネットワークを介して別のコンピュータにログインしたり，遠隔地のマシンでコマンドを実行したり，他のマシンへファイルを移動したりするためプログラム．ネットワーク上を流れるデータは暗号化されるため，インターネット経由でも一連の操作を安全に行なうことができる.

- モジュール

  「module」は「規格化された構成単位」という意味．コンピュータ分野では，一般に論理的に分離可能なハードウェア／ソフトウェアの部品を指す.

- ディレクトリ

  ディスクでファイル管理の情報を記述した部分．ファイルサイズや変更日付などの細かな情報が書き込まれる．また，階層構造のファイル管理方式では，1つ

の階層をディレクトリと呼ぶ．Macintosh ではフォルダに相当し，ファイルを分類するのに使う．

- ホスト

  パソコン通信で中心となるコンピュータのこと．メニュー表示などの処理や，フォーラム，電子メールなど各種データの記録はホストコンピュータが行なっている．利用者はホストコンピュータにアクセスして，パソコン通信で提供されている各種サービスを利用する．

- サーバ

  コンピュータネットワークにおいて，クライアントコンピュータに対し，自身の持っている機能やデータを提供するコンピュータのこと．インターネットにおける WWW サーバなどが該当する．また，クライアントソフトウェアに対し，自身の持っている機能やデータを提供するソフトウェアのこと．

- POP

  メールサーバ上のメールを読み出すときに使う手順．インターネットやイントラネット上で，電子メールを保存しているサーバからメールを受信するためのプロトコル．

- SMTP 【Simple Mail Transfer Protocol】

  インターネットやイントラネットで電子メールを送信するためのプロトコル．サーバ間でメールのやり取りをしたり，クライアントがサーバにメールを送信する際に用いられる．

- APOP

  電子メールの送受信に使われるパスワードを暗号化する認証方法．普段メールの送受信に使われる POP はパスワードを平文で (そのままで) 送るので，盗聴される危険性がある．APOP ではパスワードを暗号化して送受信するので安全性が向上している．

- クライアント

  ネットワーク上のサービスを提供する役割を持つサーバ（提供者）に対して，ネットワークに接続してサービスを利用する側のコンピュータをクライアント（依頼者）という．

- FTP

  ネットワーク上のクライアントとホストコンピュータとの間で，ファイルの転送を行なうためのプロトコル（またはそれを実装したコマンド）．UNIX では，この FTP プロトコルを実装した ftp コマンドが標準で提供される．

- RSA

  Ronald Rivest 氏，Adi Shamir 氏，Leonard Adleman 氏の 3 人が 1978 年に開発した公開鍵暗号方式の一つ．開発者の名前をとって名付けられた．公開鍵暗号の標準として広く普及している．

- 公開鍵 (public key)

  公開鍵暗号方式で使用される一対の鍵の組のうち，一般に公開されるほうの鍵．

- 秘密鍵 (secret key)

  公開鍵暗号方式で使用される一対の鍵の組のうち，一般に公開されない鍵．