# 「暗号化通信におけるリスク」

# ~ SSH に潜む落とし穴 ~"暗号化すれば安全ですか?"

<付録.1> SSH の公開鍵認証について

2015年09月

Copyright 2015 FUJITSU SOCIAL SCIENCE LABORATORY LIMITED & NTT DATA FINANCIAL CORE CORPORATION & dit Co., Ltd

## <付録.1> SSH の公開鍵認証について

#### 1. SSH の公開鍵認証のシーケンス

SSHの公開鍵認証のシーケンスを説明する。



※ 公開鍵で暗号化された認証データを、復号できるのは秘密鍵となる。

※ ハッシュ値が一致すれば正規のユーザとしてアクセスが許可される。

<付録.1> SSHの公開鍵認証について

#### 2. SSH の公開鍵認証の設定方法

本章では、SSH の公開鍵認証を安全に利用していただくための設定方法を記載する。

(1) 公開鍵と秘密鍵の生成

公開鍵ペア(秘密鍵と公開鍵)は、サーバ側で生成することが可能であるが、ここでは、SSH クライアントである Tera Term と Tectia Client を使用した生成方法を記載する。

No	ソフトウェア	作成できる鍵の形式	
1	TeraTerm	OpenSSH 形式	
2	Tectia Client	SECSH形式(ssh.com)形式	

■TeraTerm による公開鍵ペア(秘密鍵と公開鍵)の生成

- ① TeraTerm を起動する。
- ② [新しい接続]ダイアログが表示されるので、「キャンセル」ボタンを押す。

Tera Term: 新しい接	続
● TCP/IP	ホスト(T): myhost.example.com ・ ジヒストリ(O) サービス: O Telnet TCPボート#(P): 22 O SSH SSH/バージョン(V): SSH2 ・ O その他 ブロトコル(C): UNSPEC ・
● シリアル(E)	ポート(R):
	OK キャンセル ヘルブ(H)

③ Tera Term の[設定]→[SSH 鍵生成]をクリックする。



「暗号化通信	におけるリスク」~ SSHに潜む落とし穴~	"暗号化すれば安全ですか?"
<付録.1>	SSHの公開鍵認証について	

④ [鍵の種類]で作成する[鍵の暗号方式]を選択し、

[生成]ボタンをクリックする。

[鍵の種類]:RSA

[ビット数(B)]:2048

建の種類 のRSA1 ● RSA ● DSA	لاتيا ا	ト娄(B):	生成(G
C ECDSA-256 C ECDSA	-384	2048	閉じる(0
ECDSA-521	9		
鍵のバスフレーズ:			
鍵のバスフレーズ: バスフレーズの確認:			
鍵のバスフレーズ: パスフレーズの確認:			
鍵のパスフレーズ: パスフレーズの確認: コメント(0):			

⑤ [鍵のパスフレーズ]と[パスフレーズの確認]と[コメント]を入力する。

[鍵のパスフレーズ]:[任意のパスフレーズ]

[パスフレーズの確認]:[任意のパスフレーズ]

[コメント]:[任意のコメント]

- ※ 鍵のパスフレーズを[無し]にした場合、秘密鍵があるクライアントから、 パスフレーズ不要で、サーバへログインできるようになる。このため、秘 密鍵の管理は、十分に注意して運用する必要がある。
- ⑥ [公開鍵の保存]ボタンをクリックし、鍵(id\_rsa.pub)に任意の名前をつけ、 任意の場所に保存する。
- ⑦ [秘密鍵の保存]ボタンをクリックし、鍵(id\_rsa)に任意の名前をつけ、 任意の場所に保存する。

※鍵のパスフレーズを[無し]にした場合「空のパスフレーズを使用します か?」と警告画面がでるので、「はい」ボタンをクリックする。

警告	the COMP. C.	X
4	空のパスフレーズを使用	用しますか?
	(±L)(Y)	いいえ(N)

「暗号化通信におけるリスク」~ SSHに潜む落とし穴~ "暗号化すれば安全ですか?" <付録.1> SSHの公開鍵認証について

⑧ [閉じる]ボタンをクリックし、[鍵生成]ウィンドウを閉じる。

鍵の種類 ● RSA1 ● RSA ● DSA ● ECDSA-256 ● ECDSA-384 ● ECDSA-521 ● ED25519	ビット数(B): 生成(G) 2048 閉じる(C
鍵を生成しました。	
鍵のパスフレーズ: ●●●●	H
パスフレーズの確認: ●●●●	v
コメント(0): マユーザ:	名>@<ホスト名>
bcrypt KDF形式(K)	ラウンド数(N): 16

⑨ [ファイル]→[終了]をクリックし、[TeraTerm] ウィンドウを閉じる。

イル(F) 編集(E) 設定(S) ニ	レトロール(0)	ウィンドウ(W)	漢字コード(K)	ヘルプ(H)	
新しい弦続(N)	Alt+N				
セッションの複製(U)	Alt+D				
Cygwin接続(G)	Alt+G				
ログ(L)					
ログにコメントを付加(0)					
ログを表示(V)					
ログダイアログを表示(W)					
ファイル送信(S)					
転送(T)	×.				
SSH SCP					
ディレクトリを変更(C)					
ログを再生(R)					
TTY Record					
TTY Replay					
印刷(P)	Alt+P				
	Albert				
終了(X)	Alt+Q				

■Tectia Client による公開鍵ペア(秘密鍵と公開鍵)の作成

- ① Tectia Clientを起動する。
- ② [Edit]→[Tectia Connections]をクリックする。





③ [User Authentication]  $\rightarrow$  [Keys and Certificates]  $\hat{c}$   $\rho$   $\gamma$   $\rho$   $\tau$   $\delta_{\circ}$ .

Tectia Connections Configuration				
-General				
- Default Connection	Keys and Certificates			
- Proxy Rules	Manage key pairs	and certificates used in us	r authentication.	
Clients	Default keys	C:¥Users¥Administrator	AppData¥Roaming¥SSH¥UserKeys	
G-User Authentication	Default certificate	s C:¥Users¥Administrator	AppData¥Roaming¥SSH¥UserCertificates	
<ul> <li>Keys and Certificates</li> <li>Key Providero</li> </ul>	Additional Direc	tories and Files		
-Server Authentication	Type C	ontent		
-Host Keys -CA Certificates				
LDAP Servers				
-Automatic Tunnels				
	-		Add directory Add file Delete	
			Hud directory Hud me Delete	
	Key and Certific	ate List		
	File name	Status Type	Comment	
	Change passph	rase Lipipad	New key Delete	
< · · · · · · · · · · · · · · · · · · ·	Condingo passipin			
200210				
HIJSI			OK Apply Cancel Help	

④ [Key and Certificate List] → [New Key]ボタンをクリックする。

- Default Connection	Keys at	nd Certificates
-Proxy Rules	Here have	
- Logging - Clients	manage key pairs	and certificates used in user authentication.
-Connection Profiles	Default keys	U#Users#Administrator#AppData#Roaming#35H#UserKeys
User Authentication	Default certificate	es C#Users#Administrator#AppData#Roaming#SSH#UserCertificates
<ul> <li>Keys and Certificates</li> <li>Key Provident</li> </ul>	Additional Direc	ctories and Files
-Server Authentication	Type	Contant
- Host Neys CA Certificates DAP Servers ORL Prefetch Automatic Tunnels		
	Key and Certific	Add directory_ Add file_ Delete
	File hame	status type Comment
() »	Change passpl	rrase., Upload

Copyright 2015 FUJITSU SOCIAL SCIENCE LABORATORY LIMITED & NTT DATA FINANCIAL CORE CORPORATION & dit Co., Ltd

「暗号化通信における	るリスク」~ SSHに潜む落とし	し穴~ "暗号化すれば安全ですか?"
<付録.1> SSHの	公開鍵認証について	

⑤ [Enter properties for a new key]の[File name]と[Comment]を入力する。
 Filename:任意のファイル名(Key\_<ユーザ名>\_<ホスト名>)

Comment:任意のコメント

File name	key_<ユーザ名>_<ホスト名>
Comment	2048-bit dsa, <ユーザ名>_〈ホスト名〉, Mon Jun 01 2015 19:31:22 +0900
Enter a pas	sphrase for the key
Passphrase	
Retype pas	sphrase
nuvariceu oj	ntions
uvariceu o	tions

⑥ [Enter a passphrase for the key]の[Passphrase]と[Retype Passphrase]を 入力する。

Passphrase: [任意のパスフレーズ]

Retype Passphrase: [任意のパスフレーズ]

※ 鍵のパスフレーズを[無し]にした場合、秘密鍵があるクライアントから、 パスフレーズ不要で、サーバヘログインできるようになる。このため、秘 密鍵の管理は、十分に注意して運用する必要がある。

Enter properties	for a new key	
File name key_	(ユーザ名)_(木スト名)	
Comment 2048	-bit dsa, <ユーザ名>_<ホスト名>, Mon Jun 01	2015 19:31:22 +0900
Enter a passphra	use for the key	
Passphrase		
Retype passphra	se	
		Passphrase too sh
Advanced Option	8	

「暗号化通信におけるリスク」~ SSHに潜む落とし穴 ~	"暗号化すれば安全ですか?"
<付録.1> SSHの公開鍵認証について	

[Advanced Options]ボタンをクリックする。

ユーザ名>_<ホスト名>
bit rsa, <ユーザ名>_<木スト名>, Mon Jun 01 2015 19:34:51 +0900
se for the key
••••
se ••••
Passphrase

 ⑧ [Key properties]にて、[Key type]と[Key length]を設定し、 [Next]ボタンをクリックする。

[Key type] : RSA

[Key length] : 2048

File name key_<ユーザ名>_<	〈木スト名〉
Comment 2048-bit rsa, <그	L-ザ名>_<木スト名>, Mon Jun 01 2015 19:34:51 +0900
Enter a passphrase for the	key
Passphrase •••	•
Retype passphrase	•
	Passphrase too
Hide Options	Pessphrase too
Hide Options Key properties	Passphrase too
Hide Options Key properties Key type:	Pessphrase too
Hide Options Key properties Key type: Key length:	Pessphrase too RSA 20148
Hide Options Key properties Key type: Key length:	Pessphrase too RSA 2048
Hide Options Key properties Key type: Key length:	Passphrase too RSA 2048
Hide Options Key properties Key type: Key length:	Passobrase too RSA 2048
Hide Options Key properties Key type: Key length:	Passphrase to RSA 2048
Hide Options Key properties Key type: Key length:	Passobrase too RSA 2048
Hide Options Key properties Key type: Key length:	Passobrase too RSA 2048



 ③ [公開鍵(Public-Key)]をアップロードする[Public-Key Authentication Wizard]画面が表示される。ここでは、[Cancel]ボタンをクリックする。

Host name		
Quick connect User name	〈ユーザ名〉	
	Port number	22
Connection profile	e	ssh-TectiaServer-6¥.4¥.6-101 👻

⑩ [Key and Certificate List]に鍵が追加されたことを確認する。

ectia Connections Configuration	n liaili	-
-General		_
-Default Connection	Keys and Certificates	
- Logene - Clerits - Clerits - User Authentication - Broys and Certificates - Broys and Certificates - Broys Addression - Hoat Keys - CAC extinctes - CAC extinctes - CAC Pretech - Automatic Turnels	Manage key pairs and certificates used in user authentication.	
	Default keys O/#Users#Administrator#AppData#Rosming#SSH#UserKeys	
	Default certificates 0.4Users#Administrator#AppData#Roamine#SSH#UserCertificates	
	Additional Directories and Files	
	Tura Contant	
	Add directory.] Add file Delete	
	File name Status Type Comment	
	key_ida RSA 2048 bits 2048-bit dsa, ida, Mon Jun 01 2015 20:05:24 +0900	
	Charge partylener, Upbal. New hey. Defer	
· · · · · · · · · · · · · · · · · · ·		

[鍵ファイルの生成場所]

項目	内容
ファイルパス	[Windows] ①%APPDATA%¥SSH¥UserKeys [Unix/Linux] ②\$HOME/.ssh2/authorized_keys
公開鍵	<作成ファイル名>. pub
秘密鍵	〈作成ファイル名〉



⑪ [Cancel]をクリックし、ウィンドウを閉じる

General - Default Connection	Keys and Certificates	
-Proxy Rules	Manage law name and nertificated used is user authentication	
Clients	Default Leves O'All Devict Administrator (AccoData) Reamine (SS-Million Vers)	
Connection Profiles	Default nertificates ONI Insertificates	
	Additional Directories and Files	
Key Providers		
-Host Keys	Type Content	
GA Certificates		
-CRL Prefetch		
Automatic Tunnels		
	Add disectory Add Els D	eiete
	Hos shectory. Hos me	
	Hou diffectory. Hou fife_	
	Key and Certificate List	
	Key and Certificate List	
	Rey and Certificate List File name Status Type Comment Key, da K54, 2048 bits 2048-bit da, ida, Mon Jun 01 2015 20:05:24 +0900	
	Key and Certificate Lat File name Status Type Comment key_sta RSA 2046 bits 2046-bit das, ids, Mon Jun 01 2015 20:051:24 +0900	
	Key and Certificate List File name Status Type Comment Key_uba RSA 2048 bits 2048-bit dae, ida, Mon Jun 01 2015 20:05:24 +0600	
	Key and Certificate Lat File nome Status Type Comment Key_Uda R5A 2046 bits 2046-bit dia, ids, Mon Jun 01 2015 20:051:24 +0500	
	Key and Certificate List File name Status Type Comment Rey_bds RSA 2046 bits 2048-bit dae, ida, Mon Jun 01 2015 20:05:24 +0500	
	Key and Certificate Lat File nome Status Type Comment Key_sta RSA 2048 bits 2048-bit dae, ida, Mon Jun 61 2015 20:05/24 +0900	
	Rev and Centificate List File name Status Type Comment RoA 2046 bits 2046-bit dae, ida, Mon Jun 03 2015 20:05124 +0500	
	Key und Certificate Lait File nome Status Type Comment Key, uta RSA 2048 bits 2046-bit dia, ida, Mon Jun 61 2015 20:05/24 +0900	
	Rev and Certificate Lat  File name Status Type Comment key_sta  RSA 2046-bits 2046-bit dae, lde, Mon Jun 01 2015 20:05:24 +0000  Genere passphrese, Liblind. New key. D	eiste
	Terr and Certificate Lat           File name         Startus Type         Comment           File name         Startus Type         Comment           key_usta         RSA 2048 bits         2048-bit dia, ida, Mon Jun 01 2015 20:05124 +0000           Okenee passphrase         Ubload.         New key D	eiste
	Commer paraghrane, Lipited.	eiste

(2) SSH サーバの設定

SSH サーバはデフォルトの設定で、パスワード認証・公開鍵認証による接続が可能である。

ここでは、SSH のサーバ側の設定において、セキュリティを向上させるために変更 することが推奨される設定箇所と設定変更方法について記載する。

① Open SSH Server のセキュリティ強化設定

[コンフィグファイルの配置場所]

No	ソフトウェア	ファイルパス
1	Open SSH	/etc/ssh/sshd_config

[SSH サーバのセキュリティ強化設定項目]

上記のコンフ	ィガファィ	イルに記載されている	S下記の項目	の設定を行う
上山ツユマノノ	1 2 7 7 7		リーロックタロ	ツ政にてして

No	設定項目	設定方法
1	待受けポート番号の変更	SSHで接続する場合に使用するポート番号 はデフォルトで22番になっている。外部か らの不正アクセスなどの攻撃は、デフォル トの22番ポートへ行なわれる可能性が高い ため、変更しておくことが推奨される。
		[設定変更例]
		Port 2022
		※ 設定するポート番号は、現在他のサービ スなどで使用されていないポート(1025 ~ 49151番)を使用。
2	root でのログイン禁止	root ユーザで直接ログインできないように 設定する。root ユーザはデフォルトで用意 されているため、パスワード認証が利用でき る場合はパスワードだけを推測して攻撃を 受けやすくなる。その為、root ユーザで直接 ログインはできないようにしておき、一般ユ ーザでログイン後に管理者になることが推 奨される。 [設定変更例] #PermitRootLogin yes PermitRootLogin no

<付録.1> SSHの公開鍵認証について

3	パスワード認証の不許可	公開鍵認証による SSH 接続が行なえるよう に設定した場合はパスワード認証ではログ インできないようにすることが推奨され る。
		[設定変更例]
		#PasswordAuthentication yes PasswordAuthentication no
		※ 公開鍵認証の設定を行う前に上記設定 を実施した場合、コンソール接続からし かアクセスできなくなるため注意が必 要。
4	ログ設定のチューニング	OpenSSHのデフォルト設定では、通信日時/ 接続元 IP アドレス/ユーザといった情報し かログ出力されない。ログの出力ログレベ ルをチューニングし、ファイル操作に関す るログを取得する。
		■操作ログの出力レベル
		#LogLevel INF0
		LogLevel VEKBUSE
		Subsystem sftp /usr/libexec/openssh/sftp- server
		□修正後
		Subsystem sftp /usr/libexec/openssh/sftp- server -f authoriv -l VERBOSE
		※ ただしログが大量に出力されるように なるため、ログローテーションの設計が

<付録.1> SSHの公開鍵認証について

		必要になる。
5	公開鍵の配置場所の変更	一般的に公開鍵は各ユーザのホームディレ クトリ配下の.sshフォルダに配置されるこ とが多く、該当のフォルダの鍵ファイルに 意図しない公開鍵を追加されてしまった場 合、意図しないアクセスを許可してしまう ことになるため、公開鍵の配置場所を変更 しておくことが推奨される。
		[設定変更例]
		□変更前
		#AuthorizedKeysFile .ssh/authorized_keys
		□変更後
		AuthorizedKeysFile 〈任意の格納場所 〉/authorized_keys

② Tectia SSH Server のセキュリティ強化設定

[コンフィグファイルの配置場所]

No	ソフトウェア	ファイルパス	
1	Tectia SSH Server	/etc/ssh2/ ssh-server-config.xml	

[SSH サーバのセキュリティ強化設定項目]

No	設定項目	設定方法
1	待受けポート番号の 変更	SSHで接続する場合に使用するポート番号 はデフォルトで22番になっている。外部からの不正アクセスなどの攻撃は、デフォル トの22番ポートへ行なわれる可能性が高いため、変更しておくことが推奨される。 [設定変更例] □変更前 <li><li><li><li><li><li><li><li><li><li< td=""></li<></li></li></li></li></li></li></li></li></li>

<付録.1> SSHの公開鍵認証について

		スなどで使用されていないボート (1025 ~ 49151 番)を使用。
2	root でのログイン禁止	root ユーザで直接ログインできないように 設定する。root ユーザはデフォルトで用意 されているため、パスワード認証が利用でき る場合はパスワードだけを推測して攻撃を 受けやすくなる。その為、root ユーザで直接 ログインはできないようにしておき、一般ユ ーザでログイン後に管理者になることが推 奨される。
		[設定変更例]
		下記の追加設定を <authentication-methods xxx&gt;の最下部に追記する。</authentication-methods 
		□追加設定
		<authentication action="deny"> <selector> <user-privileged value="yes"></user-privileged> </selector> </authentication>
3	パスワード認証の 不許可	公開鍵認証による SSH 接続が行なえるよう に設定された場合はパスワード認証ではロ グインできないようにすることが推奨され る。
		[設定変更例]
		□削除項目
		<authentication-methods xxx="">にある下記の項目 を削除。</authentication-methods>
		<auth-password></auth-password>
		※ 公開鍵認証の設定を行う前に上記設定 を実施した場合、コンソール接続からし かアクセスできなくなるため注意が必 要。
4	公開鍵の配置場所の 変更	一般的に公開鍵は各ユーザのホームディレ クトリ配下の.sshフォルダに配置されるこ とが多く、該当のフォルダの鍵ファイルに 意図しない公開鍵を追加されてしまった場 合、意図しないアクセスを許可してしまう ことになるため、公開鍵の配置場所を変更 しておくことが推奨される。

### <付録.1> SSHの公開鍵認証について

	[設定変更例]
	□変更前
	<authentication-methods>内の下記を変更する。 <auth-publickey authorized-keys-<="" th=""></auth-publickey></authentication-methods>
	directory="%D/.ssh2/authorized_keys" $/\!\!>$
	□変更後
	<auth-publickey authorized-keys-directory="<任</th>
	意のディレクトリ>/authorized_keys″/>
	※[%D] もしくは[%homedir%]はユーザーの ホームディレクトリです。

(3) ユーザ公開鍵の配置

ここではサーバ管理者向けに、すでに下記のようなサーバのセキュア化が実施さ れている環境を想定したユーザ公開鍵の配置方法を説明する。

[サーバのセキュア化及び設定状況]

- ① 0S に公開鍵認証を行うユーザがすでに追加されている環境を想定
- ② 公開鍵のサーバへの配置は、サーバの管理者が実施する環境を想定
- ③ サーバではパスワード認証の不許可の設定がされている環境を想定
- ④ サーバへの root でのログイン禁止の設定がされている環境を想定

■OpenSSH サーバの公開鍵の配置手順

[デフォルトの公開鍵認証の配置場所]

ファイル	場所	パーミッション
SSH ディレクトリ	〈ユーザディレクトリ〉/. ssh/	700
公開鍵ファイル	〈ユーザディレクトリ〉/.ssh/	600
	authorized_keys	

- ① メンテナンスユーザの公開鍵認証で対象のサーバへログインする。
- ② 公開鍵認証のユーザに切り替える。
  - \$ su 〈公開鍵認証を行うユーザ〉
  - ※ 公開鍵ディレクトリ/公開鍵ファイルを作成するため、 所有権を該当ユーザに変更するために必要な操作になる。 所有権を後程変更できる場合は不要。
- ③ ユーザのホームディレクトリに移動する。

\$ cd \$HOME

- ④ 公開鍵認証の設定ディレクトリ".ssh"が作成されているか確認する。
  - \$ ls -la
  - ※ 初回時(ユーザ作成時)デフォルトでは".ssh"が作成されないので、
  - ".ssh"が作成されていないことを確認する。
- ⑤ 作成されていない場合は".ssh"ディレクトリを作成する。

\$ mkdir .ssh

- ⑥ ".ssh"ディレクトリのパーミッション(アクセス権)を設定する
  - $\$  chmod 700 .ssh
  - ※ 所有者にだけ読み取り権、書き込み権、実行権(パーミッション"700")を 設定する。

⑦ ".ssh"ディレクトリに移動する。

\$ cd .ssh

- ⑧ 公開鍵登録ファイルを作成する。
  - \$ touch authorized\_keys
- ⑨ 公開鍵登録ファイルのパーミッション(アクセス権)を設定する。
  - \$ chmod 600 authorized\_keys
  - ※ 所有者にだけ読み取り権、書き込み権(パーミッション"600")を設定する。
- 10 公開鍵登録ファイルに公開鍵を登録する。
  - (a) サーバ上で、公開鍵登録ファイルをテキストエディタで開く。
    - \$ vi authorized\_keys
  - (b) クライアント PC 上で公開鍵をテキストエディタで開く。
  - (c) この文字列を全て選択してコピーする。
  - (d) コピーした文字列をサーバ上の authorized\_keys(公開鍵登録ファイル) へ、ペーストする。
    - ※ あらかじめ公開鍵ファイルをアップロードできる場合は、下記のコ マンドで、公開鍵を公開鍵登録ファイルに登録することが可能。
      - \$ cat <公開鍵ファイル> >> \$home.ssh/authorized\_keys

16

■Tectia SSH Server の公開鍵の配置手順

[デフォルトの公開鍵認証の配置場所]

ファイル	場所	パーミッション
SSH ディレクトリ	〈ユーザディレクトリ〉/.ssh2/	700
公開鍵格納ディレクト リ	〈ユーザディレクトリ〉/.ssh2/ authorized_keys	700
公開鍵ファイル	<任意の公開鍵ファイル名>	600

① メンテナンスユーザの公開鍵認証で対象のサーバへログインする。

② 公開鍵認証のユーザに切り替える。

\$ su 〈公開鍵認証を行うユーザ〉

- (a) 公開鍵ディレクトリ/公開鍵ファイルを作成するため、 所有権を該当ユーザに変更するために必要な操作になる。 所有権を後程変更できる場合は不要。
- ③ ユーザのホームディレクトリに移動する。

\$ cd \$HOME

- ④ 公開鍵認証の設定ディレクトリ".ssh2"が作成されているか確認する。
  - \$ 1s -1a
  - ※ 初回時(ユーザ作成時)デフォルトでは".ssh2"が作成されないの で、".ssh2"が作成されていないことを確認する。
- ⑤ 作成されていない場合は". ssh2"ディレクトリを作成する。

**\$** mkdir .ssh2

- ⑥ ".ssh2"ディレクトリのパーミッション(アクセス権)を設定する。
  - \$ chmod 700 .ssh2
  - ※ 所有者にだけ読み取り権、書き込み権、実行権(パーミッション"700")を 設定する。
- ⑦ ".ssh2"ディレクトリに移動する。

\$ cd .ssh2

⑧ 公開鍵の格納ディレクトリを作成する。

\$ mkdir authorized\_keys

- ⑨ 公開鍵の格納ディレクトリのパーミッション(アクセス権)を設定する。
  - \$ chmod 700 authorized\_keys
  - ※ 所有者にだけ読み取り権、書き込み権、実行権(パーミッション"700")を 設定する。
- ⑩ 公開鍵を公開鍵格納ディレクトリに転送する。
  - \$ scp <公開鍵ファイル> <ユーザ名>@<サーバ>:<鍵格納ディレクトリ>
- (4) 公開鍵認証方式での接続方法
  - ここではクライアントにおける公開鍵認証での接続方法について説明する。

■TeraTerm での接続方法

- ① TeraTerm を起動する。
- ② [新しい接続]ダイアログが表示されるので、接続先のサーバのホスト名、
  - もしくは IP アドレスを入力し、「OK」ボタンを押す。

Tera Term:新しい接続	
<ul> <li>● TCP/IP</li> <li>ホスト(T): myhost.example.com</li> <li>ビヒストリ(O)</li> <li>サービス: ○ Tclnet</li> <li>TCPボート#(P): 22</li> <li>● SSH SSH/ 「ージョン(V): SSH2 ・</li> <li>● その他 プロトコルレ(C): UNSPEC ・</li> </ul>	
<ul> <li>シリアル(E) ボート(R): ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・</li></ul>	
ок キャンセル ヘルプ(H)	

③ SSH 認証ダイアログが表示されるので、ユーザ名を入力し[RSA/DSA/ECDSA 鍵を使う]にチェックをつけ、[秘密鍵]ボタンをクリックする。

SSH223
ログイン中: 192.168.7.100
認証が必要です。
ユーザ名( <u>N</u> ): test002
バスフレーズ(P):
- , バスワードをメモリ上に記憶する(M)
┌ エージェント転送する(2)
C -11. () =+ 7 k # >(1)
で <u>R</u> SA/DSA/ECDSA鍵を使う  「花密鍵低)」 に*Documents and Settings
C rhosts(SSH1)を使う ローカルのユーザ名(U)
ホスト鍵(E):
○ チャレンジレスボンス認証を使う(キーボードインタラクティブ)(①)
○ Pageantを使う

④ ファイルの選択画面が表示されるので、③で入力したユーザの秘密鍵を選択し、[開く]ボタンをクリックする。

秘密鍵ファイルの選	訳	• 6	1 1 1 1 1 1	? X
tectia-key id_rsa				
」 ファイル名(N):	identity		-	III III III III III III III III III II
	秘密線ファイルの道 デ新しいフォルダ feetia-key ■ id_rea マッイルを(M)-	秘密線ファイルの変訳 ● 新しいフォルダ ■ tectia-key ■ id_rsa	設置違ファイルの選択 ● 新しいフォルダ ■ tectia-key ■ id_rea > 2 - (1 - 2 (th)-	総密線ファイルの変訳 ● 新しいフォルダ ■ tectia-key ■ id_rsa

⑤ 下記画面に戻るので、[OK]ボタンをクリックする。

SSH퀸타	×
ログイン中: 192.168.7.100	
認証が必要です。	
ユーザ名(N): test002	
ノ ( フ つ レ ー ブ ' ( P)	
レンシレ スペン・ ロ パフロードをマモリトに記憶する(M)	
「エージェント転送する(の)	
● RSA/DSA/ECDSA鏈を使う 【 秘密鍵 ⑥ ↓ C¥Documents and Settings	
C rhosts(SSH1)を使う ローカルのユーザ名(型)	
赤スト鍵(E)	
○ チャレンジレスポンス認証を使う(キーボードインタラクティブ)(©)	
C Pageantを使う	
OK	

⑥ 下記、リモート操作画面が表示される。

192.168.7.100:22 - test002@ssh-tectiaclient-6-4-6: <sup>™</sup> VT	_ 🗆 🗙
ファイル(E) 編集(E) 設定(G) コントロール(Q) ウィンドウ(W) ヘルブ(H)	
Last login: Wed Jun 10 2015 18:01:59 +0900 from 192.168.1.96 [test002@ssh-tectiaclient-6-4-6 ~]\$ ∎	
	•

■Tectia SSH Client での接続方法

- ① Tectia SSH Client を起動する。
- ② SSH Terminal 画面が表示されるので、[Quick Connect]をクリックする。



③ Connect to Server 画面が表示されるので、[Hostname]に[接続先のホスト名もしくは IP アドレス]、[User Name]に[ユーザ名]を入力し、[Connect]を[クリック]する。

I lash Manage	-	
Inost Name:	1	Lonnec
User Name:		Cancel

④ 下記、リモート操作画面が表示される。



### 〈補足〉

上記では、鍵の作成/配布、サーバでのセキュア化設定について述べてきたが、 これらの作業はサーバ台数が多くなると、サーバ毎、ユーザ毎に手動での作業が 必要となり非常に手間がかかることが想定される。

Tectia SSH製品のUniversal SSH Key Manager を使うことで、管理 GUI 上から 一元的にかつ簡単に各サーバ上での鍵生成/配布といった操作やサーバ設定が 行えるため、結果、管理者の作業負担を軽減でき、運用作業の効率化につながる。 是非、Tectia 製品の導入をご検討ください。

~ Universal SSH Key Managerの 鍵管理画面 ~

From	То
p root@172.16.32.110	😺 root@172.16.32.111
maintenance@172.16.32.111	root@172.16.32.110
/> guest1@172.16.32.110	↓ dit@172.16.32.111
Unknown private key @172.16.32.1	🥪 🔥 root@172.16.32.110
🔑 🚹 sato@172.16.32.110	🥪 🔥 root@172.16.32.111
p root@172.16.32.111	6 Authorized keys on 1 host; eigyo@172.16.32.110 guest1@172.16.32.110 kanri@172.16.32.110 root@172.16.32.110

※送信元(From)、送信先(To)、ユーザ@IPアドレスの形式で分かり易く表示。

以上

ssh® and Tectia® are registered trademarks of SSH Communications Security Corporation in the United States and in certain other jurisdictions.

SSH and Tectia logos and names of SSH products and services are trademarks of SSH Communications Security Corporation and are protected by international copyright laws and treaties.

Logos and names of the products may be registered in certain jurisdictions.

Copyright  ${\circ}$  2014-2015 SSH Communications Security Corporation. All rights reserved.