SSH Communications Security

HULFT の通信をよりセキュアに

HULFT と SSH Tectia を組み合わせたセキュアで強力なファイル転送

Compatibility Note

2008 年 9 月

株式会社セゾン情報システムズの企業内、企業間通信ミドルウェアである HULFT は、ファイル転送のアプリケーションとして、主に流通業、製造業で大きなシェアを誇るパッケージソフトウェアです。

SSH Tectia ソリューションを HULFT と組み合わせることで、HULFT によるデータ通信をより強力なセキュリティで保護することができます。

本書では、SSH Tectia の透過的な TCP トンネリング機能を使い、HULFT の設定を変更せずに強力な暗号化通信を実現する方法を紹介します。

CONTENTS

1　シナリオ

1.1　はじめに

本書は、株式会社セゾン情報システムズの企業内、企業間通信ミドルウェアである HULFT と SSH Communications Security 社の SSH Tectia を組み合わせてセキュアで強力なファイル転送を実現するための相互接続に関する資料となっています。

HULFT では主に以下のような機能をサポートしており、ファイル転送のアプリケーションとしては導入企業数 5700 社、販売本数 105,000 本（2008 年 3 月時点）の実績を有し、主に流通業、製造業で大きなシェアを誇るパッケージソフトウェアとなっています。

1.1.1　HULFTの主な機能

- 集配信機能、集配信管理機能
- アプリケーション連携機能、ファイル連携機能
- 非同期/同期転送機能
- リモートジョブ実行
- コード変換機能、データ圧縮機能
- メッセージ送信機能

1.1.2　透過的TCPトンネリング

SSH Tectia Client の機能の一つとして透過的 TCP トンネリング機能があります。これを使うことで上位のアプリケーションの設定を変更することなく、高度な暗号化通信を提供することが可能で、安全なファイル転送を実現します。

1.1.3　SSH Tectiaでサポートしている暗号化アルゴリズム

SSH Tectia でサポートしている暗号化アルゴリズムは、以下のとおりです。

3des-cbc、aes128-cbc、aes192-cbc、aes256-cbc、arcfour、blowfish-cbc、twofish-cbc、

twofish128-cbc、twofish192-cbc、twofish256-cbc、crypticore128@ssh.com、seed-cbc@ssh.com

また、SSH Tectia では FIPS140-2 認定を受けた暗号化ライブラリをサポートしており、これを使うことで、より高度なセキュリティ要件にも柔軟に対応することが可能です。

1.1.4　FIPSモードの場合の暗号化アルゴリズム

FIPS モードの場合の暗号化アルゴリズムは、以下のとおりです。

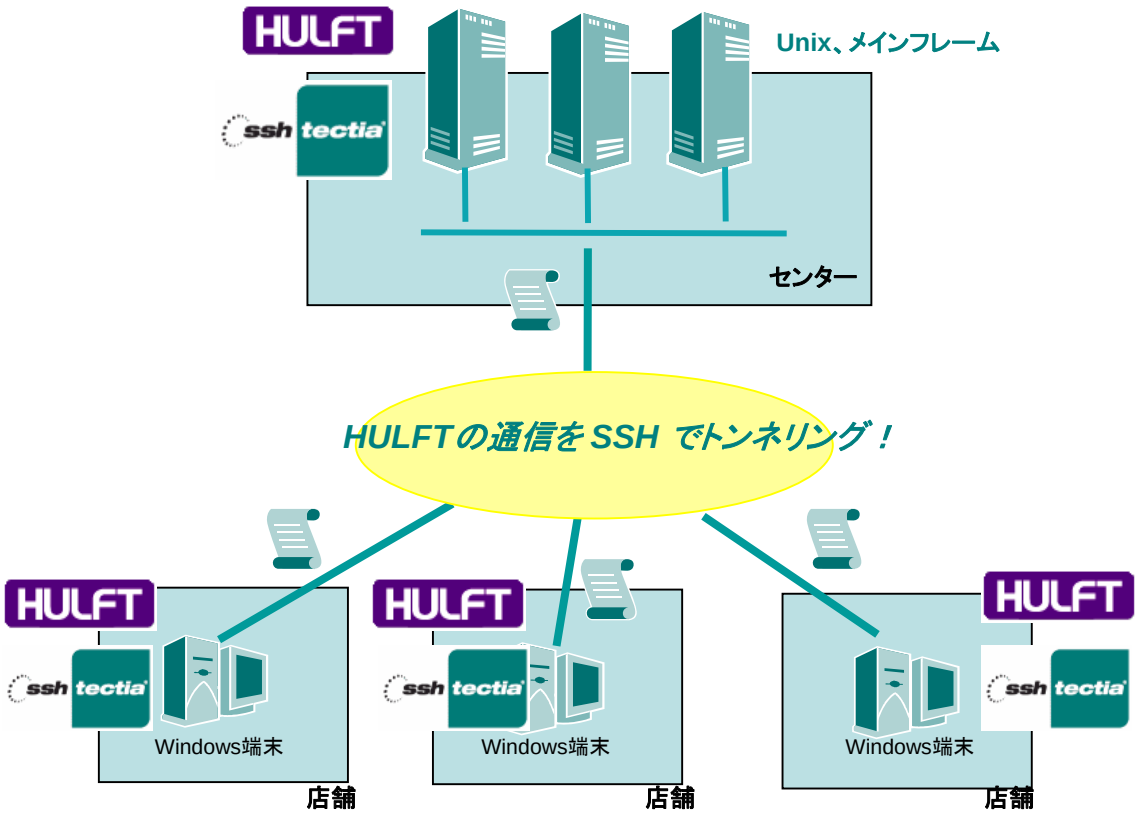
3des-cbc、aes128-cbc、aes192-cbc、aes256-cb

1.2 利用シナリオ

既に HULFT が導入されているファイル転送環境において、SSH Tectia を利用して強力な暗号化通信を行うまでを想定します。

1.2.1 利用イメージ

各店舗からセンターに対して日々の売り上げデータなどが HULFT を使ってセンターに配信されているような環境において、店舗側 Windows 端末に Tectia Client、センター側 Unix やメインフレームに Tectia Server を導入することで安全なファイル転送を実現します。



1.3 ハードウェア/ソフトウェア

本書は、以下の環境を想定しています。

	クライアント	サーバ
OS	Windows XP SP2	Red Hat Exnterprise Linux 4
ソフトウェア	SSH Tectia Client 6.0.1	SSH Tectia Server 6.0.1
HULFT ソフトウェア	HULFT for Windows Type WIN-CL Ver.06.03.03A	HULFT V06.03.02 Type-U1

HULFT を利用するにあたっては、以下の 2 点に注意する必要があります。

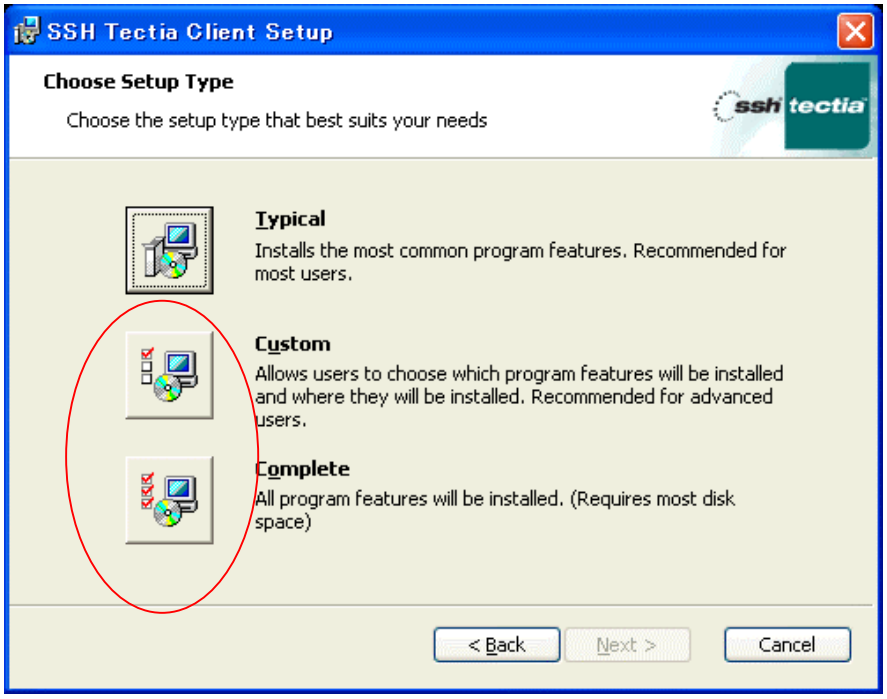
1. 配信側、集信側双方のマシンで HULFT ソフトウェアが必要です。
2. ホスト名を利用して通信を行うため、名前解決ができる必要があります。

2　インストールガイド

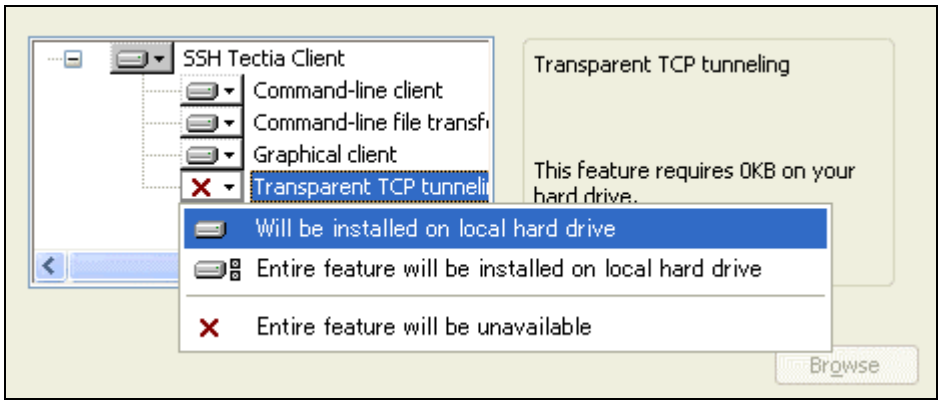
2.1　クライアント

2.1.1　クライアントのインストール

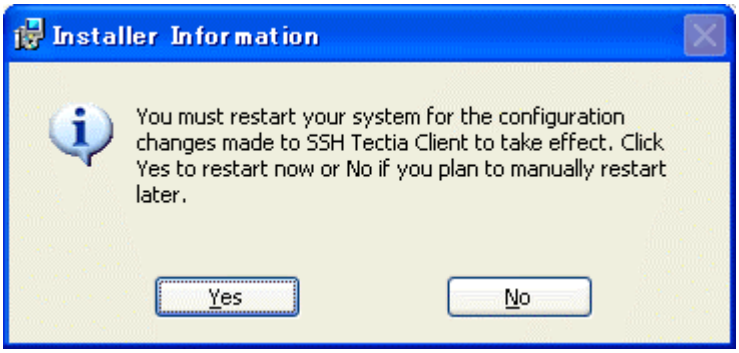
透過的 TCP トンネリングの機能を使用するためには SSH Tectia Client のインストーラを起動後、セットアップタイプのメニューで Custom か Complete を選択する必要があります。



Custom を選択した場合は、Transparent TCP tunneling（透過的 TCP トンネリング）の機能を有効にしてインストールを行ってください。



インストールは再起動を行うことで完了します。



2.2 サーバ

2.2.1 サーバのインストール

SSH Tectia Server のインストールは、必要な各パッケージをコマンドからインストールします。Red Hat Enterprise Linux の x86 用のインストールには、以下の 2 つのパッケージが必要です。

- ssh-tectia-common-6.0.1.10-linux-x86.rpm
- ssh-tectia-server-6.0.1.10-linux-x86.rpm

rpm コマンドでパッケージをインストールした後、製品版ライセンスファイル（sts60.dat）を手動で/etc/ssh2/license ディレクトリにコピーする必要があります。

2.2.2 サーバの設定

/etc/ssh2 ディレクトリ配下には、いくつかのサンプル設定があります。

基本設定として ssh-server-config-default.xml がありますので、ssh-server-config.xml としてコピーし、これを編集して利用してください。デフォルトの内容で公開鍵認証が利用できる状態なので特に変更することなく利用することが出来ます。

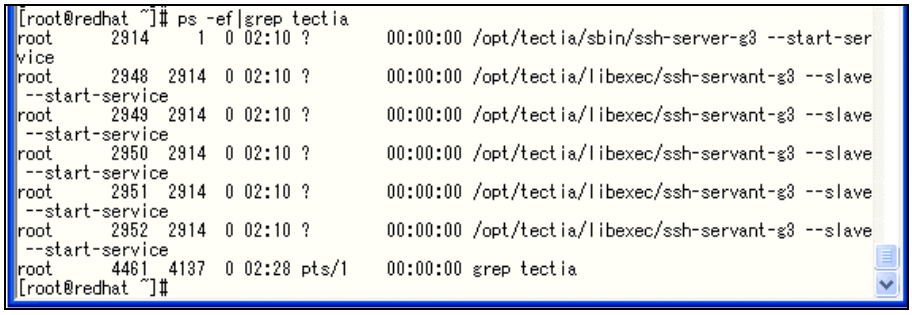
2.2.3 サーバの動作確認

SSH Tectia Server サービスの起動と停止はコマンドから行います。設定を変更した後は、必ずサービスの再起動を行ってください。

起動：/etc/init.d/ssh-server-g3 start

停止：/etc/init.d/ssh-server-g3 stop

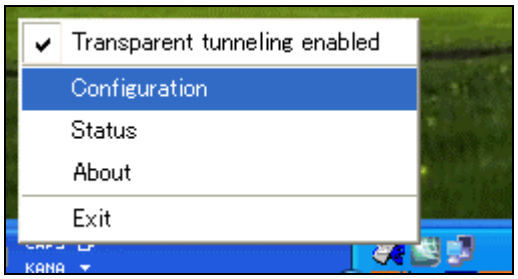
サーバが動作しているかは、ps コマンドを利用して確認してください。



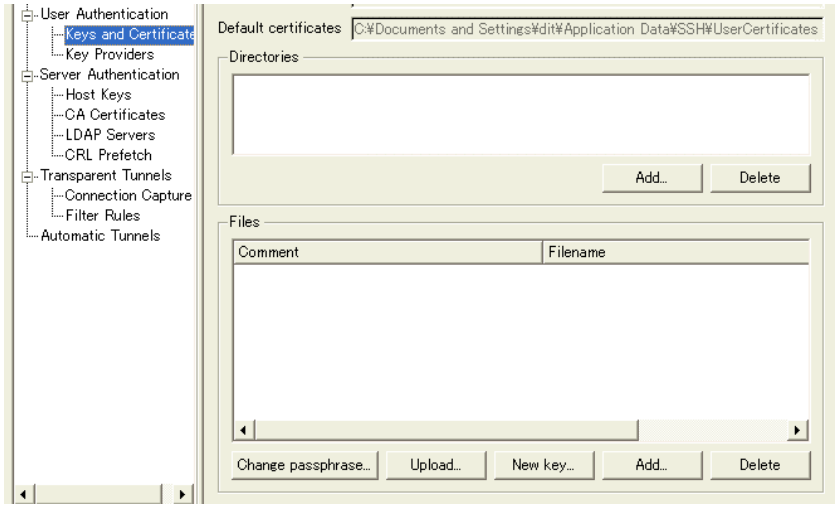
3　設定

3.1　鍵の生成とアップロード

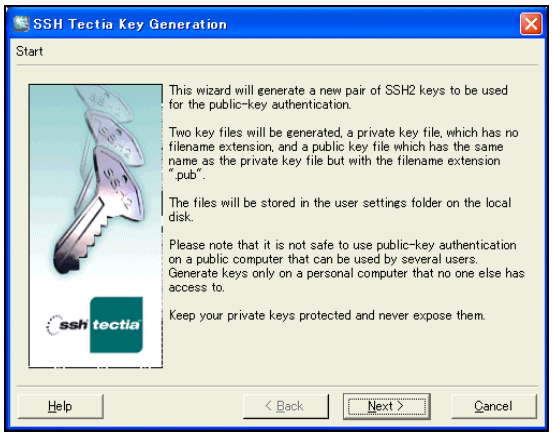
タスクトレイにある SSH Tectia のアイコンを右クリックし、Configuration を選択します。



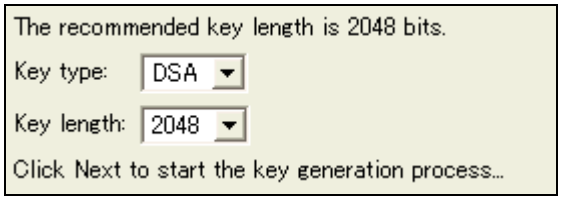
User Authentication⇒Keys and Certificate を選択します。



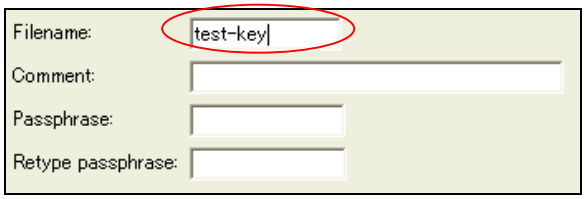
New Key を選択して、鍵の生成を開始します。



鍵の長さは 768bit から 2048bit の範囲で生成が可能です。長ければ長いほどセキュリティの強度が高くなります。

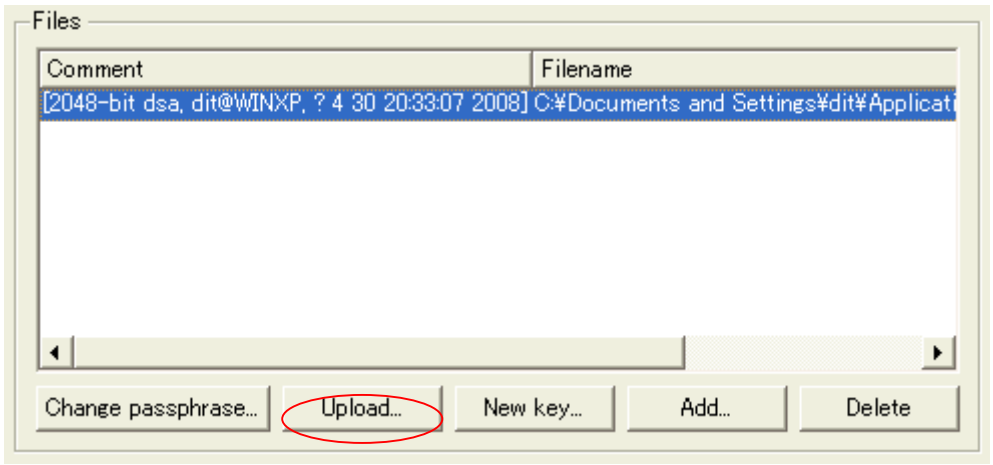


指定する鍵のファイル名は任意です。また、自動接続を行うため、パスフレーズは設定しないで次へ進んでください。

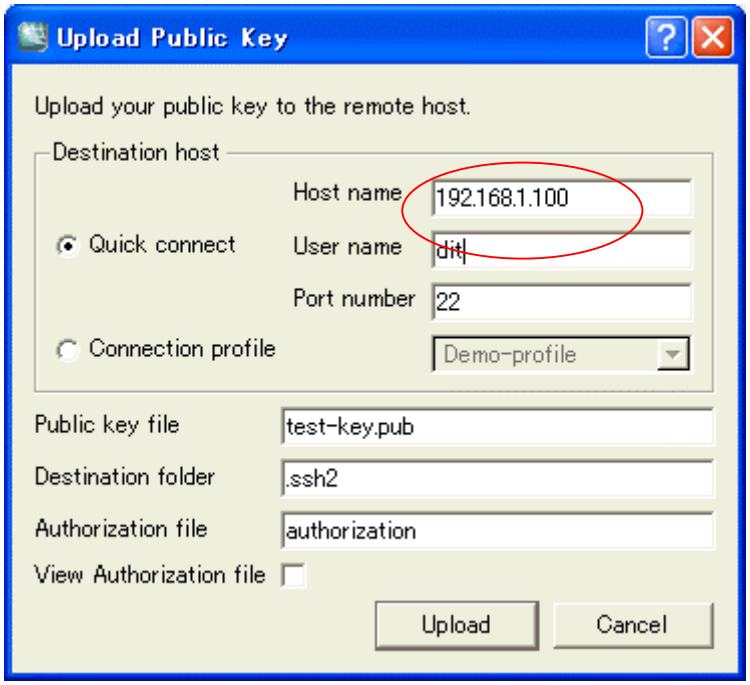


鍵の生成が終了したら接続先の SSH サーバへ公開鍵をアップロードします。

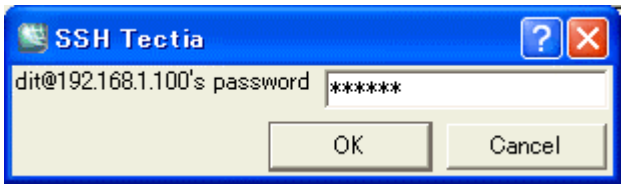
生成した鍵ファイルを選択し、Upload ボタンをクリックします。



Quick connect のラジオボタンを選択し、Host name に 192.168.1.100、User name に dit と入力し、Upload をクリックします。



パスワードを求められますのでパスワードを入力します。

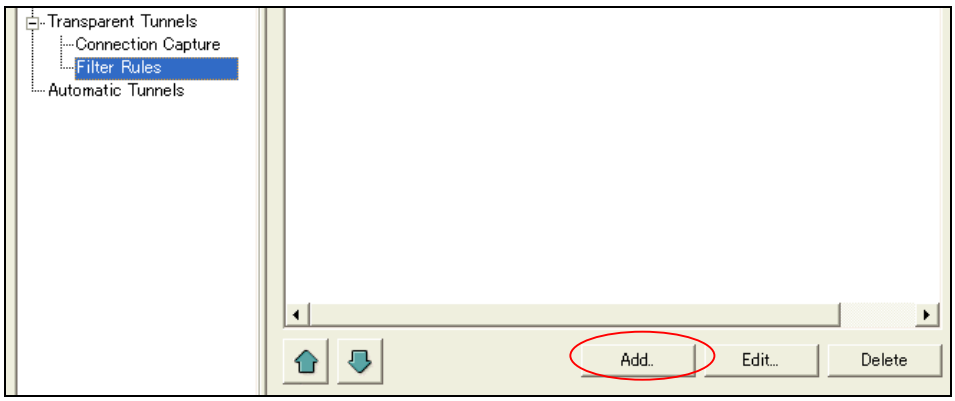


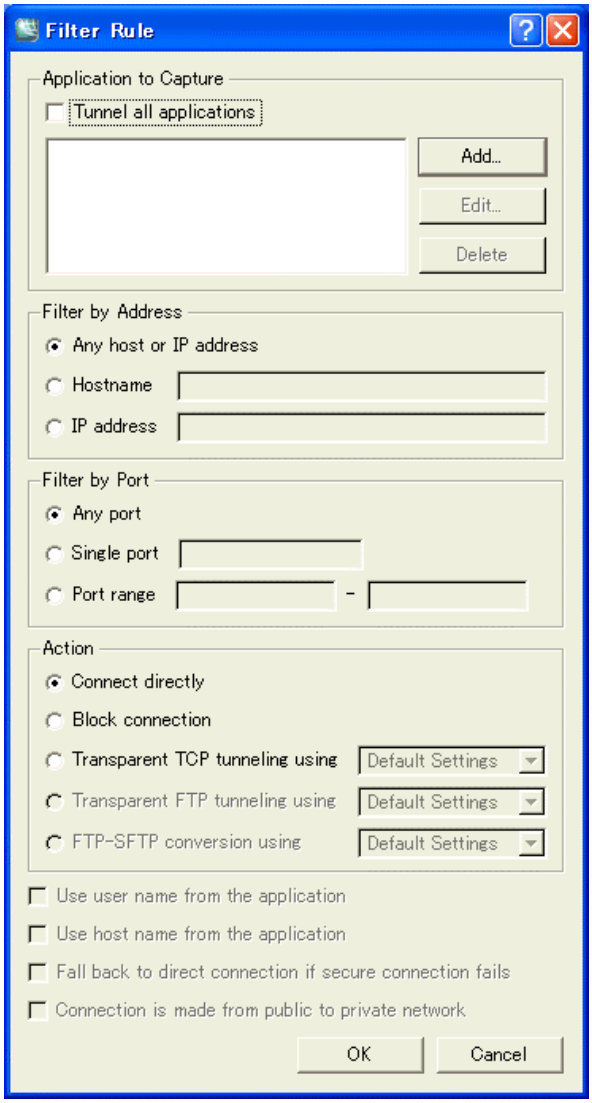
鍵のアップロードが成功すると下記のようなダイアログが表示されます。

3.2 SSHのトンネリング設定

HULFT の通信を SSH でトンネリングするために、Transparent TCP tunneling の設定を行います。

Configuration⇒Transparent Tunnels⇒Filter Rules を選択、Add をクリックします。



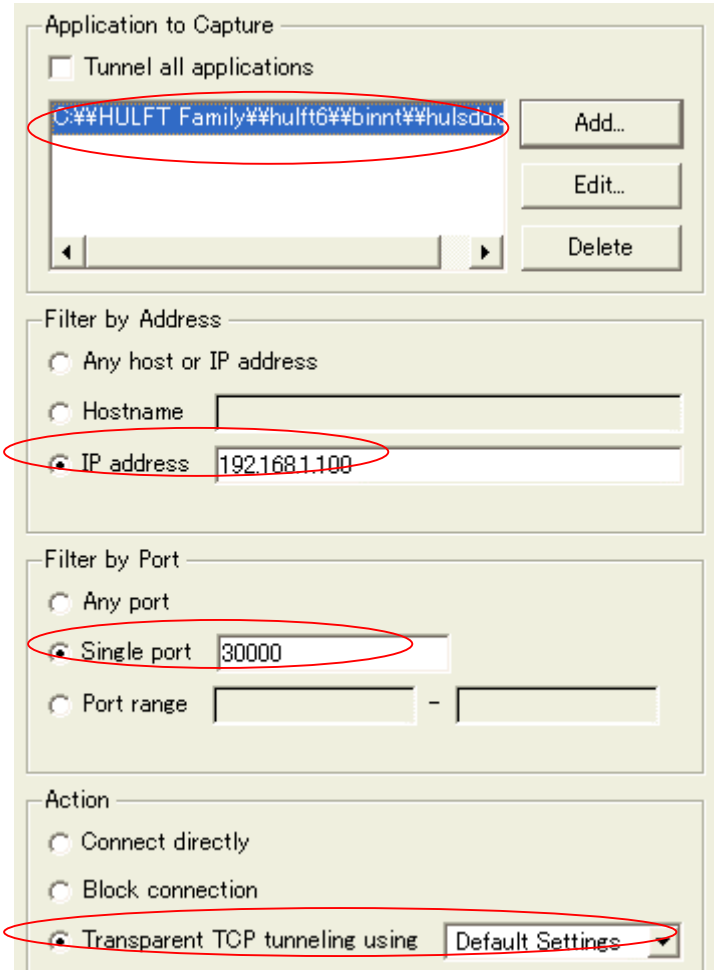


Filter Rule の内容として以下を設定します。

- トンネル対象とするアプリケーションプログラム
- トンネル対象とする接続先ホスト
- トンネル対象とする通信のポート番号

ここでは HULFT の通信をトンネル対象とするため以下のように設定を行います。

- トンネル対象とするアプリケーション：
  C:\HULFT Family\hulft6\binnt\hulsdd.exe
- トンネル対象とする接続先ホスト：
  192.168.1.100
- トンネル対象とするポート番号：
  30000



Action として、Transparent TCP tunneling を選択します。

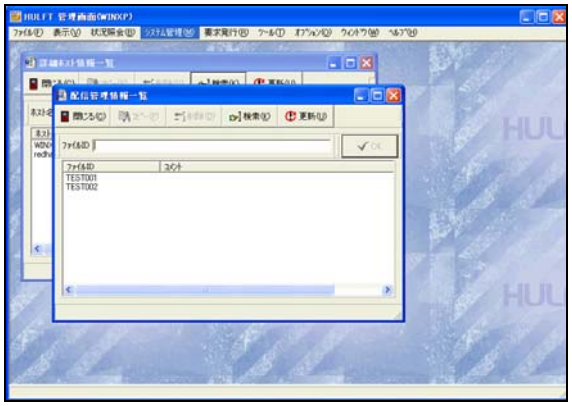
設定が終了したら、Apply⇒OK をクリックして終了してください。

4　動作確認

実際に HULFT を使ってファイル転送を行い、通信が暗号化されるか確認をします。

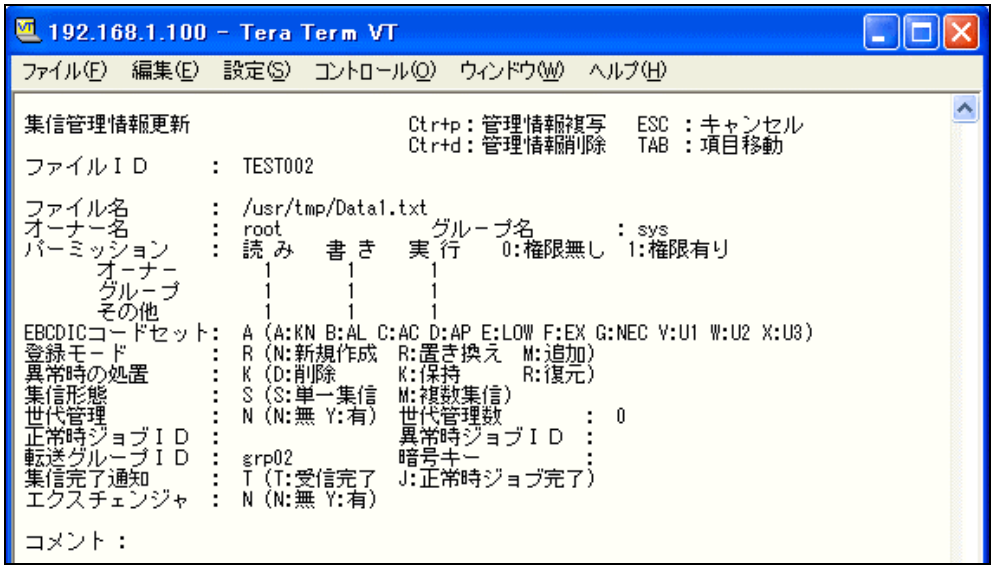
スタートメニュー⇒プログラム⇒HULFT Family⇒HULFT for Windows Ver.6 から HULFT の管理画面を選択して立ち上げます。

あらかじめ設定されている配信設定を確認します。



配信側設定

集信側設定



要求発行から配信要求を選択し、redhat サーバ（グループ ID：grp02）の HULFT へ配信を行います。

要求発行が正常終了すると、同じタイミングでファイル転送が行われ、実際のファイル転送が SSH を経由して行われます。

ファイル転送中のステータスは SSH Tectia の Status ウィンドウで確認をすることができます。