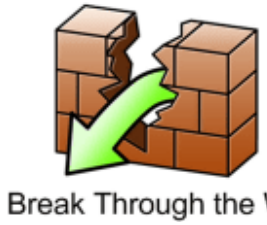


UT-VPN

Startup Guide for UNIX

このドキュメントは、UT-VPN の UNIX 版をビルドし、インストールし、起動して初期設定するまでの基本的な手順を示しています。このドキュメントは、現在日本語でのみ記述されています。後日、UT-VPN の英語版をリリースする際には、このドキュメントも英語に翻訳される予定です。

なお、UT-VPN のソースコードの改造や再配布などの方法については「UT-VPN_Developers_Guide.pdf」をお読みください。

1. UT-VPN UNIX 版の動作環境

UT-VPN UNIX 版の動作環境は以下のとおりです。

オペレーティングシステム

UT-VPN はソースコード形式で提供されますので、必ずしも以下の OS 以外の環境では動作しないということではありません。ソースコードを適切にコンパイルすることができる環境であれば、恐らく問題なく動作することができると思われます。たとえば、Solaris のバージョン 7 以前にも導入することは可能だと考えられます。

しかし、UT-VPN プロジェクトの開発者は、主に以下の環境の OS で動作を確認しています。

系列	バージョン	UT-VPN Server	UT-VPN Client
Linux	カーネル 2.4 または 2.6	◎	◎
FreeBSD	6.x 以降	◎	× (動作しません)
Solaris	8 以降	◎	× (動作しません)
Mac OS X	Tiger, Leopard または Snow Leopard	△ (ローカルブリッジ機能が動作しません)	× (動作しません)

CPU

UT-VPN はソースコード形式で提供されます。ソースコードはエンディアンの違いやビット数 (32-bit または 64-bit) の違いにかわらず適切に動作するように注意してプログラミングされているため、原理上は、上記の OS が動作するすべての CPU で動作します。

なお、UT-VPN の開発者は、以下のような環境で動作をテストしています。もちろん、これ以外の CPU でも正しく動作すると思われます。

OS	CPU
Linux	Intel x86 (32-bit) Intel x64 / AMD64 (64-bit) SH-4 MIPS (Little-Endian) ARM PPC
FreeBSD	Intel x86 (32-bit) Intel x64 / AMD64 (64-bit)
Solaris	Intel x86 (32-bit) Intel x64 / AMD64 (64-bit) SPARC (32-bit) SPARC (64-bit)
Mac OS X	Intel x86 (32-bit)

OS	CPU
	Intel x64 / AMD64 (64-bit) PowerPC (32-bit) PowerPC (64-bit)

ライブラリ

UT-VPN をコンパイル、起動するためには、以下のライブラリがインストールされていることが必要です。これらのライブラリは、OS のディストリビューションの一部として提供されていることもありますが、もしインストールされていない場合は追加のソフトウェアとして OS にインストールするか、または、自分自身でライブラリを取得（ダウンロード）してインストールする必要があります。

これらのライブラリのヘッダファイルもビルド時に必要です。

UT-VPN はこれらのライブラリを単に gcc の-l オプションとして指定してビルド時に参照しようとします。したがって、gcc がこれらのライブラリのダイナミックバージョンまたはスタティックバージョンを参照できるように適切なパスにライブラリのファイルを設置しておいてください。

ライブラリ名	gcc における指定方法	公式配布元の URL
GNU C Library (glibc)	-lm (libm)	http://www.gnu.org/software/libc/
POSIX Threads (pthread)	-lpthread	(OS に組み込まれています)
OpenSSL (crypto, ssl)	-lcrypto -lssl	http://www.openssl.org/
libiconv	-liconv	http://www.gnu.org/software/libiconv/
readline	-lreadline	http://ftp.gnu.org/gnu/readline/
ncurses	-lncurses	http://www.gnu.org/software/ncurses/

ビルドツール

UT-VPN のビルドのためには、以下のツールが必要です。これらは、通常オペレーティングシステムのインストール時にインストール可能です。または、インストール後に追加コンポーネントとしてインストールできます。たとえば、Mac OS X の場合は XCode に含まれています。

要するに、Hello World をコンパイルして実行可能ファイルを生成することができるような gcc の環境があればそれで OK だという意味です。

ソフトウェア名	公式配布元の URL
GNU Compiler Collection (gcc) およびその他のバイナリユーティリティ	http://gcc.gnu.org/

ソフトウェア名	公式配布元の URL
GNU Make (gmake)	http://www.gnu.org/software/make/

2. UT-VPN UNIX 版のダウンロードと展開

ダウンロード

UT-VPN の UNIX 版は、http://utvpn.tsukuba.ac.jp/ からダウンロードすることができます。

Windows 版と異なり、UNIX 版はソースコードの形式 (tar.gz) で提供されています。これは、UNIX のシステムには、OS の種類、依存ライブラリのバージョン、CPU 等の組み合わせによる多種多様な環境があり、すべての環境用の実行可能ファイルを UT-VPN の開発者がビルドし、インストーラを制作するコストは膨大なものになるため、それを避けるために、ソースコードでのみ配布することとしたものであります。

ダウンロードした.tar.gz ファイルは、例えば「utvpn-src-unix-v100-7092-beta-2010.06.25.tar.gz」というような名前になっています。

展開

tar.gz ファイルの展開には、tar を使用します (なお、ここでは GNU の tar を使用することを想定しています。それ以外の OS 付属の tar 等では tar ファイルの展開はできますが gz ファイルの解凍ができない場合があります。この場合は適切なコマンドを利用して展開してください)。

たとえば、/tmp に utvpn-src-unix-v100-7092-beta-2010.06.25.tar.gz をダウンロードした場合は、以下のようにして展開します。

なお、展開やコンパイル等の作業は、一般ユーザーとして実行することが推奨されています。

```
[user@linux tmp]$ tar xzvf utvpn-src-unix-v100-7092-beta-2010.06.25.tar.gz
```

展開が完了すると、たとえば「utvpn-unix-v100-7092-beta」というような展開されたディレクトリが同じディレクトリに作成されていると思われます。そのディレクトリを開き、ファイルの内容物を ls で確認してください。

```
[user@linux tmp]$ cd utvpn-unix-v100-7092-beta
[user@linux utvpn-unix-v100-7092-beta]$ ls
License-ja.txt  configure  makefiles  src
```

上記のように「configure」というファイルが展開されていれば正常です。

3. ビルド

configure の実行

UT-VPN の UNIX 版ソースコードは、手抜きのため、configure は単に makefiles ディレクトリ内の適切な Makefile をカレントディレクトリにコピーしてくるだけの設計になっています。本来は、autoconf 等を用いて伝統的な configure を生成するのが良いとされていますが、UT-VPN の UNIX 版の開発はまずは正しく動作するプログラムを公開することが最優先でしたので、現在のような形になっています。

一般的なソフトウェアをビルドするときと同様に、configure を「./configure」と入力して実行してください。すると、次のように、OS の種類や CPU のビット数を選択する画面が表示されますので、適切に指定してください。

```
[user@linux utvpn-unix-v100-7092-beta]$ ./configure
---------------------------------------------------------------
SoftEther UT-VPN for Unix

Copyright (C) 2004-2010 SoftEther Corporation.
Copyright (C) 2004-2010 University of Tsukuba, Japan.
Copyright (C) 2003-2010 Daiyuu Nobori. All Rights Reserved.

This program is free software; you can redistribute it and/or
modify it under the terms of the GNU General Public License
version 2 as published by the Free Software Foundation.
---------------------------------------------------------------

Please select your Operating System below:
 1: Linux
 2: FreeBSD
 3: Solaris
 4: Mac OS X

Which is your operating system (1-4): 1

Please select your CPU Bits below:
 1: 32-bit
 2: 64-bit

Which is the bits of your CPU (1-2): 1

Makefile is generated. Please execute 'make' to build UT-VPN.
[user@linux utvpn-unix-v100-7092-beta]$ ls -l Makefile
-rwxrwxr-x 1 user user 21313  6月 26 21:07 Makefile
```

configure によって適切な Makefile がカレントディレクトリに生成されていれば完了です。

make の実行

Makefile が作成されれば、それを用いてソースコードのビルドを行うことができます。

方法としては、Makefile があるディレクトリ上で単純に「make」と入力するだけです。

```
-rwxrwxr-x 1 user user 21313  6月 26 21:07 Makefile
[user@linux utvpn-unix-v100-7092-beta]$
[user@linux utvpn-unix-v100-7092-beta]$ make
gcc  -DNDEBUG  -DVPN_SPEED  -DUNIX  -DUNIX_LINUX  -D_REENTRANT  -DREENTRANT
-D_THREAD_SAFE    -D_THREADSAFE    -DTHREAD_SAFE    -DTHREADSAFE    -I./src/
-I./src/Cedar/  -I./src/Mayaqua/  -O2  -fsigned-char  -c  src/Mayaqua/Cfg.c  -o
```

```
tmp/objs/Mayaqua/Cfg.o
gcc  -DNDEBUG  -DVPN_SPEED  -DUNIX  -DUNIX_LINUX  -D_REENTRANT  -DREENTRANT
-D_THREAD_SAFE    -D_THREADSAFE    -DTHREAD_SAFE    -DTHREADSAFE    -I./src/
-I./src/Cedar/ -I./src/Mayaqua/ -O2 -fsigned-char -c src/Mayaqua/Encrypt.c -o
tmp/objs/Mayaqua/Encrypt.o
            :
            :  (以下略)
```

何らかの原因 (たとえばライブラリが足らないとか、include されるべきライブラリのヘッダファイルが見つからないなど) でエラーが発生すると、その旨が表示されます。その場合は、必要なライブラリを入手して再試行してください。

実行結果の確認

make に成功すれば、「output」ディレクトリに以下のような実行可能ファイル等が生成されています。これらの実行可能ファイルの生成に成功すれば、output ディレクトリ以下のこれらの実行可能ファイルを直接指定して実行すれば UT-VPN のプログラムを起動することは可能です。

```
[user@linux utvpn-unix-v100-7092-beta]$ find output/
output/
output/utvpncmd
output/utvpncmd/hamcore.utvpn
output/utvpncmd/utvpncmd
output/utvpnclient
output/utvpnclient/hamcore.utvpn
output/utvpnclient/utvpnclient
output/utvpnserver
output/utvpnserver/hamcore.utvpn
output/utvpnserver/utvpnserver
output/ham
output/ham/hamcore.utvpn
output/ham/ham
```

なお、実行可能ファイルを生成する過程で内部的に生成されたオブジェクトファイル等は、「tmp」ディレクトリに保存されています。

(参考) デバッグモードでのビルド

単純に make を実行すると、リリースモードでビルドが行われます。この場合、最適化レベルは-O2となり、また、デバッグ情報は保存されません。

代わりに「make DEBUG=YES」と指定して実行すると、デバッグモードでビルドが行われます。この場合、最適化は無効になり、デバッグ情報が保存されます。デバッグモードでビルドした実行可能ファイルは、gdb 等を用いてデバッグを行うことができます。

```
[user@linux utvpn-unix-v100-7092-beta]$ make DEBUG=YES
```

4. インストール

make install によるインストール

ビルドが成功したら、「make install」でインストールを行うことができます。インストールといっても、Windows 版のように豪華なインストーラが用意されている訳ではありません。単純に、/usr/ の下に実行可能ファイル等をコピーするだけです。

インストールを行う際には、まず su コマンドで root ユーザーになる必要があります。次に、「make install」と実行します。

```
[user@linux utvpn-unix-v100-7092-beta]$ su
Password: ********
[user@linux /tmp/utvpn-unix-v100-7092-beta]# make install
cp output/utvpnserver/hamcore.utvpn /usr/utvpnserver/hamcore.utvpn
cp output/utvpnserver/utvpnserver /usr/utvpnserver/utvpnserver
echo "#!/bin/sh" > /usr/bin/utvpnserver
echo /usr/utvpnserver/utvpnserver '"$@"' >> /usr/bin/utvpnserver
echo 'exit $?' >> /usr/bin/utvpnserver
chmod 755 /usr/bin/utvpnserver
cp output/utvpnclient/hamcore.utvpn /usr/utvpnclient/hamcore.utvpn
cp output/utvpnclient/utvpnclient /usr/utvpnclient/utvpnclient
echo "#!/bin/sh" > /usr/bin/utvpnclient
echo /usr/utvpnclient/utvpnclient '"$@"' >> /usr/bin/utvpnclient
echo 'exit $?' >> /usr/bin/utvpnclient
chmod 755 /usr/bin/utvpnclient
cp output/utvpncmd/hamcore.utvpn /usr/utvpncmd/hamcore.utvpn
cp output/utvpncmd/utvpncmd /usr/utvpncmd/utvpncmd
echo "#!/bin/sh" > /usr/bin/utvpncmd
echo /usr/utvpncmd/utvpncmd '"$@"' >> /usr/bin/utvpncmd
echo 'exit $?' >> /usr/bin/utvpncmd
chmod 755 /usr/bin/utvpncmd

---------------------------------------------------------------------
Installation completed successfully.

Please execute 'utvpnserver start' to run UT-VPN Server Background Service.
Or please execute 'utvpnclient start' to run UT-VPN Client Background Service.
And please execute 'utvpncmd' to run UT-VPN Command-Line Utility to configure
UT-Server or UT-VPN Client.
---------------------------------------------------------------------
```

上記のように「Installation completed successfully.」と表示されれば完了です。

インストールによって生成されるファイルに関する説明

インストールすると、以下のようにシェルスクリプトが/usr/bin/に作成されます。

ファイル名	UT-VPN ソフトウェア
/usr/bin/utvpnserver	UT-VPN Server バックグラウンドデーモン (サービス) プログラム
/usr/bin/utvpnclient	UT-VPN Client バックグラウンドデーモン (サービス) プログラム
/usr/bin/utvpncmd	UT-VPN Server / Client コマンドライン管理ユーティリティ

なお、上記のファイルはすべて短いシェルスクリプトになっており、実際には、以下のそれぞれのディレクトリにインストールされた実行可能ファイルを起動しているだけです。

ディレクトリ名	UT-VPN ソフトウェア
/usr/utvpnserver/	UT-VPN Server バックグラウンドデーモン (サービス) プログラム
/usr/utvpnclient/	UT-VPN Client バックグラウンドデーモン (サービス) プログラム
/usr/utvpncmd/	UT-VPN Server / Client コマンドライン管理ユーティリティ

utvpnserver および utvpnclient は、実行時にログファイルや設定ファイルを実行可能ファイルが存在するディレクトリと同一のディレクトリ、つまり上記の「ディレクトリ名」で指定されているディレクトリに保存します。/usr/bin/ には実行可能ファイルの本体を起動するための簡単なシェルスクリプトが保存されるだけで、データファイルは保存されません。

アンインストール方法

何らかの理由で UT-VPN をアンインストールするには、上記の表のファイルおよびディレクトリを削除してください。

5. UT-VPN Server の起動と設定方法

バックグラウンドサービス (デーモン) について

UT-VPN Server の本体プログラムは、バックグラウンドサービス (デーモン) として動作します。デーモンプログラムは、一度起動すると、起動したユーザーの画面 (コンソール) には一切文字出力を行わず、また、起動したユーザー (root) がログアウトしても動作し続けます。ユーザーが誰も UNIX にログインしていない状態でも動作し続けます。また、仮に何らかの原因でプログラムがクラッシュした場合、デーモンは自分自身を再起動して動作を続行します。

手動での起動方法

手動で UT-VPN Server サービスを起動するには、root 権限のユーザーとしてログインし、「utvpnserver start」と入力してください。

なお、make install を実行したインストール直後は、お使いのシェルのパスが更新されていない可能性がありますので、一度ログアウトして再度ログインするか、または、「bash」などと入力してシェルを新しく起動し、その上で実行してください。

```
[user@linux /root]# utvpnserver start
UT-VPN Server Service Started.
```

上記のように「start」引数を付けて utvpnserver を起動すると、デーモンプログラムとして動作が開始されます。その後は、ユーザー (root) がログアウトしても動作が継続されます。

なお、「utvpnserver stop」と実行することにより、UT-VPN Server サービスを正常に停止 (シャットダウン) することができます。これを行わずにプロセスや OS ごと強制終了すると、最新の設定変更が失われる可能性があります。UT-VPN Server サービスは、常に 300 秒間 (5 分間) ごとに設定データベースファイル「vpn_server.config」を保存しますが、異常終了した場合、最大 300 秒間のデータが失われる可能性があります (この間隔は、vpn_server.config の「uint AutoSaveConfigSpan 300」の値を編集して変更できます)。

OS の起動時に自動起動する方法

上記の方法で、手動で UT-VPN Server サービスを起動しても、OS が再起動するとサービスは停止してしまいます。理想的には、OS やコンピュータが停電などでいつ再起動しても、サービスが自動的に再開するのが望ましいということができます。

その場合は、OS のスタートアップスクリプトに「/usr/bin/utvpnserver start」を起動するように登録する必要があります。また、OS が終了しようとする際に「/usr/bin/utvpnserver stop」が起動されるように登録することが推奨されます。

OS のスタートアップスクリプトへの登録方法は、OS やディストリビューションによって大きく異なります。たとえば、Linux でもいろいろな方法がありますので、ここで詳しく書くことはできません。

参考情報として次の URL の PacketiX VPN Server の Linux へのインストール方法 (init.d への登録) もご参照ください: http://www.softether.co.jp/jp/vpn2/manual/web/7-3.aspx

utvpncmd (コマンドライン管理ユーティリティ) を用いた設定方法

UT-VPN Server サービスが動作している UNIX コンピュータのコンソール上で UT-VPN Server を設定・操作するためには、「utvpncmd」プログラムを使用します。utvpncmd は UT-VPN Server / UT-VPN Client を管理するための CUI (コマンドライン) のツールです。このツールはローカル (localhost) またはリモート上の UT-VPN Server / UT-VPN Client を管理することができます。

utvpncmd を起動する際は、一般ユーザー権限で OK です。

utvpncmd を普通に起動し、画面の指示に従って文字を入力すると、localhost で動作している UT-VPN Server に接続することができます。

```
[root@linux /root]# utvpncmd
utvpncmd コマンド - UT-VPN コマンドライン管理ユーティリティ
UT-VPN コマンドライン管理ユーティリティ (utvpncmd コマンド)
Version 1.00 Build 7092   (Japanese)
Compiled 2010/06/25 04:31:36 by yagi at pc25
Copyright (C) 2004-2010 SoftEther Corporation.
Copyright (C) 2004-2010 University of Tsukuba, Japan.
Copyright (C) 2003-2010 Daiyuu Nobori.
All Rights Reserved.

utvpncmd プログラムを使って以下のことができます。

1. VPN Server または VPN Bridge の管理
2. VPN Client の管理
3. VPN Tools コマンドの使用 (証明書作成や通信速度測定)

1 - 3 を選択: 1

接続先の VPN Server または VPN Bridge が動作しているコンピュータの IP アドレスまたはホスト名を指定してください。
'ホスト名:ポート番号' の形式で指定すると、ポート番号も指定できます。
(ポート番号を指定しない場合は 443 が使用されます。)
何も入力せずに Enter を押すと、localhost (このコンピュータ) のポート 443 に接続します。
接続先のホスト名または IP アドレス:

サーバーに仮想 HUB 管理モードで接続する場合は、仮想 HUB 名を入力してください。
サーバー管理モードで接続する場合は、何も入力せずに Enter を押してください。
接続先の仮想 HUB 名を入力:
VPN Server "localhost" (ポート 443) に接続しました。

VPN Server 全体の管理権限があります。

VPN Server>
```

上記のように「VPN Server>」というプロンプトが表示され、コマンド受付状態となった場合は、UT-VPN Server への接続は成功しています。

ここでコマンド名がわからない場合は「?」または「HELP」と入力することでコマンド一覧を表示することができます。また、コマンド名に続いて「--help」または「/?」と入力すると、そのコマンドの使い方を表示することができます。以下はその例です。

```
VPN Server>help
下記の 187 個のコマンドが使用できます:
 About                      - バージョン情報の表示
 AcAdd                      - 接続元 IP 制限リストにルールを追加 (IPv4)
 AcAdd6                     - 接続元 IP 制限リストにルールを追加 (IPv6)
 AcDel                      - 接続元 IP 制限リスト内のルールの削除
 AcList                     - 接続元 IP 制限リストのルール一覧の取得
 AccessAdd                   - アクセスリストへのルールの追加 (IPv4)
 AccessAdd6                  - アクセスリストへのルールの追加 (IPv6)
       :
       : (中略)
       :
 UserSet                    - ユーザー情報の変更
 UserSignedSet               - ユーザーの認証方法を署名済み証明書認証に設定

それぞれのコマンドの使用方法については、"コマンド名 /?" と入力するとヘルプが表示されます。
コマンドは正常に終了しました。

VPN Server>HubCreate /?
HubCreate コマンド - 新しい仮想 HUB の作成
コマンド "HubCreate" のヘルプ

[目的]
  新しい仮想 HUB の作成

[説明]
  VPN Server 上に新しい仮想 HUB を作成します。
  作成した仮想 HUB は、直ちに動作を開始します。
  VPN Server がクラスタ内で動作している場合は、このコマンドはクラスタコントローラに対してのみ有効です。また、新しい仮想 HUB は、ダイナミック仮想 HUB として動作します。HubSetStatic コマンドで、スタ　ティック仮想 HUB に変更することもできます。すでに VPN Server 上に存在する仮想 HUB の一覧を取得するには、HubList コマンドを使用します。
  このコマンドを実行するには、VPN Server の管理者権限が必要です。
  また、このコマンドは VPN Bridge およびクラスタメンバサーバーとして動作している VPN Server では動作しません。
  なお、クラスタ上でクラスタコントローラに対して仮想 HUB の作成コマンドを発行する場合は、HubCreateStatic コマンドまたは HubCreateDynamic コマンドを使用してください (クラスタコントローラに対して HubCreate コマンドを使用すると HubCreateDynamic コマンドと同等に動作します)。
```

```
[使用方法]
  HubCreate [name] [/PASSWORD:password]

[パラメータ]
  name      - 作成する仮想 HUB の名前を指定します。
  /PASSWORD - 作成する仮想 HUB の管理パスワードを設定する場合は、その管理パスワードを指定します。指定しない場合は、入力するためのプロンプトが表示されます。
```

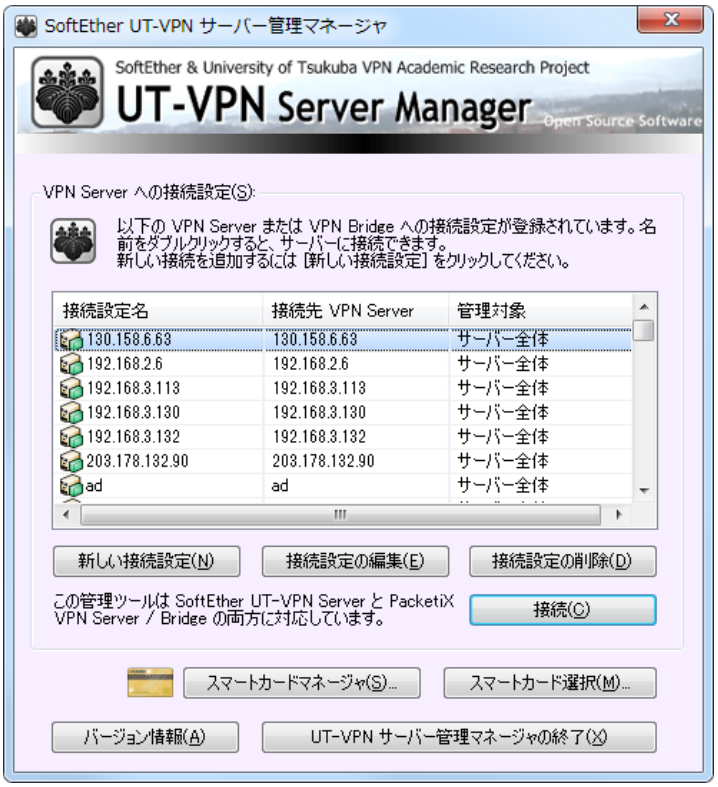
このように、utvpncmd のコマンドライン・リファレンスは、utvpncmd に内蔵されています。しかし、HTML 版で綺麗なコマンドライン・リファレンスを見たいという方もいるでしょう。

その場合は、次の URL を参照して、PacketiX VPN の vpncmd のヘルプを参考にしてください: http://www.softether.co.jp/jp/vpn3/manual/web/6.aspx

別の Windows 端末上で GUI ツール (UT-VPN サーバー管理マネージャ) を用いた管理

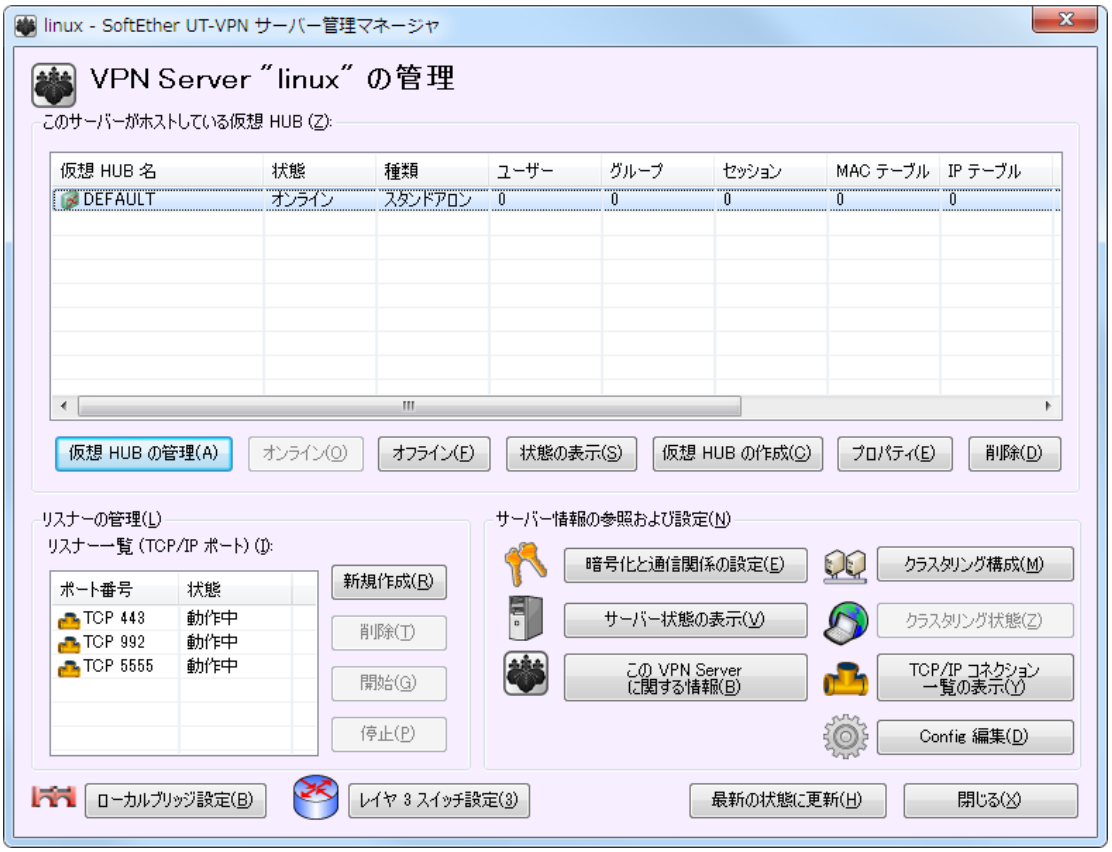
utvpncmd を用いると、すべての操作をコマンドラインで行うことができますが、とても疲れるかも知れません。そのような場合は、非常に洗練されて使いやすい GUI ツールである「UT-VPN サーバー管理マネージャ」を Windows 端末上で実行し、それを用いて UNIX 上で動作している UT-VPN Server サービスにリモート接続して管理することができます。

まず、UT-VPN の Web サイト「http://utvpn.tsukuba.ac.jp/」にアクセスし、UT-VPN Server をダウンロードしてください。UT-VPN Server のフル・インストール・バージョンでも構いませんが、「ExeOnly 版」という exe ファイルだけがパッケージングされたファイルをダウンロードすると、ZIP 形式になっており、展開すると vpnsmgr.exe を直接インストールすることなく起動できます。



UT-VPN サーバー管理マネージャが起動したら、「新しい接続設定」をクリックし、接続先のホスト

名として UNIX コンピュータの IP アドレスまたはホスト名を指定してください。そして UT-VPN Server に接続すると、とても簡単に UT-VPN Server を管理することができます。



6. UT-VPN Client の起動と設定方法

バックグラウンドサービス (デーモン) について

UT-VPN Client の本体プログラムは、バックグラウンドサービス (デーモン) として動作します。デーモンプログラムは、一度起動すると、起動したユーザーの画面 (コンソール) には一切文字出力を行わず、また、起動したユーザー (root) がログアウトしても動作し続けます。ユーザーが誰も UNIX にログインしていない状態でも動作し続けます。また、仮に何らかの原因でプログラムがクラッシュした場合、デーモンは自分自身を再起動して動作を続行します。

手動での起動方法

手動で UT-VPN Client サービスを起動するには、root 権限のユーザーとしてログインし、「utvpnclient start」と入力してください。

なお、make install を実行したインストール直後は、お使いのシェルのパスが更新されていない可能性がありますので、一度ログアウトして再度ログインするか、または、「bash」などと入力してシェルを新しく起動し、その上で実行してください。

```
[root@linux /root]# utvpnclient start
UT-VPN Client Service Started.
```

上記のように「start」引数を付けて utvpnclient を起動すると、デーモンプログラムとして動作が開始されます。その後は、ユーザー (root) がログアウトしても動作が継続されます。

OS の起動時に自動起動する方法

上記の方法で、手動で UT-VPN Client サービスを起動しても、OS が再起動するとサービスは停止してしまいます。理想的には、OS やコンピュータが停電などでいつ再起動しても、サービスが自動的に再開するのが望ましいということができます。

その場合は、OS のスタートアップスクリプトに「/usr/bin/utvpnclient start」を起動するように登録する必要があります。また、OS が終了しようとする際に「/usr/bin/utvpnclient stop」が起動されるように登録することが推奨されます。

OS のスタートアップスクリプトへの登録方法は、OS やディストリビューションによって大きく異なります。たとえば、Linux でもいろいろな方法がありますので、ここで詳しく書くことはできません。

参考情報として次の URL の PacketiX VPN Server の Linux へのインストール方法 (init.d への登録) もご参照ください: http://www.softether.co.jp/jp/vpn2/manual/web/7-3.aspx

utvpncmd (コマンドライン管理ユーティリティ) を用いた設定方法

UT-VPN Client サービスが動作している UNIX コンピュータのコンソール上で UT-VPN Client を設定・操作するためには、「utvpncmd」プログラムを使用します。utvpncmd は UT-VPN Server / UT-VPN Client を管理するための CUI (コマンドライン) のツールです。このツールはローカル (localhost) またはリモート上の UT-VPN Server / UT-VPN Client を管理することができます。

utvpncmd を起動する際は、一般ユーザー権限で OK です。

utvpncmd を普通に起動し、画面の指示に従って文字を入力すると、localhost で動作している UT-VPN Client に接続することができます。

```
[root@linux /root]# utvpncmd
utvpncmd コマンド - UT-VPN コマンドライン管理ユーティリティ
UT-VPN コマンドライン管理ユーティリティ (utvpncmd コマンド)
Version 1.00 Build 7092   (Japanese)
Compiled 2010/06/25 04:31:36 by yagi at pc25
Copyright (C) 2004-2010 SoftEther Corporation.
Copyright (C) 2004-2010 University of Tsukuba, Japan.
Copyright (C) 2003-2010 Daiyuu Nobori.
All Rights Reserved.

utvpncmd プログラムを使って以下のことができます。

1. VPN Server または VPN Bridge の管理
2. VPN Client の管理
3. VPN Tools コマンドの使用 (証明書作成や通信速度測定)

1 - 3 を選択: 2

接続先の VPN Client が動作しているコンピュータの IP アドレスまたはホスト名を指定してください。
何も入力せずに Enter を押すと、localhost (このコンピュータ) に接続します。
なお、このコマンドでは UT-VPN Client を管理できますが、PacketiX VPN Client は管理できませんのでご注意ください。
接続先のホスト名または IP アドレス:

VPN Client "localhost" に接続しました。

VPN Client>
```

上記のように「VPN Server>」というプロンプトが表示され、コマンド受付状態となった場合は、UT-VPN Server への接続は成功しています。

ここでコマンド名がわからない場合は「?」または「HELP」と入力することでコマンド一覧を表示することができます。また、コマンド名に続いて「--help」または「/?」と入力すると、そのコマンドの使い方を表示することができます。以下はその例です。

```
VPN Client>help
下記の 65 個のコマンドが使用できます:
 About                    - バージョン情報の表示
 AccountAnonymousSet      - 接続設定のユーザー認証の種類を匿名認証に設定
 AccountCertGet           - 接続設定に用いるクライアント証明書の取得
 AccountCertSet           - 接続設定のユーザー認証の種類をクライアント証明書認証に設定
 AccountCompressDisable   - 接続設定の通信時のデータ圧縮の無効化
 AccountCompressEnable    - 接続設定の通信時のデータ圧縮の有効化
```

```
 AccountConnect           - 接続設定を使用して VPN Server へ接続を開始
       :
       : (中略)
       :
 TrafficServer            - 通信スループット測定ツールサーバーの実行
 VersionGet               - VPN Client サービスのバージョン情報の取得
```

それぞれのコマンドの使用方法については、"コマンド名 /?" と入力するとヘルプが表示されます。
コマンドは正常に終了しました。

```
VPN Client>AccountCreate ?
AccountCreate コマンド - 新しい接続設定の作成
コマンド "AccountCreate" のヘルプ

[目的]
  新しい接続設定の作成

[説明]
  VPN Client に新しい接続設定を作成します。
  接続設定を作成するには、初期パラメータとして接続設定の名前と接続先のサーバー、および接続先の仮想 HUB、ユーザー名に加えて使用する仮想 LAN カード名を指定する必要があります。新しい接続設定を作成し　た場合、ユーザー認証の種類は [匿名認証] に初期設定され、プロキシサーバーの設定とサーバー証明書の検証オプションは設定されません。これらの設定やその他の詳細設定を変更するには、接続設定を作成した後　に、"Account" という名前で始まる他のコマンドを使用します。

[使用方法]
  AccountCreate      [name]      [/SERVER:hostname:port]      [/HUB:hubname]
[/USERNAME:username] [/NICNAME:nicname]

[パラメータ]
  name      - 作成する接続設定の名前を指定します。
  /SERVER   - [ホスト名:ポート番号] の形式で、接続先の VPN Server のホスト名と、ポート番号を指定します。IP アドレスで指定することもできます。
  /HUB      - 接続先の VPN Server 内の仮想 HUB を指定します。
  /USERNAME - 接続先の VPN Server に接続する際の、ユーザー認証で使用するユーザー名を指定します。
  /NICNAME  - 接続に使用する仮想 LAN カード名を指定します。
VPN Client>
```

このように、utvpncmd のコマンドライン・リファレンスは、utvpncmd に内蔵されています。しかし、HTML 版で綺麗なコマンドライン・リファレンスを見たいという方もいるでしょう。

その場合は、次の URL を参照して、PacketiX VPN の vpncmd のヘルプを参考にしてください:
http://www.softether.co.jp/jp/vpn3/manual/web/6.aspx

簡単な接続の例

UT-VPN Client を utvpncmd を用いて設定し、VPN 接続を行う例を以下に示します。

この例では、まず「VPN」という名前の仮想 LAN カードを UNIX カーネル上に作成します。次に「PUBLIC」という名前の接続設定を UT-VPN Client 上に定義（新規作成）します。接続先のサーバー名として「public.softether.com の 443 番ポート」を指定します。また、接続先の仮想 HUB として「PUBLIC」、ユーザー名として「public」を指定しています。

これは PacketiX.NET セキュアインターネット接続サービス（http://www.packetix.net/jp/secure/）に接続する例です。このサービスに接続するには、ユーザー名として「public」、認証方法は匿名認証で接続できます。もし、ユーザー名としてパスワード認証等を指定する必要がある場合は、「AccountPasswordSet」コマンド等を用いて認証データを接続設定に登録してください。

以下のように入力すると、接続設定「PUBLIC」を作成し、接続を開始し、最後に接続状況を表示することができます。

```
VPN Client>NicCreate VPN

VPN Client>AccountCreate PUBLIC /SERVER:public.softether.com:443 /HUB:PUBLIC
/USERNAME:public /NICNAME:VPN

VPN Client>AccountConnect PUBLIC

VPN Client>AccountStatusGet PUBLIC
AccountStatusGet コマンド - 接続設定の現在の状態の取得
項目                          |値
------------------------------+------------------------------------------------
接続設定名                    |PUBLIC
セッション接続状態            |接続完了（セッション確立済み）
      : (中略)
コマンドは正常に終了しました。
```

別の Windows 端末上で GUI ツール（UT-VPN クライアント接続マネージャ）を用いた管理

utvpncmd を用いると、すべての操作をコマンドラインで行うことができますが、とても疲れるかも知れません。そのような場合は、非常に洗練されて使いやすい GUI ツールである「UT-VPN クライアント接続マネージャ」を Windows 端末上で実行し、それを用いて UNIX 上で動作している UT-VPN Client サービスにリモート接続して管理することができます。

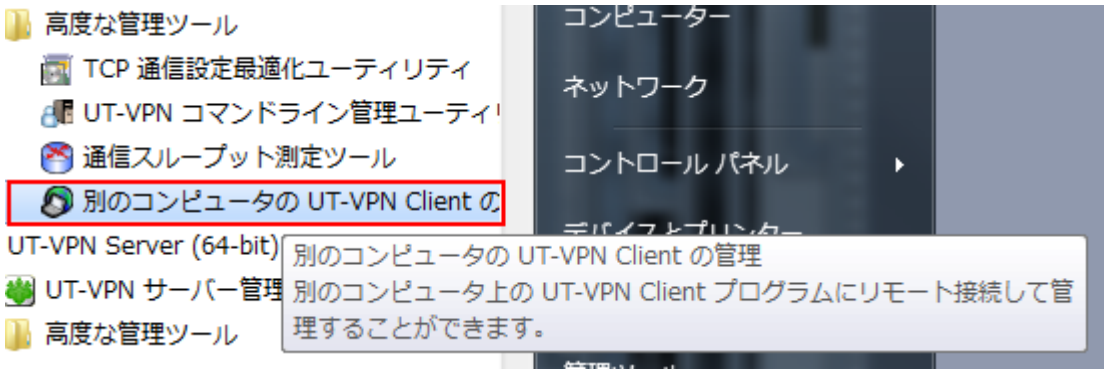
まず、対象の UNIX コンピュータ上の UT-VPN Client に utvpncmd で接続し、「RemoteEnable」コマンドを用いてリモート管理を有効にします。具体的には、次のようにします。

```
[root@linux /root]# utvpncmd /client localhost
VPN Client "localhost" に接続しました。

VPN Client>RemoteEnable
```

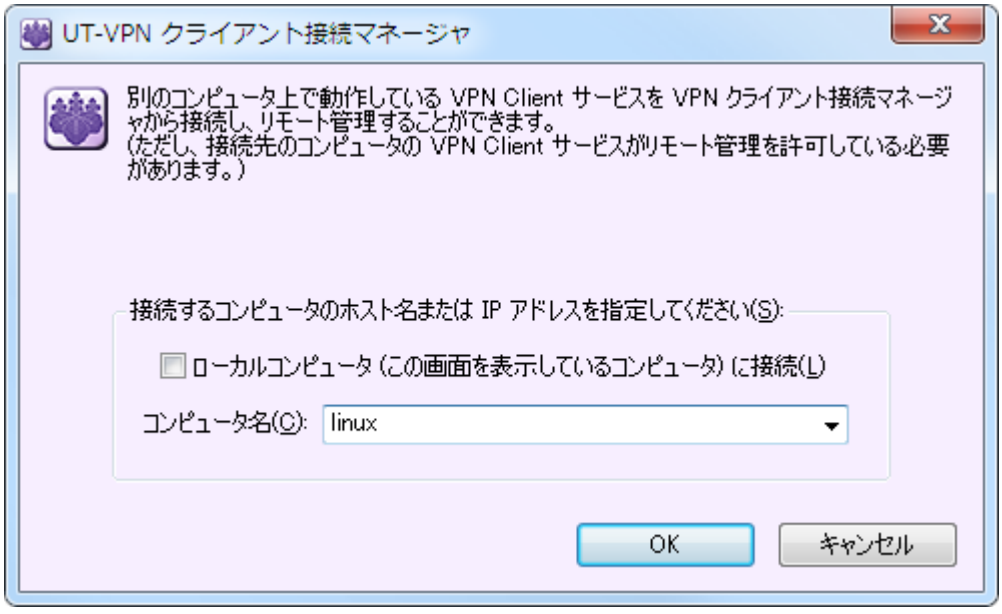
```
RemoteEnable コマンド - VPN Client サービスのリモート管理の許可
コマンドは正常に終了しました。

VPN Client>
```

次に、近くにある他のコンピュータ（Windows 端末）から、UT-VPN の Web サイト「http://utvpn.tsukuba.ac.jp/」にアクセスし、UT-VPN Client をダウンロードしてインストールしてください。

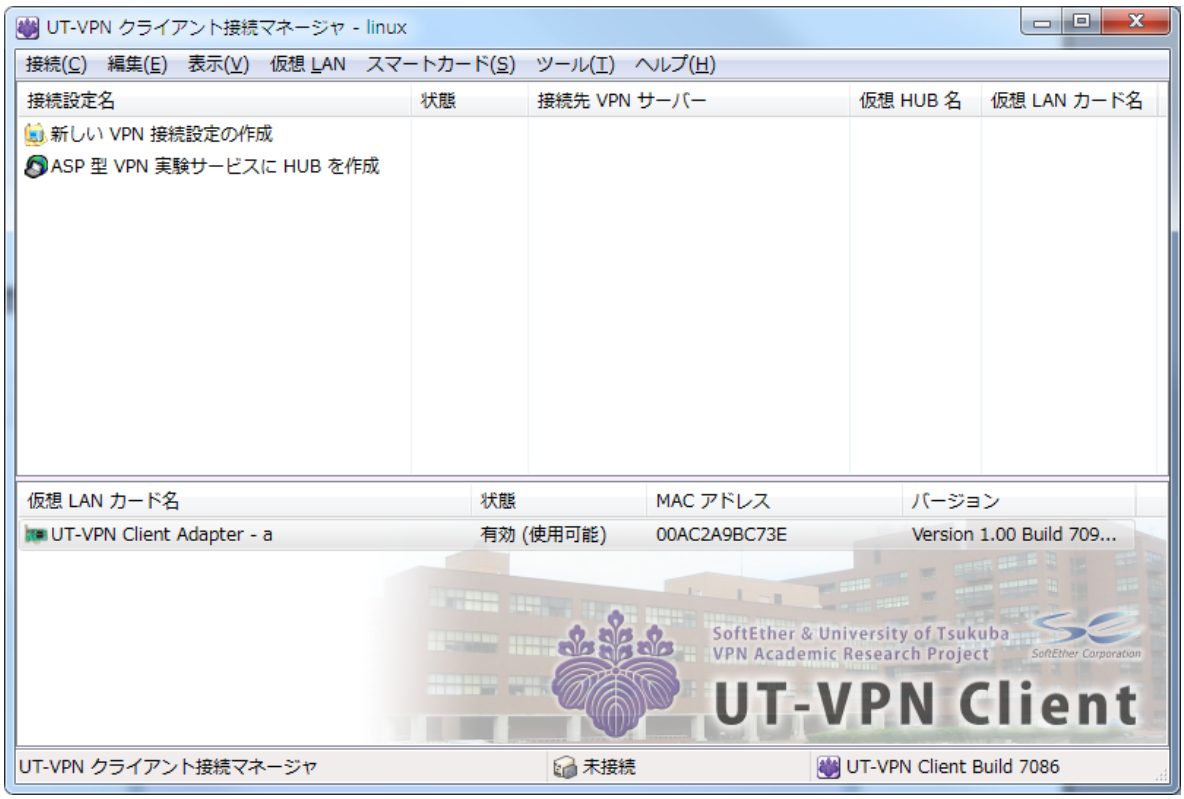
するとスタートメニューの「UT-VPN Client」に以下のように「別のコンピュータの UT-VPN Client の管理」が生成されますので、それを起動します。



以下の画面のように接続先の UT-VPN Client のホスト名を指定する画面が表示されますので、IP アドレスまたはホスト名を指定します。



接続に成功すると、以下のように、Windows 上から UNIX 上の VPN Client を管理することができるようになります。



注意事項

現在のところ、UT-VPN Client の UNIX 版は、Linux でのみ動作します。また、Tap デバイスドライバを用いて仮想 LAN カードを作成しますので、tun/tap がインストールされていない環境では動作しません。

仮想 LAN カードを作成すると、tap デバイスが Linux カーネル上に新しく作成されます。具体的には、「ifconfig -a」コマンド等で確認してみてください。

Windows と比べて、多くの Linux システムでは、tap デバイスが VPN を経由してどこか別の VPN Server に接続した場合でも、IP アドレスは DHCP によって自動取得されません。IP アドレスやルーティングテーブル等は、ifconfig コマンドや route コマンドを用いて手動設定する必要があります。または、DHCP Client を使用してください。これらの設定・処理は、普通の他の物理的な LAN カードに対する設定・処理と同様です。ただし、仮想 LAN カード (tap) に対して設定したネットワーク設定が、物理的な VPN Server との間の通信を妨害することがないようによく注意してください。たとえば、デフォルトゲートウェイを VPN Server の先にあるルータに指定する場合は、スタティックルートとして、VPN Server に対するネットマスク 255.255.255.255 宛のルーティングテーブルを記述する必要があるかも知れません。

これらの処理はいろいろと手動で行うのは面倒ですので、シェルスクリプト等を書いて自動実行するのが望ましいと思われます。たとえば、OS の起動時にこれらの処理を自動で行うようなスクリプトを書けば、常に VPN Client 経由でどこかのネットワークに接続しているコンピュータを作ることができます。