PuTTY for Symbian OS User's Guide

Table of Contents

Legal Notice

PuTTY for Symbian OS is free software, and comes with no warranty. Some PuTTY distribution packages are cryptographically signed. The signatures do not indicate any additional warranties or guarantees, they simply act as further proof that the packages originate from their original authors.

1 Introduction

PuTTY for Symbian OS is an SSH client and terminal emulator for your mobile phone. It lets you connect to to SSH servers, typically Unix or Linux machines, over the network. It can be an invaluable remote administration tool for system administrators, and is also popular for accessing IRC clients running permanently on remote servers.

PuTTY for Symbian OS is a port of the Windows and Macintosh PuTTY SSH client to Symbian OS based mobile phones. Current releases support all S60 3rd edition phones, including popular Nokia models such as the N95 and the E90, and the older Series 80 v2 platform, supporting the Nokia 9500, 9300, and 9300i communicator products. Earlier releases also supported older S60 platform versions and the 9200 Communicator series, but support for those models has been discontinued. Additionally, there are third-party ports to the UIQ platform. The latest releases are available at the PuTTY for Symbian OS WWW site [S2PuTTY08].

This document is a brief user's guide for PuTTY for Symbian OS. It describes PuTTY installation, basic usage, configuration, and public key authentication support. In addition, the last chapter lists some troubleshooting hints in case PuTTY does not work.

The document assumes that you are familiar with SSH concepts and have used SSH on other platforms before. In addition, you should be familiar with the Symbian OS phone you use, including using the system applications and installing new ones.

There is plenty of documentation on SSH protocols and applications available on the Internet. You should also check with your server's system administration for any local guides you might have. You can also check Barrett's and Silverman's book [BaSi01].

2 Installating PuTTY

This chapter documents the PuTTY installation procedure. It follows standard Symbian conventions, so readers familiar with Symbian OS application installation can skip to the next chapter.

Important security note: As an SSH client, PuTTY is security-critical software. To ensure that the copy you are installing has not been tampered with, always check the package signatures before proceeding. For the Series 80 version, the installation .SIS packages are signed themselves, so after installing the certificates on your device, the system will check the signatures automatically. For S60, use PGP or GnuPG to check the PGP signatures before installing.

2.1 Series 80

PuTTY installation packages for Series 80 are signed with a self-signed certificate. To be able to verify the packages, you'll need to install the certificate to the device. The steps needed are:

1. Fetch the certificate from http://www.s2.org/~pekangas/petteri_s80_2009_der.zipand unzip it.
2. Verify the certificate. Its MD5 sum is `9559ec393f3fecb0c34ababfc0f9727f`. A PGP signature is available at http://www.s2.org/~pekangas/petteri_s80_2009_der.cer.asc, the key is http://www.s2.org/~pekangas/petteri_pgp_2008.asc. The key is also available on OpenPGP key servers, ID E393AD7C.
3. Copy the certificate to a file in the communicator.
4. Open Control panel, select the "Security" group, and from there open "Certificate manager".
5. Change to the "Other" tab and find the new certificate, named "Petteri Kangaslampi" from the list.
6. Select "View details", select "Trust settings" and enable "Application installation"

After installing the certificate, install the .SIS package normally. The S80 versions are distributed in files named `putty_s80v2_`*`version`*`.sis`. Download the version you want, unzip the file, and transfer the .SIS package from the archive to your Communicator. You can then install the package by opening it from Messaging or using the File Manager. After installation, you can start PuTTY from the device desktop.

Note that PuTTY no longer supports the earlier 9200 Communicator series. The last PuTTY release to support those was version 1.4 beta 1, which is still available from the PuTTY web site [S2PuTTY08].

2.2 S60

PuTTY S60 3rd edition installation packages are self-signed. Many S60 devices, including all Nokia E-series phones, refuse to install self-signed applications by default. To enable this, go to the device main menu, select `Tools` and start `Application Manager`. From the application manager press `Options`, select `Settings`, and set `Software installation` to `All`. The names will be different in devices using a different language but the same setting should be present. This is a mandatory operation, otherwise PuTTY will not install, and the installer will complain about a certificate error!

Unfortunately there is no way to add new trusted application signing keys to an S60 device, so you will need to use PGP signatures to verify their authentity. All installation packages have a separate PGP signature. The key is available at http://www.s2.org/~pekangas/petteri_pgp_2008.asc and on OpenPGP key servers, ID E393AD7C.

S60 releases are distributed in files named `putty_s60v3_`*`version`*`.zip`. Download the version you want, verify its PGP signature, unzip the file, and transfer the .SISX package from the archive to your phone. You can then install the package by opening it from the e.g. the Messages Inbox or using the file manager.

Warning! After you have changed settings to install PuTTY, you have enabled installation for all self-signed applications, including possible malware. Be careful careful what you install and don't

accept installation packages from an unknown source. The installer will also list the capabilities the application needs, and you should check that those are reasonable. For example, it's natural that PuTTY needs network access, but if a simple game requires it something may be wrong.

3 Using PuTTY

3.1 First Run on Series 80: Random Number Generation

A good source of random numbers is very important for security for cryptographic applications. To ensure its random number generator is initialized well, when you run PuTTY for the first time on a Series 80 communicator, it will record noise from the phone microphone to start the random number generator. You can place the device close to a noise source, but even normal background noise has more than enough randomness.

Afterwards, PuTTY will continue to generate more randomness from keypresses and other events as it is used, but you can repeat the initialization process by selecting `Initialize random number generator` from the menu.

S60 has a system-wide pool of random number data, so on S60 PuTTY does not use the microphone.

3.2 Profile List

When you start PuTTY, it first displays a set of connection profiles. Each profile represents a group of settings, including the server to connect to and the user name to use. Profiles are equivalent to saved sessions in PuTTY for Windows and also replace explicitly managed setting files from previous versions.

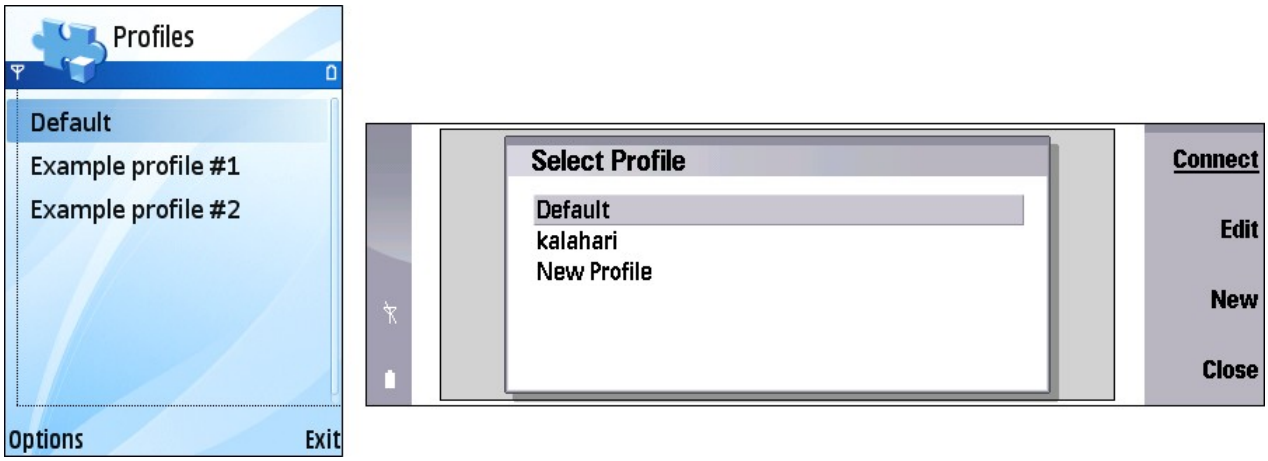

Figure 1: S60 and Series 80 Profile Lists

Figure 1 shows the S60 and Series 80 profile lists. Apart from typical platform user interface differences the lists are similar, and the functions available are the same. On S60 the different selections are available in the `Options` menu, while on Series 80 you can use the four command buttons.

Select `Connect`, press the selector, or press Enter to connect to the selected profile. Select `Edit` to edit the profile's settings or change its name. To create a new profile, select `New` or `New Profile`, and to delete one select `Delete` on S60 or select `Edit` followed by `Delete` on Series 80. You can exit PuTTY by selecting `Exit` or `Close`.

The first time you start PuTTY it creates a default profile based on default settings or if you have used previous PuTTY versions on the same phone, your earlier default settings will be used instead. On Series 80 the default profile cannot be renamed or deleted.

Editing a profile opens a multi-paged settings view or window, depending on the platform. See chapter 4 for a description of the different settings that are available.

3.3 Connecting to a Server

After you select a profile, PuTTY will start connecting to the server. If you have not set the server host name in the profile, PuTTY will now prompt you for it.

First PuTTY will connect to the network and will shortly prompt you to select the access point to use. PuTTY requires full internet access, so if your mobile operator has multiple access points try to find one titled "Internet", not "MMS" or "Streaming". In addition to mobile phone networks, you can also use WiFi access points if your phone has WiFi support and there is a suitable network around.

After the network connection is established, PuTTY will connect to the SSH server. This may take a little while, especially when using the SSH 2 protocol on an older phone, so be patient. Some common error messages you may encounter are listed in Table 1.

Host name lookup failed	The server you tried to connect to cannot be found. Check that you typed the host name correctly, and that you are not using a WAP-only access point.
Socket connect failed: Could not connect	PuTTY could not connect to the SSH server. This error typically occurs when an SSH server is not running in the target host, or it is running in a different port. Check that the host name is correct and verify that the server is not running in a non-standard port.
Socket connect failed: Timed out	PuTTY could not connect to the SSH server. This error typically occurs when trying to use a WAP or MMS only access point.

Table 1 Common connection errors

When connecting to a server for the first time, PuTTY will prompt you to verify and accept the server host key. **It is important to verify the key is correct**, otherwise the connection may not be secure.

After the server identity has been verified, PuTTY will prompt you for your username, unless it is set in the profile, and password. After this is done, the SSH connection will be open and you can start using the server. Instead of a password, you can also use public key authentication, see chapter 5.

3.4 Using the Terminal

Once the connection is open, PuTTY shows the terminal window, which is no different from any other SSH client. On both platforms you can simply start typing commands into the terminal and see the results on the screen. Beyond this, however, the Series 80 and S60 versions are rather different, and will be covered in two different subsections below.

3.4.1 Series 80

Since Series 80 devices have a full keyboard, using PuTTY is straightforward. All keys on the keyboard send the characters printed on the labels, and the `Ctrl` key works directly. Some special characters not available on the keyboard are available in the `Tools` menu, and the `Chr` key opens the system character selection window. Note that the vertical bar symbol available in the character selection window is not the same as the Unix pipe character. To send the pipe, press `Shift+Ctrl +P` or select `Tools/Send Character/Pipe` from the menu.

The Series 80 version supports two different fonts, small and large, and an optional full-screen mode. These can be changed from the `View` menu. In most cases, using large font and full-screen mode gives the best user experience. The default is small font and partial screen mode, as this results in a standard 80x24 character terminal window. The other combinations are listed in Table 2.

Font	*Display*	*Terminal*
Small	Partial screen	80x24
Small	Full-screen	106x25
Large	Partial screen	74x14
Large	Full-screen	91x14

Table 2 Communicator display modes and font sizes

3.4.2 S60

On S60 devices you can also simply start writing to the terminal window. However, most S60 phones do not have a full QWERTY keyboard, and since PuTTY does not support predictive text in the terminal this can be cumbersome. Additionally, even S60 phones with a full keyboard such as the E61 and E90 still are missing a number of important keys such as Esc and Tab.

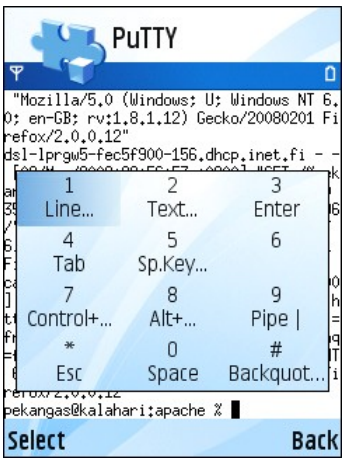

Figure 2: S60 Send Grid

To make text entry easier, and to support various special characters, PuTTY has a special Send Grid user interface control on S60. To activate the grid, select `Send` from the `Options` menu, or press the right soft key. The grid replaces the `Send` menu on earlier PuTTY versions, and supports sending special characters, various control key combinations, and plain text.

To send a line of text, followed by Enter, select `Line`. `Text` sends plain text without Enter, while other selections send either a key or a key combination directly, or open a sub-grid. The grid layout matches a typical phone keyboard, and each grid item also has an associated shortcut, shown above the item text. For example, to send an Escape character, press the asterisk (`*`), while the fastest way to send Ctrl+D is `Send-7-2`. Learning these shortcuts will make using PuTTY much faster, especially on non-QWERTY phones.

The green call ("Send") key repeats the most recent command from the Send Grid. By default it sends an Escape character, making it especially useful for E90 and E61 users. Additionally, the phone four-way selector or joystick sends cursor keys when moved and Enter when pressed.

The S60 version supports a number of different fonts that can be selected from the `View` menu. You can also set PuTTY to full screen by selecting `Toggle full screen` from the same menu.

3.4.3 Clipboard support

Both Series 80 and S60 versions support copy/paste to/from the system clipboard. To copy text to the clipboard, select `Select` from the `Edit` menu, use `Mark` to mark the beginning of the selection and `Copy` to copy it to clipboard. On Series 80 you can also use the command buttons, and on both platforms the selector can be used to mark and copy when selecting text. To paste text from the clipboard, simply select `Edit/Paste`.

4 Profile Settings

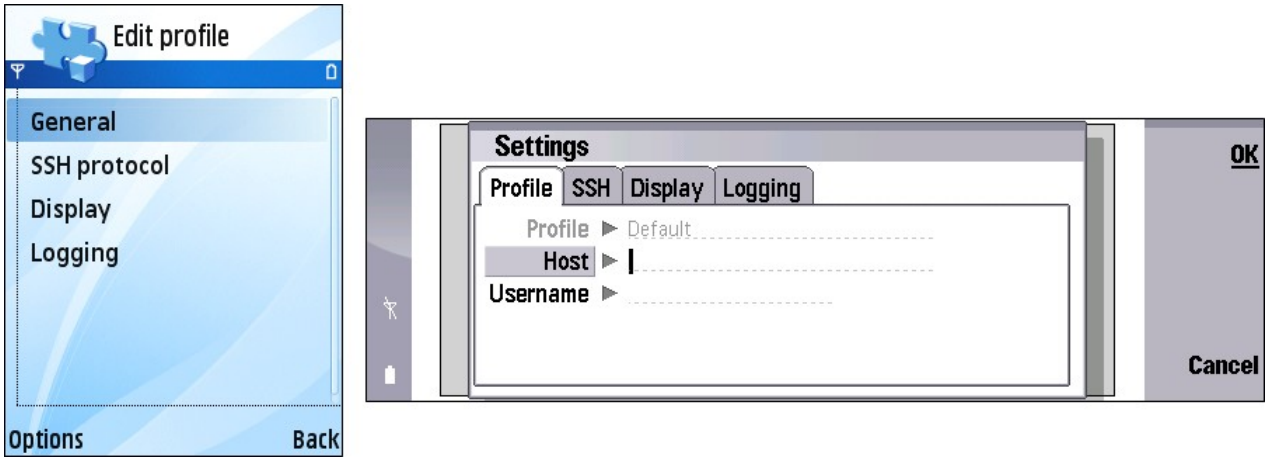

Figure 3: Settings view and window

When you edit a profile, PuTTY opens a settings view or window. The settings are divided into four groups, as shown in Figure 3. The S60 version has a settings view with four different pages, while S80 uses a dialog with four tabs. The settings however are the same for both versions and documented in this chapter.

4.1 General Settings

Profile name: The profile name as shown in the profile list. On Series 80 the initial default profile name cannot be changed.

Host: The host name or IP address for the SSH server to connect to. If you set the server host name here, PuTTY will not prompt for it later.

Username: The user name to use for logging in to the server. If you set the user name here, PuTTY will not prompt for it later.

4.2 SSH Settings

Port: The TCP port to use. Note that PuTTY always uses the SSH protocol, regardless of the port number. Changing the port is rarely useful unless you are running the SSH server in a non-standard port.

SSH Version: The SSH protocol version to use. SSH 2 is recommended.

Private key: Private key file for public key authentication. See chapter 5 for more information on public key authentication.

Compression: Controls whether SSH protocol compression is on. Compression reduces the amount of network traffic needed and makes PuTTY work faster, so it is enabled by default. Some embedded SSH servers, such as ones used in network routers, do not support compression however, so you can use this setting to disable it.

Preferred cipher: PuTTY supports two different ciphers: 128-bit Blowfish and and 256-bit AES. Blowfish is a faster algorithm and uses a shorter key length, making it more efficient, and is used by default. AES may be more secure.

Keepalive interval: If this value is not zero, it sets the number of seconds between sending SSH "Ignore" packets to the server. This can help with keeping the connection open on some networks.

4.3 Display Settings

Font: Controls the default font to use. The Series 80 version has two system fonts available, while the S60 supports a larger number of PuTTY-specific fonts, and more can be installed afterwards. The font can also be changed from the terminal view after the connection is open.

Full screen: Sets whether the terminal is full screen by default. The full screen setting can also be changed from the terminal view after the connection is open.

Palette: Controls the color palette to use for the terminal. The default palette is black text on white background; the alternatives are gray or white text on black background. The palette can also be changed from the terminal view after the connection is open.

Character set: The character set used for terminal input and output. The default is ISO-8859-15, which should work for most western users. Alternatively, UTF-8 may work better with some Linux servers.

4.4 Logging

Log type: Logging type. In normal use the type should be set to `No logging`. For troubleshooting and debugging information set the type to `SSH data & debug`. Note however that this logging setting will also log your password!

Log file: The log file. The logging information defined above is written to this file. Logging is mainly used for development and debugging, but it can be useful for troubleshooting any connection problems. The log file may contain sensitive information, such as passwords.

5 Public-key Authentication

Like all modern SSH clients, PuTTY for Symbian OS supports public-key authentication in addition to basic passwords. This chapter describes how to create suitable key pairs, and how to configure PuTTY for public-key authentication. This document does not discuss SSH server configuration, so you should make sure you can set up public-key authentication with the server you use before attempting to use it with a mobile phone. Note that OpenSSH and SSH Tectia Server from SSH Communications Security use a different configuration syntax. See [SSH04] and [OpenSSH04] for more information on server configuration.

5.1 Creating Keys

PuTTY for Symbian OS can only use key files created with PuTTYgen on a Windows PC. PuTTYgen ships with PuTTY for Windows, and is available for download at the PuTTY web page [PuTTY04]. If you do not have it installed, download the latest version, verify its authentity, and install it.

To create a key pair, start PuTTYgen, and configure it as follows:

Type of key to generate: **SSH2 RSA**
Number of bits in a generated key: **1024**

Select `Generate` to create the key and move your mouse cursor around the window to generate random numbers for the generator.

After the key has been created, PuTTYGen will prompt you to set a key comment and optionally a passphrase. The key comment should be a short description for the key, such as `Joe@E61`. Setting a passphrase for the key will make it more secure, since the key cannot be used without knowing the passphrase, but entering long passphrases can be inconvenient on a mobile phone. If you are confident you can keep your phone secure, using a public key without a passphrase can make using PuTTY much easier.

After setting the comment and passphrase, use `Save public key` and `Save private key` to save the public and private parts of the keypair to files. Transfer the private key file to your phone, using PC Suite, a memory card reader, or some other mechanism, and note the directory where it is stored. Transfer the public key to your SSH server, and configure the server to accept connections with that key.

It is a good idea to configure PuTTY for Windows to use the same private key, and verify that you can connect using the key, before attempting to use the key from a mobile phone.

5.2 Configuring PuTTY

If you created your SSH keys following the instructions in section 5.1, and have configured your SSH server correctly, configuring PuTTY to use the keys for authentication is simple. Simply start PuTTY, select the profile you wish to use, and select `Edit` to edit it. On the `SSH` settings page or tab, edit `Private key` and select the private key file you just created. When you now connect to a server, PuTTY will first attempt to authenticate using the key. If you still get prompted for a password, public key authentication failed for some reason.

5.3 Troubleshooting

SSH public key authentication can fail for several different reasons. The most common problems are incorrect key file formats, server configuration problems, and protocol mismatches. This section lists some useful troubleshooting tips.

File format problems: PuTTY only works reliably with private key files created with PuTTYgen. If you wish to use an existing key created with other tools, you can try to convert the key to PuTTY format using PuTTYgen on Windows.

Server configuration: Ensure that you have configured the server correctly, and check that you can connect from a PC using the same key. Note that OpenSSH and SSH Tectia Server from SSH Communications Security use a different configuration syntax.

Protocol mismatches: SSH protocol versions 1 and 2 use different keys. If you created a SSH 2 key, as instructed in section 5.1, make sure that the preferred SSH protocol version is set to `SSH 2`, and that the server supports SSH 2.

In general, the first thing to do when public key authentication fails is to enable logging from PuTTY. Set a log file, set logging type to `SSH data & debug`, try to connect, and see what information gets written to the log file. The log files are plain text files, and can be viewed with Notepad or any other text editor on a PC. A couple of hints:

- If the log file doesn' t contain the line `Trying public key "key_file_name"`, PuTTY didn't even attempt to use public-key authentication, and something is most likely wrong with your key.
- If you see a message "`Server refused our public key`" on the user interface, or a packet of type 15 (`SSH1_SMSG_FAILURE`) in the log as a response to the private key, the server did not accept the key, and most likely the server configuration is wrong.

6 Troubleshooting

In general PuTTY tends to either work without problems or not work at all with a given configuration. However, there are some things that may help, and this section lists a few tricks to try. The most up-to-date version of these troubleshooting tips are available on the PuTTY for Symbian OS WWW pages [S2PuTTY08].

Unless separately noted these tips apply to all PuTTY for Symbian OS versions.

6.1 General Problems

6.1.1 Create a Fresh Profile

It is possible, although unlikely, that the PuTTY profile you are trying to use can get corrupted. To work around this, delete the profile if you can, and create fresh new one. All new profiles will get the default settings and should work without problems.

If this helps, and you can figure out what you did when the configuration file got corrupted, please file a new bug report with instructions on how to reproduce the problem..

6.2 Network problems

6.2.1 Try a Different Access Point

If PuTTY starts and a network connection is set up, but the SSH connection does not open (no username or password prompt), there may be a problem in the access point or network. Try using a different access point, and ensure the access point you are trying to use supports full internet connections. If your mobile operator has multiple access points try to find one titled "Internet", not "MMS" or "Streaming"

6.2.2 Open the Browser First

If you are unsure about the network connection, try opening the web browser first and browsing to a known web site. This is especially useful with WiFi networks, since most WiFi networks that require payment or authentication will require you to open the connection first with a web browser. Once the network connection is open, and you have successfully opened the web page, try using PuTTY again and select the same network or access point as the web browser uses.

6.2.3 Try Connecting from a Computer

If the SSH connection still fails, try using the phone as a modem and connecting from a PC using the same access point. If that fails, the problem is most likely that the access point does not let SSH traffic through, and there is nothing PuTTY can do about it.

6.2.4 Enable logging

PuTTY for Symbian OS supports logging to a file. The log can contain useful information on the connection, and help you determine what the problem is. It is also useful to attach a log file to bug reports, but please check first that the log does not contain secrets such as passwords. See chapter 4 for information on logging settings.

See chapter 4 for more details on using PuTTY settings.

The log files are regular text files, and can be viewed with Notepad or other text editor. Most file manager applications can even view them on the phone itself.

References

[BaSi01] Barrett, Daniel J.; Silverman, Richard: *SSH, The Secure Shell: The Definitive Guide*. O'Reilly & Associates, 2001.

[OpenSSH04] The OpenSSH Project: *OpenSSH Manual Pages*. The OpenSSH Project, 2004. http://www.openssh.com/manual.html

[PuTTY04] Simon Tatham: *PuTTY WWW Pages*. Simon Tatham, 2004. http://www.chiark.greenend.org.uk/~sgtatham/putty/

[S2PuTTY08] Kangaslampi, Petteri: *PuTTY for Symbian OS WWW Pages*. Kangaslampi, Petteri, . http://s2putty.sourceforge.net/

[SSH04] : *SSH Tectia Server (Unix) 4.0 Documentation*. SSH Communications Security, 2004. http://www.ssh.com/support/documentation/all/server-unix/4.0/