SSH (Secure Shell) Server and Client

Document revision 2.1 (July 5, 2007, 12:16 GMT)
This document applies to V3.0

Table of Contents

General Information

Summary

SSH Client authenticates server and encrypts traffic between the client and server. You can use SSH just the same way as telnet - you run the client, tell it where you want to connect to, give your username and password, and everything is the same after that. After that you won't be able to tell that you're using SSH. The SSH feature can be used with various SSH Telnet clients to securely connect to and administrate the router. Apart form regular password-based authentication, preshared key file may be used to authenticate a user.

The MikroTik RouterOS supports:

- SSH 1.3, 1.5, and 2.0 protocol standards
- server functions for secure administration of the router
- telnet session termination with 40 bit RSA SSH encryption is supported
- secure ftp is supported
- preshared DAS key authentication

The MikroTik RouterOS has been tested with the following SSH telnet terminals:

- MikroTik RouterOS embedded SSH client
- PuTTY

- Secure CRT
- OpenSSH GNU/Linux client

Specifications

Packages required: ***security***
License required: ***level1***
Home menu level: ***/system ssh***
Standards and Technologies: ***SSH***
Hardware usage: ***Not significant***

Additional Documents

- http://www.freessh.org/

SSH Server

Home menu level: ***/ip service***

Description

SSH Server is already up and running after MikroTik router installation. The default port of the service is 22. You can set a different port number or disable the service if you do not need it. See the **System Services** manual for the detailed instructions.

SSH Client

Command name: ***/system ssh***

Property Description

port (*integer*; default: **22**) - which TCP port to use for SSH connection to a remote host
user (*text*; default: **admin**) - username for the SSH login

Example

```
[admin@MikroTik] > /system ssh 192.168.0.1 user=admin
admin@192.168.0.1's password:

  MMM      MMM       KKK                          TTTTTTTTTTT      KKK
  MMMM    MMMM       KKK                          TTTTTTTTTTT      KKK
  MMM MMMM MMM  III  KKK  KKK  RRRRRR     OOOOOO      TTT     III  KKK  KKK
  MMM  MM  MMM  III  KKKKK     RRR  RRR  OOO  OOO     TTT     III  KKKKK
  MMM      MMM  III  KKK KKK   RRRRRR    OOO  OOO     TTT     III  KKK KKK
  MMM      MMM  III  KKK  KKK  RRR  RRR   OOOOOO      TTT     III  KKK  KKK

  MikroTik RouterOS 3.0beta10 (c) 1999-2007       http://www.mikrotik.com/

Terminal xterm detected, using multiline input mode
[admin@MikroTik] >
```

SSH Preshated Key

Home menu level: ***/user ssh-keys***

Description

You can use DSA keys (only DSA keys are supported) instead of password to log into the router. This method may be preferred for automated systems that congifure router(s) with SSH protocol using RouterOS console language. It is also useful if you just don't like remembering dozens of passwords and entering them to the login prompt all the time.

Property Description

key-owner (*read-only: text*) - remote user, as specifie in key file

user (*name*) - local user to associate the key with

Command Description

import - import a DSA key file (*name*) - filename to import the SSH key from (*name*) - local user to associate the key with

Notes

Only openssh DSA keys are supported. If you use puttygen, convert generated keys to right type.

Example

Generating the DSA key on a UNIX machine:

```
sh$ ssh-keygen -t dsa -f ./id_dsa
Generating public/private dsa key pair.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in ./id_dsa.
Your public key has been saved in ./id_dsa.pub.
The key fingerprint is:
91:d7:08:be:b6:a1:67:5e:81:02:cb:4d:47:d6:a0:3b admin-ssh@beka
```

Now, after you upload the key file onto the router, you can import it:

```
[admin@MikroTik] user ssh-keys> import file=id_dsa.pub user=admin-ssh
[admin@MikroTik] user ssh-keys> print
 # USER                KEY-OWNER
 0 admin-ssh           admin-ssh@beka
[admin@MikroTik] user ssh-keys>
```