NAME

smsync — simple mtree sync client

SYNOPSIS

smsync [**-h**] [**-f** *specfile*] [**-p** *pskfile*] *host:[port:]path*

DESCRIPTION

The **smsync** program reads an *mtree* specification and synchronizes this file system hierarchy with the one rooted at the given *path* on the remote system *host* by communicating with an smsyncd(8) server.

If *port* is not specified on the command-line **smsync** tries to connect to port 4242 on *host*.

OPTIONS

The following options are available:

-h Print the online help and exit successfully.

-f *specfile*
This option causes **smsync** to read the specification from the given file rather than from stdin.

-p *pskfile*
This option causes **smsync** to read the pre-shared secret from the given file. If **-p** is not passed, **smsync** will prompt the user for the secret upon startup.

NOTES

smsync expects the input to be in the format generated by *mtree* **-C -kall**. It builds a filelist and sends the input to the remote side. After that, it will read a list of indeces from the remote side indicating which files need to be sent to the server. For each file in the received list, it will send the entire file to the server.

The client and server communicate with each other via the smsync(4) protocol. All communication is encrypted using a one-time session key generated after successful authorization using the pre-shared key.

EXIT STATUS

The **smsync** utility exits 0 on success, and >0 if an error occurs.

SEE ALSO

smsync(4), smsyncd(8), mtree(1)

HISTORY

The **smsync** program was first assigned as the final project for the class CS765 “Advanced Programming in the UNIX Environment” during the fall semester of 2005 at Stevens Institute of Technology.

CAVEATS

The client the and the server need to have a pre-shared secret in order for the authentication and encryption to work.

BUGS

Well, let’s see...