

Secure FTP on VMS

The secure file transfer protocol (FTP) is implemented as SCP2 (Secure CoPy) on the VMS platform.

SIMPLE INTERACTIVE TRANSFERS

Unlike FTP, which provides a command prompt, SCP2 uses command lines. The basic syntax is:

SCP2 <flags> <source-file> “username@<destination-host>”::<<destination-file>

The quotes (“) and colons (::) are significant.

(Note that the <source file> and <destination file> arguments in this document refer to the file that you are copying *FROM* and the file you are copying *TO*, respectively.)

For example, using FTP to send file *test.txt* on donald.uoregon.edu to daisy.uoregon.edu:

```
ftp
FTP>open daisy.uoregon.edu
Username: user
Password: *****
FTP> put test.txt
FTP> quit
```

Doing the same, using SCP2:

SCP2 /VMS test.txt “user@daisy”::

The server will prompt the user for a password, and then transfer the file. Note that the /VMS flag is important because daisy and cedar both use the VMS operating system.

The destination file name does not have to be the same as the source file. To send *test.txt* as *new_test.txt* to daisy.uoregon.edu:

SCP2 /VMS test.txt “user@daisy”::new_test.txt

2) You must specify the full URL name for off-campus servers. For instance, to transfer a file to the OUS’s Cedar server:

SCP2 /VMS test.txt “user@cedar.ous.edu”::

3) If the username on the destination server is the same as the one you’re sending from, you can omit the username:

SCP2 /VMS test.txt “cedar.ous.edu”::

Additional switches that you may use with SCP2 commands:

<code>/DIRECTORY</code>	Force the target to be a directory
<code>/HELP</code>	Display a help page on how to use file, including these switches
<code>/PRESERVE</code>	Preserve file attributes and timestamps
<code>/NOPROGRESS</code>	Suppress display of progress indicator
<code>/QUIET</code>	Suppress display of warning messages
<code>/RECURSIVE</code>	Process entire directory tree. (sends everything in a directory)
<code>/REMOVE</code>	Remove (delete) source files after copying
<code>/TRANSLATE_VMS=(ALL, NONE, VARIABLE, FIXED, VFC)</code>	(default=ALL) Select the VMS text files to be translated; <code>/VMS</code> and (Used when communicating between a VMS and non-VMS server) <code>TRANSLATE_VMS</code> are mutually exclusive
<code>/VERBOSE</code>	Display a debugging messages in detail
<code>/VERSION</code>	Display version number only
<code>/VMS</code>	Negotiate ability to transfer VMS file information; (Used when communicating between two VMS servers) <code>/VMS</code> and <code>/TRANSLATE_VMS</code> are mutually exclusive

BATCH TRANSFERS

In order to send a file non-interactively (meaning, you don't need to be present), a process called public key authentication is used. This requires a private "key" (for yourself) and a public "key" (for the destination server). The private key is a special code that is used to encrypt transfer information, and the public key is another code that is used to understand that information. There are four steps to a batch session:

1. create public and private keys for the sender
2. create/edit files telling SCP2 which keys to use
3. copy the public key to the destination server
4. use the proper command text in the batch script

The VMS servers (Donald, Daisy, Oregon) require a syntax that is slightly different than the Unix servers (Darkwing, Gladstone) for both sending and receiving files. For clarification, we have provided a separate section for each.

Public Key Authentication: VMS to VMS

Step 1: Generate a private key and a public key

```
mu sshkeygen/ssh2
```

You will be asked for a passphrase and confirmation. Disregard by hitting the enter key at the prompt (submitting the passphrase make batch mode transfers impossible, and the goal is to encrypt transmission, not to add an additional layer of login security). Sshkeygen will create an [.SSH2] directory off your home dir, and create two files:

```
id_dsa_1024_a.          (your private key)  
id_dsa_1024_a.pub.      (your public key)
```

Step 2: Create the identification and authorization files

Both you and the server you are sending to need to know which file to look at when comparing files. On your side, the identification file says which file should be used for your private key. Create a file named IDENTIFICATION in your SSH2 subdirectory with the following line in it:

```
idkey id_dsa_1024_a
```

Similarly, the authorization file tells the server which file has the public key. Create a file named AUTHORIZATION in your SSH2 subdirectory with the following line:

```
key id_dsa_1024_a.pub
```

Step 3: Copy the authorization file and the public key to destination server

Both the authorization file and the public key file (id_dsa_1024_a.pub) need to be copied to the destination server in order for SCP2 to work. You can transfer them using SCP2 in interactive mode with the following line:

```
mu scp2/vms id_dsa_1024_a.pub "login@server.edu"::[ssh2]  
mu scp2/vms AUTHORIZATION "login@server.edu"::[ssh2]
```

where "login@server.edu" is the destination account and server. If the login name is the same as the account you are currently using, you can omit the login@ and quotes. (VMS requires AUTHORIZATION must be uppercase) Note that this command *may* generate a new "key fingerprint" with the server and ask you to continue. Answer "yes". A message will display that the host key has been saved to a file in the [.SSH2.HOSTKEYS] subdirectory, and then you will be asked for the server account's password. Enter it, and the file should transfer.

If you prefer to use FTP to transfer these files, remember that the public key file must be copied in BINARY mode.

Step 4: Send your file

Once the public and private keys have been copied to the destination server, you are ready to transfer files. The general format of file transfers are:

```
scp2 <flags> <source file> <destination files>
```

where <flags> should be of the following:

```
/vms : for vms-to-vms system transfers  
/translate-vms : for vms-to-non-vms transfers
```

/batch : to send the files non-interactively, such as job submission
/help : for a complete list of flags

Examples

scp2 /vms test.txt "test@test.edu":test_at_dest.txt

This copies the local test.txt to test_at_dest at test.edu.

scp2 /vms/batch test.txt "test@test.edu":[.testdir]

This copies the local test.txt into [.testdir]test.txt at test.edu.

scp2 /vms/batch test.txt "test@test.edu"::

This copies the local test.txt to test.txt in the default dir at test.edu.

scp2 /vms/batch test.txt "test@test.edu":[.testdir]test_at_dest.txt

This copies the local test.txt to [.testdir]test_at_dest at test.edu.

scp2 /vms/batch test.txt "test@test.edu"

WARNING: This copies the local test.txt to a local file named "test@test.edu"

Public Key Authentication: VMS to Unix

Step 1: Generate a private key and a public key

Same instructions as VMS to VMS

Step 2: Create identification file

Follow the instructions as with VMS to VMS for creating the identification file only. (Unix does not use an authorization file)

Step 3: Copy the public key to destination server

As with VMS to VMS transfers, you must send the public key for the sender to the receiver:

```
mu scp2/translate_vms id_dsa_1024_a.pub "login@server.edu::.ssh/"
```

Log in to the Unix receiver, change directory to ".ssh" (note that in order to see the directory, you must type in "ls -a"; otherwise, the leading period will keep it from being listed).

Finally, type in the following command:

```
ssh-keygen -i -f id_dsa_1024_a.pub >> authorized_keys2
```

Step 4: Send your file

Same instructions as VMS to VMS, but use /TRANSLATE_VMS instead of /VMS

Public Key Authentication: Unix to VMS

Step 1: Generate a private key and a public key

To do this, issue the following command on the Unix sender:

```
ssh-keygen -t dsa
```

When prompted for a passphrase, hit “Enter” twice. This command will create the files `id_dsa` and `id_dsa.pub` in the `.ssh` directory.

Step 2: Create identification file

Change directory to the `.ssh` directory. Use your favorite editor (Pico, Emacs, Vi, etc) to write a file called “identification” (*Note: must be lowercase*). Within this file, specify the name of the private key (i.e., `id_dsa`) that was created in the step above. Save the file.

Step 3: Send public key to destination server

From the Unix sender’s `.ssh` directory, issue the following command:

```
ssh-keygen -e -f id_dsa.pub >> id_dsa_export.pub  
scp id_dsa_export.pub “login@server.edu:ssh2/”
```

Then, log into the VMS receiver and change directory (set default) to the `[.ssh2]` directory. Issue a `dir` command, and look for the “`id_dsa_export.pub`” file. After verifying that it is there, look for the “`AUTHORIZATION.`” file—if it’s not there, create it—and add the following line to it:

```
key id_dsa_export.pub
```

Step 4: Send your file

```
scp <flags> <source file> <destination files>
```

where `<flags>` should be “`-B`” to send the files non-interactively (i.e. job submission)

Examples

```
scp test.txt test@test.edu:test_at_dest.txt
```

This copies the local `test.txt` to `test_at_dest` at `test.edu`.

```
scp -B test.txt test@test.edu:testdir/
```

This copies the local `test.txt` into `testdir/test.txt` at `test.edu`.

```
scp -B test.txt test@test.edu:
```

This copies the local `test.txt` to `test.txt` in the default dir at `test.edu`.

```
scp -B test.txt test@test.edu:testdir/test_at_dest.txt
```

This copies the local `test.txt` to `testdir/test_at_dest` at `test.edu`.

```
scp -B test.txt test@test.edu
```

WARNING: This copies the local `test.txt` to a local file named “`test@test.edu`”

Public Key Authentication: Unix to Unix

Step 1: Generate a private and public keys

Same instructions as Unix to VMS

Step 2: Create identification file

Same instructions as Unix to VMS.

Step 3: Send authorization file and public key to destination server

From the Unix sender's .ssh directory, type in the following command:

```
scp id_dsa.pub login@server.edu:.ssh/
```

Log in to the Unix receiver, change directory into the .ssh directory, and type in the following command:

```
cat id_dsa.pub >> authorized_keys2
```

4. Send your file

Same instructions as Unix to VMS