

# i-Fax API

## Section 1.1 : i-Fax API for Sending Fax

Introduction :

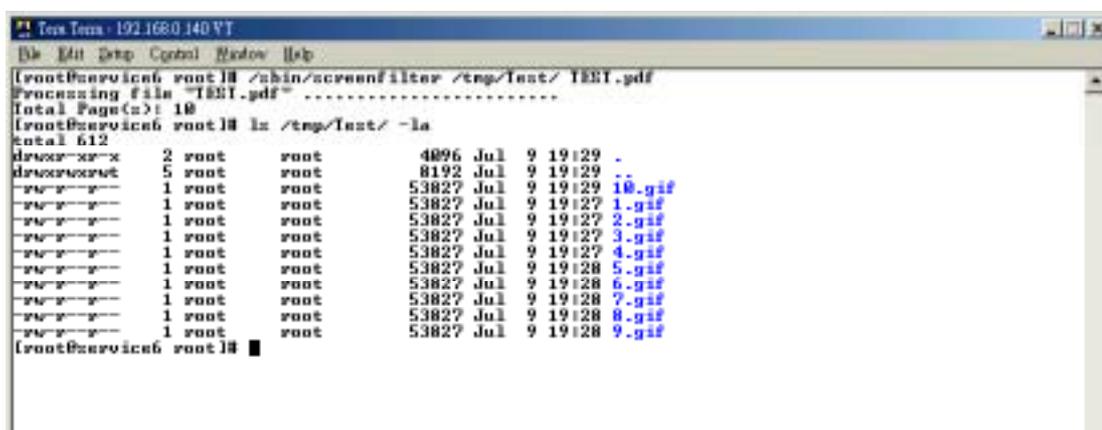
Use a command to change some files format to gif and fax format.

Change some files format to gif format, the method is :

```
export FILETYPE=gif
```

```
/sbin/screenfilter <the file's path> <file1> <file2> ...<fileN>
```

After running screenfilter, each page of your original file will be changed to one gif file. And the original file will be deleted.



```
root@service6 root# /sbin/screenfilter /tmp/Test/TEST.pdf
Processing file "TEST.pdf" .....
Total Page(s): 10
root@service6 root# ls /tmp/Test/ -la
total 612
drwxr-xr-x  2 root  root    4096 Jul  9 19:29 .
drwxr-xr-x  5 root  root    8192 Jul  9 19:29 ..
-rw-r--r--  1 root  root   53827 Jul  9 19:27 10.gif
-rw-r--r--  1 root  root   53827 Jul  9 19:27 1.gif
-rw-r--r--  1 root  root   53827 Jul  9 19:27 2.gif
-rw-r--r--  1 root  root   53827 Jul  9 19:27 3.gif
-rw-r--r--  1 root  root   53827 Jul  9 19:27 4.gif
-rw-r--r--  1 root  root   53827 Jul  9 19:28 5.gif
-rw-r--r--  1 root  root   53827 Jul  9 19:28 6.gif
-rw-r--r--  1 root  root   53827 Jul  9 19:28 7.gif
-rw-r--r--  1 root  root   53827 Jul  9 19:28 8.gif
-rw-r--r--  1 root  root   53827 Jul  9 19:28 9.gif
root@service6 root#
```

After running screenfilter, each page of your original file will be changed to one gif file.

Change some files format to fax format, the method is :

```
export FILETYPE=g3
```

```
/sbin/screenfilter <the file's path> <file1> <file2> ...<fileN>
```

After running screenfilter, each page of your original file will be changed to one g3 file. And the original file will be deleted.



### How to send fax with Class 1 fax modem:

1. /etc/rc.d/init.d/ifax stop (Stop normal i-Fax procedures in order to avoid the dial-out conflict)
2. export FAXOUT\_NOW=yes
3. export FAXOUT\_SPOOL\_DIR=/directory\_full\_path\_storing\_your\_G3\_fax\_files/
4. /sbin/ifaxmodem -d /dev/tty?? -o 1f -t Fax\_Number Faxed\_G3\_Formatted\_File1 ...Other\_G3\_Files
5. /etc/rc.d/init.d/ifax start

Example:

```
/etc/rc.d/init.d/ifax stop
export FAXOUT_NOW=yes
export FAXOUT_SPOOL_DIR=/var/spool/fax/outgoing
/sbin/ifaxmodem -d /dev/ttyS0 -o 1f -t 21992880 g3faxfile.01 g3faxfile.02
/etc/rc.d/init.d/ifax start
```

### Section 1.2 : i-Fax API for FaxMerge

Use command “/sbin/faxmerge” to do FaxMerge, the method is :

- 1) Firstly, export those fields you want to do FaxMerge. i-Fax support the following environment fields.

ID, DATE, TIME, TOTAL\_PAGES, FROM\_COMPANY  
FROM\_PERSON, TO\_COMPANY, TO\_PERSON, SUBJECT  
FROM\_FAX, FROM\_EMAIL, TO\_FAX, TO\_EMAIL  
FROM\_VOICE, TO\_VOICE, COMMENT

Notice: All environment must match those Macro fields in the \*.htm or \*.ps files.

- 2) Put the xxx.htm or xxx.ps in one folder and then do the following command.

faxmerge <Filename\_with\_Full\_Path>

E.g.: export TO\_COMPANY="Your Client's Company Name"

export TO\_PERSON="Your Client's Name"

export SUBJECT="Fax Subject"

export ... other environments

/sbin/faxmerge /tmp/test.ps

/sbin/faxmerge /var/spool/mysource/test.htm

/sbin/faxmerge /home/myname/mydata/test.html

- 3) After doing “/sbin/faxmerge” command, the content (Macros) in \*.ps or \*.htm will be replaced by content of those environment fields.

## Section 1.3 : i-Fax API for Making Fax Queues

Use another command “/sbin/printfax” to create fax queues with result codes, the method is :

```
export SERVER_PORT=API ; export REMOTE_ADDR=remote.ws.ip.addr
cat faxqueue.dat | /sbin/printfax 2> status.log
```

### Notice:

- 1) Before using this API, you must make sure the permission, owner and group to be right for /sbin/printfax. If they are wrong, please use these commands “chown root.ifax /sbin/printfax ; chmod u=rwxs,g=rx,o=rx /sbin/printfax” to correct the permission and owner for /sbin/printfax.
- 2) “remote.ws.ip.addr” means “IP Address of the remote machine connecting to this server”.
- 3) The unique Fax IDs are stored in this file “status.log” for all successful fax queues in order that you can get back its status with another command later.
- 4) This method will write the result code “No Fax Job” or Fax IDs to the file “status.log” for you to get back the status of fax queue. Before calling this command, you must create one faxqueue file “faxqueue.dat” with the following method. You can get one sample “faxqueue.dat” inside the API package from this link <http://www.unitednets.com/ifaxapi.htm>

### How to create Fax Queue file “faxqueue.dat”:

Please create one binary file with the content below and every line must be terminated by one ‘\n’ except item #79.

- 1) To Client Fax numbers (Multiple Fields) --> Use ‘\1’ to divide every fax number
- 2) i-Fax Login ID (1 Field)
- 3) Recipients' Names (Multiple Fields) --> Use ‘\1’ to divide every recipient’s name
- 4) Fax Subject Name (1 Field)
- 5) i-Fax Password (1 Field)
- 6) Recipients' Emails (Multiple Fields) --> Use ‘\1’ to divide every recipient’s email address
- 7) Recipients' Contact Phone Numbers (Multiple Fields) --> Use ‘\1’ to divide every recipient’s phone no
- 8) Recipients' Company Names (Multiple Fields) --> Use ‘\1’ to divide every recipient’s co name
- 9) Email Notification for Fax Result [Yes/No] (1 Field)
- 10) Email Fax Copy [Yes/No] (1 Field)
- 11) Fax Header [Yes/No] (1 Field)
- 12) Cover Page [Yes/No] (1 Field)
- 13) Fax Cover Template Full Path (1 Field)
- 14) Fax Cover Page Content (1 Field)
- 15) Sender's Company Name (1 Field)
- 16) Sender's Email Address(es) (Multiple Fields) --> Use ‘ ’ to divide every Email Address
- 17) Sender's Full Name (1 Field)
- 18) Sender's Company Telephone Number (1 Field)
- 19) Sender's Company Fax Number (1 Field)
- 20) Sender's Mobile Phone Number (1 Field)
- 21) Sender's Company Web Site URL (1 Field)
- 22) Sender's full Company Address (1 Field) --> Use “<BR>” to replace all “\r\n” or “\r” or “\n”
- 23) Recipients' full Company Addresses (Multiple Fields)  
--> Use ‘\1’ to divide every Company Address & use “<BR>” to replace all “\r\n” or “\r” or “\n”



### How to transfer one fax queue from one Linux machine to another Linux machine running i-Fax server:

- 1) Copy the Fax Queue file from the local Linux machine to the remote i-Fax server :-  
`scp -i identity_file faxqueue.dat user@remote_hostdomain:`
- 2) Create one new fax queue on the remote i-Fax server :-  
`ssh user@remote_hostdomain -i identity_file "export SERVER_PORT=API ; export REMOTE_ADDR=remote.ws.ip.addr ; cat faxqueue.dat | /sbin/printfax 2>&1" > status.log`
- 3) Get status of one fax queue on the remote i-Fax server:-  
`ssh user@remote_hostdomain -i identity_file "/sbin/getsentfaxstatus SentFaxID 2>&1" > faxstatus.log`  
Remark: SentFaxID is content of file "status.log" created in item #2 above.

**Notice:** Before doing the above steps, must create one key pair by running `ssh-keygen`. Then, store the private key in `/home/ifax/.ssh/id_dsa` and the public key in `/home/ifax/.ssh/authorized_keys` in the user's home directory with suitable permission and ownership. Follow all steps of "man ssh" in detail to set up one auto login for user "ifax" with key file without entering any password !! After that, you must use the following command line on the local Linux machine to login the remote i-Fax server once at least in order to create one entry of trusted-host .

```
ssh user@remote_hostdomain -i identity_file "pwd"
```

If you can login the remote i-Fax server with this command line and see home directory's name, you can start other steps for making queues.

### Example (Screen Shot) for using i-Fax API to make Fax Queues to remote server:

```
[root@Linux server]# ls -l
-rw-r--r--  1 root  root      22031  8月  7 16:54 faxqueue.dat
[root@Linux server]# scp -i /home/ifax/.ssh/id_dsa faxqueue.dat ifax@192.168.0.1:
[root@Linux server]# ssh ifax@192.168.0.1 -i /home/ifax/.ssh/id_dsa "export SERVER_PORT=API ;
export REMOTE_ADDR=192.168.0.123 ; cat faxqueue.dat | /sbin/printfax 2>&1" > status.log
[root@Linux server]# cat status.log
00000841
00000842
[root@Linux server]# ssh ifax@192.168.0.1 -i /home/ifax/.ssh/id_dsa "/sbin/getsentfaxstatus
00000841"
Server Port : Server01ttyS0
Status      : Sent
Description : Fax Sent Successfully
```