

FQA 2 – Getting To Know 9front



2.1 – Web Pages

The official websites for the 9front project are located at:

- <http://9front.org> — main website
- <http://bugs.9front.org> — bug tracker
- <http://fqa.9front.org> — official documentation, frequently questioned answers
- <http://lists.9front.org> — mailing lists
- <http://man.9front.org> — man pages
- <http://wiki.9front.org> — informal documentation, developer scratchpad

Somewhat outdated, and not entirely applicable to 9front is the Bell Labs Plan 9 wiki:
http://www.plan9.bell-labs.com/wiki/plan9/plan_9_wiki

Some users have set up their own sites and pages with 9front specific information. As always, do not blindly enter commands you do not understand into your computer.

That said, much valuable information can be found at <http://cat-v.org> regarding various aspects of UNIX, Plan 9, and software in general.

2.2 – Mailing Lists

The 9front project maintains several mailing lists which users should subscribe to and follow. To subscribe to a mailing list, send an e-mail message to list-owner@9front.org,

where *list* is the name of the mailing list. That address is an automated subscription service. In the body of your message, on a single line, you should include the subscribe command. For example:

```
subscribe
```

After subscribing, list messages are received from and sent to *list@9front.org*, where *list* is the name of the mailing list.

The *lists* are as follows:

9front — general discussion

9front-bugs — report bugs

9front-commits — read-only log of commits to the 9front mercurial repository

9front-fqa — log of commits to, and discussion about
<http://fqa.9front.org>

9front-sysinfo — read-only log of `sysinfo(1)` output from user systems

To unsubscribe from a list, you will again send an e-mail message to *list-owner@9front.org*. It should look like this:

```
unsubscribe
```

Before posting a question on the 9front mailing list, please check the rest of this FQA. If asking a question possibly related to hardware, always include the output of `sysinfo(1)`!

2.2.1 – Mailing List Archives

Mailing list archives are available via 9p:

```
% 9fs 9front
post...
% ls /n/lists
/n/lists/9atom
/n/lists/9changes
/n/lists/9fans
/n/lists/9front
/n/lists/9front-bugs
/n/lists/9front-commits
/n/lists/9front-fqa
/n/lists/9front-sysinfo
/n/lists/9nag
/n/lists/acme-sac
/n/lists/cat-v
/n/lists/cypherpunks
/n/lists/dlr
/n/lists/harvey
/n/lists/harvey-commits
/n/lists/harvey-issues
/n/lists/inferno
/n/lists/nix
/n/lists/plan9port-dev
/n/lists/sam-fans
/n/lists/sierra31
/n/lists/skunk-works
/n/lists/tscm-l
/n/lists/tuhs
/n/lists/werc
/n/lists/www-html
/n/lists/www-talk
/n/lists/www-vrml
```

Each of these directories contains messages sent to the respective mailing list in mdir format, one message per file. Erik Quanstrom's nupas upas/fs (now merged with 9front upas) can mount these directories as mailboxes, for reading directly with a mail client. **Note:** Opening large mailboxes over a slow 9p link will be very slow.

2.2.2 – Other useful mailing lists



9fans — Official mailing list of Plan 9 from Bell Labs.

9nag — Ping alerts for plan9.bell-labs.com (unofficial).

2.3 – Manual Pages

9front comes with extensive documentation in the form of manual pages. Considerable effort is made to make sure the man pages are up-to-date and accurate. In all cases, 1.) the source, followed by 2.) the man pages, are considered the authoritative source of information for 9front.

You can find all the 9front man pages on the web at the following sites:

<http://man.9front.org>

<http://felloff.net/sys/man>

<http://man.aiju.de>

<http://man.cat-v.org/9front>

<http://plan9.stanleylieber.com/man>

as well as on your 9front computer.

In general, if you know the name of a command or a manual page, you can read it by executing `man` command. For example: `man vi` to read about the MIPS binary emulator. If you don't know the name of the command, or if `man` command doesn't find the manual page, you can search the manual page database by executing `lookman something`, where `something` is a likely word that might appear in the title of the manual page you're looking for. For example:

```
% lookman vesa
man 3 vga # vga(3)
man 8 realemu # realemu(8)
man 8 vga # vga(8)
```

The resulting list is presented in the form of commands that can be highlighted and sent using rio's mouse button 2 menu.

For many, having a hardcopy of the man page can be useful. To make a printable copy of a man page:

```
man -t vga realemu | lp -dstout > vga.ps
```

or:

```
man -t vga realemu | lp -dstout | ps2pdf > vga.pdf
```

2.3.1 – How do I display a man page source file?

The command `man -w` prints the location of the man page's source file:

```
% man -w vga realemu
/sys/man/3/vga
/sys/man/8/realemu
/sys/man/8/vga
```

2.3.2 – How do I write my own manual page?

As was mentioned in *FQA 0.3.2 – Man pages*, Plan 9 man pages are divided into sections based upon the features they describe. For example, the troff source of the `io(1)` man page is located in section 1, in the file `/sys/man/1/io`. You can examine this troff source for a simple example of how to format and construct a typical man page.

For additional information, read: *How to Write a Plan 9 Manual Page*, `man(6)`, `troff(1)`

2.4 – Reporting Bugs



"I have with Plan 9 Is there a way to"

Remember: 9front developers cannot read your mind. Very detailed information is required to diagnose most serious bugs.

Also remember: It is possible you simply don't know what you're doing. If you do not understand how something is done in 9front or how it works, and can't figure out how to resolve the problem using the manual pages, *FQA 8 – Using 9front*, or *FQA 9 – Troubleshooting*, join #9front on `irc.freenode.net` or use the mailing list (`9front@9front.org`) to request help.

On the other hand, it's possible you really found a bug. If so, please file a bug report by sending an e-mail to `9front-bugs-open@9front.org`. The subject line of your e-mail will be used to open an issue on the bug tracker, and your message will be forwarded to the 9front-bugs mailing list.

2.4.1 – Your bug report SUCKS

A minimum useful bug report contains the following:

- A complete description of what you did.
- A complete description of what you expected to happen.
- A complete description of what happened, instead.

Extra credit:

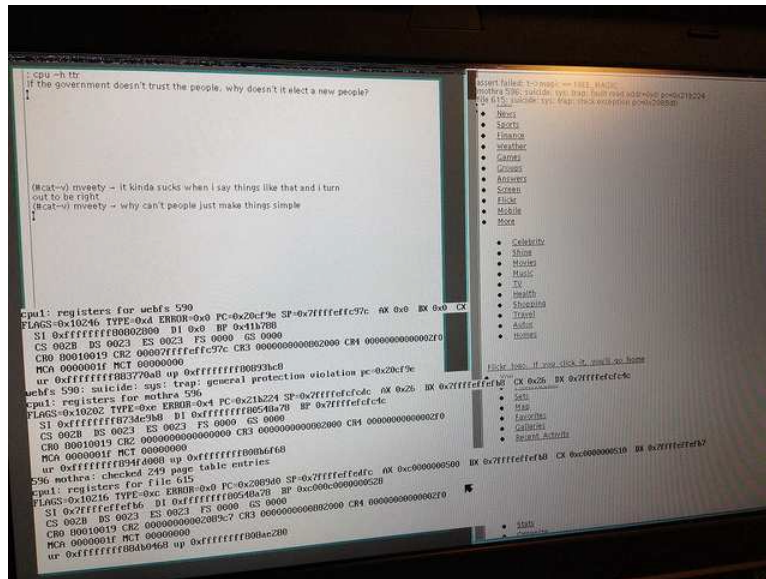
- The make and model of your computer.
- The URL returned when you run `sysinfo -p` on the affected system.

2.4.2 – How do I get more useful info for developers?

The following sub-sections offer some useful tips.

2.4.2.1 – Take a photo of the panic screen

Under some circumstances, you may not be able to recover text directly from a crashed system. In this situation, snap a photo of the screen, including as much of the screen in good focus as possible, and upload the image somewhere where it can be retrieved by developers.



2.4.2.2 – Make a stack trace of a crashed process

```
% hget http://www.jwz.org/blog/feed | rssread
panic: D2B called on non-block c580 (double-free?)
rssread 882285: suicide: sys: trap: fault read addr=0x0 pc=0x000038ef
% acid 882285
/proc/882285/text:386 plan 9 executable
/sys/lib/acid/port
/sys/lib/acid/386
acid: lstk()
abort()+0x0 /sys/src/libc/9sys/abort.c:6
ppanic(p=0xb5a0,fmt=0xbf72)+0x146 /sys/src/libc/port/malloc.c:166
    pv=0xc100
    msg=0xc310
    v=0xdffffee90
    n=0x2b
D2B(p=0xb5a0,v=0xc580)+0x5a /sys/src/libc/port/pool.c:966
    a=0xc578
poolfreel(v=0xc580,p=0xb5a0)+0x20 /sys/src/libc/port/pool.c:1190
    ab=0xc100
poolfree(p=0xb5a0,v=0xc580)+0x41 /sys/src/libc/port/pool.c:1325
free(v=0xc588)+0x23 /sys/src/libc/port/malloc.c:250
nextxmlpull(x=0xc548)+0x334 /usr/sl/src/xmlpull/xmlpull.c:221
nextxmlpull(x=0xc548)+0x2cb /usr/sl/src/xmlpull/xmlpull.c:316
main()+0x34 /usr/sl/src/rssread/rssread.c:159
    st=0x1
    f=0xc5a8
    r=0x0
    x=0xc548
_main+0x31 /sys/src/libc/386/main9.s:16
acid:
```

2.4.2.3 – Make a process snapshot

To collect even more information than a stacktrace, you can also make a full process snapshot, which includes all the memory of the program. The snapshot file can later be analyzed (even on a different machine [even on a different machine of a different architecture]).

```
% snap 882285 > rssread.snap
```