Panic for historical reasons:

Anecdotes from a life filled with BSD

Walter Belgers
walter@belge.rs
NetBSD 1.1 1.3.2 1.3.3 1.4.1 1.5.2 1.5.3 1.6 1.6.1 1.6.2 3.1 6.1 8.0
OpenBSD 2.3 2.5 3.1 3.3 3.5 3.6 5.0 5.6 5.9 6.2 6.9
FreeBSD 1.1.5.1 2.0 2.1 2.1.5 2.2.1 2.2.5 4.4 4.5 4.10 5.3 6.0 6.2
7.0 7.1 7.2 8.0 8.1 8.2 9.0 9.1 9.2 10.0 10.1 10.2 10.3 11.0 11.1
11.2 12.0 12.1 12.2 13.0 13.1 13.2
The start

TU/e
Technische Universiteit
Eindhoven
University of Technology

- SunOS
- Ultrix
The start

Ultrix V2.2 (eutws1)

login: rcstwb
Password:
Last login: Mon Feb 19 10:46:07 on tty00

You have mail.
% ls /bin /usr/bin /usr/local/bin | lpr
% man man
% man stty
% stty 0>/dev/tty3
Guido van Rooij used 386bsd

Interesting!
But my PC (8086, 640kB RAM, 20MB disk) is not suited
OOPS...!
My First BSD

- 1995-2002
- 486DX100, 64MB RAM(!)
- FreeBSD 1.1.5.1, 2.0, 2.1, 2.1.5, 2.2.1, 2.2.5, 4.4, 4.5
- UUCP, Samba, NFS, NTP, firewall, INN, GRE tunnel for /28 IPv4 and /60 IPv6 networks
- make buildworld: 24 hours
My First BSD

• 2002: Regular crashes - hardware related?
• Slowless.. (MP3 playing, SSH)
• Upgraded to Intel Celeron
• Still crashes..
Longest running issue

- Issue IPv6+NAT+ipf/fw, never solved despite low-level debugging
- 2007: workaround: disabled IPv6
- 2011: FreeBSD 9.0: really solved
At work

- Senior Security Consultant
- My own FreeBSD desktop machine
At work

- Root password: empty!

- But, only ‘walter’ in group wheel and tty’s insecure

root::0:0:Charlie &:/root:/bin/sh
walter:g3De2.eeFvY1Z:100:0:Walter Belgers:/home/walter:/bin/tcsh
+guido::::::
+arjan::::::
At work - #1

With access to the NIS master, you can choose your uid on my box..
At work - #2

- Cannot leave a live session on my box, because:
  
  walter% su
  #

- Always lock the screen! Tool: xlockmore

-/+allowroot
  The *allowroot* option allows the root password to unlock the server as well as the user who started *xlock*. May not be able to turn this on and off depending on your system and how *xlock* was configured.
Guido was able to change the motd - he got root!

Did not want to tell me how...

But when you are security aware.. you log stuff!
At work - #3

FreeBSD-SA-96:12

Security Advisory
FreeBSD, Inc.

Topic: security compromise from perl (suidperl) utility

Category: core and ports
Module: perl
Announced: 1996-06-28
Affects: FreeBSD 2.0, 2.0.5, 2.1, 2.1-stable, and 2.2-current
Corrected: 2.1-stable and 2.2-current as of 1996-06-03
FreeBSD only: no


---

III. Impact

This vulnerability can only be exploited by users with a valid account on the local system to easily obtain superuser access.
At work - #4

• Who ever did an `rm -rf /` by accident?

• Just as dangerous:

  root# tar cvf /dev/wd0 .
  ^C
  root#
Collecting UNIX servers

- I started collecting Sun, HP, IBM, SGI, Sony, Digital, RDI, Motorola, Altos, NeXT, NCR, Tadpole, MIPS, Apollo, .. servers
UNIX servers

- Installing FreeBSD, NetBSD, OpenBSD
- Challenges everywhere, e.g. this VAXstation 2000
  - Netboot only (no tape drive)
  - Disk controller not supported in OpenBSD
  - NetBSD also issues
UNIX servers

- Sun 3/50 (68020 CPU, 12MB RAM, 327MB drive)
- Compiling NetBSD 1.6.1 kernel: 14 hours, 40 mins
- Kernel must be ≤ 1008kB (1MB - VMEM)
- Third attempt fit - days of fun!
UNIX servers

- Sun 2/50 (68010 CPU, 4MB RAM, no drive)
- From before rarp/bootparam/tftp/NFS tools existed
- Uses “network disk”, transferring disk blocks
- Matt Fredette’s ndbootd on NetBSD did the trick (now integrated)
UNIX servers

- This RDI PrecisionBook 180 was used to port OpenBSD to PA-RISC architecture
- Which it is still running

*OpenBSD hppa*
HIP97

• Large outdoor hacker camp

• Theo de Raadt was there

• Although his friends said he was not
OpenBSD

Only two remote holes in the default install, in a heck of a long time!

- Still, you can run insecure stuff on OpenBSD.
PicoBSD

- FreeBSD 3.x+ based OS (I used 4.6.2)
- Runs from floppy, meant as router/firewall
- Minimal spec: 80386SX, 8MB RAM
- ‘seejpeg’ viewer is 435kB, but only 271kB free
- Put floppy 1 in RAM, then mount floppy 2 with slides + viewer
Panic

- My own FreeBSD desktop machine
- Suddenly: a panic!

**panic**: panic for historical reasons

```
Internal network

NIS server

my desktop
```
Panic

**Index: src**

FreeBSD source tree

<table>
<thead>
<tr>
<th>about</th>
<th>summary</th>
<th>refs</th>
<th>log</th>
<th>tree</th>
<th>commit</th>
<th>diff</th>
</tr>
</thead>
<tbody>
<tr>
<td>path: root/sys/i386/eisa/aha1742.c</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Blob: 77ca2a2826944f6111e6c3d67a5f05440146a02a (plain) (blame)**

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>/*</td>
</tr>
<tr>
<td>2</td>
<td>* Written by Julian Elischer (<a href="mailto:julian@tfs.com">julian@tfs.com</a>)</td>
</tr>
</tbody>
</table>
Panic

/*
 * Function to send an immediate type command to the adapter
 */
static void
ahb_send_immed(struct ahb_data *ahb, int target, u_long cmd)
{
    int port = ahb->baseport;
    int s = splbio();
    int stport = port + G2STAT;
    int wait = 100;    /* 1 ms enough? */

    while (--wait) {
        if ((inb(stport) & (G2STAT_BUSY | G2STAT_MBOX_EMPTY))
            == (G2STAT_MBOX_EMPTY))
            break;
        DELAY(10);
    }
    if (wait == 0) {
        printf("ahb%d: board is not responding\n", ahb->unit);
        Debugger("aha1742");
        fatal_if_no_DDB();
    }

    outl(port + MBOXOUT0, cmd);    /* don't know this will work */
    outb(port + G2CNTRL, G2CNTRL_SET_HOST_READY);
    outb(port + ATTN, OP_IMMED | target);
    splx(s);
}
Panic

```c
#define fatal_if_no_DDB()
#else
#define fatal_if_no_DDB() panic("panic for historical reasons")
#endif
```
Thank you

386
OpenBSD
FunBSD
NetBSD
DragonFlyBSD
FreeBSD

Computer collection: https://giga.nl/walter/computers/