An ISP Perspective, jail(8)
Virtual Private Servers

Isaac Levy (.ike) <ike@lesmuug.org>
Materials prepared for AsiaBSDCon 2007 Proceedings, University of Tokyo, Japan. These materials are Copyright 2006 Isaac Levy, under the terms of the BSD license.
.ike Context

- I have lived inside, and outside, jails extensively for web application servers and software development purposes.
- I am not a jail author, no commit bit...
- NO religious OS/Virtualization bashing, yet strong arguments presented based on specific application/threat models.
I helped run the ISP iMeme (now defunct), perhaps the first specific jail(8) based ISP in the US.

I currently operate a micro, high-availability datacenter operation for my clients, using jail-oriented systems.
(wintermute) Jon Ringuette, (partner and founder of iMeme), taught me to jail(8).
done technical talks on jail before

you have root!
chickenhawk:/home/ike ike$ sudo cat /proc/35721/status
sshd 35721 1 35721 35721 - sldr 1138578391,270482 4,956102 27,766581 select 0 0 0,0 chick.diversaform.net
chickenhawk:/home/ike ike$ sudo cat /proc/93030/status
sshd 93030 93025 93025 93025 - ndflags 1152553319,908960 0,61426 0,40950 select 1001 1001 1001,1001,1001,0 -
chickenhawk:/home/ike ike$
Warranty / Announcement

• This is a lot of information for 1 hour, will move fast.... (questions after, please)

• I’ll be around if anyone has more complex questions or strategies they want to discuss.
Warranty / Announcement

• I’m *trying* to stay close to classic UNIX process and ideas, and ‘stock’ methodology.

• I assume you all know the basics of the jail(8) mechanism on FreeBSD, but...
jail(8) is:

chroot(2) bound to an IP address

Hardware
FreeBSD UNIX Operating System
chroot(2) chdir(1) socket(2) TCP/IP
jail(8) /etc/rc
program
jail(8) /etc/rc jail(8) /etc/rc
program
program
program
program
program
jail(8) is:

chroot(2) bound to an IP address

a tool for creating virtual servers
jail(8) is:

- `chroot(2)` bound to an **IP address**
- A tool for creating virtual servers
- Native on FreeBSD since 1998
jail(8) is:

- `chroot(2)` bound to an IP address
- A tool for creating virtual servers
- Native on FreeBSD since 1998
- Designed to partition ‘mutually untrusted users’
Mutually Untrusted Users?
Mutually Untrusted Users?
Internet Service Provider (ISP)

Common Definition: a business or organization that provides users access to the Internet and related services, (web hosting, email, etc...).

Abstract Definition: providing users the ability to run programs, and maintain persistent data storage, available across a network (like the internet).
scaling, patterns, time
(an exercise)
THE ARPA NETWORK

DEC 1969

FIGURE 6.2 Drawing of 4 Node Network
(Courtesy of Alex McKenzie)
ARPA LOGICAL MAP, MARCH 1977

(Please note that while this map shows the host population of the network according to the best information obtainable, no claim can be made for its accuracy.)

Names shown are IMP names, not (necessarily) host names.

http://www.cybergeography.org/atlas/historical.html
http://www.opte.org/maps/
In 2007, 50% of the world is urban

- It was 3% in 1800
- 14% in 1900
- 61% expected in 2030
- World Population
- Internet Users

<table>
<thead>
<tr>
<th>Year</th>
<th>World Population (Billions)</th>
<th>Internet Users (Millions)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1995</td>
<td>5.674,380,000</td>
<td>20,000,000</td>
</tr>
<tr>
<td>2000</td>
<td>6.070,581,000</td>
<td>304,000,000</td>
</tr>
<tr>
<td>2005</td>
<td>6.453,628,000</td>
<td>938,000,000</td>
</tr>
</tbody>
</table>

[http://www.gazetteer.de/home.htm](http://www.gazetteer.de/home.htm)
http://www.opte.org/maps/
people ARE the city
Marshall McLuhan, well-known for coining the expression "global village".
Marshall McLuhan, well-known for coining the expression "global village".

"All media are extensions of some human faculty- psychic or physical"

"...the wheel is an extension of the foot the book is an extension of the eye, clothing, an extension of the skin,

electric circuitry, an extension of the central nervous system"

- The Medium is The Massage
people ARE the internet
Tim Berners-Lee

Tim Berners-Lee is the Director of the World Wide Web Consortium, Senior Researcher at MIT’s CSAIL where he leads the Decentralized Information Group (DIG), and Professor of Computer Science at Southampton ECS.


Bio

A graduate of Oxford University, England, Tim now holds the 3Com Founders chair at the Laboratory for Computer Science and Artificial Intelligence Lab (CSAIL) at the Massachusetts Institute of Technology (MIT). He directs the World Wide Web Consortium, an open forum of companies and organizations with the mission to lead the Web to its full potential.

With a background of system design in real-time communications and text processing software development, in 1989 he invented the World Wide Web, an internet-based hypermedia initiative for global information sharing, while working at CERN, the European Particle Physics Laboratory. He wrote the first web client (browser-editor) and server in 1990.

Before coming to CERN, Tim worked with Image Computer Systems, of Fareham, Dorset, England and
Berkeley Software Distribution

(BSD) exists here. For other uses, see BSD (disambiguation).

Berkeley Software Distribution (BSD) (sometimes called Berkeley Unix) is the Unix-derivative distributed by the University of California, Berkeley, starting in the 1970s. The name is also used collectively for the modern descendants of these distributions. BSD is one of several branches of Unix operating systems. Another one is derived from Unix System V developed by AT&T's Unix System Development Labs. A third consists of the GNU/Linux operating systems which draw from Unix System V and BSD, as well as Plan 9, and non-Unix operating systems. BSD was also identified with the versions of Unix available for workstation-class systems. This could be attributed to the ease with which it could be licensed and the familiarity it found among the founders of many technology companies during the 1980s. This popularity often came from using similar systems—notably DEC-Unix and Sun's Solaris—during their evolution. While BSD itself was largely superseded by the System V releases of OpenBSD and FreeBSD systems in the 1990s, both of which incorporated BSD code, in recent years modified open source versions of the ситуация (mostly derived from 4.4BSD-like) have been increasing in use and development.

Composed by Isaac Levy
Are my expectations of web applications growing?

On 2/12/07, Isaac Levy <ike@lesmuug.org> wrote:

Hey All,

The fine folks running AsiaBSDCon are looking to spread the word about the conference, the schedule is now online! Pretty excellent lineup, their tutorial sessions all look pretty hardcore. It seems to me the heavy focus of the conference is of course on BSD's strength in ISP and heavy network contexts- (no KDE tuning sessions? What?!)

Also, of note, lots of IPv6 stuff at this conference, (in "the" country which actually runs "production" grade IPv6 infrastructure)
homepages?
(raising user expectations of the internet)
Stewart Brand, Currently working with the Long Now Foundation
how did iMeme happen?
Typical ISP Setup (virtual hosting)

1999: bleak options (unless you want LAMP)
Typical ISP Setup (virtual hosting)

```
../vhosts
/customer1
 /www
 /cgi_bin
/customer2
 /www
 /cgi_bin
/customer3
 /www
 /cgi_bin
#customer1
#customer2
#customer3
#customer4
#customer5
#customer6
#customer7
```
The Utopian City?

“A House Is A Machine For Living In”

Le Corbusier

Le Corbusier, Contemporary city for three million inhabitants (Ville Contemporaine - 1922)
Jane Jacobs, well-known for working to maintain diverse and meaningful NYC urban life.

With regard to the ideals of Le Corbusier’s ideas for city design:

“...As in all Utopias, the right to have plans of any significance belonged only to the planners in charge.”

- The Death and Life of Great American Cities
many shapes and scales...
(growth as complex land use, not merely size)

apartment building
(note mixed use retail)

apartment building
(misc west village)

skyscraper
(Empire State Building)
Dedicated Hosting?

1999:
Dedicated hosting can be prohibitively expensive, (and often, a waste of hardware resources!)
Doug McIlroy,
the inventor of Unix pipes, part of early UNIX development.

“This is the Unix philosophy:

Write programs that do one thing and do it well.

Write programs to work together. Write programs to handle text streams, because that is a universal interface.”

“In the era of the staircase all floors above the second were considered unfit for commercial purposes, and all those above the fifth, uninhabitable.”

“Since the 1870s in Manhattan, the elevator has been the great emancipator of all horizontal surfaces above the ground floor.”
jail(8) 1998

(it does one thing)

Robert Watson worked on jail(8) as well...
Web Hosting

Welcome to imeme.net, an Internet Application Service Provider.

Everything through the web

This is what you will get:

you.imeme.net
you@imeme.net
friends@lists.imeme.net
100 MB Space

Only $10 a month
No Setup Fee

jailing server
Welcome to iMeme

iMeme is a hosting provider dedicated to Open Source. We offer solutions and conveniences for web developers. We specialize in Zope and PHP and are building a hosted developers community. Each iMeme account is overseen directly by an owner of the company. We are experienced and kind administrators and provide virtual and dedicated server colocation with prompt, professional service.

What We Offer

Our hosting packages provide you with the functionality and stability to make your ideas come to life. We specialize in reliability, systems and network administration, FreeBSD, and Zope.

iMeme offers a substantial knowledge base. Among our community of clients, we have some of the finest and most creative Zope developers. The iMeme knowledge base is focused in three ways. Canonical documentation and how-tos produced by its managing owners (for iMeme to instruct clients), interactive documentation in the form of a wiki web (so that clients may teach iMeme), and real-time tools for assistance and support (such as irc.imeme.net).

We would like to have you join our growing community of developers and allow us to provide you with the tools and expertise to help you grow.

Internet Update 1/28/04: Windows email virus ravages internet

iMeme largely unaffected by MiMal-based worm, internet traffic clogging.

Email Migration 1/3/04: Email Service Report

Email pauses under spiked spam load, no mail lost, read all about it.

Email Migration 12/24/03: Important Email Service Report

MAILSERVICE MIGRATED WITHOUT DOWNTIME, read all about it.
The real world of iMeme users:

- hacker: “I want to compile LISP”

- undergraduate sociology student: “I want to install ‘Foo’ blog software, it’s PHP and the instructions say I need to run Cron”

- web designer: “I want to run an http server on port 8080”

- business owner: “I want to run Foo web application for my business.”

- A community leader: “I want to run Mailman List Manager”

- 13 year old hacker: “I want to run both an IRC and jabber server for my friends”.

Most iMeme users simply wanted to use Python/Zope.
The UNIX Time-Sharing System*

D. M. Ritchie and K. Thompson

ABSTRACT

Unix is a general-purpose, multi-user, interactive operating system for the larger Digital Equipment Corporation PDP-11 and the Interdata 8/32 computers. It offers a number of features seldom found even in larger operating systems, including

i  A hierarchical file system incorporating demountable volumes,

ii  Compatible file, device, and inter-process I/O,

iii  The ability to initiate asynchronous processes,

iv  System command language selectable on a per-user basis,

v  Over 100 subsystems including a dozen languages,

vi  High degree of portability.

This paper discusses the nature and implementation of the file system and of the user command interface.

NOTE: * Copyright 1974, Association for Computing Machinery, Inc., reprinted by permission. This electronic edition of this paper is a reprint of the version appearing in The Bell System Technical Journal 57 no. 6, part 2 (July-August 1978). In turn, that was a revised version of an article that appeared in Communications of the ACM, 17. No. 7 (July 1974), pp. 365-375. That article was a revised version of a
Internet Service Provider (ISP)

Contemporary Datacenter
Technical Application
management issues

- lost jail?
  - [hostname lockdown]
- resource attacks
- disks full
  - [partitions, disk images]
- fork bombs, memory hogs
  - [securelevels, login.conf]
- process control
- direct driver access
  - [flags to mount devfs, procfs]
management issues

• lost jail?
  • [hostname lockdown]

• resource attacks

• disks full
  • [partitions, disk images]

• fork bombs, memory hogs
  • [securelevels, login.conf]

• process control

• direct driver access
  • [flags to mount devfs, procfs]
General Specs

• High Density 2u (and then 1u) servers

• we aimed to run 50 jails per box

• provisioned 4gb of disk space

• 100mb of what we called ‘process space’, the amalgamation of memory and cpu usage

• Bandwidth was rarely an issue, very basic QOS oriented throttling
General Specs

- Company Scale in 2003
- 1,000 domains
- 480 jails
- Less than half a rack of servers (24u)
- 3 owners, (2 of us ran operations)
Managing Disk Use

• we ran scripts from the host server which simply used du, and shoved the output into MySQL databases

• we then automated the process of enforcing policies of charging for extra disk usage. (simple cron jobs to email users, change their bill, etc...)

• iMeme gave 1 month of ‘grace time’ to trim disk usage, sometimes logfiles would explode, or users would accidentally consume undue disk space- and we felt this was a simple buffer our people appreciated
Managing Disk Use

- FreeBSD file-backed memory disks (disk images)
- `mdconfig(8)`, consult FreeBSD handbook
- Extreme I/O penalty, which is the bottleneck for jailing already!
- disk images consume RAM!
- encrypted disk images could be used as well...
Managing Memory

- Memory and CPU usage polled on a regular basis for each jail, we called it ‘Process Space’

- Shell scripts were originally setup to run as cron jobs inside each jail, which took cumulative memory consumption and cpu usage by parsing ps(1) output inside a given jail

- While iMeme originally ran these scripts inside of each jailed system, outputting totals to text files in /jail/dir/var/log/,

- this carried the risk that a user could (trivially) bypass this system to avoid increased billing or otherwise.
Managing Memory

• iMeme moved this system out of the jails themselves, to the host system with new jailing features in FreeBSD 5.x- (6.x)

• one can list/kill processes based on the jail id, information available to ps, and processes listed in the /proc filesystem.

• jail(8) info is noted in many common utilities 5.x onward
Managing Processes

- process restrictions were then handled neatly using renice(8)
- shell scripts polled every 5 minutes for ‘hog’ processes, which were logged to SQL
  - if the process wouldn’t behave in 5 minutes, it was reniced
- crude but wildly successful
Managing Processes

• Fork bombs were still a threat, but rare.

• Fix not feasible, required booting host server with escalated securelevel
Managing Processes

- from FreeBSD 5.x onward, easy to fix:
  - each jail can be set to start with an escalated securelevel
  - maxprocs could be locked for a jail
  - chflags(2) disabled in jails via host sysctl settings
Far outside the scope of this material, however, it is worth mentioning one thing:

at iMeme, each jailing host server was conceptually treated like a network border or gateway, with routing and filtering tasks carried out inside the machine.
With that, we ran NAT for our external IP blocks, and mapped addresses to our jails- which all ran using a private netblock, (192.168.x.x).
Network Mention

- ipwf and dummynet for simple fair-share traffic shaping
- ipfw performed strongly for putting out fires
- Current Diversaform systems, (me), pf replaces ipfw all around- though high-volume scale usage has not come close to ‘iMeme levels’
Hostname Annoyance

- 4.x jailing relied heavily on a jailed hostname for userland jail identification
- jailed user could change hostname, making it VERY hard to find
- FreeBSD 5.x solved this problem by:
  - pinning a ‘jail id’ to each process on the system
  - providing a sysctl feature to lock down the ability to change hostnames within a jail
deploying new systems

• iMeme used CVS to manage jail ‘skeletons’
• Tarball deployment is your friend.
  clean, simple, reliable.
  be aware of dev/proc mounts
  be aware of symlinks
  be aware of permissions, (and file flags)
• use FreeBSD Ports Mechanism?!
  (not for the ports collection proper, that’s
  insanely presumptuous, [borderline insane])
upgrading jailed systems

• Simply use buildworld, (FROM HOST SYSTEM),

• give buildworld the DESTDIR flag, with a jail’s userland path

• just follow the handbook: http://www.freebsd.org/doc/en_US.ISO8859-1/books/handbook/makeworld.html
managing jailed systems

- iMeme website, where users bought jailed systems, and managed their account and billing, was all written in Zope, and had PHP elements added over time. This could have been any web technology.

- iMeme kept a ‘Master Record Server’ (redundant), running a MySQL database with all jail/customer data:
  - jailed system details
  - resource usage
  - billing and contact information
jumping jails, backups

- every server mounted every other server
- dirty NFS setup (reliable, but clunky)
  - single point of failure for cluster of host servers
- If a jailing host server died:
  - it’s jailed systems could then be rapidly re-distributed across the entire cluster
- required intelligent Administrator intervention
best practices

- ssh into jails to manage their processes! [jexec(8) jail_attach(2) have consequences?!!]
- You always can see the jailed filesystem/userland from host server, be careful.
- Design jailing host system carefully, be creative with core UNIX utilities.
- Use your highest secure practices for host server...
best practices
(stay away from rc jail scripts in ISP context!!!)

#!/bin/sh

# simple, complete script to start a jail.

# define the absolute path to the jail,
J=/usr/local/jails/jailed.userland.directory

# define the ip address for the jail,
I=10.0.1.192

# define a hostname,
H=fqdn.com

ifconfig en0 inet alias $I/32

mount -t procfs proc $J/proc
mount_devfs devfs $J/dev
## add additonal flags to mount_devfs, to hide unnecessary devices!!!
## check the man page for mount_devfs

jail $J $H $I /bin/sh /etc/rc
important utilities

- 4.x FreeBSD, jps, jkill, jtop (ports)
- 5.x, 6.x, onward builtin ps, kill
- !plus jls(8), jexec(8) jattach(2), sysctl features for jailing
- Design your jailing system carefully, be creative (note about nullfs, devfs)
- additionally, handy: pstree, xtail, disk images via mdconfig
future directions (for me)...

and if I did it again, (a full blown ISP):

- GEOM, GGated over NFS?
- CARP, from PF/OpenBSD - makes life easier
- more NAS/SAN support (GEOM, ggated)
- Experiment with SysJail, (OpenBSD, NetBSD!)
current directions (for me)...

CATCHING MY BREATH AFTER iMeme!!!

- working freelance from NYC
- running highly redundant jailed systems for clients
- always hacking around with jail stuff...
Jailing Party - jail(8)
Friday-Monday, Sept. 3-6 2004
Next time we get more resilient hardware...
Thanks everyone for all the fun!
Special Thanks:

wintermute (partner at iMeme), taught me to jail(8).


Robert Watson wrote the extended documentation, found a few bugs, added a few new features, and cleaned up the userland jail environment.
Q&A?
jail(8)