illuria security

History of Containers

reARMSEC 2020

by: Antranig Vartanian

`whoami'

- Co-founder && CEO @ illuria Security, Inc.
- HEAD @ EVNCERT
- Daemon @ Armenia BSD User Group
- Past: CTO, Systems Engineer
- FreeBSD developer and advocate
- Runs Jabber.am :-)

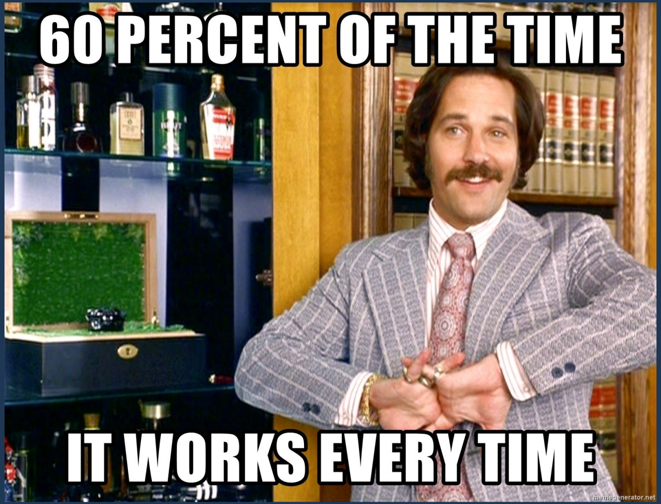
Disclaimer: I have no idea wtf I’m talking about

Agenda

- Why this talk
- History of containers
  - Containers before it was cool
  - Containers before it made money
  - Containers making (and loosing) money
- Save the whales, drop VMs
- Security in Containers
- What they didn't tell you about Virtualization
- Q&A

Why this talk

- Containers are mainstream
  - /* sigh */ Docker is mainstream
- You're doing it wrong!
- 60% of the time, it works every time



Container Prehistory

- Containers are not new
- Originated with chroot(2)
- [CHROOT]
  Dr. Marshall Kirk Mckusick, private communication: ``According to the SCCS logs, the chroot call was added by Bill Joy on March 18, 1982 approximately 1.5 years before 4.2BSD was released. That was well before we had ftp servers of any sort (ftp did not show up in the source tree until January 1983). My best guess as to its purpose was to allow Bill to chroot into the /4.2BSD build directory and build a system using only the files, include files, etc contained in that tree. That was the only use of chroot that I remember from the early days.''

`chroot(2)` & `chroot(8)`

- Detailed history written by Warner Losh
  - **Whither chroot?** https://bsdimp.blogspot.com/2020/06/whither-chroot.html
- `chroot(8)` first appeared in 4.3BSD-Reno

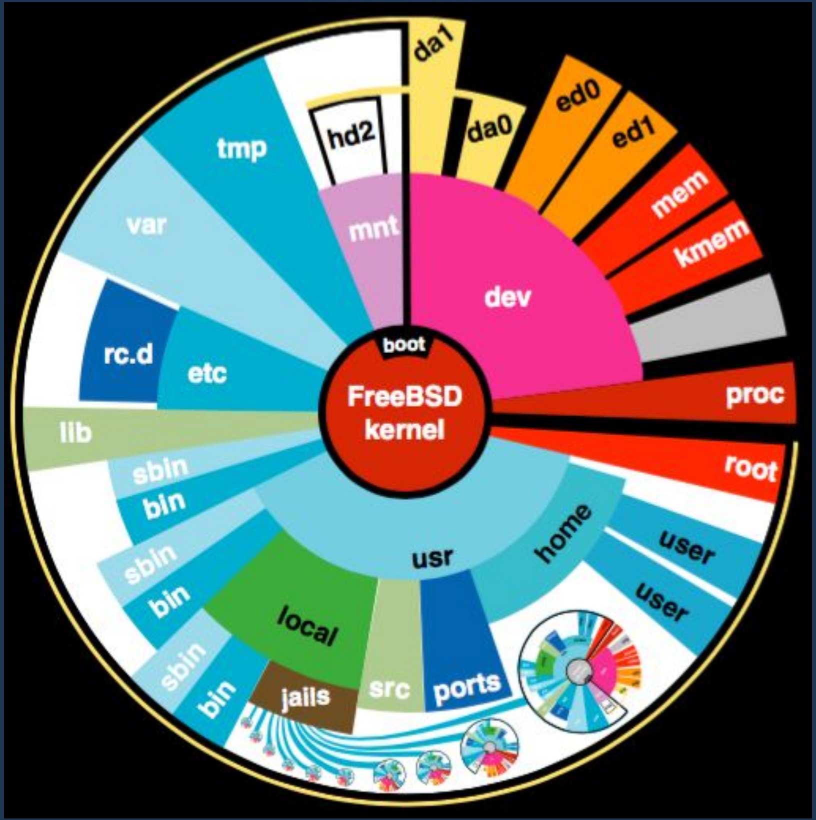
Jails: Confining the omnipotent root

Jails committed to FreeBSD in 1999 by Poul-Henning Kamp (phk)

The FreeBSD ``Jail'' facility provides the ability to partition the operating system environment, while maintaining the simplicity of the UNIX ``root'' model. In Jail, users with privilege find that the scope of their requests is limited to the jail, allowing system administrators to delegate management capabilities for each virtual machine environment. Creating virtual machines in this manner has many potential uses; the most popular thus far has been for providing virtual machine services in Internet Service Provider environments.

source: https://docs.freebsd.org/44doc/papers/jail/jail.html

Jails in a nutshell



Solaris Zones: Operating System Support for Consolidating Commercial Workloads

This paper introduces Solaris Zones (zones), a fully realized solution for server consolidation projects in a commercial UNIX operating system. By creating virtualized application execution environments within a single instance of the operating system, ... On the one hand, a system with multiple workloads needs to run those workloads in isolation, to ensure that applications can neither observe data from other applications nor affect their operation. It must also prevent applications from over-consuming system resources. On the other hand, the system as a whole has to be flexible, manageable, and observable, in order to reduce administrative costs and increase efficiency. By focusing on the support of multiple application environments rather than multiple operating system instances, **zones meets isolation requirements without sacrificing manageability**

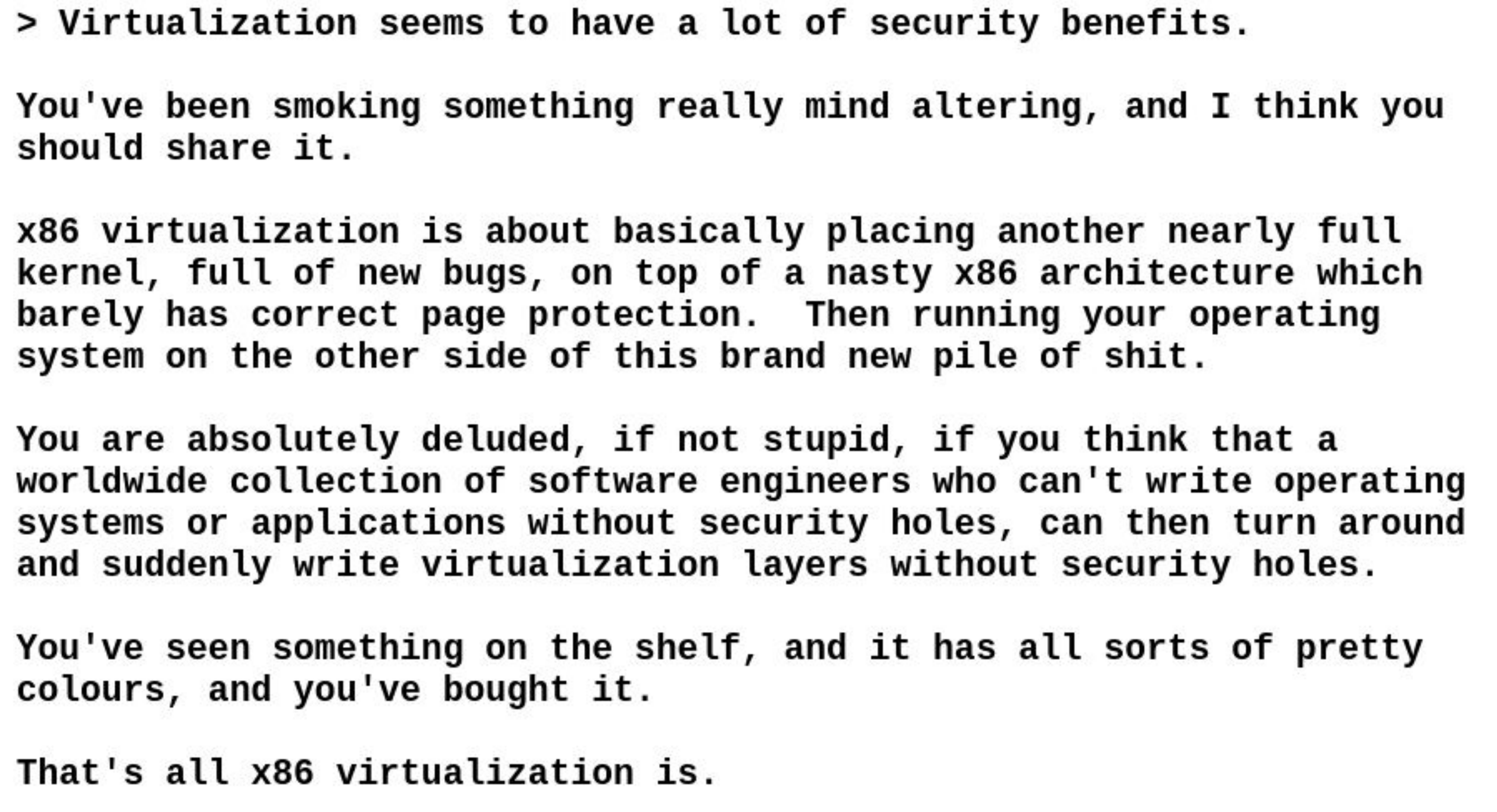
Solaris Zones (cont.)

Consolidation is common in mainframe environments, where technology to support running multiple workloads and even multiple operating systems on the same hardware has been evolving **since the late 1960's**

Hardware-level virtualization

- Existed since the 1960s
- Virtualize hardware: CPU, DRAM, I/O, etc.
- Problem:
  - Operating Systems don't work nicely with respecting resources
- Reality:
  - Hardware-level virtualization is *de facto* in the cloud

Myth: VMs are more secure



> Virtualization seems to have a lot of security benefits.

You've been smoking something really mind altering, and I think you should share it.

x86 virtualization is about basically placing another nearly full kernel, full of new bugs, on top of a nasty x86 architecture which barely has correct page protection. Then running your operating system on the other side of this brand new pile of shit.

You are absolutely deluded, if not stupid, if you think that a worldwide collection of software engineers who can't write operating systems or applications without security holes, can then turn around and suddenly write virtualization layers without security holes.

You've seen something on the shelf, and it has all sorts of pretty colours, and you've bought it.

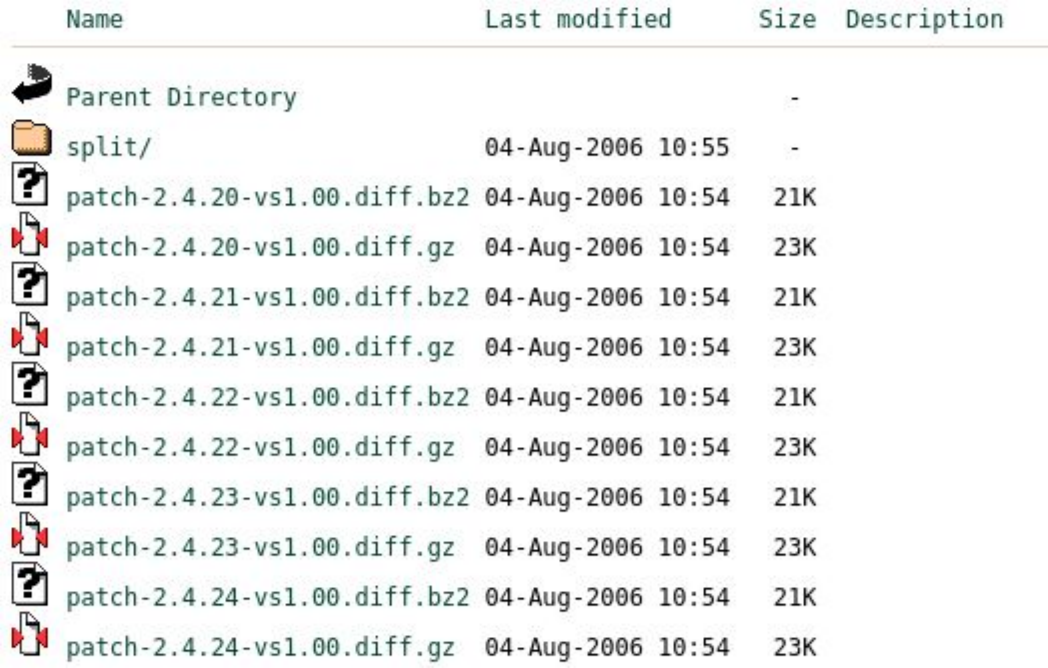
That's all x86 virtualization is.

Container Design Guideline

- Secure
- Isolated
- Integrated
- Manageable
- Transparent

Linux-VServer

- Set of kernel patches
- Implements OS-level virtualization
- Good documentation
- Old but gold



Name	Last modified	Size	Description
Parent Directory		-	
split/	04-Aug-2006 10:55	-	
patch-2.4.20-vs1.00.diff.bz2	04-Aug-2006 10:54	21K	
patch-2.4.20-vs1.00.diff.gz	04-Aug-2006 10:54	23K	
patch-2.4.21-vs1.00.diff.bz2	04-Aug-2006 10:54	21K	
patch-2.4.21-vs1.00.diff.gz	04-Aug-2006 10:54	23K	
patch-2.4.22-vs1.00.diff.bz2	04-Aug-2006 10:54	21K	
patch-2.4.22-vs1.00.diff.gz	04-Aug-2006 10:54	23K	
patch-2.4.23-vs1.00.diff.bz2	04-Aug-2006 10:54	21K	
patch-2.4.23-vs1.00.diff.gz	04-Aug-2006 10:54	23K	
patch-2.4.24-vs1.00.diff.bz2	04-Aug-2006 10:54	21K	
patch-2.4.24-vs1.00.diff.gz	04-Aug-2006 10:54	23K	

OpenVZ

- Almost(?) the same as Linux-VServer
- Released: 2005
- Modern and up-to-date

Linux Containers

- LXC, LXD, LXCFS
- Utilizes Linux Kernel features
  - namespaces
  - cgroups
- Works with mainline kernel
- Released: August 6, 2008; 12 years ago

namespace(7) & cgroups(7)

- A namespace wraps a global system resource in an abstraction that makes it appear to the processes within the namespace that they have their own isolated instance of the global resource.
- Control groups, usually referred to as cgroups, are a Linux kernel feature which allow processes to be organized into hierarchical groups whose usage of various types of resources can then be limited and monitored

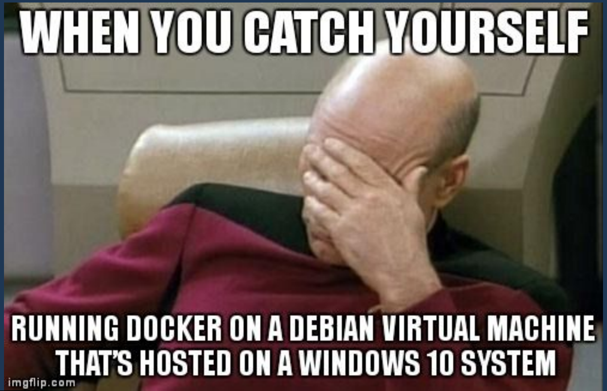
Welcome Docker!

- Released: March 20, 2013; 7 years ago
- Used LXC behind the scenes
- Moved to libcontainer one year later
- Allows developers to think **operationally**
- Encode deployment process via images
- **Docker will do to `apt` what `apt` did to `tar`**

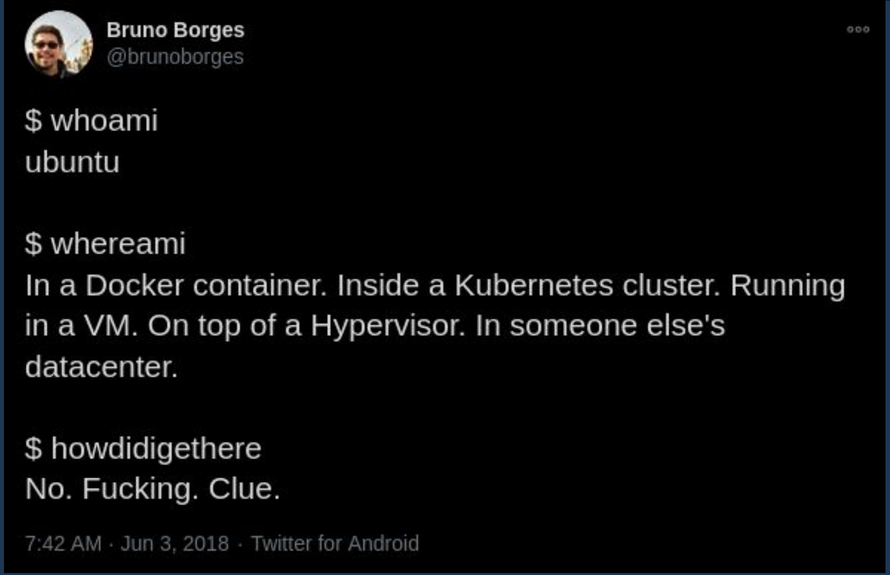
The container revolution

- Docker has shown container to the masses
- Docker's problems are **operational**: network, security, persistency
- **Security issues** is not only from **Docker**, but from **Linux "containers" implementation**
- Deploying OS containers on "Docker hosts" in VMs **negates** all the points

LET THE MEMES BEGIN


WHEN YOU CATCH YOURSELF
RUNNING DOCKER ON A DEBIAN VIRTUAL MACHINE THAT'S HOSTED ON A WINDOWS 10 SYSTEM
imgflip.com

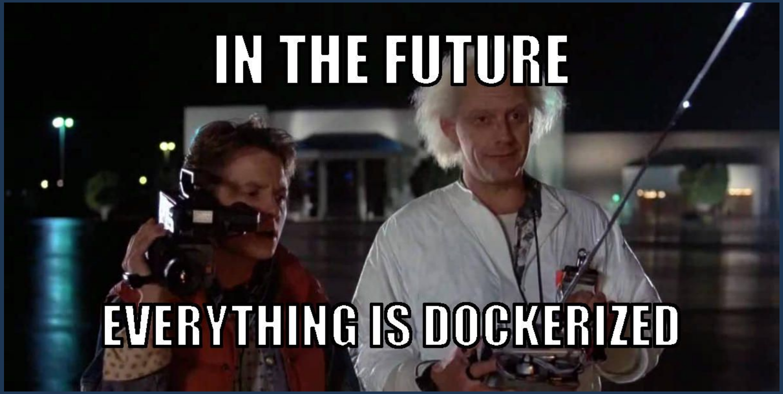
illuria security



Bruno Borges
@brunoborges

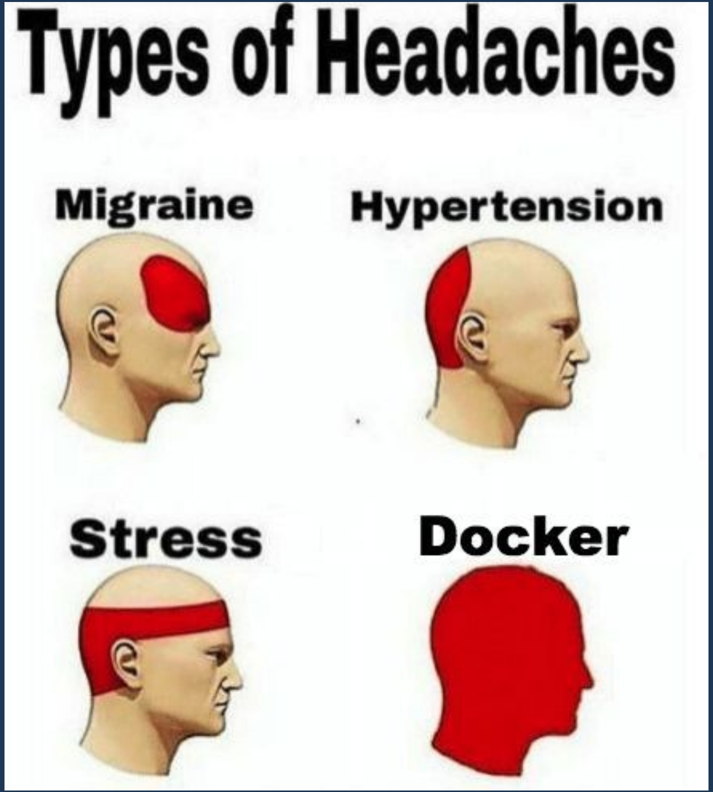
```
$ whoami
ubuntu

$ whereami
In a Docker container. Inside a Kubernetes cluster. Running in a VM. On top of a Hypervisor. In someone else's datacenter.

$ howdidigethere
No. Fucking. Clue.
```

7:42 AM · Jun 3, 2018 · Twitter for Android

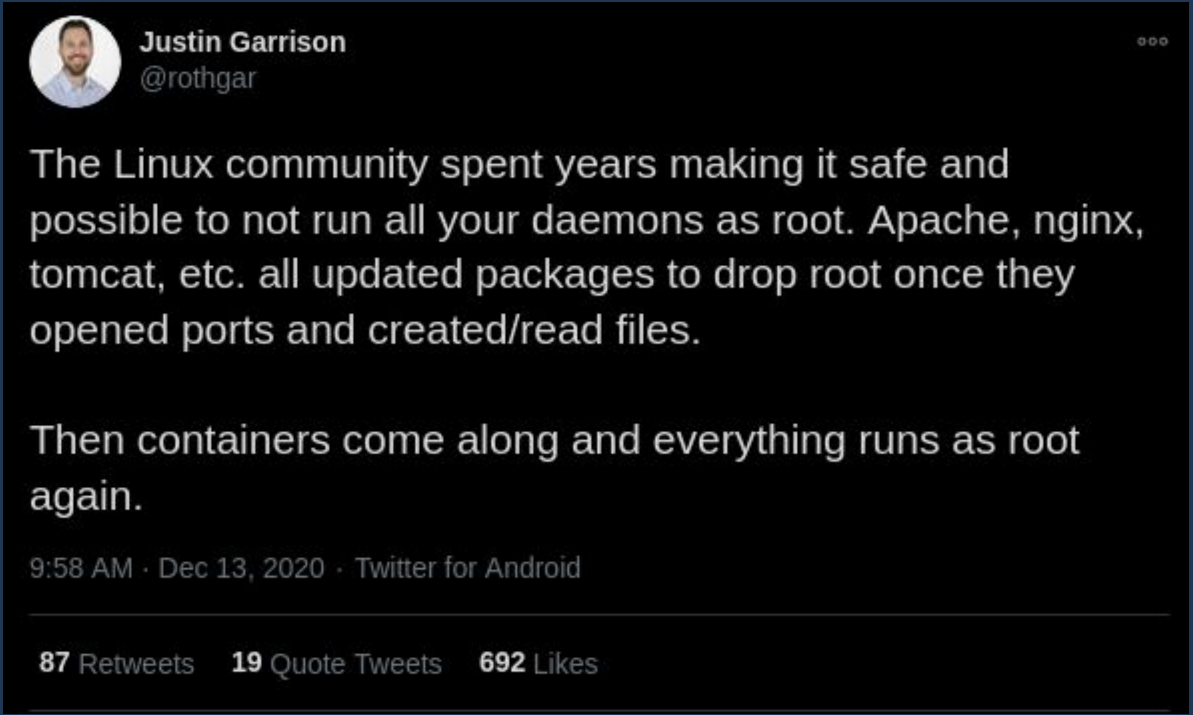


YO DAWG, I HEARD YOU LIKE JENKINS AND DOCKER
SO WE PUT JENKINS IN DOCKER SO YOU CAN RUN JENKINS ON JENKINS AND BUILD A DOCKER IMAGE
imgflip.com

CREATE NEW DEV TOOL
ZERO CONFIG, UP AND RUNNING IN MINUTES
MAKE YOUR LIFE AS A DEV EASIER
10,000 QUESTIONS ON STACK OVERFLOW
imgflip.com


Types of Headaches
Migraine
Hypertension
Stress
Docker

Me watching my senior engineer typing commands into the Unix terminal to fix my docker
Docker prune please



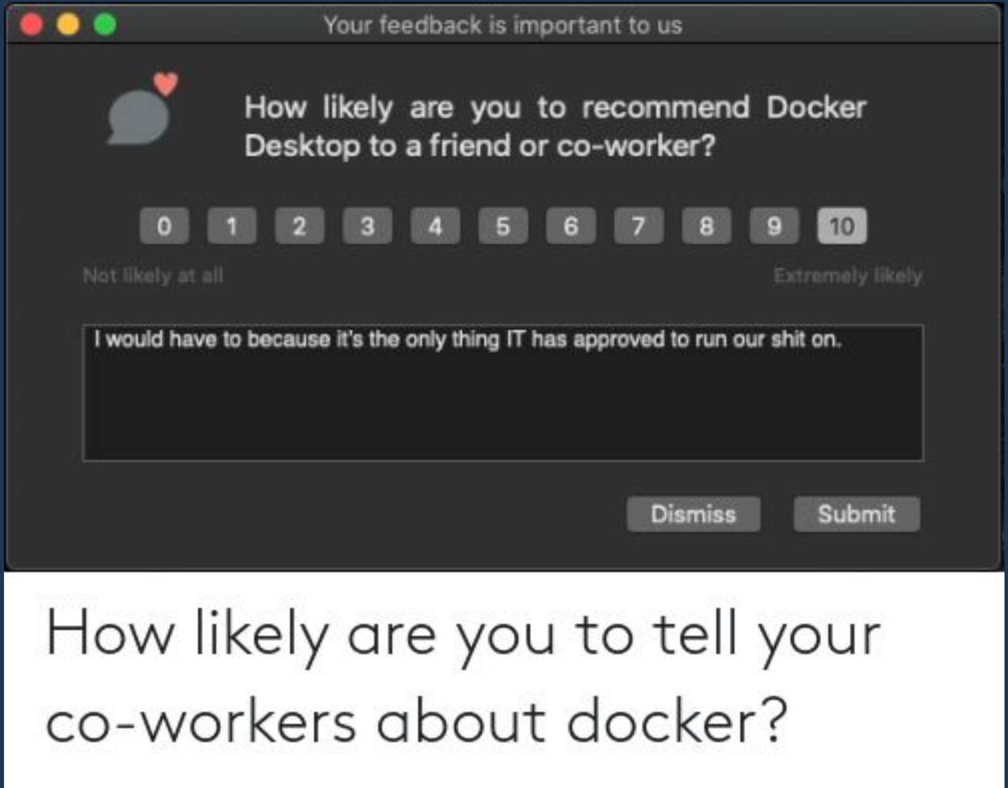
Justin Garrison
@rothgar

The Linux community spent years making it safe and possible to not run all your daemons as root. Apache, nginx, tomcat, etc. all updated packages to drop root once they opened ports and created/read files.

Then containers come along and everything runs as root again.

9:58 AM · Dec 13, 2020 · Twitter for Android

87 Retweets **19** Quote Tweets **692** Likes



Your feedback is important to us

How likely are you to recommend Docker Desktop to a friend or co-worker?

I would have to because it's the only thing IT has approved to run our shit on.

How likely are you to tell your co-workers about docker?

Moral of the story

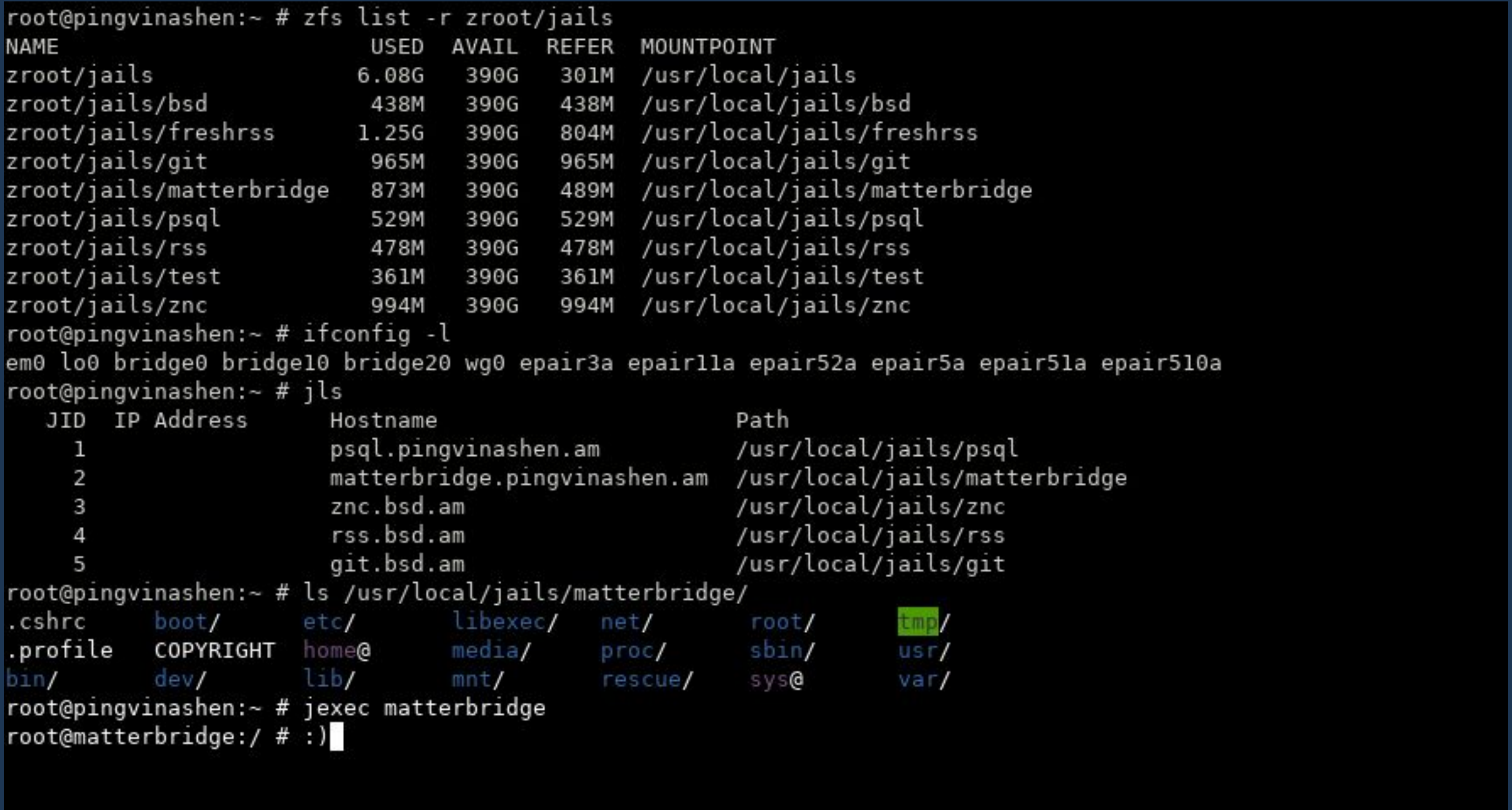
- Stop using VMs. Every time you run a VM, a whale dies in the ocean, god gets angry, aliens refuse to come, kittens die and Baby Yoda gets sad
- VMs have more attack vectors than containers
- Containers are manageable
- Containers are secure; Linux "containers" have issues
- The world is not monoculture, use other solutions
- *"Ok, Antranig, show me alternatives?"*

Alternatives

VMs alternatives; I just want a separate environment

- FreeBSD Jails
    - https://antranigv.am/weblog_en/posts/vnet-jail-howto/
- SmartOS Zones
    - https://wiki.smartos.org/how-to-create-a-zone/

FreeBSD Jails @ $HOME



```
root@pingvinashen:~ # zfs list -r zroot/jails
NAME                       USED  AVAIL  REFER  MOUNTPOINT
zroot/jails               6.08G   390G   301M  /usr/local/jails
zroot/jails/bsd            438M   390G   438M  /usr/local/jails/bsd
zroot/jails/freshrss      1.25G   390G   804M  /usr/local/jails/freshrss
zroot/jails/git            965M   390G   965M  /usr/local/jails/git
zroot/jails/matterbridge   873M   390G   489M  /usr/local/jails/matterbridge
zroot/jails/psql           529M   390G   529M  /usr/local/jails/psql
zroot/jails/rss            478M   390G   478M  /usr/local/jails/rss
zroot/jails/test           361M   390G   361M  /usr/local/jails/test
zroot/jails/znc            994M   390G   994M  /usr/local/jails/znc
root@pingvinashen:~ # ifconfig -l
em0 lo0 bridge0 bridge10 bridge20 wg0 epair3a epair11a epair52a epair5a epair51a epair510a
root@pingvinashen:~ # jls
   JID  IP Address      Hostname                      Path
     1                  psql.pingvinashen.am          /usr/local/jails/psql
     2                  matterbridge.pingvinashen.am  /usr/local/jails/matterbridge
     3                  znc.bsd.am                    /usr/local/jails/znc
     4                  rss.bsd.am                    /usr/local/jails/rss
     5                  git.bsd.am                    /usr/local/jails/git
root@pingvinashen:~ # ls /usr/local/jails/matterbridge/
.cshrc     boot/      etc/       libexec/   net/       root/      tmp/
.profile   COPYRIGHT  home@      media/     proc/      sbin/      usr/
bin/       dev/       lib/       mnt/       rescue/    sys@       var/
root@pingvinashen:~ # jexec matterbridge
root@matterbridge:/ # :)
```

Latest news

- Kubernetes dropping Docker
  - https://www.zdnet.com/article/kubernetes-dropping-docker-is-not-that-big-of-a-deal/
  - https://kubernetes.io/blog/2020/12/02/dont-panic-kubernetes-and-docker/

- Evolving Container Security With Linux User Namespaces
  - https://netflixtechblog.com/evolving-container-security-with-linux-user-namespaces-afbe3308c082

illuria security

That's all folks!

Thanks

Q&A

a@illuriasecurity.com

https://antranigv.am/