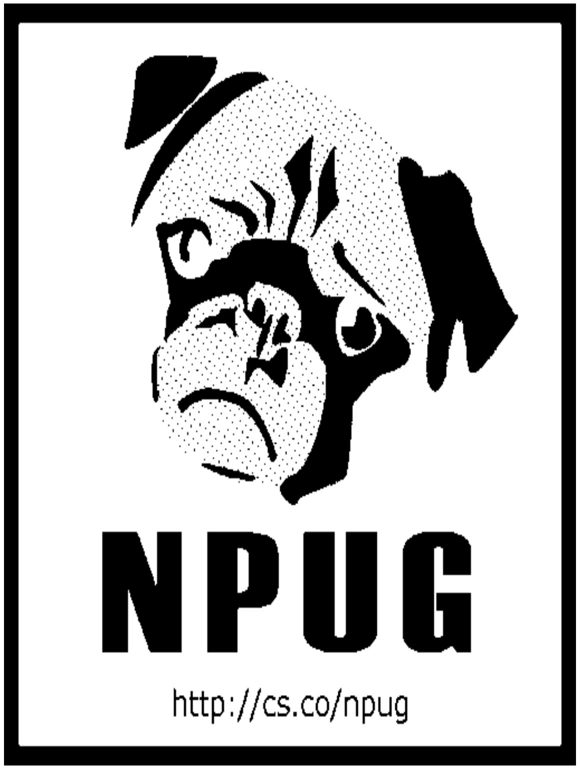
An Introduction to Networking with Linux and Linux Containers (LXC)

Travis P. Bonfigli – Technology Evangelist

Network Programmability User's Group (NPUG)

- User's Group focused on Network Programmability Topics
- Diverse Group of Users
- Combination of User Stories and Educational Material
- Recorded Meetings and Active Wiki

http://cs.co/NPUG



Agenda

- Introduction
- UNIX/Linux Networking History
- Linux Networking Files
- Linux Networking Utilities
- Linux Containers (LXC)
- Networking with LXCs
- Launching our Apache Web Server LXC
- Additional Resources

UNIX/Linux Networking History

- Bell Labs (AT&T) UNIX and UUCP at 300 baud
- DEC, Intel, and Xerox (DIX) Ethernet
- ARPANET and TCP/IP
- UC Berkeley BSD UNIX Version 4
- Columbia University and Kermit
- The Router Discovery Routing Daemon (routed)
- Gateway Routing Daemon (gated)

Linux Networking Files

- /etc/hosts
  - Plaintext file that was the de facto host lookup source prior to DNS and still used today for a variety of use cases
- /etc/resolv.conf
  - Plaintext file that is used as the resolver configuration file providing DNS customization (located in /etc/resolvconf/resolv.conf.d)
- /etc/nsswitch.conf
  - Plaintext file used to order the sources of name-service information and commonly referred to as the “Name Service Switch” configuration file
- /etc/network/interfaces
  - The Ubuntu network interface configuration file used to customize your system’s network settings
- /var/lib/lxc/<your-container-name>/config
  - The LXC configuration file used to customize the network settings for your container

Sample /etc/hosts File

```
[tpb@deathstar]:[3.2.25]:[17:13:53]:[~]:[6978]$: cat /etc/hosts

# Do not remove the following line, or various programs
# that require network functionality will fail.
127.0.0.1       deathstar.umd.edu ds localhost.localdomain localhost
#::1            localhost6.localdomain6 localhost6

10.10.0.1     pix # PIX firewall address
10.10.0.2     router # New router
10.0.5.104    mail mail.umd.cc.md.us mail.umd.edu umd-mailcla.umd.cc.md.us

# 10.10.0.3 - 10.10.0.20  (server block)

10.10.0.3     enterprise enterprise.umd.edu enterprise.umd.cc.md.us
10.10.0.5     cygnus.umd.edu cygnus
10.10.0.6     vader.umd.edu vader vader.umd.cc.md.us

10.10.0.8     Severn # DC for BCTS2
10.10.0.9     Potomac # Web server
10.10.0.10    Chester # DC for BCTS2 and DNS Server
10.10.0.11    Magothy # Remote software
10.10.0.12    Choptank # Remote software
##
## End of file...
```

Sample /etc/resolv.conf File

```
[tpb@deathstar]:[3.2.25]:[17:43:50]:[~]:[6988]$: cat /etc/resolv.conf

domain umd.edu
search students.umd.edu faculty.umd.edu
nameserver 10.10.0.31
Nameserver 10.10.0.32
Nameserver 10.10.0.3
```

Sample /etc/nsswitch.conf File

```
# To use db, put the "db" in front of "files" for entries you want to be
# looked up first in the databases
#
# Example:
#passwd:    db files nisplus nis
#shadow:    db files nisplus nis
#group:     db files nisplus nis

passwd:     files
shadow:     files
group:      files

#hosts:     db files nisplus nis dns
hosts:      files dns

# Example - obey only what nisplus tells us...
#services:   nisplus [NOTFOUND=return] files
#networks:   nisplus [NOTFOUND=return] files
#protocols:  nisplus [NOTFOUND=return] files
#rpc:        nisplus [NOTFOUND=return] files
#ethers:     nisplus [NOTFOUND=return] files
#netmasks:   nisplus [NOTFOUND=return] files
```

Sample /etc/network/interfaces File

```
# interfaces(5) file used by ifup(8) and ifdown(8) and ifdown(8)

##
## The auto and iface statements for the Loopback address...
auto lo
iface lo inet loopback

##
## The auto and iface statements for a simple static configuration...
auto eth0
iface eth0 inet static
        address 192.168.1.34
        gateway 192.168.1.254
        netmask 255.255.255.0
        network 192.168.1.0
        broadcast 192.168.1.255
##
## Sample configuration for a simple DHCP setup...
#iface eth0 inet dhcp

##
## The auto and iface statements for a simple IPv6 Global Unicast address...
iface eth0 inet6 static
        pre-up modprobe ipv6
        address 2001:db8:acad::34
        netmask 64
        gateway 2001:db8:acad::254
##
## The auto and iface statements for a simple IPv6 Link Local address...
iface eth0 inet6 static
        pre-up modprobe ipv6
        address fe80::34
        netmask 64
        gateway fe80::254
##
## End of file...
```

Sample LXC Config File

```
travispbonfigli@ubuntu:~$ sudo cat /var/lib/lxc/ubu-002/config

# Template used to create this container: /usr/share/lxc/templates/lxc-ubuntu
# Parameters passed to the template:
# For additional config options, please look at lxc.container.conf(5)

# Common configuration
lxc.include = /usr/share/lxc/config/ubuntu.common.conf

# Container specific configuration
lxc.rootfs = /var/lib/lxc/ubu-002/rootfs
lxc.utsname = ubu-002
lxc.arch = amd64

# Network configuration
lxc.network.type = macvlan
lxc.network.macvlan.mode = bridge
lxc.network.link = eth0
lxc.network.flags = up
lxc.network.ipv4 = 192.168.1.88/24 192.168.1.255
lxc.network.ipv4.gateway = 192.168.1.254
##
## End of file…
```

Linux Networking Utilities

- /sbin/ifconfig & /sbin/ip
  - Interface configuration utilities
- /usr/bin/dig
  - Domain Name Service (DNS) lookup tool
- /bin/ping & /bin/ping6
- /usr/sbin/traceroute & /usr/sbin/traceroute6
- /usr/bin/nslookup
  - Internet name server query tool
- /bin/netstat
  - Network connection and stauts utility
- /usr/bin/mtr
  - Originally known as “Matt’s Traceroute”, mtr combines ping and traceroute for some outstanding network information

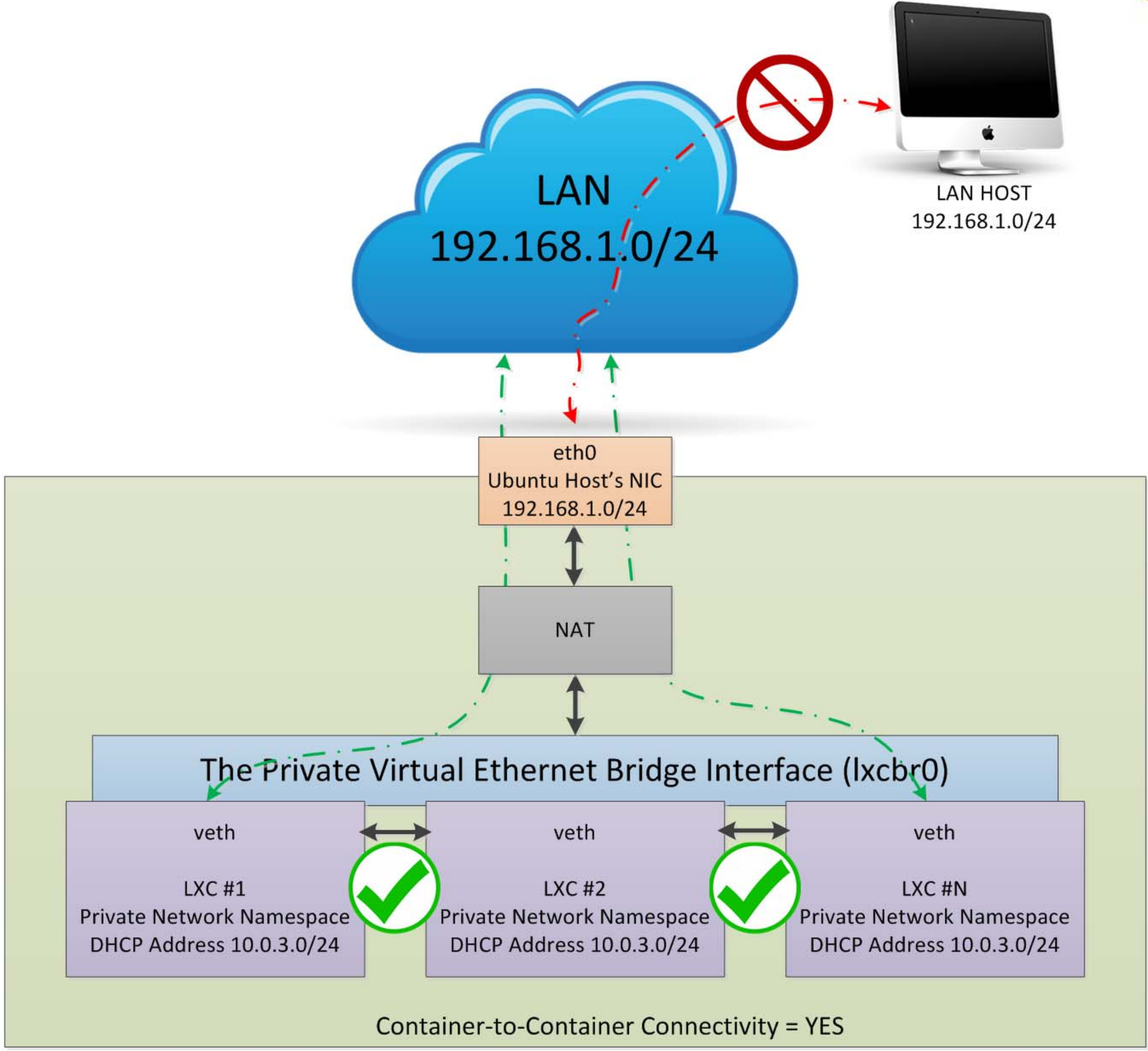
Linux Containers (LXC)

- Linux Containers (LXC)
  - LXC vs. Docker vs. Rocket
  - LXC is not VMware
  - LXC Commands (apropos lxc) & man pages
- Obtaining LXC
  - Ubuntu (sudo apt-get install lxc)
- Creating Your First LXC
  - Ubuntu (sudo lxc-create –B aufs –n ubu-lxc-001 –t ubuntu)
- Starting Your LXC
  - Ubuntu (sudo lxc-start –n ubu-lxc-001 –d)
- Listing Your LXCs
  - Ubuntu (sudo lxc-ls --fancy)
- Stopping Your LXC
  - Ubuntu (sudo lxc-stop –n ubu-lxc-001)
- Destroying Your LXC
  - Ubuntu (sudo lxc-destroy –n ubu-lxc-001)

Networking & LXC
The Default Configuration

- LXC Host Network Configuration File
  - /etc/network/interfaces
- LXC Container Network Configuration Types
  - none
  - empty
  - phys
  - veth
  - vlan
  - macvlan (private is default – Virtual Ethernet Port Aggregator/VEPA)
- LXC Container Default Network Configuration
  - NAT & DHCP (10.0.3.0/24 subnet)
  - veth Interface
  - Name Your veth Interfaces in LXC Container Configuration File
    - Ubuntu (lxc.network.veth.pair = ubu-001-veth)
- Connectivity to NATed LXCs Limited by iptables
  - Ubuntu (sudo iptables --list rules)

LXC Networking with “veth” DHCP, & NAT (the default)



Networking & LXC
Static IPs with a Bridge

- LXC Static Address Configuration Use Case
  - Create a bridge interface on HOST /etc/network/interfaces
  - Edit the /var/lib/lxc/<your-container-name>/config file on the HOST

- A Sample /etc/network/interfaces File:

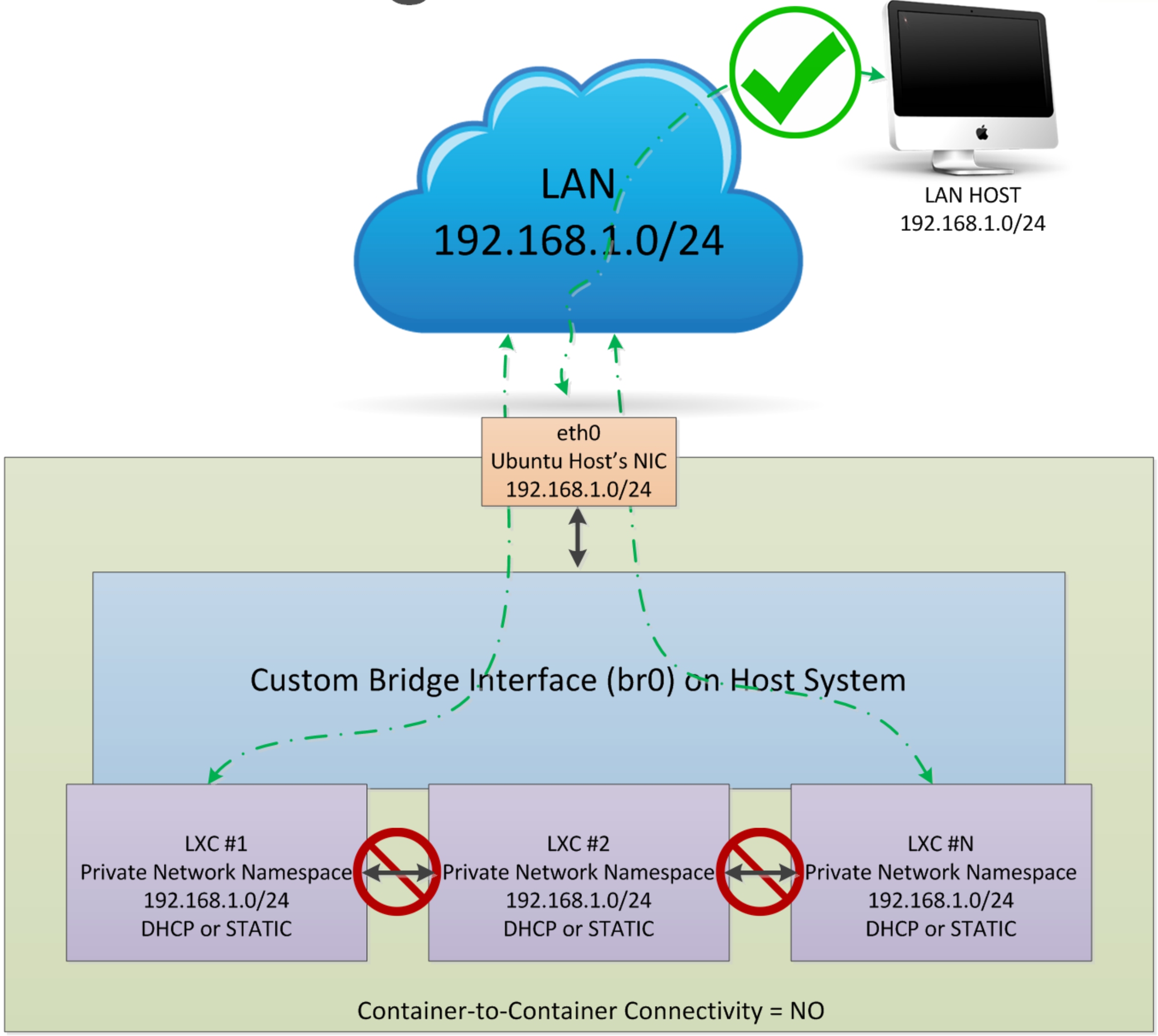
```
travispbonfigli@ubuntu:~$ cat /etc/network/interfaces
# interfaces(5) file used by ifup(8) and ifdown(8)
auto lo
iface lo inet loopback

auto mybridge001
iface mybridge001 inet dhcp
             bridge_ports eth0
             bridge_stp off
             bridge_fd 0
             bridge_maxwait 0

#Sample /var/lib/lxc/ubu-003/config file which works with the /etc/network/interfaces

# Network configuration
lxc.network.type = veth
lxc.network.flags = up
lxc.network.link = mybridge001
lxc.network.ipv4 = 192.168.1.121/24 192.168.1.255
lxc.network.ipv4.gateway = 192.168.1.254
lxc.network.veth.pair = ubu-003-veth
```

LXC Networking with Static IPs and Bridge Interfaces



LAN
192.168.1.0/24

LAN HOST
192.168.1.0/24

eth0
Ubuntu Host's NIC
192.168.1.0/24

Custom Bridge Interface (br0) on Host System

LXC #1
Private Network Namespace
192.168.1.0/24
DHCP or STATIC

LXC #2
Private Network Namespace
192.168.1.0/24
DHCP or STATIC

LXC #N
Private Network Namespace
192.168.1.0/24
DHCP or STATIC

Container-to-Container Connectivity = NO

Networking & LXC MACVLAN (Private)

- LXC MACVLAN (Private) Use Case
  - Create bridge interface on HOST or use previously created bridge interface if one already exists
  - Edit the /var/lib/lxc/<your-container-name>/config file on the HOST
  - MACVLAN (Private) LXCs do not allow HOST access to the container

- A Sample /etc/network/interfaces file:

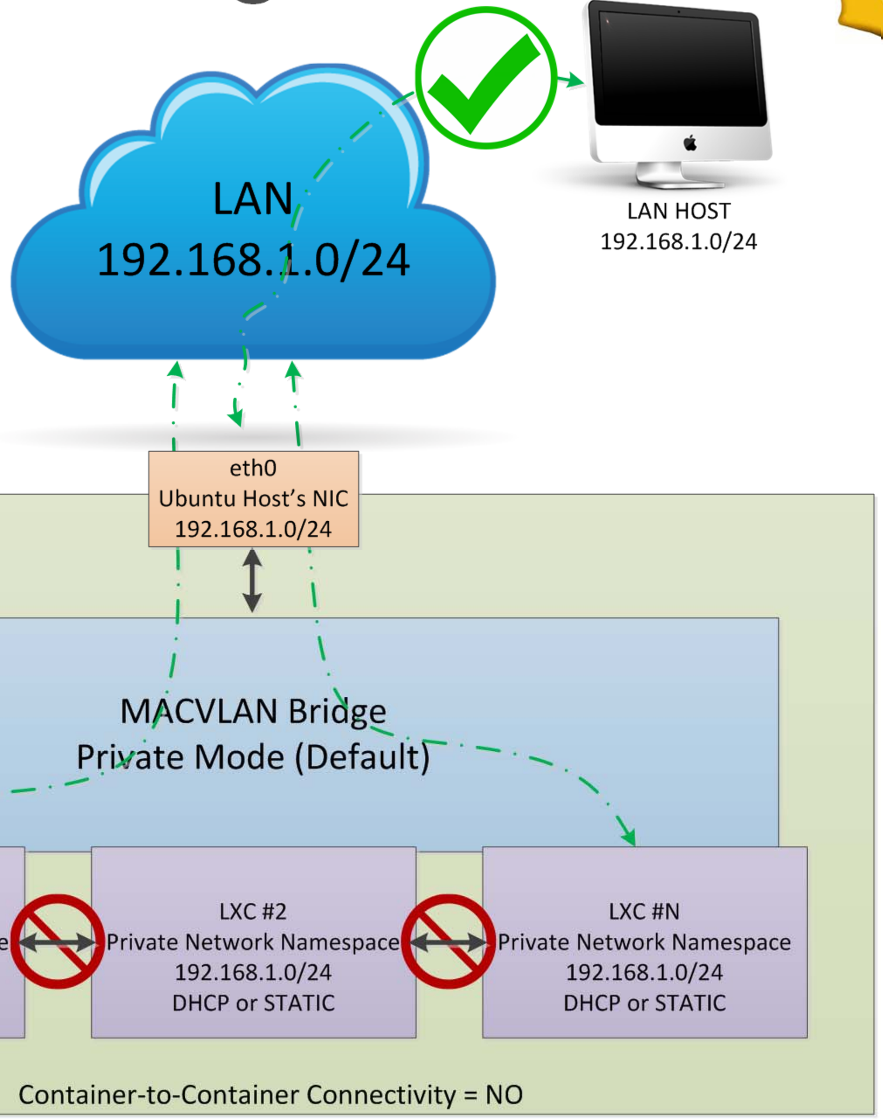
```
travispbonfigli@ubuntu:~$ cat /etc/network/interfaces
# interfaces(5) file used by ifup(8) and ifdown(8)
auto lo
iface lo inet loopback

auto mybridge001
iface mybridge001 inet dhcp
             bridge_ports eth0
             bridge_stp off
             bridge_fd 0
             bridge_maxwait 0

#Sample /var/lib/lxc/ubu-005/config file which works with the /etc/network/interfaces file
above

# Network configuration
lxc.network.type = macvlan
lxc.network.macvlan.mode = bridge
lxc.network.link = mybridge001
lxc.network.flags = up
lxc.network.ipv4 = 192.168.1.66/24 192.168.1.255
lxc.network.ipv4.gateway = 192.168.1.254
```

LXC Networking with MACVLAN Bridge Interfaces



Networking & LXC MACVLAN (VEPA)

- LXC MACVLAN (VEPA) Use Case
  - Create bridge interface on HOST or use previously created bridge interface if one already exists
  - Edit the /var/lib/lxc/<your-container-name>/config file on the HOST
  - MACVLAN (VEPA) LXCs do not allow any other containers to access them including the HOST system

- A Sample /etc/network/interfaces file:

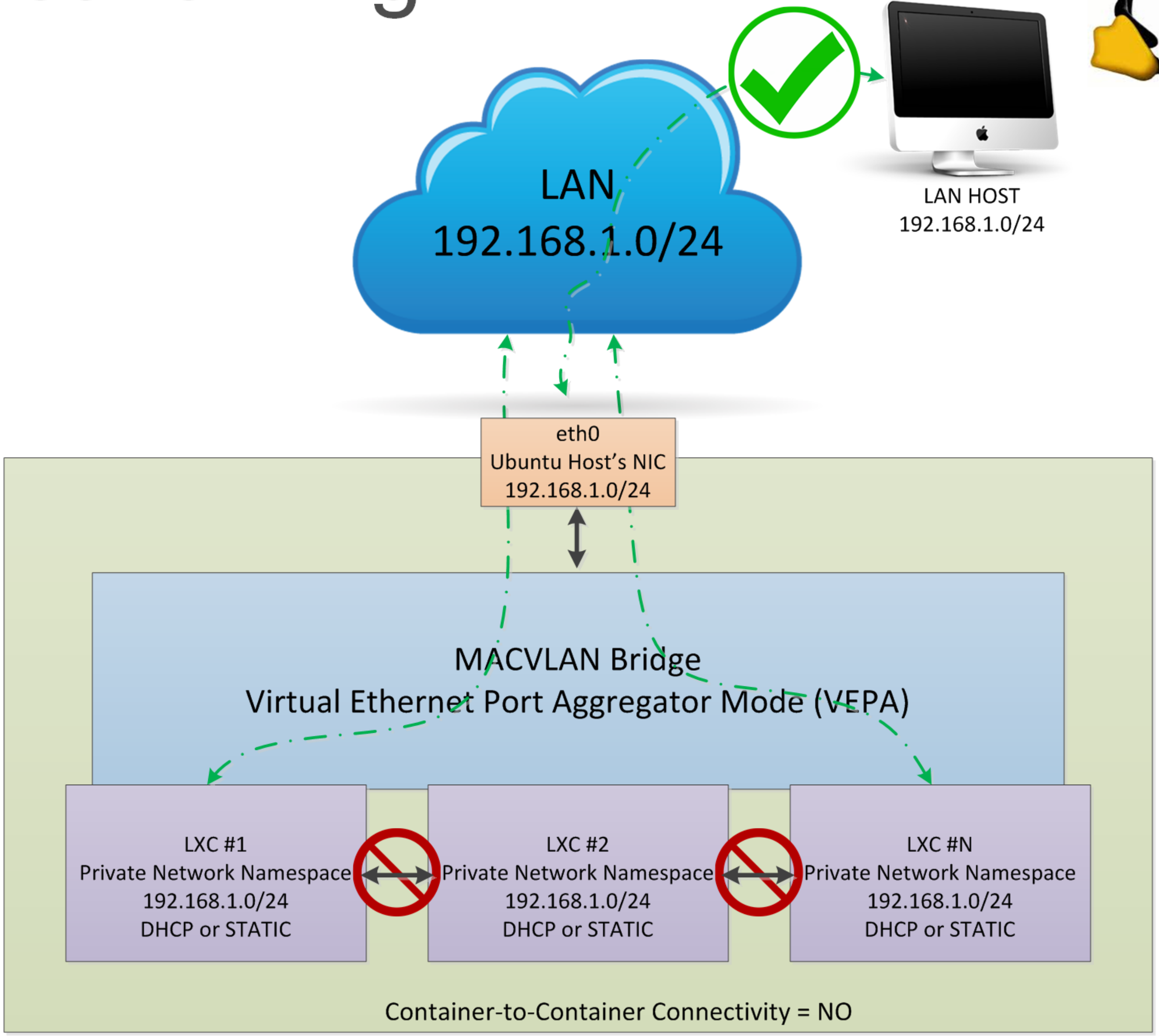
```
travispbonfigli@ubuntu:~$ cat /etc/network/interfaces
# interfaces(5) file used by ifup(8) and ifdown(8)
auto lo
iface lo inet loopback

auto mybridge001
iface mybridge001 inet dhcp
             bridge_ports eth0
             bridge_stp off
             bridge_fd 0
             bridge_maxwait 0

#Sample /var/lib/lxc/ubu-005/config file which works with the /etc/network/interfaces

# Network configuration
lxc.network.type = macvlan
lxc.network.macvlan.mode = vepa
lxc.network.link = mybridge001
lxc.network.flags = up
lxc.network.ipv4 = 192.168.1.66/24 192.168.1.255
lxc.network.ipv4.gateway = 192.168.1.254
```

LXC MACVLAN VEPA Networking



LAN

192.168.1.0/24

LAN HOST

192.168.1.0/24

eth0

Ubuntu Host's NIC

192.168.1.0/24

MACVLAN Bridge

Virtual Ethernet Port Aggregator Mode (VEPA)

LXC #1

Private Network Namespace

192.168.1.0/24

DHCP or STATIC

LXC #2

Private Network Namespace

192.168.1.0/24

DHCP or STATIC

LXC #N

Private Network Namespace

192.168.1.0/24

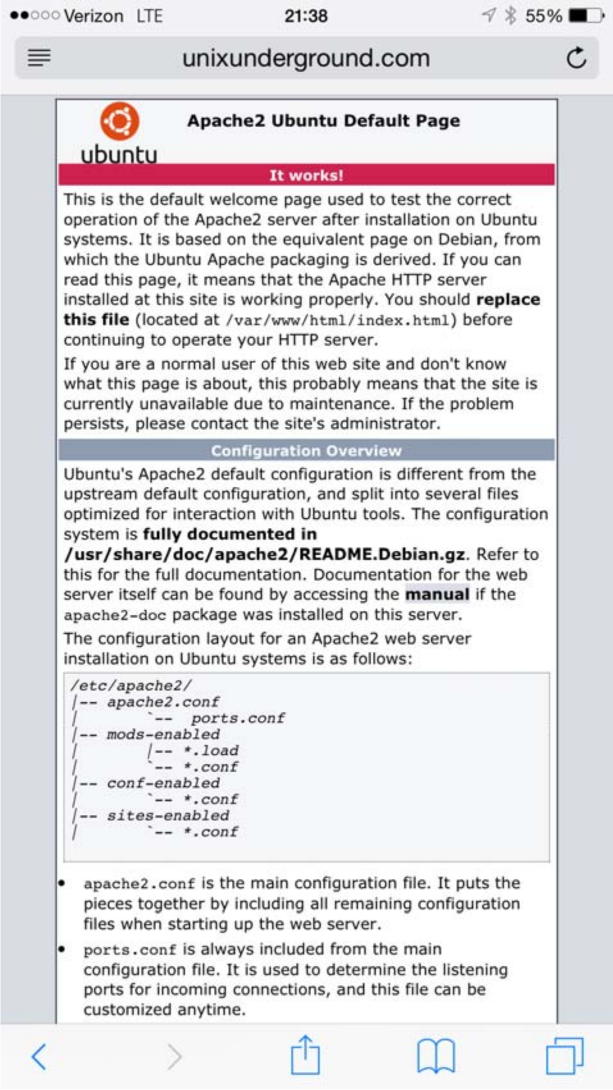
DHCP or STATIC

Container-to-Container Connectivity = NO

Let’s Create An Apache Web Server

- Create a new Ubuntu LXC
  - sudo lxc-create –B aufs –n ubu-099 –t ubuntu
- Configure Your LXC Networking (we will use MACVLAN)
  - Edit /var/lib/lxc/ubu-099/config file
- Start the LXC
  - sudo lxc-start –n ubu-099
- Connect to the LXC Console
  - sudo lxc-console –n ubu-099
- Install the Apache Web Server Software
  - sudo apt-get install apache2
- Make Port Forwarding/Firewall/index.html Modifications
- Take a look!
  - Visit www.unixunderground.com !!!

Success!!!



ubuntu

Apache2 Ubuntu Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is **fully documented in /usr/share/doc/apache2/README.Debian.gz**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:

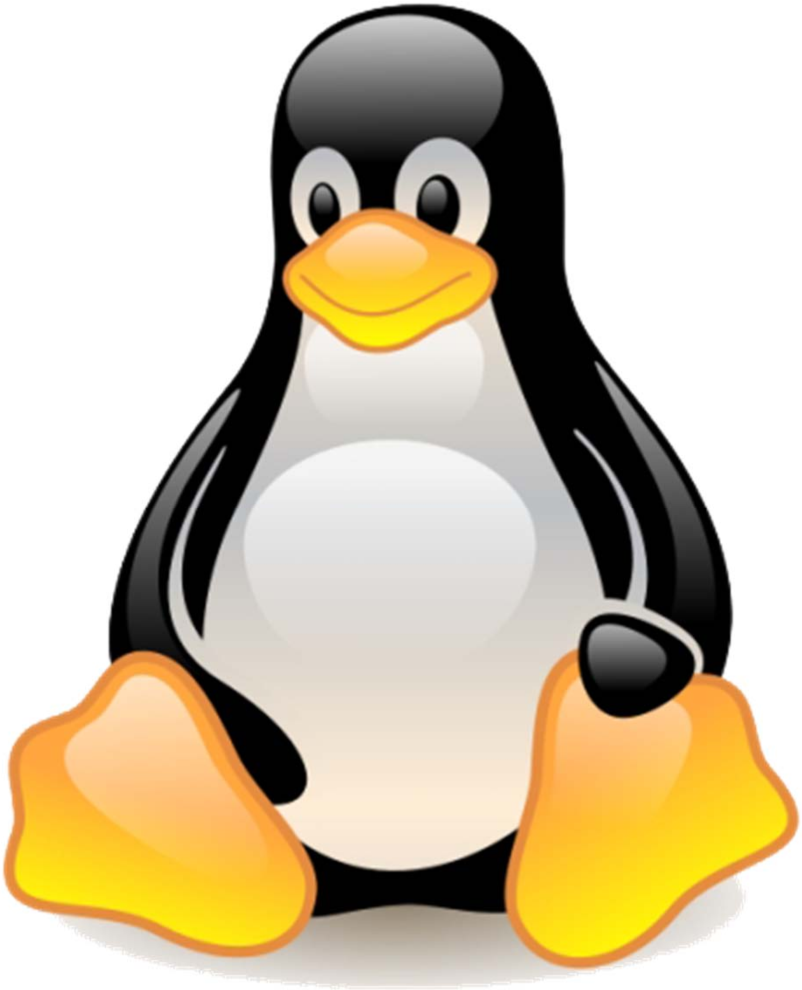
```
/etc/apache2/
|-- apache2.conf
|       `--  ports.conf
|-- mods-enabled
|       |-- *.load
|       `-- *.conf
|-- conf-enabled
|       `-- *.conf
|-- sites-enabled
|       `-- *.conf
```

- `apache2.conf` is the main configuration file. It puts the pieces together by including all remaining configuration files when starting up the web server.
- `ports.conf` is always included from the main configuration file. It is used to determine the listening ports for incoming connections, and this file can be customized anytime.

Additional Resources

- The Official Linux Containers Home Page:
  - https://linuxcontainers.org/
- The Official Ubuntu Documentation Home Page:
  - https://help.ubuntu.com/
- Stephane Graber's Official Blog:
  - https://www.stgraber.org/2013/12/20/lxc-1-0-blog-post-series/
- The Official GitHub LXC – Linux Containers Project Page:
  - https://github.com/lxc
- The Official Ubuntu Documentation Home Page:
  - https://help.ubuntu.com/
- Ubuntu Release-Specific Network Configuration Example Files:
  - Multiple network configuration examples can be found in the plaintext files located in the directory /usr/share/doc/lxc/examples

Thank You for Your Time!!!



Network Programmability User's Group (NPUG)

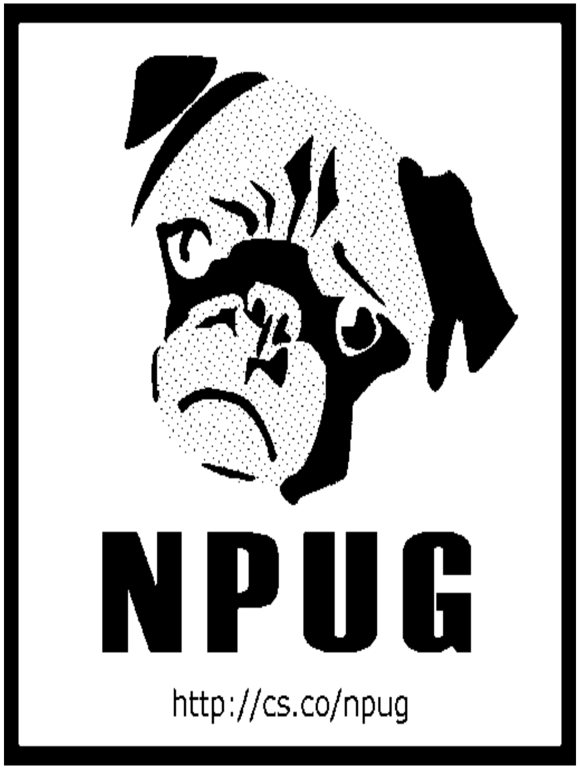
To hear more:

Check the NPUG wiki
http://cs.co/npug

Join the Mailer for Updates and Recordings
http://cs.co/npugmailer

View Previous Recordings
http://cs.co/npughistory

Feedback – good/bad, topics, speakers, help out:
Npug-advisory@googlegroups.com



http://cs.co/NPUG