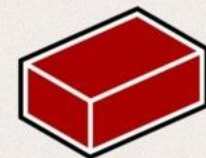


Networking with Pis

mmmmm... pie

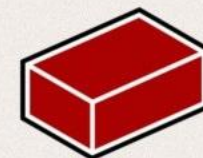


Redbrick

DCU's Networking Society

Computer Networks

- Computer network consists of two or more computers (or devices) connected together
- Computers have a unique IP address that allows other computers to find them and send data to them

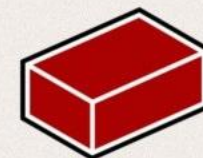


Fancy words we use

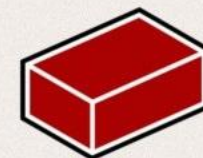
- Server: centralised resource that other computers connect to.
- Client: The computers that connect to the server
- Computer Network: The collection of servers and clients

Fancy words we use

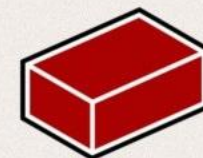
- IP address: The address of the devices on the network
- Static IP: This is an address that does not change
- LAN: Local area network, Network in a limited area



- Before the Raspberry Pis can communicate they need to be connected together via a network.
- Normally, when a device connects to a network, it is assigned a unique identifier called an IP address.
- We have to give each Raspberry Pi its own IP address.



Assigning IP addresses



Redbrick
DCU's Networking Society

Static IP addresses

- Usually when you connect the raspberry pi to a LAN it is assigned a unique identifier called an IP address. (Typically it will change everytime you connect)
- Sometimes we don't want this, if it is a once off project or a small contained network

Setup

Edit the file located at:

- `/etc/dhcpd.conf`

Add these lines to the end of the file of the first pi

- (Next Slide)


```
interface eth0
```

```
static ip_address=192.168.0.2/24
```

```
static routers=192.168.0.1
```

```
static domain_name_servers=192.168.0.1
```

Or 3 if you are the second
in your group

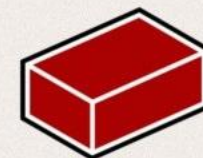
```
interface wlan0
```

```
static ip_address=192.168.0.2/24
```

```
static routers=192.168.0.1
```

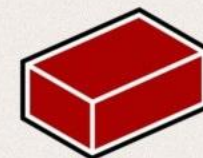
```
static domain_name_servers=192.168.0.1
```

Or 3 if you are the second
in your group

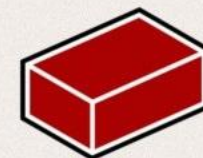


Redbrick
DCU's Networking Society

- To Test we reboot by typing
 - shutdown -r now
- and enter this command:
 - “ip a”
- We should see the ip we set in the config

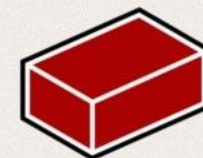


Connect the two pis together with
the ethernet cable



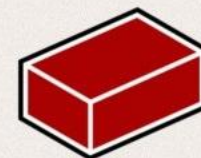
Redbrick
DCU's Networking Society

- After setting up the second pi we will ping the first to see if they can communicate :
- There is a known bug with the ping in pi so we have to first do:
 - `sudo chmod u+s /bin/ping`
- Then we can:
 - `ping 192.168.0.2`



Setting up a network

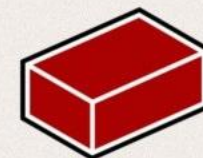
And using python code to send messages between them



Redbrick
DCU's Networking Society

Pro tip

Don't forget which is which



Redbrick
DCU's Networking Society

Setting up a chat room

- Make a file called chat.py
 - and copy what follows
 - (Can find the code on github.com/redbrick)

```
import network
import sys

def heard(phrase):
    print('them:' + phrase)

if (len(sys.argv) >= 2):
    network.call(sys.argv[1], whenHearCall=heard)
else:
    network.wait(whenHearCall=heard)

while network.isConnected():
    phrase = input()
    print('me:' + phrase)
    network.say(phrase)
```

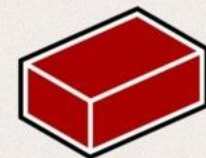
~

~

Activating the chat

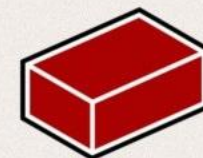
- We are setting the first pi up as the server: (192.168.0.3)
 - `python3 chat.py`
- The second pi is the client
 - `python3 chat.py 192.168.0.3`

- And now we're connected and talking



Possible Problems

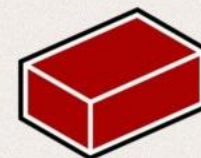
Try send a message at the same time



Redbrick
DCU's Networking Society

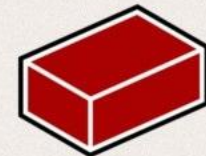
Dynamic Host Configuration Protocol

- It is more than likely clear that changing `/etc/network/interfaces` every time is not a nice solution
- Imagine if we had far more computers to add to the network
- Imagine phones and laptops that move around and regularly join and disconnect from the networks
- We also have to make sure that no device has the same address



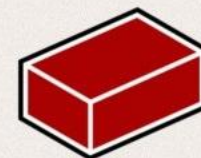
What can we do to make it better?

- We can delegate one of our pis to be a dhcp server
- We can then get the other pis to get their addresses from that pi

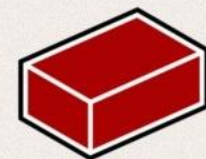


How does it do the magic?

- Imagine a computer comes on the network
- It yells out to the whole network to find out if there is a dhcp server
- The server then comes back and offers an address
- The client computer then accepts the address and takes note that it has given out that address



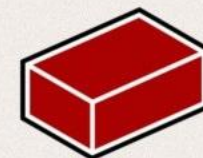
Who wants to be a server?



Redbrick
DCU's Networking Society

This is for you

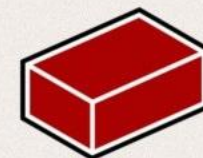
- `sudo apt-get install dnsmasq`



Redbrick
DCU's Networking Society

We need to give you a static ip

```
sudo nano /etc/network/interfaces
```

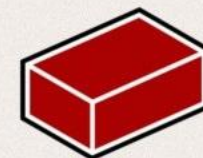


Redbrick
DCU's Networking Society

Put this in below it

```
auto eth0  
iface eth0 inet static  
address 192.168.0.1  
netmask 255.255.255.0
```

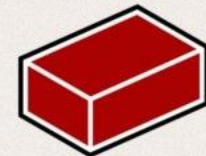
Then save and quit (ctrl-o) enter (ctrl-x)



Redbrick
DCU's Networking Society

Then we restart the network

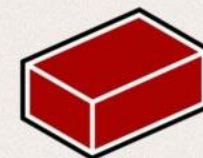
- `sudo systemctl restart networking`



Now it will always have the address

192.168.0.1

check this with ifconfig
the address should be after
inet addr



Redbrick
DCU's Networking Society

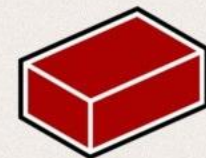
We need to configure the DHCP server

We are going to specify a config file for dnsmasq
....And make a backup in case we screw up

```
cd /etc
```

```
sudo mv dnsmasq.conf dnsmasq.default ← the backup
```

```
sudo nano dnsmasq.config
```

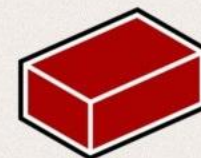


In the blank file you should...

```
interface=eth0
```

```
dhcp-range=192.168.0.10,192.168.0.254,255.255.255.0,12h
```

- First line tells dnsmasq to listen for DHCP requests on ethernet port of pi
- The rest specifies the range of addresses that can be given out
- The 12h at the end specifies how long the lease of ips are



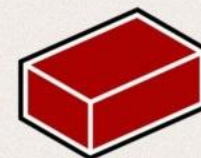
Now type this in terminal

```
sudo dnsmasq -C /etc/dnsmasq.config
```

Then we restart the network

- `sudo systemctl restart networking`

That is the server setup



Redbrick
DCU's Networking Society

On the remaining pis

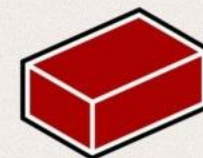
We need to check /etc/network/interfaces

```
sudo nano /etc/network/interfaces
```

Ensure that a static ip is **NOT** set, The line

```
iface eth0 inet dhcp
```

should not be commented (no # in front)



Redbrick
DCU's Networking Society

Restart

- `sudo systemctl restart networking`

Check it is working

- With the command:
 - ifconfig
- We can see that every time that we restart the pis we get a new ip address.

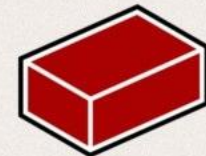
Now back to the chat program

Server pi:

```
python3 chat.py
```

Everyone else:

```
python3 chat.py 192.168.0.1
```



Redbrick

DCU's Networking Society

Helpful Links

Most of the info for this talk was resourced from:

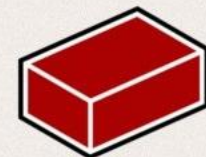
<https://www.raspberrypi.org/resources/>

The code for the chat is on github at:

<https://github.com/redbrick/HelpdeskTalks>

And along with all helpdesk talks the slides are at:

<https://www.redbrick.dcu.ie/~edu>

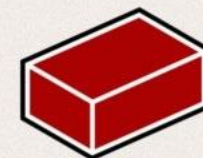


Redbrick
DCU's Networking Society

Questions?



Cat tax



Redbrick
DCU's Networking Society