How the Internet works

Director

python SOFTWARE FOUNDATION

Organizer

python™ boston

@jessicamckellar

http://jesstess.com

You type http://python.org into your browser bar and press Enter. What happens?

python™

Advanced Search

ABOUT
NEWS
DOCUMENTATION
DOWNLOAD
下载
COMMUNITY
FOUNDATION
CORE DEVELOPMENT

Help

Package Index

Quick Links (2.7.3)

» Documentation
» Windows Installer
» Source Distribution

Quick Links (3.3.0)

» Documentation

Python Programming Language – Officia

Python is a programming language that lets you work more quickly and integrate your systems more effectively. You can learn to use Python and see almost immediate gains in productivity and lower maintenance costs.

Python runs on Windows, Linux/Unix, Mac OS X, and has been ported to the Java and .NET virtual machines.

Python is free to use, even for commercial products, because of its OSI-approved open source license.

New to Python or choosing between

Support the Python

Help the Python c
becoming an associ
making a one-tim

Python 3 Poll

I wish there was Pyth

(enter PyPI pack

Vote Re

AFNIC.fr uses Pytho

Questions

What is python.org?

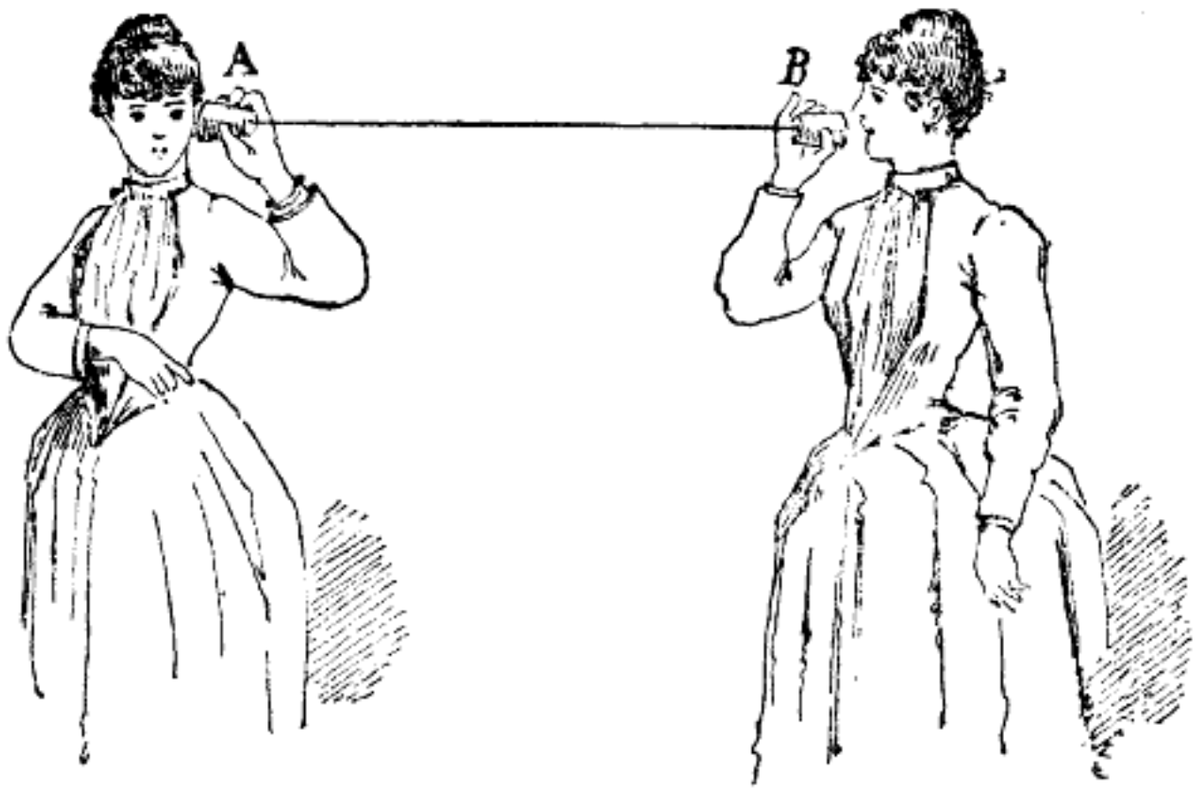
Where is python.org?

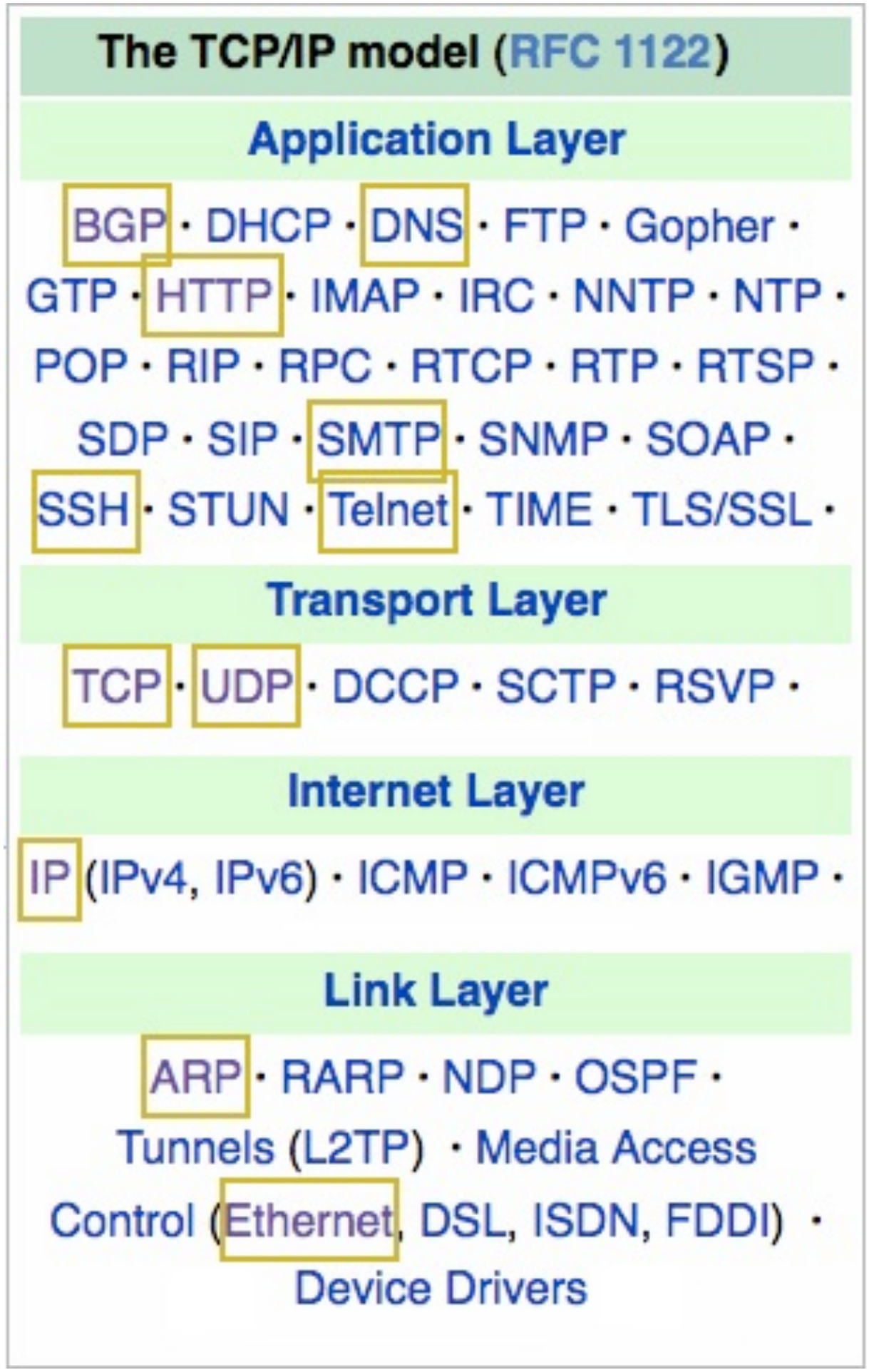
How does my computer talk to python.org?

What does my computer say to python.org?

Protocol

A format and rules for exchanging information



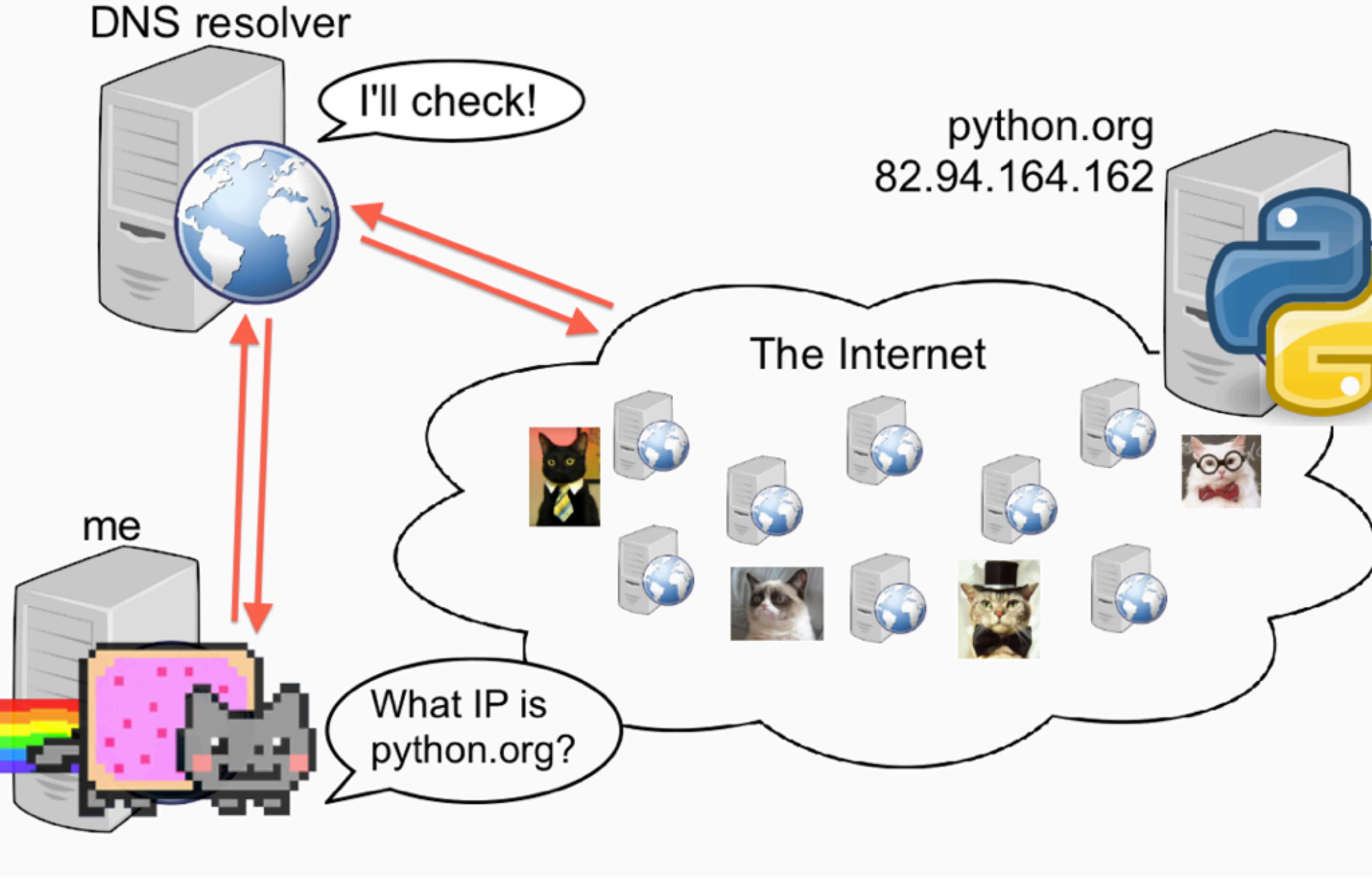


Internet protocols are layered, one protocol per task

What is python.org?

Domain Name Service (DNS) translates hostnames to IP addresses

python.org → 82.94.164.162

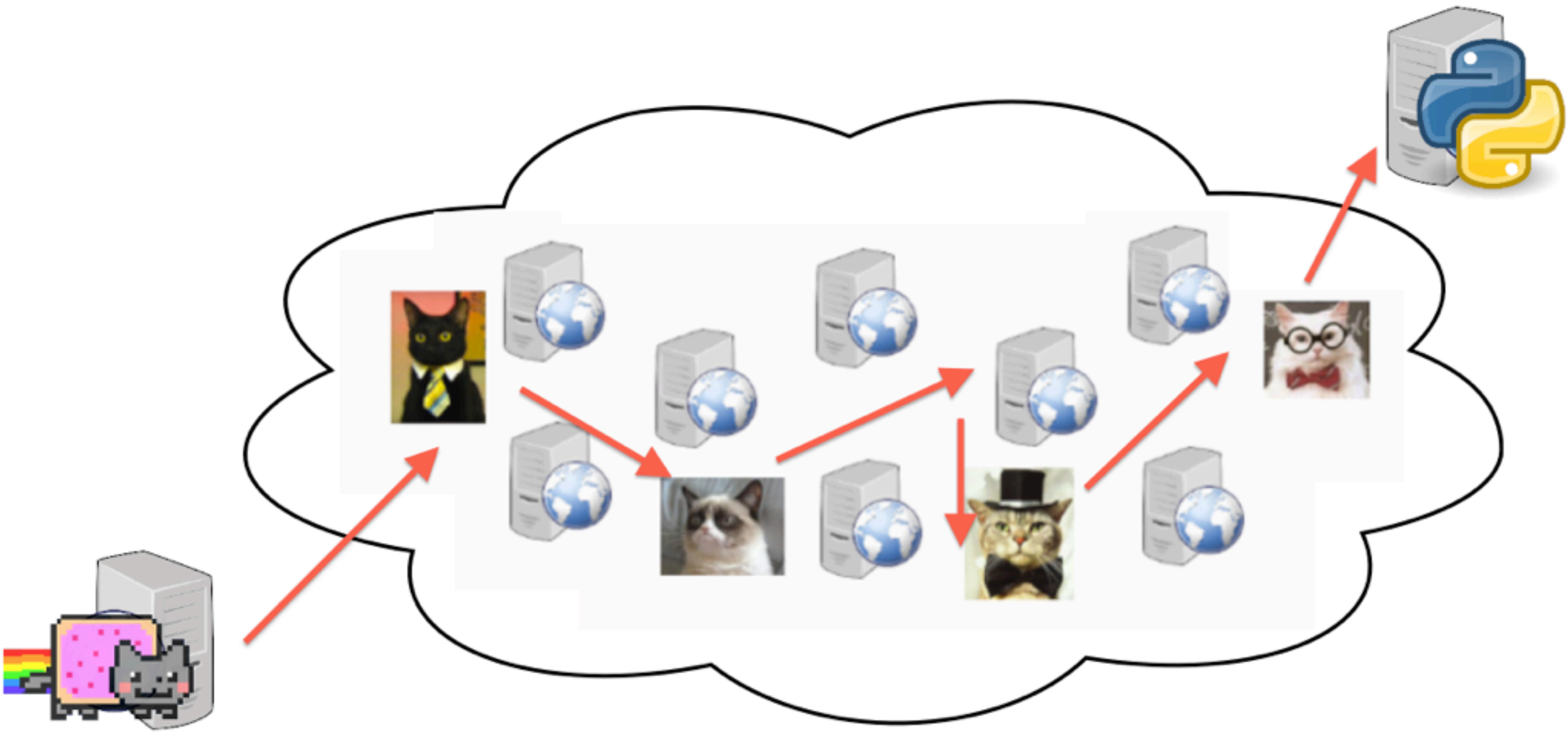

DNS resolver
I'll check!
python.org
82.94.164.162
The Internet
me
What IP is
python.org?

Where is python.org?

Internet Protocol (IP)

handles addressing and routing

HTTP
TCP
IP



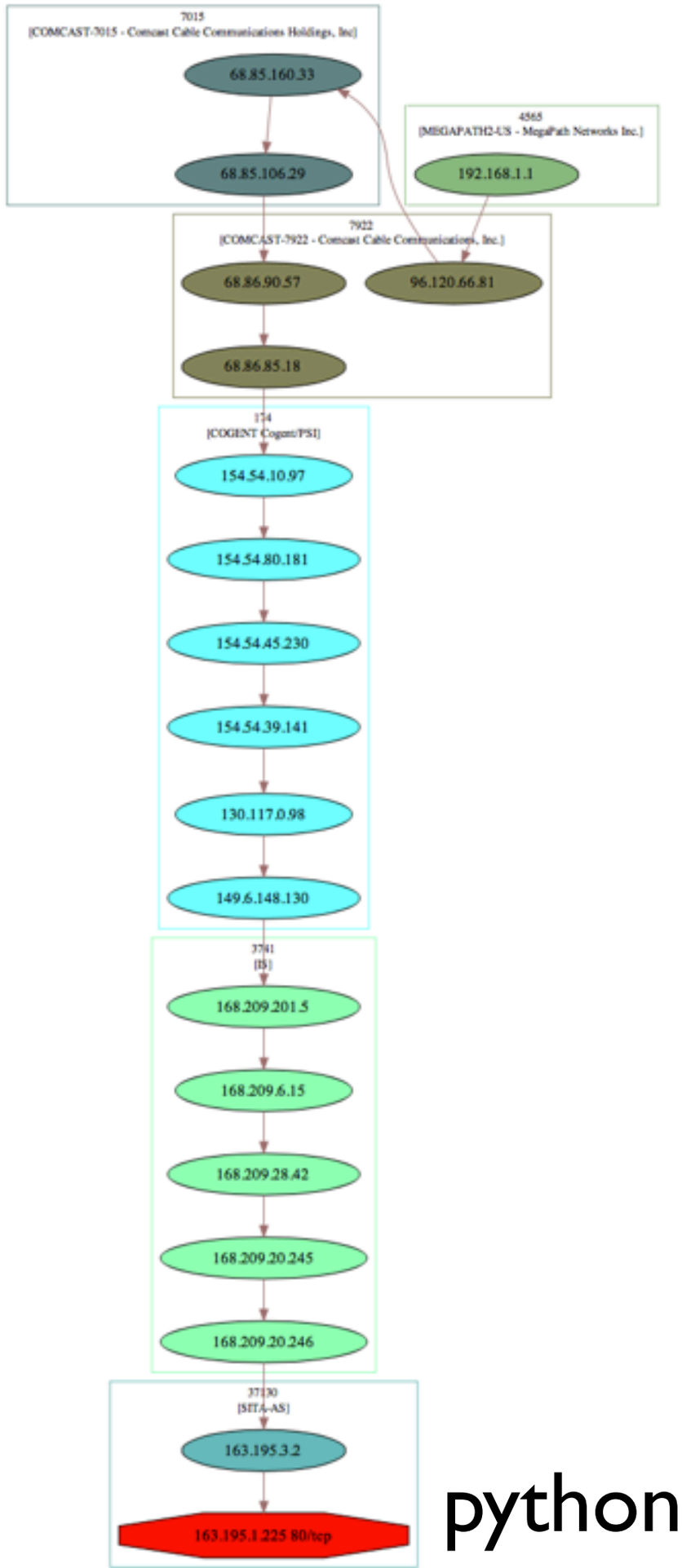
Your packets of data hop from router to router through the Internet to their destination

scapy demo!

Desktop — jesstess@kid-charlemagne: ~ — ssh — 70×19

$

```
>>> res, _ = traceroute("python.org")
Begin emission:*.*.*.*.*.*.*.*.*.*.*.*.*.
   82.94.164.162:tcp80
1  192.168.1.1      openwrt.lan
2  96.120.66.81     96.120.168.1.1
3  68.85.160.37     cambridge.ma.boston.comcast.net
4  68.85.106.29     needham.ma.boston.comcast.net
5  68.86.90.57      newyork.ny.ibone.comcast.net
6  68.86.85.22      111eighthave.ny.ibone.comcast.net
7  75.149.228.126   nyk-s2-rou-1001.us.eurorings.net
8  134.222.226.154  ldn-s2-rou-1101.uk.eurorings.net
9  134.222.231.148  ldn-s1-rou-1021.uk.eurorings.net
10 134.222.231.156  asd2-rou-1022.nl.eurorings.net
11 134.222.229.113  asd2-rou-1044.nl.eurorings.net
12 134.222.97.18    134.222.97.18
13 194.109.5.82     xr4.1d12.xs4all.net
14 194.109.12.34    swcolo2.3d12.xs4all.net
15 82.94.164.162    dinsdale.python.org

>>> res.graph(target="> traceroute.svg")
```



Me

python.org

7015
[COMCAST-7015 - Comcast Cable Communications Holdings, Inc]

68.85.160.33

68.85.106.29

Me

4565
[MEGAPATH2-US - MegaPath Networks Inc.]

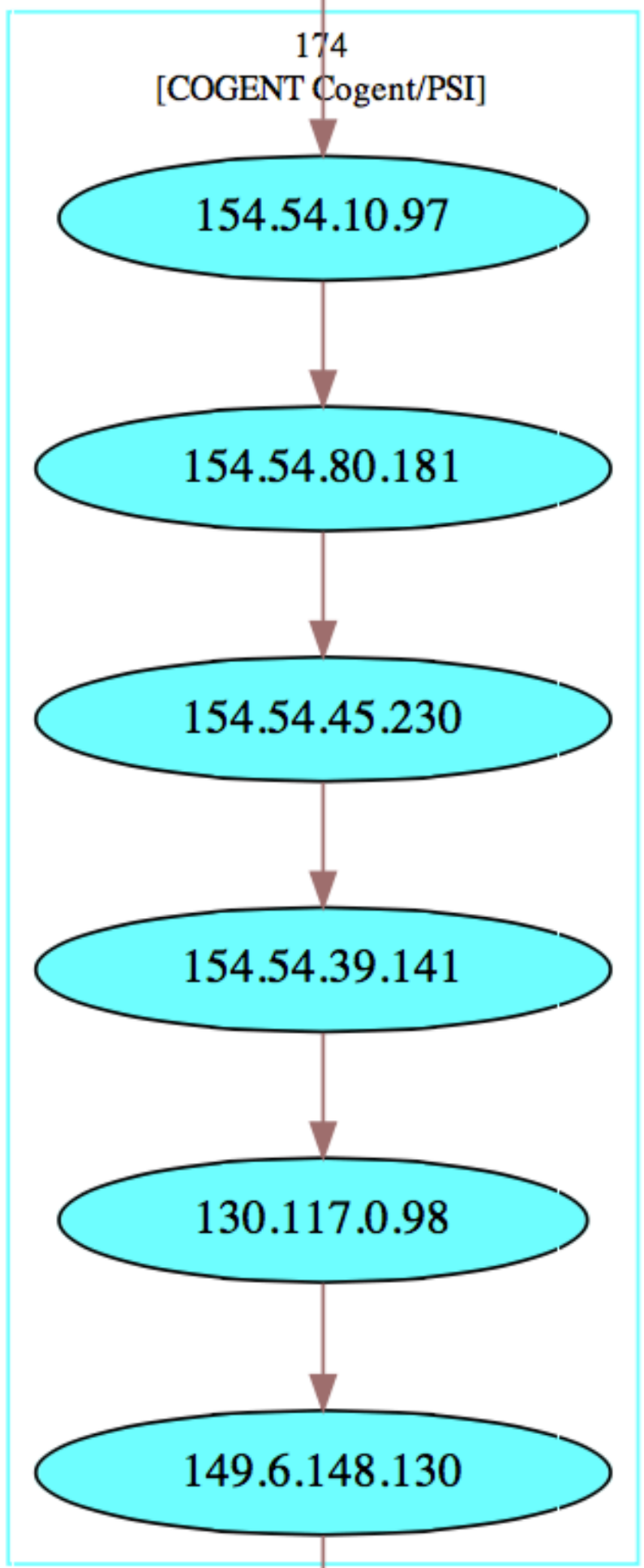
192.168.1.1

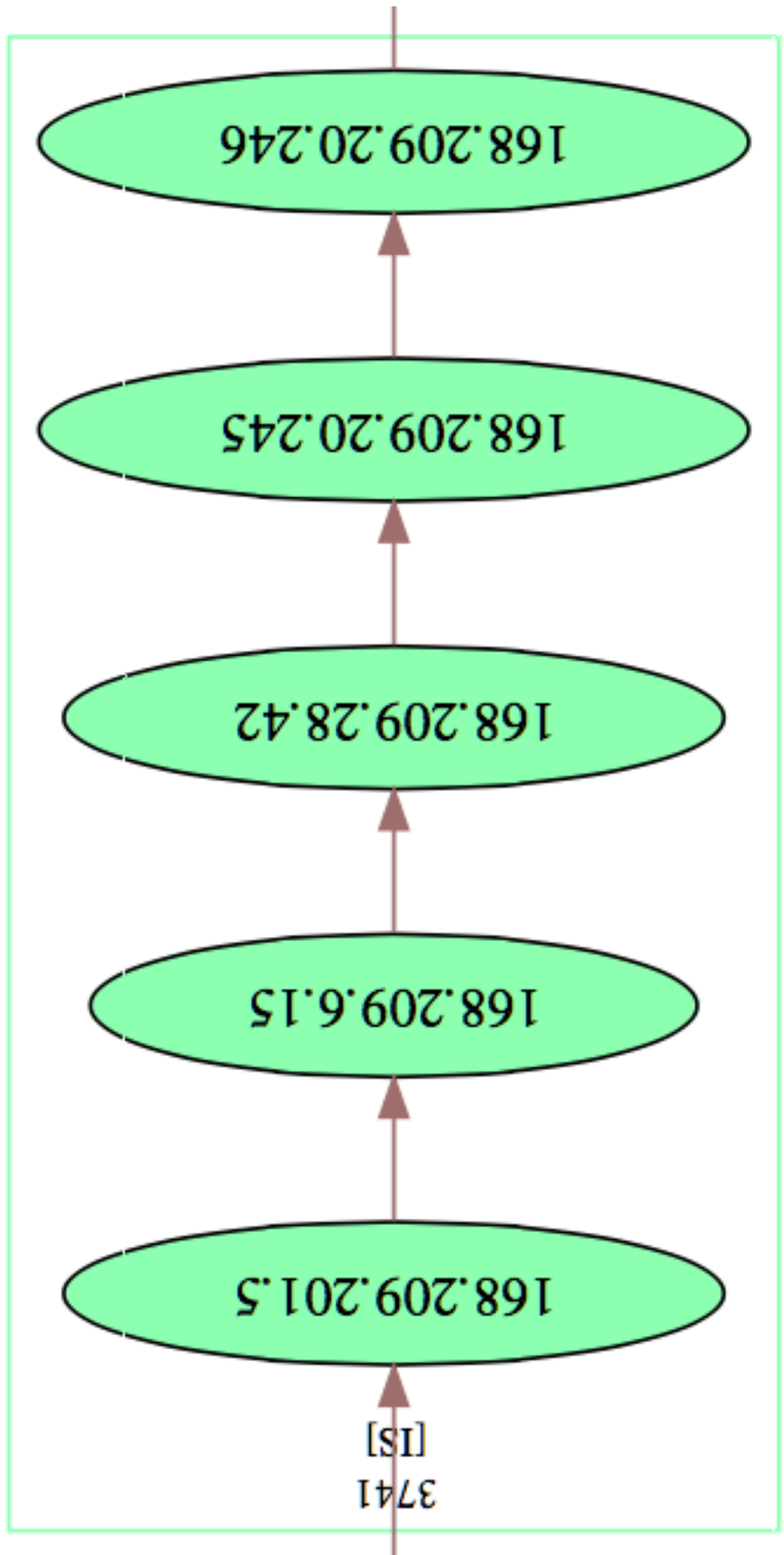
7922
[COMCAST-7922 - Comcast Cable Communications, Inc.]

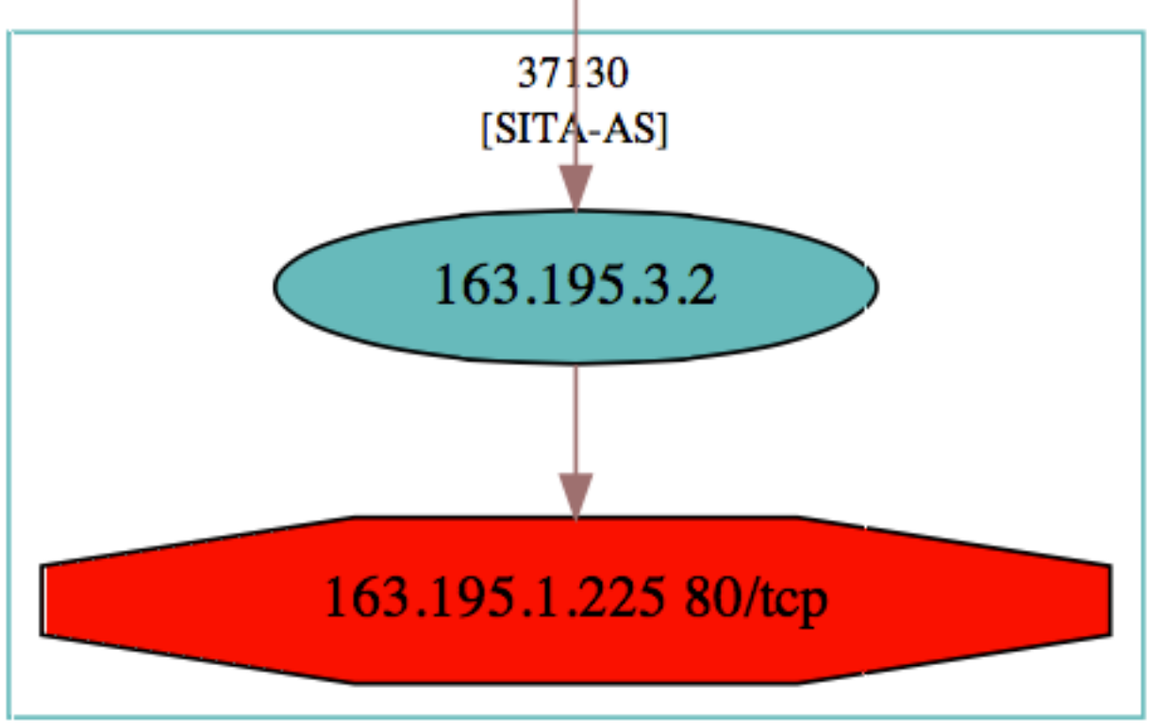
68.86.90.57

96.120.66.81

68.86.85.18


174
[COGENT Cogent/PSI]
154.54.10.97
154.54.80.181
154.54.45.230
154.54.39.141
130.117.0.98
149.6.148.130


168.209.20.246
168.209.20.245
168.209.28.42
168.209.6.15
168.209.201.5
3741
[IS]

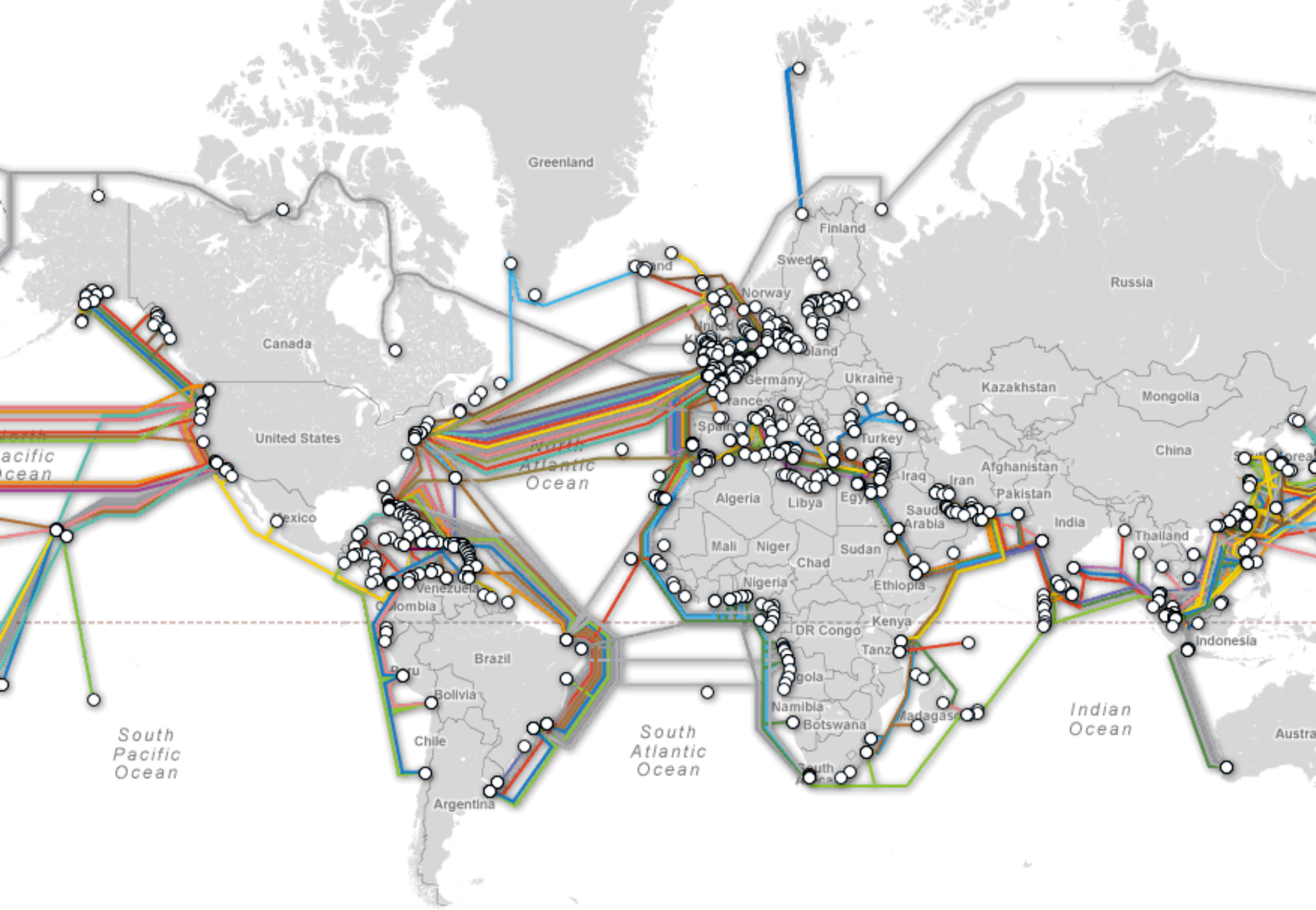


python.org

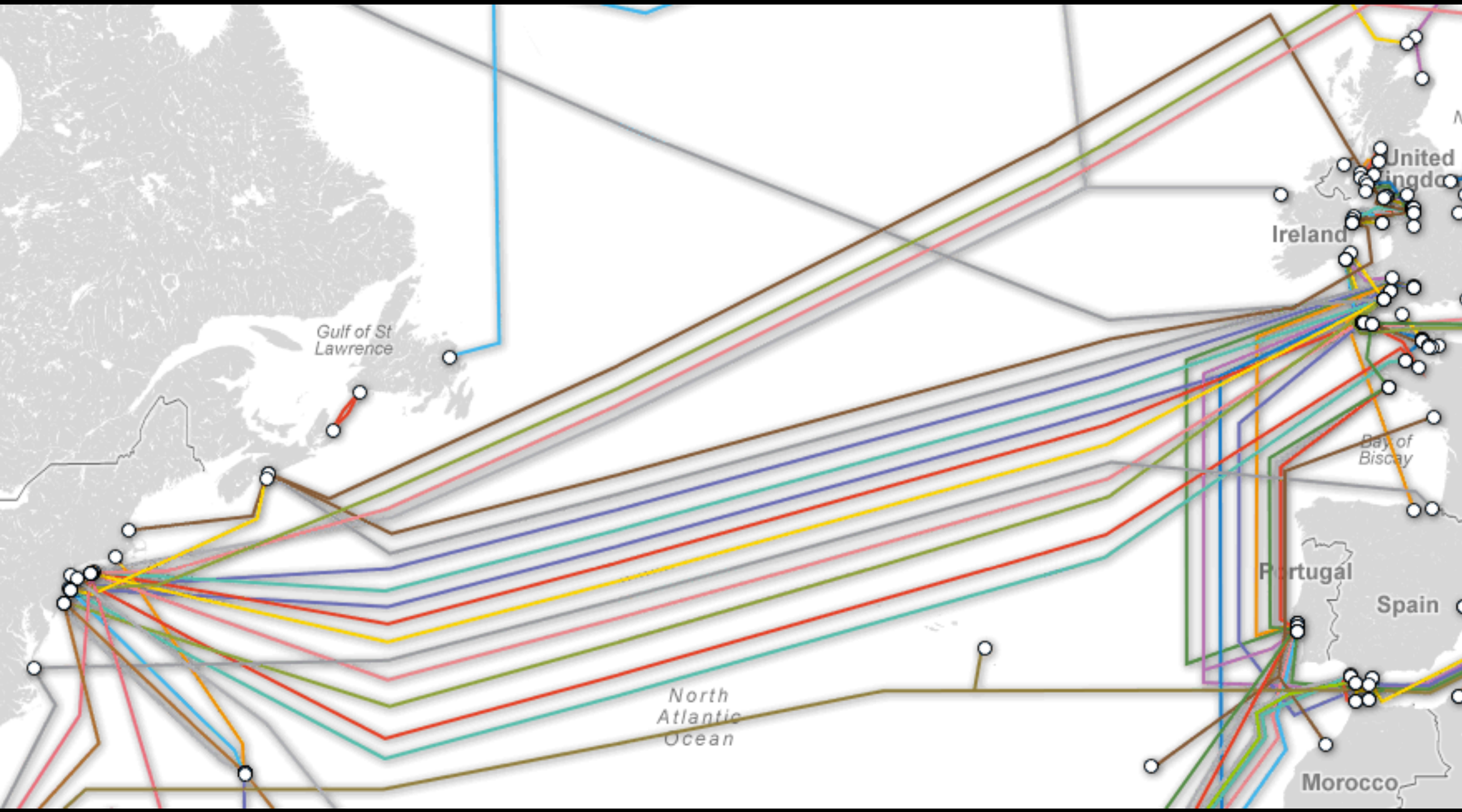
```
traceroute_ips = """
96.120.66.81
68.85.160.37
68.85.106.29
68.86.90.57
68.86.85.22
75.149.228.126
134.222.226.154
134.222.231.148
134.222.231.156
134.222.229.113
134.222.97.18
194.109.5.82
194.109.12.34
82.94.164.162
""".strip().split("\n")
```

```
import pygeoip

gi = pygeoip.GeoIP("GeoLiteCity.dat")

for ip in traceroute_ips:
    record = gi.record_by_addr(ip)
    print ip.ljust(15),
    print record["country_name"],
    if record["time_zone"]:
        print record["time_zone"],
    print ""
```

```
$ python geolocate.py
96.120.66.81     United States  Boston
68.85.160.37     United States  Boston
68.85.106.29     United States  Boston
68.86.90.57      United States  New_York
68.86.85.22      United States  New_York
75.149.228.126   United States  New_York
134.222.226.154 United Kingdom London
134.222.231.148 United Kingdom London
134.222.231.156 Netherlands    Amsterdam
134.222.229.113 Netherlands    Amsterdam
134.222.97.18   Netherlands    Amsterdam
194.109.5.82    Netherlands    Amsterdam
194.109.12.34   Netherlands    Amsterdam
82.94.164.162   Netherlands    Amsterdam
```



http://submarinecablemap.com


Gulf of St Lawrence
United Kingdom
Ireland
Bay of Biscay
Portugal
Spain
North Atlantic Ocean
Morocco

Where is python.org?

Internet Protocol (IP)

IP handles addressing and routing

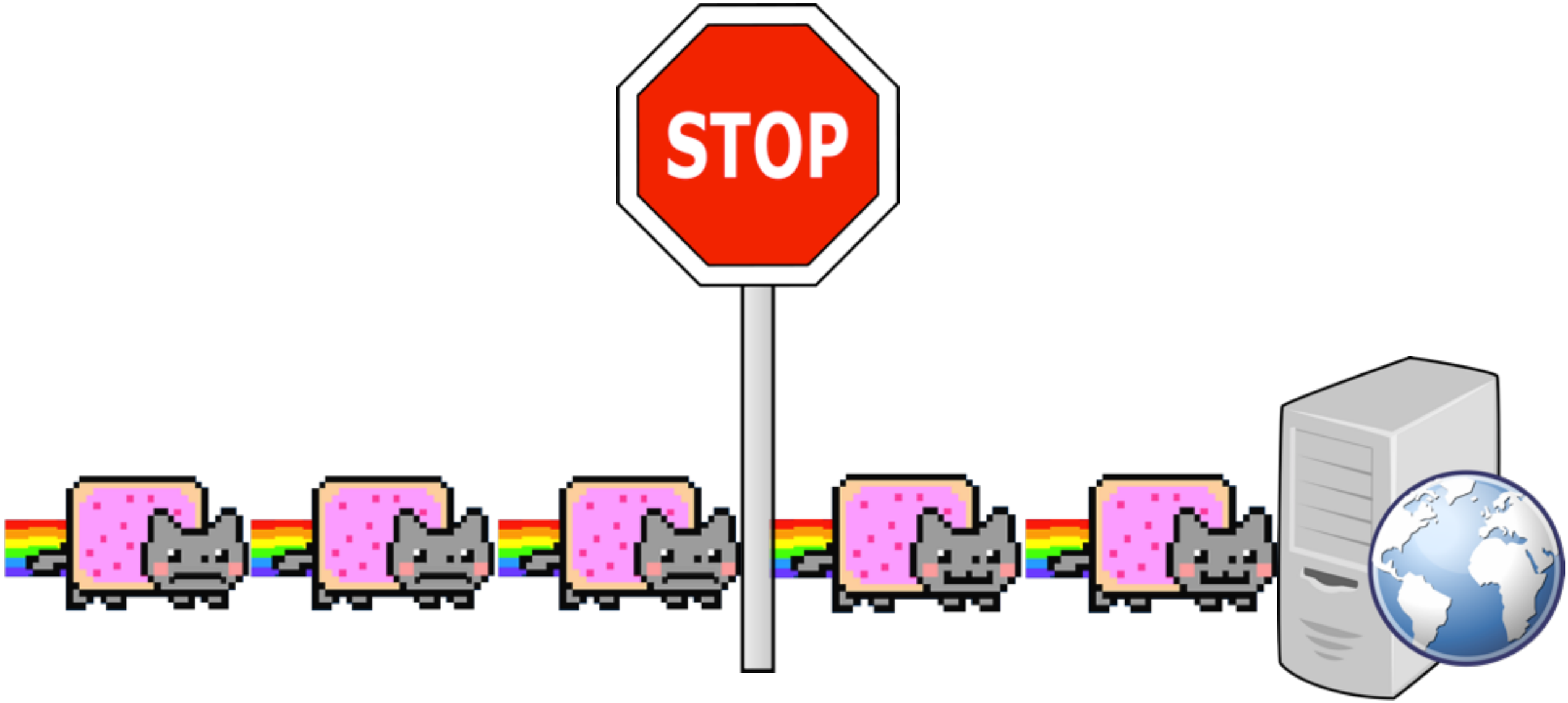
HTTP
TCP
IP

How does my computer talk to python.org?

Transmission Control Protocol (TCP)

TCP reliably delivers data

HTTP
TCP
IP



HTTP
TCP
IP

Application-layer protocols that use TCP

web: HTTP

chat: IRC, XMPP/Jabber

email: SMTP, POP3, IMAP

Each application uses a different "port" number, so many different applications can use TCP to talk to an IP address at the same time

How does my computer talk to python.org?

Transmission Control Protocol (TCP)

TCP reliably delivers data

HTTP
TCP
IP

What does my computer say to python.org?

HyperText Transfer Protocol (HTTP)

Clients use HTTP to request resources from servers

HTTP
TCP
IP

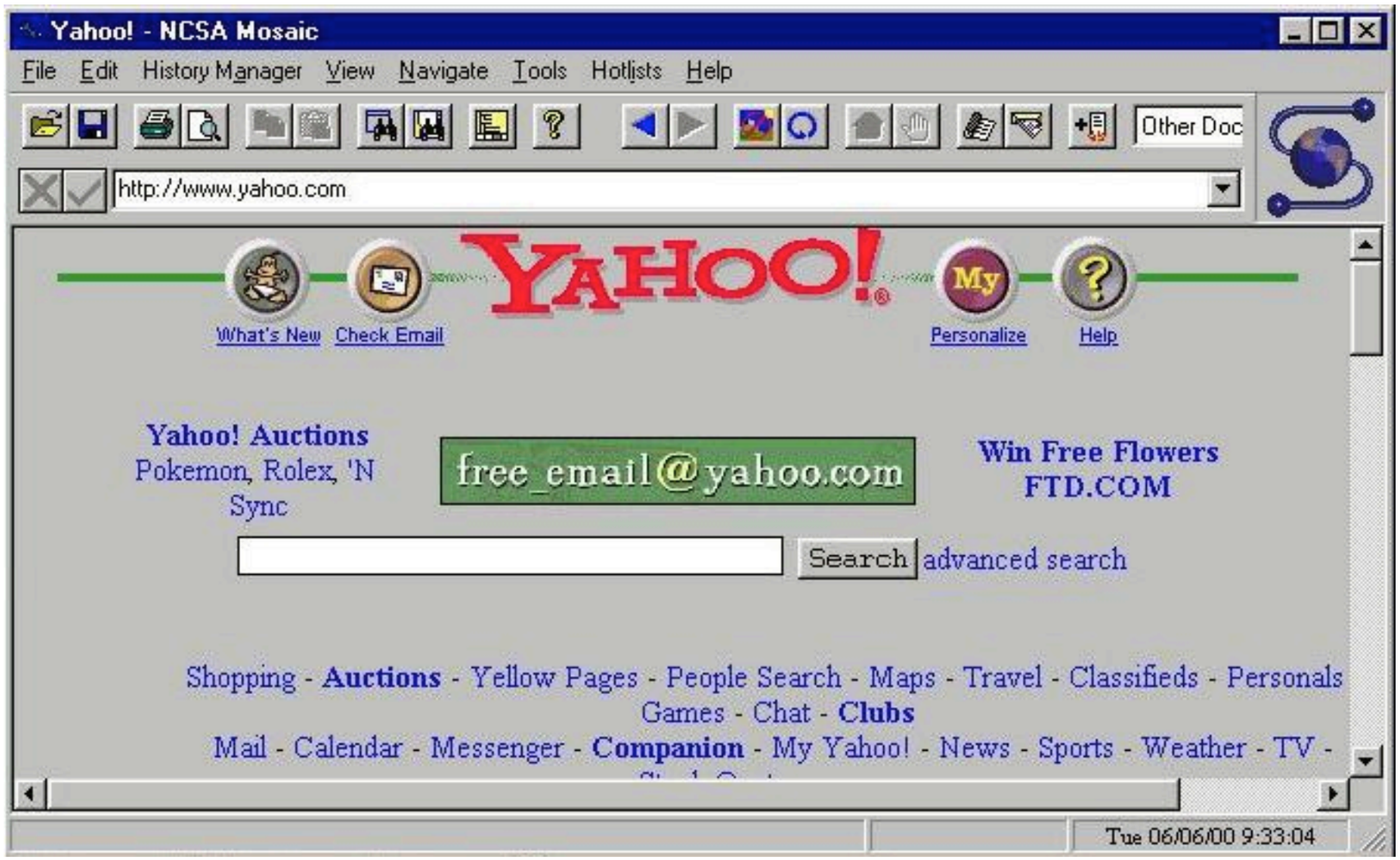
Resources

http://python.org

- HTML
- images
- textual or binary data
- dynamically-generated query results

HTTP
TCP
IP

How to be a web browser



HTTP
TCP
IP

telnet demo!

jesstess — bash — 60×16

$

```
$ telnet python.org 80
Trying 82.94.164.162...
Connected to python.org.
Escape character is '^]'.
GET / HTTP/1.1
Host: python.org
```

```
HTTP/1.1 200 OK
Date: Mon, 04 Mar 2013 00:22:30 GMT
Server: Apache/2.2.16 (Debian)
Last-Modified: Fri, 01 Mar 2013
22:31:40 GMT
Content-Type: text/html
```

```
<head>
  <title>
    Python Programming Language
    &ndash; Official Website
  </title>
</head>
```

```
HTTP/1.1 200 OK
Date: Mon, 04 Mar 2013 00:22:30 GMT
Server: Apache/2.2.16 (Debian)
Last-Modified: Fri, 01 Mar 2013
22:31:40 GMT
Content-Type: text/html
```

```
<head>
  <title>
    Python Programming Language
    &ndash; Official Website
  </title>
</head>
```

```
HTTP/1.1 200 OK
Date: Mon, 04 Mar 2013 00:22:30 GMT
Server: Apache/2.2.16 (Debian)
Last-Modified: Fri, 01 Mar 2013
22:31:40 GMT
Content-Type: text/html

<head>
  <title>
    Python Programming Language
    &ndash; Official Website
  </title>
</head>
```

```
HTTP/1.1 200 OK
Date: Mon, 04 Mar 2013 00:22:30 GMT
Server: Apache/2.2.16 (Debian)
Last-Modified: Fri, 01 Mar 2013
22:31:40 GMT
Content-Type: text/html
```

```
<head>
  <title>
    Python Programming Language
    &ndash; Official Website
  </title>
</head>
```

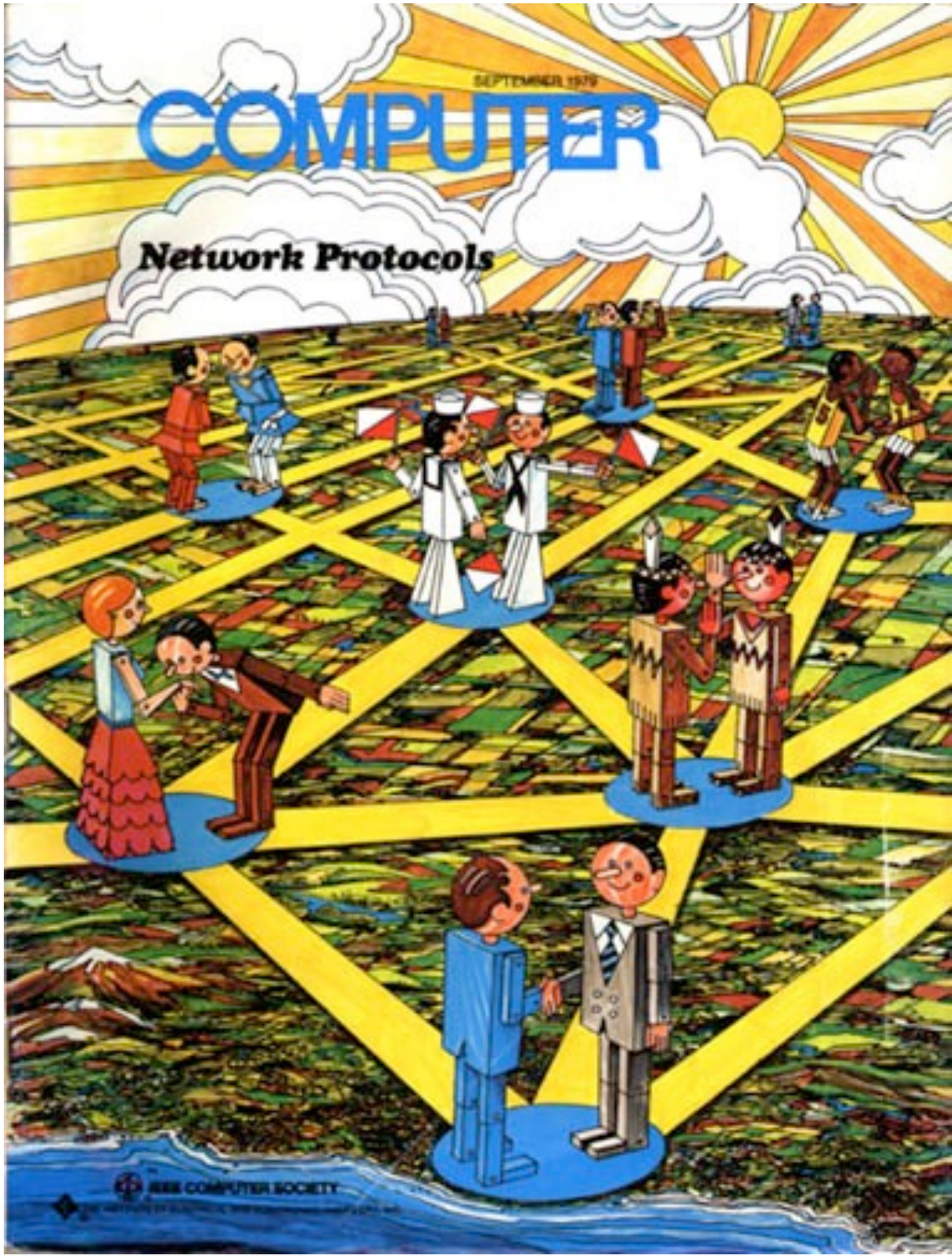
```
HTTP/1.1 200 OK
Date: Mon, 04 Mar 2013 00:22:30 GMT
Server: Apache/2.2.16 (Debian)
Last-Modified: Fri, 01 Mar 2013
22:31:40 GMT
Content-Type: text/html
```

```
<head>
  <title>
    Python Programming Language
    &ndash; Official Website
  </title>
</head>
```

```
import urllib2
urllib2.urlopen(
   "http://python.org").read()
```

HyperText Transfer Protocol
HyperText Markup Language

How to be a web server



```
class HTTPProtocol(basic.LineReceiver):
    def __init__(self):
        self.lines = []

    def lineReceived(self, line):
        self.lines.append(line)
        if not line:
            self.sendResponse()

    def sendResponse(self):
        self.sendLine("HTTP/1.1 200 OK")
        self.sendLine("")
        for line in self.lines:
            self.sendLine(line)
        self.transport.loseConnection()

reactor.listenTCP(80, HTTPFactory())
reactor.run()
```

Twisted HTTP echo server

```
class HTTPProtocol(basic.LineReceiver):
    def __init__(self):
        self.lines = []

    def lineReceived(self, line):
        self.lines.append(line)
        if not line:
            self.sendResponse()

    def sendResponse(self):
        self.sendLine("HTTP/1.1 200 OK")
        self.sendLine("")
        for line in self.lines:
            self.sendLine(line)
        self.transport.loseConnection()

reactor.listenTCP(80, HTTPFactory())
reactor.run()
```

What does my computer say to python.org?

HyperText Transfer Protocol (HTTP)

Clients use HTTP to request resources from servers

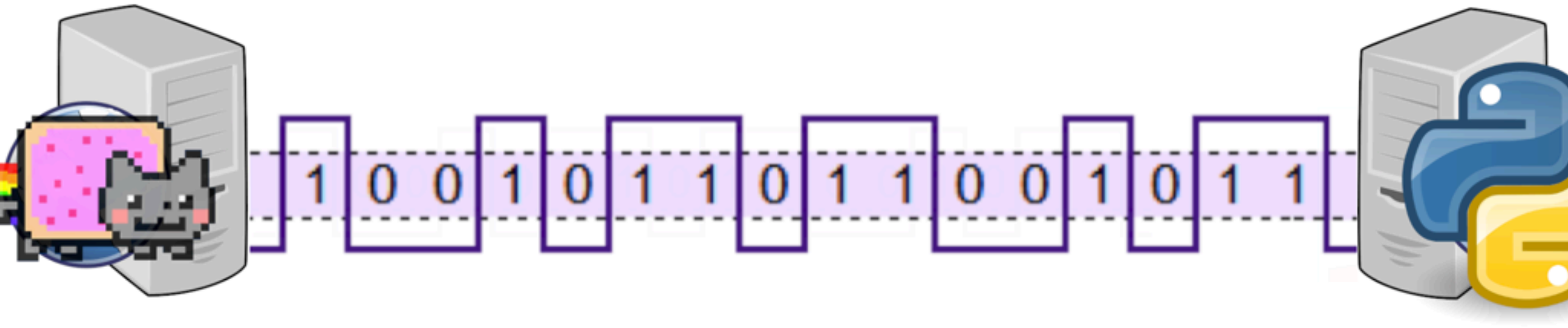
HTTP
TCP
IP

You type http://python.org into your browser bar and press Enter. What happens?

HTTP requesting resources **HTTP**

TCP reliable delivery **TCP**

IP addressing and routing **IP**



DNS hostnames → IP addresses **DNS**

<wireshark demo>

The Wireshark Network Analyzer

File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help

Filter: Expression... Clear Apply Save

WIRESHARK

The World's Most Popular Network Protocol Analyzer

Capture

Interface List
Live list of the capture interfaces
(counts incoming packets)

Start
Choose one or more interfaces to capture from, then **Start**

- fw0
- en1
- p2p0
- lo0 (loopback)

Capture Options
Start a capture with detailed options

Capture Help

How to Capture
Step by step to a successful capture setup

Network Media

Files

Open
Open a previously captured file

Open Recent:

/Users/jesstess/Desktop/pycon_capture (6106 KB)
/Users/jesstess/Desk ... op/getting_wireless_at_philly-2010-12-24 [no
/Users/jesstess/Desktop/getting_wireless_at_logan-2010-12-24 [not f

Sample Captures
A rich assortment of example capture files on the wiki

Ready to load or capture No Packets

Surfing the web

source IP

protocol

192.168.1.1	192.168.1.127	DNS	Standard query response 0x119e CNAME www.meetup.com A
192.168.1.1	192.168.1.127	DNS	Standard query response 0x16f8 A 38.123.132.30
192.168.1.127	38.123.132.30	TCP	60546 > http [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=16
192.168.1.127	38.123.132.30	TCP	60547 > http [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=16
38.123.132.30	192.168.1.127	TCP	http > 60547 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=
192.168.1.127	38.123.132.30	TCP	60547 > http [ACK] Seq=1 Ack=1 Win=65535 Len=0
192.168.1.127	38.123.132.30	HTTP	GET /bostonpython/ HTTP/1.1
38.123.132.30	192.168.1.127	TCP	http > 60546 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=
192.168.1.127	38.123.132.30	TCP	60546 > http [ACK] Seq=1 Ack=1 Win=65535 Len=0
38.123.132.30	192.168.1.127	TCP	http > 60547 [ACK] Seq=1 Ack=1127 Win=7882 Len=0
192.168.1.127	74.125.226.214	TCP	[TCP segment of a reassembled PDU]
192.168.1.127	74.125.226.214	TCP	[TCP segment of a reassembled PDU]

destination IP

protocol data

Lonely printers

source IP

protocol

192.168.1.127	192.168.1.255	BJNP	Printer Command: Discover
192.168.1.127	224.0.0.1	BJNP	Printer Command: Discover
22:e5:2a:4d:88:b3	Apple_c4:e6:47	ARP	Who has 192.168.1.127? Tell 192.168.1.1
Apple_c4:e6:47	22:e5:2a:4d:88:b3	ARP	192.168.1.127 is at 20:c9:d0:c4:e6:47

destination IP

protocol data

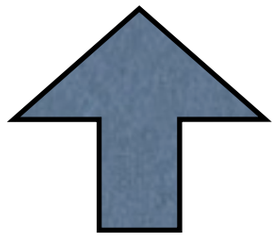
Lonely TiVos

```
9753 260.151096 192.168.1.127   72.21.91.111     TCP           60642 > https [ACK] Seq=
9754 260.759240 192.168.1.143   192.168.1.255    TiVoConnect   Discovery Beacon fizgig
9755 261.377265 54.243.101.69   192.168.1.127    TCP           http > 60646 [FIN, ACK]
```

```
▷ Frame 9754: 195 bytes on wire (1560 bits), 195 bytes captured (1560 bits)
▷ Ethernet II, Src: Tivo_38:b9:db (00:11:d9:38:b9:db), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
▷ Internet Protocol Version 4, Src: 192.168.1.143 (192.168.1.143), Dst: 192.168.1.255 (192
▷ User Datagram Protocol, Src Port: tivoconnect (2190), Dst Port: tivoconnect (2190)
▽ TiVoConnect Discovery Protocol, fizgig (7460001908A9F7D)
    Flavor: 1
    Version: 20.2.2.1-01-2-746
    Method: broadcast
    Identity: 7460001908A9F7D
    Machine: fizgig
    Platform: tcd/Series4
    Services: TiVoMediaServer:80/http
```

Errors

64.94.107.62	192.168.1.127	TLSv1	Application Data, Application Data
192.168.1.127	64.94.107.62	TCP	60706 > https [ACK] Seq=1722 Ack=460 Wir
192.168.1.127	64.94.107.62	TCP	60706 > https [FIN, ACK] Seq=1722 Ack=46
184.173.189.190	192.168.1.127	TCP	http > 60705 [ACK] Seq=1 Ack=5841 Win=21
184.173.189.190	192.168.1.127	TCP	[TCP Window Update] http > 60705 [ACK] S
184.173.189.190	192.168.1.127	TCP	[TCP Dup ACK 8064#1] http > 60705 [ACK]
184.173.189.190	192.168.1.127	TCP	[TCP Dup ACK 8064#2] http > 60705 [ACK]
192.168.1.127	184.173.189.190	TCP	[TCP Fast Retransmission] [TCP segment c
184.173.189.190	192.168.1.127	TCP	http > 60705 [ACK] Seq=1 Ack=6946 Win=26
23.22.96.69	192.168.1.127	TCP	http > 60709 [ACK] Seq=2 Ack=346 Win=156
204.13.194.186	192.168.1.127	TCP	https > 60711 [SYN, ACK] Seq=0 Ack=1 Wir



TCP protocol,
retransmitting lost data

Encrypted communication

source IP

protocol

192.168.1.127	199.59.148.10	TLSv1	Client Hello
199.59.148.10	192.168.1.127	TLSv1	Server Hello
199.59.148.10	192.168.1.127	TLSv1	Certificate, Server Hello Done
192.168.1.127	199.59.148.10	TLSv1	Client Key Exchange, Change Cipher Spec,
199.59.148.10	192.168.1.127	TLSv1	New Session Ticket, Change Cipher Spec,
192.168.1.127	199.59.148.10	TLSv1	Application Data
192.168.1.127	199.59.148.10	TLSv1	Application Data
199.59.148.10	192.168.1.127	TLSv1	Application Data
199.59.148.10	192.168.1.127	TLSv1	Application Data
199.59.148.10	192.168.1.127	TLSv1	Application Data
192.168.1.127	74.125.226.214	TLSv1.1	Application Data

destination IP

protocol data

IRC

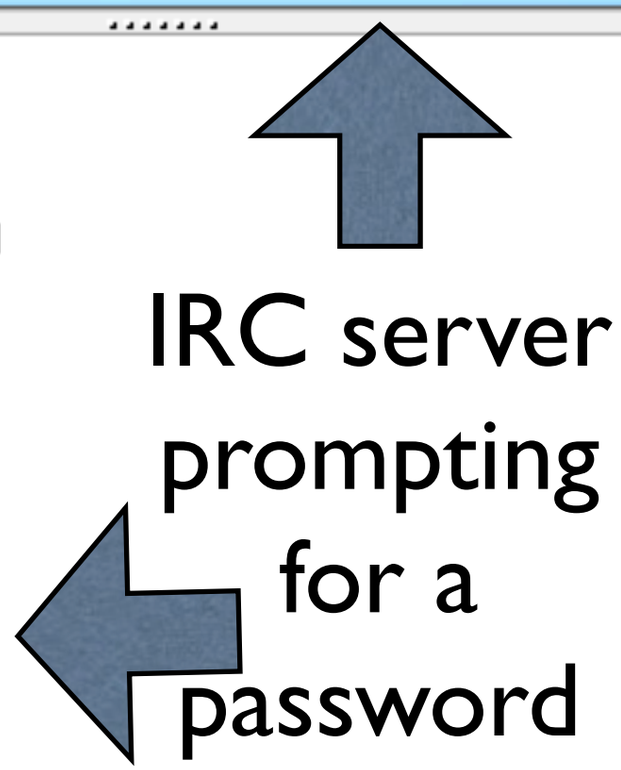
192.168.1.127	82.96.64.4	IRC	Request (CAP)
82.96.64.4	192.168.1.127	IRC	Response (NOTICE)
192.168.1.127	82.96.64.4	IRC	Request (NICK) (USER)
82.96.64.4	192.168.1.127	IRC	Response (NOTICE)
82.96.64.4	192.168.1.127	IRC	Response (NOTICE) (NOTICE) (CAP)
192.168.1.127	82.96.64.4	IRC	Request (CAP)
82.96.64.4	192.168.1.127	IRC	Response (001)
82.96.64.4	192.168.1.127	IRC	Response (002) (003) (004) (005)
82.96.64.4	192.168.1.127	IRC	Response (2) (250) (375) (372) (3
82.96.64.4	192.168.1.127	IRC	Response (372) (372) (372) (372)
82.96.64.4	192.168.1.127	IRC	Response (e) (372) (372) (372) (3
82.96.64.4	192.168.1.127	IRC	Response (372) (372) (376) (MODE)
82.96.64.4	192.168.1.127	IRC	Response (NOTICE)
192.168.1.127	82.96.64.4	IRC	Request (NICKSERV)
82.96.64.4	192.168.1.127	IRC	Response (NOTICE)
192.168.1.127	82.96.64.4	IRC	Request (JOIN)
82.96.64.4	192.168.1.127	IRC	Response (JOIN)
82.96.64.4	192.168.1.127	IRC	Response (MODE) (353) (366)
192.168.1.127	82.96.64.4	IRC	Request (WHO)
82.96.64.4	192.168.1.127	IRC	Response (352)
82.96.64.4	192.168.1.127	IRC	Response (315)

IRC

58 5.933268	82.96.64.4	192.168.1.127	IRC	Response (NOTICE)
70 12.196841	192.168.1.127	82.96.64.4	IRC	Request (NICKSERV)

```
▷ Frame 58: 231 bytes on wire (1848 bits), 231 bytes captured (1848 bits)
▷ Ethernet II, Src: 22:e5:2a:4d:88:b3 (22:e5:2a:4d:88:b3), Dst: Apple_c4:e6:47 (20:c9:d0:c4:e6:47
▷ Internet Protocol Version 4, Src: 82.96.64.4 (82.96.64.4), Dst: 192.168.1.127 (192.168.1.127)
▷ Transmission Control Protocol, Src Port: 6667 (6667), Dst Port: 61339 (61339), Seq: 6456, Ack:
▽ Internet Relay Chat
  ▷ Response: :NickServ!NickServ@services. NOTICE jesstess :This nickname is registered. Please c
```

```
010  00 d9 fb dd 40 00 33 06  f4 93 52 80 40 04 c0 a8   ....@.3. ..R.@...
020  01 7f 1a 0b ef 9b 60 6e  cf b9 db f8 e1 b1 80 18   ......`n ........
030  00 2e 42 c7 00 00 01 01  08 0a 4f d8 4e ee 42 9b   ..B..... ..O.N.B.
040  01 32 3a 4e 69 63 6b 53  65 72 76 21 4e 69 63 6b   .2:NickS erv!Nick
050  53 65 72 76 40 73 65 72  76 69 63 65 73 2e 20 4e   Serv@ser vices. N
060  4f 54 49 43 45 20 6a 65  73 73 74 65 73 73 20 3a   OTICE je sstess :
070  54 68 69 73 20 6e 69 63  6b 6e 61 6d 65 20 69 73   This nic kname is
080  20 72 65 67 69 73 74 65  72 65 64 2e 20 50 6c 65    registe red. Ple
090  61 73 65 20 63 68 6f 6f  73 65 20 61 20 64 69 66   ase choo se a dif
0a0  66 65 72 65 6e 74 20 6e  69 63 6b 6e 61 6d 65 2c   ferent n ickname,
0b0  20 6f 72 20 69 64 65 6e  74 69 66 79 20 76 69 61    or iden tify via
0c0  20 02 2f 6d 73 67 20 4e  69 63 6b 53 65 72 76 20    ./msg N ickServ
0d0  69 64 65 6e 74 69 66 79  20 3c 70 61 73 73 77 6f   identify  <passwo
0e0  72 64 3e 02 2e 0d 0a                              rd>....
```



IRC

```
70 12.196841    192.168.1.127          82.96.64.4              IRC
▷ Frame 70: 97 bytes on wire (776 bits), 97 bytes captured (776 bits)
▷ Ethernet II, Src: Apple_c4:e6:47 (20:c9:d0:c4:e6:47), Dst: 22:e5:2a:4d:
▷ Internet Protocol Version 4, Src: 192.168.1.127 (192.168.1.127), Dst: 8
▷ Transmission Control Protocol, Src Port: 61339 (61339), Dst Port: 6667
▽ Internet Relay Chat
  ▷ Request: NICKSERV IDENTIFY django4life

0000  22 e5 2a 4d 88 b3 20 c9  d0 c4 e6 47 08 00 45 00   ".*M.. . ...G..E.
0010  00 53 74 10 40 00 40 06  72 09 c0 a8 01 7f 52 60   .St.@.@. r.....R`
0020  40 04 ef 9b 1a 0b db f8  e1 b1 60 6e d0 5e 80 18   @....... ..`n.^..
0030  20 00 46 c4 00 00 01 01  08 0a 42 9b 1b 24 4f d8    .F..... ..B..$O.
0040  4e ee 4e 49 43 4b 53 45  52 56 20 49 44 45 4e 54   N.NICKSE RV IDENT
0050  49 46 59 20 64 6a 61 6e  67 6f 34 6c 69 66 65 0d   IFY djan go4life.
0060  0a                                                .
```

My password, `django4life`

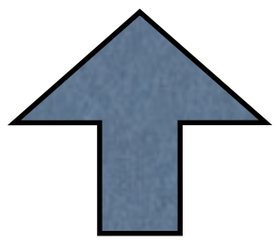
</wireshark demo>

One last demo

How to propose marriage on your local network using Scapy and ARP-cache poisoning

Lonely printers

192.168.1.127	192.168.1.255	BJNP	Printer Command: Discover
192.168.1.127	224.0.0.1	BJNP	Printer Command: Discover
22:e5:2a:4d:88:b3	Apple_c4:e6:47	ARP	Who has 192.168.1.127? Tell 192.168.1.1
Apple_c4:e6:47	22:e5:2a:4d:88:b3	ARP	192.168.1.127 is at 20:c9:d0:c4:e6:47



Address Resolution Protocol (ARP) maps hardware (MAC) addresses to IP addresses

jesstess — root@kid-charlemagne: /home/jesstess — ssh — 63×17

#

When nerds get married

What next?

- Twisted
- Scapy
- wireshark

Twisted

Glasgow Haskell Compiler

PyPy

ZeroMQ

Git

nginx

LLVM

Bash

Berkley DB

Mercurial

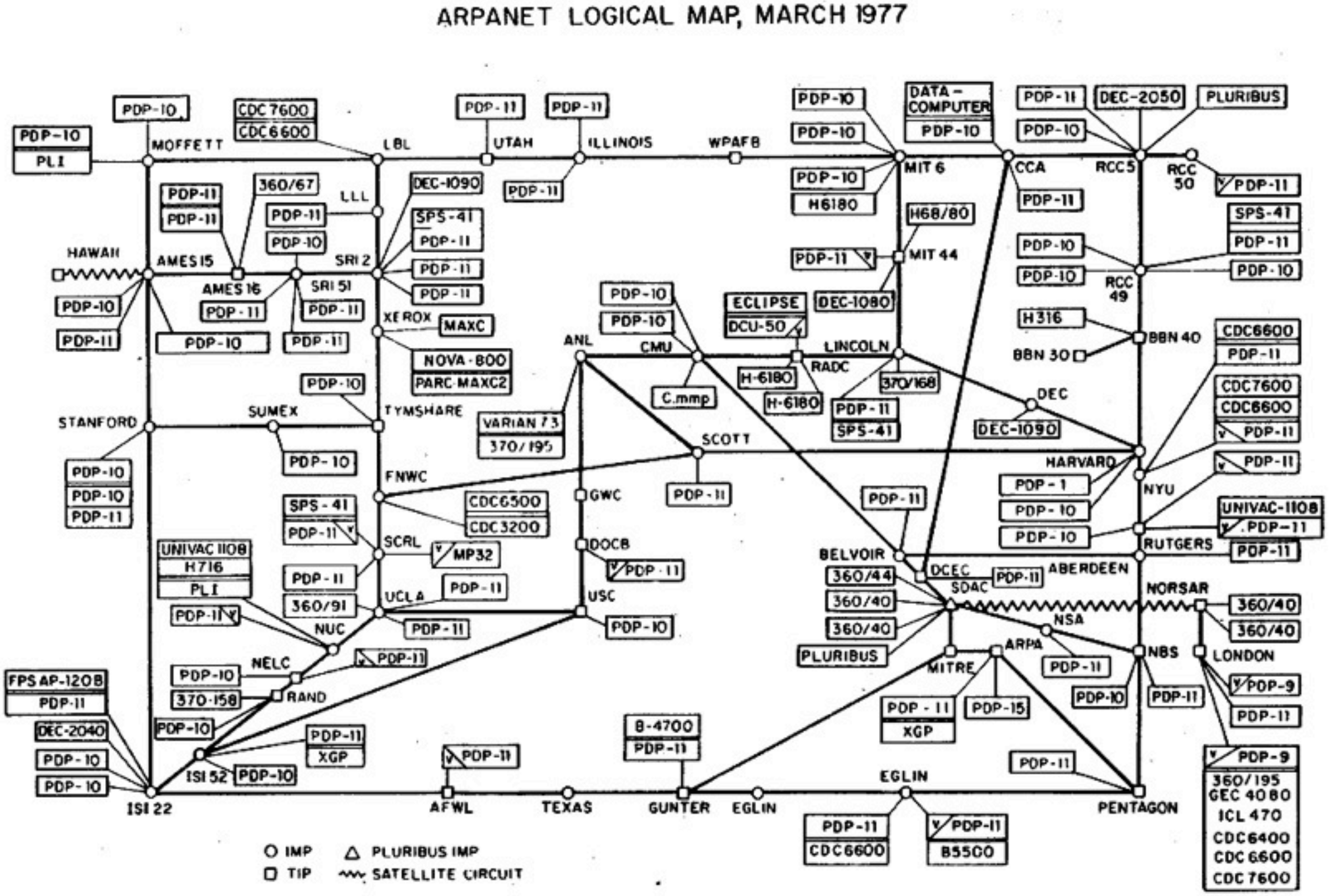
Hot off the press!

Thank you!

@jessicamckellar

http://jesstess.com

A map of the entire Internet in 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