

Get ready to do some Internet

Run the `get_mac.py` script on the course website

It will print a MAC address like **9c:9c:1f:cb:19:24**

Register that MAC address with Tufts here:

<https://device-registration.it.tufts.edu/>

Description can be "ESP32" (or whatever you want)

Device type "Other devices"

EN 1: Engineering in the Kitchen

Steven Bell

2 November 2021



Roadmap to project 4

Objectives

By the end of class today, you should be able to:

Explain what MAC and IP addresses are

Explain what an "API" is

Connect your ESP32 to the Internet

Pull interesting data from public APIs

MAC addresses

Uniquely identify a device

2^{48} possible addresses \approx 256 quadrillion

IP addresses

IP addresses are assigned, not intrinsic

Domain name system (DNS)

Like a giant address book of the Internet

Try <http://130.64.23.7>

TCP / HTTP

How do I guarantee delivery when my delivery agent makes no promises whatsoever?

Grabbing web pages with Python

```
import urequests
```

```
# Connect to the WiFi network as before
```

```
response = urequests.get('http://en1-pi.eecs.tufts.edu')  
print(response.text)
```

You can do this on your computer too using **requests**

Grabbing data from the Internet

```
response = urequests.get('http://en1-pi.eecs.tufts.edu/data')  
  
blob = response.json()
```

You can do this on your computer too using **requests**

Some interesting places with data

NOAA (<https://www.weather.gov/documentation/services-web-api>)

MBTA (<https://api-v3.mbtta.com/>)

Fantasy sports (<https://api.fantasynerd.com/docs/nfl>)