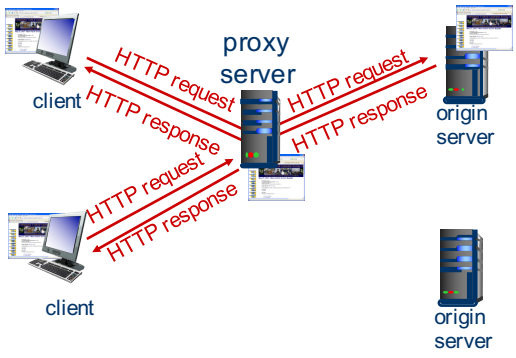
1

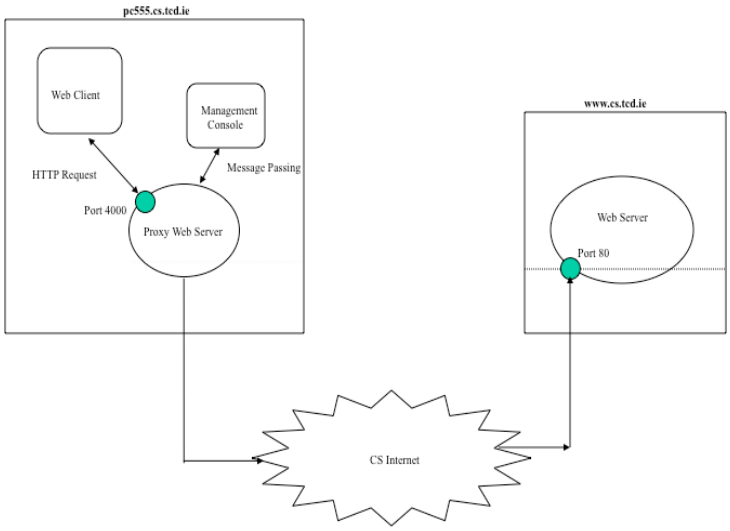
Web Caches (Proxy Server)

- Goal: satisfy client request without involving origin server
- User sets browser to accesses the Web via proxy
- Browser sends all HTTP requests to proxy server
  - Object in cache
    - Proxy returns object
  - Else proxy requests object from origin server
    - Returns object to client



2

Project 2



3

A Minimal Proxy Server

```
import re
import httplib
import string
from socket import *
from urlparse import urlparse

proxyServerPort = 8000

proxyServerSocket =
socket(AF_INET,SOCK_STREAM)

proxyServerSocket.bind(('localhost',
proxyServerPort))

proxyServerSocket.listen(1)

print 'The proxy server is ready to receive'
```

```
while 1:
    connectionSocket, addr =
    proxyServerSocket.accept()
    requestData = connectionSocket.recv(1024)

    # Match Regular Expression
    url = re.search("(?P<url>http?://[^\s]+)",
        requestData).group("url")

    # Parse HTTP Request
    output = urlparse(url)

    # Pass on HTTP Request to real web server
    conn = httplib.HTTPConnection(output.netloc)
    conn.request("GET", output.path)
    r1 = conn.getresponse()

    data1 = r1.read()
    connectionSocket.send(data1)

    connectionSocket.close()
```

4

Next Steps

- Point your browser to “localhost” on port 8000
  - e.g. Firefox Preferences → Advanced → Network → Connection Settings →Manual Proxy Configuration
- Create a more generic parser / RegEx checker to be able to extract
  - Command (GET/POST/PUT …)
  - Host
  - Path etc.
- Handle HTTPS requests
- Block IP addresses
- Cache data on proxy
- For more info see
  - Python Docs, Libraries, Frameworks etc.
    - https://www.python.org/