Network Programming Tools

Richard Ha

Agenda

Administrivia

Proxy Lab Recap

HTTP Overview

Networking Tools

Debugging Tips & Techniques

Administrivia

Proxy lab due date extension!
New due date is Aug. 6, 2015 @ 11:59pm

Will cover changes to handout in a few minutes.

New due date overlaps with final exam dates so be sure to finish the lab early.

Administrivia

Proxy lab due date extension!
New due date is Aug. 6, 2015 @ 11:59pm

Will cover changes to handout in a few minutes.

New due date overlaps with final exam dates so be sure to finish the lab early.

Lab Grading
Shell lab grades are done, please read annotated code.

Malloc grades soon to follow. (Heapchecker, style)

Administrivia

Final Exam
First round final will take place Thursday+Friday.

Recitation for final exam review will be held on Tuesday.

If you want a question answered / topic covered in the final exam review, send an e-mail titled
“FINAL RECITATION QUESTION” to the staff list.

We will pick a list of questions to cover for the recitation, as much as we can fit.

Agenda

Administrivia

Proxy Lab Recap

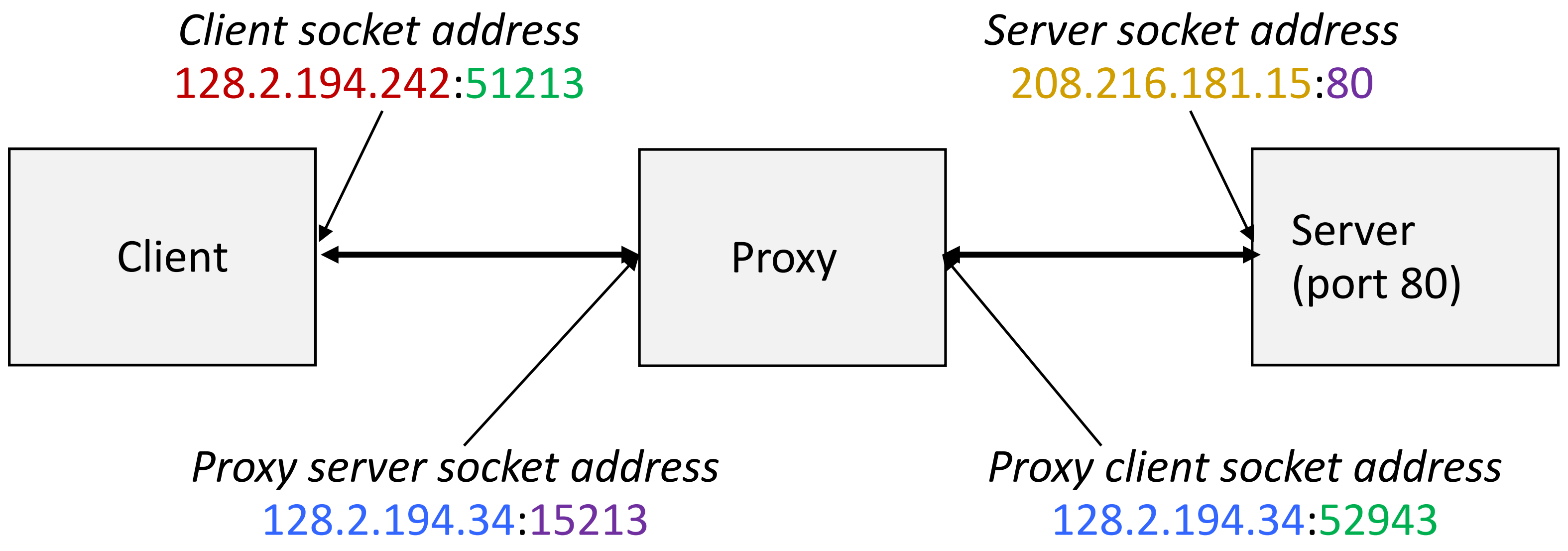
HTTP Overview

Networking Tools

Debugging Tips & Techniques

What is a Proxy?

A proxy is a go-between a client and a server



It is both a server AND a client!

Proxy Requirements

HTTP/1.0
Make and receive HTTP/1.0 requests & responses.

Convert all HTTP/1.1 requests to HTTP/1.0

Proxy Requirements

HTTP/1.0
Make and receive HTTP/1.0 requests & responses.

Convert all HTTP/1.1 requests to HTTP/1.0

Concurrency
Needs to be able to handle multiple clients at once.

Proxy Requirements

HTTP/1.0
Make and receive HTTP/1.0 requests & responses.

Convert all HTTP/1.1 requests to HTTP/1.0

Concurrency
Needs to be able to handle multiple clients at once.

Caching
Needs to temporarily cache web objects in a shared cache amongst threads.

Proxy Requirements

Robustness
Malformed requests and responses are a fact of real life, and not all of them are intentional or malicious.

Your proxy needs to be able to anticipate and deal with malformed requests and responses without crashing, segfaulting, erroring, exiting, or otherwise prematurely terminating.

Servers are expected to run (mostly) forever without human intervention. Your proxy is no exception.

Agenda

Administrivia

Proxy Lab Recap

HTTP Overview

Networking Tools

Debugging Tips & Techniques

Structure of a HTTP Request

HTTP requests are divided into two parts:

The request line
The headers
\r\n

Structure of a HTTP Request

Request Line:
<Request Method> <Request-URI> <HTTP-Version>\r\n

There are nine different HTTP methods. We only care about GET.

Examples:
GET / HTTP/1.0\r\n
GET http://cs.cmu.edu/~213 HTTP/1.0\r\n

Structure of a HTTP Request

Headers:

The headers section consists of a list of header name and value pairs.

<HeaderName>: <HeaderValue>\r\n

Example:

Host: cs.cmu.edu:80\r\n

User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:10.0.3) Gecko/20120305 Firefox/10.0.3\r\n

Structure of a HTTP Request

Headers:

The change in the lab handout was related to the Accept and Accept-Encoding headers.

Handout originally stated you should insert/modify the headers.

Now, just treat those two headers like most other headers and don't modify/change them. Just pass them as-is if they're present.

Structure of a HTTP Response

HTTP Responses are split into three sections:

Status Line
Headers
\r\n
Entity-Body

Structure of a HTTP Response

Status Line
<HTTP-Version> <Status Code> <Reason>\r\n

Example:
HTTP/1.0 200 OK\r\n
HTTP/1.0 302 Moved Permanently\r\n
HTTP/1.0 404 Not Found\r\n

Structure of a HTTP Response

Headers
Same format as request headers.

If there is an entity body that comes with the response (e.g. HTML page, image, etc.), then there will be a Content-Length header.

Structure of a HTTP Response

Content Body
Just a stream of bytes in the designated encoding.

Could be text, could be binary data, could be compressed, could be not compressed, etc.

Agenda

Administrivia

Proxy Lab Recap

HTTP Overview

Networking Tools

Debugging Tips & Techniques

Networking Tools

Just knowing the lab specifications and the HTTP/1.0 spec is not enough to do a lab.

As with any development effort, you need debugging and diagnostic tools to help you catch bugs and errors and identify where the problem is occurring.

Thankfully, Shark machines have plenty of tools to use.

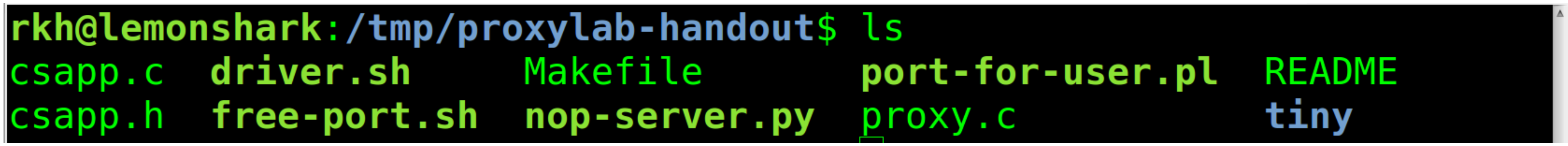
Also included some in the lab handout.

Networking Tools

- Tiny server
- cURL / wget
- Telnet
- Netcat (nc)
- Valgrind
- Your browser(s)
- And many more!

Tiny server

There is a tiny HTTP server included in your proxy lab handout.



```
rkh@lemonshark:/tmp/proxylab-handout$ ls
csapp.c  driver.sh     Makefile       port-for-user.pl  README
csapp.h  free-port.sh  nop-server.py  proxy.c           tiny
```

Tiny server

There is a tiny HTTP server included in your proxy lab handout.

It is a 200-line web server written in C.

Invoked with: ./tiny <port>

Serves everything inside ./

Comes with an example adder CGI script

Consult the README for details.

cURL / wget

Command-line programs that fetch a given web URL.

e.g. $ curl http://lemonshark.ics.cs.cmu.edu:15213/godzilla.jpg

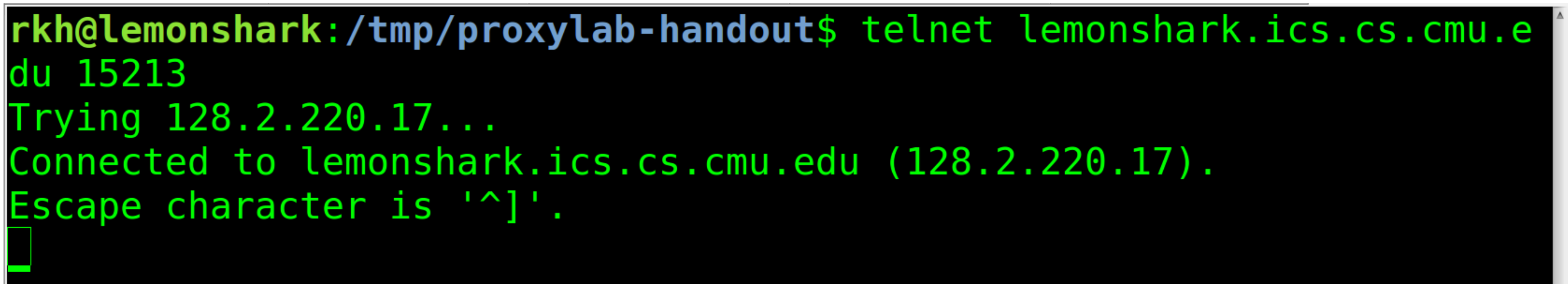
will fetch godzilla.jpg from the tiny server.

curl has a --proxy option that lets you specify a HTTP proxy that it can make requests through

telnet

Client application for the TELNET protocol

Telnet client can conveniently be used to communicate and interact with text-protocol servers like HTTP and IRC servers.

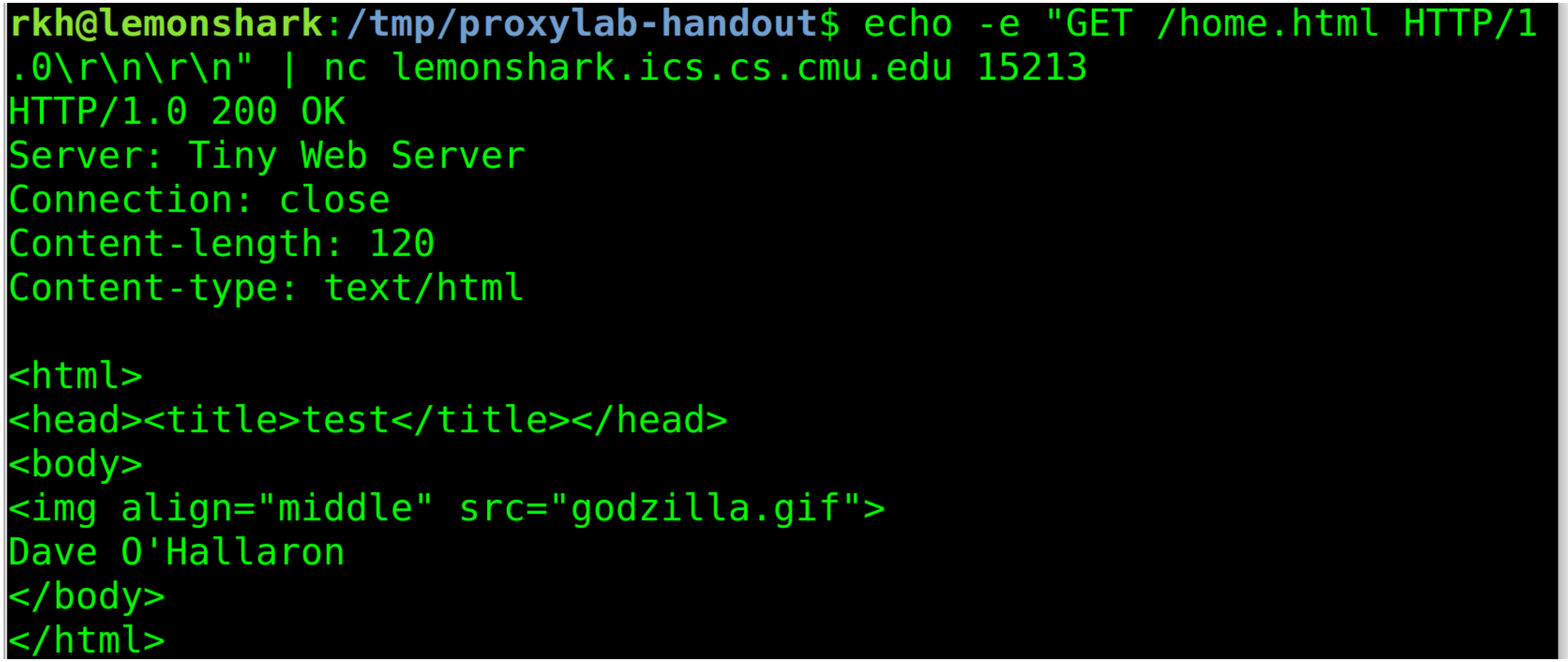


```
rkh@lemonshark:/tmp/proxylab-handout$ telnet lemonshark.ics.cs.cmu.e
du 15213
Trying 128.2.220.17...
Connected to lemonshark.ics.cs.cmu.edu (128.2.220.17).
Escape character is '^]'.
```

netcat (nc)

Like the cat program except with networking capabilities.

Can pipe strings or files to a target server and port.



```
rkh@lemonshark:/tmp/proxylab-handout$ echo -e "GET /home.html HTTP/1
.0\r\n\r\n" | nc lemonshark.ics.cs.cmu.edu 15213
HTTP/1.0 200 OK
Server: Tiny Web Server
Connection: close
Content-length: 120
Content-type: text/html

<html>
<head><title>test</title></head>
<body>
<img align="middle" src="godzilla.gif">
Dave O'Hallaron
</body>
</html>
```

netcat (nc)

Can also invoke as a listen server on a given port.

e.g. $ nc -l 15213

Easily usable in shell scripts to run tests.

Very versatile tool.

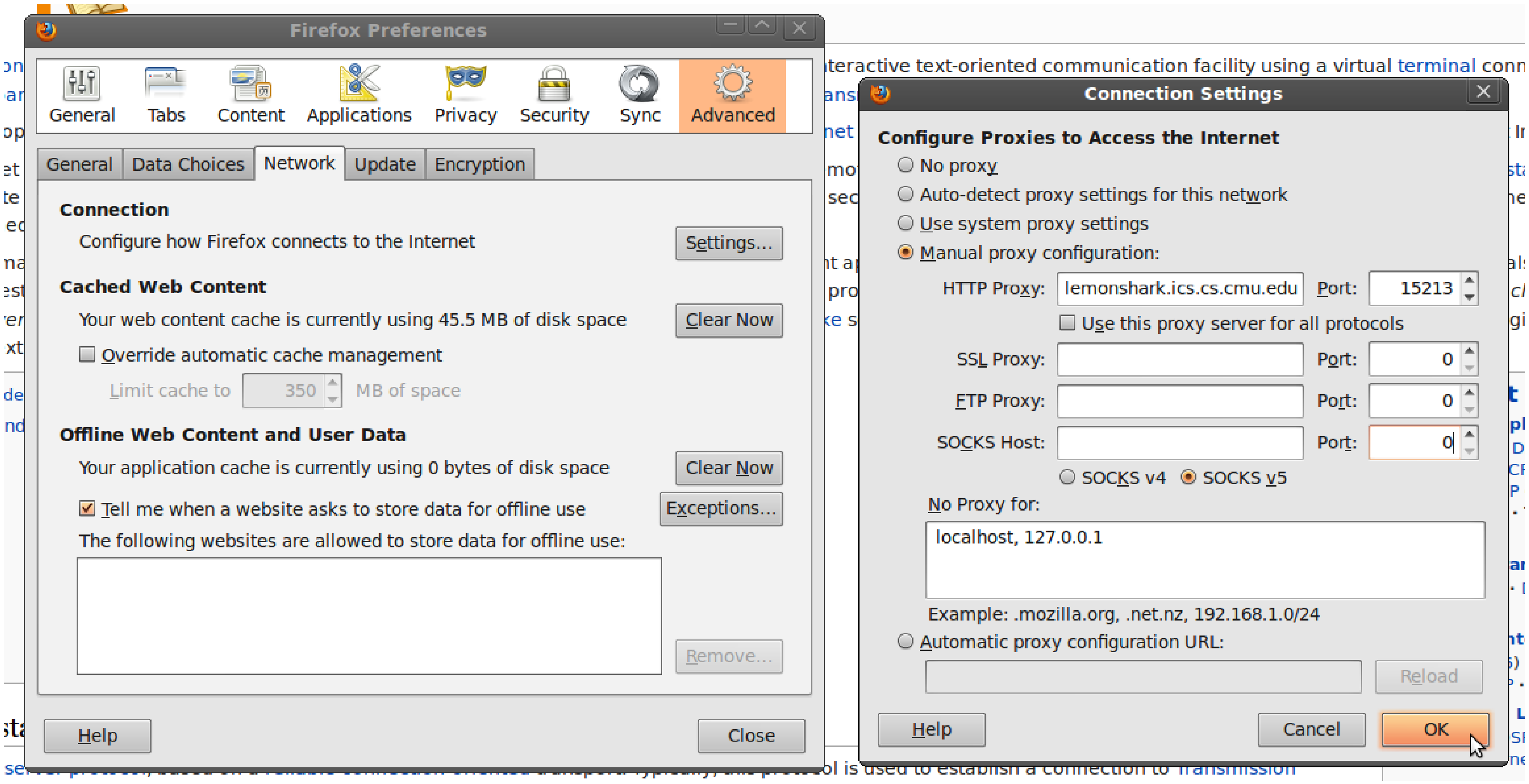
Valgrind

Very powerful memory debugging tool.

Useful in spotting memory leaks, illegal memory accesses, etc.

Your browser

Most modern browsers have a HTTP proxy option you can specify. Very useful in seeing if your proxy actually works with real-life usage.



Firefox Preferences

Connection

Configure how Firefox connects to the Internet — Settings...

Cached Web Content

Your web content cache is currently using 45.5 MB of disk space — Clear Now

☐ Override automatic cache management

Limit cache to 350 MB of space

Offline Web Content and User Data

Your application cache is currently using 0 bytes of disk space — Clear Now

☑ Tell me when a website asks to store data for offline use — Exceptions...

The following websites are allowed to store data for offline use:

Connection Settings

Configure Proxies to Access the Internet

○ No proxy
○ Auto-detect proxy settings for this network
○ Use system proxy settings
● Manual proxy configuration:

HTTP Proxy: lemonshark.ics.cs.cmu.edu Port: 15213

☐ Use this proxy server for all protocols

SSL Proxy: Port: 0

FTP Proxy: Port: 0

SOCKS Host: Port: 0

○ SOCKS v4 ● SOCKS v5

No Proxy for:

localhost, 127.0.0.1

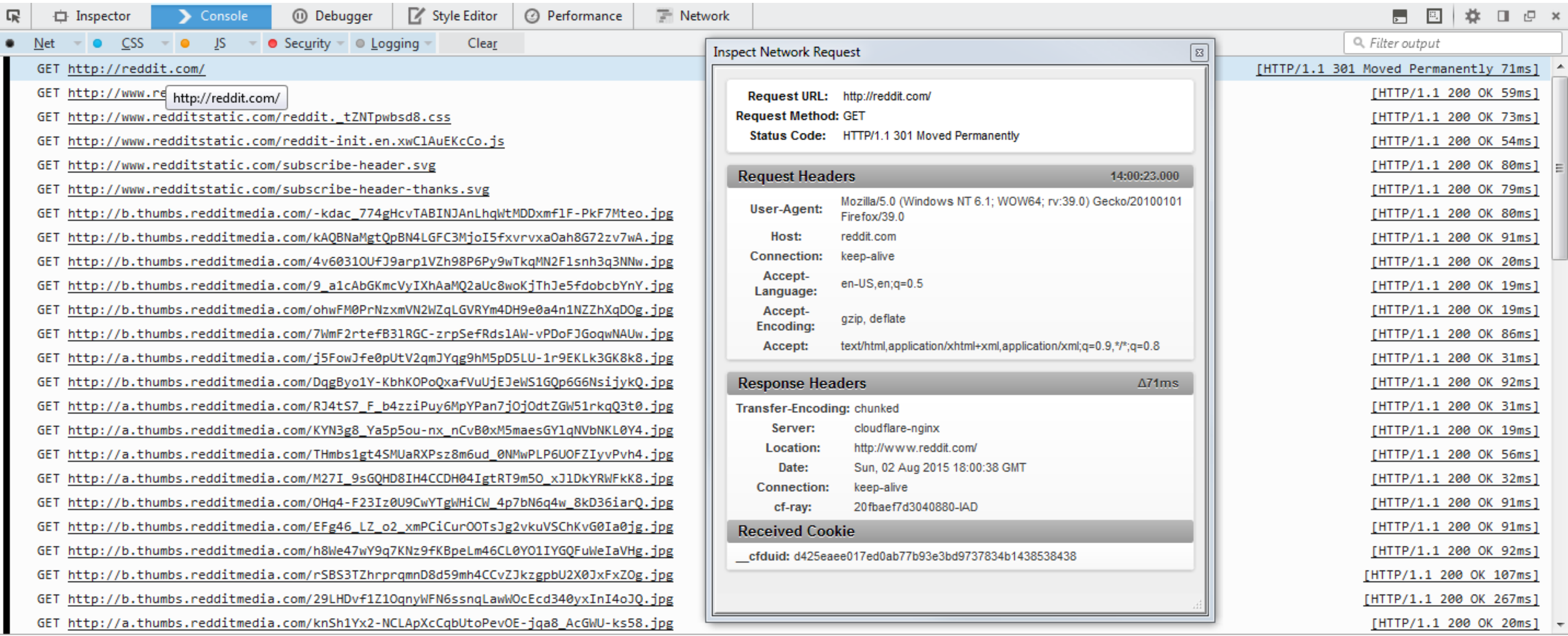
Example: .mozilla.org, .net.nz, 192.168.1.0/24

○ Automatic proxy configuration URL:

Your browser

Most modern browsers also come with a slew of networking analysis and debugging tools. You can log and inspect the requests and responses your browser makes and receives to and from your proxy and/or other web servers.

Try Ctrl+Shift+I for developer tools in Chrome or Ctrl+Shift+K for the web console in Firefox.



```
GET http://reddit.com/                                                                    [HTTP/1.1 301 Moved Permanently 71ms]
GET http://www.re  http://reddit.com/                                                     [HTTP/1.1 200 OK 59ms]
GET http://www.redditstatic.com/reddit._tZNTpwbsd8.css                                    [HTTP/1.1 200 OK 73ms]
GET http://www.redditstatic.com/reddit-init.en.xwClAuEKcCo.js                             [HTTP/1.1 200 OK 54ms]
GET http://www.redditstatic.com/subscribe-header.svg                                      [HTTP/1.1 200 OK 80ms]
GET http://www.redditstatic.com/subscribe-header-thanks.svg                               [HTTP/1.1 200 OK 79ms]
GET http://b.thumbs.redditmedia.com/-kdac_774gHcvTABINJAnLhqWtMDDxmflF-PkF7Mteo.jpg       [HTTP/1.1 200 OK 80ms]
GET http://b.thumbs.redditmedia.com/kAQBNaMgtQpBN4LGFC3MjoI5fxvrvxaOah8G72zv7wA.jpg       [HTTP/1.1 200 OK 91ms]
GET http://b.thumbs.redditmedia.com/4v6031OUfJ9arp1VZh98P6Py9wTkqMN2Flsnh3q3NNw.jpg       [HTTP/1.1 200 OK 20ms]
GET http://b.thumbs.redditmedia.com/9_a1cAbGKmcVyIXhAaMQ2aUc8woKjThJe5fdobcbYnY.jpg       [HTTP/1.1 200 OK 19ms]
GET http://b.thumbs.redditmedia.com/ohwFM0PrNzxmVN2WZqLGVRYm4DH9e0a4n1NZZhXqDOg.jpg       [HTTP/1.1 200 OK 19ms]
GET http://b.thumbs.redditmedia.com/7WmF2rtefB31RGC-zrpSefRds1AW-vPDoFJGoqwNAUw.jpg       [HTTP/1.1 200 OK 86ms]
GET http://a.thumbs.redditmedia.com/j5FowJfe0pUtV2qmJYqg9hM5pD5LU-1r9EKLk3GK8k8.jpg       [HTTP/1.1 200 OK 31ms]
GET http://b.thumbs.redditmedia.com/DqgByo1Y-KbhKOPoQxafVuUjEJeWS1GQp6G6NsijykQ.jpg       [HTTP/1.1 200 OK 92ms]
GET http://a.thumbs.redditmedia.com/RJ4tS7_F_b4zziPuy6MpYPan7jOjOdtZGW51rkqQ3t0.jpg       [HTTP/1.1 200 OK 31ms]
GET http://a.thumbs.redditmedia.com/KYN3g8_Ya5p5ou-nx_nCvB0xM5maesGY1qNVbNKL0Y4.jpg       [HTTP/1.1 200 OK 19ms]
GET http://a.thumbs.redditmedia.com/THmbs1gt4SMUaRXPsz8m6ud_0NMwPLP6UOFZIyvPvh4.jpg       [HTTP/1.1 200 OK 56ms]
GET http://a.thumbs.redditmedia.com/M27I_9sGQHD8IH4CCDH04IgtRT9m5O_xJ1DkYRWFkK8.jpg       [HTTP/1.1 200 OK 32ms]
GET http://b.thumbs.redditmedia.com/OHq4-F23Iz0U9CwYTgWHiCW_4p7bN6q4w_8kD36iarQ.jpg       [HTTP/1.1 200 OK 91ms]
GET http://b.thumbs.redditmedia.com/EFg46_LZ_o2_xmPCiCurOOTsJg2vkuVSChKvG0Ia0jg.jpg       [HTTP/1.1 200 OK 91ms]
GET http://b.thumbs.redditmedia.com/h8We47wY9q7KNz9fKBpeLm46CL0YO1IYGQFuWeIaVHg.jpg       [HTTP/1.1 200 OK 92ms]
GET http://b.thumbs.redditmedia.com/rSBS3TZhrprqmnD8d59mh4CCvZJkzgpbU2X0JxFxZOg.jpg       [HTTP/1.1 200 OK 107ms]
GET http://b.thumbs.redditmedia.com/29LHDvf1Z1OqnyWFN6ssnqLawWOcEcd340yxInI4oJQ.jpg       [HTTP/1.1 200 OK 267ms]
GET http://a.thumbs.redditmedia.com/knSh1Yx2-NCLApXcCqbUtoPevOE-jqa8_AcGWU-ks58.jpg       [HTTP/1.1 200 OK 20ms]
```

Inspect Network Request

Request URL:	http://reddit.com/
Request Method:	GET
Status Code:	HTTP/1.1 301 Moved Permanently

Request Headers 14:00:23.000

User-Agent:	Mozilla/5.0 (Windows NT 6.1; WOW64; rv:39.0) Gecko/20100101 Firefox/39.0
Host:	reddit.com
Connection:	keep-alive
Accept-Language:	en-US,en;q=0.5
Accept-Encoding:	gzip, deflate
Accept:	text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8

Response Headers Δ71ms

Transfer-Encoding:	chunked
Server:	cloudflare-nginx
Location:	http://www.reddit.com/
Date:	Sun, 02 Aug 2015 18:00:38 GMT
Connection:	keep-alive
cf-ray:	20fbaef7d3040880-IAD

Received Cookie

__cfduid: d425eaee017ed0ab77b93e3bd9737834b1438538438

Agenda

Administrivia

Proxy Lab Recap

HTTP Overview

Networking Tools

Debugging Tips & Techniques

Debugging Tips & Techniques

- Always go over your proxy's output with a fine-toothed comb.
- Use nc or tiny to display exactly what your proxy is sending.
- Try to avoid blindly using string functions like strcpy() and strlen() when handling data. Use strncpy() and strnlen() instead.
- Try to use large port numbers (e.g. 15213-65535) when running your proxy and tiny server(s) locally on the Shark machines. Smaller numbers might be reserved and already be in use by something else.

Debugging Tips & Techniques

- Make sure your underlying cache data structure implementation is correct before worrying about synchronisation. Maybe write a separate test program that includes/uses your cache library.

- Consider making your own testing framework. Save valid requests/responses your proxy breaks on to test on the proxy again later.

- Remember to include a method to monitor program flow while debugging race conditions and deadlocks.

Questions?

GOOD LUCK