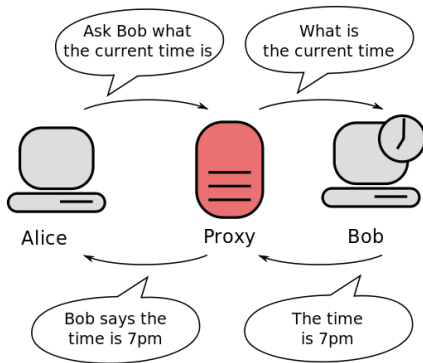http-proxy:

A library for writing HTTP

and HTTPS proxies

Erik de Castro Lopo

April 22, 2015

What's a proxy server?

A server that acts as an intermediary for requests from clients seeking resources from other servers.

HTTP Protocol:

Minimal client request:

```
GET /index.html HTTP/1.1

Host:  www.example.com
```

HTTP Protocol:

Example server response:

```
HTTP/1.1 200 OK
Date:  Wed, 22 April 2015 22:38:34 GMT
Last-Modified:  Wed, 08 Jan 2003 23:11:55 GMT
Content-Type:  text/html; charset=UTF-8
Content-Length:  138
Connection:  close


<html> .....
```

HTTP Proxy:

Minimal client request to a proxy:

```
GET http://www.example.com/index.html HTTP/1.1
```

The proxy:

- Rewrites this into the previous form and sends it upstream
- Gets the response and passes it back to the client.

For HTTP:

- All of the protocol is plain text.

- The proxy could potentially modify anything en-route.

- The proxy could cache the result of previous requests and provide the client with a cached response.

HTTPS Proxy:

Minimal client HTTPS request to a proxy:

```
CONNECT https://www.example.com/ HTTP/1.1
```

The **proxy** responds with:

```
HTTP/1.1 200 OK
```

and then simply passes bytes back and forth.

The http-proxy library:

- Built on top of libraries from the Yesod stack.

- Implemented as a `Wai` handler that is run in a `Warp` web
  server.

- Uses the wonderful `Asyc` library for the HTTPS part.

- Uses constant memory per connection.

- Has been used in anger.

- *Still* not complete!

The library spec:

- Work for HTTP and HTTPS.

- Constant max. memory usage per connection.

- Allow intercept and modification of HTTP requests and responses.

- Allow construction of a caching proxy.

- Allow construction of HTTP to HTTPS upgrade proxy.

- Allow construction of a MITM HTTPS proxy.

The Dependencies:

```
Build-Depends:

  base, async, blaze-builder, bytestring,

  bytestring-lexing, case-insensitive,

  conduit, conduit-extra, http-client,

  http-conduit, http-types, mtl, resourcet,

  tls, transformers, wai, wai-conduit, warp
```

The type:

```
proxyApp

      ::  Settings

      -> Manager

      -> Request

      -> (Response -> IO ResponseReceived)

      -> IO ResponseReceived
```

Questions?