Introduction to HTTP

Hypertext Transfer Protocol

HTTP is an application-level protocol for **distributed**, collaborative, **hypermedia** information systems.

History

- HTTP/0.9 - ~1999. Raw data transfer
- HTTP/1.0
  - MIME-like messages
  - Metainformation about the data
  - Better request/response semantics
- HTTP/1.1
  - Caching (Hierarchical proxies)
  - Persistent connections / Chunked encoding
  - Range requests
  - New methods (TRACE, PUT, DELETE, OPTIONS)

Key RFCs

- 2616: Hypertext Transfer Protocol -- HTTP/1.1
  - The Hypertext Transfer Protocol (HTTP)
  - http://tools.ietf.org/html/rfc2616
- 2617: HTTP Authentication: Basic and Digest Access Authentication
  - Basic authentication scheme and a scheme based on cryptographic hashes, referred to as "Digest Access Authentication".
  - http://tools.ietf.org/html/rfc2617
- 6265: HTTP State Management Mechanism
  - HTTP Cookie and Set-Cookie header fields
  - http://tools.ietf.org/html/rfc6265

Companion RFCs

- 2817: Upgrading to TLS Within HTTP/1.1
  - Upgrade mechanism in HTTP/1.1 to initiate Transport Layer Security (TLS) over an existing TCP connection (usually 80).
- http://tools.ietf.org/html/rfc2817

Concepts

- TCP/IP connections (only presumes a reliable transport)
- Default port 80
  - 443 for TLS connections
- URI: URL (Location) or Name (URN)

Concepts

- **Resource**: A network data object or service that can be identified by a URI. Resources may be available in multiple **representations** (e.g. multiple languages, data formats, size, and resolutions) or vary in other ways
- **Representation**: An entity included with a response that is subject to content negotiation

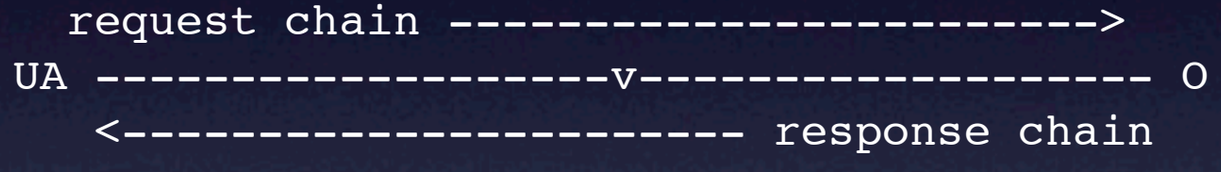
It's all about requests/ responses

- A client sends a **request** to the **server** in the form of a **request method**, **URI**, and **protocol version**, followed by a **MIME-like message** containing **request modifiers**, **client information**, and possible **body content** over **connection** with a server.
- curl -v -D - stackoverflow.com -o /dev/null
- curl -v -D - https://twitter.com/fielding -o /dev/null

...Responses

- The server responds with a **status line**, including the message's **protocol version** and a success or error **code**, followed by a **MIME-like message** containing **server information**, **entity metainformation**, and possible **entity-body content**.
- curl -i http://conekta.io

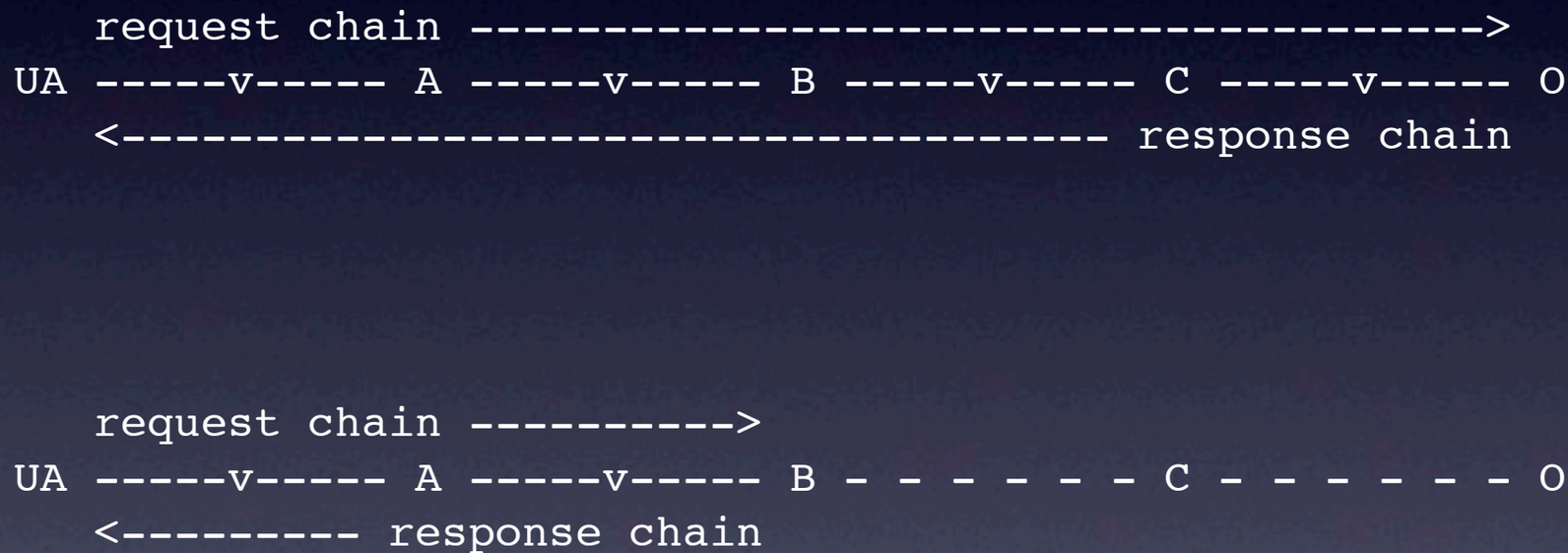
Communication



Communication with intermediaries

- Common Intermediaries:
  - **Proxy**: a forwarding agent
  - **Gateway**: a receiving agent, acting as a layer above some other server(s)
  - **Tunnel**: a relay point between two connections without changing the messages. E.g. firewalls

Communication with intermediaries



HTTPBis

- This Working Group is charged with maintaining and developing the "core" specifications for HTTP.
- Deliverables:
  - A document (or set of documents) that is suitable to supersede RFC 2616 as the definition of HTTP/1.1 and move RFC 2817 to Historic status
  - A document cataloguing the security properties of HTTP/1.1
  - A document (or set of documents) that specifies HTTP/2.0, an improved binding of HTTP's semantics to an underlying transport.

HTTP/1.1 Spec. modularity

- Several drafts for readibility and modularity
  - Messaging - low-level message parsing and connection management
  - Semantics - methods, status codes and headers
  - Conditional Requests - e.g., If-Modified-Since
  - Range Requests - getting partial content
  - Caching - browser and intermediary caches
  - Authentication - HTTP authentication framework