1. **Public Resources**

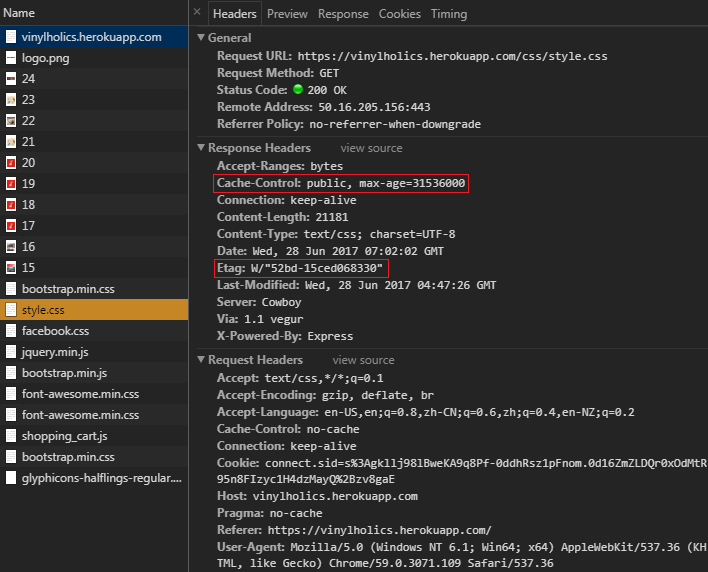
* css, scripts, fonts, static images like favicon or logo
* Cache for 1 year, public, weak ETag, stale resource allowed.
* Header example:

|  |
| --- |
| Cache-Control: public, max-age=31536000  ETag: W/"368d-15cc3a03146" |

* Expected result:
  + status 200
  + from local cache
  + have Cache-Control: public, max-age=31536000
  + have ETag
* Test result (screenshot):

1. First load of style.css



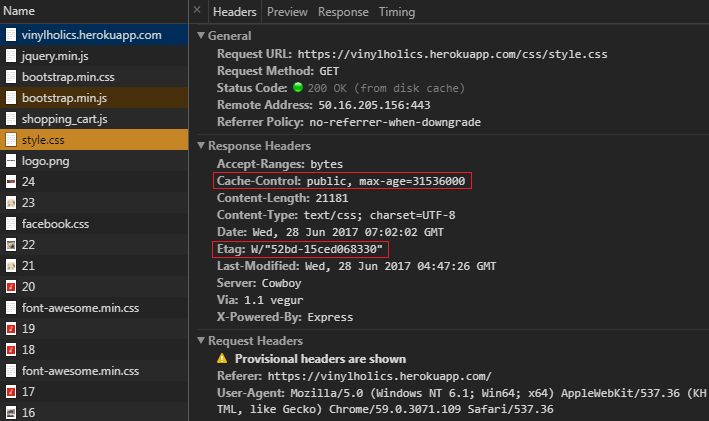


1. Reload of style.css

The resource is retrieved from disk cache, so there is no bandwidth use.

Significantly reduced the loading time from 492ms to 8ms.





1. **Relatively Static Resources**

* Images for albums or artists

Not-so-dynamic pages (browse artists, artist page, 404)

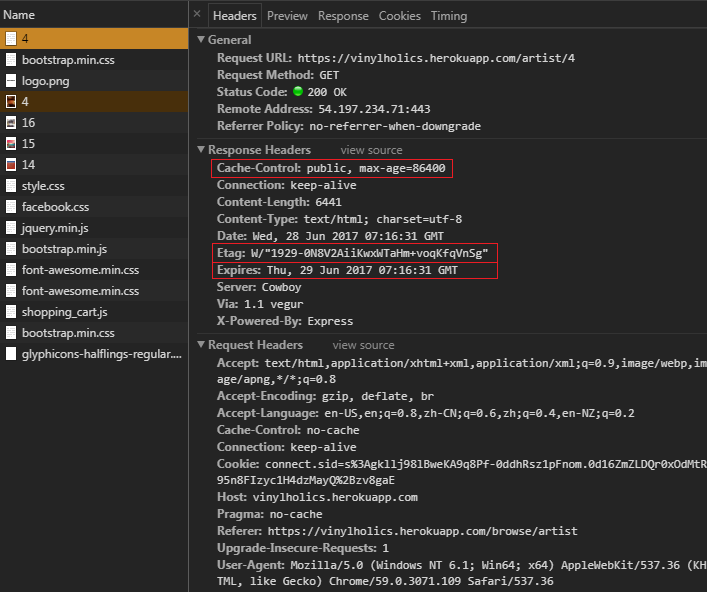
* Cache for 1 day, public, weak ETag, stale resource allowed.
* Header example:

|  |
| --- |
| Cache-Control: public, max-age=86400  ETag: W/"1885-jF9p9OVGhOjPGOXRlQu9TnQ+l0c"  Expires: Thu, 29 Jun 2017 02:45:31 GMT |

* Expected result:
  + status 200
  + from local cache
  + have Cache-Control: public, max-age=86400
  + have Expires: ... (only for old browser compatibility)
  + have ETag
* Test result (screenshot):

1. First load of https://vinylholics.herokuapp.com/artist/4



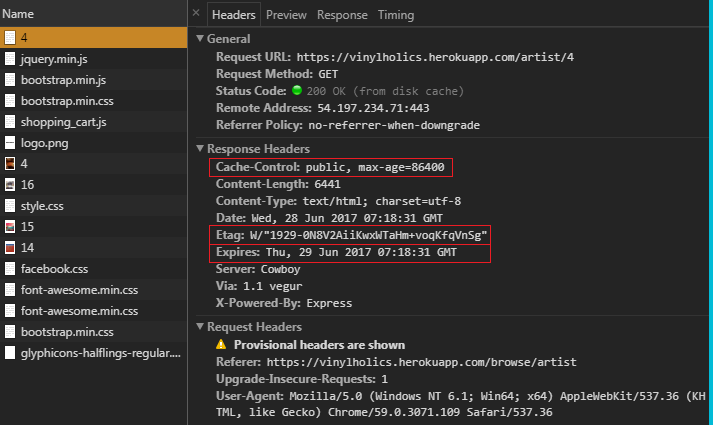


1. Reload of https://vinylholics.herokuapp.com/artist/4

The resource is retrieved from disk cache, so there is no bandwidth use.

Significantly reduced the loading time from 244ms to 2ms.





1. **Dynamic Pages**

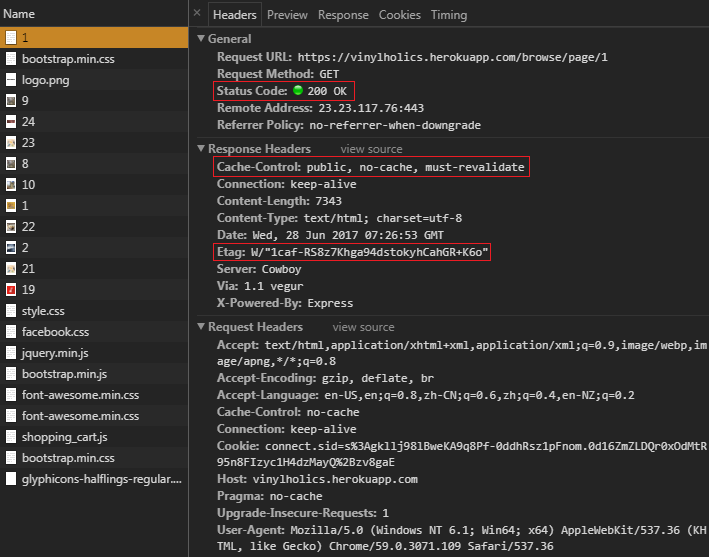
* Index, browse albums, album page
* Re-validate before using cached resource, public, weak ETag, stale resource not allowed.
* Header example:

|  |
| --- |
| Cache-Control: public, no-cache, must-revalidate  ETag: W/"1d97-t1Wdzs8mP0OTMek25jjTJosI/QY" |

* Expected result:
  + if not modified, status 304; if Modified, status 200.
  + if not modified, from local cache; if Modified, from server.
  + have Cache-Control: public, no-cache, must-revalidate
  + have ETag
* Test result (screenshot):

1. First load of https://vinylholics.herokuapp.com/browse/page/1

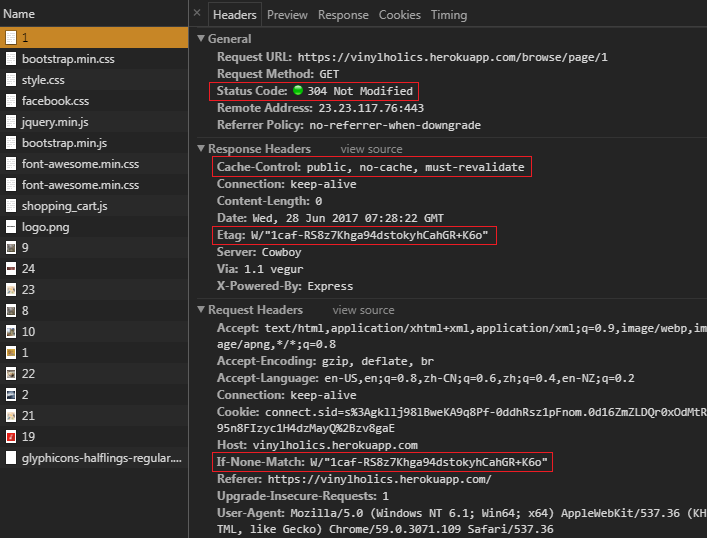




1. Reload of https://vinylholics.herokuapp.com/browse/page/1

The browser has to re-validate with server due to no-cache. Server replied with status 304 Not Modified, which means local cache can be used. Therefore, the resource is retrieved from disk cache which was saved in step a. There is a roundtrip for revalidation that consumes 258B of bandwidth.

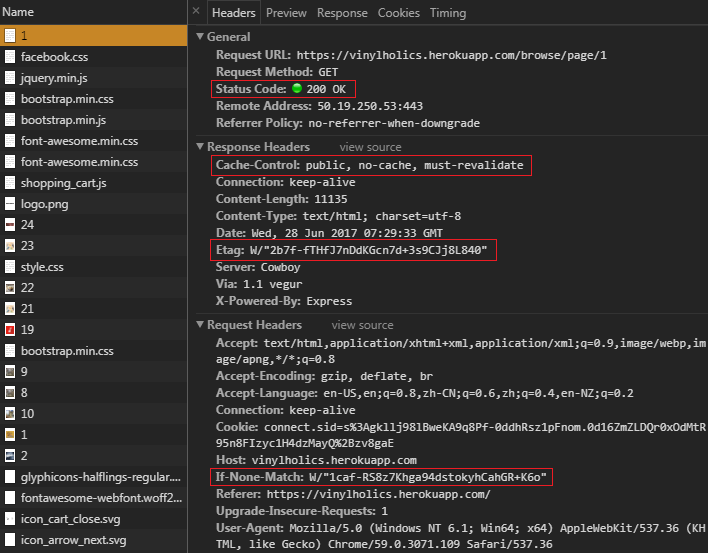




1. Log in to change the page content, then reload of https://vinylholics.herokuapp.com/browse/page/1

The browser tried to re-validate cached resource, but the server replied that the content has changed (ETag was different). Therefore, the new content was freshly retrieved from server.

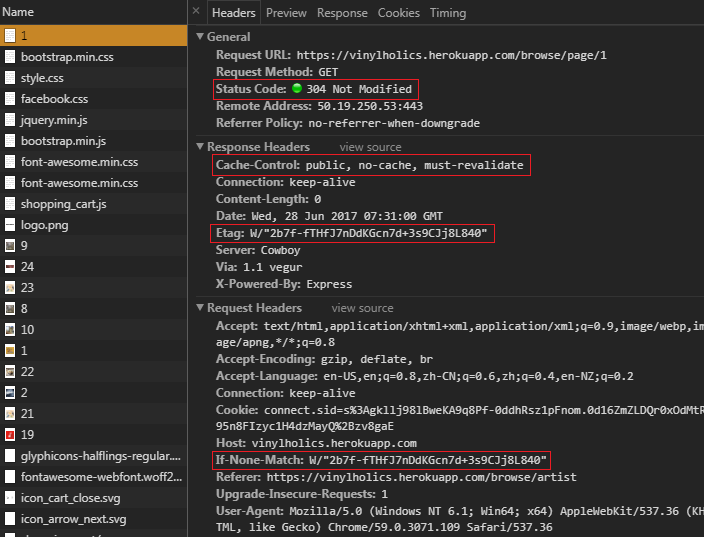




1. Reload of https://vinylholics.herokuapp.com/browse/page/1

This process is similar to step b. The only difference is that the cache in this step is from step c.





1. **Private content**

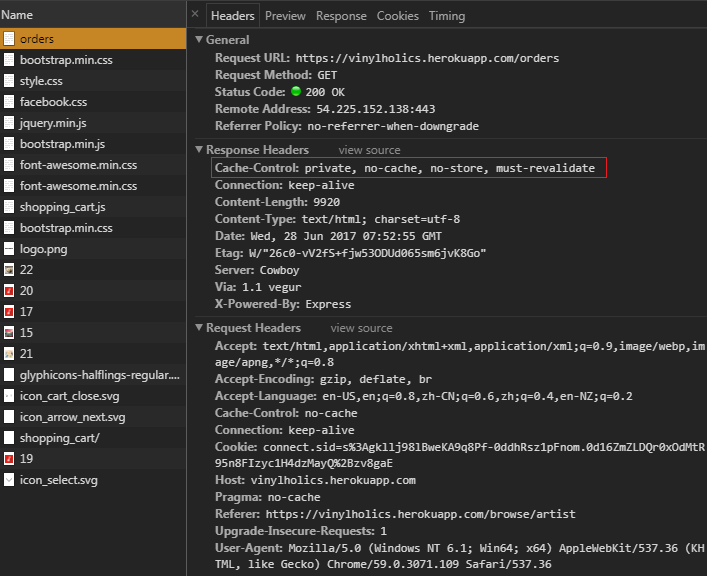
* Orders page, checkout page, user information
* No cache, no store, private, stale resource not allowed
* Header example:

|  |
| --- |
| Cache-Control: private, no-cache, no-store, must-revalidate |

* Expected result:
  + status 200
  + always from server
  + have Cache-Control: private, no-cache, no-store, must-revalidate
* Test result (screenshot):

1. First load of https://vinylholics.herokuapp.com/orders (user logged in)





1. Reload of https://vinylholics.herokuapp.com/orders (user logged in)

The browser cannot cache this resource due to no-cache, no-store. The content has to be fresh every time.

