

**Rule 1**

1. Make all style sheets into one file.

**Justification(s):**

All external stylesheets are combined into a single file and referenced in index.html just once so that the number of http requests to load the styles is reduced.

2. Make all scripts into one file.

**Justification(s):**

I. All external scripts are combined into a single file and referenced in index.html just once so that the number of http requests to load the scripts is reduced.

II. Each script inclusion using the src tag in the webpage launches a http request. We concluded that one analytics script would be good enough for tracking the website. Since, ga.js is a script from an external reference – we make sure that it is not included. Instead, we add umon2.1.js's script function(s) inside the common js file. This js file, is loaded towards the end of the body tag before before the analytic functions are accessed in index.html.

3. Remove 404s and 410s.

**Justification(s):**

I. Two files (swfobject.js and portalinclude\_v2.js) were found to not locate the resource it is referenced to and hence, were removed. This not only decreases the number of http requests but also does not change the user experience in any way.

II.

i. Both resources are javascripts which means that they block downloads of

elements below the <script> tag.

- ii. In most browsers, the javascript resource grabbing process blocks other parallel downloads.

*Such a phenomenon occurs since these scripts have the capability to change the DOM structure of the html page that is currently loading. In this particular case, a useless http request results in pausing the download of other elements of the page until the 404 response is received.*

III. After executing the download of the javascript, the engine parses the code obtained trying to find something usable in it. In this case, the 404 error delays the user from viewing other important elements of the page.

IV. This process of parsing an error http response during the load of index.html wastes resources as well.

4. Remove duplicate http requests.

Justification(s):

There were two GET requests to the same javascript file swfobject.js (one of it was a 404, since the resource location was different). The 404 duplicate http request was removed since that page was already loading the intended <script> functions from the valid http request.

5. Make sprites.

Justification(s):

- I. Instead of creating overheads by several http requests for each image that is being used in the page, the images are categorized into: repeatable (in the x-direction), repeatable (in the y-direction), repeatable (in both directions) and non-repeatable images.

II. Only static images (images that do not change with each page view or frequent enough) that belong to the website's domain are considered for this purpose because a single sprite image is pre-generated and used with parameters like background-position, width and height to display the part of the big image relevant to that particular element. The three images shown at the end of index.html are promo-s (can be changed frequently) and hence, excluded. Other images that are excluded from being part of the sprite are `get_flash_player.gif` and `privacy_certified-2008.gif`: this is because these images are loaded from a third party website and the ubi.com domain cannot take responsibility to update their sprite when the image (logo change etcetera) is initiated by the firm running those websites like adobe.

III. The padding between two images within the sprite is chosen in such a way that it is optimum.

- i. The horizontal offset should be large enough to avoid the background-repeat bug from Safari.
- ii. The vertical offset was not chosen to be zero, because that would interfere with the corresponding images being shown on the screen when the end-user increases the font.
- iii. Lesser padding will in turn, decrease the file size of the image allowing the browser to decompress and decode the image faster (mobile-friendly method).

IV. Different sprite images are created based on the repeat feature used by that image in the html element. For instance, all x-repeatable elements are aligned vertically and made symmetrical with respect to the width and vice-versa is done for y-repeatable elements.

Non-repeatable elements can be directly fed into the sprite generator.

- V. The non-repeating foreground images are also encompassed in the sprite files. The `img` tag in the `index.html` is replaced by the specification of the image as a background. This modification brought us to add an `onclick` event to the `<div>` tag with the image so instead of an image link originally present in the file. We ensured that all initial properties of the website was preserved. For instance, the `href` tag shows up a hand shaped cursor on hover. This property was added to the respective element in the new stylesheet element description.

### **Other Optimization Tweaks Applied**

1. Put style sheets on top.

#### **Justification(s):**

- I. Putting stylesheets to the top, makes the page appear to be faster because of progressive loading (especially for end-users with slow connection or pages with a lot of content).
- II. This mechanism also enhances the user experience because different elements of the page slowly appear indicating the progress level of the load.
- III. So why not put stylesheets at the bottom?
  - i. Some browsers like IE do not render the elements until the CSS is parsed because it might have to redraw the elements if otherwise.
  - ii. There is also a chance that the elements are unstyled while presented to the user during progressive load, which might not be appreciated by the user.

2. Put scripts towards the bottom.

#### **Justification(s):**

I. As mentioned above, scripts block parallel downloads (even those served from different host names to index.html). This mechanism allows the user to view the elements on the page progressively without pausing for a long time for the <javascript> to load.

II. It was also taken care that, these javascript functions were not referenced in index.html before the script source is included to avoid scoping issues.

### 3. Minification of CSS and JS.

#### Justification(s):

Minification removes comments and unnecessary characters. This improves the page speed because the file size of the above mentioned components is reduced.

### 4. Gzipping CSS and JS.

#### Justification(s):

This improves the page speed because the file size of the above mentioned components is reduced.

### 5. Make favicon.

#### Justification(s):

I. Most browsers send a request to fetch the favicon, by default. Since our website did not have one, we included an icon in the project to avoid bad requests. As long as there is a http request-response interchange, the icon (of small size) can be retrieved so that the end-user can use it for bookmarking purposes.

II. The favicon image chosen had to be small, unlike the logo itself because it interferes with the downloads of other components during page load.

III. When a page is requested multiple times, the favicon is retrieved from the server as a

cookie.

## 2. Compress sprite

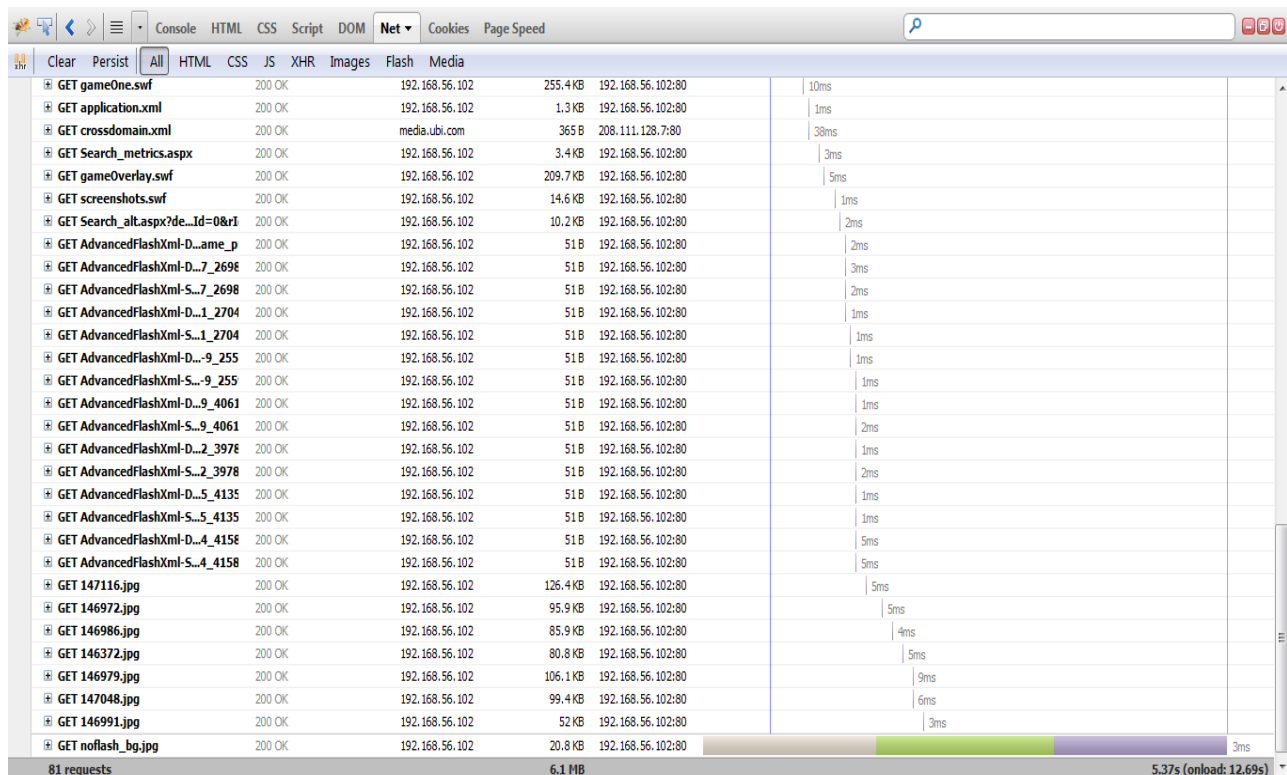
### Justification(s):

- I. The sprite image obtained is further compressed to reduce the file size.
- II. PNG format was chosen because it allows lossless compression.

*All the above results were determined and verified with the help of external tools - Google Page Speed, YSlow, Google Developer Tools, Sprite Generator Tool, Firebug, YUI Compressor etcetera.*

### Statistics

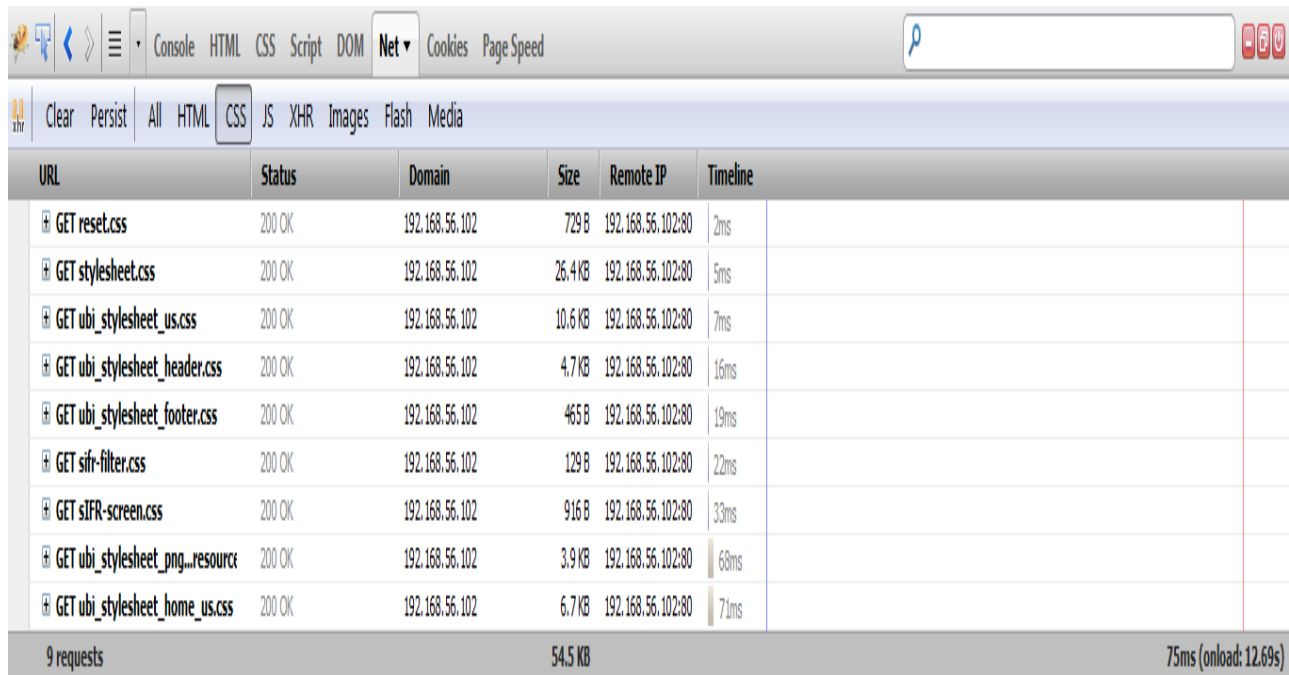
1. Below is a snapshot that shows the initial count of total http requests for the webpage to load (the cache was cleared before this screenshot was captured):



Resource	Size	Time
GET gameOne.swf	255.4 KB	10ms
GET application.xml	1.3 KB	1ms
GET crossdomain.xml	365 B	38ms
GET Search_metrics.aspx	3.4 KB	3ms
GET gameOverlay.swf	209.7 KB	5ms
GET screenshots.swf	14.6 KB	1ms
GET Search_alt.aspx?de...Id=0&rI	10.2 KB	2ms
GET AdvancedFlashXml-D...ame_p	51 B	2ms
GET AdvancedFlashXml-D...7_2698	51 B	3ms
GET AdvancedFlashXml-S...7_2698	51 B	2ms
GET AdvancedFlashXml-D...1_2704	51 B	1ms
GET AdvancedFlashXml-S...1_2704	51 B	1ms
GET AdvancedFlashXml-D...9_255	51 B	1ms
GET AdvancedFlashXml-S...9_255	51 B	1ms
GET AdvancedFlashXml-D...9_4061	51 B	1ms
GET AdvancedFlashXml-S...9_4061	51 B	2ms
GET AdvancedFlashXml-D...2_3978	51 B	1ms
GET AdvancedFlashXml-S...2_3978	51 B	2ms
GET AdvancedFlashXml-D...5_4135	51 B	1ms
GET AdvancedFlashXml-S...5_4135	51 B	1ms
GET AdvancedFlashXml-D...4_4158	51 B	5ms
GET AdvancedFlashXml-S...4_4158	51 B	5ms
GET 147116.jpg	126.4 KB	5ms
GET 146972.jpg	95.9 KB	5ms
GET 146986.jpg	85.9 KB	4ms
GET 146372.jpg	80.8 KB	5ms
GET 146979.jpg	106.1 KB	9ms
GET 147048.jpg	99.4 KB	6ms
GET 146991.jpg	52 KB	3ms
GET noflash_bg.jpg	20.8 KB	3ms

81 requests 6.1 MB 5.37s (onload: 12.69s)

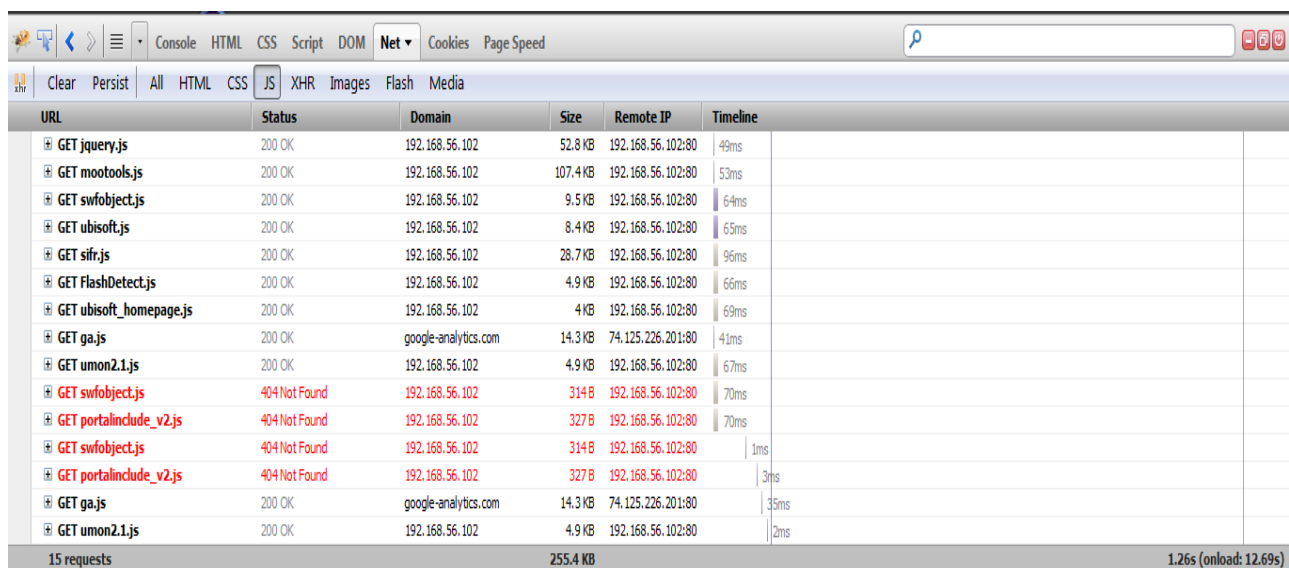
2. Below is a snapshot that shows the initial count of total http requests for retrieval of stylesheets (the cache was cleared before this screenshot was captured):



The screenshot shows the Chrome DevTools Network tab with the 'CSS' filter selected. The table lists 9 requests for various CSS files. The status for all requests is '200 OK'. The domain for all requests is '192.168.56.102'. The total size of the requests is 54.5 KB, and the total time taken is 75ms (onload: 12.69s).

URL	Status	Domain	Size	Remote IP	Timeline
GET reset.css	200 OK	192.168.56.102	729 B	192.168.56.102:80	2ms
GET stylesheet.css	200 OK	192.168.56.102	26.4 KB	192.168.56.102:80	5ms
GET ubi_stylesheet_us.css	200 OK	192.168.56.102	10.6 KB	192.168.56.102:80	7ms
GET ubi_stylesheet_header.css	200 OK	192.168.56.102	4.7 KB	192.168.56.102:80	16ms
GET ubi_stylesheet_footer.css	200 OK	192.168.56.102	465 B	192.168.56.102:80	19ms
GET sifr-filter.css	200 OK	192.168.56.102	129 B	192.168.56.102:80	22ms
GET sifr-screen.css	200 OK	192.168.56.102	916 B	192.168.56.102:80	33ms
GET ubi_stylesheet_png_resource	200 OK	192.168.56.102	3.9 KB	192.168.56.102:80	68ms
GET ubi_stylesheet_home_us.css	200 OK	192.168.56.102	6.7 KB	192.168.56.102:80	71ms
9 requests			54.5 KB		75ms (onload: 12.69s)

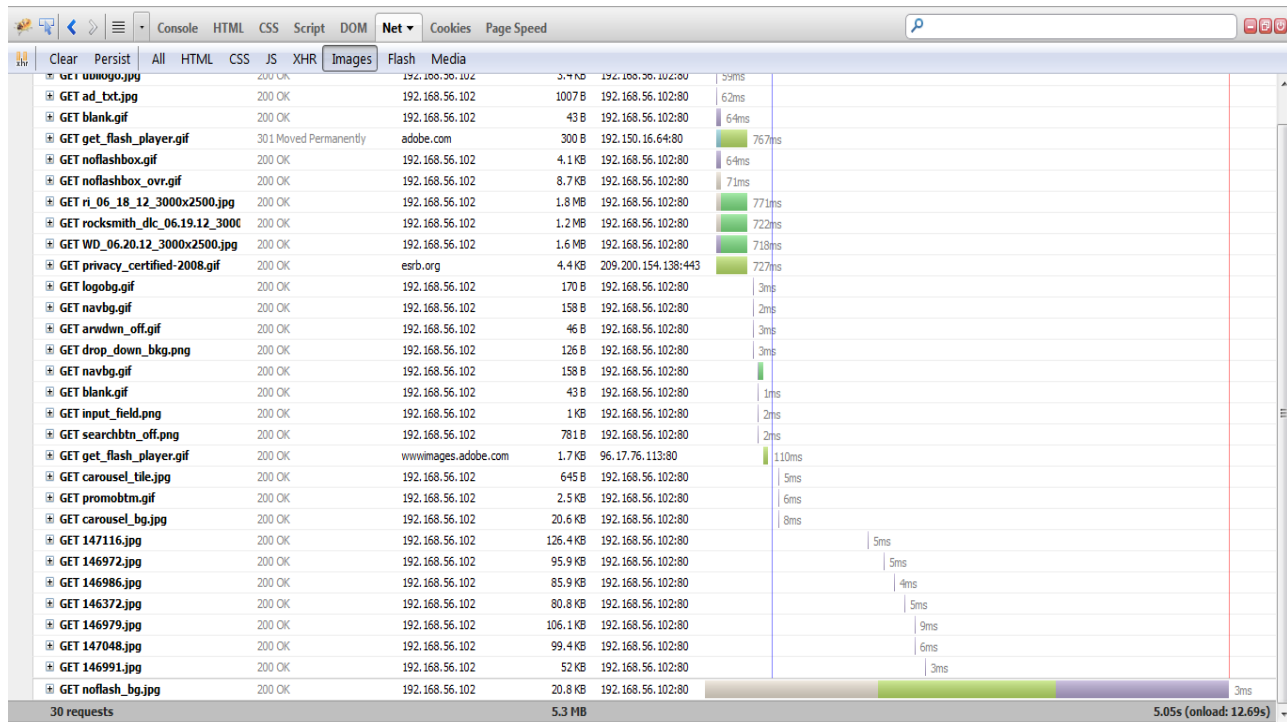
3. Below is a snapshot that shows the initial count of total http requests for retrieval of scripts (the cache was cleared before this screenshot was captured):



The screenshot shows the Chrome DevTools Network tab with the 'JS' filter selected. The table lists 15 requests for various JavaScript files. The status for most requests is '200 OK', but three requests (GET swfobject.js, GET portalinclude\_v2.js, and GET swfobject.js) have a status of '404 Not Found'. The domain for most requests is '192.168.56.102', but two requests (GET ga.js and GET umon2.1.js) have a domain of 'google-analytics.com'. The total size of the requests is 255.4 KB, and the total time taken is 1.26s (onload: 12.69s).

URL	Status	Domain	Size	Remote IP	Timeline
GET jquery.js	200 OK	192.168.56.102	52.8 KB	192.168.56.102:80	49ms
GET mootools.js	200 OK	192.168.56.102	107.4 KB	192.168.56.102:80	53ms
GET swfobject.js	200 OK	192.168.56.102	9.5 KB	192.168.56.102:80	64ms
GET ubisoft.js	200 OK	192.168.56.102	8.4 KB	192.168.56.102:80	65ms
GET sifr.js	200 OK	192.168.56.102	28.7 KB	192.168.56.102:80	96ms
GET FlashDetect.js	200 OK	192.168.56.102	4.9 KB	192.168.56.102:80	66ms
GET ubisoft_homepage.js	200 OK	192.168.56.102	4 KB	192.168.56.102:80	69ms
GET ga.js	200 OK	google-analytics.com	14.3 KB	74.125.226.201:80	41ms
GET umon2.1.js	200 OK	192.168.56.102	4.9 KB	192.168.56.102:80	67ms
GET swfobject.js	404 Not Found	192.168.56.102	314 B	192.168.56.102:80	70ms
GET portalinclude_v2.js	404 Not Found	192.168.56.102	327 B	192.168.56.102:80	70ms
GET swfobject.js	404 Not Found	192.168.56.102	314 B	192.168.56.102:80	1ms
GET portalinclude_v2.js	404 Not Found	192.168.56.102	327 B	192.168.56.102:80	3ms
GET ga.js	200 OK	google-analytics.com	14.3 KB	74.125.226.201:80	35ms
GET umon2.1.js	200 OK	192.168.56.102	4.9 KB	192.168.56.102:80	2ms
15 requests			255.4 KB		1.26s (onload: 12.69s)

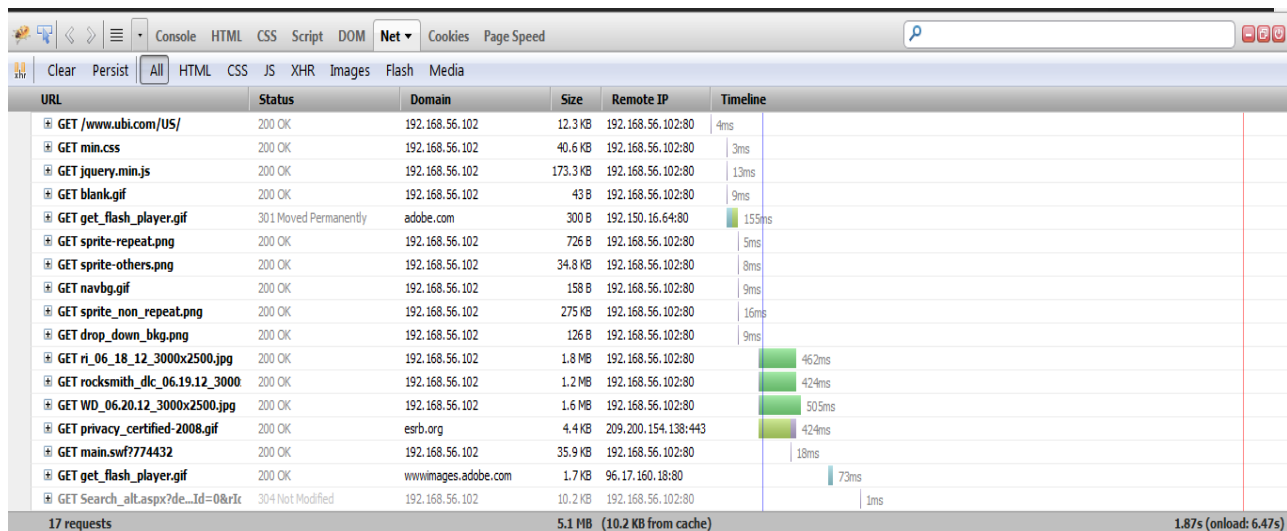
4. Below is a snapshot that shows the initial count of http requests for retrieval of images (the cache was cleared before this screenshot was captured):



This screenshot shows the initial state of the network tab with 30 requests. The requests are primarily for images, with a total size of 5.3 MB and a total time of 5.05s (onload: 12.69s).

URL	Status	Domain	Size	Remote IP	Timeline
GET /www.ubi.com/US/	200 OK	192.168.56.102	12.3 KB	192.168.56.102:80	4ms
GET min.css	200 OK	192.168.56.102	40.6 KB	192.168.56.102:80	3ms
GET jquery.min.js	200 OK	192.168.56.102	173.3 KB	192.168.56.102:80	13ms
GET blank.gif	200 OK	192.168.56.102	43 B	192.168.56.102:80	9ms
GET get_flash_player.gif	301 Moved Permanently	adobe.com	300 B	192.150.16.64:80	155ms
GET sprite-repeat.png	200 OK	192.168.56.102	726 B	192.168.56.102:80	5ms
GET sprite-others.png	200 OK	192.168.56.102	34.8 KB	192.168.56.102:80	8ms
GET navbg.gif	200 OK	192.168.56.102	158 B	192.168.56.102:80	9ms
GET sprite_non_repeat.png	200 OK	192.168.56.102	275 KB	192.168.56.102:80	16ms
GET drop_down_bkg.png	200 OK	192.168.56.102	126 B	192.168.56.102:80	9ms
GET ri_06_18_12_3000x2500.jpg	200 OK	192.168.56.102	1.8 MB	192.168.56.102:80	462ms
GET rocksmith_dlc_06.19.12_3000	200 OK	192.168.56.102	1.2 MB	192.168.56.102:80	424ms
GET WD_06.20.12_3000x2500.jpg	200 OK	192.168.56.102	1.6 MB	192.168.56.102:80	505ms
GET privacy_certified-2008.gif	200 OK	esrb.org	4.4 KB	209.200.154.138:443	424ms
GET main.swf?774432	200 OK	192.168.56.102	35.9 KB	192.168.56.102:80	18ms
GET get_flash_player.gif	200 OK	wwwimages.adobe.com	1.7 KB	96.17.160.18:80	73ms
GET Search_alt.aspx?de...Id=0&rlc	304 Not Modified	192.168.56.102	10.2 KB	192.168.56.102:80	1ms

5. Below is a snapshot that shows the current count of total http requests (the cache was cleared before this screenshot was captured):

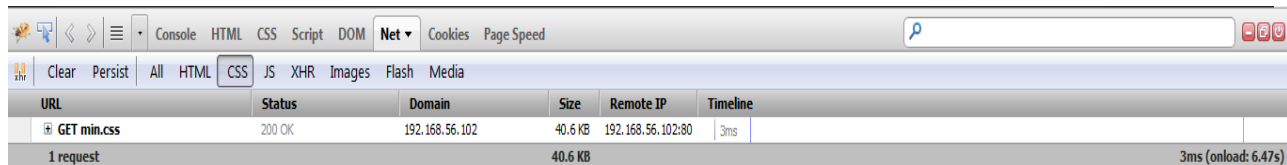


This screenshot shows the network tab after the initial load, with 17 requests. The total size is 5.1 MB (10.2 KB from cache) and the total time is 1.87s (onload: 6.47s).

URL	Status	Domain	Size	Remote IP	Timeline
GET /www.ubi.com/US/	200 OK	192.168.56.102	12.3 KB	192.168.56.102:80	4ms
GET min.css	200 OK	192.168.56.102	40.6 KB	192.168.56.102:80	3ms
GET jquery.min.js	200 OK	192.168.56.102	173.3 KB	192.168.56.102:80	13ms
GET blank.gif	200 OK	192.168.56.102	43 B	192.168.56.102:80	9ms
GET get_flash_player.gif	301 Moved Permanently	adobe.com	300 B	192.150.16.64:80	155ms
GET sprite-repeat.png	200 OK	192.168.56.102	726 B	192.168.56.102:80	5ms
GET sprite-others.png	200 OK	192.168.56.102	34.8 KB	192.168.56.102:80	8ms
GET navbg.gif	200 OK	192.168.56.102	158 B	192.168.56.102:80	9ms
GET sprite_non_repeat.png	200 OK	192.168.56.102	275 KB	192.168.56.102:80	16ms
GET drop_down_bkg.png	200 OK	192.168.56.102	126 B	192.168.56.102:80	9ms
GET ri_06_18_12_3000x2500.jpg	200 OK	192.168.56.102	1.8 MB	192.168.56.102:80	462ms
GET rocksmith_dlc_06.19.12_3000	200 OK	192.168.56.102	1.2 MB	192.168.56.102:80	424ms
GET WD_06.20.12_3000x2500.jpg	200 OK	192.168.56.102	1.6 MB	192.168.56.102:80	505ms
GET privacy_certified-2008.gif	200 OK	esrb.org	4.4 KB	209.200.154.138:443	424ms
GET main.swf?774432	200 OK	192.168.56.102	35.9 KB	192.168.56.102:80	18ms
GET get_flash_player.gif	200 OK	wwwimages.adobe.com	1.7 KB	96.17.160.18:80	73ms
GET Search_alt.aspx?de...Id=0&rlc	304 Not Modified	192.168.56.102	10.2 KB	192.168.56.102:80	1ms



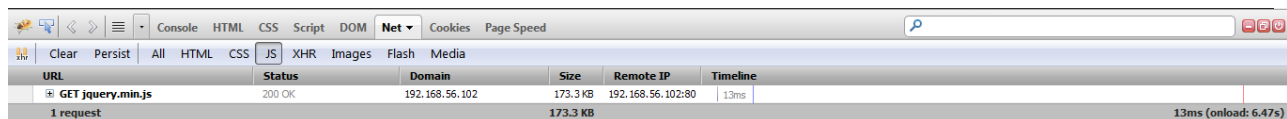
6. Below is a snapshot that shows the current count of http requests for retrieval of stylesheets (the cache was cleared before this screenshot was captured):



The screenshot shows the Chrome DevTools Network tab with the 'CSS' filter selected. A single request for 'min.css' is visible. The table below summarizes the request details.

URL	Status	Domain	Size	Remote IP	Timeline
GET min.css	200 OK	192.168.56.102	40.6 KB	192.168.56.102:80	3ms
1 request			40.6 KB	3ms (onload: 6.47s)	

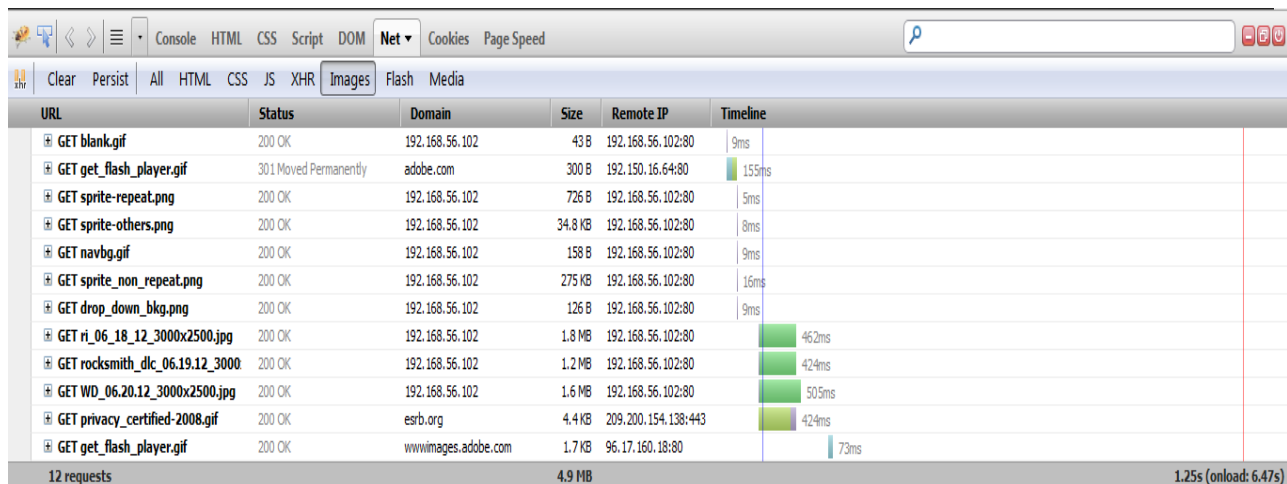
7. Below is a snapshot that shows the current count of http requests for retrieval of scripts (the cache was cleared before this screenshot was captured):



The screenshot shows the Chrome DevTools Network tab with the 'JS' filter selected. A single request for 'jquery.min.js' is visible. The table below summarizes the request details.

URL	Status	Domain	Size	Remote IP	Timeline
GET jquery.min.js	200 OK	192.168.56.102	173.3 KB	192.168.56.102:80	13ms
1 request			173.3 KB	13ms (onload: 6.47s)	

8. Below is a snapshot that shows the current count of http requests for retrieval of images (the cache was cleared before this screenshot was captured):



The screenshot shows the Chrome DevTools Network tab with the 'Images' filter selected. 12 requests for various images are visible. The table below summarizes the request details.

URL	Status	Domain	Size	Remote IP	Timeline
GET blank.gif	200 OK	192.168.56.102	43 B	192.168.56.102:80	9ms
GET get_flash_player.gif	301 Moved Permanently	adobe.com	300 B	192.150.16.64:80	155ms
GET sprite-repeat.png	200 OK	192.168.56.102	726 B	192.168.56.102:80	5ms
GET sprite-others.png	200 OK	192.168.56.102	34.8 KB	192.168.56.102:80	8ms
GET navbg.gif	200 OK	192.168.56.102	158 B	192.168.56.102:80	9ms
GET sprite_non_repeat.png	200 OK	192.168.56.102	275 KB	192.168.56.102:80	16ms
GET drop_down_bkg.png	200 OK	192.168.56.102	126 B	192.168.56.102:80	9ms
GET ri_06_18_12_3000x2500.jpg	200 OK	192.168.56.102	1.8 MB	192.168.56.102:80	462ms
GET rocksmith_dlc_06.19.12_3000	200 OK	192.168.56.102	1.2 MB	192.168.56.102:80	424ms
GET WD_06.20.12_3000x2500.jpg	200 OK	192.168.56.102	1.6 MB	192.168.56.102:80	505ms
GET privacy_certified-2008.gif	200 OK	esrb.org	4.4 KB	209.200.154.138:443	424ms
GET get_flash_player.gif	200 OK	wwwimages.adobe.com	1.7 KB	96.17.160.18:80	73ms
12 requests			4.9 MB	1.25s (onload: 6.47s)	