Do you know where your third paries are going?

Third party scripts on your site present a potentially dangerous side effect. We don't know what additonal assets third party load. That has security and performance implications. This post will look at the performance side of the issue.

As Simon Hearne writes in How to Find the Third-Parties on Your Site using third party scripts that, in turn, make calls to additional sites and parties outside their (and your) control can affect the performance of your site. Quoting the post:

> I ran the homepage through WebPageTest and sure enough, there were a bunch of calls to various subdomains of facebook.com. Thankfully WebPageTest stores initiator, referer and redirect headers; so with a little work you can find out where these third-party calls come from. The director was correct, there were no calls to Facebook on their site. It was a third-party creating fourth-party calls to Facebook! This can have serious ramifications if the Facebook (or any other third-party call) affects customer experience.

So why does this matter?

As Steve Souders (Frontend SPOF and Frontend SPOF survey), Pat Meenan (Testing for Frontend SPOF), and Joshua Bixby (How vulnerable is your site to third-party failure?) write, “fourth party” scripts (scripts called from third party code) can have adverse effects on your site’s performance. You (and potentially your client) have no way of controlling what these scripts do and whether they will block or slow down rendering.

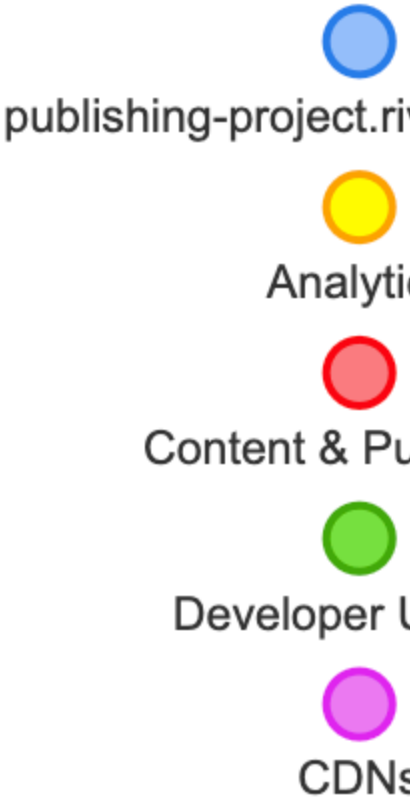
In Things to Know (and Potential Dangers) with Third-Party Scripts, Yaphi Berhanu list additional concerns about third party scripts that go beyond performance. The article is from 2017 still presents points that are still worth researching.

Testing

A very ineresting to see the number of requests your third party scripts generate is to run your site through Request Map and see see the results. I was surprised to see how many of its own submodules a script requested in the layout-experiments site.

The images that follow present Request Map results for three different websites, two sites I own and cnn.com.

For the CNN request map, note that the circles are smaller and the name of individual aassets are impossible to read because of the number of assets involved.


publishing-project.rivendellweb.net
Analytics
Content & Publishing
Developer Utilities
CDNs

res.

Figure 1: Requet map for rivendellweb.net. Created from Request Map Generator


layout-experiments.firebaseapp.com
Developer Utilities
Analytics
Tag Management

Figure 2: Requet map for layout-experiments.firebaseapp.com. Created from Request Map Generator

Figure 3: Requet map for CNN. Created from Request Map Generator

Solving the issue

In an ideal world you'd be able to trust your third party scripts that they will do a minimum due dilligence on their dependencies.

But in the real world we don't have that luxury. We use what we must and we hope that scripts that are loaded by our third party scripts will not slow down our applications.

Tools like request map will not stop the spread of unknown and unwanted scripts but they will definitely help when clients ask you where did that script come from and when you go to the third party script providers and ask why are the additional assets being loaded.