Bypass Redirect Filters

"Unvalidated Redirects and Forwards" used to be its own item in the OWASP Top Ten. It's still around, and it gives us an opportunity to look at how to get around filters.

Ideas

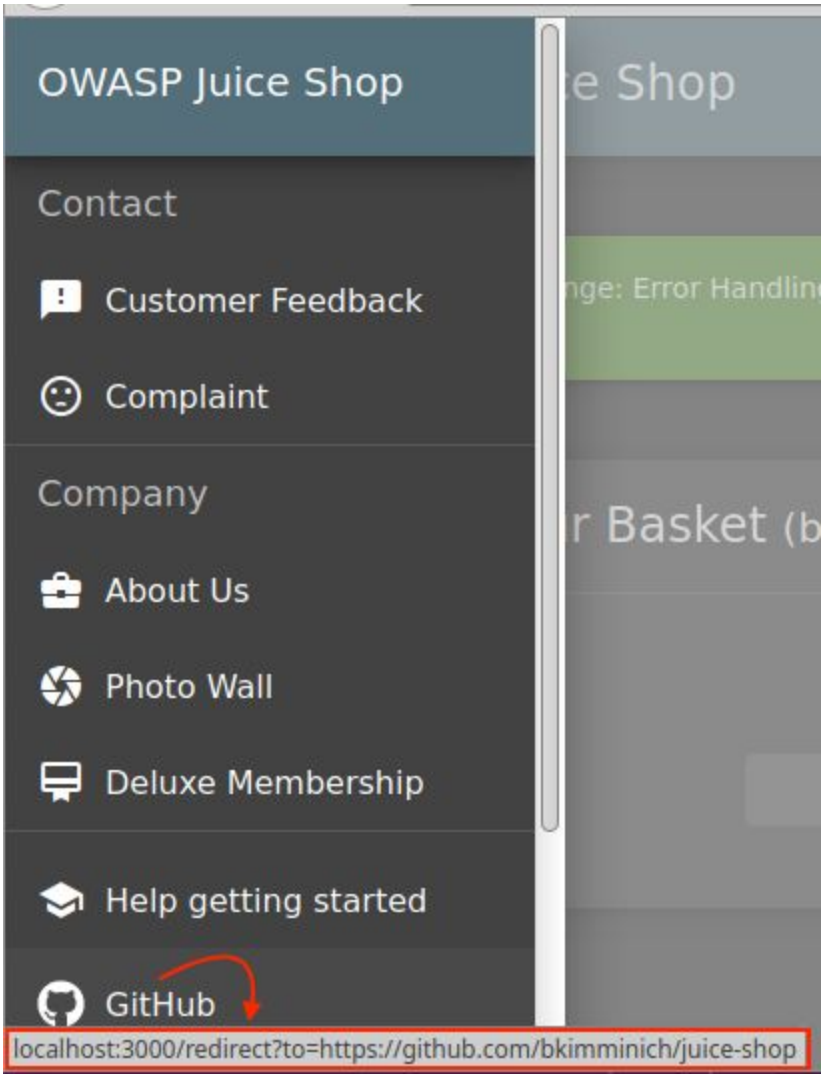
Any time you see a resource that accepts a URL as a parameter, and sends you to that URL as part of its request handling, you should try to get it to send you to other places.

These "unvalidated redirects" can be used in phishing: send a user to a legitimate site's redirector, so they see the expected hostname in the URL, then bounce them to a server you control where you show them a page that looks just like the target site's login form. See if they give you their credentials...

Walk-Thru

Make sure you have Firefox set to use your Burp Suite as a proxy, and that the Proxy > Intercept pane says "Intercept is off"

1. Notice the off-site link to GitHub, under the left-side hamburger menu, is not a direct link, but passes through a redirect script.



Redirect Script

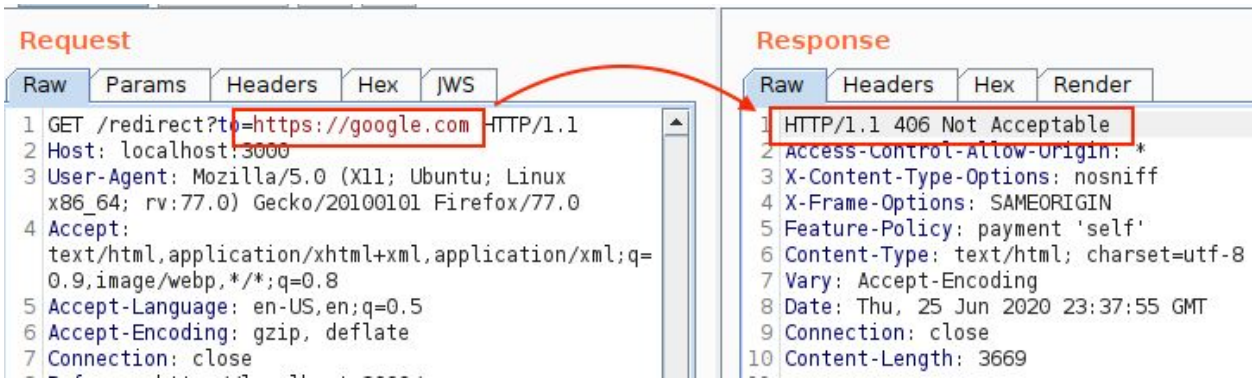
2. Click that link and notice you end up on the Juice Shop project at GitHub

3. Find the request in your Burp Proxy History and send it to Repeater

4. Send it from Repeater to make sure it still works

5. Change the destination URL (the "to" parameter) from `https://github.com/bkimminich/juice-shop` to `https://google.com` and send that request.

6. Notice an HTTP 406 "Not Acceptable" error

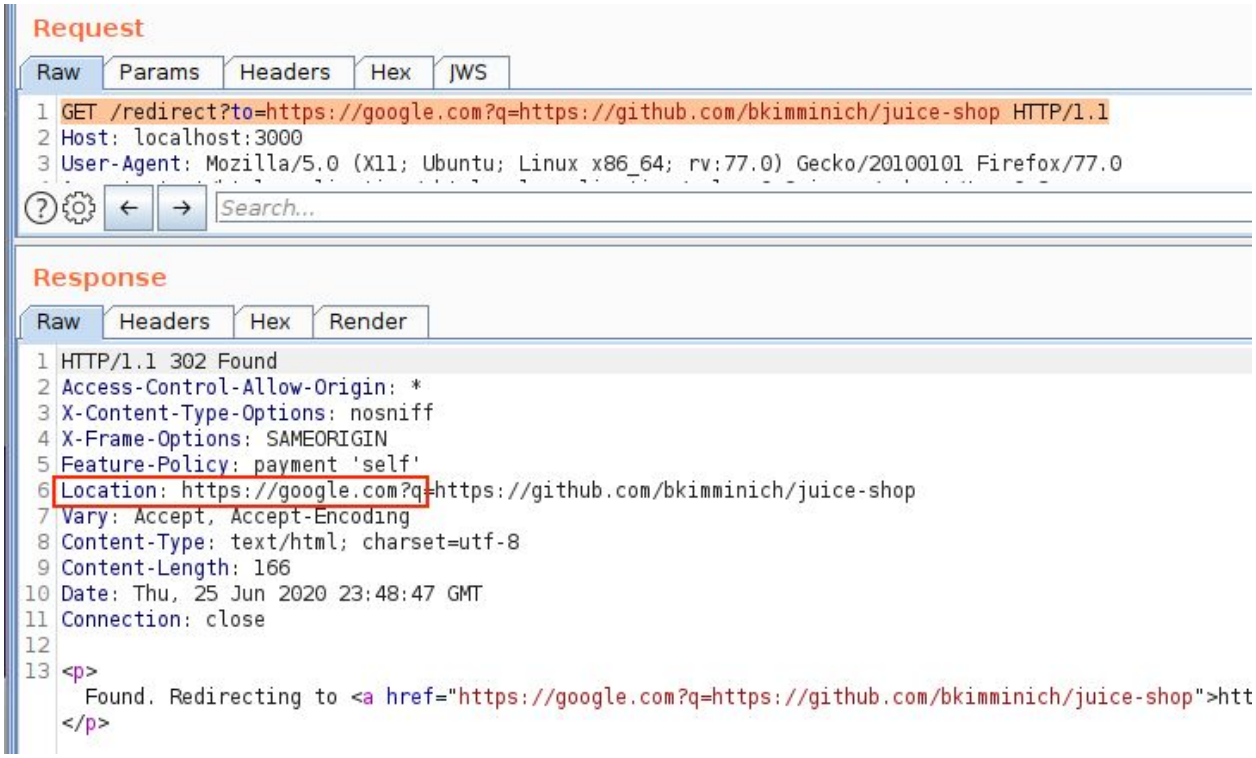


HTTP 406 - NOT ACCEPTABLE!!!!!!11!

7. How can you create a URL that you control but that also includes the original github URL?

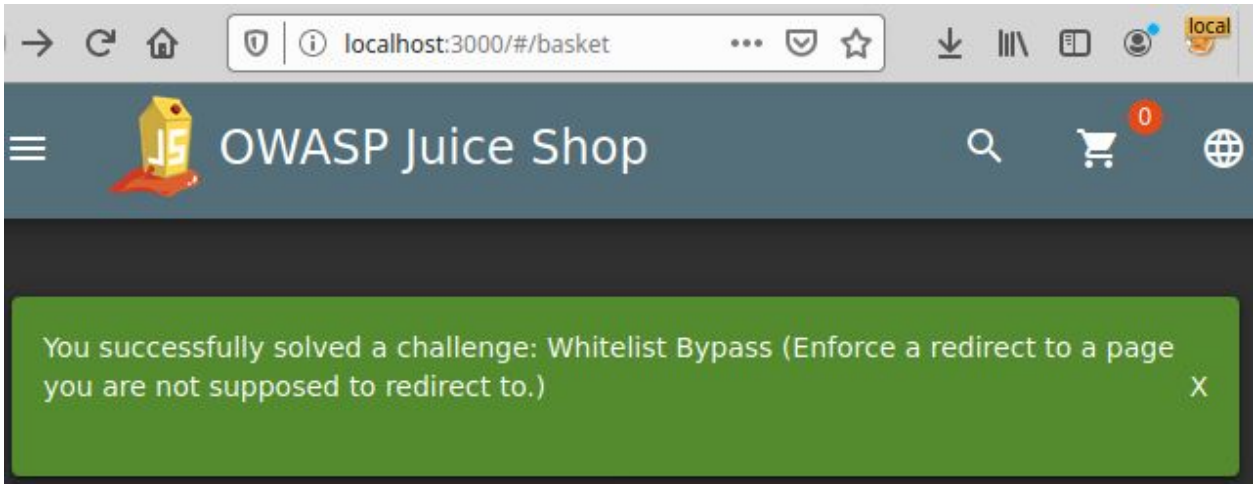
8. Add the original URL in a query string you pass to a different domain. We'll use Google here, but an attacker would use a site under their control, and made up to look like part of Juice Shop where it would try to get the visitor to give away secrets. Maybe the page says the session has timed out and prompts the user to log in again, stealing their credentials.

9.
Try this URL: `https://google.com?q=https://github.com/bkimminich/juice-shop`
The first line of your GET request should look like this, then:
`GET /redirect?to=https://google.com?q=https://github.com/bkimminich/juice-shop HTTP/1.1`



Successful Exploit of Unvalidated Redirect

10. Return to the browser to receive your reward:



References

OWASP Top Ten: Changes from 2013 to 2017
https://owasp.org/www-project-top-ten/OWASP_Top_Ten_2017/Top_10-2017_Release_Notes