Web Cache Deception Attack

Exploit

1. Browser requests `http://www.example.com/home.php/non-existent.css`.
2. Server returns the content of `http://www.example.com/home.php`, most probably with HTTP caching headers that instruct to not cache this page.
3. The response goes through the proxy.
4. The proxy identifies that the file has a css extension.
5. Under the cache directory, the proxy creates a directory named home.php, and caches the imposter "CSS" file (non-existent.css) inside.

Methodology of the attack - example

1. Normal browsing, visit home : `https://www.example.com/myaccount/home/`
2. Open the malicious link : `https://www.example.com/myaccount/home/malicious.css`
3. The page is displayed as /home and the cache is saving the page
4. Open a private tab with the previous URL : `https://www.paypal.com/myaccount/home/malicous.css`
5. The content of the cache is displayed

 (https://www.youtube.com/watch?v=pLte7SomUB8)

Video of the attack by Omer Gil - Web Cache Deception Attack in PayPal Home Page

Thanks to

- Web Cache Deception Attack - Omer Gil (http://omergil.blogspot.fr/2017/02/web-cache-deception-attack.html)
- Practical Web Cache Poisoning - James Kettle @albinowax (https://portswigger.net/blog/practical-web-cache-poisoning)