Intercepting proxies

Header Maniopulation with Burp Suite

- A1) Do you found an article that was restricted for you but after manipulating the UA string you were able to read it? What is the URL of the article? (one example is enough).
    - Answer: https://www.nzz.ch
- A2) Why do you think this website allows bypassing the paywall with a UA string similar to Googlebot?
    - Answer: the articles - even if they are restricted - should end up in the search results. Potentially users might even buy an abo if they end up on a restricted article after googling for a topic.
- A3) How would you improve the enforcement of the paywall, so that Googlebot can still access the articles but not a regular user?
    - Answer: it seems nzz.ch doesnt use all the fingerprint characteristics available to determine whether it is a google bot or not. So here what could be done:
        - use more fingerprints available at server side (makes it a bit more difficult)
        - use client side fingerprints: only possible if google bots know how to execute js and the dynamic html etc. Would probably require the article to be downloaded via ajax after a positive identification. This might not be a realistic options, as it would require additional changes and I'm also not sure how stable the google bot characteristics are.
        - maybe whitelist google bot ip's. (requires them to be in a certain range). This would probably be quite effective.

Interactive request/response mani0pulation with burp suite

- Question B1) Add an additional header to the request with a very long header name, so that the server returns an error. The error message reveals the web server (proxy server) product name and version. Which web server is used?
    - nginx/1.18.0
- Question B2) How does the HTTP body containing username and password look like? Which character is used to separate the parameters from each other?
    - "username=xxxx@xxx.ch&password=aaaa&credentialId="
        - separating character: "&"
- Question B3) Change the Content-Type response header value from text/html to text/plain. How is the website now displayed in Firefox?
    - HTML content is not interpreted. We see the the html as plain text (similar to view-source)

Fuzzing (getting using ZAP fuzzer)

C1) Value retrieved via fuzzing: ka83f342a

Remark: using a file for the different versions of a username didnt work for me. I used a regex

Forcefull browsing

- D1) What is the URL and the content (title) of the site?
    - Answer: https://d80a5d81-19ce-406a-9dc2-ab3490f96d62.idocker.vuln.land/admin
- D2) How could the operator of a web application detect and prevent such a scan?

  - Answer: an admin portal that is not public should not return an 200 Statuscode but rather an error code if user is not authorized. Usually, that us a 403 (status code) but to prevent the scan from detecting that there is a site, it would have to return a 404 even if that is not fully correct.