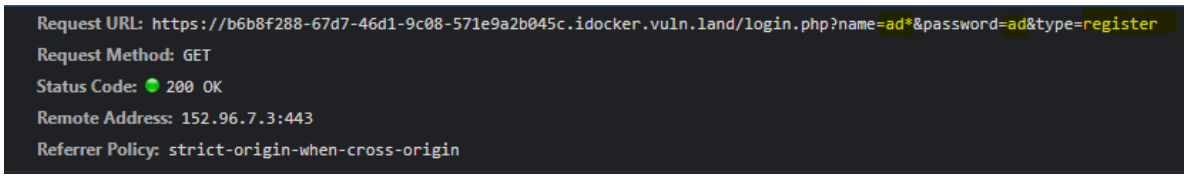
The vulnerability

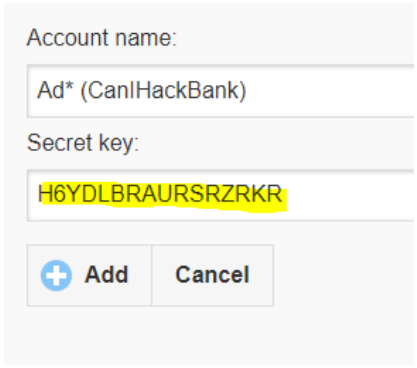
The Bank Application presumably works with LDAP. LDAP has a wildcard character (*) which allows the search/filter to match all occurrences in the parameter. This can be exploited by creating an account that starts with a substring of “admin” and ends with a trailing wildcard (*). This can be used in the filter statement used to distinguish users from admins which ultimately allows an attacker to get the TOTP (otpauth) secret of any account (including the admin account). The wildcard also allows an attacker to enter the wildcard as a password which will bypass the password login (not the 2FA).

The process

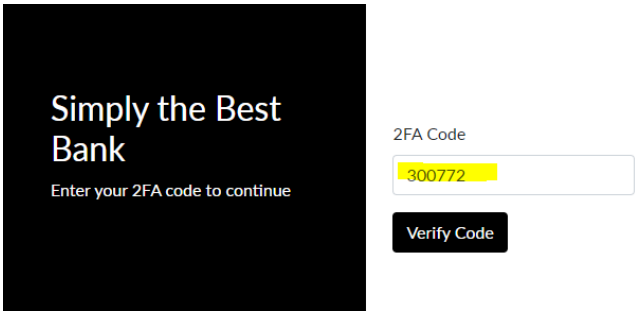
1. Create a user that contains a substring of “admin” & ends with a “*”.
   Here the username “ad*” and password “ad” have been chosen.



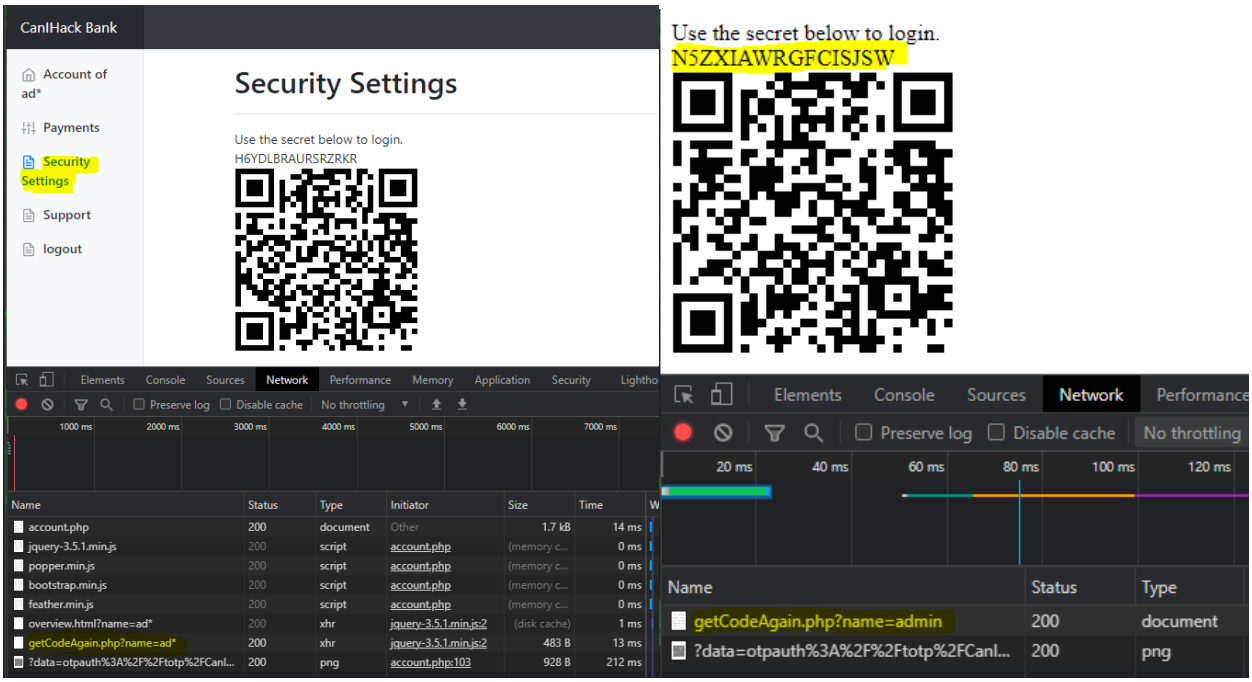
2. Use the “gauth” web app (https://gauth.apps.gbraad.nl/) to generate/manage the TOTP codes.
   Add the secret key of the newly created account to the TOTP list.



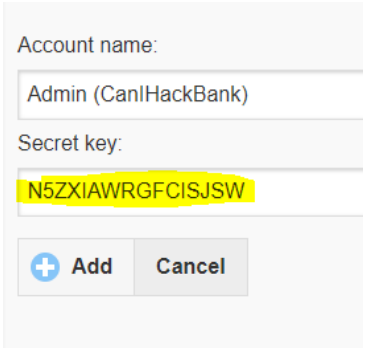
3. Login with the newly created user & enter the 2FA code generated on the “gauth” web app.



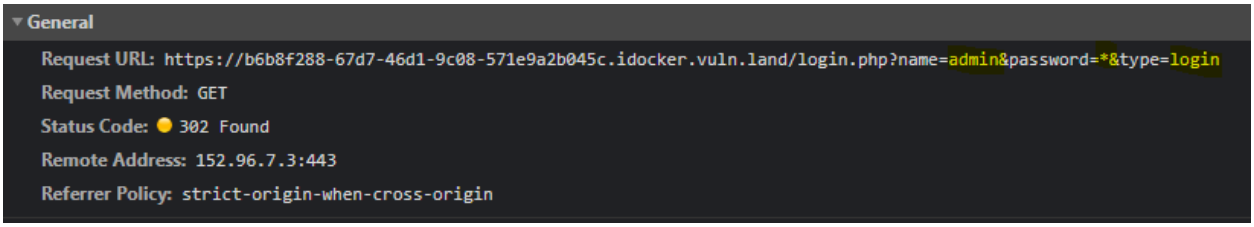
4. The “Security Settings” page makes a request to “/getCodeAgain.php?name=*<username>*”. We can make a request to “/getCodeAgain.php?name=admin” to get the secret key of the admin.



5. Add the secret of the admin to “gauth”.



6. Login with the user “admin” and password “*”. The wildcard will allow us to bypass the password login.



7. Now enter the code from “gauth” as the 2FA verification code & get the flag from the main page.