Praktikum Keamanan Jaringan

OWASP Juice Shop – Cryptographic Failures

Oleh:

Aldo Faiz Winarno (3122640039)

D4 LJ IT B

POLITEKNIK ELEKTRONIKA NEGERI SURABAYA

TAHUN 2023

Cryptographic Failures

Sumber : https://owasp.org/Top10/id/A02_2021-Cryptographic_Failures/

Deskripsi

Hal pertama adalah menentukan kebutuhan perlindungan data dalam perjalanan dan pada saat istirahat. Misalnya, kata sandi, nomor kartu kredit, catatan kesehatan, informasi pribadi, dan rahasia bisnis yang memerlukan ekstra perlindungan, terutama jika data tersebut termasuk dalam undang-undang privasi, misalnya, General Data Protection Regulation (GDPR) Uni Eropa, atau peraturan, misalnya, perlindungan data keuangan seperti PCI Data Security Standard (PCI DSS). Untuk semua data tersebut:

- Apakah ada data yang dikirimkan dalam bentuk teks yang jelas? ini menyangkut protokol seperti HTTP, SMTP, and FTP. Lalu lintas internet luar yang berbahaya. Verifikasi semua lalu lintas yang ada di internal, misalnya antara penyeimbang beban, server web, atau sistem back-end.
- Apakah ada algoritma kriptografi lama atau lemah yang digunakan baik secara default atau dalam kode yang lebih lama?
- Apakah kunci kripto bawaan sedang digunakan, kunci kripto yang lemah dihasilkan atau digunakan kembali, atau apakah kurangnya manajemen atau rotasi kunci yang tepat?
- Apakah enkripsi tidak diterapkan, misalnya, apakah ada agen pengguna (browser) yang arahan atau header keamanan hilang?
- Apakah agen pengguna (misalnya, aplikasi, klien email) tidak memverifikasi jika sertifikat yang diterima server valid?

Lihat ASVS Crypto (V7), Data Protection (V9), dan SSL/TLS (V10)

Cara Mencegah

Lakukan minimal hal berikut, dan lihat referensi:

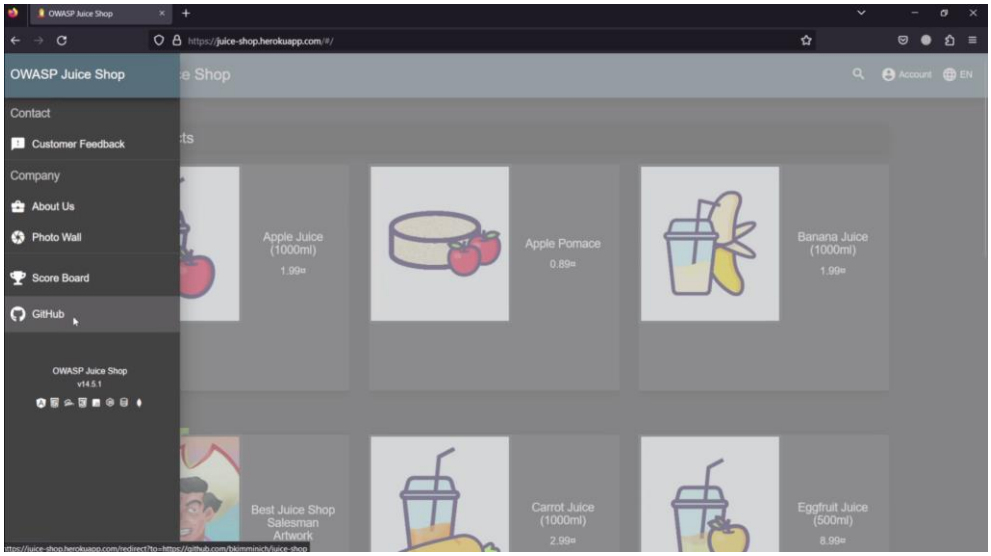
- Mengklasifikasikan data yang diproses, disimpan, atau dikirim oleh aplikasi. Identifikasi data mana yang sensitif menurut undang-undang privasi, persyaratan peraturan, atau kebutuhan bisnis.
- Tetapkan kontrol sesuai klasifikasi.
- Jangan menyimpan data sensitif yang tidak perlu. Buang sesegera mungkin atau gunakan tokenisasi yang sesuai dengan PCI DSS atau bahkan pemotongan. Data yang tidak disimpan tidak dapat dicuri.
- Pastikan untuk mengenkripsi semua data sensitif saat istirahat.
- Pastikan gunakan standar algoritma, protokol yang mutakhir dan kuat, serta kunci berada pada tempatnya; menggunakan manajemen kunci yang tepat.
- Enkripsi semua data dalam perjalanan dengan protokol aman seperti TLS dengan cipher perfect forward secrecy (PFS), prioritas cipher oleh server, dan parameter yang aman. Terapkan enkripsi menggunakan arahan seperti HTTP Strict Transport Security (HSTS).
- Menonaktifkan caching untuk respons yang berisi data sensitif.
- Simpan kata sandi menggunakan fungsi hashing adaptif dan salted yang kuat dengan faktor kerja (faktor penundaan), seperti Argon2, scrypt, bcrypt, atau PBKDF2.

- Verifikasi secara independen efektivitas konfigurasi dan pengaturan.

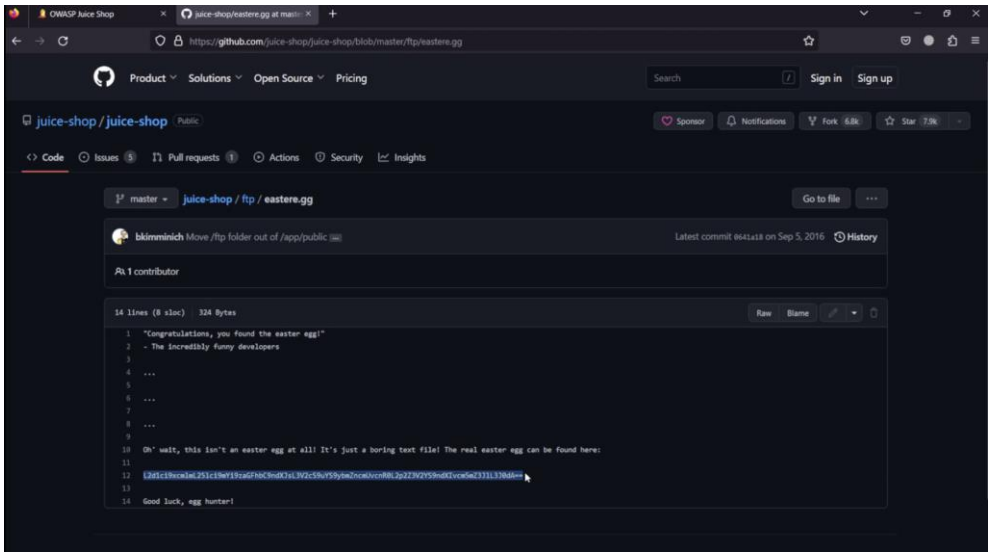
Percobaan 1

Pada percobaan ini akan mencari pesan tersembunyi yang telah disisipkan kedalam website juice shop.

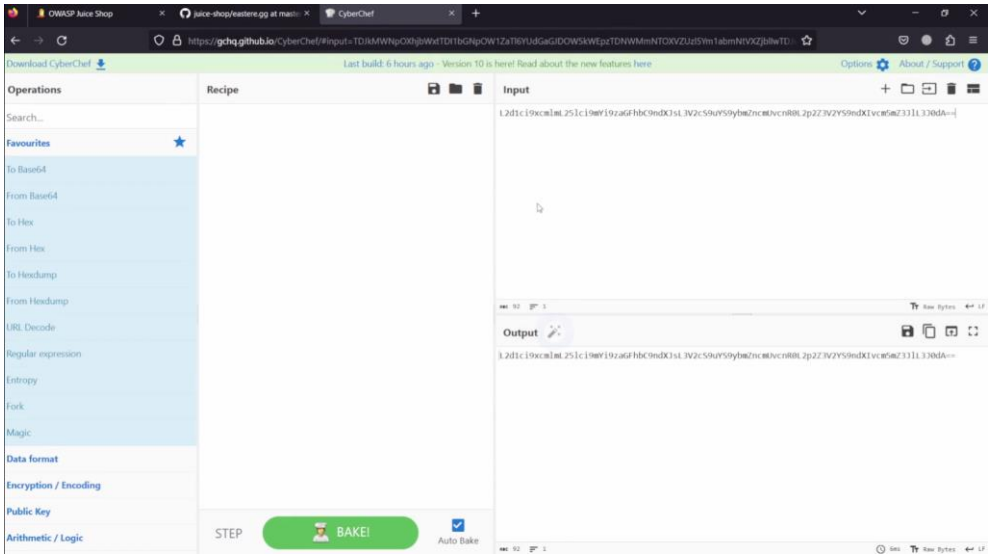
1. Menjalankan aplikasi juiceshop kemudian buka sidebar dan pilih Github.



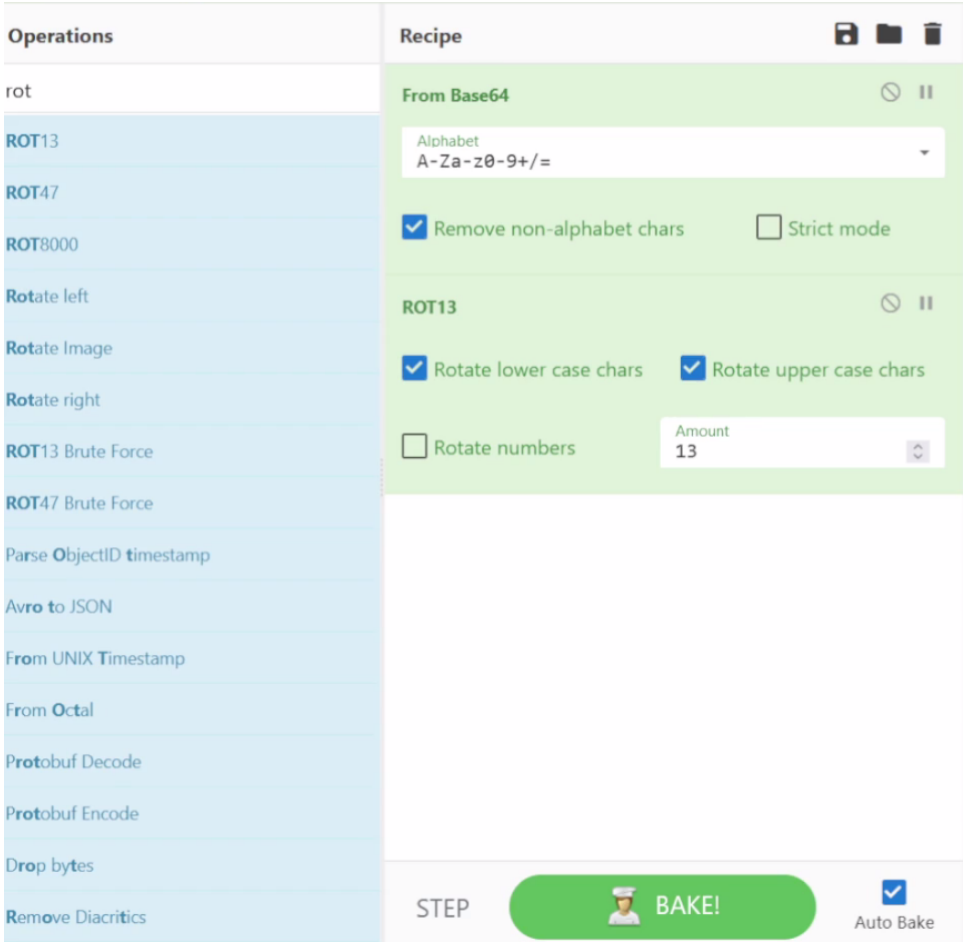
2. Masuk ke folder ftp dan klik file eastere.gg, kemudian copy text.



3. Membuka Cyber Chef dan paste ke box Input.

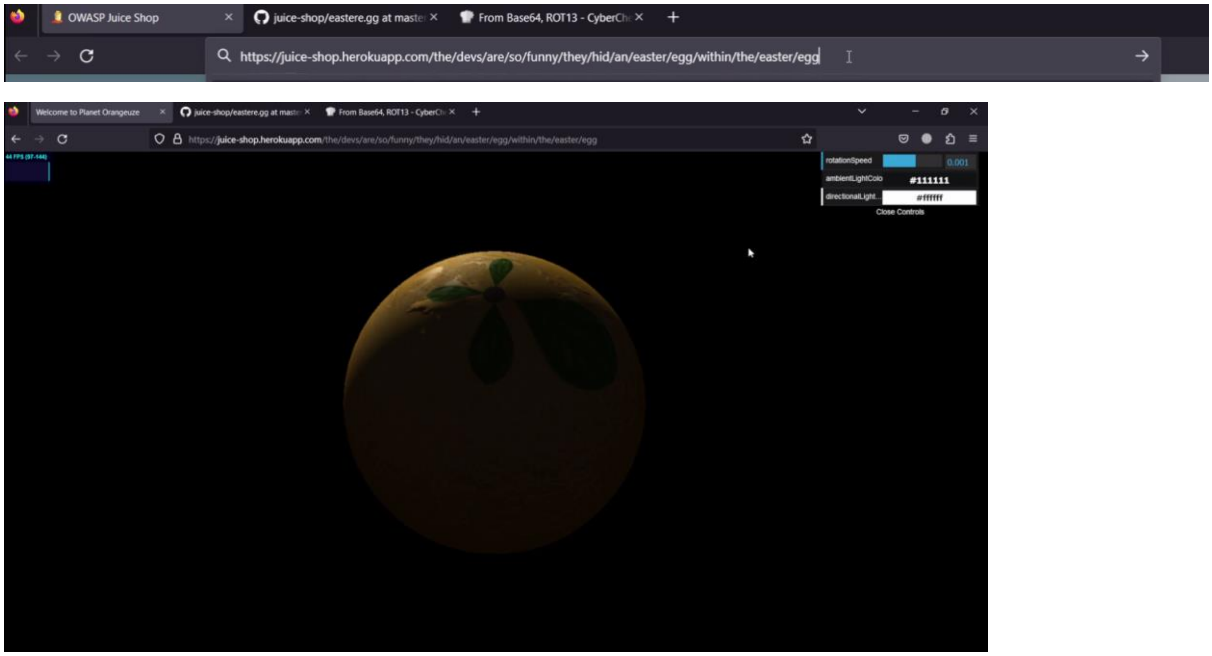


4. Drag “From Base64” dan ROT13 ke dalam Recipe.



5. Akan muncul output, kemudiac copy output tersebut.

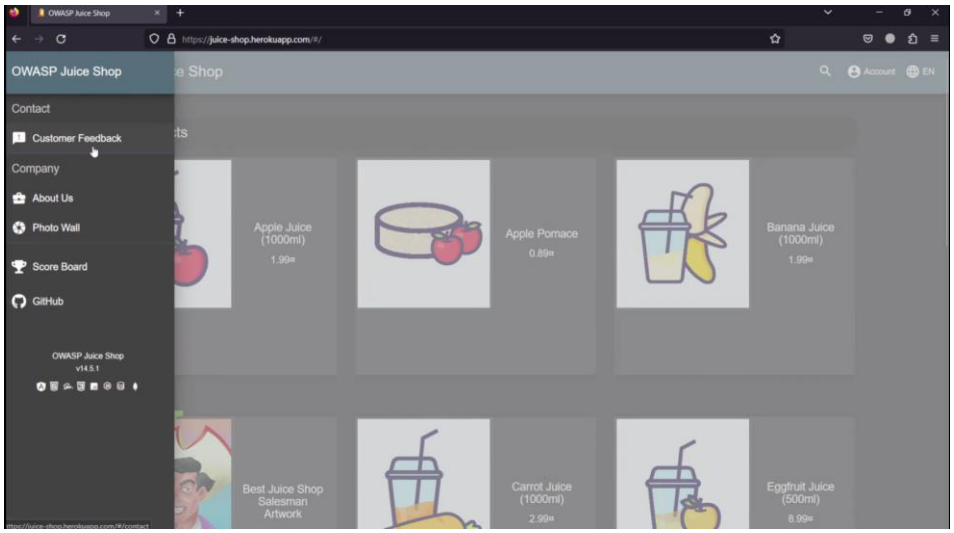
6. Tambahkan ke link juice shop dan akan muncul halaman tersembunyi.



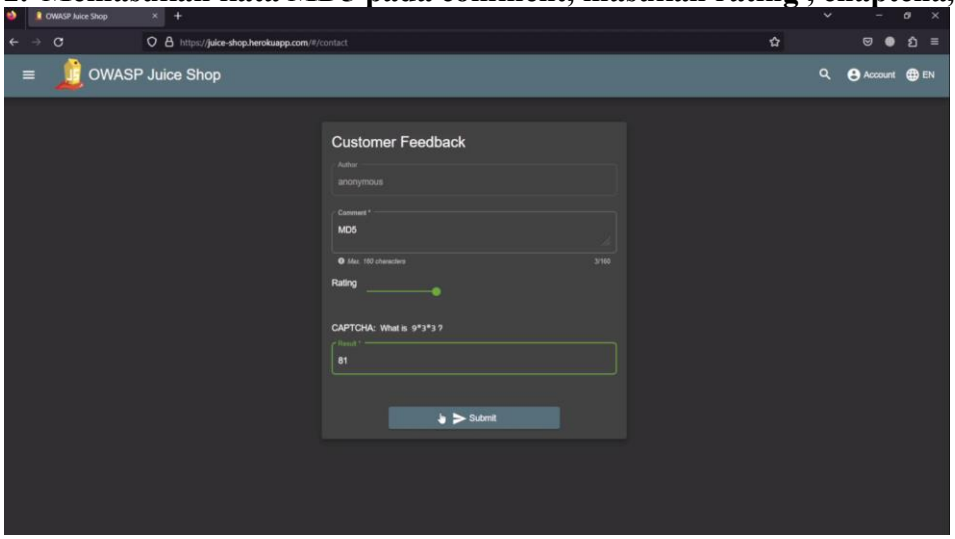
Percobaan 2

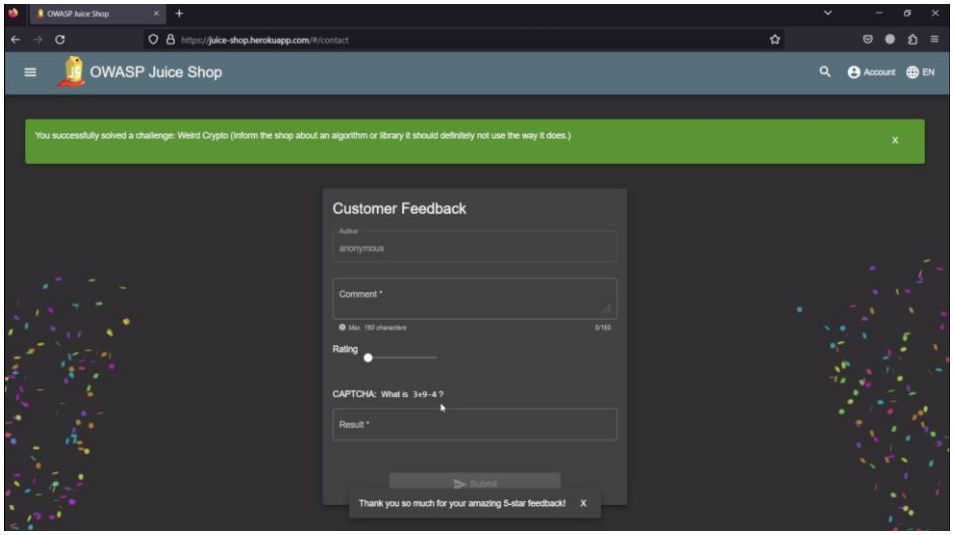
Pada percobaan ini akan mencari permasalahan kriptografi.

1. **Menjalankan aplikasi juiceshop kemudian buka sidebar dan pilih Customer Feedback.**



2. **Memasukan kata MD5 pada comment, masukan rating , chaptcha, dan submit.**





Penjelasan: MD5 merupakan salah satu algoritma yang lemah namun sering digunakan untuk melakukan kriptografi atau melakukan enkripsi pada data-data krusial, yang seharusnya memiliki privasi dan keamanan lebih. MD5 ini merupakan Collision Vulnerability, dikarenakan berapapun panjang dari sebuah text, maka tetap akan dirubah menjadi 128 bit saja. yang mana dalam skala penyimpanan data yang sangat besar akan ada kemungkinan 2 file yang berbeda akan memiliki nilai hash yang sama

Video Demo

https://drive.google.com/file/d/1jCjITtrHNVjEqTa3ARFg8pbmY8mE-WDP/view?usp=sharing