UNIVERSITAS
MUHAMMADIYAH
SIDOARJO

UMSIDA
DARI SINI PENCERAHAN BERSEMI

Rancang Bangun Password Manager Dengan Enkripsi Multi-Layer Berbasis Rust dan Typescript

Oleh:

Moch Jimmy Marchel,

Hamzah Setiawan

Informatika

Universitas Muhammadiyah Sidoarjo

Agustus, Tahun

Pendahuluan

Sejarah mencatat beberapa insiden kebocoran data berskala besar yang menjadi pengingat penting akan hal ini. Salah satu yang paling banyak dikutip dalam literatur keamanan adalah kasus RockYou pada tahun 2009, di mana sekitar 32 juta akun terekspos, dan kata sandi "123456" ditemukan digunakan oleh sekitar 290 ribu akun (±1% dari total pengguna). Insiden semacam ini menjadi salah satu motivasi awal berkembangnya penelitian tentang pengelolaan kata sandi yang lebih aman.

Di tengah pesatnya adopsi layanan digital mulai dari e-commerce, perbankan, hingga media sosial autentikasi berbasis kata sandi teks masih menjadi metode yang paling umum digunakan. Kenyamanan ini membawa konsekuensi: pengguna cenderung memakai kata sandi yang sama di berbagai platform agar mudah diingat, sehingga satu kebocoran data berpotensi berdampak berantai ke banyak akun sekaligus.

Tantangan ini diperberat oleh dilema klasik dalam keamanan kata sandi: kata sandi yang kuat dan unik untuk setiap layanan sulit diingat, sementara kata sandi yang mudah diingat cenderung lebih rentan terhadap serangan brute-force dan pencurian data.

Pertanyaan Penelitian (Rumusan Masalah)

- Aplikasi pengelola kata sandi telah banyak membantu pengguna mengelola kredensialnya dan terus berkembang seiring waktu. Penelitian di bidang keamanan informasi menunjukkan bahwa efektivitas sebuah pengelola kata sandi sangat bergantung pada detail implementasi arsitektur keamanannya terutama dalam menangani data sensitif di memori volatile dan memastikan proteksi sisi server yang optimal.

- Sebagian implementasi yang ada memanfaatkan algoritma seperti SHA-256 dan AES-GCM, yang secara umum masih dianggap aman untuk banyak kasus penggunaan. Untuk kebutuhan spesifik seperti derivasi kunci dari kata sandi pengguna, algoritma yang lebih baru seperti Argon2id menawarkan ketahanan tambahan terhadap serangan brute-force berbasis GPU dan side-channel, sehingga membuka peluang untuk memperkuat lapisan keamanan yang sudah ada.

- Penelitian ini melihat adanya ruang untuk mengeksplorasi arsitektur Zero-Knowledge yang dieksekusi sepenuhnya di sisi klien, memanfaatkan kemajuan teknologi seperti WebAssembly dan bahasa pemrograman yang memory-safe seperti Rust.

Tujuan Penelitian

- Merancang dan mengembangkan aplikasi password manager berbasis web yang mengimplementasikan prinsip arsitektur Zero-Knowledge, sehingga server tidak pernah memiliki akses terhadap kata sandi utama maupun kunci enkripsi pengguna.
- Mengimplementasikan skema enkripsi multi-lapisan yang memadukan Argon2id untuk key derivation dan XChaCha20-Poly1305 untuk enkripsi data simetris.
- Memindahkan seluruh beban komputasi kriptografi ke sisi klien melalui modul Rust yang dikompilasi ke WebAssembly (WASM).
- Menguji fungsionalitas sistem melalui metode Black Box Testing, guna memastikan seluruh fitur utama berjalan sesuai spesifikasi dan standar keamanan yang diharapkan.

Fondasi Kriptografi

Master Password

- Satu kredensial utama sekaligus basis entropi untuk derivasi seluruh kunci enkripsi dalam sistem. Kualitasnya menentukan keamanan seluruh brankas.

Salt

- Nilai acak unik per akun yang ditambahkan sebelum hashing, mencegah serangan rainbow table, serta menghalangi dekripsi massal paralel di server.

Nonce (Number Used Once)

- Nilai acak sekali-pakai pada setiap operasi enkripsi, memastikan ciphertext selalu berbeda meski data dan kunci yang digunakan sama.

Tujuan Penelitian

Argon2id

- Pemenang Password Hashing Competition 2015, didefinisikan dalam RFC 9106. Fungsi memory-hard yang tahan terhadap brute-force GPU/ASIC maupun serangan side-channel.

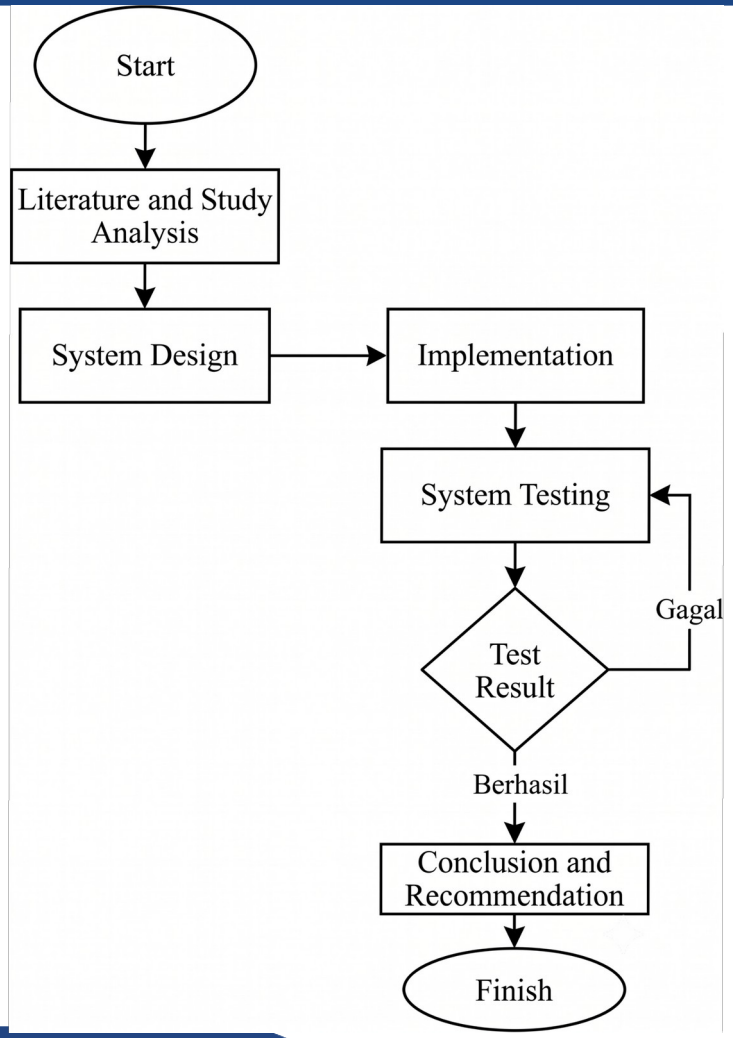
XChaCha20-Poly1305

- Skema AEAD dengan nonce 192-bit yang menekan risiko tabrakan. Berjalan cepat pada CPU umum tanpa akselerasi khusus, dan tahan cache-timing.

DEK (Data Encryption Key)

- Kunci simetris acak 256-bit yang mengenkripsi data vault secara langsung. Hanya berada di memori lokal pengguna selama sesi aktif.

Metode



Identifikasi Masalah & Studi Literatur

- Memetakan kerentanan pengelola kata sandi konvensional dan mengkaji solusi kriptografi modern.

Perancangan Sistem

- Merumuskan arsitektur Zero-Knowledge dan skema enkripsi multi-lapisan.

Implementasi

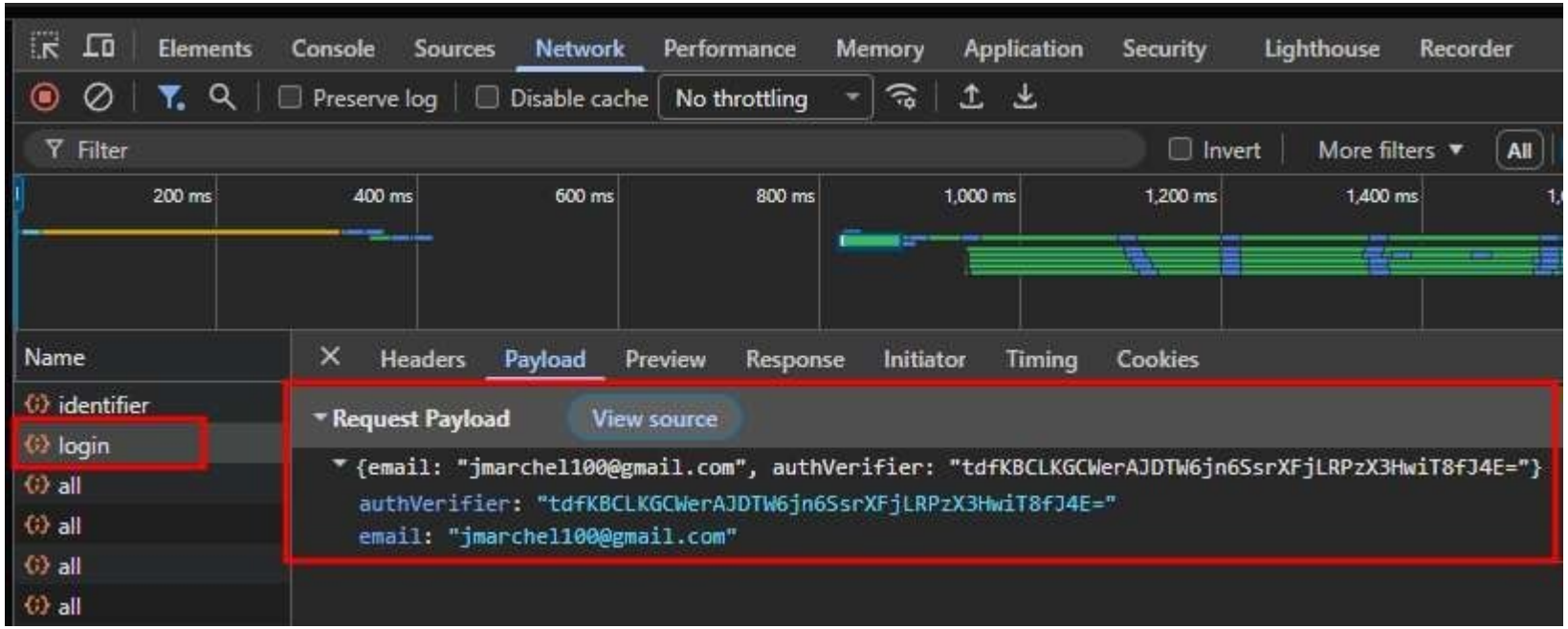
- Menerjemahkan desain ke logic backend (Rust WASM) dan frontend type-safe.

Pengujian Sistem

- Black Box Testing pada fitur utama; jika gagal, proses kembali ke tahap implementasi.

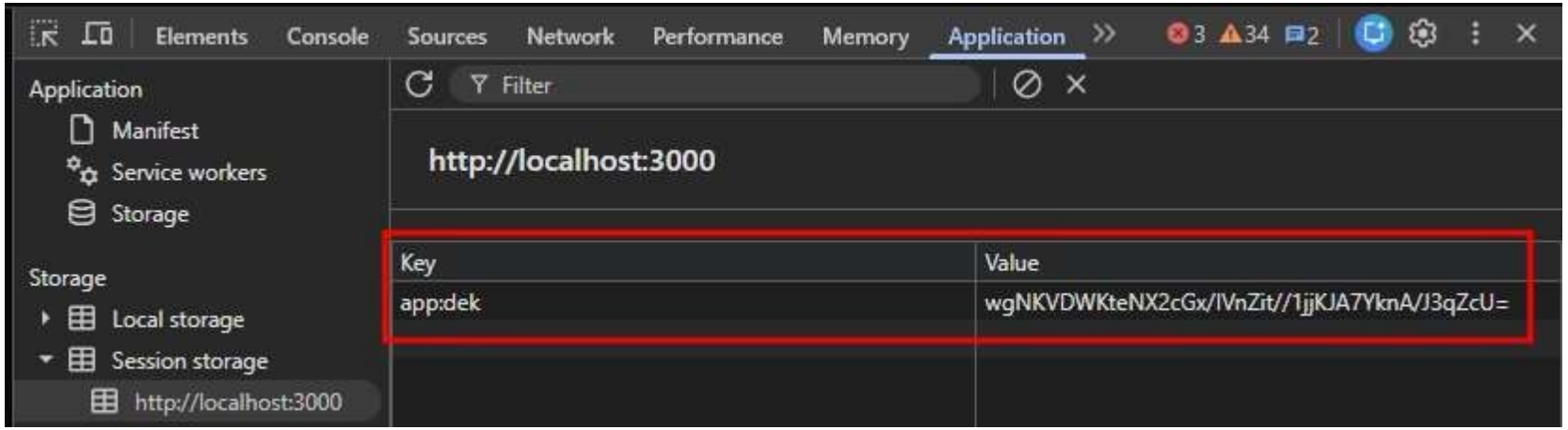
Hasil

Bukti Teknis: Payload Login dan Penyimpanan DEK



Gambar 1. Request payload saat login hanya berisi auth_verifier — bukan master password asli.

Hasil



Gambar 2. DEK plaintext hanya tersimpan sementara di sessionStorage peramban selama sesi aktif.

Hasil

```
secure_vault=# SELECT * FROM users;
-[ RECORD 1 ]-----+------------------------------------------------------------
id                | 717c2c1f-bcbe-434a-86b8-5ae07393b553
username          | Jimmy
email             | jmarchel100@gmail.com
encrypted_dek     | n5/sFV7QNBXz2siX/SzR0ocNRM5vb3jIcrJgSCLHQJgOuYaV1KaXHnWu98FSAw0e
salt              | aHF0WDhaSi96R1ZubFJ1NlNMRVhFUQ==
argon2_params     | algo=argon2id,v=19,m=65536,t=3,p=1
is_email_verified | t
created_at        | 2026-06-25 04:52:03.967882
nonce             | WirxeefeRqJ9XUjZMwyAtZHQkzIU6reu
auth_verifier     | tdfKBCLKGCWerAJDTW6jn6SsrXFjLRPzX3HwiT8fJ4E=

secure_vault=# SELECT * FROM vaults;
-[ RECORD 1 ]--+-------------------------------------------------------------------------------------------------------------------------------
---------------------------------------------------------
id             | 470d4210-1052-42f1-99ba-dc4dbb01eb2a
user_id        | 717c2c1f-bcbe-434a-86b8-5ae07393b553
encrypted_data | xqquXpBvVPkksLFN5WJrpSnSoixIEWJwwHbQK/LHs+xZ6rWBcjKHQ6oixdL9pUWEKyO0olsZLMMc9Jz4IZmGjPLuPQ0zqV40/X16b
vZDIoWUnvZdrWeXjJ4FChwo/FK1DRXqLCBhIdnqCSiiP+t1MNabGyroiQ==
nonce          | ovjGUL5eV+bhWasJYyJqjypAMlWov/Gk
created_at     | 2026-06-25 04:53:07.741144
updated_at     | 2026-06-25 04:53:07.741144
title          | Facebook
item_type      | Password
```

Gambar 3. Kolom encrypted_dek, auth_verifier, dan isi vault di PostgreSQL murni berupa ciphertext acak.

Referensi

Berdasarkan hasil perancangan dan pengujian yang telah dilakukan, penelitian ini berhasil mengimplementasikan aplikasi pengelola kata sandi berbasis Zero-Knowledge Architecture menggunakan modul Rust WebAssembly (WASM), yang memastikan seluruh komputasi kriptografi dieksekusi secara penuh di sisi klien.

- Integrasi Argon2id (tahan brute-force GPU) dan XChaCha20-Poly1305 (enkripsi simetris) terbukti memberi perlindungan berlapis bagi data kredensial pengguna.
- Server pada arsitektur ini murni menerima dan menyimpan ciphertext, nonce, dan salt, tanpa pernah memiliki akses terhadap kunci enkripsi maupun kata sandi utama pengguna.
- Pembuktian teknis melalui inspeksi jaringan, basis data, dan sessionStorage mengonfirmasi bahwa master password asli tidak pernah terekspos ke server.
- Manajemen DEK yang volatile turut berkontribusi memitigasi risiko eksploitasi memori.
- Seluruh fitur utama — registrasi, login, enkripsi, dan dekripsi — telah melalui Black Box Testing dan berjalan sesuai spesifikasi, menjadikan sistem ini salah satu upaya menghadirkan solusi manajemen kredensial yang aman dan teruji.


UMSIDA
DARI SINI PENCERAHAN BERSEMI