Laporan Report Pentest Security Lab

Dosen pengampu : Sopian Soim S.T.,M.T.



Disusun oleh :

Andrian Zola 062140352387

Faris alqhaniyyu 062140352393

**PROGRAM STUDI TEKNIK TELEKOMUNIKASI**

**JURUSAN TEKNIK ELEKTRO**

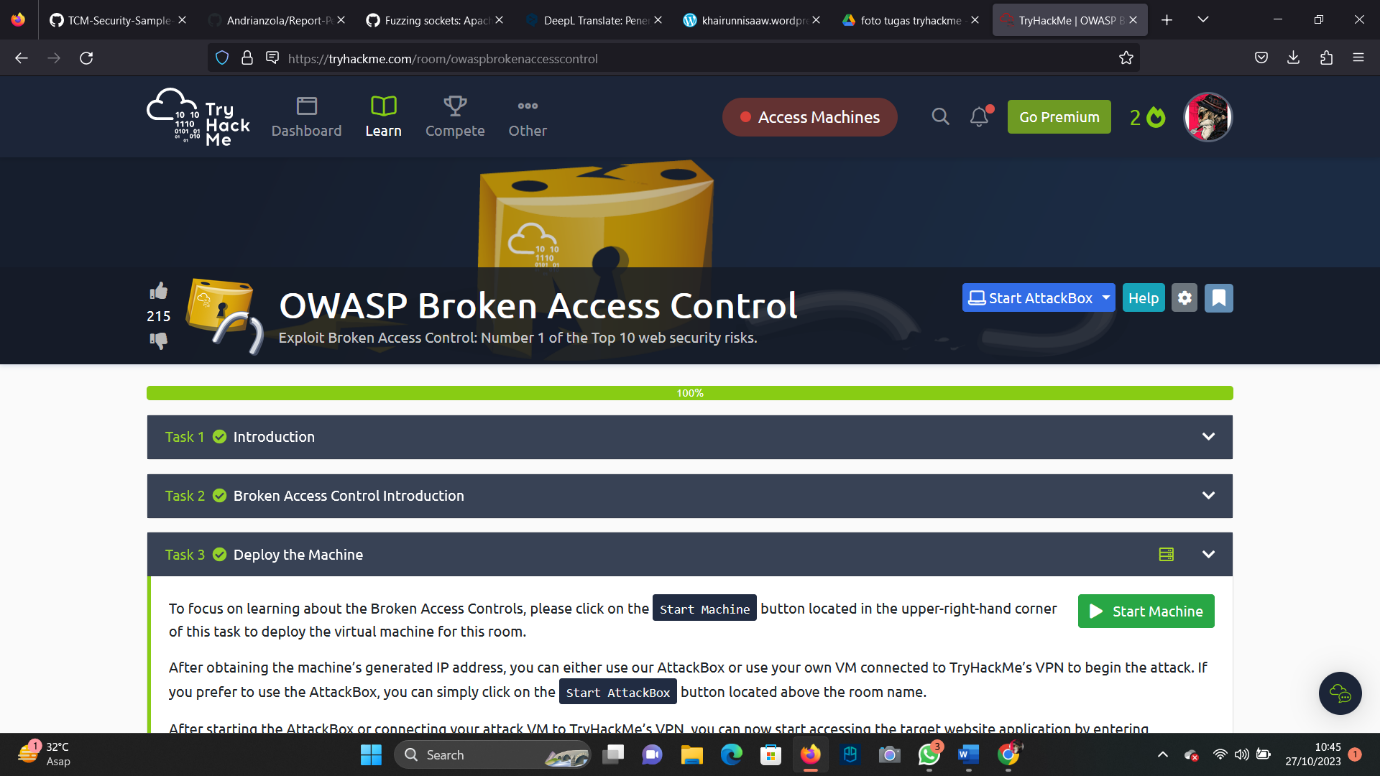
**POLITEKNIK NEGERI SRIWIJAYA**

**2023**

KATA PENGANTAR

Puji syukur diucapkan kehadirat Allah Swt. atas segala rahmat-Nya sehingga makalah ini dapat tersusun sampai selesai. Tidak lupa kami mengucapkan terima kasih terhadap bantuan dari pihak yang telah berkontribusi dengan memberikan sumbangan baik pikiran maupun materi.  
     Penulis sangat berharap semoga makalah ini dapat menambah pengetahuan dan pengalaman bagi pembaca. Bahkan kami berharap lebih jauh lagi agar makalah ini bisa pembaca praktikkan dalam kehidupan sehari-hari.  
     Bagi kami sebagai penyusun merasa bahwa masih banyak kekurangan dalam penyusunan makalah ini karena keterbatasan pengetahuan dan pengalaman kami. Untuk itu kami sangat mengharapkan kritik dan saran yang membangun dari pembaca demi kesempurnaan makalah ini.

1. **Executive Summary**



1. **OWASP Broken Access Control**

( Exploit Broken Access Control: Number 1 of the Top 10 web security risks )

Kontrol akses yang rusak adalah jenis kerentanan keamanan yang muncul ketika sebuah aplikasi atau sistem gagal membatasi akses ke data atau fungsionalitas yang sensitif dengan benar. Kerentanan ini memungkinkan penyerang mendapatkan akses tidak sah ke sumber daya yang seharusnya dibatasi, seperti akun pengguna, file, basis data, atau fungsi administratif. Kontrol akses yang rusak dapat terjadi karena berbagai faktor, termasuk desain yang buruk, kesalahan konfigurasi, atau kesalahan pengkodean.

Kontrol akses bisa diimplementasikan dengan berbagai cara, tergantung pada jenis sumber daya yang dilindungi dan persyaratan keamanan sistem. Beberapa mekanisme kontrol akses yang umum meliputi:

1. Kontrol Akses Diskresioner (DAC)

Pada jenis kontrol akses ini, pemilik sumber daya atau administrator menentukan siapa yang diizinkan mengakses sumber daya dan tindakan apa yang boleh dilakukan. DAC umumnya digunakan dalam sistem operasi dan sistem file. Dalam istilah awam, bayangkan sebuah kastil di mana raja dapat memberikan kunci kepada para penasihatnya, yang memungkinkan mereka untuk membuka pintu apa pun yang mereka suka, kapan pun mereka mau. Itulah DAC untuk Anda. Ini adalah kebebasan untuk mengontrol akses ke sumber daya Anda sendiri. Orang yang bertanggung jawab, seperti raja kastil, dapat memberikan izin kepada siapa pun yang mereka inginkan, mendikte siapa yang bisa masuk dan keluar.

1. Mandatory Access Control (MAC)

Pada jenis kontrol akses ini, akses ke sumber daya ditentukan oleh seperangkat aturan atau kebijakan yang telah ditetapkan sebelumnya yang ditegakkan oleh sistem. MAC biasanya digunakan di lingkungan yang sangat aman, seperti sistem pemerintah dan militer. Dalam istilah awam, bayangkan sebuah benteng dengan protokol keamanan berlapis besi. Hanya individu tertentu dengan izin keamanan tertentu yang dapat mengakses area tertentu, dan ini tidak dapat dinegosiasikan. Komandan tertinggi menetapkan aturan, dan aturan tersebut dipatuhi dengan ketat. Begitulah cara kerja MAC. Ini seperti petugas keamanan yang tegas yang tidak mengizinkan pengecualian terhadap aturan.

1. Kontrol Akses Berbasis Peran (RBAC)

Dalam jenis kontrol akses ini, pengguna diberi peran yang menentukan tingkat akses mereka ke sumber daya. RBAC biasanya digunakan dalam sistem perusahaan, di mana pengguna memiliki tingkat otoritas yang berbeda berdasarkan tanggung jawab pekerjaan mereka. Dalam istilah awam, bayangkan sebuah perusahaan modern. Anda memiliki manajer, eksekutif, staf penjualan, dan lain-lain. Mereka masing-masing memiliki akses yang berbeda ke dalam gedung. Beberapa bisa masuk ke ruang rapat, yang lain bisa mengakses lantai penjualan, dan seterusnya. Itulah inti dari RBAC - memberikan akses berdasarkan peran seseorang dalam organisasi.

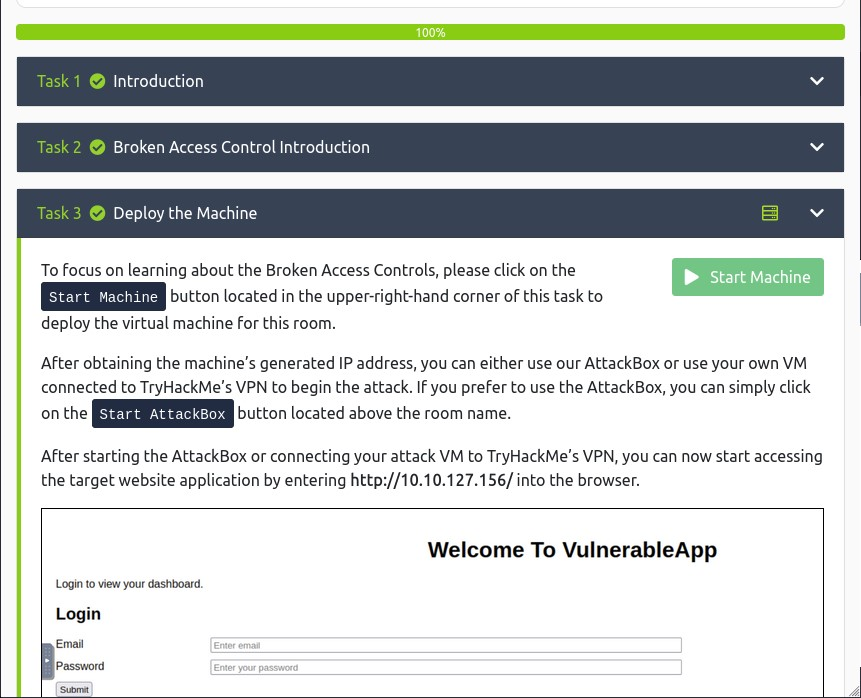
1. Kontrol Akses Berbasis Atribut (ABAC)

Pada jenis kontrol akses ini, akses ke sumber daya ditentukan oleh sekumpulan atribut, seperti peran pengguna, waktu, lokasi, dan perangkat. ABAC umumnya digunakan di lingkungan cloud dan aplikasi web. Dalam istilah awam, bayangkan sebuah sistem keamanan fiksi ilmiah yang sangat canggih yang memindai individu untuk mengetahui atribut tertentu. Mungkin sistem ini memeriksa apakah mereka berasal dari planet tertentu, apakah mereka membawa perangkat tertentu, atau apakah mereka mencoba mengakses sumber daya pada waktu tertentu. Itulah ABAC. Ini seperti keamanan masa depan yang cerdas dan fleksibel.

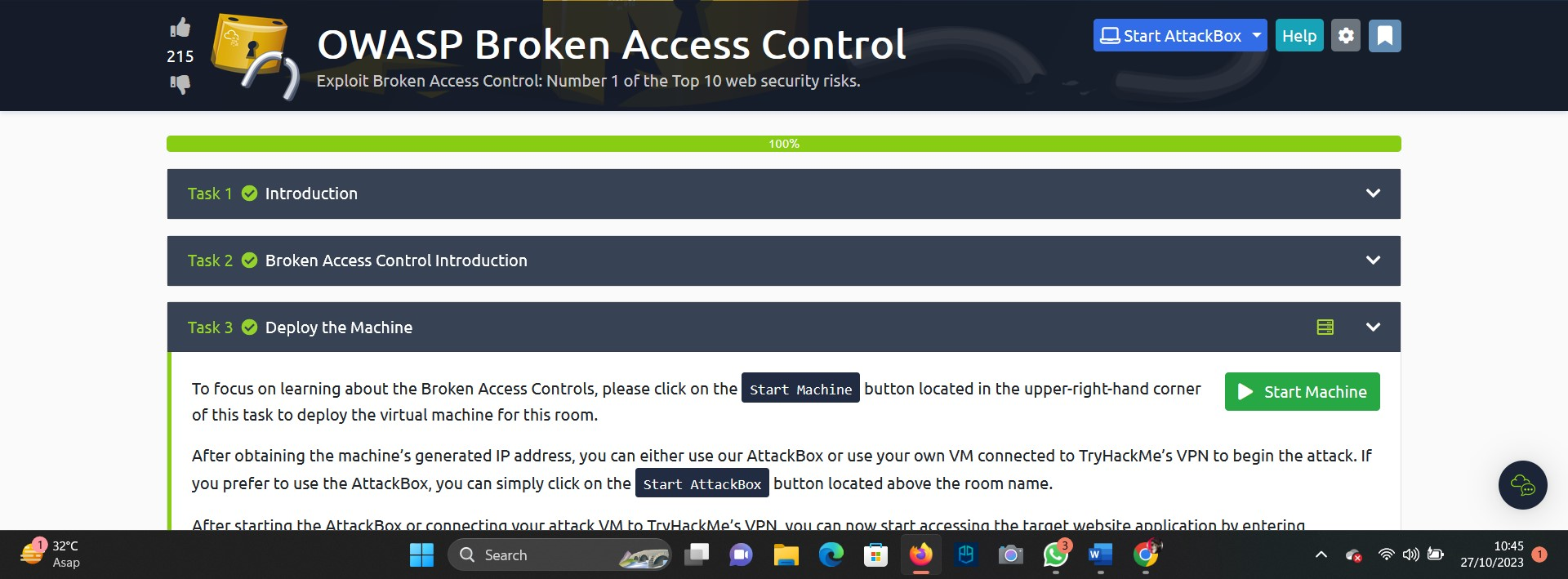
Kerentanan kontrol akses yang rusak mengacu pada situasi di mana mekanisme kontrol akses gagal menegakkan pembatasan yang tepat pada akses pengguna ke sumber daya atau data. Berikut ini beberapa eksploitasi umum untuk kontrol akses yang rusak dan contohnya:

1. **Eskalasi hak istimewa horizontal** terjadi ketika penyerang dapat mengakses sumber daya atau data milik pengguna lain dengan tingkat akses yang sama. Sebagai contoh, seorang pengguna mungkin dapat mengakses akun pengguna lain dengan mengubah ID pengguna di URL.
2. **Eskalasi hak istimewa vertikal** terjadi ketika penyerang dapat mengakses sumber daya atau data milik pengguna dengan tingkat akses yang lebih tinggi. Sebagai contoh, pengguna biasa dapat mengakses fungsi administratif dengan memanipulasi bidang formulir tersembunyi atau parameter URL.
3. **Pemeriksaan kontrol akses** yang tidak memadai terjadi ketika pemeriksaan kontrol akses tidak dilakukan dengan benar atau konsisten, sehingga memungkinkan penyerang untuk melewatinya. Sebagai contoh, sebuah aplikasi dapat mengizinkan pengguna untuk melihat data sensitif tanpa memverifikasi izin yang tepat.
4. **Referensi objek langsung** yang tidak aman terjadi ketika penyerang dapat mengakses sumber daya atau data dengan mengeksploitasi kelemahan dalam mekanisme kontrol akses aplikasi. Sebagai contoh, sebuah aplikasi mungkin menggunakan pengidentifikasi yang dapat diprediksi atau mudah ditebak untuk data sensitif, sehingga memudahkan penyerang untuk mengakses.
5. Langkah Yang Dilakukan Untuk pengujian
6. Menjalankan Mesin

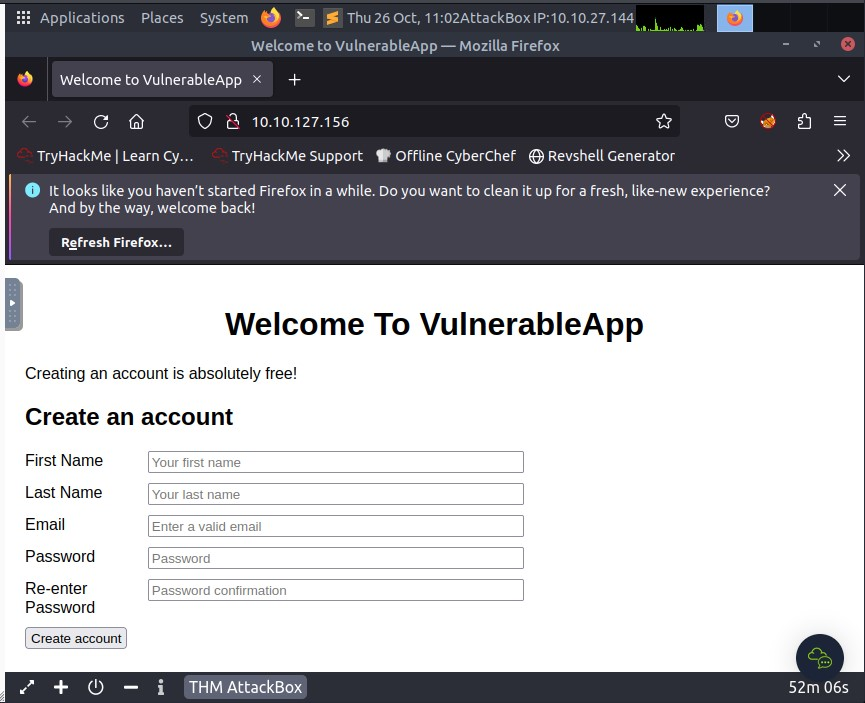
Untuk fokus mempelajari tentang Kontrol Akses yang Rusak, silakan klik pada tombol Mulai Mesin yang terletak di sudut kanan atas tugas ini untuk menggunakan mesin virtual untuk ruangan ini.



1. Setelah mendapatkan alamat IP mesin yang dihasilkan, Anda dapat menggunakan AttackBox kami atau menggunakan VM Anda sendiri yang terhubung ke VPN TryHackMe untuk memulai serangan. Jika Anda lebih suka menggunakan AttackBox, Anda cukup mengklik tombol Start AttackBox yang terletak di atas nama ruangan.



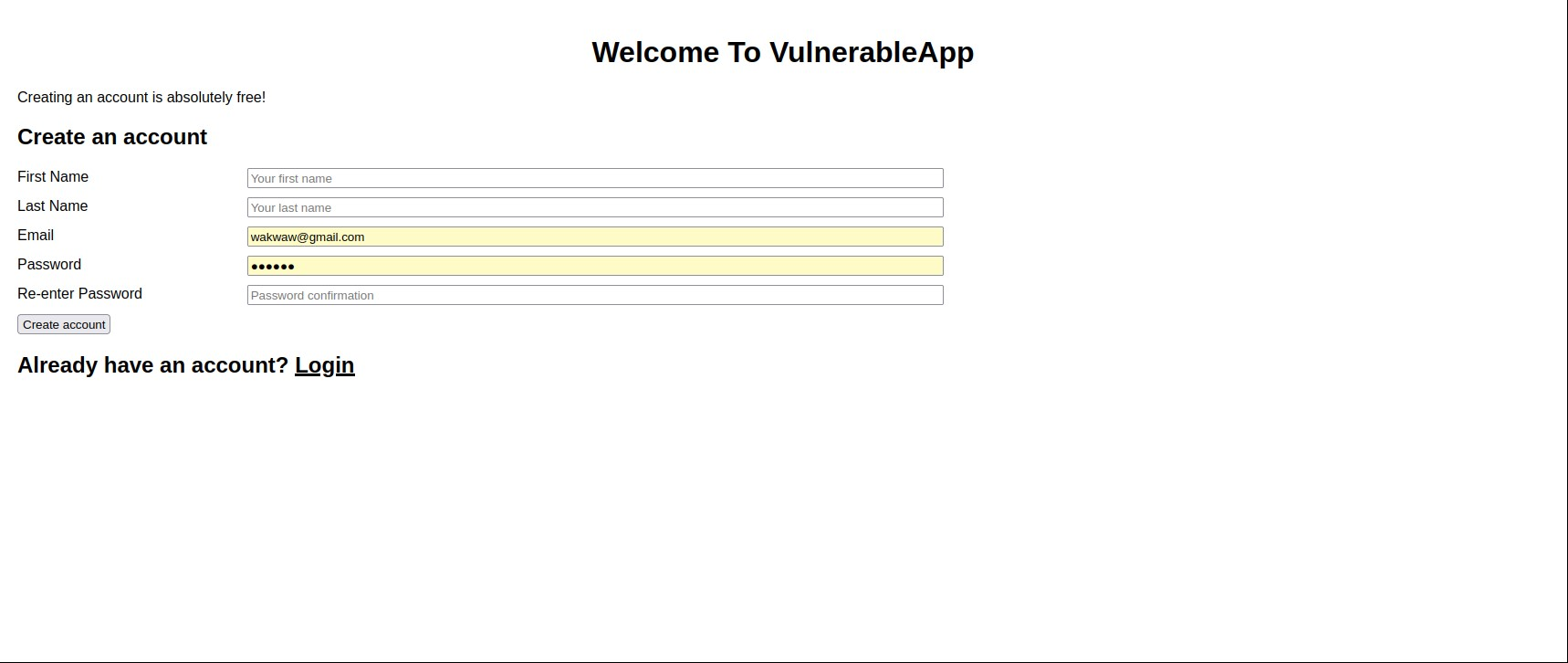
1. Setelah memulai AttackBox atau menghubungkan VM serangan Anda ke VPN TryHackMe, Anda sekarang dapat mulai mengakses aplikasi situs web target dengan memasukkan <http://10.10.127.156/> ke dalam browser.

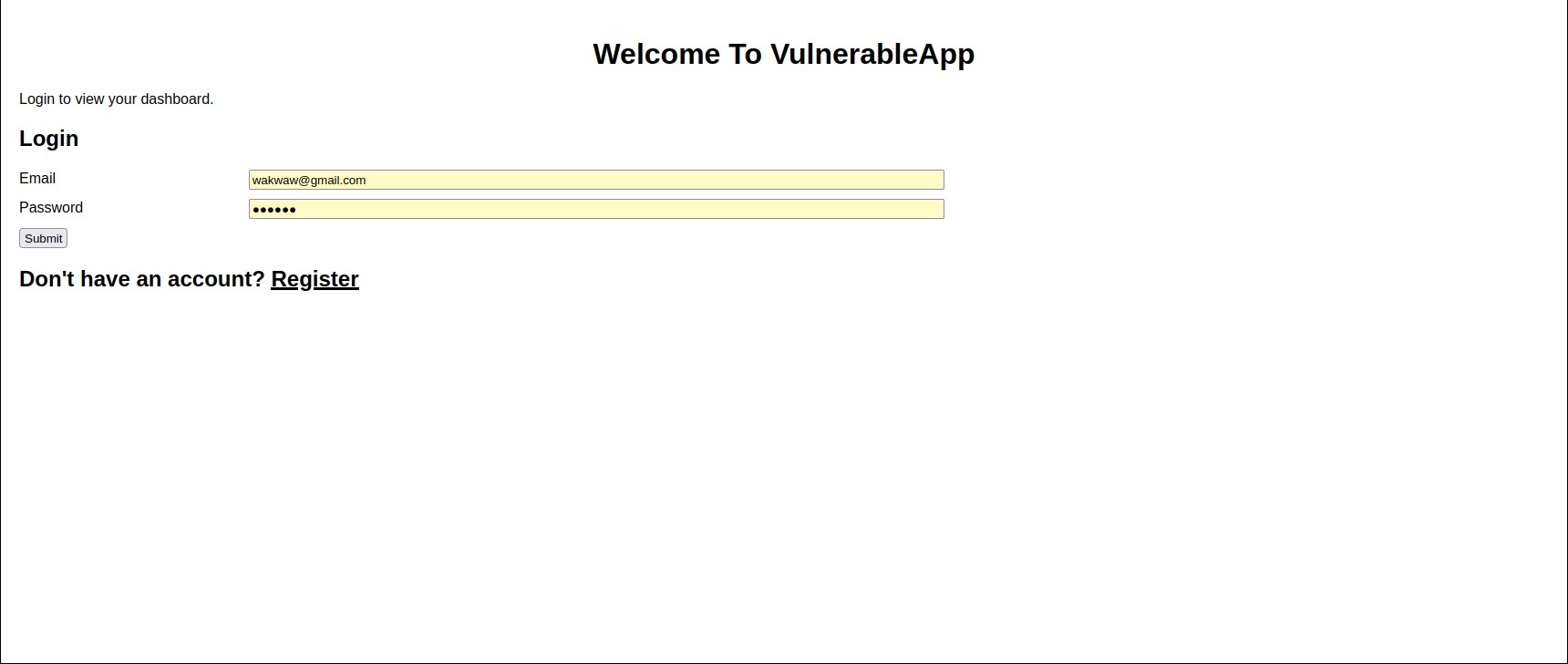


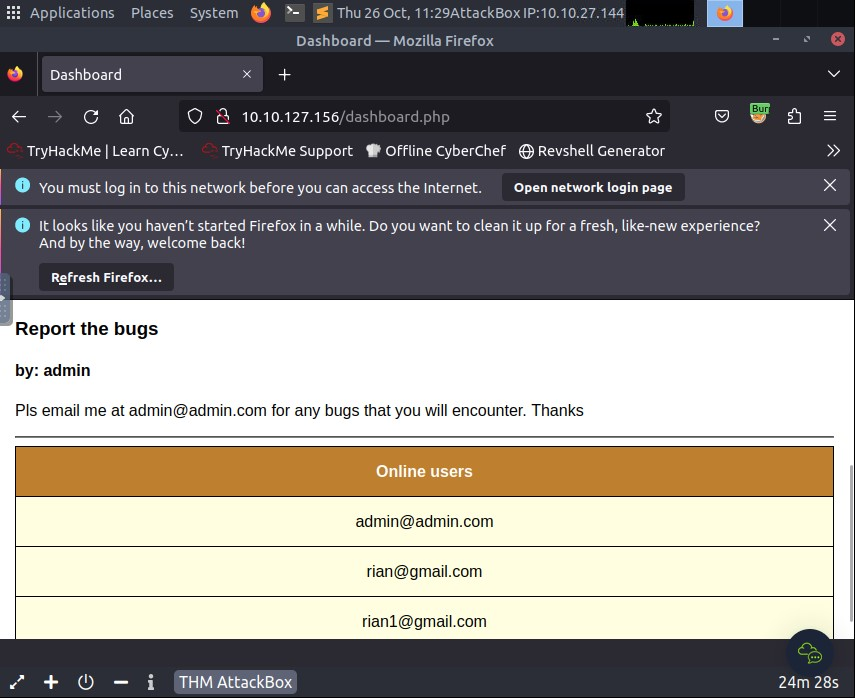
1. Menilai Aplikasi Web

Aplikasi web untuk ruangan ini memiliki Dashboard, Login, dan formulir Registrasi yang memungkinkan pengguna mengakses dashboard situs web. Dari sudut pandang seorang pengujian penetrasi aplikasi web, seorang pengujian penetrasi biasanya akan mendaftar akun. Setelah pendaftaran, pengujian penetrasi kemudian akan mencoba memeriksa fungsi login untuk menemukan kerentanan kontrol akses.

Berikut contoh penampilan setiap halaman :



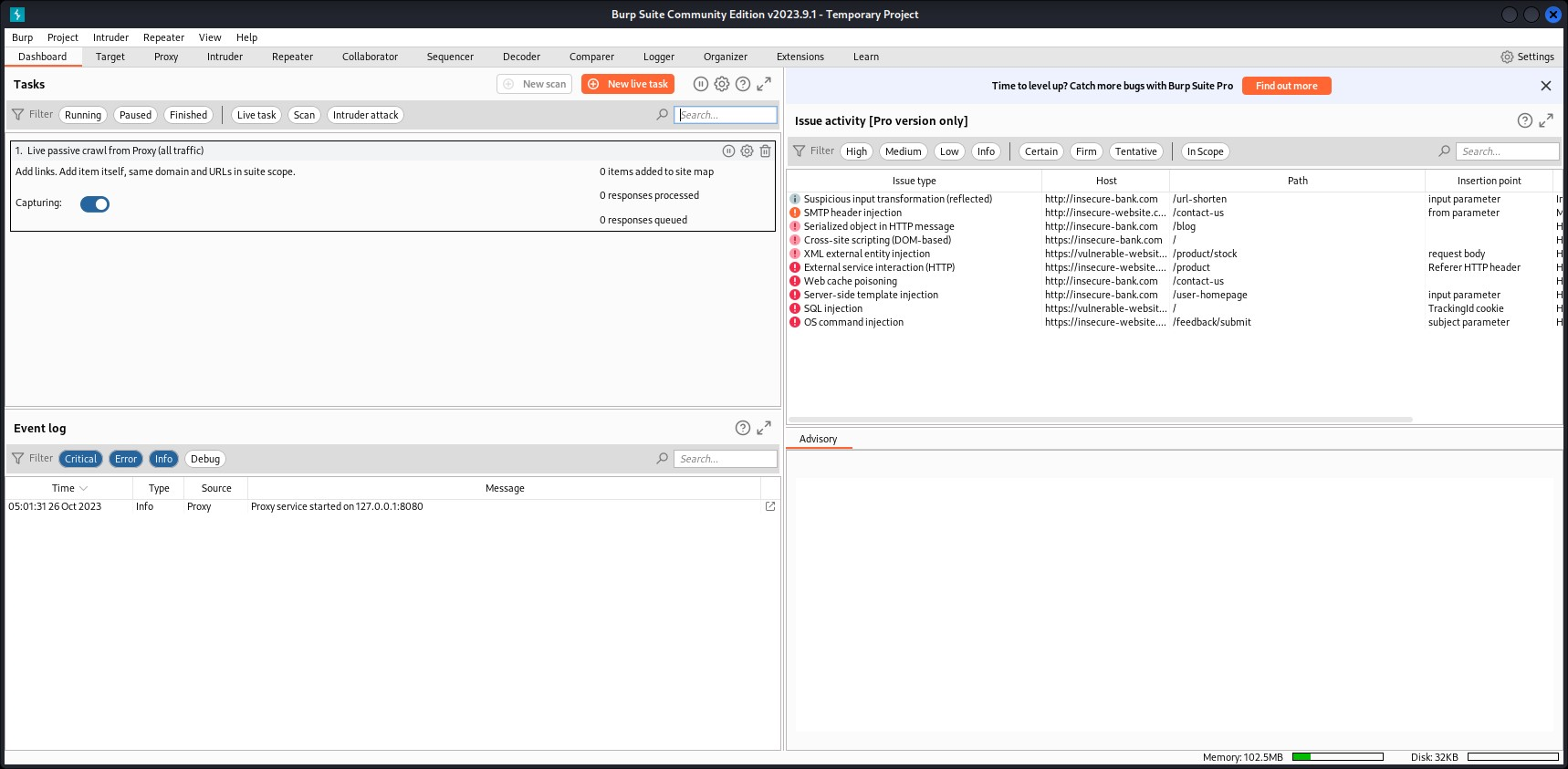


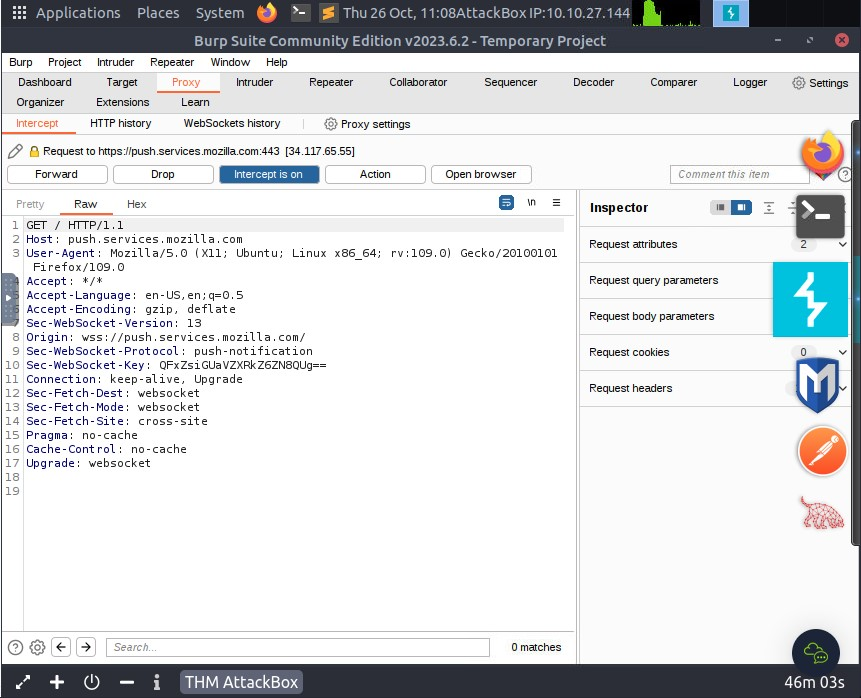


1. Menangkap lalu lintas HTTP

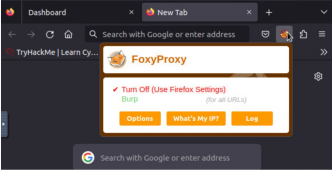
Agar kita dapat menganalisis lebih lanjut permintaan dan respons yang dikirim dan diterima dari server, kita akan menggunakan modul "Proxy" pada Burp Suite untuk menangkap lalu lintas HTTP yang dikirim ke server. Lalu lintas HTTP yang ditangkap dapat digunakan dengan modul-modul lain dari Burp Suite.

Gambar 1. Halaman pertama Burp Suite

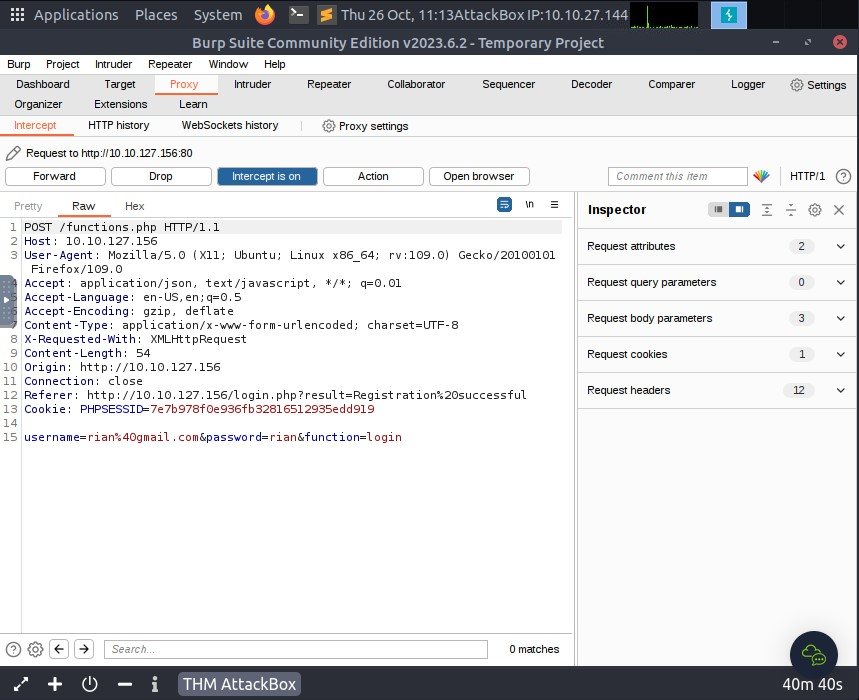




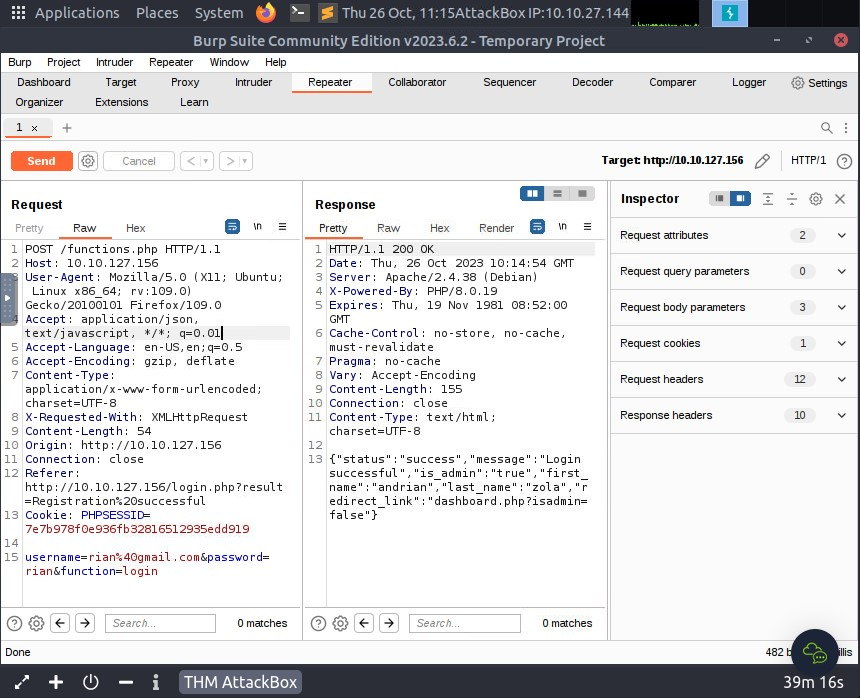
1. Untuk menyambungkan dengan **Burp Suite** browser yang harus dilakukan adalah mengubah foxy proxy menjadi **Burp.**



1. Kemudian dapat dimanipulasi dengan cara menekan **Drop**, lalu dikirimkan ke alat lain yaitu **Repeater** untuk diproses lebih lanjut.



1. Mengirimkan hasil **Proxy** ke **Repeater** dan tekan “ send “ untuk memberikan “ Response ”.

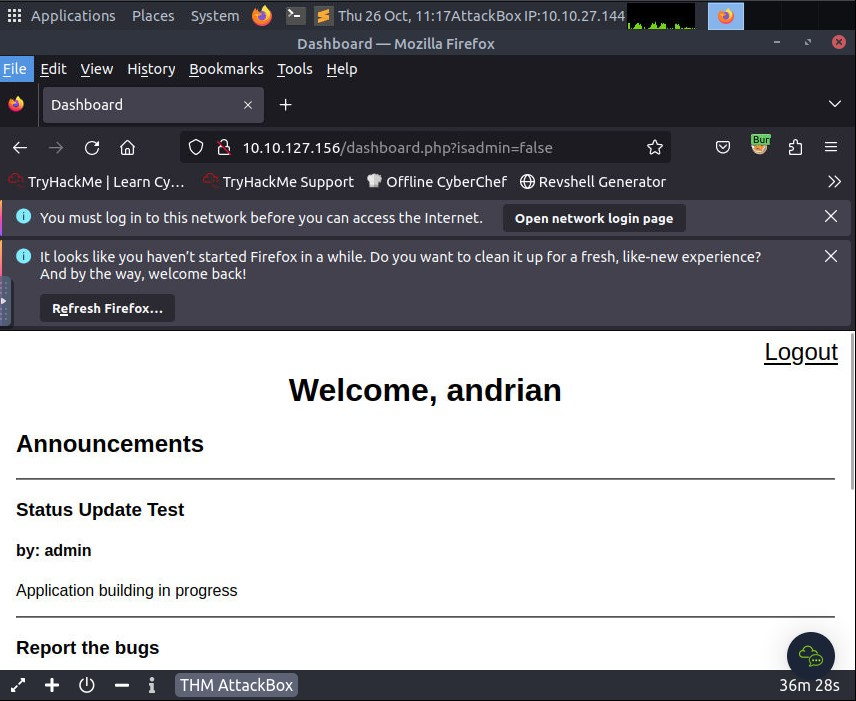


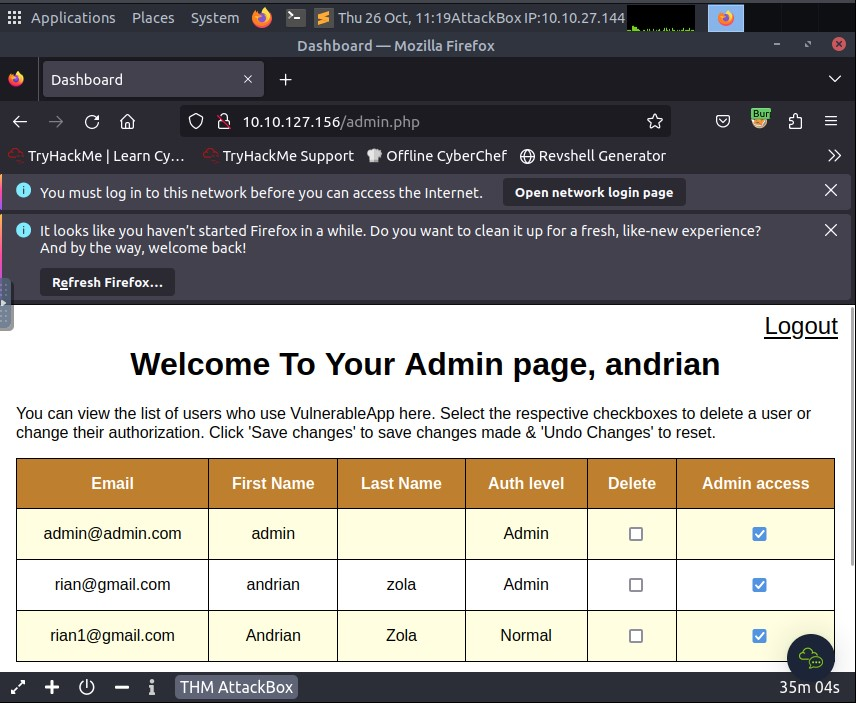
1. Berdasarkan gambar diatas, bahwa dapat melihat setelah menyelesaikan proses login, aplikasi web akan memberikan respons JSON yang berisi status, pesan, nama\_depan, nama\_belakang, is\_admin, dan redirect\_link yang digunakan oleh server untuk mengarahkan pengguna ke dashboard.php dengan parameter “isadmin” dalam URL.
2. Mengekploitasi Web Aplikasi

Pada tugas sebelumnya, kita telah mempelajari bahwa file functions.php mengembalikan respons JSON pada saat login. Respons tersebut berisi redirect\_link dengan parameter yang dapat kita gunakan untuk menguji kerentanan kontrol akses.

Untuk mulai menguji kerentanan ini, kita dapat mencegat respons HTTP dan menyalin nilai parameter redirect\_link ke bilah alamat kita.

Gambar 2. Mengubah isadmin=false menjadi isadmin=true





1. Rekomendasi Perbaikan / Kesimpulan

* Kontrol akses yang rusak adalah kerentanan keamanan yang terjadi ketika sebuah sistem gagal menegakkan kontrol akses dengan benar, yang dapat mengakibatkan pengguna yang tidak berwenang mendapatkan akses ke informasi sensitif atau melakukan tindakan yang tidak diizinkan untuk mereka lakukan.
* Eskalasi hak istimewa horizontal terjadi ketika pengguna dapat mengakses data atau melakukan tindakan yang tidak diizinkan untuk dilakukan dalam tingkat hak istimewa mereka sendiri. Hal ini dapat berbahaya karena dapat memungkinkan penyerang yang telah mendapatkan akses ke sistem untuk bergerak secara lateral melalui jaringan dan mengakses sumber daya tambahan atau data sensitif.
* Eskalasi hak istimewa vertikal terjadi ketika seorang pengguna dapat memperoleh akses ke data atau melakukan tindakan yang diperuntukkan bagi pengguna dengan tingkat hak istimewa yang lebih tinggi, seperti administrator sistem. Hal ini bisa lebih berbahaya karena memungkinkan penyerang untuk mendapatkan kontrol penuh atas sistem dan berpotensi mengambil alih seluruh jaringan.
* Dampak dari jenis eskalasi hak istimewa ini dapat bervariasi tergantung pada sistem tertentu dan tingkat akses yang diperoleh. Namun, secara umum, konsekuensinya dapat mencakup akses tidak sah ke informasi sensitif, kehilangan atau pencurian data, gangguan pada sistem atau layanan penting, dan bahkan kompromi jaringan secara keseluruhan. Oleh karena itu, penting untuk menerapkan kontrol akses yang kuat dan secara teratur memantau tanda-tanda akses atau aktivitas yang tidak sah.

Berikut ini adalah beberapa referensi yang dapat Anda berikan kepada pengembang PHP untuk membantu mereka menerapkan strategi mitigasi ini:

* Lembar Sontekan Konfigurasi PHP OWASP
* PHP dengan Cara yang Benar: Keamanan
* Pengkodean Aman dalam PHP