BAB IV

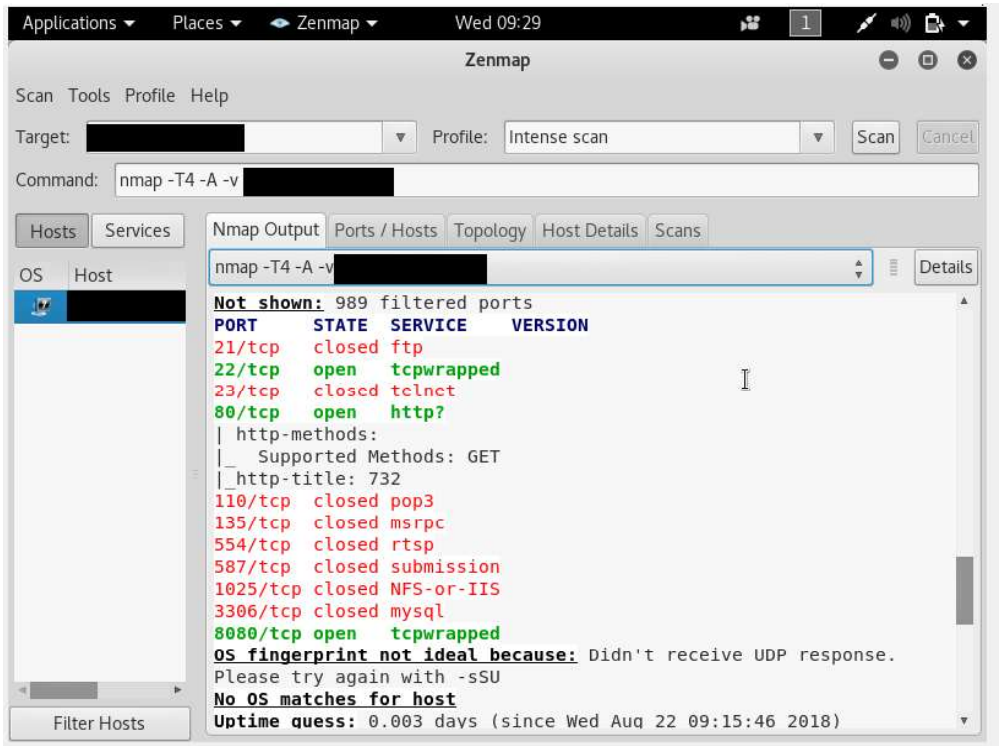
HASIL DAN PEMBAHASAN

4.1 *Scope*

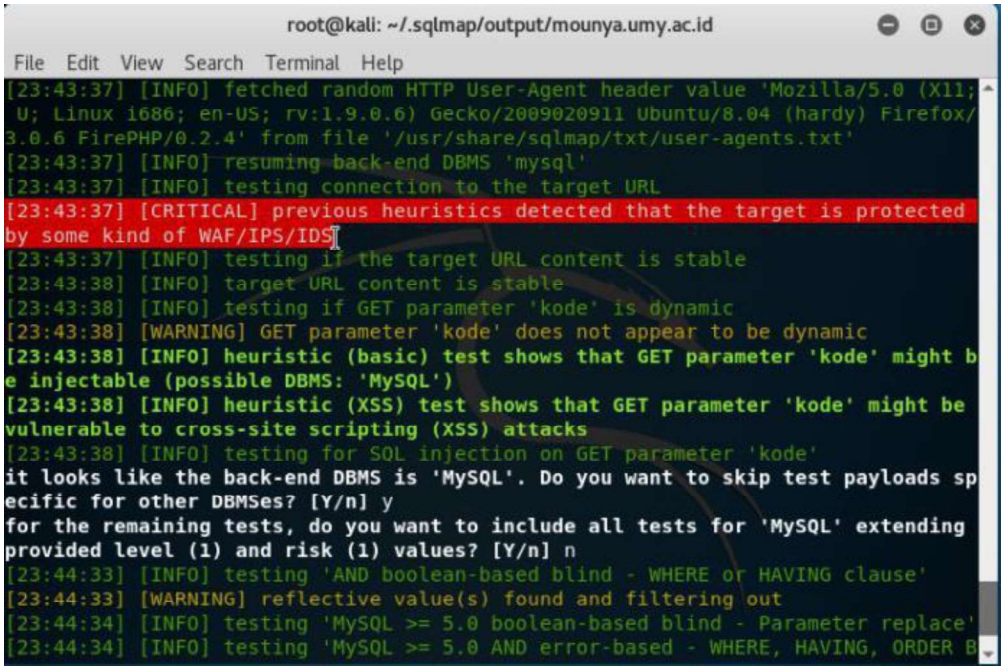
Peneliti menetapkan cakupan pada web aplikasi MoU Perguruan Tinggi XYZ untuk menjadi target *penetration testing*. Alasan ditetapkannya aset tersebut adalah merujuk pada penelitian sebelumnya yang membahas tentang daftar urutan aset penting milik Institusi Pendidikan XYZ (Yuhaz, 2018).

4.2 *Reconnaissance*

Pemindaian informasi aset yang dalam hal ini adalah web aplikasi MoU. Hasil pemindaian ini berupa informasi sistem operasi yang digunakan, yaitu Linux Ubuntu. Alamat IP 103.xxx.xxx.xxx. *Port* yang terbuka adalah *port* 22, 80, 8080 dan 25. Gambar 4.1 menerangkan tentang hasil pemindaian menggunakan *tool* Zenmap.



Gambar 4. 1 Hasil Scan Menggunakan Zenmap



Gambar 4. 2 Hasil Scan SQLMap

Kemudian peneliti melakukan *scan* menggunakan *tool* Sqlmap seperti yang ditunjukan pada gambar 4.2. *Scan* menggunakan Sqlmap bertujuan untuk memastikan *keyword* kueri yang bisa digunakan, versi *database*, jenis *database* yang dipakai dan *firewall* yang melindungi web MoU. Berdasarkan hasil *scan* dari sqlmap ditemukan *database* yang dipakai yaitu mysql dengan versi 5.5.34, ditemukan pula bahwa *web server* dari aplikasi web MoU ini dilindungi oleh *firewall* sehingga beberapa *keyword* kueri yang biasa digunakan untuk SQL *injection* terkena *filter*.

4.3 *Vulnerability Detection*

Pada tahap identifikasi celah keamanan, dilakukan pada web aplikasi MoU Perguruan Tinggi XYZ menggunakan *tool* Nessus. Celah keamanan pada MoU Perguruan Tinggi XYZ ditampilkan dalam **Tabel 4.1**.

Tabel 4. 1 Kerentanan Yang Ditemukan Menggunakan *Tool* Nessus

No	Jenis Kerentanan	Tingkat	Deskripsi
1	CGI Generic SQL *Injection*	*High*	url Web aplikasi bisa menerima parameter yang bisa disisipi *kueri* SQL *injection*
2	*CGI Generic Cookie Injection Scripting*	*Medium*	Dengan memanfaatkan masalah ini, seorang penyerang mungkin mampu melakukan *cookies injection*. Tergantung pada struktur aplikasi web jenis serangan yang bisa dilakukan *session fixation*
3	*Web Application SQL Backend Identification*	*Medium*	*database* yang di identifikasi berupa mysql
4	CGI Generic XSS	*Medium*	*Remote web server host* CGI gagal untuk membersihkan *request string* dari javascript yang berbahaya. Serangan yang mungkin terjadi penyerang menjalan kan javascript pada web tersebut
5	CGI Generic HTML *Injections*	*Medium*	*Remote web server host* CGI gagal untuk membersihkan *request string* dari javascript yang berbahaya. Serangan yang mungkin terjadi penyerang menjalan kan javascript pada web tersebut

No	Jenis Kerentanan	Tingkat	Deskripsi
6	Web Application Information Disclosure	*Medium*	web aplikasi yang ada di remote *web server* menunjukan physical path ke direktori ketika mengirimkan *malformed request*
7	Web Application Potentially Vulnerable to Clickjacking	*Medium*	*Remote web server* tidak menetapkan X-Frame-Option response header. Serangan yang mungkin dilakukan adalah clickjacking
8	Web Server Transmits Cleartext Credentials	*Low*	Web server mengirimkan data kredensial dalam bentuk *cleartext*

4.4 *Information Analysis & Planning*

Hasil dari *vulnerability scanning* menampilkan informasi terkait celah keamanan pada web aplikasi MoU Perguruan Tinggi XYZ. Peneliti mengambil celah yang digunakan untuk bahan *penetration testing*, diantaranya adalah sebagai berikut:

Tabel 4. 2 Kerentanan Yang Akan Diuji

No.	Jenis Kerentanan	Tingkat	Deskripsi
1	CGI Generic SQL *Injection*	High	url web aplikasi bisa menerima parameter yang bisa disisipi *kueri* SQL *Injection*

Alasan peneliti memilih kerentanan ini karena SQL Injection merupakan kerentanan yang paling sering ditemukan pada sebuah website dan memiliki dampak yang berbahaya jika bisa dieksploitasi.

4.5 *Penetration Testing*

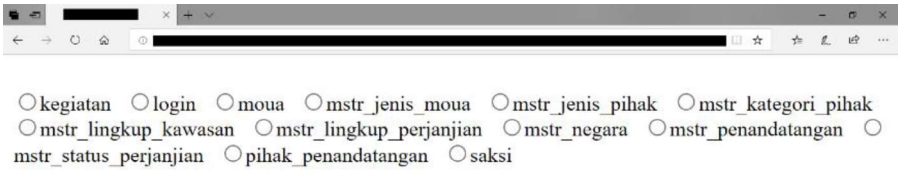
Peneliti melakukan *penetration testing* pada web aplikasi MoU Institusi Pendidikan XYZ dengan memanfaatkan celah yang telah dipilih pada **Tabel 4.2**. Hasil pengujiannya adalah sebagai berikut:

4.5.1 *CGI Generic SQL Injection*

Peneliti memanfaatkan celah keamanan ini untuk mengetahui isi *database* dari web MoU. Selanjutnya peneliti menuliskan kueri pada URL web MoU tetapi pada percobaan ini kueri yang dimasukkan terkena filter dari *firewall* dan menghasilkan *gateway time out* dari *server* seperti pada gambar 4.3.

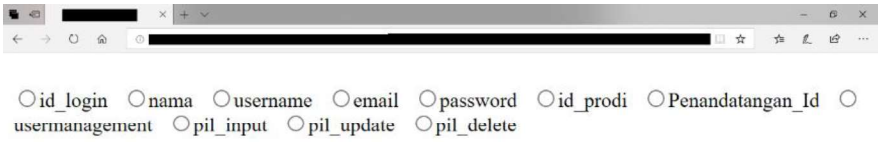
Gambar 4. 3 Gateway Time Out

Kemudian, untuk bisa melewati *filter* yang dimiliki *firewall* peneliti melakukan enkode dengan menambahkan karakter khusus dan mengganti struktur URL agar *kueri* yang dituliskan tidak terdeteksi sebagai *malicious code.* Selanjutnya, peneliti menggunakan gabungan karakter khusus dan angka untuk mengubah *kueri* yang akan dituliskan ke *url*. Dengan metode ini peneliti berhasil mendapat kan daftar tabel yang digunakan web MoU sebagaimana di tunjukan pada gambar berikut.



Gambar 4. 4 Daftar Tabel Yang Digunakan

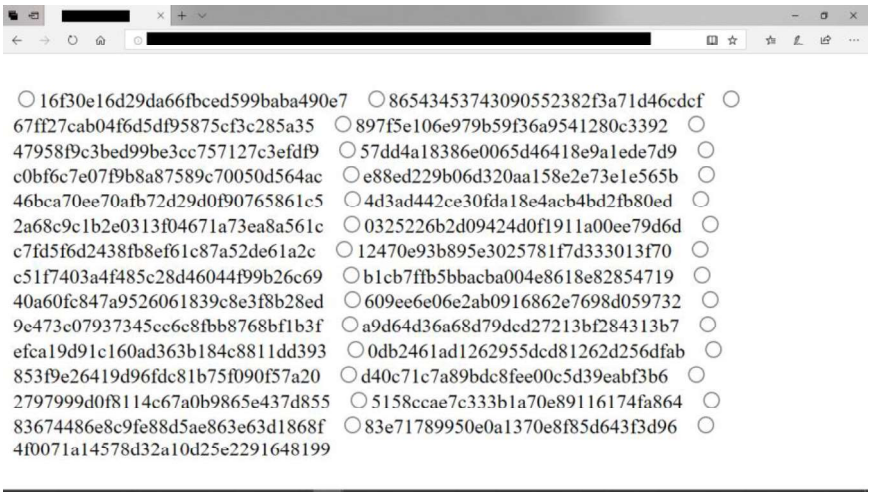
Pada gambar 4.4 menunjukkan daftar semua tabel yang digunakan aplikasi web MoU. Pada daftar tabel tersebut terdapat tabel *login* yang berisi data tentang akun yang bisa mengakses aplikasi web MoU, peneliti akan mengakses tabel *login* untuk mencoba mendapatkan nama pengguna dan kata sandi dari akun yang ada agar bisa mendapatkan hak akses.



Gambar 4. 5 Daftar Kolom Pada Tabel Login

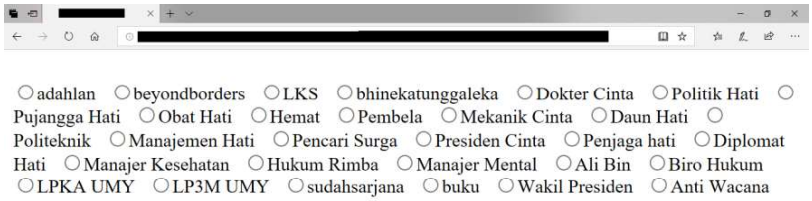
Pada gambar 4.5 menunjukkan daftar kolom yang ada pada tabel login, di sini terlihat semua kolom yang terdapat pada tabel login yaitu id_login, nama, *username*, *email*, *password*, id_prodi, Penandatangan_Id, usermanagement,

pil_input, pil_update, pil_delete. Selanjutnya, peneliti mencoba untuk melihat isi dari kolom *username* dan *password*. Hasilnya ditunjukkan pada gambar 4.6 dan 4.7.



Gambar 4. 6 Data Yang Terdapat Pada Kolom Password

Gambar 4.5 menunjukkan data dari kolom *password*. Pada kolom *password* ini telah dilakukan enkripsi untuk melindungi dan menyembunyikan data asli sehingga yang terlihat adalah karakter atau nomor yang acak. Dan gambar 4.6 merupakan daftar *username* yang berada pada kolom *username*.



Gambar 4. 7 Data Yang Terdapat Pada Kolom Username

4.6 *Privilege Escalation*

Sebagaimana yang telah dibahas pada tahap *penetration testing* peneliti telah mendapatkan *username* dan *password*. Pada kasus ini *password* yang tersimpan di *database* telah dienkripsi sehingga harus melakukan dekripsi untuk bisa mendapatkan data aslinya. Akan tetapi proses dekripsi akan memakan waktu yang cukup lama untuk menemukan algoritma yang cocok dengan enkripsi yang digunakan pada aplikasi web MoU sehingga, eksploitasi hanya sebatas membaca isi *database*.

4.7 *Reporting*

Pada tahap ini peneliti menulis laporan yang akan diberikan kepada Perguruan Tinggi XYZ. Laporan ini nantinya akan digunakan sebagai dasar untuk evaluasi dan perbaikan aplikasi web MoU Perguruan Tinggi XYZ. Laporan akan disertakan pada lampiran.