

ANALISIS DAN EVALUASI *VULNERABILITY CONTENT MANAGEMENT SYSTEM MyBB DAN PhpBB*

Irwansyah

Teknik Komputer, Universitas Binadarma
Jl. A. Yani No. 12 Plaju Palembang
email:irwansyah@mail.binadarma.ac.id

Abstrak – Kemudahan yang ditawarkan dalam membangun Website menjadikan CMS sangat diminati para webmaster, sebagai dasar untuk membangun sebuah website. Kemudahan menggunakan Content Management System belum dapat memastikan keamanan informasi yang kita masukan ke website yang kita buat. Keamanan informasi sangatlah berharga karena apabila informasi berharga didapatkan oleh pihak-pihak yang tidak bertanggung jawab, maka informasi tersebut dapat disalah gunakan. Kegiatan hacking website atau biasa disebut defacing sangat marak terjadi, Metode Penelitian yang digunakan dalam penelitian ini menggunakan penelitian tindakan atau Action Research. Ada lima tahapan dalam penelitian yang merupakan siklus dari Action Research, yaitu : Melakukan diagnosa, rencana tindakan, melakukan tindakan, melakukan evaluasi dan pembelajaran. Adapun hasil yang peneliti dapatkan pada penelitian ini adalah Mybb ditemukan 3 kerentanan yang beresiko tinggi (high risk), 1 kerentanan yang beresiko menengah (medium risk), 1 kerentanan yang beresiko rendah (low risk), dan 1 berupa informational alert. Sementara Phpbb hanya 1 kerentanan beresiko tinggi (high risk), dan 1 informational alert.

Kata Kunci: Website, CMS, Mybb, Phpbb

I. PENDAHULUAN

Pada saat ini website merupakan salah satu layanan informasi yang banyak diakses oleh pengguna internet di dunia. Sebagai salah satu layanan informasi perlu dibangun website yang mampu menangani permintaan dari banyak pengguna dengan baik. Banyak cara untuk membuat sebuah website salah satunya dengan Content Management System (CMS). Content Management System atau CMS, merupakan sebuah sistem yang memberi kemudahan dalam mengelola atau melakukan perubahan isi sebuah website. Dengan adanya Content Management System proses pembuatan website menjadi lebih mudah dan fleksibel. Kemudahan yang ditawarkan dalam membangun Website menjadikan CMS sangat diminati para webmaster, sebagai dasar untuk membangun sebuah website.

Berikut ini beberapa jenis CMS yang biasa digunakan untuk membangun sebuah website, antara lain : wordpress, mambo, joomla, drupal, phpbb, mybb, aura, dan lain – lain. Dari beberapa CMS yang tersedia ini, tentu memiliki jenis kerentanan yang berbeda – beda. Pada penelitian ini penulis akan mencoba menguji kerentanan pada CMS yaitu phpbb dan MyBB. Adapun tujuan dari penelitian ini adalah mengevaluasi kerentanan Content Management System (CMS), agar dapat menjadi pertimbangan bagi webmaster dalam memilih CMS yang aman untuk pembuatan website. Metode Penelitian yang

digunakan dalam penelitian ini menggunakan penelitian tindakan atau Action Research.

II. LANDASAN TEORI

2.1. Vulnerability Assessment

Vulnerability atau celah keamanan adalah suatu kelemahan yang mengancam nilai integrity, confidentiality dan availability dari suatu aset. Vulnerability tidak hanya berupa software bugs atau kelemahan security jaringan. Namun kelemahan seperti pegawai yang tidak ditraining, dokumentasi yang tidak tersedia maupun prosedur yang tidak dijalankan dengan benar. Vulnerability biasa dikategorikan ke dalam tiga bagian, yaitu kelemahan pada system itu sendiri, jalur akses menuju kelemahan sistem, serta kemampuan dari seorang hacker untuk melakukan attacking. [1]

2.2. CMS(Content Management System)

Content Management System (CMS) adalah sistem yang digunakan untuk mengatur situs web. Biasanya, CMS mengandung dua elemen: Content Management Application (CMA) dan Content Delivery Application (CDA). CMA merupakan elemen yang memudahkan seorang manajer isi (content manager) atau penulis – tanpa harus mengetahui Hypertext Markup Language (HTML) – untuk membuat, mengatur, mengubah dan menghapus isi dari situs web. Elemen CDA digunakan untuk

menyusun informasi untuk memperbarui isi situs web.[2]

2.3. Teknik-Teknik Attacking Yang Digunakan

Terdapat banyak sekali tipe dan jenis serangan yang terjadi di dunia maya. Sesuai dengan sifat dan karakteristiknya, semakin lama model serangan yang ada semakin kompleks dan sulit dideteksi maupun dicegah. Berikut adalah beberapa jenis model serangan yang kerap terjadi.[1].

1. SQL Injection

Pada dasarnya *SQL Injection* merupakan cara mengeksploitasi celah keamanan yang muncul pada level atau "layer" *database* dan aplikasinya. Celah keamanan tersebut ditunjukkan pada saat penyerang memasukkan nilai "string" dan karakter-karakter contoh lainnya yang ada dalam instruksi SQL; dimana perintah tersebut hanya diketahui oleh sejumlah kecil individu (baca: *hacker* maupun *cracker*) yang berusaha untuk mengeksploitasinya. Karena tipe data yang dimasukkan tidak sama dengan yang seharusnya (sesuai dengan kehendak *program*), maka terjadi sebuah aktivitas "liar" yang tidak terduga sebelumnya - dimana biasanya dapat mengakibatkan mereka yang tidak berhak masuk ke dalam sistem yang telah terproteksi menjadi memiliki hak akses dengan mudahnya. Dikatakan sebagai sebuah "injeksi" karena aktivitas penyerangan dilakukan dengan cara "memasukkan" *string* (kumpulan karakter) khusus untuk melewati filter logika hak akses pada *website* atau sistem komputer yang dimaksud.

2. XSS (Cross Site Scripting)

Cross Site Scripting (CSS) adalah suatu serangan dengan menggunakan mekanisme "injection" pada aplikasi web dengan memanfaatkan metode *HTTP GET* atau *HTTP POST*. *Cross Site Scripting* biasa digunakan oleh pihak-pihak yang berniat tidak baik dalam upaya mengacaukan konten website dengan memasukkan naskah program (biasanya *java script*) sebagai bagian dari teks masukan melalui formulir yang tersedia.

3. Missing Function Level Access Control

Hampir semua aplikasi *web* memverifikasi fungsi tingkat hak akses sebelum membuat fungsi yang terlihat di UI. Namun, aplikasi perlu ditampilkan untuk memeriksa kontrol akses yang sama pada *server* ketika setiap fungsi diakses. Jika permintaan tidak diverifikasi, penyerang akan dapat melakukan permintaan mengakses fungsi yang tidak sah.

4. Brute Force Attack

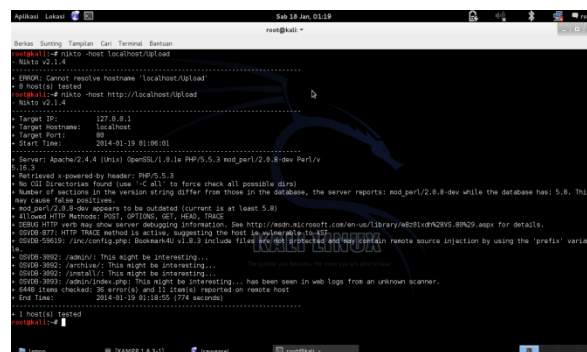
Serangan *brute-force* adalah sebuah teknik serangan terhadap sebuah sistem keamanan komputer yang menggunakan percobaan terhadap semua kunci yang mungkin. Pendekatan ini pada awalnya merujuk pada sebuah program komputer yang mengandalkan kekuatan pemrosesan komputer dibandingkan kecerdasan manusia. Sebagai contoh, untuk menyelesaikan sebuah persamaan kuadrat seperti $x^2+7x-44=0$, di mana x adalah sebuah integer, dengan menggunakan teknik serangan *brute force*, penggunaanya hanya dituntut untuk membuat program

yang mencoba semua nilai *integer* yang mungkin untuk persamaan tersebut hingga nilai x sebagai jawabannya muncul. Istilah *brute force* sendiri dipopulerkan oleh Kenneth Thompson, dengan mottonya: "When in doubt, use brute-force" (jika ragu, gunakan *brute-force*). [4].

III. PEMBAHASAN

3.1 Pengujian CMS Mybb dengan Nikto

Pada tahap pertama peneliti menguji CMS dengan Nikto. Nikto adalah *Tools Scanning Vulnerability website* yang dijalankan pada *OS Kali Linux*. Nikto berfungsi untuk mengidentifikasi direktori yang *vulnerability*. Untuk pengguna *windows* nikto tersedia pada *website* nikto itu sendiri, Peneliti menggunakan nikto yang telah ada atau sudah *terinstall* pada *OS Kali Linux*. Peneliti menggunakan nikto dengan mengetikkan perintah pada terminal "nikto -host www.target.com" pada penelitian ini peneliti mengetikkan perintah "nikto -host <http://localhost/mybb>" dengan hasil *report Tools* yang ada pada gambar 3.1.



Gambar 3.1 Hasil Scanning Mybb Vulnerability Menggunakan Nikto.

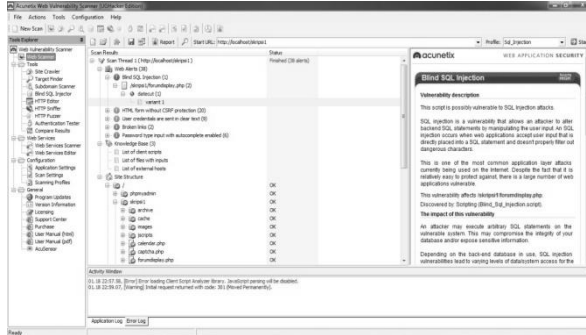
Dari hasil yang didapatkan, ditemukan beberapa direktori yang bersifat *informational alert* seperti :

- Directory `inc/config.php` = Directory not Protected and may contain remote source injection
- Directory `/admin` = the might be interesting
- Directory `/archive` = the might be interesting
- Directory `/install` = the might be interesting

Direktori `/inc/` yang berisikan file `config.php`, file `config.php` berisi konfigurasi *phpmyadmin* berupa *user* dan *password admin* tidak diproteksi atau tidak dihilangkan pada saat diakses pada *browser*. Direktori `/admin` disembunyikan, pada *panel admin* ini dapat dilakukan *Admin enum* apabila *coding* nya terdapat bug dan akan di eksekusi pada pembahasan selanjutnya.

3.2 Scanning Vulnerability pada Mybb menggunakan Tools Acunetix.

Pada scanning vulnerability menggunakan Acunetix ini dengan cara memasukkan alamat <http://localhost/case1> pada fitur web scanner dengan hasil report pada gambar 3.2 :



Gambar 3.2 Hasil Scanning Mybb Vulnerability Menggunakan Acunetix

Dari hasil report tools acunetix ditemukan pada scanning <http://localhost/case1> :

a. Direktori

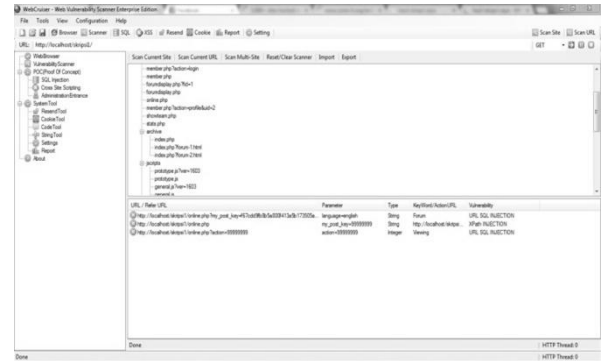
- Archive
- Cache
- Images
- jsScript
- dan file-file seperti calendar.php, captcha.php, forumdisplay.php, index.php dan file-file lainnya.

b. Vulnerability found

Blind Sql Injection pada localhost/case1/forumdisplay.php?datecut=2-2%2b4-2-2&fid=2&order=asc&sortby=starter : sedikit berbeda dari SQL Injection, teknik ini apabila tanda (') di injeksi tidak tampak pesan error tetapi ada beberapa tampilan yang berubah.

3.3 Scanning Vulnerability Mybb menggunakan tools Webcruiser

Peneliti menggunakan tools ini dikarenakan Tools ini mendeteksi parameter yang vulnerability khusus untuk SQL injection dan XSS agar dapat ditemukan secara cepat, dan Webcruiser akan menampilkan URL secara penuh apabila URL tersebut memiliki kemungkinan Vulnerability. Cara menggunakan tools ini dengan memasukkan URL "localhost/case1" pada form tools webcruiser yang telah di siapkan. Dengan hasil report pada gambar 3.3 :



Gambar 3.3 Hasil Scanning Vulnerability Mybb menggunakan Tools Webcruiser

Dari hasil report scanning tools webcruiser pada <http://localhost/case1> ditemukan beberapa kemungkinan vulnerability SQL Injection sebagai berikut :

- http://localhost/case1/online.php?my_post_key=f67cdd9fb8b5a800f413a5b173505a20&language=english
- <http://localhost/case1/online.php?action=999999999>

3.4 Teknik Validasi Vulnerability

Tahap ini adalah tahap peneliti melakukan eksekusi Vulnerability atau memastikan apakah vulnerability yang ditemukan memiliki kerentanan yang membahayakan atau tidak.

1. SQL Injection Pada Mybb

Peneliti mencoba menelusuri url-url yang memiliki kemungkinan vulnerability berdasarkan direktori yang telah didapatkan dari tahap-tahap sebelumnya. Direktori dan file yang ditemukan seperti gambar 3.4.



Gambar 3.4. Direktori yang ditemukan dari hasil scanning Acunetix pada Mybb

Dari hasil direktori yang didapatkan, pada peneliti mengidentifikasi SQL Injection didapatkan 2 (dua) Vulnerability SQL Injection pada Mybb yaitu :

- <http://localhost/case1/ajaxfs.php?usertooltip=1>
- <http://localhost/case1/ajaxfs.php?tooltip=1>

Eksekusi URL

<http://localhost/case1/ajaxfs.php?usertooltip=1>



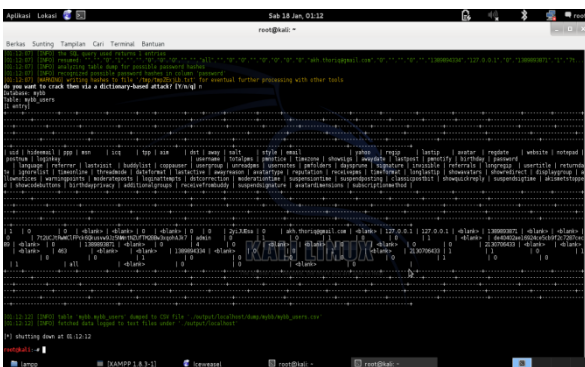
Gambar 3.5 Tampilan Eksekusi URL Sql Error yang ditambahkan (^)

Pada url diatas peneliti menemukan pesan error apa bila di injeksi dengan character (^) pada akhir url. Error tersebut membuktikan bahwa url tersebut adalah vulnerability SQL Injection. Namun url tersebut belum dapat memastikan apakah dapat menembus keamanan dari mybb dengan menginjeksi query-query sql pada url tersebut. Peneliti melakukan eksekusi vulnerability dengan menggunakan manual dan tools dengan hasil pada gambar 3.6.



Gambar 3.6 Hasil injeksi query SQL pada mybb

Dari hasil Injeksi query Sql pada Vulnerability diatas, url tidak bisa di injeksi dikarenakan pada saat injeksi order by dari 1 sampai 10 tampilan header tidak kembali utuh seperti awal, kemungkinan column lebih dari 10. Peneliti melanjutkan eksekusi dengan tools lain yaitu : sqlmap. Hasil diatas eksekusi url vulnerability SQL Injection <http://localhost/case1/ajaxfs.php?tooltip=1> menggunakan tools sqlmap pada OS Kali Linux, Tabel-tabel pada database dapat ditemukan secara keseluruhan termasuk tabel mybb_users yang berisikan user dan password admin mybb. Seperti pada gambar 3.7 berikut :



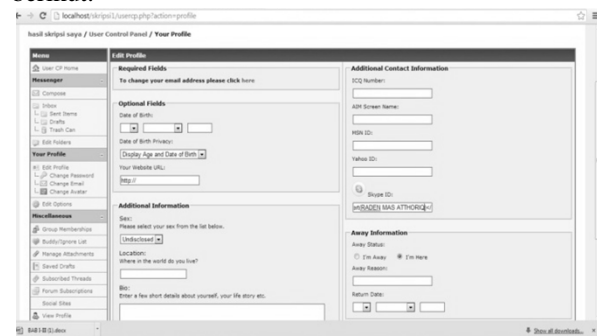
Gambar 3.7 Isi tabel mybb_users

Dari hasil tools sqlmap tabel mybb_users dapat di lihat, ini menandakan bahwa url "<http://localhost/case1/ajaxfs.php?tooltip=1>" benar-benar rentan. Dari eksekusi tools sqlmap dapat melihat isi tabel mybb_users yang berisikan informasi penting seperti username = admin dan password enskripsi(MD5)=de40402ae16924ce5cb9f2c7287cec89. Vulnerability ini sangat membahayakan dikarenakan informasi user dan password telah didapatkan. Selain itu resiko yang ditimbulkan dari kerentanan ini adalah :

- Dapat menghapus file-file penting.
- Dapat mendapatkan username dan password lainnya.
- Dapat merusak isi cms mybb tersebut.
- Dapat terjadinya Defacing atau mengganti halaman index.
- Dapat menghapus seluruh database.

2. Cross Site Scripting (XSS) pada Mybb

Peneliti melakukan identifikasi secara manual dengan cara menelusuri form-form, url-url, dan lain-lain dengan cara testing injeksi script seperti `<script>alert(PESAN)</script>,';!--></script>alert(PESAN)</script>` apabila ada pesan peringatan(alert) maka terdapat vulnerability XSS. Dari hasil identifikasi secara manual tersebut peneliti menemukan beberapa vulnerability adalah sebagai berikut:

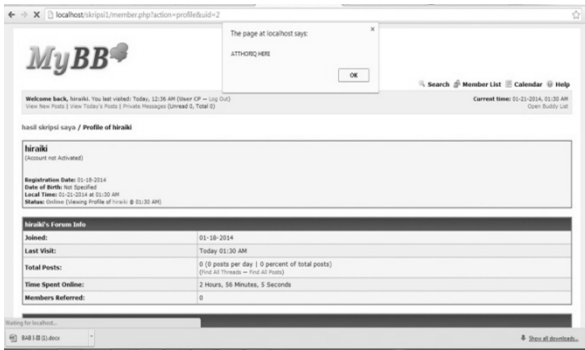


Gambar 3.8 Form Skype ID Vulnerability XSS pada Mybb

Pada Gambar 3.8 peneliti menemukan kerentanan/vulnerability XSS pada form Skype ID pada mybb. Vulnerability ini berada pada usercp atau edit profile user pada form Skype ID. Kerentanan ini apabila di injeksi dengan script bahasa pemrograman java. Peneliti akan memasukan/injeksi menggunakan script java

`><script>alert("Skype ID XSS")</script>`

Hasil dari injeksi script java tersebut akan memunculkan sebuah java alert seperti gambar 3.9 berikut ini :

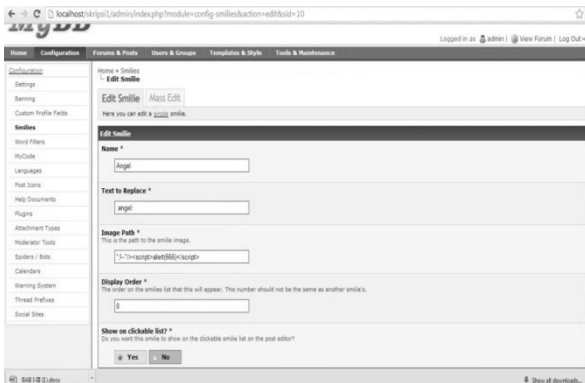


Gambar 3.9 Hasil Injeksi XSS pada form Skype ID Mybb

Dengan hasil injeksi script "><script>alert("hasil inject")</script>" pada form Skype ID, membuktikan bahwa form Skype ID pada mybb adalah benar-benar *vulnerability* XSS. Pada saat injeksi script java tampilan pada browser mengeluarkan peringatan atau biasa disebut java alert. Java alert tersebut timbul pada saat kita memiliki resiko tinggi karena dapat menimbulkan resiko-resiko lainnya seperti:

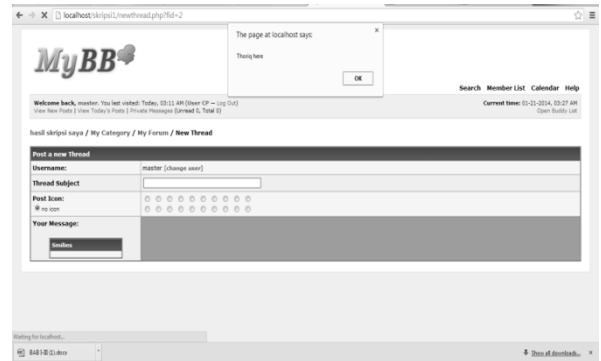
- Dapat mengacaukan konten
- Dapat mengupload *backdoor* walaupun teknik XSS untuk *upload backdoor* lebih rumit dari SQL Injection, namun jika *backdoor* diupload maka resiko lain dapat ditimbulkan seperti *defacement*, *drop database*, mencuri *username* dan *password*.

Vulnerability XSS pada Panel admin (edit smiles) pada Mybb



Gambar 3.10 Form edit Smiles admin panel vulnerability XSS pada Mybb

Peneliti menemukan kerentanan *vulnerability* XSS pada panel admin di bagian *edit smiles*, pada form *image path* terdapat kerentanan XSS. Kerentanan ini dapat diinjeksi dengan script bahasa pemrograman java. Peneliti akan memasukan/injeksi menggunakan script java "><script>alert("HASIL INJECT")</script>" untuk membuktikan bahwa form *images path* pada *smilies* adalah *vulnerability*. Hasil dari injeksi script java tersebut akan memunculkan sebuah java *alert* seperti gambar 3.11 berikut ini



Gambar 3.11 Hasil Injeksi XSS pada form edit smilies pada panel admin Mybb

Setelah diinjeksi script java ("><script>alert("hasil inject")</script>" pada form *images edit* pada *smilies admin*) tampilan pada browser mengeluarkan peringatan atau biasa disebut java alert. Java alert tersebut timbul pada saat kita membuat *NewThread* pada user biasa ataupun admin.

3. Missing Function Level Access Control pada Mybb

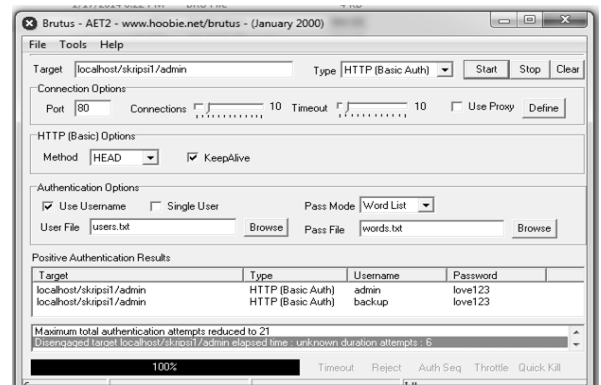
Missing Function Level Access Control adalah jenis kerentanan yang memiliki akses fungsi yang tidak sah atau tidak terproteksi secara baik. Kerentanan/*vulnerability* ini dapat berupa file upload yang tidak memiliki fungsi yang sah / tidak adanya pembatas jenis file yang di upload, direktori ataupun file yang berhubungan dengan *database*, *user*, dan *password* yang tidak terproteksi.

Dari hasil scanning *vulnerability* menggunakan tools *nikto* pada tahap sebelumnya, peneliti menemukan beberapa *vulnerability* jenis ini yaitu :

Directory inc/config.php = Directory not Protected and may contain remote source injection pada Mybb.

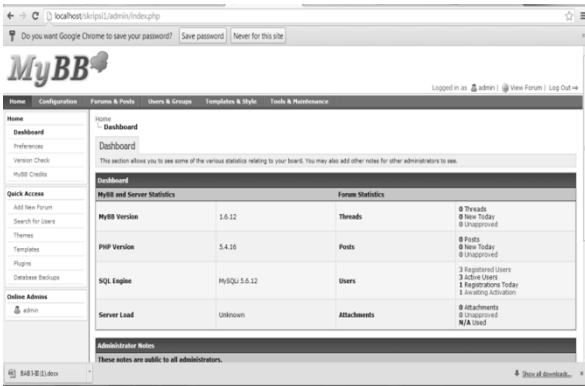
4. Brute Force Attack pada Mybb

Pada bagian ini peneliti akan melakukan pengujian menggunakan *Brute force* pada CMS mybb ditunjukkan hasil identifikasi pada Gambar 3.12. Pengujian ini dilakukan dengan menggunakan tools brutus pada *system operasi windows*.



Gambar 3.12 Hasil brute force attack pada mybb

Dari hasil *Brute force attack* menggunakan *tools* brutus ditemukan informasi yang sangat penting yaitu *username* dan *password*. *Username* dan *password* tersebut harus di uji kebenarannya dengan melakukan *session login* seperti pada gambar 3.13 :



Gambar 3.13 Hasil *session login Success*

Pada gambar diatas peneliti memasukkan *username* dan *password admin* yang didapatkan dari hasil *brute force attack* dan mendapatkan hasil *success*.

Dari hasil pengujian-pengujian Mybb dan Phpbb pada tahap sebelumnya, didapatkan hasil sebagai berikut :

Tabel 3.1 Hasil Pengujian CMS Mybb dan Phpbb

NO	Jenis Kerentanan	Level	CMS Mybb	CMS Phpbb
			Keterangan	Keterangan
1	SQL Injection	High Risk	http://localhost/skripsi1/ajaxf.php?tooltip=1 terdapat vulnerability pada url ini dan mempunyai resiko tinggi karena <i>username</i> dan <i>encrypt password</i> dapat ditemukan	Tidak ditemukan kerentanan
2	SQL Injection	Medium Risk	http://localhost/skripsi1/ajaxf.php?Usertooltip=1 Pada url ini terdapat pesan error tapi tidak bisa di injeksi menggunakan <i>tools</i> maupun <i>manual</i> .	Tidak ditemukan kerentanan
3	XSS (Cross Site Scripting)	High Risk	Pada form Skype ID pada edit profil dapat merusak tampilan jika diinjeksi <i>script</i> java "><script>alert('ATTHORIQ Here')</script>".	Tidak ditemukan kerentanan
4	XSS (Cross Site Scripting)	Low Risk	Pada form edit <i>profiles..image path</i> pada <i>admin panel</i> jika diinjeksi <i>script</i> java "><script>alert('ATTHORIQ Here')</script>" akan mengubah tampilan. Akan tetapi injeksi XSS ini harus mendapat akses <i>admin</i> .	Tidak ditemukan kerentanan
5	Missing Function Level Access Control	Informational Alert	Hasil <i>scavenging</i> nikto ditemukan letak direktori /admin, /archive/, install, dan letak file config.php (/inc/config.php)	Hasil <i>scavenging</i> nikto ditemukan direktori yang tidak diproteksi /download, /images, /docs dan lain-lain ini tidak berbahaya dikarenakan hanya bersifat <i>informational</i>
6	Brute Force Attack	High Risk	Hasil <i>Brute force</i> menggunakan <i>tools</i> brutus telah menemukan <i>username</i> dan <i>password</i> yang benar.	Hasil <i>Brute force</i> menggunakan <i>tools</i> brutus telah menemukan <i>username</i> dan <i>password</i> yang benar.

IV.KESIMPULAN

Berdasarkan hasil pengujian yang dilakukan pada 2 (dua) *Content Management System* (CMS) mybb dan phpbb dapat disimpulkan bahwa : mybb memiliki lebih banyak kerentanan atau *vulnerability* dibandingkan CMS Phpbb. Mybb ditemukan 3 kerentanan yang beresiko tinggi (*high risk*), 1 kerentanan yang beresiko menengah (*medium risk*), 1 kerentanan yang beresiko rendah (*low risk*), dan 1 berupa *informational alert*. Sementara Phpbb hanya 1 kerentanan beresiko tinggi (*high risk*), dan 1 *informational alert*.

DAFTAR REFERENSI

- [1] Santoso, Hanif dkk. *Analisis Vulnerability Aplikasi iFace IT Telkom bandung*. paper UAS CS4633 Keamanan Sistem. Bandung, 2008
- [2] Satoto, Kodrat Iman, *Analisis Keamanan Sistem Informasi Akademik Berbasis Web Di Fakultas Teknik Universitas Diponegoro*. Artikel Ilmiah. Yogyakarta: Universitas Diponegoro, . 2009
- [3] Digdo, Girindro Pringgo, *Analisis Serangan dan Keamanan Pada Aplikasi Web*, PT Elex Media Komputindo: Jakarta, 2012
- [4] ptitk.ub.ac.id/doro/download/article/file/DR00098201312

Biodata Penulis

Irwansyah.,M.M., M.Kom, memperoleh gelar Sarjana Komputer (S.Kom), Jurusan Teknik Informatika Univ. Bina Darma Palembang, lulus tahun 2000. Memperoleh gelar Magister Manajemen (M.M) Program Pasca Sarjana Magister Manajemen Universitas Bina Darma Palembang, lulus tahun 2006. Memperoleh gelar Magister Komputer (M.Kom) Program Pasca Sarjana Magister Teknik Informatika Universitas Bina Darma Palembang, lulus tahun 2011.Saat ini menjadi Dosen di Universitas Bina Darma Palembang.