VULNERABILITY ASSESSMENT PADA *WEB SERVER* WWW.BINADARMA.AC.ID

Harry Purmanta Siagian[1], M. Akbar[2] , Andri[3]
Dosen Universitas Bina Darma [2,3], Mahasiswa Universitas Bina Darma [1]
Jalan Jenderal Ahmad Yani No.12 Palembang
Pos-el: cextor.phl@gmail.com[1], akbar@mail.binadarma.ac.id[2], andri@mail.binadarma.ac.id[3]

ABSTRACT: *The virtual world is not a new thing in this day and age. Many activities of the individual to the organization using virtual worlds as a medium of information. However, just like the real world that has a negative side. The virtual world also has a negative side that cyber crimes (cyber crime). And most often the target of cyber attacks is the Web Server. For the sake of preserving digital information available on the Web server. Then any individual or organization is required to have a security system that minimizes bias cyber crime. Because one thing is for sure is that no one is safe in cyberspace. Of the other people who are not responsible for finding vulnerabilities web server www.binadarma.ac.id better my first who discovered and reported to administrators. Thus the researchers will analyze the security of the web server www.binadarma.ac.id.*

Keyword : *cyber crime, web server,* sistem keamanan informasi, *Vulnerability Assessment*.

ABSTRAK: Dunia maya sudah bukan hal yang baru pada zaman sekarang ini. Banyak kegiatan dari individu sampai ke organisasi yang menggunakan dunia maya sebagai media informasi. Akan tetapi, sama halnya seperti dunia nyata yang memiliki sisi negatif. Dunia maya juga memiliki sisi negatif yaitu kejahatan dunia maya (*cyber crime*). Dan yang paling sering menjadi target serangan *cyber crime* adalah *Web Server.* Demi menjaga informasi digital yang ada pada *Web server*. Maka setiap individu atau organisasi diharuskan memiliki sistem keamanan yang bias meminimalisir *cyber crime.* Karena satu hal yang pasti adalah tidak ada satupun yang aman di dunia maya. Dari pada orang lain yang tidak bertanggung jawab menemukan celah keamanan *web server* www.binadarma.ac.id lebih baik saya dahulu yang menemukannya dan melaporkan ke administrator . Maka dari itu peneliti akan melakukan analisa keamanan terhadap *web server* www.binadarma.ac.id.

Keyword : *cyber crime, web server,* sistem keamanan informasi, *Vulnerability Assessment*.

I. PENDAHULUAN

Sistem Informasi memiliki peranan besar di semua perusahaan atau institusi agar dapat menghasilkan keuntungan semaksimal mungkin dengan cara mengiklankan, menjual, mengadministrasi, dan mewujudkan produk baru. Perusahaan dipaksa untuk menciptakan dan memikirkan inovasi baru agar dapat bersaing dan bertahan hidup di dalam persaingan bisnis yang ketat. (ibisa, 2011:1).

Pada tahun 2005 Universitas Bina darma memiliki fasilitas informasi digital yaitu *www.binadarma.ac.id* dan hanya digunakan sebagai pengantar informasi saja. Lalu perkembangan zaman memaksa Universitas Bina Darma untuk melakukan perkembangan fasilitas pada *www.binadarma.ac.id.* Pada akhir 2011 *www.binadarma.ac.id* berkembang menjadi media mahasiswa untuk melakukan entry krs online, melihat KHS secara online dll. Fasilitas tersebut sangat membantu mahasiswa karena lebih efisien dari pada mengantri membawa kertas ke loket untuk mendapatkan KRS dan KHS. Akan tetapi satu hal yang pasti adalah tidak ada satupun yang aman di dunia cyber. (GOV-CSIRT, 2012). www.binadarma.ac.id pernah dianalisa oleh Muhammad Akbar, S.T,

M.IT dan Imam Perdana. Selaku dosen Universitas Bina Darma dan sampai pada saat ini belum pernah diuji coba lagi apakah masih aman atau belum dari adanya serangan *cyber*. Pengujian periodik terhadap sistem sangat penting. Tanpa pengujian periodik, tidak ada jaminan terhadap tindakan protektif yang dilakukan atau *patch* pengamanan yang diterapkan oleh administrator berfungsi sebagaiamana yang mestinya. *www.binadarma.ac.id* merupakan *website* terbuka yang bias diakses siapa saja. Tapi untuk fasilitas yang ada pada *www.binadarma.ac.id* seperti entry krs online, melihat transkrip nilai, dll hanya bias diakses oleh dosen dan mahasiswa universitas binadarma. Kejahatan di dunia maya sudah sangat marak dikalangan masyarakat. Bahkan anak SMA kelas 1 bisa melakukan *defacing* ke target meraka.

Meskipun umumnya kejahatan *cyber* yang terjadi hanya menimbulkan kesan negatif, memalukan atau ketidaknyamanan (seperti *defacing*), seperti situs www.presidensby.info pada rabu 1 september 2012 sempat di *deface* peretas. Pelaku meninggalkan jejak dengan menuliskan diri sebagai *Jember Hacker Team.* Namun tidak tertutup kemungkinan penyerang dapat membuat masalah yang lebih serius atau bahkan sangat merugikan.

Demi menjaga fasilitas yang sangat diperlukan oleh mahasiswa dan dari pada orang lain yang tidak bertanggung jawab yang menemukan pintu masuk atau celah ke web *server www.binadarma.ac.id,* akan lebih baik penelitian ini dilakukan. Dan jika menemukan pintu masuk atau celah keamanan akan dilaporkan ke admin *web server www.binadarma.ac.id* serta melakukan perbaikan (*patching*) secepatnya.

Berdasarkan latar belakang di atas, maka perumusan masalah dalam penelitian ini yaitu bagaimana menganalisa serta menemukan kelemahan pada web server www.binadarma.ac.id untuk meningkatkan keamanan dan kenyamanan pada layanan website www.binadarma.ac.id. Dan dapat memberikan rekomendasi perbaikan.

Agar penelitian tidak menyimpang dan tetap terarah diperlukan adanya batasan masalah.Batasan masalah dalam penelitian ini adalah web server yang di analisa adalah web server www.binadarma.ac.id serta menggunakan metode pengujian white box dan hanya sebatas pengujian saja.

Penelitian ini bertujuan untuk :

1. Mengkaji kontrol keamanan Internal maupun External sistem dengan mengidentifikasi ancaman yang dapat menimbulkan masalah serius terhadap aset organisasi.
2. Serta diharapkan dapat membantu administrator dalam mengidentifikasi celah keamanan dan menutupnya sebelum diketahui dan dimanfaatkan oleh pengguna yang tidak bertanggung jawab.

Manfaat dari penelitian ini adalah :

1. Meningkatkan keamanan pada layanan *website*.

2. Bagi pihak Universitas Bina Darma tidak perlu mengeluarkan biaya untuk menyewa suatu perusahaan yang menjual jasa keamanan.
3. Bagi penulis dapat menambah pengetahuan dan pemahaman tentang keamanan sistem terutama pada *web server*.
4. Bagi mahasiswa penelitian ini bermanfaat untuk keamanan dan kenyamanan dalam menggunakan layanan pada *website www.binadarma.ac.id.*

II. METODOLOGI PENELITIAN

2.1 Metode Pengumpulan Data

Dalam penyusunan penelitian ini penulis mengumpulkan data yang dibutuhkan dalam pengujian penetrasi menggunakan metode pengumpulan data sebagai berikut :

1. Pengamatan (*Observation*). Peneliti melakukan peninjauan langsung ke Universitas Bina Darma khususnya di bagian unit pelaksanaan teknis (UPT-SIM) yang merupakan bagian teknis system informasi di Universitas Bina Darma dengan pemilihan, pengubahan, pencatatan dan pengkodean serangkaian perilaku dan suasana berkenaan dengan objek penelitian
2. Wawancara (*Interview*). Demi mendapatkan informasi dan data-data yang berhubungan dengan penelitian ini maka penulis mengajukan beberapa pertanyaan dan diskusi kepada ketua unit satuan kerja TI Universitas Bina Darma serta jajarannya guna untuk mendapatkan informasi gambaran jaringan *web server* serta sistem keamanan *web server* untuk mempermudah penelitian.
3. Studi kepustakaan (*Literature*). Data diperoleh melalui studi kepustaka (*literature*) yaitu dengan mencari bahan dari internet, jurnal dan perpustakaan serta buku yang sesuai dengan objek yang akan diteliti.

2.2 Metode Penelitian

Peneliti menggunakan metode deskriftif dalam penelitian ini. Menurut Nasir (2003:54) bahwa metode deskriptif adalah suatu metode dalam meneliti status sekelompok manusia, suatu objek, suatu set kondisi, suatu actua pemikiran, ataupun suatu kelas peristiwa pada masa sekarang. Tujuan dari penelitian deskriptif ini adalah untuk membuat deskripsi, gambaran atau lukisan secara sistematis, actual dan akurat mengenai fakta-fakta, sifat-sifat serta hubungan antara fenomena yang diselidiki.

2.3 Metode *White Box* dan *Penetation Test*

White Box testing adalah pengujian yang memperhitungkan mekanisme internal dari sebuah sistem atau komponen (IEEE, 1990). *White Box testing* juga dikenal sebagai pengujian yang structural, pengujian kotak yang jelas (*clear box testing*), dan pengujian kotak kaca (*glass box testing*) (Beizer, 1995). Konotasi dari *clear box testing* dan *glass box testing* tepatnya menunjukkan bahwa anda memiliki visibilitas penuh terhadap internal kerja dari produk perangkat lunak, khususnya logika dan struktur dari kode.

Test Penetrasi (*Penetration Test*), yang dijelaskan oleh Anjar Priandoyo merupakan

Tingkatan ke-3 pada VA yang digunakan dalam pengukuran kerentanan. *Penetration test*sebenarnya menggunakan prinsip yang sama dengan network evaluation dimana pembedanya bahwa *penetration test* dilakukan dalam kondisi gelap, tanpa mengetahui konfigurasi dan kondisi sebenarnya seperti apa. Pada test penetrasi maka *assessor* akan menjumpai sistem sebagai sebuah kotak tertutup menghadapi penetrasi yang data dari luar.

III. HASIL

Setelah tahap demi tahap peneliti melakukan analisa dan uji coba dengan beberapa aplikasi yaitu Acunetix, Nikto, OpenVAS dan Retina Web Scanner. Maka dari hasil uji coba menggunakan aplikasi-aplikasi tersebut didapatlah beberapa kerentanan yang di akibatkan dari beberapa aplikasi yang terinstal di server kadaluwarsa.

1. ***Vulnerable applications version*** **:**

   a. Apache/2.2.6 appears to be outdated (current is at least Apache/2.2.19). Apache 1.3.42 (final release) and 2.0.64 are also current.

   b. mod_ssl/2.2.6 appears to be outdated (current is at least 2.8.31) (may depend on server version)

   c. OpenSSL/0.9.8g appears to be outdated (current is at least 1.0.0d). OpenSSL 0.9.8r is also current.

   d. PHP/5.2.5 appears to be outdated (current is at least 5.3.6)

   Vulnerable Applications Version adalah aplikasi yang sudah kadaluwarsa yang terinstal di *web server* dan membutuhkan *update* agar kelemahan-kelemahan yang diakibatkan aplikasi kadaluwarsa tertutup.

2. **Mencoba penetrasi dengan teknik penyerangan**

Pada bagian ini peneliti akan melakukan uji coba langsung terhadap *www.binadarma.ac.id* dengan menggunakan teknik penyerangan apakah berhasil atau tidak menggunakan teknik-teknik tersebut.

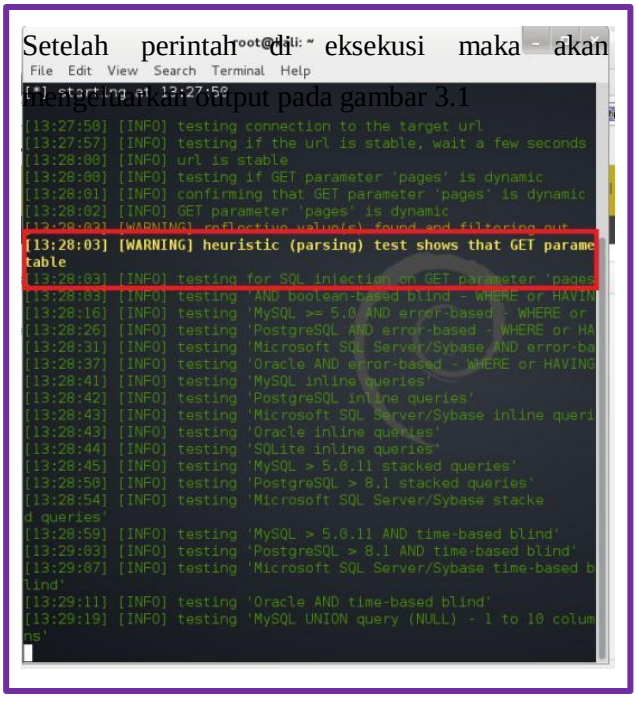
a. SQL Injection

Peneliti menggunakan perintah

```
sqlmap -u http://www.binadarma.ac.id/index.php?pages=content&id=121 --dbms=mysql
```

sqlmap merupakan tools *scanning* teknik SQL *Injection* secara automatis tergantung dari perintah si attacker. -u adalah perintah untuk membaca alamat target yang peneliti masukkan. --dbms=mysql merupakan perintah untuk melakukan *scanning* dengan aplikasi *database* pada target yaitu mysql

Setelah perintah di eksekusi maka akan mengeluarkan output pada gambar 3.1

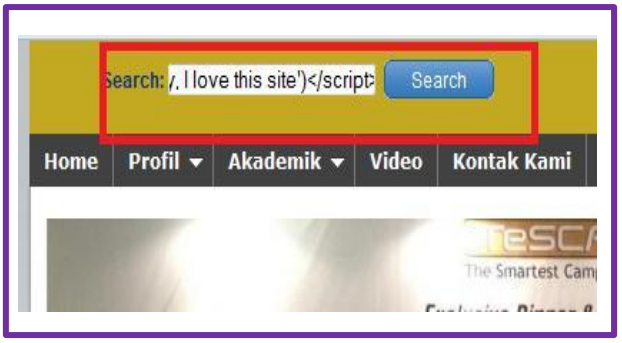


Gambar 3.1 : output dari sqlmap

Pada gambar 3.1 kotak merah merupakan bukti bahwa alamat target tidak rentan untuk SQL *Injection* metode GET. Begitu pula dengan sub domain dosen.binadarma.ac.id tidak rentan.

b. Cross-Site Scripting (XSS)

Terkadang, daripada menyerang sebuah server yang lebih sulit, seorang hacker bias saja memanfaatkan kelemahan yang ada pada sisi *client*. Dengan *Cross-Site Scripting* serangan yang dilakukan dengan cara menginjeksi/memasukkan *script* ke dalam website melalui browser.

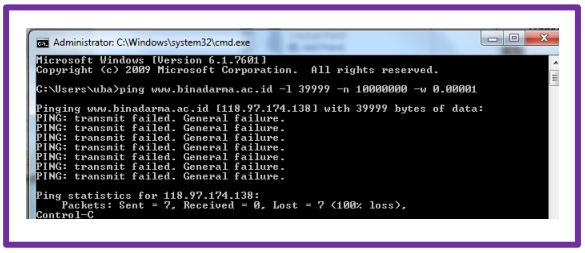


Gambar 3.2 : *form* di *inject* dengan XSS

Setelah di injeksi halaman tidak menampilkan perubahan yang membuktikan bahwa website rentan XSS.

c. *Deniel of Servcice* (DoS)

Deniel of Service adalah sebuah teknik penyerangan terhadap sebuah sistem dengan jalan menghabiskan sumber daya sistem tersebut sehingga tidak bias di akses lagi. Peneliti menggunakan *Ping of Deatch* untuk melakukan DoS pada *www.binadarma.ac.id*.



Gambar 3.3 : *Ping of Death*

Pada gambar 3.3 peneliti menggunakan perintah : Ping www.binadarma.ac.id –l 39999 –n 10000000 –w 0.00001. Akan tetapi pengiriman paket gagal dikirimkan dikarenakan adanya *firewall* pada jaringan *web server www.binadarma.ac.id*. Maka *web server www.binadarma.ac.id* tidak rentan untuk serangan DoS terutama *Ping of Death*.

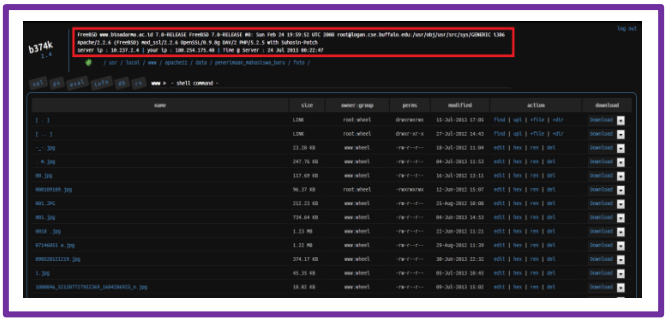
d. *Missing Function Level Access Control*

Hampir semua aplikasi web memverifikasi fungsi tingkat hak akses sebelum membuat fungsi yang terllihat di UI (*User Interface*). Namun, aplikasi perlu ditampilkan untuk memeriksa control akses yang sama pada *server* ketika setiap fungsi di akses. Jika permintaan tidak diverifikasi, seorang *attacker* akan dapat melakukan permintaan mengakses fungsi yang tidak sah. Peneliti akan mencoba beberapa fungsi *upload* pada *www.binadarma.ac.id*.

Peneliti akan mencoba fasilitas *upload* lainnya yang tersedia pada halaman penerimaan mahsiswa baru dengan alamat *http://binadarma.ac.id/penerimaan_mahasiswa_baru/pmbs1.php* dengan tampilan seperti gambar 3.4.

Gambar 3.4 : fasilitas *upload* pada halaman PMB

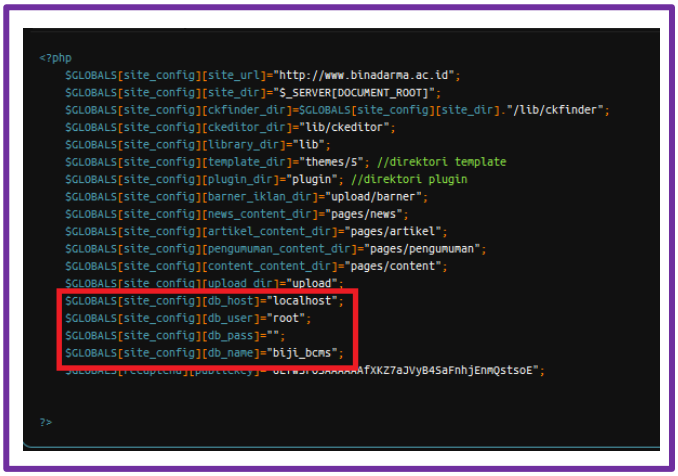
Dengan fasilitas tersebut peneliti mencoba *upload* file php shell (*backdoor*) dengan ekstensi .php. Dan berhasil di upload lalu file yang di *upload* akan terdapat pada alamat http://binadarma.ac.id/penerimaan_mahasiswa_baru/foto/ namafile.php. Tampilan shell php (*backdoor*) yang peneliti gunakan setelah berhasil di *upload* adalah sebagai berikut :



Gambar 3.5 : shell php yang berhasil di *upload*

Dengan berhasilnya shell php (backdoor) di upload akan banyak ancaman yang diakibatkan. Berikut ancaman menurut peneliti :

1. Bisa menghapus file-file penting
2. Bisa mendapatkan hak akses root (*rooting)*
3. Bisa melakukan *Jumping* ke server lain
4. Mendapatkan *username* dan *password* administrator dan akun-akun lainnya
5. Bisa akses ke *database web server www.binadarma.ac.id*
6. Melakukan *defacement*, dan lain-lain.



Gambar 3.6 : file config.php

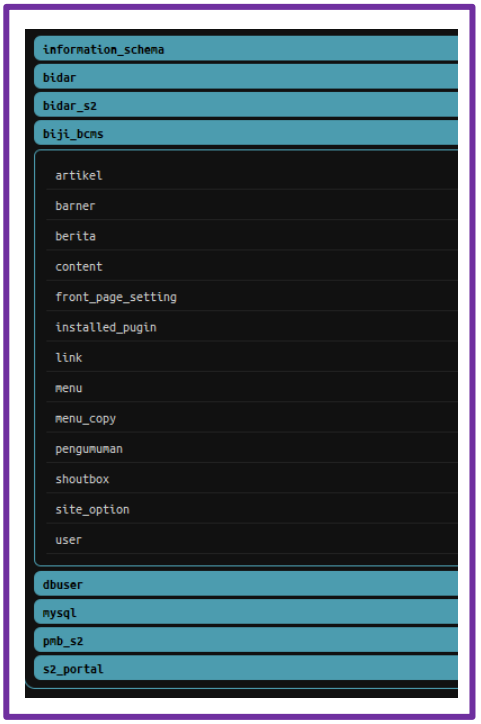
Kesalahan lain yang dilakukan adalah masih menggunakan *password default* dari konfigurasi saat menginstal program *database*. Kotak merah pada gambar merupakan informasi dari file config.php yang beralamat http://www.binadarma.ac.id/config.php. dengan db_host = localhost, db_user = root, db_pass = tidak di lindungi dengan password, db_name = biji_bcms. Dengan informasi seperti itu seorang *attacker* akan sangat mudah untuk melakukan akses ke *database* biji_bcms dengan menggunakan *backdoor* yang terbuat dari bahasa pemograman PHP yang di *coding* sedemikian rupa dan memiliki fasilitas yang hampir sama seperti CPANEL pada suatu *web hosting*. Dengan adanya *backdoor* ini *attacker* dengan sangat mudah mengakses *database* dari *web*

server dan dapat melakukan kegiatan yang merugikan lainnya. Dampak yang dilakukan *attacker* dapat dilihat pada halaman 60.

Gambar 3.7 : peneliti mencoba *connect* ke *database*

Dari hasil melihat *file* config.php pada gambar 4.27, peneliti mencoba memasuki *database web server www.binadarma.ac.id*. Dengan *Host* adalah *localhost*, *username* adalah root, akan tetapi untuk *password* administrator tidak memberikan *password*. Dengan demikian *password default* ada **“”** atau dibiarkan kosong. *Port* tidak perlu di isi karena *backdoor* peneliti gunakan sudah cukup pintar karena sudah automatis membaca dari konfigurasi *server* port-port yang digunakan.

Setelah memberikan kebutuhan untuk melakukan koneksi ke *database* peneliti menekan tombol *Connect* !. Maka hasilnya seperti gambar 4.29. Peneliti bisa melakukan akses ke *database* biji_bcms dan dapat melihat *database* lainnya yang tersimpan pada *web server www.binadarma.ac.id*.



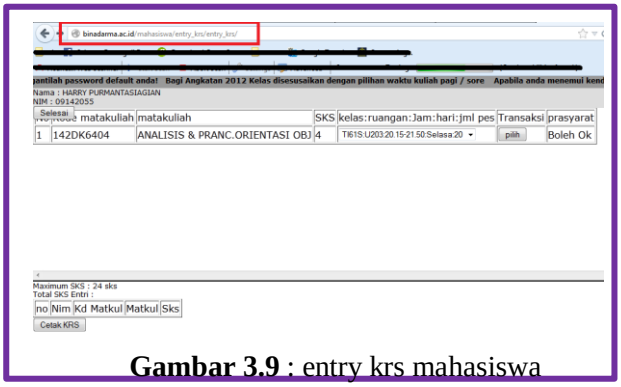
Gambar 3.8 : daftar *database* setelah *connect*

Dengan akses ini seorang *attacker* dapat mendapat semua *username* dan *password* yang tersimpan pada *database* yang tertera pada gambar 3.8.

e. *Sensitive Data Exposure*

Banyak aplikasi web tidak benar dalam melindungi data yang sensitive, seperti kartu kredi, id pajak dan pembuktian surat-surat berharga atau mandate. Penyerang dapat mencuri atau memodifikasi data yang lemah dilindungi tersebut atau melakukan pencurian identitas, penipuan kartu kredit, atau kejahatan lainnya. Data sensitive layak mendapatkan perlindungan ekstra seperti enkripsi.

Seperti halnya peneliti temukan pada bagian *Insecure Direct Reference Object* bahwa terdapat beberapa informasi yang keluar tanpa *authentication* terlebih dahulu dan memeriksa hak akses dari *user*.



Gambar 3.9 : entry krs mahasiswa

Gambar 4.31 memberitahukan bahwa seorang mahasiswa dapat memasuki halaman entry krs meskipun bukan pada saat waktu entri. Untuk membuka halaman ini *attacker* harus melakukan login terlebih dahulu dengan memasukkan NIM dan *password* mahasiswa. Dengan terbukanya halaman ini dapat membuat mahasiswa yang sudah melakukan entri lebih dulu bisa diganti oleh *attacker* dan mengacaukan jadwal kuliah seorang mahasiswa. Kerugian yang diakibatkan olleh kelemahan ini dapat membuat jadwal kulian seorang mahasiswa menjadi kacau dan bisa juga membuat mahasiswa tersebut mendapatkan kerugian materi.

Resiko : dengan adanya kerentanan ini dapat mengganggu aktifitas seorang mahasiswa bahkan bisa jadi semua mahasiswa. Dan secara otomatis mahasiswa akan komplain ke Universitas Bina Darma.

f. Insecure Direct Reference Object

Insecure Direct Reference Object terjadi ketika pengembang mengekspos referensi ke suatu objek dalam implementasi. Seperti file, petunjuk, atau tombol *database* tanpa pemeriksaan kontrol akse s atau perlindungan lainnya. Dengan begitu attacker dapat memanipulasi referensi-referensi ini untuk mengakses data yang tidak sah. Dalam bentuk yang paling sederhana kerentanan ini dapat dimanfaatkan dengan memodifikasi string URL dengan sesuatu seperti ini :

```
http://vunerableSite.com/cms/accountI
nfo?LoggedIn=True&userID=45674
```

Your Account

```
http://vunerableSite.com/cms/accountI
nfo?LoggedIn=True&userID=45675
```

Not Your Account

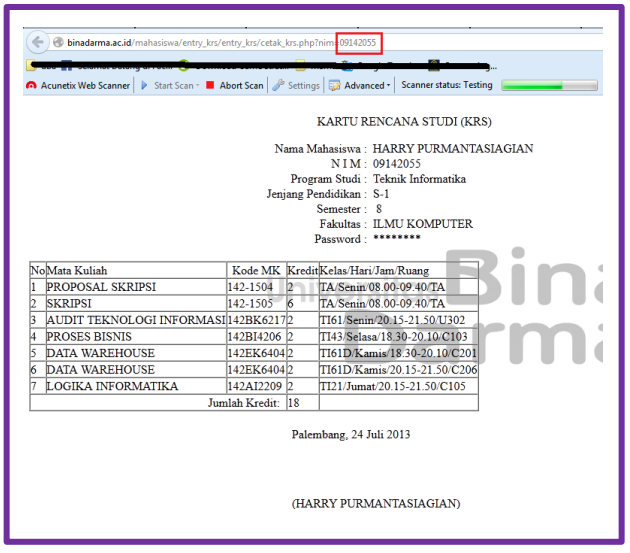
Peneliti akan melakukan hal yang sama pada *www.binadarma.ac.id* pada URL sebagai berikut :

```
http://www.binadarma.ac.id/mahasiswa/
entry_krs/entry_krs/cetak_krs.php?nim
=09142055
```

akun peneliti

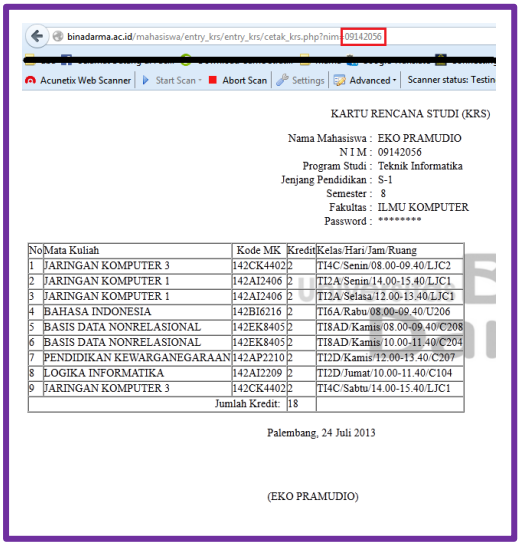
```
http://www.binadarma.ac.id/mahasiswa/
entry_krs/entry_krs/cetak_krs.php?nim
=09142056
```

akun orang lain



Gambar 3.10 : halaman KRS mahasiswa

Pada gambar 4.36 Merupakan halaman KRS mahasiswa setelah mahasiswa melakukan entry KRS. Pada address bar terdapat alamat yang memiliki kerentanan untuk dilakukannya teknik Insecure Direct Reference Object. Kotak merah pada gambar merupakan NIM dari peneliti sendiri dan peneliti akan mencoba mengganti angka atau NIM tersebut.



Gambar 3.11 : halaman user yang berbeda

Setelah peneliti mengganti angka 09142055 menjadi 09142056 halam tersebut berubah menjadi halam KRS milik Eko Pramudio yang memiliki NIM 09142056. Dengan situasi seperti ini, maka identitas atau informasi mahasiswa lain dapat diketahui oleh *attacker*. Akan tetapi *administrator* sudah mengantisipasi kejadian seperti ini dengan melakukan sensor terhadap *password user*.

1. Resiko : Bocornya informasi yang seharusnya bukan hak seorang *attacker* untuk mendapatkan informasi tersebut.
2. Solusi : Melakukan tindakan *Level access Protocol*, dengan melakukan tindakan ini. Maka siapapun yang bukan haknya tidak akan bisa membuka informasi yang memang buka termasuk dalam hak akses user.

3.1 Melakukan Evaluasi (*Evaluating*)

3.1.1 Evaluasi Hasil Pembahasan

Pada pembahasan peneliti melakukan *Vulnerability Assesment* dengan menggunakan aplikasi dan teknik-teknik yang berbeda-beda. Akan tetapi pada beberapa teknik masih bisa dikembangkan menjadi sub teknik *hacking* yang memungkinkan *attacker* untuk menemukan pintu masuk ke dalam *web server*. Namun pada teknik *Missing Function Level Access Control* peneliti menemukan pintu masuk ke *web server* dengan menginjeksi atau melakukan *upload* dari aplikasi *website www.binadarma.ac.id* berupa *backdoor* yang berbentuk shell PHP. Dengan *backdoor* ini seorang *attacker* dapat melakukan hal yang sangat merugikan untuk semua kalangan Universitas Bina Darma. Sepert :

1. *Rooting* (mengambil akses ROOT pada *web server*)
2. *Jumping* (melompat dari server satu ke server lain)
3. Mendapatkan *username* dan *password* mulai dari internet dan intranet
4. Mendapatkan file yang tidak seharusnya di publikasikan
5. Melakukan *defacement*
6. Menghapus *database* dari *web server www.binadarma.ac.id*, dan lain-lain.

Dengan adanya kerentanan ini peneliti melaporkan langsung ke ketua UPT-SIM pak M. Akbar, S.T,. M.IT. Kerentanan ini juga dapat

dikembangkan kedalam kategori ancaman *Using Components with Known Vulnerabilities*. Peneliti belum mencoba ancaman *Using Components with Known Vulnerabilities* dikarenakan waktu yang tidak memungkinkan.

Setelah melakukan *scanning* menggunakan *tools* Acunetix, Nikto, Retina, Nessus, dan OpenVAS. Peneliti mendapatkan *report* bahwa ada beberapa aplikasi di dalam *web server www.binadarma.ac.id* yang sudah tenggat waktu atau *outdated*. Dengan begitu, akan sangat mudah *attacker* menemukan celah keamanan terhadap *www.binadarma.ac.id*. Aplikasi tersebut adalah Apache, Mod_SSL, dan Open_SSL yang peneliti sarankan untuk melakukan *updating* agar *web server www.binadarma.ac.id* lebih aman.

3.2 Pembelajaran (*Learning*)

3.2.1 Dokumentasi dan Pelaporan (*Documentation and Reporting*)

Tabel 3.1 : Dokumentasi dan Laporan

No	JENIS SERANGAN	TOOLS	KETERANGAN	STATUS	KONEKSI
1	SQL Injection	Sqlmap	Melakukan injeksi ke halaman korban dengan metode POST dan GET	TIDAK BERHASIL	TIDAK LOGIN
2	Cross-Site Scripting (XSS)		Melakukan injeksi XSS ke address dan form search pada target	TIDAK BERHASIL	TIDAK LOGIN
3	Deniel of Service (DoS)	Ping of Deatch (PoD)	Mengirim paket deauthentication secara terus menerus ke target dengan jumlah paket 1000000	TIDAK BERHASIL	TIDAK LOGIN
4	Missing Function Level Access Control	PHP Shell	Memanfaatkan kelamahan fungsi aplikasi yang tidak di *coding* dengan benar	BERHASIL	MASUK KEDALAM DATABASE
5	Phising		Melakukan tindakan penipuan berupa halaman login palsu	BERHASIL	TIDAK LOGIN / INFORMASI TIDAK TERSIMPAN DI LOG
6	Sensitive Data Exposure		Memanfaatkan data atau informasi yang tidak diproteksi dengan ekstra	BERHASIL	
7	Insecure Direct Object Referencs		Melakukan tindakan terhadap alamat / URL yang memungkinkan attacker mendapatkan informasi yang tidak seharusnya di akses	BERHASIL	TIDAK LOGIN

Dokumentasi dan laporan ini merupakan hasil dari analisa dan pembahasan peneliti.

Solusi dari beberapa teknik yang peneliti terapkan pada penelitian diatas yaitu :

1. Untuk teknik *Missing Function Level Access Control* adalah dengan memeriksa kembali jika ingin mengimplementasikan aplikasi baru dan memerika kembali juga *source code* yang digunakan apakah sudah aman atau belum.
2. Untuk teknik *Phising,* sebaiknya dilakukan sosialisasi ke semua pengguna

halaman login agar lebih berhati-hati jika melihat halaman login Universitas Bina Darma.

3. *Sensistive Data Exposure,* menutup hak akses pengguna agar tidak dapat melihat data-data yang tidak semestinya dilihat pengguna tersebut.
4. *Insecure Direct Object References,* memodifikasi hak akses user dan menentukan kapan aplikasi tersebut bisa dibuka dan oleh siapa saja boleh diakses. Disarankan juga untuk melakukan enkripsi terhadap objek pada URL.
5. Menggunakan *session identifier,* dimana maksudnya dalam tiap aplikasi web, sebuah *logical session* harus ditetapkan antara browser dan *web server.* Untuk tujuan itu, suatu string atau sebuah angka dari sekumpulan data yang disebut *session identifier* perlu dikirimkan dan dikembalikan antara *web server* dan *browser.* Lebih baik, tiap *user session* pada aplikasi harus diidentifikasi oleh sebuah *session identifier* yang unik dan tidak dapat digunakan kembali dari satu *session* ke *session* yang lain. Sekalipun *user* yang sam *logon* kembali, suatu *session identifier* yang baru harus tetap dibuat.
6. Mengupdate aplikasi Apache, mod_ssl dan open_ssl agar celah yang terbuka tertutup.
7. Peneliti tidak melakukan VA pada jaringan *web server www.binadarma.ac.id*.
8. Peneliti tidak mencoba masuk ke akses intranet Universitas Bina Darma.
9. Pengujian DoS belum maksimal dikarenakan hanya menggunakan PoD (*Ping of Deathi*), peneliti menyarankan untuk melakukan DDoS.

4 SIMPULAN

Berdasarkan hasil penelitian dan pembahasan yang telah diuraikan pada bab sebelumnya, dalam penelitian yang berjudul *Vulnerability Assesment* pada *web server* www.binadarma.ac.id, maka dapat disimpulkan :

1. Teknik pengujian untuk *web server* masih begitu banyak yang belum dicobakan pada penelitian ini dan itu berarti belum semua celah pada *web server www.binadarma.ac.id.*
2. Keamanan pada *web server www.binadarma.ac.id* merupakan keamanan dan aplikasi yang termasuk *self building.* Oleh karena itu, keamanan *web server www.binadarma.ac.id* sangat bergantung pada staff dan karyawan.
3. Pengujian kerentanan pada *web server www.binadarma.ac.id* sangat diperlukan untuk menguji celah keamanan sistem. Agar memberikan kenyamanan dan keamanan dalam penggunaan fasilitas *website www.binadarma.ac.id.*
4. Hasil penelitian ini memberikan kontribusi saran perbaikan celah keamanan pada sistem *web server www.binadarma.ac.id*

Vulnrability Assessment pada *web server www.binadarma.ac.id* yang salah satunya disebabkan oleh terbatasnya waktu untuk penyelesaian penelitian. Untuk itu peneliti menyarankan sebagai berikut :

V. DAFTAR PUSTAKA

Ali, Shakel., Heriyanto, Tedy. (2011). *Backtrack 4: Assuring Security by Penetration Testing*. Birmingham-Mumbai: PACK Publishing Open Source.

Direktorat Keamanan Informasi, Direktorat Jendral Aplikasi Informatika, Kementerian Komunikasi dan Informatika Republik Indonesia. (2011). E-Book: *Panduan Keamanan Web Server.*

Digdo, Girindro Pringgo. (2012), *Analisis Serangan dan Keamanan Pada Aplikasi Web*, PT Elex Media Komputindo: Jakarta

GOV-CSIRT.(2012). *Methodology Vulnerability Assessment.* Diakses pada 12 april 2013, <http://govcsirt.kominfo.go.id/254/>.

IBISA.(2011). *Keamanan Sistem Informasi*.Yogyakarta: Andi

Kusmandani, Syaful. (2013). *Kisah Hacker Pembobol Situs Presiden SBY*. Diakses pada 12 april 2013, <http://inet.detik.com/read/2013/01/29/130724/2155140/323/kisah-hacker-pembobol-situs-presiden-sby>.

he Ten Most Critical Web Application Security Risk. E-book.Edisi ke-2.

Palembang Hacker Link, (2010), SQL *Injection Step by Step*, http://palembanghackerlink.com/thread-994.phl di akses pada 19 juli 2013.

Priandoyo, Anjar. (2006). *Vulnerablity Assessment untuk Meningkatkan Kesadaran Pentingnya Keamanan Informasi*. Jurnal Sistem Informasi. Vol. 1, No. 2, pp.73-83.

Rohman, Abdul. (2010). *Membangun DNS Server Dan Web Server Dengan Debian Linux*.ICT SMK Muhammadiyah 5 babat.

Santoso, Hanif. *Analisis Vulnerability Aplikasi iFace IT Telkom bandung*. paper UAS CS4633 Keamanan Sistem. Bandung.

Satoto, Kodrat Iman. (2009). *Analisis Keamanan Sistem Informasi Akademik Berbasis Web Di Fakultas Teknik Universitas Dipenogoro*. Artikel Ilmiah. Yogyakarta: Universitas Dipenogoro.