BAB 2
LANDASAN TEORI

2.1 Aspek dari Keamanan

Garfinkel mengemukakan bahwa keamanan komputer (*computer security*) melingkupi empat aspek, yaitu *privacy, integrity, authentication, dan availability*. Selain keempat hal di atas, masih ada dua aspek lain yang juga sering dibahas dalam kaitannya dengan *electronic commerce,* yaitu *access control* dan *nonrepudiation*[6].

1. Privacy / Confidentiality

Inti utama aspek *privacy* atau *confidentiality* adalah usaha untuk menjaga informasi dari orang yang tidak berhak mengakses. *Privacy* lebih kearah data-data yang sifatnya *privat* sedangkan *confidentiality* biasanya berhubungan dengan data yang diberikan ke pihak lain untuk keperluan tertentu dan hanya diperbolehkan untuk keperluan tertentu tersebut.

2. Integrity

Aspek ini menekankan bahwa informasi tidak boleh diubah tanpa seijin pemilik informasi. Adanya *virus, trojan horse,* atau pemakai lain yang mengubah informasi tanpa ijin merupakan contoh masalah yang harus dihadapi.

3. Authentication

Aspek ini berhubungan dengan metoda untuk menyatakan bahwa informasi betul-betul asli, atau orang yang mengakses atau memberikan informasi adalah betul-betul orang yang dimaksud.

4. Availability

Aspek *availability* atau ketersediaan berhubungan dengan ketersediaan informasi ketika dibutuhkan. Sistem informasi yang diserang atau dijebol dapat menghambat atau meniadakan akses ke informasi.

5. Access Control

Aspek ini berhubungan dengan cara pengaturan akses kepada informasi. Hal ini biasanya berhubungan dengan masalah *authentication* dan juga *privacy*. *Access control* seringkali dilakukan dengan menggunakan kombinasi *userid/password* atau dengan menggunakan mekanisme lain.

6. Non-repudiation

Aspek ini menjaga agar seseorang tidak dapat menyangkal telah melakukan sebuah transaksi.

2.2 Vulnerability

Vulnerability adalah sebagai sebuah kecacatan dalam prosedur keamanan, perangkat lunak, sistem pengendalian *internal*, atau penerapan sistem yang mempengaruhi *integrity*, *confidentiality*, *accountability*, atau *availability* data atau layanan. *Vulnerability* termasuk kekurangan yang mungkin sengaja dimanfaatkan dan dapat menyebabkan kegagalan karena tingkah manusia secara tidak sengaja ataupun karena bencana alam [7].

Jenis-Jenis *Vulnerability*

Jenis-jenis *vulnerability* menurut Lehtinen, Russell, dan Gangemi (2006) yaitu :

1. *Physical Vulnerabilities*

Kerentanan terhadap jaringan komputer Anda, dimana terdapat orang yang tidak terautorisasi mencoba untuk masuk ke dalam *server* jaringan dan menyabotase peralatan jaringan, kemudian mencuri data *back up*, *printouts* ataupun informasi penting yang memungkinkan mereka untuk lebih mudah masuk ke *server* dilain waktu.

2. *Natural Vulnerabilities*

Kerentanan terhadap komputer atau jaringan yang disebabkan oleh bencana alam dan ancaman lingkungan sepeti kebakaran, banjir, gempa bumi, petir, kehilangan daya yang dapat merusak data dan komputer. Debu, kelembapan, dan kondisi suhu yang tidak merata juga dapat menyebabkan kerusakan.

3. *Hardware and Software Vulnerabilities*

Kerentanan pada sebuah jaringan atau komputer diakibatkan karena *hardware failure* yang menyebabkan mudahnya bagi orang-orang yang tidak terautorisasi untuk membuka lubang keamanan (*security hole*). Sedangkan *software failure* dapat mengakibatkan sistem menjadi gagal.

4. *Media Vulnerabilities*

Kerentanan dapat terjadi pada sebuah media *back up* seperti kemasan *disk, tape, cartridge, chip* memori *printout* karena dapat dicuri atau rusak karena debu atau medan magnet.

5. *Emanation Vulnerabilities*

Kerentanan dapat terjadi pada sebuah peralatan elektronik yang memancarkan radiasi elektronik dan elektromagnetik, dikarenakan adanya penyadap elektronik dapat mencegat sinyal yang berasal dari komputer, jaringan dan sistem nirkabel yang mengakibatkan informasi yang disimpan dan ditransmisikan menjadi rentan.

6. *Communication Vulnerabilities.*

Kerentanan dapat timbul apabila komputer Anda terhubung dengan jaringan atau dapat diakses melalui modem atau *internet*, yang mengakibatkan orang yang tidak berhak mengakses dapat menembus sistem anda.

7. *Human Vulnerabilities*

Kerentanan terbesar yang mungkin timbul adalah dikarenakan orang-orang yang mengelola dan menggunakan sistem (*administrator*).

8. *Exploiting Vulnerabilities*

Kerentanan dapat dieksploitasi dengan berbagai cara, salah satunya seperti menggunakan logging karena logging merupakan sistem yang tidak terproteksi oleh password dan memiliki kontrol yang minimal.

2.3 Keamanan Komputer

Menurut G. J. Simons, keamanan komputer adalah bagaimana kita dapat mencegah penipuan atau paling tidak adanya penipuan disebuah sistem yang berbasis informasi, dimana informasinya sendiri tidak memiliki arti fisik.

Sedangkan menurut Jhon D. Howard :

Computer Security is preventing attackers from achieving objectives through unauthorized access or unauthorized use of computers and networks. (John D. Howard, "An Analysis Of Security Incidents On The Internet 1989 - 1995").

Dari kedua kesimpulan diatas bisa diambil kesimpulan bawah keamanan komputer adalah tindakan pencegahan dari serangan pengguna komputer atau pengakses jaringan yang tidak bertanggung jawab. Keamanan komputer berhubungan dengan pencegahan dini dan deteksi terhadapa tindakan pengganggu yang tidak dikenali dalam sistem komputer.

2.3.1 Security Assessment

Security assessment merupakan penilaian terhadap keamanan infrastruktur yang ada dalam satu organisasi atau perusahaan. *Assessment* ini merupakan suatu penilaian seberapa besar suatu organisasi atau perusahaan menerapkan prinsip keamanan komputer terhadap seluruh perangkat dan jaringan yang dalam lingkup mereka masing-masing. Penilaian terhadap keamanan infrastruktur dalam *security assessment* berupa *vulnerability assessment* dan *penetration test*[8].

2.3.2 Vulnarability Assessment

Vulnerability assessment adalah proses pencarian kelemahan yang ada pada target. *Attacker* melakukan identifikasi apa saja celah yang ada dalam sistem, lalu memberikan informasi berapa banyak kelemahan atau celah yang ada pada sistem tersebut, lalu memberikan urutan prioritas sesuai dengan besar atau kecilnya resiko yang timbul dengan adanya celah keamanan tersebut.

2.3.3 Penetration Testing

Penetration test adalah proses melakukan eksplorasi dan eksploitasi lebih jauh ke dalam sistem target setelah megetahui *bug* atau *security hole* yang di *report* pada proses *vulnerability assessment*. Pada proses ini aktivitas *gaining access* atau proses exploitasi coba dilakukan oleh *attacker*. *Attacker* akan mencoba mengeksploitasi celah keamanan sehingga berhasil mendapatkan akses ke dalam sistem target.

Ada 3 tipe *penetration test* berdasarkan lokasi dilakukannya yaitu *white box, black box* dan *grey box.*

2.3.3.1 White Box

Tipe *Penetration testing* ini diperlukan jika organisasi atau instansi perlu untuk menilai keamanan terhadap jenis serangan tertentu atau target tertentu, informasi lengkap tentang struktur jaringan dapat diberikan kepada penguji penetrasi. Informasi yang diberikan dapat mencakup dokumen, *topologi* jaringan,aset perusahaan atau inventaris aset, dan informasi penilaian bersifat rahasia lainnya.

2.3.3.2 Black Box

Tipe *Penetration testing* yang dilakukan tanpa mengetahui informasi-informasi sebelumnya yang berkaitan dengan sistem atau jaringan seperti sistem operasi yang digunakan oleh target, *topologi* jaringan, *open port* dan *service* yang sedang berjalan. *Pentester* harus terlebih dahulu menggali Informasi terkait dengan target sistem menggunakan metode-metode yang umum digunakan oleh *Attacker*.

2.3.3.3 Grey Box

Merupakan kombinasi *white box* dan *black box testing* dimana *pentester* diberikan sebagian Informasi terkait target sistem

2.4 Backdoor Shell

Backdoor atau pintu belakang dalam keamanan sistem computer merujuk kepada mekanisme yang dapat digunakan untuk mengakses sistem,aplikasi atau jaringan selain dari mekanisme yang umum digunakan atau yang lebih dikenal melalui proses autentikasi.

Backdoor pada awalnya dibuat oleh programmer computer sebagai mekanisme yang mengizinkan mereka untuk memperoleh akses khusus kedalam program mereka, seringnya digunakan untuk membenarkan dan memperbaiki kode di dalam program yang mereka buat ketika sebuah *crash* akibat *bug* yang terjadi. Salah satu pencetus backdoor yaitu Kenneth Thompson yang merupakan programmer sistem operasi unix yang membuat proses login di tahun 1983.

Backdoor yang diciptakan untuk melindungi dirinya dari pendeteksian dan pembuangan dari sistem.

Istilah *backdoor* sendiri saat ini digunakan oleh peretas untuk merujuk kepada mekanisme yang mengizinkan seorang peretas sistem dapat mengakses kembali sebuah sistem yang telah diserang. Dimana saat menyisipkan *backdoor* didalam sistem tersebut sang peretas melakukan beberapa tahapan seperti [12]

1. *Reconnaissance*

   Proses mengumpulkan data sebanyak-banyaknya terhadap target atau sasaran.
2. *Scanning*

   Proses pencarian kelemahan terhadap target. Pada tahap ini menggunakan tools. Dimana pencarian kelemahan berdasarkan data-data yang telah dikumpulkan sebelumnya.
3. *Gaining access*

   Pada tahap ini peretas melakukan penetrasi kedalam sistem target, setelah mendapat info kelemahan pada setelah proses scanning.
4. *Maintaining access*

   Setelah mendapat akses kedalam sistem target biasanya peretas ingin tetap menguasai sistem tersebut, maka pada tahap ini peretas menanam backdoor shell dengan tujuan untuk mempertahankan kontrolnya pada sistem tersebut
5. *Covering tracks*

   Merupakan tahapan untuk menghilangkan/menghapus jejak aktifitas dalam sistem sehingga korban sulit untuk mengetahui keberadaan ataupun untuk melacak pelaku.

Backdoor shell merupakan sejenis aplikasi yang memiliki *interface* yang digunakan oleh orang yang tidak memiliki akses pada sistem(*malicious user*) dapat masuk kedalam sistem (*bypass*) pada control keamanan pada sebuah *webserver*[13]. *Backdoor shell* disisipkan oleh peretas pada tahapan *maintaining access*. Dengan demikian peretas tidak perlu melakukan kembali 5 tahapan diatas

untuk kembali masuk kedalam sistem jika suatu saat admin target melakukan *patching* sistem.

Backdoor shell terdapat dalam beberapa format bahasa pemograman seperti php, asp, jsp, vb, aspx, pl, py dan bsh. Dalam penggunaannya backdoor shell cukup dieksekusi dengan alamat website target ditambah directori lokasi file disimpan seperti file backdoor c99.php disimpan pada directori "/tools/set" maka untuk mengeksekusi backdoor shell tersebut dengan mengunjungi url http://www.webtarget.com/tools/set/c99.php dengan demikian sudah dapat mengoperasikan fitur-fitur yang terdapat pada backdoor yang telah ditanam.

Berdasarkan identifikasi terhadap backdoor shell dikategorikan menjadi 2 jenis yaitu[9] :

2.4.1 Non-encrypted backdoor shell

Merupakan jenis backdoor shell yang source codenya tidak di encode sehingga kita dapat melihat isi dari source code yang terdiri dari fungsi-fungsi didalamnya. Berikut beberapa fungsi PHP yang umum digunakan pada *backdoor shell* [4]:

Tabel 2. 1 Fungsi pada backdoor shell

Fungsi	Tujuan
exec	merupakan sebuah fungsi untuk menjalankan program eksternal.
system	merupakan sebuah fungsi untuk menjalankan program eksternal serta menampilkan output.
shell_exec	merupakan sebuah fungsi yang mengeksekusi perintah melalui shell dan dikembalikan dengan output lengkap sebagai string.
passthru	merupakan sebuah fungsi untuk mengeksekusi program eksternal dan menampilkan raw ouput (keluaran baku).

Fungsi	Tujuan
eval	merupakan sebuah fungsi yang memungkinkan bahasa permograman tersebut untuk melakukan penilaian string.
proc_open dan popen	merupakan fungsi untuk mengeksekusi perintah ada cmd dan membuka fille pointer untuk input dan output.
base64_decode	Merupakan fungsi untuk melakukan decode data yang sebelumnya telah di encode dengan MIME base64
Curl_exec	Merupakan fungsi yang ditugaskan untuk mengeksekusi sebuah halaman / sebuah url.
Parse_ini_file	Merupakan fungsi yang bertujuan untuk mengambil dan mengurai konfigurasi sebuah file menjadi bentuk data array.

Fungsi diatas merupakan fungsi yang umum digunakan pada backdoor shell dimana fungsi tersebut untuk mengeksekusi perintah pada Command Line web server dalam memberi perintah tertentu.

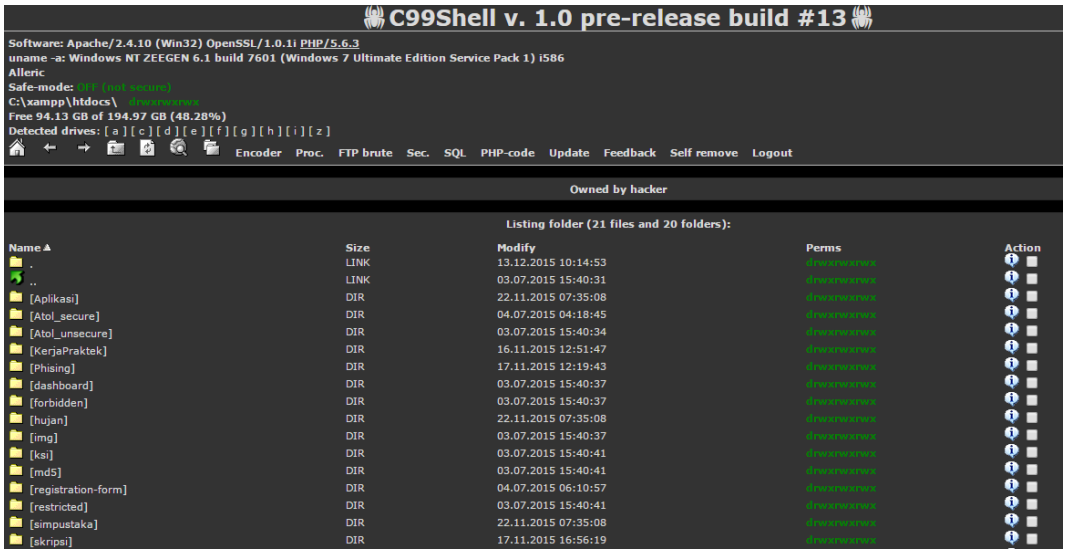
Berikut kutipan fungsi pada backdoor shell C99.php :

```
function myshellexec($cmd){
 global $disablefunc;
 $result = "";
 if (!empty($cmd)){
  if (is_callable("exec") and !in_array("exec",$disablefunc))
{exec($cmd,$result); $result = join("\n",$result);}
  elseif (($result = `$cmd`) !== false) {}
  elseif (is_callable("system") and
!in_array("system",$disablefunc)) {$v = @ob_get_contents();
@ob_clean(); system($cmd); $result = @ob_get_contents();
@ob_clean(); echo $v;}
  elseif (is_callable("passthru") and
```

Gambar 2. 1 Kutipan Code non-encrypted backdoor shell

Contoh backdoor Shell non-encrypted adalah C99.php

C99 merupakan jenis backdoor shell yang dikembangkan oleh CCteam rusia C99 memiliki beberapa fitur yang menarik berikut penjelasan fitur dan interface dari C99



Gambar 2. 2 Interface C99

Pada bagian header di interface C99 menyediakan informasi yang berkaitan dengan webserver target seperti ip server, alamat server, versi php, jenis database yang digunakan dan yang lainnya. Berikut penjelasan dari fitur yang dimiliki C99 :

- *List directory*

  Merupakan fitur yang dimiliki backdoor shell dimana menampilkan infomasi lokasi file pada webserver dan juga menyediakan eksekusi terhadap file yang terdapat pada directori seperti upload, edit, hapus dan download.

- *Shell*

  Shell merupakan fitur pada *backdoor* untuk melakukan eksekusi terhadap *command-command* linux diperuntukan untuk peretas yang ingin lebih menggali informasi yang terdapat dalam *webserver* target.

- *Encode*

  Merupakan fitur untuk melakukan *encoding* pada inputan yang diingin kan oleh pengguna dimana pada C99 hasil dapat melakukan *encode* dengan jenis keluaran MD5, Sha1, crc32, base64_encode dan dec2hex.

- FTP *brute force*

  Merupakan fitur untuk melakukan *cracking password* yang terdapat pada ftp dengan menggunakan metode *bruteforce*. dengan pencocokan dengan *wordlist* yang ada.
- SQL

  Merupakan fitur untuk melakukan pengelolaan data pada database webserver target.
- Self remove

  Merupakan fitur untuk menghapus file backdoor pada webserver. Fitur ini biasanya digunakan untuk menghilangkan jejak apabila peretas dalam kondisi terancam / memiliki kemungkinan dilacak oleh orang lain.

2.4.2 *Encrypted Backdoor Shell*

Merupakan jenis *backdoor shell* yang *source code*nya di *encode* dengan tujuan menyamarkan file menjadi tulisan code yang tidak memiliki arti, sehingga tidak menimbulkan kecurigaan terhadap file tersebut, file yang tersebut dapat dieksekusi oleh system yang sebelumnya melakukan dekripsi terhadap *source code*. Ada beberapa fungsi encode yang populer digunakan oleh *attacker* seperti *base64_encode* dan *base64_decode*. Berikut contoh backdoor yang di*encode* :

```
(base64_decode("ZXJyb3JfcmVwb3J0aW5nKDApOwokcWF6cGxtPWhlYWRlcnNfc2Vu
dCgpOwppZiAoISRxYXpwbG0pewokcmVmZXJlcj0kX1NFUlZFUlsnSFRUUF9SRUZFUkVS
J107CiR1YWc9JF9TRVJWRVJbJ0hUVFBfVVNFUl9BR0VOVCddOwppZiAoJHVhZykgewpp
ZiAoIXN0cmlzdHIoJHVhZywiTVNJRSA3LjAiKSBhbmQgIXN0cmlzdHIoJHVhZywiTVNJ
RSA2LjAiKSl7CmlmIChzdHJpc3RyKCRyZWZlcmVyLCJ5YWhvbyIpIG9yIHN0cmlzdHIo
JHJlZmVyZXIsImJpbmciKSBvciBzdHJpc3RyKCRyZWZlcmVyLCJyYW1ibGVyIikgb3Ig
c3RyaXN0cigkcmVmZXJlciwibGl2ZS5jb20iKSBvciBwcmVnX21hdGNoKCIveWFuZGV4
XC5ydVwveWFuZHNlYXJjaFw/KC4qPylcJmxyXD0vIiwkcmVmZXJlcikgb3IgcHJlZ19t
YXRjaCAoIi9nb29nbGVcLiguKj8pXC91cmxcP3NhLyIsJHJlZmVyZXIpIG9yIHN0cmlz
dHIoJHJlZmVyZXIsImZhY2Vib29rLmNvbS9sIikgb3Igc3RyaXN0cigkcmVmZXJlciwi
YW9sLmNvbSIpKSB7CmlmICghc3RyaXN0cigkcmVmZXJlciwiY2FjaGUiKSBvciAhc3Ry
aXN0cigkcmVmZXJlciwiaW51cmwiKSl7CmhlYWRlcigiTG9jYXRpb246IGh0dHA6Ly9j
dmtsYS4yNXUuY29tLyIpOwpleGl0KCk7Cn0KfQp9Cn0KfQ==")); 
```

Gambar 2. 3 Kutipan code encrypted backdoor shell

Contoh encrypted backdoor shell adalah devshell.php.



Gambar 2. 4 Interface DevShell.php

Sama seperti backdoor shell lainnya DevShell juga menyediakan informasi pada header interface yang berkaitan dengan we bserver korban seperti ip server dan software server. Berikut penjelasan fitur yang dimiliki devshell.php

- *List directory*

  Merupakan fitur yang dimiliki backdoor shell dimana menampilkan infomasi lokasi file pada webserver dan juga menyediakan eksekusi terhadap file yang terdapat pada directori seperti upload, edit, hapus dan download.

- *Shell*

  Shell merupakan fitur pada *backdoor* untuk melakukan eksekusi terhadap *command-command* linux diperuntukan untuk peretas yang ingin lebih menggali informasi yang terdapat dalam *webserver* target.

2.5 Kriptografi

Kriptografi merupakan ilmu dan seni untuk menjaga pesan agar aman, *crypto* berarti rahasia dan *graphy* berarti tulisan. Para pelaku atau praktisi kriptografi disebut *cryptographers* dan sebuah algoritma kriptografik disebut cipher, merupakan persamaan matematik yang digunakan untuk proses enkripsi dan

dekripsi. Biasanya kedua persamaan matematik tersebut memiliki hubungan matematis yang erat. Ada beberapa teknik kriptografi dalam website yaitu :

2.5.1 Encoding

Tujuan *encoding* adalah untuk merubah data sehingga data tersebut dapat digunakan oleh system lainnya secara benar dan aman, misalkan dapat dikirim via e-mail ataupun ditampilkan di halaman website. Metode ini bukan untuk menjaga kerahasiaan informasi, akan tetapi lebih mengarah untuk memastikan bahwa data dapat digunakan secara benar.

```
S8td0s5NTk/x4kZNHp6Ud9tyjAfQIiOywcFTFay+jfLUSJ9EaK
```

Gambar 2. 5 Contoh karakter base64

Encode merubah data ke dalam format yang berbeda menggunakan skema umum, sehingga data tersebut dapat dengan mudah di *decode* kembali. Metode ini tidak membutuhkan kunci khusus untuk men*decode* data. Sehingga tidak membutuhkan alat bantu untuk membacanya. Karena sudah *include* kedalam PHP dan lansung dieksekusi dengan menggunakan fungsi base64_decode() dan base64_encode().

2.5.2 Enkripsi

Tujuan *encryption* adalah untuk merubah data agar dapat dijaga kerahasiaannya. Misalkan untuk mengirim sebuah pesan rahasia yang hanya orang-orang tertentu saja yang dapat membacanya. Metode ini lebih mengacu kepada *usability*(kemampuan untuk digunakan) akan tetapi lebih mengacu kepada kemampuan untuk memastikan data tersebut tidak dapat digunakan orang lain selain dari yang telah ditentukan.

Enkripsi merubah data ke dalam format lain dengan menggunakan skema(terbatas) atau bias dikatakan hanya orang-orang tertentu saja yang bias mereverse / membalikan data tersebut kedalam bentuk normal teks. Untuk membukanya dibutuhkan fungsi cipher text, algoritma tertentu dan sebuah kunci. Missal jika menggunakan ceasar cipher pada PHP maka kita harus membuat

decryptor nya karena PHP tidak mengenali enkripsi model ini. Biasanya digunakan untuk menyimpan pesan rahasia seperti database, session dan cookies.

2.5.3 Hash

Hash sering dianggap sama dengan metode encryption maupun encode, akan tetapi sebenarnya metode tersebut berbeda satu sama lain. Tujuan hash adalah untuk membuata/menambahkan inputan secara sembarang yang akan menghasilkan karakter yang memiliki panjang yang tetap dan memiliki attribute sebagai berikut :

1. input yang sama selalu menghasilkan output yang sama.
2. Beberapa input yang berbeda tidak harus menghasilkan output yang sama.
3. Metode satu jalan input-output.
4. Setiap modifikasi masukan yang diberikan akan menghasilkan perubahan drastis untuk *hash*.

Hash digunakan di dalam otentikasi untuk menghasilkan bukti kuat bahwa data sama sekali belum di ubah. Hal ini dilakukan dengan mengambil masukan yang diberikan, *hash,* dan kemudian mengenkripsi *hash* dikirim dengan kunci publik penerima.

Ketika penerima membuka pesan dengan kunci pribadi mereka, kemudian membandikan hash pesan yang diberikan oleh pengirim dengan hash mereka sendiri. Jika hash tersebut sama makan pesan yang dikirim dapat dipastikan tidak dimodifikasi. Contoh : SHA dan MD5

2.6 Static Code Analysis

Pada bagian ini akan menjelaskan tentang static code analysis tools dan teknik yang akan digunakan dalam penelitian pendeteksi backdoor shell pada webserver.

2.6.1 Pengenalan Static Code Analysis

Dalam penjelasan pada Microsoft SDL-10 Practice, statis code analisis adalah ketika source code dianalisis secara otomatis dalam status Statis yaitu non-running source code pada dasarnya compiler menggunakan static code analisis untuk menerjemahkan source code kedalam bahasa mesin yang dapat dibaca, dan teknik

ini diterapkan pada perangkat lunak khusus untuk menemukan kerentanan dalam system dan malware. Ada beberapa tools opensource yang dapat dijalankan sebagai static code analysis seperti pada linux yaitu perintah ‘Grep’ dapat menemukan malware. Contoh penggunaan grep dalam mencari fungsi eval

```
Grep –RPnl “eval” /var/www/
```

Gambar 2. 6 Contoh pencarian pada grep

2.6.2 Teknik Static Code Analysis

Pada bagian ini akan menjelaskan jenis-jenis teknik dalam static code analysis

2.6.2.1 Taint Analysis

Potentially vulnerable Functions(PVFs) atau yang biasa dikenal sebagai ‘*sinks’* diidentifikasi pada *source code* menggunakan *function matching*, jika terdapat fungsi yang berpotensi sebagai *backdoor shell* maka file tersebut akan ditandai sebagai file *tainted*. *Taint analysis* dapat digunakan pada jenis *backdoor shell* yang *non-encripted* karena dapat mengidentifikasi secara lansung *sinks* yang terdapat pada file tersebut. Berikut *sinks* yang berpotensi sebagai *backdoor shell* : *system*(), *exec*(), *shell_exec*() ,*passthru*() ,*eval*() ,*proc_open*() ,*popen*(), *base64_decode*(), *parse_ini_file*(), *curl_exec*() [4].

2.6.2.2 String Matching

String matching merupakan bagian dari *static code analysis* yang fokus mencari karakter yang berhubungan dengan *backdoor shell* yang di enkripsi. Yang pencariannya berdasarkan dari data *signature backdoor shell*. Jika sebuah file teridentifikasi memiliki data *signature backdoor shell* maka secara lansung akan di tandai sebagai file *backdoor*.

2.7 Internet

Pengertian internet secara umum adalah suatu media informasi komputer global yang dapat dikatakan sebagai teknologi tercanggih abad ini. Internet merupakan suatu media informas yang berjalan dalam sustu komputer. Akan tetapi,

tiodak semua komputer yang ada bisa berhubungan karena suatu komputer dapat dikatakan senbagai internet jika sudah terhubung dengan suatu jaringan.

Sekitar tahun 1963, RAND Coorporation, organisasi pengendali perang dingin Amerika Serikat menghadapi masalah yang sangat pelik yaitu bagaimana Amerika Serikat tetap berkomunikasi dengan lancar setelah perang nuklir terjadi. Pemecahannya adalah dengan membangun sebuah jaringan yang menghubungkan semua tempat strategis diseluruh Amerika dan tetap memberikan *Request For Proposal* (RFP) kepada UCLA (*University of California Los Angles*) mereka yang terlibat antara lain *Victor Crt Stefen Croackern*, Jon Postel dan Robert Braclen.

Pada tahun 1964 dikeluarkan proposal RAND yang inti nya adalah bahwa jaringan yang akan dibentuk tidak berpusat pada suatu dan tetap berfungsi sekalipun dalam keadaan hancur. Proposal RAND ini juga diilhami oleh munculnya "NET" diawal tahun 1990-an yaitu suatu teknologi dimana suatu pesan disampaikan dari tempat asalnya ketempat tujuan dengan memfragmentasikan pesan tersebut menjadi bagian-bagian kecil yang disebut paket tempat asalnya dan dirangkai kembali menjadi pesan semula ditempat tujuan. Teknologi ini dikenal sebagai Paket *Switching Network* [10].

Tahun 1969 empat buah IMP (*Interface Massage Processor*) dikirimkan ke empat perguruan tinggi yakni UCLA, SRI (*Standard Reseach Institute*), UCSB(*University of California Santa Barbara*) dan *University of Utah*. Jaringan ke empat tempat ini kemudian disebut sebagai ARPANET yang disponsori oleh DARPA (*Defence* of *Advance Reseach Project Agency*) pentagon, yang sampai sekarang ARPANET merupakan pusat dari jaringan internet[10].

2.8 Web

Menurut Yuhefizar "World Wide Web atau WWW atau juga dikenal dengan WEB adalah salah satu layanan yang didapat oleh pemakai komputer yang terhubung ke internet. Web ini menyediakan informasi bagi pemakai komputer yang terhubung ke internet dari sekedar informasi kecil atau informasi yang tidak berguna sama sekali sampai informasi yang serius, dari informasi yang gratisan sampai informasi yang komersial."

Menurut Wahana Komputer "*Web* adalah formulir komunikasi interaktif yang digunakan pada suatu jaringan komputer."

Adapun cara kerja *web* adalah sebagai berikut:

a. Informasi *web* disimpan dalam dokumen dalam bentuk halaman-halaman *web* atau *web page*.
b. Halaman *web* tersebut disimpan dalam *server web* komputer.
c. Sementara dipihak pemakai ada komputer yang bertindak sebagai komputer klien dimana ditempatkan program untuk membaca halaman web yang ada di *server web* (*browser*).
d. Browser membaca halaman *web* yang ada di *server web*.

2.8.1 Website

Menurut YM Kusuma Ardhana "*Website* adalah sejumlah halaman *web* yang memiliki topik saling terkait, terkadang disertai pula dengan berkas-berkas gambar, video, atau berkas lainnya."[10]

Penemu situs *web* adalah Sir Timothy Kohn Berners-Lee, sedangkan situs *web* yang tersambung dengan jaringan pertama kali muncul pada tahun 1991.

Website mempunyai fungsi yang bermacam-macam, tergantung dari tujuan dan jenis *website* yang dibangun, tetapi secara garis besar fungsi *website* menurut YM Kusuma Ardhana adalah sebagai berikut:[10]

a) Media Promosi
b) Media Pemasaran
c) Media Informasi
d) Media Pendidikan

2.8.2 Perkembangan Website

Perkembangan *web* berdasarkan teknologi dan cara penggunaannya menurut pada praktisi informatika adalah sebagai berikut:

a. *Web* 1.0

Web 1.0 merupakan teknologi awal dari sebuah *website*, teknologi ini masih statis dimana antara pembuat *website* dan penikmat *website* hanya tejadi

komunikasi 1 arah dimana pembuat sebagai pemberi informasi dan penikmat hanya sebagai pembaca. *Web* 1.0 dikembangkan untuk pengaksesan informasi dan memiliki sifat yang sedikit interaktif. Secara garis besar, sifat *Web* 1.0 adalah *Read-Only* dan bahasa yang digunakan pada *Web* 1.0 menggunakan bahasa HTML.

b. Web 2.0

Sifat Web 2.0 adalah *Read-Write*. *Web* 2.0 lebih menekankan pada perubahan cara berpikir dalam menyajikan konten dan tampilan di dalam sebuah *website*. Di era *Web* 2.0 sekarang, *web* digunakan untuk berbagi, pertemanan, kolaborasi menjadi sesuatu yang penting. *Web* 2.0 memungkinkan pengguna untuk melakukan lebih dari sekedar mengambil informasi. Mereka dapat membangun pada fasilitas interaktif.

Kunci perbedaan dalam *Web* 2.0 dan *Web* 1.0 adalah keterbatasan pada *Web* 1.0 yang mengharuskan pengguna internet untuk datang ke dalam *website* tersebut dan melihat satu persatu konten di dalamnya. Sedangkan *Web* 2.0 memungkinkan pengguna internet dapat melihat konten suatu website tanpa harus berkunjung ke alamat situs yang bersangkutan.Kemampuan *web* 2.0 dalam melakukan aktivitas *drag and drop, auto complete, chat, voice* dapat dilakukan layaknya aplikasi *desktop*.

c. *Web* 3.0

Web 3.0 adalah generasi ketiga dari layanan internet berbasis *web*. *Web* 3.0 merupakan pengembangan dari *website* di mana konten *web* tidak hanya dalam format bahasa manusia yang umum , tetapi juga dalam format yang dapat dibaca oleh bahasa mesin. *Web* ini masih dalam tahap pengembangan. Akan tetapi, banyak yang memperkirakan *web* ini akan berkembang pesat pada tahun 2010-2020.

Konsep *Web* 3.0 itu sendiri pertama kali diperkenalkan pada tahun 2001, saat Tim Berners-Lee, penemu *World Wide Web*, menulis sebuah artikel ilmiah yang menggambarkan *Web* 3.0 sebagai sebuah sarana bagi mesin untuk membaca halaman-halaman *Web*. Fungsi *web* menjadi wadah untuk pertukaran data,

informasi, dan pengetahuan yang dapat menghasilkan kecerdasan buatan yang dapat mengerti keinginan penggunanya, dengan menggunakan *web* 3.0 kita juga diberikan keleluasaan untuk dapat melakukan modifikasi pada *website* itu sendiri.

2.8.3 Web Server

Web server adalah sebuah perangkat lunak atau *software* pada *server* yang berfungsi menerima permintaan (*request*) melalui *HTTP* atau *HTTPS* dari *client* yang dengan menggunakan *web browser* dan mengirimkan kembali (*response*) hasilnya dalam bentuk *web page* yang pada umumnya ber*extention html* maupun *php*. Sehingga dapat disimpulkan bahwa *web server* adalah sebuah perangkat lunak yang berfungsi untuk melayani permintaan *web pages* oleh *client* baik itu melalui protokol HTTP atau HTTPS dan mengirimkan dokumen dengan format html maupun *php* ke komputer *client* tersebut [].

2.8.4 Aplikasi Website

Internet pada zaman modern saat ini sudah menjadi kebutuhan pokok dan menjadi hal yang tidak terpisahkan bagi kebanyakan orang, ada pengguna internet dan ada pula pembuat aplikasi internet atau aplikasi web, pengguna internet tidak dituntut untuk memahami bagaimana halaman web dibuat, tetapi lain halnya dengan pembuat, pembuat harus mampu dan memahami bagaimana aplikasi internet tersebut bisa dibuat, bahasa pemrograman apa yang harus digunakan, bagaimana menyusun sebuah program untuk membuat aplikasi internet sehingga menjadi sebuah halaman web yang bagus sesuai dengan yang diharapkan.

Aplikasi web merupakan sebuah aplikasi yang mengunakan teknologi browser untuk menjalankan aplikasi dan diakses melalui jaringan komputer (Remick, 2011). Sedangkan meurut (Rouse, 2011) aplikasi web adalah sebuah program yang disimpan di Server dan dikirim melalui internet dan diakses melalui antarmuka browser.

Dari pengertian diatas dapat disimpulkan aplikasi web merupakan aplikasi yang diakses mengunakan web berowser melalui jaringan internet atau intranet. Aplikasi web juga merupakan suatu perangkat lunak komputer yang dikodekan dalam bahasa pemrograman yang mendukung perangkat lunak berbasis web seperti

HTML, JavaScript, CSS, Ruby, Python, Php, Java dan bahasa pemrograman lainnya. Adapun Gambaran rincian aplikasi web adalah sebagai berikut :

Ada 2 bagian pokok dalam aplikasi web, yang pertama adalah sisi client dan yang kedua adalah sisi server, sisi client dalam hal ini adalah PC atau bisa juga Perangkat mobile yang terhubung kejaringan internet, Client dapat mengakses aplikasi web melalui web browser seperti internet explorer, mozila fire fox, google crome, opera dan lain-lain, sedangkan server adalah perangkat komputer dengan spesifikasi yang bagus digunakan untuk menyimpan aplikasi web beserta database server yang siap untuk diakses oleh client, Client bertugas meminta halaman web server melalui Web Browser, Web browser akan meneruskannya ke Server dimana Aplikasi Web berada, Komputer Server akan mengolah permintaan dari client, ketika halaman web yang diminta ditemukan maka computer server akan mengirimkannya ke computer client dan halaman web yang diminta akan ditampilkan pada web browser di computer client.

2.8.5 HTML

HTML (*HyperText Markup Language*) adalah suatu format data yang digunakan untuk membuat dokumen hypertext yang dapat dibaca dari satu platform ke platform lainnya tanpa melakukan suatu perubahan apapun. Dokumen HTML disebut markup language karena mengandung tanda – tanda tertentu yang digunakan untuk menentukan tampilan teks dan tingkat kepentingan dari teks tersebut dalam suatu dokumen, contoh struktur bahasa HTML:

```
<html>
<head>
<title>Untitled Document</title>
</head>
<body>
```

Gambar 2. 7 Contoh HTML

2.8.6 CSS

Cascading Style Sheet atau CSS adalah sebuah pemrogamaran atau boleh dibilang script yang mengendalikan beberapa komponen (tag html) dalam sebuah

website sehingga tampilan akan menjadi lebih terstruktur dan seragam. Ada tiga jenis cara dalam penggunaan CSS ini, yaitu:

1. Ex*t*ernal *Style Sheet*
2. *Internal Style Sheet*
3. *Inline Style Sheet*

Berikut penjelasannya dari ketiga poin diatas :

1. *External Style Sheet*

Adalah CSS yang dibuat dalam file terpisah dengan ekstensi file .css. Untuk memanggilnya menggunakan script html yang disisipkan antara tag head sebagai berikut:

```
<link rel="stylesheet"
type="text/css"
href="ditektori/filecss.css" />
```

Gambar 2. 8 Contoh eksternal css

Pada bagian href tersebut tinggal diisi mengarah ke direktori dimana Anda menempatkan file cssnya. Penggunaan sistem embeding css ini sangat disarankan karena memang banyak kelebihan yaitu mudah untuk melakukan pengeditan, mengatur semantik dan praktis terkumpul dari satu file.

2. *Internal Style Sheet*

Adalah cara embeding css dengan menulis langsung di dalam file html yang ingin kita atus tampilannya. Penulisan ini disisipkan diantara tag head juga dengan diapit oleh tag style. Berikut ini contohnya:

```
<style type="text/css">
body{ background:#cccccc;
     font-family:Arial;}
</style>
```

Gambar 2. 9 Contoh internal css

3. *Inline Style Sheet*

Adalah penulisan script css langsung pada tag html dengan menambahkan style di dalamnya. Contoh

```
<p style="font-size:20px;">Tulisan yang di atur </p>
```

Gambar 2. 10 Contoh inline css

Dua cara terakhir ini tidak disarankan digunakan karena ribet dan memperbesar file setiap html anda sekaligus harus menulis di beberapa tempat sekaligus, kecuali memang sangat darurat. Darurat disini berarti memang tampilan segera dibutuhkan oleh pengguna saat itu sedangkan anda sedang membuka file itu dan tidak sempat membuka file css karena keterbatasan koneksi misalnya.

Namun penggunaan external css akan memudahkan Anda di waktu kedepan dalam hal update dan editing.

2.8.7 MySQL

MySQL adalah *Relational Database Managemen Sistem* (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (*General Public Licence*). Dimana setiap orang bebas untuk menggunakannya, tapi tidak boleh dijadikan produk turunan yang bersifat *Closed Source* atau komersial[12].

MySQL sebenarnya merupakan turunan salah satu konsep utama dalam database sejak lama, yaitu SQL (*Structur Query Language*). SQL adalah sebuah konsep pengoperasian database terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis[12].

Keandalan suatu sistem database (DBMS) dapat diketahui dari cara kerja optimizer-nya dalam melakukan proses perintah-perintah SQL, yang dibuat oleh user maupun program-program aplikasinya sebagai data base server lainnya dalam query data. Hal ini terbukti untuk query yang dilakukan oleh single user, kecepatan query MySQL bisa sepuluh kali lipat lebih cepat dari postgreSQL dan lima kali lebih cepat dibanding interbase[12].

Penggunaan perintah MySQL diantaranya:

a. DDL (*Data Definition Language*) yaitu perintah yang digunakan untuk mendefinisikan struktur database atau skema Contoh :

```
CREATE TABLE user
(Users varchar(20) NOT
NULL, password varchar(32)
NOT NULL,
PRIMARY KEY (users));
```

Gambar 2. 11 Contoh penggunaan ddl

b. DML (*Data Manipulation Language*) yaitu perintah yang digunakan untuk mengatur/mengelola data delam skema objek Contoh:

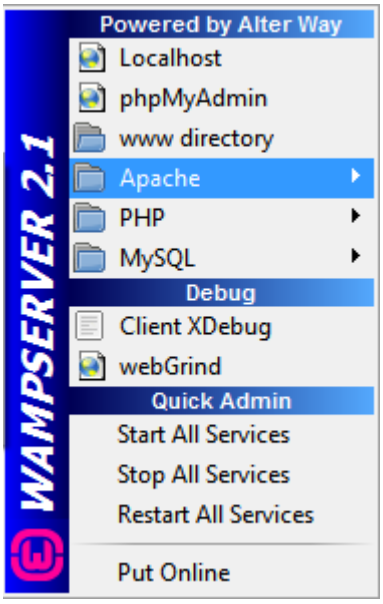
```
INSERT INTO user VALUES('ranti');
```

Gambar 2. 12 Contoh penggunaan dml

2.8.8 WAMP Server

WAMP adalah paket program yang dibuat secara independent yang dapat diinstall pada komputer yang menggunakan sistem operasi Windows. Program ini untuk melayani halaman web dinamis suatu jaringan komputer, seperti internet atau jaringan pribadi.

WAMP adalah suatu singkatan yang terbentuk dari sistem operasi (*Windows*) dan komponen utama paket : Apache, Mysql dan PHP (atau Perl atau Phyton). Apache adalah suatu server web, yang mengijinkan orang-orang dengan web browsers seperti Internet explorer atau Firefox untuk menghubungkan suatu komputer dan melihat informasi dari suatu halaman web.

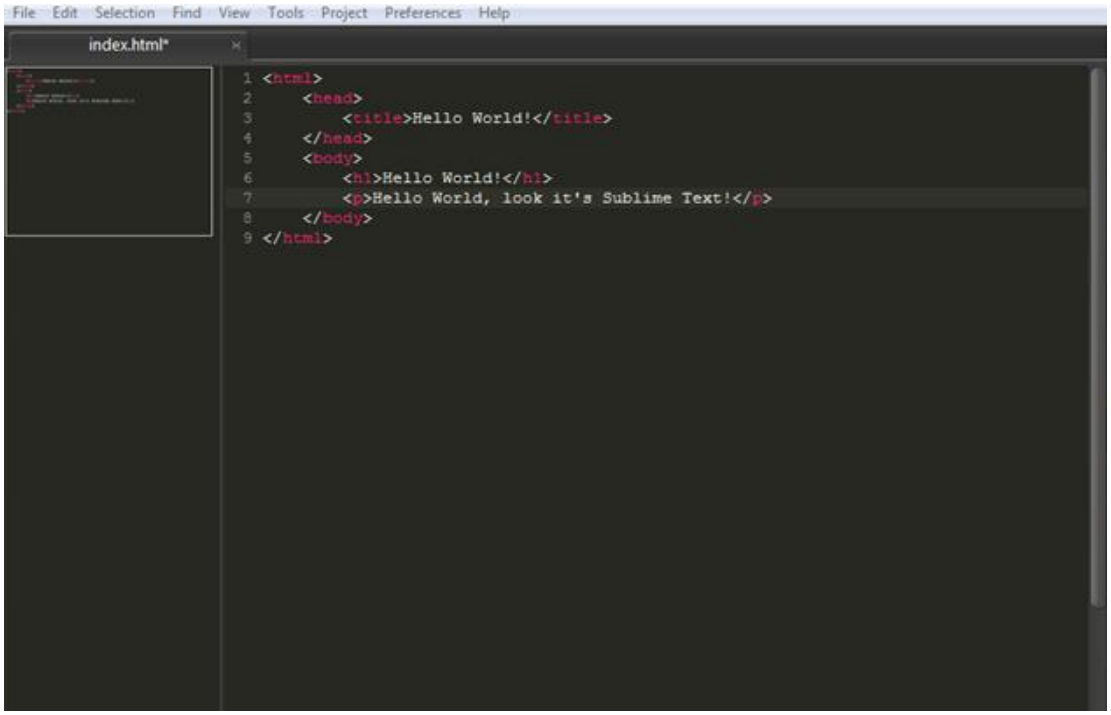


Gambar 2. 13 Wamp Server

2.8.9 Sublime

Dalam membuat situs *web* diperlukan suatu *editor*. Salah satu contoh *editor* yang sangat sederhana adalah *notepad*. *Sublime Text* merupakan *editor*

HTML yang *professional* mendesain, menulis kode program dan mengembangkan *website*, halaman *web*, dan aplikasi *web*. Berikut ini tampilan aplikasi *Editor Sublime Text* adalah:



Gambar 2. 14 Sublime text editor

2.9 Perancangan Sistem

Perancangan sistem dapat berarti membuat sistem yang baru untuk menggantikan sistem yang lama secara keseluruhan atau memperbaiki sistem yang sudah ada. Dalam merancang sebuah sistem biasanya menggunakan alat bantu berupa[11] :

2.9.1 Entity Relationship Diagram (ERD)

ERD adalah peralatan yang berfungsi untuk menggambarkan relasi atau hubungan dari dua file atau dua tabel yang dapat digolongkan dalam tiga macam bentuk relasi, yaitu [11]:

1. Entity

Adalah suatu objek yang dapat diidentifikasi dalam lingkungan pemakai, sesuatu yang penting bagi pemakai dalam konteks sistem yang akan dibuat.

2. Atribut

Entity mempunyai elemen yang disebut atribut, dan berfungsi mendeskripsikan karakter entity.

3. Hubungan

Relationship sebagaimana halnya entity maka dalam hubungannya pun harus dibedakan antara hubungan atau bentuk hubungan antar entity dengan isi dari hubungan itu sendiri. Beberapa macam hubungan antar relasi antara lain[11] :

a. Satu ke Satu (*One to One*)

Bentuk relasi antara satu entitas dengan sejumlah satu entitas dengan jumlah yang sama.



Gambar 2. 15 Relasi Satu ke Satu

b. Satu ke Banyak (*One to Many*)

Bentuk dari entitas dengan jumlah satu ke entitas lain yang berjumlah lebih dari satu (entitas dengan banyak alternatif tujuan).



Gambar 2. 16 Relasi Satu ke Banyak

c. Banyak ke Banyak (*Many to Many*)

Bentuk relasi yang mendeskripsikan permasalahan yang komplek yaitu hubungan antara entitas yang berjumlah lebih dari satu dengan entitas yang sama.



Gambar 2. 17 Relasi Banyak ke Banyak

2.9.2 Diagram Konteks

Diagram konteks merupakan pola penggambaran yang berfungsi untuk memperlihatkan interaksi tersebut dengan lingkungan dimana sistem tersebut ditempatkan. Dalam diagram konteks, sistem dianggap sebuah objek yang tidak dijelaskan secara rinci karena yng ditekankan adalah interaksi sistem dengan lingkungan yang akan mengaksesnya.

2.9.3 Data Flow Diagram (DFD)

Data Flow Diagram (DFD –DAD/Diagram Alir Data) memperlihatkan hubungan fungsional dari nilai yang dihitung oleh sistem, termasuk nilai masukan, nilai keluaran, serta tempat penyimpanan internal. DAD adalah gambaran grafis yang memperlihatkan aliran data dari sumbernya dalam objek kemudian melewati proses yang mentransformasinya ke tujuan yang lain, yang ada pada objek lain.

DAD sering digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika tanpa mempertimbangan lingkungan fisik dimana data tersebut mengalir. DFD merupakan alat yang digunakan pada metodologi pengembangan sistem yang terstruktur (structured analysis and design). DFD merupakan alat yang cukup populer sekarang ini, karena dapat menggambarkan arus data di dalam sistem dengan terstruktur jelas[11].

Beberapa simbol yang digunakan dalam Data Flow Diagram (DFD) antara lain[11]:

1. *External Entity* (kesatuan luar) atau *boundary* (batas sistem)

Setiap sistem pasti mempunyai batas sistem (*boundary*) yang memisahkan suatu sistem dengan lingkungan luarnya. Sistem akan menerima input dan menghasilkan output kepada lingkungan luarnya. Kesatuan luar (*external entity*) merupakan kesatuan (*entity*) di lingkungan luar sistem yang dapat berupa orang, organisasi atau sistem lainnya yang berada di lingkungan luarnya yang akan memberikan input atau menerima output d ari sistem.



Gambar 2. 18 Eksternal entity

2. Data Flow (arus data)

Arus data (*data flow*) di DFD diberi simbol suatu panah. Arus data ini mengalir diantara proses (*process*), simpanan data (*data strore*) dan kesatuan luar (*external entity*). Arus data ini menunjukkan arus dari data yang dapat berupa masukan untuk sistem atau hasil dari proses sistem.



Gambar 2. 19 Arus data

3. Process (proses)

Suatu proses adalah kegiatan atau kerja yang dilakukan oleh orang , mesin atau kompuiter dari hasil suatu arus data yang masuk ke dalam proses untuk dihasilkan arus data yang akan keluar dari proses. Untuk *physical* data *flow diagram* (PDFD), proses dapat dilakukan oleh orang, mesin atua komputer, sedangkan untuk logical data flow diagram (LDFD), suatu proses hanya menunjukkan proses dari komputer. Setiap proses harus diberi penjelasan yang lengkap meliputi identifikasi proses, nama proses dam pemroses.



Gambar 2. 20 Proses

4. Data Store (simpanan luar)

Simpanan data (data *store*) merupakan simpanan dari data yang dapat berupa, yaitu suatu file atau database di sistem komputer, suatu arsip atau catatan manual, suatu kotak tempat data di meja seseorang, suatu tabel acuan manual, dan suatu agenda atau buku.



Gambar 2. 21 Database