SISTEM KEAMANAN *JAIL BASH* UNTUK MENGAMANKAN AKUN LEGAL DARI KEJAHATAN *INTERNET* MENGGUNAKAN *THC-HYDRA*

Cakra Aminuddin Hamka, Haruno Sajati, Yuliani Indrianingsih
Teknik Informatika
Sekolah Tinggi Teknologi Adisutjipto Yogyakarta
informatika@stta.ac.id

ABSTRACT

Along with the developmentof current technology, making technologyis very important in today's life. Security level digital data has become more vulnerable to exploitation, the problem arises when an information technology device was attacked by people who do not want to take a responsible and important data illegally, so the administrator must act quickly to secure important data. Making a data security technology on the internet is very important information. Limitations administrator underlying the creation of a system that is able to detect and defense systems against such attacks is automation, so that it can be applied to data security. The system is built to prevent attacks on computer networks with more specificon THC-Hydra. This system will analyze the number of errors in the log into the database, and ıf the error exceeds the tolerance rules are made by the administrator. If the error is more than 3 times in one minute, then with automation, the system will create a rule that can imprison users who do not have such access, and access to the prison in the illegaluser can not perform such activities on legal access and access to activities conducted illegal user can be recorded on a file and jail.txt. Sounder any circumstances and not in supervising administrator, the security of other user data will be safe and can not be retrieved or viewed by a user of illegal access.

Keywords: Automation, THC-Hydra, the illegal user access, legal user access.

1. Pendahuluan

Keamanan jaringan komputer sebagai bagian dari sebuah sistem informasi adalah sangat penting untuk menjaga validalitas dan integritas data serta menjamin ketersediaan layanan bagi pengguna dan *administrator*. Sistem deteksi penyusupan jaringan *internet* saat ini umunya mampu mendeteksi berbagai jenis serangan, namun tidak mampu mengambil tindakan lebih lanjut, dan tidak memiliki kemanfaatan ilmu pengetahuan. Dalam penelitian ini adalah dengan membangun aplikasi yang dapat melakukan kegiatan *auto respon* terhadap aktifitas penyusup dengan menggunakan bahasa pemrograman *php*, dan dapat merekam segala aktifitas yang dilakukan akses *user* ilegal dengan menggunakan bahasa pemrograman *C++*. Dan tidak hanya itu, penelitian ini dapat melakukan proses *recovery* data. Dikarenakan saat ini serangan sangat beraneka-ragam, salah satunya dengan menggunakan *THC-Hydra,* maka hal ini sangat penting untuk dalam proses penelitian ini, maka munculah sebuah ide untuk memberikan sebuah sistem penjara untuk mengamankan akun, dan di dalam penjara tersebut, kegiatan akses *user* ilegal tidak dapat terlaksana dan kegiatan tersebut akan direkam oleh sistem tersebut.

2. Metodologi

2.1 *Shell*

Mahardika (2003), *shell* adalah program penghubung *user* dengan *kernel* sistem operasi adalah program *shell* tersebut. Setiap perintah yang inputkan oleh *user* akan diterjemahkan oleh *shell* kemudian dikirimkan hasilnya ke *kernel* dan *kernel* tersebut melakukan operasi yang diminta oleh *user*. Gambaran saat *shell* menjalankan program, yang tetap memanggil sistem *call fork* dan *exec* untuk menjalankan program yang diinginkan *user*.

2.2 *Bash*

Bash adalah *shell*, untuk penerjemah bahasa perintah, yang terdapat pada sistem operasi *GNU*, misalnya *Linux*, *BSD*, *SCO* (paket *Skunkware). Bash* kompatibel dengan *shell sh* dan kemampuan atau karakteristik yang dimiliki oleh *Korn Shell (ksh)* dan *C Shell (Csh)*. Hal ini menjadi implementasi dari *IEEEPOSIX Shell* dan bagian dari *IEEEPOSIX* spesifikasi *(Standar IEEE 1003.1). Bash* melakukan perbaikan fungsional selama *sh* untuk digunakan baik interaktif dan pemrograman. *Bash* sangat portabel saat melakukan penetrasi di hampir setiap versi *Unix*, dan didukung untuk *MS-DOS*, *OS/2*, dan *Windows* (Susanto, 2004).

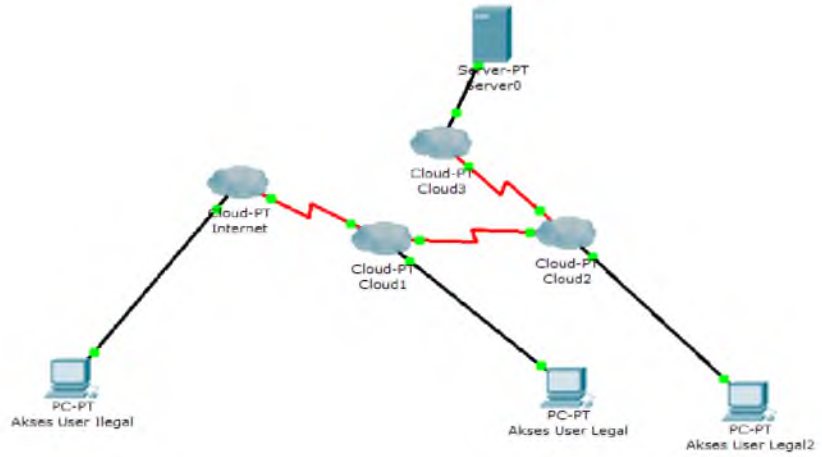
2.3 Konsep Sistem *User* di *Linux*

Sistem *user* pada *LINUX* harus terdaftar pada sistem *LINUX* secara otomatis diberikan sebuah direktori kerja sesuai nama *user* yang terdaftar. Sistem *Linux* selalu merekam *user* baru yang ditambahkan oleh *super user*. Pada saat menambahkan *user* baru, setiap *user* akan selalu diberikan sebuah direktori kerja pada*/home*. Pada direktori */home* akan terdapat direktori kerja masing-masing *user* yang telah terdaftar menggunakan nama *standard* oleh *user*nya. Sebagai keamanan, *user* biasa tidak akan dapat mengakses direktori lain atau direktori kerja *user* lainnya. *User* yang dapat mengakses direktori yang bukan miliknya apabila telah diberi hak akses pada *super user* (Nugroho, 2005).

2.4 THC-Hydra

THC Hydra adalah suatu program perangkat lunak untuk mempenetrasi *password* sebuah *user* atau pengguna pada sistem, dalam penerapannya *Hydra* sangat memungkinkan tingkat keberhasilannya apabila suatu *password* tingkat rendah, dikarenakan *Hydra* melakukan kombinasi yang umum dan secara terus menerus hingga berhasil. Penetrasi *Hydra* sangat mudah, hanya mengetikkan suatu *script* untuk menuju sebuah *user* dan aplikasi tersebut dapat berjalan. Program ini mempunyai kemudahan untuk bagaimana mendapatkan akses tidak sah dari *remote* ke sistem.

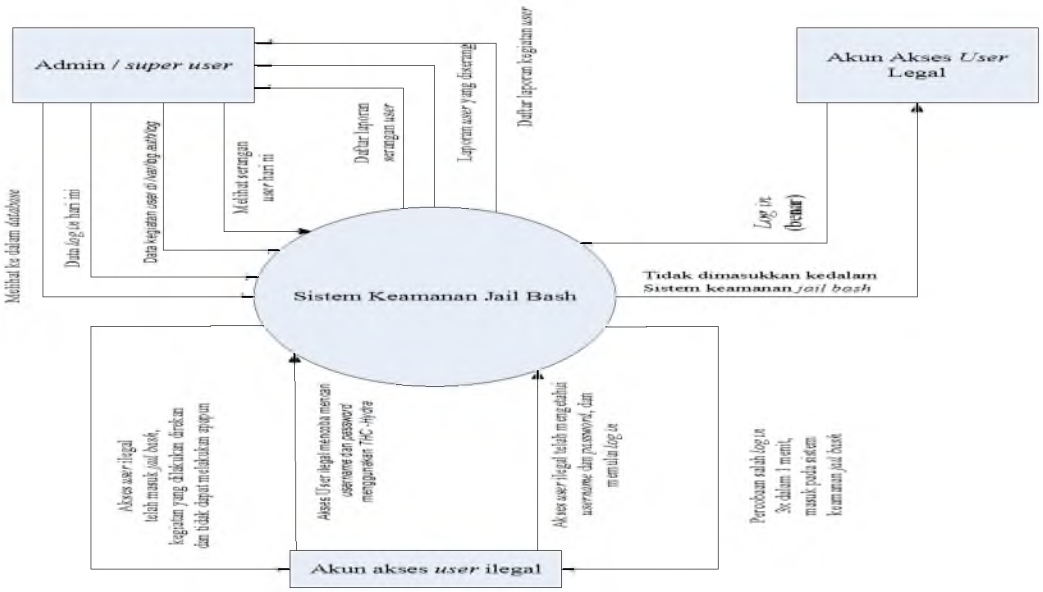
2.5 Analisa Sistem



Gambar 1 Bagan Kegiatan.

Berdasarkan bagan kegiatan di atas, dapat dijelaskan bahwa kegiatan akses *user* ilegal dapat dilakukan dengan dengan menggunakan cara mengakses *internet* terlebih dahulu, dan mampu melakukan kegiatan yang semestinya tidak dapat diakses tersebut, dengan salah satu contohnya adalah dengan melakukan penetrasi *password* akses *user* legal. Tidak hanya penetrasi *password* kepada akses *user* legal saja, namun dapat melakukan penetrasi *password* terhadap server dan dapat me-*remote access* dari berbeda tempat.

2.6 Diagram Konteks



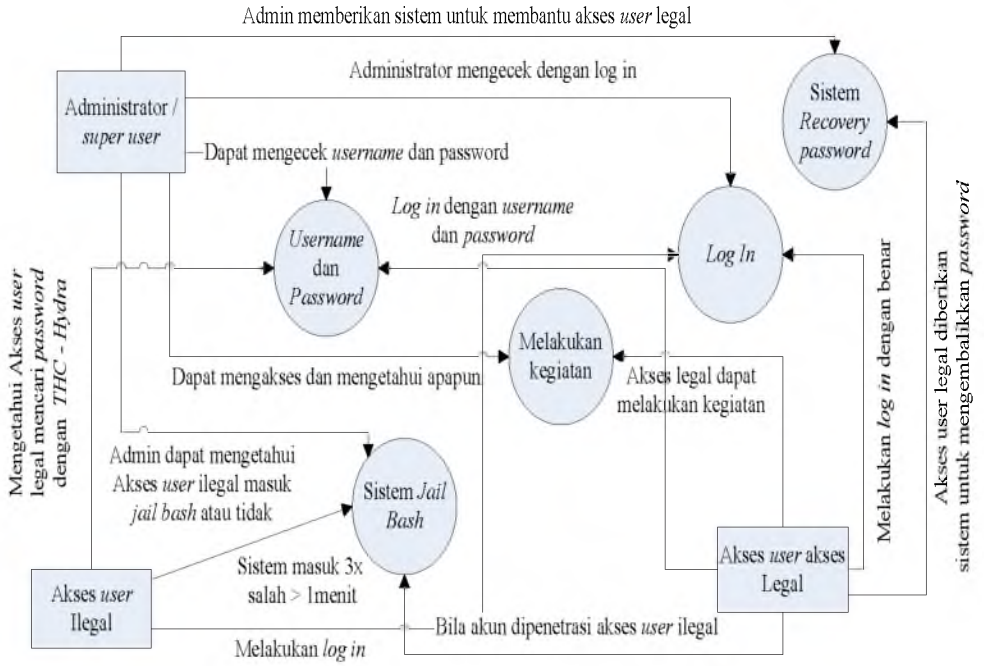
Gambar 2 Diagram Konteks.

Diagram konteks diatas, menjelaskan bahwa *administrator/super user* dapat mengakses kepada sistem keamanan yang dinamai *jail bash*. Sistem keamanan *jail bash* ini akan mengamankan akun yang ada pada sistem tersebut. Akses *user* ilegal melakukan kegiatan penetrasi *password* kepada akun yang dimiliki oleh akses *user*

legal. Saat terjadi kegiatan penetrasi tersebut menggunakan aplikasi *THC-Hydra*, maka serangan tersebut akan terus menerus. Kegiatan terus menerus untuk mengetahui *password* tersebut, terdeteksi oleh sistem dan akan secara otomatisasi memasukkan akses *user* ilegal kedalam sistem keamanan *jail bash* tersebut.

2.7 *Data Flow Diagram* (DFD)

Dapat dijelaskan bahwa *administrator* dapat melakukan dan mengetahui apapun yang dilakukan oleh akses *user* legal dan akses *user* ilegal. Akses *user* legal apabila akunnya telah dipenetrasi oleh akses *user* ilegal, maka akunnya pun akan terjebak dalam sistem *jail bash*, namun dapat dilakukan sistem *recovery* yang secara otomatisasi yang telah dibuat oleh *administrator*, dengan beberapa kriteria dan peraturan yang berlaku. Namun bagi akses *user* ilegal, untuk mendapatkan *username* dan *password* akun akses legal menggunakan aplikasi tambahan yaitu *THC-Hydra*, setelah *hydra* tersebut telah dapat mengetahui *username* dan *password* tersebut, maka langkah selanjutnya adalah sistem akan memasukkan akun yang dipenetrasi kedalam sistem *jail bash* untuk mengamankan data akun yang telah dipenetrasi.



Gambar 3 DFD sistem kerja aplikasi

2.8 Squirrelmail

Squirrelmail adalah salah satu aplikasi *web* yang sudah terpasang di *CPanel* untuk keperluan membaca *email*. *Squirrelmail* adalah aplikasi *webmail* yang mendukung protokol *IMAP* dan *SMTP* dan menampilkan halaman dalam format *HTML* tanpa membutuhkan *javascript*, sehingga bisa dengan mudah diakses menggunakan *browser* apapun dan sangat ringan. *Squirrelmail* sendiri disini digunakan untuk proses *recovery* data dan *password* akun akses legal. Didalamnya terdapat akses *link* yang mengharuskan akses legal untuk mengganti *password* tersebut dan secara otomatisasi, sistem akan mendeteksi bahwa akun akses legal tersebut dapat

melakukan aktifitas secara normal kembali dan tidak terpenjara dalam sistem *jail bash*.

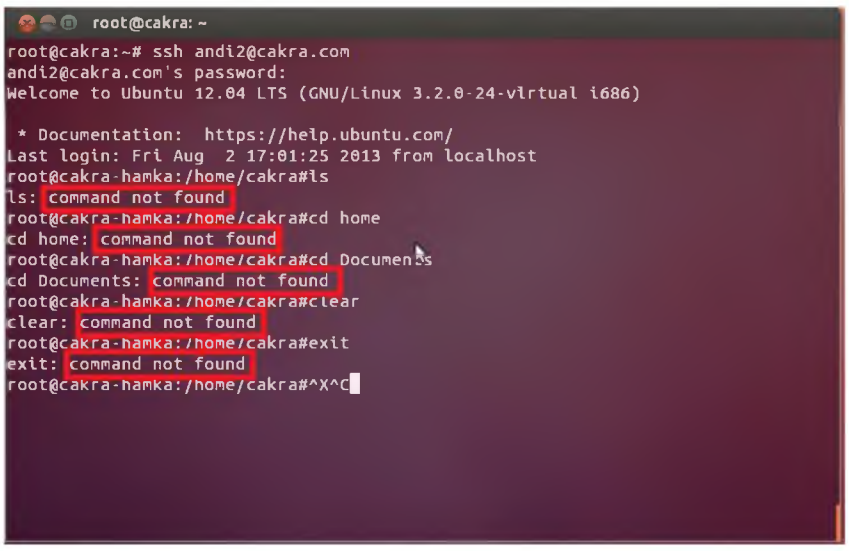
3. Hasil dan Pembahasan

3.1 Konfigurasi Aplikasi

Crontab adalah sebuah perangkat lunak yang berguna untuk penjadwalan proses yang akan di eksekusi, *crontab* berjalan dibalik layar (*daemon*) yang terdapat pada sebuah sistem operasi linux. Hal ini memungkinkah penggunauntuk melakukan eksekusi aplikasi atau skrip program sesuai dengan waktu yang telah ditentukan. *Cron* dikendalikan oleh *crontab* (*tabel cron*) *file*, sebuah *file* konfigurasi yang menentukan perintah *shell* untuk menjalankan secara berkala pada jadwal yang diberikan. Dalam penulisan *crontab* terdapat format untuk melakukan penjadwalan menunjukkan bagian-bagian format penjadwalan yang disediakan.

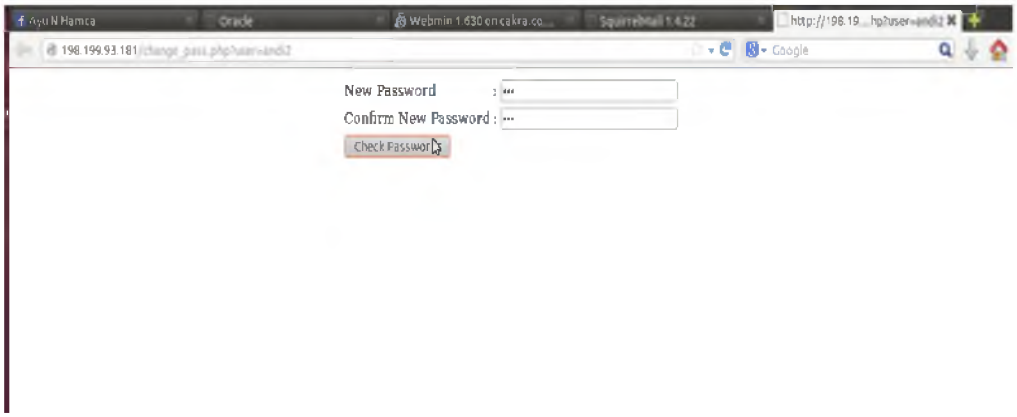
3.2 Uji Coba

Uji coba *jail bash* ini dilakukan dan diletakkan pada sebuah *web serverDigital Ocean*. Pengujian dilakukan dengan 11 kali, yaitu dengan akses *user* legal sebanyak 10 kali, akses *user* ilegal sekali. Pengujian ini dapat dilakukan dibanyak akses perangkat komputer dimanapun, dengan menggunakan *ssh*. Berikut skema pengujian yang dilakukan.



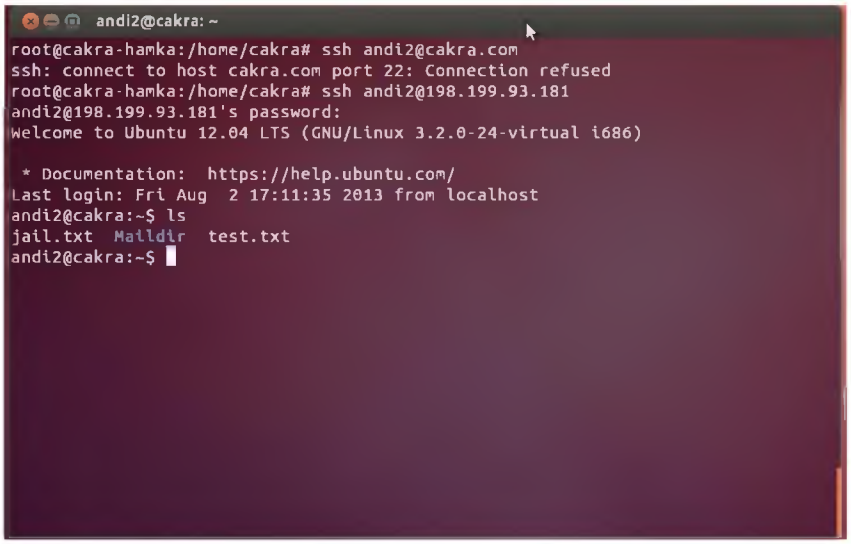
Gambar 4 kegagalan kegiatan akses *user* legal pada terminal.

Apabila saat akses *user* legal masuk dalam *terminal*, maka akan terjebak pula dalam *shell* penjara, dan tidak dapat melakukan apapun dalam *terminal*nya.



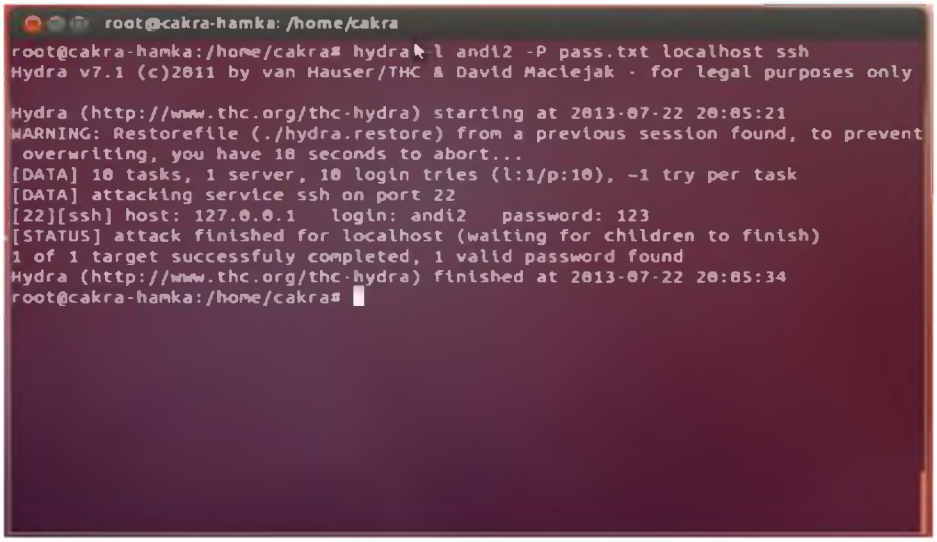
Gambar 5 Proses *recoverypassword* yang baru.

Namun dalam konteks disini, akses *user* legal telah dikirimkan oleh *administrator* sebuah *email* yang berisi perintah dan *link* untuk mengganti *password* yang baru. *Link* ini otomatis akan terkirim dan akses *user* legal dapat langsung mengganti *password* dengan ketentuan yang telah diberlakukan oleh *administrator*.



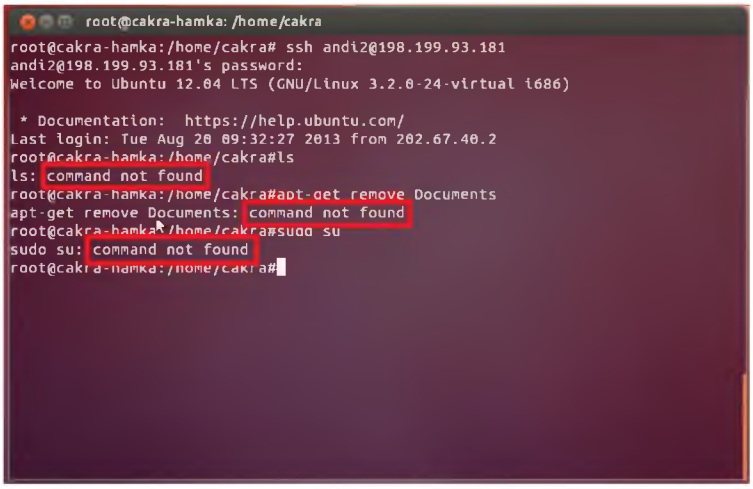
Gambar 6 Akses *user* legal dapat melakukan aktifitas normal.

Setelah kembali lagi pada dan telah *recovery password*, ini menunjukkan bahwa *shell* andi2 telah dapat digunakan seperti biasanya, dan tidak terjebak pada *shell* penjara seperti sebelumnya, dan kegiatan seperti *ls* (*list*) telah dapat diakses seperti sebelumnya.



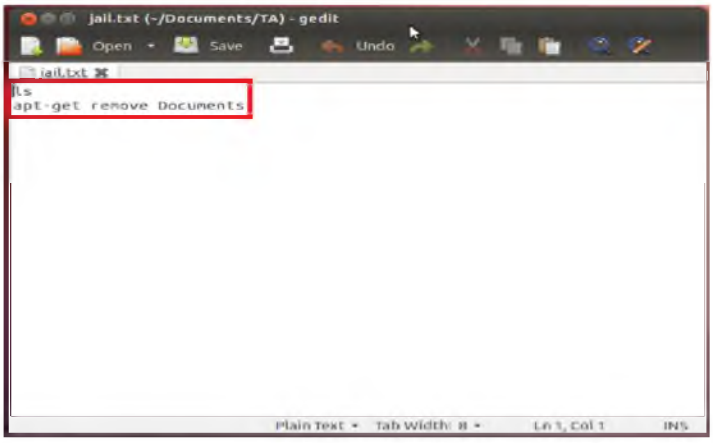
Gambar 7 Penetrasi dan aktifitas *THC-Hydra oleh* akses *user* ilegal mencari *password*.

Akses *user* ilegal adalah *user* yang tidak memiliki hak akses terhadap layanan *ssh*, maka akses *user* illegal harus mengetahui *username* dan *password* tersebut, tentu pasti tanpa sepengetahuan dan dengan cara tidak lazim, yaitu dengan mencari-cari *password* tersebut dan mencoba–coba hingga benar. Biasanya akses *user* ilegal melakukan penyerangan terhadap akun akses *user* legal dengan menggunakan THC-Hydra.



Gambar 8 Kegiatan akses *user* ilegal.

Akses *user* ilegal dapat masuk dalam *shell* penjara dikarenakan menggunakan sebuah aplikasi *Hydra*, konsep *Hydra* dengan mencoba terus menerus hingga ditemukannya *password* yang cocok, maka dengan demikian *administrator* dapat melakukan sebuah kondisi yang apabila terjadi kesalahan lebih dari 3 dalam 1 menit, maka aktifitas apapun di *terminal* tidak akan dapat terlaksana, dan aktifitas pun akan disimpan dengan sebuah file *jail.txt*.



Gambar 9 Hasil pencatatan dari *jail bash*.

4. Kesimpulan Dan Saran

Kesimpulan yang diperoleh dari uji coba dan analisis adalah sebagai berikut:

1. Penerapan dalam *shelljail* telah dapat berfungsi secara otomatisasi dalam memenjarakan *akses user ilegal* dengan penerapan aturan yang di buat administrator.
2. Selain *system* dapat melindungi dirinya secara otomatis dan *system* juga dapat dan berhasil merekam (*login*) aktifitas akses *user* ilegal dalam bentuk apapun yang dilakukan oleh *akses user ilegal* di dalam *shell* penjara.
3. Proses *recovery* terhadap *akses user legal* telah berhasil dilakukan, melalui email yang telah dibuat dan dikirim oleh *administrator*.

Dari hasil perancangan dan pengujian yang dilakukan dapat diberikan saran untuk pengembangan selanjutnya, antara lain :

1. Aplikasi ini berbasis semi otomatis, dikarenakan pemicu pertahanan dari akses *user* ilegal ke dalam *jail* dari hasil kegiatan dan kejadian pada */var/log/auth.log*, bukan berbabasis pola waktu (*realtime)* pada saat akses user ilegal sedang melakukan aksi penetrasi.
2. Aplikasi ini dapat dikembangkan dengan beberapa komputer yang banyak dan saling berhubungan satu dengan yang lainnya dan dengan beberapa jenis serangan yang lebih banyak digunakan oleh *akses user ilegal*.
3. Sistem yang dibangun masih berdasarkan pemicu waktu, yaitu 1 menit sekali, sehingga sistem pertahanan dapat dikembangkan berdasarkan pemicu kejadian penyerangan.

Daftar Pustaka

Susanto, Budi. 2004, *Pemrograman Script pada UNIX / LINUX, edisi pertama*. Yogyakarta. Penerbit Graha Ilmu.

Mahardika, Irfan. 2003. *Secure Remote Login Pada Sistem Operasi Slackware Linux*. Semarang. Universitas Diponegoro Semarang.

Mubarok, M.Husna dan dan Yoyok Bagiyo 2007. *Pemrograman Port Paralel dengan GCC/Linux dan Gambas/ M.Husna Mubarok*, hal 37.

Nugroho, Bunafit. 2005. *Instalasi dan Konfigurasi Jaringan Windows dan Linux, edisi pertama*, Yogyakarta. Penerbit Andi, hal,211.

Samsyiar, Evara. 2006. *Belajar Sendiri Administrasi Database Oracle 10g*. Jakarta: Penerbit PT Elex Media komputindo, halaman 34.

Syafii, M. 2004, *Konfigurasi Server Linux dengan Webmin*. Yogyakarta: Penerbit Andi.

MADCOMS. 2004. *Aplikasi Program PHP dan MySQL untuk membuat Website Interaktif*. Yogyakarta :Penerbit Andi.

http://www.thc.org/thc-hydra/, diakses pada tanggal 5 Juli 2013.