Implementasi Live Audio Streaming Menggunakan Raspberry Pi

Mohammad Hardian[1] **Nina Hendrarini[2]** **Ismail[3]**

[1,2,3] Fakultas Ilmu Terapan – Telkom University
[1]mohammad.hardian@gmail.com [2]ninahendrarini@tass.telkomuniversity.ac.id
[3]ismail@tass.telkomuniversity.ac.id

Abstrak

IGS (Integrated Ground System) adalah sebuah aplikasi berbasis web yang digunakan oleh dispatcher, pilot, tech team, engineer, airline, dan document manager. IGS ini berfungsi mengontrol semua kegiatan dalam bidang avionic. Sampai saat ini IGS (Integrated Ground System) Ini tidak mempunyai sistem pertahanan jadi seseorang yang tidak mempunyai hak dapat mengakses, Sehingga masih adanya celah yang membuat informasi yang ada di dalam aplikasi tersebut mudah diambil. maka diperlukan mekanisme pengaturan pertahanan pada web server IGS, dengan cara yang membangun sebuah SSL yang menciptakan sebuah sertifikat resmi admin, dan juga otentikasi Kerberos (pada sisi client dan sisi server) yang dapat membatasi hak akses atau pemberian izin kepada beberapa user untuk mengakses. Dari hasil pengujian dapat diambil sebagai berikut : Sistem memberikan koneksi jaringan dengan aman menjaga server dari ancaman user yang tidak bertanggung jawab untuk memasukan ataupun mengambil informasi yang ada pada IGS, dan sebuah user yang sudah mempunyai akses dapat memasukan informasi kegiatan penerbangan pada IGS.

Kata kunci: Keamanan, Otentikasi web, Integrated Ground System, Secure Socket Layer, Kerberos

Abstract

IGS (Integrated Ground System) is a web-based application that used by dispatchers, pilot, tech team, engineers airline, and document manager. The IGS function to control all activities on avionics system. Until now it doesn't have a defensive system to make this secure, someone who doesn't have any permission to access can through or access it. So someone can take or looking for the information contained in IGS. It would require a defense mechanism in the web server settings to install SSL which creates an official certificate admin, and Kerberos authentication (on client side and server side) which can restrict permission or granting permission for multiple user to access. From the result of the system are made can be taken : The system provides secure network connection with maintaining servers from threats unauthorized user to gain access to input of retrieve the information that existed at the IGS, and a user who already has access to information can insert aviation activities in the IGS.

Keywords: Security, Web Authentication, Integrated Ground System, Secure Socket Layer, Kerberos

1. Latar Belakang

IGS (*Integrated Ground System*) adalah aplikasi yang digunakan oleh *dispatcher, pilot, tech team, engineer, airline,* dan *document manager*. IGS ini berfungsi mengontrol semua kegiatan dalam bidang *avionic*, dan IGS ini berbasis web. Sebelumnya aplikasi IGS ini masih manual atau analog dan selanjutnya dikembangkan oleh salah satu mahasiswa Telkom University Fakultas Ilmu Terapan sebagai *Automation Web Testing Tool* yang berbentuk digital.

Sampai saat ini aplikasi IGS (*Integrated Ground System*) yang berbasis *web* ini tidak mempunyai keamanan. Sehingga masih adanya celah yang membuat informasi yang ada di dalam aplikasi tersebut mudah diambil. Seiring dengan perkembangan internet yang semakin pesat, maka diperlukan mekanisme pengaturan pertahanan pada aplikasi IGS. Salah satu cara yang dapat dilakukan untuk membuat mekanisme pertahanan pada aplikasi IGS (*Integrated Ground System*) dengan penggunaan SSL

Secure Socket Layer (SSL) adalah suatu protokol yang diciptakan oleh Netscape untuk memastikan keamanan dalam bertransaksi di internet antara *web server* dan *browser* dari klien. SSL mengubah suatu protokol seperti TCP menjadi sebuah saluran komunikasi aman yang cocok untuk transaksi yang sensitif. Sebuah sertifikat pada SSL merupakan sebuah kumpulan data identifikasi dalam format dasar

dan data tersebut digunakan dalam proses verifikasi identitas antara *web server* dan *client*.

Pada SSL juga menyediakan otentikasi (pada sisi *client* dan opsional pada sisi *server*) terhadap pihak-pihak yang berkomunikasi. Pada studi kasus ini otentikasi menggunakan Kerberos yang juga dapat mengamankan koneksi antar dua titik, dan dapat melindungi dari pihak yang dapat melakukan hal yang bersifat mengganggu pada saat proses komunikasi. Pengiriman data dilakukan dengan meng-enkrip *username* dan *password* yang akan dikirim dengan kunci tertendu dan kemudian di dekrip di sisi server yang kemungkinan akan munculnya akan *passive attack*.

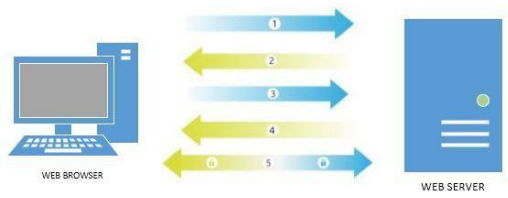
2. Dasar Teori

2.1. Secure Socket Layer (SSL)

Secure Sockets Layer, adalah metode enkripsi yang dikembangkan oleh Netscape untuk memberikan keamanan di internet. Ia mendukung beberapa protokol enkripsi dan memberikan autentikasi client dan server. SSL beroperasi pada layer transport, menciptakan saluran enkripsi yang aman untuk data, dan dapat mengenkripsi tipe data. Hal ini dapat dikatakan bahwa SSL merupakan Protokol berlapis dalam tiap lapisannya.

SSL mengambil data untuk dikirimkan, dipecahkan ke dalam blok-blok yang teratur, menerapkan MAC, dienkripsi, dan hasilnya dikirimkan. Di tempat tujuan, data didekripsi, verifikasi, dekompres, dan disusun kembali. Hasilnya dikirimkan ke klien di atasnya.

Cara kerja Secure Socket Layer (SSL) seperti pada gambar 1 dibawah ini :



Gambar 1 Proses Bekerjanya SSL

1. *Browser* menghubungkan ke server web (*website*) dijamin dengan SSL (https), *Browser* meminta agar server mengidentifikasi itu sendiri.
2. Server mengirimkan salinan Sertifikat SSL, termasuk kunci publik server.
3. *Browser* memeriksa akar sertifikat terhadap daftar CA terpercaya dan bahwa sertifikat tersebut belum berakhir, *unrevoked*, dan bahwa nama umum yang berlaku untuk situs web yang menghubungkan ke. Jika *browser* mempercayai sertifikat, menciptakan, mengenkripsi, dan mengirimkan kembali kunci simetris sesi menggunakan kunci publik server.
4. Server mendekripsi kunci simetris sesi menggunakan kunci pribadi dan mengirimkan kembali pengakuan dienkripsi dengan kunci sesi untuk memulai sesi dienkripsi.
5. Server dan Browser sekarang mengenkripsi semua data yang dikirimkan dengan kunci sesi.

2.2. Sniffing

Sniffing adalah kegiatan penyadapan pada traffic di jaringan komputer. Sniffing dibagi dua bagian, yaitu :

1. *Passive sniffing* merupakan penyadapan sebuah data dengan cara pasif, yaitu dengan menunggu data yang akan dikirim dan menangkap datanya. Cara ini efisien dalam diam – diam mengumpulkan data dari LAN. Salah satu metode yang digunakan pada passive sniffing dalam melakukan penyadapan data dengan mengorbankan keamanan fisik, dan menggunakan *virus Trojan*.
2. *Active sniffing* merupakan kegiatan perubahan paket data dalam jaringan. serangan terdiri dari *virus*, *Trojan*, *Logic-*

Bom, *worm*, dan *malware* yang mungkin menghancurkan *file* penting ,dan mensabotase sistem operasi. Pada sniffing ini berusaha untuk membebani sistem host dan memperlambat sehingga menjadi hang hingga tidak dapat digunakan.

2.3. Kerberos

Kerberos adalah sebuah protokol yang berfungsi sebagai otentikasi menggunakan pihak ketiga yang dipercaya (trusted third-party). Protokol tersebut menawarkan otentikasi pada jaringan yang tidak mempunyai keamanan.

Kerberos memlili sifat *transparent*, artinya *user* tidak perlu mengetahui tentang tahap – tahap otentikasi yang dilakukan pada jaringan, yang dilakukan user hanyalah *login* ke jaringan melalui program inisialisasi *kinit* dengan memasukan *username* dan *password*, dan *user* mendapatkan otentikasi ke server yang dituju. Perancangan Kerberos ditujukan untuk memberika solusi bagi serangan – serangan seperti :

1. *Impersonation*, yaitu memaksakan untuk mendapatkan akses dengan menggunakan *username* dan *password* yang tidak mempunyai hak akses pada *service* dari jaringan.
2. *Eavesdropping*, yaitu menyadap data – data yang lalu lalang melintas pada jaringan.
3. *Tampering*, yaitu merubah data – data yang sudah diambil dari jaringan, setelah diubah data itu dikirimkan kembali ke tujuan.

2.4. Key Distribution Center (KDC)

Protokol Kerberos digunakan untuk mengotentikasi principal. Principal merupakan pihak yang identitasnya di verifikasi. Sebuah principal bisa berbentuk user biasa, sebuah aplikasi server atau sebuah entitas jaringan lainnya yang perlu di otentikasi. Pihak yang terlibat dalam proses otentikasi adalah :

1. *Client* yang biasanya merupakan *principal.*
2. *Server* yang merupakan sebagai verifikasi.
3. *Server* Kerberos KDC.

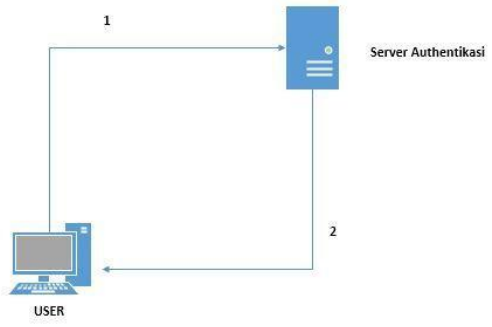
KDC ini merupakan server Kerberos yang bertugas mengantarkan *session key* kepada *server* dan *client* agar dapat melakukan koneksi. Mengotentikasi *principal* dan memberikan *session key* kepada *principal* oleh KDC melalui *Authentication Service* (AS), sedangkan untuk mempermudah *client* bekomunikasi dengan satu atau lebih server melalui *Ticket Granting Service* (TGS).

2.5. Sub Protokol pada Kerberos

Protokol Kerberos juga mempunyai 3 sub protokol untuk melakukan otentikasi, sub protokolnya sebagai berikut :

2.5.1 Authentication Service (AS)

Prosesnya sepeti pada gambar 2-2 *principal* (*client*) meminta sebuah tiket pada AS dengan mengirimkan namanya, dan selanjutnya AS menemukan *principal* itu ada dalam *database*-nya. Proses tersebut *password* atau kunci rahasia tidak berupa plainteks yang dikirimkan lewat jaringan. Kunci rahasia principal digunakan secara lokal oleh KDC untuk mengenkripsi tiket dan digunakan oleh *principal* untuk mendekripsi.



Gambar 2 Proses otentikasi

2.5.2 Ticket Granting Service (TGS)

Proses ini berlangsung saat *client* atau principal terotentikasi pada Kerberos maka *client* tersebut tidak dapat melakukan hubungan langsung dengan server yang dibutuhkan, tapi dia harus melakukan

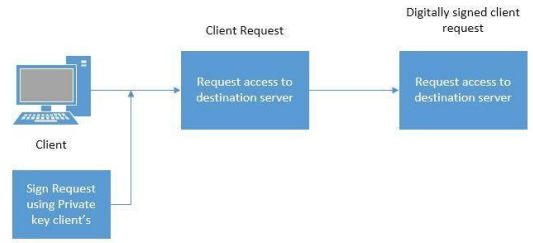
permintaan kepada KDC terlebih dahulu. Sebuah *authenticator* yang berisi *timestamp* dan checksum yang dienkripsi menggunakan *session key*. Pengiriman menggunakan enkripsi ini bertujuan untuk membuktikan identitas klien karena hanya dia yang mempunyai *session key*. *Checksum* berguna untuk menjaga informasi pesan saat pengiriman agar tidak berubah, dan timestamp berguna untuk melihat masa berlaku demi menghindari adanya gangguan.

2.5.3 Client/Server

Sub protokol ini berlangsung pada saat sebuah client digunakan untuk mengirimkan sebuah tiket sebagai pendaftaran kepada sebuah layanan.

2.6 Cara kerja otentikasi pada Kerberos

Pada saat *client* dengan server berkomunikasi otentikasi melakukan tugasnya sebagai pihak ketiga pada jaringan, berikut cara kerja otentikasi Kerberos yang menghubungkan antar client dengan server tujuan.



Gambar 3 Tahapan *client* mengirimkan permintaan

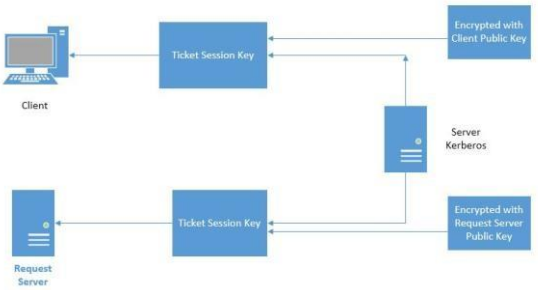
Pada gambar 2-3 diatas sebuah client mengirimkan permintaan kepada *Authentication Service*, dan meminta untuk menotentikasikan dirinya terhadap server yang di minta. Permintaan itu diberi *digital signature* atau tanda tangan digital si *client* menggunakan *private key client*.

Gambar 4 *client* mengirimkan otentikasi ke server kerberos

Selanjutnya, *client* meng-enkripsi *digitally signed* menggunakan *public key* dari server Kerberos, dan mengirimkannya permintaan yang sudah di enkripsi ke server Kerberos seperti gambar 2-4 diatas. Server Kerberos melakukan dekripsi *digitally signed* menggunakan *private key*-nya dan selanjutnya mengotentikasi *client digitally signed* dengan memverivikasi menggunakan *public key client* yang ada didalam *database* server Kerberos yang berisi semua *public key client* yang sah.

Jika server Kerberos tidak memiliki *public key client* pada *database*-nya, maka *digital signature* tidak dapat di verifikasi, dan juga *client* tersebut merupakan bukan *user* yang sah pada jaringan, dan request akan ditolak.

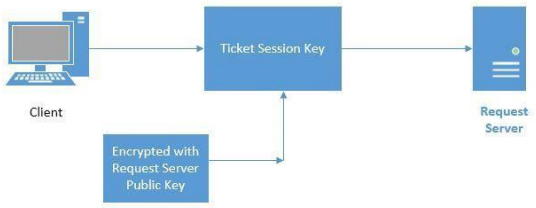
Jika server Kerberos menerima request dan mengotentikasi identitas pengirim *request*, maka server mem-verifikasi bahwa *client* tersebut memiliki otoritas yang sah untuk mengakses jaringan yg diminta.



Gambar 5 *Session ticket* diberikan server Kerberos

Jika Kerberos telah menentukan bahwa *client* memiliki otoritas untuk mengakses server yang diminta, maka server Kerberos akan mengirimkan *session ticket* yang sama baik kepada *client* maupun ke server yang diminta. Untuk mengirimkan *session ticket* kepada *client*, server Kerberos meng-enkripsi-nya menggunakan *public key* dari *client* dan sama seperti pengiriman ke server IGS, server Kerberos menggunakan *public key* server IGS.

Ketika menerima *encrypted session ticket*, baik client maupun server yang diminta akan men-dekripsi-nya menggunakan *private keys* masing-masing. *Session ticket* bisa di-tandatangani pula oleh server Kerberos untuk mencegah adanya *ticket* palsu yang dikirimkan ke *client* maupun ke server.



Gambar 6 C*lient* dengan server terhubung

Client kemudian mengirimkan salinan dari *ticket*-nya ke server. Sebelum mengirimkan *ticket*, *client* meng-enkripsi *ticket* menggunakan *public key* server IGS. Ketika menerima ticket yang di-enkripsi dari pilot, server akan men-dekripsi *ticket* menggunakan private key server . Server IGS kemudian membandingkan *ticket* yang diterima dari *client* dengan *ticket* yang berasal dari server Kerberos. Jika *ticket* sesuai (*match*) maka client akan diperbolehkan untuk terhubung ke server. Jika ticket tidak sesuai maka *client* akan ditolak . Setelah koneksi terbentuk, sistem dapat meng-enkripsi komunikasi menggunakan *session key* atau *public key* dari *client* atau tidak menggunakan enkripsi sama sekali.

Kerberos bukanlah merupakan sistem yang menjadi solusi untuk semua masalah keamanan jaringan. Dibawah ini merupakan kelebihan dan kelemahan Kerberos :

1. Kelebihan

Keunggulan utama yang dimiliki Kerberos adalah tingkat keamanannya yang tinggi. Username dan password tidak dikirimkan melintasi jaringan. Hal ini merupakan perbaikan dari sistem konvensional (password-based) yang rentan terhadap eavesdropping attack.

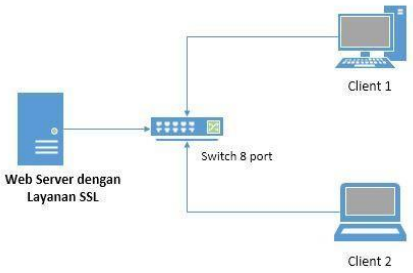
2. Kelemahan

Salah satu kelemahan Kerberos terletak pada lubang-lubang protokol Clock synchronization service. Trojan horse attack juga merupakan kelemahan dari Kerberos penyerang memodifikasi program username dan password.

3. Analisis dan Perancangan

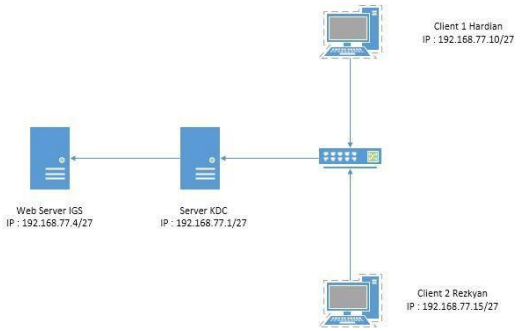
3.1. Gambaran Sistem saat ini

Usulan sistem yang sudah dirancang pada gambar 3.1 ini akan di implementasikan dengan menambahkan layanan SSL pada Web Server yang berfungsi untuk mengamankan data pada saat antar *client* dengan *web server* saling berkomunikasi.



Gambar 7 Rancangan topologi saat ini

Pada web server juga ditambahkan otentikasi menggunakan Kerberos untuk mengatasi permasalahan pada hak akeses client yang mengijinkan beberapa user untuk masuk kedalam IGS (*Integrated Ground System*).



Gambar 8 Rancangan topologi dengan kerberos

3.2. Kebutuhan Perangkat Keras

Untuk menerapkan rancangan topologi, digunakan perangkat keras dengan spesifikasi minimum seperti tabel 3-1 dibawah ini:

Tabel 1 Perangkat Keras

Jenis	Jumlah	Keterangan
Server	2	Interl Core i3; 4GB DDR3; 500GB HDD (Virtualisasi)

Client	2	Interl Core i3; 4GB DDR3; 500GB HDD (Virtualisasi)
PC *Tester*	1	Interl Core i3; 4GB DDR3; 500GB HDD
Switch	1	8 port

3.3. Kebutuhan Perangkat Lunak

Dalam pengerjaan proyek akhir ini, digunakan perangkat lunak dengan spesifikasi sebagai berikut :

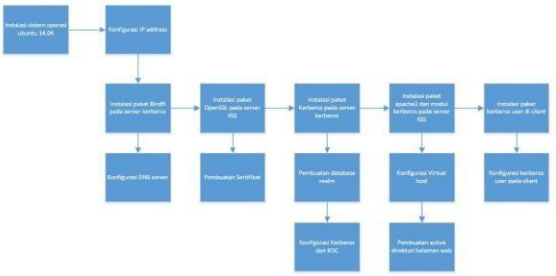
Tabel 2 Perangkat Lunak

Jenis	Versi	Keterangan
VMware	10.01	Aplikasi Virtualisasi
Ubuntu Server	14.04	Sistem Operasi Server KDC dan IGS
Ubuntu Server	14.04	Sistem Operasi Client
Windows	XP	Sistem Operasi Penyerang/Tester
OpenSSL	1.0.1	Layanan SSL
Kerberos	5	Layanan Otentikasi
Nmap	6.4	Aplikasi *Port Scanning*

3.4. Langkah Pengerjaan

Adapun tahap pengerjaan proyek akhir ini diantaranya :

1. Melakukan konfigurasi jaringan sesuai dengan topologi
2. Melakukan instalasi sistem operasi Ubuntu 14.04 dengan virtualisasi menggunakan VM Ware
3. Melakukan Instalasi SSL (*Secure Socket Layer*) dan autentikasi Kerberos pada sistem operasi 14.04.
4. Melakukan pengaksesan jaringan internal dari klien
5. Konfigurasi SSL dengan membuat sertifikat dan Kerberos untuk membatasi klien yang ingin mengakses IGS
6. Mendokumentasi semua tahapan yang telah dilakukan dan dijadikan laporan



Gambar 9 *Mapping* tahapan pengerjaan

3.5. Rencana Pengujian

Pengujian yang dilakukan meliputi :

1. Dilakukan proses ping antar jaringan internal sehingga terhubung satu dengan yang lain.
2. *Client* melakukan akses *live audio streaming*.
3. *Client* melakukan akses *shoutbox*.
4. *Client* melakukan akses *mp3 streaming*.

4. Pengujian

Adapun pengujian dari proyek akhir ini sebagai berikut.

1. Melakukan proses pengecekan jaringan internal sehingga terhubung satu dengan yang lain dengan Ping antar klien dengan *server*.

```
root@ubuntu:/home/hardian# ping 192.168.77.1
PING 192.168.77.1 (192.168.77.1) 56(84) bytes of data.
64 bytes from 192.168.77.1: icmp_seq=1 ttl=64 time=0.456 ms
64 bytes from 192.168.77.1: icmp_seq=2 ttl=64 time=0.285 ms
64 bytes from 192.168.77.1: icmp_seq=3 ttl=64 time=0.272 ms
^C
--- 192.168.77.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 1999ms
rtt min/avg/max/mdev = 0.272/0.337/0.456/0.086 ms
```

terhubung

2. Nslookup DNS server dari *client* maupun *server*.

```
root@ubuntu:/home/hardian# nslookup www.igs.
Server:         192.168.77.1
Address:        192.168.77.1#53

Name:   www.igs.com
Address: 192.168.77.4
```

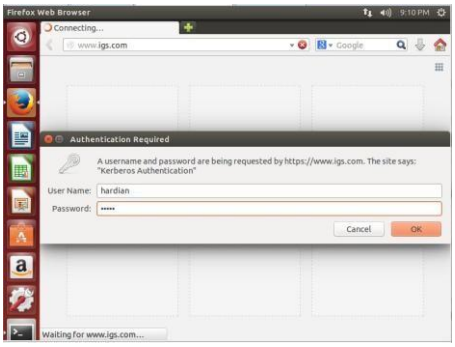
Gambar 11 Nslookup DNS Server dari client 1

3. Server KDC melakukan Pembuatan tiket pada *user*.

```
kadmin:  addprinc -clearpolicy -randkey hardian
Principal "hardian@IGS.COM" created.
kadmin:  cpw hardian
Enter password for principal "hardian@IGS.COM":
Re-enter password for principal "hardian@IGS.COM"
Password for "hardian@IGS.COM" changed.
```

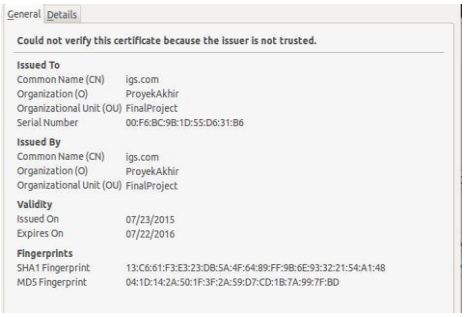
Gambar 12 Pembuatan hak akses pada *user*

4. Klien melakukan authentikasi/*log in* ke IGS.



Gambar 13 User melakukan akses ke IGS

5. Browser sisi client mendapatkan sertifikat SSL dari server.



Gambar 14 Sertifikat SSL pada *browser user*

6. Selanjutnya, penyerang melakukan port scanning pada bagian autentikasi pada saat sebuah klien mengakses IGS menggunakan Nmap.

```
443/tcp open  ssl/http Apache httpd 2.4.7 ((Ubuntu))
| http-auth:
| HTTP/1.1 401 Unauthorized

|_  Basic realm=Kerberos Authentication
|_http-methods: No Allow or Public header in OPTIONS
response (status code 401)
|_http-title: 401 Unauthorized
| ssl-cert: Subject: commonName=igs.com/
organizationName=ProyekAkhir/stateOrProvinceName=Jawa
Barat/countryName=ID
| Issuer: commonName=igs.com/
organizationName=ProyekAkhir/stateOrProvinceName=Jawa
Barat/countryName=ID
| Public Key type: rsa
| Public Key bits: 2048
| Not valid before: 2015-07-23T21:00:50+00:00
| Not valid after:  2016-07-22T21:00:50+00:00
| MD5:   041d 142a 501f 3f2a 59d7 cd1b 7a99 7fbd
|_SHA-1: 13c6 61f3 e323 db5a 4f64 89ff 9b6e 9332 2154
a148
|_ssl-date: 2014-03-12T14:02:15+00:00;
```

Gambar 15 Hasil Tester melakukan *Scanning Port*

5. Kesimpulan dan Saran

5.1. Kesimpulan

Dari penguian pada bab 4, dapat ditarik kesimpulan, yaitu:

1. SSL dan protokol Kerberos dapat di*implementasikan* sebagai metode pengamanan sebuah server IGS dari ancaman user yang tidak dikenal untuk masuk.
2. Fungsi SSL yang telah diterapkan membuat satu jalur komunikasi antar client dan server dengan melindungi informasi terdapat pada server IGS yang melintasi jalur tersebut dengan aman.
3. Protokol Kerberos dapat digunakan sebagai pengamanan server IGS dengan membatasi hak akses sebuah user dan juga sekaligus durasinya berdasarkan pengujian yang sudah dilakukan.
4. Setelah pengujian menggunakan Nmap dengan melakukan *port scanning* hanya informasi sertifikat yang terlihat.

5.2. Saran

Saran yang dapat penulis pada proyek akhir ini, yaitu:

1. Disarankan menambahkan sistem monitoring server untuk kedepannya, agar server lebih terpantau dan mudah untuk mencari dan menangani masalah yang terjadi pada server IGS.
2. Metode pengamanan SSL dan protokol Kerberos rentan terhadap virus Trojan saat ini, disarankan kedepannya dapat mencari metode lain agar aman dari virus Trojan.

Daftar Pustaka

[1] AgMAY Inc., “www.techiwarehouse.com,” 2010. [Online]. Available: http://www.techiwarehouse.com/engine/423a5281/IP-Spoofing-and-Sniffing-.

[2] A. Jumar, SSH (Secure Shell) dan SSL (Secure Socket Layer), 2003.

[3] I. Christian, "Sistem Otentikasi Kerberos pada Jaringan komputer ITB," Bandung, 2004.

[4] I. S. Host, "indositehist.com," 2015. [Online]. Available: klien.indositehost.com/knowledgebase.php?action=displayarticle&id=.

[5] Nasrullah, “www.blognazcules.com,” 2014. [Online]. Available: http://www.blognazcules.com/2014/06/mekanisme-otentikasi-pada-protokol-kerberos.html.

[6] W. Fakhri, A. R. Fauzan and I. Ahmadi, “Penerapan Kriptografi pada sitem otentikasi terpusat Kerberos v5,"Bandung, 2006.

[7] Tirtaksara, "www.bukalebar.com," 12 Januari 2015. [Online]. Available: http://www.bukalebar.com/2015/01/apa-itu-ssl-secure-sockets-layer.html.

[8] I. Prasetyo, "Secure Socket Layer (SSL)," 2013.

[9] F. Santosa, "Automation Web Testing Tool dengan Studi Kasus IGS Flight Focus PTE. LTD.," Bandung, 2013.

[10] MIT, “web.mit.edu,” 1985 – 2015. [Online]. Available: http://web.mit.edu/kerberos/.