IMPLEMENTASI SISTEM *SINGLE SIGN ON* (SSO) TERINTEGRASI ANTARA *CAPTIVE PORTAL*, STIKOM APPS DAN GOOGLE APPS DALAM JARINGAN *WIRELESS* STIKOM SURABAYA

Achmad Teguh Wibowo[(1)] Slamet[(2)] Hendra Darwintha[(3)] Satria Agung Pamuji[(4)]
1) Program Studi/Jurusan Sistem Informasi STIKOM Surabaya, email : atw@stikom.edu
2) Program Studi/Jurusan Sistem Informasi STIKOM Surabaya, email : slamet@stikom.edu
3) Program Studi/Jurusan Sistem Informasi STIKOM Surabaya, email : hendrad@stikom.edu
4) Jurusan Teknik Informatika Unitomo Surabaya, email : satria@stikom.edu

Abstract: Pemanfaatan *captive portal* dapat menjadi solusi untuk masalah pembatasan pengguna internet. Selain itu *captive portal* (chillispot) yang diimplementasikan dapat diubah tampilan dan programnya untuk pemanfaatan autentikasi internet. Teknologi ini juga dapat digabungkan dengan sistem *single sign on* (SSO) STIKOM Surabaya, yang mana telah terintegrasinya sebagian besar aplikasi milik STIKOM (STIKOM Apps) dan aplikasi buatan Google (Google Apps) bagi civitas akademika STIKOM Surabaya. Dengan adanya sistem ini, seluruh civitas memperoleh beberapa keuntungan diantaranya mengurangi *password fatigue* (kejenuhan terhadap password) berupa *username* dan *password* yang berbeda-beda dan mengurangi waktu yang terbuang karena memasukan *username* dan *password* yang sama berulang kali.

Keywords: *Captive Portal*, *Single Sign On* (SSO), STIKOM Apps, Google Apps

Pemanfaatan internet dalam infrastruktur WiFi di STIKOM Surabaya sudah mulai beralih dari sistem *proxy* (squid) menuju sistem *captive portal*. Permasalahan yang sering muncul pada saat mengimplementasikan infrastruktur WiFi adalah sistem login (*authentication*) pengguna. Sistem *proxy* bekerja dengan cara menyimpan *cache* dari suatu web dan menyimpannya di komputer sehingga pada saat komputer mengakses web yang sama untuk kali kedua dan seterusnya akan mengambil data dari komputer. Sistem seperti ini memiliki kekurangan yaitu pengguna akan melihat file yang kedaluarsa jika *cache expire time*-nya terlalu lama, pada saat website tersebut sudah berubah pengguna masih melihat file yang tersimpan di *cache memory* komputer.

Atas dasar permasalahan di atas dan telah terjalinnya kerja sama antara pihak STIKOM Surabaya dengan Google Inc. sehingga perlu dibuatkan suatu sistem yang dapat berkomunikasi antar aplikasi yang sudah ada di STIKOM Surabaya dengan beberapa produk Google Inc. diantaranya Google Mail, Google Drive, dan aplikasi lain yang masih dalam ruang lingkup Google Apps. Sistem yang dibuat menerapkan teknologi *single sign on*, dimana sistem ini memanfaatkan teknologi *captive portal* sebagai *authentication* awal. Pemanfaatan *captive portal* bukan hanya sebagai *authentication login* internet saja, tetapi digunakan juga sebagai sistem login di aplikasi STIKOM Apps (Sicyca, Digilib, dll) dan sistem login Google Apps (Google Mail, Google Drive, dll)

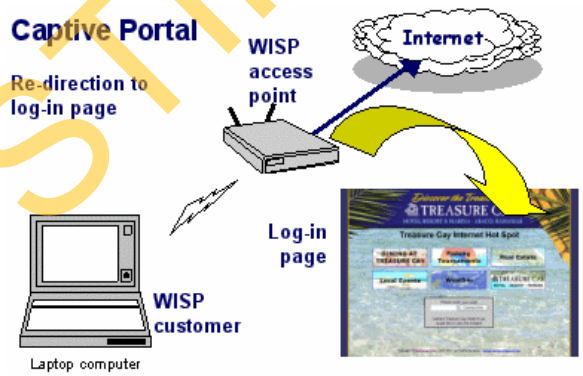
Ide dasar pemanfaatan *captive portal* yang digunakan dalam penelitian ini sebetulnya berasal dari keinginan untuk membuat jembatan antar aplikasi yang berbeda tempat atau *vendor* dalam hal ini STIKOM dan Google untuk dapat saling berkomunikasi.

Dalam penelitian ini, pemanfaatan *captive portal* berhasil diimplementasikan karena proses *authentication* yang digunakan berbasis web yaitu dengan bahasa CGI, sehingga nilai masukan yang diproses oleh aplikasi dapat ditangkap dan diubah menjadi suatu variabel *global* yang bisa dibaca oleh STIKOM Apps maupun Google Apps. Proses penangkapan variabel dari proses *authentication* menggunakan teknik *java script cookie* sehingga nilai

masukan yang berasal dari pengguna dalam hal ini *username* dan *password* dapat dikirim dan diproses menuju aplikasi yang sudah disiapkan sebelumnya. Dengan implementasi teknologi ini, hasil yang diperoleh adalah: mengurangi *password fatigue* (kejenuhan terhadap password) berupa *username* dan *password* yang berbeda-beda, mengurangi waktu yang terbuang karena memasukan *username* dan *password* yang sama berulang kali dan mengurangi IT *Costs* yang berkaitan dengan banyaknya pertanyaan mengenai *password.* Dengan demikian prinsip dasar interaksi antara manusia dan komputer (IMK) yang mengharapkan semudah dan seefisien mungkin dapat tercapai.

1. Captive Portal

Captive Portal merupakan suatu teknik authentikasi dan pengamanan data yang lewat dari *network* internal ke *network* eksternal. *Captive Portal* sebenarnya merupakan mesin router atau *gateway* yang memproteksi atau tidak mengizinkan adanya trafik, sampai pengguna melakukan registrasi terlebih dahulu ke dalam sistem. Biasanya *captive portal* ini digunakan pada infrastruktur *wireless* seperti area *hotspot*. Tetapi tidak menutup kemungkinan diterapkan pada jaringan kabel. Cara kerjanya adalah *user* dengan *wireless client*, diizinkan untuk terhubung dan mendapatkan IP *address* dari DHCP Server lalu mengarahkan semua trafik menuju *captive portal* untuk melakukan *authentication* berbasis web untuk mengijinkan *user* mendapatkan akses ke jaringan internet (hermawan, *et al*, 2012).



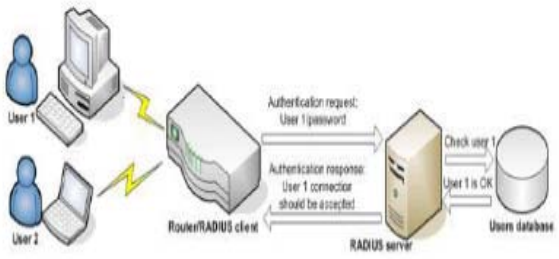
Gambar 1. Cara Kerja *Captive Portal* (Hermawan, *et al.*, 2012)

2. Remote Authentication Dial-In User Service (RADIUS)

RADIUS adalah sistem yang berfungsi untuk menyediakan mekanisme keamanan dan manajemen *user* pada jaringan komputer. Radius diterapkan jaringan dengan model client-server. RADIUS merupakan suatu protokol yang dikembangkan untuk proses *authentication, authorization, and accounting* (AAA).

Berikut ini adalah *Request For Comment* (RFC) yang berhubungan dengan RADIUS, (Edney dan William, 2003):

a. RFC2865:Remote Authentication Dial-In User Service (RADIUS)
b. RFC2866:RADIUS Accounting
c. RFC2867:RADIUS Accounting for Tunneling
d. RFC2868:RADIUS Authentication for Tunneling
e. RFC2869:RADIUS Extensions
f. RFC3162:RADIUS over IP6
g. RFC2548:Microsoft Vendor-Specific RADIUS Attributes

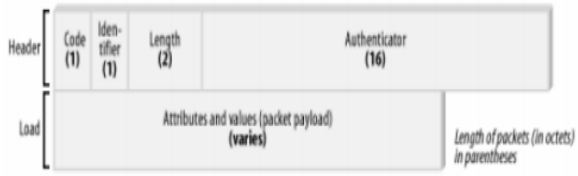


Gambar 2. InfraStruktur RADIUS (Hermawan, *et al.*, 2012)

Server Radius menyediakan mekanisme keamanan dengan menangani otentikasi dan otorisasi koneksi yang dilakukan *user*. Pada saat komputer *client* akan menghubungkan diri dengan jaringan maka server Radius akan meminta identitas *user* (username dan password) untuk kemudian dicocokkan dengan data yang ada dalam database server Radius untuk kemudian ditentukan apakah user diijinkan untuk menggunakan layanan dalam jaringan komputer. Jika proses otentikasi dan otorisasi berhasil maka proses pelaporan dilakukan, yakni dengan mencatat semua aktifitas koneksi user,

menghitung durasi waktu dan jumlah transfer data dilakukan oleh user. Proses pelaporan yang dilakukan server Radius bisa dalam bentuk waktu (detik, menit, jam, dll) maupun dalam bentuk besar transfer data (Byte, KByte, Mbyte) (Hermawan, *et al.*, 2012).

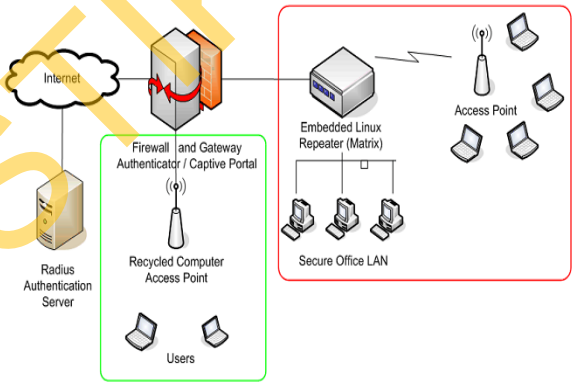
Pada awal pengembangannya, RADIUS menggunakan port 1645, yang ternyata bertubrukan dengan layanan datametrics. Sekarang, port yang dipakai RADIUS adalah port 1812. Gambar 3 menunjukan struktur paket data RADIUS.



Gambar 3. Struktur Paket Data RADIUS (Hassel, 2002)

3. Chillispot

ChilliSpot merupakan *open source* atau jalur akses *wireless* LAN *controller*. Digunakan untuk otentikasi pengguna *wireless* LAN yang mendukung *web login* berbasis *perl script* (*hotspotlogin* .cgi) yang merupakan enkripsi data *username* dan *password* dari *client*. Untuk melakukan otentikasi sebelum mendapatkan hak akses layanan *internet*, chillispot sangat besar peranannya dalam memvalidkan otentikasi tersebut. Beberapa layanan yang disediakan oleh chillispot atau yang biasa disebut sebagai *captive portal* ini adalah AAA. (Hermawan, *et al.*, 2012).



Gambar 4. Arsitektur Chillispot Dalam Sebuah Jaringan (Hermawan, *et al.*, 2012)

4. *Single Sign On*

Teknologi *Single-sign-on* (sering disingkat menjadi SSO) adalah teknologi yang mengizinkan pengguna jaringan agar dapat mengakses sumber daya dalam jaringan hanya dengan menggunakan satu akun pengguna saja. Teknologi ini sangat diminati, khususnya dalam jaringan yang sangat besar dan bersifat heterogen (di saat sistem operasi serta aplikasi yang digunakan oleh komputer adalah berasal dari banyak *vendor*, dan pengguna dimintai untuk mengisi informasi dirinya ke dalam setiap *platform* yang berbeda tersebut yang hendak diakses oleh pengguna). Dengan menggunakan SSO, seorang *user* hanya cukup melakukan proses autentikasi sekali saja untuk mendapatkan izin akses terhadap semua layanan yang terdapat di dalam jaringan (Pangestu, *et al.*, 2013).

5. Google Apps

Google Apps adalah salah satu seperangkat aplikasi Google yang menyatukan layanan penting untuk membantu bisnis Anda. Ini adalah sebuah layanan terpusat yang membuat bisnis, sekolah, dan institusi dapat memakai berbagai produk Google termasuk Google Email, Google Documents, Google Kalender, dan Google Talk pada sebuah domain yang anda miliki. (http://support.google.com, 2013).

6. Cookie

Cookie sering digunakan untuk mengidentifikasi pengguna. Cookie adalah file kecil yang ditanamkan server pada komputer pengguna (http://www.w3schools.com, 2013). Jenis cookie ada dua yaitu : cookie pihak pertama yang disetting oleh domain situs yang dikunjungi dan cookie pihak ketiga yang berasal dari sumber domain lain yang dimiliki contoh: iklan atau gambar yang tersemat di situs tersebut. (http://support.google.com, 2013). Secara garis besar struktur cookie tampak dalam gambar 5, dimana:

```
setcookie(name, value, expire, path, domain);
```

Gambar 5. Struktur Cookie

a. *Name* berfungsi untuk memberi nama cookie.
b. *Values* berfungsi untuk memberi nilai cookie.
c. *Expire* berfungsi untuk memberi batasan waktu hidup cookie
d. *Path* berfungsi untuk mengatur letak dimana cookie akan dijalankan.
e. *Domain* berfungsi untuk mengatur cookie hanya aktif di domain mana.

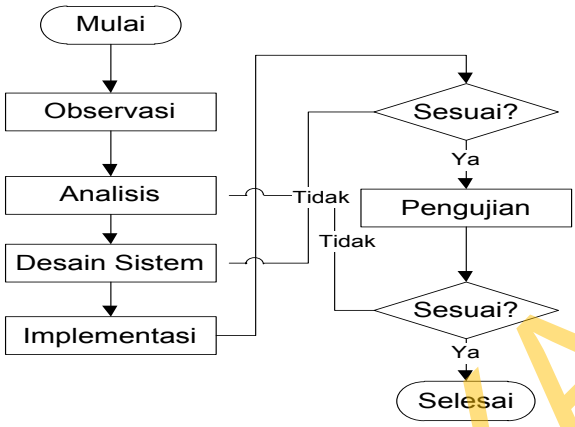
Pada dasarnya prinsip kerja cookie tidak membutuhkan lima variabel tersebut, cukup dengan 2 variabel *name* dan *values* cookie sudah dapat digunakan. Tetapi fungsi cookie yang paling penting adalah *expire* sehingga cookie sering digunakan menggunakan tiga variabel yaitu *name, values dan expire*.

7. Barkeley Internet Name Domain (BIND)

Berkeley Internet Name Domain (BIND) adalah nama program server DNS yang umum digunakan di Internet. Sejarah BIND sering diwarnai dengan celah-celah keamanan yang serius. Pernah beberapa kali ada versi BIND yang mampu dieksploitasi sedemikian hingga orang dari luar server bisa masuk ke dalam server sebagai root. Walaupun demikian, dengan konfigurasi yang benar, BIND dapat digunakan sebagai server DNS yang cepat, aman dan tangguh. Serial BIND yang terakhir dirilis adalah BIND seri 9.

8. METODE

Metodologi penelitian dalam penelitian implementasi sistem *single sign on* (SSO) terintegrasi antara *captive portal*, STIKOM *Apps* dan Google *Apps* dalam jaringan *Wireless* STIKOM Surabaya tampak dalam gambar 5.



Gambar 5. Metodologi Penelitian.

8.1 Alat dan Bahan

Alat dan bahan yang digunakan dalam penelitian ini adalah :

a. Perangkat Lunak

Perangkat lunak yang digunakan dalam penelitian ini adalah :

1. Ubuntu Server 12.04
2. Freeradius
3. Chillispot
4. MySQL
5. Bind9
6. Browser
7. Web Server (Apache, SSL, PHP)

b. Perangkat Keras:

1. Memory kapasitas 512 mb atau lebih
2. Hardisk 20 gb atau lebih
3. 2 lan card, kabel lan
4. Access point
5. Processor Intel Pentium IV dengan kecepatan 1.86 atau lebih.
6. *Mouse, Keyboard* dan *monitor* dalam keadaan baik.

8.2 Perancangan Sistem

Perancangan sistem dalam pelaksanaan penelitian ini. Terdiri dari beberapa tahap, yaitu:

Tahap Pertama:

Observasi: cara ini dilakukan terhadap objek secara langsung guna mendapatkan informasi dasar terhadap objek yang diteliti.

Tahap Kedua:

Analisis: semua data yang diperoleh melalui tahap studi literatur dan observasi, dikumpulkan dan diakuisisi menjadi pengetahuan dasar tentang sistema sso

Tahap Ketiga

Desain sistem: merancang sistem secara keseluruhan mulai dari antar muka, desain infrastruktur jaringan ,pengolahan input dan menghasilkan output yang sesuai dengan kebutuhan sistem.

Tahap Keempat

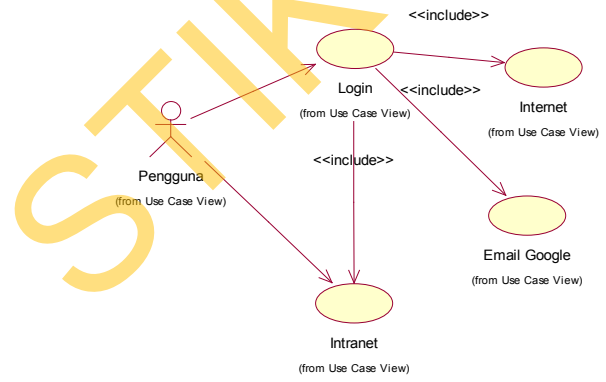
Implementasi: memindahkan hasil rancangan pada tahap sebelumnya kedalam sistem komputerisasi. Pada bagian ini, membuat aplikasi hasil dari rancangan desain sistem yang dibuat.

Tahap Kelima

Pengujian: dalam tahap ini dilakukan dengan beberapa tahap. Tahap pertama melakukan pengujian perangkat lunak dengan memasukkan data mahasiswa dengan benar, pengujian ke dua dengan memasukkan data mahasiswa yang tidak benar dan pengujian ke tiga menggunakan software netcut untuk mengetahui keamanan sistem SSO yang dibuat.

8.3 Diagram *Use Case Authentication*

Diagram use case dalam penelitian ini tampak dalam gambar 6.

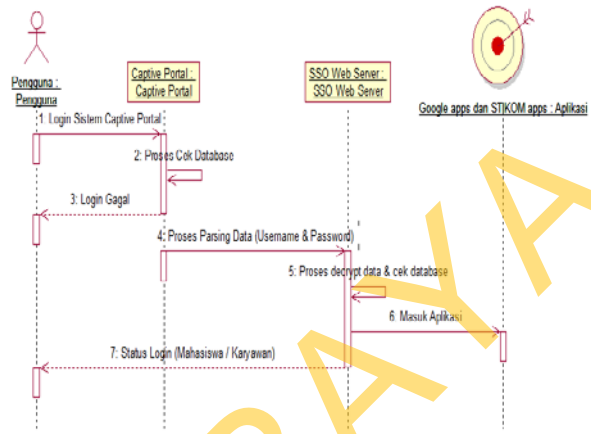


Gambar 6. Diagram *Use Case Authentication*.

8.4 *Sequence* Diagram Sistem SSO STIKOM

Sequence diagram dalam penelitian ini tampak dalam gambar 7.

x



Gambar 7. *Sequence* Diagram.

9. HASIL DAN PEMBAHASAN

9.1 Tampilan Login *Captive Portal*

Halaman login *captive portal* yang dibuat tampak dalam gambar 8.

Gambar 8. Halaman Login

Dalam gambar 8 terdapat *form* masukan berupa *username* dan *password*, dalam tampilan form login *captive portal* terdapat beberapa link yang langsung menuju ke aplikasi STIKOM tanpa harus login di sistem *captive portal.*

9.2 Uji Coba Dengan Menggunakan *Username* dan *Password* yang salah

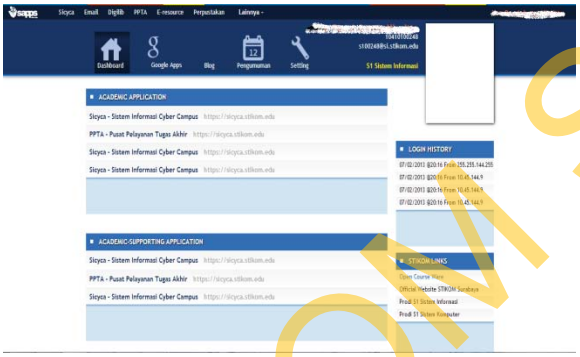
Halaman login *captive portal* menunjukkan *error*, tampak dalam gambar 9.

Gambar 9. *Error* Halaman Login

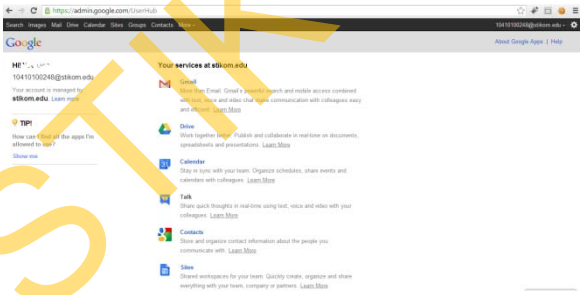
Dalam gambar 9 terdapat peringatan login gagal di bawah tampilan *textbox password*, karena data yang dimasukan kedalam *textbox username* dan *password* berupa data yang tidak tersedia di database.

9.3 Uji Coba Dengan Menggunakan *Username* dan *Password* Mahasiswa

Captive portal akan di*redirect* masuk ke aplikasi SSO. Tampilan SSO tampak dalam gambar 10.



Gambar 10. Tampilan SSO.

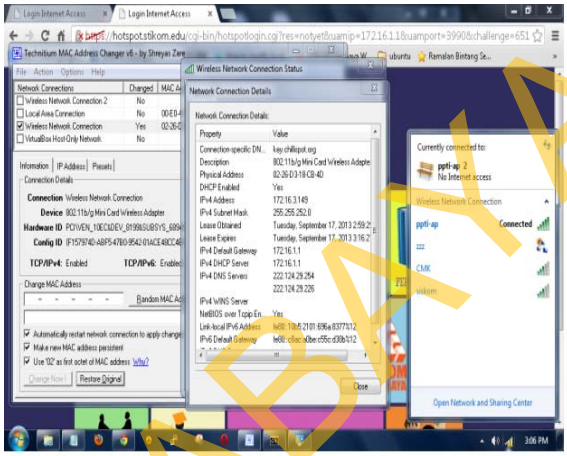


Gambar 11. Tampilan Google Apps

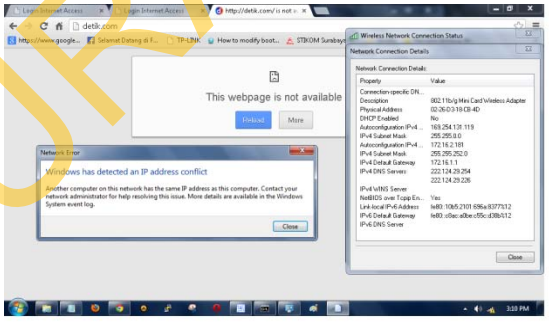
Dalam tampilan SSO terdapat *link* untuk menuju sicyca, digilib, google apps, dll. Link-link tersebut apabila ditekan akan langsung masuk ke dalam aplikasi tanpa harus login lagi.

9.4 Uji Coba Dengan Menggunakan *Software* Netcut

Pada pengujian menggunakan *software* netcut, hasil yang diperoleh tampak dalam gambar 12 dan 13.



Gambar 11. Konfigurasi IP Sesuai Netcut



Gambar 12. Hasil Uji Coba Menggunakan Netcut

Dalam uji coba menggunakan netcut dengan cara merubah ip dan *mac address* komputer yang digunakan. Tampak *captive portal* tidak mendeteksi ip yang digunakan.

10.KESIMPULAN DAN SARAN

10.1 Kesimpulan

Dari perancangan sistem, implementasi dan pengujian maka dapat diambil kesimpulan :

a. Sistem *captive portal* dapat diimplementasikan sebagai portal akses internet didalam infrastruktur WiFi.
b. Sistem *captive portal* dapat berkomunikasi dengan sistem SSO yang diimplementasikan melalui cookies javascript.

c. SSO yang diimplementasikan dapat langsung berkomunikasi dengan aplikasi dari pihak luar (Google Apps)

10.2 Saran

Berikut beberapa saran dalam penelitian ini :

a. Sistem *captive portal* dapat ditambahakan dengan Squid proxy agar dapat menghemat *bandwith.*Sebaiknya SSO
b. Dapat ditambahkan teknologi *Hierarchical Token Bucket* (HTB) sebagai pengatur *bandwith* untuk mahasiswa, karyawan, dosen dan tamu.
c. SSO yang dibuat sebaiknya tidak hanya tersinkronisasi dengan aplikasi Google apps saja, bisa ditambahkan dengan twitter outh, facebook dan lain-lain.

RUJUKAN

Barth, Adam, 2011, *HTTP State Management Mechanism*, University of California, Berkeley

Edney, Jon. Arbaugh, William A. 2003, *Real 802.11 Security: Wi-Fi Protected Access and 802.11i*. Addison Wesley.

Google Apps. https://support.google.com/accounts /answer/72709?hl=id. diakses tanggal 17/9/ 2013.

Hassel, Jonathan. 2002, *RADIUS*, O'Reilly.

Hermawan, D. K.. Sudarsono, A. Winarno, I 2012, *Implementasi Bandwith Management Captive Portal Pada Jaringan Wireless Di PENS-ITS*, Tugas Akhir tidak diterbitkan, Jurusan Statistika.

Kesan, Jay P., 2004, *Deconstructing Code*, Social Science Research Network Electronic Paper, University of Illinois College of Law

Pangestu, H. Periyadi. Deshanta, P. 2007, *Implementasi SSO (Single Sign On) Menggunakan Autentikas NCSA Untuk Website*, Open Jurnal Politeknik Telkom. Diakses tanggal 19/9/2013.