Implementasi Transport Layer Security pada LDAP

Ratdhian Cipta Sukmana
ratdix@yahoo.com
http://ratdix.wordpress.com

Pendahuluan

Artikel ini merupakan materi lanjutan dari "pengenalan LDAP" yang saya tulis sebelumnya, agar Anda dapat mengimplementasikan penggunaan TLS pada LDAP mengingat LDAP merupakan *layanan direktori* yang dapat dijadikan *Backend* bagi aplikasi maupun server lain sehingga jaminan dan keamanan transaksi untuk melayani *client request* dapat dilakukan oleh LDAP server.

TLS *(Transport Layer Security),* merupakan protokol yang menyediakan keamanan koneksi antara client dan server. TLS mempunyai kapabilitas untuk melayani servis autentikasi antara client dan server dengan membuat *encrypted connection* antara keduanya sehingga memberikan garansi integritas dan kerahasiaan pengiriman data. TLS sendiri memiliki dua *layers*, yaitu *TLS Record Protocol* dan *TLS Handshake Protocol*. *TLS Record Protokol* menyediakan keamanan koneksi dengan metode enkripsi seperti *Data Encryption Standart (DES),* walaupun begitu *TLS Record Protokol* dapat juga di gunakan tanpa enkripsi. Sedangkan *TLS Handshake Protokol* mengizinkan autentikasi antara client dan server yang akan bernegoisasi dengan menggunakan *cryptographic keys* sebelum data berpindah. TLS sendiri merupakan penyempurnaan dari protokol *SSL (Secure Sockets Layer).* Bagaimanakah cara protokol TLS bekerja sehingga dia menjamin bahwa pertukaran data dapat dijaga dengan aman……?

Konsep dasar

Sebenarnya ada empat ciri yang berbeda tentang sistem yang aman, yaitu:

- ***Privacy (Privasi)***
  Untuk memperoleh privasi, solusi keamanan harus memastikan bahwa tidak ada seorangpun yang dapat melihat, mengakses, atau menggunakan informasi privat yang di transmisikan melalui internet. Privasi juga akan memastikan bahwa hanya pengirim dan penerima yang sah yang dapat menggunakan informasi yang di pindahkan.
- ***Integrity (Integritas)***
  Integritas menjamin pendeteksian adanya perubahan informasi yang terjadi di antara waktu pengiriman dan penerimaan. Bila informasi di ubah dengan cara apapun selama transmisi berlangsung, maka sistem keamanan akan mendeteksi dan melaporkan perubahan ini sehingga sistem penerima akan meminta pengiriman informasi ulang.
- ***Otenticity (Otentisitas)***
  Otentikasi menyediakan aturan bagi pengguna untuk melakukan verifikasi bahwa mereka benar-benar berkomunikasi dengan pelaku yang sah dan memberikan jaminan bahwa semua pelaku dalam komunikasi adalah otentik.
- ***Non-Repudiation (Tidak terjadi penolakan)***
  Non-Repudiation menyediakan metode untuk menjamin bahwa tidak terjadi kesalahan dalam pertukaran informasi kepada pihak yang melakukan transaksi. Sehingga tanda tangan (Sign) maupun sertifikat digital mutlak di perlukan agar tidak terjadi penolakan.

Dengan adanya *username*, *password* dan *sertifikat digital* di lengkapi dengan protokol *SSL/TLS* maka dapat memenuhi sistem keamanan seperti diatas. Pada proses autentikasi, TLS client mengirimkan pesan kepada TLS server, dan server akan merespon dengan menanyakan informasi yang di butuhkan server untuk otentikasi, setelah itu client dan server akan melakukan perubahan pada *session keys* dan autentikasi dialog-pun berakhir. Apabila proses otentikasi gagal maka komunikasi antara client dan server tidak dapat berlanjut, tetapi apabila autentikasi berhasil, maka komunikasi yang aman menggunakan SSL/TLS dimulai dengan menanfaatkan *symmetric encryption keys* yang telah di tetapkan ketika proses autentikasi.

Instalasi

Untuk mengaktifkan protokol TLS pastikan paket OpenSSL telah terinstal pada mesin Anda. Setelah itu Anda harus membuat *server certificate* yang berfungsi untuk memastikan keamanan dan komunikasi yang telah terenkripsi. Sebaiknya Anda membuat satu direktori khusus untuk menyimpan sertifikat digital yang akan Anda buat.

- **Generating the Certificate Authority (CA)**

[root@ldap ~]# mkdir /myCA
[root@ldap ~]# cd /myCA
[root@ldap myCA]# /usr/share/ssl/misc/CA.pl -newca
CA certificate filename (or enter to create)

Making CA certificate ...
Generating a 1024 bit RSA private key
.++++++
...++++++
writing new private key to './demoCA/private/cakey.pem'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:

You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.

Country Name (2 letter code) [AU]:ID
State or Province Name (full name) [Some-State]:DKI
Locality Name (eg, city) []:JAKARTA
Organization Name (eg, company) [Internet Widgits Pty Ltd]:SMARTBEE Edu
Organizational Unit Name (eg, section) []:IT
Common Name (eg, YOUR name) []:ldap.smartbee.com
Email Address []:ratdhian.sukmana@smartbee.com

Catatan: Pastikan Anda menuliskan nama server LDAP anda secara lengkap (fully-qualified distinguished name of server) pada Common Name, dan PEM pass phase yang Anda ketik harus di ingat untuk menandai server certificate (SA).

Lihat Hasilnya
[root@ldap myCA]# ls -l
total 1
drwxr-xr-x 3 root root 72 Aug 1 21:53 .
drwxr-xr-x 25 root root 584 Aug 1 21:45 ..
drwxr-xr-x 6 root root 232 Aug 1 21:54 **demoCA**
[root@ldap myCA]# ls -l demoCA/
total 8
drwxr-xr-x 6 root root 232 Aug 1 21:54 .
drwxr-xr-x 3 root root 72 Aug 1 21:53 ..
-rw-r--r-- 1 root root 1310 Aug 1 21:54 **cacert.pem**
drwxr-xr-x 2 root root 48 Aug 1 21:53 certs
drwxr-xr-x 2 root root 48 Aug 1 21:53 crl

```
-rw-r--r--  1 root root   0  Aug  1 21:53 index.txt
drwxr-xr-x  2 root root  48  Aug  1 21:53 newcerts
drwxr-xr-x  2 root root  80  Aug  1 21:53 private
-rw-r--r--  1 root root  17  Aug  1 21:54 serial
```

- **Generating the Server Certificate (SA)**

[root@ldap myCA]# openssl req -new -nodes -keyout newreq.pem -out newreq.pem
Generating a 1024 bit RSA private key
...++++++
.++++++
writing new private key to '**newreq.pem**'

You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.

Country Name (2 letter code) [AU]:ID
State or Province Name (full name) [Some-State]:DKI
Locality Name (eg, city) []:JAKARTA
Organization Name (eg, company) [Internet Widgits Pty Ltd]:SMARTBEE Edu
Organizational Unit Name (eg, section) []:IT
Common Name (eg, YOUR name) []:ldap.smartbee.com
Email Address []:ratdhian.sukmana@smartbee.com

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
[root@ldap myCA]#

Catatan: Pastikan Anda menuliskan nama server LDAP anda secara lengkap (fully-qualified distinguished name of server) pada Common Name, dan sebaiknya chalenge password tidak perlu anda isi.

- **Sign the Certificate with the new CA**

[root@ldap myCA]# /usr/share/ssl/misc/CA.pl -sign
Using configuration from /etc/ssl/openssl.cnf
Enter pass phrase for ./demoCA/private/cakey.pem:
Check that the request matches the signature
Signature ok

Certificate Details:
Serial Number:
ad:df:c2:a8:a9:4e:0a:e5
Validity
Not Before: Aug 2 04:04:30 2006 GMT
Not After : Aug 2 04:04:30 2007 GMT
Subject:
countryName = ID
stateOrProvinceName = DKI
localityName = JAKARTA
organizationName = SMARTBEE Edu
organizationalUnitName = IT
commonName = ldap.smartbee.com
emailAddress = ratdhian.sukmana@smartbee.com
X509v3 extensions:
X509v3 Basic Constraints:
CA:FALSE
Netscape Comment:
OpenSSL Generated Certificate
X509v3 Subject Key Identifier:
11:EE:E3:FE:B2:73:F1:99:E9:A7:FE:48:89:21:87:BE:AF:22:C9:64
X509v3 Authority Key Identifier:
keyid:65:E5:67:32:B0:0D:AE:70:72:47:83:39:A4:35:5C:A5:01:FF:F6:63
DirName:/C=ID/ST=DKI/L=JAKARTA/O=SMARTBEE
Edu/OU=IT/CN=ldap.smartbee.com/emailAddress=ratdhian.sukmana@smartbee.com
serial:AD:DF:C2:A8:A9:4E:0A:E4

Certificate is to be certified until Aug 2 04:04:30 2007 GMT (365 days)
Sign the certificate? [y/n]:y

1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
Signed certificate is in **newcert.pem**
[root@ldap myCA]#

- ## Installing the Certificates on LDAP

Apabila anda telah menyelesaikan tahapan di atas, berarti anda telah berhasil membuat suatu *secure certificate* yang akan kita implementasikan ke LDAP.

[root@ldap myCA]# cp demoCA/cacert.pem /etc/openldap/
[root@ldap myCA]# cp newcert.pem /etc/openldap/servercrt.pem
[root@ldap myCA]# cp newreq.pem /etc/openldap/serverkey.pem
[root@ldap myCA]# chown ldap:ldap /etc/openldap/*.pem
[root@ldap myCA]# chmod 640 /etc/openldap/cacert.pem
[root@ldap myCA]# chmod 600 /etc/openldap/serverkey.pem

Pasang di file slapd.conf untuk standalone/master ldap
Bukalah file slapd.conf, lalu aktifkan/tambahkan ke tiga baris di bawah dan arahkan path ke direktori tempat anda menyimpan semua sertifikat

[root@ldap myCA]# vim /etc/openldap/slapd.conf

```
----------------------slapd.conf--------------------
TLSCACertificateFile /etc/openldap/cacert.pem
TLSCertificateFile /etc/openldap/servercrt.pem
TLSCertificateKeyFile /etc/openldap/serverkey.pem
----------------------cut file---------------------
```

Pasang di file ldap.conf untuk client/slave ldap
Bukalah file ldap.conf, lalu aktifkan/tambahkan ke tiga baris di bawah dan arahkan path ke direktori tempat anda menyimpan semua sertifikat

[root@ldap myCA]# vim /etc/ldap.conf

```
----------------------ldap.conf-----------------------
# CA certificates for server certificate verification
# At least one of these are required if tls_checkpeer is "yes"
tls_cacertfile /etc/openldap/cacert.pem
tls_cacertdir /etc/openldap
# Client certificate and key
# Use these, if your server requires client authentication.
tls_cert
tls_key
----------------------cut file----------------------
```

lakukan test anonymous user dengan menggunakan opsi –ZZ.
‘-ZZ’ flag mendorong kesuksesan *TLS handshake*, sedangkan apabila anda menggunakan ‘-Z’ flag akan mangaktifkan TLS dan memprosesnya tanpa menggunakan *encripted connection* apabila *TLS Handshake* gagal.

[root@ldap myCA]# ldapsearch -x -b "dc=smartbee,dc=com" -H 'ldap://ldap.smartbee.com:389' -ZZ

Referensi

- Soper, D.Kent — Open*LDAP Server With Server-Side SSL/TLS and Client Authentication*
- Weber, Steffo — *Securing LDAP Though TLS/SSL Sun Blue Prints, 2002*
- http://tldp.org — *LDAP Implementation HOWTO*
- http://wikipedia.org — *Transport Layer Security*
- www.elektroindonesia.com — *Keamanan Internet Berbasis WAP*

Biografi Penulis

Ratdhian Cipta Sukmana.
Mempelajari Ilmu Komputer berawal dari hobi, sejak SMU telah mengikuti pelatihan-pelatihan komputer hingga akhirnya dapat menyelesaikan S1 pada jurusan System Komputer Universitas Gunadarma Jakarta di akhir tahun 2001. Memulai karirnya sebagai Technical Support di beberapa perusahaan dan hingga kini masih aktif sebagai staff IT salah satu perusahaan Media di Jakarta. Sangat tertarik dengan Open Source dan Networking. Kopetensi inti pada bidang IT Support, Network Security, Administrator dan System Developer. Aktif di berbagai milis, dan selalu berusaha menggemakan konsep keterbukaan akan ilmu pengetahuan dengan semangat "Open Content". Berbagai artikel komputasi menarik lain yang di tuliskan berdasarkan pengalaman tersedia di situs blog *http://ratdix.wordpress.com*