

PERANCANGAN REPLIKASI BASIS DATA MYSQL DENGAN MEKANISME PENGAMANAN MENGGUNAKAN SSL ENCRYPTION

Herman Yuliansyah¹

¹Program Studi Teknik Informatika Universitas Ahmad Dahlan

Jl. Prof. Dr. Soepomo, Janturan, Yogyakarta 55164.

E-mail : herman.yuliansyah@tif.uad.ac.id

Abstrak

Karena faktor pentingnya data dan informasi pada organisasi small medium business (SMB), maka diperlukan suatu replikasi basis data sehingga jika terjadi suatu keadaan force majeure, organisasi tersebut tidak akan mengalami kehilangan data. Replikasi basis data ini melibatkan paling sedikit 2 sistem basis data yang terhubung dalam suatu jaringan komputer. Konsep CIA Triad yaitu Confidentiality, Integrity dan Availability, merupakan dasar keamanan untuk mengelola data dan informasi. Keterkaitan antara faktor replikasi data dan keamanan informasi sangat perlu diperhatikan untuk diteliti lebih lanjut sehingga diharapkan dapat meningkatkan faktor keamanan data dalam jaringan komputer.

Metodologi yang digunakan dalam penelitian ini yaitu dengan melakukan pengumpulan data terkait topik penelitian melalui studi pustaka, observasi, dan membuat pengujian di laboratorium dengan membuat simulasi replikasi basis data tanpa mekanisme pengamanan dan melakukan pengujian keamanan komunikasinya kemudian membandingkannya dengan replikasi data dengan mekanisme pengamanan dengan SSL serta melakukan pengujian keamanan komunikasinya melalui proses penyadapan paket data yang dikirimkan dari server master dan server slave.

Hasil dari penelitian ini dapat disimpulkan yaitu replikasi basis data MySQL dengan tanpa mekanisme pengamanan SSL memunculkan celah keamanan karena pesan yang dikirimkan dalam bentuk pesan plain tidak tersantikan. Sedangkan melalui penambahan dukungan SSL telah memberikan manfaat keamanan terhadap replikasi basis data MySQL, hal ini dibuktikan bahwa hasil proses penyadapan pesan terlihat paket data yang dikomunikasikan dari server master ke server slave sudah dalam keadaan terenkripsi.

Kata Kunci : *Replikasi basis data, MySQL, SSL Encryption.*

1. PENDAHULUAN

Menurut Stallings, ada tiga tujuan utama dari keamanan komputer yang konsep tersebut dikenal sebagai CIA Triad. CIA Triad tersebut adalah *Confidentiality, Integrity* dan *Availability*. *Confidentiality* mencakup dua konsep, yaitu: *Data confidentiality*, adalah memastikan bahwa informasi pribadi atau rahasia tidak dibuat untuk tersedia atau diungkapkan kepada individu yang tidak sah, dan *Privasi*, adalah memastikan bahwa kontrol individu atau pengaruh

informasi apa yang berkaitan dengan mereka dapat dikumpulkan dan di simpan dan oleh siapa dan kepada siapa informasi yang dapat diungkapkan. *Integrity*, mencakup dua konsep, yaitu: *Data Integrity*, adalah memastikan bahwa informasi dan program yang diubah hanya dengan cara tertentu dan berwenang. *System Integrity*, adalah memastikan bahwa sistem melakukan fungsi yang ditujukan secara tak terhalang, bebas dari manipulasi yang tidak sah disengaja atau tidak disengaja dari sistem. *Availability* yaitu meyakinkan bahwa sistem bekerja segera dan layanan tidak di tolak untuk pengguna yang berwenang[1]. Jika melihat dari konsep tersebut, nampak bahwa ketiga bertujuan sebagai keamanan mendasar untuk kedua data dan informasi dan layanan komputasi, sehingga dapat digunakan menjadi acuan untuk menghindarkan dari berbagai macam serangan yang ada.

Salah satu mekanisme serangan dalam jaringan komputer adalah dengan teknik *sniffing* atau penyadapan. Teknik ini dilakukan oleh seorang *hacker* dengan alat bantu yang di sebut *sniffer*. Menurut Kimberly Graves, *Sniffer* merupakan sebuah alat bantu yang digunakan untuk menangkap lalu lintas data yang dikirimkan antara dua sistem dan dapat menghasilkan banyak informasi, tergantung bagaimana *sniffer* itu digunakan langkah pengamanan sistem tersebut[2].

Sebuah basis data mempunyai fungsi untuk mengkoleksi banyak data. Sedangkan data didefinisikan sebagai deskripsi tentang benda, kejadian, aktivitas dan transaksi, yang tidak mempunyai makna atau tidak berpengaruh secara langsung ke pada pemakainya[3]. Sedangkan menurut McFadden, dkk dalam kadir, mendefinisikan informasi sebagai data yang telah diproses sedemikian rupa sehingga meningkatkan pengetahuan seseorang yang menggunakan data tersebut[3].

Saat ini sudah menjadi pengetahuan umum bahwa banyak sekali aplikasi pada organisasi *small medium business* (SMB) yang di bangun dengan menggunakan sistem basis data. Karena faktor pentingnya data dan informasi tersebut, maka diperlukan suatu mekanisme untuk mereplikasi atau menggandakan sistem basis data sehingga jika terjadi suatu keadaan yang termasuk ke dalam *force majeure*, organisasi tersebut tidak akan mengalami kehilangan data. Replikasi basis data ini melibatkan paling sedikit 2 sistem basis data yang terhubung dalam suatu jaringan komputer.

MySQL adalah sebuah *database manajemen system* (DBMS) populer yang memiliki fungsi sebagai *relational database manajemen system* (RDBMS). Selain itu MySQL *software* merupakan suatu aplikasi yang sifatnya *open source* serta *server* basis data MySQL memiliki kinerja sangat cepat, reliable, dan mudah untuk digunakan serta bekerja dengan arsitektur *client server* atau *embedded systems*[4]. Dikarenakan faktor *open source* dan populer tersebut maka cocok untuk mendemonstrasikan proses replikasi basis data.

Berkaitan dengan uraian di atas, maka peneliti bermaksud untuk melakukan penelitian yang berkaitan dengan keamanan sistem dan jaringan komputer dan replikasi basis data yang di buat dalam bentuk perancangan replikasi basis data MySQL dengan mekanisme pengamanan menggunakan *SSL Encryption* dengan tujuan untuk meningkatkan faktor keamanan data dalam jaringan komputer.

2. KAJIAN PUSTAKA

I Gde Budi Rinanta Putra melakukan penelitian tentang implementasi MySQL *Cluster* pada basis data terdistribusi menyatakan bahwa kegagalan sistem informasi ketika di akses oleh klien lebih banyak disebabkan karena pada sisi *server* terjadi *failure*. *Failure* ini dapat disebabkan oleh karena *server* mati dan tidak ada cadangan server lain yang dapat langsung menggantikan server utama yang mati tersebut. Sehingga solusi untuk mengatasi masalah diatas adalah dengan menggunakan teknologi MySQL *Cluster*. Pada teknologi MySQL *Cluster* terdapat replikasi *database* juga terdapat sistem yang mampu mengatasi *failure* sistem *database* itu sendiri. Oleh karena itu dengan implementasi MySQL *cluster* ini diharapkan sistem penyimpanan *database* itu bersifat *high availability*. Sehingga apabila terjadi sistem *failure* pada server utama bisa langsung digantikan dengan *secondary server*. Kesimpulan dari penelitian ini adalah MySQL *Cluster* sebagai sebuah solusi replikasi pada basis data terdistribusi dan MySQL *cluster* merupakan sebuah *database* yang menggunakan arsitektur *shared-nothing*[5].

Toga Aldila Cinderatama melakukan penelitian tentang basis data terdistribusi untuk aplikasi kependudukan berbasis web menyatakan bahwa implementasi *database* terdistribusi pada suatu sistem aplikasi dapat menghasilkan performansi yang baik menyangkut ketersediaan data. Adanya replikasi *database* yang dapat menghasilkan kesamaan posisi data pada beberapa *master site*, maka memungkinkan adanya pembagian beban dalam pengaksesan kerja *server*, sehingga kegagalan pengaksesan data dapat diminimalisasikan[6].

Dessyanto dalam penelitiannya menyebutkan bahwa IPsec (IP *Security*) dan SSL (*Secure Socket Layer*) merupakan teknik yang paling banyak digunakan untuk mengamankan komunikasi data melalui internet. Kedua teknik ini memiliki keunggulan dan kelemahan masing-masing. Tujuan dari penelitiannya adalah untuk menyajikan analisis terhadap kedua teknik di atas dalam segi keamanan dan kinerja, dan kesimpulannya adalah masing-masing protokol memiliki fitur unik. Pemilihan IPsec atau SSL tergantung pada kebutuhan keamanan data yang diperlukan. IPsec sangat cocok digunakan untuk komunikasi data antar *gateway*. SSL dapat bekerja dibelakang *firewall* dengan sangat baik jika dibandingkan dengan IPsec. Dalam implementasi IPsec, *client* perlu aplikasi IPsec khusus untuk *remote access*. Penggunaan kompresi data dalam jaringan dengan bandwidth rendah sangat menguntungkan dan hal ini terdapat dalam IPsec. IPsec memiliki kemampuan untuk memproteksi jaringan *wireless*[7].

Bhiogade melakukan penelitian tentang keamanan di Internet, dan penggunaan protokol SSL, serta penggunaan sertifikat untuk memenuhi permintaan untuk interaksi yang aman melalui Internet. Hasil dari penelitian ini adalah usaha untuk meyakinkan pengguna suatu domain bahwa pengguna tersebut tidak berisiko bila mengirim data melalui Internet, maka dapat terpenuhi apabila pengguna tersebut mendapatkan sertifikat. Jika pemilik situs memiliki lebih dari satu nama domain yang harus diamankan, maka pemilik situs harus memiliki lebih dari satu sertifikat. Sertifikat adalah nama domain dan nama *host* tertentu, jadi

pemilik situs akan membutuhkan sertifikat sebanyak pemilik situs memiliki nama domain. Jaminan membayar pada bisnis *e-commerce* akan mendapatkan keuntungan apabila SSL *web server* dalam keadaan diaktifkan dan memiliki sertifikat. SSL menyediakan kepercayaan terhadap integritas dan keamanan dalam bisnis *online* dan infrastruktur jaringan. Pelanggan semakin menyadari keuntungan dari keamanan SSL dan akan sering tidak membeli secara *online* dari toko yang tidak aman. Semua pemilik situs yang menggunakan keamanan SSL akan di dukung oleh jaminan yang kuat untuk mendorong pelanggan untuk membeli secara *online*[8].

3. METODE PENELITIAN

Di dalam melakukan penelitian ini, dilakukan cara-cara penelitian sebagai berikut:

3.1. Studi Pustaka

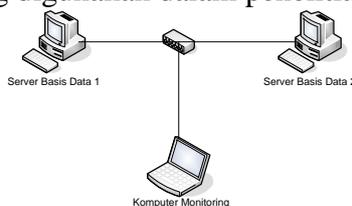
Merupakan pengumpulan data yang dilakukan dengan mencari, membaca dan mengumpulkan dokumen-dokumen sebagai referensi seperti buku, artikel dan literatur-literatur dan *browsing* di internet yang berhubungan dengan replikasi basis data dan SSL *encryption*.

3.2. Observasi

Merupakan pengumpulan data yang dilakukan dengan melihat langsung ke suatu pusat data (*data center*) di suatu organisasi yang telah menerapkan sistem *data center* terpadu.

3.3. Topologi Jaringan

Topologi jaringan yang digunakan dalam penelitian ini adalah sebagai berikut:



Gambar 1. Topologi jaringan dalam penelitian

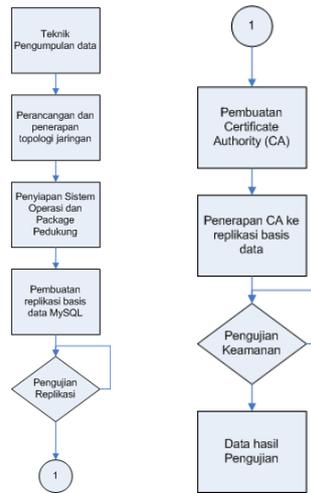
Gambar 1 menjelaskan tentang topologi jaringan yang akan digunakan sebagai sarana simulasi untuk penelitian ini. Terdiri dari 2 *server* basis data yang dikomunikasikan secara *host-to-host* dan diletakkan 1 buah komputer yang digunakan untuk memonitor, menganalisa dan mengevaluasi pertukaran data yang terjadi antara 2 *server* basis data tersebut.

Untuk pengalamatan dari topologi jaringan tersebut, *server* basis data 1 akan dikenakan pengalamatan dengan IP Address 192.168.1.1/24 dan server basis data 2 dengan IP Address 192.168.1.2/24 serta komputer monitoring dengan IP Address 192.168.1.3/24.

3.4. Jalannya Penelitian

Metode yang dilakukan dengan membuat simulasi suatu sistem jaringan di laboratorium jaringan komputer Teknik Informatika UAD, untuk merepresentasikan keadaan yang terjadi sesungguhnya di infrastruktur teknologi

informasi suatu organisasi. Di pengujian simulasi laboratorium ini akan ditempatkan 2 *server* basis data dan 1 buah komputer monitoring yang berguna untuk memantau dan menganalisa serta mengevaluasi pertukaran data antara 2 *server* basis data yang diujikan.



Gambar 2. Flow chart jalannya penelitian

Berdasarkan gambar 2, penelitian ini akan dilakukan dengan tahapan sebagai berikut:

- 1) Server basis data 1 dan server basis data 2 akan dilakukan instalasi sistem operasi linux dan aplikasi Apache, PHP dan MySQL serta openssl.
- 2) Pemasangan IP Address pada server basis data 1, server basis data 2 serta komputer monitoring, setelah itu dilakukan pengujian untuk memastikan bahwa ketiganya dapat saling terkoneksi dengan baik.
- 3) Melakukan tahapan proses replikasi basis data MySQL dengan mengatur bahwa server basis data 1 merupakan master replikasi dan server basis data 2 merupakan slave replikasi.
- 4) Menguji hasil replikasi dengan cara memasukkan data simultan di server basis data 1 dan melihat hasil replikasi di server basis data 2. Mekanisme memasukkan data secara simultan dilakukan dengan membuat perulangan dengan kode program PHP untuk memasukkan data tertentu dalam suatu tabel.
- 5) Membuat *certificate authority* (CA) dengan aplikasi openssl.
- 6) Mengimplementasikan CA tersebut ke replikasi basis data.
- 7) Melakukan pengujian di komputer monitoring dengan cara menyadap semua komunikasi yang terjadi antara server basis data 1 dan server basis data 2.

4. HASIL DAN PEMBAHASAN

4.1. Implementasi Replikasi Basis Data dengan MySQL Server

Implementasi replikasi basis data dengan MySQL server ini menggunakan topologi jaringan seperti pada gambar 1. Pada gambar 1 terlihat bahwa terdapat dua buah *server* yaitu *server* sebagai *master* replikasi dan sebagai *slave* replikasi. Tahapan yang dilakukan untuk mengimplementasikan replikasi basis data dengan MySQL *server* tersebut:

4.1.1. Instalasi MySQL pada Server Master dan Server Slave

Proses instalasi aplikasi MySQL dilakukan dengan langkah berikut ini pada *server master* dan *server slave*:

```
root@server1:~# aptitude install mysql-server mysql-client
```

4.1.2. Konfigurasi Server Master

Untuk memastikan bahwa replikasi dapat bekerja, maka harus MySQL harus dapat mendengarkan pada semua antarmuka pada *server master* (*server 1*). Hal ini dilakukan dengan memberi tanda komentar (#) pada baris *bind-address = 127.0.0.1* pada file */etc/mysql/my.cnf*.

Tahap selanjutnya dilakukan untuk membuat nama pengguna, basis data dan struktur tabel untuk *slave_user* yang akan digunakan oleh *server slave* (*server 2*) untuk mengakses aplikasi basis data MySQL pada server master (*server 1*).

```
root@server1:~# mysql -u root -p
mysql> GRANT REPLICATION SLAVE ON *.* TO 'slave_user'@'%'
IDENTIFIED BY 'slave_password';
mysql> FLUSH PRIVILEGES;
mysql> CREATE DATABASE exampledb;
mysql> USE exampledb;
mysql> CREATE TABLE `mahasiswa` (
  `nim` int(10) NOT NULL AUTO_INCREMENT,
  `nama` varchar(255) NOT NULL,
  `tanggal_lahir` date NOT NULL,
  PRIMARY KEY (`nim`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
mysql> quit;
```

Kemudian mem-*backup database* tersebut dan mengirimkannya ke *server slave*.

```
root@server1:~# cd /tmp
root@server1:/tmp# mysqldump -u root -pyourrootsqlpassword
--opt exampledb > snapshot.sql
root@server1:/tmp# scp snapshot.sql root@192.168.1.3:/tmp
```

4.1.3. Konfigurasi Server Slave

Sebelum memulai konfigurasi replikasi, dapat di buat terlebih dahulu sebuah *database* kosong dengan nama *exampledb* pada *server slave*.

```
root@server2:~# mysql -u root -p
mysql> CREATE DATABASE exampledb;
mysql> quit;
```

Pada *server slave*, dilakukan *impor* MySQL dump *snapshot.sql* dari *server master* sebelumnya.

```
root@server2:~# /usr/bin/mysqladmin --user=root --
password=yourrootsqlpassword stop-slave
root@server2:~# cd /tmp
root@server1:/tmp# mysql -u root -pyourrootsqlpassword
exampledb < snapshot.sql
```

Tahap selanjutnya yaitu menyiapkan akun yang digunakan untuk berkomunikasi dengan *server master*. Dilakukan dengan perintah berikut:

```
mysql> mysql -u root -p
mysql> CHANGE MASTER TO MASTER_HOST='192.168.1.2',
MASTER_USER='slave_user', MASTER_PASSWORD='slave_password',
MASTER_LOG_FILE='mysql-bin.000018', MASTER_LOG_POS=107;
mysql> START SLAVE;
mysql> quit;
```

4.2. Pengujian Replikasi Basis Data dengan MySQL Server

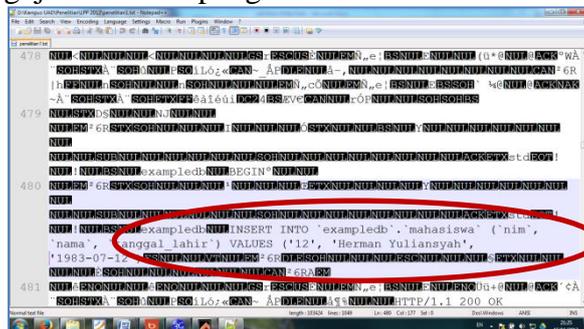
Pengujian replikasi ini dimaksudkan untuk memastikan hasil implementasi replikasi dapat berjalan sesuai dengan yang diharapkan. Skenario pengujian ini dilakukan dengan membuat suatu algoritma program array data yang tujuannya untuk melakukan proses *insert* data pada *database* di *server master*.

Untuk mengetahui hasil replikasi dapat berjalan dilakukan dengan melihat isi *database* di server *slave* dan memvalidasi bahwa isi *database* di *server master* akan sama dengan isi *database* di server *slave*.

4.3. Pengujian Keamanan Replikasi Basis Data dengan MySQL Server

Melalui penggunaan *tools* analisis paket data jaringan dengan aplikasi wireshark, dilakukanlah mekanisme *network forensics*. *Network forensics* merupakan kegiatan untuk melakukan pencarian data yang berhubungan dengan kejahatan di lingkungan jaringan komputer. *Network forensics* bukanlah sebuah produk, namun proses yang cukup kompleks dimana pengguna dapat mengumpulkan data dan menganalisa dan kemudian melakukan investigasi sesuai dengan kebutuhan[9]. Skenario pengujian ini membutuhkan sebuah komputer pasif tambahan seperti gambar 1 yang berguna untuk menganalisa semua komunikasi yang terjadi antara *server master* dan *server slave*. Skenario yang dilakukan adalah *server master* melakukan proses penyimpanan data ke dalam basis data internalnya dan kemudian pada saat yang bersamaan data yang disimpan tersebut dikirimkan kepada server *slave* serta komputer tambahan tersebut berfungsi untuk men-*snifing*/menyadap komunikasi yang terjadi.

Dari hasil pengujian ini di dapat gambaran 3 berikut:



Gambar 3. Hasil Pengujian Keamanan Komunikasi Replikasi Basis Data dengan MySQL Server

Gambar 3 menunjukkan bahwa komunikasi replikasi basis data MySQL tanpa mekanisme enkripsi tertentu memiliki celah karena pesan yang dikirimkan dapat terbaca oleh pihak ketiga.

4.4. Implementasi Replikasi Basis Data MySQL Server Dengan SSL

4.4.1. Pembuatan *certificate authority* SSL

Untuk dapat mengimplementasikan replikasi basis data MySQL dengan SSL *encryption*, langkah awal yang dapat dilakukan adalah dengan membuat *certificate authority* (CA). Berikut langkah yang dapat dilakukan:

```
root@server1:~# openssl genrsa 2048 > ca-key.pem
root@server1:~# openssl req -new -x509 -nodes -days 1000 -key ca-key.pem > ca-cert.pem
root@server1:~# openssl req -newkey rsa:2048 -days 1000 -nodes -keyout server-key.pem > server-req.pem
root@server1:~# openssl x509 -req -in server-req.pem -days 1000 -CA ca-cert.pem -CAkey ca-key.pem -set_serial 01 > server-cert.pem
root@server1:~# openssl req -newkey rsa:2048 -days 1000 -nodes -keyout client-key.pem > client-req.pem
```

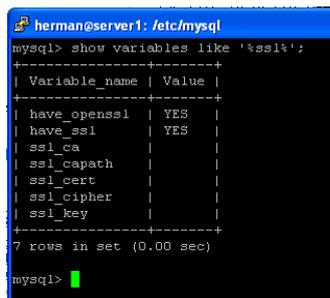
```
root@server1:~# openssl x509 -req -in client-req.pem -days
1000 -CA ca-cert.pem -CAkey ca-key.pem -set_serial 01 >
client-cert.pem
```

4.4.2. Mengaktifkan Dukungan SSL Untuk Server Master dan Slave

Pengaktifan SSL tersebut dapat dilakukan dengan melakukan perubahan terhadap file `/etc/mysql/my.cnf`. Setelah menambahkan kata SSL, maka untuk mengaplikasikan perubahan konfigurasi tersebut, basis data MySQL dapat di muat ulang dengan perintah `/etc/init.d/mysql restart`. Untuk menguji hasil konfigurasi tersebut dapat dilakukan dengan perintah berikut pada terminal MySQL:

```
mysql> show variables like '%ssl%';
```

Hasil pengujian tersebut dapat di lihat pada gambar 4, terlihat nama *variable* `have_openssl` dan `have_ssl` nilainya berubah menjadi `YES` seperti pada gambar 4.



```
herman@server1: /etc/mysql
mysql> show variables like '%ssl%';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| have_openssl  | YES   |
| have_ssl      | YES   |
| ssl_ca        |       |
| ssl_capath    |       |
| ssl_cert      |       |
| ssl_cipher    |       |
| ssl_key       |       |
+-----+-----+
7 rows in set (0.00 sec)

mysql>
```

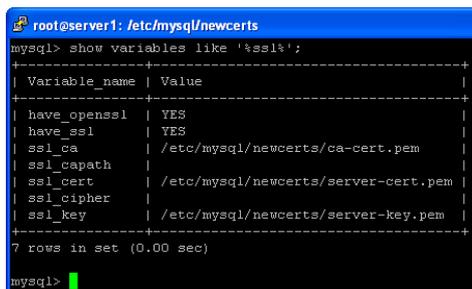
Gambar 4. MySQL Telah Memiliki Dukungan Terhadap SSL

4.4.3. Konfigurasi *certificate authority* SSL Untuk Server Master

Konfigurasi CA SSL untuk server master dilakukan dengan melakukan editing terhadap file `/etc/mysql/my.cnf`. Pada file ini yang dapat dilakukan adalah mengilangkan tanda komentar pada baris `ssl-ca`, `ssl-cert`, dan `ssl-key` serta mengisi dengan nilai yang sesuai yaitu nilai lokasi tempat file CA yang telah dibuat sebelumnya. Terminal MySQL dapat ditambahkan sebuah *replication user* `slave_user` yang dapat digunakan oleh *server slave* untuk mengakses basis data MySQL pada *server master*. Penambahan user ini dilakukan dengan perintah berikut:

```
root@server1:/tmp# mysql -u root -p
mysql> GRANT REPLICATION SLAVE ON *.* TO 'slave_user'@'%'
IDENTIFIED BY 'slave_password' REQUIRE SSL;
mysql> FLUSH PRIVILEGES;
mysql> quit;
```

Hasil dari konfigurasi ini dapat diujikan dengan perintah `show variables like '%ssl%'` pada terminal MySQL sehingga hasilnya akan tampak seperti pada gambar 5.



```
root@server1: /etc/mysql/newcerts
mysql> show variables like '%ssl%';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| have_openssl  | YES   |
| have_ssl      | YES   |
| ssl_ca        | /etc/mysql/newcerts/ca-cert.pem |
| ssl_capath    | /etc/mysql/newcerts/server-cert.pem |
| ssl_cert      | /etc/mysql/newcerts/server-cert.pem |
| ssl_cipher    |       |
| ssl_key       | /etc/mysql/newcerts/server-key.pem |
+-----+-----+
7 rows in set (0.00 sec)

mysql>
```

Gambar 5. Hasil Konfigurasi *certificate authority* SSL Untuk *Server Master*

4.4.4. Konfigurasi *certificate authority* SSL Untuk *Server Slave*

Konfigurasi CA SSL untuk server slave dilakukan dengan melakukan perubahan terhadap file `/etc/mysql/my.cnf`. Pada file ini yang dapat dilakukan adalah menghilangkan tanda komentar pada baris `ssl-ca`, `ssl-cert`, dan `ssl-key` serta mengisi dengan nilai yang sesuai yaitu nilai lokasi tempat file CA yang telah dibuat sebelumnya. Hasil dari konfigurasi ini dapat diujikan dengan perintah `show variables like '%ssl%'` pada terminal MySQL sehingga hasilnya akan tampak seperti pada gambar 6.

```
herman@server2: /etc/mysql/newcerts
mysql> show variables like '%ssl%';
+-----+-----+
| Variable name | Value |
+-----+-----+
| have_openssl  | YES   |
| have_ssl      | YES   |
| ssl_ca        | /etc/mysql/newcerts/ca-cert.pem |
| ssl_capath    |       |
| ssl_cert      | /etc/mysql/newcerts/client-cert.pem |
| ssl_cipher    |       |
| ssl_key       | /etc/mysql/newcerts/client-key.pem |
+-----+-----+
7 rows in set (0.00 sec)

mysql>
```

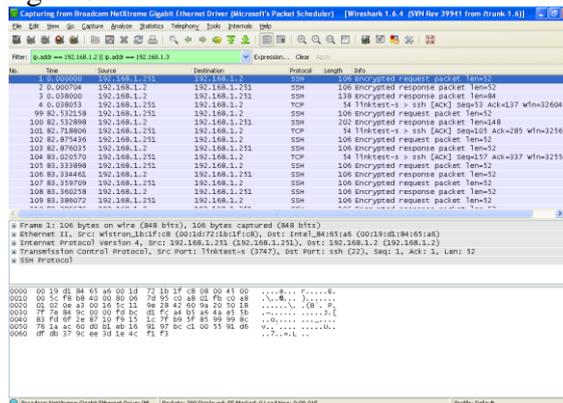
Gambar 6. Hasil Konfigurasi *certificate authority* SSL Untuk *Server Slave*

Setelah *server slave* mendapat dukungan terhadap SSL, maka untuk mengkoneksikan *server slave* dengan *server master* dilakukan dengan perintah berikut pada terminal MySQL:

```
root@server1:/tmp# mysql -u root -p
mysql> CHANGE MASTER TO MASTER_HOST='192.168.1.2',
MASTER_USER='slave_user', MASTER_PASSWORD='slave_password',
MASTER_LOG_FILE='mysql-bin.000001', MASTER_LOG_POS=3096424,
MASTER_SSL=1, MASTER_SSL_CA = '/etc/mysql/newcerts/ca-cert.pem',
MASTER_SSL_CERT = '/etc/mysql/newcerts/client-cert.pem',
MASTER_SSL_KEY = '/etc/mysql/newcerts/client-key.pem';
mysql> START SLAVE;
mysql> quit;
```

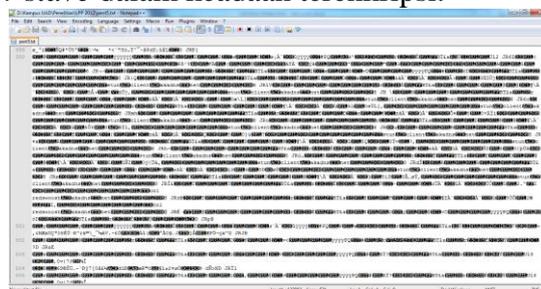
4.5. Pengujian Keamanan Replikasi Basis Data MySQL Server Dengan SSL

Pengujian keamanan komunikasi replikasi basis data MySQL *Server* dengan SSL dilakukan sama seperti pengujian replikasi tanpa SSL yaitu dengan menempatkan satu komputer pasif penyadap komunikasi dari *server* dengan skenario seperti gambar 1.



Gambar 7. Penggunaan Tools Analisis Paket Data Jaringan Dengan Aplikasi Wireshark.

Gambar 8 menunjukkan gambaran tentang proses penyadapan paket data pada komunikasi replikasi basis data dari *server master* ke *server slave*. Hasil analisa data ini disimpan dalam sebuah file log seperti gambar 9. Berdasarkan gambar 9 terlihat bahwa seluruh isi dari file log berisikan data dalam bentuk terenkripsi. Dari gambar 3 tidak ditemukan lagi query SQL *insert* seperti yang terjadi pada gambar 3. Dengan demikian dari hasil penerapan replikasi basis data MySQL dengan SSL ini menunjukkan bahwa keamanan komunikasi data telah dapat ditingkatkan, terbukti bahwa paket data yang dikomunikasikan dari *server master* ke *server slave* dalam keadaan terenkripsi.



Gambar 9. Hasil Log Penyadapan Komunikasi Replikasi Basis Data Dengan SSL.

5. KESIMPULAN

Berdasarkan hasil penelitian dan pembahasan, maka dapat disimpulkan beberapa hal-hal berikut:

1. Telah berhasil dilakukan replikasi basis data MySQL dengan dua buah server yaitu server master dan server slave, baik menggunakan mekanisme pengamanan SSL maupun tanpa pengamanan SSL.
2. Replikasi basis data MySQL dengan tanpa mekanisme pengamanan SSL memunculkan celah tersendiri yaitu hasil komunikasi paket datanya dapat di sadap dan pesan yang dikirimkan dalam bentuk pesan plain tidak tersantikan.
3. Penambahan dukungan SSL telah memberikan manfaat keamanan terhadap replikasi basis data MySQL, hal ini dibuktikan bahwa hasil proses penyadapan pesan terlihat paket data yang dikomunikasikan dari server master ke server slave sudah dalam keadaan terenkripsi

DAFTAR PUSTAKA

[1] W. Stallings, *Cryptography and Network Security Principles and Practice* United States of America: Prentice Hall, 2011.

[2] K. Graves, *CEH : certified ethical hacker study guide*. Indianapolis: Wiley Publishing, Inc., 2010.

[3] A. Kadir, *Pengenalan Sistem Informasi*. Yogyakarta: Penerbit ANDI, 2003.

[4] MySQL. (2012, 15 Oktober). *Why MySQL?* Available: <http://www.mysql.com/why-mysql/>

[5] I. G. B. R. Putra, "Implemetasi MySQL Cluster Pada Basis Data Terdistribusi," *Jurnal Elektronik Ilmu Komputer Universitas Udayana*, vol. Volume 1, pp. 11-20, 2012.

[6] T. A. Cinderatama, W. Yuwono, and R. Asmara, "Basis Data Terditribusi Untuk Aplikasi Kependudukan Berbasis Web," ed: <http://repo.eepis-its.edu>, 2011.

[7] B. P. Dessyanto. (2012, 15 November). *PERBANDINGAN KINERJA IP SEC DAN SSL*. Available: <http://repository.upnyk.ac.id/1971/>

- [8] M. Bhigade, "Secure Socket Layer," *Informing Science + IT Education Conference* pp. 0085-0090, 2002.
- [9] A. Kurniawan, *Network Forensics Panduan Analisis dan Investigasi Paket Data Jaringan Menggunakan Wireshark*. Yogyakarta: Penerbit Andi, 2012.