Instituto Tecnologico Superior de Coatzacoalcos

IV Semana Academica y Cultural

Seguridad en Base de Datos

Farid Alfredo Bielma Lopez

fbielma@fbielma.org

http://www.fbielma.org/talks/

Agenda

- Introduccion
- Estructura de MySQL/PostgreSQL
- Seguridad en MySQL/PostgreSQL
- Copias de seguridad
- Importacion de Datos
- Mantenimiento de Bases de Datos

Introducción

Al tratar el tema de la seguridad en Base de Datos, es importante considerar la necesidad de proteger totalmente la máquina completa contra todos los tipos de ataques posibles: intercepción pasiva de paquetes, reproducción de comandos, y denegación de servicio.

Por ejemplo...

Intente escanear sus puertos desde Internet utilizando una herramienta como nmap. MySQL utiliza el puerto 3306 por defecto y PosgreSQL el 5432. Estos puertos no debería ser accesible desde lugares no confiables.

```
shell>telnet server_host 3306

shell>telnet server_host 5432
```

Consejos Generales

No transmita datos sin cifrar por Internet. Esta información es accesible para cualquiera que tenga el tiempo y la habilidad para interceptarla y utilizarla para sus propios propósitos. En vez de eso, utilice un protocolo de cifrado como SSL o SSH. MySQL soporta conexiones SSL internas desde la versión 4.0.0. El redireccionamiento de puertos de SSH se puede utilizar para crear un tunel cifrado (y comprimido) para la comunicación.

A poco es muy dificil...?

```
shell> tcpdump -l -i eth0 -w - src or dst port 3306 |
  strings
```

Estructura de MySQL

Los directorios /include y /lib contiene los fichero *.h y las librerias necesarias, en /bin estan los ficheros ejecutables y en /data encontraremos como subdirectorio cada una de las bases de datos que hayamos creado.

Estructura de MySQL

Por cada tabla que definamos MySQL va ha crear tres archivos: **mitabla.ISD, mitabla.ISM, mitabla.frm.**

El sistema de permisos MySQL lo guarda en una base de datos llamada mysql, la cuál se componen de cinco tablas: host, user, db, tables_priv, colums_priv.

La Tabla User

CAMPO	TIPO	POR DEFECTO
Host	char(60)	
User	char(16)	
Password	char(16)	
Select_priv	enum('N','Y')	N
Insert_priv	enum('N','Y')	N
Update_priv	enum('N','Y')	N
Delete_priv	enum('N','Y')	N
Create_priv	enum('N','Y')	N
Drop_priv	enum('N','Y')	N
Reload_priv	enum('N','Y')	N
Shutdown_priv	enum('N','Y')	N
File_priv	enum('N','Y')	N
Grant_priv	enum('N','Y')	N
References_priv	enum('N','Y')	N
Alter_priv	enum('N','Y')	N

Recomendaciones en MySQL

- No dé nunca a nadie (excepto a la cuenta root de MySQL acceso a la tabla User en la base de datos mysql) Esto es crítico. La clave cifrada es la verdadera clave en MySQL.
- Estudie el sistema de privilegios de acceso de MySQL. Las sentencias GRANT y REVOKE se utilizan para controlar el acceso a MySQL. No otorgue más privilegios de los necesarios.

Recomendaciones en MySQL

- No elija claves que puedan aparecer en un diccionario. Existen programas especiales para romperlas. Incluso claves como ``perro98'' son muy malas. Es mucho mejor ``oweei98''.
- Invierta en un firewall. Le protegerá de al menos el 50% de todos los tipos de vulnerabilidades de cualquier software. Ponga MySQL tras el firewall o en una zona desmilitarizada (DMZ).

Recomendaciones en MySQL

- Intente escanear sus puertos desde Internet utilizando una herramienta como nmap. MySQL utiliza el puerto 3306 por defecto.
- Probar si el puerte MySQL está abierto, intente el siguiente comando desde alguna máquina remota, donde su servidor MySQL se estÃ¡ ejecutando:

```
shell> telnet server_host 3306
```

Seguridad en la Base de Datos

Pruebe el comando mysql -u root. Si es capaz de conectar al servidor sin la necesidad de introducir una clave, tiene problemas.

Cualquier persona podrá conectar a su servidor MySQL como el usuario root de MySQL con privilegios totales!

Asignar Password de Root

Luego de instalar MySQL en Unix, se necesita inicializar las tablas de permisos, ejecutar el servidor, y asegurarse de que éste funciona correctamente.

Las tablas de permisos se configuran mediante el programa mysql_install_db.

Asignar Password de Root

Con mysqladmin:

```
mysqladmin -u root password 'newpass`
```

Con SET PASSWORD:

```
shell> mysql -u root

mysql> SET PASSWORD FOR ''@'localhost'
= PASSWORD('newpwd');
```

Asignar Password de Root

Con la sentencia UPDATE:

```
shell> mysql -u root

mysql> UPDATE mysql.user SET Password
= PASSWORD('newpwd')WHERE User = '';

mysql> FLUSH PRIVILEGES;
```

La Sentencia GRANT

Utilice la sentencia SHOW GRANTS y compruebe quien tiene acceso a qué recurso. Después utilice la sentencia REVOKE para denegar los privilegios que no son necesarios.

Uso de Grant:

```
mysql> GRANT ALL PRIVILEGES ON test.*  TO
'root'@'localhost' IDENTIFIED BY 'goodsecret';
```

Carácteres de Escape

Muchas interfaces de programación de aplicaciones proveen alguna manera de preceder con caracteres de escape en sus datos, y entonces se pueden generar estas consultas:

```
``; DROP DATABASE mysql;''
```

```
SELECT * FROM table WHERE ID=234 OR 1=1
```

Carácteres de Escape

Apis de Programación:

- API MySQL de C: Utilice la función **mysql_real_escape_string()**.
- PHP: Utilice la función **mysql_escape_string()**. Con versiones anteriores a PHP 4.0.3, utilice **addslashes()**. En PHP 5, puede utilizar la extensión **mysqli**, que soporta los protocolo de autentificación y clave de acceso mejorados de MySQL, así como las sentencias preparadas con placeholders.
- DBI de Perl: Utilice el método **quote()** o utilice placeholders.
- JDBC de Java: Utilice un objeto **PreparedStatement** y placeholders.

Sentencia LOAD DATA

Hay dos aspectos de seguridad potenciales al soportar la versión LOCAL de los comandos LOAD DATA:

- La transferencia del fichero desde el equipo cliente al equipo servidor se inicia mediante el servidor MySQL. En teoría, puede construirse un servidor modificado de forma que le diga al programa cliente que transfiera un fichero elegido por el servidor en lugar de el fichero especificado por el cliente en el comando LOAD DATA .
- En un entorno Web en el que los clientes se conecten mediante un servidor Web, un usuario podría usar LOAD DATA LOCAL para leer cualquier fichero al que el servidor Web tuviese acceso de lectura (asumiendo que el usuario puediese ejecutar cualquier comando contra el servidor SQL).

Sentencia LOAD DATA

- Puede desactivar todos los comandos **LOAD DATA LOCAL** desde el lado del servidor arrancando **mysqld** con la opción **--local-infile=0.**

- Para el cliente de línea de comando **mysql**, **LOAD DATA LOCAL** puede activarse especificando la opción **--local-infile[=1]** , o deshabilitarse con la opción **--local-infile=0**

Encriptación

No almacene ninguna clave sin cifrar en su base de datos. Si alguien tuviera acceso a su ordenador, el intruso podría obtener la lista completa de claves y utilizarlas. En vez de eso, utilice **MD5()**, **SHA1()**, o cualquier otra función de hashing de un sentido.

Copias de Seguridad

El cliente mysqldump es un programa de respaldo de base de datos, pueden tranferiste los datos a otro servidor SQL (no necesariamente MySQL).

El respaldo tipicamente contiene sentencias SQL para crear la tabla y su contenido.

mysqldump

Para tener una lista de las opciones que soporta el programa ejecuta:
mysqldump --help

Forma de Uso:

shell> mysqldump *db_name* > *backup-file.sql*

Mantenimiento de Base de Datos

El cliente mysqlcheck analiza, repara y optimiza el contenido de las Tablas.

mysqlcheck es similar en funcion a myisamchk, pero trabaja diferente. La diferencia radica en que el primero no es necesario detener el servidor para realizar el mantenimiento, mientrás en el segundo si.

mysqldump

Para tener una lista de las opciones que soporta el programa ejecuta: mysqlcheck --help

Forma de Uso:

shell> mysqldcheck -- analize --all

Chin... perdi el password de root

- Localizar el pid que contiene el ID del proceso mysql.

- Detener el servicio mysql, de las formas conocidas, una puede ser:

shell> **kill `cat /mysql-data-directory/host_name.pid`**

- Crear un archivo de texto y coloca algo como lo siguiente:

SET PASSWORD FOR

Chin... perdi el password de root

Reiniciar el Servidor con la opcion --init-file

shell> **mysqld_safe --init-file=~/mysql-init &**

Reiniciar el Servidor con la opcion --init-file

Alternativamente con el programa cliente puedes hacer algo como detener el servicio y reiniciar con la opcion:

--skip-grant-tables --user=root

Esquema de Seguridad en PostgreSQL

Protección de los ficheros de la base de datos. Todos los ficheros almacenados en la base de datos estan protegidos contra escritura por cualquier cuenta que no sea la del superusuario de Postgres.

Las conexiones de los clientes al servidor de la base de datos estan permitidas, por defecto, únicamente mediante sockets Unix locales y no mendiante sockets TCP/IP. Ha de arrancarse el demonio con la opcion -i para permitir la conexion de clientes no locales.

Esquema de Seguridad en PostgreSQL

Las conexiones de los clientes se pueden restringir por dirección IP y/o por nombre de usuario mediante el fichero pg_hba.conf

A cada usuario de Postgres se le asigna un nombre de usuario y (opcionalmente) una contraseña. Por defecto, los usarios no tienen permiso de escritura a bases de datos que no hayan creado.

Los usuarios pueden ser incluidos en *grupos*, y el acceso a las tablas puede restringirse en base a esos grupos.

Copias de Seguridad y Restauracion

Las copias de Seguridad para una sola Base de Datos se establece:

% pg_dump *nombredb* > *nombredb*.pgdump

Y se puede restaurar de esta manera:

% cat *nombredb*.pgdump | psql *nombredb*

Conclusiones

Deshabilitar el Acceso Remoto

Cambiar el password de root por default

Eliminar cuentas anónimas y passwords en blanco.

Eliminar la Base de Datos test

Correr MySQL y Postgres como un usuario sin privilegios

Conceder privilegios mínimos a los usuarios.

Acerca de Coatzacoalcos, Mexico

Preguntas ???

Farid Alfredo Bielma Lopez

Instituto Tecnológico Superior de Coatzacoalcos

http://www.fbielma.org/talks/

fbielma@fbielma.org

MSN: fbielma@hotmail.com