

**Laporan Keamanan Basis Data**  
**Wordpress Multi Database & MySQL ACL**



Disusun Oleh :  
M. Arghian Taufahena  
4332101007

PROGRAM STUDI REKAYASA KEAMANAN SIBER  
JURUSAN TEKNIK INFORMATIKA  
POLITEKNIK NEGERI BATAM  
2022

**M. Arghian Taufahena**

**4332101007**

**RKS 3A Pagi**

---

## Keamanan Basis Data | Wordpress Multi Database

---

### 1. Database Server

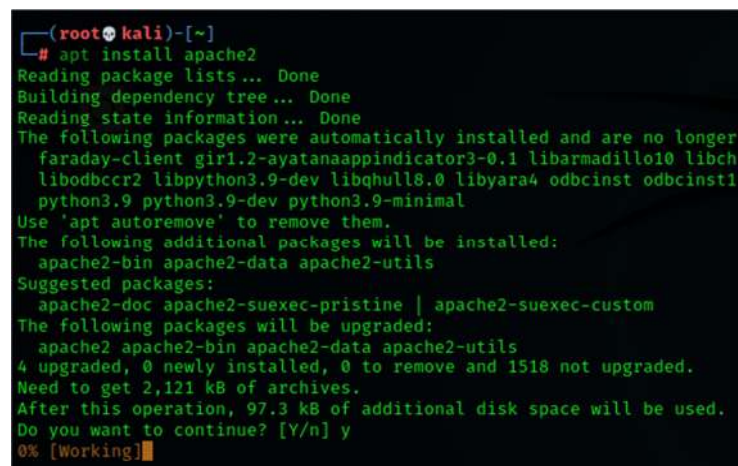
Untuk server database saya menggunakan os ubuntu server yang sudah saya install pada mesin virtual. Database server disini bertujuan untuk menyimpan ketiga web database yang berbeda dengan user privilege yang berbeda pula.

#### a. Set Up Layanan Server

Sebelum itu pertama kita akan set up server database dengan menginstall beberapa service ( disini saya menggunakan server dummy, hanya untuk menginstall service database server yang diperlukan, karena sebelumnya untuk Ubuntu database server saya tidak sempat untuk mengambil screenshot. ) ;

##### 1. Apache2

Apache2 disini berfungsi sebagai penghubung server dengan website yang akan kita tampilkan nantinya. Gunakan perintah *apt install apache2* untuk menginstall.



```
(root@kali)-[~]
# apt install apache2
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer
required:
  faraday-client gir1.2-ayatanaappindicator3-0.1 libarmadillo10 libch
  libodbccr2 libpython3.9-dev libqhull8.0 libyara4 odbcinst odbcinst1
  python3.9 python3.9-dev python3.9-minimal
Use 'apt autoremove' to remove them.
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils
Suggested packages:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom
The following packages will be upgraded:
  apache2 apache2-bin apache2-data apache2-utils
4 upgraded, 0 newly installed, 0 to remove and 1518 not upgraded.
Need to get 2,121 kB of archives.
After this operation, 97.3 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
0% [Working]
```

## 2. MySql

Selanjutnya adalah mysql, mysql disini berfungsi sebagai service untuk database wordpress kita nantinya. Gunakan perintah *apt install mariadb-server mariadb-client*

```
(root@kali)-[~]
# apt install mariadb-server mariadb-client
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer
required:
  faraday-client girl1.2-ayatanaappindicator3-0.1 libarmadillo10 libch
  libodbc2 libpython3.9-dev libqhull8.0 libyara4 odbcinst odbcinst1
  python3.9 python3.9-dev python3.9-minimal
Use 'apt autoremove' to remove them.
The following additional packages will be installed:
  default-mysql-server libdaxctl1 libndctl6 libpmem1 mariadb-client-1
Suggested packages:
  mailx mariadb-test netcat-openbsd
The following packages will be REMOVED:
  mariadb-client-10.5 mariadb-client-core-10.5 mariadb-server-10.5 ma
The following NEW packages will be installed:
  libdaxctl1 libndctl6 libpmem1 mariadb-client mariadb-client-10.6 ma
The following packages will be upgraded:
  default-mysql-server mariadb-common
2 upgraded, 9 newly installed, 4 to remove and 1516 not upgraded.
Need to get 13.7 MB of archives.
After this operation, 4,417 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://kali.darklab.sh/kali kali-rolling/main amd64 mariadb-com
```

## 3. PHP

Selanjutnya PHP, berfungsi sebagai service atau web browser, ketika ada permintaan dari browser ke web server, PHP akan menghubungi MySQL server untuk mencari data yang dibutuhkan di database. Gunakan perintah *apt install php php-mysql* untuk menginstall.

```
(root@kali)-[~]
# apt install php php-mysql
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are r
  faraday-client girl1.2-ayatanaappindicator3-0.1 libarmadillo
  libodbc2 libpython3.9-dev libqhull8.0 libyara4 odbcinst o
  python3-pyhp python3.9 python3.9-dev python3.9-minimal
Use 'apt autoremove' to remove them.
The following additional packages will be installed:
  libapache2-mod-php8.1 php8.1 php8.1-cli php8.1-common php8.
Suggested packages:
  php-pear
The following NEW packages will be installed:
  libapache2-mod-php8.1 php8.1 php8.1-cli php8.1-common php8.
The following packages will be upgraded:
  php php-mysql
2 upgraded, 7 newly installed, 0 to remove and 1514 not upgra
Need to get 4,857 kB of archives.
After this operation, 21.6 MB of additional disk space will b
Do you want to continue? [Y/n] y
Get:2 http://http.kali.org/kali kali-rolling/main amd64 php8.
Get:6 http://archive-4.kali.org/kali kali-rolling/main amd64
Get:7 http://http.kali.org/kali kali-rolling/main amd64 php a
13% [Waiting for headers]
```

## b. Set Up Database Server

Setelah selesai penginstallan semua service yang dibutuhkan, selanjutnya saya akan membuat ketiga database web nya, dengan cara :

- Masuk ke mysql root ( `mysql -u root -p` )
- Membuat database ( `Create database nama_database;` )
- Membuat user dengan untuk database sebelumnya ( `create user 'nama'@'ip host yang akan diberikan akses untuk remote user' identified by 'password';` )
- Beri semua hak akses pada akun yang telah kita buat ( `grant all privileges on nama_database.* to 'user'@'ip host';` )
- Refresh privileges ( `flush privileges;` )

Berikut database `wp_sucxsz` dengan user `sucxsz` yang sudah saya buat, dan bisa di remote pada atau di akses di mysql atau wordpress web server nantinya. Dengan ketentuan ip web server harus sama dengan ip host yang sudah saya masukkan untuk user `sucxsz`.

```
MariaDB [(none)]> create database wp_sucxsz;
Query OK, 1 row affected (1.541 sec)

MariaDB [(none)]> create user 'sucxsz'@'192.168.20.6' identified by 'gamma';
Query OK, 0 rows affected (0.438 sec)

MariaDB [(none)]> grant all privileges on wp_sucxsz.* to 'sucxsz'@'192.168.20.6';
ERROR 1133 (28000): Can't find any matching row in the user table
MariaDB [(none)]> grant all privileges on wp_sucxsz.* to 'sucxsz'@'192.168.20.6';
Query OK, 0 rows affected (0.099 sec)

MariaDB [(none)]> flush privileges;
Query OK, 0 rows affected (0.102 sec)

MariaDB [(none)]> _
```

Selanjutnya kita akan konfigurasi mysql server untuk menambahkan ip address server. Dengan cara buka dan edit file yang ada pada direktori `/etc/mysql/mariadb.conf.d/50-server.conf`. disini saya menggunakan nano sebagai layanan file editing konfigurasi. Masukkan ip server pada bagian bind-address, ini akan berguna sebagai host layanan yang akan kita berikan oleh server kita.



```
root@sucxsz:~# mysql -u sucxsz -h 192.168.20.5 -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 119
Server version: 10.6.7-MariaDB-2ubuntu1.1 Ubuntu 22.04

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| information_schema |
| wp_sucxsz |
+-----+
2 rows in set (0.07 sec)

MariaDB [(none)]>
```

Dapat dilihat disini sudah bisa remote database sesuai ketentuan akun yang sudah dibuat, yaitu user sucxsz dapat hak akses pada databases wp\_sucxsz.

## b. Install Wordpress CMS

Untuk pengisntallan wordpress pada webserver, disini saya menggunakan beberapa perintah yaitu :

- Mengunduh file wordpress pada domain [wordpress.org/latest.tar.gz](https://wordpress.org/latest.tar.gz) yang ditempatkan pada direktori /tmp ( `cd /tmp && wget https://wordpress.org/latest.tar.gz` )

```
root@sucxsz:~# cd /tmp && wget https://wordpress.org/latest.tar.gz
--2022-09-22 00:57:48-- https://wordpress.org/latest.tar.gz
Resolving wordpress.org (wordpress.org)... 198.143.164.252
Connecting to wordpress.org (wordpress.org)|198.143.164.252|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 21172479 (20M) [application/octet-stream]
Saving to: 'latest.tar.gz'

latest.tar.gz          5%[=>]
```

- Selanjutnya copy file atau folder wordpress ke direktori html ( `cp -R wordpress /var/www/html/` )
- Ubah permission dir wordpress ( `chown -R` ) dan ( `chmod -R 755` )
- Buat direktori uploads di dalam `/wordpress/wp-content/` ( `mkdir` )
- Kembali ubah permission dir uploads seperti wordpress ( `chown -R` )

```

root@sucxsz:/tmp# cp -R wordpress /var/www/html/
root@sucxsz:/tmp# chown -R www-data:www-data /var/www/html/wordpress/
root@sucxsz:/tmp# chmod -R 755 /var/www/html/wordpress/
root@sucxsz:/tmp# mkdir /var/www/html/wordpress/wp-content/uploads
root@sucxsz:/tmp# chown -R www-data:www-data /var/www/html/wordpress/wp-content/uploads/
root@sucxsz:/tmp#

```



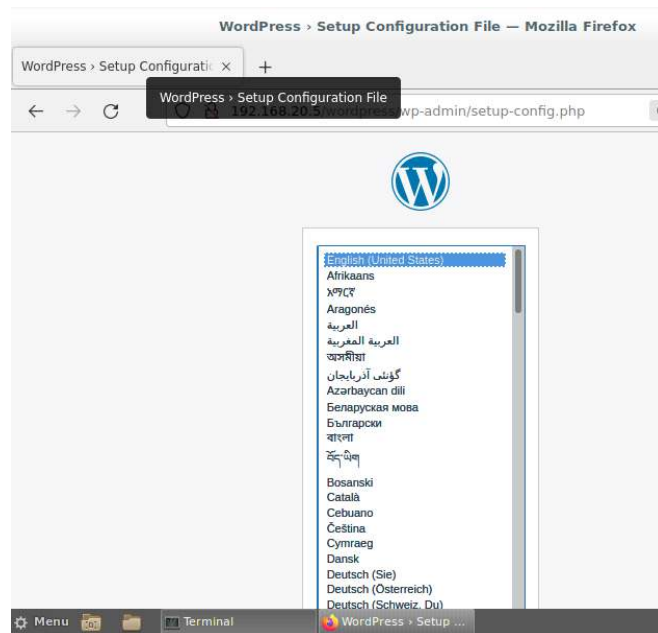
- Penjelasan Command
  - o `cp -R` yang berfungsi untuk menyalin sebuah direktori
  - o `chown -R` yang berfungsi untuk mengganti owner atau kepemilikan folder, disini `www-data:www-data` agar jikalau ada hacker yang mendapat exploit ke file, maka hacker tersebut hanya dapat masuk sebagai `www-data`.
  - o `Chmod -R` yang berfungsi mengubah permission sebuah file atau hak akses, disini `755` hanya bisa mengakses file untuk melihat dan mengeksekusi.

### c. Konfigurasi Wordpress

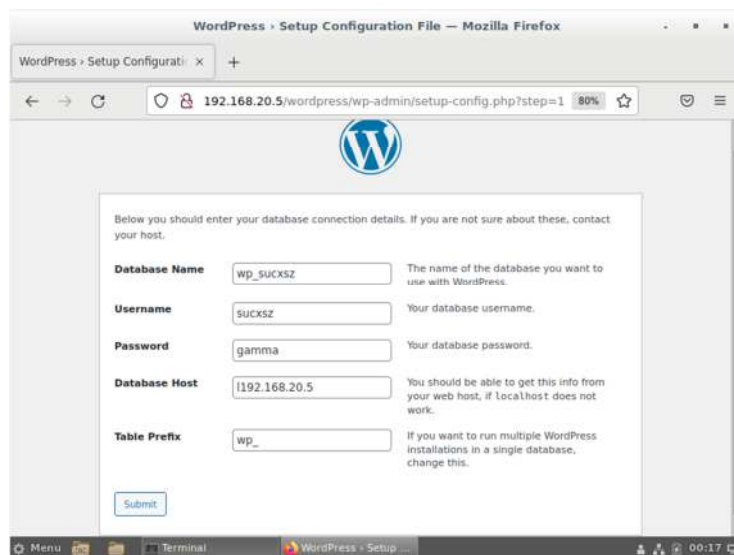
Untuk konfigurasi atau membuat website wordpress, kita gunakan browser untuk membuatnya, buka ip address web server pada kolom domain, disertai direktori wordpress. **192.168.20.6/wordpress.**

Pertama kita disuruh memilih bahasa yang akan kita gunakan



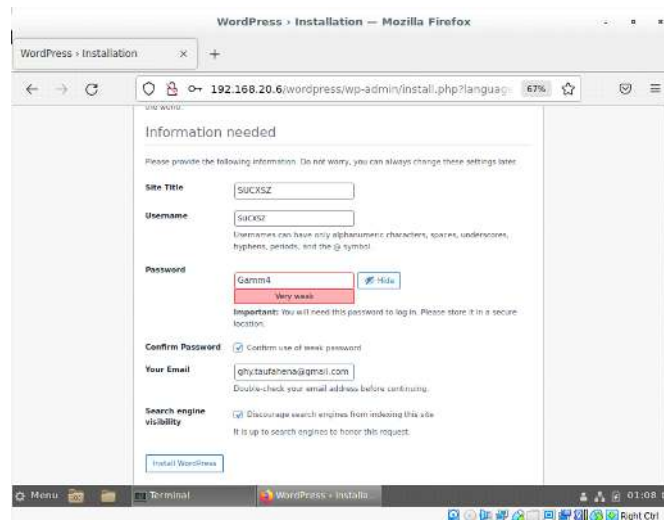


Selanjutnya kita akan memilih database yang akan kita gunakan sesuai dengan database yang ada di database server. Untuk website pertama, saya gunakan database wp\_sucxsz. Dan juga gunakan username, password dan host ip ( database server ) yang sesuai dengan database nya. Untuk table prefix tetap diisi dengan wp\_ saja.

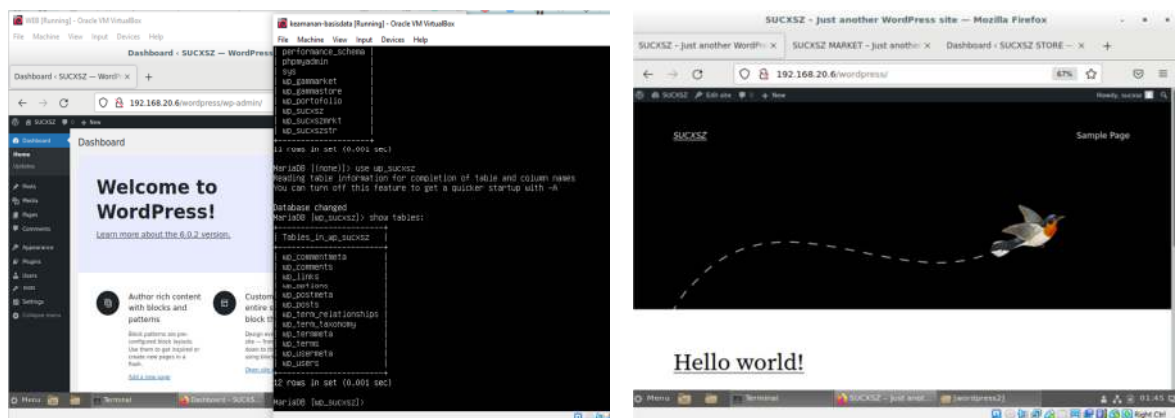


Selanjutnya kita isi beberapa kolom yang akan digunakan untuk wordpress, disini terdapat judul situs, username, password dan email. Diisi saja sesuai keinginan kita membuat website dan akun nya.





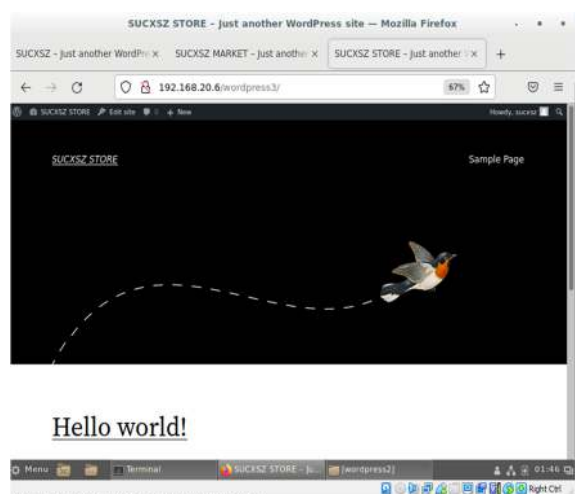
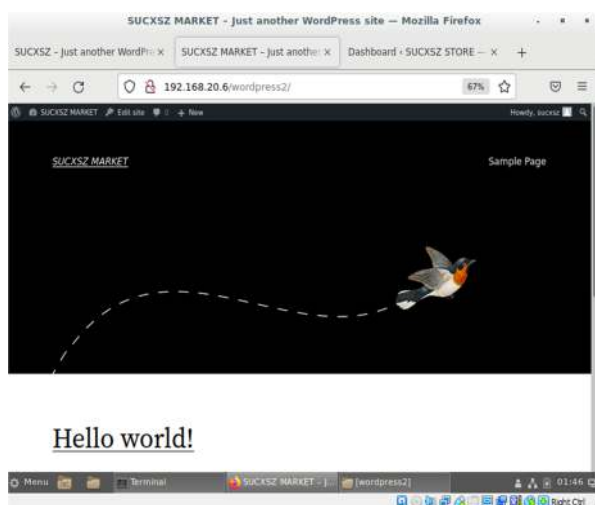
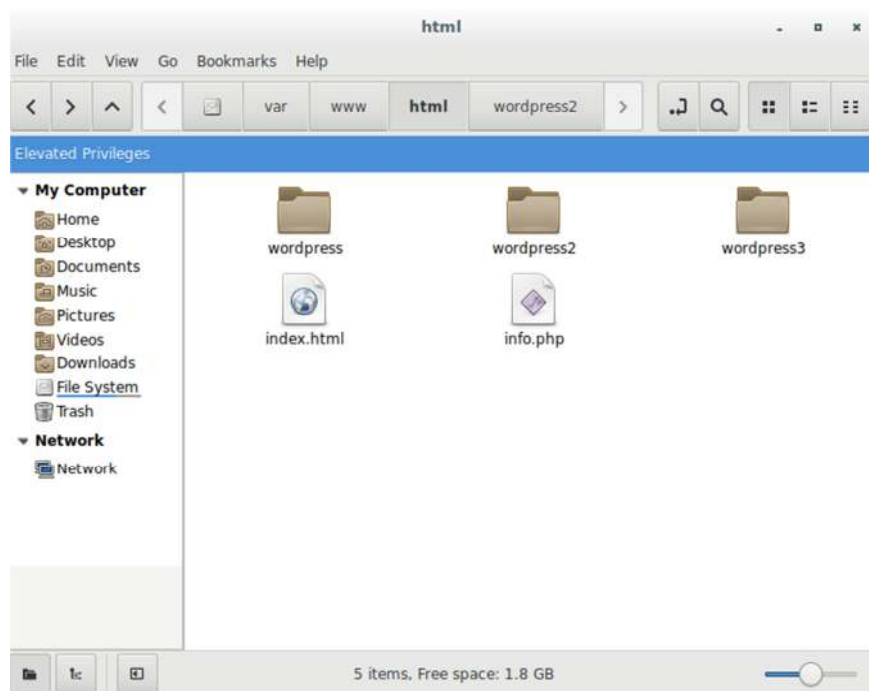
Setelah semua proses selesai, maka wordpress pertama sudah jadi. Dan lanjut untuk membuat web wordpress kedua dengan sisa 2 database yang masih kosong. Caranya tidak berbeda dengan sebelumnya, hanya saja ada sedikit perubahan pada direktori wordpress nya.



Untuk membuat dua wordpress lainnya adalah dengan cara menduplikat file wordpress yang sebelumnya telah ada. contoh disini saya buat2 duplikat file wordpress dengan tambahan angka 2 dan 3 sebagai penanda.

Tidak hanya duplikat direktori, salah satu file yang ada dalam wordpress harus kita hapus terlebih dahulu, yaitu file **wp-config.php**. kenapa dihapus ? karena file config ini sudah menyimpan konfigurasi dari wordpress pertama yang kita buat, jadi kita perlu membuat file config baru. Untuk membuatnya gampang, wordpress akan otomatis menyuruh kita membuat file **wp-config.php**, apabila di file wordpress yang digunakan tidak ada pada saat penginstallan atau pembuatan web berlangsung.

Dan untuk membuat website kedua ini kita hanya perlu mengganti direktori pada domain awal, yaitu **192.168.20.6/wordpress2**



Terakhir, saya akan mencoba melihat isi dalam database ketiga, yang harusnya didalamnya sudah terisi oleh database dari wordpress website ketiga saya. Disini saya akan menggunakan remote akses mysql user ke 3.

```
root@sucxsz:~# mysql -u sucxsz3 -h 192.168.20.5 -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 120
Server version: 10.6.7-MariaDB-2ubuntu1.1 Ubuntu 22.04


Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| information_schema |
| wp_sucxsz3 |
+-----+
2 rows in set (0.00 sec)

MariaDB [(none)]> use wp_sucxsz3
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

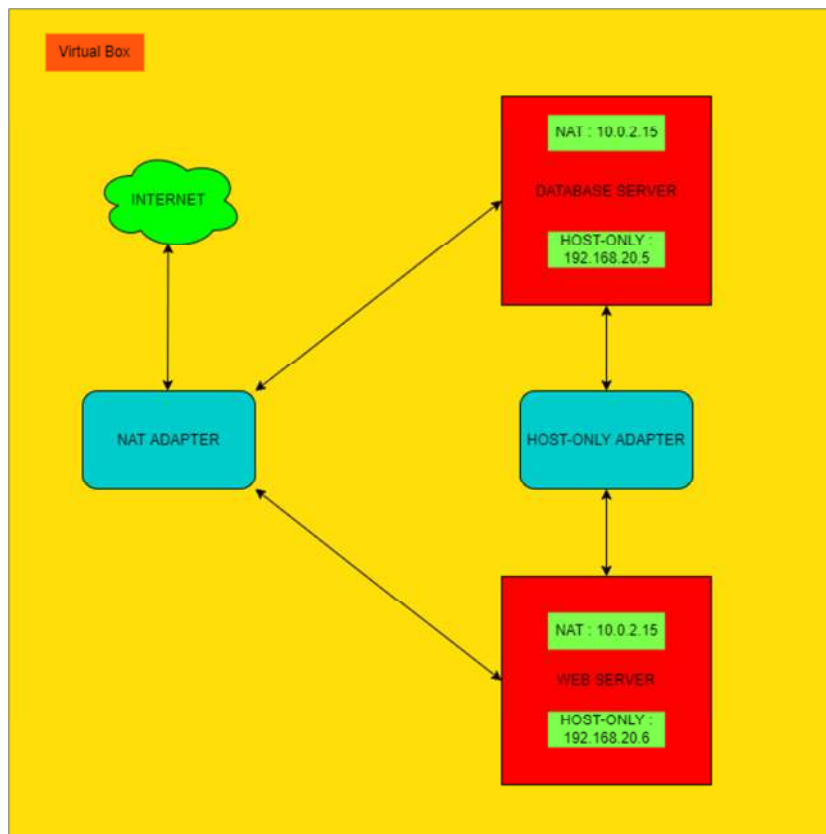
Database changed
MariaDB [wp_sucxsz3]> show tables;
+-----+
| Tables_in_wp_sucxsz3 |
+-----+
| wp_commentmeta |
| wp_comments |
| wp_links |
| wp_options |
| wp_postmeta |
| wp_posts |
| wp_term_relationships |
| wp_term_taxonomy |
+-----+
```



Nah, sudah dapat dilihat bahwa database ke 3 saya sudah terisi dengan database wordpress ke 3.

### 3. Topologi

Topologi jaringan yang saya gunakan untuk menjalankan kedua server adalah sebagai berikut :

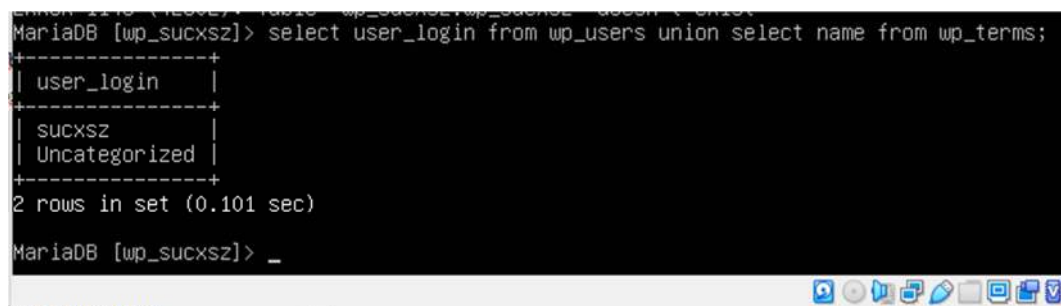


## 4. Mysql UNION, SELECT, SLEEP & --

### a. UNION

Fungsi MySQL UNION adalah Operator MySQL yang digunakan untuk menggabungkan kumpulan hasil dari 2 atau lebih pernyataan SELECT. Ini menghapus duplikat baris antara berbagai pernyataan SELECT. Setiap pernyataan SELECT di dalam operator UNION harus memiliki jumlah field yang sama pada set hasil dengan tipe data yang sama.

**Contoh :** *select nama\_kolom from nama\_tabel UNION select nama\_kolom from tabel2.*



```

MariaDB [wp_sucxsz]> select user_login from wp_users union select name from wp_terms;
+-----+
| user_login |
+-----+
| sucxsz    |
| Uncategorized |
+-----+
2 rows in set (0.101 sec)

MariaDB [wp_sucxsz]> _
```

Disini saya menggunakan kolom `user_login` dari table `wp_users` dan kolom `name` dari table `wp_terms`

### b. SELECT

Select digunakan untuk memilih data dari sebuah tabel atau beberapa tabel. Data yang dipilih menggunakan select dapat dalam satu kondisi atau beberapa kondisi yang mengkombinasikan select dengan operator lain.

**CONTOH : SELECT \* FROM nama\_tabel**



```

MariaDB [wp_sucxsz]> select user_pass from wp_users;
+-----+
| user_pass |
+-----+
| $P$BUjkBSZ4g9eVkjtoE2EJUDwjbpcYz/ |
+-----+
1 row in set (0.000 sec)

MariaDB [wp_sucxsz]> _
```

Disini saya menggunakan kolom `user_pass` pada table `wp_users`.

### c. SLEEP

Sleep digunakan menunggu batas waktu berakhir. Pada contoh di bawah saya mau menampilkan semua record dari table wp\_users kemudian saya gunakan sleep(5) maka akan membutuhkan waktu 5 detik untuk menampilkan semua recordnya.

**CONTOH : Select \*, SLEEP from nama\_tabel**

```
Database changed
MariaDB [wp_sucxsz]> show tables;
+-----+
| Tables_in_wp_sucxsz |
+-----+
wp_commentmeta
wp_comments
wp_links
wp_options
wp_postmeta
wp_posts
wp_term_relationships
wp_term_taxonomy
wp_termmeta
wp_terms
wp_usermeta
wp_users
+-----+
12 rows in set (0.00 sec)

MariaDB [wp_sucxsz]> select *, sleep(7) from wp_posts;
```

### d. SQL Comments ( -- )

Perintah -- pada sql digunakan untuk menambahkan komentar dengan pernyataan dengan contoh kode .

**CONTOH : select\* all records from nama table; -- dari table nama table**

```
MariaDB [wp_sucxszmkt]> select* -- selects all records
-> from wp_users; -- dari table wp_users
+-----+
| ID | user_login | user_pass | user_nicename | user_email | user_activation_key | user_status | display_name |
+-----+
| 1 | sucxsz | $P$B0puExtMfDu2DUaiz8VjTKAu9H3PYp. | sucxsz | ghy.taufahena@gmail.com | 0 | sucxsz |
+-----+
1 row in set (0.000 sec)

MariaDB [wp_sucxszmkt]>
```

## 5. Mysql ACL

MySQL menggunakan keamanan berdasarkan Akses Control List ( ACL ) untuk semua koneksi, kueri, dan operasi lain yang dapat dilakukan pengguna. Ada juga dukungan untuk koneksi terenkripsi SSL antara klien dan server MySQL.

Sama seperti yang sudah saya jelaskan diatas, tentang remote MySQL Database Server di Web server.

```
root@sucxsz:~# mysql -u sucxsz -h 192.168.20.5 -p
Enter password:
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 119
Server version: 10.6.7-MariaDB-2ubuntu1.1 Ubuntu 22.04

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| information_schema |
| wp_sucxsz |
+-----+
2 rows in set (0.07 sec)

MariaDB [(none)]>
```

### a. Fundamental Access Control

ACL biasa digunakan untuk manajemen user dan control akses , Contoh disini, perintah **Show Grants** untuk melihat privilege sebuah user dan **Revoke** untuk mneghapus hak akses yang tidak diperlukan.

```
MariaDB [(none)]> show grants for 'sucxsz'@'192.168.20.6';
+-----+
| Grants for sucxsz@192.168.20.6 |
+-----+
| GRANT USAGE ON *.* TO `sucxsz`@`192.168.20.6` IDENTIFIED BY PASSWORD '*3DD86F1DCB51287BC6DFA271770DD805ACAE1224' |
| GRANT ALL PRIVILEGES ON `wp_sucxsz`.* TO `sucxsz`@`192.168.20.6` |
+-----+
2 rows in set (0.106 sec)

MariaDB [(none)]> _
```



Investasikan dalam firewall. Ini melindungi Anda dari setidaknya 50% dari semua jenis eksploitasi dalam perangkat lunak apa pun. Letakkan MySQL di belakang firewall atau di zona demiliterisasi ( DMZ ).

MySQL menggunakan port 3306 secara default. Port ini seharusnya tidak dapat diakses dari host yang tidak dipercaya. Sebagai cara sederhana untuk memeriksa apakah port MySQL Anda terbuka, coba perintah berikut dari beberapa mesin jarak jauh,.

Contohnya disini saya menggunakan perintah telnet untuk mengetahui apakah port 3306 dari MySQL dapat diakses dari pihak luar , dengan perintah *telnet ip\_server 3306*

```
(root@kali)-[/home/kali]
# telnet 192.168.20.5 3306
Trying 192.168.20.5 ...
Connected to 192.168.20.5.
Escape character is '^]'.
HHost '192.168.20.10' is not allowed to connect to this MariaDB server
Connection closed by foreign host.

(root@kali)-[/home/kali]
#
```

## b. Mandatory Access Control ( MAC )

Kontrol Akses Wajib ( MAC ) mengacu pada jenis kontrol akses yang dengannya sistem operasi atau basis data membatasi kemampuan subjek atau inisiator untuk mengakses atau umumnya melakukan semacam operasi pada objek atau target.

Dalam sistem operasi, subjek biasanya merupakan proses atau utas; objek adalah konstruksi seperti file, direktori, port TCP / UDP, segmen memori bersama, perangkat IO, dll. Subjek dan objek masing-masing memiliki satu set atribut keamanan. Setiap kali subjek mencoba mengakses objek, aturan otorisasi yang diberlakukan oleh kernel sistem operasi memeriksa atribut keamanan ini dan memutuskan apakah akses dapat terjadi.

Setiap operasi oleh subjek apa pun pada objek apa pun diuji terhadap seperangkat aturan otorisasi ( alias kebijakan ) untuk menentukan apakah operasi diizinkan. Sistem manajemen basis data, dalam mekanisme kontrol aksesnya, juga dapat menerapkan kontrol akses wajib; dalam hal ini, objek adalah tabel, tampilan, prosedur, dll.

Contoh implementasi MAC dalam MySQL :

Disini saya berikan akses khusus berupa select untuk database wp\_sucxsz table wp\_users kepada user sucxsz.

```
11 rows in set (0.000 sec)

MariaDB [(none)]> grant select on wp_sucxsz.wp_users to 'sucxsz'@'192.168.20.6';
Query OK, 0 rows affected (0.183 sec)

MariaDB [(none)]> flush privileges;
Query OK, 0 rows affected (0.672 sec)

MariaDB [(none)]>
```

Selanjutnya langsung kita coba login MySQL dari Web Server dengan menggunakan user sucxsz, Dan coba perintah select yang sudah diberikan.

```
MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| information_schema |
| wp_sucxsz |
+-----+
2 rows in set (0.00 sec)

MariaDB [(none)]> use wp_sucxsz;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
MariaDB [wp_sucxsz]> select * from wp_users
-> ;
+----+-----+-----+-----+-----+-----+-----+-----+
| ID | user_login | user_pass | user_registered | user_activation_key | user_nicename | user_email | user_url |
+----+-----+-----+-----+-----+-----+-----+-----+
| 1 | sucxsz | $P$BUjkBSZ4g9eVkjtoE2EJUDwjbpcYz/ | 2022-09-22 05:08:51 | sucxsz | ghy.taufahena@gmail.com | http://192.168.20.6/wordpress |
+----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

Berhasil, dapat diketahui bahwa pemberian access control tipe mandatorian ini dapat berjalan. Users sucxsz ini tidak dapat menggunakan perintah sql lain selain select, karena saya hanya memberikan akses perintah select saja.

### c. Discretionary Access Control ( DAC )

Berbeda dengan MAC, terkadang suatu system memiliki kontrol akses “diskresioner”, atau tidak memiliki control akses wajib. Sistem dapat mengimplementasikan MAC dan DAC secara bersamaan, DAC mengacu pada kategaori kontrol akses yang dapat ditransfer satu sama lain.

DAC menitik beratkan pada security objects, security subjecst,dan accessprivileges . DAC akan membatasi akses pada objel – objek berdasarkan identitas subjek atau grup dimana mereka berada.

Contoh implementasi DAC dalam MySQL :

Disini pada database saya beri izin untuk perintah Select, Create, Update, dan Delete pada database wp\_gammarket, yang ditujukan ke user sucxsz.

```
sys
| wp_gammarket
| wp_gammastore
| wp_portofolio
| wp_sucxsz
| wp_sucxszmkt
| wp_sucxszstr
+-----+
11 rows in set (0.002 sec)

MariaDB [(none)]> grant select,create,insert,update,delete on wp_gammarket.* to 'sucxsz'@'192.168.20
.6';
Query OK, 0 rows affected (0.077 sec)

MariaDB [(none)]> flush privileges;
Query OK, 0 rows affected (0.032 sec)

MariaDB [(none)]> _
```

```

MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| information_schema |
| wp_gammarket |
| wp_sucxsz |
+-----+
3 rows in set (0.00 sec)

MariaDB [(none)]> use wp_gammarket;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
MariaDB [wp_gammarket]> use wp_gammarket;
Database changed
MariaDB [wp_gammarket]> create table data_pelanggan (nomor int(8) not null, nama char(50) not null, prodi varchar(255) not null);
Query OK, 0 rows affected (0.14 sec)

MariaDB [wp_gammarket]> show tables;
+-----+
| Tables_in_wp_gammarket |
+-----+
| data_pelanggan |
| wp_commentmeta |
| wp_comments |
| wp_links |
| wp_options |
| wp_postmeta |
| wp_posts |
| wp_term_relationships |
| wp_term_taxonomy |
| wp_termmeta |
| wp_terms |
+-----+

```

Dapat dilihat bahwa user sucxsz dapat melakukan perintah create atau membuat table pada database wp\_gammarket.

Tidak hanya create, saya juga bisa melakukan perintah select, update, dan delete pada database ini, karena sebelumnya sudah diberikan hak akses kepada user sucxsz.

Sekian Penjelasan Mengenai MySQL ACL Dan  
Wordpress Multi Database  
Semoga Bermanfaat Dan Dapat Dipahami.

Terima Kasih

