PENGELOLAAN KEAMANAN

Lab. Sistem Basis Data (MySQL)

By Galih Hermawan

PENDAHULUAN

- Topik bahasan:
  - Mengelola hak akses user terhadap suatu database atau bagian-bagian tertentu dari database
  - Memastikan user tertentu saja yang bisa mengakses database atau bagian-bagiannya
  - Menentukan aksi apa saja yang bisa dilakukan oleh user tersebut saat mengakses data
  - Penggunaan perintah MySQL: GRANT, SHOW GRANTS, SET PASSWORD, FLUSH PRIVILEGES, REVOKE, dan DROP USER



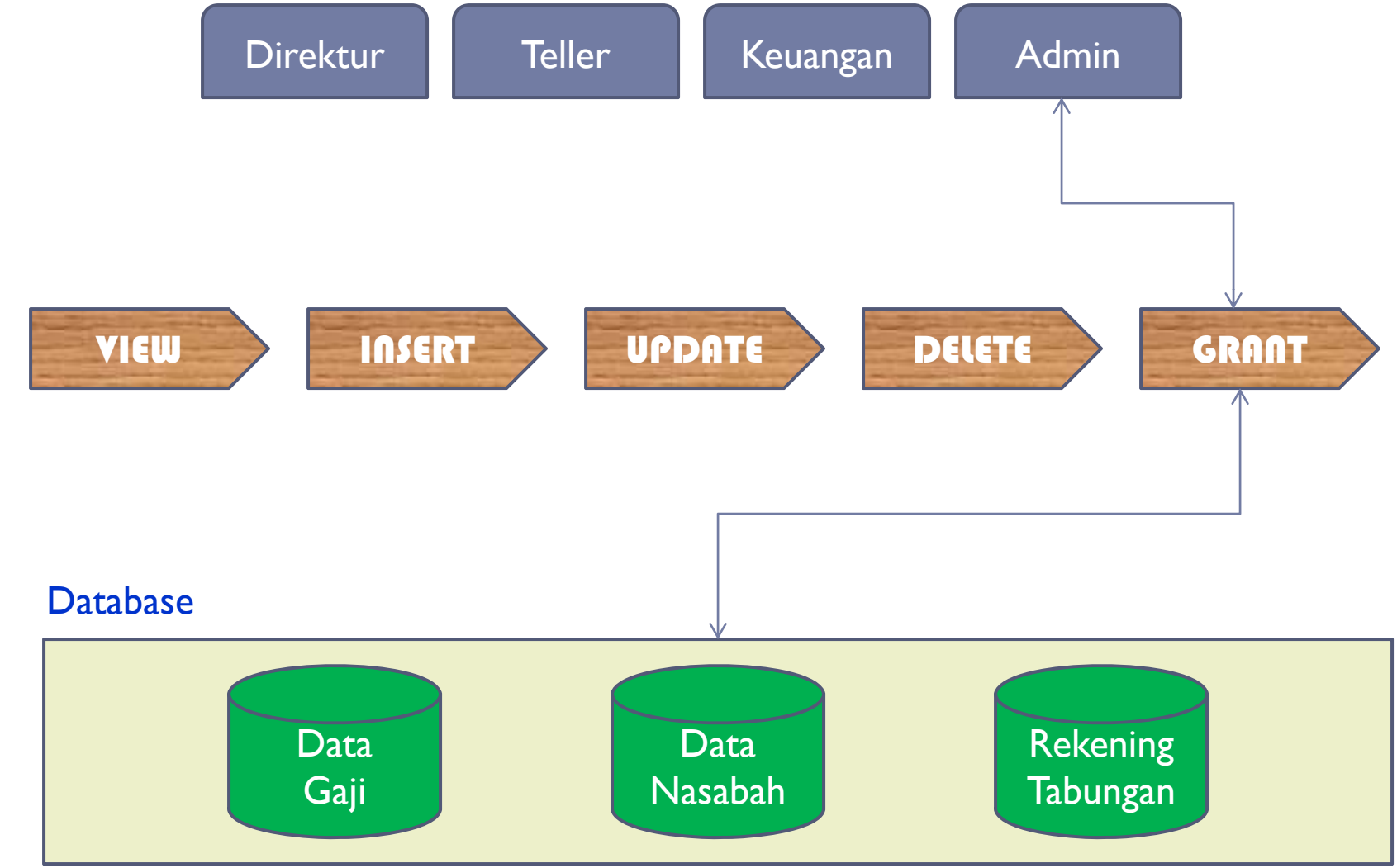
SISTEM HAK AKSES

- Tabel-tabel yang mengelola hak akses pengguna adalah *grant tables*
- Apa yang tersimpan dalam *grant tables*?
  - User mana / siapa yang berhak mengakses MySQL
  - Dari komputer mana si user boleh mengakses data
  - Aksi-aksi apa saja yang bisa dilakukan user tersebut
- Contoh kasus:

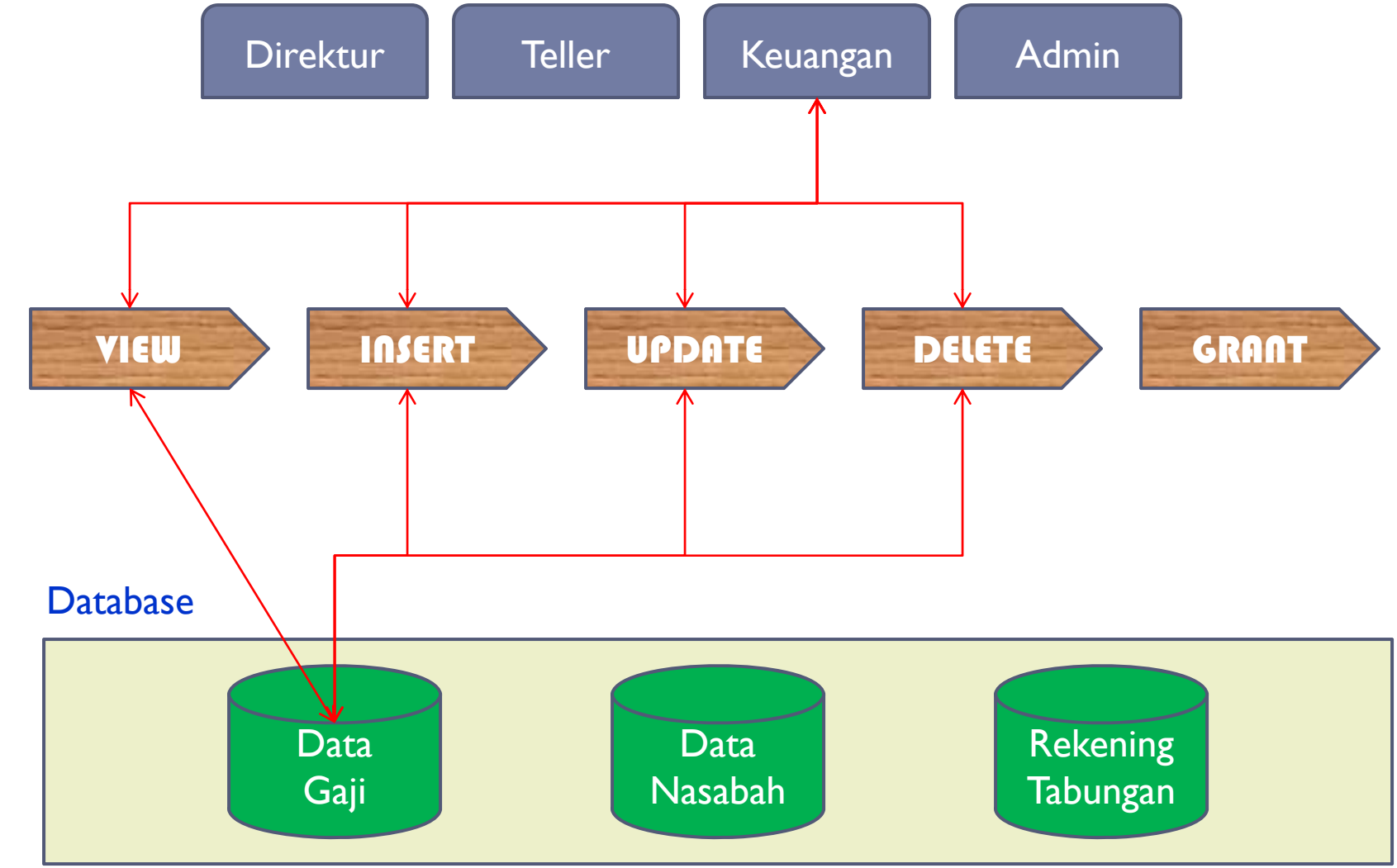
Ada user yang bisa melihat atau menampilkan data tertentu saja, sedangkan dia tidak bisa melakukan peng-update-an maupun penghapusan, ataupun penambahan data baru



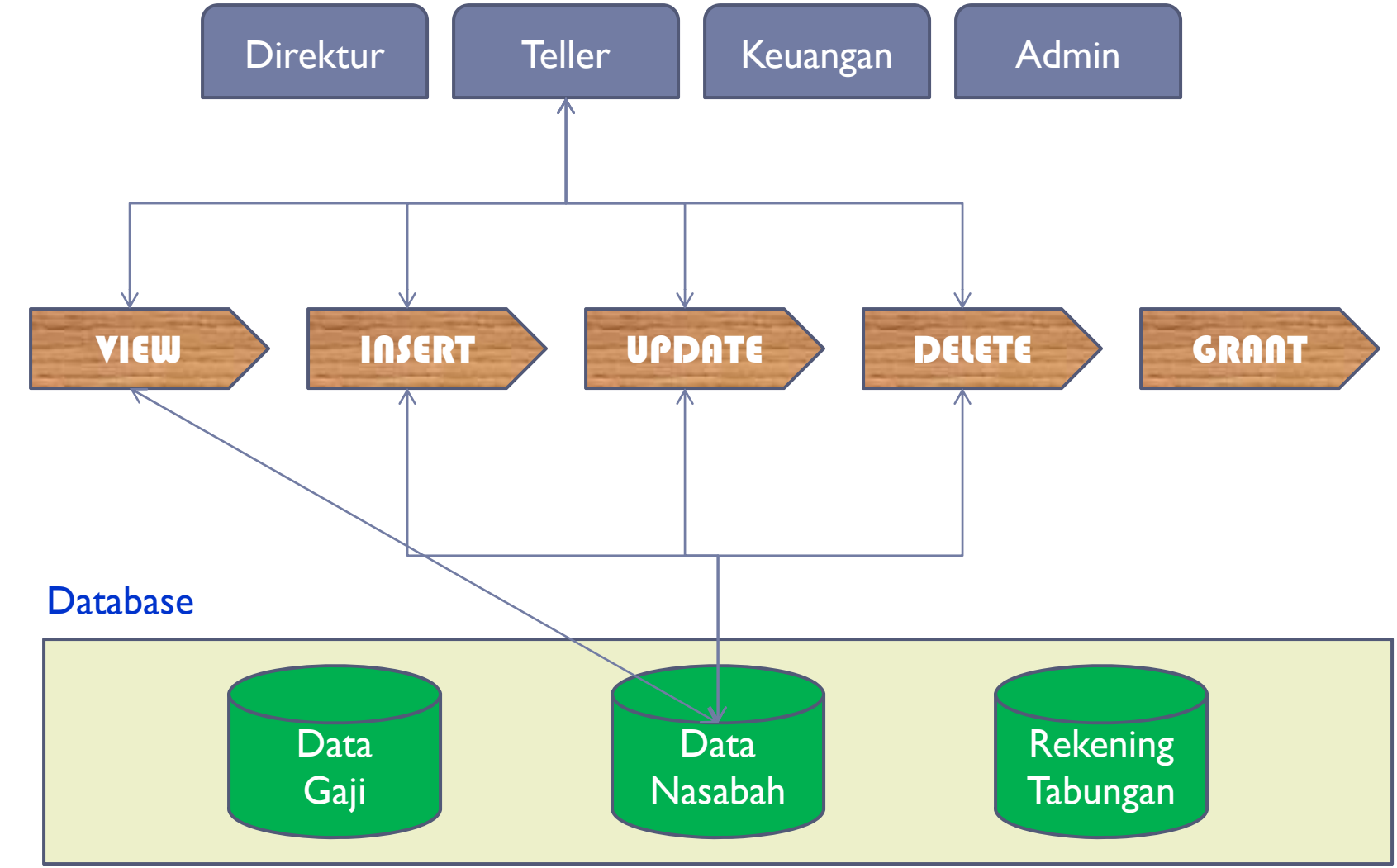
STUDI KASUS


Direktur
Teller
Keuangan
Admin
VIEW
INSERT
UPDATE
DELETE
GRANT
Database
Data Gaji
Data Nasabah
Rekening Tabungan

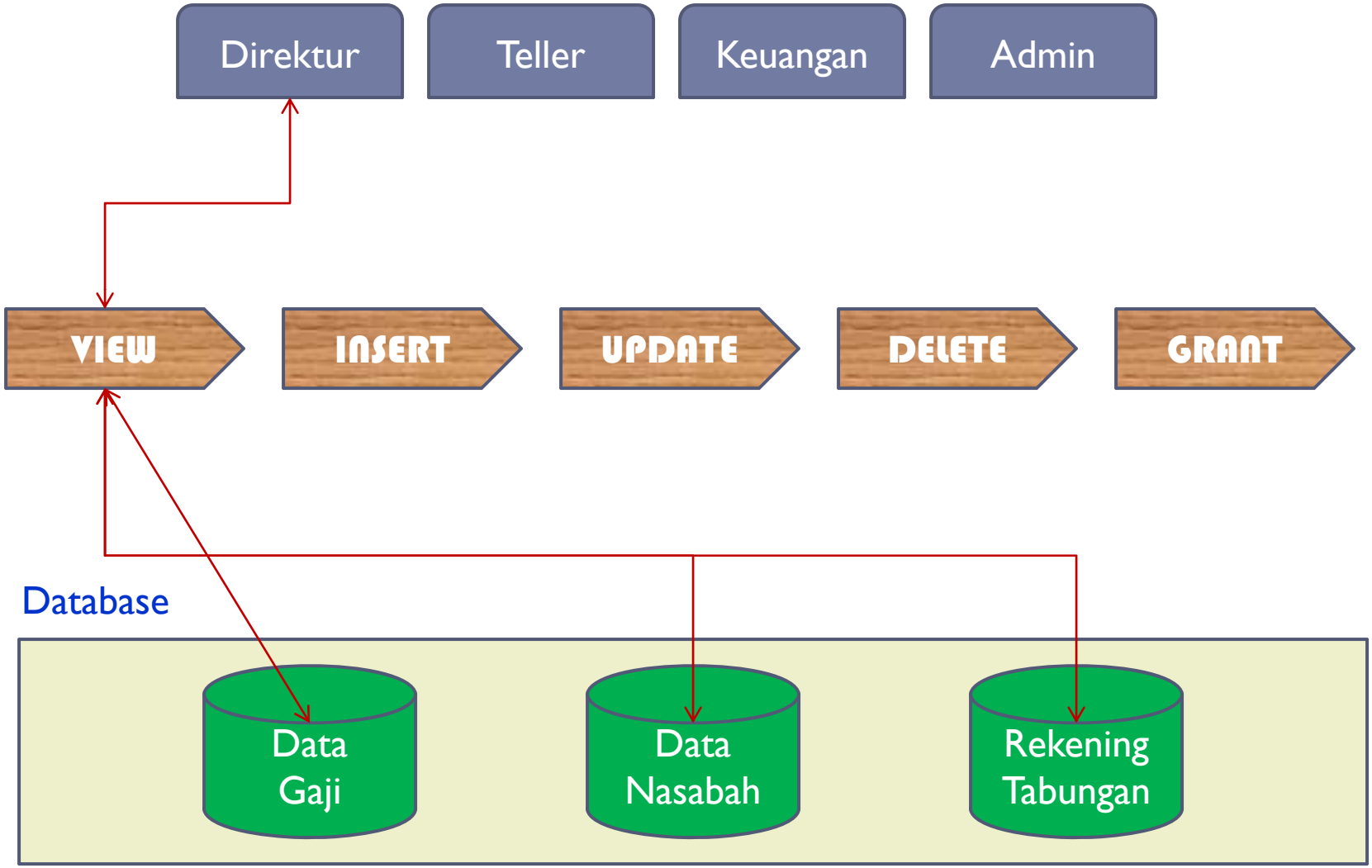
STUDI KASUS


Direktur
Teller
Keuangan
Admin
VIEW
INSERT
UPDATE
DELETE
GRANT
Database
Data Gaji
Data Nasabah
Rekening Tabungan

STUDI KASUS


Direktur
Teller
Keuangan
Admin
VIEW
INSERT
UPDATE
DELETE
GRANT
Database
Data Gaji
Data Nasabah
Rekening Tabungan

STUDI KASUS


Direktur
Teller
Keuangan
Admin
VIEW
INSERT
UPDATE
DELETE
GRANT
Database
Data Gaji
Data Nasabah
Rekening Tabungan

MySQL GRANT TABLES

- Tabel User
  - Mengatur siapa yang bisa terhubung, dari komputer mana dia terhubung, dan hak akses global apa yang dia miliki

```
mysql> SELECT HOST, USER, Select_priv, Delete_priv, Insert_priv, Update_priv, Create_priv
    -> FROM mysql.user;
+-----------+-------------+-------------+-------------+-------------+-------------+-------------+
| HOST      | USER        | Select_priv | Delete_priv | Insert_priv | Update_priv | Create_priv |
+-----------+-------------+-------------+-------------+-------------+-------------+-------------+
| localhost | root        | Y           | Y           | Y           | Y           | Y           |
| 127.0.0.1 | root        | Y           | Y           | Y           | Y           | Y           |
| localhost | galih       | Y           | Y           | Y           | Y           | Y           |
| %         | informatika | Y           | N           | Y           | Y           | N           |
+-----------+-------------+-------------+-------------+-------------+-------------+-------------+
4 rows in set (0.00 sec)
```



MySQL GRANT TABLES

- Tabel Db
  - Memberikan hak akses pada user secara spesifik pada database tertentu

```
mysql> SELECT Host, Db, User, Select_priv, Update_priv, delete_priv
    -> FROM mysql.db;
+------+---------+-------------+-------------+--------------+--------------+
| Host | Db      | User        | Select_priv | Update_priv  | delete_priv  |
+------+---------+-------------+-------------+--------------+--------------+
| %    | test    |             | Y           | Y            | Y            |
| %    | test\_% |             | Y           | Y            | Y            |
| %    | world   | informatika | Y           | Y            | N            |
+------+---------+-------------+-------------+--------------+--------------+
3 rows in set (0.00 sec)
```



MySQL GRANT TABLES

- Tabel Host
  - Tabel host berkaitan dengan tabel db dan diperiksa hanya ketika seorang user terdaftar dalam tabel db namun kolom Host kosong
- Tabel tables_priv
  - Tabel ini lebih spesifik ke hak akses tingkat tabel. Hak-hak akses yang terdaat di tabel ini diterapkan hanya pada tabel yang dispesifikasikan pada tabel tables_priv
- Tabel columns_priv
  - Tabel ini menunjukkan hak-hak akses yang berhubungan dengan kolom-kolom secara individu



MySQL PRIVILEGES

- Tabel user, host dan db berisi kolom yang masing-masing kolom tersebut mewakili hak-hak akses secara spesifik
- Tipe: berhubungan dengan data, administrasi, enkripsi, dan koneksi
- Detail hak-hak akses terdapat di modul

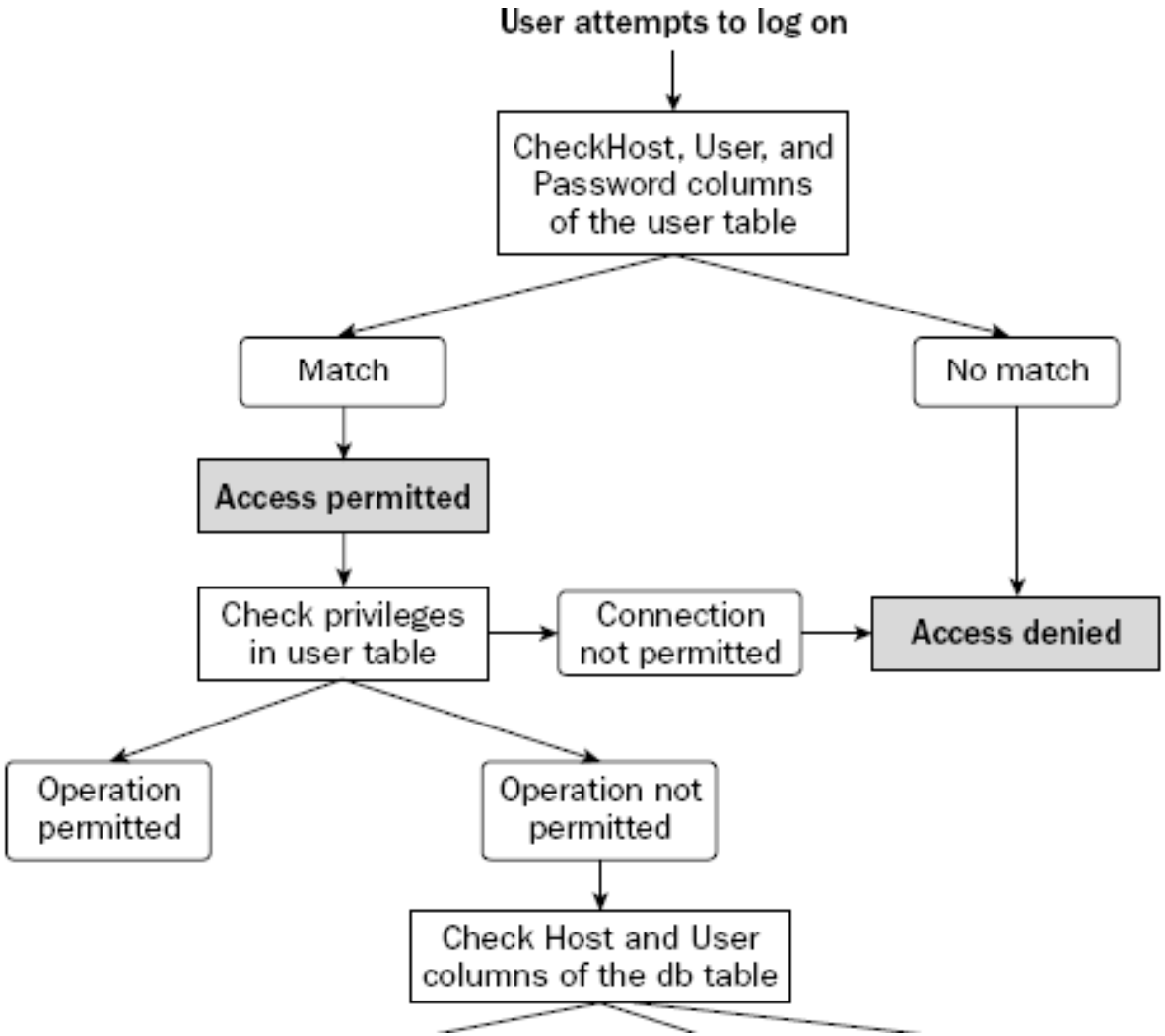


AUTENTIKASI KONEKSI

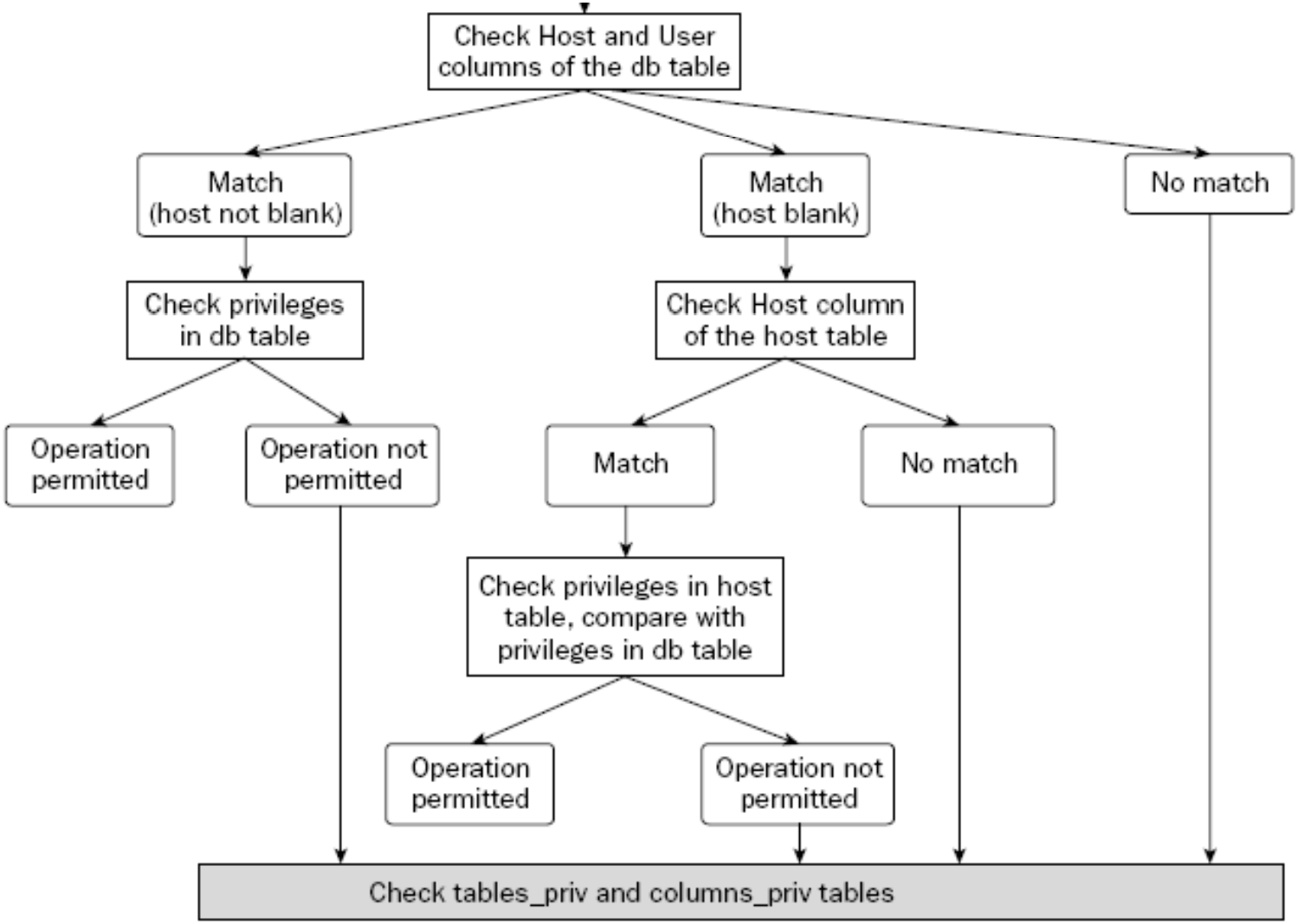
- Ketika MySQL server mulai, data dari tabel user disalin ke memori.
- Ketika klien berusaha *log in* ke server, akun user diperiksa/disesuaikan dengan data user di memori.
- Server menggunakan inputan user untuk mengautentikasi user, berdasarkan nilai Host dan User.



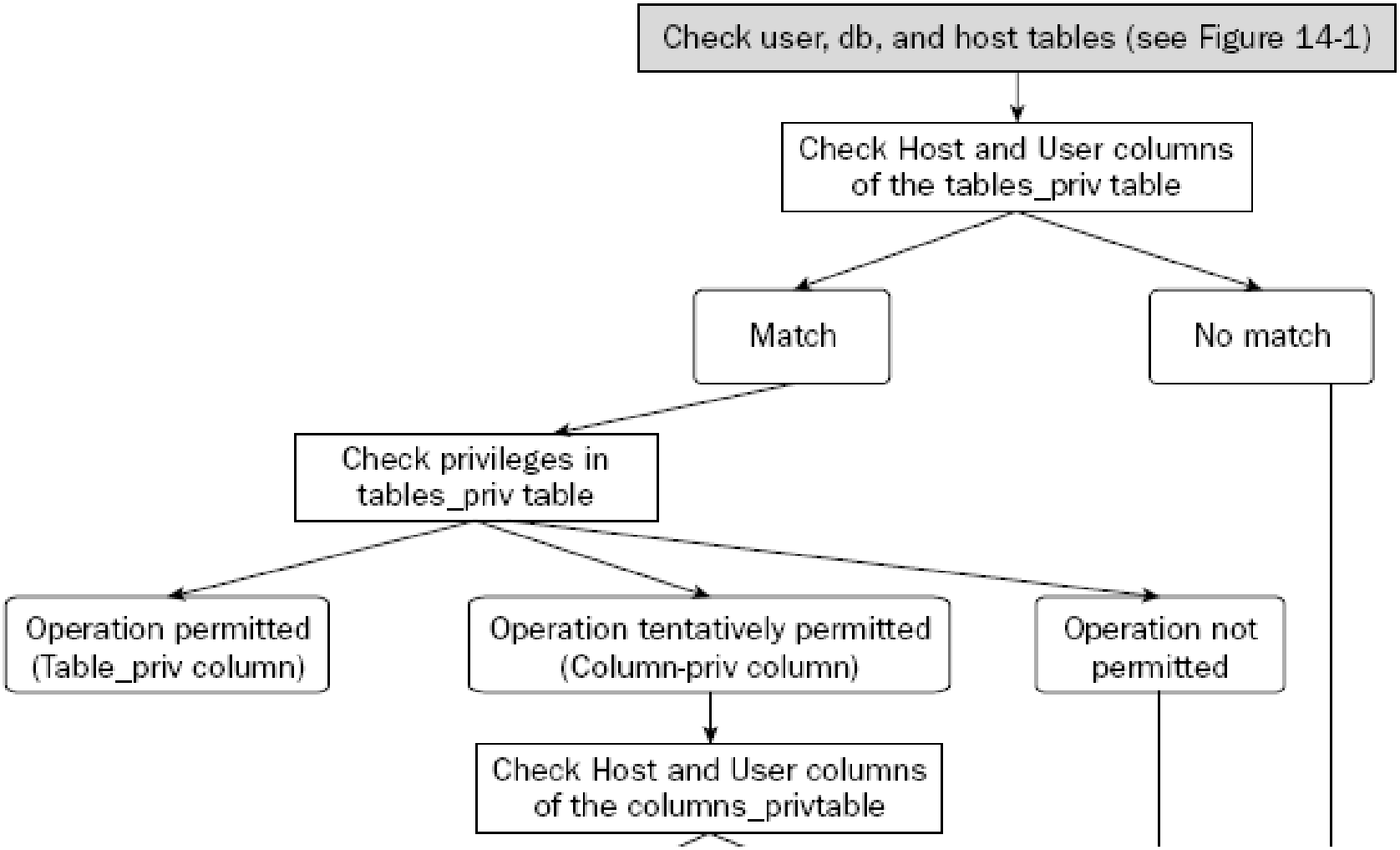
VERIFIKASI HAK AKSES



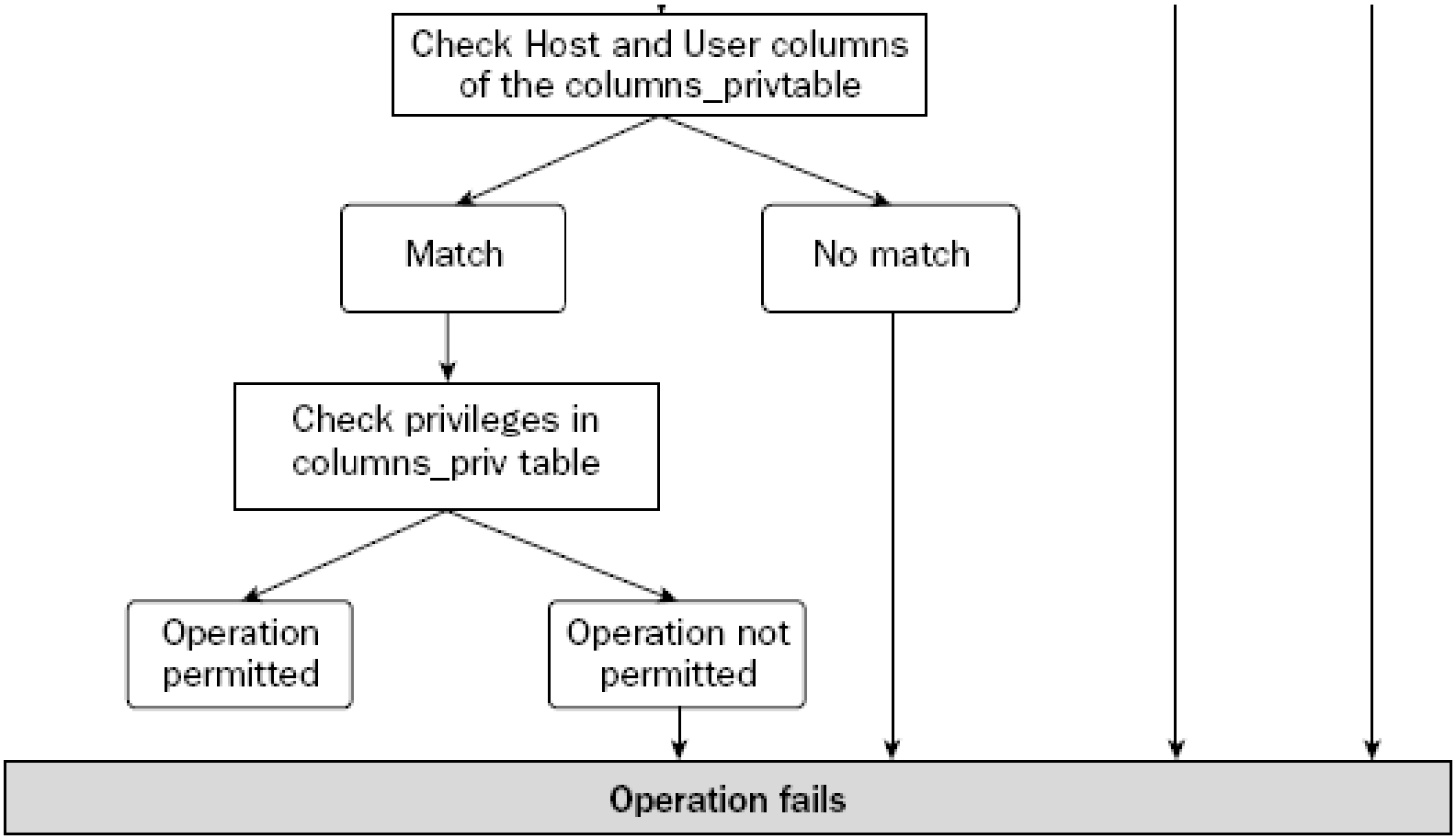
VERIFIKASI HAK AKSES



VERIFIKASI HAK AKSES



VERIFIKASI HAK AKSES



MENGELOLA AKUN USER

- Menambah user melalui perintah GRANT

```
GRANT <privilege> [(<column> [{, <column>}...])]
   [{, <privilege> [(<column> [{, <column>}...])]}...]
ON {<table> | * | *.* | <database>.*}
TO '<user>'@'<host>' [IDENTIFIED BY [PASSWORD] '<new password>']
   [{, '<user>'@'<host>' [IDENTIFIED BY [PASSWORD] '<new password>']}...]
[REQUIRE {NONE | SSL | X509 | {<require definition>}]
[WITH <with option> [<with option>...]]

<require definition>::=
<require option> [[AND] <require option>] [[AND] <require option>]

<require option>::=
{CIPHER '<string>'}
| {ISSUER '<string>'}
| {SUBJECT '<string>'}

<with option>::=
{GRANT OPTION}
| {MAX_QUERIES_PER_HOUR <count>}
| {MAX_UPDATES_PER_HOUR <count>}
| {MAX_CONNECTIONS_PER_HOUR <count>}
```



MENGELOLA AKUN USER

- Tabel berikut berisi sensivitas karakter pada tiap-tiap kolom

Column	Case sensitive?
Host	No
User	Yes
Password	Yes
Db	Yes
Table_name	Yes
Column_name	No



SINTAKS PEMBERIAN HAK AKSES

- GRANT: All, Select, Insert, Update, Delete, dll
- ON: *.*, db_mahasiswa.*, db_mahasiswa.nilai, dll
- TO: nama_pengguna (bisa dikosongkan untuk *anonymous*)
- @host: 192.168.1.2, unikom.ac.id, localhost, %, dll
- IDENTIFIED BY: password



CONTOH PEMBERIAN HAK AKSES

- GRANT ALL
  ON *.*
  TO ‘galih’@‘localhost’ IDENTIFIED BY ‘1234’

- GRANT SELECT
  ON world.*
  TO ‘budi’@‘%’ IDENTIFIED BY ‘abc’

- GRANT SELECT, INSERT, UPDATE (District, Population)
  ON world.city
  TO ‘if’@‘galih.eu’ IDENTIFIED BY ‘abc’

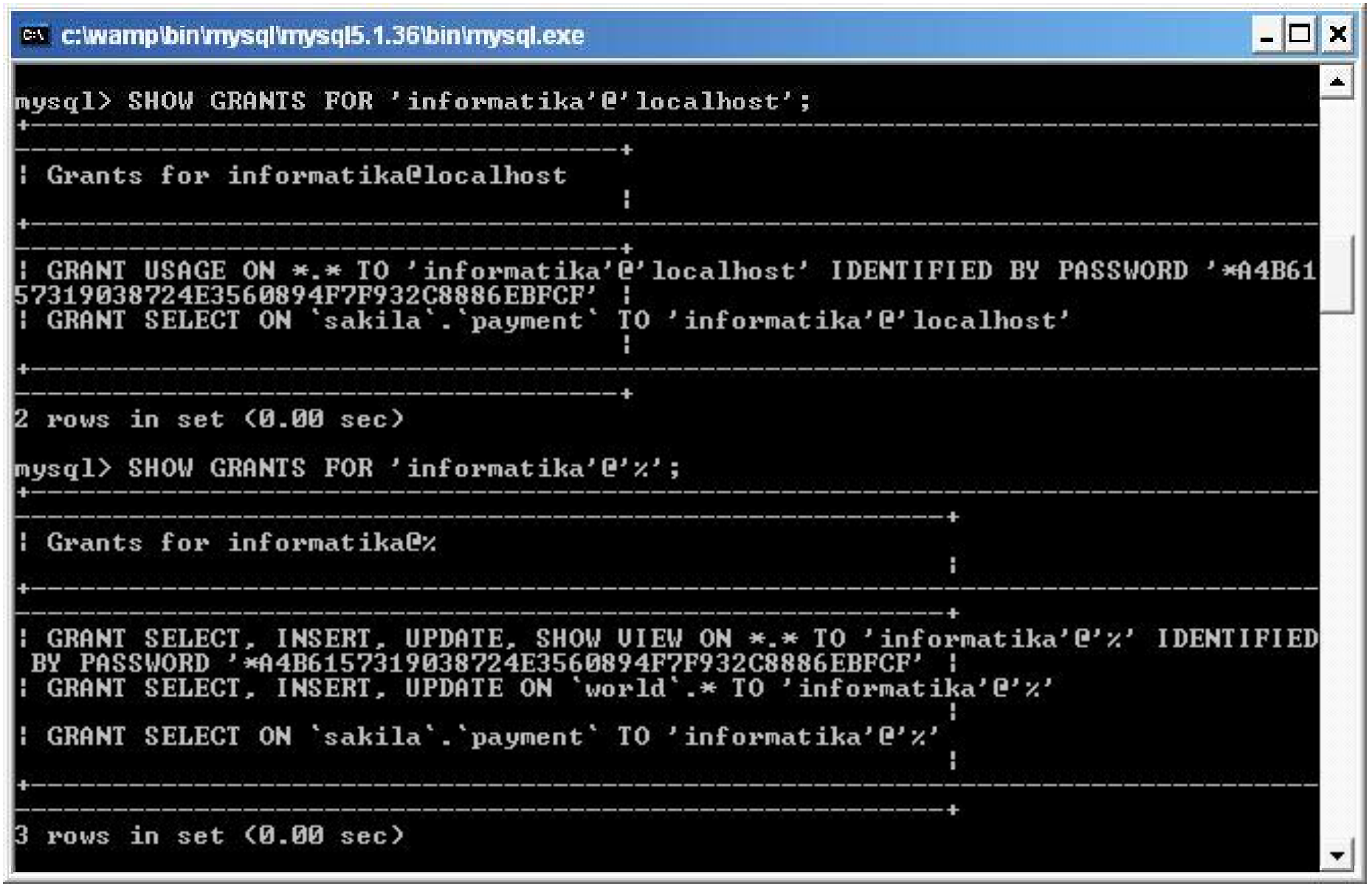


SHOW GRANTS

- Menampilkan hak akses user
- Sintaks: SHOW GRANTS FOR '<user>'@'<host>'
- Contoh:

```
SHOW GRANTS FOR 'galih'@'localhost';
```




c:\wamp\bin\mysql\mysql5.1.36\bin\mysql.exe
mysql> SHOW GRANTS FOR 'informatika'@'localhost';
+--
--+
| Grants for informatika@localhost
|
+--
--+
| GRANT USAGE ON *.* TO 'informatika'@'localhost' IDENTIFIED BY PASSWORD '*A4B61
57319038724E3560894F7F932C8886EBFCF' |
| GRANT SELECT ON `sakila`.`payment` TO 'informatika'@'localhost'
|
+--
--+
2 rows in set (0.00 sec)
mysql> SHOW GRANTS FOR 'informatika'@'%';
+--

--+
| Grants for informatika@%
|
+--

--+
| GRANT SELECT, INSERT, UPDATE, SHOW VIEW ON *.* TO 'informatika'@'%' IDENTIFIED
BY PASSWORD '*A4B61573190387 24E3560894F7F932C8886EBFCF' |
| GRANT SELECT, INSERT, UPDATE ON `world`.* TO 'informatika'@'%'
|
| GRANT SELECT ON `sakila`.`payment` TO 'informatika'@'%'
|
+--

--+
3 rows in set (0.00 sec)

SETTING PASSWORD

- Memasukkan password baru atau mengubah password untuk user yang sudah ada
- Sintaks:

  SET PASSWORD [FOR '<user>'@'<host>'] = PASSWORD('<new password>');
- Contoh:

```
SET PASSWORD FOR 'galih'@'localhost'=PASSWORD('1234');
```



FLUSH PRIVILEGES

- Berguna untuk me-*reload* tabel GRANT agar supaya dapat langsung diterapkan.

```
Flush privileges;
```



REVOKE UNTUK MENCABUT HAK AKSES

- Perintah REVOKE digunakan untuk mencabut kembali hak akses yang telah diberikan pada pengguna
- Sintaks:

  REVOKE ALL PRIVILEGES, GRANT OPTION

  FROM '<user>'@'<host>' [{,'<user>'@'<host>'}...]
- ATAU

  REVOKE <privilege> [(<column> [{, <column>}...])]

  [{, <privilege> [(<column> [{, <column>}...])]}...]

  ON {<table> | * | *.* | <database>.*}

  FROM '<user>'@'<host>' [{,'<user>'@'<host>'}...]



CONTOH REVOKE

```
GRANT SELECT, UPDATE
ON test.*
TO 'user1'@'domain1.com' IDENTIFIED BY 'pw1'
WITH GRANT OPTION MAX_QUERIES_PER_HOUR 50 MAX_UPDATES_PER_HOUR 50;
```

```
REVOKE ALL PRIVILEGES, GRANT OPTION
FROM 'user1'@'domain1.com';
```

```
REVOKE SELECT, UPDATE, GRANT OPTION
ON test.*
FROM 'user1'@'domain1.com';
```



MENGHAPUS USER

- Perintah untuk menghapus user dari sistem
- Sintaks:

  DROP USER '<user>'@'<host>' [{, '<user>'@'<host>'}...]

- Contoh:

  DROP USER 'galih'@'localhost';



END SESSION

