MySQL Security
Past and Present

Georgi Kodinov
Team Lead,
MySQL Server General Team

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle’s products remains at the sole discretion of Oracle.

Agenda

- Introduction to MySQL Security
- What’s New in MySQL 5.6 ?

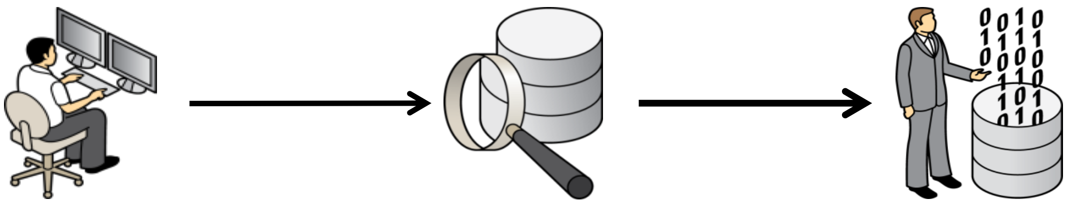
MySQL Security
Logical Model

User Accounts

- Two parts : ‘user name’@’host name’.
- Can use wildcards in host names: empty (‘’), ‘%’ and ‘_’.
- Host part can be IP addresses/net masks too.
- Authentication methods are pluggable.
- No passwords are stored in tables, only hashes.
- GRANT commands can create users too.
- Account can “inherit” other, more general accounts.

“Logged in” and “Current” User IDs

- Joe connects from the data entry department
- MySQL finds a record *”@’%.dataentry’* and authenticates Joe to it
- Joe can use all privileges granted to *”@’%.dataentry’*



user()

Users table

current_user()

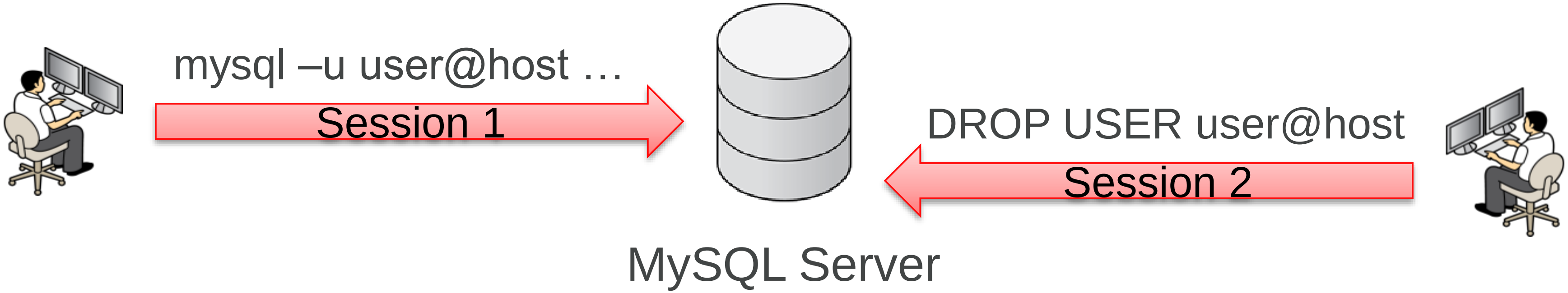
joe@p1.dataentry

”@%.dataentry

Inheriting rights from “less specific” accounts

- Setup
  - CREATE USER user@localhost;
  - GRANT SELECT ON mysql.user to user@’%’;
- Login as **user@localhost** and:
  - “SELECT * FROM mysql.user” works !
  - But “SHOW GRANTS” doesn’t show the SELECT privilege inherited from user@’%’
  - Observe two distinct users defined: user@localhost and user@%

QUIZ : How Does “DROP USER” Work ?



mysql –u user@host …

Session 1

DROP USER user@host

Session 2

MySQL Server

What Happens ?

A. DROP USER fails with an “active session” error

B. Session 1 is terminated

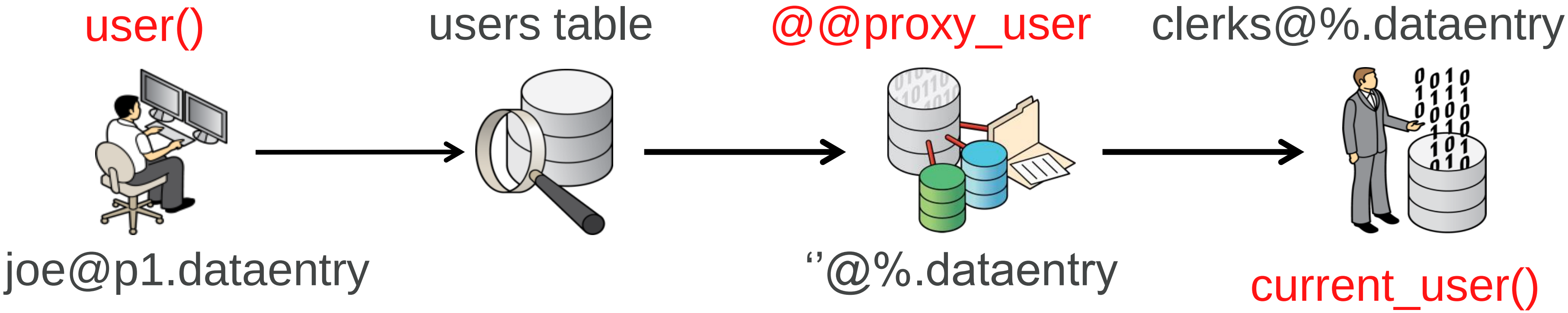
C. Session 1 is unaffected by DROP USER

Using Plugins to Authenticate

- All MySQL users authenticate via plugins
- Plugins come in pairs: client + server
- The server picks the plugin pair
- Built-in plugins
- Backward compatible
- Plugins can “re-route” (proxy) to other MySQL user ids

Proxy Users

- Joe connects and gets resolved to ’’@’%.dataentry’
- ’’@’%.dataentry’ uses a plugin to authenticate Joe
- The plugin “re-routes” Joe to ’clerks’@’%.dataentry’



user()

users table

@@proxy_user

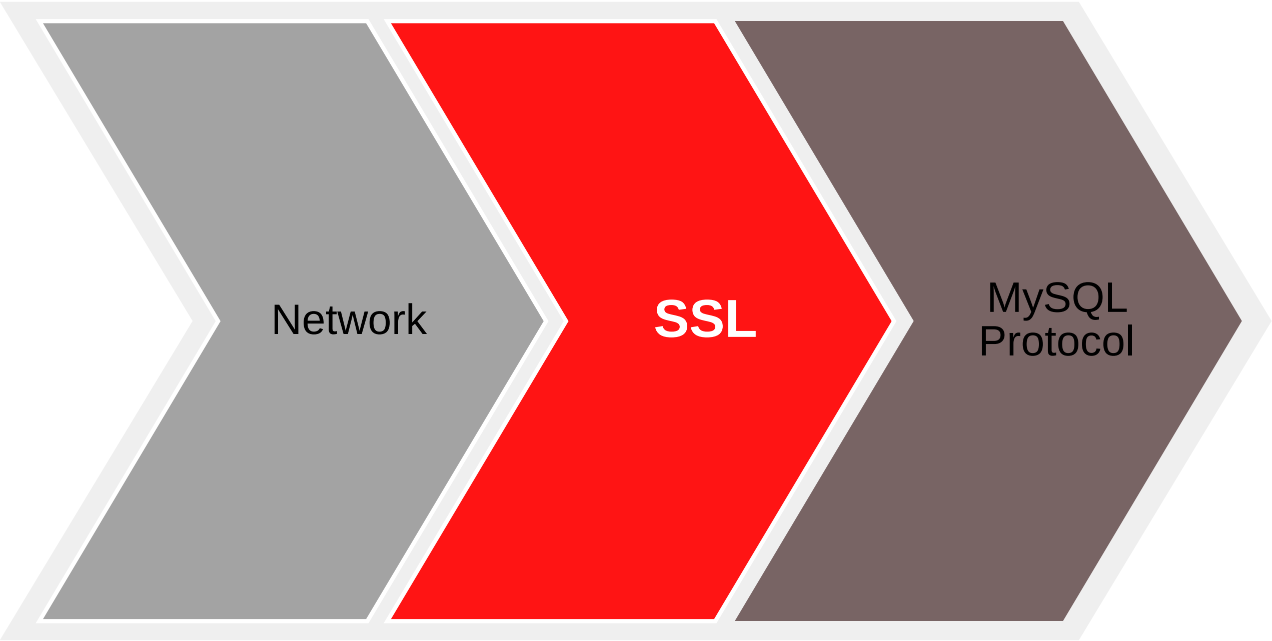
clerks@%.dataentry

joe@p1.dataentry

‘’@%.dataentry

current_user()

SSL is a Separate Layer



Privileges available in MySQL

- **Server** : *USAGE, SHUTDOWN, SUPER, CREATE …*
- **Database** : *CREATE, DROP, EXECUTE, SELECT*, …
- **Table** : *INSERT, SELECT*, …
- **Column** : *INSERT, SELECT, UPDATE*
- **Stored routine** : *EXECUTE, CREATE ROUTINE, ALTER ROUTINE*
- **User** : *PROXY*

Granting Privileges

- Always granted to a user+host combination
- Only *DROP USER* is guaranteed to clean up properly !
- *INSERT* can be granted on a column subset
- Privileges are checked against *current_user()*

MySQL Security
Physical Model

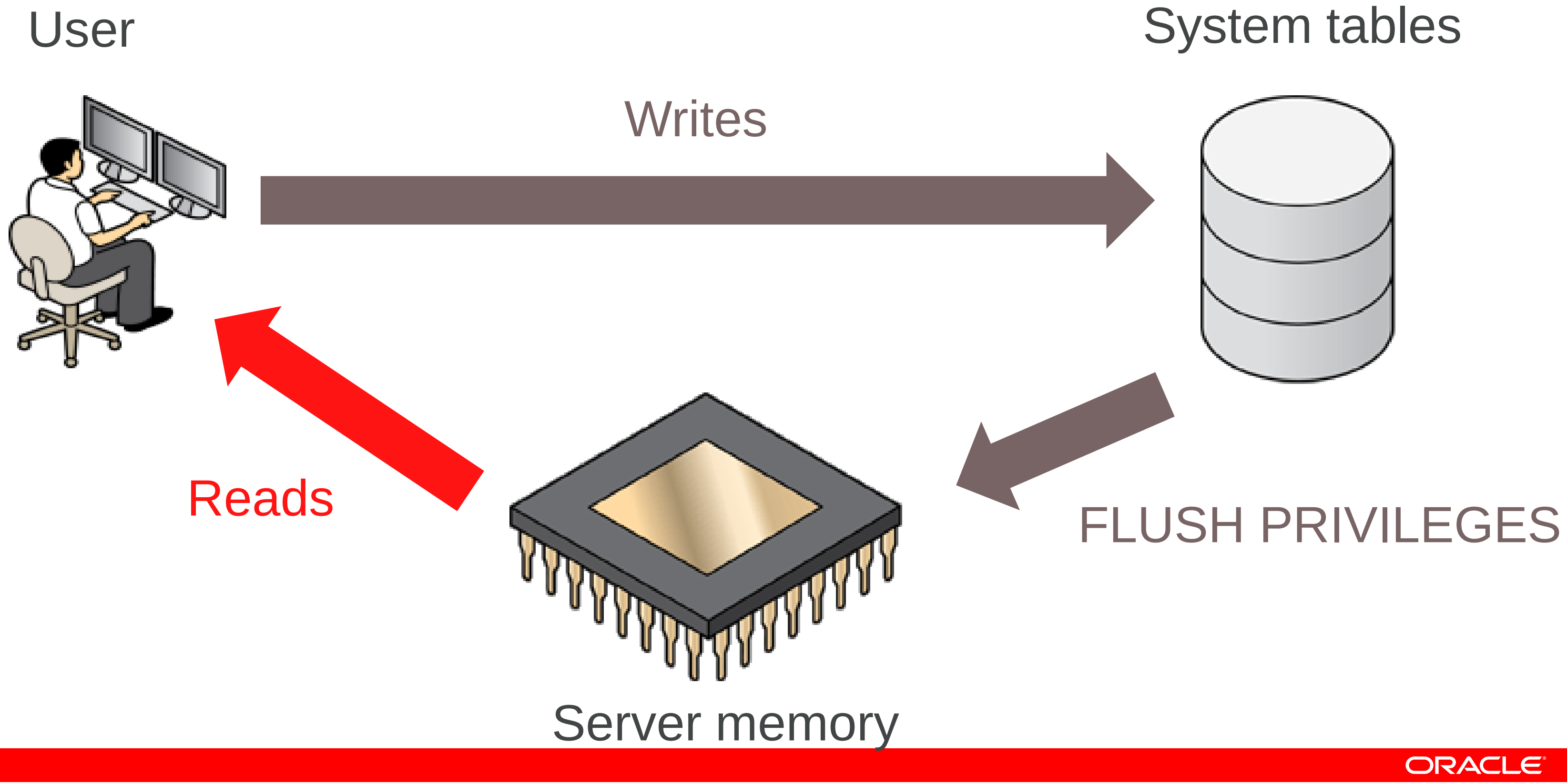
Authentication Tables

Table	Usage
user	User accounts definitions
plugin	Definitions for authentication plugins

Authorization Tables

Table	Applicable to
user	Users
db	Databases
tables_priv	Tables
columns_priv	Table columns
procs_priv	Stored procedures and functions
proxies_priv	User proxying by plugins

Memory Cache for User Data


User

System tables

Writes

Reads

FLUSH PRIVILEGES

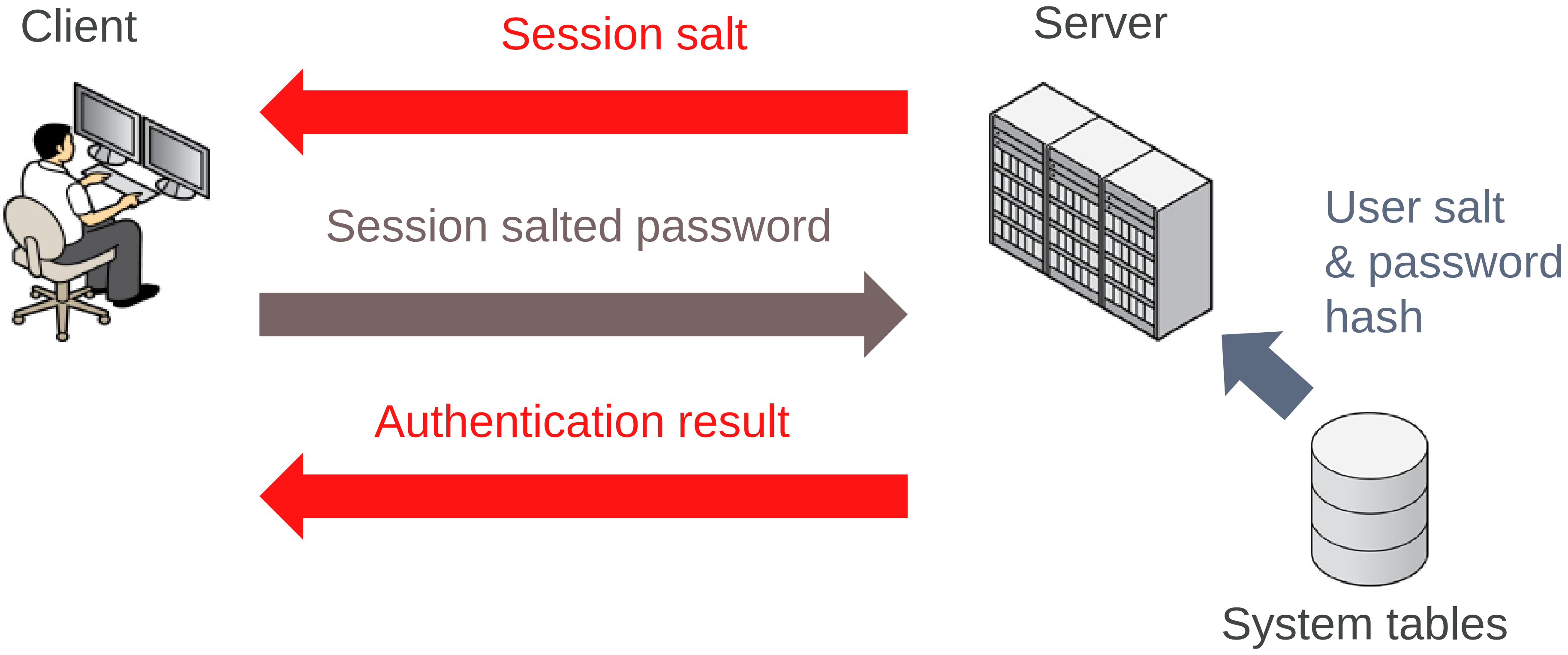
Server memory

WHAT'S NEW IN MYSQL 5.6 ?

New Authentication Method

- Adds per-user password salt
- Modern hash algorithms (SHA256) for an extensible format
- Makes use of hardware acceleration for cryptography calculations
- Falls back to RSA to encrypt the password when no SSL

How does SHA256 work



Client

Session salt

Server

Session salted password

User salt
& password
hash

Authentication result

System tables

Change Password at Next Login

- Safe way to force password reset
- Introduces *ALTER USER … EXPIRE*
- New field in *mysql.user*

Auditing Password Security

- New plugin API
- Works for *PASSWORD(), CREATE USER* etc
- Strength evaluation aid for GUIs : *validate_password_strength()*

Policy	Length	Mixed case + digits + special chars	Dictionary
Low	✓		
Medium	✓	✓	
High	✓	✓	✓

Enterprise Auditing

- Versatile standards based output (XML)
- Designed for minimum impact (Flexible disk flushing/buffering)
- Supports log file rotation
- Part of the MySQL Enterprise offering
- Supported by MySQL Workbench GUI tool
- Works with Oracle Audit Vault

Alternative to Passwords on the Command Line

- Wallet type of design
- Named “login paths” : user/password/host combos
- An alternative to passwords in scripts
- Use encryption to make it harder to abuse the password files
- Designated configuration editor command line tool

Connect Strings Support

- Arbitrary strings sent by clients at login time
- Some MySQL clients send client PIDs, command lines, platforms, versions, program names, OSes, etc.
- Client programs can supply more via mysql_options4()
- PERFORMANCE_SCHEMA tables to read the data

Refactoring

- Obfuscate passwords from slow query log (WL#5706)
- Use the C++ Standard Template Library (WL#5825)
- Use standard random generator (WL#5605)
- Replace custom encryption code with approved packages (WL#6008)
- Support SSL certificate revocation lists (Bug#31224)
- Support SSL keys with pass phrases (Bug#44559)

ORACLE®