
ORACLE®

# 第5回：セキュリティ基礎編

## 初心者向け！MySQL入門

Ryusuke Kajiyama / 梶山隆輔
MySQL Sales Consulting Senior Manager, Asia Pacific & Japan

# Safe Harbor Statement

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.

# 43%

過去1年間にデータへの不正
アクセス被害にあった企業の割合

Source: Ponemon Institute, 2014

# 大規模な不正アクセス

2013年には5億レコード以上の個人情報が流出。レコード数は1年で5倍増

**77%**

Webサイトの脆弱性
さらに8件に1件は
深刻な脆弱性

**8**

2013年に1,000万レコード以上の被害に遭った不正アクセス事件

2013年の不正アクセス件数は62%増

Source: Internet Security Threat Report 2014, Symantec

# データベースの脆弱性と対策

- 設定の不備
  - デフォルト設定からの変更
- 過剰な権限の付与
  - 権限設定ポリシー
- 貧弱なアクセス制御
  - 管理者権限付与の限定
- 貧弱な認証
  - 複雑なパスワードの強制
- 貧弱な監査
  - コンプライアンス & 監査ポリシー
- 暗号化の不足
  - データ、バックアップ & ネットワークの暗号化
- 認証情報の不十分な管理
  - mysql_config_editor利用, Key Vaults
- 無防備なバックアップデータ
  - バックアップデータの暗号化
- 監視の不備
  - ユーザ & オブジェクト監視
- アプリケーションの脆弱性
  - データベースファイアウォール

# データベースへの攻撃パターン

- SQLインジェクション
  - 対策: データベースファイアウォール、ホワイトリスト、入力バリデーション
- バッファオーバーフロー
  - 対策: データベースソフトウェアの定期的な更新、データベースファイアウォール、ホワイトリスト、入力バリデーション
- ブルートフォースアタック (総当たり攻撃)
  - 対策: 設定回数を超えた回数ログインを試みたアカウントをロック
- ネットワーク傍受
  - 対策: 全ての接続とデータ転送に SSL/TLS を必須化
- マルウェア
  - 対策: 強固なアクセスコントロール、接続元IPアドレスの制限、デフォルト設定の変更

# データベースへ攻撃による問題

- 情報流出: クレジットカードやその他の個人情報の不正な取得
  - 対策: データやネットワークの暗号化、強固なアクセス制御
- DoS(サービス拒否)攻撃: 負荷の極めて高いクエリの実行
  - 対策:各種のリソース利用制限 (最大接続数、セッション数、タイムアウトなど)
- 権限の昇格: 管理者の認証情報の不正な取得
  - 対策: 認証の強化、アクセス制御、監査
- なりすまし: 他社の認証情報を不正に利用
  - 対策: 強力なアカウントおよびパスワードポリシー
- 不正な改変: データの変更、トランザクション記録の削除
  - 対策: 認証の強化、監査、監視、バックアップ

# DBAによるセキュリティ対策

- アクセス権を得る必要のあるユーザーのみが、
  権限を取得出来るよう確実にする
- ユーザーおよびアプリケーションが、何をできるかを制限する
- ユーザーおよびアプリケーションが、
  どこからデータにアクセスできるかを制限する
- 何が、いつ発生したかを適切に監視しておく
- バックアップが安全でセキュアに取得されている事を
  確実にしておく
- 攻撃経路を最小化しておく

# ファイルシステム


- 実行バイナリの所有者はroot
- データディレクトリの所有者は一般ユーザ
  (例 mysql)
  - 他のユーザがOS上でこのディレクトリにアクセスできないように設定すること
- ログディレクトリの所有者は一般ユーザ (例 mysql)
- 一般ユーザ(例 mysql)はログイン不可にしておく
- ソケットファイルにアクセス可能であること
  (全てのユーザから参照可能に)

# ファイルシステム

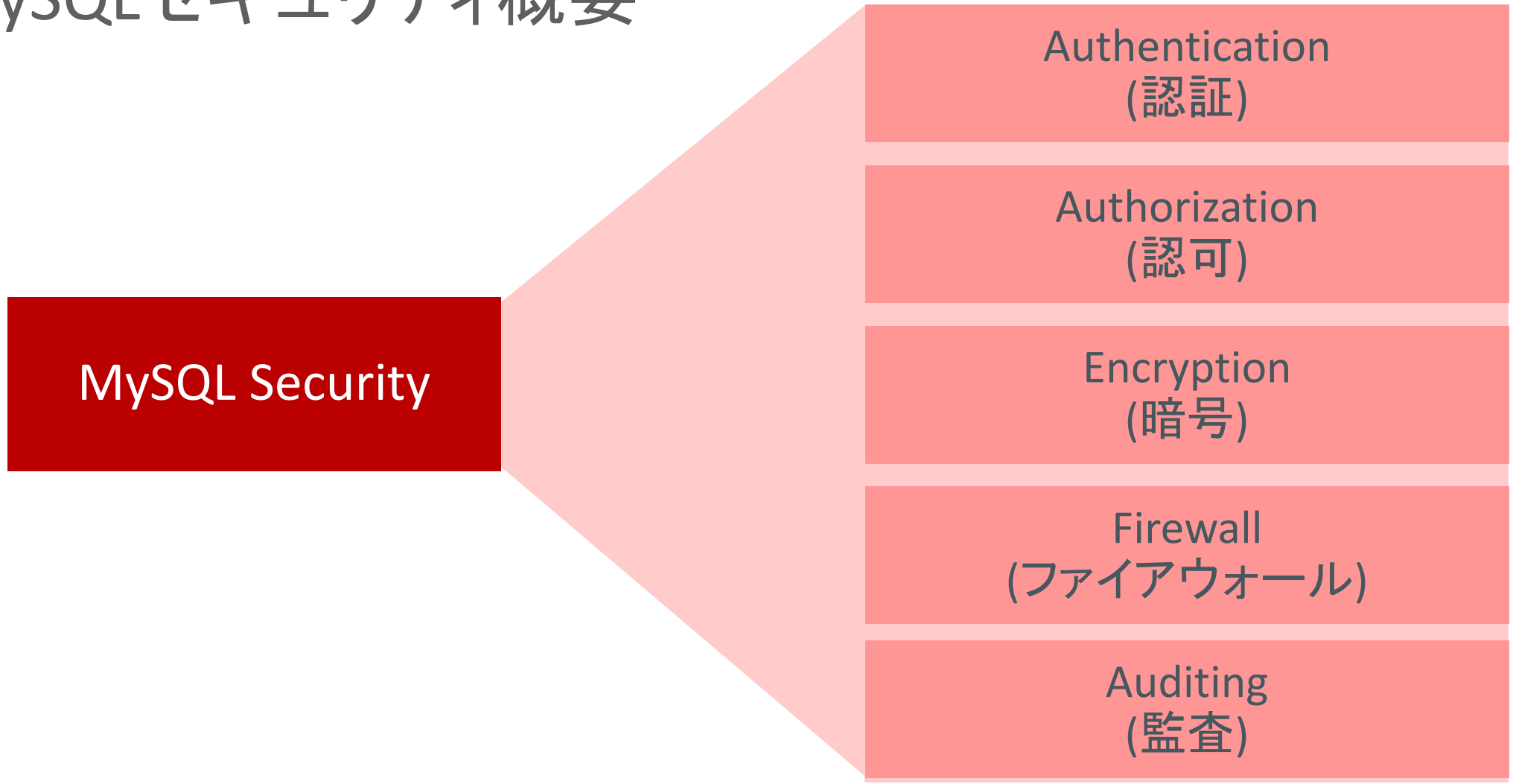
```
$ groupadd mysql
$ useradd -g mysql mysql
$ chown -R root:mysql $MYSQL_HOME
$ chmod 750 $MYSQL_HOME
$ chown -R mysql:mysql $MYSQL_HOME/data
$ chmod 700 $MYSQL_HOME/data
```

NOTE: デフォルトのログ出力先は$MYSQL_HOME/data

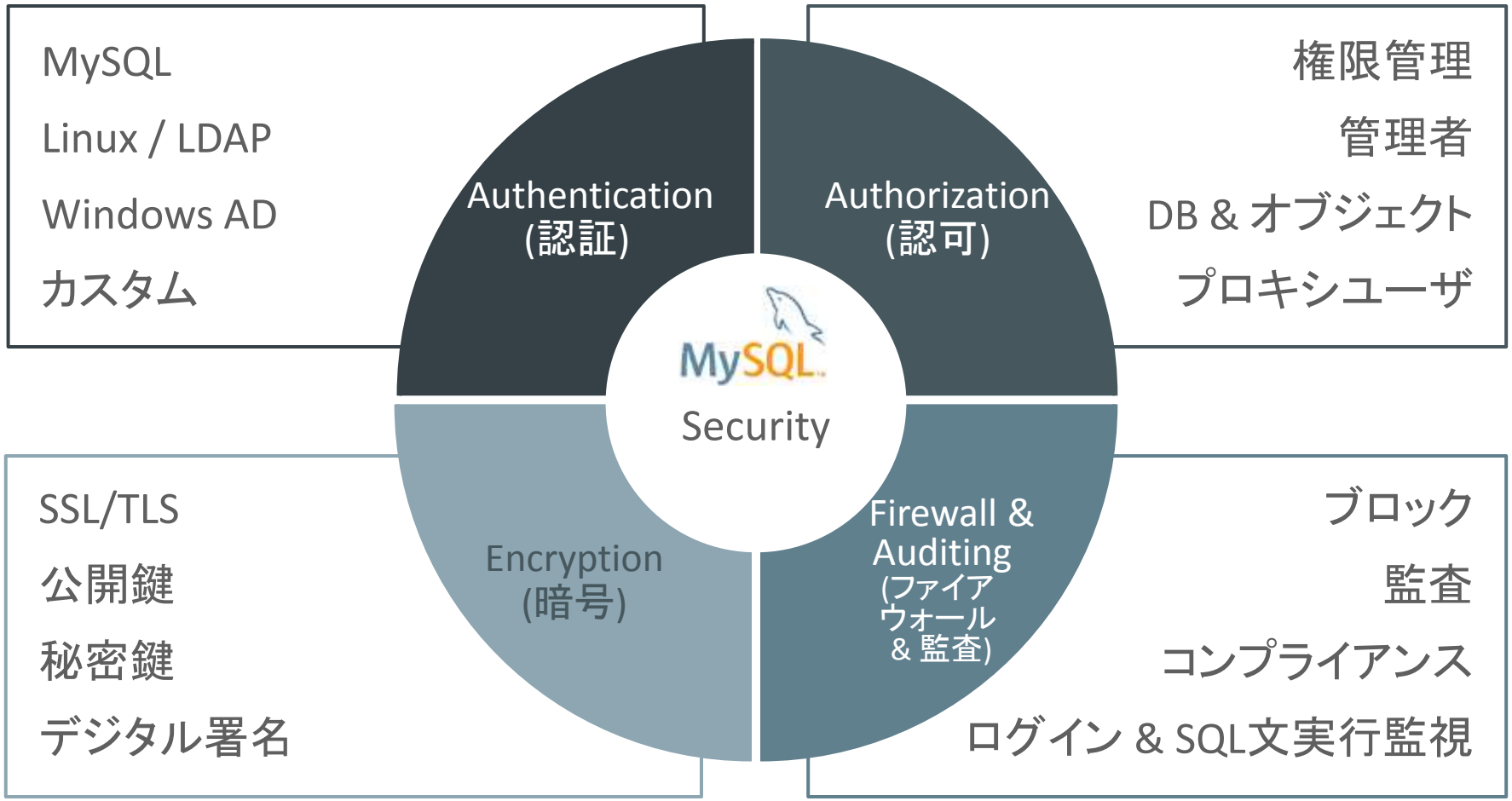
# データベース

- MySQLサーバは ‘root’ ユーザ以外で起動
  - デフォルトでは‘root’ユーザでの起動不可
  - ‘mysql’ユーザとして起動する場合 : --user=mysql
  - MySQLサーバの起動ユーザはOS上で必要以上の権限やファイルシステムのアクセス許可を与えないこと
  - ファイルシステム上のファイル操作にはFILE権限が必要
- デフォルトでは全ネットワークインターフェースを使用
  - 特定のインターフェースを使用する場合 : --bind-address
  - TCP/IP経由でのアクセスを無効にする場合 : --skip-networking

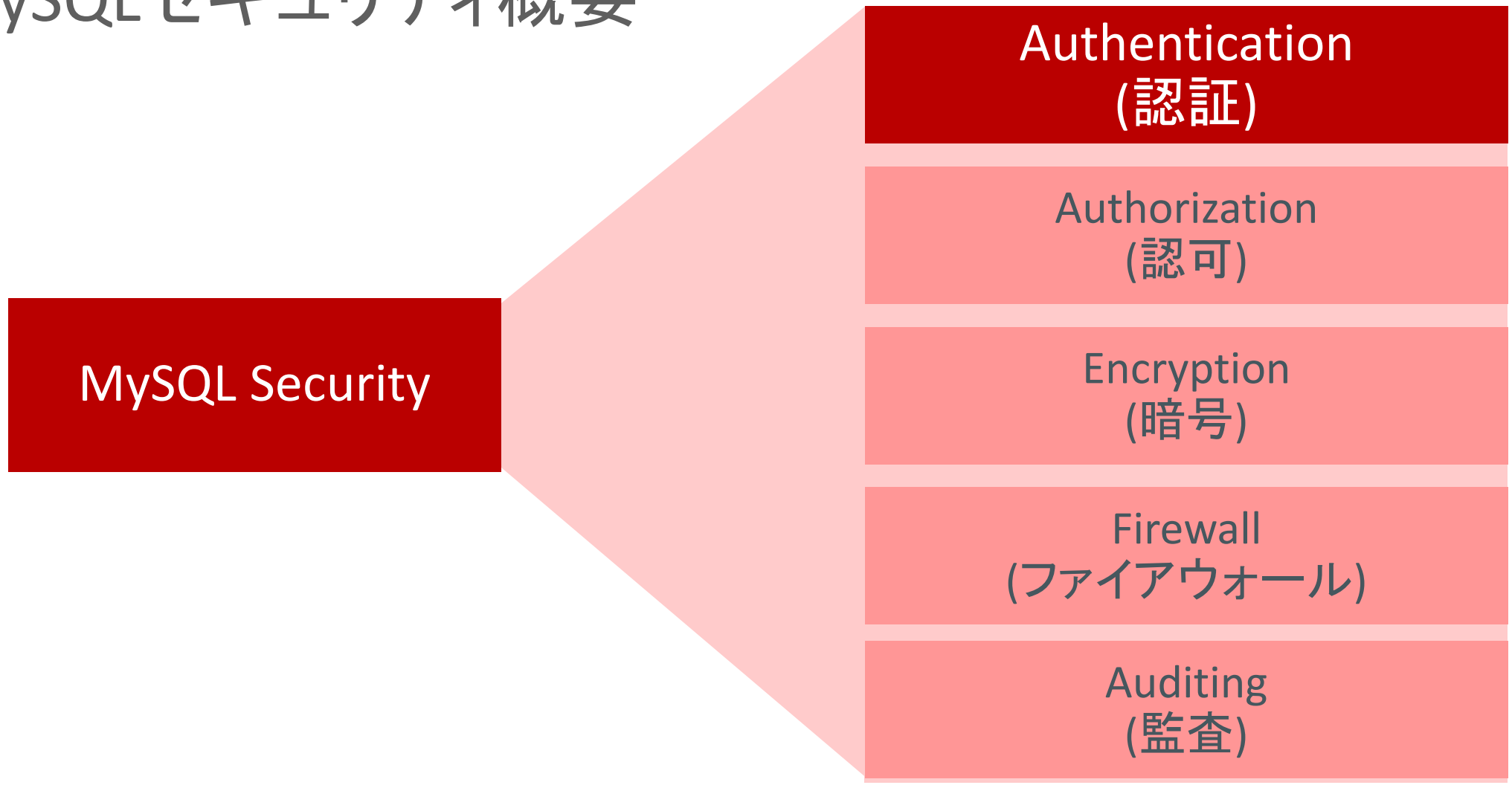
# MySQLセキュリティ概要



# MySQLセキュリティ概要



# MySQLセキュリティ概要

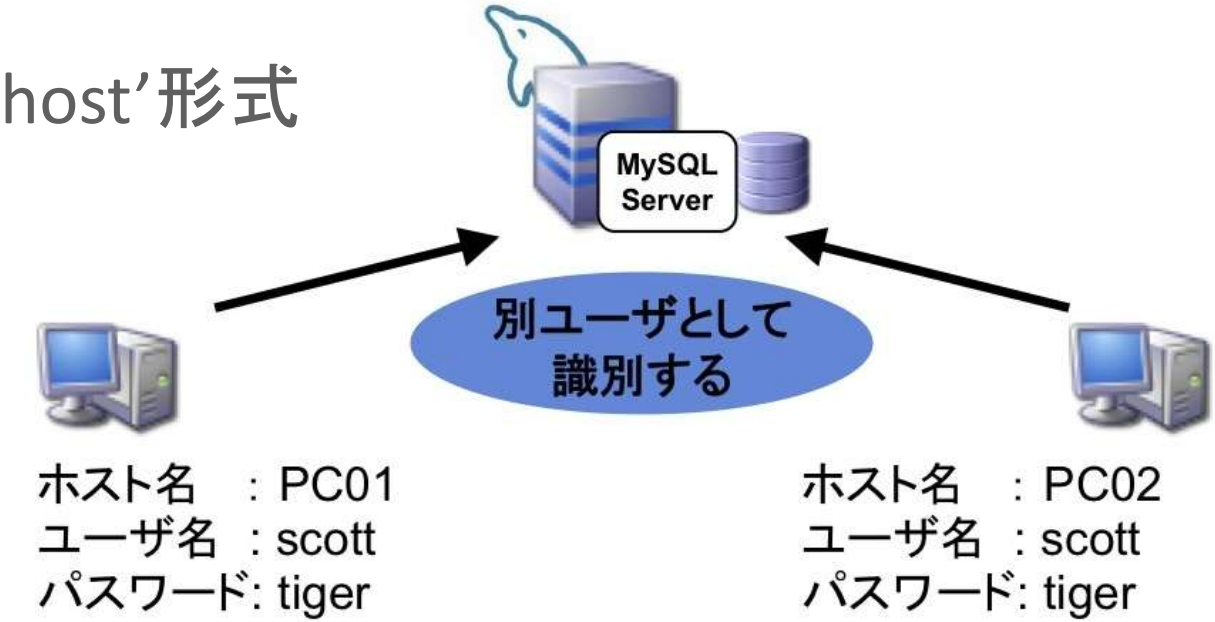


# MySQLにおけるユーザ認証

- 管理者権限 (データベース全体に関する権限)
- データベース権限 (スキーマやデータに関する権限)
- セッション制限およびオブジェクト権限
- ユーザ権限に加えさらに粒度の細かい制御
  - データベースの作成、変更および削除
  - テーブルの作成、変更および削除
  - インデックスの作成、変更および削除
  - INSERT, SELECT, UPDATE, DELETE の各SQL文の実行
  - ストアドプロシージャの作成、変更、削除、および実行時権限の設定

# MySQLのユーザアカウント
## ユーザ名と接続元ホストの組み合わせ

- CREATE USER文でユーザを作成
- ユーザアカウントは、'username'@'host'形式
- 接続元ホストの指定
  - ホスト名またはIPアドレス
  - %をワイルドカードとして利用可能
  - サブネットでの指定も可能
- 権限の付与/剥奪は GRANT文/REVOKE文にて




```
mysql> CREATE USER 'scott'@'localhost' IDENTIFIED BY 'tiger';
mysql> CREATE USER 'scott'@'PC%' IDENTIFIED BY 'tiger';
```

# MySQLのユーザーアカウント管理テーブル

- MySQLデータベース内のuserテーブルにてユーザアカウントを管理
- 同一ユーザ名でも接続元ホストが異なる場合は別アカウントとできる

```
root@localhost [mysql]>select user,host,password from mysql.user;
+---------------+-----------------------------+-------------------------------------------+
| user          | host                        | password                                  |
+---------------+-----------------------------+-------------------------------------------+
| root          | localhost                   | *A41ECFBE1191DDE4713F2B6F5A6CD5D0D0D5DC35 |
| root          | centos01                    | *A41ECFBE1191DDE4713F2B6F5A6CD5D0D0D5DC35 |
| GTID_SSL_USER | 192.168.56.%                | *84AE91CE95DAA59A02F658041290FEECF1BEE392 |
| admin         | %                           | *A41ECFBE1191DDE4713F2B6F5A6CD5D0D0D5DC35 |
| root          | 192.168.56.%                | *A41ECFBE1191DDE4713F2B6F5A6CD5D0D0D5DC35 |
| audited_user  | 192.168.56.0/255.255.255.0  | *CED32AC7E202DBC0858C9E8935348E7ED3E083D4 |
| GTID_USER     | 192.168.56.%                | *A41ECFBE1191DDE4713F2B6F5A6CD5D0D0D5DC35 |
| sec_password  | 192.168.56.%                | *6FDB5274DE9FC54EDC81BD5909DEBE93CF739D48 |
+---------------+-----------------------------+-------------------------------------------+
```

ユーザー名　　アクセス元 Host / IP　　パスワード

～MySQL 5.6x

# 不要なアカウントの削除

**[Drop the Extra Roots]** root@localhost [mysql]>DROP USER root@'127.0.0.1', root@'::1';

```
root@localhost [mysql]>SELECT user,host,plugin,password,password_expired FROM mysql.user;
+--------------+---------------------------+-----------------------+-------------------------------------------+------------------+
| user         | host                      | plugin                | password                                  | password_expired |
+--------------+---------------------------+-----------------------+-------------------------------------------+------------------+
| root         | localhost                 |                       | *A41ECFBE1191DDE4713F2B6F5A6CD5D0D0D5DC35 | N                |
| root         | centos01                  |                       | *A41ECFBE1191DDE4713F2B6F5A6CD5D0D0D5DC35 | N                |
| root         | 127.0.0.1                 |                       |                                           | N                |
| root         | ::1                       |                       |                                           | N                |
| admin        | %                         | mysql_native_password | *A41ECFBE1191DDE4713F2B6F5A6CD5D0D0D5DC35 | N                |
| root         | 192.168.56.%              | mysql_native_password | *A41ECFBE1191DDE4713F2B6F5A6CD5D0D0D5DC35 | N                |
| audited_user | 192.168.56.0/255.255.255.0 | mysql_native_password | *CED32AC7E202DBC0858C9E8935348E7ED3E083D4 | N                |
| GTID_USER    | 192.168.56.%              | mysql_native_password | *A41ECFBE1191DDE4713F2B6F5A6CD5D0D0D5DC35 | N                |
+--------------+---------------------------+-----------------------+-------------------------------------------+------------------+
8 rows in set (0.00 sec)

root@localhost [mysql]>DROP USER root@'127.0.0.1', root@'::1';
Query OK, 0 rows affected (0.01 sec)

root@localhost [mysql]>SELECT user,host,plugin,password,password_expired FROM mysql.user;
+--------------+---------------------------+-----------------------+-------------------------------------------+------------------+
| user         | host                      | plugin                | password                                  | password_expired |
+--------------+---------------------------+-----------------------+-------------------------------------------+------------------+
| root         | localhost                 |                       | *A41ECFBE1191DDE4713F2B6F5A6CD5D0D0D5DC35 | N                |
| root         | centos01                  |                       | *A41ECFBE1191DDE4713F2B6F5A6CD5D0D0D5DC35 | N                |
| admin        | %                         | mysql_native_password | *A41ECFBE1191DDE4713F2B6F5A6CD5D0D0D5DC35 | N                |
| root         | 192.168.56.%              | mysql_native_password | *A41ECFBE1191DDE4713F2B6F5A6CD5D0D0D5DC35 | N                |
| audited_user | 192.168.56.0/255.255.255.0 | mysql_native_password | *CED32AC7E202DBC0858C9E8935348E7ED3E083D4 | N                |
| GTID_USER    | 192.168.56.%              | mysql_native_password | *A41ECFBE1191DDE4713F2B6F5A6CD5D0D0D5DC35 | N                |
+--------------+---------------------------+-----------------------+-------------------------------------------+------------------+
6 rows in set (0.00 sec)
```

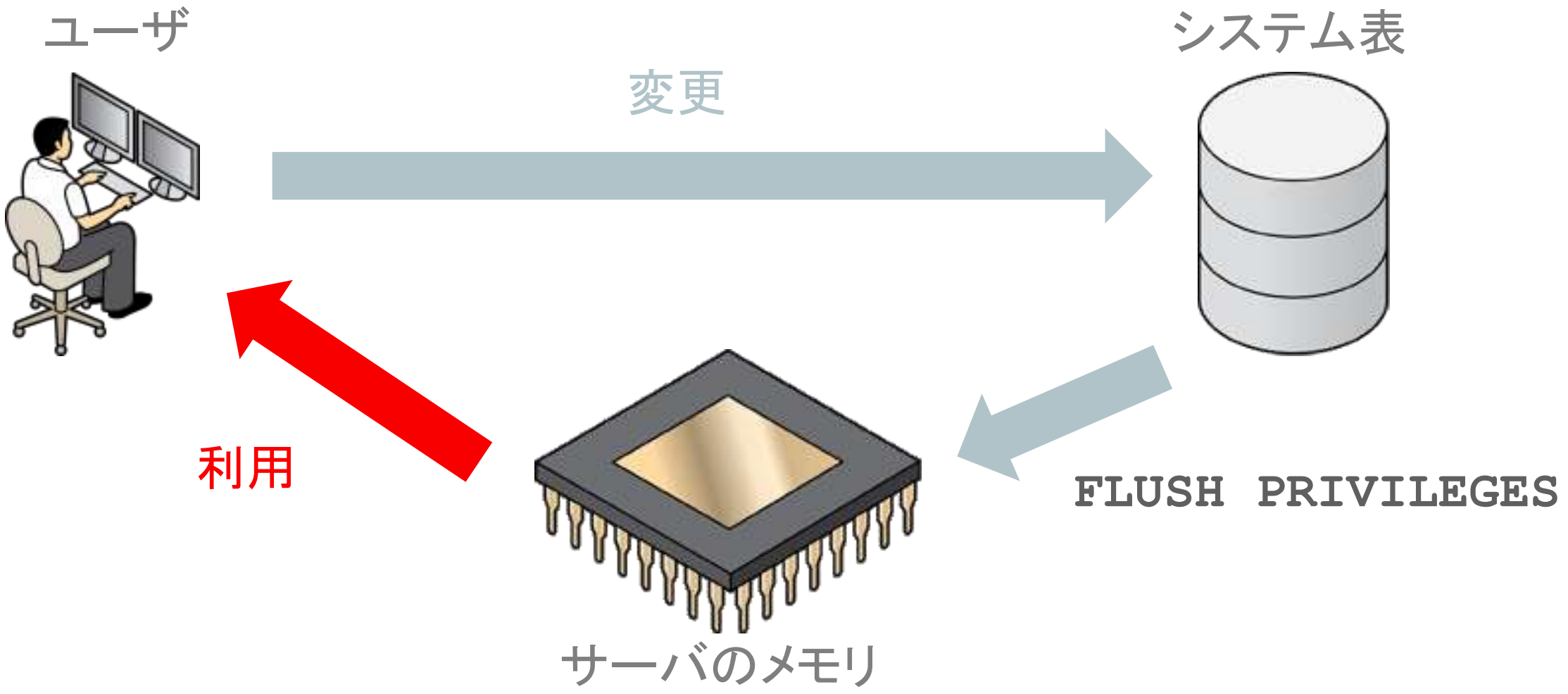
# MySQLの古典的ユーザ管理方法[非推奨]

- 以前のバージョンでのユーザ追加方法 (5.0未満)
  [現在は推奨されていない方法]

```
mysql> INSERT INTO mysql.user(...) VALUES (...);
mysql> FLUSH PRIVILEGES;
```

- usersテーブルを含むmysqlデータベースに対する
  DMLでの直接操作は推奨されていない

# メモリにキャッシュされたユーザ情報を利用

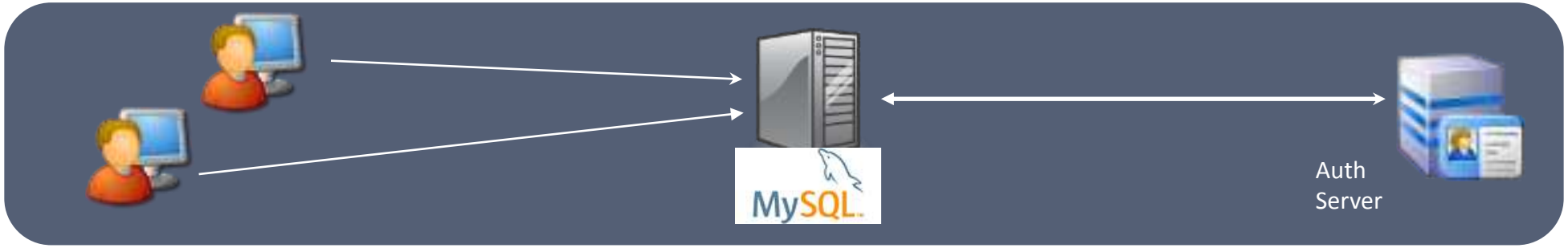


# MySQL 認証方式オプション

- ビルトインの認証方式 (デフォルト)
  - userテーブルにユーザアカウントとパスワードを格納
- X.509
  - クライアント証明書にてサーバが認証
- MySQLネイティブ, SHA 256パスワードプラグイン
  - ネイティブのSHA1かプラグインでの SHA-256 によるハッシュが選択可能
- MySQL Enterprise Authentication
  - Microsoft Active Directory
  - Linux PAMs (Pluggable Authentication Modules)
    - LDAPなどをサポート
- カスタム認証

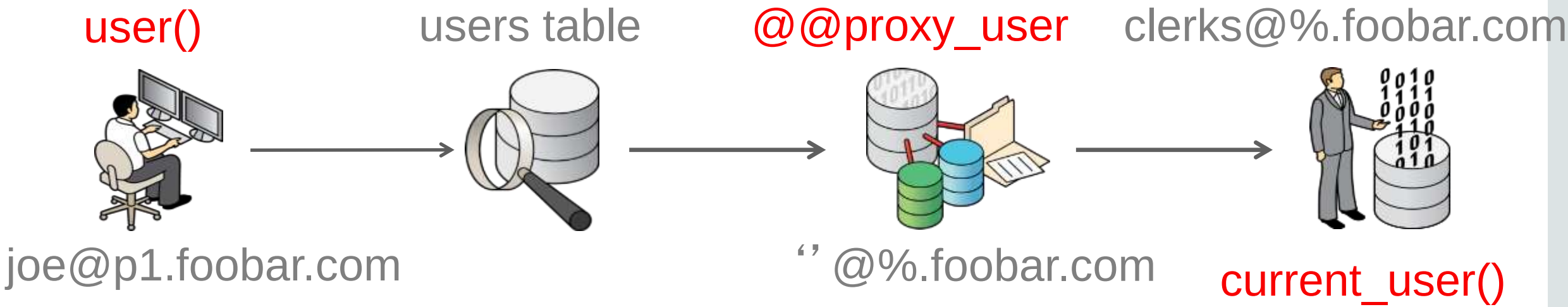
# MySQL Enterprise Authentication

- PAM (Pluggable Authentication Modules)
  - 外部の認証システムを利用
  - 標準的なインターフェース (Unix, LDAP)
- Windows
  - Windowsのネイティブサービスを利用
  - Windowsログイン時に認証済みの情報を利用可能 (Windows Active Directory)
- Pluggable Authentication API



# Authentication APIを利用したプラグイン

- Joeの接続は*''@'%.foobar.com'*にマッピングされる
- *''@'%.foobar.com'*を使ってプラグインでの認証を行う
- プラグインがJoeを*'clerks'@'%.foobar.com'*としてアクセスさせる



# MySQLパスワードポリシー

- パスワード無しのアカウント
  - 全てのアカウントにパスワードを設定して不正な利用を防ぐ必要有り
- パスワード検証プラグイン
  - より強力なパスワードを強制
- パスワードの失効/ローテーション
  - 次回ログイン時にパスワードのリセットが必要
- アカウントのロック (MySQL 5.7より利用可能)

# パスワード

- MySQL 5.6までのパスワードの変更は

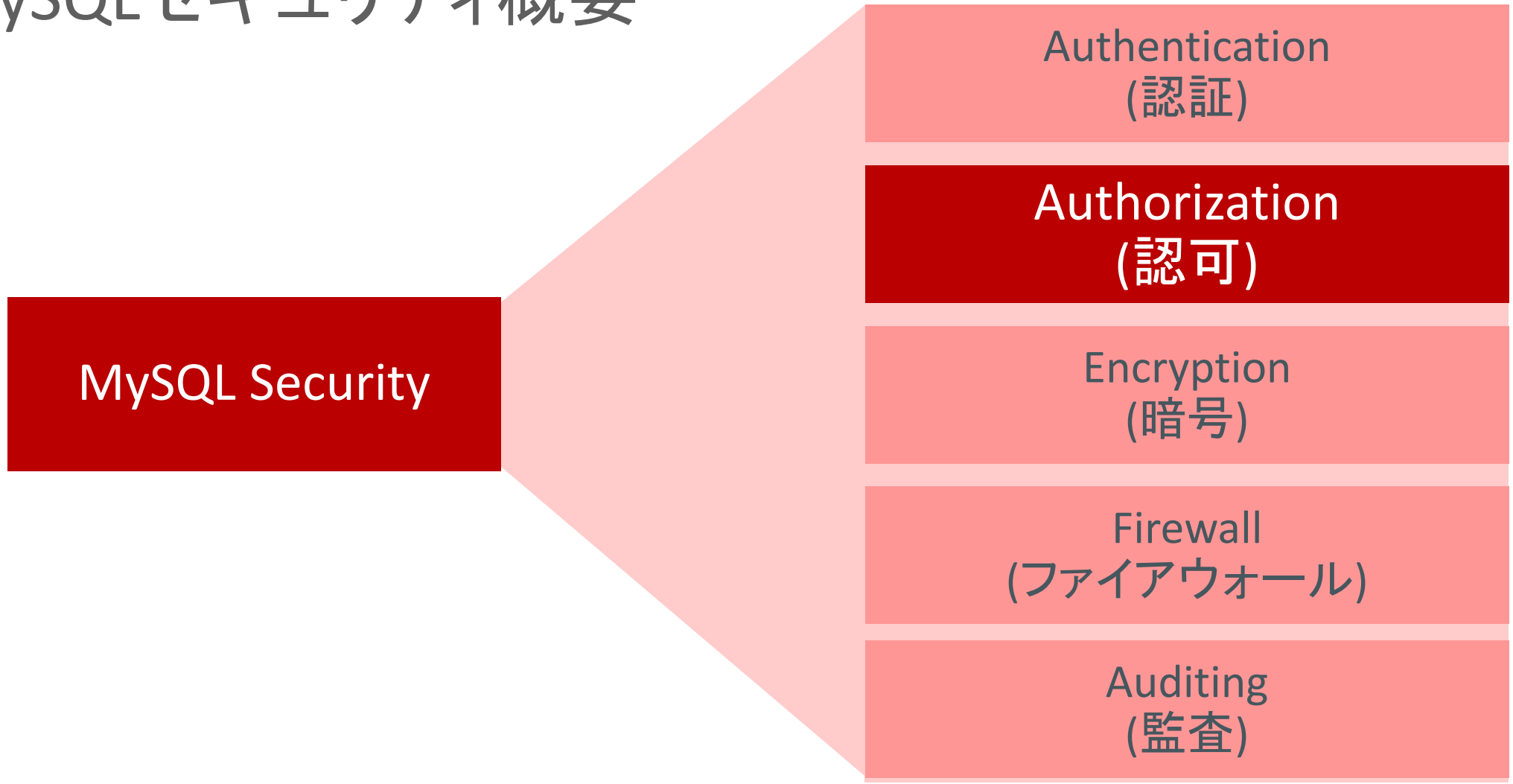
```
mysql> SET PASSWORD [FOR user] = PASSWORD('some password');
```

  - SETコマンドでのパスワード変更はMySQL 5.7では非推奨
  - MySQL 5.7では下記

```
mysql> ALTER USER user IDENTIFIED BY 'auth_string';
```

- 注: 4.1にてパスワードのハッシュ化方式変更
  - 古いパスワードハッシュ化の利用が必要な場合は--old-passwordsオプション利用
  - MySQL 5.7では古いパスワードハッシュ化は非サポート

# MySQLセキュリティ概要



# MySQL権限管理テーブル

## user

- ユーザアカウント
- グローバル権限

## db

- データベースレベル権限
- データベース、テーブル、オブジェクト
- ユーザ、ホスト

## tables_priv

- テーブルレベル権限
- テーブルおよび列

## columns_priv

- 特定の列

## procs_priv

- ストアドプロシージャ
- ファンクション
- 各ストアドプログラム単位

## proxies_priv

- プロキシユーザ
- プロキシ権限

# MySQL権限管理

- 連続的にチェック
  - 設定
  - ユーザ
  - 許可および権利
- 監査 & レビューのポイント
  - Who – どのような操作を行うことが想定されているのか
  - What – 何を制限することを想定されているのか
  - When – 想定されていないタイミングでの利用がないか
  - Where – どのホストから接続されることが想定されているのか
- MySQLでは確実に制限を行うための設定をシンプルに利用可能

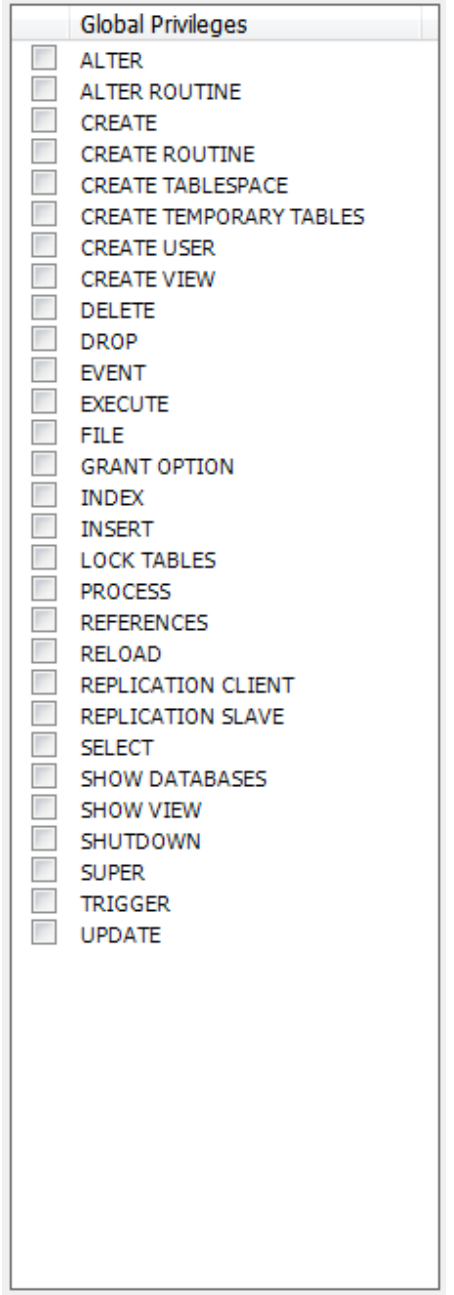
# Grant/Revoke

- 権限が影響するタイミングは
  - テーブルとカラム：データ参照/変更時
  - データベース：USE <dbname>実行時
  - グローバル権限とパスワード：接続時

http://dev.mysql.com/doc/refman/5.6/ja/privilege-changes.html

# グローバル権限

- SUPER
  - CHANGE MASTER, KILL, PURGE MASTER LOGS, SET GLOBAL
- SHUTDOWN
- RELOAD
- PROCESS
  - SHOW ENGINE INNODB STATUSの実行にも必要
- FILE
- ALL
- WITH GRANT OPTION



# 付与されている権限の確認

- ユーザに付与されている権限の確認コマンド

```
mysql> SHOW GRANTS [FOR user]
```

http://dev.mysql.com/doc/refman/5.6/ja/show-grants.html

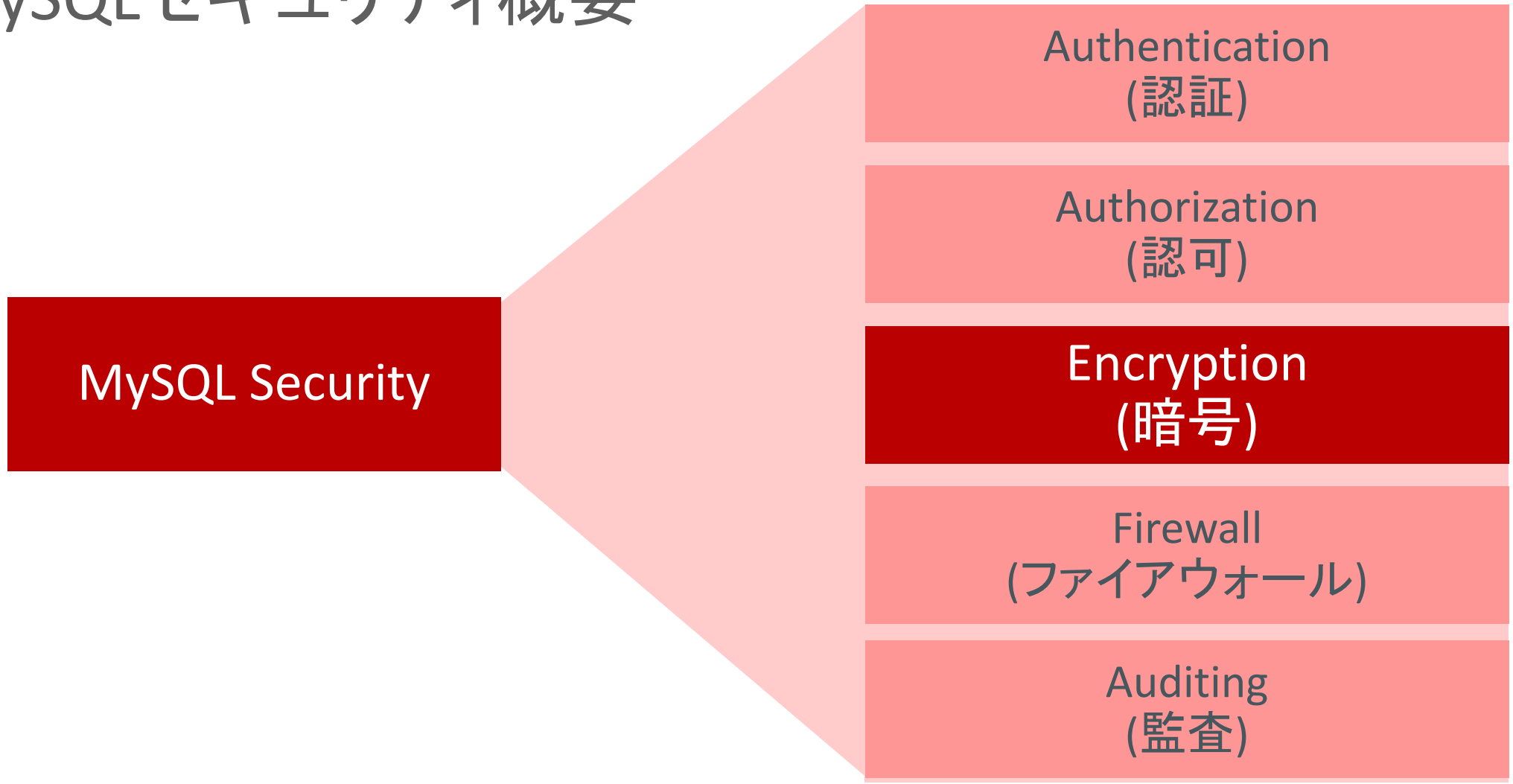
# 利用するリソースの制限

- **MAX_QUERIES_PER_HOUR**
- **MAX_UPDATES_PER_HOUR**
- **MAX_CONNECTIONS_PER_HOUR**
- **MAX_USER_CONNECTIONS**

http://dev.mysql.com/doc/refman/5.6/ja/user-resources.html

http://dev.mysql.com/doc/refman/5.6/ja/grant.html

# MySQLセキュリティ概要



# MySQLでの暗号

- SSL/TLS 暗号化
  - MySQLクライアント/サーバ間
  - レプリケーション: マスター/スレーブ間
- データ暗号化
  - AES 暗号/復号

- MySQL Enterprise Encryption
  - 非対称暗号/復号
  - 公開鍵と秘密鍵の生成
  - セッション固有鍵
  - デジタル署名
- MySQL Enterprise Backup
  - AES 暗号/復号

# SSL/TLS

- 暗号化接続
  - MySQLクライアント/サーバ間
  - レプリケーション: マスター/スレーブ間
- MySQLでは接続ごとの暗号化が可能
  - X509 標準準拠のIdentity Verification (アイデンティティの検証)
- 適切なSSL証明書やキーファイルの指定
- 信頼できるCA (Certificate Authorities / 認証局) を利用可能
- CRL (Certificate Revocation List / 証明書失効リスト) サポート

# SSL Trust Levels

| クライアント | サーバ | 暗号化 | クライアントキーペア | | サーバキーペア | |
|---|---|---|---|---|---|---|
| | | | Signed | Valid | Signed | Valid |
| (*--skip-ssl*) | (*--skip-ssl*) | ✗ | ✗ | ✗ | ✗ | ✗ |
| CA | key pair | ✓ | ✗ | ✗ | ✓ | ✗ |
| key pair | *REQUIRE X509* | ✓ | ✓ | ✗ | ✓ | ✗ |
| key pair | *REQUIRE SUBJECT* | ✓ | ✓ | ✓ | ✓ | ✗ |
| *--verify-server-cert* | key pair | | | | | ✓ |

# SSLに関する権限

- 権限付与時にREQUIRE SSLを利用可能
  - **REQUIRE NONE**
  - **REQUIRE SSL**
  - **REQUIRE X509**
  - **REQUIRE ISSUER 'issuer'**
  - **REQUIRE SUBJECT 'subject'**
  - **REQUIRE CIPHER 'cipher'**

http://dev.mysql.com/doc/refman/5.6/ja/ssl-connections.html

# SSLに関する権限

```
GRANT ALL PRIVILEGES ON test.* TO 'root'@'localhost'
  IDENTIFIED BY 'goodsecret'
  REQUIRE SUBJECT '/C=EE/ST=Some-State/L=Tallinn/
    O=MySQL demo client certificate/
    CN=Tonu Samuel/Email=tonu@example.com'
  AND ISSUER '/C=FI/ST=Some-State/L=Helsinki/
    O=MySQL Finland AB/CN=Tonu Samuel/Email=tonu@example.com'
  AND CIPHER 'EDH-RSA-DES-CBC3-SHA';
```

# MySQL Enterprise Encryption

- MySQLの暗号化関数
  - AES256による対称暗号 (全Edition共通)
  - 公開鍵方式 / 非対称鍵暗号方式 – RSA暗号
- 鍵管理機能
  - 公開鍵および秘密鍵の生成
  - 鍵の交換方式: DH (ディフィー・ヘルマン鍵共有)
- 署名および検証Sign and verify data functions
  - デジタル署名の暗号化ハッシュ、検証およびバリデーション – RSA, DSA

# MySQL Enterprise Encryption

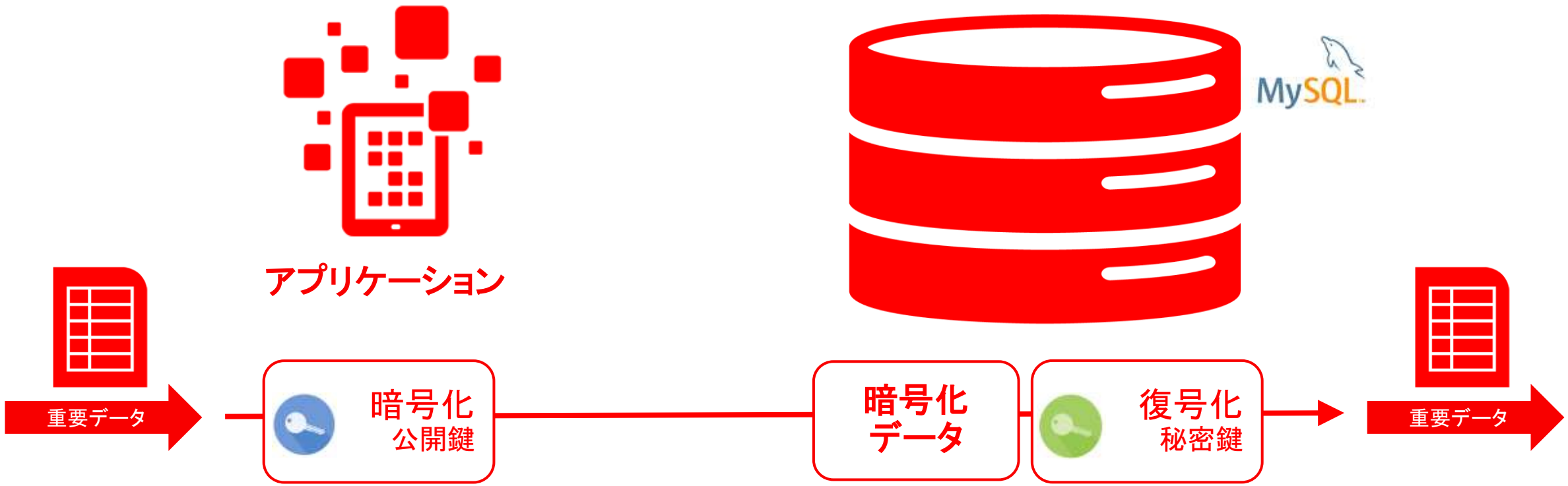
## MySQLサーバ内で暗号化/復号化


**秘密鍵 / 公開鍵のペア**

- MySQL Enterprise Encryptionの関数で生成
- 外部で生成 (例: OpenSSL)

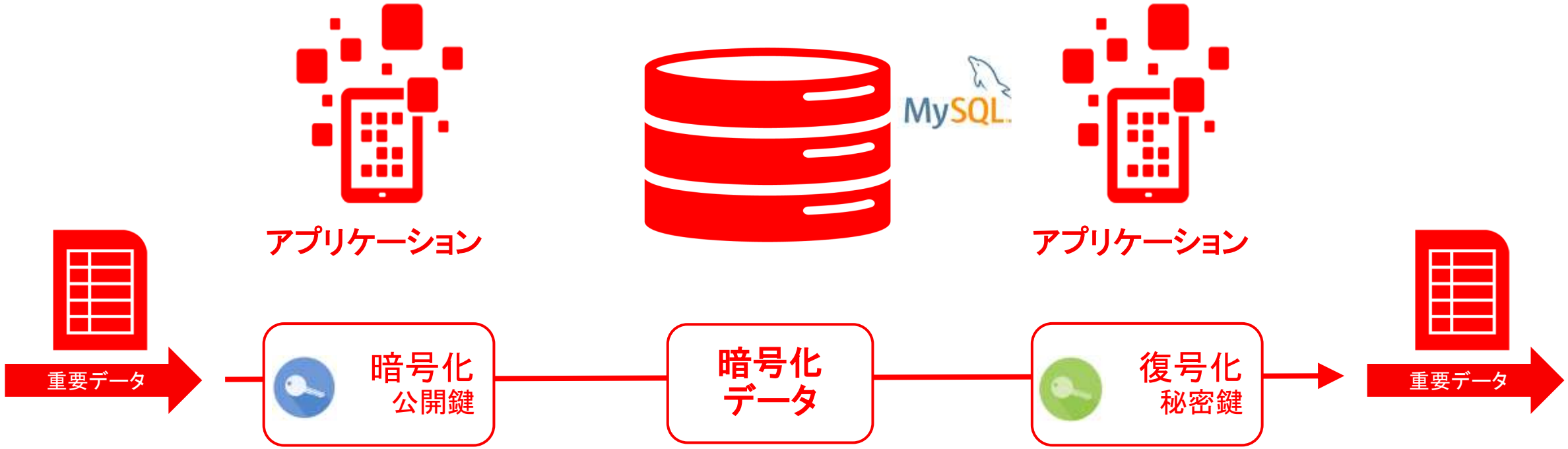
# MySQL Enterprise Encryption

## アプリで暗号化/MySQLで復号化



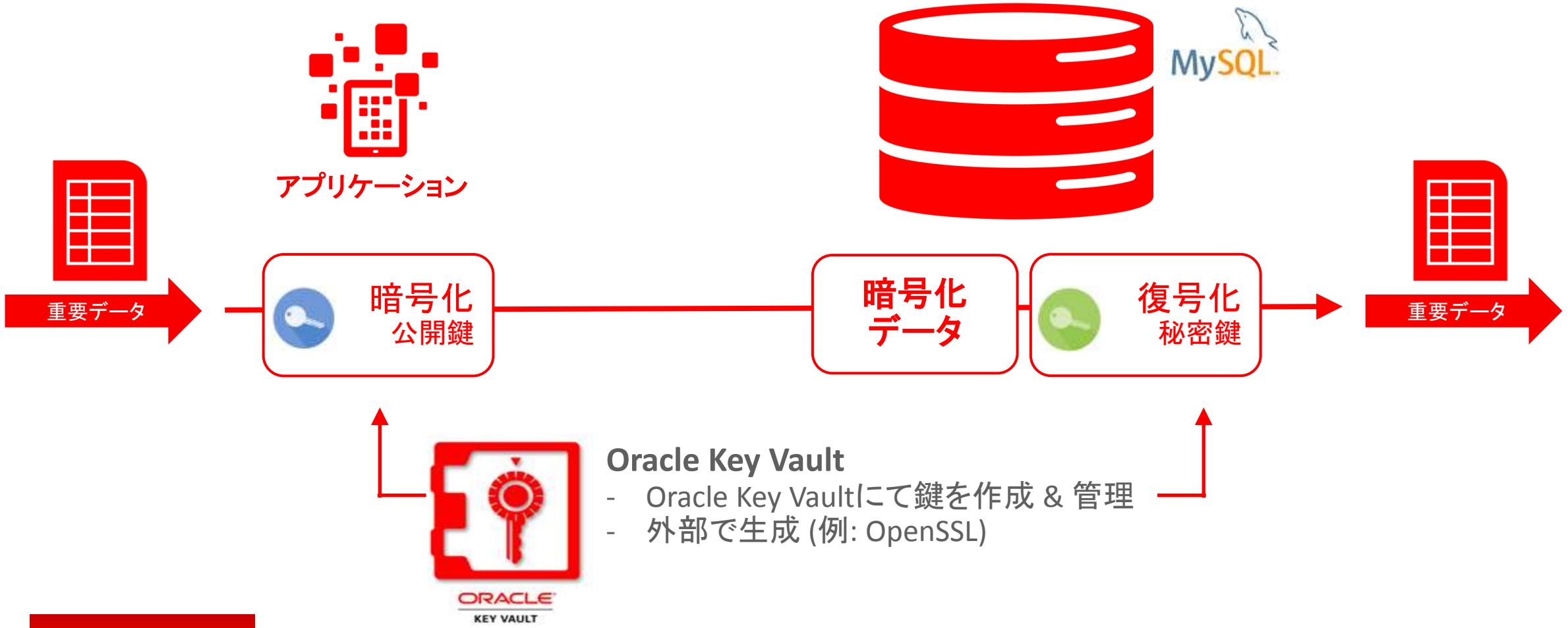
# MySQL Enterprise Encryption

## アプリで暗号化/MySQLで格納/アプリで復号化



# MySQL Enterprise Encryption

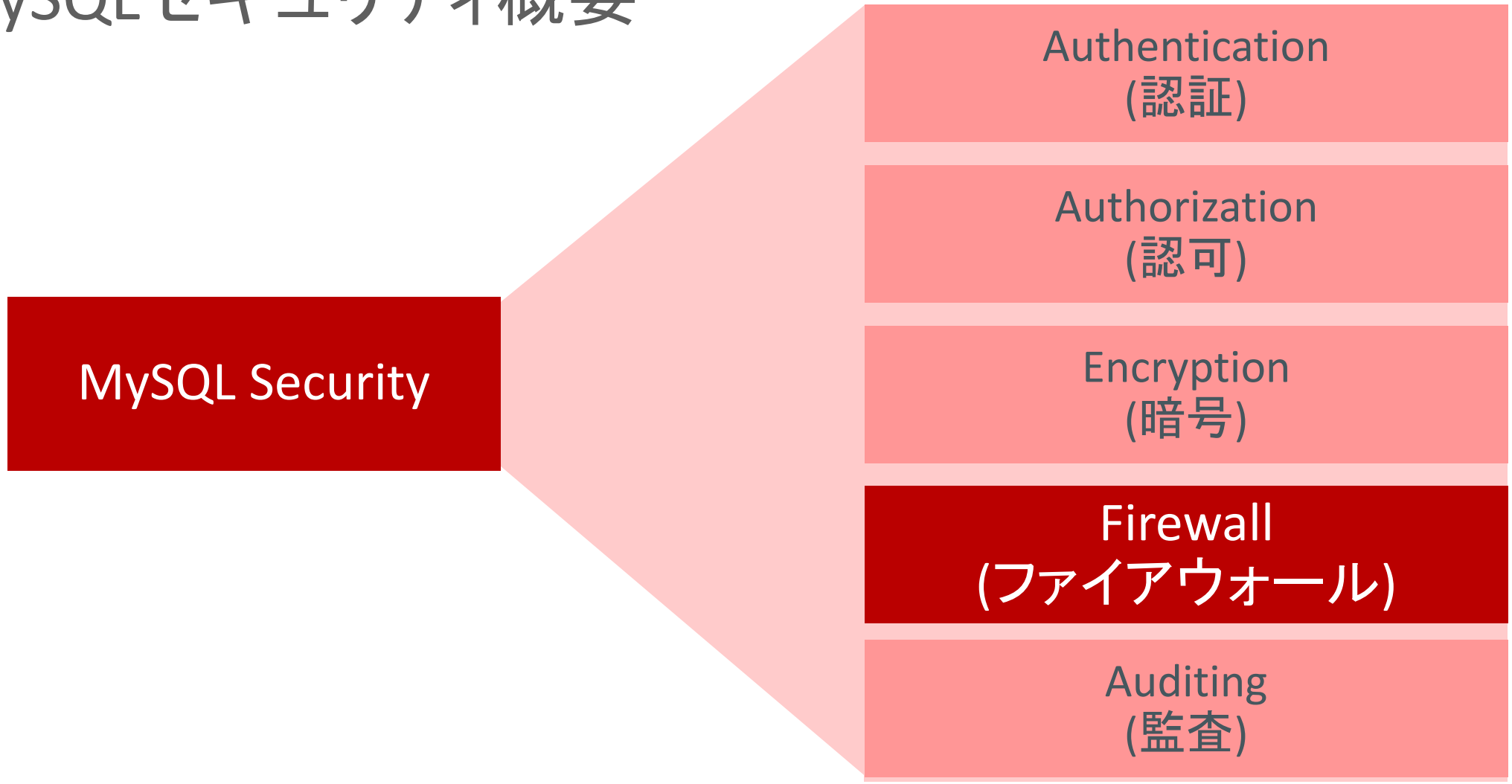
## Oracle Key Vault にて鍵を生成 (または外部で生成)




**Oracle Key Vault**

- Oracle Key Vaultにて鍵を作成 & 管理
- 外部で生成 (例: OpenSSL)

# MySQLセキュリティ概要



# データベース ファイアウォール

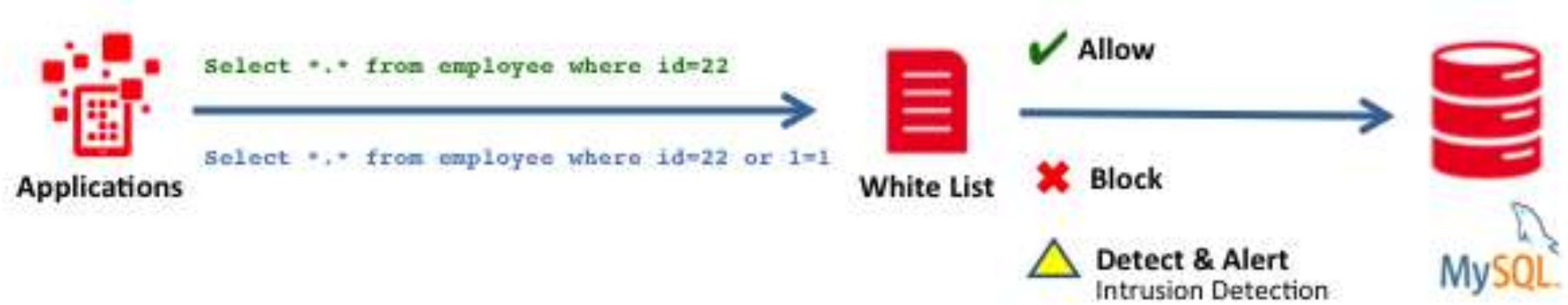
- SQLインジェクション: Webアプリケーション最頻出の脆弱性
  - 77%のWebに脆弱性
  - 8件中1件に深刻な脆弱性
- MySQL Enterprise Firewall
  - SQL文の実行をリアルタイムでモニタリング
  - ホワイトリストに基づく「ルール」を自動的に生成
  - ポリシーに違反するトランザクションを「検知」または「ブロック」

# MySQL Enterprise Firewall

- SQLインジェクション対策、リアルタイムで保護
  - ポジティブ・セキュリティ・モデル、実行されるクエリーを分析してホワイトリストと照合
- アプリケーション処理を学習してホワイトリストを自動作成
  - ユーザー毎に、SQL実行パターンを記録して自動的にホワイトリストを作成
- 不審なアクセスをブロック
  - ポリシーに違反するトランザクションを「検知」または「ブロック」
- 透過的
  - アプリケーションを変更する必要無し

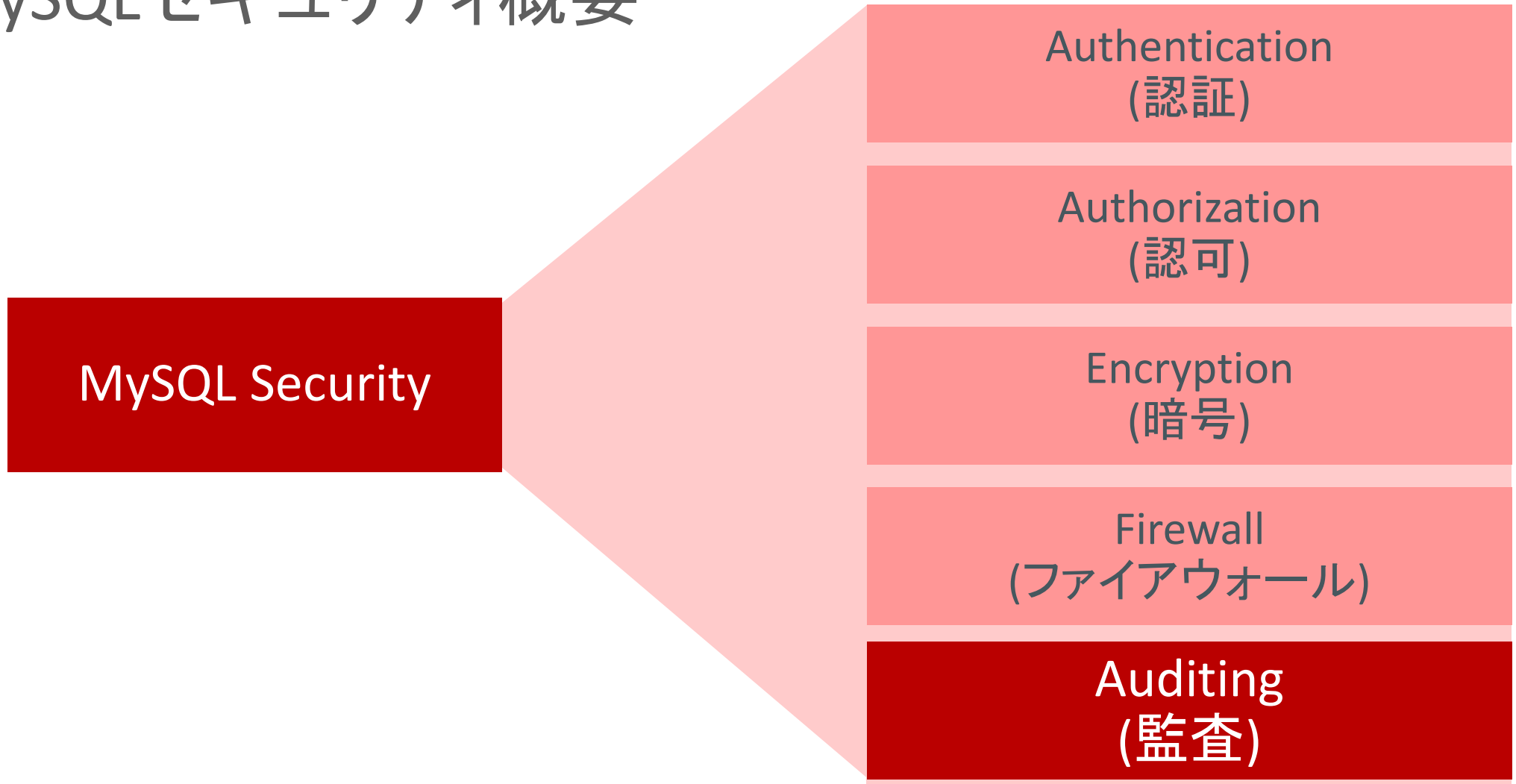
# MySQL Enterprise Firewall

- ホワイトリストを用いたSQLインジェクション対策




- ポリシーに違反するトランザクションを「検知」または「ブロック」
- ログに記録 & 検証

# MySQLセキュリティ概要



# データベース監査

- セキュリティおよびコンプライアンスのための監査
  - FIPS, HIPAA, PCI-DSS, SOX, DISA STIG, …
- MySQLのビルトインのログ:
  - 一般ログ、エラーログ
- MySQL Enterprise Audit
  - ログオン、クエリーの情報を監査可能
  - ポリシーを設定可能:フィルタリング、ログローテーション
  - 動的に設定を変更可能:Audit設定時にサーバの再起動が不要
  - Oracle Audit Vault対応

# MySQL Enterprise Audit

```
mysql> INSTALL PLUGIN audit_log SONAME 'audit_log.so';

mysql> SHOW VARIABLES LIKE 'audit_log%';
+-------------------------------+---------------+
| audit_log_buffer_size         | 1048576       |
| audit_log_connection_policy   | ALL           |
| audit_log_current_session     | OFF           |
| audit_log_exclude_accounts    |               |
| audit_log_file                | audit.log     |
| audit_log_flush               | OFF           |
| audit_log_format              | NEW           |
| audit_log_include_accounts    |               |
| audit_log_policy              | ALL           |
| audit_log_rotate_on_size      | 0             |
| audit_log_statement_policy    | ALL           |
| audit_log_strategy            | ASYNCHRONOUS  |
+-------------------------------+---------------+
```

1. DBAがAuditプラグインを有効化

```
shell> mysql -h joeshost -u joe -p
Enter password: ********

mysql> SELECT * FROM joes_table;
+---------------+--------------+
| FIRST_NAME    | LAST_NAME    |
+---------------+--------------+
| Joe           | User         |
+---------------+--------------+
```

2. ユーザ Joe が接続しSQLを実行


3. Joe の接続と実行したSQLをログに記録

```
<?xml version="1.0" encoding="UTF-8"?>
<AUDIT>
  <AUDIT_RECORD
    TIMESTAMP="2012-08-02T14:52:12"
    NAME="Audit"
    SERVER_ID="1"
    VERSION="1"
    STARTUP_OPTIONS="--port=3306"
    OS_VERSION="i686-Linux"
    MYSQL_VERSION="5.5.28-debug-log"/>
  <AUDIT_RECORD
    TIMESTAMP="2012-08-02T14:52:41"
    NAME="Connect"
    CONNECTION_ID="1"
    STATUS="0"
    USER="joe"
    PRIV_USER="root"
    OS_LOGIN=""
    PROXY_USER=""
    HOST="SERVER1"
    IP="127.0.0.1"
    DB="joes_db"/>
  <AUDIT_RECORD
    TIMESTAMP="2012-08-02T14:53:45"
    NAME="Query"
    CONNECTION_ID="1"
    STATUS="0"
    SQLTEXT="SELECT * FROM joes_table;"/>
</AUDIT>
```

# MySQLのセキュリティ強化策

## インストール

- mysql_secure_installation
- MySQLを常に最新版に
  - MySQL Installer for Windows
  - Yum/Apt レポジトリ

## 設定

- ファイアウォール
- 監査とログ
- ネットワークアクセスの制限
- 変更点の監視

## ユーザ管理

- 余分なアカウントの削除
- 最低限の権限付与
- ユーザと権限の監査

## パスワード

- 強力なパスワードポリシー
- ハッシュ、失効
- パスワード検証プラグイン

## 暗号化

- セキュアな接続のための SSL/TLS
- データ暗号化 (AES, RSA)

## バックアップ

- バックアップの監視
- バックアップの暗号化

# MySQL 5.7 Linux パッケージ – セキュリティ関連の改良

- testデータベースの削除
- 無名ユーザの作成を割愛
- rootユーザはローカルホストのみ
- 暗号化された通信を利用できるように設定
  - 自動的に SSL/RSA 証明書/キーを作成
    - EE : 証明書/キーがなければ起動時に
    - CE : 新ユーティリティ mysql_ssl_rsa_setup で
- SSL証明書/キーを自動検出
- 暗号化されたTLS接続を試行
- データのインポート/エクスポート用のディレクトリをコンパイル時に指定済み
- OSのmysqlユーザとグループを作成し、ログイン抑止など設定
- データのインポート/エクスポートの無効化が可能に
  - secure-file-privの値をNULLに

各種のセキュリティセットアップのステップをMySQL Installer for Windowsに追加

# MySQLのセキュリティ強化策: インストール

- MySQL_Secure_Installation / MySQL Installer for Windows
  - rootアカウントに「強固」なパスワードを設定
  - ローカルホスト以外からアクセス可能なrootアカウントを削除
  - 無名ユーザアカウントを削除
  - testデータベースを削除
    - 無名ユーザを含め全てのユーザからアクセス可能
- MySQLを常に最新版に
  - レポジトリ – YUM/APT/SUSE
  - MySQL Installer for Windows

# mysql_secure_installationを利用した不要アカウント削除

```
[root@Centos03 mysql]# ./bin/mysql_secure_installation


NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MySQL
      SERVERS IN PRODUCTION USE!  PLEASE READ EACH STEP CAREFULLY!

In order to log into MySQL to secure it, we'll need the current
password for the root user.  If you've just installed MySQL, and
you haven't set the root password yet, the password will be blank,
so you should just press enter here.

Enter current password for root (enter for none):
OK, successfully used password, moving on...

Setting the root password ensures that nobody can log into the MySQL
root user without the proper authorisation.

Set root password? [Y/n] Y
New password:
Re-enter new password:
Password updated successfully!
Reloading privilege tables..
 ... Success!


By default, a MySQL installation has an anonymous user, allowing anyone
to log into MySQL without having to have a user account created for
them.  This is intended only for testing, and to make the installation
go a bit smoother.  You should remove them before moving into a
production environment.

Remove anonymous users? [Y/n] Y
 ... Success!
```

rootアカウントパスワード設定
不要なアカウントの削除
rootのリモートログイン不可
testデータベースの削除

```
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
+--------------------+
3 rows in set (0.00 sec)
```

```
mysql> select user,host,password from mysql.user;
+------+-----------+-------------------------------------------+
| user | host      | password                                  |
+------+-----------+-------------------------------------------+
| root | localhost | *A41ECFBE1191DDE4713F2B6F5A6CD5D0D0D5DC35 |
| root | 127.0.0.1 | *A41ECFBE1191DDE4713F2B6F5A6CD5D0D0D5DC35 |
| root | ::1       | *A41ECFBE1191DDE4713F2B6F5A6CD5D0D0D5DC35 |
+------+-----------+-------------------------------------------+
3 rows in set (0.00 sec)
```

参照：4.4.5 mysql_secure_installation

# 補足：mysql_secure_installation

- MySQL 5.6をrpmでインストールした場合、rootのパスワードは自動的に設定され、後で変更する必要がある。
  - パスワードは $HOME/. mysql_secret ファイルに記載されている
  - パスワードを変更するまではrootユーザで何も実行できない
- MySQL 5.6の場合、mysql_install_db に--random-password が指定でき、DB作成時に合わせて以下の操作を実行可能
  - rootユーザにランダムなパスワードを設定
    - パスワードの設定ファイル、パスワードを変更するまでの操作制限は上記と同様
  - anonymousユーザを削除

# ソフトウェア更新 – データベースおよびOSのメンテナンス

- セキュリティを確保するためには、常にOSおよびMySQLに最新パッチを適用することが求められる
  - 再起動を求められることもあり、運用計画の工夫が必要
- MySQLレプリケーションを利用して停止時間を最小限に抑える工夫も
  - 順番に更新を行う
    - ベストプラクティスとしてはスレーブから先に更新
  - MySQL 5.6以上では GTIDベースのレプリケーションをサポート
    - 順番に更新を行うことがよりシンプルに可能
- OSベンダが用意したセキュリティガイドラインも参考に
  - 例: http://www.oracle.com/technetwork/articles/servers-storage-admin/tips-harden-oracle-linux-1695888.html

# MySQLのセキュリティ強化策: 設定

- SQL実行や接続の監査
  - MySQL Enterprise Audit利用
  - または一時的に一般ログを有効
  - 常に監視と検査
- リモートアクセス無効化/制限
  - ローカルのみの場合は
    “skip-networking”
    または bind-address=127.0.0.1
  - リモートアクセスがある場合は接続元ホストまたはIPアドレスを制限
- rootユーザ名を変更
- サーバ内のローカルファイルへのアクセスを無効化
  - LOAD DATA LOCAL INFILEを無効化
- MySQLをデフォルトとは異なるTCP/IPポートで運用
  - MySQLの発見リスクを下げる
- OSのMySQLユーザの権限を削減
- secure-authを有効に
  - 古いパスワードハッシュの接続を拒否
  - 5.7からは同様の挙動がデフォルトに

# MySQLのセキュリティ強化策: ベストプラクティス

| パラメタ名 | 推奨値 | 備考 |
|---|---|---|
| secure_file_priv | データのロード専用ディレクトリ | 特定の場所からのみデータのロードを許可。<br>MySQLにOSの様々な場所にアクセスさせない |
| symbolic_links | Boolean – NO | ファイルシステムのセキュアではないディレクトリへのリダイレクトを防ぐ |
| general-log | Boolean – OFF | デバッグ時にのみONに |
| log-raw | デフォルト値 - OFF | デバッグ時にのみONに |
| skip-networking<br>または bind-address | ON<br>127.0.0.1 | ローカルアクセスのみの場合は外部からの接続をブロック |
| SSL オプション | 値を有効に | ネットワークの暗号化推奨 |

# ファイルシステム (Secure_file_privの設定)

- LOAD_FILE() 関数および LOAD DATA および SELECT ... INTO OUTFILE文がアクセス可能なディレクトリを、設定したディレクトリに限定

1. 専用のディレクトリを作成
2. mysqlをディレクトリのオーナーに設定
3. my.cnfの[mysqld]セクションに“secure_file_priv "を追加
4. mysqlを再起動

```
[root@GA01 mysql_share]# cat /etc/my.cnf | grep secure_file
secure_file_priv                = /home/mysql/mysql_share
[root@GA01 mysql_share]# /etc/init.d/mysql.server restart
Shutting down MySQL.. SUCCESS!
Starting MySQL. SUCCESS!
[root@GA01 mysql_share]#
```

```
admin@192.168.56.201 [(none)]> SELECT @@global.secure_file_priv;
+---------------------------+
| @@global.secure_file_priv |
+---------------------------+
| /home/mysql/mysql_share/  |
+---------------------------+
1 row in set (0.00 sec)

admin@192.168.56.201 [(none)]> SELECT LOAD_FILE('/etc/passwd')\G
*************************** 1. row ***************************
LOAD_FILE('/etc/passwd'): NULL
1 row in set (0.01 sec)
```

Since: MySQL 5.0.38～

# Account Management (File_privの設定)

## インストール後の作業：OS上のファイルへのアクセスを制限

1. 専用のディレクトリを作成
2. mysqlをディレクトリのオーナーに設定
3. my.cnfの[mysqld]セクションに“secure_file_priv "を追加
4. mysqlを再起動

```
admin@192.168.56.201 [mysql]> SELECT @@global.secure_file_priv;
+---------------------------+
| @@global.secure_file_priv |
+---------------------------+
| NULL                      |
+---------------------------+
1 row in set (0.00 sec)
```

[設定前]

```
admin@192.168.56.201 [mysql]> SELECT LOAD_FILE('/etc/passwd')\G
*************************** 1. row ***************************
LOAD_FILE('/etc/passwd'): root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
```

```
[root@GA01 mysql_share]# cat /etc/my.cnf | grep secure_file
secure_file_priv                = /home/mysql/mysql_share
[root@GA01 mysql_share]# /etc/init.d/mysql.server restart
Shutting down MySQL.. SUCCESS!
Starting MySQL. SUCCESS!
[root@GA01 mysql_share]#
```

[設定後]

```
admin@192.168.56.201 [(none)]> SELECT @@global.secure_file_priv;
+---------------------------+
| @@global.secure_file_priv |
+---------------------------+
| /home/mysql/mysql_share/  |
+---------------------------+
1 row in set (0.00 sec)

admin@192.168.56.201 [(none)]> SELECT LOAD_FILE('/etc/passwd')\G
*************************** 1. row ***************************
LOAD_FILE('/etc/passwd'): NULL
1 row in set (0.01 sec)
```

# MySQLのセキュリティ強化策: パスワードポリシー

- 強力なパスワードポリシーの強制
- パスワードハッシュ
- パスワードの失効
- パスワード検証プラグイン
- 認証プラグイン
  - 各コンポーネントからのパスワードポリシーを引き継ぎ
  - LDAP, Windows Active Directory, など
- 使用していないアカウントを無効に
  - アカウントのロック (MySQL 5.7)

Since: MySQL 5.6.6～

# Account Management (Password Validation Plugin)

## インストール後の作業：パスワードポリシーのインストール

英数字の混在を強制する、文字数をN文字以上にする、特定のキーワードはパスワードに指定できなくする、といった対応が可能

```
[mysql]> install plugin validate_password soname 'validate_password.so';
```

```
root@localhost [mysql]>SHOW VARIABLES LIKE 'validate_password%';
+--------------------------------------+---------------------------------------+
| Variable_name                        | Value                                 |
+--------------------------------------+---------------------------------------+
| validate_password_dictionary_file    | /usr/local/mysql/data/band_dictionary.txt |
| validate_password_length             | 8                                     |
| validate_password_mixed_case_count   | 1                                     |
| validate_password_number_count       | 1                                     |
| validate_password_policy             | STRONG                                |
| validate_password_special_char_count | 1                                     |
+--------------------------------------+---------------------------------------+
6 rows in set (0.00 sec)
```

例) 8文字以下、小文字のみ、数字未入力、辞書に登録済みの文字はパスワードとしては不適切な為、ERROR 1819で拒否される。

```
[admin@CentOS01 ~]$ perror 1819;
MySQL error code 1819 (ER_NOT_VALID_PASSWORD): Your password does not satisfy the current policy requirements
[admin@CentOS01 ~]$
```

# MySQLのセキュリティ強化策: 暗号化

- 暗号化された通信など
- SSL/TLSによる通信の暗号化
- X.509による「要素」の追加 – ユーザ/パスワード、その他の認証
  - クライアントの認証を確実に – さらなる信頼性
- データベースおよびアプリケーションレベルで重要データを暗号化
- データベースまたはアプリケーションの関数でデータをマスク/匿名化
  - 個人ID, クレジットカード, …
- 公開鍵はアプリケーションでの暗号時のみ利用することを検討

# MySQLのセキュリティ強化策: バックアップ

- バックアップはビジネス面からも重要
  - 攻撃や障害後のリストアに利用
  - 他のサーバへの移行やマイグレーション、複製
  - 監査の試行に利用
- 定期的にバックアップ
- バックアップの監視
- バックアップデータの暗号化

# アプリケーションおよび認証情報 – ベストプラクティス

- アプリケーション – ユーザ名/パスワードなど認証情報の共有は最小限
  - より高い粒度で – アプリケーションやサーバ全体で共有しない
- 認証情報の変更機能を持たせる
  - 全てのパスワードを同時に変更する必要はない
- 認証情報の変更自体もセキュアかつ分かりやすく
  - アプリケーションコード内への認証情報埋め込みは避ける
    - アプリケーションの再デプロイをせずに変更できるように
    - バージョン管理システム内にアプリケーションと同様に格納されないように
    - アプリケーションから認証情報へのアクセスもセキュアな方法で

# MySQL Enterprise Edition

# MySQL Enterprise Edition

## ビジネスクリティカルなシステムの運用を効率化

- MySQL導入の最適化
- ROIの最適化をサポート
- ユーザビリティ・顧客満足の向上

# MySQL Enterprise Edition

## 拡張機能

- 拡張性
- 高可用性
- セキュリティ
- 監査
- 暗号化

## 管理ツール

- 監視
- バックアップ
- 開発
- 管理
- マイグレーション

## サポート

- 技術サポート
- コンサルティングサポート
- オラクル製品との動作保証

# MySQL Enterprise Editionによるデータ保護

## MySQL Enterprise Backup

- オンラインバックアップ/リカバリ
- クラウドストレージへバックアップ
- 差分バックパック & ポイントインタイムリカバリ

## MySQL Enterprise Auth.

- 外部認証との統合 (PAM, Windows, LDAP, etc.)
- MySQL Enterprise Monitorでのセキュリティアドバイザ

## MySQL Enterprise Encryption

- AES256による対称暗号
- 公開鍵方式 / 非対称暗号
- 暗号学的ハッシュによる電子署名、照合および妥当性確認

## MySQL Enterprise Audit

- 接続、ログインおよびSQL実行の記録
- ポリシーベースのフィルタリングおよびログ切り替え
- オラクルの監査仕様に準拠したXMLベースの出力

| | MySQL Editions | | |
|---|---|---|---|
| | Standard Edition | Enterprise Edition | Cluster CGE |
| **機能概要** | | | |
| MySQL Database | ✔ | ✔ | ✔ |
| MySQL Connectors | ✔ | ✔ | ✔ |
| MySQL Replication | ✔ | ✔ | ✔ |
| MySQL Fabric, Utilities | | ✔ | ✔ |
| MySQL Partitioning | | ✔ | ✔ |
| Storage Engine: MyISAM, InnoDB | ✔ | ✔ | ✔ |
| Storage Engine: NDB (ndbcluster) | | | ✔ |
| MySQL Workbench SE/EE* | ✔ | ✔ | ✔ |
| MySQL Enterprise Monitor* | | ✔ | ✔ |
| MySQL Enterprise Backup* | | ✔ | ✔ |
| MySQL Enterprise Authentication (外部認証サポート) * | | ✔ | ✔ |
| MySQL Enterprise Audit (ポリシーベース監査機能) * | | ✔ | ✔ |
| MySQL Enterprise Encryption（非対称暗号化）* | | ✔ | ✔ |
| MySQL Enterprise Firewall（SQLインジェクション対策）* | | ✔ | ✔ |
| MySQL Enterprise Scalability (スレッドプール) * | | ✔ | ✔ |
| MySQL Enterprise High Availability (HAサポート) * | | ✔ | ✔ |
| Oracle Enterprise Manager for MySQL* | | ✔ | ✔ |
| MySQL Cluster Manager (MySQL Cluster管理) * | | | ✔ |
| MySQL Cluster Geo-Replication | | | ✔ |

*商用版のみで利用可能な追加機能

| | MySQL Editions | | |
|---|---|---|---|
| | Standard SE | Enterprise EE | Cluster CGE |
| **Oracle Premium Support** | | | |
| 24時間365日サポート | ✔ | ✔ | ✔ |
| インシデント数無制限 | ✔ | ✔ | ✔ |
| ナレッジベース | ✔ | ✔ | ✔ |
| バグ修正&パッチ提供 | ✔ | ✔ | ✔ |
| コンサルティングサポート | ✔ | ✔ | ✔ |
| **オラクル製品との動作保証** | | | |
| Oracle Linux | ✔ | ✔ | ✔ |
| Oracle VM | ✔ | ✔ | ✔ |
| Oracle Solaris | ✔ | ✔ | ✔ |
| Oracle Enterprise Manager | | ✔ | ✔ |
| Oracle GoldenGate | | ✔ | ✔ |
| Oracle Data Integrator | | ✔ | ✔ |
| Oracle Fusion Middleware | | ✔ | ✔ |
| Oracle Secure Backup | | ✔ | ✔ |
| Oracle Audit Vault and Database Firewall | | ✔ | ✔ |

※最新の対比表は、MySQL Editionsのサイトを参照下さい。

# MySQL Supportの特徴

- 「パフォーマンス・チューニング」や「SQLチューニング」まで通常サポートの範囲内
  - コンサルティングサポートが含まれており、「クエリ・レビュー」、「パフォーマンス・チューニング」、「レプリケーション・レビュー」、「パーティショニング・レビュー」などに対応可能
    http://www-jp.mysql.com/support/consultative.html
- ソースコードレベルでサポート可能
  - ほとんどのサポートエンジニアがソースを読めるため、対応が早い
  - 開発エンジニアとサポートエンジニアも密に連携している
- 物理サーバー単位課金
  - CPU数、コア数に依存しない価格体系
- オラクルのライフタイムサポート
  - サポートポリシーが明確であるため、長期的な計画を立てやすい
    http://www-jp.mysql.com/support/

# 参考情報

# 参考情報

- MySQL Webサイト
  https://www-jp.mysql.com/
- MySQLコミュニティWebページ
  http://dev.mysql.com/
- 日本MySQLユーザー会(メーリングリスト有り)
  http://www.mysql.gr.jp/
- イベント案内
  - mysql.comのイベントページ
    https://www-jp.mysql.com/news-and-events/events/
  - オラクル社全体のイベントページ(OTN Japan - イベント・セミナー)
    http://events.oracle.com/search/search

# MySQLのドキュメント

- MySQL Developer Zone(http://dev.mysql.com/)にドキュメント類が公開されている
- 以下のドキュメントは2015年6月に日本語版が公開された
  - MySQL 5.6 リファレンスマニュアル (含むMySQL Cluster 7.3-7.4マニュアル)
    http://dev.mysql.com/doc/refman/5.6/ja/index.html
  - MySQL Enterprise Monitor 3.0.18 マニュアル
    http://dev.mysql.com/doc/mysql-monitor/3.0/ja/index.html
  - MySQL Enterprise Backup ユーザーズガイド (バージョン 3.11.1)
    http://dev.mysql.com/doc/mysql-enterprise-backup/3.11/ja/index.html
- 上記日本語版公開以降に英語版ドキュメントのみ修正されている内容もあるため、ドキュメント参照時は英語版ドキュメントも合わせてご参照下さい。(URLの”ja”部分を”en”に変更すると、英語版ドキュメントが表示可能)

# MySQLのドキュメント

- MySQL Documentation: MySQL Reference Manuals
  http://dev.mysql.com/doc/
- MySQL Documentation: MySQL Workbench
  http://dev.mysql.com/doc/index-gui.html
- MySQL Documentation: MySQL Utilities/MySQL Fabric
  http://dev.mysql.com/doc/index-utils-fabric.html
- MySQL Documentation: Connectors and APIs
  http://dev.mysql.com/doc/index-connectors.html

# MySQLのドキュメント

- MySQL Documentation: Other MySQL Documentation
  http://dev.mysql.com/doc/index-other.html
  ⇒”world database”などのサンプルデータベースもダウンロード可能
- MySQL Documentation: MySQL Enterprise Products
  http://dev.mysql.com/doc/index-enterprise.html
  ⇒商用版製品に関するドキュメント

# MySQL Enterprise Edition & Cluster CGEの試使用
## 30日間トライアル


- Oracle Software Delivery Cloud
  http://edelivery.oracle.com/
- 製品パックを選択:
  “MySQL Database”
- 製品マニュアル
  http://dev.mysql.com/doc/index-enterprise.html

# オラクルユニバーシティ MySQL 研修

| コース名 | 日数 | 価格 (税込) | 開催日程 |
|---|---|---|---|
| MySQL for Beginners | 4 | ¥220,320 | お問い合わせください |
| MySQL データベース管理 I | 3 | ¥165,240 | 2015/09/07 - 09, 2015/10/26 - 28, 2015/12/02 – 04 |
| MySQL データベース管理 II | 2 | ¥110,160 | 2015/09/24 - 25, 2015/10/05 – 06, 2015/09/29 – 30, 2015/12/07 – 08 |
| MySQL High Availability | 3 | ¥231,336 | お問い合わせください |

※ MySQL データーベース管理 I/II, MySQL Performance TuningはMySQL5.5対応、MySQL 入門は MySQL 5.0/5.1対応です。
※ コース開催予定は2015年7月現在のものです。開催日程の最新情報はOracle University ホームページ ( http://www.oracle.com/jp/education/ ) にてご確認ください。
※ 価格(税込み)は**2015年7月現在の価格です**。Oracle PartnerNetwork 会員様は、パートナー割引価格で受講いただけます。

ORACLE

# 管理者向け MySQL 5.6 対応認定資格

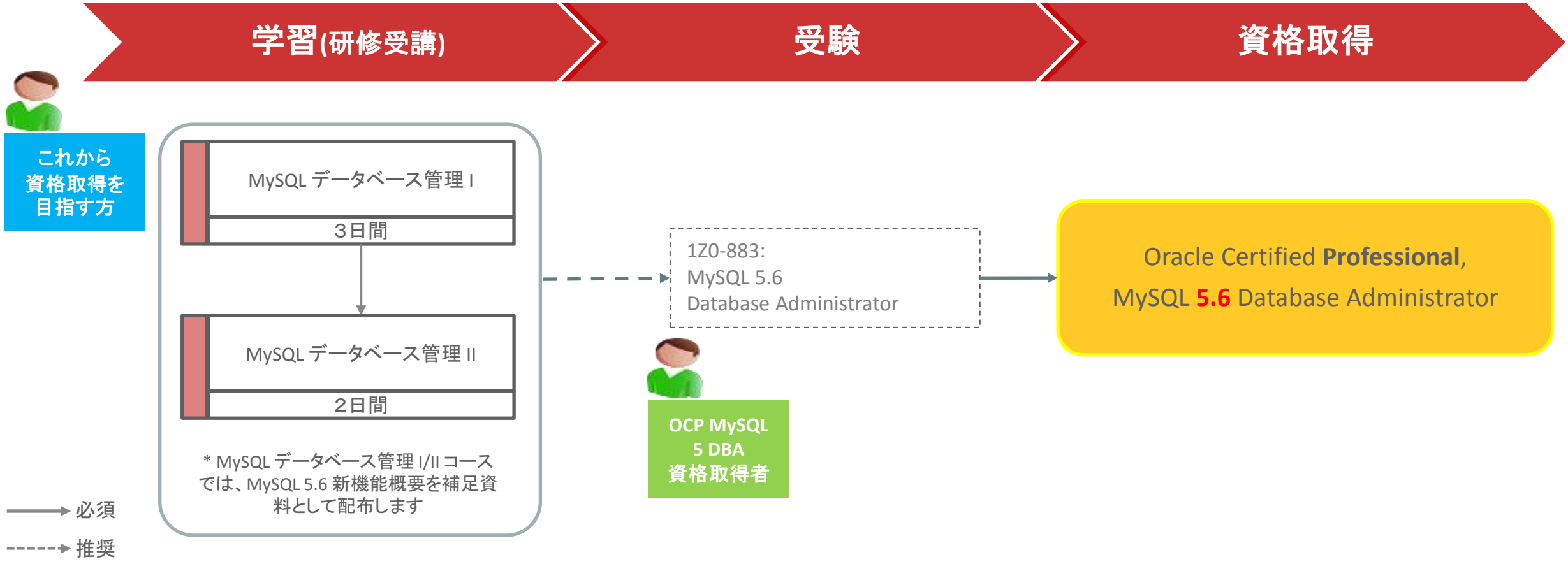
- Oracle Certified **Professional**, MySQL **5.6** Database Administrator
  - Oracle Certified Professional, MySQL 5.6 Database Administrator 資格は、パーティショニング、およびレプリケーションにおける機能強化やパフォーマンス監視と診断のPERFORMANCE_SCHEMAの使用などMySQL 5.6の新機能を含むMySQLデータベースのインストール、複製、チューニング、およびセキュリティ設定など幅広い管理スキルを証明します。
- 認定試験:
  - MySQL 5.6 Database Administrator (1Z0-883)
    - 本試験に合格することで、資格取得できます
    - 日本語試験、英語試験共に受験可能

# 開発者向け MySQL 5.6 対応認定資格

- Oracle Certified **Professional**, MySQL **5.6** Developer
  - Oracle Certified Professional, MySQL 5.6 Database Administrator 資格は、MySQL データ・タイプや SQLシンタックス、テーブルやスキーマなどの各種オブジェクト、ストアド・プロシージャ、ビュー、結合など、MySQL データベースを使用したアプリケーション開発に必要なスキルを証明します。
- 認定試験:
  - MySQL 5.6 Developer (1Z0-882)
    - 本試験に合格することで、資格取得できます
    - 日本語試験、英語試験共に受験可能

# 管理者向け MySQL 5.6 認定資格取得パス

## 新規取得もアップグレードも一試験で。



# Integrated Cloud

## Applications & Platform Services


ORACLE®