ZDRIVE

システム開発実習

DB利用のログイン認証の実装

株式会社ジードライブ

ZDRIVE

この講義で学ぶこと

- データベースを利用したログイン認証機構を実装する際のポイント

パスワード情報の保護

ZDRIVE

ログイン情報を格納するテーブル

- ログイン情報を格納するDBのテーブルには、以下のフィールドが最低限含まれる：
  - ログインID
  - パスワード

例：loginテーブル

login_id	password
taro	catchABall123
newton	fallingRedApple789
kaguya	princessBamb00!

ZDRIVE

パスワード情報の保護

- パスワードをそのままの文字列(平文：ひらぶん)でデータベースに保存するのは危険
  - 万が一データベースの中身が第三者に漏えいした場合に、ユーザのパスワードが知られてしまう
  - 他のサイトでも同じパスワードを使い回しているユーザが多い現状では、１つのパスワードが漏えいすると、そのユーザが利用する他のサイトでも「なりすまし」の被害に遭う可能性がある

パスワード情報の保護（つづき）

- そこで、パスワードは何らかの暗号化を行って保存するのが一般的
- よく利用されるのは、ハッシュ関数と呼ばれる、ある文字列から復元不可能な文字列を生成するしくみ
  - この方法を使ってパスワードを保存しておくと、サイト管理者でもユーザのパスワードを知ることはできない

login_id	password
taro	cc3336549cf5473a01b59fa301b4ce298fcdb962988c32d59a1c3eefa7d5366c ↑「catchABall123」をハッシュ化したもの

- ハッシュ関数の例
  - MD5, SHA-1, **SHA-2**, etc. (参考：ハッシュ関数まとめ)

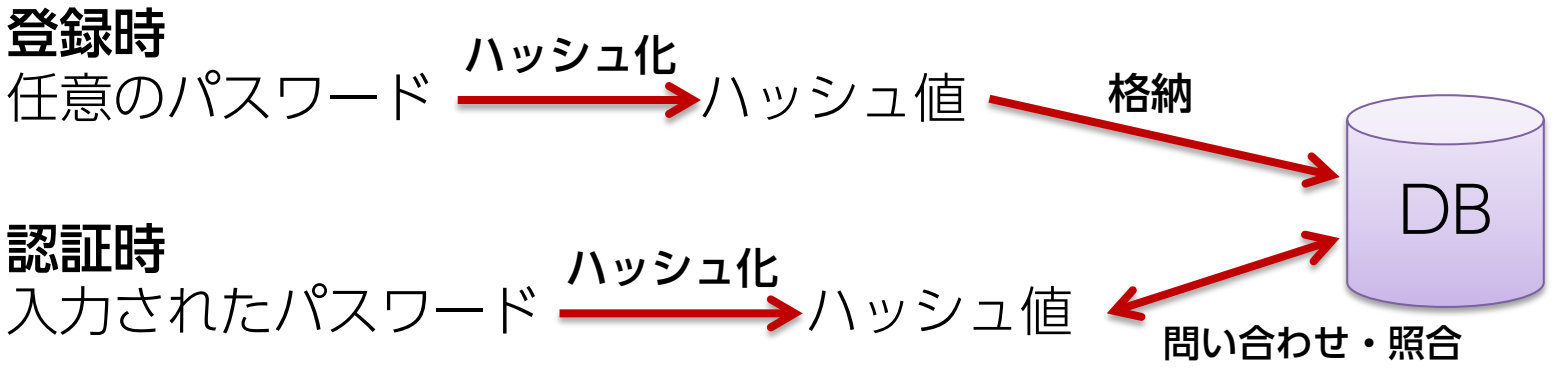
ハッシュ関数の例

- SHA-256（SHA-2規格の一つ）
  - 任意の文字列から生成される256bit（16進数で64桁）の**ハッシュ値**
    例：「hello」という文字列のSHA-256値
    - 2cf24dba5fb0a30e26e83b2ac5b9e29e1b161e5c1fa7425e73043362938b9824

- PHP, MySQL共に、SHA-2を計算する関数がある
  - PHP: **hash("sha256", "文字列")**
  - MySQL: **SHA2("文字列", 256)**

ハッシュ利用時のパスワード照合方法

- パスワード登録時にパスワードのハッシュ値を計算しDBに登録する
- 認証時にはユーザの入力したパスワードのハッシュ値を計算しDBに問い合わせる



さらに安全なパスワード保護

- 単純にハッシュ化するのではなく、**Salt**を加えることで、パスワードをより安全に保護することができる
  - パスワードにSaltと呼ばれる任意の文字列を連結してハッシュ化を行う

例：パスワード「catchABall123」にSalt「riceball」を連結
⇒「catchABall123riceball」をハッシュ化しDBに登録する

login_id	password
taro	98e91552310182f024248c7d3691348ca9e1f8befe887b422010042c7b1c3b5b ↑「catchABall123riceball」をハッシュ化したもの

参考：PHPでのhash化にSaltを併用したパスワードの保護について
参考：安全なパスワードの保存方法

DBを利用した認証処理

DBを利用したログイン認証

- SELECT文を使い、ユーザ名とパスワードの組み合わせがテーブルに存在するかどうかをチェックする

Salt + SHA-256を使った認証の例

```
$salt = "riceball";
$id = $_POST["id"];
$pass = $_POST["pass"];
$sql = "SELECT * FROM login WHERE id=? AND pass=?";
$stmt = $pdo->prepare($sql);
$stmt->execute([$id, hash("sha256", $pass . $salt)]);
$info = $stmt->fetch();
if ($info != FALSE) {
  // ログイン認証成功
}
```

セッションIDの再発行

- セキュリティーを高めるために、ログイン認証成功後、セッションIDの再発行を行う
  - session_regenerate_id() という関数を使う

```
 ・ ・ ・
$info = $stmt->fetch();
if ($info != FALSE) {
 // ログイン認証成功
 // セッションIDの再発行
 session_regenerate_id();
}
```

練習

- ログイン用テーブルの作成
- ユーザ情報の登録
- ログイン認証の実験

ZDRIVE

実習課題

- 実習課題04-1 を行う
- 実習課題04-2 を行う