1. SQL Injection

Insecure Code Example

```
$unsafe_id = $_GET['id'];
$query = "SELECT  FROM users WHERE id = $unsafe_id";
$result = mysqli_query($conn, $query);
```

Secure Code Example (Using PDO):

```
$pdo = new PDO("mysql:host=localhost;dbname=mydatabase", "username",
"password");
$stmt = $pdo->prepare("SELECT  FROM users WHERE id = :id");
$stmt->execute(['id' => $_GET['id']]);
$user = $stmt->fetch();
```

2. Cross-Site Scripting (XSS)

Insecure Code Example:

```
echo "<p>Welcome, ". $_POST['username']. "!</p>";
```

Secure Code Example :

```
$safe_username = htmlspecialchars($_POST['username'], ENT_QUOTES, 'UTF-8');
echo "<p>Welcome, ". $safe_username . "!</p>";
```

3. Password Hashing

Insecure Practice: Storing passwords in plain text.

Secure Code Example for Hashing a Password:

```
$password = $_POST['password'];
$hashed_password = password_hash($password, PASSWORD_DEFAULT);
```

Secure Code Example for Verifying a Password:

```
$entered_password = $_POST['entered_password'];
if (password_verify($entered_password, $hashed_password_from_db)) {
    // Password is correct
} else {
    // Invalid password
}
```

4. CSRF (Cross-Site Request Forgery)

Secure Code Example for Generating a CSRF Token:

```
session_start();
if (empty($_SESSION['csrf_token'])) {
    $_SESSION['csrf_token'] = bin2hex(random_bytes(32));
}
$token = $_SESSION['csrf_token'];
```

Secure Code Example for Including the Token in a Form:

```
<input type="hidden" name="csrf_token" value="<?php echo htmlspecialchars($token);
?>">
```

Secure Code Example for Verifying the Token on Form Submission:

```
if (!isset($_POST['csrf_token']) || !hash_equals($_SESSION['csrf_token'],
$_POST['csrf_token'])) {
    die("CSRF validation failed");
}
```

5. Session Management

Secure Session Configuration Code:

```
ini_set('session.cookie_secure', 1);
ini_set('session.cookie_httponly', 1);
ini_set('session.use_strict_mode', 1);
session_start();
session_regenerate_id(true);
```