Validación y desinfección de entradas

¿Qué es la validación y desinfección de entrada?

La validación comprueba si la entrada cumple un conjunto de criterios (por ejemplo, una cadena no contiene comillas simples independientes).

La desinfección modifica la entrada para garantizar que sea válida (como duplicar comillas simples).

Normalmente, combinaría estas dos técnicas para proporcionar una defensa en profundidad a su aplicación. Por ejemplo, puede cambiar todas las comillas simples en una cadena por comillas dobles (desinfectar) y luego verificar que todas las comillas se hayan cambiado a comillas dobles (validar).

Las comprobaciones de validación incluyen pruebas de longitud, formato, rango y caracteres permitidos. Por ejemplo, si su aplicación espera una entrada de número entero positivo, debe validar que cualquier entrada de cadena consta solo de los dígitos del 0 al 9.

Oracle Database proporciona un paquete PL / SQL llamado DBMS_ASSERT, que contiene funciones que pueden usarse para filtrar y desinfectar cadenas de entrada.

Para evitar la inyección de SQL, todas las entradas que se van a concatenar en SQL dinámico deben filtrarse y desinfectarse correctamente.

Para MySQL Protection:

```
      txtUserId = getRequestString("UserId");
txtSQL = "SELECT * FROM Users WHERE UserId = @0";
db.Execute(txtSQL,txtUserId);
```

```
      txtNam = getRequestString("CustomerName");
txtAdd = getRequestString("Address");  txtCit =
getRequestString("City");
      txtSQL = "INSERT INTO Customers (CustomerName,Address,City)
Values(@0,@1,@2)";
      db.Execute(txtSQL,txtNam,txtAdd,txtCit);
```

```
      txtNam = getRequestString("CustomerName");
txtAdd = getRequestString("Address");  txtCit =
getRequestString("City");
      txtSQL = "INSERT INTO Customers (CustomerName,Address,City)
Values(@0,@1,@2)";
      command = new SqlCommand(txtSQL);
command.Parameters.AddWithValue("@0",txtNam);
command.Parameters.AddWithValue("@1",txtAdd);
command.Parameters.AddWithValue("@2",txtCit);
command.ExecuteNonQuery();
```

```
      $stmt = $dbh->prepare("INSERT INTO Customers
(CustomerName,Address,City)
```

```
VALUES (:nam, :add, :cit)");
$stmt->bindParam(':nam', $txtNam);
$stmt->bindParam(':add', $txtAdd);
$stmt->bindParam(':cit', $txtCit);
$stmt->execute();
```