- Una buona soluzione è quella di utilizzare una libreria ORM, come EntityFramework, Hibernate o iBatis.
- Limitare l'accesso agli oggetti e alle funzionalità di database, in base al principio del minimo privilegio.

Esempio - Javascript per Client Second Order Sql Injection.
Forma non corretta:

```
var userId = 5;
var query = connection.query('SELECT * FROM users WHERE id = ?', [userId],
function(err, results) {
 //query.sql returns SELECT * FROM users WHERE id = '5'
});
```

Forma corretta:

```
var post = {id: 1, title: 'Hello MySQL'};
var query = connection.query('INSERT INTO posts SET ?', post, function(err,
result) {
 //query.sql returns INSERT INTO posts SET `id` = 1, `title` = 'Hello MySQL'
});
```

7.11.7 Client Sql Injection

Come riconoscerla

Utilizzando questa vulnerabilità un attaccante potrebbe utilizzare i canali di comunicazione tra l'applicazione e il suo database, ossia modificando ad arte una query SQL testuale. Ciò è reso possibile nei casi in cui l'applicazione costruisce dinamicamente le query concatenandole con l'input dell'utente. Se non a questo non sono stati applicati i controlli di validità, l'attaccante potrebbe modificare i comandi SQL nel senso da lui desiderato.

Come difendersi

Valgono le considerazioni e le contromisure esposte nel punto precedente.

Esempio - Javascript per Client SQL Injection
Forma non corretta:

```
var info = {
 userid: message.author.id
}

connection.query("SELECT * FROM table WHERE userid = '" + message.author.id + "'",
info, function(error) {
 if (error) throw error;
});
```

Forma corretta:

```
var sql = "SELECT * FROM table WHERE userid = ?";
var inserts = [message.author.id];
sql = mysql.format(sql, inserts);
```

Per ulteriori informazioni vedere: http://cwe.mitre.org/data/definitions/89.html,
CWE-89: Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection').

7.11.8 Cross-Site Request Forgery (CSRF)

Come riconoscerla

Il Cross-Site Request Forgery , abbreviato CSRF o anche XSRF, è una vulnerabilità a cui sono esposti i siti web dinamici quando sono progettati per ricevere richieste da un client senza meccanismi per controllare