```
        echo "Funzione non attendibile!";
```

Per ulteriori informazioni si veda: http://cwe.mitre.org/data/definitions/470.html,
Use of Externally-Controlled Input to Select Classes or Code ('Unsafe Reflection').

7.9.9 SQL Injection

Come riconoscerla

I dati forniti dall'utente, in un modulo (form) o attraverso i parametri URL, devono essere sempre considerati non attendibili e potenzialmente corrotti. La composizione dinamica di query SQL, a partire da dati non verificati, consente agli aggressori di inserire valori appositamente predisposti per modificarne il contenuto e il significato. Gli attacchi di SQL injection possono leggere, modificare o eliminare informazioni riservate dal database e talvolta persino metterlo fuori uso o eseguire comandi arbitrari di sistema operativo.

Come difendersi

In genere, la soluzione consiste nel fare affidamento su query parametriche, piuttosto che sulla concatenazione di stringhe. Questa tecnica fornisce un valido escaping dei caratteri pericolosi.
Validare tutti gli input, indipendentemente dalla loro provenienza. La validazione dovrebbe essere basata su una white list: dovrebbero essere accettati solo i dati che adattano a una struttura specificata, scartando quelli che non rispettano la white list. Occorre controllare il tipo del dato, la sua dimensione, l'intervallo di validità (range), il formato ed eventuali valori attesi (white list).
Limitare l'accesso agli oggetti e alle funzionalità di database, in base al "Principle of Least Privilege" (non fornire agli utenti permessi più elevati di quelli strettamente necessari per svolgere il loro lavoro).
Esempio:
Codice vulnerabile:

```
$unsafe_variable = $_POST['user_input'];
mysql_query("SELECT * FROM tabella WHERE name = '$unsafe_variablè");
```

Codice sicuro:
Utilizzando i Php Data Objets (PDO) si può scrivere una query con i prepared statement:

```
$stmt = $pdo->prepare('SELECT * FROM tabella WHERE name = :namè);
$stmt->execute(array('namè => $name));
foreach ($stmt as $row) {
  // Ciclo sulla riga ($row)
}
```

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

7.9.10 XPath Injection

Come riconoscerla

Gli attacchi XPath Injection sono possibili quando un sito Web utilizza le informazioni fornite dall'utente per costruire una query XPath per i dati XML. Inviando informazioni intenzionalmente malformate nel sito Web, un utente malintenzionato può scoprire come sono strutturati i dati XML o accedere a dati a cui normalmente non avrebbe accesso. Potrebbe persino essere in grado di elevare i suoi privilegi sul sito Web se i dati XML vengono utilizzati per l'autenticazione (come un file utente basato su XML).

Come difendersi

- Evitare che la costruzione della query XPath dipenda dalle informazioni inserite dall'utente. Possibilmente mapparla con i parametri utente mantenendo la separazione tra dati e codice. Nel