**Understanding NoSQL Injection**

**What is NoSQL injection?**

NoSQL injection is a security weakness that is primarily in a web application that incorporates a NoSQL database. Not Only SQL, which is also known as NoSQL, refers to a database that uses data formats that are more flexible and do not support Structured Query Language, or SQL. They store and manage data as key-value pairs, documents, or data graphs.

A NoSQL injection is very similar to a SQL injection. It allows cyber attackers to bypass all types of authentication, remove sensitive and confidential data, tamper with data that are present in the database, or even compromise the database and the background server. Maximum NoSQL injection vulnerabilities take place since developers accept and process inputs given by users without properly sanitizing or authenticating them.

NoSQL databases do not support SQL databases. Thus, the queries also depend on a lot of factors like database factors such as MongoDB, a programming language such as Python, and development tools like Angular.

**How does it work?**

NoSQL injection occurs when a query, which is most commonly delivered by an end-user, is not sanitized or protected. This non-sanitization allows the cyber attacker to insert malicious and harmful input that executes an unwanted command on the database.

Traditional SQL injection techniques do not work on NoSQL databases, since they use a specific query language and it does not support SQL. To attack NoSQL databases, attackers generally adjust their techniques and methods to product-specific query, which is generally written in the same language that is used to code the database engine. Since some NoSQL databases use application code, cyber attackers can successively perform unwanted and unexpected actions on a NoSQL database and execute malicious and defective code and unvalidated input within the application. This allows attackers to hijack servers and perform different vulnerable attacks, making NoSQL injections more severe than SQL injections.

In a general NoSQL network architecture, all the data access is managed and taken care of by a software driver. Libraries in multiple languages are available to the clients which allow them to access the database with ease. Even if these drivers are not vulnerable and harmful, they may possess insecure APIs, which in turn, are harmful. This is another threat that allows arbitrary and unknown code execution in the database and within the applications.

**How to prevent NoSQL Injection attacks?**

There are several ways by which NoSQL Injection attacks can be prevented. Some of them are enlisted down below.

### **1)Improvement of Developer Skills**

NoSQL attacks can be more difficult to prevent than traditional SQL injection because many NoSQL databases contain different unsafe or malicious code and functionality, which is unfamiliar to all the developers. The first step is to thoroughly go through the documentation and security guidelines for NoSQL databases. Apart from all these, teams should invest in proper training to make sure that all the developers are used to and well equipped with the database engine being used, and understand how to correctly implement security practices.

### **2)Using the Latest Version**

### Different popular NoSQL products are in their active development phase, so it is very important to use the latest and updated version and upgrade regularly. Different vulnerabilities are discovered in NoSQL databases daily. To cite an example, older versions of MongoDB are less secure and more vulnerable, whereas newer versions are more secure and less vulnerable.

### **3)Avoid Accepting Raw User Input**

### The best and most effective way to prevent NoSQL injection attacks is to avoid using raw user input in your application code, especially during writing database queries. For example, MongoDB has different built-in functionalities to help build secure queries without using JavaScript.

**References**

1)<https://www.imperva.com/learn/application-security/nosql-injection/#:~:text=NoSQL%20injection%20occurs%20when%20a,unwanted%20command%20on%20the%20database>.

2)<https://www.invicti.com/blog/web-security/what-is-nosql-injection/>

3)<https://www.indusface.com/blog/what-is-nosql-injection-attack-and-how-to-prevent-it/>

4)<https://www.acunetix.com/blog/web-security-zone/nosql-injections/>

5)<https://protectonce.com/appsec-academy-for-developer/what-is-nosql-injection/>

6)<https://book.hacktricks.xyz/pentesting-web/nosql-injection>

7)<https://www.wallarm.com/what/nosql-injection-attack>