Nanyang Technological University

AY2021-2022 Semester 2

# CE/CZ4067 SOFTWARE SECURITY

## Tutorial 9: NoSQL Injection and Taint Analysis

1. A NoSQL injection vulnerability is an error in a web application that uses a NoSQL database. This Web application security issue lets a malicious party bypass authentication, extract data, modify data, or even gain complete control over the application. Collect information about NoSQL injection on the Internet (for example, [2, 1, 3]) and answer the following questions.

   (a) What is the purpose of the following query?

   ```
   $collection->find(array(
       "username" => $_GET['username'],
       "passwd" => $_GET['passwd']));
   ```

   (b) How is the query affected if the following input is provided by the attacker? Write down the equivalent SQL query sent to the database.

   ```
   login.php?username=admin&passwd[$ne]=1
   ```

   (c) List at least three common defenses against NoSQL injection.

2. Taint analysis is a process used in computer security to identify the flow of user input through a system to understand the security implications of the system design. Taint analysis can be performed either dynamically or statically. Figure 1 shows a simple Java program used to process user inputs and construct queries accordingly. Perform dynamic and static taint analyses on the program to issue warnings whenever a query statement may contain unsanitized user inputs.

   (a) Which line(s) should be marked as taint source? Which line(s) should be marked as sink?

```
1 a = read();
2 c = read();
3 if (a.equals("hello")) {
4   b = a + "world";
5 } else {
6   a = sanitize(c);
7 }
8 query(c);
9 query(b);
```

Figure 1: A simple Java program used to handle user inputs.

(b) Fill in Table 1 to simulate the taint propagation in a dynamic program run. Does the source flow into the sink?

Table 1: Taint tracking in a dynamic run.

| | Line | 1 | 2 |
|---|---|---|---|
| a | Value | "4067" | "4067" |
| | Taint | T | T |
| b | Value | $\perp$ | $\perp$ |
| | Taint | N | N |
| c | Value | $\perp$ | "attack" |
| | Taint | N | T |

(c) Fill in Fig. 2 to simulate the taint propagation in a static taint analysis. Does the source flow into the sink?

# References

[1] Hacking NodeJS and MongoDB. https://blog.websecurify.com/2014/08/hacking-nodejs-and-mongodb.html, November 2021.

[2] NoSQL injection in MongoDB. https://zanon.io/posts/nosql-injection-in-mongodb/, November 2021.

[3] Patrick Spiegel. NoSQL injection. https://entwicklertag.de/karlsruhe/2017/sites/entwicklertag.de.karlsruhe.2017/files/folien/PresentationNoSQLiPatrick.pdf, November 2021.

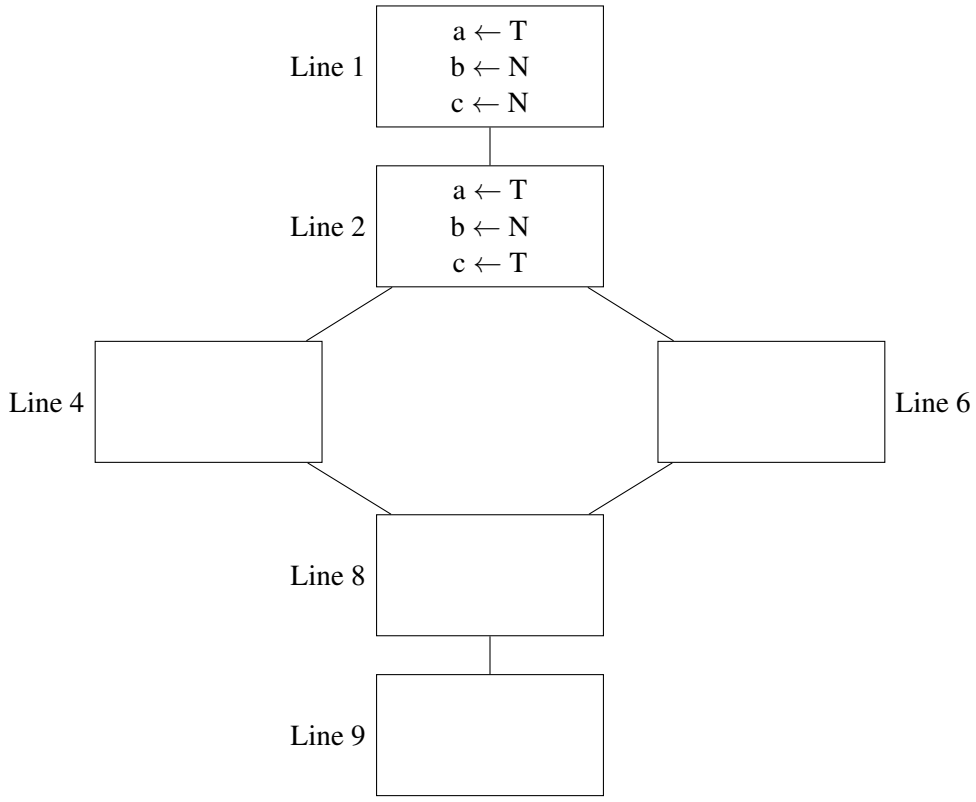


Figure 2: Control-flow graph of the simple Java program.