CSE: 5382-001: SECURE PROGRAMMING

ASSIGNMENT 10

Tharoon T Thiagarajan

1001704601

Part 1:

Manual Static Analysis:

From the given program, when I performed manual static analysis on the program I was able to see that the BufferReader class was not closed and this can cause leakage of memory and exhausting of resources leading to security breach.

```
45  public void processRequest(Socket s) throws Exception {
46      /* used to read data from the client */
47      BufferedReader br =
48          new BufferedReader (
49                  new InputStreamReader (s.getInputStream()));
```

In the same program at line 103, I was able to notice that the FileReader was also not closed. Failing to close the FileReader will also cause leakage of memory leading to severe performance issue. This performance issue will cause improper allocation of memory and can lead to serious threat to the application.

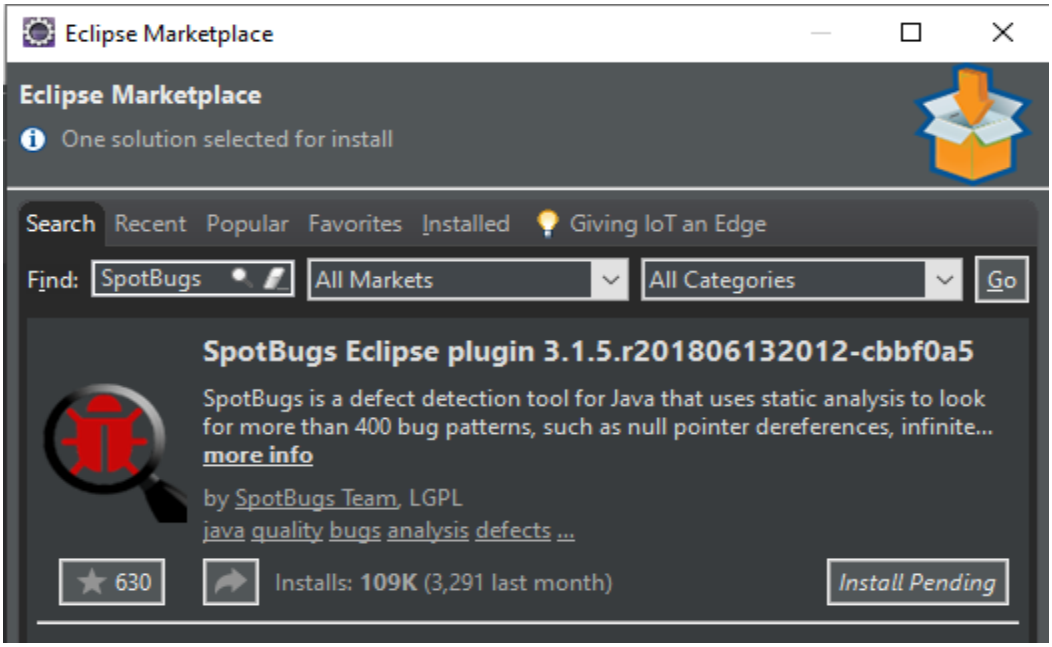
```
101     /* try to open file specified by pathname */
102     try {
103         fr = new FileReader (pathname);
104         c = fr.read();
105     }
106     catch (Exception e) {
107         /* if the file is not found,return the
108            appropriate HTTP response code  */
109         osw.write ("HTTP/1.0 404 Not Found\n\n");
110         return;
111     }
```

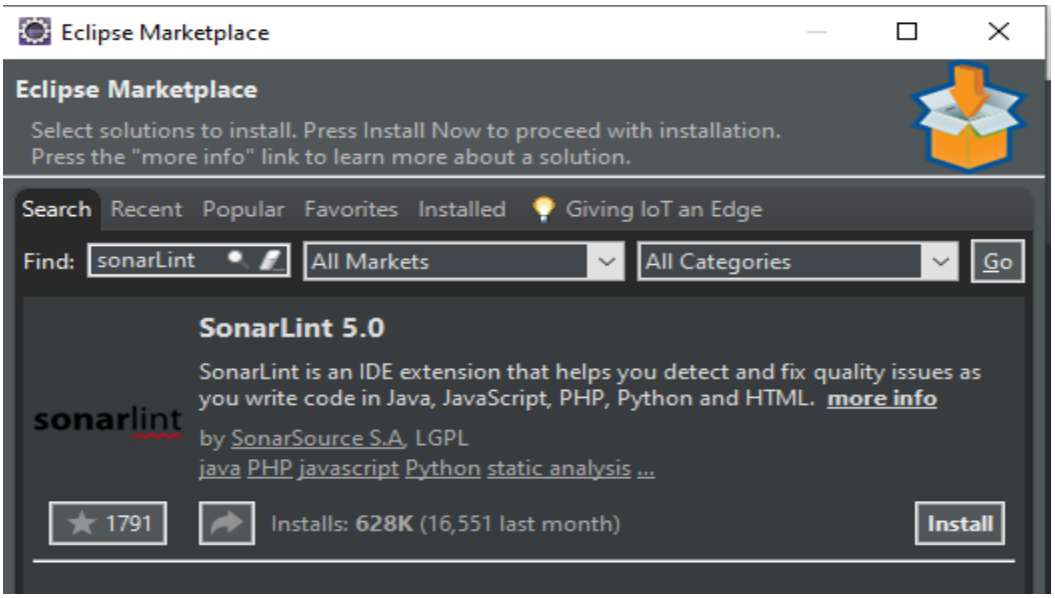
Tool Choices/Versions

For the tools of choices for the static code analysis, I chose SpotBugs and SonarLint as the second tool of choice. I used Eclipse IDE for inspecting the java code and I have attached the screenshots of the tools from the Eclipse marketplace.

The version of SpotBugs I used was SpotBubs 3.1.5

The version of SonarLint I used was SonarLint 5.0

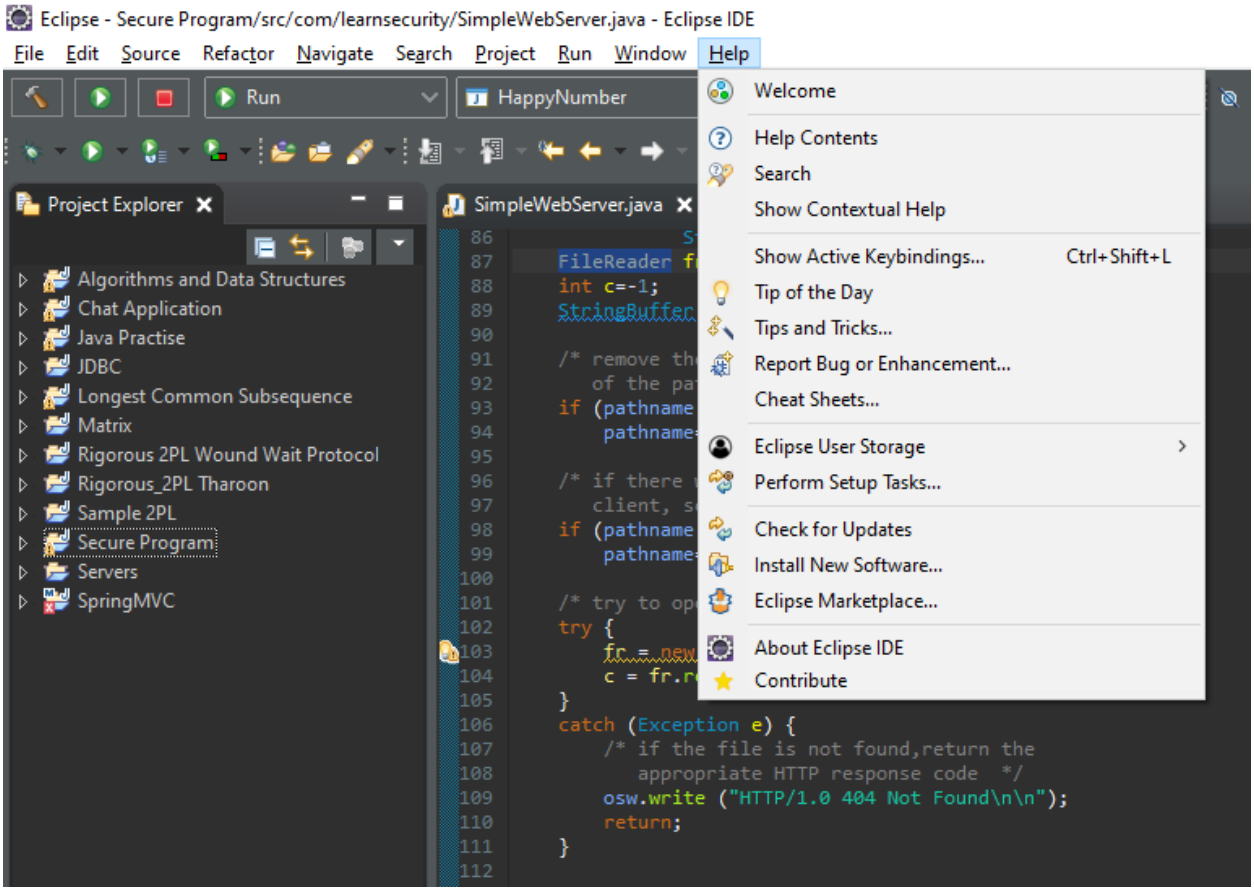




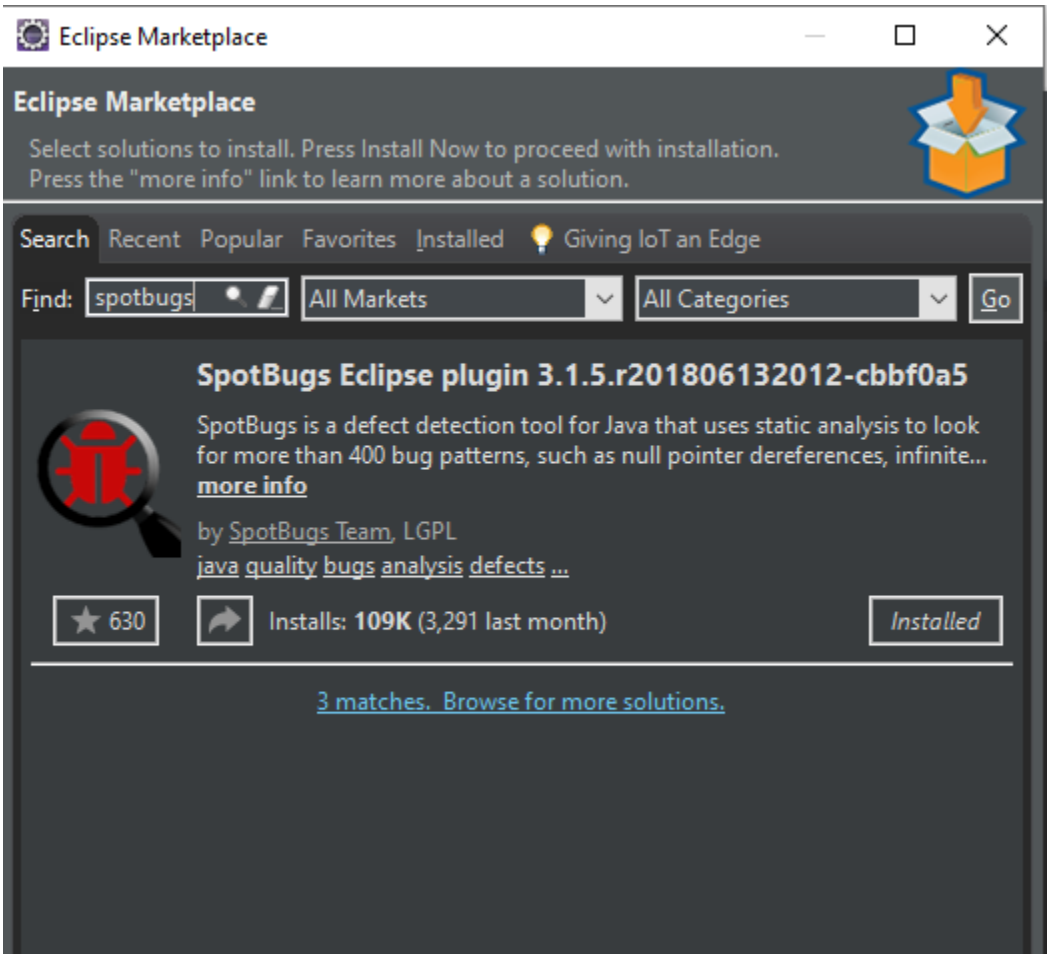
Tool Invocation Process:

To install the SpotBugs plug in in Eclipse IDE,

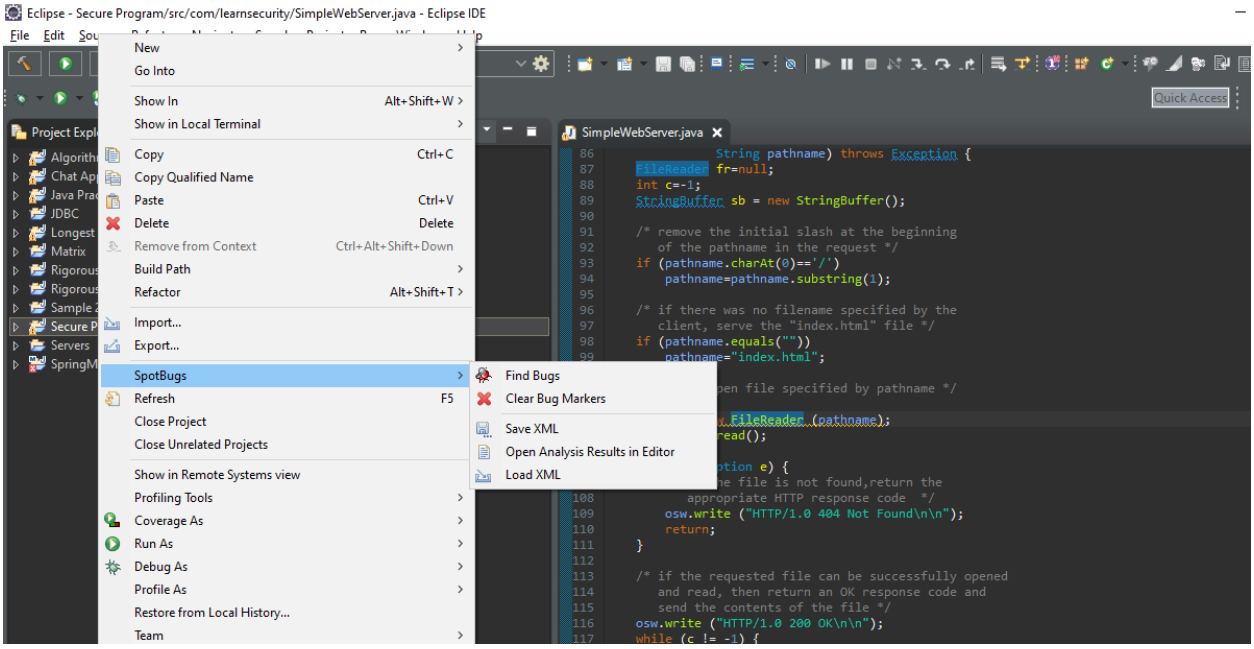
Go to Eclipse -> Help -> Eclipse Marketplace



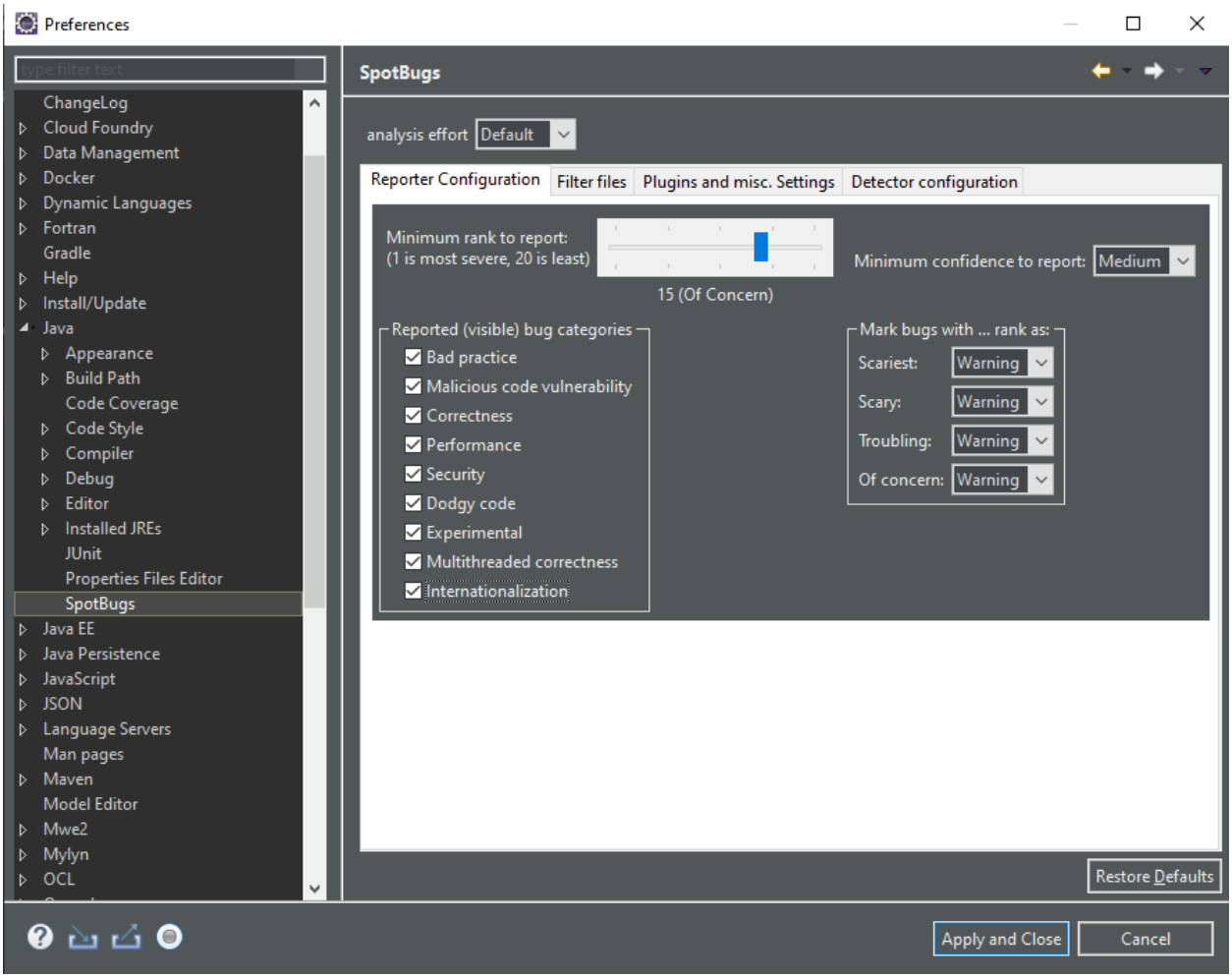
In the eclipse marketplace window, type SpotBugs in the Find field and click go. You will be able to see the plugin for SpotBugs getting displayed. Click on the install option to install the plugin. Accept the Terms and conditions to install SpotBugs in Eclipse.



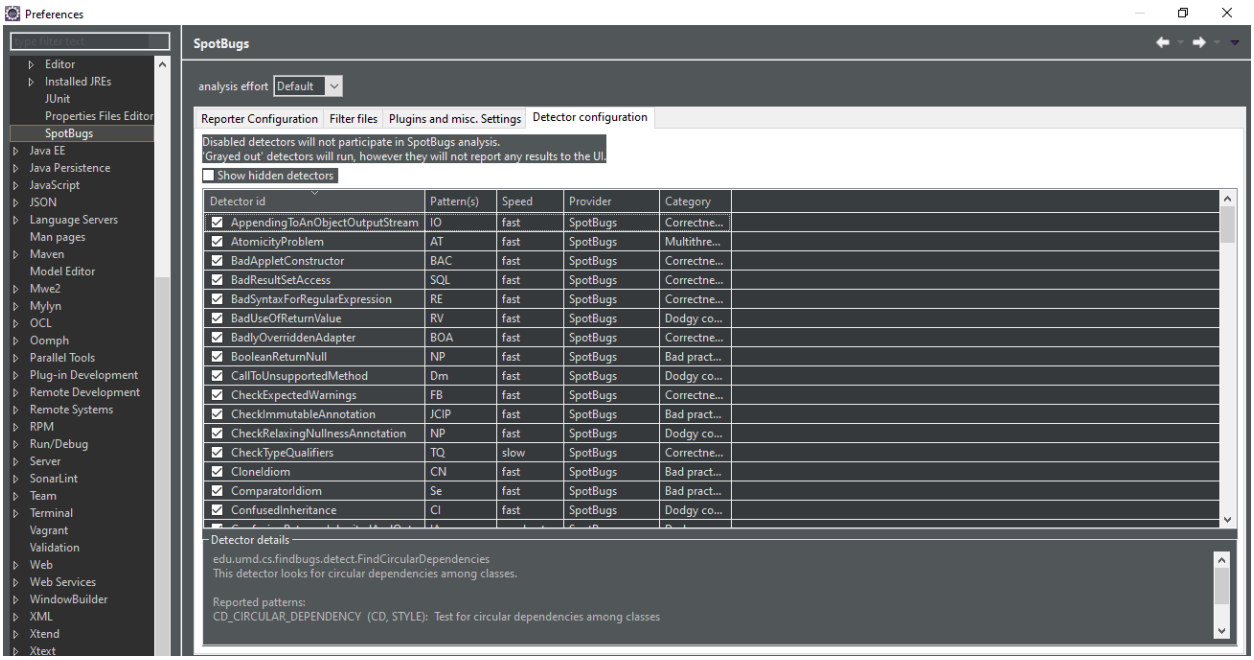
To check if SpotBugs is installed in Eclipse, Right click on your project and you will be able to see an option for SpotBugs.



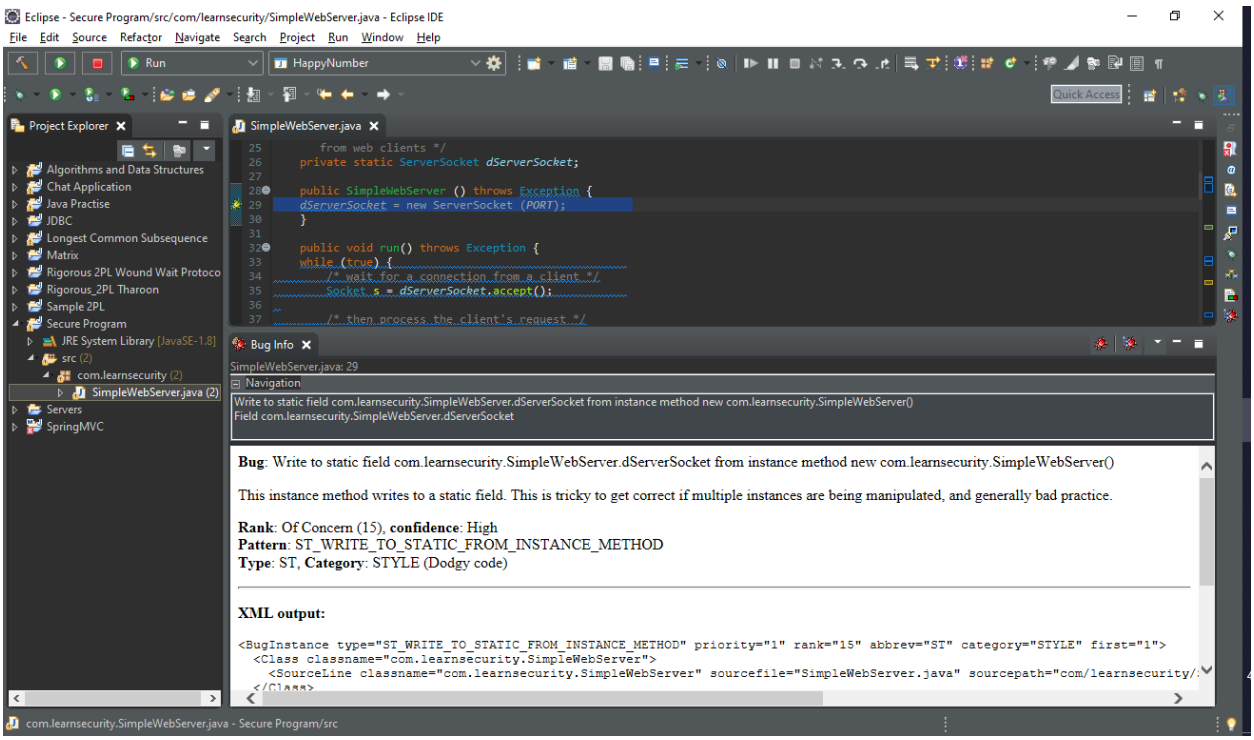
Then go to Window -> Preferences and choose SpotBugs under the Java option on the left pane. Under Reporter Configuration check all the visible bug categories so that the tool reports all the security related vulnerabilities.



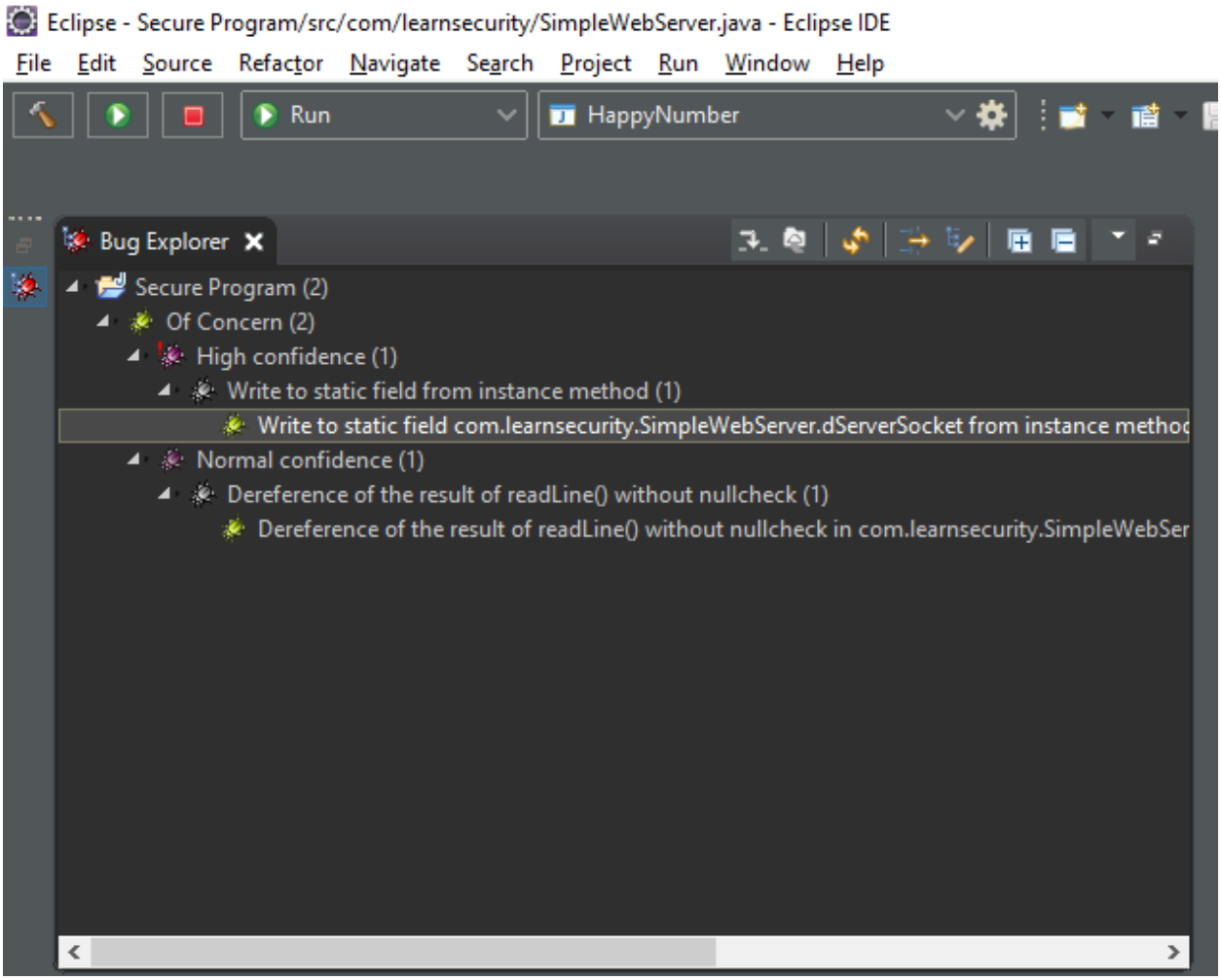
Then switch to Detector Configuration tab in the same window and check all the options so that the SpotBugs plugin will participate in all bug analysis.



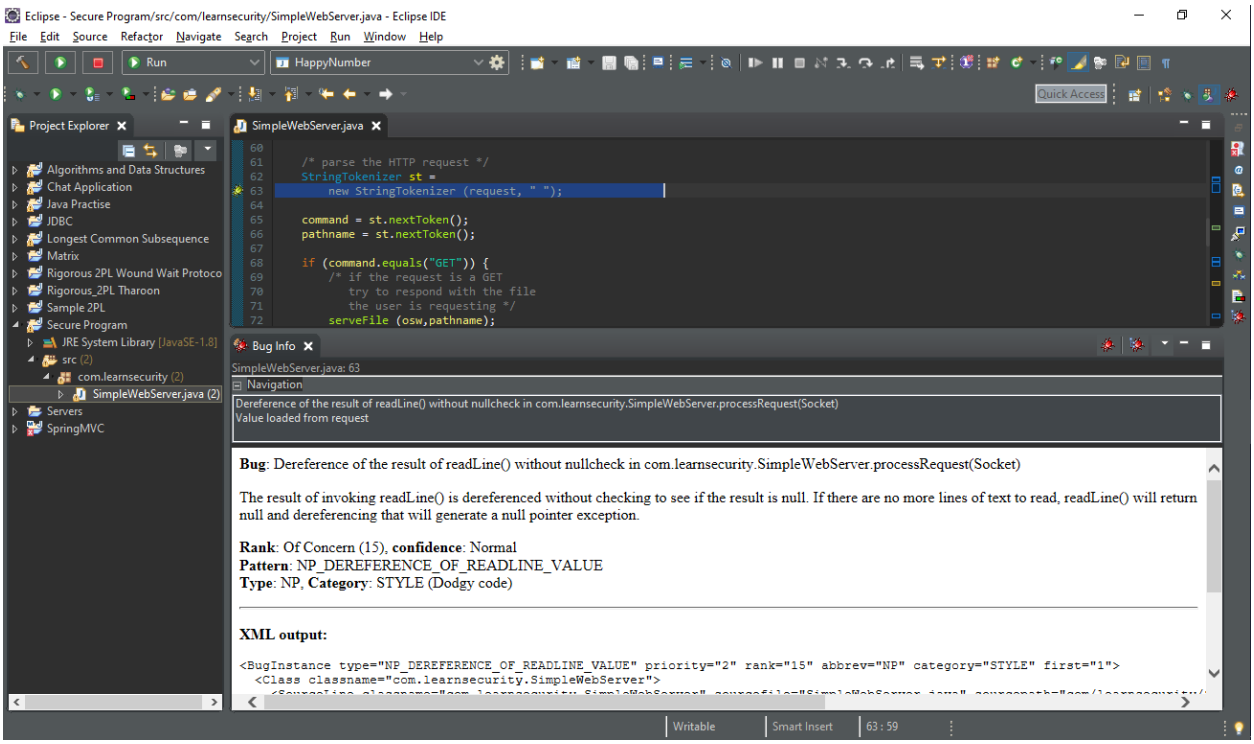
After installing and configuring the SpotBugs plugin, I created a Java Project in Eclipse and created a Java Class for the given program to perform static code analysis. Then I right clicked on the java program and chose the option Find bugs under the SpotBugs. Once I clicked the Find Bugs option, I was able to see 2 bugs getting displayed. The first bug is on line 29 where the SpotBug reported a ST_WRITE_TO_STATIC_FROM_INSTANCE_METHOD issue with High confidence. The bug states that the instance method writes to a static field multiple times.



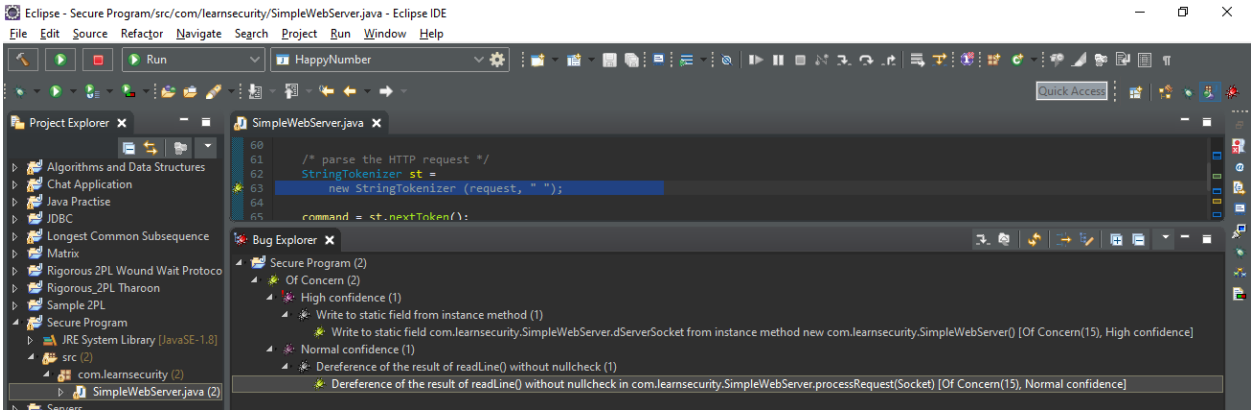
SpotBugs plugin has two types of viewing the bug. One is the Bug Info view and the other one is the Bug Explorer view. This is the snapshot of the Bug Explorer view showing the number of bugs reported.



The second bug is on the line 63. SpotBug reported it as a NP_DEREFERENCE_OF_READLINE_VALUE with normal confidence. The bug states that the readLine() is dereferenced without checking to see if the result is null. If there are no more lines of text to read, readLine() will return null and dereferencing that will generate a null pointer exception.



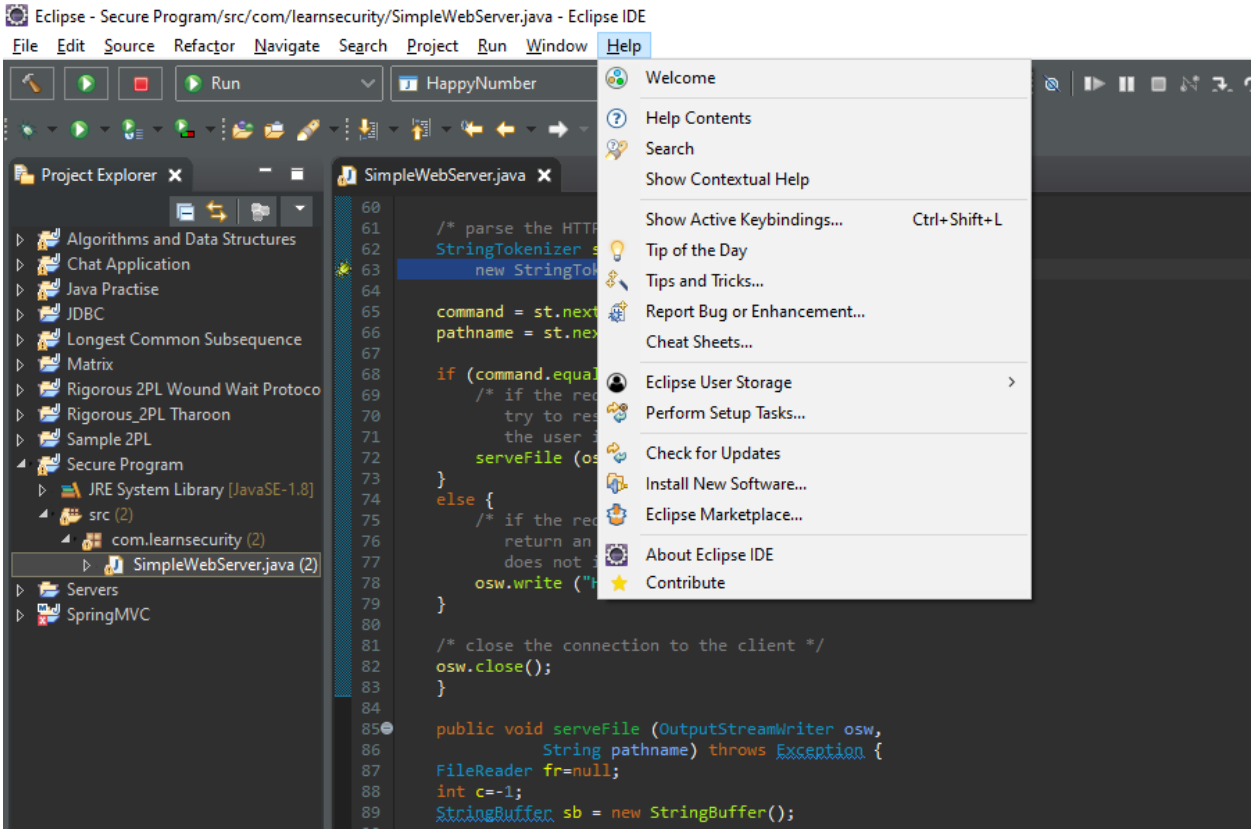
This is the snapshot of the Bug Explorer view for the second bug in the program.



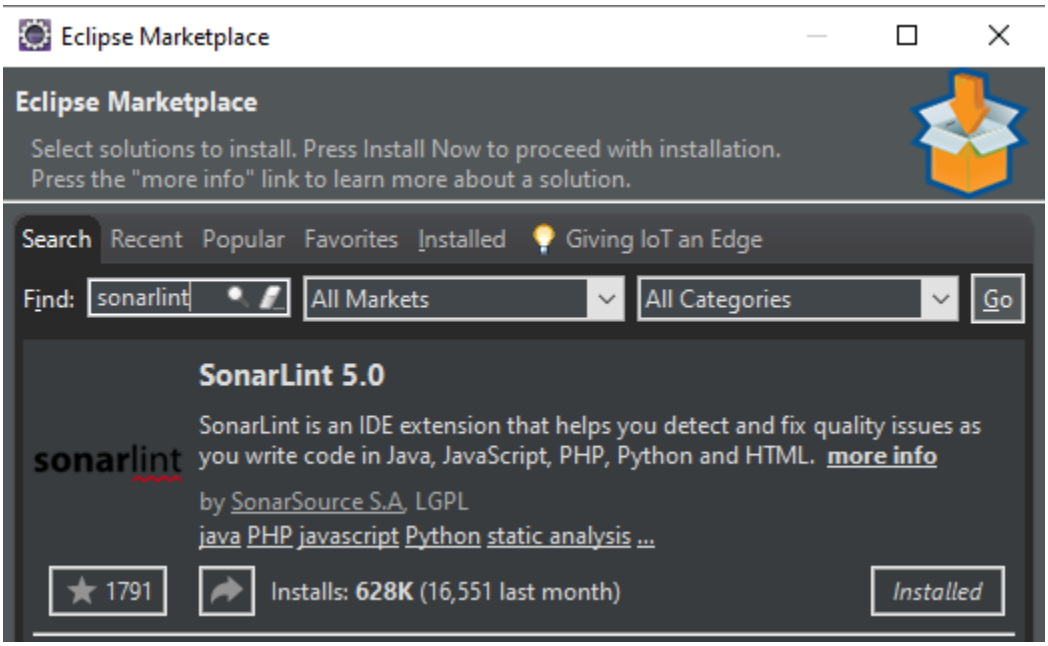
Using SonarLint:

To install the SonarLint plug in in Eclipse IDE,

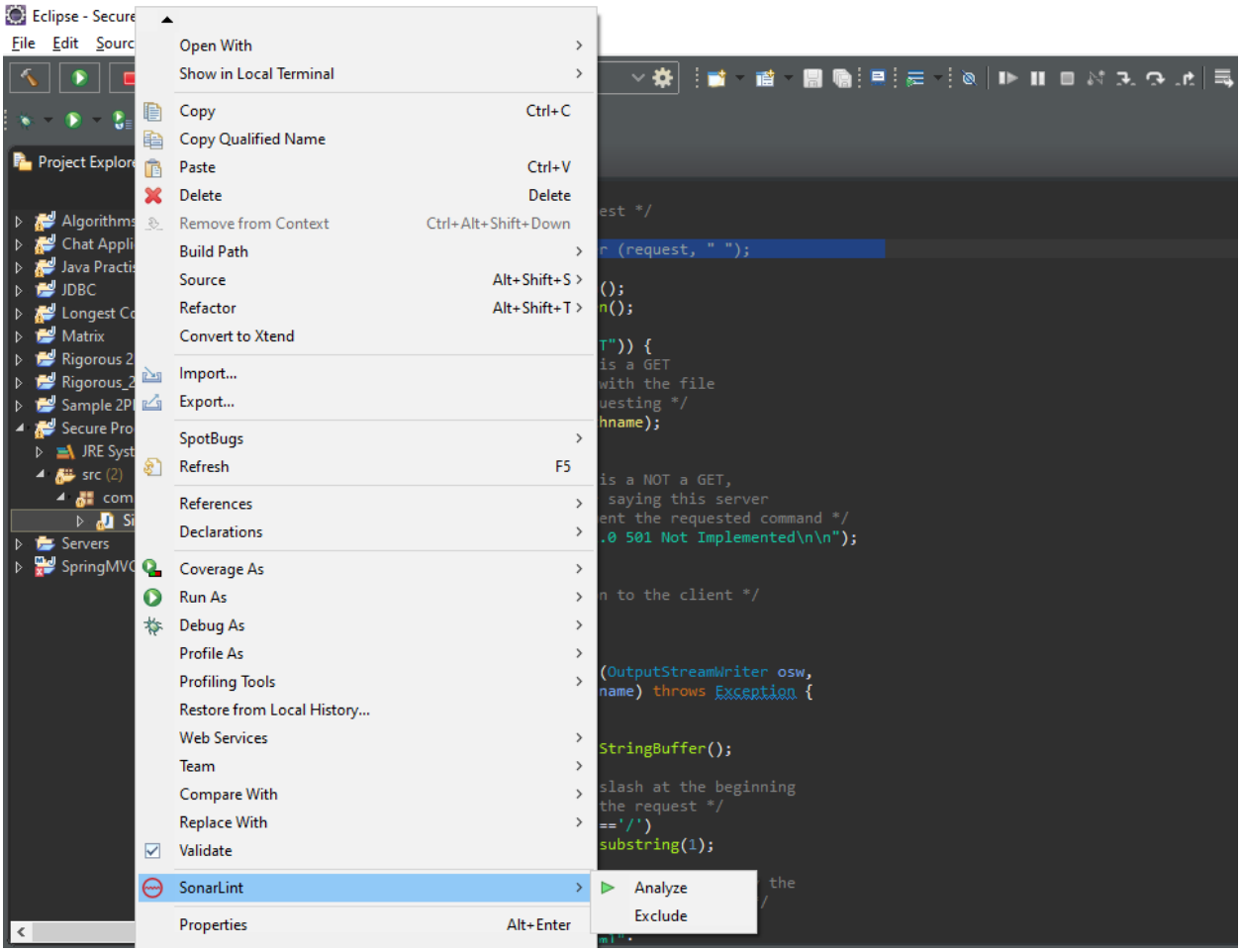
Go to Eclipse -> Help -> Eclipse Marketplace



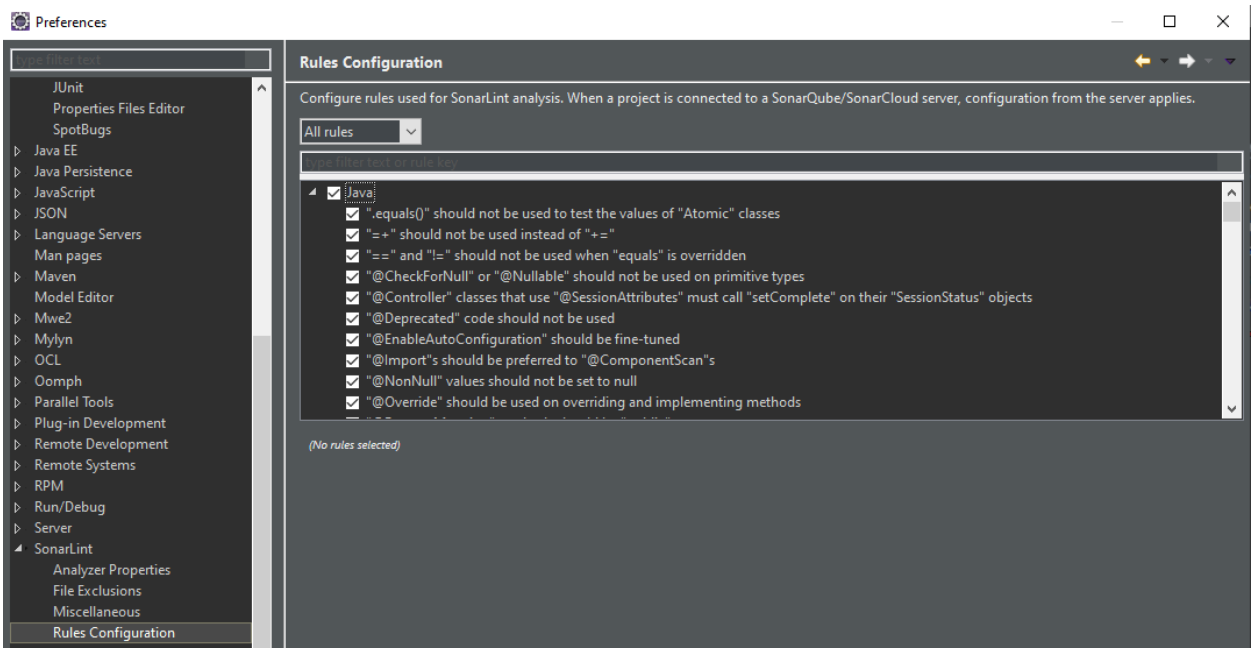
In the eclipse marketplace window, type SonarLint in the Find field and click go. You will be able to see the plugin for SonarLint getting displayed. Click on the install option to install the plugin. Accept the Terms and conditions to install SonarLint in Eclipse.



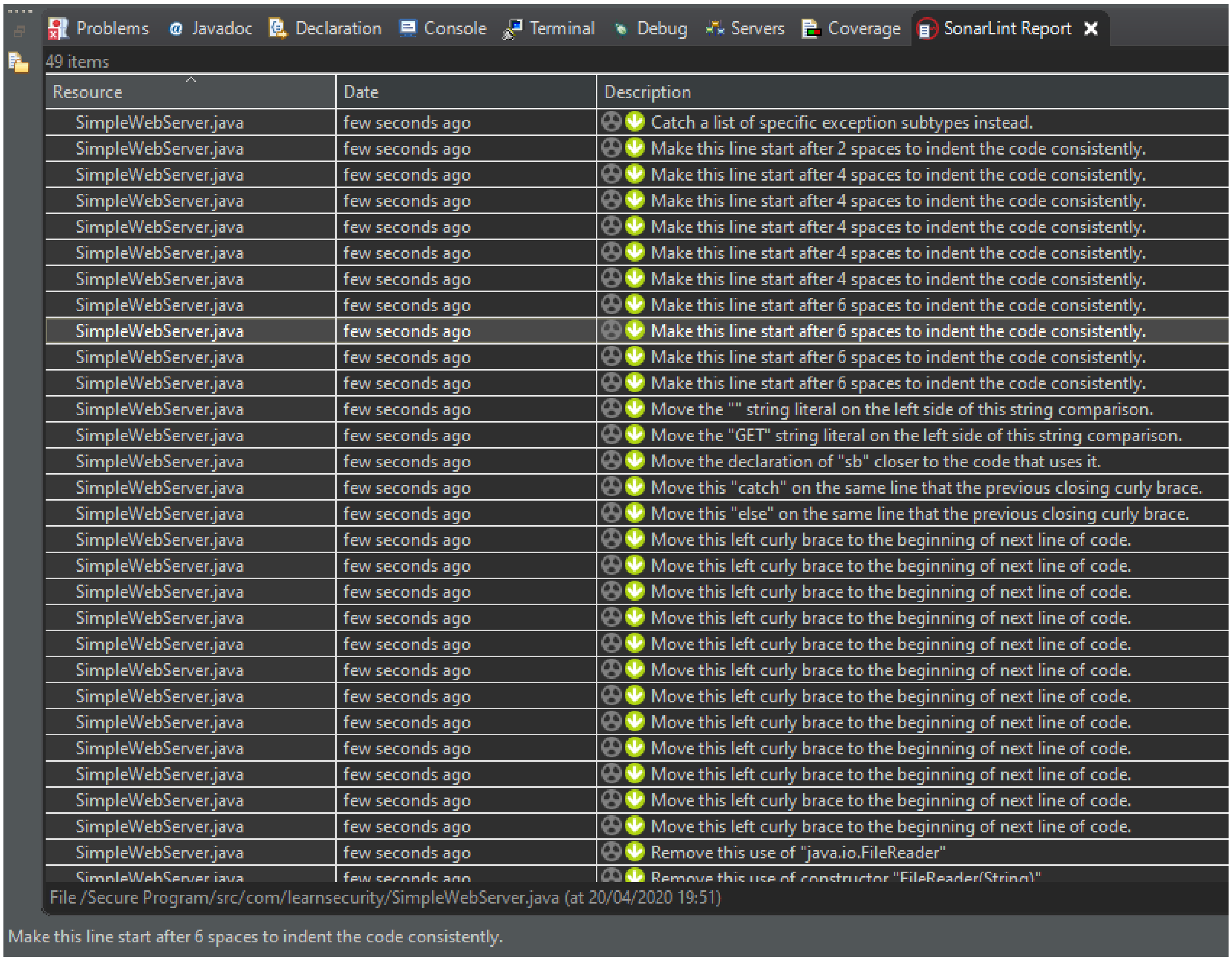
To check if SonarLint is installed in Eclipse, Right click on your project and you will be able to see an option for SonarLint.



Then go to Window -> Preferences and choose SonarLint on the left pane of the window. Under SonarLint option choose Rules configuration. Under the Rules Configuration choose Java because we are going to analyze the java code. Check all the options under Java so that we can analyze the java code rigorously.



After installing and configuring the SpotBugs plugin, I created a Java Project in Eclipse and created a Java Class for the given program to perform static code analysis. Then I right clicked on the java program and chose the option Analyze under SonarLint option. I was able to see that there was a detailed description of the bug present in the program.



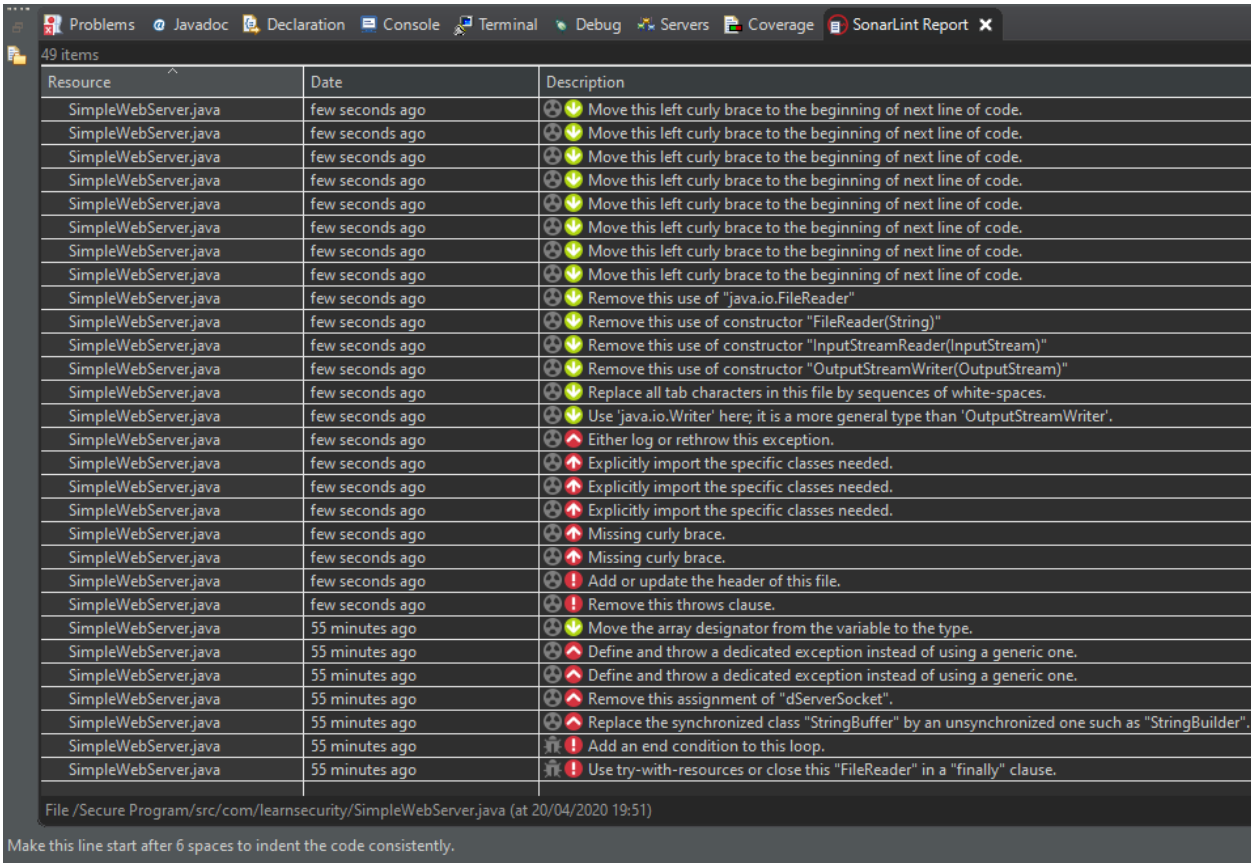
49 items

Resource	Date	Description
SimpleWebServer.java	few seconds ago	Catch a list of specific exception subtypes instead.
SimpleWebServer.java	few seconds ago	Make this line start after 2 spaces to indent the code consistently.
SimpleWebServer.java	few seconds ago	Make this line start after 4 spaces to indent the code consistently.
SimpleWebServer.java	few seconds ago	Make this line start after 4 spaces to indent the code consistently.
SimpleWebServer.java	few seconds ago	Make this line start after 4 spaces to indent the code consistently.
SimpleWebServer.java	few seconds ago	Make this line start after 4 spaces to indent the code consistently.
SimpleWebServer.java	few seconds ago	Make this line start after 4 spaces to indent the code consistently.
SimpleWebServer.java	few seconds ago	Make this line start after 6 spaces to indent the code consistently.
SimpleWebServer.java	few seconds ago	Make this line start after 6 spaces to indent the code consistently.
SimpleWebServer.java	few seconds ago	Make this line start after 6 spaces to indent the code consistently.
SimpleWebServer.java	few seconds ago	Make this line start after 6 spaces to indent the code consistently.
SimpleWebServer.java	few seconds ago	Move the "" string literal on the left side of this string comparison.
SimpleWebServer.java	few seconds ago	Move the "GET" string literal on the left side of this string comparison.
SimpleWebServer.java	few seconds ago	Move the declaration of "sb" closer to the code that uses it.
SimpleWebServer.java	few seconds ago	Move this "catch" on the same line that the previous closing curly brace.
SimpleWebServer.java	few seconds ago	Move this "else" on the same line that the previous closing curly brace.
SimpleWebServer.java	few seconds ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	few seconds ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	few seconds ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	few seconds ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	few seconds ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	few seconds ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	few seconds ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	few seconds ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	few seconds ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	few seconds ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	few seconds ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	few seconds ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	few seconds ago	Remove this use of "java.io.FileReader"
SimpleWebServer.java	few seconds ago	Remove this use of constructor "FileReader(String)"

File /Secure Program/src/com/learnsecurity/SimpleWebServer.java (at 20/04/2020 19:51)

Make this line start after 6 spaces to indent the code consistently.

The red color indication depicts that there is a serious bug in the program and that those lines must be corrected in order to avoid any severe security breach to the application.



49 items

Resource	Date	Description
SimpleWebServer.java	few seconds ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	few seconds ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	few seconds ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	few seconds ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	few seconds ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	few seconds ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	few seconds ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	few seconds ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	few seconds ago	Remove this use of "java.io.FileReader"
SimpleWebServer.java	few seconds ago	Remove this use of constructor "FileReader(String)"
SimpleWebServer.java	few seconds ago	Remove this use of constructor "InputStreamReader(InputStream)"
SimpleWebServer.java	few seconds ago	Remove this use of constructor "OutputStreamWriter(OutputStream)"
SimpleWebServer.java	few seconds ago	Replace all tab characters in this file by sequences of white-spaces.
SimpleWebServer.java	few seconds ago	Use 'java.io.Writer' here; it is a more general type than 'OutputStreamWriter'.
SimpleWebServer.java	few seconds ago	Either log or rethrow this exception.
SimpleWebServer.java	few seconds ago	Explicitly import the specific classes needed.
SimpleWebServer.java	few seconds ago	Explicitly import the specific classes needed.
SimpleWebServer.java	few seconds ago	Explicitly import the specific classes needed.
SimpleWebServer.java	few seconds ago	Missing curly brace.
SimpleWebServer.java	few seconds ago	Missing curly brace.
SimpleWebServer.java	few seconds ago	Add or update the header of this file.
SimpleWebServer.java	few seconds ago	Remove this throws clause.
SimpleWebServer.java	55 minutes ago	Move the array designator from the variable to the type.
SimpleWebServer.java	55 minutes ago	Define and throw a dedicated exception instead of using a generic one.
SimpleWebServer.java	55 minutes ago	Define and throw a dedicated exception instead of using a generic one.
SimpleWebServer.java	55 minutes ago	Remove this assignment of "dServerSocket".
SimpleWebServer.java	55 minutes ago	Replace the synchronized class "StringBuffer" by an unsynchronized one such as "StringBuilder".
SimpleWebServer.java	55 minutes ago	Add an end condition to this loop.
SimpleWebServer.java	55 minutes ago	Use try-with-resources or close this "FileReader" in a "finally" clause.

File /Secure Program/src/com/learnsecurity/SimpleWebServer.java (at 20/04/2020 19:51)

Make this line start after 6 spaces to indent the code consistently.

Comparison/Contrast:

Does the tool analyze source or binary as input?

SpotBugs:

SpotBugs is a static analysis tool and it analysis the byte code of the source program. It analysis the byte code to find the bugs in the program.

SonarLint:

SonarLint analyses the source code of the program to analyse bugs. The main objective of SonarLint is to make the quality of the code available to everyone with very precise information about the bug.

Which category of tools is it?

Spotbug is a,

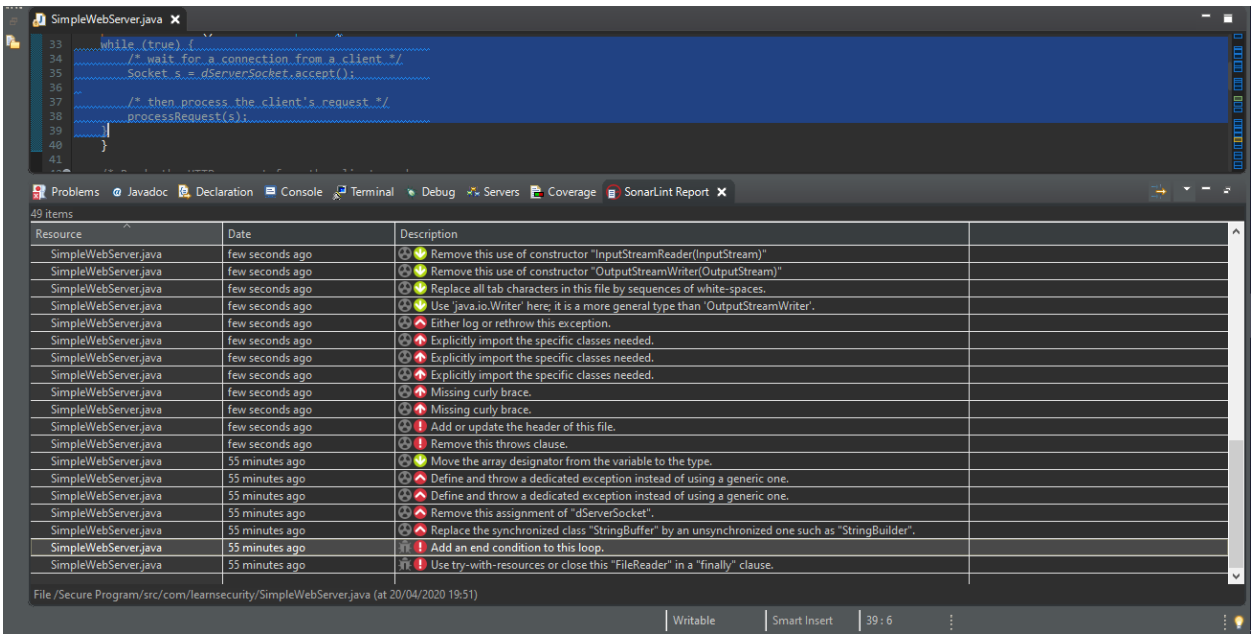
- Typechecking
- Security review
- Bug finding

SonarLint is a,

- StyleChecking
- Bug Finding
- Security review

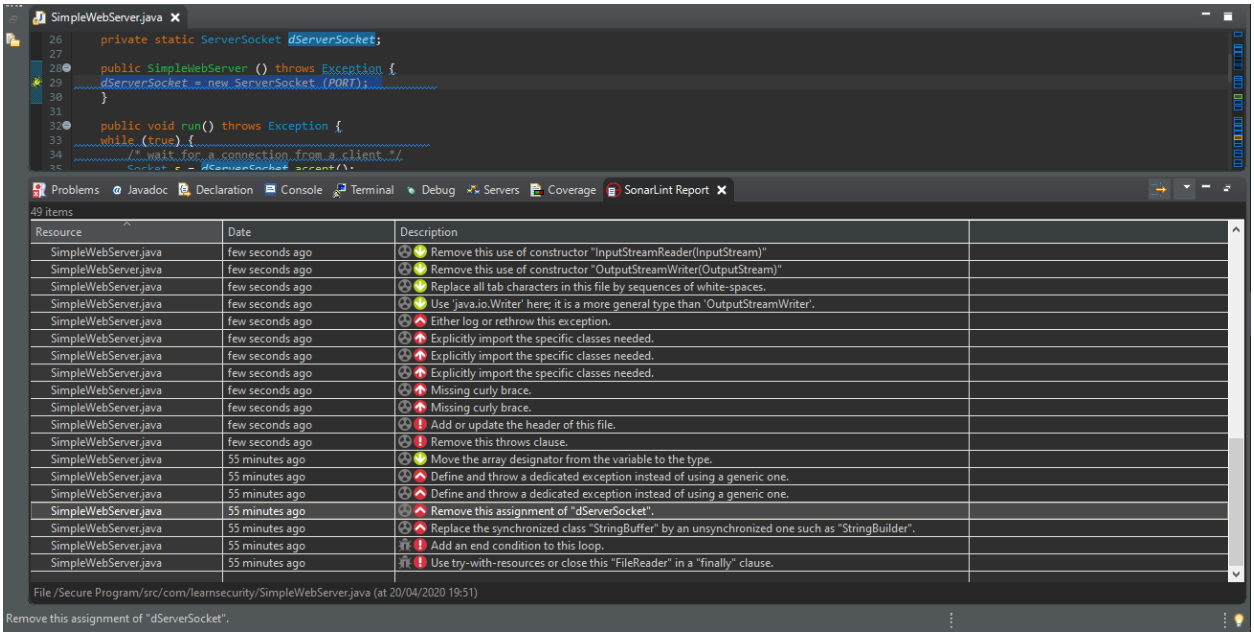
Show an example (if one exists) of a finding that is reported by one tool and not others.

There was a bug reported by SonarLint in line 33, where there was no end condition for the while loop. Whereas the SpotBug did not report this bug.

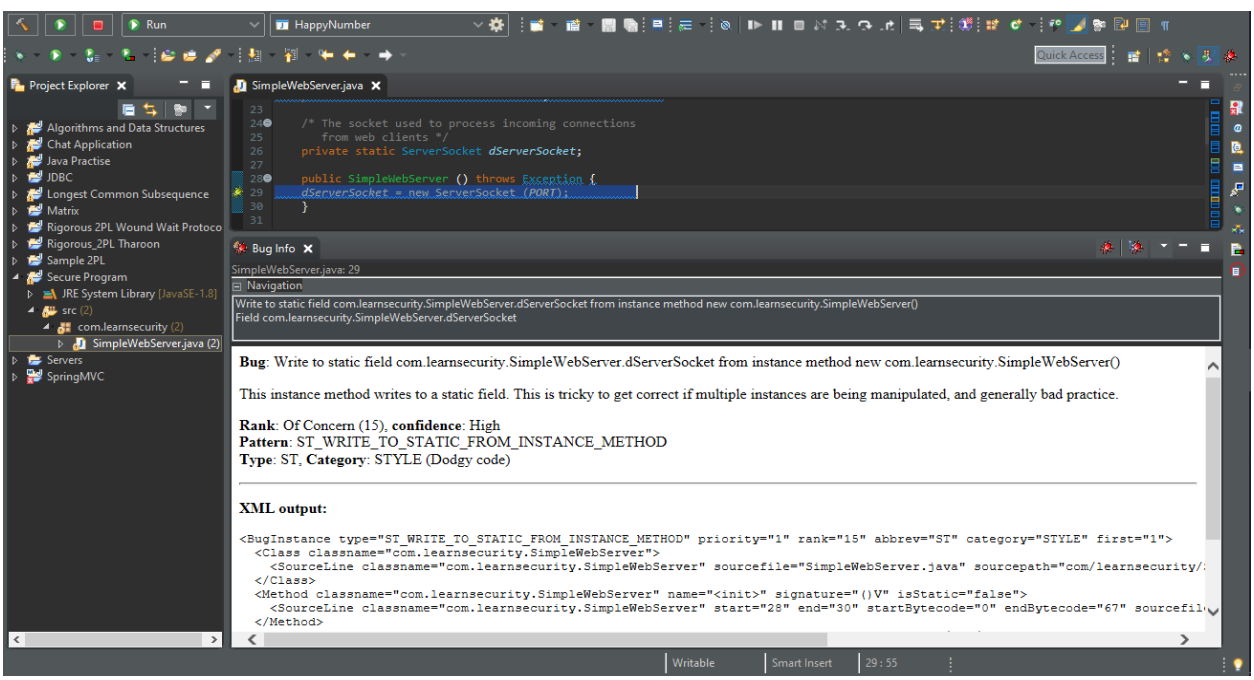


Show an example (if one exists) of a finding reported by multiple tools

There was a bug reported by SonarLint on the line 29 of the given program. The description of the bug was to remove the assignment dServerSocket in the program. The same bug was reported by SpotBugs on line 29 of the given program. SpotBugs reported the same problem of dServerSocket. Both the tool spotted the same error in the given program.

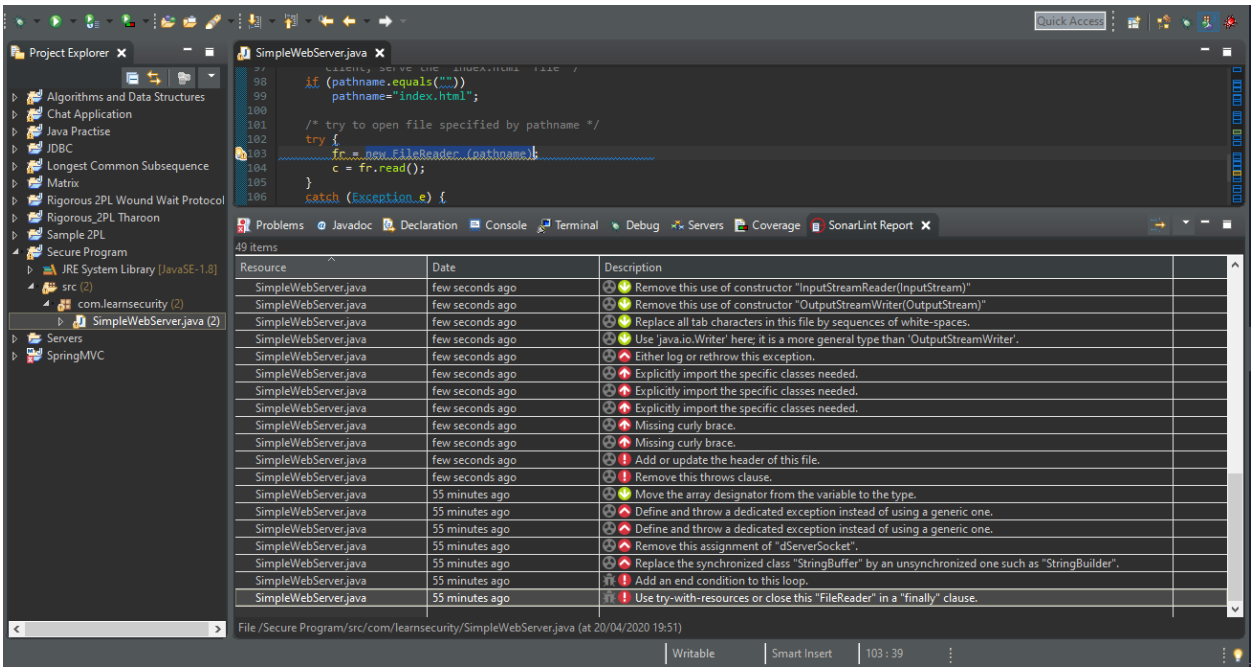


This is the snapshot of Spotbug reporting the same bug in the given program.

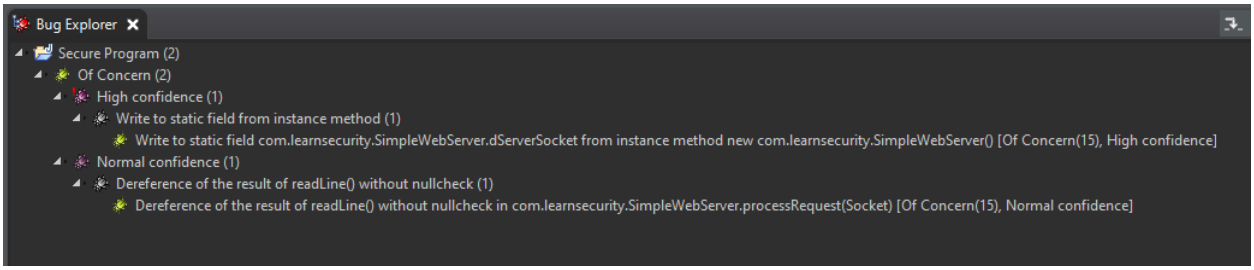


For the known flaw in the code used, document which tools reported it (true negative) and which tools did not (false positive).

There is a bug on line 103 of the program. I have reported the bug in Manual static analysis of the given program. The bug is that the FileReader is not closed properly in the program. SonarLint has reported the bug of FileReader that it has to be closed to prevent any security vulnerability. The bug reported by SonarLint is that “Use try with resources or close this FileReader in a finally clause”. Whereas SpotBug has failed to report this bug.



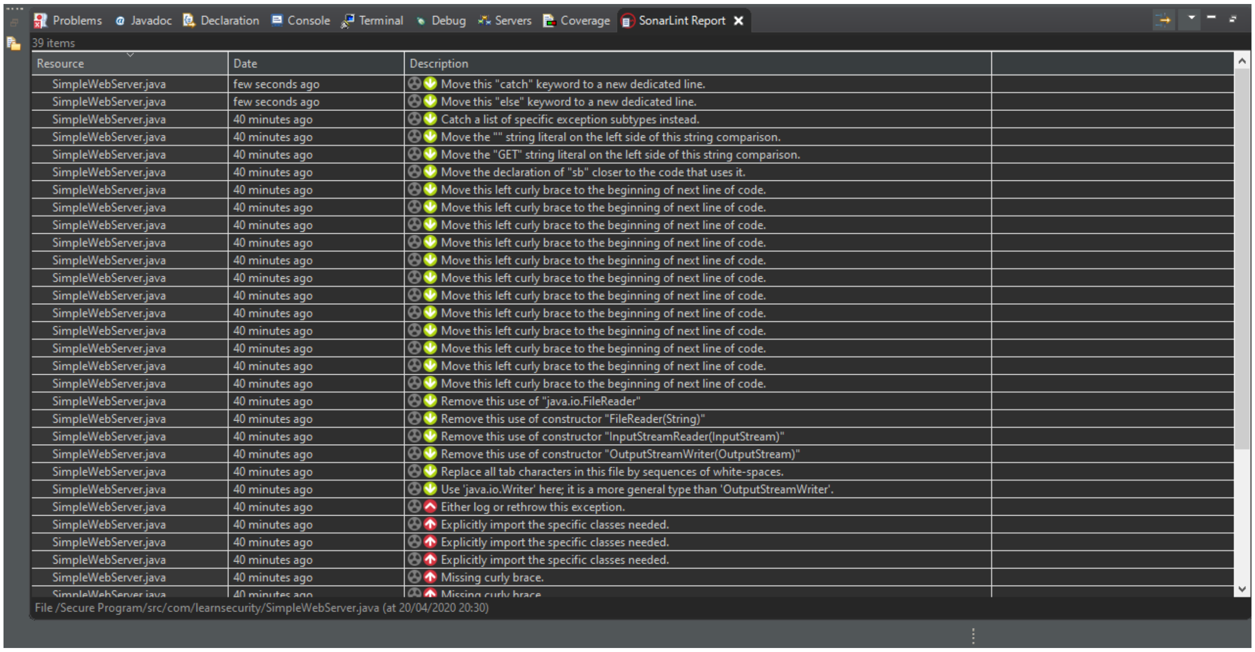
This is the snapshot of SpotBug report analysis where it only reported two bugs in the program.



Results:

a) Raw results provided:

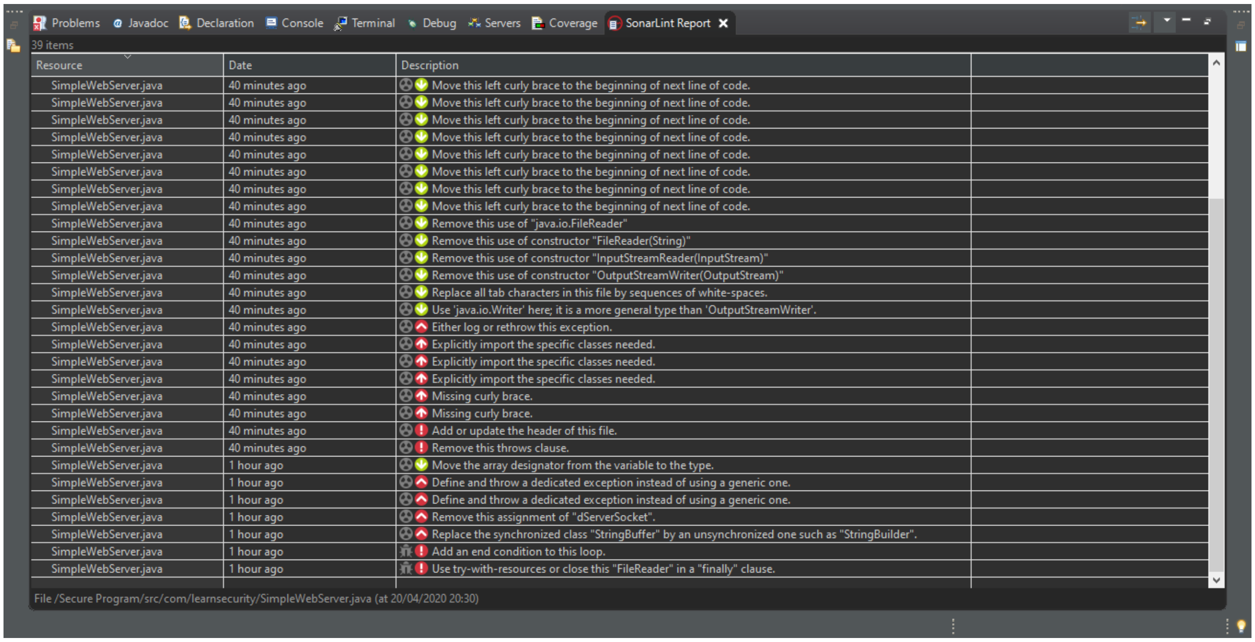
I have attached the snapshots of the bug report by both the tools SpotBugs and SonarLint. I have also attached the reports exported in the Excel format from both the tools. The name of the files are SonarLint part1.xlsx, SpotBugs part1.xlsx and SpotBugsReport.xml.



39 items

Resource	Date	Description
SimpleWebServer.java	few seconds ago	Move this "catch" keyword to a new dedicated line.
SimpleWebServer.java	few seconds ago	Move this "else" keyword to a new dedicated line.
SimpleWebServer.java	40 minutes ago	Catch a list of specific exception subtypes instead.
SimpleWebServer.java	40 minutes ago	Move the "" string literal on the left side of this string comparison.
SimpleWebServer.java	40 minutes ago	Move the "GET" string literal on the left side of this string comparison.
SimpleWebServer.java	40 minutes ago	Move the declaration of "sb" closer to the code that uses it.
SimpleWebServer.java	40 minutes ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	40 minutes ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	40 minutes ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	40 minutes ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	40 minutes ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	40 minutes ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	40 minutes ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	40 minutes ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	40 minutes ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	40 minutes ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	40 minutes ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	40 minutes ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	40 minutes ago	Remove this use of "java.io.FileReader"
SimpleWebServer.java	40 minutes ago	Remove this use of constructor "FileReader(String)"
SimpleWebServer.java	40 minutes ago	Remove this use of constructor "InputStreamReader(InputStream)"
SimpleWebServer.java	40 minutes ago	Remove this use of constructor "OutputStreamWriter(OutputStream)"
SimpleWebServer.java	40 minutes ago	Replace all tab characters in this file by sequences of white-spaces.
SimpleWebServer.java	40 minutes ago	Use 'java.io.Writer' here; it is a more general type than 'OutputStreamWriter'.
SimpleWebServer.java	40 minutes ago	Either log or rethrow this exception.
SimpleWebServer.java	40 minutes ago	Explicitly import the specific classes needed.
SimpleWebServer.java	40 minutes ago	Explicitly import the specific classes needed.
SimpleWebServer.java	40 minutes ago	Explicitly import the specific classes needed.
SimpleWebServer.java	40 minutes ago	Missing curly brace.
SimpleWebServer.java	40 minutes ago	Missing curly brace

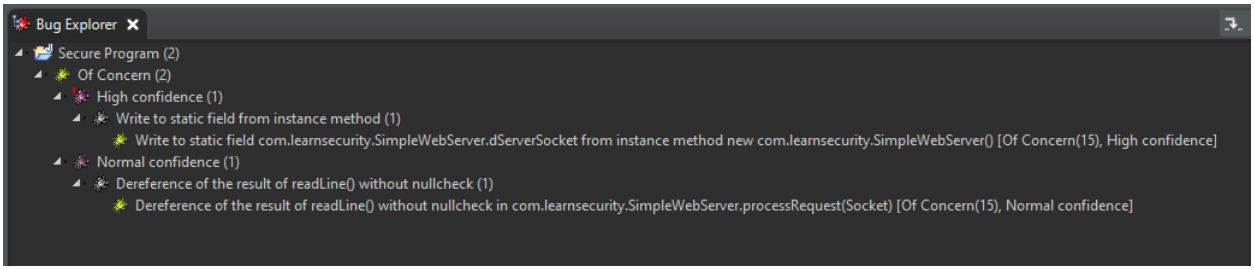
File /Secure Program/src/com/learnsecurity/SimpleWebServer.java (at 20/04/2020 20:30)



39 items

Resource	Date	Description
SimpleWebServer.java	40 minutes ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	40 minutes ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	40 minutes ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	40 minutes ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	40 minutes ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	40 minutes ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	40 minutes ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	40 minutes ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	40 minutes ago	Remove this use of "java.io.FileReader"
SimpleWebServer.java	40 minutes ago	Remove this use of constructor "FileReader(String)"
SimpleWebServer.java	40 minutes ago	Remove this use of constructor "InputStreamReader(InputStream)"
SimpleWebServer.java	40 minutes ago	Remove this use of constructor "OutputStreamWriter(OutputStream)"
SimpleWebServer.java	40 minutes ago	Replace all tab characters in this file by sequences of white-spaces.
SimpleWebServer.java	40 minutes ago	Use 'java.io.Writer' here; it is a more general type than 'OutputStreamWriter'.
SimpleWebServer.java	40 minutes ago	Either log or rethrow this exception.
SimpleWebServer.java	40 minutes ago	Explicitly import the specific classes needed.
SimpleWebServer.java	40 minutes ago	Explicitly import the specific classes needed.
SimpleWebServer.java	40 minutes ago	Explicitly import the specific classes needed.
SimpleWebServer.java	40 minutes ago	Missing curly brace.
SimpleWebServer.java	40 minutes ago	Missing curly brace.
SimpleWebServer.java	40 minutes ago	Add or update the header of this file.
SimpleWebServer.java	40 minutes ago	Remove this throws clause.
SimpleWebServer.java	1 hour ago	Move the array designator from the variable to the type.
SimpleWebServer.java	1 hour ago	Define and throw a dedicated exception instead of using a generic one.
SimpleWebServer.java	1 hour ago	Define and throw a dedicated exception instead of using a generic one.
SimpleWebServer.java	1 hour ago	Remove this assignment of "dServerSocket".
SimpleWebServer.java	1 hour ago	Replace the synchronized class "StringBuffer" by an unsynchronized one such as "StringBuilder".
SimpleWebServer.java	1 hour ago	Add an end condition to this loop.
SimpleWebServer.java	1 hour ago	Use try-with-resources or close this "FileReader" in a "finally" clause.

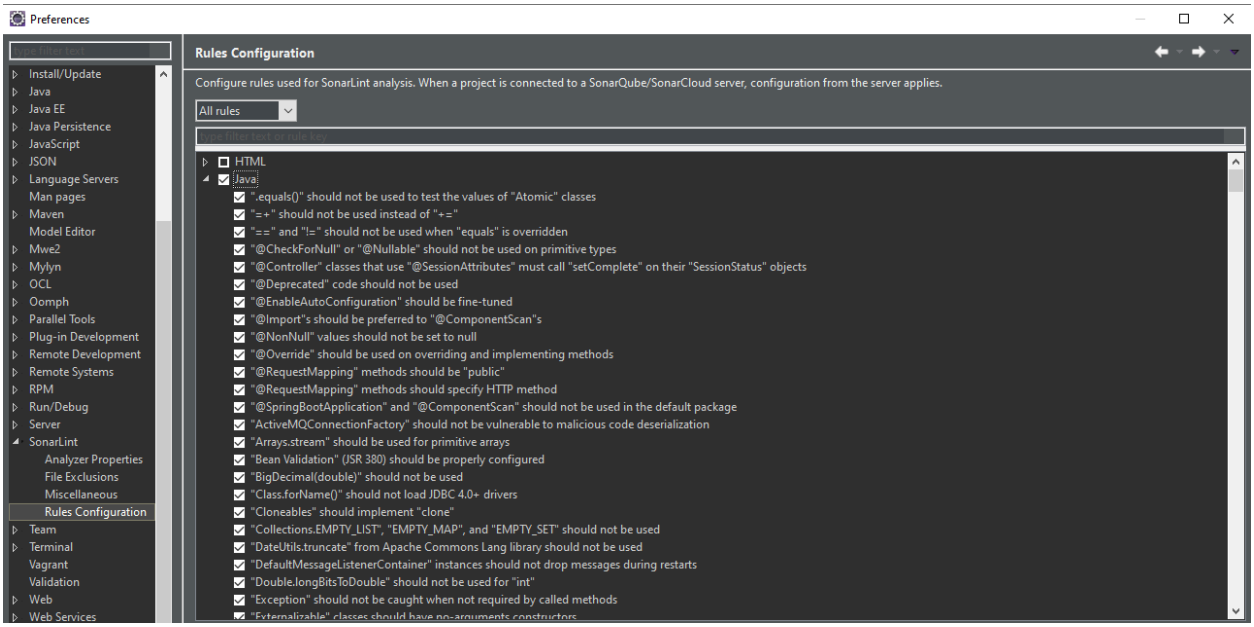
File /Secure Program/src/com/learnsecurity/SimpleWebServer.java (at 20/04/2020 20:30)



b) Security rules enabled:

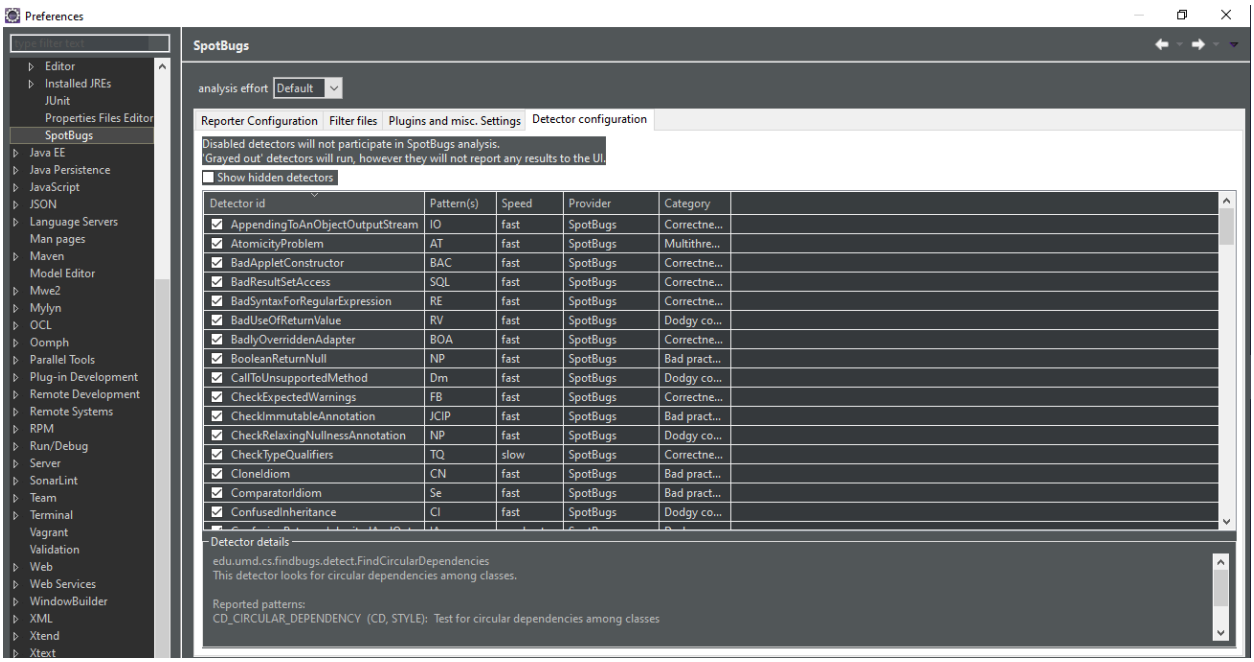
SonarLint:

For SonarLint I have enabled all the security rules under the preference of Eclipse. I have checked all the rules for Java since we are using java code for static code analysis.



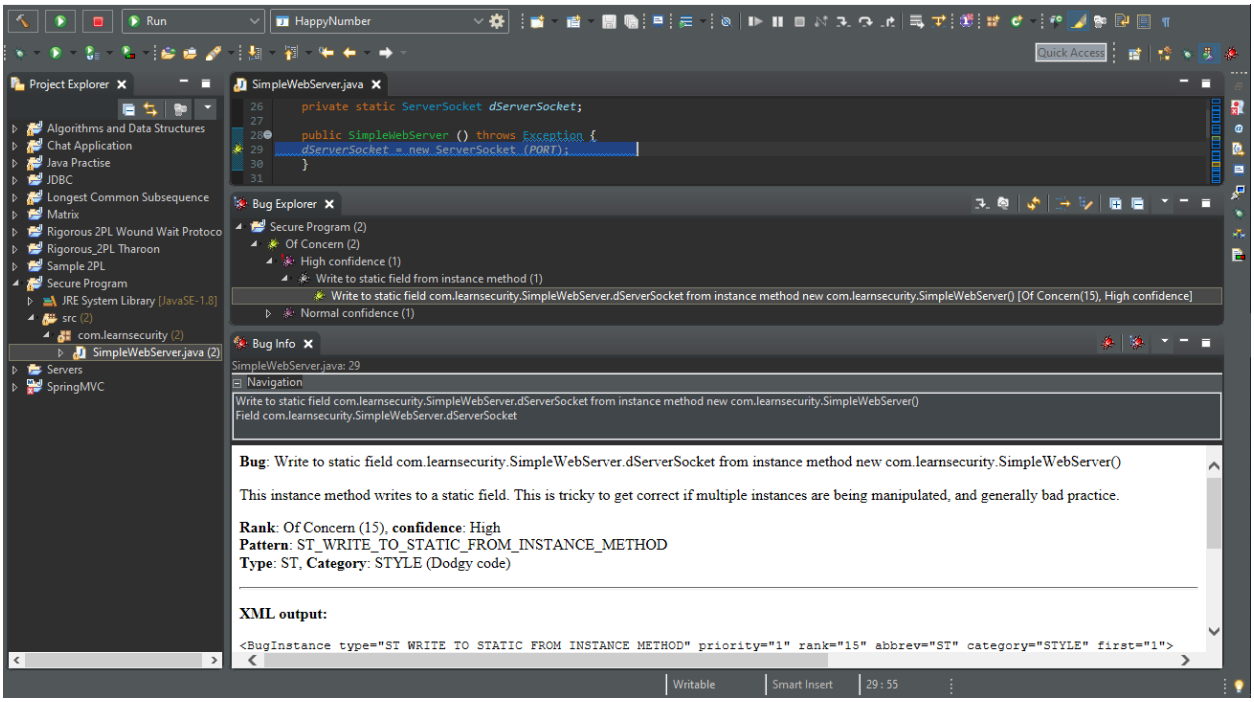
SpotBugs:

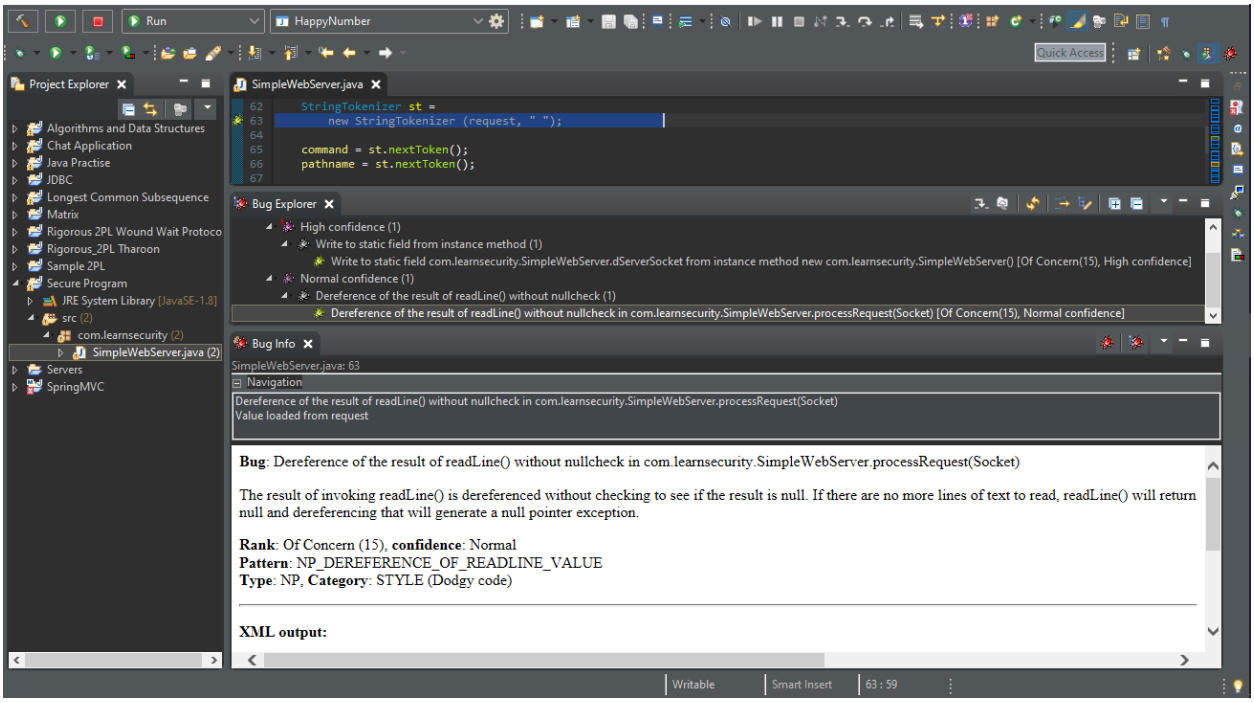
For SpotBugs also I have enabled all the security rules for analyzing the code.

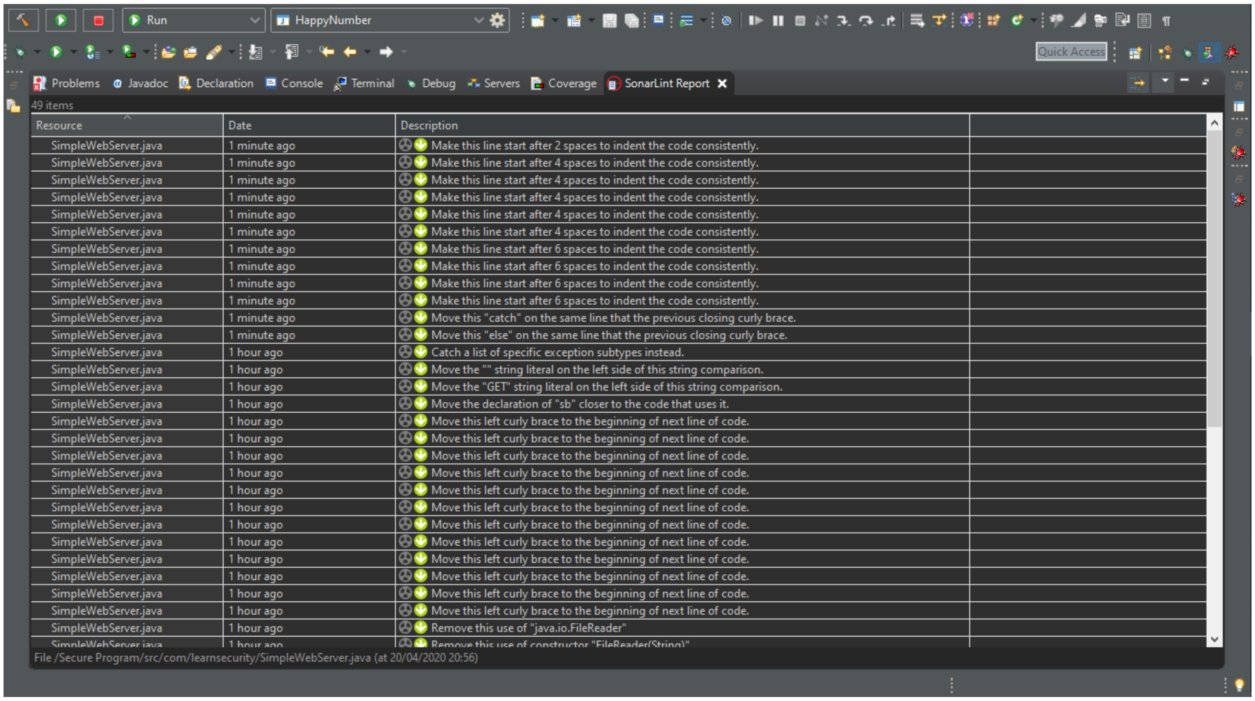


Turned on most aggressive mode in tool for finding defects:

I have turned on the most aggressive mode for both tools by enabling all the security rules. I have attached the snapshots of the test result from both the tools. I was able see that both the tools reported all the severe bugs in the program.



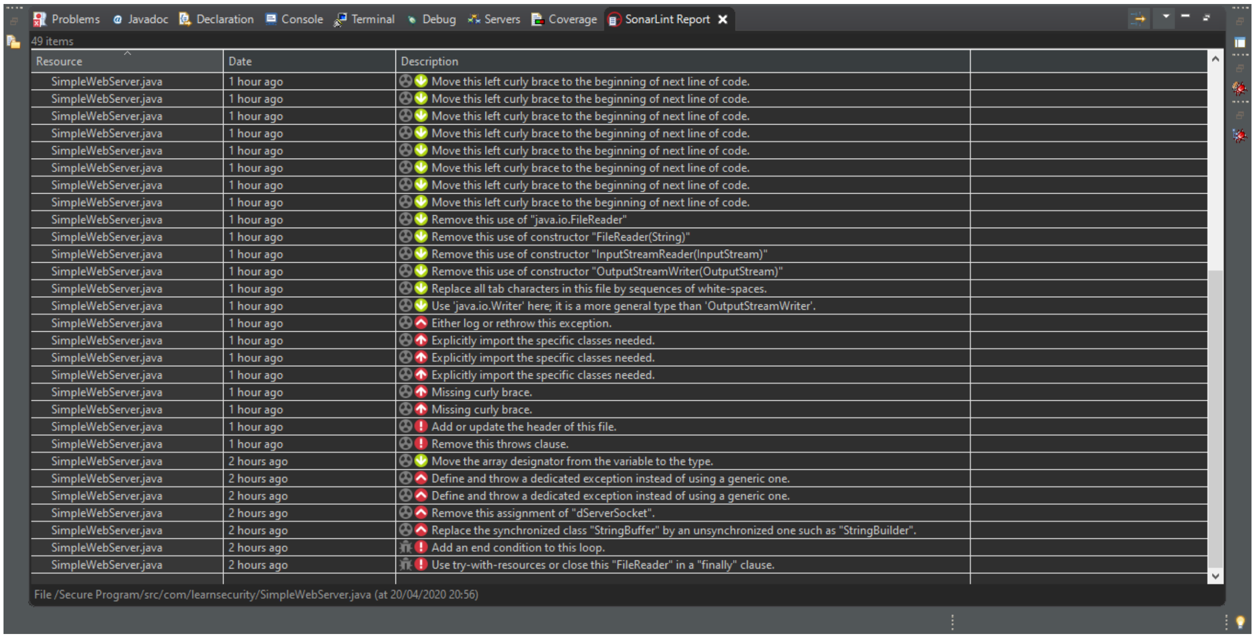




49 items

Resource	Date	Description
SimpleWebServer.java	1 minute ago	Make this line start after 2 spaces to indent the code consistently.
SimpleWebServer.java	1 minute ago	Make this line start after 4 spaces to indent the code consistently.
SimpleWebServer.java	1 minute ago	Make this line start after 4 spaces to indent the code consistently.
SimpleWebServer.java	1 minute ago	Make this line start after 4 spaces to indent the code consistently.
SimpleWebServer.java	1 minute ago	Make this line start after 4 spaces to indent the code consistently.
SimpleWebServer.java	1 minute ago	Make this line start after 4 spaces to indent the code consistently.
SimpleWebServer.java	1 minute ago	Make this line start after 6 spaces to indent the code consistently.
SimpleWebServer.java	1 minute ago	Make this line start after 6 spaces to indent the code consistently.
SimpleWebServer.java	1 minute ago	Make this line start after 6 spaces to indent the code consistently.
SimpleWebServer.java	1 minute ago	Make this line start after 6 spaces to indent the code consistently.
SimpleWebServer.java	1 minute ago	Move this "catch" on the same line that the previous closing curly brace.
SimpleWebServer.java	1 minute ago	Move this "else" on the same line that the previous closing curly brace.
SimpleWebServer.java	1 hour ago	Catch a list of specific exception subtypes instead.
SimpleWebServer.java	1 hour ago	Move the "" string literal on the left side of this string comparison.
SimpleWebServer.java	1 hour ago	Move the "GET" string literal on the left side of this string comparison.
SimpleWebServer.java	1 hour ago	Move the declaration of "sb" closer to the code that uses it.
SimpleWebServer.java	1 hour ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	1 hour ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	1 hour ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	1 hour ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	1 hour ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	1 hour ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	1 hour ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	1 hour ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	1 hour ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	1 hour ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	1 hour ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	1 hour ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	1 hour ago	Remove this use of "java.io.FileReader"
SimpleWebServer.java	1 hour ago	Remove this use of constructor "FileReader(String)"

File /Secure Program/src/com/learnsecurity/SimpleWebServer.java (at 20/04/2020 20:56)



49 items

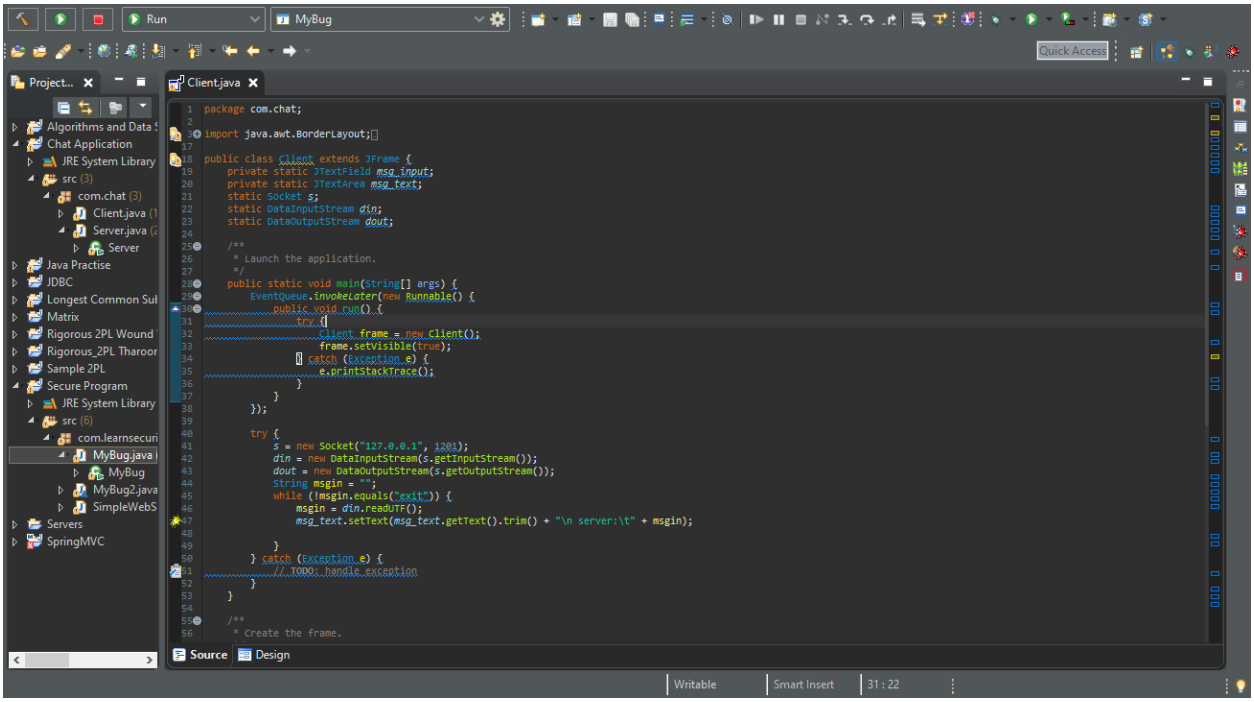
Resource	Date	Description
SimpleWebServer.java	1 hour ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	1 hour ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	1 hour ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	1 hour ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	1 hour ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	1 hour ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	1 hour ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	1 hour ago	Move this left curly brace to the beginning of next line of code.
SimpleWebServer.java	1 hour ago	Remove this use of "java.io.FileReader"
SimpleWebServer.java	1 hour ago	Remove this use of constructor "FileReader(String)"
SimpleWebServer.java	1 hour ago	Remove this use of constructor "InputStreamReader(InputStream)"
SimpleWebServer.java	1 hour ago	Remove this use of constructor "OutputStreamWriter(OutputStream)"
SimpleWebServer.java	1 hour ago	Replace all tab characters in this file by sequences of white-spaces.
SimpleWebServer.java	1 hour ago	Use 'java.io.Writer' here; it is a more general type than 'OutputStreamWriter'.
SimpleWebServer.java	1 hour ago	Either log or rethrow this exception.
SimpleWebServer.java	1 hour ago	Explicitly import the specific classes needed.
SimpleWebServer.java	1 hour ago	Explicitly import the specific classes needed.
SimpleWebServer.java	1 hour ago	Explicitly import the specific classes needed.
SimpleWebServer.java	1 hour ago	Missing curly brace.
SimpleWebServer.java	1 hour ago	Missing curly brace.
SimpleWebServer.java	1 hour ago	Add or update the header of this file.
SimpleWebServer.java	1 hour ago	Remove this throws clause.
SimpleWebServer.java	2 hours ago	Move the array designator from the variable to the type.
SimpleWebServer.java	2 hours ago	Define and throw a dedicated exception instead of using a generic one.
SimpleWebServer.java	2 hours ago	Define and throw a dedicated exception instead of using a generic one.
SimpleWebServer.java	2 hours ago	Remove this assignment of "dServerSocket".
SimpleWebServer.java	2 hours ago	Replace the synchronized class "StringBuffer" by an unsynchronized one such as "StringBuilder".
SimpleWebServer.java	2 hours ago	Add an end condition to this loop.
SimpleWebServer.java	2 hours ago	Use try-with-resources or close this "FileReader" in a "finally" clause.

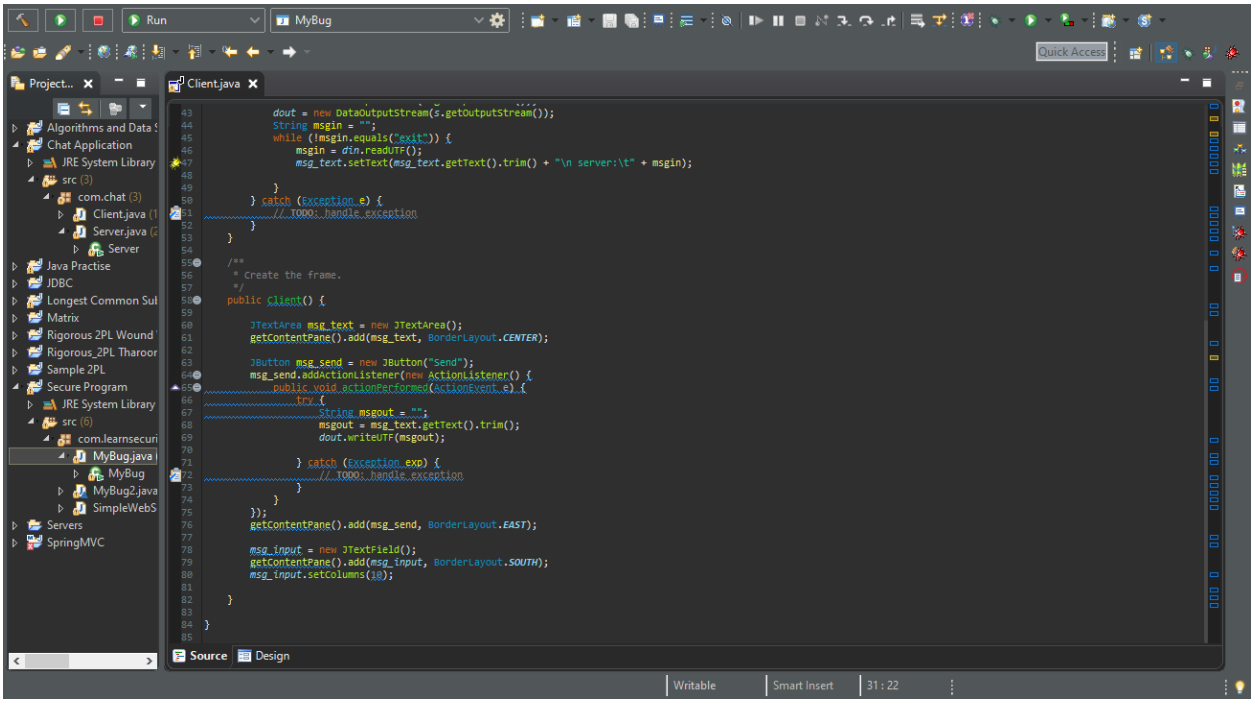
File /Secure Program/src/com/learnsecurity/SimpleWebServer.java (at 20/04/2020 20:56)

Part 2:

For the part 2, I had a java program for chat application that was implemented using sockets and Java GUI. The program basically has 2 parts, one is the server part and the other is the client part. I did a static code analysis on the client.java using the tools SpotBugs and SonarLint.

I have attached the snapshot of the code and I have also provided the source code in the submission file.





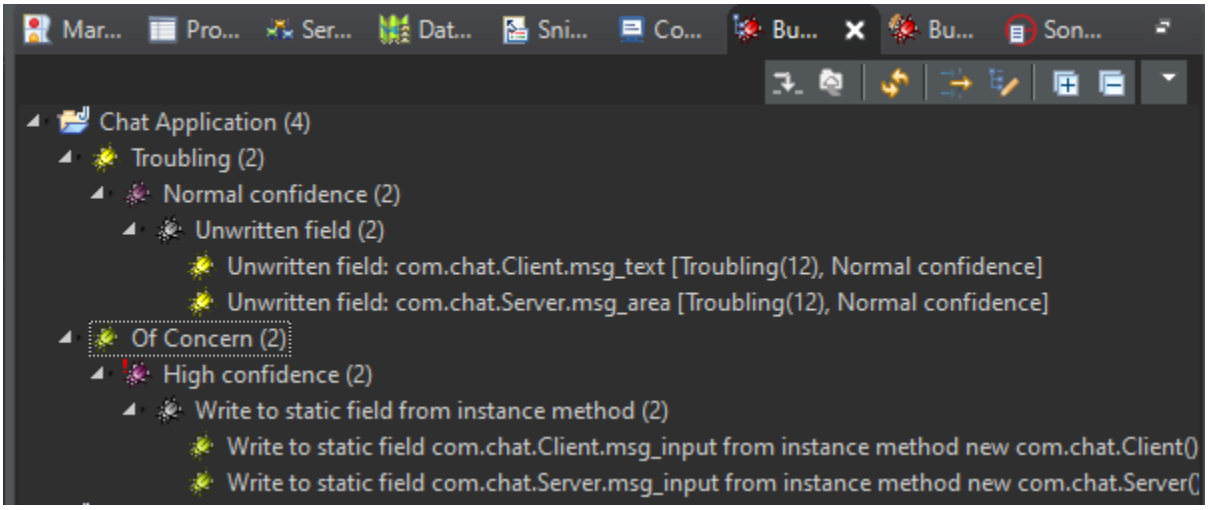
Manual Analysis:

When I did a manual static analysis on the client.java program I was able to see that I did not close the socket connection after establishing a socket connection at the starting of the program. I failed to close the socket connection in the program which can lead many severe threats to the application. One major threat is that if the socket is not closed, the connection keeps accepting requests and there are many possibilities that an attacker can use this connection and inject malicious code into the application. Failing to close the connection will also lead to unwanted leak of resources and file descriptors.

Results:

I have attached the snapshots of the report generated by both tools. I have also included the report generated from both tools. The attached files are SonarLint part2.xslx and SpotBugsReport Part2.xml.

```java
            while (!msgin.equals("exit")) {
                msgin = din.readUTF();
                msg_text.setText(msg_text.getText().trim() + "\n server:\t" + msgin);

            }
        } catch (Exception e) {
            // TODO: handle exception

        }
    }

    /**
     * Create the frame.
     */
    public Client() {

        JTextArea msg_text = new JTextArea();
        getContentPane().add(msg_text, BorderLayout.CENTER);

        JButton msg_send = new JButton("Send");
        msg_send.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                try {
                    String msgout = "";
                    msgout = msg_text.getText().trim();
                    dout.writeUTF(msgout);

                } catch (Exception exp) {
                    // TODO: handle exception
                }
            }
        });
        getContentPane().add(msg_send, BorderLayout.EAST);

        msg_input = new JTextField();
        getContentPane().add(msg_input, BorderLayout.SOUTH);
        msg_input.setColumns(10);

    }


}
```



*Client.java

```
                // TODO: handle exception
            }
        }
    });
    getContentPane().add(msg_send, BorderLayout.EAST);

    msg_input = new JTextField();
    getContentPane().add(msg_input, BorderLayout.SOUTH);
    msg_input.setColumns(10);

}
```

Source | Design

Markers | Properties | Servers | Data Source Explorer | Snippets | Console | Bug Explorer | Bug Info | SonarLint Report

Client.java: 78

Navigation

Write to static field com.chat.Client.msg_input from instance method new com.chat.Client()
Field com.chat.Client.msg_input

Bug: Write to static field com.chat.Client.msg_input from instance method new com.chat.Client()

This instance method writes to a static field. This is tricky to get correct if multiple instances are being manipulated, and generally bad practice.

Rank: Of Concern (15), **confidence**: High
Pattern: ST_WRITE_TO_STATIC_FROM_INSTANCE_METHOD
Type: ST, **Category**: STYLE (Dodgy code)

XML output:

```
<BugInstance type="ST_WRITE_TO_STATIC_FROM_INSTANCE_METHOD" priority="1" rank="15" abbrev="ST" category="STYLE" first="1">
  <Class classname="com.chat.Client">
    <SourceLine classname="com.chat.Client" sourcefile="Client.java" sourcepath="com/chat/Client.java"/>
  </Class>
```

Writable | Smart Insert | 80 : 36

*Client.java

```
        String msgin = "";
        while (!msgin.equals("exit")) {
            msgin = din.readUTF();
            msg_text.setText(msg_text.getText().trim() + "\n server:\t" + msgin);

        }
```

Source | Design

Markers | Properties | Servers | Data Source Explorer | Snippets | Console | Bug Explorer | Bug Info | SonarLint Report

Client.java: 47

Navigation

Unwritten field: com.chat.Client.msg_text
Field com.chat.Client.msg_text

Bug: Unwritten field: com.chat.Client.msg_text

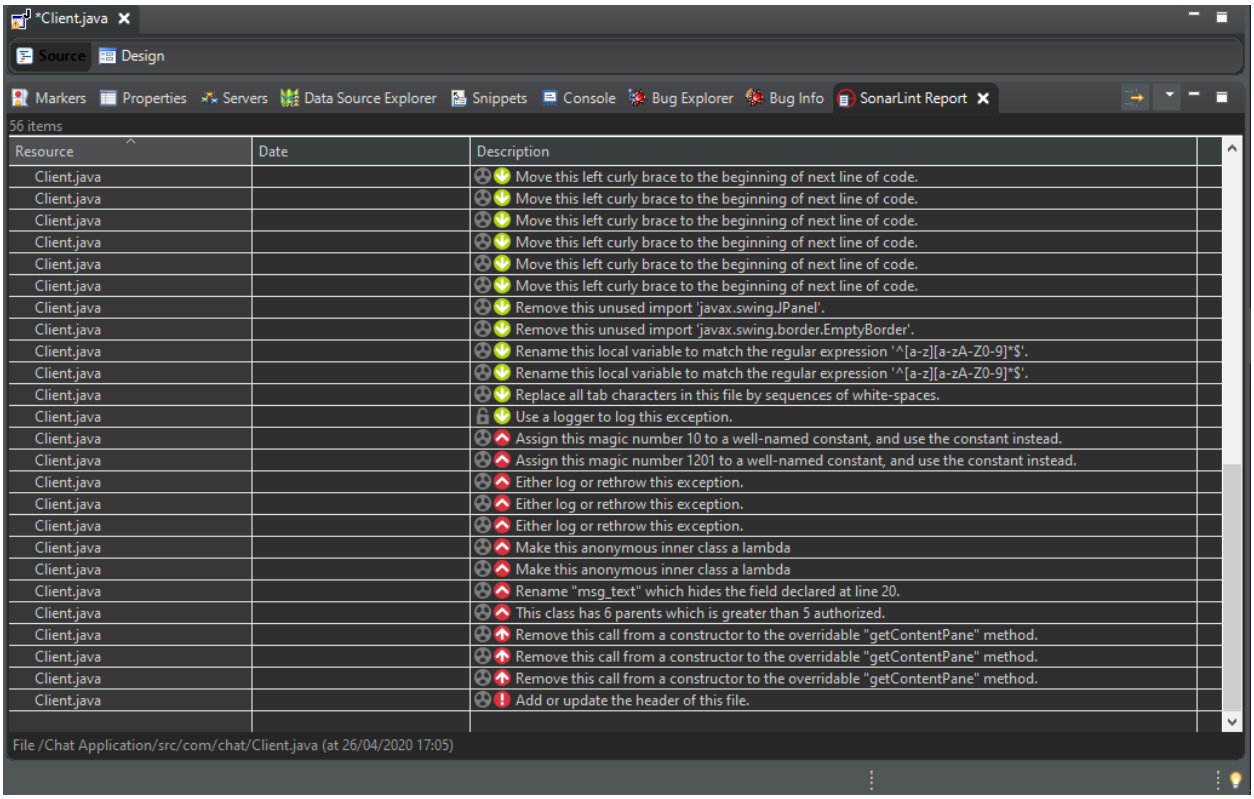
This field is never written. All reads of it will return the default value. Check for errors (should it have been initialized?), or remove it if it is useless.

Rank: Troubling (12), **confidence**: Normal
Pattern: UWF_UNWRITTEN_FIELD
Type: UwF, **Category**: CORRECTNESS (Correctness)

XML output:

```
<BugInstance type="UWF_UNWRITTEN_FIELD" priority="2" rank="12" abbrev="UwF" category="CORRECTNESS" first="1">
  <Class classname="com.chat.Client">
    <SourceLine classname="com.chat.Client" sourcefile="Client.java" sourcepath="com/chat/Client.java"/>
  </Class>
  <Field classname="com.chat.Client" name="msg_text" signature="Ljavax/swing/JTextArea;" isStatic="true">
    <SourceLine classname="com.chat.Client" sourcefile="Client.java" sourcepath="com/chat/Client.java"/>
  </Field>
  <SourceLine classname="com.chat.Client" start="47" end="47" startBytecode="70" endBytecode="70" sourcefile="Client.java" s
</BugInstance>
```

Writable | Smart Insert | 47 : 86



*Client.java

Source | Design

Markers | Properties | Servers | Data Source Explorer | Snippets | Console | Bug Explorer | Bug Info | SonarLint Report

56 items

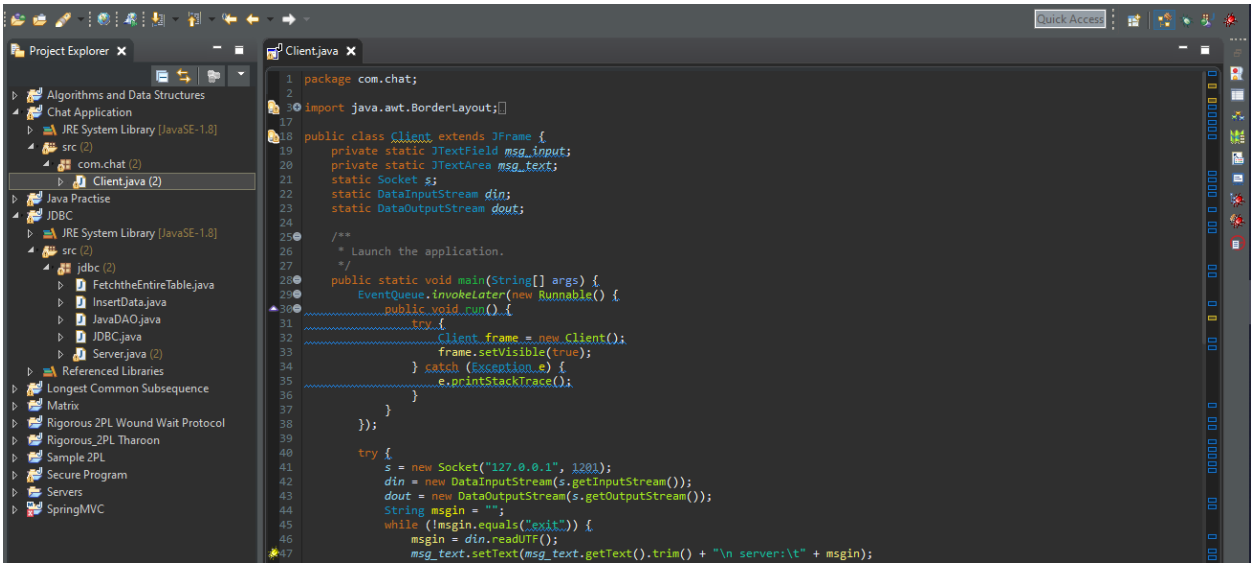
Resource	Date	Description
Client.java		Move this left curly brace to the beginning of next line of code.
Client.java		Move this left curly brace to the beginning of next line of code.
Client.java		Move this left curly brace to the beginning of next line of code.
Client.java		Move this left curly brace to the beginning of next line of code.
Client.java		Move this left curly brace to the beginning of next line of code.
Client.java		Move this left curly brace to the beginning of next line of code.
Client.java		Remove this unused import 'javax.swing.JPanel'.
Client.java		Remove this unused import 'javax.swing.border.EmptyBorder'.
Client.java		Rename this local variable to match the regular expression '^[a-z][a-zA-Z0-9]*$'.
Client.java		Rename this local variable to match the regular expression '^[a-z][a-zA-Z0-9]*$'.
Client.java		Replace all tab characters in this file by sequences of white-spaces.
Client.java		Use a logger to log this exception.
Client.java		Assign this magic number 10 to a well-named constant, and use the constant instead.
Client.java		Assign this magic number 1201 to a well-named constant, and use the constant instead.
Client.java		Either log or rethrow this exception.
Client.java		Either log or rethrow this exception.
Client.java		Either log or rethrow this exception.
Client.java		Make this anonymous inner class a lambda
Client.java		Make this anonymous inner class a lambda
Client.java		Rename "msg_text" which hides the field declared at line 20.
Client.java		This class has 6 parents which is greater than 5 authorized.
Client.java		Remove this call from a constructor to the overridable "getContentPane" method.
Client.java		Remove this call from a constructor to the overridable "getContentPane" method.
Client.java		Remove this call from a constructor to the overridable "getContentPane" method.
Client.java		Add or update the header of this file.

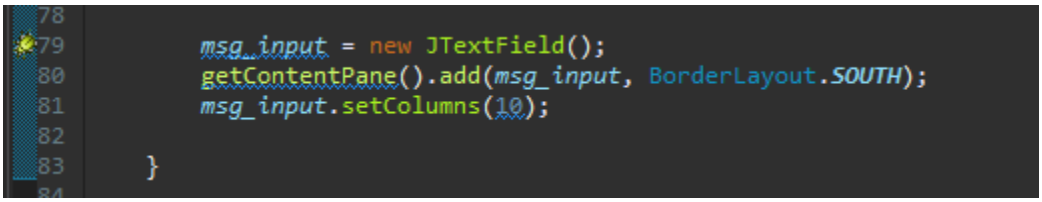
File /Chat Application/src/com/chat/Client.java (at 26/04/2020 17:05)

Fixes:

The bug found by the SpotBug tool on line 79 is a ST_WRITE_TO_STATIC_FROM_INSTANCE_METHOD. This is because msg_input is a static variable and I create a new object and assign it to msg_input each time when the Client class is invoked, that is the reason I am getting that bug in SpotBug. Since it is static whenever an object is assigned to msg_input the other objects referring to msg_input also gets changed and gets the new value that has been generated by the new object. The fix to this bug is to make the msg_input non static member. So that whenever a new object is created the msg_input is not overwritten and each object has its own instance of the variable msg_input.

Before changing msg_input to non static:





```
        msg_input = new JTextField();
        getContentPane().add(msg_input, BorderLayout.SOUTH);
        msg_input.setColumns(10);

    }
```

After removing static in msg_input:

```java
import java.awt.BorderLayout;

public class Client extends JFrame {
    private JTextField msg_input;
    private static JTextArea msg_text;
    static Socket s;
    static DataInputStream din;
    static DataOutputStream dout;

    /**
     * Launch the application.
     */
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    Client frame = new Client();
                    frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });

        try {
            s = new Socket("127.0.0.1", 1201);
            din = new DataInputStream(s.getInputStream());
            dout = new DataOutputStream(s.getOutputStream());
            String msgin = "";
            while (!msgin.equals("exit")) {
                msgin = din.readUTF();
                msg_text.setText(msg_text.getText().trim() + "\n server:\t" + msgin);

            }
        } catch (Exception e) {
```

```java

        msg_input = new JTextField();
        getContentPane().add(msg_input, BorderLayout.SOUTH);
        msg_input.setColumns(10);

    }


}
```

The same bug has been reported by SonarLint also:

Resource	Date	Description
Client.java	4 hours ago	Rename this field "msg_input" to match the regular expression '^[a-z][a-zA-Z0-9]*$'.
Client.java	4 hours ago	Remove this assignment of "msg_input".

There is an another bug that has been reported by SpotBug on line 47. The pattern of the bug is UWF_UNWRITTEN_FIELD. msg_txt field is never written. This is because msg_txt is an instance of the class JTextArea and that we are calling a predefined method using the msg_txt. This bug cannot be fixed and that we are assigning any values and that it is only an instance of the class JTextArea and calling a method a that has been defined in it. And hence this bug cannot be fixed and there cannot be any potential threat for the application. Application will not work as expected upon removal of msg_txt variable.