Fortify Audit Workbench

Developer Workbook

fastester2.0-REST_07262019

Table of Contents

Executive Summary
Project Description
Issue Breakdown by Fortify Categories
Results Outline

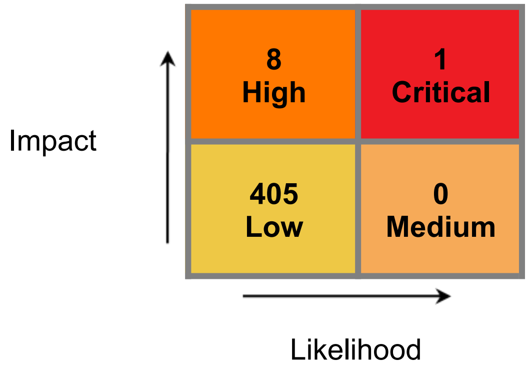
Executive Summary

This workbook is intended to provide all necessary details and information for a developer to understand and remediate the different issues discovered during the fastester2.0-REST_07262019 project audit. The information contained in this workbook is targeted at project managers and developers.

This section provides an overview of the issues uncovered during analysis.

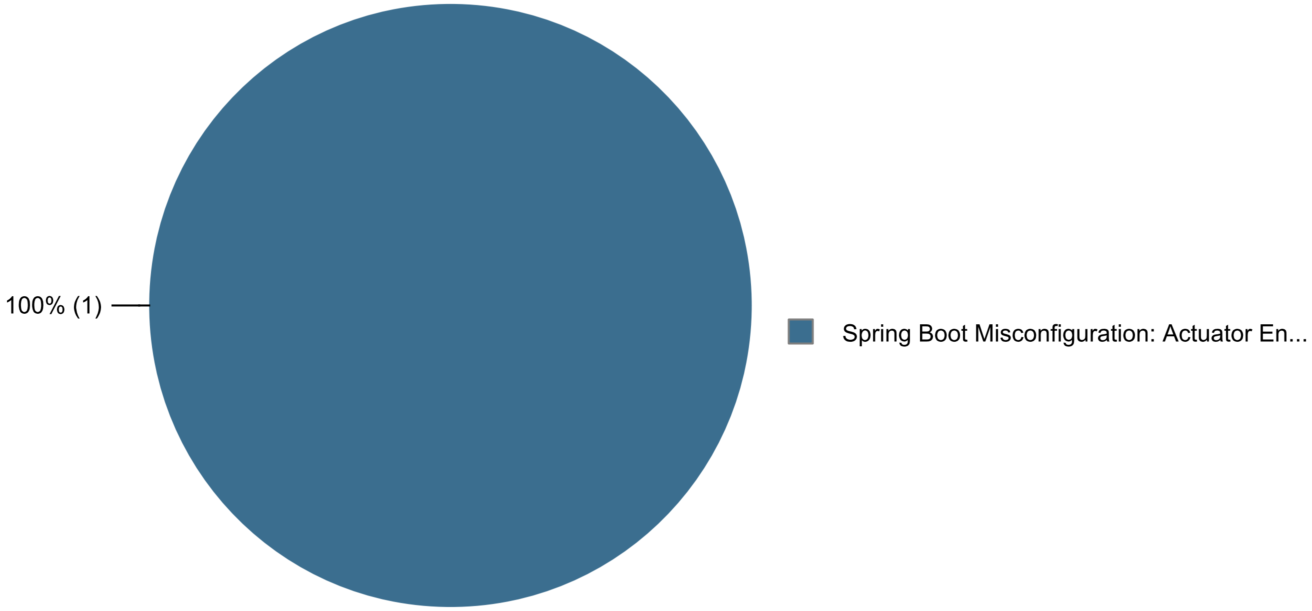
Project Name:	fastester2.0-REST_0726
Project Version:	
SCA:	Results Present
WebInspect:	Results Not Present
WebInspect Agent:	Results Not Present
Other:	Results Not Present

Issues by Priority



8 High	1 Critical
405 Low	0 Medium

Top Ten Critical Categories



Project Description

This section provides an overview of the Fortify scan engines used for this project, as well as the project meta-information.

SCA

Date of Last Analysis:	Jul 26, 2019, 3:30 PM	**Engine Version:**	18.20.1071
Host Name:	CTSINMBPPROS1	**Certification:**	VALID
Number of Files:	173	**Lines of Code:**	14,450

Issue Breakdown by Fortify Categories

The following table depicts a summary of all issues grouped vertically by Fortify Category. For each category, the total number of issues is shown by Fortify Priority Order, including information about the number of audited issues.

Category	Fortify Priority (audited/total)				Total Issues
	Critical	High	Medium	Low	
Build Misconfiguration: External Maven Dependency Repository	0	0	0	0 / 1	0 / 1
Code Correctness: Byte Array to String Conversion	0	0	0	0 / 6	0 / 6
Code Correctness: Class Does Not Implement equals	0	0	0	0 / 4	0 / 4
Code Correctness: Erroneous String Compare	0	0	0	0 / 2	0 / 2
Code Correctness: Hidden Method	0	0	0	0 / 1	0 / 1
Code Correctness: null Argument to equals()	0	0	0	0 / 1	0 / 1
Code Correctness: toString on Array	0	0	0	0 / 1	0 / 1
Dead Code: Expression is Always false	0	0	0	0 / 3	0 / 3
Dead Code: Unused Field	0	0	0	0 / 40	0 / 40
Dead Code: Unused Method	0	0	0	0 / 7	0 / 7
Denial of Service	0	0	0	0 / 3	0 / 3
J2EE Bad Practices: Leftover Debug Code	0	0	0	0 / 2	0 / 2
Null Dereference	0	0 / 5	0	0	0 / 5
Password Management: Null Password	0	0	0	0 / 1	0 / 1
Password Management: Password in Comment	0	0	0	0 / 53	0 / 53
Poor Error Handling: Empty Catch Block	0	0	0	0 / 1	0 / 1
Poor Error Handling: Overly Broad Catch	0	0	0	0 / 51	0 / 51
Poor Error Handling: Overly Broad Throws	0	0	0	0 / 57	0 / 57
Poor Error Handling: Program Catches NullPointerException	0	0	0	0 / 1	0 / 1
Poor Error Handling: Throw Inside Finally	0	0	0	0 / 3	0 / 3
Poor Logging Practice: Use of a System Output Stream	0	0	0	0 / 3	0 / 3
Poor Style: Confusing Naming	0	0	0	0 / 1	0 / 1
Poor Style: Redundant Initialization	0	0	0	0 / 1	0 / 1
Poor Style: Value Never Read	0	0	0	0 / 78	0 / 78
Privacy Violation: Heap Inspection	0	0 / 1	0	0	0 / 1
Redundant Null Check	0	0	0	0 / 5	0 / 5
Spring Boot Misconfiguration: Actuator Endpoint Security Disabled	0 / 1	0	0	0	0 / 1
Spring Boot Misconfiguration: DevTools Enabled	0	0 / 1	0	0	0 / 1
System Information Leak	0	0	0	0 / 75	0 / 75
System Information Leak: Spring Boot Actuators Enabled	0	0	0	0 / 1	0 / 1
Unchecked Return Value	0	0	0	0 / 3	0 / 3
Unreleased Resource: Streams	0	0 / 1	0	0	0 / 1

Results Outline

Build Misconfiguration: External Maven Dependency Repository (1 issue)

Abstract

This maven build script relies on external sources, which could allow an attacker to insert malicious code into the final product or to take control of the build machine.

Explanation

Several tools exist within the Java development world to aid in dependency management: both Apache Ant and Apache Maven build systems include functionality specifically designed to help manage dependencies and Apache Ivy is developed explicitly as a dependency manager. Although there are differences in their behavior, these tools share the common functionality that they automatically download external dependencies specified in the build process at build time. This makes it much easier for developer B to build software in the same manner as developer A. Developers just store dependency information in the build file, which means that each developer and build engineer has a consistent way to obtain dependencies, compile the code, and deploy without the dependency management hassles involved in manual dependency management. The following examples illustrate how Ivy, Ant and Maven can be used to manage external dependencies as part of a build process. Under Maven, instead of listing explicit URLs from which to retrieve the dependencies, developers specify the dependency names and versions and Maven relies on its underlying configuration to identify the server(s) from which to retrieve the dependencies. For commonly used components this saves the developer from having to researching dependency locations. **Example 1:** The following except from a Maven pom.xml file shows how a developer can specify multiple external dependencies using their name and version:

```
<dependencies>
  <dependency>
    <groupId>commons-logging</groupId>
    <artifactId>commons-logging</artifactId>
    <version>1.1</version>
  </dependency>
  <dependency>
    <groupId>javax.jms</groupId>
    <artifactId>jms</artifactId>
    <version>1.1</version>
  </dependency>
  ...
</dependencies>
```

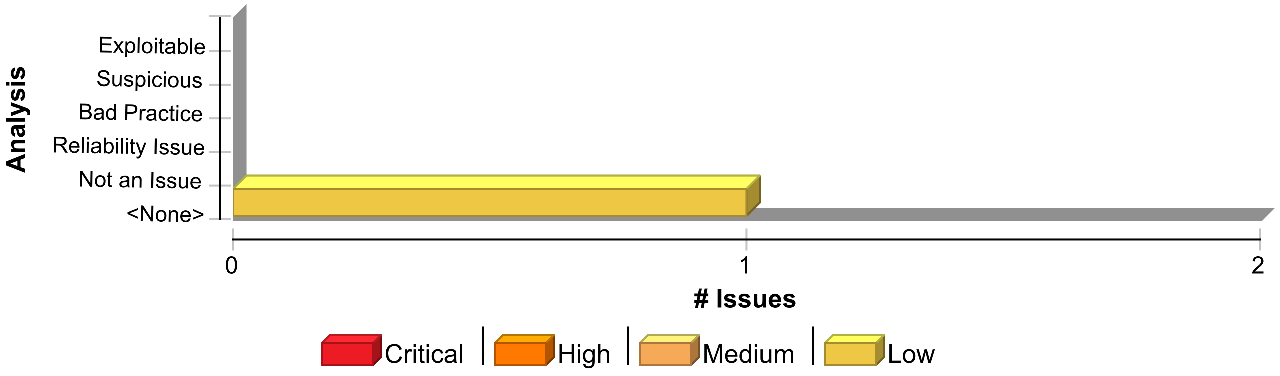
Two distinct types of attack scenarios affect these systems: An attacker could either compromise the server hosting the dependency or compromise the DNS server the build machine uses to redirect requests for hostname of the server hosting the dependency to a machine controlled by the attacker. Both scenarios result in the attacker gaining the ability to inject a malicious version of a dependency into a build running on an otherwise uncompromised machine. Regardless of the attack vector used to deliver the Trojan dependency, these scenarios share the common element that the build system blindly accepts the malicious binary and includes it in the build. Because the build system has no recourse for rejecting the malicious binary and existing security mechanisms, such as code review, typically focus on internally-developed code rather than external dependencies, this type of attack has a strong potential to go unnoticed as it spreads through the development environment and potentially into production. Although there is some risk of a compromised dependency being introduced into a manual build process, by the tendency of automated build systems to retrieve the dependency from an external source each time the build system is run in a new environment greatly increases the window of opportunity for an attacker. An attacker need only compromise the dependency server or the DNS server during one of the many times the dependency is retrieved in order to compromise the machine on which the build is occurring.

Recommendation

The simplest solution is to refrain from adopting automated dependency management systems altogether. Managing dependencies manually eliminates the potential for unexpected behavior caused by the build system. Obviously, the an attacker could still mount one of the attacks described above to coincide with the manual retrieval of a dependency, but limiting the frequency with which the dependency must be retrieved significantly reduces the window of opportunity for an attacker. Finally, this solution forces the development organization to rely on what is ostensibly an antiquated build system. A system based on manual dependency management is often more difficult to use and maintain, and may be unacceptable in some software development environments. The second solution is a hybrid of the traditional manual dependency management approach and the fully automated solution that is popular today. The biggest advantage of the manual build process is the decreased window of attack, which can be achieved in a semi-automated system by replicating external dependency servers internally. Any build system that requires an external dependency can then point to the internal server using a hard-coded internal IP address to bypass the risk of DNS-based attacks. As new dependencies are added and new versions released, they can be downloaded once and included on the internal repository. This solution reduces the attack opportunities and allows the organization leverage existing internal network security infrastructure. To implement this solution using Maven, a project should have the IP address for an internal repository hard coded the pom.xml. Specifying the IP address in the pom.xml ensures the internal repository will be used by the corresponding build, but is tied to a specific project. Alternatively, the IP address can be specified in settings.xml, which makes the configuration easier to share across multiple projects. **Example 2:** The following Maven pom.xml demonstrates the use of an explicit internal IP address (the entries can also be used in settings.xml):

```
<project>
  ...
  <repositories>
    <repository>
      <releases>
        <enabled>true</enabled>
        <updatePolicy>always</updatePolicy>
        <checksumPolicy>warn</checksumPolicy>
      </releases>
      <snapshots>
        <enabled>true</enabled>
        <updatePolicy>never</updatePolicy>
        <checksumPolicy>fail</checksumPolicy>
      </snapshots>
      <id>central</id>
      <name>Internal Repository</name>
      <url>http://172.16.1.13/maven2</url>
      <layout>default</layout>
    </repository>
  </repositories>
  <pluginRepositories>
    ...
  </pluginRepositories>
  ...
</project>
```

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Build Misconfiguration: External Maven Dependency Repository	1	0	0	1
Total	**1**	**0**	**0**	**1**

Build Misconfiguration: External Maven Dependency Repository — Low

Package: <none>

pom.xml, line 189 (Build Misconfiguration: External Maven Dependency Repository) — Low

Issue Details

Kingdom: Environment
Scan Engine: SCA (Configuration)

Sink Details

Sink: //project/repositories
File: pom.xml:189
Taint Flags:

```
186  </dependency>
187
188  </dependencies>
189  <repositories>
190  <repository>
191  <id>repository.springsource.milestone</id>
192  <name>SpringSource Milestone Repository</name>
```

Code Correctness: Byte Array to String Conversion (6 issues)

Abstract

Converting a byte array into a `String` may lead to data loss.

Explanation

When data from a byte array is converted into a `String`, it is unspecified what will happen to any data that is outside of the applicable character set. This can lead to data being lost, or a decrease in the level of security when binary data is needed to ensure proper security measures are followed. **Example 1:** The following code converts data into a String in order to create a hash.

```
  ...
  FileInputStream fis = new FileInputStream(myFile);
  byte[] byteArr = byte[BUFSIZE];
  ...
  int count = fis.read(byteArr);
  ...
  String fileString = new String(byteArr);
  String fileSHA256Hex = DigestUtils.sha256Hex(fileString);
  // use fileSHA256Hex to validate file
  ...
```

Assuming the size of the file is less than `BUFSIZE`, this works fine as long as the information in `myFile` is encoded the same as the default character set, however if it's using a different encoding, or is a binary file, it will lose information. This in turn will cause the resulting SHA hash to be less reliable, and could mean it's far easier to cause collisions, especially if any data outside of the default character set is represented by the same value, such as a question mark.

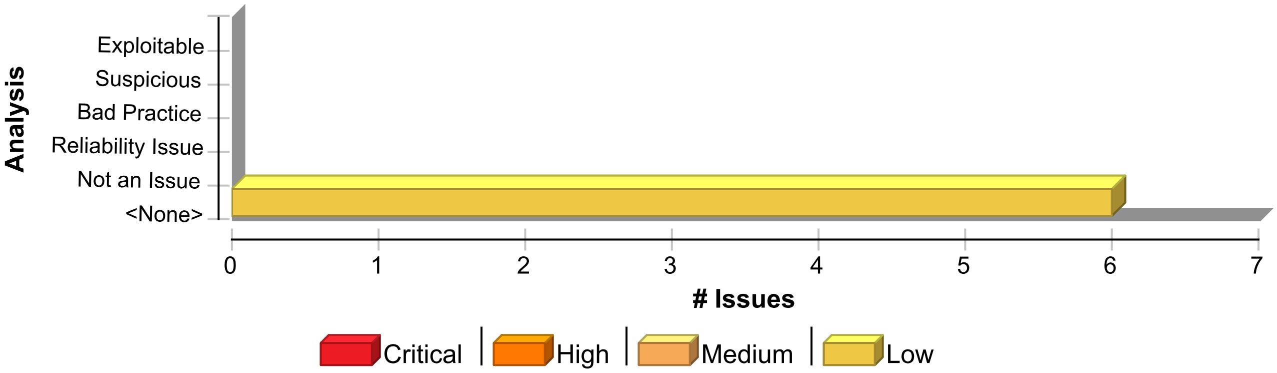
Recommendation

Generally speaking, a byte array potentially containing noncharacter data should never be converted into a `String` object as it may break functionality, but in some cases this can cause much larger security concerns. In a lot of cases there is no need to actually convert a byte array into a String, but if there is a specific reason to be able to create a `String` object from binary data, it must first be encoded in a way such that it will fit into the default character set. **Example 2:** The following uses a different variant of the API in Example 1 to prevent any validation problems.

```
  ...
  FileInputStream fis = new FileInputStream(myFile);
  byte[] byteArr = byte[BUFSIZE];
  ...
  int count = fis.read(byteArr);
  ...
  byte[] fileSHA256 = DigestUtils.sha256(byteArr);
  // use fileSHA256 to validate file, comparing hash byte-by-byte.
  ...
```

In this case, it is straightforward to rectify, since this API has overloaded variants including one that accepts a byte array, and this could be simplified even further by using another overloaded variant of `DigestUtils.sha256()` that accepts a `FileInputStream` object as its argument. Other scenarios may need careful consideration as to whether it's possible that the byte array could contain data outside of the character set, and further refactoring may be required.

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Code Correctness: Byte Array to String Conversion	6	0	0	6
Total	**6**	**0**	**0**	**6**

Code Correctness: Byte Array to String Conversion — Low

Package: com.cts.dcrowd.rest.app.security

src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java, line 431 (Code Correctness: Byte Array to String Conversion) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Semantic)

Sink Details

Sink: String()
Enclosing Method: decryptLoginCredential()
File: src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java:431
Taint Flags:

```
428  Cipher cipher = Cipher.getInstance("Blowfish/CBC/PKCS5Padding");
429  cipher.init(Cipher.DECRYPT_MODE, skeyspec);
430  byte[] decrypted = cipher.doFinal(blowFishCredentials.getBytes());
431  userCredential = new String(decrypted);
432  }
433  catch (Exception e) {
434  userCredential = null;
```

Package: com.cts.dcrowd.rest.app.util

src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java, line 101 (Code Correctness: Byte Array to String Conversion) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Semantic)

Sink Details

Code Correctness: Byte Array to String Conversion — Low

Package: com.cts.dcrowd.rest.app.util

src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java, line 101 (Code Correctness: Byte Array to String Conversion) — Low

Sink: String()
Enclosing Method: generateBlowFishDecryptValue()
File: src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java:101
Taint Flags:

```
98   Cipher cipher = Cipher.getInstance("Blowfish/CBC/PKCS5Padding");
99   cipher.init(Cipher.DECRYPT_MODE, skeyspec);
100  byte[] decrypted = cipher.doFinal(Base64.getDecoder().decode(encryptedValue));
101  strData = new String(decrypted);
102  System.out.println(strData);
103  } catch (Exception e) {
104  e.printStackTrace();
```

src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java, line 48 (Code Correctness: Byte Array to String Conversion) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Semantic)

Sink Details

Sink: String()
Enclosing Method: base64Encoder()
File: src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java:48
Taint Flags:

```
45   * @return the string
46   */
47   public static String base64Encoder(byte[] source){
48   return new String(Base64.getEncoder().encode(source));
49   }
50
51   /**
```

src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java, line 80 (Code Correctness: Byte Array to String Conversion) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Semantic)

Sink Details

Sink: String()
Enclosing Method: generateBlowFishEncryptValue()
File: src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java:80
Taint Flags:

Code Correctness: Byte Array to String Conversion — Low

Package: com.cts.dcrowd.rest.app.util

src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java, line 80 (Code Correctness: Byte Array to String Conversion) — Low

```
77  Cipher cipher=Cipher.getInstance("Blowfish/CBC/PKCS5Padding");
78  cipher.init(Cipher.ENCRYPT_MODE, skeyspec);
79  byte[] encrypted=cipher.doFinal(sourceValue.getBytes());
80  strData = new String(Base64.getEncoder().encode(encrypted));
81  } catch (Exception e) {
82  e.printStackTrace();
83  }
```

Package: com.cts.dcrowd.rest.app.utils

src/main/java/com/cts/dcrowd/rest/app/utils/Util.java, line 546 (Code Correctness: Byte Array to String Conversion) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Semantic)

Sink Details

Sink: String()
Enclosing Method: base64Encoder()
File: src/main/java/com/cts/dcrowd/rest/app/utils/Util.java:546
Taint Flags:

```
543   * @return the string
544   */
545   public static String base64Encoder(byte[] source){
546   return new String(Base64.getEncoder().encode(source));
547   }
548
549   /**
```

src/main/java/com/cts/dcrowd/rest/app/utils/Util.java, line 747 (Code Correctness: Byte Array to String Conversion) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Semantic)

Sink Details

Sink: String()
Enclosing Method: decryptLoginCredential()
File: src/main/java/com/cts/dcrowd/rest/app/utils/Util.java:747
Taint Flags:

```
744   Cipher cipher = Cipher.getInstance("Blowfish/CBC/PKCS5Padding");
745   cipher.init(Cipher.DECRYPT_MODE, skeyspec);
746   byte[] decrypted = cipher.doFinal(blowFishCredentials.getBytes());
```

Code Correctness: Byte Array to String Conversion

Low

Package: com.cts.dcrowd.rest.app.utils

src/main/java/com/cts/dcrowd/rest/app/utils/Util.java, line 747 (Code Correctness: Byte Array to String Conversion)

Low

```
747  userCredential = new String(decrypted);
748  }
749  catch (Exception e) {
750  userCredential = null;
```

Code Correctness: Class Does Not Implement equals (4 issues)

Abstract

The `equals()` method is called on an object that does not implement `equals()`.

Explanation

When comparing objects, developers usually want to compare properties of objects. However, calling `equals()` on a class (or any super class/interface) that does not explicitly implement `equals()` results in a call to the `equals()` method inherited from `java.lang.Object`. Instead of comparing object member fields or other properties, `Object.equals()` compares two object instances to see if they are the same. Although there are legitimate uses of `Object.equals()`, it is often an indication of buggy code. **Example 1:**

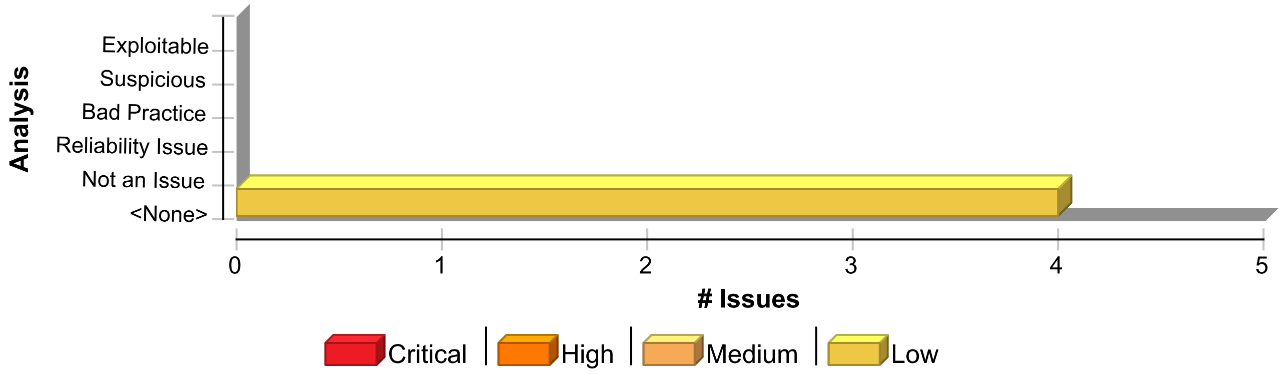
```
public class AccountGroup
{
    private int gid;

    public int getGid()
    {
        return gid;
    }

    public void setGid(int newGid)
    {
        gid = newGid;
    }
}
...
public class CompareGroup
{
    public boolean compareGroups(AccountGroup group1, AccountGroup group2)
    {
        return group1.equals(group2);   //equals() is not implemented in
AccountGroup
    }
}
```

Recommendation

Verify that the use of `Object.equals()` is really the method you intend to call. If not, implement an `equals()` method or use a different method for comparing objects. **Example 2:** The following code adds an `equals()` method to the example from the Explanation section.

```
public class AccountGroup
{
    private int gid;

    public int getGid()
    {
        return gid;
    }

    public void setGid(int newGid)
    {
        gid = newGid;
    }
    public boolean equals(Object o)
    {
        if (!(o instanceof AccountGroup))
            return false;
        AccountGroup other = (AccountGroup) o;
        return (gid == other.getGid());
    }
}
...
public class CompareGroup
{
    public static boolean compareGroups(AccountGroup group1, AccountGroup
group2)
    {
        return group1.equals(group2);
    }
}
```

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Code Correctness: Class Does Not Implement equals	4	0	0	4
Total	**4**	**0**	**0**	**4**

Code Correctness: Class Does Not Implement equals — Low

Package: .com.cts.dcrowd.rest.app.service.helper

src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java, line 371 (Code Correctness: Class Does Not Implement equals) — **Low**

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionCall: equals
Enclosing Method: accept()

Code Correctness: Class Does Not Implement equals — Low

Package: .com.cts.dcrowd.rest.app.service.helper

src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java, line 371 (Code Correctness: Class Does Not Implement equals) — Low

File: src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java:371
Taint Flags:

```
368  for(Object filterTagValue : (List<?>) filterTags.getValue()) {
369  try {
370  JSONObject filterObject = convertObjectToJson(filterTagValue);
371  if(filterObject.getString("state_code").equals(filterByState.getAddress().getState())){
372
if(filterObject.getString("country_code").equals(filterByState.getAddress().getCountry())) {
373  userListAfterStateFilter.add(filterByState);
374  } else {
```

src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java, line 372 (Code Correctness: Class Does Not Implement equals) — Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionCall: equals
Enclosing Method: accept()
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java:372
Taint Flags:

```
369  try {
370  JSONObject filterObject = convertObjectToJson(filterTagValue);
371  if(filterObject.getString("state_code").equals(filterByState.getAddress().getState())){
372
if(filterObject.getString("country_code").equals(filterByState.getAddress().getCountry())) {
373  userListAfterStateFilter.add(filterByState);
374  } else {
375  continue;
```

src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java, line 377 (Code Correctness: Class Does Not Implement equals) — Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionCall: equals
Enclosing Method: accept()

Code Correctness: Class Does Not Implement equals — Low

Package: .com.cts.dcrowd.rest.app.service.helper

src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java, line 377 (Code Correctness: Class Does Not Implement equals) — Low

File: src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java:377
Taint Flags:

```
374  } else {
375  continue;
376  }
377  } else if(!filterObject.getString("country_code").equals(filterByState.getAddress().getCountry())) {
378  userListAfterStateFilter.add(filterByState);
379  }
380  }catch(Exception e) {
```

Package: com.cts.dcrowd.rest.app.service.helper

src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java, line 341 (Code Correctness: Class Does Not Implement equals) — Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionCall: equals
Enclosing Method: orCriteriaFilterByCountryMatch()
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java:341
Taint Flags:

```
338  try {
339  //1. Convert object to JSON
340  JSONObject filterObject = convertObjectToJson(filterTagValue);
341  if(filterObject.getString("country_code").equals(addressObject.getCountry())){
342  matchFound = true;
343  }
344  } catch (JSONException | ParseException e) {
```

Code Correctness: Erroneous String Compare (2 issues)

Abstract

Strings should be compared with the `equals()` method, not == or `!=`.

Explanation

This program uses == or `!=` to compare two strings for equality, which compares two objects for equality, not their values. Chances are good that the two references will never be equal. **Example 1:** The following branch will never be taken.

```
  if (args[0] == STRING_CONSTANT) {
      logger.info("miracle");
  }
```

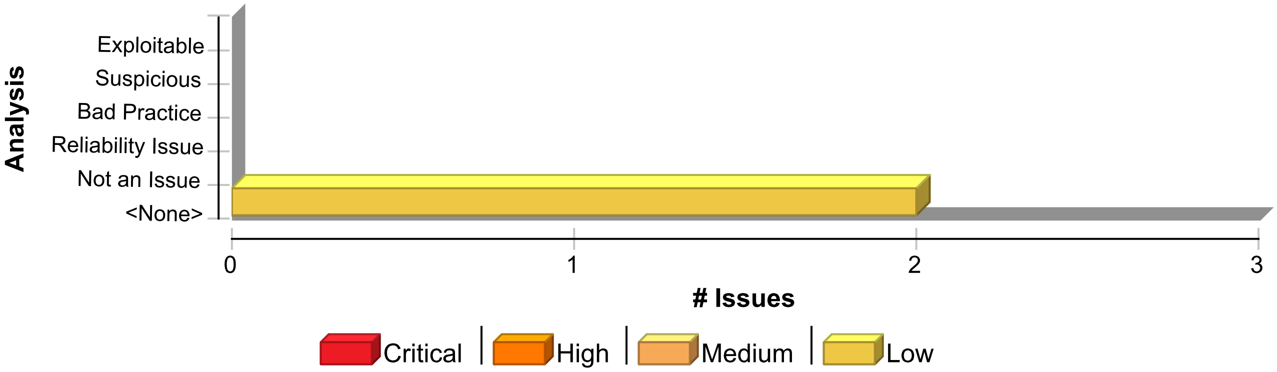
The == and `!=` operators will only behave as expected when they are used to compare strings contained in objects that are equal. The most common way for this to occur is for the strings to be interned, whereby the strings are added to a pool of objects maintained by the `String` class. Once a string is interned, all uses of that string will use the same object and equality operators will behave as expected. All string literals and string-valued constants are interned automatically. Other strings can be interned manually be calling `String.intern()`, which will return a canonical instance of the current string, creating one if necessary.

Recommendation

Use `equals()` to compare strings. **Example 2:** The code in Example 1 could be rewritten in the following way:

```
  if (STRING_CONSTANT.equals(args[0])) {
      logger.info("could happen");
  }
```

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Code Correctness: Erroneous String Compare	2	0	0	2
Total	**2**	**0**	**0**	**2**

Code Correctness: Erroneous String Compare — Low

Package: com.cts.dcrowd.rest.app.security

src/main/java/com/cts/dcrowd/rest/app/security/ApplicationAuthProvider.java, line 142 (Code Correctness: Erroneous String Compare) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Operation
Enclosing Method: authenticateEmployee()
File: src/main/java/com/cts/dcrowd/rest/app/security/ApplicationAuthProvider.java:142
Taint Flags:

```
139  username = authentication.getName();
140  password = authentication.getCredentials().toString();
141
142  if(password != null && password != ""){
143  requestFactory = new HttpComponentsClientHttpRequestFactory();
144  SSLContextBuilder sslContextBuilder = new SSLContextBuilder();
145  sslContextBuilder.loadTrustMaterial(null, new TrustStrategy() {
```

Package: com.cts.dcrowd.rest.app.service

src/main/java/com/cts/dcrowd/rest/app/service/CampaignService.java, line 628 (Code Correctness: Erroneous String Compare) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Operation
Enclosing Method: setExternalModLvl()
File: src/main/java/com/cts/dcrowd/rest/app/service/CampaignService.java:628
Taint Flags:

```
625  else {
626  ExternalUserVO externalUser =
externalUserRepository.findByUsername(userModel.getUserEmailId().trim());
627  if (externalUser != null) {
628  if (externalUser.get_id() != null && externalUser.get_id() != "") {
629  updateExternalModeratoruser(externalUser, campaign, moderatorRoleObj);
630  userModel.setMailSent(true);
631
```

Code Correctness: Hidden Method (1 issue)

Abstract

Static methods cannot be overridden, but may appear to be hidden when called as an instance method.

Explanation

Static methods cannot be overridden by definition, since they belong to the class rather than an instance of the class. Although there are cases where it looks like a static method has been overridden in a subclass, which may cause confusion and can lead to the incorrect version of the method being called. **Example 1:** The following tries to define an API for authenticating users.

```
class AccessLevel{
  public static final int ROOT = 0;
  //...
  public static final int NONE = 9;
}
//...
class User {
  private static int access;
  public User(){
    access = AccessLevel.ROOT;
  }
  public static int getAccessLevel(){
    return access;
  }
  //...
}
class RegularUser extends User {
  private static int access;
  public RegularUser(){
    access = AccessLevel.NONE;
  }
  public static int getAccessLevel(){
    return access;
  }
  public static void escalatePrivilege(){
    access = AccessLevel.ROOT;
  }
  //...
}
//...
class SecureArea {
  //...
  public static void doRestrictedOperation(User user){
    if (user instanceof RegularUser){
      if (user.getAccessLevel() == AccessLevel.ROOT){
        System.out.println("doing a privileged operation");
      }else{
        throw new RuntimeException();
      }
    }
  }
}
```

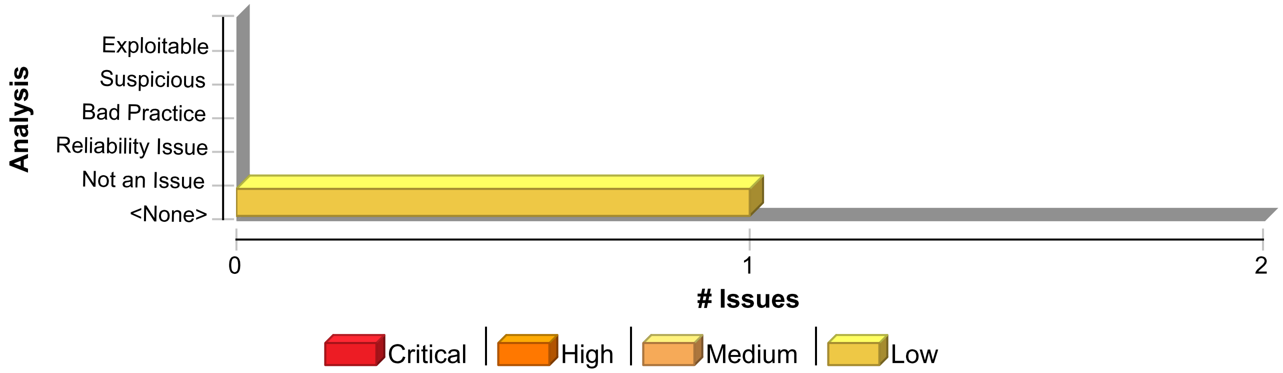
At first glance, this code looks fine. However, since we are calling the method `getAccessLevel()` against the instance `user` and not against the classes `User` or `RegularUser`, it will mean that this condition will always return `true`, and the restricted operation will be performed even though `instanceof` was used in order to get into this part of the `if/else` block.

Recommendation

When possible, try not to give static methods a signature similar to those in superclasses to avoid confusion. Ideally the code should be refactored to make it clearer, such as putting these into separate `final` classes, which cannot be subclassed, or making the constructor `private` therefore preventing the static methods being called against an instance of a class instead of the class itself. **Example 2:** The following refactors the code to make the permissions clearer, and uses the principle of least privilege to make sure it fails securely.

```
class AccessLevel{
  public static final int ROOT = 0;
  ...
  public static final int NONE = 9;
}
...
class User {
  protected int access;
  public User(){
    access = AccessLevel.NONE;
    ...
  }
  public int getAccessLevel(){
    return access;
  }
  ...
}
final class RegularUser extends User {
  ...
}
final class AdminUser extends User {
  public AdminUser(){
    access = AccessLevel.ROOT;
  }
}
...
class SecureArea {
  ...
  public void doRestrictedOperation(User user) throws
PermissionNotAllowedException {
    if (user.getAccessLevel() == AccessLevel.ROOT){
      //do operation
      ...
    }else{
      throw new PermissionNotAllowedException();
    }
  }
}
```

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Code Correctness: Hidden Method	1	0	0	1
Total	**1**	**0**	**0**	**1**

Code Correctness: Hidden Method — Low

Package: com.cts.dcrowd.rest.defect.app.domain

src/main/java/com/cts/dcrowd/rest/defect/app/domain/DefectEntry.java, line 464 (Code Correctness: Hidden Method) — **Low**

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: getSerialversionuid
Enclosing Method: getSerialversionuid()
File: src/main/java/com/cts/dcrowd/rest/defect/app/domain/DefectEntry.java:464
Taint Flags:

```
461  *
462  * @return the serialversionuid
463  */
464  public static long getSerialversionuid() {
465  return serialVersionUID;
466  }
467
```

Code Correctness: null Argument to equals() (1 issue)

Abstract

The expression `obj.equals(null)` will always be false.

Explanation

The program uses the `equals()` method to compare an object with `null`. This comparison will always return false, since the object is not `null`. (If the object is null, the program will throw a `NullPointerException`).

Recommendation

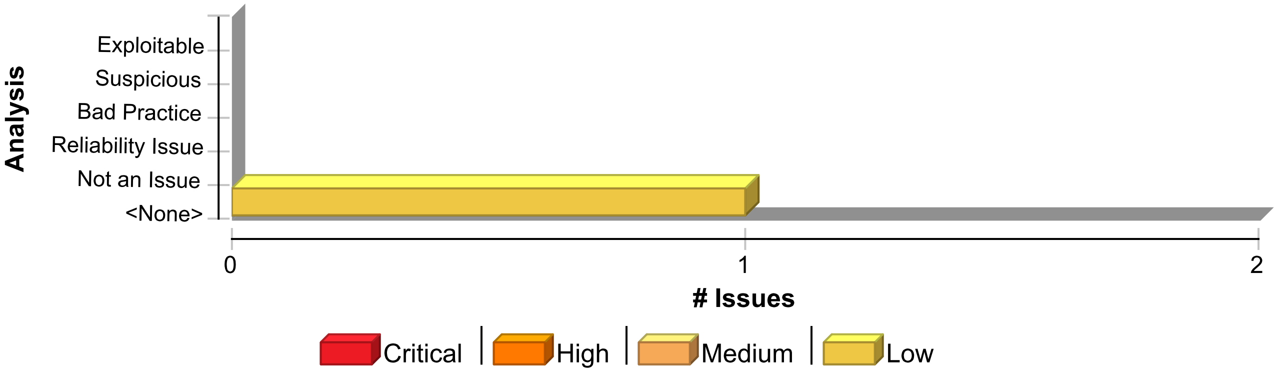
It is possible that the programmer intended to check to see if the object was null. Instead of writing

```
obj.equals(null)
```

they intended

```
obj == null
```

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Code Correctness: null Argument to equals()	1	0	0	1
Total	**1**	**0**	**0**	**1**

Code Correctness: null Argument to equals() — Low

Package: com.cts.dcrowd.rest.app.utils

src/main/java/com/cts/dcrowd/rest/app/utils/Util.java, line 522 (Code Correctness: null Argument to equals()) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionCall: equals

Code Correctness: null Argument to equals() Low

Package: com.cts.dcrowd.rest.app.utils

src/main/java/com/cts/dcrowd/rest/app/utils/Util.java, line 522 (Code Correctness: null Argument to equals()) Low

Enclosing Method: generateBlowFishDecryptValue()
File: src/main/java/com/cts/dcrowd/rest/app/utils/Util.java:522
Taint Flags:

```
519  * @return the string
520  */
521  public static String generateBlowFishDecryptValue(String encryptedValue) {
522  if(encryptedValue == null || encryptedValue.equals(null))
523  {
524  return null;
525  }
```

Code Correctness: toString on Array (1 issue)

Abstract

`toString()` is called on an array.

Explanation

In most cases, a call to `toString()` on an array indicates a developer is interested in returning the contents of the array as a String. However, a direct call to `toString()` on an array will return a string value containing the array's type and hashcode in memory. **Example 1:** The following code will output `[Ljava.lang.String;@1232121`.

```
String[] strList = new String[5];
...
System.out.println(strList);
```

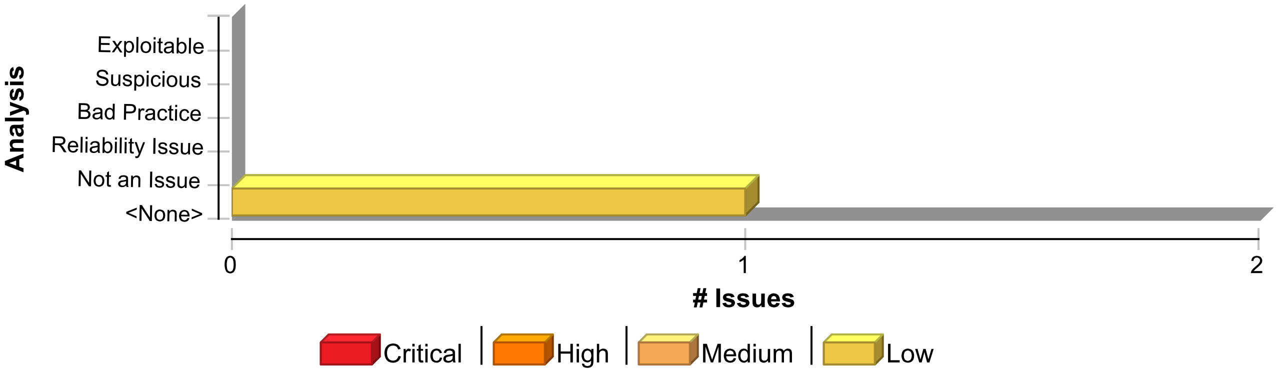
Recommendation

Several `Array.toString()` methods were introduced in Java 5 [1]. These methods will return a string representation of the array contents in comma delimited format. **Example 2:**

```
String[] strList = new String[5];
...
System.out.println(Arrays.toString(strList));
```

If you are using an older version of Java, you will need to iterate through the array to get the value of each element.

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Code Correctness: toString on Array	1	0	0	1
Total	**1**	**0**	**0**	**1**

Code Correctness: toString on Array	**Low**
Package: com.cts.dcrowd.rest.app.utils	
src/main/java/com/cts/dcrowd/rest/app/utils/Util.java, line 507 (Code Correctness: toString on Array)	**Low**

Issue Details

Code Correctness: toString on Array — Low

Package: com.cts.dcrowd.rest.app.utils

src/main/java/com/cts/dcrowd/rest/app/utils/Util.java, line 507 (Code Correctness: toString on Array) — Low

Kingdom: API Abuse
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionCall: toString
Enclosing Method: generateBlowFishEncryptValue()
File: src/main/java/com/cts/dcrowd/rest/app/utils/Util.java:507
Taint Flags:

```
504  cipher.init(Cipher.ENCRYPT_MODE, skeyspec);
505  byte[] encrypted=cipher.doFinal(sourceValue.getBytes());
506  //strData = base64Encoder(encrypted);
507  strData = encrypted.toString();
508  } catch (Exception e) {
509  e.printStackTrace();
510  }
```

Dead Code: Expression is Always false (3 issues)

Abstract

This expression (or part of it) will always evaluate to `false`.

Explanation

This expression (or part of it) will always evaluate to `false`; the program could be rewritten in a simpler form. The nearby code may be present for debugging purposes, or it may not have been maintained along with the rest of the program. The expression may also be indicative of a bug earlier in the method.
Example 1: The following method never sets the variable `secondCall` after initializing it to `false`. (The variable `firstCall` is mistakenly used twice.) The result is that the expression `firstCall && secondCall` will always evaluate to `false`, so `setUpDualCall()` will never be invoked.

```
public void setUpCalls() {
  boolean firstCall = false;
  boolean secondCall = false;

  if (fCall > 0) {
    setUpFCall();
    firstCall = true;
  }
  if (sCall > 0) {
    setUpSCall();
    firstCall = true;
  }

  if (firstCall && secondCall) {
    setUpDualCall();
  }
}
```

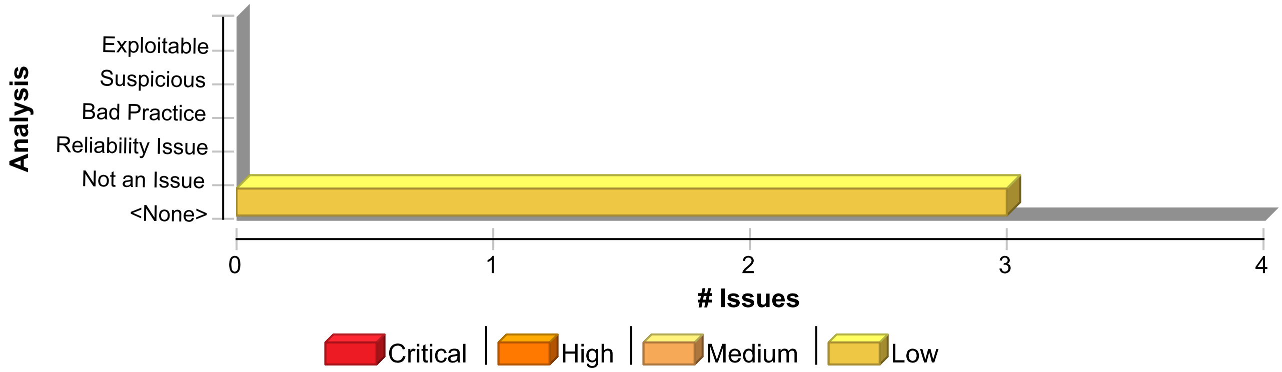
Example 2: The following method never sets the variable `firstCall` to `true`. (The variable `firstCall` is mistakenly set to `false` after the first conditional statement.) The result is that the first part of the expression `firstCall && secondCall` will always evaluate to `false`.

```
public void setUpCalls() {
  boolean firstCall = false;
  boolean secondCall = false;

  if (fCall > 0) {
    setUpFCall();
    firstCall = false;
  }
  if (sCall > 0) {
    setUpSCall();
    secondCall = true;
  }

  if (firstCall || secondCall) {
    setUpForCall();
  }
}
```

Recommendation

In general, you should repair or remove unused code. It causes additional complexity and maintenance burden without contributing to the functionality of the program.

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Dead Code: Expression is Always false	3	0	0	3
Total	**3**	**0**	**0**	**3**

Dead Code: Expression is Always false — Low

Package: com.cts.dcrowd.rest.app.controller

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 789 (Dead Code: Expression is Always false) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: IfStatement
Enclosing Method: removeCampaign()
File: src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java:789
Taint Flags:

```
786  } catch (DcrowdException de){
787  de.printStackTrace();
788  LOGGER.error("DcrowdException : ", de.getMessage());
789  if(status != 200) {
790  LOGGER.error("DcrowdException : Failed in CampaignController - Unable to remove
Campaign");
791  return buildFailureResponse( 226, "Failed in CampaignController - Unable to remove
Campaign");
792  }
```

Dead Code: Expression is Always false — Low

Package: com.cts.dcrowd.rest.defect.app.services.impl

src/main/java/com/cts/dcrowd/rest/defect/app/services/impl/DcrowdDefectEntryServiceImpl.java, line 143 (Dead Code: Expression is Always false) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: IfStatement
Enclosing Method: save()
File: src/main/java/com/cts/dcrowd/rest/defect/app/services/impl/DcrowdDefectEntryServiceImpl.java:143
Taint Flags:

```
140  saveException = true;
141  }
142  });
143  if(saveException) {
144  throw new DcrowdException("not able to write the attachments to file server");
145  }
146  this.save(defectentry);
```

src/main/java/com/cts/dcrowd/rest/defect/app/services/impl/DcrowdDefectEntryServiceImpl.java, line 225 (Dead Code: Expression is Always false) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: IfStatement
Enclosing Method: update()
File: src/main/java/com/cts/dcrowd/rest/defect/app/services/impl/DcrowdDefectEntryServiceImpl.java:225
Taint Flags:

```
222  saveException = true;
223  }
224  });
225  if(saveException) {
226  throw new DcrowdException("not able to write the attachments to file server");
227  }
228  defectentry = defaultServiceImpl.modifyDefect(defectentry);
```

Dead Code: Unused Field (40 issues)

Abstract

This field is never used.

Explanation

This field is never accessed, except perhaps by dead code. Dead code is defined as code that is never directly or indirectly executed by a public method. It is likely that the field is simply vestigial, but it is also possible that the unused field points out a bug. **Example 1:** The field named `glue` is not used in the following class. The author of the class has accidentally put quotes around the field name, transforming it into a string constant.

```
public class Dead {

  String glue;

  public String getGlue() {
    return "glue";
  }

}
```

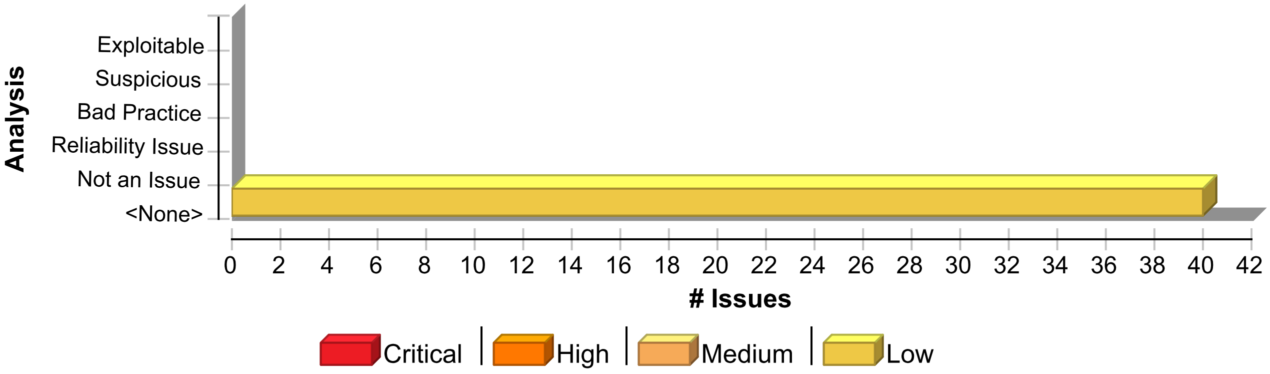
Example 2: The field named `glue` is used in the following class, but only from a method that is never called.

```
public class Dead {

  String glue;

  private String getGlue() {
    return glue;
  }

}
```

Recommendation

In general, you should repair or remove dead code. To repair dead code, execute the dead code directly or indirectly through a public method. Dead code causes additional complexity and maintenance burden without contributing to the functionality of the program.

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Dead Code: Unused Field	40	0	0	40
Total	**40**	**0**	**0**	**40**

Dead Code: Unused Field — Low

Package: com.cts.dcrowd.rest.app.controller

src/main/java/com/cts/dcrowd/rest/app/controller/LoginController.java, line 75 (Dead Code: Unused Field) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: externalUserRepository
File: src/main/java/com/cts/dcrowd/rest/app/controller/LoginController.java:75
Taint Flags:

```
72
73  /** The external user repository. */
74  @Autowired
75  private ExternalUserRepository externalUserRepository;
76
77  /** The application query processor. */
78  @Autowired
```

Package: com.cts.dcrowd.rest.app.repository

src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java, line 52 (Dead Code: Unused Field) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: wac
File: src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java:52
Taint Flags:

```
49
50  /** The wac. */
51  @Autowired
52  private WebApplicationContext wac;
53
54  /** The mock mvc. */
55  MockMvc mockMvc;
```

Dead Code: Unused Field — Low

Package: com.cts.dcrowd.rest.app.repository

src/test/java/com/cts/dcrowd/rest/app/repository/BadgeControllerTestNG.java, line 54 (Dead Code: Unused Field) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: wac
File: src/test/java/com/cts/dcrowd/rest/app/repository/BadgeControllerTestNG.java:54
Taint Flags:

```
51
52  /** The wac. */
53  @Autowired
54  private WebApplicationContext wac;
55
56
57  /** The Constant OBJECT_MAPPER. */
```

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 50 (Dead Code: Unused Field) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: wac
File: src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java:50
Taint Flags:

```
47
48  /** The wac. */
49  @Autowired
50  private WebApplicationContext wac;
51
52  /** The Constant OBJECT_MAPPER. */
53  public static final ObjectMapper OBJECT_MAPPER = new ObjectMapper();
```

Package: com.cts.dcrowd.rest.app.security

src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java, line 63 (Dead Code: Unused Field) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Dead Code: Unused Field — Low

Package: com.cts.dcrowd.rest.app.security

src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java, line 63 (Dead Code: Unused Field) — Low

Sink Details

Sink: Field: expiration
File: src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java:63
Taint Flags:

```
60
61  /** The expiration. */
62  @Value("${dcrowd.token.expiration}")
63  private String expiration;
64
65  /** The dbname. */
66  @Value("${application.dbname}")
```

src/main/java/com/cts/dcrowd/rest/app/security/LdapUserDetailsMapper.java, line 68 (Dead Code: Unused Field) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: masterService
File: src/main/java/com/cts/dcrowd/rest/app/security/LdapUserDetailsMapper.java:68
Taint Flags:

```
65
66  /** The master service. */
67  @Autowired
68  private MasterService masterService;
69
70  /** The badge service. */
71  @Autowired
```

src/main/java/com/cts/dcrowd/rest/app/security/LdapUserDetailsMapperWS.java, line 77 (Dead Code: Unused Field) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: userActivityRepository
File: src/main/java/com/cts/dcrowd/rest/app/security/LdapUserDetailsMapperWS.java:77
Taint Flags:

```
74
75  /** The user activity repository. */
```

Dead Code: Unused Field — Low

Package: com.cts.dcrowd.rest.app.security

src/main/java/com/cts/dcrowd/rest/app/security/LdapUserDetailsMapperWS.java, line 77 (Dead Code: Unused Field) — Low

```
76  @Autowired
77  private UserActivityRepository userActivityRepository;
78
79  /** The activity handler. */
80  @Autowired
```

src/main/java/com/cts/dcrowd/rest/app/security/LdapUserDetailsMapper.java, line 84 (Dead Code: Unused Field) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: userActivityRepository
File: src/main/java/com/cts/dcrowd/rest/app/security/LdapUserDetailsMapper.java:84
Taint Flags:

```
81
82  /** The user activity repository. */
83  @Autowired
84  private UserActivityRepository userActivityRepository;
85
86  /*
87   * (non-Javadoc)
```

src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java, line 59 (Dead Code: Unused Field) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: secret
File: src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java:59
Taint Flags:

```
56
57  /** The secret. */
58  @Value("${dcrowd.token.secret}")
59  private String secret;
60
61  /** The expiration. */
62  @Value("${dcrowd.token.expiration}")
```

Dead Code: Unused Field

Low

Package: com.cts.dcrowd.rest.app.security.configuration

src/main/java/com/cts/dcrowd/rest/app/security/configuration/RestFilter.java, line 36 (Dead Code: Unused Field)

Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: allowedMethodsList
File: src/main/java/com/cts/dcrowd/rest/app/security/configuration/RestFilter.java:36
Taint Flags:

```
33
34  /** The allowed methods list. */
35  @Value("${ui.origin.allow.methods.aslist}")
36  private String allowedMethodsList;
37
38  /** The allowed origins. */
39  @Value("${ui.origin.uri.aslist}")
```

src/main/java/com/cts/dcrowd/rest/app/security/configuration/WebSecurityConfiguration.java, line 59 (Dead Code: Unused Field)

Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: ldapUrl
File: src/main/java/com/cts/dcrowd/rest/app/security/configuration/WebSecurityConfiguration.java:59
Taint Flags:

```
56
57  /** The ldap url. */
58  @Value("${ldap.url}")
59  private String ldapUrl;
60
61  /** The ldap base. */
62  @Value("${ldap.base}")
```

src/main/java/com/cts/dcrowd/rest/app/security/configuration/WebSecurityConfiguration.java, line 63 (Dead Code: Unused Field)

Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Dead Code: Unused Field — Low

Package: com.cts.dcrowd.rest.app.security.configuration

src/main/java/com/cts/dcrowd/rest/app/security/configuration/WebSecurityConfiguration.java, line 63 (Dead Code: Unused Field) — Low

Sink: Field: ldapBase
File: src/main/java/com/cts/dcrowd/rest/app/security/configuration/WebSecurityConfiguration.java:63
Taint Flags:

```
60
61  /** The ldap base. */
62  @Value("${ldap.base}")
63  private String ldapBase;
64
65  /** The user dn. */
66  @Value("${ldap.userDn}")
```

Package: com.cts.dcrowd.rest.app.security.service.impl

src/main/java/com/cts/dcrowd/rest/app/security/service/impl/SecurityServiceImpl.java, line 37 (Dead Code: Unused Field) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: authenticationManager
File: src/main/java/com/cts/dcrowd/rest/app/security/service/impl/SecurityServiceImpl.java:37
Taint Flags:

```
34
35  /** The authentication manager. */
36  @Autowired
37  private AuthenticationManager authenticationManager;
38
39  @Autowired
40  private DcrowdUserDetailsService dcrowdUserDetailsService;
```

Package: com.cts.dcrowd.rest.app.service

src/main/java/com/cts/dcrowd/rest/app/service/UserActivityService.java, line 28 (Dead Code: Unused Field) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: userActivityRepository
File: src/main/java/com/cts/dcrowd/rest/app/service/UserActivityService.java:28
Taint Flags:

Dead Code: Unused Field — Low

Package: com.cts.dcrowd.rest.app.service

src/main/java/com/cts/dcrowd/rest/app/service/UserActivityService.java, line 28 (Dead Code: Unused Field) — Low

```
25
26  /** The user activity repository. */
27  @Autowired
28  private UserActivityRepository userActivityRepository;
29
30
31  /**
```

src/main/java/com/cts/dcrowd/rest/app/service/EmailService.java, line 72 (Dead Code: Unused Field) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: emailRepository
File: src/main/java/com/cts/dcrowd/rest/app/service/EmailService.java:72
Taint Flags:

```
69
70  /** The email repository. */
71  @Autowired
72  private EmailRepository emailRepository;
73
74 /** The free marker configuration. */
75  /*@Autowired
```

src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java, line 45 (Dead Code: Unused Field) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: messageSource
File: src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java:45
Taint Flags:

```
42
43  /** The message source. */
44  @Autowired
45  private MessageSource messageSource;
46
47  /** The badge service. */
```

Dead Code: Unused Field — Low

Package: com.cts.dcrowd.rest.app.service

src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java, line 45 (Dead Code: Unused Field) — Low

```
48  @Autowired
```

src/main/java/com/cts/dcrowd/rest/app/service/UserService.java, line 77 (Dead Code: Unused Field) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: messageSource
File: src/main/java/com/cts/dcrowd/rest/app/service/UserService.java:77
Taint Flags:

```
74
75  /** The message source. */
76  @Autowired
77  private MessageSource messageSource;
78
79  /** The badge service. */
80  @Autowired
```

src/main/java/com/cts/dcrowd/rest/app/service/MerchandiseService.java, line 45 (Dead Code: Unused Field) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: mongoTemplate
File: src/main/java/com/cts/dcrowd/rest/app/service/MerchandiseService.java:45
Taint Flags:

```
42
43  /** The mongo template. */
44  @Autowired
45  private MongoTemplate mongoTemplate;
46
47
48  /**
```

src/main/java/com/cts/dcrowd/rest/app/service/EmailService.java, line 61 (Dead Code: Unused Field) — Low

Issue Details

Dead Code: Unused Field — Low

Package: com.cts.dcrowd.rest.app.service

src/main/java/com/cts/dcrowd/rest/app/service/EmailService.java, line 61 (Dead Code: Unused Field) — Low

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: fromId
File: src/main/java/com/cts/dcrowd/rest/app/service/EmailService.java:61
Taint Flags:

```
58
59  /** The from id. */
60  @Value("${app.from.id}")
61  private String fromId;
62
63  /** The from send url. */
64  @Value("${email.send.url}")
```

Package: com.cts.dcrowd.rest.app.service.helper

src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java, line 73 (Dead Code: Unused Field) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: defectQueryProcessor
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java:73
Taint Flags:

```
70
71  /** The query processor. */
72  @Autowired
73  private DefectsQueryProcessor defectQueryProcessor;
74
75  /** The activity master service. */
76  @Autowired
```

src/main/java/com/cts/dcrowd/rest/app/service/helper/CustomerServiceHelper.java, line 47 (Dead Code: Unused Field) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Dead Code: Unused Field — Low

Package: com.cts.dcrowd.rest.app.service.helper

src/main/java/com/cts/dcrowd/rest/app/service/helper/CustomerServiceHelper.java, line 47 (Dead Code: Unused Field) — Low

Sink: Field: addressRepository
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/CustomerServiceHelper.java:47
Taint Flags:

```
44
45  /** The address repository. */
46  @Autowired
47  private AddressRepository addressRepository;
48
49  /** The company repository. */
50  @Autowired
```

src/main/java/com/cts/dcrowd/rest/app/service/helper/ExternalServiceHelper.java, line 41 (Dead Code: Unused Field) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: userActivityRepository
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/ExternalServiceHelper.java:41
Taint Flags:

```
38
39  /** The user activity repository. */
40  @Autowired
41  private UserActivityRepository userActivityRepository;
42
43  /**
44  * Creates or update external user.
```

src/main/java/com/cts/dcrowd/rest/app/service/helper/CustomerServiceHelper.java, line 63 (Dead Code: Unused Field) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: userActivityRepository
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/CustomerServiceHelper.java:63
Taint Flags:

```
60
61  /** The user activity repository. */
62  @Autowired
```

Dead Code: Unused Field | Low

Package: com.cts.dcrowd.rest.app.service.helper

src/main/java/com/cts/dcrowd/rest/app/service/helper/CustomerServiceHelper.java, line 63 (Dead Code: Unused Field) | Low

```
63  private UserActivityRepository userActivityRepository;
64
65  /** The master service. */
66  @Autowired
```

src/main/java/com/cts/dcrowd/rest/app/service/helper/CustomerServiceHelper.java, line 51 (Dead Code: Unused Field) | Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: companyRepository
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/CustomerServiceHelper.java:51
Taint Flags:

```
48
49  /** The company repository. */
50  @Autowired
51  private CompanyRepository companyRepository;
52
53  /** The role repository. */
54  @Autowired
```

src/main/java/com/cts/dcrowd/rest/app/service/helper/ActivityMessageHandler.java, line 35 (Dead Code: Unused Field) | Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: messageSource
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/ActivityMessageHandler.java:35
Taint Flags:

```
32
33  /** The message source. */
34  @Autowired
35  private MessageSource messageSource;
36
37  /** The security service. */
38  @Autowired
```

Dead Code: Unused Field — Low

Package: com.cts.dcrowd.rest.app.service.helper

src/main/java/com/cts/dcrowd/rest/app/service/helper/ActivityMessageHandler.java, line 39 (Dead Code: Unused Field) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: securityService
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/ActivityMessageHandler.java:39
Taint Flags:

```
36
37  /** The security service. */
38  @Autowired
39  private SecurityService securityService;
40
41  /** The user activity repository. */
42  @Autowired
```

src/main/java/com/cts/dcrowd/rest/app/service/helper/ActivityMessageHandler.java, line 43 (Dead Code: Unused Field) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: userActivityRepository
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/ActivityMessageHandler.java:43
Taint Flags:

```
40
41  /** The user activity repository. */
42  @Autowired
43  private UserActivityRepository userActivityRepository;
44
45
46  /**
```

src/main/java/com/cts/dcrowd/rest/app/service/helper/ExternalServiceHelper.java, line 33 (Dead Code: Unused Field) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Dead Code: Unused Field — Low

Package: com.cts.dcrowd.rest.app.service.helper

src/main/java/com/cts/dcrowd/rest/app/service/helper/ExternalServiceHelper.java, line 33 (Dead Code: Unused Field) — Low

Sink: Field: externalUserRepository
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/ExternalServiceHelper.java:33
Taint Flags:

```
30
31  /** The external user repository. */
32  @Autowired
33  private ExternalUserRepository externalUserRepository;
34
35  /** The role repository. */
36  @Autowired
```

src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java, line 61 (Dead Code: Unused Field) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: addressRepository
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java:61
Taint Flags:

```
58
59  /** The address repository. */
60  @Autowired
61  private AddressRepository addressRepository;
62
63  /** The badge repository. */
64  @Autowired
```

src/main/java/com/cts/dcrowd/rest/app/service/helper/ApplicationQueryProcessor.java, line 80 (Dead Code: Unused Field) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: maxNewCrewMembers
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/ApplicationQueryProcessor.java:80
Taint Flags:

```
77
78  /** The ldap full url. */
79  @Value("${splashscreen.newestcrew.max.count}")
```

Dead Code: Unused Field — Low

Package: com.cts.dcrowd.rest.app.service.helper

src/main/java/com/cts/dcrowd/rest/app/service/helper/ApplicationQueryProcessor.java, line 80 (Dead Code: Unused Field) — Low

```
80  private String maxNewCrewMembers;
81
82  /** The mongo template. */
83  @Autowired
```

src/main/java/com/cts/dcrowd/rest/app/service/helper/ApplicationQueryProcessor.java, line 92 (Dead Code: Unused Field) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: cityRepository
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/ApplicationQueryProcessor.java:92
Taint Flags:

```
89
90  /** The city repository. */
91  @Autowired
92  private CityRepository cityRepository;
93
94  /** The campaign repository. */
95  @Autowired
```

src/main/java/com/cts/dcrowd/rest/app/service/helper/ApplicationQueryProcessor.java, line 84 (Dead Code: Unused Field) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: mongoTemplate
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/ApplicationQueryProcessor.java:84
Taint Flags:

```
81
82  /** The mongo template. */
83  @Autowired
84  private MongoTemplate mongoTemplate;
85
86  /** The badge repository. */
87  @Autowired
```

Dead Code: Unused Field — Low

Package: com.cts.dcrowd.rest.defect.app.controller

src/main/java/com/cts/dcrowd/rest/defect/app/controller/DMLoginController.java, line 54 (Dead Code: Unused Field) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: authType
File: src/main/java/com/cts/dcrowd/rest/defect/app/controller/DMLoginController.java:54
Taint Flags:

```
51
52  /** The auth type as db. */
53  @Value("${auth.type}")
54  private String authType;
55
56  /** The authentication manager. */
57  @Autowired
```

src/main/java/com/cts/dcrowd/rest/defect/app/controller/DMLoginController.java, line 58 (Dead Code: Unused Field) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: authenticationManager
File: src/main/java/com/cts/dcrowd/rest/defect/app/controller/DMLoginController.java:58
Taint Flags:

```
55
56  /** The authentication manager. */
57  @Autowired
58  private AuthenticationManager authenticationManager;
59
60  /** The user service. */
61  @Autowired
```

Package: com.cts.dcrowd.rest.defect.app.jira.services

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 67 (Dead Code: Unused Field) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Dead Code: Unused Field — Low

Package: com.cts.dcrowd.rest.defect.app.jira.services

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 67 (Dead Code: Unused Field) — Low

Sink Details

Sink: Field: securityService
File: src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java:67
Taint Flags:

```
64
65  /** The security service. */
66  @Autowired
67  private SecurityService securityService;
68
69  /** The dcrowd defect service. */
70  @Autowired
```

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 75 (Dead Code: Unused Field) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: defectLayoutConfigService
File: src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java:75
Taint Flags:

```
72
73  /** The defect layout config service. */
74  @Autowired
75  private DefectLayoutConfigService defectLayoutConfigService;
76
77  /**
78  * Gets the jira authentication.
```

Package: com.cts.dcrowd.rest.defect.app.services.helper

src/main/java/com/cts/dcrowd/rest/defect/app/services/helper/DefectsQueryProcessor.java, line 62 (Dead Code: Unused Field) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: mongoTemplate
File: src/main/java/com/cts/dcrowd/rest/defect/app/services/helper/DefectsQueryProcessor.java:62
Taint Flags:

Dead Code: Unused Field — Low

Package: com.cts.dcrowd.rest.defect.app.services.helper

src/main/java/com/cts/dcrowd/rest/defect/app/services/helper/DefectsQueryProcessor.java, line 62 (Dead Code: Unused Field) — Low

```
59
60  /** The mongo template. */
61  @Autowired
62  private MongoTemplate mongoTemplate;
63  /**
64  * Count defects by campaign.
65  *
```

Package: com.cts.dcrowd.rest.defect.app.services.impl

src/main/java/com/cts/dcrowd/rest/defect/app/services/impl/DcrowdDefectEntryServiceImpl.java, line 96 (Dead Code: Unused Field) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: filePathUrl
File: src/main/java/com/cts/dcrowd/rest/defect/app/services/impl/DcrowdDefectEntryServiceImpl.java:96
Taint Flags:

```
93  private String filePath;
94
95  @Value("${app.attachment.url}")
96  private String filePathUrl;
97
98  @Value("${app.filePath.seperator}")
99  private String filePathSeperator;
```

src/main/java/com/cts/dcrowd/rest/defect/app/services/impl/DefaultDefectServiceImpl.java, line 84 (Dead Code: Unused Field) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: userActivityRepository
File: src/main/java/com/cts/dcrowd/rest/defect/app/services/impl/DefaultDefectServiceImpl.java:84
Taint Flags:

```
81
82  /** The user activity repository. */
83  @Autowired
84  private UserActivityRepository userActivityRepository;
85
```

Dead Code: Unused Field — Low

Package: com.cts.dcrowd.rest.defect.app.services.impl

src/main/java/com/cts/dcrowd/rest/defect/app/services/impl/DefaultDefectServiceImpl.java, line 84 (Dead Code: Unused Field) — Low

```
86  /** The campaign service. */
87  @Autowired
```

Package: com.cts.dcrowd.rest.defect.app.util

src/main/java/com/cts/dcrowd/rest/defect/app/util/TaskSchedulerService.java, line 23 (Dead Code: Unused Field) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: emailRepo
File: src/main/java/com/cts/dcrowd/rest/defect/app/util/TaskSchedulerService.java:23
Taint Flags:

```
20  private static final Logger LOGGER = Logger.getLogger(TaskSchedulerService.class);
21
22  @Autowired
23  private EmailRepository emailRepo;
24
25  /** The badge service. */
26  @Autowired
```

Dead Code: Unused Method (7 issues)

Abstract

This method is not reachable from any method outside the class.

Explanation

This method is never called or is only called from other dead code. **Example 1:** In the following class, the method `doWork()` can never be called.

```
public class Dead {
  private void doWork() {
    System.out.println("doing work");
  }
  public static void main(String[] args) {
    System.out.println("running Dead");
  }
}
```

Example 2: In the following class, two private methods call each other, but since neither one is ever invoked from anywhere else, they are both dead code.

```
public class DoubleDead {
  private void doTweedledee() {
    doTweedledumb();
  }
  private void doTweedledumb() {
    doTweedledee();
  }
  public static void main(String[] args) {
    System.out.println("running DoubleDead");
  }
}
```

(In this case it is a good thing that the methods are dead: invoking either one would cause an infinite loop.)

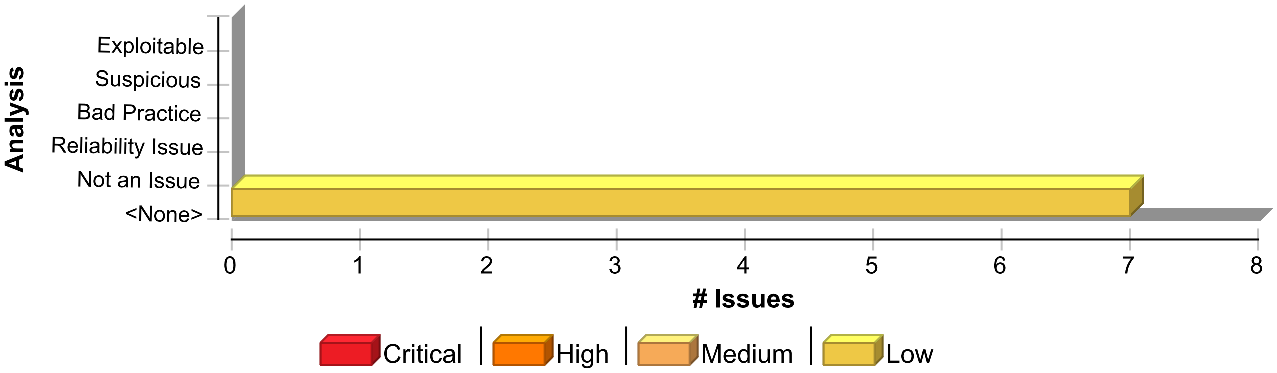
Recommendation

A dead method may indicate a bug in dispatch code. **Example 3:** If method is flagged as dead named `getWitch()` in a class that also contains the following dispatch method, it may be because of a copy-and-paste error. The 'w' case should return `getWitch()` not `getMummy()`.

```
public ScaryThing getScaryThing(char st) {
  switch(st) {
    case 'm':
      return getMummy();
    case 'w':
      return getMummy();
    default:
      return getBlob();
  }
}
```

In general, you should repair or remove dead code. To repair dead code, execute the dead code directly or indirectly through a public method. Dead code causes additional complexity and maintenance burden without contributing to the functionality of the program.

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Dead Code: Unused Method	7	0	0	7
Total	**7**	**0**	**0**	**7**

Dead Code: Unused Method — Low

Package: com.cts.dcrowd.rest.app.security

src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java, line 256 (Dead Code: Unused Method) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: generateExpirationDate
Enclosing Method: generateExpirationDate()
File: src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java:256
Taint Flags:

```
253  *
254  * @return the date
255  */
256  private Date generateExpirationDate() {
257  return new Date(System.currentTimeMillis() + Long.valueOf(this.expiration) * 1000);
258  }
259
```

Package: com.cts.dcrowd.rest.app.service.helper

src/main/java/com/cts/dcrowd/rest/app/service/helper/ExternalServiceHelper.java, line 177 (Dead Code: Unused Method) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Dead Code: Unused Method — Low

Package: com.cts.dcrowd.rest.app.service.helper

src/main/java/com/cts/dcrowd/rest/app/service/helper/ExternalServiceHelper.java, line 177 (Dead Code: Unused Method) — Low

Sink: Function: setUserActivity
Enclosing Method: setUserActivity()
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/ExternalServiceHelper.java:177
Taint Flags:

```
174  * @param user the user
175  * @return the user VO
176  */
177  private UserVO setUserActivity(UserVO user) {
178  UserActivityVO userActivity = new UserActivityVO();
179  userActivity.set_id(Util.generateMongoID());
180  userActivity.setUserName(Util.buildFullName(user));
```

src/main/java/com/cts/dcrowd/rest/app/service/helper/ExternalServiceHelper.java, line 194 (Dead Code: Unused Method) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: validateAndSetRoleAndDesignation
Enclosing Method: validateAndSetRoleAndDesignation()
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/ExternalServiceHelper.java:194
Taint Flags:

```
191  * @param externalCustomerData the external customer data
192  * @return the user VO
193  */
194  private UserVO validateAndSetRoleAndDesignation(UserVO user, ExternalCustomerInvite
externalCustomerData) {
195  String inviteType = externalCustomerData.getInviteEventTag();
196  String designation = Role.valueOf(inviteType).getDisplayName();
197  UserRolesVO roles = roleRepository.findByRole(inviteType);
```

src/main/java/com/cts/dcrowd/rest/app/service/helper/ExternalServiceHelper.java, line 162 (Dead Code: Unused Method) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: validateAndSetEncodedPassword
Enclosing Method: validateAndSetEncodedPassword()
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/ExternalServiceHelper.java:162
Taint Flags:

Dead Code: Unused Method — Low

Package: com.cts.dcrowd.rest.app.service.helper

src/main/java/com/cts/dcrowd/rest/app/service/helper/ExternalServiceHelper.java, line 162 (Dead Code: Unused Method) — Low

```
159  * @return the user VO
160  * @throws DcrowdException the dcrowd exception
161  */
162  private UserVO validateAndSetEncodedPassword(UserVO user, Map<String, Object>
requestParam) throws DcrowdException {
163  String decodedPassword =
Util.base64Decoder(Util.assertNullAndGetValue(Util.assertKeyExists(requestParam, "password")));
164  if(!Util.regexPatternMatchByType(decodedPassword, RegExType.PASSWORD)){
165  throw new DcrowdException("Password should compliant to - Minimum 8 characters, 1
uppercase & 1 lowercase letter, 1 number");
```

Package: com.cts.dcrowd.rest.app.utils

src/main/java/com/cts/dcrowd/rest/app/utils/Util.java, line 478 (Dead Code: Unused Method) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: stopWordRemove
Enclosing Method: stopWordRemove()
File: src/main/java/com/cts/dcrowd/rest/app/utils/Util.java:478
Taint Flags:

```
475  * @param splitWords the split words
476  * @return the string[]
477  */
478  private static String[] stopWordRemove(String[] splitWords) {
479  return StopwordUtility.getInstance().removeStopWords(splitWords);
480  }
481
```

Package: com.cts.dcrowd.rest.defect.app.jira.services

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 583 (Dead Code: Unused Method) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: getDefect
Enclosing Method: getDefect()
File: src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java:583

Dead Code: Unused Method — Low

Package: com.cts.dcrowd.rest.defect.app.jira.services

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 583 (Dead Code: Unused Method) — Low

Taint Flags:

```
580  * @return the defect
581  * @throws DcrowdException the dcrowd exception
582  */
583  private DefectEntry getDefect(JsonNode rootNode, String defectName) throws
DcrowdException {
584
585  JsonNode defectNode = rootNode.path("defectList");
586  DefectEntry defectEntry = null;
```

Package: com.cts.dcrowd.rest.defect.app.services.impl

src/main/java/com/cts/dcrowd/rest/defect/app/services/impl/DefaultDefectServiceImpl.java, line 390 (Dead Code: Unused Method) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: notifyCommenters
Enclosing Method: notifyCommenters()
File: src/main/java/com/cts/dcrowd/rest/defect/app/services/impl/DefaultDefectServiceImpl.java:390
Taint Flags:

```
387  * @param eventType the event type
388  * @throws DcrowdException the dcrowd exception
389  */
390  private void notifyCommenters(DefectEntry defectEntry, EventType eventType) throws
DcrowdException {
391  List<Comment> commentsList = defectEntry.getCommentsForDefects();
392  for(Comment comment : commentsList){
393  UserVO user = userRepository.findByUsername(comment.getCommentedUserEmpId());
```

Denial of Service (3 issues)

Abstract

An attacker could cause the program to crash or otherwise become unavailable to legitimate users.

Explanation

Attackers may be able to deny service to legitimate users by flooding the application with requests, but flooding attacks can often be defused at the network layer. More problematic are bugs that allow an attacker to overload the application using a small number of requests. Such bugs allow the attacker to specify the quantity of system resources their requests will consume or the duration for which they will use them. **Example 1:** The following code allows a user to specify the amount of time for which a thread will sleep. By specifying a large number, an attacker may tie up the thread indefinitely. With a small number of requests, the attacker may deplete the application's thread pool.

```
  int usrSleepTime = Integer.parseInt(usrInput);
  Thread.sleep(usrSleepTime);
```

Example 2: The following code reads a String from a zip file. Because it uses the `readLine()` method, it will read an unbounded amount of input. An attacker may take advantage of this code to cause an `OutOfMemoryException` or to consume a large amount of memory so that the program spends more time performing garbage collection or runs out of memory during some subsequent operation.

```
  InputStream zipInput = zipFile.getInputStream(zipEntry);
  Reader zipReader = new InputStreamReader(zipInput);
  BufferedReader br = new BufferedReader(zipReader);
  String line = br.readLine();
```

Recommendation

Validate user input to ensure that it will not cause inappropriate resource utilization. **Example 3:** The following code allows a user to specify the amount of time for which a thread will sleep just as in Example 1, but only if the value is within reasonable bounds.

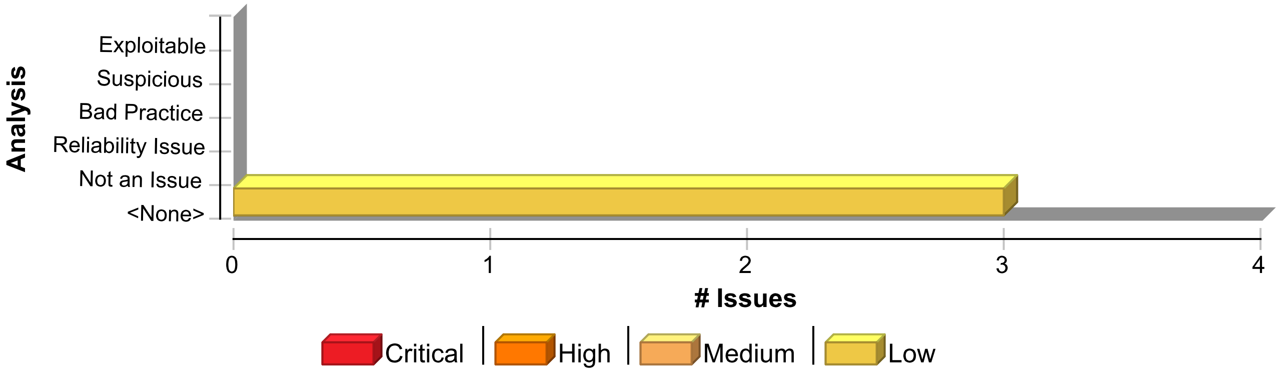
```
  int usrSleepTime = Integer.parseInt(usrInput);
  if (usrSleepTime >= SLEEP_MIN &&
      usrSleepTime <= SLEEP_MAX) {
    Thread.sleep(usrSleepTime);
  } else {
    throw new Exception("Invalid sleep duration");
  }
}
```

Example 4: The following code reads a String from a zip file just as in Example 2, but the maximum string length it will read is `MAX_STR_LEN` characters.

```
  InputStream zipInput = zipFile.getInputStream(zipEntry);
  Reader zipReader = new InputStreamReader(zipInput);
  BufferedReader br = new BufferedReader(zipReader);
  StringBuffer sb = new StringBuffer();
  int intC;
  while ((intC = br.read()) != -1) {
    char c = (char) intC;
    if (c == '\n') {
      break;
    }
    if (sb.length() >= MAX_STR_LEN) {
      throw new Exception("input too long");
    }
    sb.append(c);
```

```
}
String line = sb.toString();
```

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Denial of Service	3	0	0	3
Total	**3**	**0**	**0**	**3**

Denial of Service — Low

Package: com.cts.dcrowd.rest.app.utils

src/main/java/com/cts/dcrowd/rest/app/utils/Util.java, line 329 (Denial of Service) — Low

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Semantic)

Sink Details

Sink: readLine()
Enclosing Method: decompressByteArray()
File: src/main/java/com/cts/dcrowd/rest/app/utils/Util.java:329
Taint Flags:

```
326  BufferedReader bf = new BufferedReader(new InputStreamReader(gis, "UTF-8"));
327
328  String line;
329  while ((line=bf.readLine())!=null) {
330  outStr += line;
331  }
332  } catch (Exception e) {
```

src/main/java/com/cts/dcrowd/rest/app/utils/StopwordUtility.java, line 60 (Denial of Service) — Low

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Semantic)

Denial of Service — Low

Package: com.cts.dcrowd.rest.app.utils

src/main/java/com/cts/dcrowd/rest/app/utils/StopwordUtility.java, line 60 (Denial of Service) — Low

Sink Details

Sink: readLine()
Enclosing Method: loadStopList()
File: src/main/java/com/cts/dcrowd/rest/app/utils/StopwordUtility.java:60
Taint Flags:

```
57  try (BufferedReader br = new BufferedReader(
58  new InputStreamReader(StopwordUtility.class.getResourceAsStream("/stoplist.txt")))) {
59  String line;
60  while ((line = br.readLine()) != null) {
61  String stopword = line.trim();
62  if (stopword.length() > 0) {
63  stoplist.add(stopword);
```

src/main/java/com/cts/dcrowd/rest/app/utils/EmailDomainFilterUtility.java, line 58 (Denial of Service) — Low

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Semantic)

Sink Details

Sink: readLine()
Enclosing Method: loadDomainList()
File: src/main/java/com/cts/dcrowd/rest/app/utils/EmailDomainFilterUtility.java:58
Taint Flags:

```
55  try (BufferedReader br = new BufferedReader(
56  new InputStreamReader(EmailDomainFilterUtility.class.getResourceAsStream("/
domain_filter.txt")))) {
57  String line;
58  while ((line = br.readLine()) != null) {
59  String stopword = line.trim();
60  if (stopword.length() > 0) {
61  emailDomainList.add(stopword);
```

J2EE Bad Practices: Leftover Debug Code (2 issues)

Abstract

Debug code can create unintended entry points in a deployed web application.

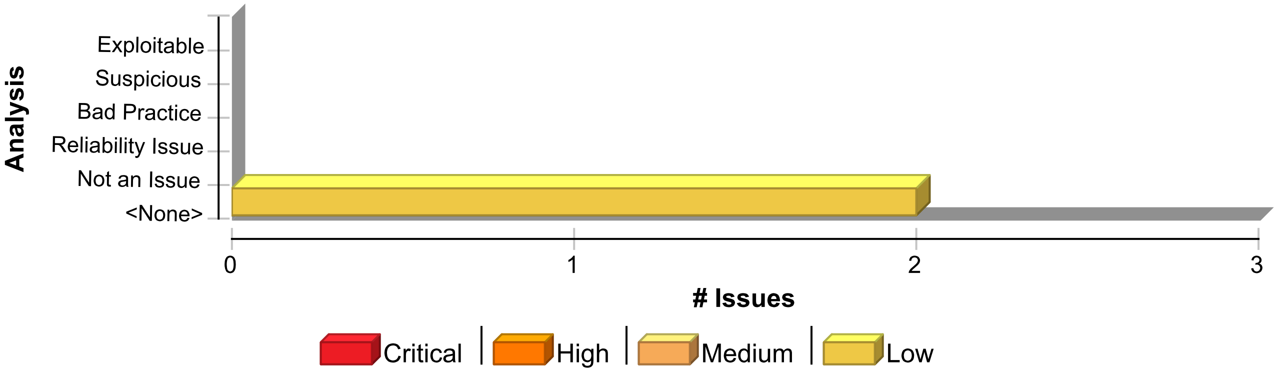
Explanation

A common development practice is to add "back door" code specifically designed for debugging or testing purposes that is not intended to be shipped or deployed with the application. When this sort of debug code is accidentally left in the application, the application is open to unintended modes of interaction. These back door entry points create security risks because they are not considered during design or testing and fall outside of the expected operating conditions of the application. The most common example of forgotten debug code is a `main()` method appearing in a web application. Although this is an acceptable practice during product development, classes that are part of a production J2EE application should not define a `main()`.

Recommendation

Remove debug code before deploying a production version of an application. Regardless of whether a direct security threat can be articulated, it is unlikely that there is a legitimate reason for such code to remain in the application after the early stages of development.

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
J2EE Bad Practices: Leftover Debug Code	2	0	0	2
Total	**2**	**0**	**0**	**2**

J2EE Bad Practices: Leftover Debug Code	Low
Package: com.cts.dcrowd.rest.app	
src/main/java/com/cts/dcrowd/rest/app/ApplicationStarter.java, line 29 (J2EE Bad Practices: Leftover Debug Code)	**Low**

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Structural)

J2EE Bad Practices: Leftover Debug Code | Low

Package: com.cts.dcrowd.rest.app

src/main/java/com/cts/dcrowd/rest/app/ApplicationStarter.java, line 29 (J2EE Bad Practices: Leftover Debug Code) | Low

Sink Details

Sink: Function: main
Enclosing Method: main()
File: src/main/java/com/cts/dcrowd/rest/app/ApplicationStarter.java:29
Taint Flags:

```
26  *
27  * @param args the arguments
28  */
29  public static void main(String[] args) {
30  SpringApplication.run(new Object[] { ApplicationStarter.class }, args);
31  }
32
```

Package: com.cts.dcrowd.rest.app.util

src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java, line 32 (J2EE Bad Practices: Leftover Debug Code) | Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: main
Enclosing Method: main()
File: src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java:32
Taint Flags:

```
29  * @param args the arguments
30  * @return the JSON object
31  */
32  public static void main(String[] args) {
33  String encrypted = generateBlowFishEncryptValue("password");
34  // System.out.println(generateBlowFishDecryptValue("Y3isgEgqkQUyh525Wy5M+w=="));
35
```

Null Dereference (5 issues)

Abstract

The program can potentially dereference a null pointer, thereby causing a null pointer exception.

Explanation

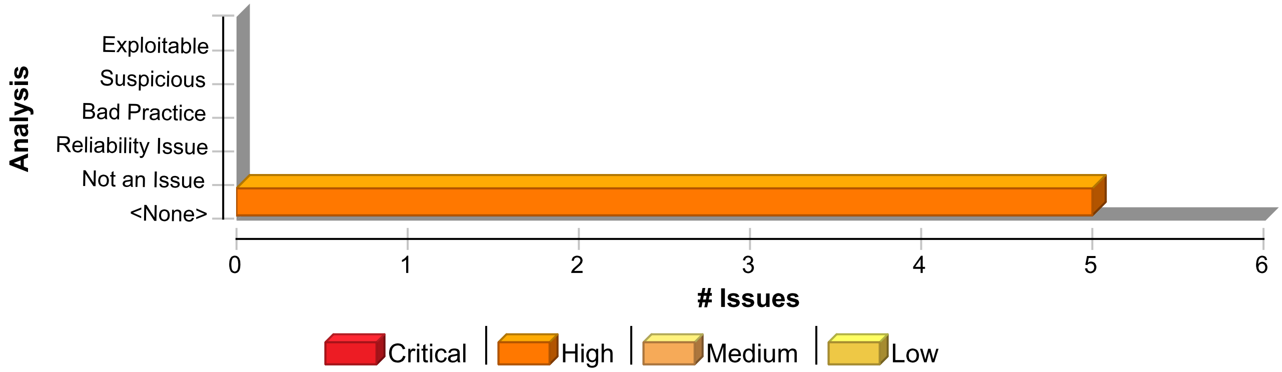
Null pointer exceptions usually occur when one or more of the programmer's assumptions is violated. A dereference-after-store error occurs when a program explicitly sets an object to `null` and dereferences it later. This error is often the result of a programmer initializing a variable to `null` when it is declared. Most null pointer issues result in general software reliability problems, but if attackers can intentionally trigger a null pointer dereference, they can use the resulting exception to bypass security logic or to cause the application to reveal debugging information that will be valuable in planning subsequent attacks. **Example:** In the following code, the programmer explicitly sets the variable `foo` to `null`. Later, the programmer dereferences `foo` before checking the object for a null value.

```
Foo foo = null;
...
foo.setBar(val);
...
}
```

Recommendation

Implement careful checks before dereferencing objects that might be null. When possible, abstract null checks into wrappers around code that manipulates resources to ensure that they are applied in all cases and to minimize the places where mistakes can occur.

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Null Dereference	5	0	0	5
Total	**5**	**0**	**0**	**5**

Null Dereference — High

Package: com.cts.dcrowd.rest.app.service

src/main/java/com/cts/dcrowd/rest/app/service/CampaignService.java, line 260 (Null Dereference) — High

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Control Flow)

Sink Details

Sink: Dereferenced : campaignFromDb
Enclosing Method: saveCampaignWithStatus()
File: src/main/java/com/cts/dcrowd/rest/app/service/CampaignService.java:260
Taint Flags:

```
257  moderatorEmailList.addAll(getNewModerators(campaign.getModeratorLvl1PreviousUserIdList(),
campaign.getModeratorLvl1UserIdList(), true));
258
moderatorEmailList.addAll(getNewModerators(campaign.getModeratorLvl2PreviousUserEmailList(),
campaign.getModeratorLvl2UserEmailList(), true));
259  if(campaign.getExternalModeratorLvl1() != null &&
campaign.getExternalModeratorLvl1().size() > 0) {
260  moderatorEmailList.addAll(getNewModerators(campaignFromDb.getExternalModeratorLvl1(),
campaign.getExternalModeratorLvl1(), false));
261  }
262  if(campaign.getExternalModeratorLvl2() != null &&
campaign.getExternalModeratorLvl2().size() > 0) {
263  moderatorEmailList.addAll(getNewModerators(campaignFromDb.getExternalModeratorLvl2(),
campaign.getExternalModeratorLvl2(), false));
```

src/main/java/com/cts/dcrowd/rest/app/service/CampaignService.java, line 263 (Null Dereference) — High

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Control Flow)

Sink Details

Sink: Dereferenced : campaignFromDb
Enclosing Method: saveCampaignWithStatus()
File: src/main/java/com/cts/dcrowd/rest/app/service/CampaignService.java:263
Taint Flags:

```
260  moderatorEmailList.addAll(getNewModerators(campaignFromDb.getExternalModeratorLvl1(),
campaign.getExternalModeratorLvl1(), false));
261  }
262  if(campaign.getExternalModeratorLvl2() != null &&
campaign.getExternalModeratorLvl2().size() > 0) {
263  moderatorEmailList.addAll(getNewModerators(campaignFromDb.getExternalModeratorLvl2(),
campaign.getExternalModeratorLvl2(), false));
264  }
265
266
campaign.setIsModeratorMailSent(emailEventService.moderatorsInvitationForCampaigns(moderatorEmai
```

Null Dereference High

Package: com.cts.dcrowd.rest.app.service

src/main/java/com/cts/dcrowd/rest/app/service/CampaignService.java, line 263 (Null Dereference) High

```
campaign.getName()));
```

src/main/java/com/cts/dcrowd/rest/app/service/CampaignService.java, line 305 (Null Dereference) High

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Control Flow)

Sink Details

Sink: Dereferenced : campaignFromDb
Enclosing Method: saveCampaignWithStatus()
File: src/main/java/com/cts/dcrowd/rest/app/service/CampaignService.java:305
Taint Flags:

```
302  if (!Util.isEmpty(campaign.getImageData().getImageData())) {
303  if (!Util.isEmpty(campaign.get_id())) {
304  if (!Util.isEmpty(campaignFromDb)) {
305  if (!Util.isEmpty(campaignFromDb.getImageData().getImageFilePath())) {
306  Util.writeFileToFileServer(null, null, campaignFromDb.getImageData().getImageFilePath(),
307  true);
308  }
```

src/main/java/com/cts/dcrowd/rest/app/service/CampaignService.java, line 322 (Null Dereference) High

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Control Flow)

Sink Details

Sink: Dereferenced : campaignFromDb
Enclosing Method: saveCampaignWithStatus()
File: src/main/java/com/cts/dcrowd/rest/app/service/CampaignService.java:322
Taint Flags:

```
319  }
320  } else {
321  if (!Util.isEmpty(campaignFromDb)) {
322  campaign.setImageData(campaignFromDb.getImageData());
323  }
324  }
325
```

Null Dereference — High

Package: com.cts.dcrowd.rest.defect.app.services.impl

src/main/java/com/cts/dcrowd/rest/defect/app/services/impl/DcrowdDefectEntryServiceImpl.java, line 696 (Null Dereference) — High

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Control Flow)

Sink Details

Sink: Dereferenced : user
Enclosing Method: cloneDefectEntriesForExport()
File: src/main/java/com/cts/dcrowd/rest/defect/app/services/impl/DcrowdDefectEntryServiceImpl.java:696
Taint Flags:

```
693  // TODO Auto-generated catch block
694  e.printStackTrace();
695  }
696  cloneDefectMap.put("LOGGEDEMAILID", user.getEmail());
697  cloneDefectMap.put("USERLOCATION", defectEntry.getUserLocation());
698  if(!Util.isEmpty(defectEntry.getDeAttachments())) {
699  defectEntry.getDeAttachments().forEach((deAttachment) -> {
```

Password Management: Null Password (1 issue)

Abstract

Null passwords can compromise security.

Explanation

Assigning `null` to password variables is a bad idea because it can allow attackers to bypass password verification or might indicate that resources are protected by an empty password. **Example 1:** The code below initializes a password variable to `null`, attempts to read a stored value for the password, and compares it against a user-supplied value.

```
    ...
    String storedPassword = null;
    String temp;

    if ((temp = readPassword()) != null) {
        storedPassword = temp;
    }

    if(Utils.verifyPassword(userPassword, storedPassword))
        // Access protected resources
        ...
    }
    ...
```

If `readPassword()` fails to retrieve the stored password due to a database error or another problem, then an attacker could trivially bypass the password check by providing a null value for `userPassword`. In the mobile environment, password management is especially important given that there is such a high chance of device loss. **Example 2:** The code below initializes username and password variables to `null`, reads credentials from an Android WebView store if they have not been previously rejected by the server for the current request, and uses them to setup authentication for viewing protected pages.

```
...
webview.setWebViewClient(new WebViewClient() {
  public void onReceivedHttpAuthRequest(WebView view,
        HttpAuthHandler handler, String host, String realm) {
    String username = null;
    String password = null;

    if (handler.useHttpAuthUsernamePassword()) {
      String[] credentials = view.getHttpAuthUsernamePassword(host, realm);
      username = credentials[0];
      password = credentials[1];
    }
    handler.proceed(username, password);
  }
});
...
```

Similar to Example 1, if `useHttpAuthUsernamePassword()` returns `false`, an attacker will be able to view protected pages by supplying a null password.

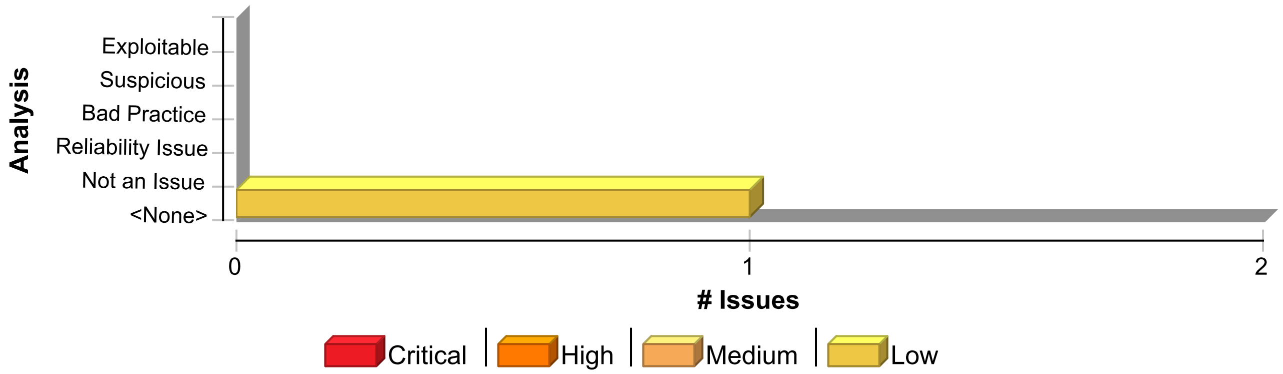
Recommendation

Always read stored password values from encrypted, external resources and assign password variables meaningful values. Ensure that sensitive resources are never protected with empty or null passwords. For Android, as well as any other platform that uses SQLite database, SQLCipher is a good alternative. SQLCipher is an extension to the SQLite database that provides transparent 256-bit AES encryption of database files. Thus, credentials can be stored in an encrypted database. **Example 3:** The code below demonstrates how to integrate SQLCipher into an Android application after downloading the necessary binaries, and store credentials into the database file.

```
import net.sqlcipher.database.SQLiteDatabase;
...
  SQLiteDatabase.loadLibs(this);
  File dbFile = getDatabasePath("credentials.db");
  dbFile.mkdirs();
  dbFile.delete();
  SQLiteDatabase db = SQLiteDatabase.openOrCreateDatabase(dbFile,
"credentials", null);
  db.execSQL("create table credentials(u, p)");
  db.execSQL("insert into credentials(u, p) values(?, ?)", new Object[]
{username, password});
...
```

Note that references to `android.database.sqlite.SQLiteDatabase` are substituted with those of `net.sqlcipher.database.SQLiteDatabase`. To enable encryption on the WebView store, WebKit has to be re-compiled with the `sqlcipher.so` library.

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Password Management: Null Password	1	0	0	1
Total	**1**	**0**	**0**	**1**

Password Management: Null Password — Low

Package: com.cts.dcrowd.rest.app.security

src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java, line 127 (Password Management: Null Password) — **Low**

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: password
Enclosing Method: getPasswordFromBase64Token()
File: src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java:127
Taint Flags:

Password Management: Null Password — Low

Package: com.cts.dcrowd.rest.app.security

src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java, line 127 (Password Management: Null Password) — Low

```
124  try {
125  password = new String(Base64.getDecoder().decode(token),UTF8).split(">:<")[1];
126  } catch (Exception e) {
127  password = null;
128  LOGGER.debug("Exception : {} \n Stack Trace : {}", e.getMessage(),
String.valueOf(e.getStackTrace()));
129  }
130  return password;
```

Password Management: Password in Comment (53 issues)

Abstract

Storing passwords or password details in plain text anywhere in the system or system code may compromise system security in a way that cannot be easily remedied.

Explanation

It is never a good idea to hardcode a password. Storing password details within comments is equivalent to hardcoding passwords. Not only does it allow all of the project's developers to view the password, it also makes fixing the problem extremely difficult. After the code is in production, the password is now leaked to the outside world and cannot be protected or changed without patching the software. If the account protected by the password is compromised, the owners of the system must choose between security and availability. **Example:** The following comment specifies the default password to connect to a database:

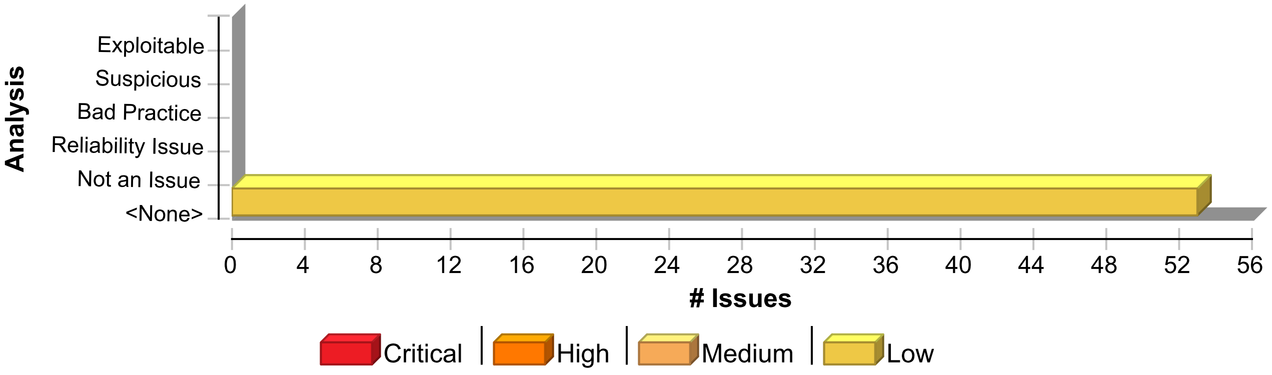
```
...
// Default username for database connection is "scott"
// Default password for database connection is "tiger"
...
```

This code will run successfully, but anyone who has access to it will have access to the password. After the program has shipped, there is likely no way to change the database user "scott" with a password of "tiger" unless the program is patched. An employee with access to this information can use it to break into the system.

Recommendation

Passwords should never be hardcoded and should generally be obfuscated and managed in an external source. Storing passwords in plain text anywhere on the system allows anyone with sufficient permissions to read and potentially misuse the password.

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Password Management: Password in Comment	53	0	0	53
Total	**53**	**0**	**0**	**53**

Password Management: Password in Comment

Low

Package: .com.cts.dcrowd.rest.app.controller

src/main/java/com/cts/dcrowd/rest/app/controller/ExternalController.java, line 78 (Password Management: Password in Comment)

Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/controller/ExternalController.java:78
Taint Flags:

```
75  }
76  }
77
78  /**
79  * Send Mail for Password Reset
80  *
81  * @param external user EmailID
```

src/main/java/com/cts/dcrowd/rest/app/controller/LoginController.java, line 103 (Password Management: Password in Comment)

Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/controller/LoginController.java:103
Taint Flags:

```
100  String username = securityService.getAutheticatedUserName(basicAuthToken);
101  String password = securityService.getAutheticatedPassword(basicAuthToken);
102  //String username = securityService.getAutheticatedUser(basicAuthToken, 0);
103  //String password = securityService.getAutheticatedUser(basicAuthToken, 1);
104
105  user = userService.findByUsername(username);
106  boolean isNewUser = Util.isEmpty(user);
```

src/main/java/com/cts/dcrowd/rest/app/controller/ExternalController.java, line 97 (Password Management: Password in Comment)

Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Password Management: Password in Comment — Low

Package: .com.cts.dcrowd.rest.app.controller

src/main/java/com/cts/dcrowd/rest/app/controller/ExternalController.java, line 97 (Password Management: Password in Comment) — Low

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/controller/ExternalController.java:97
Taint Flags:

```
94  }
95  return buildSuccessResponse(status);
96  }
97  /**
98  * Reset Password for externals
99  *
100  * @param external user EmailID
```

src/main/java/com/cts/dcrowd/rest/app/controller/LoginController.java, line 130 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/controller/LoginController.java:130
Taint Flags:

```
127  return ResponseEntity.badRequest().body("Invalid Authentication source : User not found in Database");
128  }*/
129
130  //securityService.doAuthenticateExternalCustomer(username, password, user);
131  user = userService.setAuthTokenToUser(device, user);
132  user.setPhoneNumber(null);
133
```

Package: .com.cts.dcrowd.rest.app.domain

src/main/java/com/cts/dcrowd/rest/app/domain/ExternalUserVO.java, line 295 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/domain/ExternalUserVO.java:295
Taint Flags:

Password Management: Password in Comment — Low

Package: .com.cts.dcrowd.rest.app.domain

src/main/java/com/cts/dcrowd/rest/app/domain/ExternalUserVO.java, line 295 (Password Management: Password in Comment) — Low

```
292  this.modifiedDate = modifiedDate;
293  }
294
295  /**
296  * @return the lastPasswordReset
297  */
298  public Date getLastPasswordReset() {
```

src/main/java/com/cts/dcrowd/rest/app/domain/ExternalUserVO.java, line 203 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/domain/ExternalUserVO.java:203
Taint Flags:

```
200  return password;
201  }
202
203  /**
204  * @param password the password to set
205  */
206  @JsonProperty
```

src/main/java/com/cts/dcrowd/rest/app/domain/ExternalUserVO.java, line 471 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/domain/ExternalUserVO.java:471
Taint Flags:

```
468  return customerPassword;
469  }
470
471  /**
472  * @param customerPassword the customerPassword to set
473  */
```

Password Management: Password in Comment — Low

Package: .com.cts.dcrowd.rest.app.domain

src/main/java/com/cts/dcrowd/rest/app/domain/ExternalUserVO.java, line 471 (Password Management: Password in Comment) — Low

```
474  @JsonProperty
```

src/main/java/com/cts/dcrowd/rest/app/domain/UserVO.java, line 1053 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/domain/UserVO.java:1053
Taint Flags:

```
1050  return customerPassword;
1051  }
1052
1053  /**
1054  * @param customerPassword the customerPassword to set
1055  */
1056  @JsonProperty
```

src/main/java/com/cts/dcrowd/rest/app/domain/UserVO.java, line 363 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/domain/UserVO.java:363
Taint Flags:

```
360  @Override
361  @JsonIgnore
362  public String getPassword() {
363  //return Util.generateBlowFishDecryptValue(password);
364  return "password";
365
366  }
```

src/main/java/com/cts/dcrowd/rest/app/domain/ExternalUserVO.java, line 50 (Password Management: Password in Comment) — Low

Issue Details

Password Management: Password in Comment — Low

Package: .com.cts.dcrowd.rest.app.domain

src/main/java/com/cts/dcrowd/rest/app/domain/ExternalUserVO.java, line 50 (Password Management: Password in Comment) — Low

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/domain/ExternalUserVO.java:50
Taint Flags:

```
47  /** The last login. */
48  private Date lastLogin;
49
50  /** The password. */
51  @JsonProperty(value="reg_password")
52  private String password;
53
```

src/main/java/com/cts/dcrowd/rest/app/domain/UserVO.java, line 80 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/domain/UserVO.java:80
Taint Flags:

```
77  /** The is registered. */
78  private boolean isRegistered;
79
80  /** The password. */
81  private String password;
82
83  /** The auth token. */
```

src/main/java/com/cts/dcrowd/rest/app/domain/UserVO.java, line 368 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/domain/UserVO.java:368

Password Management: Password in Comment — Low

Package: .com.cts.dcrowd.rest.app.domain

src/main/java/com/cts/dcrowd/rest/app/domain/UserVO.java, line 368 (Password Management: Password in Comment) — Low

Taint Flags:

```
365
366   }
367
368   /**
369   * Sets the password.
370   *
371   * @param password the new password
```

src/main/java/com/cts/dcrowd/rest/app/domain/UserVO.java, line 498 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/domain/UserVO.java:498
Taint Flags:

```
495
496
497
498   /**
499   * Gets the last password reset.
500   *
501   * @return the last password reset
```

src/main/java/com/cts/dcrowd/rest/app/domain/ExternalUserVO.java, line 302 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/domain/ExternalUserVO.java:302
Taint Flags:

```
299   return lastPasswordReset;
300   }
301
302   /**
```

Password Management: Password in Comment

Low

Package: .com.cts.dcrowd.rest.app.domain

src/main/java/com/cts/dcrowd/rest/app/domain/ExternalUserVO.java, line 302 (Password Management: Password in Comment)

Low

```
303  * @param lastPasswordReset the lastPasswordReset to set
304  */
305  public void setLastPasswordReset(Date lastPasswordReset) {
```

src/main/java/com/cts/dcrowd/rest/app/domain/UserVO.java, line 507 (Password Management: Password in Comment)

Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/domain/UserVO.java:507
Taint Flags:

```
504  return lastPasswordReset;
505  }
506
507  /**
508  * Sets the last password reset.
509  *
510  * @param lastPasswordreset the new last password reset
```

src/main/java/com/cts/dcrowd/rest/app/domain/ExternalUserVO.java, line 72 (Password Management: Password in Comment)

Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/domain/ExternalUserVO.java:72
Taint Flags:

```
69  /** The organization. */
70  private String organization;
71
72  /** The last password reset. */
73  private Date lastPasswordReset;
74
75  /** The nominated campaign. */
```

Password Management: Password in Comment — Low

Package: .com.cts.dcrowd.rest.app.domain

src/main/java/com/cts/dcrowd/rest/app/domain/UserVO.java, line 104 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/domain/UserVO.java:104
Taint Flags:

```
101  /** The company. */
102  private String companyId;
103
104  /** The last password reset. */
105  private Date lastPasswordReset;
106
107  /** The user badges. */
```

src/main/java/com/cts/dcrowd/rest/app/domain/UserVO.java, line 44 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/domain/UserVO.java:44
Taint Flags:

```
41  /** The Constant MAX_LENGTH_USERNAME. */
42  public static final int MAX_LENGTH_USERNAME = 100;
43
44  /** The Constant MAX_LENGTH_PASSWORD. */
45  public static final int MAX_LENGTH_PASSWORD = 25;
46
47  /** The username. */
```

src/main/java/com/cts/dcrowd/rest/app/domain/ExternalUserVO.java, line 195 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Password Management: Password in Comment — Low

Package: .com.cts.dcrowd.rest.app.domain

src/main/java/com/cts/dcrowd/rest/app/domain/ExternalUserVO.java, line 195 (Password Management: Password in Comment) — Low

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/domain/ExternalUserVO.java:195
Taint Flags:

```
192  this.lastLogin = lastLogin;
193  }
194
195  /**
196  * @return the password
197  */
198  @JsonIgnore
```

src/main/java/com/cts/dcrowd/rest/app/domain/ExternalUserVO.java, line 108 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/domain/ExternalUserVO.java:108
Taint Flags:

```
105  /** The invited users. */
106  private List<String> invitedUsers = new ArrayList<String>();
107
108  /** The Customer password */
109  private String customerPassword;
110
111  /**
```

src/main/java/com/cts/dcrowd/rest/app/domain/UserVO.java, line 173 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/domain/UserVO.java:173
Taint Flags:

```
170  /** The is eligible for gonein 60 seconds badge. */
171  private boolean isEligibleForGonein60SecondsBadge;
172
```

Password Management: Password in Comment — Low

Package: .com.cts.dcrowd.rest.app.domain

src/main/java/com/cts/dcrowd/rest/app/domain/UserVO.java, line 173 (Password Management: Password in Comment) — Low

```
173  /** The Customer password */
174  private String customerPassword;
175
176
```

src/main/java/com/cts/dcrowd/rest/app/domain/UserVO.java, line 357 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/domain/UserVO.java:357
Taint Flags:

```
354  this.isRegistered = isRegistered;
355  }
356
357  /*  (non-Javadoc)
358  * @see org.springframework.security.core.userdetails.UserDetails#getPassword()
359  */
360  @Override
```

src/main/java/com/cts/dcrowd/rest/app/domain/ExternalUserVO.java, line 463 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/domain/ExternalUserVO.java:463
Taint Flags:

```
460  this.invitedUsers = invitedUsers;
461  }
462
463  /**
464  * @return the customerPassword
465  */
466  @JsonIgnore
```

Password Management: Password in Comment — Low

Package: .com.cts.dcrowd.rest.app.domain

src/main/java/com/cts/dcrowd/rest/app/domain/UserVO.java, line 1045 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/domain/UserVO.java:1045
Taint Flags:

```
1042  this.moderatingCampaign = moderatingCampaign;
1043  }
1044
1045  /**
1046  * @return the customerPassword
1047  */
1048  @JsonIgnore
```

Package: .com.cts.dcrowd.rest.app.repository

src/main/java/com/cts/dcrowd/rest/app/repository/UserRepository.java, line 40 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/repository/UserRepository.java:40
Taint Flags:

```
37  */
38  UserVO findByEmail(String email);
39
40  /**
41  * Find by password.
42  *
43  * @param password the password
```

src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java, line 105 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Password Management: Password in Comment | Low

Package: .com.cts.dcrowd.rest.app.repository

src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java, line 105 (Password Management: Password in Comment) | Low

Sink Details

Sink: Comment
File: src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java:105
Taint Flags:

```
102  public void setup() throws Exception {
103  this.mockMvc = MockMvcBuilders.webAppContextSetup(this.wac).build();
104
105  /*int i;
106  for (i = 65; i <= 67; i++) {
107  String name = String.valueOf((char)(i));
108  createUser(name, name, name);
```

Package: .com.cts.dcrowd.rest.app.security

src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java, line 133 (Password Management: Password in Comment) | Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java:133
Taint Flags:

```
130  return password;
131  }
132
133  /**
134  * Gets the username or password from BlowFish token.
135  *
136  * @param token the token
```

src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java, line 341 (Password Management: Password in Comment) | Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java:341
Taint Flags:

Password Management: Password in Comment — Low

Package: .com.cts.dcrowd.rest.app.security

src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java, line 341 (Password Management: Password in Comment) — Low

```
338  .compact();
339  }
340
341  /**
342  * Can token be refreshed.
343  *
344  * @param token the token
```

src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java, line 114 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java:114
Taint Flags:

```
111  }
112
113
114  /**
115  * Gets the password from base 64 token.
116  *
117  * @param token the token
```

src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java, line 271 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java:271
Taint Flags:

```
268  return expiration.before(this.generateCurrentDate());
269  }
270
271  /**
272  * Checks if is created before last password reset.
273  *
```

Password Management: Password in Comment | Low

Package: .com.cts.dcrowd.rest.app.security

src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java, line 271 (Password Management: Password in Comment) | Low

```
274  * @param created the created
```

Package: .com.cts.dcrowd.rest.app.security.configuration

src/main/java/com/cts/dcrowd/rest/app/security/configuration/MongoSecurityConfiguration.java, line 58 (Password Management: Password in Comment) | Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/security/configuration/MongoSecurityConfiguration.java:58
Taint Flags:

```
55  @Value("${spring.data.mongodb.username}")
56  private String username;
57
58  /** The password. */
59  @Value("${spring.data.mongodb.password}")
60  private String password;
61
```

src/main/java/com/cts/dcrowd/rest/app/security/configuration/WebSecurityConfiguration.java, line 69 (Password Management: Password in Comment) | Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/security/configuration/WebSecurityConfiguration.java:69
Taint Flags:

```
66  @Value("${ldap.userDn}")
67  private String userDn;
68
69  /** The password. */
70  @Value("${ldap.password}")
71  private String password;
72
```

Password Management: Password in Comment

Low

Package: .com.cts.dcrowd.rest.app.security.service

src/main/java/com/cts/dcrowd/rest/app/security/service/SecurityService.java, line 54 (Password Management: Password in Comment)

Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/security/service/SecurityService.java:54
Taint Flags:

```
51  */
52  public String getAutheticatedUserName(String basicAuthToken) throws DcrowdException;
53
54  /**
55  * Gets the autheticated password.
56  *
57  * @param basicAuthToken the basic auth token
```

src/main/java/com/cts/dcrowd/rest/app/security/service/SecurityService.java, line 72 (Password Management: Password in Comment)

Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/security/service/SecurityService.java:72
Taint Flags:

```
69  */
70  public String getAutheticatedUser(String basicAuthToken, int index) throws DcrowdException;
71
72  /**
73  * Do authenticate external customer.
74  *
75  * @param username the username
```

Package: .com.cts.dcrowd.rest.app.security.service.impl

src/main/java/com/cts/dcrowd/rest/app/security/service/impl/SecurityServiceImpl.java, line 93 (Password Management: Password in Comment)

Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Password Management: Password in Comment — Low

Package: .com.cts.dcrowd.rest.app.security.service.impl

src/main/java/com/cts/dcrowd/rest/app/security/service/impl/SecurityServiceImpl.java, line 93 (Password Management: Password in Comment) — Low

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/security/service/impl/SecurityServiceImpl.java:93
Taint Flags:

```
90  String username=getAutheticatedUserName(basicAuthToken);
91  String password=getAutheticatedPassword(basicAuthToken);
92  //String username=getAutheticatedUser(basicAuthToken, 0);
93  //String password=getAutheticatedUser(basicAuthToken, 1);
94  LOGGER.debug("Inside login request username : {} and password {} ", username, password);
95
96  Authentication authentication = this.authenticationManager.authenticate(
```

src/main/java/com/cts/dcrowd/rest/app/security/service/impl/SecurityServiceImpl.java, line 118 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/security/service/impl/SecurityServiceImpl.java:118
Taint Flags:

```
115  return tokenUtils.getUsernameFromBase64Token(basicAuthToken);
116  }
117
118  /* (non-Javadoc)
119  * @see
com.cts.dcrowd.rest.app.security.service.SecurityService#getAutheticatedPassword(java.lang.Strin
120  */
121  @Override
```

Package: .com.cts.dcrowd.rest.app.service

src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java, line 349 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment

Password Management: Password in Comment — Low

Package: .com.cts.dcrowd.rest.app.service

src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java, line 349 (Password Management: Password in Comment) — Low

File: src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java:349
Taint Flags:

```
346  }
347  }
348
349  /**
350  * External Password reset.
351  *
352  * @param customerEmailId the customer email id
```

src/main/java/com/cts/dcrowd/rest/app/service/UserService.java, line 209 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/service/UserService.java:209
Taint Flags:

```
206  }
207
208
209  /**
210  * Find by password.
211  *
212  * @param password the password
```

src/main/java/com/cts/dcrowd/rest/app/service/ExternalService.java, line 212 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/service/ExternalService.java:212
Taint Flags:

```
209  return isValidUser;
210  }
211
212  /**
```

Password Management: Password in Comment — Low

Package: .com.cts.dcrowd.rest.app.service

src/main/java/com/cts/dcrowd/rest/app/service/ExternalService.java, line 212 (Password Management: Password in Comment) — Low

```
213  * password reset.
214  *
215  * @param userId the user id
```

src/main/java/com/cts/dcrowd/rest/app/service/ExternalService.java, line 191 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/service/ExternalService.java:191
Taint Flags:

```
188  return true;
189  }
190
191  /**
192  * send mail for password reset.
193  *
194  * @param userId the user id
```

src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java, line 79 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java:79
Taint Flags:

```
76  @Value("${email.templates.registration.acknowledgement}")
77  private String registrationAcknowldgement;
78
79  /** The reset password. */
80  @Value("${email.templates.reset.password}")
81  private String resetPasswordTemplate;
82
```

Password Management: Password in Comment — Low

Package: .com.cts.dcrowd.rest.app.service.helper

src/main/java/com/cts/dcrowd/rest/app/service/helper/CustomerServiceHelper.java, line 106 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/CustomerServiceHelper.java:106
Taint Flags:

```
103  return user;
104  }
105
106  /**
107  * Validate and set password.
108  *
109  * @param user the password
```

src/main/java/com/cts/dcrowd/rest/app/service/helper/ExternalServiceHelper.java, line 154 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/ExternalServiceHelper.java:154
Taint Flags:

```
151  return user;
152  }
153
154  /**
155  * Validate and set password.
156  *
157  * @param user the password
```

src/main/java/com/cts/dcrowd/rest/app/service/helper/ExternalServiceHelper.java, line 74 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Password Management: Password in Comment — Low

Package: .com.cts.dcrowd.rest.app.service.helper

src/main/java/com/cts/dcrowd/rest/app/service/helper/ExternalServiceHelper.java, line 74 (Password Management: Password in Comment) — Low

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/ExternalServiceHelper.java:74
Taint Flags:

```
71
72  // 1. Set User full name
73  user.setUserFullName(String.valueOf(new StringBuilder().append(user.getFirstname()).append(" ").append(user.getLastname())));
74  // 2. Set Password(Encrypted)
75  //user = validateAndSetEncodedPassword(user, requestParam);
76  // 3. Set Role to the User
77  user = setRoleToDesignation(user, Role.ROLE_CROWD_TESTER.getMessage());
```

src/main/java/com/cts/dcrowd/rest/app/service/helper/ExternalServiceHelper.java, line 75 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/ExternalServiceHelper.java:75
Taint Flags:

```
72  // 1. Set User full name
73  user.setUserFullName(String.valueOf(new StringBuilder().append(user.getFirstname()).append(" ").append(user.getLastname())));
74  // 2. Set Password(Encrypted)
75  //user = validateAndSetEncodedPassword(user, requestParam);
76  // 3. Set Role to the User
77  user = setRoleToDesignation(user, Role.ROLE_CROWD_TESTER.getMessage());
78  // 4. Set UserActivity
```

src/main/java/com/cts/dcrowd/rest/app/service/helper/CustomerServiceHelper.java, line 80 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Password Management: Password in Comment — Low

Package: .com.cts.dcrowd.rest.app.service.helper

src/main/java/com/cts/dcrowd/rest/app/service/helper/CustomerServiceHelper.java, line 80 (Password Management: Password in Comment) — Low

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/CustomerServiceHelper.java:80
Taint Flags:

```
77  */
78  public UserVO buildCustomerUserObject(UserVO user, Map<String, Object> requestParam,
ExternalCustomerInvite externalCustomerData) throws DcrowdException {
79
80  // 0. Set Urename - Password(Encrypted)
81  user.setEmail(user.getUsername());
82  user.setFirstname(externalCustomerData.getFirstname());
83  user.setLastname(externalCustomerData.getLastname());
```

Package: .com.cts.dcrowd.rest.app.util

src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java, line 150 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java:150
Taint Flags:

```
147  }
148  }
149
150  /**
151  * Convert base 64.
152  *
153  * @param username the username
```

src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java, line 36 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java:36
Taint Flags:

Password Management: Password in Comment — Low

Package: .com.cts.dcrowd.rest.app.util

src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java, line 36 (Password Management: Password in Comment) — Low

```
33  String encrypted = generateBlowFishEncryptValue("password");
34  // System.out.println(generateBlowFishDecryptValue("Y3isgEgqkQUyh525Wy5M+w=="));
35
36  /* String encodedValue = base64Encoder("password".getBytes());
37  String decodedValue = base64Decoder(encodedValue);
38  System.out.println(decodedValue);*/
39  }
```

Package: .com.cts.dcrowd.rest.defect.app.controller

src/main/java/com/cts/dcrowd/rest/defect/app/controller/DMLoginController.java, line 101 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/defect/app/controller/DMLoginController.java:101
Taint Flags:

```
98
99  SecurityContextHolder.getContext().setAuthentication(authentication);
100  UserVO user = new UserVO();
101  // Reload password post-authentication so we can generate token
102  try
103  {
104  user = userService.findByUsername(username);
```

Package: .com.cts.dcrowd.rest.defect.app.jira.services

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 92 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java:92
Taint Flags:

```
89  // securityService.getAutheticatedUserName(rootNode.path("credentials").asText());
90  String userName = getUserName(rootNode);
91  String password = getPassword(rootNode);
```

Password Management: Password in Comment — Low

Package: .com.cts.dcrowd.rest.defect.app.jira.services

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 92 (Password Management: Password in Comment) — Low

```
92  // String password =
93  // securityService.getAutheticatedPassword(rootNode.path("credentials").asText());
94
95  HttpHost targetHost = new HttpHost(rootNode.path("domainName").asText(), 8080, "http");
```

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 392 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java:392
Taint Flags:

```
389  HttpEntity entity = new StringEntity(issueObj.toString(), ContentType.APPLICATION_JSON);
390  httpPost.setEntity(entity);
391  // String encoding =
392  // Base64.encodeBase64String("vmohandass:password".getBytes());
393  // httpGet.setHeader("Authorization", "Basic " + encoding);
394
395  HttpResponse httpResponse = httpClient.execute(targetHost, httpPost, context);
```

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 93 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java:93
Taint Flags:

```
90  String userName = getUserName(rootNode);
91  String password = getPassword(rootNode);
92  // String password =
93  // securityService.getAutheticatedPassword(rootNode.path("credentials").asText());
94
95  HttpHost targetHost = new HttpHost(rootNode.path("domainName").asText(), 8080, "http");
96
```

Password Management: Password in Comment — Low

Package: .com.cts.dcrowd.rest.defect.app.jira.services

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 493 (Password Management: Password in Comment) — Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java:493
Taint Flags:

```
490  return securityService.getAutheticatedUserName(rootNode.path("credentials").asText());
491  }
492
493  /**
494  * Gets the password.
495  *
496  * @param rootNode the root node
```

Poor Error Handling: Empty Catch Block (1 issue)

Abstract

Ignoring an exception can cause the program to overlook unexpected states and conditions.

Explanation

Just about every serious attack on a software system begins with the violation of a programmer's assumptions. After the attack, the programmer's assumptions seem flimsy and poorly founded, but before an attack many programmers would defend their assumptions well past the end of their lunch break. Two dubious assumptions that are easy to spot in code are "this method call can never fail" and "it doesn't matter if this call fails". When a programmer ignores an exception, they implicitly state that they are operating under one of these assumptions. **Example 1:** The following code excerpt ignores a rarely-thrown exception from `doExchange()`.

```
try {
  doExchange();
}
catch (RareException e) {
  // this can never happen
}
```

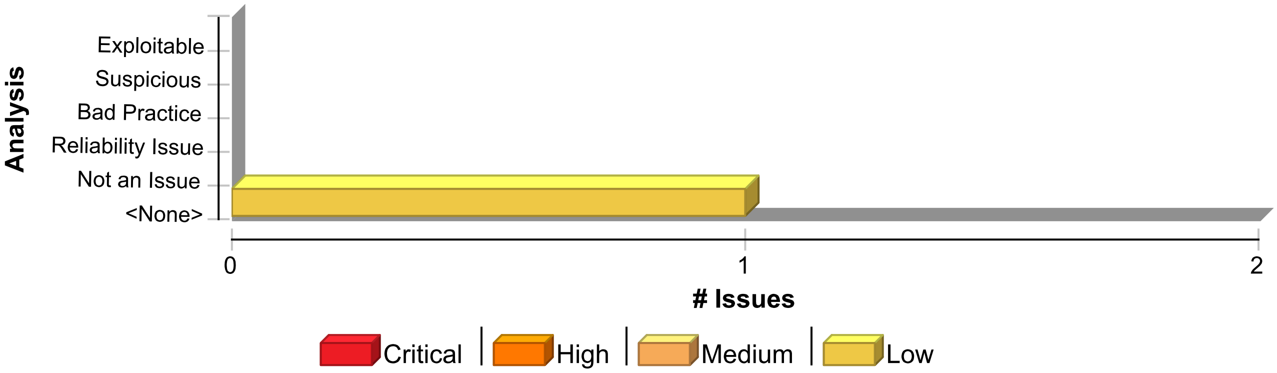
If a `RareException` were to ever be thrown, the program would continue to execute as though nothing unusual had occurred. The program records no evidence indicating the special situation, potentially frustrating any later attempt to explain the program's behavior.

Recommendation

At a minimum, log the fact that the exception was thrown so that it will be possible to come back later and make sense of the resulting program behavior. Better yet, abort the current operation. If the exception is being ignored because the caller cannot properly handle it but the context makes it inconvenient or impossible for the caller to declare that it throws the exception itself, consider throwing a `RuntimeException` or an `Error`, both of which are unchecked exceptions. As of JDK 1.4, `RuntimeException` has a constructor that makes it easy to wrap another exception. **Example 2:** The code in Example 1 could be rewritten in the following way:

```
try {
  doExchange();
}
catch (RareException e) {
  throw new RuntimeException("This can never happen", e);
}
```

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Poor Error Handling: Empty Catch Block	1	0	0	1
Total	**1**	**0**	**0**	**1**

Poor Error Handling: Empty Catch Block — Low

Package: com.cts.dcrowd.rest.app.util

src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java, line 197 (Poor Error Handling: Empty Catch Block) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: dump()
File: src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java:197
Taint Flags:

```
194  }
195  sb.append(field.getName()).append('=').append('\'')
196  .append(value).append('\'');
197  } catch (IllegalAccessException ignore) {
198  //this should never happen
199  }
200
```

Poor Error Handling: Overly Broad Catch (51 issues)

Abstract

The catch block handles a broad swath of exceptions, potentially trapping dissimilar issues or problems that should not be dealt with at this point in the program.

Explanation

Multiple catch blocks can get repetitive, but "condensing" catch blocks by catching a high-level class such as `Exception` can obscure exceptions that deserve special treatment or that should not be caught at this point in the program. Catching an overly broad exception essentially defeats the purpose of Java's typed exceptions, and can become particularly dangerous if the program grows and begins to throw new types of exceptions. The new exception types will not receive any attention. **Example:** The following code excerpt handles three types of exceptions in an identical fashion.

```
  try {
    doExchange();
  }
  catch (IOException e) {
    logger.error("doExchange failed", e);
  }
  catch (InvocationTargetException e) {
    logger.error("doExchange failed", e);
  }
  catch (SQLException e) {
    logger.error("doExchange failed", e);
  }
```

At first blush, it may seem preferable to deal with these exceptions in a single catch block, as follows:

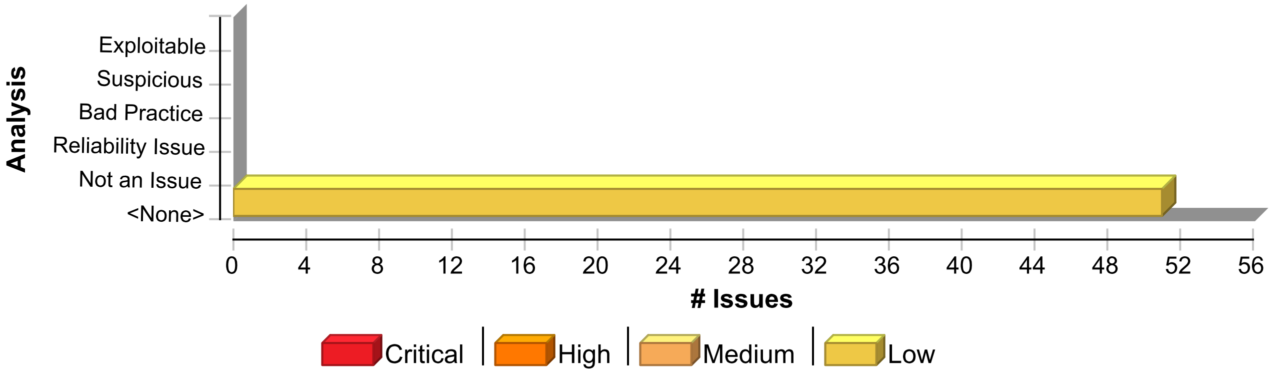
```
  try {
    doExchange();
  }
  catch (Exception e) {
    logger.error("doExchange failed", e);
  }
```

However, if `doExchange()` is modified to throw a new type of exception that should be handled in some different kind of way, the broad catch block will prevent the compiler from pointing out the situation. Further, the new catch block will now also handle exceptions derived from `RuntimeException` such as `ClassCastException`, and `NullPointerException`, which is not the programmer's intent.

Recommendation

Do not catch broad exception classes such as `Exception`, `Throwable`, `Error`, or `RuntimeException` except at the very top level of the program or thread.

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Poor Error Handling: Overly Broad Catch	51	0	0	51
Total	**51**	**0**	**0**	**51**

Poor Error Handling: Overly Broad Catch — Low

Package: .com.cts.dcrowd.rest.app.service.helper

src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java, line 380 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: accept()
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java:380
Taint Flags:

```
377   } else if(!
filterObject.getString("country_code").equals(filterByState.getAddress().getCountry())) {
378   userListAfterStateFilter.add(filterByState);
379   }
380   }catch(Exception e) {
381   e.printStackTrace();
382   }
383   }
```

src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java, line 476 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock

Poor Error Handling: Overly Broad Catch — Low

Package: .com.cts.dcrowd.rest.app.service.helper

src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java, line 476 (Poor Error Handling: Overly Broad Catch) — Low

Enclosing Method: accept()
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java:476
Taint Flags:

```
473  }
474  }
475  }
476  } catch (Exception e) {
477  e.printStackTrace();
478  }
479  }
```

Package: .com.cts.dcrowd.rest.defect.app.jira.services

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 244 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: accept()
File: src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java:244
Taint Flags:

```
241  ResponseEntity<String> response1 = restTemplate.postForEntity(serverUrl, requestEntity,
String.class);
242  System.out.println(response1);
243
244  } catch (Exception e) {
245  // TODO Auto-generated catch block
246  e.printStackTrace();
247  }
```

Package: com.cts.dcrowd.rest.app.controller

src/main/java/com/cts/dcrowd/rest/app/controller/UserController.java, line 535 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock

Poor Error Handling: Overly Broad Catch — Low

Package: com.cts.dcrowd.rest.app.controller

src/main/java/com/cts/dcrowd/rest/app/controller/UserController.java, line 535 (Poor Error Handling: Overly Broad Catch) — Low

Enclosing Method: saveProfileImage()
File: src/main/java/com/cts/dcrowd/rest/app/controller/UserController.java:535
Taint Flags:

```
532  throw new DcrowdException("Invalid Attachment");
533  }
534
535  } catch (Exception e) {
536  LOGGER.debug("DcrowdException : ", e);
537  return buildFailureResponse( 219, "Failed in UserController - "+e);
538  }
```

Package: com.cts.dcrowd.rest.app.security

src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java, line 143 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: getUserCredentialFromBlowFish()
File: src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java:143
Taint Flags:

```
140  String credential;
141  try {
142  credential = new String(decryptLoginCredential(token)).split(":")[index];
143  } catch (Exception e) {
144  credential = null;
145  LOGGER.debug("Exception : {} \n Stack Trace : {}", e.getMessage(),
String.valueOf(e.getStackTrace()));
146  }
```

src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java, line 161 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: getUsernameFromToken()
File: src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java:161

Poor Error Handling: Overly Broad Catch | Low

Package: com.cts.dcrowd.rest.app.security

src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java, line 161 (Poor Error Handling: Overly Broad Catch) | Low

Taint Flags:

```
158  try {
159  final Claims claims = this.getClaimsFromToken(token);
160  username = claims.getSubject();
161  } catch (Exception e) {
162  username = null;
163  LOGGER.debug("Exception : {} \n Stack Trace : {}", e.getMessage(),
String.valueOf(e.getStackTrace()));
164  }
```

src/main/java/com/cts/dcrowd/rest/app/security/LdapUserDetailsMapperWS.java, line 308 (Poor Error Handling: Overly Broad Catch) | Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: insertNewUserWebService()
File: src/main/java/com/cts/dcrowd/rest/app/security/LdapUserDetailsMapperWS.java:308
Taint Flags:

```
305  emailEventService.afterRegistration(user.getEmail());
306  return user;
307
308  } catch (Exception e) {
309  e.printStackTrace();
310  return null;
311  }
```

src/main/java/com/cts/dcrowd/rest/app/security/ApplicationAuthProvider.java, line 106 (Poor Error Handling: Overly Broad Catch) | Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: authenticate()
File: src/main/java/com/cts/dcrowd/rest/app/security/ApplicationAuthProvider.java:106
Taint Flags:

```
103  return new UsernamePasswordAuthenticationToken(finalUser,
104  authentication.getCredentials(), grantedAuths);
```

Poor Error Handling: Overly Broad Catch — Low

Package: com.cts.dcrowd.rest.app.security

src/main/java/com/cts/dcrowd/rest/app/security/ApplicationAuthProvider.java, line 106 (Poor Error Handling: Overly Broad Catch) — Low

```
105  }
106  catch (Exception e) {
107  e.printStackTrace();
108  return null;
109  }
```

src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java, line 215 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: getAudienceFromToken()
File: src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java:215
Taint Flags:

```
212  try {
213  final Claims claims = this.getClaimsFromToken(token);
214  audience = (String) claims.get("audience");
215  } catch (Exception e) {
216  audience = null;
217  LOGGER.debug("Exception : {} \n Stack Trace : {}", e.getMessage(),
String.valueOf(e.getStackTrace()));
218  }
```

src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java, line 368 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: refreshToken()
File: src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java:368
Taint Flags:

```
365  final Claims claims = this.getClaimsFromToken(token);
366  claims.put("created", this.generateCurrentDate());
367  refreshedToken = this.generateToken(claims);
368  } catch (Exception e) {
369  refreshedToken = null;
370  LOGGER.debug("Exception : {} \n Stack Trace : {}", e.getMessage(),
```

Poor Error Handling: Overly Broad Catch — Low

Package: com.cts.dcrowd.rest.app.security

src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java, line 368 (Poor Error Handling: Overly Broad Catch) — Low

```
String.valueOf(e.getStackTrace()));
371  }
```

src/main/java/com/cts/dcrowd/rest/app/security/LdapUserDetailsMapperWS.java, line 102 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: loadUserByUsername()
File: src/main/java/com/cts/dcrowd/rest/app/security/LdapUserDetailsMapperWS.java:102
Taint Flags:

```
99  if (Util.isEmpty(user)) {
100  user = insertNewUserWebService(username);
101  }
102  } catch (Exception e) {
103  e.printStackTrace();
104  return null;
105  }
```

src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java, line 179 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: getCreatedDateFromToken()
File: src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java:179
Taint Flags:

```
176  try {
177  final Claims claims = this.getClaimsFromToken(token);
178  created = new Date((Long) claims.get("created"));
179  } catch (Exception e) {
180  created = null;
181  LOGGER.debug("Exception : {} \n Stack Trace : {}", e.getMessage(),
String.valueOf(e.getStackTrace()));
182  }
```

Poor Error Handling: Overly Broad Catch Low

Package: com.cts.dcrowd.rest.app.security

src/main/java/com/cts/dcrowd/rest/app/security/ApplicationAuthProvider.java, line 166 (Poor Error Handling: Overly Broad Catch) Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: authenticateEmployee()
File: src/main/java/com/cts/dcrowd/rest/app/security/ApplicationAuthProvider.java:166
Taint Flags:

```
163  return false;
164  }
165
166  } catch (Exception e) {
167  LOGGER.debug("Exception on connectExternalUser catch block : {} \n Stack Trace : {}", e.getMessage(), String.valueOf(e.getStackTrace()));
168  } finally {
169  try {
```

src/main/java/com/cts/dcrowd/rest/app/security/ApplicationAuthProvider.java, line 172 (Poor Error Handling: Overly Broad Catch) Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: authenticateEmployee()
File: src/main/java/com/cts/dcrowd/rest/app/security/ApplicationAuthProvider.java:172
Taint Flags:

```
169  try {
170  requestFactory.destroy();
171  HttpClientUtils.closeQuietly(httpClient);
172  } catch (Exception e) {
173  // TODO Auto-generated catch block
174  LOGGER.debug("Exception on authenticateEmployee finally block : {} \n Stack Trace : {}", e.getMessage(), String.valueOf(e.getStackTrace()));
175  }
```

src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java, line 126 (Poor Error Handling: Overly Broad Catch) Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Poor Error Handling: Overly Broad Catch — Low

Package: com.cts.dcrowd.rest.app.security

src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java, line 126 (Poor Error Handling: Overly Broad Catch) — Low

Sink Details

Sink: CatchBlock
Enclosing Method: getPasswordFromBase64Token()
File: src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java:126
Taint Flags:

```
123  //String base64Credentials = token.substring(BASIC.length()).trim();
124  try {
125  password = new String(Base64.getDecoder().decode(token),UTF8).split(">:<")[1];
126  } catch (Exception e) {
127  password = null;
128  LOGGER.debug("Exception : {} \n Stack Trace : {}", e.getMessage(),
String.valueOf(e.getStackTrace()));
129  }
```

src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java, line 85 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: validateBasicAuthToken()
File: src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java:85
Taint Flags:

```
82  String[] basicAuthString = new String(Base64.getDecoder().decode(token),
UTF8).split(">:<");
83  //String[] basicAuthString = new String(decryptLoginCredential(token)).split(":");
84  isValid = (basicAuthString.length == 2) ? true : false;
85  } catch (Exception e) {
86  isValid = false;
87  LOGGER.debug("UnsupportedEncodingException : {} \n Stack Trace : {}", e.getMessage(),
String.valueOf(e.getStackTrace()));
88  }
```

src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java, line 197 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Poor Error Handling: Overly Broad Catch — Low

Package: com.cts.dcrowd.rest.app.security

src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java, line 197 (Poor Error Handling: Overly Broad Catch) — Low

Sink: CatchBlock
Enclosing Method: getExpirationDateFromToken()
File: src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java:197
Taint Flags:

```
194  try {
195  final Claims claims = this.getClaimsFromToken(token);
196  expiration = claims.getExpiration();
197  } catch (Exception e) {
198  expiration = null;
199  LOGGER.debug("Exception : {} \n Stack Trace : {}", e.getMessage(),
String.valueOf(e.getStackTrace()));
200  }
```

src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java, line 433 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: decryptLoginCredential()
File: src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java:433
Taint Flags:

```
430  byte[] decrypted = cipher.doFinal(blowFishCredentials.getBytes());
431  userCredential = new String(decrypted);
432  }
433  catch (Exception e) {
434  userCredential = null;
435  LOGGER.debug("Exception : {} \n Stack Trace : {}", e.getMessage(),
String.valueOf(e.getStackTrace()));
436  }
```

src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java, line 106 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: getUsernameFromBase64Token()
File: src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java:106

Poor Error Handling: Overly Broad Catch — Low

Package: com.cts.dcrowd.rest.app.security

src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java, line 106 (Poor Error Handling: Overly Broad Catch) — Low

Taint Flags:

```
103  //String base64Credentials = token.substring(BASIC.length()).trim();
104  try {
105  username = new String(Base64.getDecoder().decode(token), UTF8).split(">:<")[0];
106  } catch (Exception e) {
107  username = null;
108  LOGGER.debug("Exception : {} \n Stack Trace : {}", e.getMessage(),
String.valueOf(e.getStackTrace()));
109  }
```

src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java, line 408 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: getDatabaseName()
File: src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java:408
Taint Flags:

```
405  try {
406  final Claims claims = this.getClaimsFromToken(token);
407  dbname = claims.getIssuer();
408  } catch (Exception e) {
409  dbname = null;
410  LOGGER.debug("Exception : {} \n Stack Trace : {}", e.getMessage(),
String.valueOf(e.getStackTrace()));
411  }
```

src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java, line 235 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: getClaimsFromToken()
File: src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java:235
Taint Flags:

```
232  .setSigningKey(this.secret)
```

Poor Error Handling: Overly Broad Catch — Low

Package: com.cts.dcrowd.rest.app.security

src/main/java/com/cts/dcrowd/rest/app/security/TokenUtils.java, line 235 (Poor Error Handling: Overly Broad Catch) — Low

```
233  .parseClaimsJws(token)
234  .getBody();
235  } catch (Exception e) {
236  claims = null;
237  LOGGER.debug("Exception : {} \n Stack Trace : {}", e.getMessage(),
String.valueOf(e.getStackTrace()));
238  }
```

Package: com.cts.dcrowd.rest.app.security.service.impl

src/main/java/com/cts/dcrowd/rest/app/security/service/impl/SecurityServiceImpl.java, line 103 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: doAuthenticate()
File: src/main/java/com/cts/dcrowd/rest/app/security/service/impl/SecurityServiceImpl.java:103
Taint Flags:

```
100  );
101
102  SecurityContextHolder.getContext().setAuthentication(authentication);
103  } catch(Exception e) {
104  e.printStackTrace();
105  return false;
106  }
```

Package: com.cts.dcrowd.rest.app.service

src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java, line 202 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: moderatorsInvitationForCampaigns()
File: src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java:202
Taint Flags:

```
199  return false;
200  }
```

Poor Error Handling: Overly Broad Catch — Low

Package: com.cts.dcrowd.rest.app.service

src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java, line 202 (Poor Error Handling: Overly Broad Catch) — Low

```
201
202   } catch (Exception e) {
203   e.printStackTrace();
204   return false;
205   }
```

src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java, line 237 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: afterRegistration()
File: src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java:237
Taint Flags:

```
234   try {
235   EmailVO emailConf = emailService.createEmail(configuration,false);
236   emailService.sendEmail(emailConf);
237   } catch (Exception e) {
238   e.printStackTrace();
239   }
240   }
```

src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java, line 405 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: externalCustomerRegistered()
File: src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java:405
Taint Flags:

```
402   try {
403   EmailVO emailConf = emailService.createEmail(configuration,false);
404   emailService.sendEmail(emailConf);
405   } catch (Exception e) {
406   e.printStackTrace();
407   }
```

Poor Error Handling: Overly Broad Catch — Low

Package: com.cts.dcrowd.rest.app.service

src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java, line 405 (Poor Error Handling: Overly Broad Catch) — Low

```
408   }
```

src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java, line 135 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: afterInvitedTester()
File: src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java:135
Taint Flags:

```
132   EmailVO emailConf = emailService.createEmail(configuration,false);
133   emailService.sendEmail(emailConf);
134   return true;
135   } catch (Exception e) {
136   e.printStackTrace();
137   return false;
138   }
```

src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java, line 316 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: campaignEmail()
File: src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java:316
Taint Flags:

```
313   toIndex = Math.min(toIndex + 100 , email.size());
314   }
315   return true;
316   } catch (Exception e) {
317   e.printStackTrace();
318   return false;
319   }
```

Poor Error Handling: Overly Broad Catch — Low

Package: com.cts.dcrowd.rest.app.service

src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java, line 427 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: userAssignedMail()
File: src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java:427
Taint Flags:

```
424  try {
425  EmailVO emailConf = emailService.createEmail(configuration,false);
426  emailService.sendEmail(emailConf);
427  } catch (Exception e) {
428  e.printStackTrace();
429  }
430  }
```

src/main/java/com/cts/dcrowd/rest/app/service/AddressService.java, line 51 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: getAddressIdByStreetAddress()
File: src/main/java/com/cts/dcrowd/rest/app/service/AddressService.java:51
Taint Flags:

```
48  try{
49  return addressRepository.findByOfficeAddress(streetAddress);
50  }
51  catch(Exception ex)
52  {
53  return null;
54  }
```

src/main/java/com/cts/dcrowd/rest/app/service/CampaignService.java, line 996 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Poor Error Handling: Overly Broad Catch — Low

Package: com.cts.dcrowd.rest.app.service

src/main/java/com/cts/dcrowd/rest/app/service/CampaignService.java, line 996 (Poor Error Handling: Overly Broad Catch) — Low

Sink Details

Sink: CatchBlock
Enclosing Method: confirmTeam()
File: src/main/java/com/cts/dcrowd/rest/app/service/CampaignService.java:996
Taint Flags:

```
993  try {
994  user = userService.findByUserId(userId);
995  userFlag = true;
996  } catch (Exception e) {
997  externalUser = externalUserRepository.findOne(userId);
998  }
999  if (userFlag) {
```

src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java, line 344 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: fastesterRegistration()
File: src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java:344
Taint Flags:

```
341  try {
342  EmailVO emailConf = emailService.createEmail(configuration,false);
343  emailService.sendEmail(emailConf);
344  } catch (Exception e) {
345  e.printStackTrace();
346  }
347  }
```

src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java, line 168 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: inviteExternalModerators()
File: src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java:168

Poor Error Handling: Overly Broad Catch — Low

Package: com.cts.dcrowd.rest.app.service

src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java, line 168 (Poor Error Handling: Overly Broad Catch) — Low

Taint Flags:

```
165  EmailVO emailConf = emailService.createEmail(configuration,false);
166  emailService.sendEmail(emailConf);
167  return true;
168  } catch (Exception e) {
169  e.printStackTrace();
170  return false;
171  }
```

src/main/java/com/cts/dcrowd/rest/app/service/UserService.java, line 234 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: setUserProfile()
File: src/main/java/com/cts/dcrowd/rest/app/service/UserService.java:234
Taint Flags:

```
231  if(!Util.isEmpty(user)){
232  user.setUserProfile(configureAndGetUserProfile(user));
233  }
234  }catch(Exception e)
235  {
236  e.printStackTrace();
237  }
```

src/main/java/com/cts/dcrowd/rest/app/service/EmailService.java, line 136 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: createEmail()
File: src/main/java/com/cts/dcrowd/rest/app/service/EmailService.java:136
Taint Flags:

```
133  }
134  return email;
135
```

Poor Error Handling: Overly Broad Catch — Low

Package: com.cts.dcrowd.rest.app.service

src/main/java/com/cts/dcrowd/rest/app/service/EmailService.java, line 136 (Poor Error Handling: Overly Broad Catch) — Low

```
136  } catch (Exception e) {
137  e.printStackTrace();
138
139  }
```

src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java, line 267 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: defectStatusChangeEmail()
File: src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java:267
Taint Flags:

```
264  try {
265  EmailVO emailConf = emailService.createEmail(configuration,false);
266  emailService.sendEmail(emailConf);
267  } catch (Exception e) {
268  e.printStackTrace();
269  }
270  }
```

src/main/java/com/cts/dcrowd/rest/app/service/ProfileImageService.java, line 108 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: saveImage()
File: src/main/java/com/cts/dcrowd/rest/app/service/ProfileImageService.java:108
Taint Flags:

```
105  }
106  Util.writeFileToFileServer(imageFile.getBytes(), fileDIR, finalFilePath, false);
107  userService.createUpdateUser(user);
108  } catch (Exception e) {
109  e.printStackTrace();
110
111  }
```

Poor Error Handling: Overly Broad Catch — Low

Package: com.cts.dcrowd.rest.app.service

src/main/java/com/cts/dcrowd/rest/app/service/EmailService.java, line 243 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: getVelocityEngineTemplateContent()
File: src/main/java/com/cts/dcrowd/rest/app/service/EmailService.java:243
Taint Flags:

```
240
  .processTemplateIntoString( freeMarkerConfiguration.getTemplate(configuration.getTemplateID())
241   content.append(VelocityEngineUtils.mergeTemplateIntoString(velocityEngine,
configuration.getTemplateID(),configuration.getParameters()));
242   return content.toString();
243   }catch(Exception e){
244   LOGGER.debug("Inside getVelocityEngineTemplateContent method : error"+e.getMessage());
245   }
246   return "";
```

Package: com.cts.dcrowd.rest.app.service.helper

src/main/java/com/cts/dcrowd/rest/app/service/helper/LdapWebserviceHelper.java, line 88 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: extractLdapValueMap()
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/LdapWebserviceHelper.java:88
Taint Flags:

```
85  try {
86  requestFactory.destroy();
87  HttpClientUtils.closeQuietly(httpClient);
88  } catch (Exception e) {
89  // TODO Auto-generated catch block
90  LOGGER.debug("Exception on connectExternalUser finally block : {} \n Stack Trace : {}",
e.getMessage(), String.valueOf(e.getStackTrace()));
91  }
```

Poor Error Handling: Overly Broad Catch | Low

Package: com.cts.dcrowd.rest.app.service.helper

src/main/java/com/cts/dcrowd/rest/app/service/helper/ApplicationQueryProcessor.java, line 421 (Poor Error Handling: Overly Broad Catch) | Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: getUserBadgeToGenerateReport()
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/ApplicationQueryProcessor.java:421
Taint Flags:

```
418  List<BadgeReport> badgeReport = output.getMappedResults();
419  return badgeReport;
420  }
421  catch(Exception e)
422  {
423  e.printStackTrace();
424  }
```

Package: com.cts.dcrowd.rest.app.util

src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java, line 81 (Poor Error Handling: Overly Broad Catch) | Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: generateBlowFishEncryptValue()
File: src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java:81
Taint Flags:

```
78  cipher.init(Cipher.ENCRYPT_MODE, skeyspec);
79  byte[] encrypted=cipher.doFinal(sourceValue.getBytes());
80  strData = new String(Base64.getEncoder().encode(encrypted));
81  } catch (Exception e) {
82  e.printStackTrace();
83  }
84  return strData;
```

src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java, line 145 (Poor Error Handling: Overly Broad Catch) | Low

Issue Details

Kingdom: Errors

Poor Error Handling: Overly Broad Catch — Low

Package: com.cts.dcrowd.rest.app.util

src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java, line 145 (Poor Error Handling: Overly Broad Catch) — Low

Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: readJsonObject()
File: src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java:145
Taint Flags:

```
142  Object obj = parser.parse(fileReader);
143  JSONObject jsonObject = (JSONObject) obj;
144  return jsonObject;
145  } catch (Exception e) {
146  return null;
147  }
148  }
```

src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java, line 61 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: base64Decoder()
File: src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java:61
Taint Flags:

```
58  String sourceValue = "";
59  try {
60  sourceValue = new String(Base64.getDecoder().decode(encodedValue), "utf-8");
61  } catch (Exception e) {
62  e.printStackTrace();
63  }
64  return sourceValue;
```

src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java, line 103 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock

Poor Error Handling: Overly Broad Catch — Low

Package: com.cts.dcrowd.rest.app.util

src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java, line 103 (Poor Error Handling: Overly Broad Catch) — Low

Enclosing Method: generateBlowFishDecryptValue()
File: src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java:103
Taint Flags:

```
100  byte[] decrypted = cipher.doFinal(Base64.getDecoder().decode(encryptedValue));
101  strData = new String(decrypted);
102  System.out.println(strData);
103  } catch (Exception e) {
104  e.printStackTrace();
105  }
106  return strData;
```

Package: com.cts.dcrowd.rest.app.utils

src/main/java/com/cts/dcrowd/rest/app/utils/Util.java, line 508 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: generateBlowFishEncryptValue()
File: src/main/java/com/cts/dcrowd/rest/app/utils/Util.java:508
Taint Flags:

```
505  byte[] encrypted=cipher.doFinal(sourceValue.getBytes());
506  //strData = base64Encoder(encrypted);
507  strData = encrypted.toString();
508  } catch (Exception e) {
509  e.printStackTrace();
510  }
511  return strData;
```

src/main/java/com/cts/dcrowd/rest/app/utils/Util.java, line 531 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: generateBlowFishDecryptValue()
File: src/main/java/com/cts/dcrowd/rest/app/utils/Util.java:531
Taint Flags:

Poor Error Handling: Overly Broad Catch | Low

Package: com.cts.dcrowd.rest.app.utils

src/main/java/com/cts/dcrowd/rest/app/utils/Util.java, line 531 (Poor Error Handling: Overly Broad Catch) | Low

```
528  String[] basicAuthString = null;
529  try {
530  basicAuthString = new String(Base64.getDecoder().decode(encryptedValue),
StandardCharsets.UTF_8).split("fastestCustomers");
531  } catch (Exception e) {
532  e.printStackTrace();
533  }
534  return basicAuthString!=null && basicAuthString.length != 0 ? basicAuthString[0] : null;
```

src/main/java/com/cts/dcrowd/rest/app/utils/Util.java, line 559 (Poor Error Handling: Overly Broad Catch) | Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: base64Decoder()
File: src/main/java/com/cts/dcrowd/rest/app/utils/Util.java:559
Taint Flags:

```
556  String sourceValue = "";
557  try {
558  sourceValue = new String(Base64.getDecoder().decode(encodedValue), "utf-8");
559  } catch (Exception e) {
560  LOGGER.debug("UnsupportedEncodingException : {} \n Stack Trace : {}", e.getMessage(),
String.valueOf(e.getStackTrace()));
561  }
562  return sourceValue;
```

src/main/java/com/cts/dcrowd/rest/app/utils/StopwordUtility.java, line 36 (Poor Error Handling: Overly Broad Catch) | Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: StopwordUtility()
File: src/main/java/com/cts/dcrowd/rest/app/utils/StopwordUtility.java:36
Taint Flags:

```
33  private StopwordUtility(){
34  try {
35  loadStopList();
```

Poor Error Handling: Overly Broad Catch

Low

Package: com.cts.dcrowd.rest.app.utils

src/main/java/com/cts/dcrowd/rest/app/utils/StopwordUtility.java, line 36 (Poor Error Handling: Overly Broad Catch)

Low

```
36  } catch (Exception e) {
37  e.printStackTrace();
38  }
39  }
```

src/main/java/com/cts/dcrowd/rest/app/utils/Util.java, line 749 (Poor Error Handling: Overly Broad Catch)

Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: decryptLoginCredential()
File: src/main/java/com/cts/dcrowd/rest/app/utils/Util.java:749
Taint Flags:

```
746  byte[] decrypted = cipher.doFinal(blowFishCredentials.getBytes());
747  userCredential = new String(decrypted);
748  }
749  catch (Exception e) {
750  userCredential = null;
751  LOGGER.debug("Exception : {} \n Stack Trace : {}", e.getMessage(),
String.valueOf(e.getStackTrace()));
752  }
```

src/main/java/com/cts/dcrowd/rest/app/utils/Util.java, line 653 (Poor Error Handling: Overly Broad Catch)

Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: connectExternalUser()
File: src/main/java/com/cts/dcrowd/rest/app/utils/Util.java:653
Taint Flags:

```
650  try {
651  requestFactory.destroy();
652  HttpClientUtils.closeQuietly(httpClient);
653  } catch (Exception e) {
654  // TODO Auto-generated catch block
655  LOGGER.debug("Exception on connectExternalUser finally block : {} \n Stack Trace : {}",
e.getMessage(), String.valueOf(e.getStackTrace()));
```

Poor Error Handling: Overly Broad Catch — Low

Package: com.cts.dcrowd.rest.app.utils

src/main/java/com/cts/dcrowd/rest/app/utils/Util.java, line 653 (Poor Error Handling: Overly Broad Catch) — Low

```
656  }
```

src/main/java/com/cts/dcrowd/rest/app/utils/EmailDomainFilterUtility.java, line 34 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: EmailDomainFilterUtility()
File: src/main/java/com/cts/dcrowd/rest/app/utils/EmailDomainFilterUtility.java:34
Taint Flags:

```
31  private EmailDomainFilterUtility(){
32  try {
33  loadDomainList();
34  } catch (Exception e) {
35  e.printStackTrace();
36  }
37  }
```

Package: com.cts.dcrowd.rest.defect.app.services.impl

src/main/java/com/cts/dcrowd/rest/defect/app/services/impl/DefaultDefectServiceImpl.java, line 234 (Poor Error Handling: Overly Broad Catch) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: modifyDefect()
File: src/main/java/com/cts/dcrowd/rest/defect/app/services/impl/DefaultDefectServiceImpl.java:234
Taint Flags:

```
231  emailEventService.defectStatusChangeEmail(recieverEmailId, defectEntryFromReq);
232  // Set is email flag to true
233  defectEntryFromReq.setIsEmailSentForRejectOrOnhold(true);
234  } catch(Exception exp) {
235  defectEntryFromReq.setIsEmailSentForRejectOrOnhold(false);
236  LOGGER.error("UndeliverableEmailException : Email not sent - Reason below \n" +
exp.getMessage());
237  }
```

Poor Error Handling: Overly Broad Catch	Low
Package: com.cts.dcrowd.rest.defect.app.services.impl	
src/main/java/com/cts/dcrowd/rest/defect/app/services/impl/DefaultDefectServiceImpl.java, line 234 (Poor Error Handling: Overly Broad Catch)	Low

Poor Error Handling: Overly Broad Throws (57 issues)

Abstract

The method throws a generic exception making it harder for callers to do a good job of error handling and recovery.

Explanation

Declaring a method to throw `Exception` or `Throwable` makes it difficult for callers to do good error handling and error recovery. Java's exception mechanism is set up to make it easy for callers to anticipate what can go wrong and write code to handle each specific exceptional circumstance. Declaring that a method throws a generic form of exception defeats this system. **Example:** The following method throws three types of exceptions.

```
public void doExchange()
  throws IOException, InvocationTargetException,
         SQLException {
  ...
}
```

While it might seem tidier to write

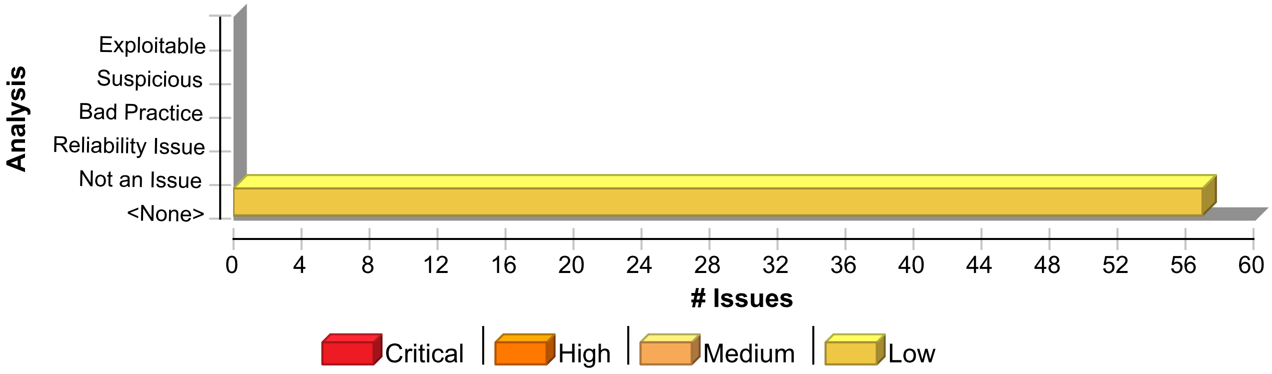
```
public void doExchange()
  throws Exception {
  ...
}
```

doing so hampers the caller's ability to understand and handle the exceptions that occur. Further, if a later revision of `doExchange()` introduces a new type of exception that should be treated differently than previous exceptions, there is no easy way to enforce this requirement.

Recommendation

Do not declare methods to throw `Exception` or `Throwable`. If the exceptions thrown by a method are not recoverable or should not generally be caught by the caller, consider throwing unchecked exceptions rather than checked exceptions. This can be accomplished by implementing exception classes that extend `RuntimeException` or `Error` instead of `Exception`, or add a try/catch wrapper in your method to convert checked exceptions to unchecked exceptions.

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Poor Error Handling: Overly Broad Throws	57	0	0	57
Total	**57**	**0**	**0**	**57**

Poor Error Handling: Overly Broad Throws — Low

Package: com.cts.dcrowd.rest.app.repository

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 126 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testgetNominatedCampaignOfUser
Enclosing Method: testgetNominatedCampaignOfUser()
File: src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java:126
Taint Flags:

```
123  * @throws Exception the exception
124  */
125  @Test
126  public void testgetNominatedCampaignOfUser() throws Exception {
127
128  mockMvc.perform(get("http://localhost:8007/api/v1/crowd/campaigns/nominatedCampaign")
129  .header("X_Auth_Token", user.getAuthToken())
```

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 289 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testsaveFeedback
Enclosing Method: testsaveFeedback()
File: src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java:289
Taint Flags:

```
286  * @throws Exception the exception
287  */
288  @Test
289  public void testsaveFeedback() throws Exception {
290  JSONObject jsonObj = TestUtil.readJsonObject(new File("src/test/resources/
CampaignJson.txt"));
291  String feedback = jsonObj.get("feedback").toString();
292  mockMvc.perform(post("http://localhost:8007/api/v1/crowd/campaigns/moderator/
```

Poor Error Handling: Overly Broad Throws

Low

Package: com.cts.dcrowd.rest.app.repository

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 289 (Poor Error Handling: Overly Broad Throws)

Low

```
saveFeedback/"+campaign.get_id())
```

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 157 (Poor Error Handling: Overly Broad Throws)

Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testsaveCampaign
Enclosing Method: testsaveCampaign()
File: src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java:157
Taint Flags:

```
154  *
155  * @throws Exception the exception
156  */
157  public void testsaveCampaign() throws Exception {
158
159  JSONObject jsonObj = TestUtil.readJsonObject(new File("src/test/resources/
CampaignJson.txt"));
160  String campaignJson = jsonObj.get("campaign").toString();
```

src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java, line 343 (Poor Error Handling: Overly Broad Throws)

Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testgetRedeemProduct
Enclosing Method: testgetRedeemProduct()
File: src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java:343
Taint Flags:

```
340  * @throws Exception the exception
341  */
342  @Test
343  public void testgetRedeemProduct() throws Exception {
344  mockMvc.perform(get("http://localhost:8007/api/v1/crowd/users/redeem/view/1")
345  .header("X_Auth_Token", user1.getAuthToken())
346  .contentType(MediaType.APPLICATION_JSON)).
```

Poor Error Handling: Overly Broad Throws — Low

Package: com.cts.dcrowd.rest.app.repository

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 275 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testcompleteCampaign
Enclosing Method: testcompleteCampaign()
File: src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java:275
Taint Flags:

```
272  * @throws Exception the exception
273  */
274  @Test
275  public void testcompleteCampaign() throws Exception {
276  mockMvc.perform(post("http://localhost:8007/api/v1/crowd/campaigns/
completeCampaign/"+campaign.get_id())
277  .header("X_Auth_Token", user.getAuthToken())
278  .contentType(MediaType.APPLICATION_JSON)).
```

src/test/java/com/cts/dcrowd/rest/app/repository/BadgeControllerTestNG.java, line 78 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: setup
Enclosing Method: setup()
File: src/test/java/com/cts/dcrowd/rest/app/repository/BadgeControllerTestNG.java:78
Taint Flags:

```
75  * @throws Exception the exception
76  */
77  @BeforeClass
78  public void setup() throws Exception {
79  this.mockMvc = MockMvcBuilders.webAppContextSetup(this.wac).build();
80  mockMvc.perform(post("http://localhost:8007/api/v1/crowd/
login").header("Authorization",TestUtil.convertBase64("602937", "Mar@2017"))
81  .contentType(MediaType.APPLICATION_JSON)).andExpect(status().isOk());
```

src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java, line 171 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors

Poor Error Handling: Overly Broad Throws — Low

Package: com.cts.dcrowd.rest.app.repository

src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java, line 171 (Poor Error Handling: Overly Broad Throws) — Low

Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testfindAllUsers
Enclosing Method: testfindAllUsers()
File: src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java:171
Taint Flags:

```
168  * @throws Exception the exception
169  */
170  @Test
171  public void testfindAllUsers() throws Exception {
172  mockMvc.perform(get("http://localhost:8007/api/v1/crowd/users/admin/allUsersList/1/2")
173  .header("X_Auth_Token", user1.getAuthToken())
174  .contentType(MediaType.APPLICATION_JSON))
```

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 305 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testgetFeedback
Enclosing Method: testgetFeedback()
File: src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java:305
Taint Flags:

```
302  * @throws Exception the exception
303  */
304  @Test
305  public void testgetFeedback() throws Exception {
306  mockMvc.perform(get("http://localhost:8007/api/v1/crowd/campaigns/moderator/
getFeedback/"+campaign.get_id()+"/"+user.get_id())
307  .header("X_Auth_Token", user.getAuthToken())
308  .contentType(MediaType.APPLICATION_JSON)).
```

src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java, line 329 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Poor Error Handling: Overly Broad Throws — Low

Package: com.cts.dcrowd.rest.app.repository

src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java, line 329 (Poor Error Handling: Overly Broad Throws) — Low

Sink Details

Sink: Function: testuserRedemption
Enclosing Method: testuserRedemption()
File: src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java:329
Taint Flags:

```
326  * @throws Exception the exception
327  */
328  @Test
329  public void testuserRedemption() throws Exception {
330  mockMvc.perform(post("http://localhost:8007/api/v1/crowd/users/redemption/purchase/1")
331  .header("X_Auth_Token", user1.getAuthToken())
332  .contentType(MediaType.APPLICATION_JSON)).
```

src/test/java/com/cts/dcrowd/rest/app/repository/BadgeControllerTestNG.java, line 176 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testgetBadgeImageById
Enclosing Method: testgetBadgeImageById()
File: src/test/java/com/cts/dcrowd/rest/app/repository/BadgeControllerTestNG.java:176
Taint Flags:

```
173  * @throws Exception the exception
174  */
175  @Test
176  public void testgetBadgeImageById() throws Exception
177  {
178  mockMvc.perform(get("http://localhost:8007/api/v1/crowd/badge/badgeImage/
download/"+badge.get_id())
179  .header("X_Auth_Token", user.getAuthToken())
```

src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java, line 200 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testuserProfileInfo
Enclosing Method: testuserProfileInfo()
File: src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java:200

Poor Error Handling: Overly Broad Throws — Low

Package: com.cts.dcrowd.rest.app.repository

src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java, line 200 (Poor Error Handling: Overly Broad Throws) — Low

Taint Flags:

```
197  * @throws Exception the exception
198  */
199  @Test
200  public void testuserProfileInfo() throws Exception {
201  mockMvc.perform(get("http://localhost:8007/api/v1/crowd/users/userProfile/"+user1.get_id())
202  .header("X_Auth_Token", user1.getAuthToken())
203  .contentType(MediaType.APPLICATION_JSON)).
```

src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java, line 257 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testgetUserBadge
Enclosing Method: testgetUserBadge()
File: src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java:257
Taint Flags:

```
254  * @throws Exception the exception
255  */
256  @Test
257  public void testgetUserBadge() throws Exception {
258  mockMvc.perform(get("http://localhost:8007/api/v1/crowd/users/badge/"+user1.get_id())
259  .header("X_Auth_Token", user1.getAuthToken())
260  .contentType(MediaType.APPLICATION_JSON)).
```

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 197 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testnominateCampaign
Enclosing Method: testnominateCampaign()
File: src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java:197
Taint Flags:

```
194  * @throws Exception the exception
```

Poor Error Handling: Overly Broad Throws | Low

Package: com.cts.dcrowd.rest.app.repository

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 197 (Poor Error Handling: Overly Broad Throws) | Low

```
195  */
196  @Test
197  public void testnominateCampaign() throws Exception {
198
199  mockMvc.perform(post("http://localhost:8007/api/v1/crowd/campaigns/
nominateCampaign/"+campaign.get_id())
200  .header("X_Auth_Token", user.getAuthToken())
```

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 228 (Poor Error Handling: Overly Broad Throws) | Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testcloseNomination
Enclosing Method: testcloseNomination()
File: src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java:228
Taint Flags:

```
225  * @throws Exception the exception
226  */
227  @Test
228  public void testcloseNomination() throws Exception {
229  mockMvc.perform(post("http://localhost:8007/api/v1/crowd/campaigns/
closeNomination/"+campaign.get_id())
230  .header("X_Auth_Token", user.getAuthToken())
231  .contentType(MediaType.APPLICATION_JSON)).
```

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 177 (Poor Error Handling: Overly Broad Throws) | Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testopenForNomination
Enclosing Method: testopenForNomination()
File: src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java:177
Taint Flags:

```
174  * @throws Exception the exception
```

Poor Error Handling: Overly Broad Throws — Low

Package: com.cts.dcrowd.rest.app.repository

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 177 (Poor Error Handling: Overly Broad Throws) — Low

```
175  */
176  @Test
177  public void testopenForNomination() throws Exception {
178  ObjectMapper mapper = new ObjectMapper();
179  String createdCampaign = mapper.writeValueAsString(campaign);
180  String openForNomination = "{ \"param\": " +createdCampaign+"}";
```

src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java, line 242 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testLocationgetTopPerformers
Enclosing Method: testLocationgetTopPerformers()
File: src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java:242
Taint Flags:

```
239  * @throws Exception the exception
240  */
241  @Test
242  public void testLocationgetTopPerformers() throws Exception {
243  City cityId = masterService.getCityIdByName("coimbatore");
244  mockMvc.perform(get("http://localhost:8007/api/v1/crowd/users/
topLocationPerformers/"+cityId+"/1/2")
245  .header("X_Auth_Token", user1.getAuthToken())
```

src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java, line 271 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testaddUserBadge
Enclosing Method: testaddUserBadge()
File: src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java:271
Taint Flags:

```
268  * @throws Exception the exception
269  */
270  @Test
271  public void testaddUserBadge() throws Exception {
```

Poor Error Handling: Overly Broad Throws — Low

Package: com.cts.dcrowd.rest.app.repository

src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java, line 271 (Poor Error Handling: Overly Broad Throws) — Low

```
272  mockMvc.perform(post("http://localhost:8007/api/v1/crowd/users/moderator/addBadge/"+user1.get_id()+"/1")
273  .header("X_Auth_Token", user1.getAuthToken())
274  .contentType(MediaType.APPLICATION_JSON)).
```

src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java, line 127 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testHandleLogin
Enclosing Method: testHandleLogin()
File: src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java:127
Taint Flags:

```
124  * @throws Exception the exception
125  */
126  @Test
127  public void testHandleLogin() throws Exception {
128
129  mockMvc.perform(post("http://localhost:8007/api/v1/crowd/login").header("Authorization",TestUtil.convertBase64(user1.getUsername(), "Mar@2017"))
130  .contentType(MediaType.APPLICATION_JSON)).andExpect(status().isOk());
```

src/test/java/com/cts/dcrowd/rest/app/repository/BadgeControllerTestNG.java, line 114 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testremoveBadge
Enclosing Method: testremoveBadge()
File: src/test/java/com/cts/dcrowd/rest/app/repository/BadgeControllerTestNG.java:114
Taint Flags:

```
111  * @throws Exception the exception
112  */
113  @Test
114  public void testremoveBadge() throws Exception
115  {
116  mockMvc.perform(post("http://localhost:8007/api/v1/crowd/admin/badge/remove/"+badge.getBadgeId())
```

Poor Error Handling: Overly Broad Throws — Low

Package: com.cts.dcrowd.rest.app.repository

src/test/java/com/cts/dcrowd/rest/app/repository/BadgeControllerTestNG.java, line 114 (Poor Error Handling: Overly Broad Throws) — Low

```
117  .header("X_Auth_Token", user.getAuthToken())
```

src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java, line 141 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testgetUserDetailsById
Enclosing Method: testgetUserDetailsById()
File: src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java:141
Taint Flags:

```
138  * @throws Exception the exception
139  */
140  @Test
141  public void testgetUserDetailsById() throws Exception {
142  //String id = user1.get_id();
143  mockMvc.perform(get("http://localhost:8007/api/v1/crowd/users/userDetailbyId/"+user1.get_id())
144  .header("X_Auth_Token", user1.getAuthToken())
```

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 333 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testgetCampaignerImageById
Enclosing Method: testgetCampaignerImageById()
File: src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java:333
Taint Flags:

```
330  * @throws Exception the exception
331  */
332  @Test
333  public void testgetCampaignerImageById() throws Exception {
334  mockMvc.perform(get("http://localhost:8007/api/v1/crowd/campaigns/campaignerImage/download/"+campaign.get_id())
335  .header("X_Auth_Token", user.getAuthToken())
336  .contentType(MediaType.APPLICATION_JSON)).
```

Poor Error Handling: Overly Broad Throws

Low

Package: com.cts.dcrowd.rest.app.repository

src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java, line 214 (Poor Error Handling: Overly Broad Throws)

Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testgetUserActivities
Enclosing Method: testgetUserActivities()
File: src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java:214
Taint Flags:

```
211  * @throws Exception the exception
212  */
213  @Test
214  public void testgetUserActivities() throws Exception {
215  mockMvc.perform(get("http://localhost:8007/api/v1/crowd/users/activity/"+user1.get_id())
216  .header("X_Auth_Token", user1.getAuthToken())
217  .contentType(MediaType.APPLICATION_JSON)).
```

src/test/java/com/cts/dcrowd/rest/app/repository/BadgeControllerTestNG.java, line 130 (Poor Error Handling: Overly Broad Throws)

Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testviewBadgeList
Enclosing Method: testviewBadgeList()
File: src/test/java/com/cts/dcrowd/rest/app/repository/BadgeControllerTestNG.java:130
Taint Flags:

```
127  * @throws Exception the exception
128  */
129  @Test
130  public void testviewBadgeList() throws Exception
131  {
132  mockMvc.perform(get("http://localhost:8007/api/v1/crowd/admin/badge/viewBadgeList/1/1")
133  .header("X_Auth_Token", user.getAuthToken())
```

src/test/java/com/cts/dcrowd/rest/app/repository/BadgeControllerTestNG.java, line 146 (Poor Error Handling: Overly Broad Throws)

Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Poor Error Handling: Overly Broad Throws — Low

Package: com.cts.dcrowd.rest.app.repository

src/test/java/com/cts/dcrowd/rest/app/repository/BadgeControllerTestNG.java, line 146 (Poor Error Handling: Overly Broad Throws) — Low

Sink Details

Sink: Function: testmoderatorViewBadgeList
Enclosing Method: testmoderatorViewBadgeList()
File: src/test/java/com/cts/dcrowd/rest/app/repository/BadgeControllerTestNG.java:146
Taint Flags:

```
143  * @throws Exception the exception
144  */
145  @Test
146  public void testmoderatorViewBadgeList() throws Exception
147  {
148  mockMvc.perform(get("http://localhost:8007/api/v1/crowd/moderator/badge/viewBadgeList")
149  .header("X_Auth_Token", user.getAuthToken())
```

src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java, line 228 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testgetTopPerformers
Enclosing Method: testgetTopPerformers()
File: src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java:228
Taint Flags:

```
225  * @throws Exception the exception
226  */
227  @Test
228  public void testgetTopPerformers() throws Exception {
229  mockMvc.perform(get("http://localhost:8007/api/v1/crowd/users/topPerformers/1/2")
230  .header("X_Auth_Token", user1.getAuthToken())
231  .contentType(MediaType.APPLICATION_JSON)).
```

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 319 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testsendRating
Enclosing Method: testsendRating()

Poor Error Handling: Overly Broad Throws — Low

Package: com.cts.dcrowd.rest.app.repository

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 319 (Poor Error Handling: Overly Broad Throws) — Low

File: src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java:319
Taint Flags:

```
316  * @throws Exception the exception
317  */
318  @Test(dependsOnMethods = {"testsaveFeedback"})
319  public void testsendRating() throws Exception {
320  mockMvc.perform(post("http://localhost:8007/api/v1/crowd/campaigns/moderator/
sendRating/"+campaign.get_id())
321  .header("X_Auth_Token", user.getAuthToken())
322  .contentType(MediaType.APPLICATION_JSON)).
```

src/test/java/com/cts/dcrowd/rest/app/repository/BadgeControllerTestNG.java, line 93 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testaddBadge
Enclosing Method: testaddBadge()
File: src/test/java/com/cts/dcrowd/rest/app/repository/BadgeControllerTestNG.java:93
Taint Flags:

```
90  * @throws Exception the exception
91  */
92  @Test
93  public void testaddBadge() throws Exception
94  {
95  JSONObject jsonObj = TestUtil.readJsonObject(new File("src/test/resources/badgeJson.txt"));
96  String badgeJson = jsonObj.get("badge").toString();
```

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 96 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testgetProfileImageById
Enclosing Method: testgetProfileImageById()
File: src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java:96
Taint Flags:

Poor Error Handling: Overly Broad Throws — Low

Package: com.cts.dcrowd.rest.app.repository

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 96 (Poor Error Handling: Overly Broad Throws) — Low

```
93
94  }*/
95  @Test
96  public void testgetProfileImageById() throws Exception {
97
98  mockMvc.perform(get("http://localhost:8007/api/v1/crowd/campaigns/1/2")
99  .header("X_Auth_Token", user.getAuthToken())
```

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 261 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: teststartCampaign
Enclosing Method: teststartCampaign()
File: src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java:261
Taint Flags:

```
258  * @throws Exception the exception
259  */
260  @Test
261  public void teststartCampaign() throws Exception {
262  mockMvc.perform(post("http://localhost:8007/api/v1/crowd/campaigns/
startCampaign/"+campaign.get_id())
263  .header("X_Auth_Token", user.getAuthToken())
264  .contentType(MediaType.APPLICATION_JSON)).
```

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 361 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testmoderatorArchivedCampaign
Enclosing Method: testmoderatorArchivedCampaign()
File: src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java:361
Taint Flags:

```
358  * @throws Exception the exception
```

Poor Error Handling: Overly Broad Throws — Low

Package: com.cts.dcrowd.rest.app.repository

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 361 (Poor Error Handling: Overly Broad Throws) — Low

```
359  */
360  @Test
361  public void testmoderatorArchivedCampaign() throws Exception {
362  mockMvc.perform(get("http://localhost:8007/api/v1/crowd/campaigns/moderator/
archivedCampaign/1/1")
363  .header("X_Auth_Token", user.getAuthToken())
364  .contentType(MediaType.APPLICATION_JSON)).
```

src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java, line 387 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testuserAddressById
Enclosing Method: testuserAddressById()
File: src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java:387
Taint Flags:

```
384  * @throws Exception the exception
385  */
386  @Test
387  public void testuserAddressById() throws Exception {
388
389  mockMvc.perform(get("http://localhost:8007/api/v1/crowd/user/
address/"+user1.getAddressId())
390  .header("X_Auth_Token", user1.getAuthToken())
```

src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java, line 315 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testgetRedemption
Enclosing Method: testgetRedemption()
File: src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java:315
Taint Flags:

```
312  * @throws Exception the exception
313  */
314  @Test
```

Poor Error Handling: Overly Broad Throws — Low

Package: com.cts.dcrowd.rest.app.repository

src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java, line 315 (Poor Error Handling: Overly Broad Throws) — Low

```
315  public void testgetRedemption() throws Exception {
316  mockMvc.perform(get("http://localhost:8007/api/v1/crowd/users/redeemStore")
317  .header("X_Auth_Token", user1.getAuthToken())
318  .contentType(MediaType.APPLICATION_JSON)).
```

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 71 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: setup
Enclosing Method: setup()
File: src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java:71
Taint Flags:

```
68  * @throws Exception the exception
69  */
70  @BeforeClass
71  public void setup() throws Exception {
72  this.mockMvc = MockMvcBuilders.webAppContextSetup(this.wac).build();
73  mockMvc.perform(post("http://localhost:8007/api/v1/crowd/
login").header("Authorization",TestUtil.convertBase64("602937", "Mar@2017"))
74  .contentType(MediaType.APPLICATION_JSON)).andExpect(status().isOk());
```

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 212 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testviewCampaign
Enclosing Method: testviewCampaign()
File: src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java:212
Taint Flags:

```
209  * @throws Exception the exception
210  */
211  @Test
212  public void testviewCampaign() throws Exception {
213
```

Poor Error Handling: Overly Broad Throws — Low

Package: com.cts.dcrowd.rest.app.repository

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 212 (Poor Error Handling: Overly Broad Throws) — Low

```
214  mockMvc.perform(get("http://localhost:8007/api/v1/crowd/campaigns/
viewCampaign/"+campaign.get_id())
215  .header("X_Auth_Token", user.getAuthToken())
```

src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java, line 372 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testgetProfileImageById
Enclosing Method: testgetProfileImageById()
File: src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java:372
Taint Flags:

```
369  * @throws Exception the exception
370  */
371  @Test
372  public void testgetProfileImageById() throws Exception {
373
374  mockMvc.perform(get("http://localhost:8007/api/v1/crowd/users/profileImage/
download/"+user1.get_id())
375  .header("X_Auth_Token", user1.getAuthToken())
```

src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java, line 301 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testpodiumData
Enclosing Method: testpodiumData()
File: src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java:301
Taint Flags:

```
298  * @throws Exception the exception
299  */
300  @Test
301  public void testpodiumData() throws Exception {
302  mockMvc.perform(get("http://localhost:8007/api/v1/crowd/users/podium")
303  .header("X_Auth_Token", user1.getAuthToken())
304  .contentType(MediaType.APPLICATION_JSON)).
```

Poor Error Handling: Overly Broad Throws

Low

Package: com.cts.dcrowd.rest.app.repository

src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java, line 301 (Poor Error Handling: Overly Broad Throws)

Low

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 142 (Poor Error Handling: Overly Broad Throws)

Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testgetCompletedCampaignOfUser
Enclosing Method: testgetCompletedCampaignOfUser()
File: src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java:142
Taint Flags:

```
139  * @throws Exception the exception
140  */
141  @Test
142  public void testgetCompletedCampaignOfUser() throws Exception {
143
144  mockMvc.perform(get("http://localhost:8007/api/v1/crowd/campaigns/completedCampaign")
145  .header("X_Auth_Token", user.getAuthToken())
```

src/test/java/com/cts/dcrowd/rest/app/repository/BadgeControllerTestNG.java, line 161 (Poor Error Handling: Overly Broad Throws)

Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testgetUserBadge
Enclosing Method: testgetUserBadge()
File: src/test/java/com/cts/dcrowd/rest/app/repository/BadgeControllerTestNG.java:161
Taint Flags:

```
158  * @throws Exception the exception
159  */
160  @Test
161  public void testgetUserBadge() throws Exception
162  {
163  mockMvc.perform(get("http://localhost:8007/api/v1/crowd/badge/view/"+badge.getBadgeId())
164  .header("X_Auth_Token", user.getAuthToken())
```

Poor Error Handling: Overly Broad Throws — Low

Package: com.cts.dcrowd.rest.app.repository

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 243 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testconfirmTeam
Enclosing Method: testconfirmTeam()
File: src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java:243
Taint Flags:

```
240  * @throws Exception the exception
241  */
242  @Test
243  public void testconfirmTeam() throws Exception {
244
245  JSONObject jsonObj = TestUtil.readJsonObject(new File("src/test/resources/CampaignJson.txt"));
246  String users = jsonObj.get("users").toString();
```

src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java, line 357 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testgetProfileImage
Enclosing Method: testgetProfileImage()
File: src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java:357
Taint Flags:

```
354  * @throws Exception the exception
355  */
356  @Test
357  public void testgetProfileImage() throws Exception {
358
359  mockMvc.perform(get("http://localhost:8007/api/v1/crowd/users/profileImage/download")
360  .header("X_Auth_Token", user1.getAuthToken())
```

src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java, line 156 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Poor Error Handling: Overly Broad Throws — Low

Package: com.cts.dcrowd.rest.app.repository

src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java, line 156 (Poor Error Handling: Overly Broad Throws) — Low

Sink Details

Sink: Function: testgetUserDetailsByName
Enclosing Method: testgetUserDetailsByName()
File: src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java:156
Taint Flags:

```
153  * @throws Exception the exception
154  */
155  @Test
156  public void testgetUserDetailsByName() throws Exception {
157  //String id = user1.get_id();
158  mockMvc.perform(get("http://localhost:8007/api/v1/crowd/users/
userDetailbyName/"+user1.getUsername())
159  .header("X_Auth_Token", user1.getAuthToken())
```

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 347 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testmoderatorCurrentCampaign
Enclosing Method: testmoderatorCurrentCampaign()
File: src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java:347
Taint Flags:

```
344  * @throws Exception the exception
345  */
346  @Test
347  public void testmoderatorCurrentCampaign() throws Exception {
348  mockMvc.perform(get("http://localhost:8007/api/v1/crowd/campaigns/moderator/
currentCampaign/1/1")
349  .header("X_Auth_Token", user.getAuthToken())
350  .contentType(MediaType.APPLICATION_JSON)).
```

src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java, line 286 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Poor Error Handling: Overly Broad Throws | Low

Package: com.cts.dcrowd.rest.app.repository

src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java, line 286 (Poor Error Handling: Overly Broad Throws) | Low

Sink: Function: testremoveUserBadge
Enclosing Method: testremoveUserBadge()
File: src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java:286
Taint Flags:

```
283  * @throws Exception the exception
284  */
285  @Test
286  public void testremoveUserBadge() throws Exception {
287  mockMvc.perform(post("http://localhost:8007/api/v1/crowd/users/moderator/
removeBadge/"+user1.get_id()+"/1")
288  .header("X_Auth_Token", user1.getAuthToken())
289  .contentType(MediaType.APPLICATION_JSON)).
```

src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java, line 186 (Poor Error Handling: Overly Broad Throws) | Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testgetMyDetails
Enclosing Method: testgetMyDetails()
File: src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java:186
Taint Flags:

```
183  * @throws Exception the exception
184  */
185  @Test
186  public void testgetMyDetails() throws Exception {
187  mockMvc.perform(get("http://localhost:8007/api/v1/crowd/user/myDetails")
188  .header("X_Auth_Token", user1.getAuthToken())
189  .contentType(MediaType.APPLICATION_JSON)).
```

src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java, line 102 (Poor Error Handling: Overly Broad Throws) | Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: setup
Enclosing Method: setup()
File: src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java:102
Taint Flags:

Poor Error Handling: Overly Broad Throws — Low

Package: com.cts.dcrowd.rest.app.repository

src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java, line 102 (Poor Error Handling: Overly Broad Throws) — Low

```
99   * @throws Exception the exception
100   */
101   @BeforeClass
102   public void setup() throws Exception {
103   this.mockMvc = MockMvcBuilders.webAppContextSetup(this.wac).build();
104
105   /*int i;
```

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 111 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: testgetCurrentCampaignOfUser
Enclosing Method: testgetCurrentCampaignOfUser()
File: src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java:111
Taint Flags:

```
108   * @throws Exception the exception
109   */
110   @Test
111   public void testgetCurrentCampaignOfUser() throws Exception {
112
113   mockMvc.perform(get("http://localhost:8007/api/v1/crowd/campaigns/currentCampaign")
114   .header("X_Auth_Token", user.getAuthToken())
```

Package: com.cts.dcrowd.rest.app.security.configuration

src/main/java/com/cts/dcrowd/rest/app/security/configuration/WebSecurityConfiguration.java, line 172 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: authenticationManagerBean
Enclosing Method: authenticationManagerBean()
File: src/main/java/com/cts/dcrowd/rest/app/security/configuration/WebSecurityConfiguration.java:172
Taint Flags:

```
169   */
```

Poor Error Handling: Overly Broad Throws — Low

Package: com.cts.dcrowd.rest.app.security.configuration

src/main/java/com/cts/dcrowd/rest/app/security/configuration/WebSecurityConfiguration.java, line 172 (Poor Error Handling: Overly Broad Throws) — Low

```
170  @Bean
171  @Override
172  public AuthenticationManager authenticationManagerBean() throws Exception {
173  return super.authenticationManagerBean();
174  }
175
```

src/main/java/com/cts/dcrowd/rest/app/security/configuration/WebSecurityConfiguration.java, line 97 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: configure
Enclosing Method: configure()
File: src/main/java/com/cts/dcrowd/rest/app/security/configuration/WebSecurityConfiguration.java:97
Taint Flags:

```
94  * @see
org.springframework.security.config.annotation.web.configuration.WebSecurityConfigurerAdapter#cc
95  */
96  @Override
97  protected void configure(HttpSecurity httpSecurity) throws Exception {
98
99  httpSecurity
100  .sessionManagement()
```

src/main/java/com/cts/dcrowd/rest/app/security/configuration/WebSecurityConfiguration.java, line 182 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: configure
Enclosing Method: configure()
File: src/main/java/com/cts/dcrowd/rest/app/security/configuration/WebSecurityConfiguration.java:182
Taint Flags:

```
179  * @see
org.springframework.security.config.annotation.web.configuration.WebSecurityConfigurerAdapter#cc
```

Poor Error Handling: Overly Broad Throws — Low

Package: com.cts.dcrowd.rest.app.security.configuration

src/main/java/com/cts/dcrowd/rest/app/security/configuration/WebSecurityConfiguration.java, line 182 (Poor Error Handling: Overly Broad Throws) — Low

```
180  */
181  @Override
182  public void configure(AuthenticationManagerBuilder auth) throws Exception {
183  if("db".equals(this.authType)) {
184  auth.userDetailsService(userDetailsServiceBean());
185  }
```

src/main/java/com/cts/dcrowd/rest/app/security/configuration/WebSecurityConfiguration.java, line 150 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: authenticationTokenFilterBean
Enclosing Method: authenticationTokenFilterBean()
File: src/main/java/com/cts/dcrowd/rest/app/security/configuration/WebSecurityConfiguration.java:150
Taint Flags:

```
147  * @throws Exception the exception
148  */
149  @Bean
150  public AuthenticationTokenFilter authenticationTokenFilterBean() throws Exception {
151  AuthenticationTokenFilter authenticationTokenFilter = new AuthenticationTokenFilter();
152  authenticationTokenFilter.setAuthenticationManager(authenticationManagerBean());
153  LOGGER.debug("AuthenticationTokenFilter :: Bean Intialized");
```

src/main/java/com/cts/dcrowd/rest/app/security/configuration/WebSecurityConfiguration.java, line 244 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: ldapAuthenticator
Enclosing Method: ldapAuthenticator()
File: src/main/java/com/cts/dcrowd/rest/app/security/configuration/WebSecurityConfiguration.java:244
Taint Flags:

```
241  * @throws Exception the exception
242  */
243  @Bean
```

Poor Error Handling: Overly Broad Throws — Low

Package: com.cts.dcrowd.rest.app.security.configuration

src/main/java/com/cts/dcrowd/rest/app/security/configuration/WebSecurityConfiguration.java, line 244 (Poor Error Handling: Overly Broad Throws) — Low

```
244  public LdapAuthenticator ldapAuthenticator() throws Exception {
245  BaseLdapPathContextSource contextSource = ldapContextSource();
246  BindAuthenticator authenticator = new BindAuthenticator(contextSource);
247  authenticator.setUserSearch(new FilterBasedLdapUserSearch("", searchFilter,
contextSource));
```

src/main/java/com/cts/dcrowd/rest/app/security/configuration/MongoSecurityConfiguration.java, line 74 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: mongo
Enclosing Method: mongo()
File: src/main/java/com/cts/dcrowd/rest/app/security/configuration/MongoSecurityConfiguration.java:74
Taint Flags:

```
71  * @see org.springframework.data.mongodb.config.AbstractMongoConfiguration#mongo()
72  */
73  @Override
74  public Mongo mongo() throws Exception {
75  return new MongoClient(Collections.singletonList(new ServerAddress(this.host,
Integer.valueOf(this.port)))
76  , Collections.singletonList(MongoCredential.createScramSha1Credential(this.username,
this.database, this.password.toCharArray())));
77  }
```

src/main/java/com/cts/dcrowd/rest/app/security/configuration/MongoSecurityConfiguration.java, line 105 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: mongoTemplate
Enclosing Method: mongoTemplate()
File: src/main/java/com/cts/dcrowd/rest/app/security/configuration/MongoSecurityConfiguration.java:105
Taint Flags:

```
102  */
103  @Override
```

Poor Error Handling: Overly Broad Throws — Low

Package: com.cts.dcrowd.rest.app.security.configuration

src/main/java/com/cts/dcrowd/rest/app/security/configuration/MongoSecurityConfiguration.java, line 105 (Poor Error Handling: Overly Broad Throws) — Low

```
104  @Bean
105  public MongoTemplate mongoTemplate() throws Exception{
106  return new MongoTemplate(mongo(), getDatabaseName());
107  }
108 }
```

src/main/java/com/cts/dcrowd/rest/app/security/configuration/WebSecurityConfiguration.java, line 259 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: ldapAuthoritiesPopulator
Enclosing Method: ldapAuthoritiesPopulator()
File: src/main/java/com/cts/dcrowd/rest/app/security/configuration/WebSecurityConfiguration.java:259
Taint Flags:

```
256  * @throws Exception the exception
257  */
258  @Bean
259  public LdapAuthoritiesPopulator ldapAuthoritiesPopulator() throws Exception {
260  DefaultLdapAuthoritiesPopulator defaultAuthPopulator = new
DefaultLdapAuthoritiesPopulator(ldapContextSource(), "");
261  defaultAuthPopulator.setGroupRoleAttribute(groupRoleAttribute);
262  LOGGER.debug("LdapAuthoritiesPopulator :: Bean Intialized");
```

src/main/java/com/cts/dcrowd/rest/app/security/configuration/WebSecurityConfiguration.java, line 215 (Poor Error Handling: Overly Broad Throws) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: ldapAuthenticationProvider
Enclosing Method: ldapAuthenticationProvider()
File: src/main/java/com/cts/dcrowd/rest/app/security/configuration/WebSecurityConfiguration.java:215
Taint Flags:

```
212  * @throws Exception the exception
213  */
214  @Bean
```

Poor Error Handling: Overly Broad Throws | Low

Package: com.cts.dcrowd.rest.app.security.configuration

src/main/java/com/cts/dcrowd/rest/app/security/configuration/WebSecurityConfiguration.java, line 215 (Poor Error Handling: Overly Broad Throws) | Low

```
215  public LdapAuthenticationProvider ldapAuthenticationProvider() throws Exception {
216  LdapAuthenticationProvider authProvider = new
LdapAuthenticationProvider(ldapAuthenticator(), ldapAuthoritiesPopulator());
217  authProvider.setUserDetailsContextMapper(userDetailsContextMapper());
218  LOGGER.debug("LdapAuthenticationProvider :: Bean Intialized");
```

src/main/java/com/cts/dcrowd/rest/app/security/configuration/WebSecurityConfiguration.java, line 229 (Poor Error Handling: Overly Broad Throws) | Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: ldapContextSource
Enclosing Method: ldapContextSource()
File: src/main/java/com/cts/dcrowd/rest/app/security/configuration/WebSecurityConfiguration.java:229
Taint Flags:

```
226  * @throws Exception the exception
227  */
228  @Bean
229  public BaseLdapPathContextSource ldapContextSource() throws Exception {
230  DefaultSpringSecurityContextSource contextSource = new
DefaultSpringSecurityContextSource(ldapFullUrl);
231  contextSource.setUserDn(userDn);
232  contextSource.setPassword(password);
```

src/main/java/com/cts/dcrowd/rest/app/security/configuration/WebSecurityConfiguration.java, line 202 (Poor Error Handling: Overly Broad Throws) | Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: userDetailsServiceBean
Enclosing Method: userDetailsServiceBean()
File: src/main/java/com/cts/dcrowd/rest/app/security/configuration/WebSecurityConfiguration.java:202
Taint Flags:

```
199  * @see
org.springframework.security.config.annotation.web.configuration.WebSecurityConfigurerAdapter#us
200  */
```

Poor Error Handling: Overly Broad Throws

Low

Package: com.cts.dcrowd.rest.app.security.configuration

src/main/java/com/cts/dcrowd/rest/app/security/configuration/WebSecurityConfiguration.java, line 202 (Poor Error Handling: Overly Broad Throws)

Low

```
201  @Override
202  public UserDetailsService userDetailsServiceBean() throws Exception {
203  return dCrowdUserDetailsService();
204  }
205
```

Poor Error Handling: Program Catches NullPointerException (1 issue)

Abstract

It is generally a bad practice to catch `NullPointerException`.

Explanation

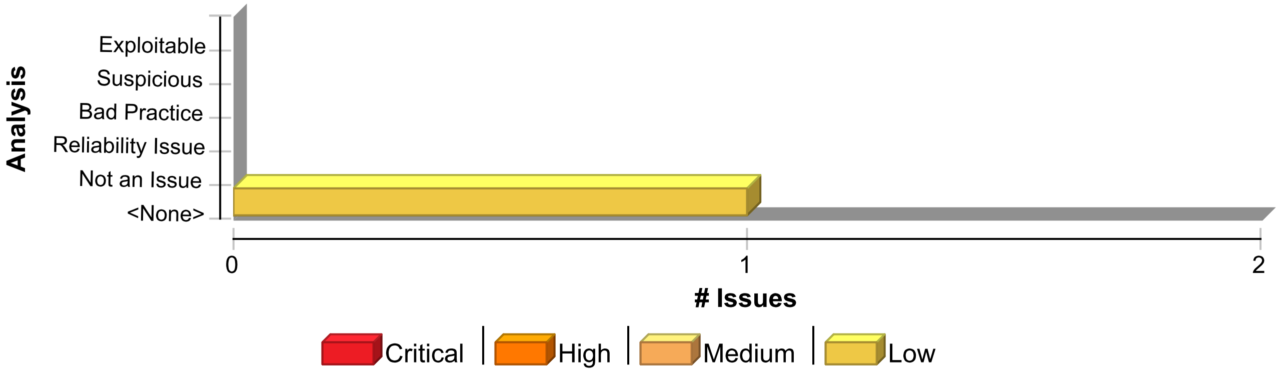
Programmers typically catch `NullPointerException` under three circumstances: 1. The program contains a null pointer dereference. Catching the resulting exception was easier than fixing the underlying problem. 2. The program explicitly throws a `NullPointerException` to signal an error condition. 3. The code is part of a test harness that supplies unexpected input to the classes under test. Of these three circumstances, only the last is acceptable. **Example:** The following code mistakenly catches a `NullPointerException`.

```
  try {
    mysteryMethod();
  }
  catch (NullPointerException npe) {
  }
```

Recommendation

The program should not dereference null pointers. If you cannot eliminate the cause of the null pointer dereference, you must carefully review the code to make sure that the exception is handled in such a way that the program does not enter into an unexpected or illegal state. If the `NullPointerException` is being thrown explicitly, change the program to throw an exception derived from `RuntimeException` or `Error` instead.

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Poor Error Handling: Program Catches NullPointerException	1	0	0	1
Total	**1**	**0**	**0**	**1**

Poor Error Handling: Program Catches NullPointerException — Low

Package: com.cts.dcrowd.rest.defect.app.services.impl

src/main/java/com/cts/dcrowd/rest/defect/app/services/impl/DefaultDefectServiceImpl.java, line 504 (Poor Error Handling: Program Catches NullPointerException) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: defectNameBuilder()
File: src/main/java/com/cts/dcrowd/rest/defect/app/services/impl/DefaultDefectServiceImpl.java:504
Taint Flags:

```
501  String defPrefix;
502  try {
503  campaignsCount = defectEntryRepository.countByDeCampaignId(defectEntry.getDeCampaignId());
504  } catch(NullPointerException e) {
505  campaignsCount = 0;
506  }
507  if(defectEntry.getDeCampaignName().length()>=3){
```

Poor Error Handling: Throw Inside Finally (3 issues)

Abstract

Using a `throw` statement inside a `finally` block breaks the logical progression through the `try-catch-finally`.

Explanation

In Java, `finally` blocks are always executed after their corresponding `try-catch` blocks and are often used to free allocated resources, such as file handles or database cursors. Throwing an exception in a `finally` block can bypass critical cleanup code since normal program execution will be disrupted.
Example 1: In the following code, the call to `stmt.close()` is bypassed when the `FileNotFoundException` is thrown.

```
public void processTransaction(Connection conn) throws FileNotFoundException
{
    FileInputStream fis = null;
    Statement stmt = null;
    try
    {
        stmt = conn.createStatement();
        fis = new FileInputStream("badFile.txt");
        ...
    }
    catch (FileNotFoundException fe)
    {
        log("File not found.");
    }
    catch (SQLException se)
    {
        //handle error
    }
    finally
    {
        if (fis == null)
        {
            throw new FileNotFoundException();
        }

        if (stmt != null)
        {
            try
            {
                stmt.close();
            }
            catch (SQLException e)
            {
                log(e);
            }
        }
    }
}
```

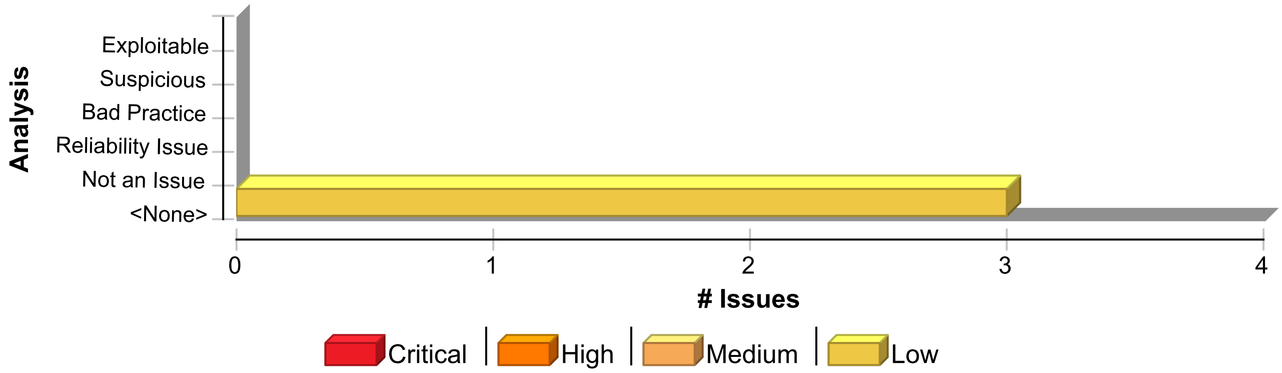
This category is from the Cigital Java Rulepack. http://www.cigital.com/

Recommendation

Never throw exceptions from within `finally` blocks. If you must re-throw an exception, do it inside a `catch` block so as not to interrupt the normal execution of the `finally` block. **Example 2:** The following code re-throws the `FileNotFoundException` in the `catch` block.

```
public void processTransaction(Connection conn) throws FileNotFoundException
{
    FileInputStream fis = null;
    Statement stmt = null;
    try
    {
        stmt = conn.createStatement();
        fis = new FileInputStream("badFile.txt");
        ...
    }
    catch (FileNotFoundException fe)
    {
        log("File not found.");
        throw fe;
    }
    catch (SQLException se)
    {
        //handle error
    }
    finally
    {
        if (fis != null)
        {
            try
            {
                fis.close();
            }
            catch (IOException ie)
            {
                log(ie);
            }
        }

        if (stmt != null)
        {
            try
            {
                stmt.close();
            }
            catch (SQLException e)
            {
                log(e);
            }
        }
    }
}
```

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Poor Error Handling: Throw Inside Finally	3	0	0	3
Total	**3**	**0**	**0**	**3**

Poor Error Handling: Throw Inside Finally

Low

Package: com.cts.dcrowd.rest.app.utils

src/main/java/com/cts/dcrowd/rest/app/utils/Util.java, line 679 (Poor Error Handling: Throw Inside Finally)

Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: FinallyBlock
Enclosing Method: writeFileToFileServer()
File: src/main/java/com/cts/dcrowd/rest/app/utils/Util.java:679
Taint Flags:

```
676  new File(fileDIR).mkdirs();
677  }
678  file = new File(finalFilePath);
679  try (ByteArrayInputStream inputStream = new ByteArrayInputStream(imageData)) {
680
681  if (!file.exists()) {
682  file.createNewFile();
```

src/main/java/com/cts/dcrowd/rest/app/utils/StopwordUtility.java, line 58 (Poor Error Handling: Throw Inside Finally)

Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: FinallyBlock
Enclosing Method: loadStopList()

Poor Error Handling: Throw Inside Finally — Low

Package: com.cts.dcrowd.rest.app.utils

src/main/java/com/cts/dcrowd/rest/app/utils/StopwordUtility.java, line 58 (Poor Error Handling: Throw Inside Finally) — Low

File: src/main/java/com/cts/dcrowd/rest/app/utils/StopwordUtility.java:58
Taint Flags:

```
55  private synchronized void loadStopList() throws IOException {
56  stoplist = new LinkedHashSet<>(256);
57  try (BufferedReader br = new BufferedReader(
58  new InputStreamReader(StopwordUtility.class.getResourceAsStream("/stoplist.txt")))) {
59  String line;
60  while ((line = br.readLine()) != null) {
61  String stopword = line.trim();
```

src/main/java/com/cts/dcrowd/rest/app/utils/EmailDomainFilterUtility.java, line 56 (Poor Error Handling: Throw Inside Finally) — Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: FinallyBlock
Enclosing Method: loadDomainList()
File: src/main/java/com/cts/dcrowd/rest/app/utils/EmailDomainFilterUtility.java:56
Taint Flags:

```
53  private synchronized void loadDomainList() throws IOException {
54  emailDomainList = new LinkedHashSet<>(256);
55  try (BufferedReader br = new BufferedReader(
56  new InputStreamReader(EmailDomainFilterUtility.class.getResourceAsStream("/
domain_filter.txt")))) {
57  String line;
58  while ((line = br.readLine()) != null) {
59  String stopword = line.trim();
```

Poor Logging Practice: Use of a System Output Stream (3 issues)

Abstract

Using `System.out` or `System.err` rather than a dedicated logging facility makes it difficult to monitor the behavior of the program.

Explanation

Example 1: The first Java program that a developer learns to write often looks like this:

```
public class MyClass
  public static void main(String[] args) {
    System.out.println("hello world");
  }
}
```

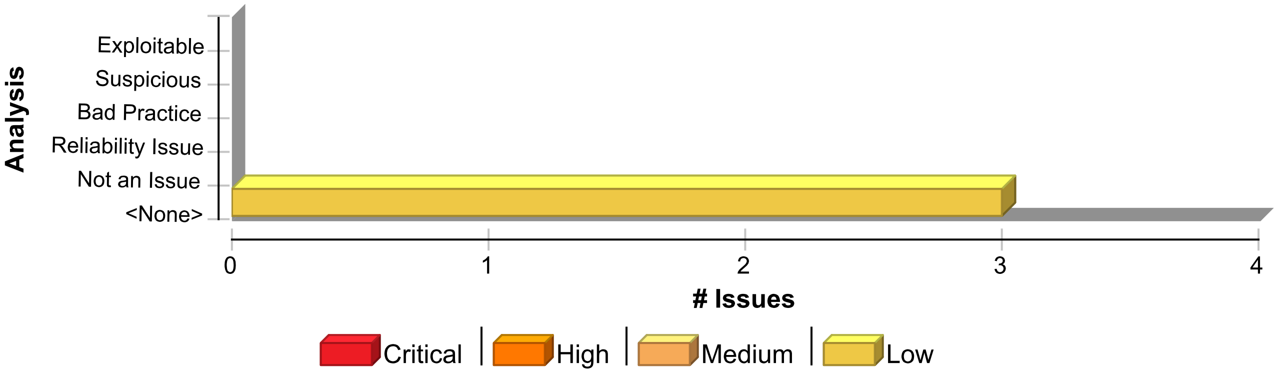
While most programmers go on to learn many nuances and subtleties about Java, a surprising number hang on to this first lesson and never give up on writing messages to standard output using `System.out.println()`. The problem is that writing directly to standard output or standard error is often used as an unstructured form of logging. Structured logging facilities provide features like logging levels, uniform formatting, a logger identifier, timestamps, and, perhaps most critically, the ability to direct the log messages to the right place. When the use of system output streams is jumbled together with the code that uses loggers properly, the result is often a well-kept log that is missing critical information. Developers widely accept the need for structured logging, but many continue to use system output streams in their "pre-production" development. If the code you are reviewing is past the initial phases of development, use of `System.out` or `System.err` may indicate an oversight in the move to a structured logging system.

Recommendation

Use a Java logging facility rather than `System.out` or `System.err`. **Example 2:** For example, the "hello world" program above can be re-written using log4j like this:

```
import org.apache.log4j.Logger;
import org.apache.log4j.BasicConfigurator;

public class MyClass {
  private final static Logger logger =
            Logger.getLogger(MyClass.class);

  public static void main(String[] args) {
    BasicConfigurator.configure();
    logger.info("hello world");
  }
}
```

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Poor Logging Practice: Use of a System Output Stream	3	0	0	3
Total	**3**	**0**	**0**	**3**

Poor Logging Practice: Use of a System Output Stream — Low

Package: .com.cts.dcrowd.rest.defect.app.jira.services

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 242 (Poor Logging Practice: Use of a System Output Stream) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionCall: println
Enclosing Method: accept()
File: src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java:242
Taint Flags:

```
239  String serverUrl = issueURL + "/attachments";
240  RestTemplate restTemplate = new RestTemplate();
241  ResponseEntity<String> response1 = restTemplate.postForEntity(serverUrl, requestEntity,
String.class);
242  System.out.println(response1);
243
244  } catch (Exception e) {
245  // TODO Auto-generated catch block
```

Package: com.cts.dcrowd.rest.app.repository

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 181 (Poor Logging Practice: Use of a System Output Stream) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Structural)

Poor Logging Practice: Use of a System Output Stream — Low

Package: com.cts.dcrowd.rest.app.repository

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 181 (Poor Logging Practice: Use of a System Output Stream) — Low

Sink Details

Sink: FunctionCall: println
Enclosing Method: testopenForNomination()
File: src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java:181
Taint Flags:

```
178  ObjectMapper mapper = new ObjectMapper();
179  String createdCampaign = mapper.writeValueAsString(campaign);
180  String openForNomination = "{ \"param\": " +createdCampaign+"}";
181  System.out.println("openForNomination:"+openForNomination);
182  mockMvc.perform(post("http://localhost:8007/api/v1/crowd/campaigns/openForNomination")
183  .header("X_Auth_Token", user.getAuthToken())
184  .contentType(MediaType.APPLICATION_JSON)
```

Package: com.cts.dcrowd.rest.app.util

src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java, line 102 (Poor Logging Practice: Use of a System Output Stream) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionCall: println
Enclosing Method: generateBlowFishDecryptValue()
File: src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java:102
Taint Flags:

```
99  cipher.init(Cipher.DECRYPT_MODE, skeyspec);
100  byte[] decrypted = cipher.doFinal(Base64.getDecoder().decode(encryptedValue));
101  strData = new String(decrypted);
102  System.out.println(strData);
103  } catch (Exception e) {
104  e.printStackTrace();
105  }
```

Poor Style: Confusing Naming (1 issue)

Abstract

The class contains a field and a method with the same name.

Explanation

It is confusing to have a member field and a method with the same name. It makes it easy for a programmer to accidentally call the method when attempting to access the field or vice versa. **Example 1:**

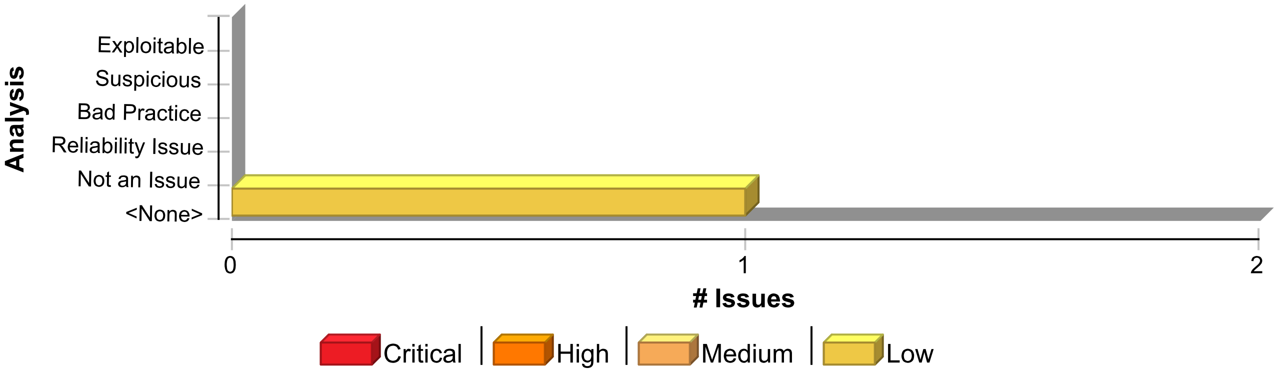
```
public class Totaller {
  private int total;
  public int total() {
    ...
  }
}
```

Recommendation

Rename either the method or the field. If the method returns the field, consider following the standard getter/setter naming convention. **Example 2:** The code in Example 1 could be rewritten in the following way:

```
public class Totaller {
  private int total;
  public int getTotal() {
    ...
  }
}
```

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Poor Style: Confusing Naming	1	0	0	1
Total	**1**	**0**	**0**	**1**

Poor Style: Confusing Naming — Low

Package: com.cts.dcrowd.rest.app.security.configuration

src/main/java/com/cts/dcrowd/rest/app/security/configuration/WebSecurityConfiguration.java, line 91 (Poor Style: Confusing Naming) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: securityService
File: src/main/java/com/cts/dcrowd/rest/app/security/configuration/WebSecurityConfiguration.java:91
Taint Flags:

```
88
89  /** The security service. */
90  @Autowired
91  private SecurityService securityService;
92
93  /* (non-Javadoc)
94   * @see
org.springframework.security.config.annotation.web.configuration.WebSecurityConfigurerAdapter#cc
```

Poor Style: Redundant Initialization (1 issue)

Abstract

The variable's value is assigned but never used, making it a dead store.

Explanation

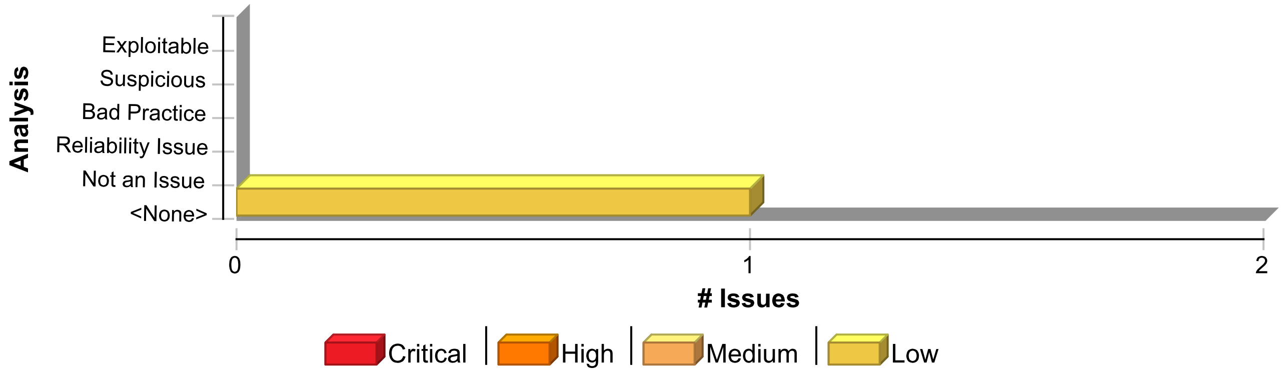
This variable's initial value is not used. After initialization, the variable is either assigned another value or goes out of scope. **Example:** The following code excerpt assigns to the variable `r` and then overwrites the value without using it.

```
  int r = getNum();
  r = getNewNum(buf);
```

Recommendation

Remove unnecessary assignments in order to make the code easier to understand and maintain.

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Poor Style: Redundant Initialization	1	0	0	1
Total	**1**	**0**	**0**	**1**

Poor Style: Redundant Initialization **Low**

Package: com.cts.dcrowd.rest.app.utils

src/main/java/com/cts/dcrowd/rest/app/utils/Util.java, line 444 (Poor Style: Redundant Initialization) **Low**

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: score
Enclosing Method: setFuzzyScore()
File: src/main/java/com/cts/dcrowd/rest/app/utils/Util.java:444

Poor Style: Redundant Initialization | Low

Package: com.cts.dcrowd.rest.app.utils

src/main/java/com/cts/dcrowd/rest/app/utils/Util.java, line 444 (Poor Style: Redundant Initialization) | Low

Taint Flags:

```
441   * @return the long
442   */
443   public static double setFuzzyScore(String defectTerm, String defectQuery) {
444   double score = 0.0;
445   try {
446   FuzzyScore fScore = new FuzzyScore(Locale.getDefault());
447   score = fScore.fuzzyScore(defectTerm, defectQuery);
```

Poor Style: Value Never Read (78 issues)

Abstract

The variable's value is assigned but never used, making it a dead store.

Explanation

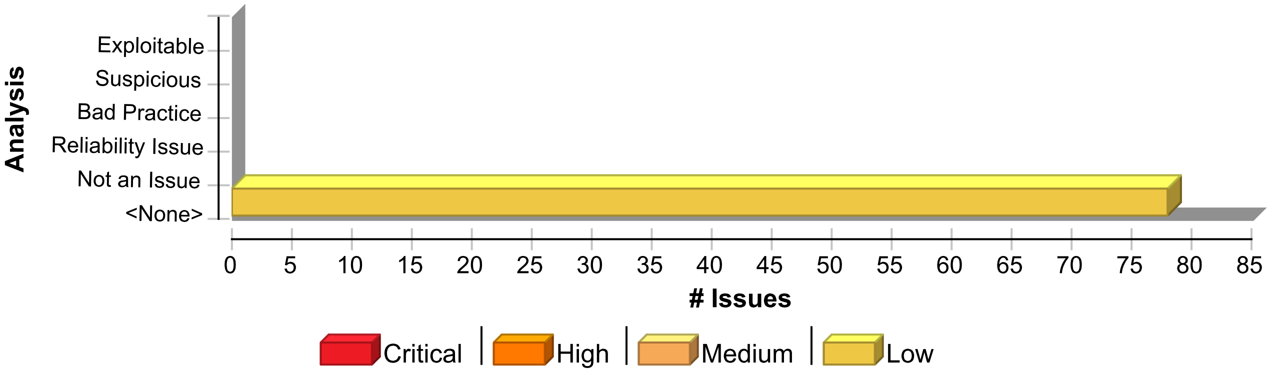
This variable's value is not used. After the assignment, the variable is either assigned another value or goes out of scope. **Example:** The following code excerpt assigns to the variable `r` and then overwrites the value without using it.

```
  r = getName();
  r = getNewBuffer(buf);
```

Recommendation

Remove unnecessary assignments in order to make the code easier to understand and maintain.

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Poor Style: Value Never Read	78	0	0	78
Total	**78**	**0**	**0**	**78**

Poor Style: Value Never Read **Low**

Package: .com.cts.dcrowd.rest.app.service.helper

src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java, line 455 (Poor Style: Value Never Read) **Low**

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: valueStringObj
Enclosing Method: apply()
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java:455

Poor Style: Value Never Read — Low

Package: .com.cts.dcrowd.rest.app.service.helper

src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java, line 455 (Poor Style: Value Never Read) — Low

Taint Flags:

```
452  if(!category.equalsIgnoreCase("department")){
453  List<?> valueObjList = (List<?>) value;
454  for(Object valueObj : valueObjList) {
455  String valueStringObj = String.valueOf(valueObj);
456  if(category.equalsIgnoreCase("devices")){
457  type = "deviceType";
458  }
```

src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java, line 370 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: filterObject
Enclosing Method: accept()
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java:370
Taint Flags:

```
367  if(!Util.isEmpty(filterByState)) {
368  for(Object filterTagValue : (List<?>) filterTags.getValue()) {
369  try {
370  JSONObject filterObject = convertObjectToJson(filterTagValue);
371  if(filterObject.getString("state_code").equals(filterByState.getAddress().getState())){
372
if(filterObject.getString("country_code").equals(filterByState.getAddress().getCountry())) {
373  userListAfterStateFilter.add(filterByState);
```

Package: .com.cts.dcrowd.rest.defect.app.jira.services

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 240 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: restTemplate
Enclosing Method: accept()
File: src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java:240
Taint Flags:

Poor Style: Value Never Read — Low

Package: .com.cts.dcrowd.rest.defect.app.jira.services

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 240 (Poor Style: Value Never Read) — Low

```
237  org.springframework.http.HttpEntity<LinkedMultiValueMap<String, Object>> requestEntity =
new org.springframework.http.HttpEntity<>(
238  map, headers);
239  String serverUrl = issueURL + "/attachments";
240  RestTemplate restTemplate = new RestTemplate();
241  ResponseEntity<String> response1 = restTemplate.postForEntity(serverUrl, requestEntity,
String.class);
242  System.out.println(response1);
243
```

Package: com.cts.dcrowd.rest.app.controller

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 784 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: status
Enclosing Method: removeCampaign()
File: src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java:784
Taint Flags:

```
781  {
782  int status = 200;
783  try {
784  status = campaignService.removeCampaign(campaignId);
785  return buildSuccessResponse("Campaign removed from the system successfully");
786  } catch (DcrowdException de){
787  de.printStackTrace();
```

Package: com.cts.dcrowd.rest.app.repository

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 291 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: feedback
Enclosing Method: testsaveFeedback()
File: src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java:291
Taint Flags:

Poor Style: Value Never Read | Low

Package: com.cts.dcrowd.rest.app.repository

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 291 (Poor Style: Value Never Read) | Low

```
288  @Test
289  public void testsaveFeedback() throws Exception {
290  JSONObject jsonObj = TestUtil.readJsonObject(new File("src/test/resources/
CampaignJson.txt"));
291  String feedback = jsonObj.get("feedback").toString();
292  mockMvc.perform(post("http://localhost:8007/api/v1/crowd/campaigns/moderator/
saveFeedback/"+campaign.get_id())
293  .header("X_Auth_Token", user.getAuthToken())
294  .contentType(MediaType.APPLICATION_JSON).content(feedback)).
```

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 160 (Poor Style: Value Never Read) | Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: campaignJson
Enclosing Method: testsaveCampaign()
File: src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java:160
Taint Flags:

```
157  public void testsaveCampaign() throws Exception {
158
159  JSONObject jsonObj = TestUtil.readJsonObject(new File("src/test/resources/
CampaignJson.txt"));
160  String campaignJson = jsonObj.get("campaign").toString();
161  mockMvc.perform(post("http://localhost:8007/api/v1/crowd/campaigns/saveCampaign")
162  .header("X_Auth_Token", user.getAuthToken())
163  .contentType(MediaType.APPLICATION_JSON)
```

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 178 (Poor Style: Value Never Read) | Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: mapper
Enclosing Method: testopenForNomination()
File: src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java:178
Taint Flags:

```
175  */
176  @Test
```

Poor Style: Value Never Read — Low

Package: com.cts.dcrowd.rest.app.repository

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 178 (Poor Style: Value Never Read) — Low

```
177  public void testopenForNomination() throws Exception {
178  ObjectMapper mapper = new ObjectMapper();
179  String createdCampaign = mapper.writeValueAsString(campaign);
180  String openForNomination = "{ \"param\": " +createdCampaign+"}";
181  System.out.println("openForNomination:"+openForNomination);
```

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 246 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: users
Enclosing Method: testconfirmTeam()
File: src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java:246
Taint Flags:

```
243  public void testconfirmTeam() throws Exception {
244
245  JSONObject jsonObj = TestUtil.readJsonObject(new File("src/test/resources/
CampaignJson.txt"));
246  String users = jsonObj.get("users").toString();
247  mockMvc.perform(post("http://localhost:8007/api/v1/crowd/campaigns/
confirmTeam/"+campaign.get_id())
248  .header("X_Auth_Token", user.getAuthToken())
249  .contentType(MediaType.APPLICATION_JSON).content(users)).
```

src/test/java/com/cts/dcrowd/rest/app/repository/BadgeControllerTestNG.java, line 95 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: jsonObj
Enclosing Method: testaddBadge()
File: src/test/java/com/cts/dcrowd/rest/app/repository/BadgeControllerTestNG.java:95
Taint Flags:

```
92  @Test
93  public void testaddBadge() throws Exception
94  {
95  JSONObject jsonObj = TestUtil.readJsonObject(new File("src/test/resources/badgeJson.txt"));
96  String badgeJson = jsonObj.get("badge").toString();
```

Poor Style: Value Never Read — Low

Package: com.cts.dcrowd.rest.app.repository

src/test/java/com/cts/dcrowd/rest/app/repository/BadgeControllerTestNG.java, line 95 (Poor Style: Value Never Read) — Low

```
97
98  mockMvc.perform(post("http://localhost:8007/api/v1/crowd/admin/badge/add")
```

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 159 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: jsonObj
Enclosing Method: testsaveCampaign()
File: src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java:159
Taint Flags:

```
156  */
157  public void testsaveCampaign() throws Exception {
158
159  JSONObject jsonObj = TestUtil.readJsonObject(new File("src/test/resources/CampaignJson.txt"));
160  String campaignJson = jsonObj.get("campaign").toString();
161  mockMvc.perform(post("http://localhost:8007/api/v1/crowd/campaigns/saveCampaign")
162  .header("X_Auth_Token", user.getAuthToken())
```

src/test/java/com/cts/dcrowd/rest/app/repository/BadgeControllerTestNG.java, line 96 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: badgeJson
Enclosing Method: testaddBadge()
File: src/test/java/com/cts/dcrowd/rest/app/repository/BadgeControllerTestNG.java:96
Taint Flags:

```
93  public void testaddBadge() throws Exception
94  {
95  JSONObject jsonObj = TestUtil.readJsonObject(new File("src/test/resources/badgeJson.txt"));
96  String badgeJson = jsonObj.get("badge").toString();
97
98  mockMvc.perform(post("http://localhost:8007/api/v1/crowd/admin/badge/add")
99  .header("X_Auth_Token", user.getAuthToken())
```

Poor Style: Value Never Read — Low

Package: com.cts.dcrowd.rest.app.repository

src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java, line 243 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: cityId
Enclosing Method: testLocationgetTopPerformers()
File: src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java:243
Taint Flags:

```
240  */
241  @Test
242  public void testLocationgetTopPerformers() throws Exception {
243  City cityId = masterService.getCityIdByName("coimbatore");
244  mockMvc.perform(get("http://localhost:8007/api/v1/crowd/users/
topLocationPerformers/"+cityId+"/1/2")
245  .header("X_Auth_Token", user1.getAuthToken())
246  .contentType(MediaType.APPLICATION_JSON)).
```

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 290 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: jsonObj
Enclosing Method: testsaveFeedback()
File: src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java:290
Taint Flags:

```
287  */
288  @Test
289  public void testsaveFeedback() throws Exception {
290  JSONObject jsonObj = TestUtil.readJsonObject(new File("src/test/resources/
CampaignJson.txt"));
291  String feedback = jsonObj.get("feedback").toString();
292  mockMvc.perform(post("http://localhost:8007/api/v1/crowd/campaigns/moderator/
saveFeedback/"+campaign.get_id())
293  .header("X_Auth_Token", user.getAuthToken())
```

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 245 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality

Poor Style: Value Never Read | Low

Package: com.cts.dcrowd.rest.app.repository

src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java, line 245 (Poor Style: Value Never Read) | Low

Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: jsonObj
Enclosing Method: testconfirmTeam()
File: src/test/java/com/cts/dcrowd/rest/app/repository/CampaignControllerTestNG.java:245
Taint Flags:

```
242  @Test
243  public void testconfirmTeam() throws Exception {
244
245  JSONObject jsonObj = TestUtil.readJsonObject(new File("src/test/resources/
CampaignJson.txt"));
246  String users = jsonObj.get("users").toString();
247  mockMvc.perform(post("http://localhost:8007/api/v1/crowd/campaigns/
confirmTeam/"+campaign.get_id())
248  .header("X_Auth_Token", user.getAuthToken())
```

Package: com.cts.dcrowd.rest.app.security

src/main/java/com/cts/dcrowd/rest/app/security/LdapUserDetailsMapper.java, line 217 (Poor Style: Value Never Read) | Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: city
Enclosing Method: insertNewUser()
File: src/main/java/com/cts/dcrowd/rest/app/security/LdapUserDetailsMapper.java:217
Taint Flags:

```
214  addressNew.setCity(context.getStringAttribute("l"));
215  City city = masterService.getCityIdByName(context.getStringAttribute("l"));
216  if(Util.isEmpty(city)) {
217  city = masterService.createMissingCity(context.getStringAttribute("l"));
218  }
219
220
```

src/main/java/com/cts/dcrowd/rest/app/security/ApplicationAuthProvider.java, line 144 (Poor Style: Value Never Read) | Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Poor Style: Value Never Read | Low

Package: com.cts.dcrowd.rest.app.security

src/main/java/com/cts/dcrowd/rest/app/security/ApplicationAuthProvider.java, line 144 (Poor Style: Value Never Read) | Low

Sink Details

Sink: VariableAccess: sslContextBuilder
Enclosing Method: authenticateEmployee()
File: src/main/java/com/cts/dcrowd/rest/app/security/ApplicationAuthProvider.java:144
Taint Flags:

```
141
142  if(password != null && password != ""){
143  requestFactory = new HttpComponentsClientHttpRequestFactory();
144  SSLContextBuilder sslContextBuilder = new SSLContextBuilder();
145  sslContextBuilder.loadTrustMaterial(null, new TrustStrategy() {
146  @Override
147  public boolean isTrusted(X509Certificate[] chain, String authType) throws
CertificateException {
```

src/main/java/com/cts/dcrowd/rest/app/security/ApplicationAuthProvider.java, line 151 (Poor Style: Value Never Read) | Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: sslConnectionSocketFactory
Enclosing Method: authenticateEmployee()
File: src/main/java/com/cts/dcrowd/rest/app/security/ApplicationAuthProvider.java:151
Taint Flags:

```
148  return true;
149  }
150  });
151  SSLConnectionSocketFactory sslConnectionSocketFactory = new
SSLConnectionSocketFactory(sslContextBuilder.build());
152  httpClient = HttpClients.custom().setSSLSocketFactory(sslConnectionSocketFactory).build();
153  requestFactory.setHttpClient(httpClient);
154  RestTemplate restTemplate = new RestTemplate(requestFactory);
```

src/main/java/com/cts/dcrowd/rest/app/security/ApplicationAuthProvider.java, line 154 (Poor Style: Value Never Read) | Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: restTemplate

Poor Style: Value Never Read — Low

Package: com.cts.dcrowd.rest.app.security

src/main/java/com/cts/dcrowd/rest/app/security/ApplicationAuthProvider.java, line 154 (Poor Style: Value Never Read) — Low

Enclosing Method: authenticateEmployee()
File: src/main/java/com/cts/dcrowd/rest/app/security/ApplicationAuthProvider.java:154
Taint Flags:

```
151  SSLConnectionSocketFactory sslConnectionSocketFactory = new
SSLConnectionSocketFactory(sslContextBuilder.build());
152  httpClient = HttpClients.custom().setSSLSocketFactory(sslConnectionSocketFactory).build();
153  requestFactory.setHttpClient(httpClient);
154  RestTemplate restTemplate = new RestTemplate(requestFactory);
155  MultiValueMap<String, String> map = new LinkedMultiValueMap<>();
156  map.add("username", username);
157  map.add("password", password);
```

src/main/java/com/cts/dcrowd/rest/app/security/LdapUserDetailsMapperWS.java, line 193 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: city
Enclosing Method: insertNewUserWebService()
File: src/main/java/com/cts/dcrowd/rest/app/security/LdapUserDetailsMapperWS.java:193
Taint Flags:

```
190  addressNew.setCity(ldapValues.get("city").toString());
191  City city = masterService.getCityIdByName(ldapValues.get("city").toString());
192  if (Util.isEmpty(city)) {
193  city = masterService.createMissingCity(ldapValues.get("city").toString());
194  }
195
196  }
```

Package: com.cts.dcrowd.rest.app.service

src/main/java/com/cts/dcrowd/rest/app/service/BadgeService.java, line 240 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: gfsPhoto
Enclosing Method: getImageByFileName()
File: src/main/java/com/cts/dcrowd/rest/app/service/BadgeService.java:240

Poor Style: Value Never Read — Low

Package: com.cts.dcrowd.rest.app.service

src/main/java/com/cts/dcrowd/rest/app/service/BadgeService.java, line 240 (Poor Style: Value Never Read) — Low

Taint Flags:

```
237  * @return the image by file name
238  */
239  private GridFSDBFile getImageByFileName(String fileName) {
240  GridFS gfsPhoto = new GridFS(db, BADGE_IMAGE_REPO);
241  return gfsPhoto.findOne(fileName);
242  }
243
```

src/main/java/com/cts/dcrowd/rest/app/service/BadgeService.java, line 251 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: gfsPhoto
Enclosing Method: removeImage()
File: src/main/java/com/cts/dcrowd/rest/app/service/BadgeService.java:251
Taint Flags:

```
248  */
249  public void removeImage(String fileName)
250  {
251  GridFS gfsPhoto = new GridFS(db, BADGE_IMAGE_REPO);
252  gfsPhoto.remove(gfsPhoto.findOne(fileName));
253  }
254
```

src/main/java/com/cts/dcrowd/rest/app/service/ProfileImageService.java, line 176 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: gfsPhoto
Enclosing Method: getImageByFileName()
File: src/main/java/com/cts/dcrowd/rest/app/service/ProfileImageService.java:176
Taint Flags:

```
173  * @return the image by file name
174  */
175  private GridFSDBFile getImageByFileName(String fileName) {
```

Poor Style: Value Never Read

Low

Package: com.cts.dcrowd.rest.app.service

src/main/java/com/cts/dcrowd/rest/app/service/ProfileImageService.java, line 176 (Poor Style: Value Never Read)

Low

```
176  GridFS gfsPhoto = new GridFS(db, USER_IMAGE_REPO);
177  GridFSDBFile gridFsFile = gfsPhoto.findOne(fileName);
178  return gridFsFile;
179  }
```

src/main/java/com/cts/dcrowd/rest/app/service/ReportGenerationService.java, line 41 (Poor Style: Value Never Read)

Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: user
Enclosing Method: generateBadgeReport()
File: src/main/java/com/cts/dcrowd/rest/app/service/ReportGenerationService.java:41
Taint Flags:

```
38  * @throws DcrowdException the dcrowd exception
39  */
40  public List<BadgeReport> generateBadgeReport(long badgeId) throws DcrowdException {
41  UserVO user = securityService.getLoggedUser();
42  List<BadgeReport> userBadgeList = appQueryProcessor.getUserBadgeToGenerateReport(badgeId);
43  userBadgeList = userBadgeList.stream().filter(p ->
p.getCompanyId().equals(user.getCompanyId())).collect(Collectors.toList());
44  return userBadgeList;
```

src/main/java/com/cts/dcrowd/rest/app/service/EmailService.java, line 162 (Poor Style: Value Never Read)

Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: emailJson
Enclosing Method: sendEmail()
File: src/main/java/com/cts/dcrowd/rest/app/service/EmailService.java:162
Taint Flags:

```
159  map.put("id", email.getId());
160  map.put("authToken", "ZW1haWxBdXRob3JpemF0aW9uVG9rZW4=");*/
161  try {
162  JSONObject emailJson = new JSONObject();
163  emailJson.put("authToken", authToken);
164  emailJson.put("bcc", email.getBcc());
165  emailJson.put("cc", email.getCc());
```

Poor Style: Value Never Read — Low

Package: com.cts.dcrowd.rest.app.service

src/main/java/com/cts/dcrowd/rest/app/service/EmailService.java, line 162 (Poor Style: Value Never Read) — Low

src/main/java/com/cts/dcrowd/rest/app/service/CommonService.java, line 40 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: user
Enclosing Method: getUserDetailFilter()
File: src/main/java/com/cts/dcrowd/rest/app/service/CommonService.java:40
Taint Flags:

```
37  * @throws DcrowdException the dcrowd exception
38  */
39  public List<CampaignUsersModel> getUserDetailFilter(Map<String, Object> filters) throws DcrowdException {
40  UserVO user = securityService.getLoggedUser();
41  List<CampaignUsersModel> userInfoList = appQueryProcessor.getUserDetailsByFilter(filters);
42  userInfoList = userInfoList.stream().filter(p -> p.getCompanyId().equals(user.getCompanyId())).collect(Collectors.toList());
43  return userInfoList;
```

Package: com.cts.dcrowd.rest.app.service.helper

src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java, line 539 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: campaignUser
Enclosing Method: getModeratorLevel1RoleList()
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java:539
Taint Flags:

```
536  }else if(Util.isEmpty(source1)) {
537  finalModeratorRoleList.addAll(source2);
538  } else {
539  for(CampaignUsersModel campaignUser : source1) {
540  finalModeratorRoleList = source2.stream().filter(p -> ! p.get_id().equals(campaignUser.get_id())).collect(Collectors.toList());
541  }
542  }
```

Poor Style: Value Never Read — Low

Package: com.cts.dcrowd.rest.app.service.helper

src/main/java/com/cts/dcrowd/rest/app/service/helper/ApplicationQueryProcessor.java, line 794 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: completeCriteria
Enclosing Method: buildFilterCriteria()
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/ApplicationQueryProcessor.java:794
Taint Flags:

```
791  @SuppressWarnings("unchecked")
792  public MatchOperation buildFilterCriteria(Map<String, Object> filters)
793  {
794  Criteria completeCriteria = new Criteria();
795  List<Criteria> andCriteria = new ArrayList<>();
796  List<Criteria> orCriteria = new ArrayList<>();
797  List<String> filterVal = new ArrayList<>();
```

src/main/java/com/cts/dcrowd/rest/app/service/helper/LdapWebserviceHelper.java, line 63 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: sslContextBuilder
Enclosing Method: extractLdapValueMap()
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/LdapWebserviceHelper.java:63
Taint Flags:

```
60  public Map<String, Object> extractLdapValueMap(String username)
61  {
62  HttpComponentsClientHttpRequestFactory requestFactory = new
HttpComponentsClientHttpRequestFactory();
63  SSLContextBuilder sslContextBuilder = new SSLContextBuilder();
64  Map<String, Object> ldapValues = null;
65  CloseableHttpClient httpClient = null;
66  try {
```

src/main/java/com/cts/dcrowd/rest/app/service/helper/ApplicationQueryProcessor.java, line 466 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Poor Style: Value Never Read — Low

Package: com.cts.dcrowd.rest.app.service.helper

src/main/java/com/cts/dcrowd/rest/app/service/helper/ApplicationQueryProcessor.java, line 466 (Poor Style: Value Never Read) — Low

Sink Details

Sink: VariableAccess: output
Enclosing Method: getUserDetailsByFilter()
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/ApplicationQueryProcessor.java:466
Taint Flags:

```
463
  .and("email").as("userEmailId").and("address.state").as("state").and("address.country").as("cc
464  .and("cityName").as("city").and("userProfile.completedProjects").as("projectsCompleted")
465  );
466  AggregationResults<CampaignUsersModel> output = mongoTemplate.aggregate(aggregation,
UserVO.class,CampaignUsersModel.class);
467  userModel = output.getMappedResults();
468  return userModel;
469  }
```

src/main/java/com/cts/dcrowd/rest/app/service/helper/LdapWebserviceHelper.java, line 76 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: restTemplate
Enclosing Method: extractLdapValueMap()
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/LdapWebserviceHelper.java:76
Taint Flags:

```
73  SSLConnectionSocketFactory sslConnectionSocketFactory = new
SSLConnectionSocketFactory(sslContextBuilder.build());
74  httpClient = HttpClients.custom().setSSLSocketFactory(sslConnectionSocketFactory).build();
75  requestFactory.setHttpClient(httpClient);
76  RestTemplate restTemplate = new RestTemplate(requestFactory);
77  MultiValueMap<String, String> map = new LinkedMultiValueMap<String, String>();
78
79  Map<String, Object> result = restTemplate.postForObject(searchUser + username, map,
Map.class);
```

src/main/java/com/cts/dcrowd/rest/app/service/helper/LdapWebserviceHelper.java, line 73 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Poor Style: Value Never Read — Low

Package: com.cts.dcrowd.rest.app.service.helper

src/main/java/com/cts/dcrowd/rest/app/service/helper/LdapWebserviceHelper.java, line 73 (Poor Style: Value Never Read) — Low

Sink: VariableAccess: sslConnectionSocketFactory
Enclosing Method: extractLdapValueMap()
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/LdapWebserviceHelper.java:73
Taint Flags:

```
70  return true;
71  }
72  });
73  SSLConnectionSocketFactory sslConnectionSocketFactory = new
SSLConnectionSocketFactory(sslContextBuilder.build());
74  httpClient = HttpClients.custom().setSSLSocketFactory(sslConnectionSocketFactory).build();
75  requestFactory.setHttpClient(httpClient);
76  RestTemplate restTemplate = new RestTemplate(requestFactory);
```

src/main/java/com/cts/dcrowd/rest/app/service/helper/ApplicationQueryProcessor.java, line 417 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: output
Enclosing Method: getUserBadgeToGenerateReport()
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/ApplicationQueryProcessor.java:417
Taint Flags:

```
414  and("cityName").as("city").and("username").as("employeeId"));
415  try
416  {
417  AggregationResults<BadgeReport> output = mongoTemplate.aggregate(aggregation,
UserVO.class, BadgeReport.class);
418  List<BadgeReport> badgeReport = output.getMappedResults();
419  return badgeReport;
420  }
```

src/main/java/com/cts/dcrowd/rest/app/service/helper/ApplicationQueryProcessor.java, line 716 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: user
Enclosing Method: curateModeratorCampaignListByUserId()
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/ApplicationQueryProcessor.java:716

Poor Style: Value Never Read — Low

Package: com.cts.dcrowd.rest.app.service.helper

src/main/java/com/cts/dcrowd/rest/app/service/helper/ApplicationQueryProcessor.java, line 716 (Poor Style: Value Never Read) — Low

Taint Flags:

```
713  private List<CampaignVO> curateModeratorCampaignListByUserId(List<CampaignVO>
campaignList, UserDetails userDetails, boolean isExternal) {
714  List<CampaignVO> curatedCampaignList = new ArrayList<>();
715  if(!isExternal){
716  UserVO user = (UserVO) userDetails;
717  for(CampaignVO campaign : campaignList) {
718  if(CollectionUtils.isNotEmpty(campaign.getModeratorLvl1UserIdList()) &&
campaign.getModeratorLvl1UserIdList().stream().anyMatch(e -> e.get_id() != null &&
e.get_id().equals(user.get_id()))) {
719  curatedCampaignList.add(campaign);
```

src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java, line 571 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: campaignUser
Enclosing Method: getModeratorLevel2RoleList()
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java:571
Taint Flags:

```
568  else if(Util.isEmpty(source1)) {
569  finalModeratorRoleList.addAll(source2);
570  }else {
571  for(CampaignUsersModel campaignUser : source1) {
572  finalModeratorRoleList = source2.stream().filter(p -> !
p.get_id().equals(campaignUser.get_id())).collect(Collectors.toList());
573  }
574  }
```

src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java, line 340 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: filterObject
Enclosing Method: orCriteriaFilterByCountryMatch()
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java:340
Taint Flags:

Poor Style: Value Never Read — Low

Package: com.cts.dcrowd.rest.app.service.helper

src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java, line 340 (Poor Style: Value Never Read) — Low

```
337  for(Object filterTagValue : (List<?>) filterTags.getValue()) {
338  try {
339  //1. Convert object to JSON
340  JSONObject filterObject = convertObjectToJson(filterTagValue);
341  if(filterObject.getString("country_code").equals(addressObject.getCountry())){
342  matchFound = true;
343  }
```

src/main/java/com/cts/dcrowd/rest/app/service/helper/ApplicationQueryProcessor.java, line 371 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: output
Enclosing Method: findLocationLeaderByXpPoints()
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/ApplicationQueryProcessor.java:371
Taint Flags:

```
368  .and("totalXPPoints").divide("noOfTesters").as("avgValue"),
369  sort(Sort.Direction.DESC,"avgValue"), limit(3));
370
371  AggregationResults<LocationLeader> output = mongoTemplate.aggregate(aggregation,
UserVO.class,
372  LocationLeader.class);
373
374  List<LocationLeader> locationLeaders = output.getMappedResults();
```

src/main/java/com/cts/dcrowd/rest/app/service/helper/ApplicationQueryProcessor.java, line 443 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: output
Enclosing Method: getRedeemDetailsToGenerateReport()
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/ApplicationQueryProcessor.java:443
Taint Flags:

```
440  unwind("userRedeem"),
441  sort(Sort.Direction.ASC,"userRedeem.createdDate"));
442
443  AggregationResults<RedeemReport> output = mongoTemplate.aggregate(aggregation,
```

Poor Style: Value Never Read — Low

Package: com.cts.dcrowd.rest.app.service.helper

src/main/java/com/cts/dcrowd/rest/app/service/helper/ApplicationQueryProcessor.java, line 443 (Poor Style: Value Never Read) — Low

```
UserVO.class,RedeemReport.class);
444  List<RedeemReport> redeemReport = new ArrayList<RedeemReport>();
445  redeemReport = output.getMappedResults();
446  return redeemReport;
```

src/main/java/com/cts/dcrowd/rest/app/service/helper/ApplicationQueryProcessor.java, line 726 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: externalUser
Enclosing Method: curateModeratorCampaignListByUserId()
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/ApplicationQueryProcessor.java:726
Taint Flags:

```
723  }
724  }
725  }else {
726  ExternalUserVO externalUser = (ExternalUserVO) userDetails;
727  for(CampaignVO campaign : campaignList) {
728  if(CollectionUtils.isNotEmpty(campaign.getExternalModeratorLvl1()) &&
campaign.getExternalModeratorLvl1().stream().anyMatch(e -> e.get_id() != null &&
e.get_id().equals(externalUser.get_id()))) {
729  curatedCampaignList.add(campaign);
```

Package: com.cts.dcrowd.rest.app.util

src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java, line 140 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: parser
Enclosing Method: readJsonObject()
File: src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java:140
Taint Flags:

```
137  */
138  public static JSONObject readJsonObject(File file) {
139
140  JSONParser parser = new JSONParser();
```

Poor Style: Value Never Read — Low

Package: com.cts.dcrowd.rest.app.util

src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java, line 140 (Poor Style: Value Never Read) — Low

```
141  try (FileReader fileReader = new FileReader(file)) {
142  Object obj = parser.parse(fileReader);
143  JSONObject jsonObject = (JSONObject) obj;
```

src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java, line 33 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: encrypted
Enclosing Method: main()
File: src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java:33
Taint Flags:

```
30  * @return the JSON object
31  */
32  public static void main(String[] args) {
33  String encrypted = generateBlowFishEncryptValue("password");
34  // System.out.println(generateBlowFishDecryptValue("Y3isgEgqkQUyh525Wy5M+w=="));
35  
36  /* String encodedValue = base64Encoder("password".getBytes());
```

Package: com.cts.dcrowd.rest.app.utils

src/main/java/com/cts/dcrowd/rest/app/utils/ImageUtil.java, line 96 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: gfsPhoto
Enclosing Method: getImageByFileId()
File: src/main/java/com/cts/dcrowd/rest/app/utils/ImageUtil.java:96
Taint Flags:

```
93  */
94  public static GridFSDBFile getImageByFileId(DB db, String fileId, String repoName) {
95  ObjectId objFileId = new ObjectId(fileId);
96  GridFS gfsPhoto = new GridFS(db, repoName);
97  return gfsPhoto.findOne(objFileId);
98  }
99  
```

Poor Style: Value Never Read — Low

Package: com.cts.dcrowd.rest.app.utils

src/main/java/com/cts/dcrowd/rest/app/utils/ImageUtil.java, line 96 (Poor Style: Value Never Read) — Low

src/main/java/com/cts/dcrowd/rest/app/utils/ImageUtil.java, line 76 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: gfsPhoto
Enclosing Method: createNewFile()
File: src/main/java/com/cts/dcrowd/rest/app/utils/ImageUtil.java:76
Taint Flags:

```
73  * @throws IOException Signals that an I/O exception has occurred.
74  */
75  private static GridFSInputFile createNewFile(DB db, MultipartFile file, String
repoName,String CampaginId, String DefectId) throws IOException {
76  GridFS gfsPhoto = new GridFS(db, repoName);
77  GridFSInputFile gfsFile = gfsPhoto.createFile(file.getBytes());
78  gfsFile.put("campaignId", CampaginId);
79  gfsFile.put("defectId", DefectId);
```

src/main/java/com/cts/dcrowd/rest/app/utils/Util.java, line 446 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: fScore
Enclosing Method: setFuzzyScore()
File: src/main/java/com/cts/dcrowd/rest/app/utils/Util.java:446
Taint Flags:

```
443  public static double setFuzzyScore(String defectTerm, String defectQuery) {
444  double score = 0.0;
445  try {
446  FuzzyScore fScore = new FuzzyScore(Locale.getDefault());
447  score = fScore.fuzzyScore(defectTerm, defectQuery);
448  } catch (IllegalArgumentException ile) {
449  score = 0;
```

Poor Style: Value Never Read — Low

Package: com.cts.dcrowd.rest.app.utils

src/main/java/com/cts/dcrowd/rest/app/utils/Util.java, line 637 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: sslConnectionSocketFactory
Enclosing Method: connectExternalUser()
File: src/main/java/com/cts/dcrowd/rest/app/utils/Util.java:637
Taint Flags:

```
634  return true;
635  }
636  });
637  sslConnectionSocketFactory = new SSLConnectionSocketFactory(sslContextBuilder.build());
638  httpClient = HttpClients.custom().setSSLSocketFactory(sslConnectionSocketFactory).build();
639  requestFactory.setHttpClient(httpClient);
640  restTemplate = new RestTemplate(requestFactory);
```

src/main/java/com/cts/dcrowd/rest/app/utils/Util.java, line 640 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: restTemplate
Enclosing Method: connectExternalUser()
File: src/main/java/com/cts/dcrowd/rest/app/utils/Util.java:640
Taint Flags:

```
637  sslConnectionSocketFactory = new SSLConnectionSocketFactory(sslContextBuilder.build());
638  httpClient = HttpClients.custom().setSSLSocketFactory(sslConnectionSocketFactory).build();
639  requestFactory.setHttpClient(httpClient);
640  restTemplate = new RestTemplate(requestFactory);
641  result = restTemplate.postForObject(url, email , Map.class, " ");
642  LOGGER.debug("Inside Email Sending Class...Result..."+result.toString());
643  if (result != null && result.containsKey("success")) {
```

src/main/java/com/cts/dcrowd/rest/app/utils/ImageUtil.java, line 45 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Poor Style: Value Never Read — Low

Package: com.cts.dcrowd.rest.app.utils

src/main/java/com/cts/dcrowd/rest/app/utils/ImageUtil.java, line 45 (Poor Style: Value Never Read) — Low

Sink Details

Sink: VariableAccess: gfsFile
Enclosing Method: saveImage()
File: src/main/java/com/cts/dcrowd/rest/app/utils/ImageUtil.java:45
Taint Flags:

```
42  removeImage(db, imageFileName, repoName);
43  gfsFile = createNewFile(db, file, repoName);
44  }*/
45  gfsFile = createNewFile(db, file, repoName,CampaginId,DefectId);
46  imageFileId = gfsFile.getId().toString();
47  } catch (Exception e) {
48  throw new DcrowdException("Image save failed");
```

src/main/java/com/cts/dcrowd/rest/app/utils/ImageUtil.java, line 123 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: gfsPhoto
Enclosing Method: removeImage()
File: src/main/java/com/cts/dcrowd/rest/app/utils/ImageUtil.java:123
Taint Flags:

```
120  public static void removeImage(DB db, String fileId, String repoName)
121  {
122  ObjectId objFileId = new ObjectId(fileId);
123  GridFS gfsPhoto = new GridFS(db, repoName);
124  if(gfsPhoto.findOne(objFileId)!=null){
125  gfsPhoto.remove(gfsPhoto.findOne(objFileId));
126  }
```

src/main/java/com/cts/dcrowd/rest/app/utils/ImageUtil.java, line 109 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: gfsPhoto
Enclosing Method: getImageByFileName()
File: src/main/java/com/cts/dcrowd/rest/app/utils/ImageUtil.java:109

Poor Style: Value Never Read — Low

Package: com.cts.dcrowd.rest.app.utils

src/main/java/com/cts/dcrowd/rest/app/utils/ImageUtil.java, line 109 (Poor Style: Value Never Read) — Low

Taint Flags:

```
106  * @return the image by file name
107  */
108  public static GridFSDBFile getImageByFileName(DB db, String fileName, String repoName) {
109  GridFS gfsPhoto = new GridFS(db, repoName);
110  return gfsPhoto.findOne(fileName);
111  }
112
```

src/main/java/com/cts/dcrowd/rest/app/utils/Util.java, line 625 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: sslContextBuilder
Enclosing Method: connectExternalUser()
File: src/main/java/com/cts/dcrowd/rest/app/utils/Util.java:625
Taint Flags:

```
622  public static boolean connectExternalUser(String url, String email){
623  boolean output = false;
624  HttpComponentsClientHttpRequestFactory requestFactory = new
HttpComponentsClientHttpRequestFactory();
625  SSLContextBuilder sslContextBuilder = new SSLContextBuilder();
626  CloseableHttpClient httpClient = null;
627  RestTemplate restTemplate;
628  Map<String, Object> result;
```

Package: com.cts.dcrowd.rest.defect.app.controller

src/main/java/com/cts/dcrowd/rest/defect/app/controller/DefectEntryController.java, line 642 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: mapper
Enclosing Method: exportDefectsToJira()
File: src/main/java/com/cts/dcrowd/rest/defect/app/controller/DefectEntryController.java:642
Taint Flags:

Poor Style: Value Never Read — Low

Package: com.cts.dcrowd.rest.defect.app.controller

src/main/java/com/cts/dcrowd/rest/defect/app/controller/ DefectEntryController.java, line 642 (Poor Style: Value Never Read) — Low

```
639  LOGGER.debug("Received request to saveDefectEntry");
640  try
641  {
642  ObjectMapper mapper = new ObjectMapper();
643  String jsonData = mapper.writeValueAsString(jiradetails);
644  JsonNode rootNode = mapper.readTree(jsonData);
645  jiraService.getJiraAuthentication(rootNode);
```

Package: com.cts.dcrowd.rest.defect.app.jira.services

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 551 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: jArray
Enclosing Method: getPostData()
File: src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java:551
Taint Flags:

```
548  String dCrowdLabel = entry1.getValue().optString("label");
549  try {
550  String custObjVal = objectMapper.writeValueAsString(entry.getValue());
551  JSONArray jArray = new JSONArray(custObjVal);
552  for (int index = 0; index < jArray.length(); index++) {
553  String label = jArray.getJSONObject(index).optString("label");
554  if (dCrowdLabel.equals(label)) {
```

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 183 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: issueRespObj
Enclosing Method: addCommentsAndAttachments()
File: src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java:183
Taint Flags:

```
180  private void addCommentsAndAttachments(JsonNode rootNode, HttpHost targetHost,
HttpClientContext context,
181  HttpClient httpClient, String issueResponse)
```

Poor Style: Value Never Read — Low

Package: com.cts.dcrowd.rest.defect.app.jira.services

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 183 (Poor Style: Value Never Read) — Low

```
182  throws JSONException, IOException, ClientProtocolException, HttpResponseException,
DcrowdException {
183  JSONObject issueRespObj = new JSONObject(issueResponse);
184  JSONArray createdIssues = issueRespObj.optJSONArray("issues");
185  for (int index = 0; index < createdIssues.length(); index++) {
186  JSONObject jiraIssue = createdIssues.optJSONObject(index);
```

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 194 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: jsonNode
Enclosing Method: addCommentsAndAttachments()
File: src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java:194
Taint Flags:

```
191  JSONObject summaryData = new JSONObject(response);
192  String summary = summaryData.optJSONObject("fields").optString("summary");
193  JsonNode defectNode = rootNode.path("defectList");
194  for (JsonNode jsonNode : defectNode) {
195  String defectId = jsonNode.path("id").asText();
196  DefectEntry defectEntry = dcrowdDefectService.fetchByDefectId(defectId);
197  if (defectEntry != null && summary.contains(defectEntry.getDefectName())) {
```

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 265 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: commentArray
Enclosing Method: doAddCommentsPost()
File: src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java:265
Taint Flags:

```
262  String issueURL, Comment comment) {
263  JSONObject jobj = new JSONObject();
264  JSONObject updateObj = new JSONObject();
265  JSONArray commentArray = new JSONArray();
266  JSONObject addObj = new JSONObject();
267  try {
```

Poor Style: Value Never Read — Low

Package: com.cts.dcrowd.rest.defect.app.jira.services

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 265 (Poor Style: Value Never Read) — Low

```
268  addObj.put("add", new JSONObject().put("body", comment.getCommentDescription()));
```

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 346 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: arrayOptions
Enclosing Method: addCustomFieldOptions()
File: src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java:346
Taint Flags:

```
343  String custId = String.valueOf(responseJson.opt("id"));
344
345  String options = String.valueOf(customFields.opt("options"));
346  JSONArray arrayOptions = new JSONArray(options);
347  JSONArray array = new JSONArray();
348  for (int index = 0; index < arrayOptions.length(); index++) {
349  JSONObject optionsJson = new JSONObject();
```

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 587 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: jsonNode
Enclosing Method: getDefect()
File: src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java:587
Taint Flags:

```
584
585  JsonNode defectNode = rootNode.path("defectList");
586  DefectEntry defectEntry = null;
587  for (JsonNode jsonNode : defectNode) {
588  String defectId = jsonNode.path("id").asText();
589  defectEntry = dcrowdDefectService.fetchByDefectId(defectId);
590  if (defectEntry != null && defectEntry.getDefectName().equals(defectName)) {
```

Poor Style: Value Never Read — Low

Package: com.cts.dcrowd.rest.defect.app.jira.services

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 342 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: responseJson
Enclosing Method: addCustomFieldOptions()
File: src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java:342
Taint Flags:

```
339  HttpClient httpClient, JSONObject customFields, HttpResponse custResponse)
340  throws HttpResponseException, IOException, JSONException, ClientProtocolException,
DcrowdException {
341  String resp = new BasicResponseHandler().handleResponse(custResponse);
342  JSONObject responseJson = new JSONObject(resp);
343  String custId = String.valueOf(responseJson.opt("id"));
344
345  String options = String.valueOf(customFields.opt("options"));
```

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 122 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: avail
Enclosing Method: getJiraAuthentication()
File: src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java:122
Taint Flags:

```
119  List<JSONObject> custFields = new ArrayList<>();
120
121  List<String> availFieldNames = new ArrayList<>();
122  for (JSONObject avail : availFields) {
123  availFieldNames.add(String.valueOf(avail.opt("name")));
124  }
125  for (JSONObject frmKeys : formFields) {
```

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 139 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Poor Style: Value Never Read — Low

Package: com.cts.dcrowd.rest.defect.app.jira.services

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 139 (Poor Style: Value Never Read) — Low

Sink Details

Sink: VariableAccess: avail
Enclosing Method: getJiraAuthentication()
File: src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java:139
Taint Flags:

```
136  List<String> custoFields = new ArrayList<>();
137
138  List<JSONObject> createdFields = new ArrayList<>();
139  for (JSONObject avail : allCustomFields) {
140  custoFields.add(String.valueOf(avail.opt("name")));
141  }
142  for (JSONObject frmKeys : formFields) {
```

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 264 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: updateObj
Enclosing Method: doAddCommentsPost()
File: src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java:264
Taint Flags:

```
261  private void doAddCommentsPost(HttpHost targetHost, HttpClientContext context, HttpClient
httpClient,
262  String issueURL, Comment comment) {
263  JSONObject jobj = new JSONObject();
264  JSONObject updateObj = new JSONObject();
265  JSONArray commentArray = new JSONArray();
266  JSONObject addObj = new JSONObject();
267  try {
```

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 347 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: array
Enclosing Method: addCustomFieldOptions()
File: src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java:347

Poor Style: Value Never Read

Low

Package: com.cts.dcrowd.rest.defect.app.jira.services

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 347 (Poor Style: Value Never Read)

Low

Taint Flags:

```
344
345  String options = String.valueOf(customFields.opt("options"));
346  JSONArray arrayOptions = new JSONArray(options);
347  JSONArray array = new JSONArray();
348  for (int index = 0; index < arrayOptions.length(); index++) {
349  JSONObject optionsJson = new JSONObject();
350  optionsJson.put("name", arrayOptions.get(index));
```

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 263 (Poor Style: Value Never Read)

Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: jobj
Enclosing Method: doAddCommentsPost()
File: src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java:263
Taint Flags:

```
260  */
261  private void doAddCommentsPost(HttpHost targetHost, HttpClientContext context, HttpClient
httpClient,
262  String issueURL, Comment comment) {
263  JSONObject jobj = new JSONObject();
264  JSONObject updateObj = new JSONObject();
265  JSONArray commentArray = new JSONArray();
266  JSONObject addObj = new JSONObject();
```

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 517 (Poor Style: Value Never Read)

Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: array
Enclosing Method: getPostData()
File: src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java:517
Taint Flags:

```
514  public JSONObject getPostData(JsonNode rootNode, Map<String, JSONObject> map)
515  throws DcrowdException, JSONException {
```

Poor Style: Value Never Read — Low

Package: com.cts.dcrowd.rest.defect.app.jira.services

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 517 (Poor Style: Value Never Read) — Low

```
516  JSONObject issues = new JSONObject();
517  JSONArray array = new JSONArray();
518
519  JsonNode defectNode = rootNode.path("defectList");
520  for (JsonNode jsonNode : defectNode) {
```

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 387 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: issueObj
Enclosing Method: doCreateBulkIssuePost()
File: src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java:387
Taint Flags:

```
384  "http://" + rootNode.path("domainName").asText() + ":8080/rest/api/2/issue/bulk");
385
386  JSONObject issueObj = new JSONObject();
387  issueObj = getPostData(rootNode, map);
388
389  HttpEntity entity = new StringEntity(issueObj.toString(), ContentType.APPLICATION_JSON);
390  httpPost.setEntity(entity);
```

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 359 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: custOptResponse
Enclosing Method: addCustomFieldOptions()
File: src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java:359
Taint Flags:

```
356
357  HttpEntity custFieldoption = new StringEntity(array.toString(),
ContentType.APPLICATION_JSON);
358  custFieldOptionsPost.setEntity(custFieldoption);
359  HttpResponse custOptResponse = httpClient.execute(targetHost, custFieldOptionsPost,
context);
360  if (custOptResponse.getStatusLine().getStatusCode() != 204) {
```

Poor Style: Value Never Read — Low

Package: com.cts.dcrowd.rest.defect.app.jira.services

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 359 (Poor Style: Value Never Read) — Low

```
361  throw new DcrowdException("Error in creating custom fields");
362  }
```

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 468 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: jArray
Enclosing Method: getAvailableCustomFields()
File: src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java:468
Taint Flags:

```
465  String response = new BasicResponseHandler().handleResponse(httpResponseGet);
466  List<JSONObject> availableCustomFields = new ArrayList<>();
467  try {
468  JSONArray jArray = new JSONArray(response);
469  for (int index = 0; index < jArray.length(); index++) {
470  if ((boolean) jArray.getJSONObject(index).get("custom")) {
471  availableCustomFields.add(jArray.getJSONObject(index));
```

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 520 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: jsonNode
Enclosing Method: getPostData()
File: src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java:520
Taint Flags:

```
517  JSONArray array = new JSONArray();
518
519  JsonNode defectNode = rootNode.path("defectList");
520  for (JsonNode jsonNode : defectNode) {
521  String defectId = jsonNode.path("id").asText();
522  DefectEntry defectEntry = dcrowdDefectService.fetchByDefectId(defectId);
523
```

Poor Style: Value Never Read — Low

Package: com.cts.dcrowd.rest.defect.app.jira.services

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 303 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: customFieldData
Enclosing Method: createCustomFields()
File: src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java:303
Taint Flags:

```
300  String searcherKey = getSearcherKey(component);
301  String typeKey = getTypeKey(component);
302
303  JSONObject customFieldData = new JSONObject();
304  customFieldData.put("searcherKey", searcherKey);
305  customFieldData.put("name", String.valueOf(customFields.opt("label")));
306  customFieldData.put("description", String.valueOf(customFields.opt("description")));
```

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 630 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: defectConfigFields
Enclosing Method: getFormData()
File: src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java:630
Taint Flags:

```
627  DefectLayoutConfig defectLayConfig = defectLayoutConfigService
628  .getDefectLayoutByCampaignId(rootNode.path("defectCampaignId").asText());
629  String formData = String.valueOf(defectLayConfig.getConfigParamMap().get("param1"));
630  JSONArray defectConfigFields = new JSONArray(formData);
631  String defLabel = null;
632  List<JSONObject> formFields = new ArrayList<>();
633  for (int i = 0; i < defectConfigFields.length(); i++) {
```

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 191 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Poor Style: Value Never Read — Low

Package: com.cts.dcrowd.rest.defect.app.jira.services

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 191 (Poor Style: Value Never Read) — Low

Sink Details

Sink: VariableAccess: summaryData
Enclosing Method: addCommentsAndAttachments()
File: src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java:191
Taint Flags:

```
188  HttpGet httpGet = new HttpGet(issueURL);
189  HttpResponse httpResponseGet = httpClient.execute(targetHost, httpGet, context);
190  String response = new BasicResponseHandler().handleResponse(httpResponseGet);
191  JSONObject summaryData = new JSONObject(response);
192  String summary = summaryData.optJSONObject("fields").optString("summary");
193  JsonNode defectNode = rootNode.path("defectList");
194  for (JsonNode jsonNode : defectNode) {
```

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 184 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: createdIssues
Enclosing Method: addCommentsAndAttachments()
File: src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java:184
Taint Flags:

```
181  HttpClient httpClient, String issueResponse)
182  throws JSONException, IOException, ClientProtocolException, HttpResponseException,
DcrowdException {
183  JSONObject issueRespObj = new JSONObject(issueResponse);
184  JSONArray createdIssues = issueRespObj.optJSONArray("issues");
185  for (int index = 0; index < createdIssues.length(); index++) {
186  JSONObject jiraIssue = createdIssues.optJSONObject(index);
187  String issueURL = jiraIssue.optString("self");
```

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 186 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: jiraIssue
Enclosing Method: addCommentsAndAttachments()
File: src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java:186

Poor Style: Value Never Read

Low

Package: com.cts.dcrowd.rest.defect.app.jira.services

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 186 (Poor Style: Value Never Read)

Low

Taint Flags:

```
183  JSONObject issueRespObj = new JSONObject(issueResponse);
184  JSONArray createdIssues = issueRespObj.optJSONArray("issues");
185  for (int index = 0; index < createdIssues.length(); index++) {
186  JSONObject jiraIssue = createdIssues.optJSONObject(index);
187  String issueURL = jiraIssue.optString("self");
188  HttpGet httpGet = new HttpGet(issueURL);
189  HttpResponse httpResponseGet = httpClient.execute(targetHost, httpGet, context);
```

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 537 (Poor Style: Value Never Read)

Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: objectMapper
Enclosing Method: getPostData()
File: src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java:537
Taint Flags:

```
534  fields.put("fields", project);
535  issues.put("issueUpdates", array.put(fields));
536
537  ObjectMapper objectMapper = new ObjectMapper();
538  Map<String, Object> defectMap = objectMapper.convertValue(defectEntry, Map.class);
539
540  for (Map.Entry<String, JSONObject> entry1 : map.entrySet()) {
```

Package: com.cts.dcrowd.rest.defect.app.services.helper

src/main/java/com/cts/dcrowd/rest/defect/app/services/helper/DefectsQueryProcessor.java, line 470 (Poor Style: Value Never Read)

Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: output
Enclosing Method: aggregateDefectsCountForModerator2()
File: src/main/java/com/cts/dcrowd/rest/defect/app/services/helper/DefectsQueryProcessor.java:470
Taint Flags:

```
467  project("value", "key"),
```

Poor Style: Value Never Read — Low

Package: com.cts.dcrowd.rest.defect.app.services.helper

src/main/java/com/cts/dcrowd/rest/defect/app/services/helper/DefectsQueryProcessor.java, line 470 (Poor Style: Value Never Read) — Low

```
468  sort(Sort.Direction.DESC, "key"));
469
470  AggregationResults<PieChart> output = mongoTemplate.aggregate(aggregation, DefectEntry.class,
471  PieChart.class);
472
473  return output.getMappedResults();
```

src/main/java/com/cts/dcrowd/rest/defect/app/services/helper/DefectsQueryProcessor.java, line 603 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: groupResults
Enclosing Method: getDefectsCountByUserId()
File: src/main/java/com/cts/dcrowd/rest/defect/app/services/helper/DefectsQueryProcessor.java:603
Taint Flags:

```
600  Aggregation.limit(limit)
601  );
602
603  AggregationResults<TopUserDefect> groupResults= mongoTemplate.aggregate(aggregation,DefectEntry.class,TopUserDefect.class);
604  List<TopUserDefect> topUserList = groupResults.getMappedResults();
605
606  /*for(TopUserDefect topUser : topUserList) {
```

src/main/java/com/cts/dcrowd/rest/defect/app/services/helper/DefectsQueryProcessor.java, line 441 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: output
Enclosing Method: aggregateDefectsCount()
File: src/main/java/com/cts/dcrowd/rest/defect/app/services/helper/DefectsQueryProcessor.java:441
Taint Flags:

```
438  project("value", "key"),
439  sort(Sort.Direction.DESC, "key"));
440
441  AggregationResults<PieChart> output = mongoTemplate.aggregate(aggregation,
```

Poor Style: Value Never Read — Low

Package: com.cts.dcrowd.rest.defect.app.services.helper

src/main/java/com/cts/dcrowd/rest/defect/app/services/helper/ DefectsQueryProcessor.java, line 441 (Poor Style: Value Never Read) — Low

```
DefectEntry.class,
442   PieChart.class);
443
444   return output.getMappedResults();
```

Package: com.cts.dcrowd.rest.defect.app.services.impl

src/main/java/com/cts/dcrowd/rest/defect/app/services/impl/ DefaultDefectServiceImpl.java, line 433 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: externalUser
Enclosing Method: addWatcherToDefect()
File: src/main/java/com/cts/dcrowd/rest/defect/app/services/impl/DefaultDefectServiceImpl.java:433
Taint Flags:

```
430   watcher.setWatcherUserImageFileUrl(user.getImageData().getImageFileUrl());
431   }
432   } else {
433   externalUser = externalUserRepository.findByUsername(watcher.getWatcherUserId());
434   watcher.setWatcherFullName(Util.getLoggedUserFullName());
435   watcher.setWatcherInitial(getUserInitial());
436   }
```

src/main/java/com/cts/dcrowd/rest/defect/app/services/impl/ DcrowdDefectEntryServiceImpl.java, line 616 (Poor Style: Value Never Read) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: VariableAccess: defectLayout
Enclosing Method: exportCuratedDefectList()
File: src/main/java/com/cts/dcrowd/rest/defect/app/services/impl/DcrowdDefectEntryServiceImpl.java:616
Taint Flags:

```
613   DefectLayoutConfig
defectLayoutConfig=defectLayoutConfigRepository.findByCampaignId(campaignId);
614   Map<String, Object> configParamMap = defectLayoutConfig.getConfigParamMap();
615   String data=String.valueOf(configParamMap.get("param1"));
616   JSONArray defectLayout=new JSONArray(data);
617
```

Poor Style: Value Never Read — Low

Package: com.cts.dcrowd.rest.defect.app.services.impl

src/main/java/com/cts/dcrowd/rest/defect/app/services/impl/DcrowdDefectEntryServiceImpl.java, line 616 (Poor Style: Value Never Read) — Low

```
618   List<String> defectFormFields=new ArrayList<>();
619   List<String> defaultFields = Arrays.asList("CAMPAIGNNAME", "DEFECTNAME",
"DEFECTSTATUS","LOGGEDDATE","LASTUPDATEDDATE","LASTUPDATEDBY","LOGGEDBY","LOGGEDUSERID","LOGGEDE
```

Privacy Violation: Heap Inspection (1 issue)

Abstract

Storing sensitive data in a `String` object makes it impossible to reliably purge the data from memory.

Explanation

Sensitive data (such as passwords, social security numbers, credit card numbers etc) stored in memory can be leaked if memory is not cleared after use. Often, `Strings` are used store sensitive data, however, since `String` objects are immutable, removing the value of a `String` from memory can only be done by the JVM garbage collector. The garbage collector is not required to run unless the JVM is low on memory, so there is no guarantee as to when garbage collection will take place. In the event of an application crash, a memory dump of the application might reveal sensitive data. **Example 1:** The following code converts a password from a character array to a `String`.

```
private JPasswordField pf;
...
final char[] password = pf.getPassword();
...
String passwordAsString = new String(password);
```

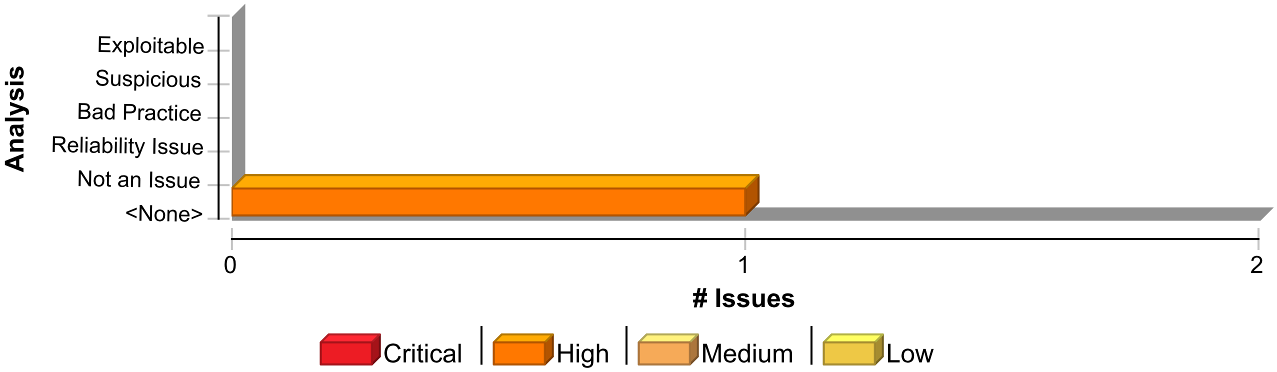
This category was derived from the Cigital Java Rulepack. http://www.cigital.com/

Recommendation

Always be sure to clear sensitive data that is no longer needed. Instead of storing sensitive data in immutable objects such as `Strings`, use byte arrays or character arrays that you can clear programmatically. **Example 2:** The following code clears memory after a password is used.

```
private JPasswordField pf;
...
final char[] password = pf.getPassword();
// use the password
...
// erase when finished
Arrays.fill(password, ' ');
```

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Privacy Violation: Heap Inspection	1	0	0	1
Total	**1**	**0**	**0**	**1**

Privacy Violation: Heap Inspection — High

Package: com.cts.dcrowd.rest.app.utils

src/main/java/com/cts/dcrowd/rest/app/utils/Util.java, line 530 (Privacy Violation: Heap Inspection) — High

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Data Flow)

Source Details

Source: Read this.password
From: com.cts.dcrowd.rest.app.domain.ExternalUserVO.getPassword
File: src/main/java/com/cts/dcrowd/rest/app/domain/ExternalUserVO.java:200

```
197  */
198  @JsonIgnore
199  public String getPassword() {
200  return password;
201  }
202
203  /**
```

Sink Details

Sink: java.lang.String.String()
Enclosing Method: generateBlowFishDecryptValue()
File: src/main/java/com/cts/dcrowd/rest/app/utils/Util.java:530
Taint Flags: PRIVATE, WEAKCRYPTO

```
527  {
528  String[] basicAuthString = null;
529  try {
530  basicAuthString = new String(Base64.getDecoder().decode(encryptedValue),
StandardCharsets.UTF_8).split("fastestCustomers");
531  } catch (Exception e) {
532  e.printStackTrace();
533  }
```

Redundant Null Check (5 issues)

Abstract

The program can potentially dereference a null pointer, thereby causing a null pointer exception.

Explanation

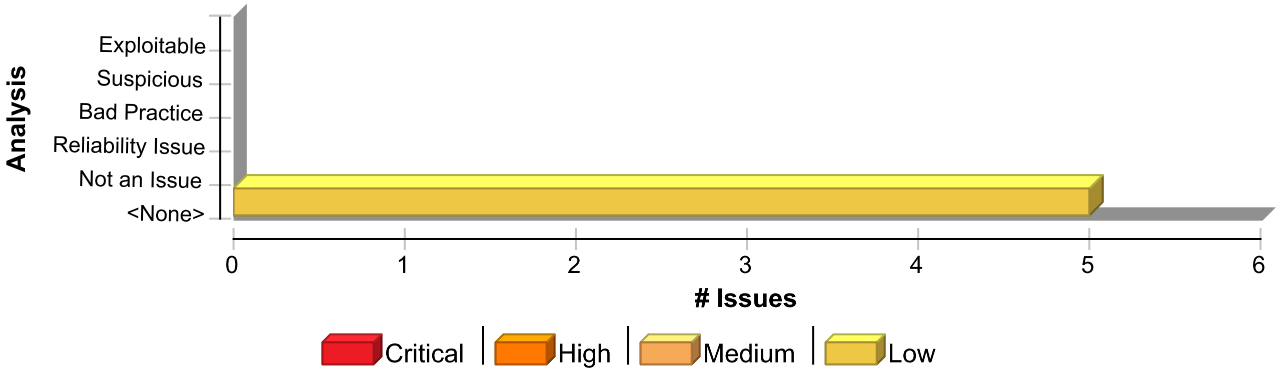
Null pointer exceptions usually occur when one or more of the programmer's assumptions is violated. A check-after-dereference error occurs when a program dereferences an object that can be `null` before checking if the object is `null`. Most null pointer issues result in general software reliability problems, but if attackers can intentionally trigger a null pointer dereference, they can use the resulting exception to bypass security logic or to cause the application to reveal debugging information that will be valuable in planning subsequent attacks. **Example:** In the following code, the programmer assumes that the variable `foo` is not `null` and confirms this assumption by dereferencing the object. However, the programmer later contradicts the assumption by checking `foo` against `null`. If `foo` can be `null` when it is checked in the `if` statement then it can also be `null` when it is dereferenced and might cause a null pointer exception. Either the dereference is unsafe or the subsequent check is unnecessary.

```
foo.setBar(val);
...
if (foo != null) {
    ...
}
```

Recommendation

Implement careful checks before dereferencing objects that might be `null`. When possible, abstract null checks into wrappers around code that manipulates resources to ensure that they are applied in all cases and to minimize the places where mistakes can occur.

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Redundant Null Check	5	0	0	5
Total	**5**	**0**	**0**	**5**

Redundant Null Check — Low

Package: com.cts.dcrowd.rest.app.service

src/main/java/com/cts/dcrowd/rest/app/service/CampaignService.java, line 609 (Redundant Null Check) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Control Flow)

Sink Details

Sink: Dereferenced : externalModerators
Enclosing Method: setExternalModLvl()
File: src/main/java/com/cts/dcrowd/rest/app/service/CampaignService.java:609
Taint Flags:

```
606  List<ExternalUsersModel> externalModeratorslvl1 = campaign.getExternalModeratorLvl1();
607  List<ExternalUsersModel> externalModeratorslvl2 = campaign.getExternalModeratorLvl2();
608  if(externalModeratorslvl1!=null) {
609  externalModerators.addAll(externalModeratorslvl1);
610  }
611  if(externalModeratorslvl2!=null) {
612  externalModerators.addAll(externalModeratorslvl2);
```

src/main/java/com/cts/dcrowd/rest/app/service/CampaignService.java, line 990 (Redundant Null Check) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Control Flow)

Sink Details

Sink: Dereferenced : campaign
Enclosing Method: confirmTeam()
File: src/main/java/com/cts/dcrowd/rest/app/service/CampaignService.java:990
Taint Flags:

```
987  boolean userFlag = false;
988  List<String> userListObj = (List<String>) userList;
989  // UserDetails userDetails;
990  campaign.getConfirmedTeamMembers().clear();
991  try {
992  for (String userId : userListObj) {
993  try {
```

src/main/java/com/cts/dcrowd/rest/app/service/CampaignService.java, line 990 (Redundant Null Check) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Control Flow)

Redundant Null Check — Low

Package: com.cts.dcrowd.rest.app.service

src/main/java/com/cts/dcrowd/rest/app/service/CampaignService.java, line 990 (Redundant Null Check) — Low

Sink Details

Sink: Dereferenced : campaign
Enclosing Method: confirmTeam()
File: src/main/java/com/cts/dcrowd/rest/app/service/CampaignService.java:990
Taint Flags:

```
987  boolean userFlag = false;
988  List<String> userListObj = (List<String>) userList;
989  // UserDetails userDetails;
990  campaign.getConfirmedTeamMembers().clear();
991  try {
992  for (String userId : userListObj) {
993  try {
```

src/main/java/com/cts/dcrowd/rest/app/service/CampaignService.java, line 1008 (Redundant Null Check) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Control Flow)

Sink Details

Sink: Dereferenced : campaign
Enclosing Method: confirmTeam()
File: src/main/java/com/cts/dcrowd/rest/app/service/CampaignService.java:1008
Taint Flags:

```
1005  user.setModifiedBy(Util.buildFullName(securityService.getLoggedUser()));
1006  userService.createUpdateUser(user);
1007  activityHandler.setActivityToUser(user, activityHandler.addActivityWithOneParam(
1008  String.valueOf(campaign.getName()), EventType.NOMINATION_ACCEPTED));
1009
1010  // activityHandler.setActivityToUser(user,
1011  // activityHandler.addActivityWithMessageOnly(EventType.NOMINATION_ACCEPTED));
```

Package: com.cts.dcrowd.rest.app.utils

src/main/java/com/cts/dcrowd/rest/app/utils/Util.java, line 642 (Redundant Null Check) — Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Control Flow)

Sink Details

Sink: Dereferenced : result
Enclosing Method: connectExternalUser()

Redundant Null Check — Low

Package: com.cts.dcrowd.rest.app.utils

src/main/java/com/cts/dcrowd/rest/app/utils/Util.java, line 642 (Redundant Null Check) — Low

File: src/main/java/com/cts/dcrowd/rest/app/utils/Util.java:642
Taint Flags:

```
639  requestFactory.setHttpClient(httpClient);
640  restTemplate = new RestTemplate(requestFactory);
641  result = restTemplate.postForObject(url, email , Map.class, " ");
642  LOGGER.debug("Inside Email Sending Class...Result..."+result.toString());
643  if (result != null && result.containsKey("success")) {
644  output = (boolean) result.get("success");
645  }
```

Spring Boot Misconfiguration: Actuator Endpoint Security Disabled (1 issue)

Abstract

The Spring Boot application uses Actuator endpoints requiring no authentication.

Explanation

Spring Boot applications can be configured to deploy Actuators, which are REST endpoints that allow users to monitor different aspects of the application. There are different built-in Actuators which may expose sensitive data and are labeled as "sensitive". By default all sensitive HTTP endpoints are secured such that only users that have an `ACTUATOR` role may access them. This application is either disabling the authentication requirement for sensitive endpoints: **Example 1**:

```
management.security.enabled=false
```

Or marking sensitive endpoints as non-sensitive: **Example 2**:

```
endpoints.health.sensitive=false
```

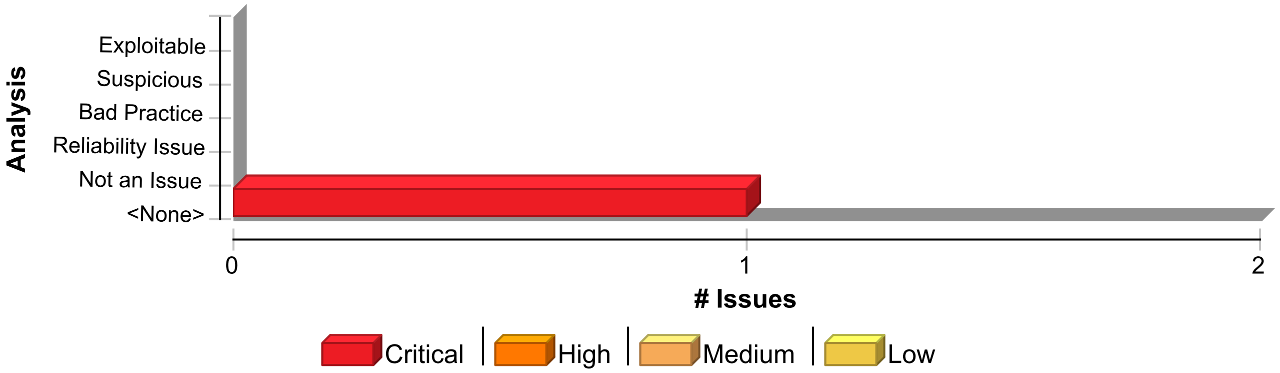
Or a custom Actuator is set as non-sensitive:

```
@Component
public class CustomEndpoint implements Endpoint<List<String>> {

    public String getId() {
        return "customEndpoint";
    }

    public boolean isEnabled() {
        return true;
    }

    public boolean isSensitive() {
        return false;
    }

    public List%lt;String> invoke() {
        // Custom logic to build the output
        ...
    }
}
```

Recommendation

All endpoints exposing sensitive information or operations should be protected with the correct levels of authentication and authorization. It is always a good practice to require authentication even for internal servers as a security in-depth mechanism. Take into account that even in the case that the application is deployed internally, behind a firewall, an attacker may still be able to reach it using a Server-Side Request Forgery vulnerability found in a different application.

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Spring Boot Misconfiguration: Actuator Endpoint Security Disabled	1	0	0	1
Total	**1**	**0**	**0**	**1**

Spring Boot Misconfiguration: Actuator Endpoint Security Disabled — Critical

Package: src.main.resources

src/main/resources/application.properties, line 4 (Spring Boot Misconfiguration: Actuator Endpoint Security Disabled) — Critical

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Configuration)

Sink Details

Sink: endpoints.refresh.sensitive
File: src/main/resources/application.properties:4
Taint Flags:

```
1 # Default profile
2 #spring.profiles.active=local
3 spring.profiles.include=message
4 endpoints.refresh.sensitive=false
5
6 undefined
7 undefined
```

Spring Boot Misconfiguration: DevTools Enabled (1 issue)

Abstract

The Spring Boot application is configured in developer mode.

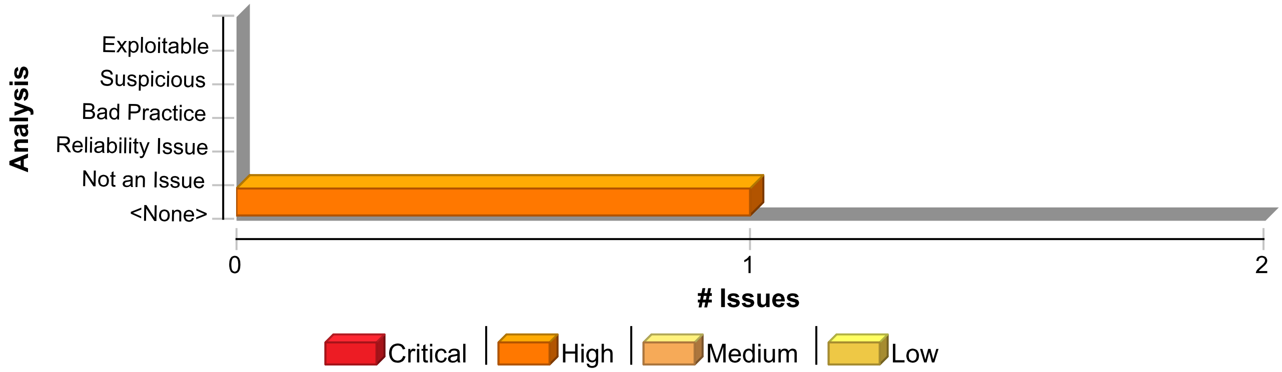
Explanation

The Spring Boot application has DevTools enabled. DevTools include an additional set of tools which can make the application development experience a little more pleasant, but DevTools are not recommended to use on applications in a production environment. As stated in the official Spring Boot documentation: "Enabling `spring-boot-devtools` on a remote application is a security risk. You should never enable support on a production deployment."

Recommendation

Remove `spring-boot-devtoos` dependency on production deployments.

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Spring Boot Misconfiguration: DevTools Enabled	1	0	0	1
Total	**1**	**0**	**0**	**1**

Spring Boot Misconfiguration: DevTools Enabled	High
Package: <none>	
pom.xml, line 38 (Spring Boot Misconfiguration: DevTools Enabled)	**High**

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Configuration)

Sink Details

File: pom.xml:38
Taint Flags:

```
35  </dependency>
36  <dependency>
```

Spring Boot Misconfiguration: DevTools Enabled High

Package: <none>

pom.xml, line 38 (Spring Boot Misconfiguration: DevTools Enabled) High

```
37  <groupId>org.springframework.boot</groupId>
38  <artifactId>spring-boot-devtools</artifactId>
39  </dependency>
40  <dependency>
41  <groupId>org.mongodb</groupId>
```

System Information Leak (75 issues)

Abstract

Revealing system data or debugging information helps an adversary learn about the system and form a plan of attack.

Explanation

An information leak occurs when system data or debugging information leaves the program through an output stream or logging function. **Example 1:** The following code prints an exception to the standard error stream:

```
try {
    ...
} catch (Exception e) {
    e.printStackTrace();
}
```

Depending upon the system configuration, this information can be dumped to a console, written to a log file, or exposed to a remote user. For example, with scripting mechanisms it is trivial to redirect output information from "Standard error" or "Standard output" into a file or another program. Alternatively the system that the program runs on could have a remote logging mechanism such as a "syslog" server that will send the logs to a remote device. During development you will have no way of knowing where this information may end up being displayed. In some cases the error message tells the attacker precisely what sort of an attack the system is vulnerable to. For example, a database error message can reveal that the application is vulnerable to a SQL injection attack. Other error messages can reveal more oblique clues about the system. In the example above, the leaked information could imply information about the type of operating system, the applications installed on the system, and the amount of care that the administrators have put into configuring the program. Here is another scenario, specific to the mobile world. Most mobile devices now implement a Near-Field Communication (NFC) protocol for quickly sharing information between devices using radio communication. It works by bringing devices to close proximity or simply having them touch each other. Even though the communication range of NFC is limited to just a few centimeters, eavesdropping, data modification and various other types of attacks are possible, since NFC alone does not ensure secure communication. **Example 2:** The Android platform provides support for NFC. The following code creates a message that gets pushed to the other device within the range.

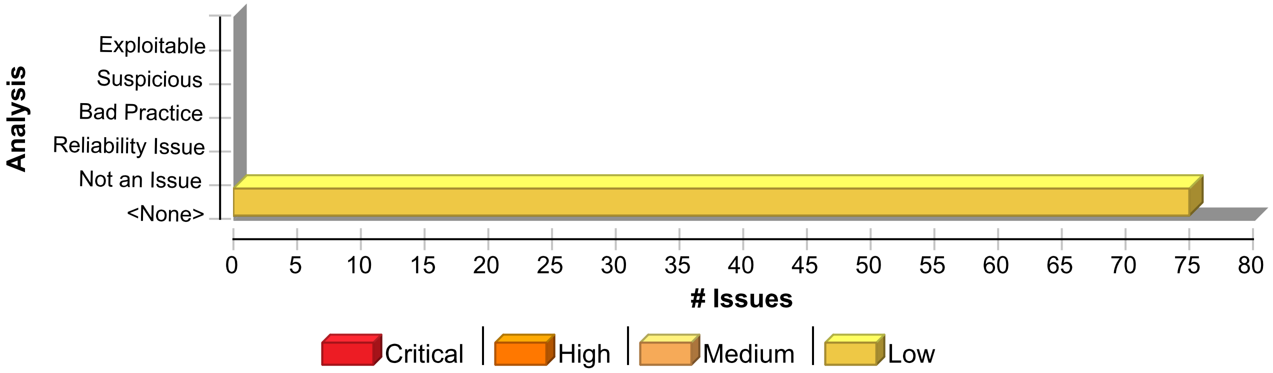
```
...
public static final String TAG = "NfcActivity";
private static final String DATA_SPLITTER = "__:DATA:__";
private static final String MIME_TYPE = "application/my.applications.mimetype";
...
public NdefMessage createNdefMessage(NfcEvent event) {
    TelephonyManager tm =
(TelephonyManager)Context.getSystemService(Context.TELEPHONY_SERVICE);
    String VERSION = tm.getDeviceSoftwareVersion();
    String text = TAG + DATA_SPLITTER + VERSION;
    NdefRecord record = new NdefRecord(NdefRecord.TNF_MIME_MEDIA,
            MIME_TYPE.getBytes(), new byte[0], text.getBytes());
    NdefRecord[] records = { record };
    NdefMessage msg = new NdefMessage(records);
    return msg;
}
...
```

NFC Data Exchange Format (NDEF) message contains typed data, a URI, or a custom application payload. If the message contains information about the application, such as its name, MIME type, or device software version, this information could be leaked to an eavesdropper. In the example above, Fortify Static Code Analyzer reports a System Information Leak vulnerability on the return statement.

Recommendation

Write error messages with security in mind. In production environments, turn off detailed error information in favor of brief messages. Restrict the generation and storage of detailed output that can help administrators and programmers diagnose problems. Be careful, debugging traces can sometimes appear in non-obvious places (embedded in comments in the HTML for an error page, for example). Even brief error messages that do not reveal stack traces or database dumps can potentially aid an attacker. For example, an "Access Denied" message can reveal that a file or user exists on the system. If you are concerned about leaking system data via NFC on an Android device, you could do one of the following three things. Either do not include system data in the messages pushed to other devices in range, or encrypt the payload of the message, or establish secure communication channel at a higher layer.

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
System Information Leak	75	0	0	75
Total	**75**	**0**	**0**	**75**

System Information Leak — Low

Package: .com.cts.dcrowd.rest.app.service

src/main/java/com/cts/dcrowd/rest/app/service/CampaignService.java, line 1657 (System Information Leak) — **Low**

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: accept()
File: src/main/java/com/cts/dcrowd/rest/app/service/CampaignService.java:1657
Taint Flags:

```
1654  userRepository.save(user);
1655  } catch (DcrowdException e1) {
1656  // TODO Auto-generated catch block
1657  e1.printStackTrace();
1658  }
```

System Information Leak — Low

Package: .com.cts.dcrowd.rest.app.service

src/main/java/com/cts/dcrowd/rest/app/service/CampaignService.java, line 1657 (System Information Leak) — Low

```
1659
1660  });
```

src/main/java/com/cts/dcrowd/rest/app/service/CampaignService.java, line 295 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: accept()
File: src/main/java/com/cts/dcrowd/rest/app/service/CampaignService.java:295
Taint Flags:

```
292  userRepository.save(user);
293  } catch (DcrowdException e1) {
294  // TODO Auto-generated catch block
295  e1.printStackTrace();
296  }
297  });
298  campaignId = null;
```

Package: .com.cts.dcrowd.rest.app.service.helper

src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java, line 477 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: accept()
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java:477
Taint Flags:

```
474  }
475  }
476  } catch (Exception e) {
477  e.printStackTrace();
478  }
479  }
480  }
```

System Information Leak

Low

Package: .com.cts.dcrowd.rest.app.service.helper

src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java, line 381 (System Information Leak)

Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: accept()
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java:381
Taint Flags:

```
378  userListAfterStateFilter.add(filterByState);
379  }
380  }catch(Exception e) {
381  e.printStackTrace();
382  }
383  }
384  }
```

src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java, line 230 (System Information Leak)

Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: accept()
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java:230
Taint Flags:

```
227  }
228  } catch (DcrowdException e) {
229  // TODO Auto-generated catch block
230  e.printStackTrace();
231  }
232  campaignResponseModel.setSystemDate(campaign.getCreatedDate());
233
```

src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java, line 187 (System Information Leak)

Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

System Information Leak — Low

Package: .com.cts.dcrowd.rest.app.service.helper

src/main/java/com/cts/dcrowd/rest/app/service/helper/ CampaignServiceHelper.java, line 187 (System Information Leak) — Low

Sink Details

Sink: printStackTrace()
Enclosing Method: accept()
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/CampaignServiceHelper.java:187
Taint Flags:

```
184  }
185  } catch (DcrowdException e) {
186  // TODO Auto-generated catch block
187  e.printStackTrace();
188  }
189  campaignResponseModel.setSystemDate(campaign.getCreatedDate());
190
```

Package: .com.cts.dcrowd.rest.defect.app.jira.services

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 246 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: accept()
File: src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java:246
Taint Flags:

```
243
244  } catch (Exception e) {
245  // TODO Auto-generated catch block
246  e.printStackTrace();
247  }
248
249  });
```

Package: .com.cts.dcrowd.rest.defect.app.services.impl

src/main/java/com/cts/dcrowd/rest/defect/app/services/impl/ DcrowdDefectEntryServiceImpl.java, line 221 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

System Information Leak — Low

Package: .com.cts.dcrowd.rest.defect.app.services.impl

src/main/java/com/cts/dcrowd/rest/defect/app/services/impl/DcrowdDefectEntryServiceImpl.java, line 221 (System Information Leak) — Low

Sink: printStackTrace()
Enclosing Method: accept()
File: src/main/java/com/cts/dcrowd/rest/defect/app/services/impl/DcrowdDefectEntryServiceImpl.java:221
Taint Flags:

```
218  }
219  } catch (IOException | DcrowdException e) {
220  // TODO Auto-generated catch block
221  e.printStackTrace();
222  saveException = true;
223  }
224  });
```

src/main/java/com/cts/dcrowd/rest/defect/app/services/impl/DcrowdDefectEntryServiceImpl.java, line 139 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: accept()
File: src/main/java/com/cts/dcrowd/rest/defect/app/services/impl/DcrowdDefectEntryServiceImpl.java:139
Taint Flags:

```
136  deAttachment.setAttachmentFilePath(filePathUrl+"/defect/"+campaignId+"/"+defectId+"/"+deAttachment.getAttachmentFileName());
137  } catch (IOException | DcrowdException e) {
138  // TODO Auto-generated catch block
139  e.printStackTrace();
140  saveException = true;
141  }
142  });
```

Package: com.cts.dcrowd.rest.app.controller

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 918 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: moderatorClosedCampaign()

System Information Leak — Low

Package: com.cts.dcrowd.rest.app.controller

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 918 (System Information Leak) — Low

File: src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java:918
Taint Flags:

```
915  Page<CampaignMetricsModel> campaignWithMetricsList =
campaignService.getClosedCampaignByModeratorId(offset-1,limit,isExternal);
916  return buildSuccessResponse(campaignWithMetricsList);
917  }catch(DcrowdException de) {
918  de.printStackTrace();
919  LOGGER.error("DcrowdException : ", de.getMessage());
920  return buildFailureResponse( 232, "Failed in CampaignController - Unable to fetch closed
campaign list");
921  }
```

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 968 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: getNominationHistoryOfUser()
File: src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java:968
Taint Flags:

```
965  return buildSuccessResponse(nominationList);
966  }
967  catch(DcrowdException de) {
968  de.printStackTrace();
969  LOGGER.error("DcrowdException : ", de.getMessage());
970  return buildFailureResponse(203, "Failed in CampaignController - Unable to
getNominationHistoryOfUser");
971  }
```

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 946 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: searchCampaigns()
File: src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java:946
Taint Flags:

System Information Leak — Low

Package: com.cts.dcrowd.rest.app.controller

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 946 (System Information Leak) — Low

```
943  }
944  return buildSuccessResponse(campaignList);
945  }catch(DcrowdException de) {
946  de.printStackTrace();
947  LOGGER.error("DcrowdException : ", de.getMessage());
948  return buildFailureResponse( 233, "Failed in CampaignController - Unable to search campaigns");
949  }
```

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 479 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: adminArchivedCampaign()
File: src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java:479
Taint Flags:

```
476  Pagination campaign = campaignService.getClosedCampaignByCreatorId(offset-1, limit);
477  return buildSuccessResponse(campaign);
478  }catch(DcrowdException de) {
479  de.printStackTrace();
480  LOGGER.error("DcrowdException : ", de.getMessage());
481  return buildFailureResponse(215, "Failed in CampaignController - Unable to fetch archived campaign list");
482  }
```

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 855 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: getDefectLayoutsByCampaignName()
File: src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java:855
Taint Flags:

```
852  return buildSuccessResponse(defectLayoutConfigService.getDefectLayoutsByCampaignName(campaignName));
853  }catch(DcrowdException de)
```

System Information Leak — Low

Package: com.cts.dcrowd.rest.app.controller

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 855 (System Information Leak) — Low

```
854  {
855  de.printStackTrace();
856  LOGGER.error("DcrowdException : ", de.getMessage());
857  return buildFailureResponse(229, "Failed in CampaignController - Unable to get Defect
Layouts By CampaignName");
858  }
```

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 548 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: sendRating()
File: src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java:548
Taint Flags:

```
545  campaignService.updateRatingForUser(campaignId);
546  return buildSuccessResponse("Send Rating : Successfully Saved");
547  }catch(DcrowdException | IOException de) {
548  de.printStackTrace();
549  LOGGER.error("DcrowdException : ", de.getMessage());
550  return buildFailureResponse( 218, "Failed in CampaignController - Unable to send rating");
551  }
```

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 320 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: viewCampaign()
File: src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java:320
Taint Flags:

```
317  }
318  catch(DcrowdException de)
319  {
320  de.printStackTrace();
321  LOGGER.error("DcrowdException : ", de.getMessage());
322  return buildFailureResponse(208, "Failed in CampaignController - error during
```

System Information Leak — Low

Package: com.cts.dcrowd.rest.app.controller

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 320 (System Information Leak) — Low

```
viewCampaign");
323   }
```

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 787 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: removeCampaign()
File: src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java:787
Taint Flags:

```
784   status = campaignService.removeCampaign(campaignId);
785   return buildSuccessResponse("Campaign removed from the system successfully");
786   } catch (DcrowdException de){
787   de.printStackTrace();
788   LOGGER.error("DcrowdException : ", de.getMessage());
789   if(status != 200) {
790   LOGGER.error("DcrowdException : Failed in CampaignController - Unable to remove
Campaign");
```

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 875 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: getAllDefectLayouts()
File: src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java:875
Taint Flags:

```
872   return buildSuccessResponse(defectLayoutConfigService.getDefectLayouts());
873   }catch(DcrowdException de)
874   {
875   de.printStackTrace();
876   LOGGER.error("DcrowdException : ", de.getMessage());
877   return buildFailureResponse(230, "Failed in CampaignController - Unable to get All Defect
Layouts");
878   }
```

System Information Leak — Low

Package: com.cts.dcrowd.rest.app.controller

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 897 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: moderatorCurrentCampaign()
File: src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java:897
Taint Flags:

```
894  Page<CampaignMetricsModel> campaignWithMetricsList = campaignService.getCurrentCampaignByModeratorId(offset-1,limit,isExternal);
895  return buildSuccessResponse(campaignWithMetricsList);
896  }catch(DcrowdException de) {
897  de.printStackTrace();
898  LOGGER.error("DcrowdException : ", de.getMessage());
899  return buildFailureResponse( 231, "Failed in CampaignController - Unable to fetch current campaign list");
900  }
```

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 433 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: completeCampaign()
File: src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java:433
Taint Flags:

```
430  campaignService.completeCampaign(campaignId);
431  return buildSuccessResponse("Successfully completed campaign");
432  } catch (DcrowdException | IOException de){
433  de.printStackTrace();
434  LOGGER.error("DcrowdException : ", de.getMessage());
435  return buildFailureResponse( 213, "Failed in CampaignController - Unable to complete campaign");
436  }
```

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 413 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation

System Information Leak

Low

Package: com.cts.dcrowd.rest.app.controller

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 413 (System Information Leak)

Low

Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: startCampaign()
File: src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java:413
Taint Flags:

```
410  campaignService.startCampaign(campaignId);
411  return buildSuccessResponse("Campaign started successfully");
412  } catch(DcrowdException | IOException de) {
413  de.printStackTrace();
414  LOGGER.error("DcrowdException : ", de.getMessage());
415  return buildFailureResponse(212, "Failed in CampaignController - Unable to start
campaign");
416  }
```

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 238 (System Information Leak)

Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: saveCampaign()
File: src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java:238
Taint Flags:

```
235  }
236  catch(DcrowdException | IOException de)
237  {
238  de.printStackTrace();
239  LOGGER.error("DcrowdException : ", de);
240  return buildFailureResponse(206, "Failed in CampaignController - error during campaign
creation");
241  }
```

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 458 (System Information Leak)

Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

System Information Leak — Low

Package: com.cts.dcrowd.rest.app.controller

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 458 (System Information Leak) — Low

Sink: printStackTrace()
Enclosing Method: adminCurrentCampaign()
File: src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java:458
Taint Flags:

```
455  Pagination campaign = campaignService.getCurrentCampaignByCreatorId(offset-1,limit);
456  return buildSuccessResponse(campaign);
457  }catch(DcrowdException de) {
458  de.printStackTrace();
459  LOGGER.error("DcrowdException : ", de.getMessage());
460  return buildFailureResponse( 214, "Failed in CampaignController - Unable to fetch current campaign list");
461  }
```

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 813 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: defectConfig()
File: src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java:813
Taint Flags:

```
810  defectLayoutConfigService.configure(campaignId, crowdRequest.getParam());
811  return buildSuccessResponse("Defect Layout saved successfully");
812  } catch (DcrowdException de){
813  de.printStackTrace();
814  LOGGER.error("DcrowdException : ", de.getMessage());
815  return buildFailureResponse( 227, "Failed in CampaignController - Unable to save Defect Layout");
816  }
```

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 345 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: nominateCampaign()
File: src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java:345

System Information Leak — Low

Package: com.cts.dcrowd.rest.app.controller

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 345 (System Information Leak) — Low

Taint Flags:

```
342  else
343  return buildFailureResponse(209, "Campaign is not in open for nomination status");
344  } catch (DcrowdException de) {
345  de.printStackTrace();
346  LOGGER.error("DcrowdException:", de.getMessage());
347  return buildFailureResponse(209, "Failed in CampaignController - Unable to nominate in the campaign");
348  }
```

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 527 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: getFeedback()
File: src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java:527
Taint Flags:

```
524  return buildSuccessResponse(feedback);
525  }catch(DcrowdException de)
526  {
527  de.printStackTrace();
528  LOGGER.error("DcrowdException : ", de.getMessage());
529  return buildFailureResponse(217, "Failed in CampaignController - Unable to get user rating");
530  }
```

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 599 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: getCampaignerImageById()
File: src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java:599
Taint Flags:

```
596  return buildSuccessResponse(image);
```

System Information Leak — Low

Package: com.cts.dcrowd.rest.app.controller

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 599 (System Information Leak) — Low

```
597  }
598  catch (DcrowdException de) {
599  de.printStackTrace();
600  LOGGER.error("DcrowdException : ",de.getMessage());
601  return buildFailureResponse(220, "Failed in CampaignController - Unable to
getCampaignerImageById");
602  }
```

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 188 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: getNominatedCampaignOfUser()
File: src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java:188
Taint Flags:

```
185  return buildSuccessResponse(campaignList);
186  }
187  catch(DcrowdException de) {
188  de.printStackTrace();
189  LOGGER.error("DcrowdException : ", de.getMessage());
190  return buildFailureResponse(203, "Failed in CampaignController - Unable to
getNominatedCampaignOfUser");
191  }
```

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 574 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: saveCampaignerImage()
File: src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java:574
Taint Flags:

```
571  }
572
573  } catch (DcrowdException | IOException de) {
574  de.printStackTrace();
```

System Information Leak — Low

Package: com.cts.dcrowd.rest.app.controller

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 574 (System Information Leak) — Low

```
575   LOGGER.error("DcrowdException : ",de.getMessage());
576   return buildFailureResponse(219, "Failed in CampaignController - "+de);
577   }
```

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 392 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: closeNomination()
File: src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java:392
Taint Flags:

```
389   campaignService.closeNomination(campaignId);
390   return buildSuccessResponse("Nomination closed");
391   } catch(DcrowdException | IOException de) {
392   de.printStackTrace();
393   LOGGER.error("DcrowdException : ", de.getMessage());
394   return buildFailureResponse( 211, "Failed in CampaignController - Unable to close
nomination");
395   }
```

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 212 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: getCompletedCampaignOfUser()
File: src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java:212
Taint Flags:

```
209   return buildSuccessResponse(campaignList);
210   }
211   catch(DcrowdException de) {
212   de.printStackTrace();
213   LOGGER.error("DcrowdException : ", de.getMessage());
214   return buildFailureResponse(204, "Failed in CampaignController - Unable to
getCompletedCampaignOfUser");
215   }
```

System Information Leak — Low

Package: com.cts.dcrowd.rest.app.controller

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 212 (System Information Leak) — Low

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 287 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: updateForNomination()
File: src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java:287
Taint Flags:

```
284  updateCampaignUsersList = campaignService.updateCampaignNominationAndConfirmedUsersList(campaign,emailStatus);
285  } catch (IOException | DcrowdException e) {
286  // TODO Auto-generated catch block
287  e.printStackTrace();
288  }
289  if(updateCampaignUsersList){
290  return buildSuccessResponse("Success - update Nomination mail sent Success");
```

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 372 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: confirmTeam()
File: src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java:372
Taint Flags:

```
369  campaignService.confirmTeam(campaignId, userList,confirmStatus);
370  return buildSuccessResponse("Team confirmed");
371  } catch(DcrowdException de) {
372  de.printStackTrace();
373  LOGGER.error("DcrowdException: ", de.getMessage());
374  return buildFailureResponse(210, "Failed in CampaignController - Unable to confirm team");
375  }
```

System Information Leak | Low

Package: com.cts.dcrowd.rest.app.controller

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 262 (System Information Leak) | Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: openForNomination()
File: src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java:262
Taint Flags:

```
259  }
260  catch(DcrowdException | IOException e)
261  {
262  e.printStackTrace();
263  LOGGER.error( "DcrowdException : ", e);
264  return buildFailureResponse(207, "Failed in CampaignController - errors in
openForNomination");
265  }
```

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 669 (System Information Leak) | Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: emailImageUpload()
File: src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java:669
Taint Flags:

```
666  throw new DcrowdException("Invalid Attachment");
667  }
668  } catch (DcrowdException de) {
669  de.printStackTrace();
670  LOGGER.error("DcrowdException : ",de.getMessage());
671  return buildFailureResponse(223, "Failed in CampaignController - "+de);
672  }
```

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 834 (System Information Leak) | Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

System Information Leak — Low

Package: com.cts.dcrowd.rest.app.controller

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 834 (System Information Leak) — Low

Sink Details

Sink: printStackTrace()
Enclosing Method: getDefectLayout()
File: src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java:834
Taint Flags:

```
831  return buildSuccessResponse(defectLayoutConfigService.getDefectLayoutByCampaignId(campaignId));
832  }catch(DcrowdException de)
833  {
834  de.printStackTrace();
835  LOGGER.error("DcrowdException : ", de.getMessage());
836  return buildFailureResponse(228, "Failed in CampaignController - Unable to get Defect Layout");
837  }
```

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 133 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: getCampaigns()
File: src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java:133
Taint Flags:

```
130  usersCampaignList.setCurrentCampaignListWithMetrics(homeCampaignPage);
131  return buildSuccessResponse(usersCampaignList);
132  } catch(DcrowdException de) {
133  de.printStackTrace();
134  LOGGER.error("DcrowdException : ", de.getMessage());
135  return buildFailureResponse(201, "Failed in CampaignController - Unable to retrieve all campaigns");
136  }
```

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 164 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()

System Information Leak — Low

Package: com.cts.dcrowd.rest.app.controller

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 164 (System Information Leak) — Low

Enclosing Method: getCurrentCampaignOfUser()
File: src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java:164
Taint Flags:

```
161  return buildSuccessResponse(usersCampaignList);
162  }
163  catch(DcrowdException de) {
164  de.printStackTrace();
165  LOGGER.error("DcrowdException : ", de.getMessage());
166  return buildFailureResponse(202, "Failed in CampaignController - Unable to
getCurrentCampaignOfUser");
167  }
```

src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java, line 505 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: saveFeedback()
File: src/main/java/com/cts/dcrowd/rest/app/controller/CampaignController.java:505
Taint Flags:

```
502  return buildSuccessResponse("Feedback : Successfully Saved");
503  }catch(DcrowdException de)
504  {
505  de.printStackTrace();
506  LOGGER.error("DcrowdException : ", de.getMessage());
507  return buildFailureResponse(216, "Failed in CampaignController - Unable to send user
rating");
508  }
```

Package: com.cts.dcrowd.rest.app.repository

src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java, line 92 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: createUser()

System Information Leak — Low

Package: com.cts.dcrowd.rest.app.repository

src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java, line 92 (System Information Leak) — Low

File: src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java:92
Taint Flags:

```
89  userService.createUpdateUser(user);
90  } catch (DcrowdException e) {
91  // TODO Auto-generated catch block
92  e.printStackTrace();
93  }
94  }
95
```

src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java, line 79 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: createUser()
File: src/test/java/com/cts/dcrowd/rest/app/repository/UserControllerTestNG.java:79
Taint Flags:

```
76  user.setCreatedDate(Util.getDateWithTimeAsDate());
77  } catch (DcrowdException e) {
78  // TODO Auto-generated catch block
79  e.printStackTrace();
80  }
81  //user.setSystemDate(new Date());
82
```

Package: com.cts.dcrowd.rest.app.security

src/main/java/com/cts/dcrowd/rest/app/security/LdapUserDetailsMapperWS.java, line 309 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: insertNewUserWebService()
File: src/main/java/com/cts/dcrowd/rest/app/security/LdapUserDetailsMapperWS.java:309
Taint Flags:

System Information Leak Low

Package: com.cts.dcrowd.rest.app.security

src/main/java/com/cts/dcrowd/rest/app/security/LdapUserDetailsMapperWS.java, line 309 (System Information Leak) Low

```
306  return user;
307
308  } catch (Exception e) {
309  e.printStackTrace();
310  return null;
311  }
312
```

src/main/java/com/cts/dcrowd/rest/app/security/ApplicationAuthProvider.java, line 107 (System Information Leak) Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: authenticate()
File: src/main/java/com/cts/dcrowd/rest/app/security/ApplicationAuthProvider.java:107
Taint Flags:

```
104  authentication.getCredentials(), grantedAuths);
105  }
106  catch (Exception e) {
107  e.printStackTrace();
108  return null;
109  }
110  }
```

src/main/java/com/cts/dcrowd/rest/app/security/LdapUserDetailsMapperWS.java, line 103 (System Information Leak) Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: loadUserByUsername()
File: src/main/java/com/cts/dcrowd/rest/app/security/LdapUserDetailsMapperWS.java:103
Taint Flags:

```
100  user = insertNewUserWebService(username);
101  }
102  } catch (Exception e) {
103  e.printStackTrace();
```

System Information Leak Low

Package: com.cts.dcrowd.rest.app.security

src/main/java/com/cts/dcrowd/rest/app/security/LdapUserDetailsMapperWS.java, line 103 (System Information Leak) Low

```
104  return null;
105  }
106  return user;
```

Package: com.cts.dcrowd.rest.app.security.service.impl

src/main/java/com/cts/dcrowd/rest/app/security/service/impl/SecurityServiceImpl.java, line 104 (System Information Leak) Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: doAuthenticate()
File: src/main/java/com/cts/dcrowd/rest/app/security/service/impl/SecurityServiceImpl.java:104
Taint Flags:

```
101
102  SecurityContextHolder.getContext().setAuthentication(authentication);
103  } catch(Exception e) {
104  e.printStackTrace();
105  return false;
106  }
107  return true;
```

Package: com.cts.dcrowd.rest.app.service

src/main/java/com/cts/dcrowd/rest/app/service/UserService.java, line 457 (System Information Leak) Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: addUserBadge()
File: src/main/java/com/cts/dcrowd/rest/app/service/UserService.java:457
Taint Flags:

```
454  }
455  catch (DcrowdException e) {
456  // TODO Auto-generated catch block
457  e.printStackTrace();
458  }
```

System Information Leak — Low

Package: com.cts.dcrowd.rest.app.service

src/main/java/com/cts/dcrowd/rest/app/service/UserService.java, line 457 (System Information Leak) — Low

```
459  }
460
```

src/main/java/com/cts/dcrowd/rest/app/service/ProfileImageService.java, line 109 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: saveImage()
File: src/main/java/com/cts/dcrowd/rest/app/service/ProfileImageService.java:109
Taint Flags:

```
106  Util.writeFileToFileServer(imageFile.getBytes(), fileDIR, finalFilePath, false);
107  userService.createUpdateUser(user);
108  } catch (Exception e) {
109  e.printStackTrace();
110
111  }
112
```

src/main/java/com/cts/dcrowd/rest/app/service/UserService.java, line 236 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: setUserProfile()
File: src/main/java/com/cts/dcrowd/rest/app/service/UserService.java:236
Taint Flags:

```
233  }
234  }catch(Exception e)
235  {
236  e.printStackTrace();
237  }
238  return user;
239  }
```

System Information Leak — Low

Package: com.cts.dcrowd.rest.app.service

src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java, line 203 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: moderatorsInvitationForCampaigns()
File: src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java:203
Taint Flags:

```
200  }
201
202  } catch (Exception e) {
203  e.printStackTrace();
204  return false;
205  }
206  }
```

src/main/java/com/cts/dcrowd/rest/app/service/CampaignService.java, line 1264 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: updateRatingForUser()
File: src/main/java/com/cts/dcrowd/rest/app/service/CampaignService.java:1264
Taint Flags:

```
1261
1262  }
1263  } catch (Exception e) {
1264  e.printStackTrace();
1265  throw new DcrowdException("Fails during updateRatingForUser");
1266  }
1267  saveCampaignWithStatus(campaign, CampaignStatus.CAMPAIGN_CLOSED);
```

src/main/java/com/cts/dcrowd/rest/app/service/CampaignService.java, line 1039 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

System Information Leak — Low

Package: com.cts.dcrowd.rest.app.service

src/main/java/com/cts/dcrowd/rest/app/service/CampaignService.java, line 1039 (System Information Leak) — Low

Sink Details

Sink: printStackTrace()
Enclosing Method: confirmTeam()
File: src/main/java/com/cts/dcrowd/rest/app/service/CampaignService.java:1039
Taint Flags:

```
1036  }
1037
1038  catch (Exception e) {
1039  e.printStackTrace();
1040  throw new DcrowdException("Fails during Confirm Team Service" + e.getMessage());
1041  }
1042
```

src/main/java/com/cts/dcrowd/rest/app/service/CampaignService.java, line 1992 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: updateCampaignNominationAndConfirmedUsersList()
File: src/main/java/com/cts/dcrowd/rest/app/service/CampaignService.java:1992
Taint Flags:

```
1989  return false;
1990  }
1991  } catch (DcrowdException e) {
1992  e.printStackTrace();
1993  return false;
1994  }
1995
```

src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java, line 345 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: fastesterRegistration()
File: src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java:345

System Information Leak — Low

Package: com.cts.dcrowd.rest.app.service

src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java, line 345 (System Information Leak) — Low

Taint Flags:

```
342 EmailVO emailConf = emailService.createEmail(configuration,false);
343 emailService.sendEmail(emailConf);
344 } catch (Exception e) {
345 e.printStackTrace();
346 }
347 }
348
```

src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java, line 376 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: externalPasswordReset()
File: src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java:376
Taint Flags:

```
373 EmailVO emailConf = emailService.createEmail(configuration,false);
374 emailService.sendEmail(emailConf);
375 } catch (Exception e) {
376 e.printStackTrace();
377 throw new DcrowdException("Failed during sending mail to external");
378 }
379 }
```

src/main/java/com/cts/dcrowd/rest/app/service/EmailService.java, line 137 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: createEmail()
File: src/main/java/com/cts/dcrowd/rest/app/service/EmailService.java:137
Taint Flags:

```
134 return email;
135
136 } catch (Exception e) {
```

System Information Leak — Low

Package: com.cts.dcrowd.rest.app.service

src/main/java/com/cts/dcrowd/rest/app/service/EmailService.java, line 137 (System Information Leak) — Low

```
137  e.printStackTrace();
138
139  }
140  return null;
```

src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java, line 406 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: externalCustomerRegistered()
File: src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java:406
Taint Flags:

```
403  EmailVO emailConf = emailService.createEmail(configuration,false);
404  emailService.sendEmail(emailConf);
405  } catch (Exception e) {
406  e.printStackTrace();
407  }
408  }
409
```

src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java, line 428 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: userAssignedMail()
File: src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java:428
Taint Flags:

```
425  EmailVO emailConf = emailService.createEmail(configuration,false);
426  emailService.sendEmail(emailConf);
427  } catch (Exception e) {
428  e.printStackTrace();
429  }
430  }
431
```

System Information Leak — Low

Package: com.cts.dcrowd.rest.app.service

src/main/java/com/cts/dcrowd/rest/app/service/UserService.java, line 496 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: removeUserBadge()
File: src/main/java/com/cts/dcrowd/rest/app/service/UserService.java:496
Taint Flags:

```
493  }
494  } catch (DcrowdException e) {
495  // TODO Auto-generated catch block
496  e.printStackTrace();
497  }
498
499  }
```

src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java, line 268 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: defectStatusChangeEmail()
File: src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java:268
Taint Flags:

```
265  EmailVO emailConf = emailService.createEmail(configuration,false);
266  emailService.sendEmail(emailConf);
267  } catch (Exception e) {
268  e.printStackTrace();
269  }
270  }
271
```

src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java, line 136 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

System Information Leak — Low

Package: com.cts.dcrowd.rest.app.service

src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java, line 136 (System Information Leak) — Low

Sink Details

Sink: printStackTrace()
Enclosing Method: afterInvitedTester()
File: src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java:136
Taint Flags:

```
133  emailService.sendEmail(emailConf);
134  return true;
135  } catch (Exception e) {
136  e.printStackTrace();
137  return false;
138  }
139  }
```

src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java, line 317 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: campaignEmail()
File: src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java:317
Taint Flags:

```
314  }
315  return true;
316  } catch (Exception e) {
317  e.printStackTrace();
318  return false;
319  }
320  }
```

src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java, line 169 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: inviteExternalModerators()
File: src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java:169

System Information Leak — Low

Package: com.cts.dcrowd.rest.app.service

src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java, line 169 (System Information Leak) — Low

Taint Flags:

```
166  emailService.sendEmail(emailConf);
167  return true;
168  } catch (Exception e) {
169  e.printStackTrace();
170  return false;
171  }
172  }
```

src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java, line 238 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: afterRegistration()
File: src/main/java/com/cts/dcrowd/rest/app/service/EmailEventService.java:238
Taint Flags:

```
235  EmailVO emailConf = emailService.createEmail(configuration,false);
236  emailService.sendEmail(emailConf);
237  } catch (Exception e) {
238  e.printStackTrace();
239  }
240  }
241
```

Package: com.cts.dcrowd.rest.app.service.helper

src/main/java/com/cts/dcrowd/rest/app/service/helper/ApplicationQueryProcessor.java, line 423 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: getUserBadgeToGenerateReport()
File: src/main/java/com/cts/dcrowd/rest/app/service/helper/ApplicationQueryProcessor.java:423
Taint Flags:

```
420  }
```

System Information Leak — Low

Package: com.cts.dcrowd.rest.app.service.helper

src/main/java/com/cts/dcrowd/rest/app/service/helper/ApplicationQueryProcessor.java, line 423 (System Information Leak) — Low

```
421  catch(Exception e)
422  {
423  e.printStackTrace();
424  }
425  return null;
426  }
```

Package: com.cts.dcrowd.rest.app.util

src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java, line 104 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: generateBlowFishDecryptValue()
File: src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java:104
Taint Flags:

```
101  strData = new String(decrypted);
102  System.out.println(strData);
103  } catch (Exception e) {
104  e.printStackTrace();
105  }
106  return strData;
107  }
```

src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java, line 62 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: base64Decoder()
File: src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java:62
Taint Flags:

```
59  try {
60  sourceValue = new String(Base64.getDecoder().decode(encodedValue), "utf-8");
61  } catch (Exception e) {
62  e.printStackTrace();
```

System Information Leak

Low

Package: com.cts.dcrowd.rest.app.util

src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java, line 62 (System Information Leak)

Low

```
63  }
64  return sourceValue;
65  }
```

src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java, line 82 (System Information Leak)

Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: generateBlowFishEncryptValue()
File: src/test/java/com/cts/dcrowd/rest/app/util/TestUtil.java:82
Taint Flags:

```
79  byte[] encrypted=cipher.doFinal(sourceValue.getBytes());
80  strData = new String(Base64.getEncoder().encode(encrypted));
81  } catch (Exception e) {
82  e.printStackTrace();
83  }
84  return strData;
85  }
```

Package: com.cts.dcrowd.rest.app.utils

src/main/java/com/cts/dcrowd/rest/app/utils/EmailDomainFilterUtility.java, line 35 (System Information Leak)

Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: EmailDomainFilterUtility()
File: src/main/java/com/cts/dcrowd/rest/app/utils/EmailDomainFilterUtility.java:35
Taint Flags:

```
32  try {
33  loadDomainList();
34  } catch (Exception e) {
35  e.printStackTrace();
36  }
37  }
38
```

System Information Leak — Low

Package: com.cts.dcrowd.rest.app.utils

src/main/java/com/cts/dcrowd/rest/app/utils/EmailDomainFilterUtility.java, line 35 (System Information Leak) — Low

src/main/java/com/cts/dcrowd/rest/app/utils/Util.java, line 509 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: generateBlowFishEncryptValue()
File: src/main/java/com/cts/dcrowd/rest/app/utils/Util.java:509
Taint Flags:

```
506  //strData = base64Encoder(encrypted);
507  strData = encrypted.toString();
508  } catch (Exception e) {
509  e.printStackTrace();
510  }
511  return strData;
512  }
```

src/main/java/com/cts/dcrowd/rest/app/utils/StopwordUtility.java, line 37 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: StopwordUtility()
File: src/main/java/com/cts/dcrowd/rest/app/utils/StopwordUtility.java:37
Taint Flags:

```
34  try {
35  loadStopList();
36  } catch (Exception e) {
37  e.printStackTrace();
38  }
39  }
40
```

src/main/java/com/cts/dcrowd/rest/app/utils/ZipUtil.java, line 38 (System Information Leak) — Low

Issue Details

System Information Leak — Low

Package: com.cts.dcrowd.rest.app.utils

src/main/java/com/cts/dcrowd/rest/app/utils/ZipUtil.java, line 38 (System Information Leak) — Low

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: gzipCompress()
File: src/main/java/com/cts/dcrowd/rest/app/utils/ZipUtil.java:38
Taint Flags:

```
35  gzipOS.close();
36  result = bos.toByteArray();
37  } catch (IOException e) {
38  e.printStackTrace();
39  throw new DcrowdException("Exception in Util Library : Compression");
40  }
41  return result;
```

src/main/java/com/cts/dcrowd/rest/app/utils/Util.java, line 532 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: generateBlowFishDecryptValue()
File: src/main/java/com/cts/dcrowd/rest/app/utils/Util.java:532
Taint Flags:

```
529  try {
530  basicAuthString = new String(Base64.getDecoder().decode(encryptedValue),
StandardCharsets.UTF_8).split("fastestCustomers");
531  } catch (Exception e) {
532  e.printStackTrace();
533  }
534  return basicAuthString!=null && basicAuthString.length != 0 ? basicAuthString[0] : null;
535  }
```

src/main/java/com/cts/dcrowd/rest/app/utils/ZipUtil.java, line 64 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

System Information Leak — Low

Package: com.cts.dcrowd.rest.app.utils

src/main/java/com/cts/dcrowd/rest/app/utils/ZipUtil.java, line 64 (System Information Leak) — Low

Sink: printStackTrace()
Enclosing Method: gzipUncompress()
File: src/main/java/com/cts/dcrowd/rest/app/utils/ZipUtil.java:64
Taint Flags:

```
61  }
62  result = bos.toByteArray();
63  } catch (IOException e) {
64  e.printStackTrace();
65  throw new DcrowdException("Exception in Util Library : DeCompression");
66  }
67  return result;
```

Package: com.cts.dcrowd.rest.defect.app.jira.services

src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java, line 277 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: doAddCommentsPost()
File: src/main/java/com/cts/dcrowd/rest/defect/app/jira/services/JiraService.java:277
Taint Flags:

```
274
275  } catch (JSONException | IOException e) {
276  // TODO Auto-generated catch block
277  e.printStackTrace();
278  }
279  }
280
```

Package: com.cts.dcrowd.rest.defect.app.services.impl

src/main/java/com/cts/dcrowd/rest/defect/app/services/impl/DcrowdDefectEntryServiceImpl.java, line 694 (System Information Leak) — Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()

System Information Leak	Low
Package: com.cts.dcrowd.rest.defect.app.services.impl	
src/main/java/com/cts/dcrowd/rest/defect/app/services/impl/ DcrowdDefectEntryServiceImpl.java, line 694 (System Information Leak)	Low

Enclosing Method: cloneDefectEntriesForExport()
File: src/main/java/com/cts/dcrowd/rest/defect/app/services/impl/DcrowdDefectEntryServiceImpl.java:694
Taint Flags:

```
691  user = userService.findByUsername(defectEntry.getDeUserId());
692  } catch (DcrowdException e) {
693  // TODO Auto-generated catch block
694  e.printStackTrace();
695  }
696  cloneDefectMap.put("LOGGEDEMAILID", user.getEmail());
697  cloneDefectMap.put("USERLOCATION", defectEntry.getUserLocation());
```

System Information Leak: Spring Boot Actuators Enabled (1 issue)

Abstract

Enabling Spring Boot Actuators may reveal system data and debugging information which may help an adversary learn about the system and form a plan of attack.

Explanation

Spring Boot Actuators allow users to monitor and interact with the application. There are different built-in Actuators which expose system data and debugging information through HTTP endpoints, JMX or even by remote shell (SSH or Telnet). Attackers may benefit from this information to learn about the system and gather information that could be used to attack the application.

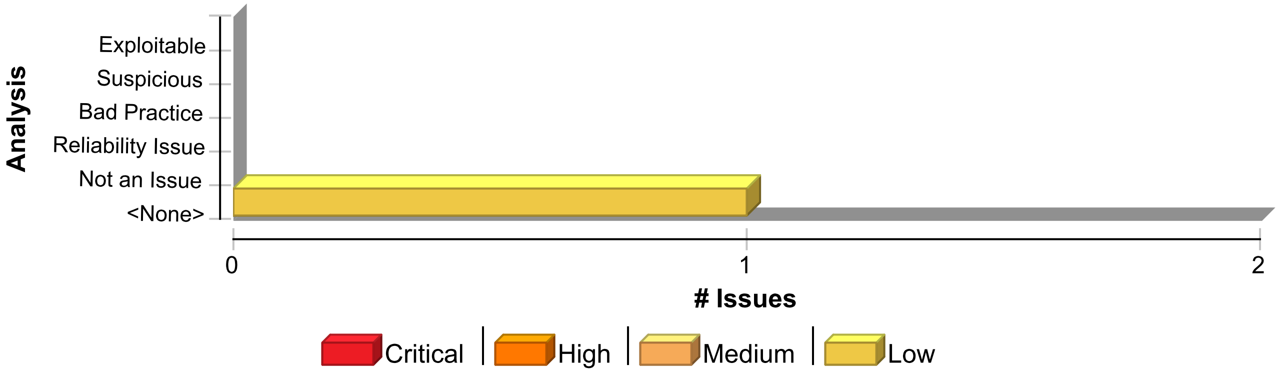
Recommendation

Configure Actuators with security in mind. If the Actuator reveals sensitive data, it should be flagged as such so that unauthenticated users cannot access it. **Example**: The following property can be used to make all Actuator endpoints sensitive:

```
endpoints.sensitive=true
```

In production environments, turn off detailed error information in favor of brief messages. Restrict the generation and storage of detailed output that can help administrators and programmers diagnose problems.

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
System Information Leak: Spring Boot Actuators Enabled	1	0	0	1
Total	**1**	**0**	**0**	**1**

System Information Leak: Spring Boot Actuators Enabled **Low**

Package: <none>

pom.xml, line 26 (System Information Leak: Spring Boot Actuators Enabled) **Low**

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Configuration)

System Information Leak: Spring Boot Actuators Enabled — Low

Package: <none>

pom.xml, line 26 (System Information Leak: Spring Boot Actuators Enabled) — Low

Sink Details

File: pom.xml:26
Taint Flags:

```
23  <dependencies>
24  <dependency>
25  <groupId>org.springframework.boot</groupId>
26  <artifactId>spring-boot-starter-actuator</artifactId>
27  </dependency>
28  <dependency>
29  <groupId>org.springframework.boot</groupId>
```

Unchecked Return Value (3 issues)

Abstract

Ignoring a method's return value can cause the program to overlook unexpected states and conditions.

Explanation

It is not uncommon for Java programmers to misunderstand `read()` and related methods that are part of many `java.io` classes. Most errors and unusual events in Java result in an exception being thrown. (This is one of the advantages that Java has over languages like C: Exceptions make it easier for programmers to think about what can go wrong.) But the stream and reader classes do not consider it unusual or exceptional if only a small amount of data becomes available. These classes simply add the small amount of data to the return buffer, and set the return value to the number of bytes or characters read. There is no guarantee that the amount of data returned is equal to the amount of data requested. This behavior makes it important for programmers to examine the return value from `read()` and other IO methods to ensure that they receive the amount of data they expect. **Example:** The following code loops through a set of users, reading a private data file for each user. The programmer assumes that the files are always exactly 1 kilobyte in size and therefore ignores the return value from `read()`. If an attacker can create a smaller file, the program will recycle the remainder of the data from the previous user and handle it as though it belongs to the attacker.

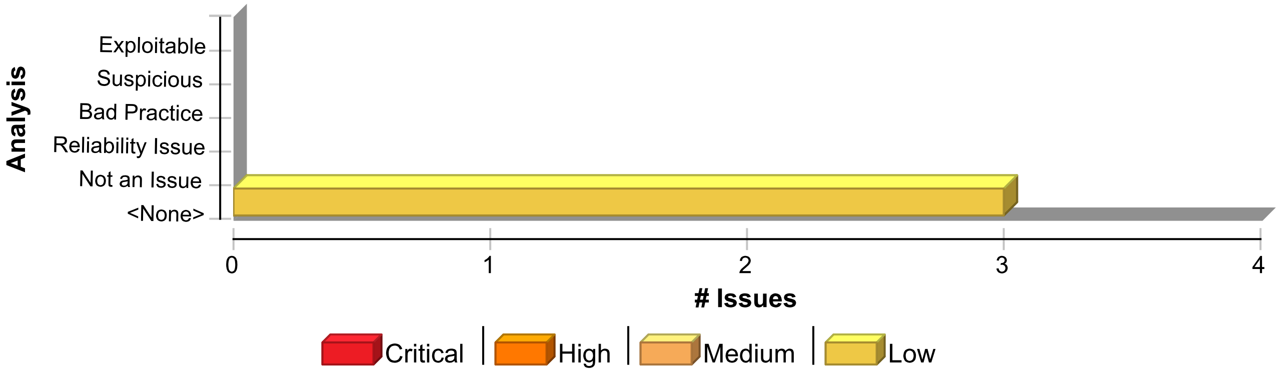
```
FileInputStream fis;
byte[] byteArray = new byte[1024];
for (Iterator i=users.iterator(); i.hasNext();) {
    String userName = (String) i.next();
    String pFileName = PFILE_ROOT + "/" + userName;
    FileInputStream fis = new FileInputStream(pFileName);
    fis.read(byteArray); // the file is always 1k bytes
    fis.close();
    processPFile(userName, byteArray);
}
```

Recommendation

```
  FileInputStream fis;
  byte[] byteArray = new byte[1024];
  for (Iterator i=users.iterator(); i.hasNext();) {
    String userName = (String) i.next();
    String pFileName = PFILE_ROOT + "/" + userName;
    fis = new FileInputStream(pFileName);
    int bRead = 0;
    while (bRead < 1024) {
        int rd = fis.read(byteArray, bRead, 1024 - bRead);
        if (rd == -1) {
          throw new IOException("file is unusually small");
        }
        bRead += rd;
    }
    // could add check to see if file is too large here
    fis.close();
    processPFile(userName, byteArray);
  }
```

Note: Because the fix for this problem is relatively complicated, you might be tempted to use a simpler approach, such as checking the size of the file before you begin reading. Such an approach would render the application vulnerable to a file system race condition, whereby an attacker could replace a well-formed file with a malicious file between the file size check and the call to read data from the file.

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Unchecked Return Value	3	0	0	3
Total	**3**	**0**	**0**	**3**

Unchecked Return Value — Low

Package: com.cts.dcrowd.rest.app.utils

src/main/java/com/cts/dcrowd/rest/app/utils/Util.java, line 762 (Unchecked Return Value) — Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: read()
Enclosing Method: encodeFileToBase64Binary()
File: src/main/java/com/cts/dcrowd/rest/app/utils/Util.java:762
Taint Flags:

```
759  try {
760  FileInputStream fileInputStreamReader = new FileInputStream(file);
761  byte[] bytes = new byte[(int)file.length()];
762  fileInputStreamReader.read(bytes);
763  encodedfile = new String(org.apache.commons.codec.binary.Base64.encodeBase64(bytes),
"UTF-8");
764  fileInputStreamReader.close();
765  } catch (FileNotFoundException e) {
```

src/main/java/com/cts/dcrowd/rest/app/utils/Util.java, line 672 (Unchecked Return Value) — Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Unchecked Return Value

Low

Package: com.cts.dcrowd.rest.app.utils

src/main/java/com/cts/dcrowd/rest/app/utils/Util.java, line 672 (Unchecked Return Value)

Low

Sink Details

Sink: delete()
Enclosing Method: writeFileToFileServer()
File: src/main/java/com/cts/dcrowd/rest/app/utils/Util.java:672
Taint Flags:

```
669  file = new File(finalFilePath);
670
671  if (file.exists()) {
672  file.delete();
673  }
674  } else {
675  if (!new File(fileDIR).exists()) {
```

src/main/java/com/cts/dcrowd/rest/app/utils/Util.java, line 676 (Unchecked Return Value)

Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: mkdirs()
Enclosing Method: writeFileToFileServer()
File: src/main/java/com/cts/dcrowd/rest/app/utils/Util.java:676
Taint Flags:

```
673  }
674  } else {
675  if (!new File(fileDIR).exists()) {
676  new File(fileDIR).mkdirs();
677  }
678  file = new File(finalFilePath);
679  try (ByteArrayInputStream inputStream = new ByteArrayInputStream(imageData)) {
```

Unreleased Resource: Streams (1 issue)

Abstract

The program can potentially fail to release a system resource.

Explanation

The program can potentially fail to release a system resource. Resource leaks have at least two common causes: - Error conditions and other exceptional circumstances. - Confusion over which part of the program is responsible for releasing the resource. Most unreleased resource issues result in general software reliability problems. However, if an attacker can intentionally trigger a resource leak, the attacker may be able to launch a denial of service attack by depleting the resource pool. **Example:** The following method never closes the file handle it opens. The `finalize()` method for `FileInputStream` eventually calls `close()`, but there is no guarantee as to how long it will take before the `finalize()` method will be invoked. In a busy environment, this can result in the JVM using up all of its file handles.

```
private void processFile(String fName) throws FileNotFoundException,
IOException {
  FileInputStream fis = new FileInputStream(fName);
  int sz;
  byte[] byteArray = new byte[BLOCK_SIZE];
  while ((sz = fis.read(byteArray)) != -1) {
    processBytes(byteArray, sz);
  }
}
```

Recommendation

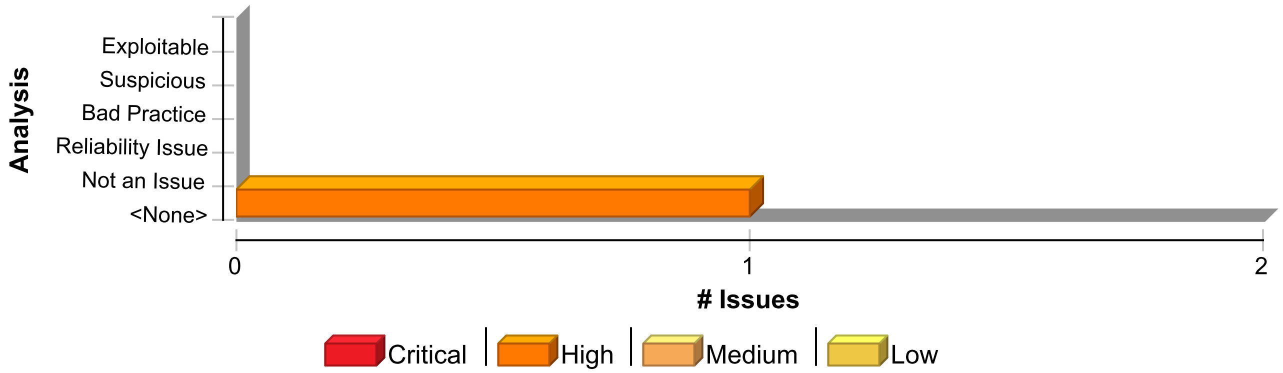
1. Never rely on `finalize()` to reclaim resources. In order for an object's `finalize()` method to be invoked, the garbage collector must determine that the object is eligible for garbage collection. Because the garbage collector is not required to run unless the JVM is low on memory, there is no guarantee that an object's `finalize()` method will be invoked in an expedient fashion. When the garbage collector finally does run, it may cause a large number of resources to be reclaimed in a short period of time, which can lead to "bursty" performance and lower overall system throughput. This effect becomes more pronounced as the load on the system increases. Finally, if it is possible for a resource reclamation operation to hang (if it requires communicating over a network to a database, for example), then the thread that is executing the `finalize()` method will hang. 2. Release resources in a `finally` block. The code for the Example should be rewritten as follows:

```
public void processFile(String fName) throws FileNotFoundException,
IOException {
  FileInputStream fis;
  try {
    fis = new FileInputStream(fName);
    int sz;
    byte[] byteArray = new byte[BLOCK_SIZE];
    while ((sz = fis.read(byteArray)) != -1) {
      processBytes(byteArray, sz);
    }
  }
  finally {
    if (fis != null) {
      safeClose(fis);
    }
  }
}
```

```
public static void safeClose(FileInputStream fis) {
  if (fis != null) {
    try {
      fis.close();
    } catch (IOException e) {
      log(e);
    }
  }
}
```

This solution uses a helper function to log the exceptions that might occur when trying to close the stream. Presumably this helper function will be reused whenever a stream needs to be closed. Also, the `processFile` method does not initialize the `fis` object to null. Instead, it checks to ensure that `fis` is not `null` before calling `safeClose()`. Without the `null` check, the Java compiler reports that `fis` might not be initialized. This choice takes advantage of Java's ability to detect uninitialized variables. If `fis` is initialized to `null` in a more complex method, cases in which `fis` is used without being initialized will not be detected by the compiler.

Issue Summary



Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Unreleased Resource: Streams	1	0	0	1
Total	**1**	**0**	**0**	**1**

Unreleased Resource: Streams **High**

Package: com.cts.dcrowd.rest.app.utils

src/main/java/com/cts/dcrowd/rest/app/utils/Util.java, line 760 (Unreleased Resource: Streams) **High**

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Control Flow)

Sink Details

Sink: fileInputStreamReader = new FileInputStream(...)
Enclosing Method: encodeFileToBase64Binary()
File: src/main/java/com/cts/dcrowd/rest/app/utils/Util.java:760
Taint Flags:

Unreleased Resource: Streams — High

Package: com.cts.dcrowd.rest.app.utils

src/main/java/com/cts/dcrowd/rest/app/utils/Util.java, line 760 (Unreleased Resource: Streams) — High

```
757  File file = new File(filePath);
758  String encodedfile = null;
759  try {
760  FileInputStream fileInputStreamReader = new FileInputStream(file);
761  byte[] bytes = new byte[(int)file.length()];
762  fileInputStreamReader.read(bytes);
763  encodedfile = new String(org.apache.commons.codec.binary.Base64.encodeBase64(bytes), "UTF-8");
```