Hewlett Packard Enterprise

HPE Security Fortify Audit Workbench

Developer Workbook

PS_INB_ERX_Fortify-2.0.3.043

FORTIFY®

Table of Contents
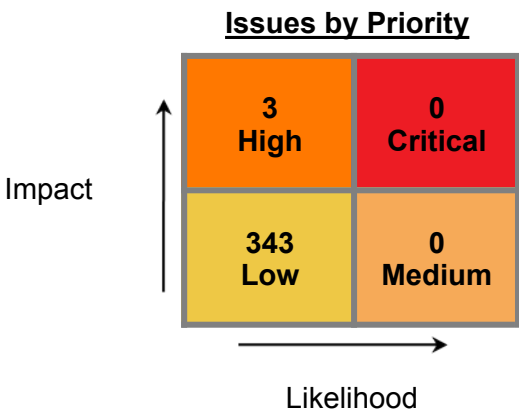
Executive Summary

This workbook is intended to provide all necessary details and information for a developer to understand and remediate the different issues discovered during the PS_INB_ERX_Fortify-2.0.3.043 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: PS_INB_ERX_Fortify-2.0

Project Version:

SCA: Results Present

WebInspect: Results Not Present

SecurityScope: Results Not Present

Other: Results Not Present

Issues by Priority

Impact

| 3 High | 0 Critical |
| 343 Low | 0 Medium |

Likelihood

Top Ten Critical Categories

This project does not contain any critical issues

Project Description

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

SCA

Date of Last Analysis:	Apr 25, 2017, 10:30 PM	**Engine Version:**	17.10.0156
Host Name:	HAM-LT51592	**Certification:**	VALID
Number of Files:	2,141	**Lines of Code:**	107,422

FORTIFY®

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	
Access Control: Database	0	1 / 1	0	0	1 / 1
Build Misconfiguration: External Maven Dependency Repository	0	0	0	0 / 7	0 / 7
Code Correctness: Byte Array to String Conversion	0	0	0	0 / 2	0 / 2
Code Correctness: Erroneous String Compare	0	0	0	0 / 5	0 / 5
Code Correctness: Misleading Method Signature	0	0	0	0 / 4	0 / 4
Cross-Site Request Forgery	0	0	0	0 / 8	0 / 8
Dead Code: Expression is Always false	0	0	0	0 / 1	0 / 1
Dead Code: Expression is Always true	0	0	0	0 / 1	0 / 1
Denial of Service	0	0	0	0 / 1	0 / 1
Denial of Service: StringBuilder	0	0	0	0 / 1	0 / 1
Hidden Field	0	0	0	0 / 2	0 / 2
J2EE Misconfiguration: Excessive Session Timeout	0	0	0	0 / 2	0 / 2
J2EE Misconfiguration: Incomplete Error Handling	0	0	0	0 / 1	0 / 1
J2EE Misconfiguration: Missing Error Handling	0	0	0	0 / 2	0 / 2
Missing Check against Null	0	0	0	0 / 6	0 / 6
Obsolete: Deprecated by ESAPI	0	0	0	0 / 52	0 / 52
Password Management	0	0	0	0 / 2	0 / 2
Password Management: Hardcoded Password	0	2 / 2	0	0	2 / 2
Password Management: Password in Comment	0	0	0	0 / 18	0 / 18
Poor Error Handling: Overly Broad Catch	0	0	0	0 / 19	0 / 19
Poor Error Handling: Overly Broad Throws	0	0	0	0 / 16	0 / 16
Poor Error Handling: Throw Inside Finally	0	0	0	0 / 1	0 / 1
Poor Style: Confusing Naming	0	0	0	0 / 167	0 / 167
Poor Style: Value Never Read	0	0	0	0 / 1	0 / 1
Redundant Null Check	0	0	0	0 / 2	0 / 2
SQL Injection	0	0	0	0 / 1	0 / 1
System Information Leak: External	0	0	0	0 / 7	0 / 7
System Information Leak: HTML Comment in JSP	0	0	0	0 / 5	0 / 5
System Information Leak: Internal	0	0	0	0 / 3	0 / 3
Trust Boundary Violation	0	0	0	0 / 6	0 / 6

Results Outline

Access Control: Database (1 issue)

Abstract

Without proper access control, executing a SQL statement that contains a user-controlled primary key can allow an attacker to view unauthorized records.

Explanation

Database access control errors occur when:

1. Data enters a program from an untrusted source.

2. The data is used to specify the value of a primary key in a SQL query.

Example 1: The following code uses a parameterized statement, which escapes metacharacters and prevents SQL injection vulnerabilities, to construct and execute a SQL query that searches for an invoice matching the specified identifier [1]. The identifier is selected from a list of all invoices associated with the current authenticated user.

```
...
id = Integer.decode(request.getParameter("invoiceID"));
String query = "SELECT * FROM invoices WHERE id = ?";
PreparedStatement stmt = conn.prepareStatement(query);
stmt.setInt(1, id);
ResultSet results = stmt.execute();
...
```

The problem is that the developer has failed to consider all of the possible values of `id`. Although the interface generates a list of invoice identifiers that belong to the current user, an attacker may bypass this interface to request any desired invoice. Because the code in this example does not check to ensure that the user has permission to access the requested invoice, it will display any invoice, even if it does not belong to the current user.

Some think that in the mobile world, classic web application vulnerabilities, such as database access control errors, do not make sense -- why would the user attack themself? However, keep in mind that the essence of mobile platforms is applications that are downloaded from various sources and run alongside each other on the same device. The likelihood of running a piece of malware next to a banking application is high, which necessitates expanding the attack surface of mobile applications to include inter-process communication.

Example 2: The following code adapts Example 1 to the Android platform.

```
...
```

```
        String id = this.getIntent().getExtras().getString("invoiceID");
        String query = "SELECT * FROM invoices WHERE id = ?";
        SQLiteDatabase db = this.openOrCreateDatabase("DB", MODE_PRIVATE,
null);
        Cursor c = db.rawQuery(query, new Object[]{id});
...
```

A number of modern web frameworks provide mechanisms for performing validation of user input. Struts and Spring MVC are among them. To highlight the unvalidated sources of input, the rulepacks dynamically re-prioritize the issues reported by HPE Security Fortify Static Code Analyzer by lowering their probability of exploit and providing pointers to the supporting evidence whenever the framework validation mechanism is in use. We refer to this feature as Context-Sensitive Ranking. To further assist the HPE Security Fortify user with the auditing process, the HPE Security Fortify Software Security Research group makes available the Data Validation project template that groups the issues into folders based on the validation mechanism applied to their source of input.

Recommendation

Rather than relying on the presentation layer to restrict values submitted by the user, access control should be handled by the application and database layers. Under no circumstances should a user be allowed to retrieve or modify a row in the database without the appropriate permissions. Every query that accesses the database should enforce this policy, which can often be accomplished by simply including the current authenticated username as part of the query.

Example 3: The following code implements the same functionality as Example 1 but imposes an additional constraint requiring that the current authenticated user have specific access to the invoice.

```
...
userName = ctx.getAuthenticatedUserName();
id = Integer.decode(request.getParameter("invoiceID"));
String query =
        "SELECT * FROM invoices WHERE id = ? AND user = ?";
PreparedStatement stmt = conn.prepareStatement(query);
stmt.setInt(1, id);
stmt.setString(2, userName);
ResultSet results = stmt.execute();
...
```
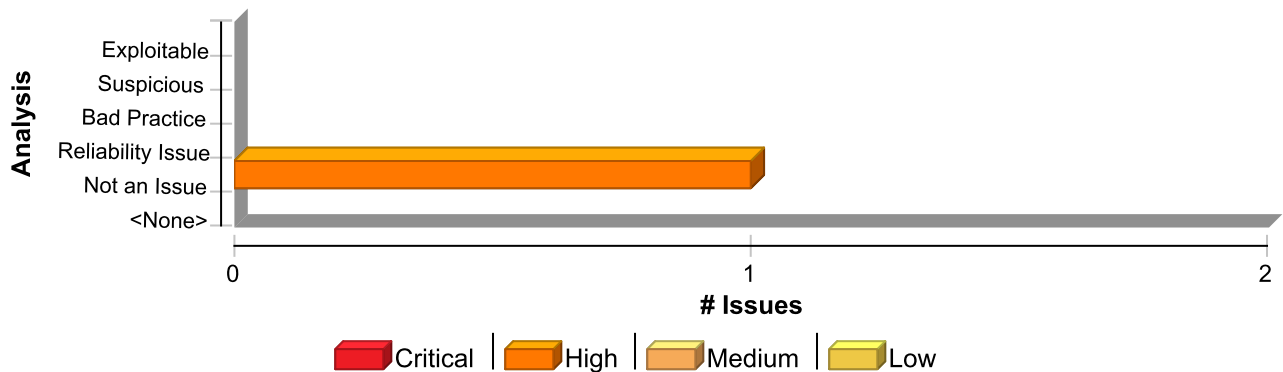
And here is an Android equivalent:

```
...
        PasswordAuthentication pa = authenticator.getPasswordAuthentication();
        String userName = pa.getUserName();
        String id = this.getIntent().getExtras().getString("invoiceID");
        String query = "SELECT * FROM invoices WHERE id = ? AND user = ?";
        SQLiteDatabase db = this.openOrCreateDatabase("DB", MODE_PRIVATE,
null);
        Cursor c = db.rawQuery(query, new Object[]{id, userName});
...
```

Issue Summary

Analysis (y-axis): Exploitable, Suspicious, Bad Practice, Reliability Issue, Not an Issue, <None>

 # Issues (x-axis): 0, 1, 2

Legend: Critical | High | Medium | Low

Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Access Control: Database	1	0	0	1
Total	**1**	**0**	**0**	**1**

Access Control: Database	High

URL: /managePharm/updatePharmacy

PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/ BaseDao.java, line 29 (Access Control: Database)	High

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Data Flow)

Audit Details

Analysis   Not an Issue

Audit Comments

VHAHAMWandlJ: Tue Apr 25 2017 22:36:57 GMT-0400 (EDT)
The method getByKey() identified in this finding is only available to an authenticated and authorized user thereby mitigating the risk of access to unauthorized records.

Source Details

Source: updatePharmacy(1)
From: gov.va.med.pharmacy.web.administration.PharmacyManagementController.updatePha
rmacy
File: PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/administration/PharmacyMa
nagementController.java:122
URL: /managePharm/updatePharmacy

```
119   }
120
121   @RequestMapping(value = "/updatePharmacy", method = RequestMethod.POST)
122   public ModelAndView updatePharmacy(HttpServletRequest request,@Valid
@ModelAttribute("pharmacyEditForm") PharmacyForm pharmacyForm, BindingResult
bindingResult) throws IOException {
123
```

Access Control: Database	High

PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/ BaseDao.java, line 29 (Access Control: Database)	High

```
124   LOG.info("Trying to persist pharmacy information.");
125
```

Sink Details

Sink: org.hibernate.Session.get()
Enclosing Method: getByKey()
File: PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/BaseDao.java:29
Taint Flags: NUMBER, PRIMARY_KEY, WEB

```
26
27   @SuppressWarnings("unchecked")
28   public T getByKey(long id) {
29   return (T) getSession().get(persistentClass, id);
30   }
31
32   public void persist(T entity) {
```

Build Misconfiguration: External Maven Dependency Repository (7 issues)

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>
```
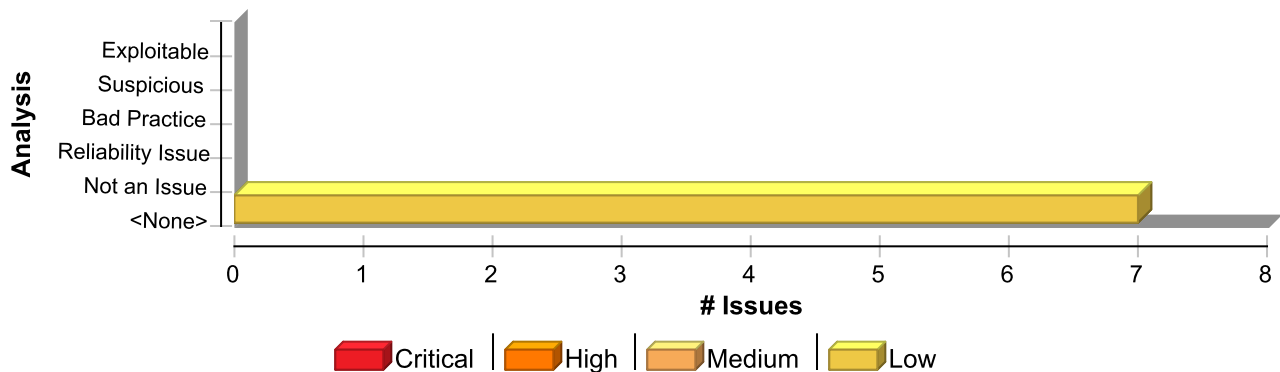
```
    <pluginRepositories>
      ...
    </pluginRepositories>
    ...
</project>
```

Issue Summary

Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Build Misconfiguration: External Maven Dependency Repository	7	0	0	7
Total	**7**	**0**	**0**	**7**

Build Misconfiguration: External Maven Dependency Repository	Low
Package: PS_INB_ERX_Common	
PS_INB_ERX_Common/pom.xml, line 2 (Build Misconfiguration: External Maven Dependency Repository)	Low

Issue Details

Kingdom: Environment
Scan Engine: SCA (Configuration)

Sink Details

Sink: //project/repositories
File: PS_INB_ERX_Common/pom.xml:2
Taint Flags:

```
1 <project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/
  XMLSchema-instance"
2   xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-
  v4_0_0.xsd">
3
4   <modelVersion>4.0.0</modelVersion>
5   <artifactId>PS_INB_ERX_Common</artifactId>
6
7 undefined
```

Build Misconfiguration: External Maven Dependency Repository	Low

Package: PS_INB_ERX_Fortify

PS_INB_ERX_Fortify/pom.xml, line 22 (Build Misconfiguration: External Maven Dependency Repository)	Low

Issue Details

Kingdom: Environment
Scan Engine: SCA (Configuration)

Sink Details

Sink: //project/repositories
File: PS_INB_ERX_Fortify/pom.xml:22
Taint Flags:

```
19
20  </modules>
21
22  <repositories>
23  <repository>
24  <id>project.local</id>
25  <name>project</name>
```

Package: PS_INB_ERX_JMS

PS_INB_ERX_JMS/pom.xml, line 15 (Build Misconfiguration: External Maven Dependency Repository)	Low

Issue Details

Kingdom: Environment
Scan Engine: SCA (Configuration)

Sink Details

Sink: //project/repositories
File: PS_INB_ERX_JMS/pom.xml:15
Taint Flags:

```
12  <version>${version.number}</version>
13  <relativePath>../PS_INB_ERX/pom.xml</relativePath>
14  </parent>
15  <repositories>
16  <repository>
17  <id>project.local</id>
18  <name>project</name>
```

Package: PS_INB_ERX_Model

PS_INB_ERX_Model/pom.xml, line 2 (Build Misconfiguration: External Maven Dependency Repository)	Low

Issue Details

Kingdom: Environment
Scan Engine: SCA (Configuration)

Build Misconfiguration: External Maven Dependency Repository	Low

Package: PS_INB_ERX_Model

PS_INB_ERX_Model/pom.xml, line 2 (Build Misconfiguration: External Maven Dependency Repository)	Low

Sink Details

Sink: //project/repositories
File: PS_INB_ERX_Model/pom.xml:2
Taint Flags:

```
1 <project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/
  XMLSchema-instance"
2  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-
  v4_0_0.xsd">
3
4  <modelVersion>4.0.0</modelVersion>
5  <name>PS_INB_ERX_Model</name>
6
7 undefined
```

Package: PS_INB_ERX_Service

PS_INB_ERX_Service/pom.xml, line 16 (Build Misconfiguration: External Maven Dependency Repository)	Low

Issue Details

Kingdom: Environment
Scan Engine: SCA (Configuration)

Sink Details

Sink: //project/repositories
File: PS_INB_ERX_Service/pom.xml:16
Taint Flags:

```
13  <relativePath>../PS_INB_ERX/pom.xml</relativePath>
14  </parent>
15
16  <repositories>
17  <repository>
18  <id>project.local</id>
19  <name>project</name>
```

Package: PS_INB_ERX_UI

PS_INB_ERX_UI/pom.xml, line 17 (Build Misconfiguration: External Maven Dependency Repository)	Low

Issue Details

Kingdom: Environment
Scan Engine: SCA (Configuration)

Sink Details

Build Misconfiguration: External Maven Dependency Repository	Low

Package: PS_INB_ERX_UI

PS_INB_ERX_UI/pom.xml, line 17 (Build Misconfiguration: External Maven Dependency Repository)	Low

Sink: //project/repositories
File: PS_INB_ERX_UI/pom.xml:17
Taint Flags:

14	`</parent>`
15	
16	
17	`<repositories>`
18	`<repository>`
19	`<id>project.local</id>`
20	`<name>project</name>`

Package: PS_INB_ERX_WS

PS_INB_ERX_WS/pom.xml, line 16 (Build Misconfiguration: External Maven Dependency Repository)	Low

Issue Details

Kingdom: Environment
Scan Engine: SCA (Configuration)

Sink Details

Sink: //project/repositories
File: PS_INB_ERX_WS/pom.xml:16
Taint Flags:

13	`<relativePath>../PS_INB_ERX/pom.xml</relativePath>`
14	`</parent>`
15	
16	`<repositories>`
17	`<repository>`
18	`<id>project.local</id>`
19	`<name>project</name>`

Code Correctness: Byte Array to String Conversion (2 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.
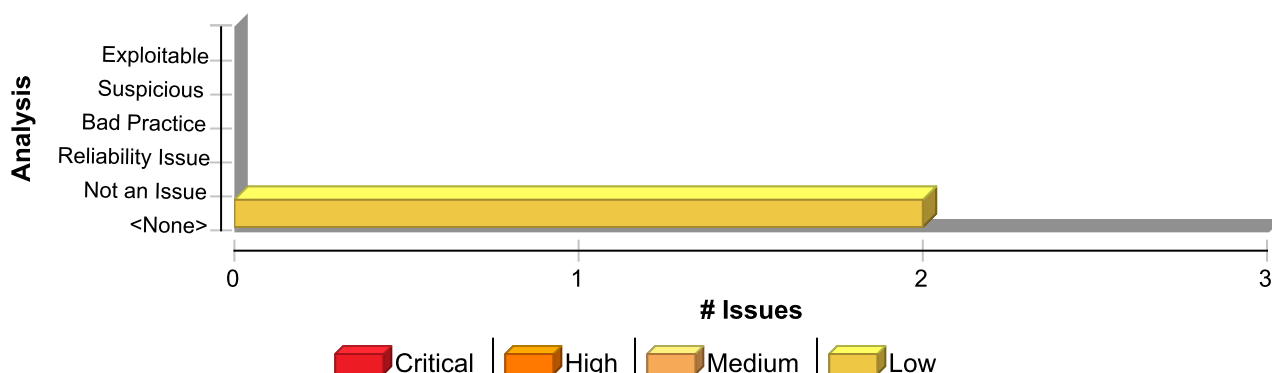
Recommendation

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	2	0	0	2
Total	**2**	**0**	**0**	**2**

Code Correctness: Byte Array to String Conversion	Low
Package: gov.va.med.pharmacy.jms.messaging	
PS_INB_ERX_JMS/src/main/java/gov/va/med/pharmacy/jms/messaging/ InboundMsgProcessor.java, line 43 (Code Correctness: Byte Array to String Conversion)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Semantic)

Sink Details

Sink: String()
Enclosing Method: getERxMessage()
File: PS_INB_ERX_JMS/src/main/java/gov/va/med/pharmacy/jms/messaging/InboundMsgProcessor.java:43
Taint Flags:

```
40
41   eRxMessage.readBytes(messageByteArray, messageLength);
42
43   String message = new String(messageByteArray);
44
45   inboundeRx.setMessage(message);
46
```

Code Correctness: Byte Array to String Conversion	Low

Package: gov.va.med.pharmacy.jms.messaging

PS_INB_ERX_JMS/src/main/java/gov/va/med/pharmacy/jms/messaging/ InboundMsgProducer.java, line 39 (Code Correctness: Byte Array to String Conversion)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Semantic)

Sink Details

Sink: String()
Enclosing Method: postProcessMessage()
File: PS_INB_ERX_JMS/src/main/java/gov/va/med/pharmacy/jms/messaging/InboundMsgProducer.java:39
Taint Flags:

```
36   byte[] newMessage = new byte[length.intValue()];
37   int messageRead = byteMessage.readBytes(newMessage);
38
39   LOG.debug("--- sending message", new String(newMessage));
40
41   return msg;
42   }
```

Code Correctness: Erroneous String Compare (5 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.
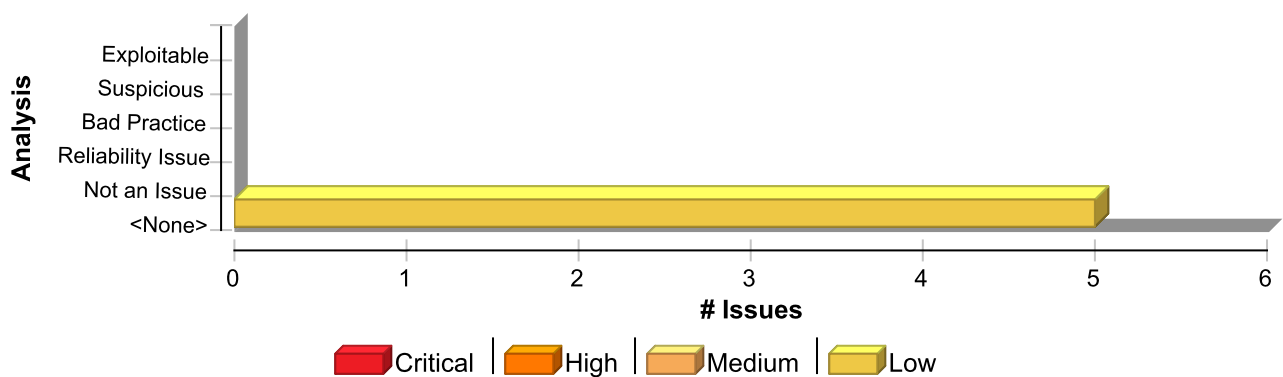
Recommendation

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	5	0	0	5
Total	**5**	**0**	**0**	**5**

Code Correctness: Erroneous String Compare	Low

Package: gov.va.med.pharmacy.persistence.track.impl	

PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/track/impl/ NcpdpMessagesDaoImpl.java, line 633 (Code Correctness: Erroneous String Compare)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Operation
Enclosing Method: searchMessages()
File: PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/track/impl/NcpdpMessagesDaoImpl.java:633
Taint Flags:

```
630   sql = sql + " and t.pharmacy_id in (select pharmacy_id from pharmacy where visn = '" +
visn + "')\r\n";
631   };
632   sql = sql + " and t.pharmacy_id in (select pharmacy_id from pharmacy where va_station_id
like ? ) \r\n";
633   if (patientSsn != "") {
634   sql = sql + " and x.patient_ssn = '" + patientSsn + "'\r\n";
635   };
636   sql = sql +
```

PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/track/impl/ NcpdpMessagesDaoImpl.java, line 624 (Code Correctness: Erroneous String Compare)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Operation
Enclosing Method: searchMessages()
File: PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/track/impl/NcpdpMessagesDaoImpl.java:624
Taint Flags:

```
621   if (messageType != ""){
622   sql = sql + " and t.message_type = '" + messageType + "' \r\n";
623   };
624   if (messageStatus != ""){
625   sql = sql + " and t.message_status = '" + messageStatus + "' \r\n";
626   } else { //excluding the 3006 status messages because they are invalid XML
```

Code Correctness: Erroneous String Compare	Low
Package: gov.va.med.pharmacy.persistence.track.impl	

PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/track/impl/ NcpdpMessagesDaoImpl.java, line 624 (Code Correctness: Erroneous String Compare)	Low

```
627   sql = sql + " and t.message_status not in ('3006') \r\n";
```

PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/track/impl/ NcpdpMessagesDaoImpl.java, line 639 (Code Correctness: Erroneous String Compare)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Operation
Enclosing Method: searchMessages()
File: PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/track/impl/NcpdpMessagesDaoImpl.java:639
Taint Flags:

```
636   sql = sql +
637   " and nvl(x.patient_last_name,' ') like ?\r\n" +
638   " and nvl(x.patient_first_name,' ') like ?\r\n";
639   if (patientDob != "") {
640   sql = sql + " and x.patient_dob = to_date('" + patientDob + "','YYYY-MM-DD') \r\n";
641   };
642   sql = sql +
```

PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/track/impl/ NcpdpMessagesDaoImpl.java, line 629 (Code Correctness: Erroneous String Compare)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Operation
Enclosing Method: searchMessages()
File: PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/track/impl/NcpdpMessagesDaoImpl.java:629
Taint Flags:

```
626   } else { //excluding the 3006 status messages because they are invalid XML
627   sql = sql + " and t.message_status not in ('3006') \r\n";
628   };
629   if (visn != "") {
630   sql = sql + " and t.pharmacy_id in (select pharmacy_id from pharmacy where visn = '" +
visn + "')\r\n";
631   };
```

Code Correctness: Erroneous String Compare	Low
Package: gov.va.med.pharmacy.persistence.track.impl	
PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/track/impl/ NcpdpMessagesDaoImpl.java, line 629 (Code Correctness: Erroneous String Compare)	**Low**

```
632   sql = sql + " and t.pharmacy_id in (select pharmacy_id from pharmacy where va_station_id
like ? ) \r\n";
```

PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/track/impl/ NcpdpMessagesDaoImpl.java, line 621 (Code Correctness: Erroneous String Compare)	**Low**

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Operation
Enclosing Method: searchMessages()
File: PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/track/impl/NcpdpMessagesDaoImpl.java:621
Taint Flags:

```
618   System.out.println("va_station_id:" + vaStationId);
619   */
620
621   if (messageType != ""){
622   sql = sql + " and t.message_type = '" + messageType + "' \r\n";
623   };
624   if (messageStatus != ""){
```

Code Correctness: Misleading Method Signature (4 issues)

Abstract

This looks like an effort to override a common Java method, but it probably does not have the intended effect.

Explanation

This method's name is similar to a common Java method name, but it is either spelled incorrectly or the argument list causes it to not override the intended method.

Example 1: The following method is meant to override `Object.equals()`:

```
public boolean equals(Object obj1, Object obj2) {
  ...
}
```

But since `Object.equals()` only takes a single argument, the method above is never called.

Recommendation

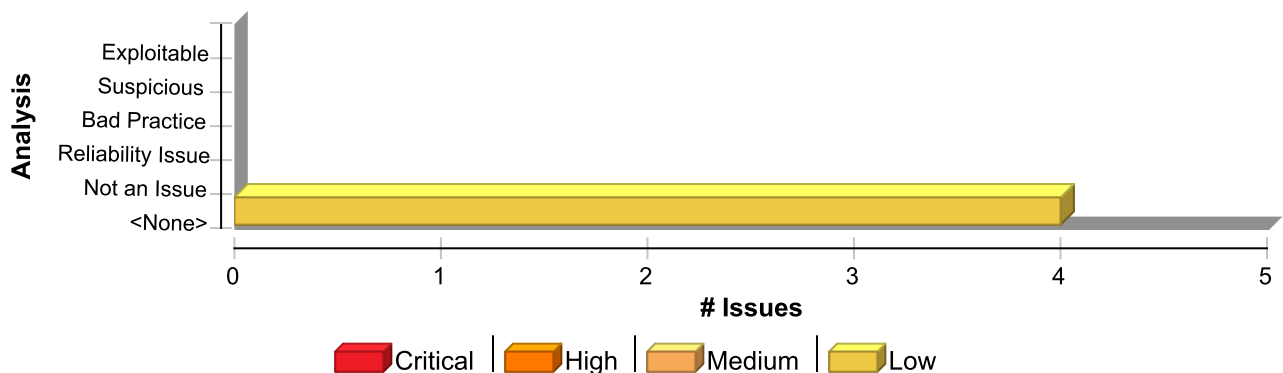Carefully check to make sure the method does what was intended.

Example 2: The code in Example 1 could be rewritten in the following way:

```
public boolean equals(Object obj) {
  ...
}
```

Issue Summary

Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Code Correctness: Misleading Method Signature	4	0	0	4
Total	**4**	**0**	**0**	**4**

Code Correctness: Misleading Method Signature	Low

| **Package: gov.va.med.pharmacy.persistence.model.usertypes** | |

PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/model/usertypes/OutboundNcpdpMsgUserType.java, line 56 (Code Correctness: Misleading Method Signature)	**Low**

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: hashCode
Enclosing Method: hashCode()
File: PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/model/usertypes/OutboundNcpdpMsgUserType.java:56
Taint Flags:

```
53   }
54
55   @Override
56   public int hashCode(final Object x) {
57   return (x != null) ? x.hashCode() : 0;
58   }
59
```

PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/model/usertypes/InboundNcpdpMsgUserType.java, line 49 (Code Correctness: Misleading Method Signature)	**Low**

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: hashCode
Enclosing Method: hashCode()
File: PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/model/usertypes/InboundNcpdpMsgUserType.java:49
Taint Flags:

```
46   }
47
48   @Override
49   public int hashCode(final Object x) {
50   return (x != null) ? x.hashCode() : 0;
51   }
52
```

Code Correctness: Misleading Method Signature	Low

Package: gov.va.med.pharmacy.persistence.model.usertypes

PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/model/ usertypes/InboundNcpdpMsgUserType.java, line 44 (Code Correctness: Misleading Method Signature)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: equals
Enclosing Method: equals()
File: PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/model/usertypes/
InboundNcpdpMsgUserType.java:44
Taint Flags:

```
41  }
42
43  @Override
44  public boolean equals(final Object x, final Object y) {
45  return (x != null) && x.equals(y);
46  }
47
```

PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/model/ usertypes/OutboundNcpdpMsgUserType.java, line 51 (Code Correctness: Misleading Method Signature)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: equals
Enclosing Method: equals()
File: PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/model/usertypes/
OutboundNcpdpMsgUserType.java:51
Taint Flags:

```
48  }
49
50  @Override
51  public boolean equals(final Object x, final Object y) {
52  return (x != null) && x.equals(y);
53  }
54
```

Cross-Site Request Forgery (8 issues)

Abstract

Form posts must contain a user-specific secret in order to prevent an attacker from making unauthorized requests.

Explanation

A cross-site request forgery (CSRF) vulnerability occurs when:
1. A Web application uses session cookies.

2. The application acts on an HTTP request without verifying that the request was made with the user's consent.

A nonce is a cryptographic random value that is sent with a message to prevent replay attacks. If the request does not contain a nonce that proves its provenance, the code that handles the request is vulnerable to a CSRF attack (unless it does not change the state of the application). This means a Web application that uses session cookies has to take special precautions in order to ensure that an attacker can't trick users into submitting bogus requests. Imagine a Web application that allows administrators to create new accounts by submitting this form:

```
<form method="POST" action="/new_user" >
  Name of new user: <input type="text" name="username">
  Password for new user: <input type="password" name="user_passwd">
    <input type="submit" name="action" value="Create User">
</form>
```

An attacker might set up a Web site with the following:

```
<form method="POST" action="http://www.example.com/new_user">
  <input type="hidden" name="username" value="hacker">
  <input type="hidden" name="user_passwd" value="hacked">
</form>
<script>
  document.usr_form.submit();
</script>
```

If an administrator for `example.com` visits the malicious page while she has an active session on the site, she will unwittingly create an account for the attacker. This is a CSRF attack. It is possible because the application does not have a way to determine the provenance of the request. Any request could be a legitimate action chosen by the user or a faked action set up by an attacker. The attacker does not get to see the Web page that the bogus request generates, so the attack technique is only useful for requests that alter the state of the application.

Applications that pass the session identifier in the URL rather than as a cookie do not have CSRF problems because there is no way for the attacker to access the session identifier and include it as part of the bogus request.

CSRF is entry number five on the 2007 OWASP Top 10 list.

Recommendation

Applications that use session cookies must include some piece of information in every form post that the back-end code can use to validate the provenance of the request. One way to do that is to include a random request identifier or nonce, like this:

```
RequestBuilder rb = new RequestBuilder(RequestBuilder.POST, "/new_user");
body = addToPost(body, new_username);
body = addToPost(body, new_passwd);
body = addToPost(body, request_id);
rb.sendRequest(body, new NewAccountCallback(callback));
```

Then the back-end logic can validate the request identifier before processing the rest of the form data. When possible, the request identifier should be unique to each server request rather than shared across every request for a particular session. As with session identifiers, the harder it is for an attacker to guess the request identifier, the harder it is to conduct a successful CSRF attack. The token should not be easily guessed and it should be protected in the same way that session tokens are protected, such as using SSLv3.

Additional mitigation techniques include:

Framework protection: Most modern web application frameworks embed CSRF protection and they will automatically include and verify CSRF tokens.
Use a Challenge-Response control: Forcing the customer to respond to a challenge sent by the server is a strong defense against CSRF. Some of the challenges that can be used for this purpose are: CAPTCHAs, password re-authentication and one-time tokens.
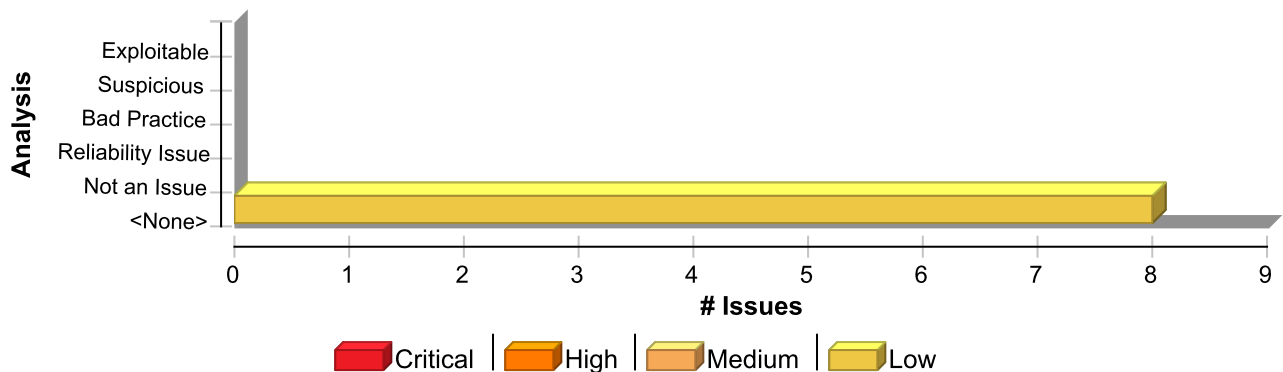**Check HTTP Referer/Origin headers:** An attacker won't be able to spoof these headers while performing a CSRF attack. This makes these headers a useful method to prevent CSRF attacks.
Double-submit Session Cookie: Sending the session ID Cookie as a hidden form value in addition to the actual session ID Cookie is a good protection against CSRF attacks. The server will check both values and make sure they are identical before processing the rest of the form data. If an attacker submits a form in behalf of a user, he won't be able to modify the session ID cookie value as per the same-origin-policy.
Limit Session Lifetime: When accessing protected resources using a CSRF attack, the attack will only be valid as long as the session ID sent as part of the attack is still valid on the server. Limiting the Session lifetime will reduce the probability of a successful attack.

The techniques described here can be defeated with XSS attacks. Effective CSRF mitigation includes XSS mitigation techniques.

Issue Summary

Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Cross-Site Request Forgery	8	0	0	8
Total	**8**	**0**	**0**	**8**

Cross-Site Request Forgery	Low
Package: .src.main.webapp.WEB-INF	

PS_INB_ERX_UI/src/main/webapp/WEB-INF/trackeRx.jsp, line 11 (Cross-Site Request Forgery)	Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Content)

Sink Details

File: PS_INB_ERX_UI/src/main/webapp/WEB-INF/trackeRx.jsp:11
Taint Flags:

```
8 <h1 class="titleBar"><span class="headerText">Track/Audit eRx</span></h1>
9
10 <br/>
11 <form id="searchCriteriaForm">
12
13 <div style="width: 1000px;" >
14 <div style="width: 100px;" >
```

PS_INB_ERX_UI/src/main/webapp/WEB-INF/reports.jsp, line 19 (Cross-Site Request Forgery)	Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Content)

Sink Details

File: PS_INB_ERX_UI/src/main/webapp/WEB-INF/reports.jsp:19
Taint Flags:

```
16 </div>
```

Cross-Site Request Forgery	Low

Package: .src.main.webapp.WEB-INF	

PS_INB_ERX_UI/src/main/webapp/WEB-INF/reports.jsp, line 19 (Cross-Site Request Forgery)	Low

17	
18	`<div id="summaryReportFilter">`
19	`<form id="summaryReportForm" >`
20	`<h2 class="h2title"><span class="reportTitle"></span></h2>`
21	`<label for="visnSelection">VISN:  </label>`
22	`<div id="visnSelect" tabindex="0" style="width: 40px; ">`

PS_INB_ERX_UI/src/main/webapp/WEB-INF/managepharmacy.jsp, line 35 (Cross-Site Request Forgery)	Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Content)

Sink Details

File: PS_INB_ERX_UI/src/main/webapp/WEB-INF/managepharmacy.jsp:35
Taint Flags:

32	`</c:if>`
33	
34	`<div id="pharmacyManagement">`
35	`<form:form id="pharmacyFilterForm" name="pharmacyFilterForm" method="POST" action="$ {pageContext.request.contextPath}/inb-erx/managePharm/addPharmacy">`
36	
37	`<label for="visnSelection" style="width: 40px; padding-top: 6px; padding-left: 5px;">VISN: </label>`
38	`<div id="visnSelect" tabindex="0" style="width: 40px; ">`

PS_INB_ERX_UI/src/main/webapp/WEB-INF/loginErrorLayout.jsp, line 24 (Cross-Site Request Forgery)	Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Content)

Sink Details

File: PS_INB_ERX_UI/src/main/webapp/WEB-INF/loginErrorLayout.jsp:24
Taint Flags:

21	`<hr />`
22	`<div id="content">`
23	` `
24	`<form action="${index}" method="post" >`
25	
26	`<div align="center">`
27	` `

Cross-Site Request Forgery	Low

Package: .src.main.webapp.WEB-INF	

PS_INB_ERX_UI/src/main/webapp/WEB-INF/loginErrorLayout.jsp, line 24 (Cross-Site Request Forgery)	Low

PS_INB_ERX_UI/src/main/webapp/WEB-INF/manageusers.jsp, line 10 (Cross-Site Request Forgery)	Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Content)

Sink Details

File: PS_INB_ERX_UI/src/main/webapp/WEB-INF/manageusers.jsp:10
Taint Flags:

```
7   <div id="userRoles">
8   <h2 class="titleBar"><span> Users</span>
9   </h2>
10  <form:form name="manageUsersForm" id ="manageUsersForm"
    modelAttribute="userManagementModel" method="POST" action="${pageContext.request.contextPath}/
    inb-erx/manageUsers/saveUserData">
11  <c:if test="${not empty data.errorMessage }">
12  <div id="errorMessages" class="error" aria-describedby="" tabindex="0">
13  <p id="errorTitle">
```

PS_INB_ERX_UI/src/main/webapp/WEB-INF/addpharmacy.jsp, line 33 (Cross-Site Request Forgery)	Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Content)

Sink Details

File: PS_INB_ERX_UI/src/main/webapp/WEB-INF/addpharmacy.jsp:33
Taint Flags:

```
30
31  <div id="pharmacyManagement">
32
33  <form:form id="pharmacyAddForm" name="pharmacyAddForm" commandName="pharmacyAddForm"
    method="POST" action="${pageContext.request.contextPath}/inb-erx/managePharm/addNewPharmacy">
34
35  <div id="list">
36
```

PS_INB_ERX_UI/src/main/webapp/WEB-INF/editpharmacy.jsp, line 33 (Cross-Site Request Forgery)	Low

Issue Details

Cross-Site Request Forgery Low

Package: .src.main.webapp.WEB-INF

PS_INB_ERX_UI/src/main/webapp/WEB-INF/editpharmacy.jsp, line 33 (Cross-Site Request Forgery) Low

Kingdom: Encapsulation
Scan Engine: SCA (Content)

Sink Details

File: PS_INB_ERX_UI/src/main/webapp/WEB-INF/editpharmacy.jsp:33
Taint Flags:

```
30
31 <div id="pharmacyEditScreen">
32
33 <form:form id="pharmacyEditForm" name="pharmacyEditForm" commandName="pharmacyEditForm"
   method="POST" action="${pageContext.request.contextPath}/inb-erx/managePharm/updatePharmacy">
34
35  <div id="list">
36
```

PS_INB_ERX_UI/src/main/webapp/WEB-INF/landing.jsp, line 28 (Cross-Site Request Forgery) Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Content)

Sink Details

File: PS_INB_ERX_UI/src/main/webapp/WEB-INF/landing.jsp:28
Taint Flags:

```
25 <br />    <br />
26
27 <br/>
28 <form action="${index}" method="post" >
29
30 <div align="center">
31 <p>
```

Dead Code: Expression is Always false (1 issue)

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();
```
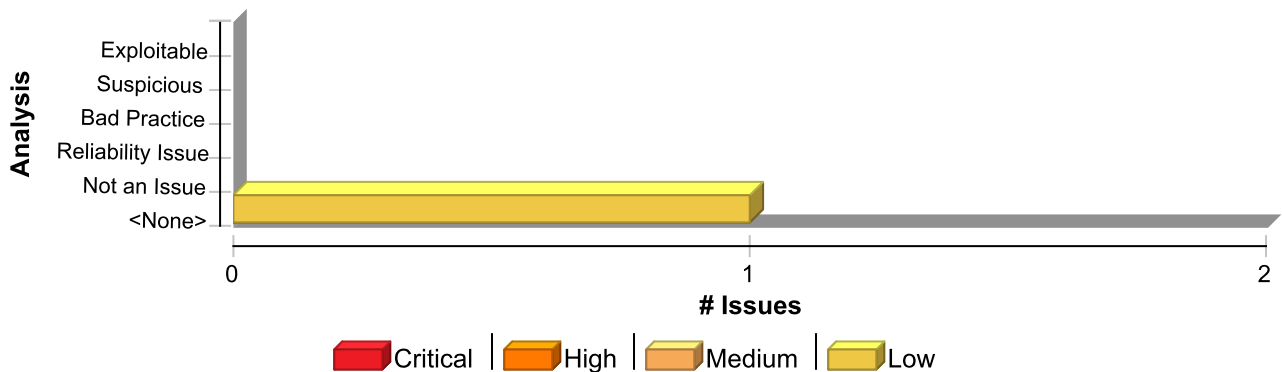
```
        }
}
```

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	1	0	0	1
Total	**1**	**0**	**0**	**1**

Dead Code: Expression is Always false	Low

Package: gov.va.med.pharmacy.jaxrs.eAnde.dao.impl	

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/eAnde/dao/impl/ PatientInfoWSDao.java, line 19 (Dead Code: Expression is Always false)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: IfStatement
Enclosing Method: lookupPatientByKey()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/eAnde/dao/impl/PatientInfoWSDao.java:19
Taint Flags:

```
16
17  try {
18    getEESummaryResponse eeSummaryResponse = eeClient.returnResponse(key);
19    if (eeSummaryResponse != null){
20
patientInfo.setEligibilityStatus(eeSummaryResponse.getSummary().getEligibilityVerificationInfo()
21
patientInfo.setEnrollmentStatus(eeSummaryResponse.getSummary().getEnrollmentDeterminationInfo().
22  };
```

Dead Code: Expression is Always false	Low
Package: gov.va.med.pharmacy.jaxrs.eAnde.dao.impl	
PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/eAnde/dao/impl/ PatientInfoWSDao.java, line 19 (Dead Code: Expression is Always false)	Low

Dead Code: Expression is Always true (1 issue)

Abstract

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

Explanation

This expression (or part of it) will always evaluate to `true`; 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 `true`. (The variable `firstCall` is mistakenly used twice.) The result is that the expression `firstCall || secondCall` will always evaluate to `true`, so `setUpForCall()` will always be invoked.

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

  if (fCall < 0) {
    cancelFCall();
    firstCall = false;
  }
  if (sCall < 0) {
    cancelSCall();
    firstCall = false;
  }

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

Example 2: The following method tries to check the variables `firstCall` and `secondCall`. (The variable `firstCall` is mistakenly set to `true` instead of being checked.) The result is that the first part of the expression `firstCall = true && secondCall == true` will always evaluate to `true`.

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

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

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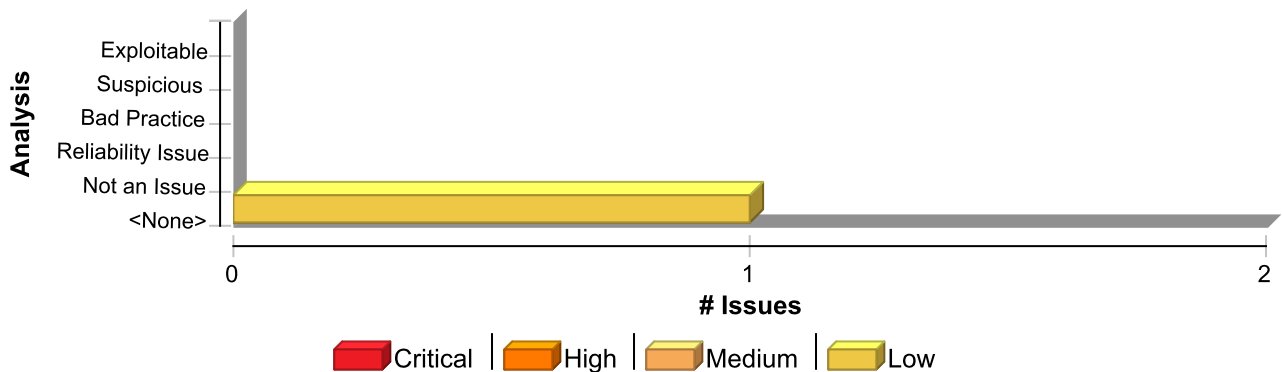
```
        }
    }
}
```

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 true	1	0	0	1
Total	**1**	**0**	**0**	**1**

Dead Code: Expression is Always true	Low
Package: gov.va.med.pharmacy.jaxrs.eAnde.services.impl	
PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/eAnde/services/impl/ EESummary_Client.java, line 72 (Dead Code: Expression is Always true)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: IfStatement
Enclosing Method: returnResponse()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/eAnde/services/impl/EESummary_Client.java:72
Taint Flags:

```
69
70   URL wsdlURL = EeSummaryPortService.WSDL_LOCATION;
71
72   if (url.length() > 0 && url != null && !"".equals(url)) {
73   File wsdlFile = new File(url);
74   try {
75   if (wsdlFile.exists()) {
```

Denial of Service (1 issue)

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.
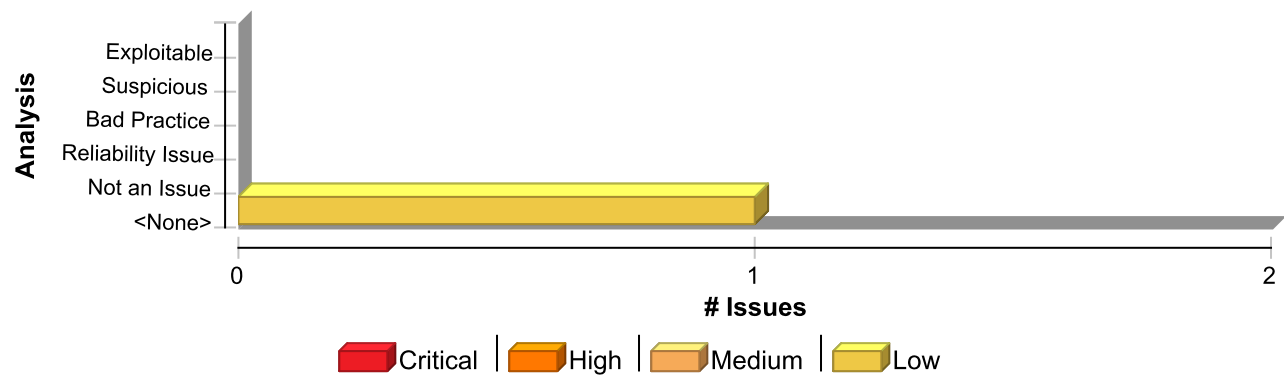
```
InputStream zipInput = zipFile.getInputStream(zipEntry);
```

```
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	1	0	0	1
Total	**1**	**0**	**0**	**1**

Denial of Service	Low
Package: gov.va.med.pharmacy.jaxrs.tovista.dao.impl	
PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/tovista/dao/impl/ VistaInboundRpc.java, line 68 (Denial of Service)	Low

Issue Details

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

Sink Details

Sink: readLine()
Enclosing Method: sendMessage()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/tovista/dao/impl/VistaInboundRpc.java:68
Taint Flags:

```
65   String line = null;
```

Denial of Service	Low

Package: gov.va.med.pharmacy.jaxrs.tovista.dao.impl

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/tovista/dao/impl/ VistaInboundRpc.java, line 68 (Denial of Service)	Low

```
66   int i = 0;
67   try{
68   while(null != (line = br.readLine())) {
69   sb.append(line.trim());
70   LOG.debug("xml("+ i +")=" +line.trim());
71   i++;
```

Denial of Service: StringBuilder (1 issue)

Abstract

Appending untrusted data to `StringBuilder` instance initialized with the default backing-array size can cause the JVM to over-consume heap memory space.

Explanation

Appending user-controlled data to a `StringBuilder` instance initialized with the default backing character array size (16) can cause the application to consume large amounts of heap memory while resizing the underlying array to fit user's data. Everytime new data is appended to a `StringBuilder` instance, it will try to fit it on its backing character array. If data does not fit, a new array will be created doubling the previous size while the old array will remain in the heap until it is garbage collected. This defect can be used to execute a Denial of Service (DoS) attack.

Example 1: User-controlled data is appended to a `StringBuilder` instance initialized with the default constructor.

```
...
StringBuilder sb = new StringBuilder();
sb.append(request.getParameter("foo"));
...
```
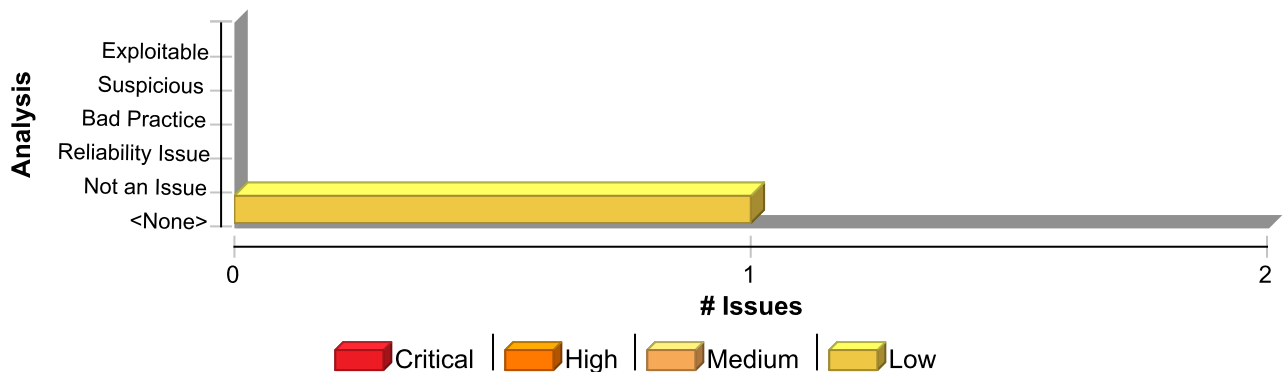
Recommendation

Initialize the `StringBuilder` with a size similar to the length of the expected data in order to reduce the number of times the backing array needs to be resized. Check the size of the data before appending it to a `StringBuilder` instance.

Example 2: User-controlled data is appended to a `StringBuilder` instance initialized with the default constructor.

```
...
private final int MAX_DATA = 128;
private final int EXPECTED_BUFFER_DATA = 1024;
StringBuilder sb = new StringBuilder(EXPECTED_BUFFER_DATA);
...
String data = request.getParameter("foo");
if (data.length() < MAX_DATA) sb.append(data);
...
```

Issue Summary

Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Denial of Service: StringBuilder	1	0	0	1
Total	**1**	**0**	**0**	**1**

Denial of Service: StringBuilder	Low

Package: gov.va.med.pharmacy.jaxrs.tovista.dao.impl	

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/tovista/dao/impl/VistaInboundRpc.java, line 69 (Denial of Service: StringBuilder)	Low

Issue Details

Kingdom: Input Validation and Representation
Scan Engine: SCA (Data Flow)

Source Details

Source: java.io.BufferedReader.readLine()
From: gov.va.med.pharmacy.jaxrs.tovista.dao.impl.VistaInboundRpc.sendMessage
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/tovista/dao/impl/VistaInboundRpc.java:68

```
65   String line = null;
66   int i = 0;
67   try{
68   while(null != (line = br.readLine())) {
69   sb.append(line.trim());
70   LOG.debug("xml("+ i +")=" +line.trim());
71   i++;
```

Sink Details

Sink: java.lang.StringBuilder.append()
Enclosing Method: sendMessage()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/tovista/dao/impl/VistaInboundRpc.java:69
Taint Flags: NO_NEW_LINE, STREAM

```
66   int i = 0;
67   try{
68   while(null != (line = br.readLine())) {
```

Denial of Service: StringBuilder | Low

Package: gov.va.med.pharmacy.jaxrs.tovista.dao.impl

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/tovista/dao/impl/ VistaInboundRpc.java, line 69 (Denial of Service: StringBuilder) | **Low**

```
69    sb.append(line.trim());
70    LOG.debug("xml("+ i +")=" +line.trim());
71    i++;
72    }
```

Hidden Field (2 issues)

Abstract

A hidden form field is used.

Explanation

Programmers often trust the contents of hidden fields, expecting that users will not be able to view them or manipulate their contents. Attackers will violate these assumptions. They will examine the values written to hidden fields and alter them or replace the contents with attack data.

Example: An tag of type `hidden` indicates the use of a hidden field.
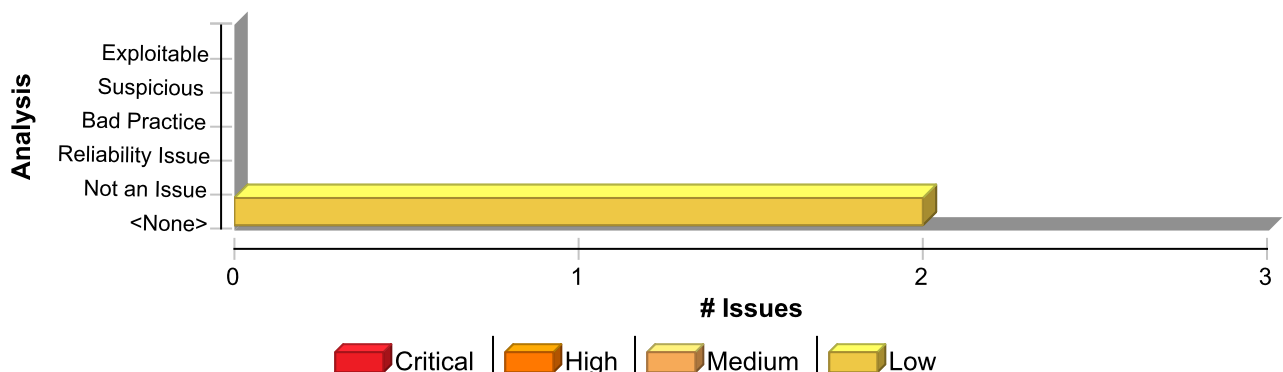
```
<input type="hidden">
```

If hidden fields carry sensitive information, this information will be cached the same way the rest of the page is cached. This can lead to sensitive information being tucked away in the browser cache without the user's knowledge.

Recommendation

Expect that attackers will study and decode all uses of hidden fields in the application. Treat hidden fields as untrusted input. Don't store information in hidden fields if the information should not be cached along with the rest of the page.

Issue Summary

Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Hidden Field	2	0	0	2
Total	**2**	**0**	**0**	**2**

Hidden Field	Low
Package: .src.main.webapp.WEB-INF	
PS_INB_ERX_UI/src/main/webapp/WEB-INF/manageusers.jsp, line 31 (Hidden Field)	**Low**
Issue Details	

Hidden Field		Low

PS_INB_ERX_UI/src/main/webapp/WEB-INF/manageusers.jsp, line 31 (Hidden Field)	Low

Kingdom: Encapsulation
Scan Engine: SCA (Content)

Sink Details

File: PS_INB_ERX_UI/src/main/webapp/WEB-INF/manageusers.jsp:31
Taint Flags:

```
28  <div id="successMessages" class="warning">${data.successMessage} </div>
29  </c:if>
30
31  <form:hidden path="modifiedIds" value="" />
32  <form:hidden path="modifiedFields" value="" />
33  <div id="userRolesGrid" style="display: block;width:88%; height: 200px" ></div>
34  <div class="manageUsersFields">
```

PS_INB_ERX_UI/src/main/webapp/WEB-INF/manageusers.jsp, line 32 (Hidden Field)	Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Content)

Sink Details

File: PS_INB_ERX_UI/src/main/webapp/WEB-INF/manageusers.jsp:32
Taint Flags:

```
29  </c:if>
30
31  <form:hidden path="modifiedIds" value="" />
32  <form:hidden path="modifiedFields" value="" />
33  <div id="userRolesGrid" style="display: block;width:88%; height: 200px" ></div>
34  <div class="manageUsersFields">
35  <br>
```

J2EE Misconfiguration: Excessive Session Timeout (2 issues)

Abstract

An overly long session timeout gives attackers more time to potentially compromise user accounts.

Explanation

The longer a session stays open, the larger the window of opportunity an attacker has to compromise user accounts. While a session remains active, an attacker may be able to brute-force a user's password, crack a user's wireless encryption key, or commandeer a session from an open browser. Longer session timeouts can also prevent memory from being released and eventually result in a denial of service if a sufficiently large number of sessions are created.

Example 1: If the session timeout is zero or less than zero, the session never expires. The following example shows a session timeout set to -1, which will cause the session to remain active indefinitely.

```
<session-config>
    <session-timeout>-1</session-timeout>
</session-config>
```

The tag defines the default session timeout interval for all sessions in the web application. If the tag is missing, it is left to the container to set the default timeout.
This category was derived from the Cigital Java Rulepack. http://www.cigital.com/
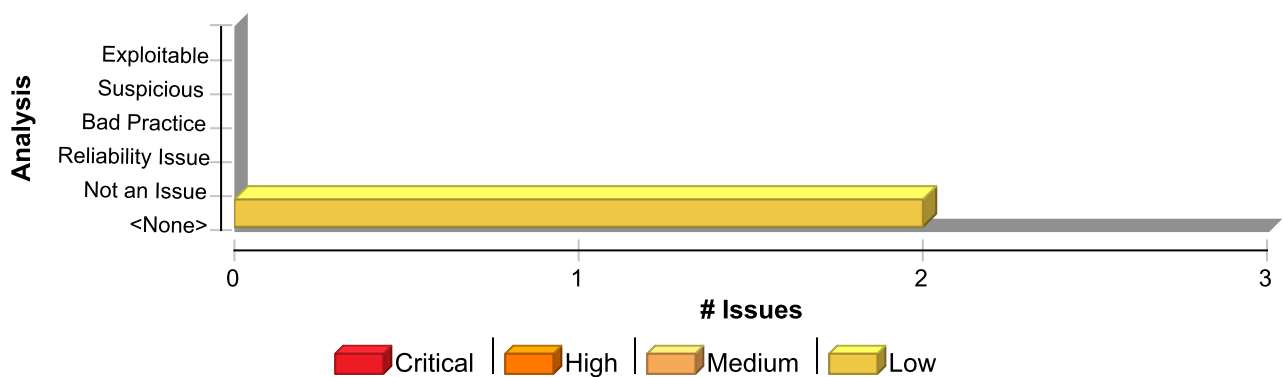
Recommendation

Set a session timeout that is 30 minutes or less, which both allows users to interact with the application over a period of time and provides a reasonable bound for the window of attack.

Example 2: The following example sets the session timeout to 20 minutes.

```
<session-config>
  <session-timeout>20</session-timeout>
</session-config>
```

Issue Summary

Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
J2EE Misconfiguration: Excessive Session Timeout	2	0	0	2
Total	**2**	**0**	**0**	**2**

J2EE Misconfiguration: Excessive Session Timeout	Low

Package: .src.main.webapp.WEB-INF	

PS_INB_ERX_JMS/src/main/webapp/WEB-INF/web.xml, line 5 (J2EE Misconfiguration: Excessive Session Timeout)	Low

Issue Details

Kingdom: Environment
Scan Engine: SCA (Configuration)

Sink Details

File: PS_INB_ERX_JMS/src/main/webapp/WEB-INF/web.xml:5
Taint Flags:

```
2   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3   xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
4   http://java.sun.com/xml/ns/javaee/web-app_3_0.xsd"
5   version="3.0">
6   <display-name>Inbound eRx JMS Receiver</display-name>
7
8   <listener>
```

PS_INB_ERX_WS/src/main/webapp/WEB-INF/web.xml, line 6 (J2EE Misconfiguration: Excessive Session Timeout)	Low

Issue Details

Kingdom: Environment
Scan Engine: SCA (Configuration)

Sink Details

File: PS_INB_ERX_WS/src/main/webapp/WEB-INF/web.xml:6
Taint Flags:

```
3   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4   xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
5   http://java.sun.com/xml/ns/javaee/web-app_3_0.xsd"
6   metadata-complete="true">
7
8   <display-name>Inbound eRx WS Application</display-name>
9   <description></description>
```

J2EE Misconfiguration: Incomplete Error Handling (1 issue)

Abstract

A web application must define a default error page for 500 errors to prevent attackers from mining information from the application container's built-in error response.

Explanation

When an attacker explores a web site looking for vulnerabilities, the amount of information that the site provides is crucial to the eventual success or failure of any attempted attacks. If the application shows the attacker a stack trace, it relinquishes information that makes the attacker's job significantly easier. For example, a stack trace might show the attacker a malformed SQL query string, the type of database being used, and the version of the application container. This information enables the attacker to target known vulnerabilities in these components.

The application configuration should specify a default error page in order to guarantee that the application will never leak error messages to an attacker. Handling standard HTTP error codes is useful and user-friendly in addition to being a good security practice, and a good configuration will also define a last-chance error handler that catches any exception that could possibly be thrown by the application.

Recommendation

A web application must be configured with a default error page. Your `web.xml` should include at least the following entries:

```
<error-page>
    <exception-type>java.lang.Throwable</exception-type>
<location>/error.jsp</location>
</error-page>
<error-page>
    <error-code>404</error-code>
<location>/error.jsp</location>
</error-page>
<error-page>
    <error-code>500</error-code>
<location>/error.jsp</location>
</error-page>
```

Issue Summary

Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
J2EE Misconfiguration: Incomplete Error Handling	1	0	0	1
Total	**1**	**0**	**0**	**1**

J2EE Misconfiguration: Incomplete Error Handling	Low
Package: .src.main.webapp.WEB-INF	

PS_INB_ERX_UI/src/main/webapp/WEB-INF/web.xml, line 88 (J2EE Misconfiguration: Incomplete Error Handling)	Low

Issue Details

Kingdom: Environment
Scan Engine: SCA (Configuration)

Sink Details

Sink: /web-app/error-page/error-code
File: PS_INB_ERX_UI/src/main/webapp/WEB-INF/web.xml:88
Taint Flags:

85	`</error-page>`
86	
87	`<error-page>`
88	`<error-code>403</error-code>`
89	`<location>/WEB-INF/accessDenied.jsp</location>`
90	`</error-page>`
91	

J2EE Misconfiguration: Missing Error Handling (2 issues)

Abstract

A web application must define default error pages in order to prevent attackers from mining information from the application container's built-in error response.

Explanation

When an attacker explores a web site looking for vulnerabilities, the amount of information that the site provides is crucial to the eventual success or failure of any attempted attacks. If the application shows the attacker a stack trace, it relinquishes information that makes the attacker's job significantly easier. For example, a stack trace might show the attacker a malformed SQL query string, the type of database being used, and the version of the application container. This information enables the attacker to target known vulnerabilities in these components.

The application configuration should specify a default error page in order to guarantee that the application will never leak error messages to an attacker. Handling standard HTTP error codes is useful and user-friendly in addition to being a good security practice, and a good configuration will also define a last-chance error handler that catches any exception that could possibly be thrown by the application.

Recommendation

A web application must be configured with a default error page. Your `web.xml` should include at least the following entries:

```
<error-page>
    <exception-type>java.lang.Throwable</exception-type>
<location>/error.jsp</location>
</error-page>
<error-page>
    <error-code>404</error-code>
<location>/error.jsp</location>
</error-page>
<error-page>
    <error-code>500</error-code>
<location>/error.jsp</location>
</error-page>
```

Issue Summary

Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
J2EE Misconfiguration: Missing Error Handling	2	0	0	2
Total	**2**	**0**	**0**	**2**

J2EE Misconfiguration: Missing Error Handling	Low

Package: .src.main.webapp.WEB-INF

PS_INB_ERX_JMS/src/main/webapp/WEB-INF/web.xml, line 5 (J2EE Misconfiguration: Missing Error Handling)	Low

Issue Details

Kingdom: Environment
Scan Engine: SCA (Configuration)

Sink Details

Sink: /web-app
File: PS_INB_ERX_JMS/src/main/webapp/WEB-INF/web.xml:5
Taint Flags:

```
2   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3   xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
4   http://java.sun.com/xml/ns/javaee/web-app_3_0.xsd"
5   version="3.0">
6   <display-name>Inbound eRx JMS Receiver</display-name>
7
8   <listener>
```

PS_INB_ERX_WS/src/main/webapp/WEB-INF/web.xml, line 6 (J2EE Misconfiguration: Missing Error Handling)	Low

Issue Details

Kingdom: Environment
Scan Engine: SCA (Configuration)

Sink Details

Sink: /web-app
File: PS_INB_ERX_WS/src/main/webapp/WEB-INF/web.xml:6
Taint Flags:

```
3   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4   xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
5   http://java.sun.com/xml/ns/javaee/web-app_3_0.xsd"
6   metadata-complete="true">
7
8   <display-name>Inbound eRx WS Application</display-name>
9   <description></description>
```

Missing Check against Null (6 issues)

Abstract

The program can dereference a null pointer because it does not check the return value of a function that might return null.

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 function call can never fail" and "it doesn't matter if this function call fails". When a programmer ignores the return value from a function, they implicitly state that they are operating under one of these assumptions.

Example 1: The following code does not check to see if the string returned by `getParameter()` is `null` before calling the member function `compareTo()`, potentially causing a null dereference.

```
String itemName = request.getParameter(ITEM_NAME);
   if (itemName.compareTo(IMPORTANT_ITEM)) {
      ...
    }
 ...
```

Example 2:. The following code shows a system property that is set to `null` and later dereferenced by a programmer who mistakenly assumes it will always be defined.

```
System.clearProperty("os.name");
...
String os = System.getProperty("os.name");
if (os.equalsIgnoreCase("Windows 95") )
 System.out.println("Not supported");
```

The traditional defense of this coding error is:

"I know the requested value will always exist because.... If it does not exist, the program cannot perform the desired behavior so it doesn't matter whether I handle the error or simply allow the program to die dereferencing a null value."

But attackers are skilled at finding unexpected paths through programs, particularly when exceptions are involved.

Recommendation

If a function can return an error code or any other evidence of its success or failure, always check for the error condition, even if there is no obvious way for it to occur. In addition to preventing security errors, many initially mysterious bugs have eventually led back to a failed method call with an unchecked return
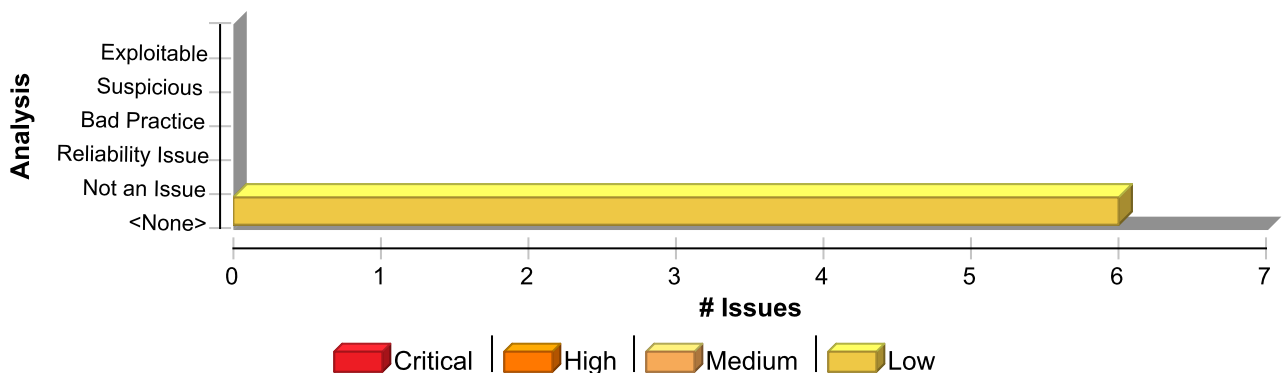
value.

Create an easy to use and standard way for dealing with failure in your application. If error handling is straightforward, programmers will be less inclined to omit it. One approach to standardized error handling is to write wrappers around commonly-used functions that check and handle error conditions without additional programmer intervention. When wrappers are implemented and adopted, the use of non-wrapped equivalents can be prohibited and enforced by using custom rules.

Example 3: The following code implements a wrapper around `getParameter()` that checks the return value of `getParameter()` against `null` and uses a default value if the requested parameter is not defined.

```
String safeGetParameter (HttpRequest request, String name)
{
    String value = request.getParameter(name);
    if (value == null) {
        return getDefaultValue(name)
    }
    return value;
}
```

Issue Summary

Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Missing Check against Null	6	0	0	6
Total	**6**	**0**	**0**	**6**

Missing Check against Null	Low
Package: gov.va.med.pharmacy.jaxrs.eAnde.services.impl	
PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/eAnde/services/impl/ EESummary_Client.java, line 50 (Missing Check against Null)	Low

Issue Details

> **Kingdom:** API Abuse
> **Scan Engine:** SCA (Control Flow)

Sink Details

Missing Check against Null	Low

Package: gov.va.med.pharmacy.jaxrs.eAnde.services.impl	
PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/eAnde/services/impl/ EESummary_Client.java, line 50 (Missing Check against Null)	Low

Sink: getClassLoader() : Class.getClassLoader may return NULL
Enclosing Method: returnResponse()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/eAnde/services/impl/EESummary_Client.java:50
Taint Flags:

```
47
48   InputStream inputStream = null;
49
50   inputStream =
this.getClass().getClassLoader().getResourceAsStream("gov.va.med.pharmacy.wsclients.properties")
51   try{
52   if(null!=inputStream){
53   properties.load(inputStream);
```

Package: gov.va.med.pharmacy.jaxrs.mvi.dao.impl	
PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/dao/impl/ MVIClient.java, line 733 (Missing Check against Null)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Control Flow)

Sink Details

Sink: classloader = getClassLoader() : Class.getClassLoader may return NULL
Enclosing Method: setupTLS()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/dao/impl/MVIClient.java:733
Taint Flags:

```
730   */
731   private void setupTLS(VAIdMPort port, Properties properties) throws
FileNotFoundException, IOException, GeneralSecurityException {
732
733   ClassLoader classloader = this.getClass().getClassLoader();
734
735   File file = new
File(classloader.getResource(properties.getProperty(KEYSTORE_FILE_NAME_PROPERTY).trim()).getFile
736   HTTPConduit httpConduit = (HTTPConduit) ClientProxy.getClient(port).getConduit();
```

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/dao/impl/ MVIClient.java, line 162 (Missing Check against Null)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Control Flow)

Sink Details

Missing Check against Null	Low

Package: gov.va.med.pharmacy.jaxrs.mvi.dao.impl	
PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/dao/impl/ MVIClient.java, line 162 (Missing Check against Null)	**Low**

Sink: getClassLoader() : Class.getClassLoader may return NULL
Enclosing Method: getMVIData()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/dao/impl/MVIClient.java:162
Taint Flags:

```
159
160   InputStream inputStream = null;
161   try {
162   inputStream = this.getClass().getClassLoader()
163   .getResourceAsStream("gov.va.med.pharmacy.wsclients.properties");
164   if (null != inputStream) {
165   properties.load(inputStream);
```

Package: gov.va.med.pharmacy.jaxrs.outboundncpdpmessage.service.impl	
PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/ outboundncpdpmessage/service/impl/ OutboundNCPDPMessageServiceImpl.java, line 90 (Missing Check against Null)	**Low**

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Control Flow)

Sink Details

Sink: getClassLoader() : Class.getClassLoader may return NULL
Enclosing Method: sendOutboundMessage()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/outboundncpdpmessage/service/impl/
OutboundNCPDPMessageServiceImpl.java:90
Taint Flags:

```
87
88   InputStream inputStream = null;
89   try {
90   inputStream =
this.getClass().getClassLoader().getResourceAsStream("gov.va.med.pharmacy.wsclients.properties")
91   if (null != inputStream) {
92   properties.load(inputStream);
93   }
```

Package: gov.va.med.pharmacy.utility	
PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/ VistaLinkConnectionUtility.java, line 68 (Missing Check against Null)	**Low**

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Control Flow)

Missing Check against Null	Low

Package: gov.va.med.pharmacy.utility	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/ VistaLinkConnectionUtility.java, line 68 (Missing Check against Null)	Low

Sink Details

Sink: getClassLoader() : Class.getClassLoader may return NULL
Enclosing Method: sendRequest()
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/VistaLinkConnectionUtility.java:68
Taint Flags:

```
65  InputStream inputStream = null;
66  try{
67
68  inputStream =
this.getClass().getClassLoader().getResourceAsStream("gov.va.med.pharmacy.vistalink.properties")
69
70  if(null!=inputStream){
71  properties.load(inputStream);
```

Package: gov.va.med.pharmacy.wsclients.eAnde	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/ ClientPasswordCallback.java, line 30 (Missing Check against Null)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Control Flow)

Sink Details

Sink: getClassLoader() : Class.getClassLoader may return NULL
Enclosing Method: handle()
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/ClientPasswordCallback.java:30
Taint Flags:

```
27
28  String password = null;
29
30  inputStream =
this.getClass().getClassLoader().getResourceAsStream("gov.va.med.pharmacy.wsclients.properties")
31  try{
32  if (null != inputStream) {
33  properties.load(inputStream);
```

Obsolete: Deprecated by ESAPI (52 issues)

Abstract

ESAPI offers a safer version of this method.

Explanation

The ESAPI secure coding guidelines contain a list of banned APIs for which a safer alternative is available in ESAPI.

The list of banned and substitute API's:
Banned 001 System.out.println()
Banned 002 Throwable.printStackTrace()
Banned 003 Runtime.exec()
Banned 004 Session.getId()
Banned 005 ServletRequest.getUserPrincipal()
Banned 006 ServletRequest.isUserInRole()
Banned 007 Session.invalidate()
Banned 008 Math.Random.*
Banned 009 File.createTempFile()
Banned 010 ServletResponse.setContentType()
Banned 011 ServletResponse.sendRedirect()
Banned 012 RequestDispatcher.forward()
Banned 013 ServletResponse.addHeader()
Banned 014 ServletResponse.addCookie()
Banned 015 ServletRequest.isSecure()
Banned 016 Properties.*
Banned 017 ServletContext.log()
Banned 018 java.security and javax.crypto
Banned 019 java.net.URLEncoder/Decoder
Banned 021 ServletResponse.encodeURL
Banned 022 ServletResponse.encodeRedirectURL
Banned 023 javax.servlet.ServletInputStream.readLine

Recommendation

Replace banned API calls with the recommended safer version provided by ESAPI.

The list of banned and replacement API's:

Banned 001 System.out.println()
ESAPI Replacement: Logger.*
Banned 002 Throwable.printStackTrace()
ESAPI Replacement: Logger.*
Banned 003 Runtime.exec()
ESAPI Replacement: Executor.safeExec()
Banned 004 Session.getId()
ESAPI Replacement: Randomizer.getRandomString (better not to use at all)
Banned 005 ServletRequest.getUserPrincipal()
ESAPI Replacement: Authenticator.getCurrentUser()
Banned 006 ServletRequest.isUserInRole()
ESAPI Replacement: AccessController.isAuthorized*()
Banned 007 Session.invalidate()

ESAPI Replacement: Authenticator.logout()
Banned 008 Math.Random.*
ESAPI Replacement: Randomizer.*
Banned 009 File.createTempFile()
ESAPI Replacement: Randomizer.getRandomFilename()
Banned 010 ServletResponse.setContentType()
ESAPI Replacement: HTTPUtilities.setContentType()
Banned 011 ServletResponse.sendRedirect()
ESAPI Replacement: HTTPUtilities.safeSendRedirect()
Banned 012 RequestDispatcher.forward()
ESAPI Replacement: HTTPUtilities.safeSendForward()
Banned 013 ServletResponse.addHeader()
ESAPI Replacement: HTTPUtilities.safeSetHeader()/safeSetHeader()
Banned 014 ServletResponse.addCookie()
ESAPI Replacement: HTTPUtilities.safeAddCookie()
Banned 015 ServletRequest.isSecure()
ESAPI Replacement: HTTPUtilties.isSecureChannel()
Banned 016 Properties.*
ESAPI Replacement: EncryptedProperties.*
Banned 017 ServletContext.log()
ESAPI Replacement: Logger.*
Banned 018 java.security and javax.crypto
ESAPI Replacement: Encryptor.*
Banned 019 java.net.URLEncoder/Decoder
ESAPI Replacement: Encoder.encodeForURL()/decodeForURL()
Banned 021 ServletResponse.encodeURL
ESAPI Replacement: HTTPUtilities.safeEncodeURL() (better not to use at all)
Banned 022 ServletResponse.encodeRedirectURL
ESAPI Replacement: HTTPUtilities.safeEncodeRedirectURL() (better not to use at all)
Banned 023 javax.servlet.ServletInputStream.readLine
ESAPI Replacement: Validator.safeReadLine()

Issue Summary

Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Obsolete: Deprecated by ESAPI	52	0	0	52
Total	**52**	**0**	**0**	**52**

Obsolete: Deprecated by ESAPI	Low

Package: gov.va.med.pharmacy.jaxrs.eAnde.services.impl

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/eAnde/services/impl/ EESummary_Client.java, line 61 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: getProperty()
Enclosing Method: returnResponse()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/eAnde/services/impl/EESummary_Client.java:61
Taint Flags:

```
58   StreamUtilities.safeClose(inputStream);
59   }
60   }
61   eAndeEndpoint = properties.getProperty(E_AND_E_CLIENT_ENDPOINT).trim();
62
63   username = properties.getProperty(E_AND_E_USERNAME).trim();
64
```

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/eAnde/services/impl/ EESummary_Client.java, line 42 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: Properties()
Enclosing Method: returnResponse()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/eAnde/services/impl/EESummary_Client.java:42
Taint Flags:

```
39
40   public getEESummaryResponse returnResponse(String key) throws java.lang.Exception {
41
42   Properties properties = new Properties();
43
44   String username = null;
45   String eAndeEndpoint = null;
```

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/eAnde/services/impl/ EESummary_Client.java, line 63 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Obsolete: Deprecated by ESAPI	Low

Package: gov.va.med.pharmacy.jaxrs.eAnde.services.impl

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/eAnde/services/impl/ EESummary_Client.java, line 63 (Obsolete: Deprecated by ESAPI)	Low

Sink Details

Sink: getProperty()
Enclosing Method: returnResponse()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/eAnde/services/impl/EESummary_Client.java:63
Taint Flags:

```
60  }
61  eAndeEndpoint = properties.getProperty(E_AND_E_CLIENT_ENDPOINT).trim();
62
63  username = properties.getProperty(E_AND_E_USERNAME).trim();
64
65  eAndeRequestName = properties.getProperty(E_AND_E_REQUEST_NAME).trim();
66
```

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/eAnde/services/impl/ EESummary_Client.java, line 53 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: load()
Enclosing Method: returnResponse()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/eAnde/services/impl/EESummary_Client.java:53
Taint Flags:

```
50  inputStream =
this.getClass().getClassLoader().getResourceAsStream("gov.va.med.pharmacy.wsclients.properties")
51  try{
52  if(null!=inputStream){
53  properties.load(inputStream);
54  }
55  }
56  finally{
```

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/eAnde/services/impl/ EESummary_Client.java, line 65 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: getProperty()
Enclosing Method: returnResponse()

Obsolete: Deprecated by ESAPI	Low

Package: gov.va.med.pharmacy.jaxrs.eAnde.services.impl

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/eAnde/services/impl/ EESummary_Client.java, line 65 (Obsolete: Deprecated by ESAPI)	Low

File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/eAnde/services/impl/EESummary_Client.java:65
Taint Flags:

```
62
63   username = properties.getProperty(E_AND_E_USERNAME).trim();
64
65   eAndeRequestName = properties.getProperty(E_AND_E_REQUEST_NAME).trim();
66
67   String url = "";
68
```

Package: gov.va.med.pharmacy.jaxrs.mvi.dao.impl

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/dao/impl/ MVIClient.java, line 165 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: load()
Enclosing Method: getMVIData()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/dao/impl/MVIClient.java:165
Taint Flags:

```
162   inputStream = this.getClass().getClassLoader()
163   .getResourceAsStream("gov.va.med.pharmacy.wsclients.properties");
164   if (null != inputStream) {
165   properties.load(inputStream);
166   }
167   } finally {
168   if (null != inputStream) {
```

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/dao/impl/ MVIClient.java, line 158 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: Properties()
Enclosing Method: getMVIData()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/dao/impl/MVIClient.java:158
Taint Flags:

Obsolete: Deprecated by ESAPI	Low

Package: gov.va.med.pharmacy.jaxrs.mvi.dao.impl	

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/dao/impl/ MVIClient.java, line 158 (Obsolete: Deprecated by ESAPI)	Low

```
155
156   // System.setProperty("javax.net.debug", "ssl, handshake");
157
158   Properties properties = new Properties();
159
160   InputStream inputStream = null;
161   try {
```

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/dao/impl/ MVIClient.java, line 745 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: getProperty()
Enclosing Method: setupTLS()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/dao/impl/MVIClient.java:745
Taint Flags:

```
742   FileInputStream fileIn1 = null;
743   FileInputStream fileIn2 = null;
744   TLSClientParameters tlsCP = new TLSClientParameters();
745   String keyPasswd = properties.getProperty(KEYSTORE_PASSWD_PROPERTY).trim();
746   KeyStore keyStore = KeyStore.getInstance(KEYSTORE_FILE_TYPE);
747
748   try {
```

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/dao/impl/ MVIClient.java, line 735 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: getProperty()
Enclosing Method: setupTLS()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/dao/impl/MVIClient.java:735
Taint Flags:

```
732
733   ClassLoader classloader = this.getClass().getClassLoader();
734
735   File file = new
```

Obsolete: Deprecated by ESAPI	Low

Package: gov.va.med.pharmacy.jaxrs.mvi.dao.impl

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/dao/impl/ MVIClient.java, line 735 (Obsolete: Deprecated by ESAPI)	Low

```
File(classloader.getResource(properties.getProperty(KEYSTORE_FILE_NAME_PROPERTY).trim()).getFile
736  HTTPConduit httpConduit = (HTTPConduit) ClientProxy.getClient(port).getConduit();
737
738  // We do two FileInpustreams otherwise it gives EOFException as the file
```

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/dao/impl/ MVIClient.java, line 192 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: getProperty()
Enclosing Method: getMVIData()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/dao/impl/MVIClient.java:192
Taint Flags:

```
189  VAIdMPort port = ss.getVAIdMPort();
190
191
192  mviClientEndpoint = properties.getProperty(MVI_CLIENT_ENDPOINT_PROPERTY).trim(); // trim
the value for any potential whitespace.
193  // Secure Dev
194  ((BindingProvider)
port).getRequestContext().put(BindingProvider.ENDPOINT_ADDRESS_PROPERTY, mviClientEndpoint);
195
```

Package: gov.va.med.pharmacy.jaxrs.outboundncpdpmessage.service.impl

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/ outboundncpdpmessage/service/impl/ OutboundNCPDPMessageServiceImpl.java, line 86 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: Properties()
Enclosing Method: sendOutboundMessage()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/outboundncpdpmessage/service/impl/ OutboundNCPDPMessageServiceImpl.java:86
Taint Flags:

```
83  //TODO - Asynchronous invocation with future......
```

Obsolete: Deprecated by ESAPI	Low

Package: gov.va.med.pharmacy.jaxrs.outboundncpdpmessage.service.impl	

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/ outboundncpdpmessage/service/impl/ OutboundNCPDPMessageServiceImpl.java, line 86 (Obsolete: Deprecated by ESAPI)	Low

```
84
85
86   Properties properties = new Properties();
87
88   InputStream inputStream = null;
89   try {
```

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/ outboundncpdpmessage/service/impl/ OutboundNCPDPMessageServiceImpl.java, line 92 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: load()
Enclosing Method: sendOutboundMessage()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/outboundncpdpmessage/service/impl/ OutboundNCPDPMessageServiceImpl.java:92
Taint Flags:

```
89   try {
90   inputStream =
this.getClass().getClassLoader().getResourceAsStream("gov.va.med.pharmacy.wsclients.properties")
91   if (null != inputStream) {
92   properties.load(inputStream);
93   }
94   } finally {
95   if (null != inputStream) {
```

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/ outboundncpdpmessage/service/impl/ OutboundNCPDPMessageServiceImpl.java, line 100 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: getProperty()
Enclosing Method: sendOutboundMessage()

Obsolete: Deprecated by ESAPI	Low

Package: gov.va.med.pharmacy.jaxrs.outboundncpdpmessage.service.impl

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/ outboundncpdpmessage/service/impl/ OutboundNCPDPMessageServiceImpl.java, line 100 (Obsolete: Deprecated by ESAPI)	Low

File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/outboundncpdpmessage/service/impl/
OutboundNCPDPMessageServiceImpl.java:100
Taint Flags:

```
97   }
98   }
99
100  String dasOutboundEndpoint = properties.getProperty("das.outbound.endpoint").trim();
101
102  WebClient client = WebClient.create(dasOutboundEndpoint);
103
```

Package: gov.va.med.pharmacy.jaxrs.provider.dao.impl

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/provider/dao/impl/ ProviderSearchDaoImpl.java, line 29 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: lookupProvider()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/provider/dao/impl/ProviderSearchDaoImpl.java:29
Taint Flags:

```
26
27   } catch (Exception e) {
28   LOG.error("Error in ProviderSearchDaoImpl:"+e.getMessage());
29   e.printStackTrace();
30   throw new Exception(e); // let the exception go all the way up.
31   }
32
```

Package: gov.va.med.pharmacy.jaxrs.tovista.dao.impl

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/tovista/dao/impl/ VistaInboundDaoImpl.java, line 33 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Obsolete: Deprecated by ESAPI	Low

Package: gov.va.med.pharmacy.jaxrs.tovista.dao.impl

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/tovista/dao/impl/ VistaInboundDaoImpl.java, line 33 (Obsolete: Deprecated by ESAPI)	Low

Sink: printStackTrace()
Enclosing Method: sendMessage()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/tovista/dao/impl/VistaInboundDaoImpl.java:33
Taint Flags:

```
30   sendResults = vistaInboundRpc.sendMessage(vistaInboundObj);
31
32   } catch (Exception e) {
33   e.printStackTrace();
34   throw new Exception(e); // let the exception go all the way up.
35   }
36
```

Package: gov.va.med.pharmacy.persistence.dao.impl

PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/dao/impl/ SummaryReportDaoImpl.java, line 48 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionCall: print
Enclosing Method: find()
File: PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/dao/impl/SummaryReportDaoImpl.java:48
Taint Flags:

```
45   @Override
46   public List<SummaryReportVw> find(SummaryReportFilter summaryReportFilter) {
47
48   System.out.print("DateFrom: " + summaryReportFilter.getDateFrom()
49   + " DateTo: " + summaryReportFilter.getDateTo() + "VISN: " + summaryReportFilter.getVisn()
50   + "Station ID: " + summaryReportFilter.getStationId());
51
```

Package: gov.va.med.pharmacy.persistence.service.impl

PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/service/ impl/UserServiceImpl.java, line 42 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()

Obsolete: Deprecated by ESAPI	Low

Package: gov.va.med.pharmacy.persistence.service.impl

PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/service/impl/UserServiceImpl.java, line 42 (Obsolete: Deprecated by ESAPI)	Low

Enclosing Method: addNewVAUser()
File: PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/service/impl/UserServiceImpl.java:42
Taint Flags:

```
39
40  } catch (Exception e) {
41  // TODO Auto-generated catch block
42  e.printStackTrace();
43  }
44  }
45
```

Package: gov.va.med.pharmacy.utility

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/VistaLinkConnectionUtility.java, line 63 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: Properties()
Enclosing Method: sendRequest()
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/VistaLinkConnectionUtility.java:63
Taint Flags:

```
60  try {
61
62
63  Properties properties = new Properties();
64
65  InputStream inputStream = null;
66  try{
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/VistaLinkConnectionUtility.java, line 71 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: load()
Enclosing Method: sendRequest()
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/VistaLinkConnectionUtility.java:71
Taint Flags:

Obsolete: Deprecated by ESAPI	Low

Package: gov.va.med.pharmacy.utility

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/ VistaLinkConnectionUtility.java, line 71 (Obsolete: Deprecated by ESAPI)	Low

```
68  inputStream =
this.getClass().getClassLoader().getResourceAsStream("gov.va.med.pharmacy.vistalink.properties")
69
70  if(null!=inputStream){
71  properties.load(inputStream);
72  proxyUser = properties.getProperty("vistalink.proxyuser").trim();
73  }
74
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/ StreamUtilities.java, line 25 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionCall: print
Enclosing Method: safeClose()
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/StreamUtilities.java:25
Taint Flags:

```
22  try {
23  inputStream.close();
24  } catch (IOException e) {
25  System.out.print(e);
26  }
27  }
28  }
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/ StreamUtilities.java, line 15 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionCall: print
Enclosing Method: safeClose()
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/StreamUtilities.java:15
Taint Flags:

```
12  try {
13  fileInputStream.close();
14  } catch (IOException e) {
```

Obsolete: Deprecated by ESAPI	Low

Package: gov.va.med.pharmacy.utility

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/ StreamUtilities.java, line 15 (Obsolete: Deprecated by ESAPI)	Low

```
15  System.out.print(e);
16  }
17  }
18  }
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/ VistaLinkConnectionUtility.java, line 110 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: sendRequest()
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/VistaLinkConnectionUtility.java:110
Taint Flags:

```
107  connection.close();
108  } catch (ResourceException e) {
109  //LOG.error("Error closing VistALink connection.", e);
110  e.printStackTrace();
111  }
112  }
113  }
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/ StreamUtilities.java, line 35 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionCall: print
Enclosing Method: safeClose()
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/StreamUtilities.java:35
Taint Flags:

```
32  try {
33  bufferedReader.close();
34  } catch (IOException e) {
35  System.out.print(e);
36  }
37  }
38  }
```

Obsolete: Deprecated by ESAPI	Low

Package: gov.va.med.pharmacy.utility

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/ StreamUtilities.java, line 35 (Obsolete: Deprecated by ESAPI)	Low

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/ VistaLinkConnectionUtility.java, line 72 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: getProperty()
Enclosing Method: sendRequest()
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/VistaLinkConnectionUtility.java:72
Taint Flags:

```
69
70   if(null!=inputStream){
71   properties.load(inputStream);
72   proxyUser = properties.getProperty("vistalink.proxyuser").trim();
73   }
74
75   } finally {
```

Package: gov.va.med.pharmacy.web.administration

PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/administration/ PharmacyManagementController.java, line 601 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: addNewPharmacy()
File: PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/administration/PharmacyManagementController.java:601
Taint Flags:

```
598  }
599  catch (Exception e) {
600
601  e.printStackTrace();
602
603  errorsList.add("Add Failed. Unable to save record. Please contact production support.");
604
```

Obsolete: Deprecated by ESAPI	Low

Package: gov.va.med.pharmacy.web.administration	

PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/administration/ PharmacyManagementController.java, line 566 (Obsolete: Deprecated by ESAPI)	**Low**

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: addNewPharmacy()
File: PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/administration/PharmacyManagementController.java:566
Taint Flags:

```
563
564  } catch (HibernateException e) {
565
566  e.printStackTrace();
567
568  Throwable cause = e.getCause();
569
```

PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/administration/ PharmacyManagementController.java, line 266 (Obsolete: Deprecated by ESAPI)	**Low**

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: updatePharmacy()
File: PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/administration/PharmacyManagementController.java:266
Taint Flags:

```
263  }
264  catch (Exception e) {
265
266  e.printStackTrace();
267
268  errorsList.add("Update Failed. Unable to save record. Please contact production
support.");
269
```

PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/administration/ PharmacyManagementController.java, line 233 (Obsolete: Deprecated by ESAPI)	**Low**

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Obsolete: Deprecated by ESAPI	Low

Package: gov.va.med.pharmacy.web.administration

PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/administration/ PharmacyManagementController.java, line 233 (Obsolete: Deprecated by ESAPI)	Low

Sink Details

Sink: printStackTrace()
Enclosing Method: updatePharmacy()
File: PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/administration/PharmacyManagementController.java:233
Taint Flags:

```
230
231   } catch (HibernateException e) {
232
233   e.printStackTrace();
234
235   Throwable cause = e.getCause();
236
```

Package: gov.va.med.pharmacy.web.auth

PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/auth/ LogoutController.java, line 24 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: invalidate()
Enclosing Method: logout()
File: PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/auth/LogoutController.java:24
Taint Flags:

```
21
22   session.removeAttribute(Constants.SESSION_USER);
23
24   session.invalidate();
25
26   String logoutMessage;
27
```

PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/auth/ LoginController.java, line 115 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()

Obsolete: Deprecated by ESAPI	Low

Package: gov.va.med.pharmacy.web.auth

PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/auth/ LoginController.java, line 115 (Obsolete: Deprecated by ESAPI)	Low

Enclosing Method: authenticateUser()
File: PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/auth/LoginController.java:115
Taint Flags:

```
112   e.printStackTrace();
113   } catch (Exception e) {
114   // TODO Auto-generated catch block
115   e.printStackTrace();
116   }
117   session.removeAttribute("loginErrorMessage");
118
```

PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/auth/ LoginController.java, line 112 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: authenticateUser()
File: PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/auth/LoginController.java:112
Taint Flags:

```
109   session.setAttribute(Constants.SESSION_USER, user);
110   } catch (IntrusionException e) {
111   // TODO Auto-generated catch block
112   e.printStackTrace();
113   } catch (Exception e) {
114   // TODO Auto-generated catch block
115   e.printStackTrace();
```

PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/auth/ LogoutController.java, line 65 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: validateInput()
File: PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/auth/LogoutController.java:65
Taint Flags:

```
62
```

Obsolete: Deprecated by ESAPI	Low

Package: gov.va.med.pharmacy.web.auth

PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/auth/ LogoutController.java, line 65 (Obsolete: Deprecated by ESAPI)	Low

```
63   } catch (Exception e) {
64
65     e.printStackTrace();
66
67   throw new RuntimeException("Invalid logoutURL.");
68   }
```

Package: gov.va.med.pharmacy.web.interceptor

PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/interceptor/ LoginInterceptor.java, line 56 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: sendRedirect()
Enclosing Method: preHandle()
File: PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/interceptor/LoginInterceptor.java:56
Taint Flags:

```
53   }
54   */
55   // logger.debug("no user redirecting to welcome page");
56   response.sendRedirect("/inbound/public/landing");
57   return false;
58   }
59
```

Package: gov.va.med.pharmacy.web.reports

PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/reports/ SummaryReportController.java, line 167 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: csvExport()
File: PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/reports/SummaryReportController.java:167
Taint Flags:

```
164
165   } catch (Exception e) {
```

Obsolete: Deprecated by ESAPI	Low

Package: gov.va.med.pharmacy.web.reports

PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/reports/ SummaryReportController.java, line 167 (Obsolete: Deprecated by ESAPI)	Low

```
166   // TODO Auto-generated catch block
167   e.printStackTrace();
168   }
169
170   }
```

PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/reports/ SummaryReportController.java, line 144 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionCall: setContentType
Enclosing Method: csvExport()
File: PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/reports/SummaryReportController.java:144
Taint Flags:

```
141
142   }
143
144   response.setContentType("text/csv");
145
146   response.setHeader(responseHeaderKey, responseHeaderValue);
147
```

Package: gov.va.med.pharmacy.web.track

PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/track/ TrackRxController.java, line 147 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionCall: println
Enclosing Method: trackAuditSearch()
File: PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/track/TrackRxController.java:147
Taint Flags:

```
144   messageStatus = node.get("messageStatus").asText();}
145   if (node.get("inboundOutbound") != null){
146   inboundOutbound = node.get("inboundOutbound").asText();
147   System.out.println("inboundOutbound is:" + inboundOutbound);
148   }
```

Obsolete: Deprecated by ESAPI	Low

Package: gov.va.med.pharmacy.web.track

PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/track/ TrackRxController.java, line 147 (Obsolete: Deprecated by ESAPI)	Low

```
149
150
```

PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/track/ TrackRxController.java, line 199 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: csvExport()
File: PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/track/TrackRxController.java:199
Taint Flags:

```
196
197  } catch (Exception e) {
198  // TODO Auto-generated catch block
199  e.printStackTrace();
200  }
201
202  }
```

PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/track/ TrackRxController.java, line 169 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Structural)

Sink Details

Sink: FunctionCall: setContentType
Enclosing Method: csvExport()
File: PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/track/TrackRxController.java:169
Taint Flags:

```
166
167  String responseHeaderValue = String.format("attachment; filename=\"%s\"", csvFileName);
168
169  response.setContentType("text/csv");
170
171  response.setHeader(responseHeaderKey, responseHeaderValue);
172
```

Obsolete: Deprecated by ESAPI	Low

Package: gov.va.med.pharmacy.wsclients.eAnde

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/ EeSummaryPortSoap11Impl.java, line 135 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: getEESummaryHistory()
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/EeSummaryPortSoap11Impl.java:135
Taint Flags:

```
132   gov.va.med.pharmacy.wsclients.eAnde.GetEESummaryHistoryResponse _return = null;
133   return _return;
134   } catch (java.lang.Exception ex) {
135   ex.printStackTrace();
136   throw new RuntimeException(ex);
137   }
138   }
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/ EeSummaryPortSoap11Impl.java, line 60 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: getDeliveryPreference()
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/EeSummaryPortSoap11Impl.java:60
Taint Flags:

```
57   gov.va.med.pharmacy.wsclients.eAnde.GetDeliveryPreferenceResponse _return = null;
58   return _return;
59   } catch (java.lang.Exception ex) {
60   ex.printStackTrace();
61   throw new RuntimeException(ex);
62   }
63   }
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/ EeSummaryPortSoap11Impl.java, line 180 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Obsolete: Deprecated by ESAPI	Low

Package: gov.va.med.pharmacy.wsclients.eAnde

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/ EeSummaryPortSoap11Impl.java, line 180 (Obsolete: Deprecated by ESAPI)	Low

Sink Details

Sink: printStackTrace()
Enclosing Method: getEESummary()
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/EeSummaryPortSoap11Impl.java:180
Taint Flags:

```
177  gov.va.med.pharmacy.wsclients.eAnde.getEESummaryResponse _return = null;
178  return _return;
179  } catch (java.lang.Exception ex) {
180  ex.printStackTrace();
181  throw new RuntimeException(ex);
182  }
183  }
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/ EeSummaryPortSoap11Impl.java, line 75 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: retrieveIVMCandidates()
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/EeSummaryPortSoap11Impl.java:75
Taint Flags:

```
72  gov.va.med.pharmacy.wsclients.eAnde.RetrieveIVMCandidatesResponse _return = null;
73  return _return;
74  } catch (java.lang.Exception ex) {
75  ex.printStackTrace();
76  throw new RuntimeException(ex);
77  }
78  }
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/ ClientPasswordCallback.java, line 22 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: Properties()
Enclosing Method: handle()
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/ClientPasswordCallback.java:22

Obsolete: Deprecated by ESAPI	Low

Package: gov.va.med.pharmacy.wsclients.eAnde	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/ ClientPasswordCallback.java, line 22 (Obsolete: Deprecated by ESAPI)	Low

Taint Flags:

```
19
20   public void handle(Callback[] callbacks) throws IOException, UnsupportedCallbackException {
21
22   Properties properties = new Properties();
23
24   InputStream inputStream = null;
25
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/ EeSummaryPortSoap11Impl.java, line 165 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: getEligibilityDetermination()
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/EeSummaryPortSoap11Impl.java:165
Taint Flags:

```
162   gov.va.med.pharmacy.wsclients.eAnde.GetEligibilityDeterminationResponse _return = null;
163   return _return;
164   } catch (java.lang.Exception ex) {
165   ex.printStackTrace();
166   throw new RuntimeException(ex);
167   }
168   }
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/ EeSummaryPortSoap11Impl.java, line 150 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: updateDeliveryPreference()
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/EeSummaryPortSoap11Impl.java:150
Taint Flags:

```
147   gov.va.med.pharmacy.wsclients.eAnde.UpdateDeliveryPreferenceResponse _return = null;
148   return _return;
```

Obsolete: Deprecated by ESAPI	Low

Package: gov.va.med.pharmacy.wsclients.eAnde

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/ EeSummaryPortSoap11Impl.java, line 150 (Obsolete: Deprecated by ESAPI)	Low

```
149   } catch (java.lang.Exception ex) {
150   ex.printStackTrace();
151   throw new RuntimeException(ex);
152   }
153   }
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/ EeSummaryPortSoap11Impl.java, line 90 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: getPersonNotification()
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/EeSummaryPortSoap11Impl.java:90
Taint Flags:

```
87    gov.va.med.pharmacy.wsclients.eAnde.GetPersonNotificationResponse _return = null;
88    return _return;
89    } catch (java.lang.Exception ex) {
90    ex.printStackTrace();
91    throw new RuntimeException(ex);
92    }
93    }
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/ EeSummaryPortSoap11Impl.java, line 105 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: getIVMLetterStatuses()
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/EeSummaryPortSoap11Impl.java:105
Taint Flags:

```
102   gov.va.med.pharmacy.wsclients.eAnde.GetIVMLetterStatusesResponse _return = null;
103   return _return;
104   } catch (java.lang.Exception ex) {
105   ex.printStackTrace();
106   throw new RuntimeException(ex);
107   }
```

Obsolete: Deprecated by ESAPI	Low

Package: gov.va.med.pharmacy.wsclients.eAnde	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/EeSummaryPortSoap11Impl.java, line 105 (Obsolete: Deprecated by ESAPI)	Low

```
108   }
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/ClientPasswordCallback.java, line 33 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: load()
Enclosing Method: handle()
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/ClientPasswordCallback.java:33
Taint Flags:

```
30   inputStream =
this.getClass().getClassLoader().getResourceAsStream("gov.va.med.pharmacy.wsclients.properties")
31   try{
32   if (null != inputStream) {
33   properties.load(inputStream);
34   }
35   }
36   finally{
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/ClientPasswordCallback.java, line 42 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: getProperty()
Enclosing Method: handle()
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/ClientPasswordCallback.java:42
Taint Flags:

```
39   }
40   }
41   username = properties.getProperty(E_AND_E_USERNAME).trim();
42   password = properties.getProperty(E_AND_E_PASSWD).trim();
43
44   for (int i = 0; i < callbacks.length; i++) {
45   WSPasswordCallback pc = (WSPasswordCallback) callbacks[i];
```

Obsolete: Deprecated by ESAPI	Low

Package: gov.va.med.pharmacy.wsclients.eAnde

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/ EeSummaryPortSoap11Impl.java, line 120 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: sendIVMUpdates()
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/EeSummaryPortSoap11Impl.java:120
Taint Flags:

```
117  gov.va.med.pharmacy.wsclients.eAnde.SendIVMUpdatesResponse _return = null;
118  return _return;
119  } catch (java.lang.Exception ex) {
120  ex.printStackTrace();
121  throw new RuntimeException(ex);
122  }
123  }
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/ EeSummaryPortSoap11Impl.java, line 45 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Sink Details

Sink: printStackTrace()
Enclosing Method: updateIVMRecordStatus()
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/EeSummaryPortSoap11Impl.java:45
Taint Flags:

```
42  gov.va.med.pharmacy.wsclients.eAnde.UpdateIVMRecordStatusResponse _return = null;
43  return _return;
44  } catch (java.lang.Exception ex) {
45  ex.printStackTrace();
46  throw new RuntimeException(ex);
47  }
48  }
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/ ClientPasswordCallback.java, line 41 (Obsolete: Deprecated by ESAPI)	Low

Issue Details

Kingdom: API Abuse
Scan Engine: SCA (Semantic)

Obsolete: Deprecated by ESAPI	Low

Package: gov.va.med.pharmacy.wsclients.eAnde

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/ ClientPasswordCallback.java, line 41 (Obsolete: Deprecated by ESAPI)	Low

Sink Details

Sink: getProperty()
Enclosing Method: handle()
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/ClientPasswordCallback.java:41
Taint Flags:

```
38   StreamUtilities.safeClose(inputStream);
39   }
40   }
41   username = properties.getProperty(E_AND_E_USERNAME).trim();
42   password = properties.getProperty(E_AND_E_PASSWD).trim();
43
44   for (int i = 0; i < callbacks.length; i++) {
```

Password Management (2 issues)

Abstract

Storing a password in plaintext can result in a system compromise.

Explanation

Password management issues occur when a password is stored in plaintext in an application's properties or configuration file.

Example 1: The following code reads a password from a properties file and uses the password to connect to a database.

```
...
Properties prop = new Properties();
prop.load(new FileInputStream("config.properties"));
String password = prop.getProperty("password");

DriverManager.getConnection(url, usr, password);
...
```

This code will run successfully, but anyone who has access to config.properties can read the value of `password`. If a devious employee has access to this information, they can use it to break into the system.

In the mobile world, password management is even trickier, considering a much higher chance of device loss.

Example 2: The code below reads username and password from an Android WebView store 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[] credentials = view.getHttpAuthUsernamePassword(host, realm);
    String username = credentials[0];
    String password = credentials[1];
    handler.proceed(username, password);
  }
});
...
```

By default, WebView credentials are stored in plaintext and are not hashed. So if a user has a rooted device (or uses an emulator), she is able to read stored passwords for given sites.

Recommendation

A password should never be stored in plaintext. Instead, the password should be entered by an administrator when the system starts. If that approach is impractical, a less secure but often adequate solution is to obfuscate the password and scatter the de-obfuscation material around the system so that an attacker has to obtain and correctly combine multiple system resources to decipher the password. At the very least, passwords should be hashed before being stored.

Some third-party products claim the ability to manage passwords in a more secure way. For example, WebSphere Application Server 4.x uses a simple XOR encryption algorithm for obfuscating values, but be skeptical about such facilities. WebSphere and other application servers offer outdated and relatively weak encryption mechanisms that are insufficient for security-sensitive environments. For a secure generic solution, the best option today appears to be a proprietary mechanism that you create.

For Android, as well as any other platform that uses SQLite database, a good option is SQLCipher -- an extension to SQLite database that provides transparent 256-bit AES encryption of database files. Thus, credentials can be stored in an encrypted database.
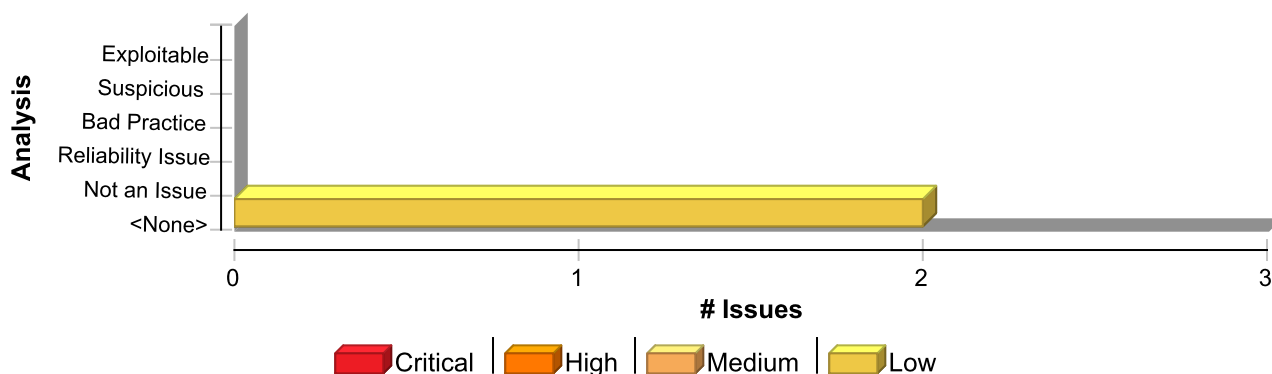
Example 3: The code below demonstrates how to integrate SQLCipher into an Android application after downloading the necessary binaries, and store credentials into the database file.

```
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	2	0	0	2
Total	**2**	**0**	**0**	**2**

Password Management	Low

Package: gov.va.med.pharmacy.jaxrs.mvi.dao.impl	

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/dao/impl/ MVIClient.java, line 762 (Password Management)	Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Data Flow)

Source Details

Source: java.util.Properties.load()
From: gov.va.med.pharmacy.jaxrs.mvi.dao.impl.MVIClient.getMVIData
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/dao/impl/MVIClient.java:165

```
162  inputStream = this.getClass().getClassLoader()
163  .getResourceAsStream("gov.va.med.pharmacy.wsclients.properties");
164  if (null != inputStream) {
165  properties.load(inputStream);
166  }
167  } finally {
168  if (null != inputStream) {
```

Sink Details

Sink: java.security.KeyStore.load()
Enclosing Method: setupTLS()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/dao/impl/MVIClient.java:762
Taint Flags: PROPERTY

```
759  try {
760  fileIn2 = new FileInputStream(file);
761  KeyStore trustStore = KeyStore.getInstance(KEYSTORE_FILE_TYPE);
762  trustStore.load(fileIn2, keyPasswd.toCharArray());
763  TrustManager[] myTrustStoreKeyManagers = getTrustManagers(trustStore);
764  tlsCP.setTrustManagers(myTrustStoreKeyManagers);
765  httpConduit.setTlsClientParameters(tlsCP);
```

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/dao/impl/ MVIClient.java, line 750 (Password Management)	Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Data Flow)

Source Details

Source: java.util.Properties.load()
From: gov.va.med.pharmacy.jaxrs.mvi.dao.impl.MVIClient.getMVIData
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/dao/impl/MVIClient.java:165

Password Management	Low
Package: gov.va.med.pharmacy.jaxrs.mvi.dao.impl	
PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/dao/impl/ MVIClient.java, line 750 (Password Management)	Low

```
162    inputStream = this.getClass().getClassLoader()
163    .getResourceAsStream("gov.va.med.pharmacy.wsclients.properties");
164    if (null != inputStream) {
165    properties.load(inputStream);
166    }
167    } finally {
168    if (null != inputStream) {
```

Sink Details

Sink: java.security.KeyStore.load()
Enclosing Method: setupTLS()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/dao/impl/MVIClient.java:750
Taint Flags: PROPERTY

```
747
748    try {
749    fileIn1 = new FileInputStream(file);
750    keyStore.load(fileIn1, keyPasswd.toCharArray());
751    KeyManager[] myKeyManagers = getKeyManagers(keyStore, keyPasswd);
752    tlsCP.setKeyManagers(myKeyManagers);
753    } finally {
```

Password Management: Hardcoded Password (2 issues)

Abstract

Hardcoded passwords may compromise system security in a way that cannot be easily remedied.

Explanation

It is never a good idea to hardcode a password. Not only does hardcoding a password allow all of the project's developers to view the password, it also makes fixing the problem extremely difficult. Once the code is in production, the password cannot be changed without patching the software. If the account protected by the password is compromised, the owners of the system will be forced to choose between security and availability.

Example 1: The following code uses a hardcoded password to connect to a database:

```
...
DriverManager.getConnection(url, "scott", "tiger");
...
```

This code will run successfully, but anyone who has access to it will have access to the password. Once 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 could use it to break into the system. Even worse, if attackers have access to the bytecode for the application they can use the `javap -c` command to access the disassembled code, which will contain the values of the passwords used. The result of this operation might look something like the following for the example above:

```
javap -c ConnMngr.class

22: ldc    #36; //String jdbc:mysql://ixne.com/rxsql
24: ldc    #38; //String scott
26: ldc    #17; //String tiger
```

In the mobile world, password management is even trickier, considering a much higher chance of device loss.

Example 2: The code below uses hardcoded username and password to setup authentication for viewing protected pages with Android's WebView.

```
...
webview.setWebViewClient(new WebViewClient() {
  public void onReceivedHttpAuthRequest(WebView view,
        HttpAuthHandler handler, String host, String realm) {
    handler.proceed("guest", "allow");
  }
});
...
```

Similar to Example 1, this code will run successfully, but anyone who has access to it will have access to the password.

Recommendation

Passwords should never be hardcoded and should generally be obfuscated and managed in an external source. Storing passwords in plaintext anywhere on the system allows anyone with sufficient permissions to read and potentially misuse the password. At the very least, passwords should be hashed before being stored.

Some third-party products claim the ability to manage passwords in a more secure way. For example, WebSphere Application Server 4.x uses a simple XOR encryption algorithm for obfuscating values, but be skeptical about such facilities. WebSphere and other application servers offer outdated and relatively weak encryption mechanisms that are insufficient for security-sensitive environments. For a secure generic solution, the best option today appears to be a proprietary mechanism that you create.

For Android, as well as any other platform that uses SQLite database, a good option is SQLCipher -- an extension to 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: Hardcoded Password	2	0	0	2
Total	**2**	**0**	**0**	**2**

Password Management: Hardcoded Password	High

Package: gov.va.med.pharmacy.jaxrs.mvi.dao.impl	

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/dao/impl/ MVIClient.java, line 98 (Password Management: Hardcoded Password)	**High**

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Audit Details

Analysis   Not an Issue

Audit Comments

VHAHAMWandlJ: Sun Dec 18 2016 17:51:45 GMT-0500 (EST)
The references are not hardcoded passwords, but System Property names that refer to passwords.

Sink Details

Sink: FieldAccess: KEYSTORE_PASSWD_PROPERTY
Enclosing Method: ()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/dao/impl/MVIClient.java:98
Taint Flags:

```
95   private static final String MVI_CLIENT_ENDPOINT_PROPERTY = "mvi.client.endpoint";
96   private static final String KEYSTORE_FILE_TYPE = "JKS";
97   private static final String KEYSTORE_FILE_NAME_PROPERTY = "keystore.filename";
98   private static final String KEYSTORE_PASSWD_PROPERTY = "keystore.password";
99   private static final String LEGAL_NAME = "Legal Name";
100  private static final String L_CHARACTER = "L";
101  private static final String EMPTY_STRING = "";
```

Package: gov.va.med.pharmacy.wsclients.eAnde	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/ ClientPasswordCallback.java, line 17 (Password Management: Hardcoded Password)	**High**

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Audit Details

Analysis   Not an Issue

Audit Comments

VHAHAMWandlJ: Sun Dec 18 2016 17:51:45 GMT-0500 (EST)
The references are not hardcoded passwords, but System Property names that refer to passwords.

Sink Details

Password Management: Hardcoded Password	High
Package: gov.va.med.pharmacy.wsclients.eAnde	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/ ClientPasswordCallback.java, line 17 (Password Management: Hardcoded Password)	**High**

Sink: FieldAccess: E_AND_E_PASSWD
Enclosing Method: ()
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/ClientPasswordCallback.java:17
Taint Flags:

```
14
15 public class ClientPasswordCallback implements CallbackHandler {
16
17   private static final String E_AND_E_PASSWD = "eAnde.password";
18   private static final String E_AND_E_USERNAME = "eAnde.username";
19
20   public void handle(Callback[] callbacks) throws IOException, UnsupportedCallbackException {
```

Password Management: Password in Comment (18 issues)

Abstract

Storing passwords or password details in plaintext 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. Once 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 will be forced to 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. Once 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 could 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 plaintext 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	18	0	0	18
Total	**18**	**0**	**0**	**18**

Password Management: Password in Comment	Low

Package: .src.main.java.gov.va.med.pharmacy.ncpdpmsgs	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/ PasswordType.java, line 19 (Password Management: Password in Comment)	Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/PasswordType.java:19
Taint Flags:

```
16  import javax.xml.bind.annotation.XmlValue;
17
18
19  /**
20   * <p>Java class for PasswordType complex type.
21   *
22   * <p>The following schema fragment specifies the expected content contained within this
class.
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/ PasswordRequestType.java, line 61 (Password Management: Password in Comment)	Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/PasswordRequestType.java:61
Taint Flags:

```
58  return oldPassword;
59  }
60
61  /**
62   * Sets the value of the oldPassword property.
63   *
64   * @param value
```

Password Management: Password in Comment	Low

Package: .src.main.java.gov.va.med.pharmacy.ncpdpmsgs	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/ UsernameTokenType.java, line 96 (Password Management: Password in Comment)	Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/UsernameTokenType.java:96
Taint Flags:

```
93   return password;
94   }
95
96   /**
97   * Sets the value of the password property.
98   *
99   * @param value
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/ PasswordChange.java, line 59 (Password Management: Password in Comment)	Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/PasswordChange.java:59
Taint Flags:

```
56   return request;
57   }
58
59   /**
60   * Sets the value of the request property.
61   *
62   * @param value
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/ PasswordRequestType.java, line 49 (Password Management: Password in Comment)	Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Password Management: Password in Comment	Low

Package: .src.main.java.gov.va.med.pharmacy.ncpdpmsgs	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/ PasswordRequestType.java, line 49 (Password Management: Password in Comment)	Low

Sink Details

Sink: Comment
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/PasswordRequestType.java:49
Taint Flags:

```
46   @XmlElement(name = "NewPassword", required = true)
47   protected Object newPassword;
48
49   /**
50    * Gets the value of the oldPassword property.
51    *
52    * @return
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/ ObjectFactory.java, line 828 (Password Management: Password in Comment)	Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/ObjectFactory.java:828
Taint Flags:

```
825   return new SecurityIdentificationType();
826   }
827
828   /**
829    * Create an instance of {@link PasswordType }
830    *
831    */
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/ BodyType.java, line 17 (Password Management: Password in Comment)	Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/BodyType.java:17
Taint Flags:

Password Management: Password in Comment	Low

Package: .src.main.java.gov.va.med.pharmacy.ncpdpmsgs	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/ BodyType.java, line 17 (Password Management: Password in Comment)	Low

```
14  import javax.xml.bind.annotation.XmlType;
15
16
17  /**
18   * <p>Java class for BodyType complex type.
19   *
20   * <p>The following schema fragment specifies the expected content contained within this
class.
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/ ObjectFactory.java, line 524 (Password Management: Password in Comment)	Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/ObjectFactory.java:524
Taint Flags:

```
521  return new PasswordChange();
522  }
523
524  /**
525   * Create an instance of {@link PasswordRequestType }
526   *
527   */
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/ UsernameTokenType.java, line 84 (Password Management: Password in Comment)	Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/UsernameTokenType.java:84
Taint Flags:

```
81  this.username = value;
82  }
83
84  /**
```

Password Management: Password in Comment	Low

Package: .src.main.java.gov.va.med.pharmacy.ncpdpmsgs	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/ UsernameTokenType.java, line 84 (Password Management: Password in Comment)	Low

```
85   * Gets the value of the password property.
86   *
87   * @return
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/ BodyType.java, line 349 (Password Management: Password in Comment)	Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/BodyType.java:349
Taint Flags:

```
346   this.getMessage = value;
347   }
348
349   /**
350   * Gets the value of the passwordChange property.
351   *
352   * @return
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/ ObjectFactory.java, line 516 (Password Management: Password in Comment)	Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/ObjectFactory.java:516
Taint Flags:

```
513   return new GetMessage();
514   }
515
516   /**
517   * Create an instance of {@link PasswordChange }
518   *
519   */
```

Password Management: Password in Comment	Low

Package: .src.main.java.gov.va.med.pharmacy.ncpdpmsgs	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/ PasswordRequestType.java, line 17 (Password Management: Password in Comment)	Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/PasswordRequestType.java:17
Taint Flags:

```
14  import javax.xml.bind.annotation.XmlType;
15
16
17  /**
18   * <p>Java class for PasswordRequestType complex type.
19   *
20   * <p>The following schema fragment specifies the expected content contained within this
     class.
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/ PasswordRequestType.java, line 85 (Password Management: Password in Comment)	Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/PasswordRequestType.java:85
Taint Flags:

```
82   return newPassword;
83   }
84
85   /**
86    * Sets the value of the newPassword property.
87    *
88    * @param value
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/ PasswordChange.java, line 18 (Password Management: Password in Comment)	Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Password Management: Password in Comment	Low

Package: .src.main.java.gov.va.med.pharmacy.ncpdpmsgs	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/ PasswordChange.java, line 18 (Password Management: Password in Comment)	Low

Sink Details

Sink: Comment
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/PasswordChange.java:18
Taint Flags:

```
15  import javax.xml.bind.annotation.XmlType;
16
17
18  /**
19   * <p>Java class for anonymous complex type.
20   *
21   * <p>The following schema fragment specifies the expected content contained within this
class.
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/ UsernameTokenType.java, line 19 (Password Management: Password in Comment)	Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/UsernameTokenType.java:19
Taint Flags:

```
16  import javax.xml.datatype.XMLGregorianCalendar;
17
18
19  /**
20   * <p>Java class for UsernameTokenType complex type.
21   *
22   * <p>The following schema fragment specifies the expected content contained within this
class.
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/ BodyType.java, line 361 (Password Management: Password in Comment)	Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/BodyType.java:361

Password Management: Password in Comment	Low
Package: .src.main.java.gov.va.med.pharmacy.ncpdpmsgs	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/ BodyType.java, line 361 (Password Management: Password in Comment)	Low

Taint Flags:

358	return passwordChange;
359	}
360	
361	/**
362	* Sets the value of the passwordChange property.
363	*
364	* @param value

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/ PasswordRequestType.java, line 73 (Password Management: Password in Comment)	Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/PasswordRequestType.java:73
Taint Flags:

70	this.oldPassword = value;
71	}
72	
73	/**
74	* Gets the value of the newPassword property.
75	*
76	* @return

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/ PasswordChange.java, line 47 (Password Management: Password in Comment)	Low

Issue Details

Kingdom: Security Features
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/PasswordChange.java:47
Taint Flags:

44	@XmlElement(name = "Request", required = true)
45	protected PasswordRequestType request;
46	

Password Management: Password in Comment	Low
Package: .src.main.java.gov.va.med.pharmacy.ncpdpmsgs	
PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/ncpdpmsgs/ PasswordChange.java, line 47 (Password Management: Password in Comment)	Low

47	/**
48	* Gets the value of the request property.
49	*
50	* @return

Poor Error Handling: Overly Broad Catch (19 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 ugly and repetitive, but "condensing" catch blocks by catching a high-level class like `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:
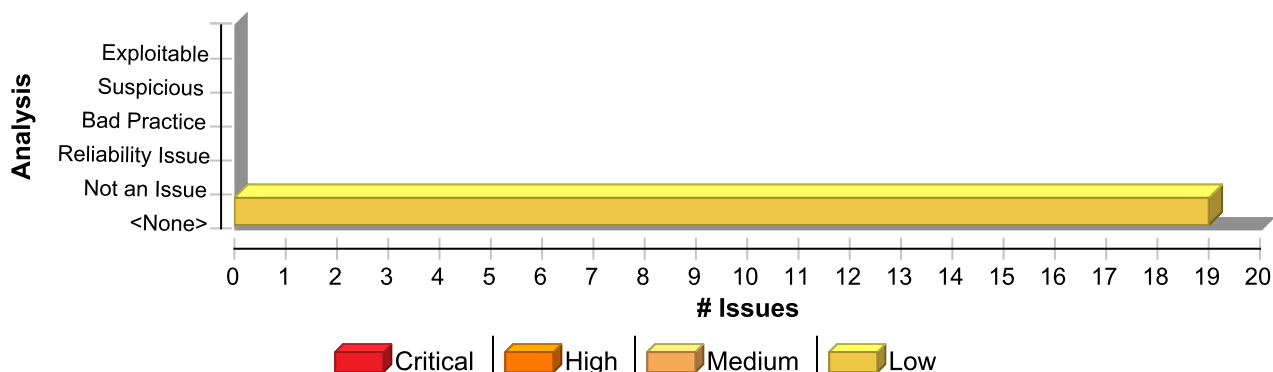
```
try {
  doExchange();
}
catch (Exception e) {
  logger.error("doExchange failed", e);
}
```

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 like `Exception`, `Throwable`, `Error`, or except at the very top level of the program or thread.

Issue Summary

Analysis (y-axis): Exploitable, Suspicious, Bad Practice, Reliability Issue, Not an Issue, <None>

X-axis: **# Issues** (0–20)

Legend: ■ Critical | ■ High | ■ Medium | ■ Low

Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Poor Error Handling: Overly Broad Catch	19	0	0	19
Total	**19**	**0**	**0**	**19**

Poor Error Handling: Overly Broad Catch	Low

Package: gov.va.med.pharmacy.jaxrs.drug.service.impl	

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/drug/service/impl/DrugSearchImpl.java, line 54 (Poor Error Handling: Overly Broad Catch)	Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: search()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/drug/service/impl/DrugSearchImpl.java:54
Taint Flags:

```
51   searchResponse.setSuccess(searchResults.getSuccess());
52
53
54   } catch (Exception e) {
55   searchResponse.setSuccess(false);
56
57   searchResponse.setErrorMessage("Error while searching drug: " + e.getMessage());
```

Package: gov.va.med.pharmacy.jaxrs.eAnde.services.impl	

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/eAnde/services/impl/PatientInfoService.java, line 47 (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: gov.va.med.pharmacy.jaxrs.eAnde.services.impl

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/eAnde/services/impl/ PatientInfoService.java, line 47 (Poor Error Handling: Overly Broad Catch)	Low

Sink: CatchBlock
Enclosing Method: fetchEligibilityStatusByKey()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/eAnde/services/impl/PatientInfoService.java:47
Taint Flags:

```
44   // response.setEnrollmentStatus(getPatientInfoDao().lookupPatientByKey(
45   //request.getPatientInfo().getKey()).getEnrollmentStatus());
46   }
47   catch (Exception e)
48   {
49   response.setSuccess(false);
50   response.setErrorMessage(e.getClass() + ": " + e.getMessage());
```

Package: gov.va.med.pharmacy.jaxrs.inboundncpdpmessage.service.impl

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/ inboundncpdpmessage/service/impl/InboundNCPDPMessageServiceImpl.java, line 158 (Poor Error Handling: Overly Broad Catch)	Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: getInboundMessage()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/inboundncpdpmessage/service/impl/ InboundNCPDPMessageServiceImpl.java:158
Taint Flags:

```
155   + "</Body>"
156   + "</Message>";
157
158   } catch (Exception e) {
159
160   LOG.error("Error in InboundNCPDPMessageServiceImpl:" + e.getMessage());
161
```

Package: gov.va.med.pharmacy.jaxrs.mvi.service.impl

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/service/impl/ PatientSearchImpl.java, line 93 (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: gov.va.med.pharmacy.jaxrs.mvi.service.impl	

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/service/impl/PatientSearchImpl.java, line 93 (Poor Error Handling: Overly Broad Catch)	Low

Sink: CatchBlock
Enclosing Method: searchPatientByName()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/service/impl/PatientSearchImpl.java:93
Taint Flags:

```
90   break;
91   }
92
93   } catch (Exception e) {
94   LOG.error("Error in PatientSearchImpl:"+e.getMessage());
95
96   searchResponse.setSuccess(false);
```

Package: gov.va.med.pharmacy.jaxrs.outboundncpdpmessage.service.impl	

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/outboundncpdpmessage/service/impl/OutboundNCPDPMessageServiceImpl.java, line 160 (Poor Error Handling: Overly Broad Catch)	Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: sendOutboundMessage()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/outboundncpdpmessage/service/impl/OutboundNCPDPMessageServiceImpl.java:160
Taint Flags:

```
157   }
158
159
160   } catch (Exception e) {
161
162   LOG.error("Error in OutboundNCPDPMessageServiceImpl:"+e.getMessage());
163
```

Package: gov.va.med.pharmacy.jaxrs.provider.service.impl	

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/provider/service/impl/ProviderSearchImpl.java, line 53 (Poor Error Handling: Overly Broad Catch)	Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Poor Error Handling: Overly Broad Catch	Low

Package: gov.va.med.pharmacy.jaxrs.provider.service.impl

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/provider/service/impl/ProviderSearchImpl.java, line 53 (Poor Error Handling: Overly Broad Catch)	Low

Sink Details

Sink: CatchBlock
Enclosing Method: search()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/provider/service/impl/ProviderSearchImpl.java:53
Taint Flags:

```
50
51
52
53   } catch (Exception e) {
54
55   LOG.error("Error in ProviderSearchImpl:"+e.getMessage());
56
```

Package: gov.va.med.pharmacy.jaxrs.tovista.service.impl

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/tovista/service/impl/VistaInboundSendImpl.java, line 47 (Poor Error Handling: Overly Broad Catch)	Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: send()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/tovista/service/impl/VistaInboundSendImpl.java:47
Taint Flags:

```
44   sendResponse.setSuccess(sendResults.getSuccess());
45
46
47   } catch (Exception e) {
48   sendResponse.setSuccess(false);
49
50   sendResponse.setErrorMessage("Error while sending vista inbound message: " +
e.getMessage());
```

Package: gov.va.med.pharmacy.jaxrs.vistaoutboundmsg.service.impl

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/vistaoutboundmsg/service/impl/VistaOutboundMsgImpl.java, line 132 (Poor Error Handling: Overly Broad Catch)	Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Poor Error Handling: Overly Broad Catch	Low

Package: gov.va.med.pharmacy.jaxrs.vistaoutboundmsg.service.impl

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/vistaoutboundmsg/service/impl/VistaOutboundMsgImpl.java, line 132 (Poor Error Handling: Overly Broad Catch)	Low

Sink Details

Sink: CatchBlock
Enclosing Method: persistXMLMessage()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/vistaoutboundmsg/service/impl/
VistaOutboundMsgImpl.java:132
Taint Flags:

```
129  wsResponse.setSuccess(true);
130
131
132  } catch (Exception e) {
133
134  LOG.error("Error in VistaOutboundMsgImpl:"+e.getMessage());
135
```

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/vistaoutboundmsg/service/impl/VistaOutboundMsgImpl.java, line 88 (Poor Error Handling: Overly Broad Catch)	Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: persistMessage()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/vistaoutboundmsg/service/impl/
VistaOutboundMsgImpl.java:88
Taint Flags:

```
85  wsResponse.setSuccess(true);
86
87
88  } catch (Exception e) {
89
90  LOG.error("Error in VistaOutboundMsgImpl:"+e.getMessage());
91
```

Package: gov.va.med.pharmacy.jms.messaging

PS_INB_ERX_JMS/src/main/java/gov/va/med/pharmacy/jms/messaging/InboundMsgProcessor.java, line 57 (Poor Error Handling: Overly Broad Catch)	Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Poor Error Handling: Overly Broad Catch	Low

Package: gov.va.med.pharmacy.jms.messaging	

PS_INB_ERX_JMS/src/main/java/gov/va/med/pharmacy/jms/messaging/ InboundMsgProcessor.java, line 57 (Poor Error Handling: Overly Broad Catch)	Low

Sink Details

Sink: CatchBlock
Enclosing Method: getERxMessage()
File: PS_INB_ERX_JMS/src/main/java/gov/va/med/pharmacy/jms/messaging/InboundMsgProcessor.java:57
Taint Flags:

```
54
55   inboundNcpdpMsgService.saveInboundERx(inboundeRx);
56
57   } catch (Exception e) {
58
59   LOG.error("Error in getERxMessage:"+e.getMessage());
60   }
```

Package: gov.va.med.pharmacy.persistence.service.impl	

PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/service/ impl/UserServiceImpl.java, line 40 (Poor Error Handling: Overly Broad Catch)	Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: addNewVAUser()
File: PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/service/impl/UserServiceImpl.java:40
Taint Flags:

```
37   {
38   userDao.save(user);
39
40   } catch (Exception e) {
41   // TODO Auto-generated catch block
42   e.printStackTrace();
43   }
```

Package: gov.va.med.pharmacy.web.administration	

PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/administration/ PharmacyManagementController.java, line 264 (Poor Error Handling: Overly Broad Catch)	Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Poor Error Handling: Overly Broad Catch	Low

Package: gov.va.med.pharmacy.web.administration	

PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/administration/ PharmacyManagementController.java, line 264 (Poor Error Handling: Overly Broad Catch)	**Low**

Sink Details

Sink: CatchBlock
Enclosing Method: updatePharmacy()
File: PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/administration/PharmacyManagementController.java:264
Taint Flags:

```
261  errorsList.add("Update Failed. Unable to save record. Please contact production
support.");
262  }
263  }
264  catch (Exception e) {
265
266  e.printStackTrace();
267
```

PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/administration/ PharmacyManagementController.java, line 599 (Poor Error Handling: Overly Broad Catch)	**Low**

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: addNewPharmacy()
File: PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/administration/PharmacyManagementController.java:599
Taint Flags:

```
596
597  hasErrors = true;
598  }
599  catch (Exception e) {
600
601  e.printStackTrace();
602
```

PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/administration/ UserManagementController.java, line 313 (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: gov.va.med.pharmacy.web.administration	
PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/administration/ UserManagementController.java, line 313 (Poor Error Handling: Overly Broad Catch)	Low

Sink: CatchBlock
Enclosing Method: processForm()
File: PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/administration/UserManagementController.java:313
Taint Flags:

310	`}`
311	
312	
313	`} catch (Exception ex) {`
314	
315	`result.addError(new ObjectError(USER_MANAGEMENT_MODEL, "A system problem is preventing the action to add a new user at this time. Please report this error to the production support."));`
316	

Package: gov.va.med.pharmacy.web.auth	
PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/auth/ LoginController.java, line 113 (Poor Error Handling: Overly Broad Catch)	Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: authenticateUser()
File: PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/auth/LoginController.java:113
Taint Flags:

110	`} catch (IntrusionException e) {`
111	`// TODO Auto-generated catch block`
112	`e.printStackTrace();`
113	`} catch (Exception e) {`
114	`// TODO Auto-generated catch block`
115	`e.printStackTrace();`
116	`}`

Package: gov.va.med.pharmacy.web.reports	
PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/reports/ SummaryReportController.java, line 165 (Poor Error Handling: Overly Broad Catch)	Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Poor Error Handling: Overly Broad Catch	Low

Package: gov.va.med.pharmacy.web.reports	
PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/reports/ SummaryReportController.java, line 165 (Poor Error Handling: Overly Broad Catch)	Low

Sink Details

Sink: CatchBlock
Enclosing Method: csvExport()
File: PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/reports/SummaryReportController.java:165
Taint Flags:

```
162   response.getOutputStream().flush();
163
164
165   } catch (Exception e) {
166   // TODO Auto-generated catch block
167   e.printStackTrace();
168   }
```

Package: gov.va.med.pharmacy.web.track	
PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/track/ TrackRxController.java, line 197 (Poor Error Handling: Overly Broad Catch)	Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: csvExport()
File: PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/track/TrackRxController.java:197
Taint Flags:

```
194   }
195
196
197   } catch (Exception e) {
198   // TODO Auto-generated catch block
199   e.printStackTrace();
200   }
```

Package: gov.va.med.pharmacy.wsclients.eAnde	
PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/ EeSummaryPortService.java, line 33 (Poor Error Handling: Overly Broad Catch)	Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Poor Error Handling: Overly Broad Catch	Low

Package: gov.va.med.pharmacy.wsclients.eAnde

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/ EeSummaryPortService.java, line 33 (Poor Error Handling: Overly Broad Catch)	Low

Sink Details

Sink: CatchBlock
Enclosing Method: ()
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/EeSummaryPortService.java:33
Taint Flags:

```
30  URL url = null;
31  try {
32  url = EeSummaryPortService.class.getResource("wsdl/eAnde/eeSummary.wsdl");
33  } catch (Exception e) {
34  java.util.logging.Logger.getLogger(EeSummaryPortService.class.getName())
35  .log(java.util.logging.Level.INFO,
36  "Can not initialize the default wsdl from {0}", "wsdl/eAnde/eeSummary.wsdl");
```

Package: gov.va.med.pharmacy.wsclients.mvi

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ VAIdM.java, line 30 (Poor Error Handling: Overly Broad Catch)	Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: CatchBlock
Enclosing Method: ()
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/VAIdM.java:30
Taint Flags:

```
27  URL url = null;
28  try {
29  url = VAIdM.class.getResource("wsdl/mvi/PREeRX_IdmHl7v3_FlatFile.wsdl");
30  } catch (Exception e) {
31  java.util.logging.Logger.getLogger(VAIdM.class.getName())
32  .log(java.util.logging.Level.INFO,
33  "Can not initialize the default wsdl from {0}", "wsdl/mvi/PREeRX_IdmHl7v3_FlatFile.wsdl");
```

Poor Error Handling: Overly Broad Throws (16 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 {
  ...
}
```

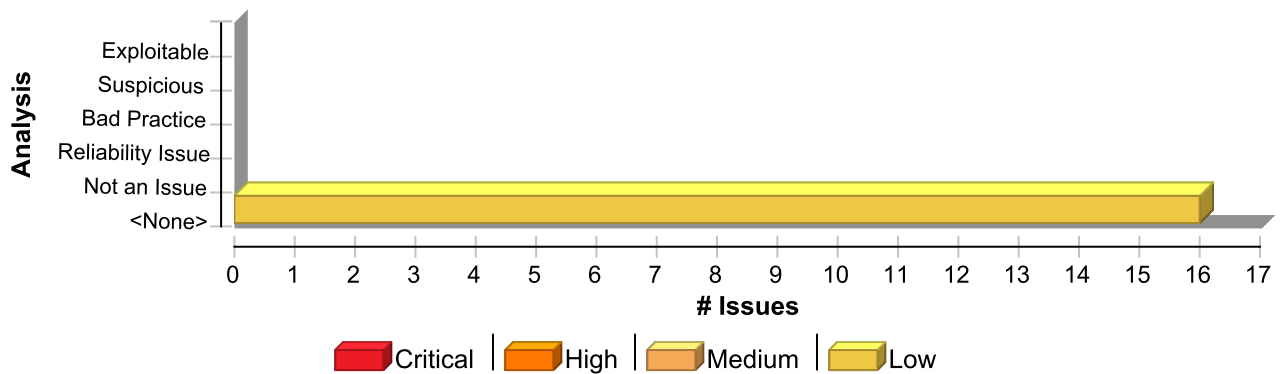While it might seem tidier to write

```
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	16	0	0	16
Total	**16**	**0**	**0**	**16**

Poor Error Handling: Overly Broad Throws	Low

Package: gov.va.med.pharmacy.jaxrs.drug.dao	

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/drug/dao/ DrugSearchDao.java, line 13 (Poor Error Handling: Overly Broad Throws)	Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: lookupDrug
Enclosing Method: lookupDrug()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/drug/dao/DrugSearchDao.java:13
Taint Flags:

```
10   * @return
11   * @throws Exception
12   */
13   public Drug lookupDrug(Drug drugObj) throws Exception;
14  }
15
16  undefined
```

Package: gov.va.med.pharmacy.jaxrs.drug.dao.impl	

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/drug/dao/impl/ DrugRpc.java, line 27 (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: gov.va.med.pharmacy.jaxrs.drug.dao.impl	
PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/drug/dao/impl/ DrugRpc.java, line 27 (Poor Error Handling: Overly Broad Throws)	**Low**

Sink: Function: lookupDrug
Enclosing Method: lookupDrug()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/drug/dao/impl/DrugRpc.java:27
Taint Flags:

```
24   this.vistaLinkConnectionUtility = vistaLinkConnectionUtility;
25   }
26
27   public Drug lookupDrug(Drug searchObj) throws Exception {
28
29   Drug searchResults = new Drug();
30   //inject
```

Package: gov.va.med.pharmacy.jaxrs.eAnde.dao	
PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/eAnde/dao/ PatientInfoDao.java, line 7 (Poor Error Handling: Overly Broad Throws)	**Low**

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: lookupPatientByKey
Enclosing Method: lookupPatientByKey()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/eAnde/dao/PatientInfoDao.java:7
Taint Flags:

```
4
5 public interface PatientInfoDao {
6
7   public PatientInfo lookupPatientByKey(String key) throws Exception;
8 }
9
10 undefined
```

Package: gov.va.med.pharmacy.jaxrs.eAnde.services.impl	
PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/eAnde/services/impl/ EESummary_Client.java, line 40 (Poor Error Handling: Overly Broad Throws)	**Low**

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: returnResponse

Poor Error Handling: Overly Broad Throws	Low

Package: gov.va.med.pharmacy.jaxrs.eAnde.services.impl	

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/eAnde/services/impl/ EESummary_Client.java, line 40 (Poor Error Handling: Overly Broad Throws)	Low

Enclosing Method: returnResponse()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/eAnde/services/impl/EESummary_Client.java:40
Taint Flags:

```
37  private static final String E_AND_E_CLIENT_ENDPOINT = "eAnde.client.endpoint";
38  private static final QName SERVICE_NAME = new QName("http://
jaxws.webservices.esr.med.va.gov/schemas", "eeSummaryPortService");
39
40  public getEESummaryResponse returnResponse(String key) throws java.lang.Exception {
41
42  Properties properties = new Properties();
43
```

Package: gov.va.med.pharmacy.jaxrs.mvi.dao	

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/dao/ PatientSearchDao.java, line 15 (Poor Error Handling: Overly Broad Throws)	Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: lookupPatientByName
Enclosing Method: lookupPatientByName()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/dao/PatientSearchDao.java:15
Taint Flags:

```
12  * @return Patient
13  * @throws Exception
14  */
15  public Patient lookupPatientByName(Patient patientObj) throws Exception;
16  }
17
18 undefined
```

Package: gov.va.med.pharmacy.jaxrs.mvi.dao.impl	

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/dao/impl/ MVIClient.java, line 150 (Poor Error Handling: Overly Broad Throws)	Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: getMVIData

Poor Error Handling: Overly Broad Throws	Low

Package: gov.va.med.pharmacy.jaxrs.mvi.dao.impl	

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/dao/impl/ MVIClient.java, line 150 (Poor Error Handling: Overly Broad Throws)	Low

Enclosing Method: getMVIData()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/dao/impl/MVIClient.java:150
Taint Flags:

```
147  //@Autowired
148  //private Environment environment;
149
150  public Patient getMVIData(Patient patientSearchObj) throws java.lang.Exception {
151
152  // The following prints out a lot of debug information.
153  // It should only be uncommented in dev environment or for
```

Package: gov.va.med.pharmacy.jaxrs.provider.dao	

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/provider/dao/ ProviderSearchDao.java, line 14 (Poor Error Handling: Overly Broad Throws)	Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: lookupProvider
Enclosing Method: lookupProvider()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/provider/dao/ProviderSearchDao.java:14
Taint Flags:

```
11   * @return Provider
12   * @throws Exception
13   */
14   public Provider lookupProvider(Provider providerObj) throws Exception;
15   }
16
17   undefined
```

Package: gov.va.med.pharmacy.jaxrs.provider.dao.impl	

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/provider/dao/impl/ ProviderRPC.java, line 19 (Poor Error Handling: Overly Broad Throws)	Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: lookupProvider
Enclosing Method: lookupProvider()

Poor Error Handling: Overly Broad Throws	Low

Package: gov.va.med.pharmacy.jaxrs.provider.dao.impl

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/provider/dao/impl/ ProviderRPC.java, line 19 (Poor Error Handling: Overly Broad Throws)	Low

File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/provider/dao/impl/ProviderRPC.java:19
Taint Flags:

```
16
17  private static final org.apache.logging.log4j.Logger LOG =
org.apache.logging.log4j.LogManager.getLogger(ProviderRPC.class);
18
19  public Provider lookupProvider(Provider searchObj) throws Exception {
20
21  Provider searchResults = new Provider();
22
```

Package: gov.va.med.pharmacy.jaxrs.tovista.dao

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/tovista/dao/ VistaInboundDao.java, line 13 (Poor Error Handling: Overly Broad Throws)	Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: sendMessage
Enclosing Method: sendMessage()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/tovista/dao/VistaInboundDao.java:13
Taint Flags:

```
10  * @return
11  * @throws Exception
12  */
13  public VistaInbound sendMessage(VistaInbound vistaInboundObj) throws Exception;
14 }
15
16 undefined
```

Package: gov.va.med.pharmacy.jaxrs.tovista.dao.impl

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/tovista/dao/impl/ VistaInboundRpc.java, line 42 (Poor Error Handling: Overly Broad Throws)	Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: sendMessage
Enclosing Method: sendMessage()

Poor Error Handling: Overly Broad Throws	Low

Package: gov.va.med.pharmacy.jaxrs.tovista.dao.impl	
PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/tovista/dao/impl/VistaInboundRpc.java, line 42 (Poor Error Handling: Overly Broad Throws)	Low

File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/tovista/dao/impl/VistaInboundRpc.java:42
Taint Flags:

```
39   }
40
41
42   public VistaInbound sendMessage(VistaInbound sendObj) throws Exception {
43
44
45   VistaInbound searchResults = new VistaInbound();
```

Package: gov.va.med.pharmacy.jms.messaging	
PS_INB_ERX_JMS/src/main/java/gov/va/med/pharmacy/jms/messaging/InboundMsgProducer.java, line 21 (Poor Error Handling: Overly Broad Throws)	Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: enqueueERxMessage
Enclosing Method: enqueueERxMessage()
File: PS_INB_ERX_JMS/src/main/java/gov/va/med/pharmacy/jms/messaging/InboundMsgProducer.java:21
Taint Flags:

```
18   @Autowired
19   private JmsTemplate jmsTemplate;
20
21   public void enqueueERxMessage(MessageType eRxMessage) throws Exception {
22
23   try {
24
```

Package: gov.va.med.pharmacy.persistence.service	
PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/service/OutboundNcpdpMsgService.java, line 7 (Poor Error Handling: Overly Broad Throws)	Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: saveOutboundMsg
Enclosing Method: saveOutboundMsg()

Poor Error Handling: Overly Broad Throws	Low

Package: gov.va.med.pharmacy.persistence.service	

PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/service/OutboundNcpdpMsgService.java, line 7 (Poor Error Handling: Overly Broad Throws)	Low

File: PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/service/OutboundNcpdpMsgService.java:7
Taint Flags:

```
4
5  public interface OutboundNcpdpMsgService {
6
7    public void saveOutboundMsg(OutboundNcpdpMsgEntity outboundMsg) throws Exception ;
8
9  public OutboundNcpdpMsgEntity findById(long id);
10 }
```

PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/service/PharmacyService.java, line 17 (Poor Error Handling: Overly Broad Throws)	Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: findById
Enclosing Method: findById()
File: PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/service/PharmacyService.java:17
Taint Flags:

```
14
15   public PharmacyEntity findByNCPDPId(String NCPDPId);
16
17   public PharmacyEntity findById(Long pharmacyId) throws Exception;
18
19   public void updatePharmacyInfo(PharmacyEntity pharmacy) throws HibernateException;
20
```

PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/service/InboundNcpdpMsgService.java, line 7 (Poor Error Handling: Overly Broad Throws)	Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: saveInboundERx
Enclosing Method: saveInboundERx()
File: PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/service/InboundNcpdpMsgService.java:7
Taint Flags:

Poor Error Handling: Overly Broad Throws	Low

PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/service/ InboundNcpdpMsgService.java, line 7 (Poor Error Handling: Overly Broad Throws)	Low

```
4
5  public interface InboundNcpdpMsgService {
6
7    void saveInboundERx(InboundNcpdpMsgEntity inboundeRx) throws Exception ;
8
9  }
10
```

Package: gov.va.med.pharmacy.utility	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/ VistaLinkConnectionUtility.java, line 51 (Poor Error Handling: Overly Broad Throws)	Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: sendRequest
Enclosing Method: sendRequest()
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/VistaLinkConnectionUtility.java:51
Taint Flags:

```
48   * @return The RpcResponse.
49   * @throws Exception
50   */
51   public RpcResponse sendRequest(String division, String rpcContext, String rpcName,
ArrayList<Object> rpcParams) throws Exception {
52
53   // LOG.debug("VistA request [" + rpcContext + ", " + rpcName + "] with parameters: " +
rpcParams);
54
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/ VistaLinkConnectionUtility.java, line 122 (Poor Error Handling: Overly Broad Throws)	Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: Function: getConnection
Enclosing Method: getConnection()
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/VistaLinkConnectionUtility.java:122

Poor Error Handling: Overly Broad Throws	Low
Package: gov.va.med.pharmacy.utility	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/ VistaLinkConnectionUtility.java, line 122 (Poor Error Handling: Overly Broad Throws)	Low

Taint Flags:

119	`* @return A VistaLinkConnection.`
120	`* @throws Exception`
121	`*/`
122	`private VistaLinkConnection getConnection(String division, String proxyUser) throws Exception {`
123	
124	`// LOG.debug("Connecting to VistAlink as: " + userId + ", " + division);`
125	

FORTIFY®

Poor Error Handling: Throw Inside Finally (1 issue)

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);
            }
        }
    }
}
```

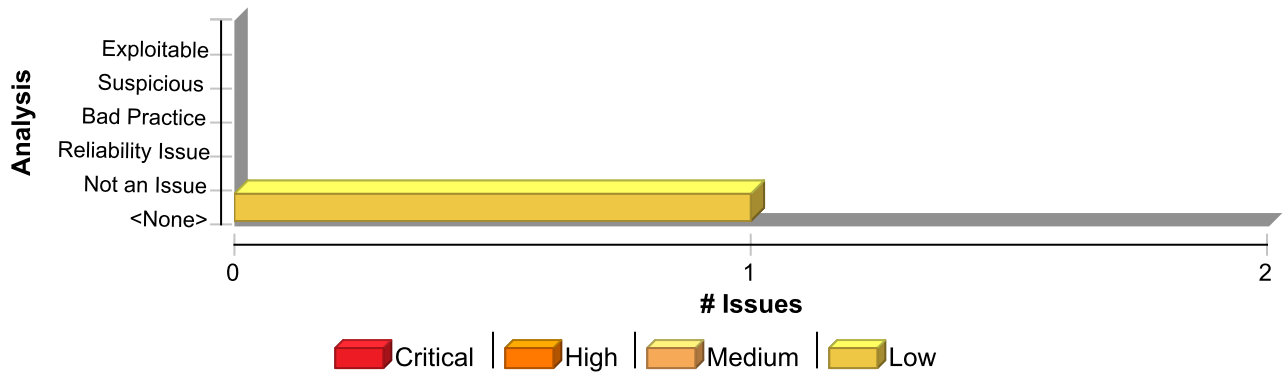This category is from the Cigital Java Rulepack. http://www.cigital.com/

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	1	0	0	1
Total	**1**	**0**	**0**	**1**

Poor Error Handling: Throw Inside Finally	Low
Package: gov.va.med.pharmacy.utility	
PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/ VistaLinkConnectionUtility.java, line 146 (Poor Error Handling: Throw Inside Finally)	Low

Issue Details

Kingdom: Errors
Scan Engine: SCA (Structural)

Sink Details

Sink: FinallyBlock
Enclosing Method: getConnection()
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/VistaLinkConnectionUtility.java:146
Taint Flags:

```
143  // LOG.error("VistALink error while retrieving connection.", e);
144  //throw new InterfaceException(e, InterfaceException.INTERFACE_ERROR,
     InterfaceException.PRE_ENCAPSULATION);
145  throw new Exception (e);
146  } finally {
147  if (ic != null){
148  ic.close();
149  };
```

Poor Style: Confusing Naming (167 issues)

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() {
    ...
  }
}
```
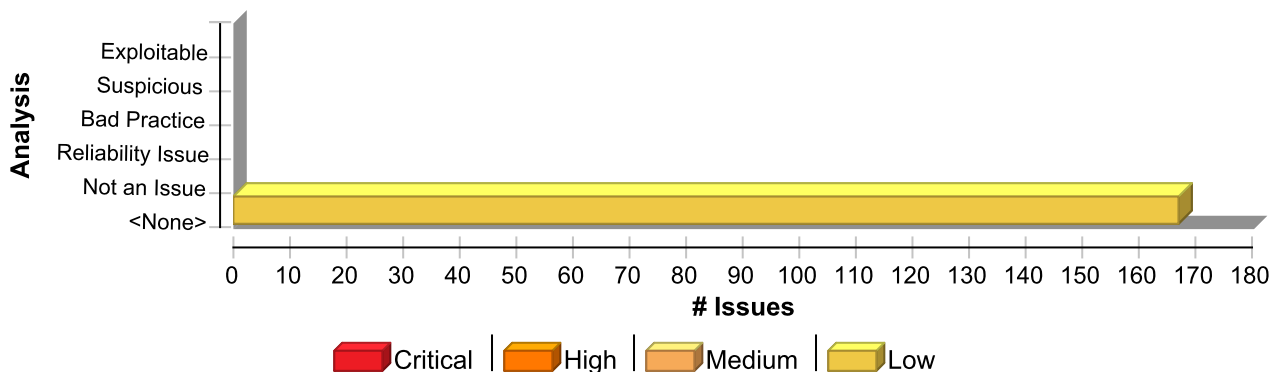
Recommendation

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

FORTIFY®

Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Poor Style: Confusing Naming	167	0	0	167
Total	**167**	**0**	**0**	**167**

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.eAnde	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/NotificationType.java, line 35 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/eAnde/NotificationType.java:35
Taint Flags:

```
32   LINK("Link"),
33   @XmlEnumValue("Move")
34   MOVE("Move");
35   private final String value;
36
37   NotificationType(String v) {
38   value = v;
```

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/TableCellVerticalAlign.java, line 38 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/TableCellVerticalAlign.java:38
Taint Flags:

```
35   MIDDLE("middle"),
36   @XmlEnumValue("top")
37   TOP("top");
38   private final String value;
39
40   TableCellVerticalAlign(String v) {
41   value = v;
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ParameterizedDataTypeInterval.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ParameterizedDataTypeInterval.java:29
Taint Flags:

```
26
27    @XmlEnumValue("IVL<T>")
28    IVL_T("IVL<T>");
29    private final String value;
30
31    ParameterizedDataTypeInterval(String v) {
32    value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeUncertainProbabilisticConceptDescriptor.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/
DataTypeUncertainProbabilisticConceptDescriptor.java:29
Taint Flags:

```
26
27    @XmlEnumValue("UVP<CD>")
28    UVP_CD("UVP<CD>");
29    private final String value;
30
31    DataTypeUncertainProbabilisticConceptDescriptor(String v) {
32    value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ CentralNumic.java, line 35 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Poor Style: Confusing Naming	Low
Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ CentralNumic.java, line 35 (Poor Style: Confusing Naming)	Low

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/CentralNumic.java:35
Taint Flags:

```
32   X_PAR("x-PAR"),
33   @XmlEnumValue("x-SHH")
34   X_SHH("x-SHH");
35   private final String value;
36
37   CentralNumic(String v) {
38   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeSetOfCharacterStrings.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/DataTypeSetOfCharacterStrings.java:29
Taint Flags:

```
26
27   @XmlEnumValue("SET<ST>")
28   SET_ST("SET<ST>");
29   private final String value;
30
31   DataTypeSetOfCharacterStrings(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ParameterizedDataTypeHistorical.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ParameterizedDataTypeHistorical.java:29
Taint Flags:

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ParameterizedDataTypeHistorical.java, line 29 (Poor Style: Confusing Naming)	Low

```
26
27    @XmlEnumValue("HXIT<T>")
28    HXIT_T("HXIT<T>");
29    private final String value;
30
31    ParameterizedDataTypeHistorical(String v) {
32    value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ConceptCodeRelationship.java, line 35 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ConceptCodeRelationship.java:35
Taint Flags:

```
32    HAS_SUBTYPE("hasSubtype"),
33    @XmlEnumValue("smallerThan")
34    SMALLER_THAN("smallerThan");
35    private final String value;
36
37    ConceptCodeRelationship(String v) {
38    value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ Pai.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Pai.java:29
Taint Flags:

```
26
27    @XmlEnumValue("x-YUF")
28    X_YUF("x-YUF");
29    private final String value;
30
31    Pai(String v) {
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Pai.java, line 29 (Poor Style: Confusing Naming)	Low

```
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/RealmOfUse.java, line 36 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/RealmOfUse.java:36
Taint Flags:

```
33   NORTH_AMERICA("NorthAmerica"),
34   USA("USA"),
35   UV("UV");
36   private final String value;
37
38   RealmOfUse(String v) {
39   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/TextMediaType.java, line 44 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/TextMediaType.java:44
Taint Flags:

```
41   TEXT_SGML("text/sgml"),
42   @XmlEnumValue("text/xml")
43   TEXT_XML("text/xml");
44   private final String value;
45
46   TextMediaType(String v) {
47   value = v;
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	
PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeParametricProbabilityDistribution.java, line 29 (Poor Style: Confusing Naming)	**Low**

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeParametricProbabilityDistribution.java:29
Taint Flags:

```
26
27   @XmlEnumValue("PPD<QTY>")
28   PPD_QTY("PPD<QTY>");
29   private final String value;
30
31   DataTypeParametricProbabilityDistribution(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ParameterizedDataTypeUncertainValueProbabilistic.java, line 29 (Poor Style: Confusing Naming)	**Low**

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ParameterizedDataTypeUncertainValueProbabilistic.java:29
Taint Flags:

```
26
27   @XmlEnumValue("UVP<T>")
28   UVP_T("UVP<T>");
29   private final String value;
30
31   ParameterizedDataTypeUncertainValueProbabilistic(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ OregonAthapaskan.java, line 32 (Poor Style: Confusing Naming)	**Low**

Issue Details

Kingdom: Code Quality

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ OregonAthapaskan.java, line 32 (Poor Style: Confusing Naming)	Low

Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/OregonAthapaskan.java:32
Taint Flags:

```
29   X_TOL("x-TOL"),
30   @XmlEnumValue("x-TUU")
31   X_TUU("x-TUU");
32   private final String value;
33
34   OregonAthapaskan(String v) {
35   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ Palaihnihan.java, line 32 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Palaihnihan.java:32
Taint Flags:

```
29   X_ACH("x-ACH"),
30   @XmlEnumValue("x-ATW")
31   X_ATW("x-ATW");
32   private final String value;
33
34   Palaihnihan(String v) {
35   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ XLabSpecimenCollectionProviders.java, line 47 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/XLabSpecimenCollectionProviders.java:

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ XLabSpecimenCollectionProviders.java, line 47 (Poor Style: Confusing Naming)	Low

47
Taint Flags:

```
44   SELF("self"),
45   @XmlEnumValue("thirdParty")
46   THIRD_PARTY("thirdParty");
47   private final String value;
48
49   XLabSpecimenCollectionProviders(String v) {
50   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ApplicationMediaType.java, line 35 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ApplicationMediaType.java:35
Taint Flags:

```
32   APPLICATION_MSWORD("application/msword"),
33   @XmlEnumValue("application/pdf")
34   APPLICATION_PDF("application/pdf");
35   private final String value;
36
37   ApplicationMediaType(String v) {
38   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ActExposureCode.java, line 63 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ActExposureCode.java:63
Taint Flags:

```
60   SOCIAL_2("SOCIAL2"),
61   @XmlEnumValue("WORK2")
62   WORK_2("WORK2");
```

Poor Style: Confusing Naming	Low

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ActExposureCode.java, line 63 (Poor Style: Confusing Naming)	Low

```
63   private final String value;
64
65   ActExposureCode(String v) {
66   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ CodingRationale.java, line 33 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/CodingRationale.java:33
Taint Flags:

```
30   @XmlEnumValue("HL7")
31   HL_7("HL7"),
32   SRC("SRC");
33   private final String value;
34
35   CodingRationale(String v) {
36   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ Tsimshianic.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Tsimshianic.java:29
Taint Flags:

```
26
27   @XmlEnumValue("x-TSI")
28   X_TSI("x-TSI");
29   private final String value;
30
31   Tsimshianic(String v) {
32   value = v;
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ MdfRmimRowType.java, line 41 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/MdfRmimRowType.java:41
Taint Flags:

```
38   RMIM("rmim"),
39   @XmlEnumValue("stc")
40   STC("stc");
41   private final String value;
42
43   MdfRmimRowType(String v) {
44   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ XPhoneOrEmailURLScheme.java, line 35 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/XPhoneOrEmailURLScheme.java:35
Taint Flags:

```
32   MAILTO("mailto"),
33   @XmlEnumValue("tel")
34   TEL("tel");
35   private final String value;
36
37   XPhoneOrEmailURLScheme(String v) {
38   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ValueSetPropertyId.java, line 35 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ValueSetPropertyId.java, line 35 (Poor Style: Confusing Naming)	Low

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ValueSetPropertyId.java:35
Taint Flags:

```
32   HOW_APPLIES("howApplies"),
33   @XmlEnumValue("openIssue")
34   OPEN_ISSUE("openIssue");
35   private final String value;
36
37   ValueSetPropertyId(String v) {
38   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ KutchinHan.java, line 32 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/KutchinHan.java:32
Taint Flags:

```
29   X_HAA("x-HAA"),
30   @XmlEnumValue("x-KUC")
31   X_KUC("x-KUC");
32   private final String value;
33
34   KutchinHan(String v) {
35   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ActMoodCriterion.java, line 31 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ActMoodCriterion.java:31
Taint Flags:

```
28   CRT("CRT"),
29   @XmlEnumValue("EVN.CRT")
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ActMoodCriterion.java, line 31 (Poor Style: Confusing Naming)	Low

```
30   EVN_CRT("EVN.CRT");
31   private final String value;
32
33   ActMoodCriterion(String v) {
34   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ Pomoan.java, line 41 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Pomoan.java:41
Taint Flags:

```
38   X_PEO("x-PEO"),
39   @XmlEnumValue("x-PEQ")
40   X_PEQ("x-PEQ");
41   private final String value;
42
43   Pomoan(String v) {
44   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ WesternMiwok.java, line 32 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/WesternMiwok.java:32
Taint Flags:

```
29   X_CSI("x-CSI"),
30   @XmlEnumValue("x-LMW")
31   X_LMW("x-LMW");
32   private final String value;
33
34   WesternMiwok(String v) {
35   value = v;
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ Tsamosan.java, line 38 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Tsamosan.java:38
Taint Flags:

```
35   X_QUN("x-QUN"),
36   @XmlEnumValue("x-CJH")
37   X_CJH("x-CJH");
38   private final String value;
39
40   Tsamosan(String v) {
41   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeBagOfPhysicalQuantities.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/DataTypeBagOfPhysicalQuantities.java:29
Taint Flags:

```
26
27   @XmlEnumValue("BAG<PQ>")
28   BAG_PQ("BAG<PQ>");
29   private final String value;
30
31   DataTypeBagOfPhysicalQuantities(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeEventRelatedInterval.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeEventRelatedInterval.java, line 29 (Poor Style: Confusing Naming)	Low

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/DataTypeEventRelatedInterval.java:29
Taint Flags:

```
26
27   @XmlEnumValue("EIVL<TS>")
28   EIVL_TS("EIVL<TS>");
29   private final String value;
30
31   DataTypeEventRelatedInterval(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ UnitOfMeasureAtomBaseUnitSens.java, line 45 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/UnitOfMeasureAtomBaseUnitSens.java:45
Taint Flags:

```
42   RAD("rad"),
43   @XmlEnumValue("s")
44   S("s");
45   private final String value;
46
47   UnitOfMeasureAtomBaseUnitSens(String v) {
48   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ArtificialDentition.java, line 578 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ArtificialDentition.java:578
Taint Flags:

```
575   TID_9_PD("TID9pd"),
576   @XmlEnumValue("TID9pm")
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ArtificialDentition.java, line 578 (Poor Style: Confusing Naming)	**Low**

```
577   TID_9_PM("TID9pm");
578   private final String value;
579
580   ArtificialDentition(String v) {
581   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeParametricProbabilityDistributionOfPhysicalQuantities.java, line 29 (Poor Style: Confusing Naming)	**Low**

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/
DataTypeParametricProbabilityDistributionOfPhysicalQuantities.java:29
Taint Flags:

```
26
27   @XmlEnumValue("PPD<PQ>")
28   PPD_PQ("PPD<PQ>");
29   private final String value;
30
31   DataTypeParametricProbabilityDistributionOfPhysicalQuantities(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ChiwereWinnebago.java, line 32 (Poor Style: Confusing Naming)	**Low**

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ChiwereWinnebago.java:32
Taint Flags:

```
29   X_IOW("x-IOW"),
30   @XmlEnumValue("x-WIN")
31   X_WIN("x-WIN");
32   private final String value;
33
34   ChiwereWinnebago(String v) {
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ChiwereWinnebago.java, line 32 (Poor Style: Confusing Naming)	Low

```
35   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ XActStatusActiveComplete.java, line 32 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/XActStatusActiveComplete.java:32
Taint Flags:

```
29   ACTIVE("active"),
30   @XmlEnumValue("completed")
31   COMPLETED("completed");
32   private final String value;
33
34   XActStatusActiveComplete(String v) {
35   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ InuitInupiaq.java, line 32 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/InuitInupiaq.java:32
Taint Flags:

```
29   X_ESI("x-ESI"),
30   @XmlEnumValue("x-ESK")
31   X_ESK("x-ESK");
32   private final String value;
33
34   InuitInupiaq(String v) {
35   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ActSupplyFulfillmentRefusalReason.java, line 44 (Poor Style: Confusing Naming)	Low

Issue Details

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ActSupplyFulfillmentRefusalReason.java, line 44 (Poor Style: Confusing Naming)	Low

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/
ActSupplyFulfillmentRefusalReason.java:44
Taint Flags:

```
41   FRR_02("FRR02"),
42   @XmlEnumValue("FRR06")
43   FRR_06("FRR06");
44   private final String value;
45
46   ActSupplyFulfillmentRefusalReason(String v) {
47   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ParameterizedDataTypeNonParametricProbabilityDistribution.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/
ParameterizedDataTypeNonParametricProbabilityDistribution.java:29
Taint Flags:

```
26
27   @XmlEnumValue("NPPD<T>")
28   NPPD_T("NPPD<T>");
29   private final String value;
30
31   ParameterizedDataTypeNonParametricProbabilityDistribution(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ManagedParticipationStatusNormal.java, line 41 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Poor Style: Confusing Naming	Low
Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ManagedParticipationStatusNormal.java, line 41 (Poor Style: Confusing Naming)	Low

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ManagedParticipationStatusNormal.java:
41
Taint Flags:

```
38   COMPLETED("completed"),
39   @XmlEnumValue("pending")
40   PENDING("pending");
41   private final String value;
42
43   ManagedParticipationStatusNormal(String v) {
44   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ IntegrityCheckAlgorithm.java, line 32 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/IntegrityCheckAlgorithm.java:32
Taint Flags:

```
29   SHA_1("SHA-1"),
30   @XmlEnumValue("SHA-256")
31   SHA_256("SHA-256");
32   private final String value;
33
34   IntegrityCheckAlgorithm(String v) {
35   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ Nootkan.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Nootkan.java:29
Taint Flags:

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ Nootkan.java, line 29 (Poor Style: Confusing Naming)	Low

```
26
27   @XmlEnumValue("x-MYH")
28   X_MYH("x-MYH");
29   private final String value;
30
31   Nootkan(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ WesternNumic.java, line 32 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/WesternNumic.java:32
Taint Flags:

```
29   X_MON("x-MON"),
30   @XmlEnumValue("x-PAO")
31   X_PAO("x-PAO");
32   private final String value;
33
34   WesternNumic(String v) {
35   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ TableCellHorizontalAlign.java, line 41 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/TableCellHorizontalAlign.java:41
Taint Flags:

```
38   LEFT("left"),
39   @XmlEnumValue("right")
40   RIGHT("right");
41   private final String value;
42
43   TableCellHorizontalAlign(String v) {
```

Poor Style: Confusing Naming	Low

| Package: gov.va.med.pharmacy.wsclients.mvi | |

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ TableCellHorizontalAlign.java, line 41 (Poor Style: Confusing Naming)	Low

```
44   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ Sahaptian.java, line 38 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Sahaptian.java:38
Taint Flags:

```
35   X_WAA("x-WAA"),
36   @XmlEnumValue("x-YAK")
37   X_YAK("x-YAK");
38   private final String value;
39
40   Sahaptian(String v) {
41   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ SirenikskiYupik.java, line 35 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/SirenikskiYupik.java:35
Taint Flags:

```
32   X_ESS("x-ESS"),
33   @XmlEnumValue("x-EMS")
34   X_EMS("x-EMS");
35   private final String value;
36
37   SirenikskiYupik(String v) {
38   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ WesternMuskogean.java, line 32 (Poor Style: Confusing Naming)	Low

Issue Details

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/WesternMuskogean.java, line 32 (Poor Style: Confusing Naming)	Low

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/WesternMuskogean.java:32
Taint Flags:

```
29   X_CIC("x-CIC"),
30   @XmlEnumValue("x-CCT")
31   X_CCT("x-CCT");
32   private final String value;
33
34   WesternMuskogean(String v) {
35   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/SerranoGabrielino.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/SerranoGabrielino.java:29
Taint Flags:

```
26
27   @XmlEnumValue("x-SER")
28   X_SER("x-SER");
29   private final String value;
30
31   SerranoGabrielino(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Maiduan.java, line 35 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ Maiduan.java, line 35 (Poor Style: Confusing Naming)	Low

File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Maiduan.java:35
Taint Flags:

```
32   X_NMU("x-NMU"),
33   @XmlEnumValue("x-MAI")
34   X_MAI("x-MAI");
35   private final String value;
36
37   Maiduan(String v) {
38   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ Coosan.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Coosan.java:29
Taint Flags:

```
26
27   @XmlEnumValue("x-COS")
28   X_COS("x-COS");
29   private final String value;
30
31   Coosan(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ WesternApachean.java, line 35 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/WesternApachean.java:35
Taint Flags:

```
32   X_APM("x-APM"),
33   @XmlEnumValue("x-APW")
34   X_APW("x-APW");
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ WesternApachean.java, line 35 (Poor Style: Confusing Naming)	Low

```
35   private final String value;
36
37   WesternApachean(String v) {
38   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ Charset.java, line 55 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Charset.java:55
Taint Flags:

```
52   UTF_7("UTF-7"),
53   @XmlEnumValue("UTF-8")
54   UTF_8("UTF-8");
55   private final String value;
56
57   Charset(String v) {
58   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ PermanentDentition.java, line 158 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/PermanentDentition.java:158
Taint Flags:

```
155   TID_8("TID8"),
156   @XmlEnumValue("TID9")
157   TID_9("TID9");
158   private final String value;
159
160   PermanentDentition(String v) {
161   value = v;
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ KoyukonIngalik.java, line 35 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/KoyukonIngalik.java:35
Taint Flags:

```
32   X_HOI("x-HOI"),
33   @XmlEnumValue("x-KOY")
34   X_KOY("x-KOY");
35   private final String value;
36
37   KoyukonIngalik(String v) {
38   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ConceptPropertyId.java, line 43 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ConceptPropertyId.java:43
Taint Flags:

```
40   INVERSE_RELATIONSHIP("inverseRelationship"),
41   @XmlEnumValue("openIssue")
42   OPEN_ISSUE("openIssue");
43   private final String value;
44
45   ConceptPropertyId(String v) {
46   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ XActStatusActiveSuspended.java, line 32 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ XActStatusActiveSuspended.java, line 32 (Poor Style: Confusing Naming)	Low

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/XActStatusActiveSuspended.java:32
Taint Flags:

```
29   ACTIVE("active"),
30   @XmlEnumValue("suspended")
31   SUSPENDED("suspended");
32   private final String value;
33
34   XActStatusActiveSuspended(String v) {
35   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ParameterizedDataTypeEventRelatedInterval.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ParameterizedDataTypeEventRelatedInterval.java:29
Taint Flags:

```
26
27   @XmlEnumValue("EIVL<T>")
28   EIVL_T("EIVL<T>");
29   private final String value;
30
31   ParameterizedDataTypeEventRelatedInterval(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ CentralSalish.java, line 35 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/CentralSalish.java:35
Taint Flags:

```
32   X_LUT("x-LUT"),
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ CentralSalish.java, line 35 (Poor Style: Confusing Naming)	Low

```
33   @XmlEnumValue("x-STR")
34   X_STR("x-STR");
35   private final String value;
36
37   CentralSalish(String v) {
38   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ AudioMediaType.java, line 35 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/AudioMediaType.java:35
Taint Flags:

```
32   AUDIO_K_32_ADPCM("audio/k32adpcm"),
33   @XmlEnumValue("audio/mpeg")
34   AUDIO_MPEG("audio/mpeg");
35   private final String value;
36
37   AudioMediaType(String v) {
38   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ MissouriRiver.java, line 32 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/MissouriRiver.java:32
Taint Flags:

```
29   X_CRO("x-CRO"),
30   @XmlEnumValue("x-HID")
31   X_HID("x-HID");
32   private final String value;
33
34   MissouriRiver(String v) {
35   value = v;
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ MissouriRiver.java, line 32 (Poor Style: Confusing Naming)	Low

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ TableFrame.java, line 53 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/TableFrame.java:53
Taint Flags:

```
50  VOID("void"),
51  @XmlEnumValue("vsides")
52  VSIDES("vsides");
53  private final String value;
54
55  TableFrame(String v) {
56  value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ SupernumeraryTooth.java, line 182 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/SupernumeraryTooth.java:182
Taint Flags:

```
179  TID_SS("TIDSs"),
180  @XmlEnumValue("TIDTs")
181  TID_TS("TIDTs");
182  private final String value;
183
184  SupernumeraryTooth(String v) {
185  value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ Chimakuan.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Chimakuan.java, line 29 (Poor Style: Confusing Naming)	Low

Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Chimakuan.java:29
Taint Flags:

```
26
27   @XmlEnumValue("x-QUI")
28   X_QUI("x-QUI");
29   private final String value;
30
31   Chimakuan(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/PaymentTerms.java, line 37 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/PaymentTerms.java:37
Taint Flags:

```
34   N_60("N60"),
35   @XmlEnumValue("N90")
36   N_90("N90");
37   private final String value;
38
39   PaymentTerms(String v) {
40   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/SouthernNumic.java, line 32 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/SouthernNumic.java:32

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ SouthernNumic.java, line 32 (Poor Style: Confusing Naming)	Low

Taint Flags:

```
29   X_KAW("x-KAW"),
30   @XmlEnumValue("x-UTE")
31   X_UTE("x-UTE");
32   private final String value;
33
34   SouthernNumic(String v) {
35   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ Diegueno.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Diegueno.java:29
Taint Flags:

```
26
27   @XmlEnumValue("x-DIH")
28   X_DIH("x-DIH");
29   private final String value;
30
31   Diegueno(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ Aleut.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Aleut.java:29
Taint Flags:

```
26
27   @XmlEnumValue("x-ALW")
28   X_ALW("x-ALW");
29   private final String value;
```

FORTIFY®

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ Aleut.java, line 29 (Poor Style: Confusing Naming)	Low

```
30
31   Aleut(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeSetOfCodedSimpleValue.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/DataTypeSetOfCodedSimpleValue.java:
29
Taint Flags:

```
26
27   @XmlEnumValue("SET<CS>")
28   SET_CS("SET<CS>");
29   private final String value;
30
31   DataTypeSetOfCodedSimpleValue(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeNonParametricProbabilityDistributionOfIntervalOfPhysicalQuantities.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/
DataTypeNonParametricProbabilityDistributionOfIntervalOfPhysicalQuantities.java:29
Taint Flags:

```
26
27   @XmlEnumValue("NPPD<IVL<PQ>>")
28   NPPD_IVL_PQ("NPPD<IVL<PQ>>");
29   private final String value;
30
31   DataTypeNonParametricProbabilityDistributionOfIntervalOfPhysicalQuantities(String v) {
32   value = v;
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeNonParametricProbabilityDistributionOfIntervalOfPhysicalQuantities.java, line 29 (Poor Style: Confusing Naming)	Low

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ OrderedListStyle.java, line 41 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/OrderedListStyle.java:41
Taint Flags:

```
38   LITTLE_ALPHA("LittleAlpha"),
39   @XmlEnumValue("LittleRoman")
40   LITTLE_ROMAN("LittleRoman");
41   private final String value;
42
43   OrderedListStyle(String v) {
44   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ RoleStatusNormal.java, line 44 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/RoleStatusNormal.java:44
Taint Flags:

```
41   SUSPENDED("suspended"),
42   @XmlEnumValue("terminated")
43   TERMINATED("terminated");
44   private final String value;
45
46   RoleStatusNormal(String v) {
47   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ QueryStatusCode.java, line 41 (Poor Style: Confusing Naming)	Low

Issue Details

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ QueryStatusCode.java, line 41 (Poor Style: Confusing Naming)	Low

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/QueryStatusCode.java:41
Taint Flags:

```
38   NEW("new"),
39   @XmlEnumValue("waitContinuedQueryResponse")
40   WAIT_CONTINUED_QUERY_RESPONSE("waitContinuedQueryResponse");
41   private final String value;
42
43   QueryStatusCode(String v) {
44   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ InteriorSalish.java, line 38 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/InteriorSalish.java:38
Taint Flags:

```
35   X_FLA("x-FLA"),
36   @XmlEnumValue("x-OKA")
37   X_OKA("x-OKA");
38   private final String value;
39
40   InteriorSalish(String v) {
41   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ EasternMiwok.java, line 38 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ EasternMiwok.java, line 38 (Poor Style: Confusing Naming)	Low

File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/EasternMiwok.java:38
Taint Flags:

```
35   X_PMW("x-PMW"),
36   @XmlEnumValue("x-SKD")
37   X_SKD("x-SKD");
38   private final String value;
39
40   EasternMiwok(String v) {
41   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeIntervalOfIntegerNumbers.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/DataTypeIntervalOfIntegerNumbers.java:29
Taint Flags:

```
26
27   @XmlEnumValue("IVL<INT>")
28   IVL_INT("IVL<INT>");
29   private final String value;
30
31   DataTypeIntervalOfIntegerNumbers(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ FontStyle.java, line 38 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/FontStyle.java:38
Taint Flags:

```
35   ITALICS("italics"),
36   @XmlEnumValue("underline")
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ FontStyle.java, line 38 (Poor Style: Confusing Naming)	Low

```
37   UNDERLINE("underline");
38   private final String value;
39
40   FontStyle(String v) {
41   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ LocalMarkupIgnore.java, line 32 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/LocalMarkupIgnore.java:32
Taint Flags:

```
29   ALL("all"),
30   @XmlEnumValue("markup")
31   MARKUP("markup");
32   private final String value;
33
34   LocalMarkupIgnore(String v) {
35   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ActSpecObsDilutionCode.java, line 38 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ActSpecObsDilutionCode.java:38
Taint Flags:

```
35   AUTO_LOW("AUTO-LOW"),
36   PRE("PRE"),
37   RERUN("RERUN");
38   private final String value;
39
40   ActSpecObsDilutionCode(String v) {
41   value = v;
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ SouthernCaddoan.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/SouthernCaddoan.java:29
Taint Flags:

```
26
27   @XmlEnumValue("x-CAD")
28   X_CAD("x-CAD");
29   private final String value;
30
31   SouthernCaddoan(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ExtendedReleaseSuspension.java, line 34 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ExtendedReleaseSuspension.java:34
Taint Flags:

```
31   ERSUSP_12("ERSUSP12"),
32   @XmlEnumValue("ERSUSP24")
33   ERSUSP_24("ERSUSP24");
34   private final String value;
35
36   ExtendedReleaseSuspension(String v) {
37   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeSetOfCodedValue.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeSetOfCodedValue.java, line 29 (Poor Style: Confusing Naming)	Low

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/DataTypeSetOfCodedValue.java:29
Taint Flags:

```
26
27   @XmlEnumValue("SET<CV>")
28   SET_CV("SET<CV>");
29   private final String value;
30
31   DataTypeSetOfCodedValue(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ImageMediaType.java, line 41 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ImageMediaType.java:41
Taint Flags:

```
38   IMAGE_PNG("image/png"),
39   @XmlEnumValue("image/tiff")
40   IMAGE_TIFF("image/tiff");
41   private final String value;
42
43   ImageMediaType(String v) {
44   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ExtendedReleaseTablet.java, line 36 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ExtendedReleaseTablet.java:36
Taint Flags:

```
33   @XmlEnumValue("ERTAB24")
34   ERTAB_24("ERTAB24"),
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ExtendedReleaseTablet.java, line 36 (Poor Style: Confusing Naming)	Low

```
35   ERECTAB("ERECTAB");
36   private final String value;
37
38   ExtendedReleaseTablet(String v) {
39   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ Kalapuyan.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Kalapuyan.java:29
Taint Flags:

```
26
27   @XmlEnumValue("x-KAL")
28   X_KAL("x-KAL");
29   private final String value;
30
31   Kalapuyan(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ParameterizedDataTypeBag.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ParameterizedDataTypeBag.java:29
Taint Flags:

```
26
27   @XmlEnumValue("BAG<T>")
28   BAG_T("BAG<T>");
29   private final String value;
30
31   ParameterizedDataTypeBag(String v) {
32   value = v;
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeIntervalOfRealNumbers.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/DataTypeIntervalOfRealNumbers.java: 29
Taint Flags:

```
26
27   @XmlEnumValue("IVL<REAL>")
28   IVL_REAL("IVL<REAL>");
29   private final String value;
30
31   DataTypeIntervalOfRealNumbers(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ExtendedReleaseCapsule.java, line 36 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ExtendedReleaseCapsule.java:36
Taint Flags:

```
33   @XmlEnumValue("ERCAP24")
34   ERCAP_24("ERCAP24"),
35   ERECCAP("ERECCAP");
36   private final String value;
37
38   ExtendedReleaseCapsule(String v) {
39   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ Delawaran.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ Delawaran.java, line 29 (Poor Style: Confusing Naming)	Low

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Delawaran.java:29
Taint Flags:

```
26
27   @XmlEnumValue("x-DEL")
28   X_DEL("x-DEL");
29   private final String value;
30
31   Delawaran(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ Shasta.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Shasta.java:29
Taint Flags:

```
26
27   @XmlEnumValue("x-SHT")
28   X_SHT("x-SHT");
29   private final String value;
30
31   Shasta(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ AcknowledgementDetailNotSupportedCode.java, line 44 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/
AcknowledgementDetailNotSupportedCode.java:44
Taint Flags:

```
41   NS_202("NS202"),
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ AcknowledgementDetailNotSupportedCode.java, line 44 (Poor Style: Confusing Naming)	Low

```
42   @XmlEnumValue("NS203")
43   NS_203("NS203");
44   private final String value;
45
46   AcknowledgementDetailNotSupportedCode(String v) {
47   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ParameterizedDataTypeSequence.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ParameterizedDataTypeSequence.java:29
Taint Flags:

```
26
27   @XmlEnumValue("LIST<T>")
28   LIST_T("LIST<T>");
29   private final String value;
30
31   ParameterizedDataTypeSequence(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeSetOfCodedWithEquivalents.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeSetOfCodedWithEquivalents.java:29
Taint Flags:

```
26
27   @XmlEnumValue("SET<CE>")
28   SET_CE("SET<CE>");
29   private final String value;
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeSetOfCodedWithEquivalents.java, line 29 (Poor Style: Confusing Naming)	Low

```
30
31   DataTypeSetOfCodedWithEquivalents(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ActNonObservationIndicationCode.java, line 41 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ActNonObservationIndicationCode.java:
41
Taint Flags:

```
38   IND_03("IND03"),
39   @XmlEnumValue("IND04")
40   IND_04("IND04");
41   private final String value;
42
43   ActNonObservationIndicationCode(String v) {
44   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ EntityStatusNormal.java, line 35 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/EntityStatusNormal.java:35
Taint Flags:

```
32   ACTIVE("active"),
33   @XmlEnumValue("inactive")
34   INACTIVE("inactive");
35   private final String value;
36
37   EntityStatusNormal(String v) {
38   value = v;
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ HL7DefinedRoseProperty.java, line 120 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/HL7DefinedRoseProperty.java:120
Taint Flags:

```
117  VOCAB_DOMAIN("Vocab_domain"),
118  @XmlEnumValue("Vocab_strength")
119  VOCAB_STRENGTH("Vocab_strength");
120  private final String value;
121
122  HL7DefinedRoseProperty(String v) {
123  value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ProbabilityDistributionType.java, line 45 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ProbabilityDistributionType.java:45
Taint Flags:

```
42  LN("LN"),
43  N("N"),
44  U("U");
45  private final String value;
46
47  ProbabilityDistributionType(String v) {
48  value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ HtmlLinkType.java, line 71 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ HtmlLinkType.java, line 71 (Poor Style: Confusing Naming)	Low

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/HtmlLinkType.java:71
Taint Flags:

```
68   STYLESHEET("stylesheet"),
69   @XmlEnumValue("subsection")
70   SUBSECTION("subsection");
71   private final String value;
72
73   HtmlLinkType(String v) {
74   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ CodeSystem.java, line 709 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/CodeSystem.java:709
Taint Flags:

```
706   W_4("W4"),
707   @XmlEnumValue("W1-W2")
708   W_1_W_2("W1-W2");
709   private final String value;
710
711   CodeSystem(String v) {
712   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ Cahitan.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Cahitan.java:29
Taint Flags:

```
26
27   @XmlEnumValue("x-YAQ")
```

Poor Style: Confusing Naming	Low

| Package: gov.va.med.pharmacy.wsclients.mvi | |

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ Cahitan.java, line 29 (Poor Style: Confusing Naming)	Low

```
28   X_YAQ("x-YAQ");
29   private final String value;
30
31   Cahitan(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ Dhegiha.java, line 38 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Dhegiha.java:38
Taint Flags:

```
35   X_OSA("x-OSA"),
36   @XmlEnumValue("x-QUA")
37   X_QUA("x-QUA");
38   private final String value;
39
40   Dhegiha(String v) {
41   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeUncertainNarrativeConceptDescriptor.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeUncertainNarrativeConceptDescriptor.java:29
Taint Flags:

```
26
27   @XmlEnumValue("UVN<CD>")
28   UVN_CD("UVN<CD>");
29   private final String value;
30
31   DataTypeUncertainNarrativeConceptDescriptor(String v) {
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeUncertainNarrativeConceptDescriptor.java, line 29 (Poor Style: Confusing Naming)	Low

```
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ArapahoGrosVentre.java, line 32 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ArapahoGrosVentre.java:32
Taint Flags:

```
29   X_ARP("x-ARP"),
30   @XmlEnumValue("x-ATS")
31   X_ATS("x-ATS");
32   private final String value;
33
34   ArapahoGrosVentre(String v) {
35   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ Ojibwayan.java, line 32 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Ojibwayan.java:32
Taint Flags:

```
29   X_POT("x-POT"),
30   @XmlEnumValue("x-OJB")
31   X_OJB("x-OJB");
32   private final String value;
33
34   Ojibwayan(String v) {
35   value = v;
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ TableCellScope.java, line 38 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/TableCellScope.java:38
Taint Flags:

```
35   ROW("row"),
36   @XmlEnumValue("rowgroup")
37   ROWGROUP("rowgroup");
38   private final String value;
39
40   TableCellScope(String v) {
41   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ TransdermalPatch.java, line 42 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/TransdermalPatch.java:42
Taint Flags:

```
39   @XmlEnumValue("TPATH2WK")
40   TPATH_2_WK("TPATH2WK"),
41   TPATHWK("TPATHWK");
42   private final String value;
43
44   TransdermalPatch(String v) {
45   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeUncertainProbabilisticIntervalOfPhysicalQuantities.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeUncertainProbabilisticIntervalOfPhysicalQuantities.java, line 29 (Poor Style: Confusing Naming)	Low

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/
DataTypeUncertainProbabilisticIntervalOfPhysicalQuantities.java:29
Taint Flags:

```
26
27   @XmlEnumValue("UVP<IVL<PQ>>")
28   UVP_IVL_PQ("UVP<IVL<PQ>>");
29   private final String value;
30
31   DataTypeUncertainProbabilisticIntervalOfPhysicalQuantities(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ TableRules.java, line 41 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/TableRules.java:41
Taint Flags:

```
38   NONE("none"),
39   @XmlEnumValue("rows")
40   ROWS("rows");
41   private final String value;
42
43   TableRules(String v) {
44   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ BinaryDataEncoding.java, line 31 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/BinaryDataEncoding.java:31
Taint Flags:

```
28   @XmlEnumValue("B64")
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ BinaryDataEncoding.java, line 31 (Poor Style: Confusing Naming)	Low

```
29   B_64("B64"),
30   TXT("TXT");
31   private final String value;
32
33   BinaryDataEncoding(String v) {
34   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ParameterizedDataTypeParametricProbabilityDistribution.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ParameterizedDataTypeParametricProbabilityDistribution.java:29
Taint Flags:

```
26
27   @XmlEnumValue("PPD<T>")
28   PPD_T("PPD<T>");
29   private final String value;
30
31   ParameterizedDataTypeParametricProbabilityDistribution(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeNonParametricProbabilityDistribution.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeNonParametricProbabilityDistribution.java:29
Taint Flags:

```
26
27   @XmlEnumValue("NPPD<ANY>")
28   NPPD_ANY("NPPD<ANY>");
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeNonParametricProbabilityDistribution.java, line 29 (Poor Style: Confusing Naming)	Low

```
29   private final String value;
30
31   DataTypeNonParametricProbabilityDistribution(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ SaukFoxKickapoo.java, line 35 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/SaukFoxKickapoo.java:35
Taint Flags:

```
32   X_SAC("x-SAC"),
33   @XmlEnumValue("x-SJW")
34   X_SJW("x-SJW");
35   private final String value;
36
37   SaukFoxKickapoo(String v) {
38   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ AcknowledgementDetailSyntaxErrorCode.java, line 49 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/
AcknowledgementDetailSyntaxErrorCode.java:49
Taint Flags:

```
46   SYN_101("SYN101"),
47   @XmlEnumValue("SYN103")
48   SYN_103("SYN103");
49   private final String value;
50
51   AcknowledgementDetailSyntaxErrorCode(String v) {
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ AcknowledgementDetailSyntaxErrorCode.java, line 49 (Poor Style: Confusing Naming)	Low

```
52   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ Tepiman.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Tepiman.java:29
Taint Flags:

```
26
27   @XmlEnumValue("x-PAP")
28   X_PAP("x-PAP");
29   private final String value;
30
31   Tepiman(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ Cupan.java, line 35 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Cupan.java:35
Taint Flags:

```
32   X_CUP("x-CUP"),
33   @XmlEnumValue("x-LUI")
34   X_LUI("x-LUI");
35   private final String value;
36
37   Cupan(String v) {
38   value = v;
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeNonParametricProbabilityDistributionOfConceptDescriptor.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeNonParametricProbabilityDistributionOfConceptDescriptor.java:29
Taint Flags:

```
26
27   @XmlEnumValue("NPPD<CD>")
28   NPPD_CD("NPPD<CD>");
29   private final String value;
30
31   DataTypeNonParametricProbabilityDistributionOfConceptDescriptor(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ CombinedPharmacyOrderSuspendReasonCode.java, line 53 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ CombinedPharmacyOrderSuspendReasonCode.java:53
Taint Flags:

```
50   CLARIF("CLARIF"),
51   ALTCHOICE("ALTCHOICE"),
52   WASHOUT("WASHOUT");
53   private final String value;
54
55   CombinedPharmacyOrderSuspendReasonCode(String v) {
56   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeSetOfRealNumbers.java, line 32 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeSetOfRealNumbers.java, line 32 (Poor Style: Confusing Naming)	Low

Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/DataTypeSetOfRealNumbers.java:32
Taint Flags:

```
29   SET_REAL("SET<REAL>"),
30   @XmlEnumValue("IVL<REAL>")
31   IVL_REAL("IVL<REAL>");
32   private final String value;
33
34   DataTypeSetOfRealNumbers(String v) {
35   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ SetOperator.java, line 39 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/SetOperator.java:39
Taint Flags:

```
36   I("I"),
37   A("A"),
38   P("P");
39   private final String value;
40
41   SetOperator(String v) {
42   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ Pidgin.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Pidgin.java:29

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Pidgin.java, line 29 (Poor Style: Confusing Naming)	Low

Taint Flags:

```
26
27   @XmlEnumValue("x-CHH")
28   X_CHH("x-CHH");
29   private final String value;
30
31   Pidgin(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Abenakian.java, line 35 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Abenakian.java:35
Taint Flags:

```
32   X_MAC("x-MAC"),
33   @XmlEnumValue("x-ABE")
34   X_ABE("x-ABE");
35   private final String value;
36
37   Abenakian(String v) {
38   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/NonDrugAgentEntity.java, line 77 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/NonDrugAgentEntity.java:77
Taint Flags:

```
74   NDA_11("NDA11"),
75   @XmlEnumValue("NDA07")
76   NDA_07("NDA07");
77   private final String value;
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ NonDrugAgentEntity.java, line 77 (Poor Style: Confusing Naming)	Low

```
78
79   NonDrugAgentEntity(String v) {
80   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ Dakotan.java, line 38 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Dakotan.java:38
Taint Flags:

```
35   X_LKT("x-LKT"),
36   @XmlEnumValue("x-NKT")
37   X_NKT("x-NKT");
38   private final String value;
39
40   Dakotan(String v) {
41   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ River.java, line 35 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/River.java:35
Taint Flags:

```
32   X_MOV("x-MOV"),
33   @XmlEnumValue("x-YUM")
34   X_YUM("x-YUM");
35   private final String value;
36
37   River(String v) {
38   value = v;
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ XPhoneURLScheme.java, line 32 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/XPhoneURLScheme.java:32
Taint Flags:

```
29   FAX("fax"),
30   @XmlEnumValue("tel")
31   TEL("tel");
32   private final String value;
33
34   XPhoneURLScheme(String v) {
35   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ Cree.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Cree.java:29
Taint Flags:

```
26
27   @XmlEnumValue("x-CRP")
28   X_CRP("x-CRP");
29   private final String value;
30
31   Cree(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ParameterizedDataTypeAnnotated.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ParameterizedDataTypeAnnotated.java, line 29 (Poor Style: Confusing Naming)	Low

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ParameterizedDataTypeAnnotated.java: 29
Taint Flags:

```
26
27   @XmlEnumValue("ANT<T>")
28   ANT_T("ANT<T>");
29   private final String value;
30
31   ParameterizedDataTypeAnnotated(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeBagOfConceptDescriptors.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/DataTypeBagOfConceptDescriptors.java: 29
Taint Flags:

```
26
27   @XmlEnumValue("BAG<CD>")
28   BAG_CD("BAG<CD>");
29   private final String value;
30
31   DataTypeBagOfConceptDescriptors(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ModelMediaType.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ModelMediaType.java:29
Taint Flags:

```
26
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ModelMediaType.java, line 29 (Poor Style: Confusing Naming)	Low

```
27   @XmlEnumValue("model/vrml")
28   MODEL_VRML("model/vrml");
29   private final String value;
30
31   ModelMediaType(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Tanana.java, line 35 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Tanana.java:35
Taint Flags:

```
32   X_TCB("x-TCB"),
33   @XmlEnumValue("x-TAU")
34   X_TAU("x-TAU");
35   private final String value;
36
37   Tanana(String v) {
38   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/DataTypeSetOfIntegerNumbers.java, line 32 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/DataTypeSetOfIntegerNumbers.java:32
Taint Flags:

```
29   SET_INT("SET<INT>"),
30   @XmlEnumValue("IVL<INT>")
31   IVL_INT("IVL<INT>");
32   private final String value;
33
34   DataTypeSetOfIntegerNumbers(String v) {
35   value = v;
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeSetOfIntegerNumbers.java, line 32 (Poor Style: Confusing Naming)	Low

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ UnorderedListStyle.java, line 35 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/UnorderedListStyle.java:35
Taint Flags:

```
32   DISC("Disc"),
33   @XmlEnumValue("Square")
34   SQUARE("Square");
35   private final String value;
36
37   UnorderedListStyle(String v) {
38   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ActStatusNormal.java, line 50 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ActStatusNormal.java:50
Taint Flags:

```
47   NEW("new"),
48   @XmlEnumValue("suspended")
49   SUSPENDED("suspended");
50   private final String value;
51
52   ActStatusNormal(String v) {
53   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeAnnotatedPhysicalQuantity.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeAnnotatedPhysicalQuantity.java, line 29 (Poor Style: Confusing Naming)	Low

Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/
DataTypeAnnotatedPhysicalQuantity.java:29
Taint Flags:

```
26
27   @XmlEnumValue("ANT<PQ>")
28   ANT_PQ("ANT<PQ>");
29   private final String value;
30
31   DataTypeAnnotatedPhysicalQuantity(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ UpperChinook.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/UpperChinook.java:29
Taint Flags:

```
26
27   @XmlEnumValue("x-WAC")
28   X_WAC("x-WAC");
29   private final String value;
30
31   UpperChinook(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ GTSAbbreviationHolidaysUSNational.java, line 59 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ GTSAbbreviationHolidaysUSNational.java, line 59 (Poor Style: Confusing Naming)	Low

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ GTSAbbreviationHolidaysUSNational.java:59
Taint Flags:

```
56  @XmlEnumValue("JHNUSTKS5")
57  JHNUSTKS_5("JHNUSTKS5"),
58  JHNUSVET("JHNUSVET");
59  private final String value;
60
61  GTSAbbreviationHolidaysUSNational(String v) {
62  value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ MaterialEntityAdditive.java, line 147 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/MaterialEntityAdditive.java:147
Taint Flags:

```
144  TOLU("TOLU"),
145  URETM("URETM"),
146  VIRTM("VIRTM");
147  private final String value;
148
149  MaterialEntityAdditive(String v) {
150  value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ParameterizedDataTypeUncertainValueNarrative.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ParameterizedDataTypeUncertainValueNarrative.java:29

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ParameterizedDataTypeUncertainValueNarrative.java, line 29 (Poor Style: Confusing Naming)	Low

Taint Flags:

```
26
27  @XmlEnumValue("UVN<T>")
28  UVN_T("UVN<T>");
29  private final String value;
30
31  ParameterizedDataTypeUncertainValueNarrative(String v) {
32  value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ EasternApachean.java, line 32 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/EasternApachean.java:32
Taint Flags:

```
29  X_APJ("x-APJ"),
30  @XmlEnumValue("x-APL")
31  X_APL("x-APL");
32  private final String value;
33
34  EasternApachean(String v) {
35  value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ CaliforniaAthapaskan.java, line 32 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/CaliforniaAthapaskan.java:32
Taint Flags:

```
29  X_KTW("x-KTW"),
30  @XmlEnumValue("x-HUP")
31  X_HUP("x-HUP");
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ CaliforniaAthapaskan.java, line 32 (Poor Style: Confusing Naming)	Low

```
32   private final String value;
33
34   CaliforniaAthapaskan(String v) {
35   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ TableRuleStyle.java, line 38 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/TableRuleStyle.java:38
Taint Flags:

```
35   RRULE("Rrule"),
36   @XmlEnumValue("Toprule")
37   TOPRULE("Toprule");
38   private final String value;
39
40   TableRuleStyle(String v) {
41   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ Wintuan.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Wintuan.java:29
Taint Flags:

```
26
27   @XmlEnumValue("x-WIT")
28   X_WIT("x-WIT");
29   private final String value;
30
31   Wintuan(String v) {
32   value = v;
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ VideoMediaType.java, line 32 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/VideoMediaType.java:32
Taint Flags:

```
29   VIDEO_MPEG("video/mpeg"),
30   @XmlEnumValue("video/x-avi")
31   VIDEO_X_AVI("video/x-avi");
32   private final String value;
33
34   VideoMediaType(String v) {
35   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ MultipartMediaType.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/MultipartMediaType.java:29
Taint Flags:

```
26
27   @XmlEnumValue("multipart/x-hl7-cda-level-one")
28   MULTIPART_X_HL_7_CDA_LEVEL_ONE("multipart/x-hl7-cda-level-one");
29   private final String value;
30
31   MultipartMediaType(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ NorthernIroquoian.java, line 44 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Poor Style: Confusing Naming	Low
Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ NorthernIroquoian.java, line 44 (Poor Style: Confusing Naming)	Low

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/NorthernIroquoian.java:44
Taint Flags:

```
41  X_SEE("x-SEE"),
42  @XmlEnumValue("x-TUS")
43  X_TUS("x-TUS");
44  private final String value;
45
46  NorthernIroquoian(String v) {
47  value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeHistoryOfAddress.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/DataTypeHistoryOfAddress.java:29
Taint Flags:

```
26
27  @XmlEnumValue("HIST<AD>")
28  HIST_AD("HIST<AD>");
29  private final String value;
30
31  DataTypeHistoryOfAddress(String v) {
32  value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ CentralMuskogean.java, line 35 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/CentralMuskogean.java:35
Taint Flags:

```
32  X_CKU("x-CKU"),
33  @XmlEnumValue("x-MIK")
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/CentralMuskogean.java, line 35 (Poor Style: Confusing Naming)	Low

```
34   X_MIK("x-MIK");
35   private final String value;
36
37   CentralMuskogean(String v) {
38   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/OtherIndicationValue.java, line 35 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/OtherIndicationValue.java:35
Taint Flags:

```
32   INDICATION_VALUE("_IndicationValue"),
33   @XmlEnumValue("_SymptomValue")
34   SYMPTOM_VALUE("_SymptomValue");
35   private final String value;
36
37   OtherIndicationValue(String v) {
38   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/HL7CommitteeIDInRIM.java, line 59 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/HL7CommitteeIDInRIM.java:59
Taint Flags:

```
56   C_01("C01"),
57   @XmlEnumValue("C21")
58   C_21("C21");
59   private final String value;
60
61   HL7CommitteeIDInRIM(String v) {
62   value = v;
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Yokutsan.java, line 38 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Yokutsan.java:38
Taint Flags:

```
35   X_PYL("x-PYL"),
36   @XmlEnumValue("x-TKH")
37   X_TKH("x-TKH");
38   private final String value;
39
40   Yokutsan(String v) {
41   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/HL7StandardVersionCode.java, line 57 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/HL7StandardVersionCode.java:57
Taint Flags:

```
54   PAST("PAST"),
55   PREVSUM("PREVSUM"),
56   SUM("SUM");
57   private final String value;
58
59   HL7StandardVersionCode(String v) {
60   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/DataTypeSetOfPhysicalQuantities.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeSetOfPhysicalQuantities.java, line 29 (Poor Style: Confusing Naming)	Low

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/DataTypeSetOfPhysicalQuantities.java: 29
Taint Flags:

```
26
27   @XmlEnumValue("SET<PQ>")
28   SET_PQ("SET<PQ>");
29   private final String value;
30
31   DataTypeSetOfPhysicalQuantities(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ParameterizedDataTypePeriodicInterval.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ParameterizedDataTypePeriodicInterval.java:29
Taint Flags:

```
26
27   @XmlEnumValue("PIVL<T>")
28   PIVL_T("PIVL<T>");
29   private final String value;
30
31   ParameterizedDataTypePeriodicInterval(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ Keresan.java, line 32 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Keresan.java:32
Taint Flags:

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ Keresan.java, line 32 (Poor Style: Confusing Naming)	Low

```
29  X_KJQ("x-KJQ"),
30  @XmlEnumValue("x-KEE")
31  X_KEE("x-KEE");
32  private final String value;
33
34  Keresan(String v) {
35  value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ Yukian.java, line 32 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Yukian.java:32
Taint Flags:

```
29  X_WAO("x-WAO"),
30  @XmlEnumValue("x-YUK")
31  X_YUK("x-YUK");
32  private final String value;
33
34  Yukian(String v) {
35  value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeParametricProbabilityDistributionOfRealNumbers.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/
DataTypeParametricProbabilityDistributionOfRealNumbers.java:29
Taint Flags:

```
26
27  @XmlEnumValue("PPD<REAL>")
28  PPD_REAL("PPD<REAL>");
29  private final String value;
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeParametricProbabilityDistributionOfRealNumbers.java, line 29 (Poor Style: Confusing Naming)	Low

```
30
31   DataTypeParametricProbabilityDistributionOfRealNumbers(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeAnnotatedConceptDescriptor.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeAnnotatedConceptDescriptor.java:29
Taint Flags:

```
26
27   @XmlEnumValue("ANT<CD>")
28   ANT_CD("ANT<CD>");
29   private final String value;
30
31   DataTypeAnnotatedConceptDescriptor(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ActAdministrativeRuleDetectedIssueCode.java, line 32 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ActAdministrativeRuleDetectedIssueCode.java:32
Taint Flags:

```
29   KEY_205("KEY205"),
30   @XmlEnumValue("KEY204")
31   KEY_204("KEY204");
32   private final String value;
33
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ActAdministrativeRuleDetectedIssueCode.java, line 32 (Poor Style: Confusing Naming)	Low

```
34   ActAdministrativeRuleDetectedIssueCode(String v) {
35   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ EntityHandling.java, line 63 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/EntityHandling.java:63
Taint Flags:

```
60   NTR("NTR"),
61   PSO("PSO"),
62   UPR("UPR");
63   private final String value;
64
65   EntityHandling(String v) {
66   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ ActRelationshipSplit.java, line 36 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ActRelationshipSplit.java:36
Taint Flags:

```
33   @XmlEnumValue("I1")
34   I_1("I1"),
35   IW("IW");
36   private final String value;
37
38   ActRelationshipSplit(String v) {
39   value = v;
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ SouthernAlaska.java, line 32 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/SouthernAlaska.java:32
Taint Flags:

```
29   X_AHT("x-AHT"),
30   @XmlEnumValue("x-TFN")
31   X_TFN("x-TFN");
32   private final String value;
33
34   SouthernAlaska(String v) {
35   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ HL7ITSVersionCode.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/HL7ITSVersionCode.java:29
Taint Flags:

```
26
27   @XmlEnumValue("XMLV1PR1")
28   XMLV_1_PR_1("XMLV1PR1");
29   private final String value;
30
31   HL7ITSVersionCode(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypePeriodicIntervalOfTime.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypePeriodicIntervalOfTime.java, line 29 (Poor Style: Confusing Naming)	Low

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/DataTypePeriodicIntervalOfTime.java:29
Taint Flags:

```
26
27   @XmlEnumValue("PIVL<TS>")
28   PIVL_TS("PIVL<TS>");
29   private final String value;
30
31   DataTypePeriodicIntervalOfTime(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ Tiwa.java, line 32 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Tiwa.java:32
Taint Flags:

```
29   X_TAO("x-TAO"),
30   @XmlEnumValue("x-TIX")
31   X_TIX("x-TIX");
32   private final String value;
33
34   Tiwa(String v) {
35   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ Ritwan.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/Ritwan.java:29
Taint Flags:

```
26
27   @XmlEnumValue("x-YUR")
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ Ritwan.java, line 29 (Poor Style: Confusing Naming)	Low

```
28   X_YUR("x-YUR");
29   private final String value;
30
31   Ritwan(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ XDocumentStatus.java, line 38 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/XDocumentStatus.java:38
Taint Flags:

```
35   NEW("new"),
36   @XmlEnumValue("obsolete")
37   OBSOLETE("obsolete");
38   private final String value;
39
40   XDocumentStatus(String v) {
41   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ NorthernCaddoan.java, line 35 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/NorthernCaddoan.java:35
Taint Flags:

```
32   X_PAW("x-PAW"),
33   @XmlEnumValue("x-WIC")
34   X_WIC("x-WIC");
35   private final String value;
36
37   NorthernCaddoan(String v) {
38   value = v;
```

Poor Style: Confusing Naming	Low

Package: gov.va.med.pharmacy.wsclients.mvi	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ DataTypeHistoricalAddress.java, line 29 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/DataTypeHistoricalAddress.java:29
Taint Flags:

```
26
27   @XmlEnumValue("HXIT<AD>")
28   HXIT_AD("HXIT<AD>");
29   private final String value;
30
31   DataTypeHistoricalAddress(String v) {
32   value = v;
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/ MdfHmdRowType.java, line 44 (Poor Style: Confusing Naming)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Sink Details

Sink: Field: value
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/wsclients/mvi/MdfHmdRowType.java:44
Taint Flags:

```
41   CLASS("class"),
42   @XmlEnumValue("stc")
43   STC("stc");
44   private final String value;
45
46   MdfHmdRowType(String v) {
47   value = v;
```

Poor Style: Value Never Read (1 issue)

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.
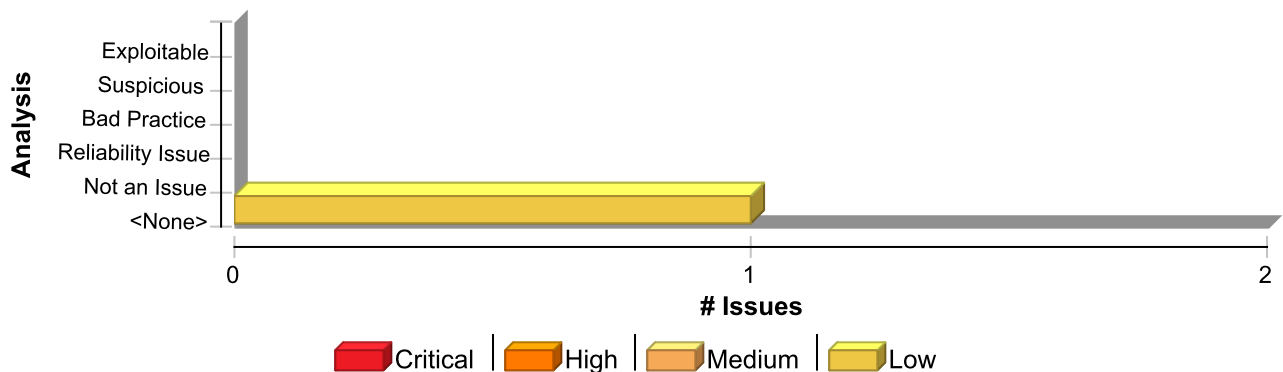
Example: The following code excerpt assigns to the variable `r` and then overwrites the value without using it.

```
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	1	0	0	1
Total	**1**	**0**	**0**	**1**

Poor Style: Value Never Read	Low
Package: gov.va.med.pharmacy.jms.messaging	
PS_INB_ERX_JMS/src/main/java/gov/va/med/pharmacy/jms/messaging/ InboundMsgProducer.java, line 37 (Poor Style: Value Never Read)	Low

Issue Details

Kingdom: Code Quality
Scan Engine: SCA (Structural)

Poor Style: Value Never Read	Low

Package: gov.va.med.pharmacy.jms.messaging	

PS_INB_ERX_JMS/src/main/java/gov/va/med/pharmacy/jms/messaging/ InboundMsgProducer.java, line 37 (Poor Style: Value Never Read)	Low

Sink Details

Sink: VariableAccess: messageRead
Enclosing Method: postProcessMessage()
File: PS_INB_ERX_JMS/src/main/java/gov/va/med/pharmacy/jms/messaging/InboundMsgProducer.java:37
Taint Flags:

```
34   Long length = byteMessage.getBodyLength();

35

36   byte[] newMessage = new byte[length.intValue()];
37   int messageRead = byteMessage.readBytes(newMessage);

38

39   LOG.debug("--- sending message", new String(newMessage));

40
```

Redundant Null Check (2 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	2	0	0	2
Total	**2**	**0**	**0**	**2**

Redundant Null Check	Low

Package: gov.va.med.pharmacy.jaxrs.eAnde.services.impl	

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/eAnde/services/impl/ EESummary_Client.java, line 72 (Redundant Null Check)	Low

Issue Details

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

Sink Details

Sink: Dereferenced : url
Enclosing Method: returnResponse()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/eAnde/services/impl/EESummary_Client.java:72
Taint Flags:

```
69
70   URL wsdlURL = EeSummaryPortService.WSDL_LOCATION;
71
72   if (url.length() > 0 && url != null && !"".equals(url)) {
73   File wsdlFile = new File(url);
74   try {
75   if (wsdlFile.exists()) {
```

Package: gov.va.med.pharmacy.persistence.dao.impl	

PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/dao/impl/ SummaryReportDaoImpl.java, line 97 (Redundant Null Check)	Low

Issue Details

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

Sink Details

Sink: Dereferenced : visn
Enclosing Method: find()
File: PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/dao/impl/SummaryReportDaoImpl.java:97
Taint Flags:

```
94   if (visn != null && visn.length() > 0){
95   visn = visn.substring(0, visn.length()-1);
96   }
97   if (visn.compareTo("") == 0){ //check for All value
98   visn = "%";
99   }
100
```

SQL Injection (1 issue)

Abstract

Constructing a dynamic SQL statement with input coming from an untrusted source could allow an attacker to modify the statement's meaning or to execute arbitrary SQL commands.

Explanation

SQL injection errors occur when:

1. Data enters a program from an untrusted source.

In this case HPE Security Fortify Static Code Analyzer could not determine that the source of the data is trusted.

2. The data is used to dynamically construct a SQL query.

Example 1: The following code dynamically constructs and executes a SQL query that searches for items matching a specified name. The query restricts the items displayed to those where the owner matches the user name of the currently-authenticated user.

```
...
  String userName = ctx.getAuthenticatedUserName();
  String itemName = request.getParameter("itemName");
    String query = "SELECT * FROM items WHERE owner = '"
               + userName + "' AND itemname = '"
            + itemName + "'";
  ResultSet rs = stmt.execute(query);
...
```

The query that this code intends to execute follows:

```
    SELECT * FROM items
    WHERE owner = <userName>
 AND itemname = <itemName>;
```

However, because the query is constructed dynamically by concatenating a constant base query string and a user input string, the query only behaves correctly if `itemName` does not contain a single-quote character. If an attacker with the user name `wiley` enters the string "`name' OR 'a'='a`" for `itemName`, then the query becomes the following:

```
  SELECT * FROM items
   WHERE owner = 'wiley'
 AND itemname = 'name' OR 'a'='a';
```

The addition of the `OR 'a'='a'` condition causes the where clause to always evaluate to true, so the

query becomes logically equivalent to the much simpler query:

```
    SELECT * FROM items;
```

This simplification of the query allows the attacker to bypass the requirement that the query only return items owned by the authenticated user; the query now returns all entries stored in the `items` table, regardless of their specified owner.

Example 2: This example examines the effects of a different malicious value passed to the query constructed and executed in Example 1. If an attacker with the user name `wiley` enters the string "`name'; DELETE FROM items; --`" for `itemName`, then the query becomes the following two queries:

```
 SELECT * FROM items
    WHERE owner = 'wiley'
  AND itemname = 'name';

    DELETE FROM items;

    --'
```

Many database servers, including Microsoft(R) SQL Server 2000, allow multiple SQL statements separated by semicolons to be executed at once. While this attack string results in an error on Oracle and other database servers that do not allow the batch-execution of statements separated by semicolons, on databases that do allow batch execution, this type of attack allows the attacker to execute arbitrary commands against the database.

Notice the trailing pair of hyphens (--), which specifies to most database servers that the remainder of the statement is to be treated as a comment and not executed [4]. In this case the comment character serves to remove the trailing single-quote left over from the modified query. On a database where comments are not allowed to be used in this way, the general attack could still be made effective using a trick similar to the one used in Example 1. If an attacker enters the string "`name'); DELETE FROM items; SELECT * FROM items WHERE 'a'='a`", the following three valid statements will be created:

```
   SELECT * FROM items
    WHERE owner = 'wiley'
  AND itemname = 'name';

   DELETE FROM items;

   SELECT * FROM items WHERE 'a'='a';
```

Some think that in the mobile world, classic web application vulnerabilities, such as SQL injection, do not make sense -- why would the user attack themself? However, keep in mind that the essence of mobile platforms is applications that are downloaded from various sources and run alongside each other on the same device. The likelihood of running a piece of malware next to a banking application is high, which necessitates expanding the attack surface of mobile applications to include inter-process communication.

Example 3: The following code adapts Example 1 to the Android platform.

```
...
        PasswordAuthentication pa = authenticator.getPasswordAuthentication();
        String userName = pa.getUserName();
        String itemName = this.getIntent().getExtras().getString("itemName");
        String query = "SELECT * FROM items WHERE owner = '"
                            + userName + "' AND itemname = '"
                            + itemName + "'";
        SQLiteDatabase db = this.openOrCreateDatabase("DB", MODE_PRIVATE,
null);
        Cursor c = db.rawQuery(query, null);
...
```

One traditional approach to preventing SQL injection attacks is to handle them as an input validation problem and either accept only characters from a whitelist of safe values or identify and escape a blacklist of potentially malicious values. Whitelisting can be a very effective means of enforcing strict input validation rules, but parameterized SQL statements require less maintenance and can offer more guarantees with respect to security. As is almost always the case, blacklisting is riddled with loopholes that make it ineffective at preventing SQL injection attacks. For example, attackers may:

 - Target fields that are not quoted
 - Find ways to bypass the need for certain escaped meta-characters
 - Use stored procedures to hide the injected meta-characters

Manually escaping characters in input to SQL queries can help, but it will not make your application secure from SQL injection attacks.

Another solution commonly proposed for dealing with SQL injection attacks is to use stored procedures. Although stored procedures prevent some types of SQL injection attacks, they fail to protect against many others. Stored procedures typically help prevent SQL injection attacks by limiting the types of statements that can be passed to their parameters. However, there are many ways around the limitations and many interesting statements that can still be passed to stored procedures. Again, stored procedures can prevent some exploits, but they will not make your application secure against SQL injection attacks.

Recommendation

The root cause of a SQL injection vulnerability is the ability of an attacker to change context in the SQL query, causing a value that the programmer intended to be interpreted as data to be interpreted as a command instead. When a SQL query is constructed, the programmer knows what should be interpreted as part of the command and what should be interpreted as data. Parameterized SQL statements can enforce this behavior by disallowing data-directed context changes and preventing nearly all SQL injection attacks. Parameterized SQL statements are constructed using strings of regular SQL, but when user-supplied data needs to be included, they create bind parameters, which are placeholders for data that is subsequently inserted. Bind parameters allow the program to explicitly specify to the database what should be treated as a command and what should be treated as data. When the program is ready to execute a statement, it specifies to the database the runtime values to use for the value of each of the bind parameters, without the risk of the data being interpreted as commands.

Example 1 can be rewritten to use parameterized SQL statements (instead of concatenating user supplied strings) as follows:

```
...
  String userName = ctx.getAuthenticatedUserName();
  String itemName = request.getParameter("itemName");
    String query =
```

```
      "SELECT * FROM items WHERE itemname=? AND owner=?";
   PreparedStatement stmt = conn.prepareStatement(query);
 stmt.setString(1, itemName);
   stmt.setString(2, userName);
   ResultSet results = stmt.execute();
...
```

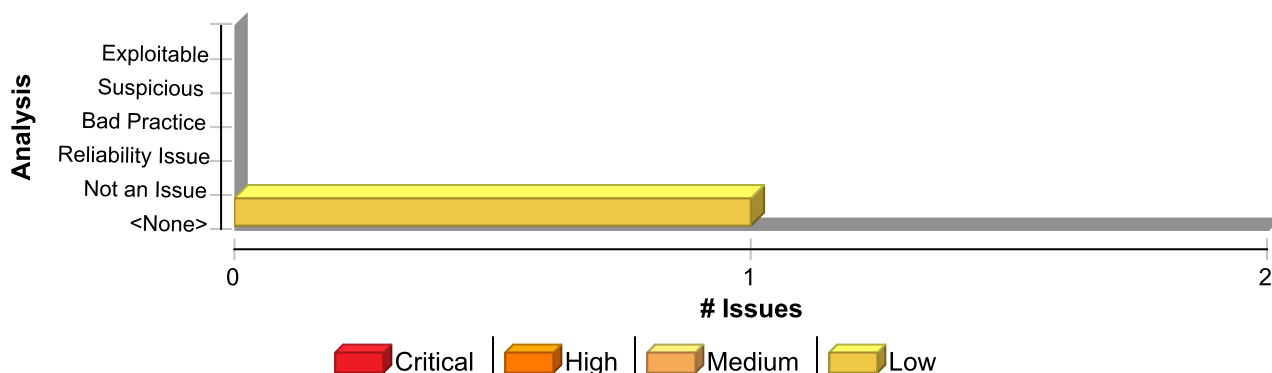And here is an Android equivalent:

```
...
  PasswordAuthentication pa = authenticator.getPasswordAuthentication();
 String userName = pa.getUserName();
    String itemName = this.getIntent().getExtras().getString("itemName");
  String query = "SELECT * FROM items WHERE itemname=? AND owner=?";
 SQLiteDatabase db = this.openOrCreateDatabase("DB", MODE_PRIVATE, null);
   Cursor c = db.rawQuery(query, new Object[]{itemName, userName});
...
```

More complicated scenarios, often found in report generation code, require that user input affect the command structure of the SQL statement, such as the addition of dynamic constraints in the `WHERE` clause. Do not use this requirement to justify concatenating user input into query strings. Prevent SQL injection attacks where user input must affect statement command structure with a level of indirection: create a set of legitimate strings that correspond to different elements you might include in a SQL statement. When constructing a statement, use input from the user to select from this set of application-controlled values.

Issue Summary

Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
SQL Injection	1	0	0	1
Total	**1**	**0**	**0**	**1**

SQL Injection	Low

Package: gov.va.med.pharmacy.persistence.track.impl	

PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/track/impl/ NcpdpMessagesDaoImpl.java, line 660 (SQL Injection)	Low

Issue Details	

SQL Injection	Low
Package: gov.va.med.pharmacy.persistence.track.impl	
PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/track/impl/ NcpdpMessagesDaoImpl.java, line 660 (SQL Injection)	Low

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

Sink Details

Sink: query()
Enclosing Method: searchMessages()
File: PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/track/impl/NcpdpMessagesDaoImpl.java:660
Taint Flags:

```
657  try {
658  LOG.info("Retrieving NCPDP message details.");
659  //System.out.println("Retrieving NCPDP message details.");
660  ncpdpMsgList = jdbcTemplate.query(sql,new NcpdpMsgListRowMapper(),messageId, relatesToId,
     vaStationId, patientLastName, patientFirstName, prescriberNpi, prescribedDrug,
     inboundNcpdpMsgId);
661  } catch (DataAccessException e) {
662  //ncpdpMsg.setDataError(e.getMessage());
663  //System.out.println("data error is:" + e.getMessage());
```

System Information Leak: External (7 issues)

Abstract

Revealing system data or debugging information helps an adversary learn about the system and form a plan of attack.

Explanation

An external information leak occurs when system data or debugging information leaves the program to a remote machine via a socket or network connection. External leaks can help an attacker by revealing specific data about operating systems, full pathnames, the existence of usernames, or locations of configuration files, and are more serious than internal information leaks which are more difficult for an attacker to access.

Example 1: The following code leaks Exception information in the HTTP response:

```
protected void doPost (HttpServletRequest req, HttpServletResponse res) throws
IOException {
    ...
    PrintWriter out = res.getWriter();
    try {
        ...
    } catch (Exception e) {
      out.println(e.getMessage());
    }
}
```

This information can be exposed to a remote user. 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.

In the mobile world, information leaks are also a concern. The essence of mobile platforms is applications that are downloaded from various sources and run alongside each other on the same device. The likelihood of running a piece of malware next to a banking application is high, which is why application authors need to be careful about what information they include in messages addressed to other applications running on the device.

Example 2: The code below broadcasts the stack trace of a caught exception to all the registered Android receivers.

```
...
try {
  ...
} catch (Exception e) {
    String exception = Log.getStackTraceString(e);
    Intent i = new Intent();
    i.setAction("SEND_EXCEPTION");
```

```
        i.putExtra("exception", exception);
        view.getContext().sendBroadcast(i);
}
...
```

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 3: 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";
...
TelephonyManager tm =
(TelephonyManager)Context.getSystemService(Context.TELEPHONY_SERVICE);
String VERSION = tm.getDeviceSoftwareVersion();
...
NfcAdapter nfcAdapter = NfcAdapter.getDefaultAdapter(this);
if (nfcAdapter == null)
  return;

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);
nfcAdapter.setNdefPushMessage(msg, this);
...
```

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.

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. Due to this, it's advised to always keep information instead of sending it to a resource directly outside the program.

Example 4: The code below broadcasts the stack trace of a caught exception within your app only, so that it cannot be leaked to other apps on the system. There is also the added bonus that this is more efficient than globally broadcasting through the system.
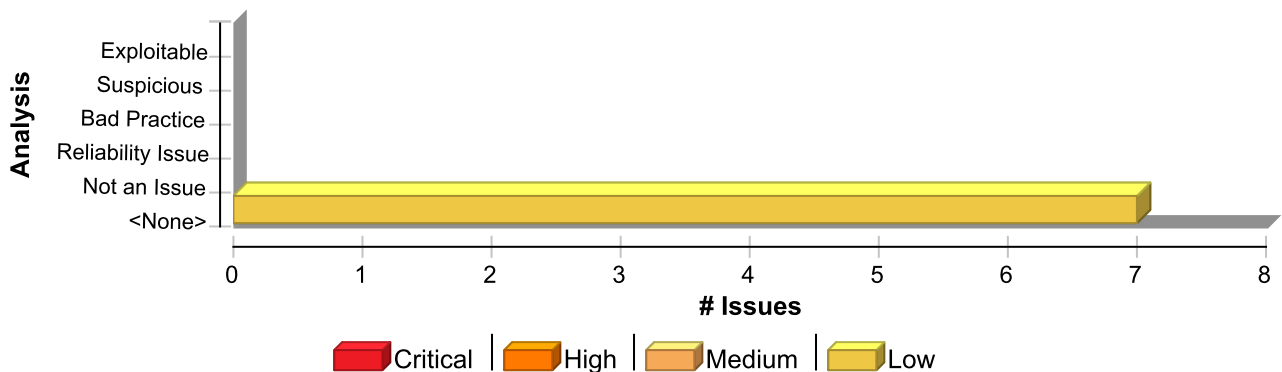
```
...
try {
  ...
} catch (Exception e) {
    String exception = Log.getStackTraceString(e);
    Intent i = new Intent();
    i.setAction("SEND_EXCEPTION");
    i.putExtra("exception", exception);
    LocalBroadcastManager.getInstance(view.getContext()).sendBroadcast(i);
}
...
```

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: External	7	0	0	7
Total	**7**	**0**	**0**	**7**

System Information Leak: External	Low
Package: gov.va.med.pharmacy.jaxrs.drug.service.impl	
PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/drug/service/impl/ DrugSearchImpl.java, line 62 (System Information Leak: External)	**Low**

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Data Flow)

Source Details

Source: java.lang.Throwable.getMessage()
From: gov.va.med.pharmacy.jaxrs.drug.service.impl.DrugSearchImpl.search
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/drug/service/impl/DrugS
earchImpl.java:57

System Information Leak: External	Low

Package: gov.va.med.pharmacy.jaxrs.drug.service.impl	

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/drug/service/impl/ DrugSearchImpl.java, line 62 (System Information Leak: External)	Low

```
54   } catch (Exception e) {
55   searchResponse.setSuccess(false);
56
57   searchResponse.setErrorMessage("Error while searching drug: " +
     e.getMessage());
58
59   LOG.error("Error in DrugSearchImpl:"+e.getMessage());
60   }
```

Sink Details

Sink: Return searchResponse
Enclosing Method: search()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/drug/service/impl/DrugSearchImpl.java:62
Taint Flags: EXCEPTIONINFO, SYSTEMINFO

```
59   LOG.error("Error in DrugSearchImpl:"+e.getMessage());
60   }
61
62   return searchResponse;
63   }
64
65 undefined
```

Package: gov.va.med.pharmacy.jaxrs.mvi.service.impl	

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/service/impl/ PatientSearchImpl.java, line 101 (System Information Leak: External)	Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Data Flow)

Source Details

Source: java.lang.Throwable.getMessage()
From: gov.va.med.pharmacy.jaxrs.mvi.service.impl.PatientSearchImpl.searchPatientByName
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/service/impl/PatientSearchImpl.java:98

```
95
96   searchResponse.setSuccess(false);
97
98   searchResponse.setErrorMessage("Error while searching patient: " +
     e.getMessage());
99   }
```

System Information Leak: External	Low

Package: gov.va.med.pharmacy.jaxrs.mvi.service.impl	

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/service/impl/PatientSearchImpl.java, line 101 (System Information Leak: External)	Low

```
100
101   return searchResponse;
```

Sink Details

Sink: Return searchResponse
Enclosing Method: searchPatientByName()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/mvi/service/impl/PatientSearchImpl.java:101
Taint Flags: EXCEPTIONINFO, SYSTEMINFO

```
98   searchResponse.setErrorMessage("Error while searching patient: " + e.getMessage());
99   }
100
101   return searchResponse;
102   }
103
104 }
```

Package: gov.va.med.pharmacy.jaxrs.outboundncpdpmessage.service.impl	

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/outboundncpdpmessage/service/impl/OutboundNCPDPMessageServiceImpl.java, line 169 (System Information Leak: External)	Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Data Flow)

Source Details

Source: java.lang.Throwable.getMessage()
From: gov.va.med.pharmacy.jaxrs.outboundncpdpmessage.service.impl.OutboundNCPDPMessageServiceImpl.sendOutboundMessage
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/outboundncpdpmessage/service/impl/OutboundNCPDPMessageServiceImpl.java:166

```
163
164   wsResponse.setSuccess(false);
165
166   wsResponse.setErrorMessage("Error while processing outbound message: " + e.getMessage());
167   }
168
169   return wsResponse;
```

Sink Details

System Information Leak: External	Low

Package: gov.va.med.pharmacy.jaxrs.outboundncpdpmessage.service.impl	
PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/outboundncpdpmessage/service/impl/OutboundNCPDPMessageServiceImpl.java, line 169 (System Information Leak: External)	Low

Sink: Return wsResponse
Enclosing Method: sendOutboundMessage()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/outboundncpdpmessage/service/impl/OutboundNCPDPMessageServiceImpl.java:169
Taint Flags: EXCEPTIONINFO, SYSTEMINFO

```
166  wsResponse.setErrorMessage("Error while processing outbound message: " + e.getMessage());
167  }
168
169  return wsResponse;
170  }
171
172 undefined
```

Package: gov.va.med.pharmacy.jaxrs.provider.service.impl	
PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/provider/service/impl/ProviderSearchImpl.java, line 62 (System Information Leak: External)	Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Data Flow)

Source Details

Source: java.lang.Throwable.getMessage()
From: gov.va.med.pharmacy.jaxrs.provider.service.impl.ProviderSearchImpl.search
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/provider/service/impl/ProviderSearchImpl.java:59

```
56
57  searchResponse.setSuccess(false);
58
59  searchResponse.setErrorMessage("Error while searching provider: " +
e.getMessage());
60  }
61
62  return searchResponse;
```

Sink Details

Sink: Return searchResponse
Enclosing Method: search()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/provider/service/impl/ProviderSearchImpl.java:62
Taint Flags: EXCEPTIONINFO, SYSTEMINFO

System Information Leak: External	Low

Package: gov.va.med.pharmacy.jaxrs.provider.service.impl	

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/provider/service/impl/ProviderSearchImpl.java, line 62 (System Information Leak: External)	Low

```
59   searchResponse.setErrorMessage("Error while searching provider: " + e.getMessage());
60   }
61
62   return searchResponse;
63   }
64
65   undefined
```

Package: gov.va.med.pharmacy.jaxrs.tovista.service.impl	

PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/tovista/service/impl/VistaInboundSendImpl.java, line 53 (System Information Leak: External)	Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Data Flow)

Source Details

Source: java.lang.Throwable.getMessage()
From: gov.va.med.pharmacy.jaxrs.tovista.service.impl.VistaInboundSendImpl.send
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/tovista/service/impl/VistaInboundSendImpl.java:50

```
47   } catch (Exception e) {
48   sendResponse.setSuccess(false);
49
50   sendResponse.setErrorMessage("Error while sending vista inbound message: " + e.getMessage());
51   }
52
53   return sendResponse;
```

Sink Details

Sink: Return sendResponse
Enclosing Method: send()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/tovista/service/impl/VistaInboundSendImpl.java:53
Taint Flags: EXCEPTIONINFO, SYSTEMINFO

```
50   sendResponse.setErrorMessage("Error while sending vista inbound message: " + e.getMessage());
51   }
52
53   return sendResponse;
54   }
55
```

System Information Leak: External	Low

Package: gov.va.med.pharmacy.jaxrs.tovista.service.impl	
PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/tovista/service/impl/ VistaInboundSendImpl.java, line 53 (System Information Leak: External)	Low

```
56 undefined
```

Package: gov.va.med.pharmacy.jaxrs.vistaoutboundmsg.service.impl	
PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/vistaoutboundmsg/ service/impl/VistaOutboundMsgImpl.java, line 97 (System Information Leak: External)	Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Data Flow)

Source Details

Source: java.lang.Throwable.getMessage()
From: gov.va.med.pharmacy.jaxrs.vistaoutboundmsg.service.impl.VistaOutboundMsgImpl.
persistMessage
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/vistaoutboundmsg/servic
e/impl/VistaOutboundMsgImpl.java:94

```
91
92   wsResponse.setSuccess(false);
93
94   wsResponse.setErrorMessage("Error while saving vista outbound message: "
+ e.getMessage());
95   }
96
97   return wsResponse;
```

Sink Details

Sink: Return wsResponse
Enclosing Method: persistMessage()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/vistaoutboundmsg/service/impl/
VistaOutboundMsgImpl.java:97
Taint Flags: EXCEPTIONINFO, SYSTEMINFO

```
94   wsResponse.setErrorMessage("Error while saving vista outbound message: " + e.getMessage());
95   }
96
97   return wsResponse;
98   }
99
100  @POST
```

System Information Leak: External	Low
Package: gov.va.med.pharmacy.jaxrs.vistaoutboundmsg.service.impl	
PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/vistaoutboundmsg/service/impl/VistaOutboundMsgImpl.java, line 141 (System Information Leak: External)	Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Data Flow)

Source Details

Source: java.lang.Throwable.getMessage()
From: gov.va.med.pharmacy.jaxrs.vistaoutboundmsg.service.impl.VistaOutboundMsgImpl.
persistXMLMessage
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/vistaoutboundmsg/servic
e/impl/VistaOutboundMsgImpl.java:138

```
135
136   wsResponse.setSuccess(false);
137
138   wsResponse.setErrorMessage("Error while saving vista outbound message: "
+ e.getMessage());
139   }
140
141   return wsResponse;
```

Sink Details

Sink: Return wsResponse
Enclosing Method: persistXMLMessage()
File: PS_INB_ERX_WS/src/main/java/gov/va/med/pharmacy/jaxrs/vistaoutboundmsg/service/impl/
VistaOutboundMsgImpl.java:141
Taint Flags: EXCEPTIONINFO, SYSTEMINFO

```
138   wsResponse.setErrorMessage("Error while saving vista outbound message: " +
e.getMessage());
139   }
140
141   return wsResponse;
142   }
143
144 undefined
```

System Information Leak: HTML Comment in JSP (5 issues)

Abstract

Any information revealed in an HTML comment might help an adversary learn about the system and form a plan of attack.

Explanation

HTML comments provide an attacker with an easy source of information about a dynamically generated web page.

Example 1:

```
<!-- TBD: this needs a security audit -->
<form method="POST" action="recalcOrbit">
...
```

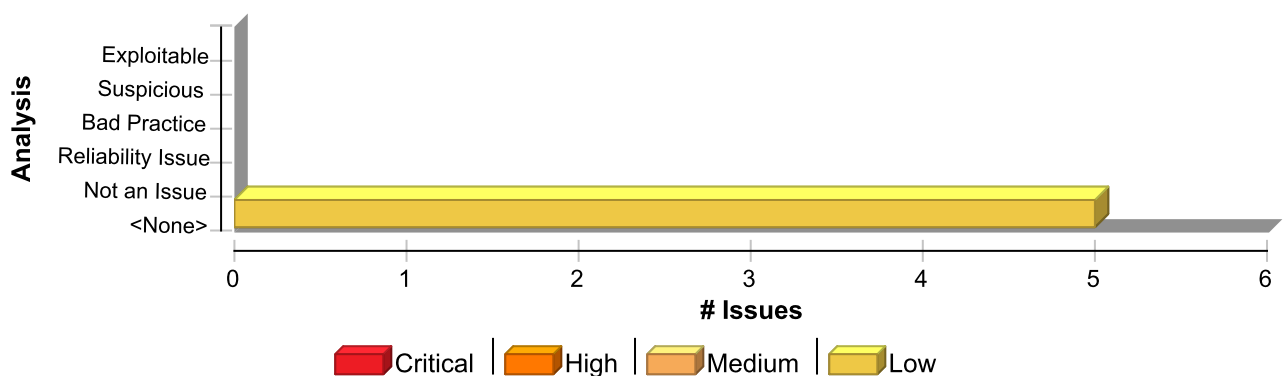Even comments that seem innocuous may be useful to someone trying to understand the way the system is built.

Recommendation

Replace HTML comments with JSP comments (which will not be transmitted to the user).

Example 2: The previous example is rewritten to use JSP comments, which will not be displayed to the user.

```
<%-- TBD: this needs a security audit --%>
<form method="POST" action="recalcOrbit">
...
```

Issue Summary

Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
System Information Leak: HTML Comment in JSP	5	0	0	5
Total	**5**	**0**	**0**	**5**

System Information Leak: HTML Comment in JSP	Low

Package: .src.main.webapp.WEB-INF	

PS_INB_ERX_UI/src/main/webapp/WEB-INF/logout.jsp, line 30 (System Information Leak: HTML Comment in JSP)	Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: PS_INB_ERX_UI/src/main/webapp/WEB-INF/logout.jsp:30
Taint Flags:

```
27   <hr />
28   </div>
29   </body>
30   <!-- Included to resolve IE caching bugs -->
31   <head>
32   <meta http-equiv="pragma" content="no-cache" />
33   <meta http-equiv="cache-control" content="no-cache" />
```

PS_INB_ERX_UI/src/main/webapp/WEB-INF/help.jsp, line 24 (System Information Leak: HTML Comment in JSP)	Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: PS_INB_ERX_UI/src/main/webapp/WEB-INF/help.jsp:24
Taint Flags:

```
21 href="/inbound/images/help/colorschememapping.xml">
22
23 <style>
24 <!--
25 /* Font Definitions */
26 @font-face
27 {font-family:Wingdings;
```

System Information Leak: HTML Comment in JSP	Low

Package: .src.main.webapp.WEB-INF	
PS_INB_ERX_UI/src/main/webapp/WEB-INF/reports.jsp, line 44 (System Information Leak: HTML Comment in JSP)	**Low**

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: PS_INB_ERX_UI/src/main/webapp/WEB-INF/reports.jsp:44
Taint Flags:

```
41  <input type="text" name="dateTo" id="dateTo" value="now"
42  data-dojo-type="dijit/form/DateTextBox" required="true" />
43
44  <!-- <button id="runReportButton" type="button" >Run Report</button> -->
45  <button dojoType="dijit.form.Button" type="button">Run Report
46  <script type="dojo/method" event="onClick" args="evt">
47  getSummaryReport();
```

PS_INB_ERX_UI/src/main/webapp/WEB-INF/errorPage.jsp, line 32 (System Information Leak: HTML Comment in JSP)	**Low**

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Structural)

Sink Details

Sink: Comment
File: PS_INB_ERX_UI/src/main/webapp/WEB-INF/errorPage.jsp:32
Taint Flags:

```
29  <br>
30  </p>
31  <br />
32  <!--
33  <p>
34  Please <a href="${pageContext.request.contextPath}/public/welcome">login</a> again.
35  </p>
```

Package: .src.main.webapp.WEB-INF.layouts	
PS_INB_ERX_UI/src/main/webapp/WEB-INF/layouts/basicLayout.jsp, line 18 (System Information Leak: HTML Comment in JSP)	**Low**

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Structural)

System Information Leak: HTML Comment in JSP	Low

Package: .src.main.webapp.WEB-INF.layouts	

PS_INB_ERX_UI/src/main/webapp/WEB-INF/layouts/basicLayout.jsp, line 18 (System Information Leak: HTML Comment in JSP)	Low

Sink Details

Sink: Comment
File: PS_INB_ERX_UI/src/main/webapp/WEB-INF/layouts/basicLayout.jsp:18
Taint Flags:

```
15
16 <body id="home" class="claro">
17
18 <!--
19 <div id="preloader"></div>
20 -->
21
```

System Information Leak: Internal (3 issues)

Abstract

Revealing system data or debugging information helps an adversary learn about the system and form a plan of attack.

Explanation

An internal information leak occurs when system data or debugging information is sent to a local file, console, or screen via printing or logging.

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 user. 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.

In the mobile world, information leaks are also a concern.

Example 2: The code below logs the stack trace of a caught exception on the Android platform.
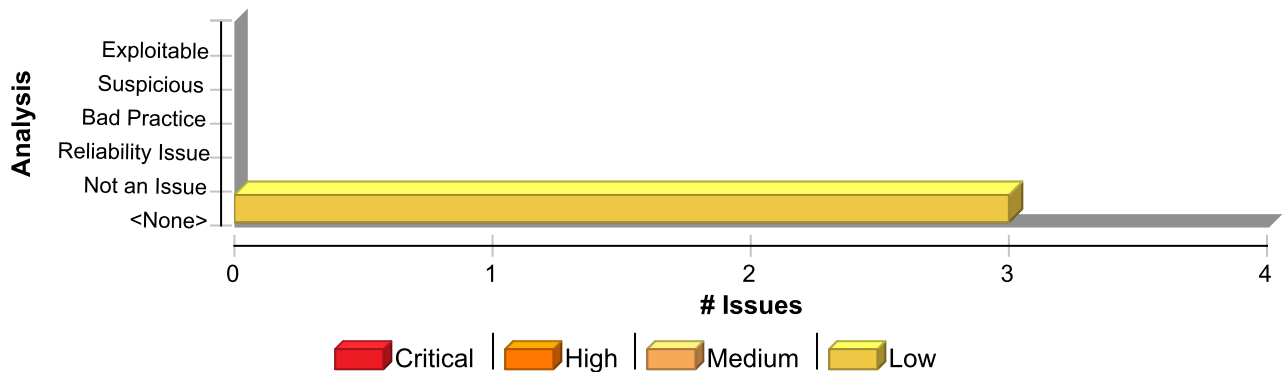
```
...
try {
  ...
} catch (Exception e) {
    Log.e(TAG, Log.getStackTraceString(e));
}
...
```

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.

Issue Summary

Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
System Information Leak: Internal	3	0	0	3
Total	**3**	**0**	**0**	**3**

System Information Leak: Internal	Low

Package: gov.va.med.pharmacy.utility	

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/ StreamUtilities.java, line 15 (System Information Leak: Internal)	Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read e
From: gov.va.med.pharmacy.utility.StreamUtilities.safeClose
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/StreamUtilities.java:15

```
12   try {
13   fileInputStream.close();
14   } catch (IOException e) {
15   System.out.print(e);
16   }
17   }
18   }
```

Sink Details

Sink: java.io.PrintStream.print()
Enclosing Method: safeClose()
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/StreamUtilities.java:15
Taint Flags: EXCEPTIONINFO, SYSTEMINFO

```
12   try {
```

System Information Leak: Internal	Low

Package: gov.va.med.pharmacy.utility

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/ StreamUtilities.java, line 15 (System Information Leak: Internal)	Low

```
13   fileInputStream.close();
14   } catch (IOException e) {
15   System.out.print(e);
16   }
17   }
18   }
```

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/ StreamUtilities.java, line 25 (System Information Leak: Internal)	Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read e
From: gov.va.med.pharmacy.utility.StreamUtilities.safeClose
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/StreamUtilities.java:25

```
22   try {
23   inputStream.close();
24   } catch (IOException e) {
25   System.out.print(e);
26   }
27   }
28   }
```

Sink Details

Sink: java.io.PrintStream.print()
Enclosing Method: safeClose()
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/StreamUtilities.java:25
Taint Flags: EXCEPTIONINFO, SYSTEMINFO

```
22   try {
23   inputStream.close();
24   } catch (IOException e) {
25   System.out.print(e);
26   }
27   }
28   }
```

System Information Leak: Internal	Low

Package: gov.va.med.pharmacy.utility

PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/ StreamUtilities.java, line 35 (System Information Leak: Internal)	Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Data Flow)

Source Details

Source: Read e
From: gov.va.med.pharmacy.utility.StreamUtilities.safeClose
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/StreamUtilities.java:35

```
32   try {
33   bufferedReader.close();
34   } catch (IOException e) {
35   System.out.print(e);
36   }
37   }
38   }
```

Sink Details

Sink: java.io.PrintStream.print()
Enclosing Method: safeClose()
File: PS_INB_ERX_Common/src/main/java/gov/va/med/pharmacy/utility/StreamUtilities.java:35
Taint Flags: EXCEPTIONINFO, SYSTEMINFO

```
32   try {
33   bufferedReader.close();
34   } catch (IOException e) {
35   System.out.print(e);
36   }
37   }
38   }
```

Trust Boundary Violation (6 issues)

Abstract

Commingling trusted and untrusted data in the same data structure encourages programmers to mistakenly trust unvalidated data.

Explanation

A trust boundary can be thought of as line drawn through a program. On one side of the line, data is untrusted. On the other side of the line, data is assumed to be trustworthy. The purpose of validation logic is to allow data to safely cross the trust boundary--to move from untrusted to trusted.

A trust boundary violation occurs when a program blurs the line between what is trusted and what is untrusted. The most common way to make this mistake is to allow trusted and untrusted data to commingle in the same data structure.

Example: The following Java code accepts an HTTP request and stores the `usrname` parameter in the HTTP session object before checking to ensure that the user has been authenticated.

```
usrname = request.getParameter("usrname");
if (session.getAttribute(ATTR_USR) != null) {
    session.setAttribute(ATTR_USR, usrname);
}
```
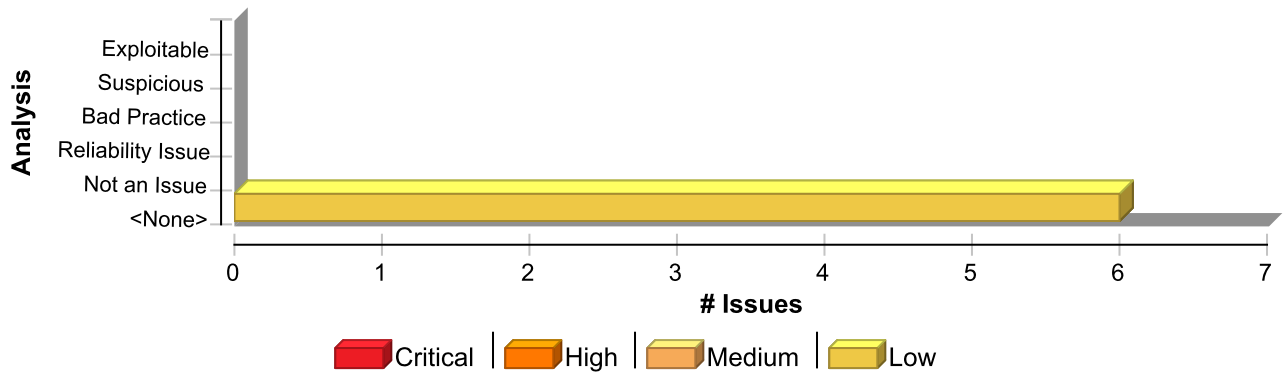
Without well-established and maintained trust boundaries, programmers will inevitably lose track of which pieces of data have been validated and which have not. This confusion will eventually allow some data to be used without first being validated.

Recommendation

Define clear trust boundaries in the application. Do not use the same data structure to hold trusted data in some contexts and untrusted data in other contexts. Minimize the number of ways that data can move across a trust boundary.

Trust boundary violations sometimes occur when input needs to be built up over a series of user interactions before being processed. It may not be possible to do complete input validation until all of the data has arrived. In these situations, it is still important to maintain a trust boundary. The untrusted data should be built up in a single untrusted data structure, validated, and then moved into a trusted location.

Issue Summary

Engine Breakdown

	SCA	WebInspect	SecurityScope	Total
Trust Boundary Violation	6	0	0	6
Total	**6**	**0**	**0**	**6**

Trust Boundary Violation	Low
Package: gov.va.med.pharmacy.web.administration	
PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/administration/ UserManagementController.java, line 91 (Trust Boundary Violation)	**Low**

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Data Flow)

Source Details

Source: org.hibernate.Criteria.list()
From: gov.va.med.pharmacy.persistence.dao.impl.UserDaoImpl.findAll
File: PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/dao/impl/UserD
aoImpl.java:65

62	
63	`criteria.setResultTransformer(Criteria.DISTINCT_ROOT_ENTITY);// for unique results`
64	
65	`List<VaUser> users = (List<VaUser>) criteria.list();`
66	
67	`return users;`
68	`}`

Sink Details

Sink: org.springframework.web.servlet.ModelAndView.addObject()
Enclosing Method: loadUsers()
File: PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/administration/UserManagementController.java:91
Taint Flags: DATABASE, NUMBER, PRIMARY_KEY, XSS

88	`userRoles.add(userModel);`
89	`}`

Trust Boundary Violation	Low

Package: gov.va.med.pharmacy.web.administration	

PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/administration/ UserManagementController.java, line 91 (Trust Boundary Violation)	**Low**

```
90
91   mav.addObject("items", userRoles);
92
93   return mav;
94   }
```

PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/administration/ PharmacyManagementController.java, line 99 (Trust Boundary Violation)	**Low**

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Data Flow)

Source Details

Source: org.hibernate.Criteria.uniqueResult()
From: gov.va.med.pharmacy.persistence.dao.impl.PharmacyDaoImpl.findByNCPDPId
File: PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/dao/impl/PharmacyDaoImpl.java:67

```
64   }
65
66
67   PharmacyEntity pharmacy = (PharmacyEntity) criteria.uniqueResult();
68
69   return pharmacy;
70   }
```

Sink Details

Sink: org.springframework.web.servlet.ModelAndView.addObject()
Enclosing Method: editPharmacy()
File: PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/administration/PharmacyManagementController.java:99
Taint Flags: DATABASE, XSS

```
96
97    PharmacyEntity pharmacyInfo = pharmacyService.findByNCPDPId(NCPDPId);
98
99    view.addObject("pharmacyInfo",pharmacyInfo );
100
101   Map<String, String> statesMap = getStatesHashMap();
102
```

PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/administration/ PharmacyManagementController.java, line 283 (Trust Boundary Violation)	**Low**

Issue Details

Trust Boundary Violation	Low

Package: gov.va.med.pharmacy.web.administration	
PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/administration/ PharmacyManagementController.java, line 283 (Trust Boundary Violation)	**Low**

Kingdom: Encapsulation
Scan Engine: SCA (Data Flow)

Source Details

Source: org.hibernate.Session.get()
From: gov.va.med.pharmacy.persistence.BaseDao.getByKey
File: PS_INB_ERX_Model/src/main/java/gov/va/med/pharmacy/persistence/BaseDao.java:2
9

```
26
27  @SuppressWarnings("unchecked")
28  public T getByKey(long id) {
29  return (T) getSession().get(persistentClass, id);
30  }
31
32  public void persist(T entity) {
```

Sink Details

Sink: org.springframework.web.servlet.ModelAndView.addObject()
Enclosing Method: updatePharmacy()
File: PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/administration/PharmacyManagementController.java:283
Taint Flags: DATABASE, PRIMARY_KEY, XSS

```
280  }
281
282
283  view.addObject("pharmacyInfo",pharmacyInfo );
284
285  Map<String, String> statesMap = getStatesHashMap();
286
```

Package: gov.va.med.pharmacy.web.auth	
PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/auth/ LoginController.java, line 79 (Trust Boundary Violation)	**Low**

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Data Flow)

Source Details

Source: javax.servlet.http.HttpServletRequest.getHeader()
From: gov.va.med.pharmacy.web.auth.LoginController.authenticateUser
File: PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/auth/LoginController.java
:73

Trust Boundary Violation	Low

Package: gov.va.med.pharmacy.web.auth

PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/auth/ LoginController.java, line 79 (Trust Boundary Violation)	Low

```
70
71   if(request.getHeader(Constants.SSOI_LANDING_URL)!=null)
72   {
73   String logoutURL = request.getHeader(Constants.SSOI_LANDING_URL);
74
75   if(ESAPI.validator().isValidInput("IAMLogout", logoutURL, "IAMLogout",
     300, false, false))
76   {
```

Sink Details

Sink: javax.servlet.http.HttpSession.setAttribute()
Enclosing Method: authenticateUser()
File: PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/auth/LoginController.java:79
Taint Flags: WEB, XSS

```
76   {
77   session.removeAttribute(Constants.SSOI_LANDING_URL);
78
79   session.setAttribute(Constants.SSOI_LANDING_URL, logoutURL);
80   }
81
82   }
```

URL: /managePharm/addNewPharmacy

PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/administration/ PharmacyManagementController.java, line 626 (Trust Boundary Violation)	Low

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Data Flow)

Source Details

Source: addNewPharmacy(1)
From: gov.va.med.pharmacy.web.administration.PharmacyManagementController.addNewPharmacy
File: PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/administration/PharmacyManagementController.java:458
URL: /managePharm/addNewPharmacy

```
455   }
456
457   @RequestMapping(value = "/addNewPharmacy", method = RequestMethod.POST)
458   public ModelAndView addNewPharmacy(HttpServletRequest request,@Valid
      @ModelAttribute("pharmacyAddForm") PharmacyForm pharmacyForm, BindingResult
```

Trust Boundary Violation	Low

URL: /managePharm/addNewPharmacy	
PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/administration/ PharmacyManagementController.java, line 626 (Trust Boundary Violation)	**Low**

```
bindingResult) throws IOException {
```

459	
460	LOG.info("Trying to persist pharmacy information.");
461	

Sink Details

Sink: org.springframework.web.servlet.ModelAndView.addObject()
Enclosing Method: addNewPharmacy()
File: PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/administration/PharmacyManagementController.java:626
Taint Flags: WEB, XSS

623	responseMessage.setSuccessMessage("Pharmacy information added successfully.");
624	}
625	
626	view.addObject("pharmacyInfo",pharmacyInfo);
627	
628	Map<String, String> statesMap = getStatesHashMap();
629	

URL: /managePharm/updatePharmacy	
PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/administration/ PharmacyManagementController.java, line 283 (Trust Boundary Violation)	**Low**

Issue Details

Kingdom: Encapsulation
Scan Engine: SCA (Data Flow)

Source Details

Source: updatePharmacy(1)
From: gov.va.med.pharmacy.web.administration.PharmacyManagementController.updatePharmacy
File: PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/administration/PharmacyManagementController.java:122
URL: /managePharm/updatePharmacy

119	}
120	
121	@RequestMapping(value = "/updatePharmacy", method = RequestMethod.POST)
122	public ModelAndView updatePharmacy(HttpServletRequest request,@Valid @ModelAttribute("pharmacyEditForm") PharmacyForm pharmacyForm, BindingResult bindingResult) throws IOException {
123	
124	LOG.info("Trying to persist pharmacy information.");
125	

Trust Boundary Violation	Low

URL: /managePharm/updatePharmacy

PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/administration/ PharmacyManagementController.java, line 283 (Trust Boundary Violation)	Low

Sink Details

Sink: org.springframework.web.servlet.ModelAndView.addObject()
Enclosing Method: updatePharmacy()
File: PS_INB_ERX_UI/src/main/java/gov/va/med/pharmacy/web/administration/PharmacyManagementController.java:283
Taint Flags: WEB, XSS

```
280  }
281
282
283  view.addObject("pharmacyInfo",pharmacyInfo );
284
285  Map<String, String> statesMap = getStatesHashMap();
286
```