BLOCK AUDIT REPORT GOLD

Security Audit Report

POTENTWALLET

Disclaimer

BlockAudit reports are not, nor should be considered, an "endorsement" or "disapproval" of any particular project or team. These reports are not, nor should be considered, an indication of the economics or value of any "product" or "asset" created by any team or project that contracts BlockAudit to perform a security review.

BlockAudit Reports do not provide any warranty or guarantee regarding the absolute bug-free nature of the technology analyzed, nor do they provide any indication of the technologies proprietors, business, business model or legal compliance.

BlockAudit Reports should not be used in any way to make decisions around investment or involvement with any particular project. These reports in no way provide investment advice, nor should be leveraged as investment advice of any sort.

BlockAudit Reports represent an extensive auditing process intending to help our customers increase the quality of their code while reducing the high level of risk presented by cryptographic tokens and blockchain technology.

Blockchain technology and cryptographic assets present a high level of ongoing risk. BlockAudit's position is that each company and individual are responsible for their own due diligence and continuous security. BlockAudit's goal is to help reduce the attack vectors and the high level of variance associated with utilizing new and consistently changing technologies, and in no way claims any guarantee of security or functionality of the technology we agree to analyze.

What is a BlockAudit report?

A document describing in detail an in-depth analysis of a particular piece(s) of source code provided to BlockAudit by a Client.

An organized collection of testing results, analysis and inferences made about the structure, implementation and overall best practices of a particular piece of source code.

Representation that a Client of BlockAudit has completed a round of auditing with the intention to increase the quality of the company/product's IT infrastructure and or source code.

Overview

Project Name	PotentWallet
Description	Round three audit of the PotentWallet
Languages	Java, Kotlin
Codebase	GitHub Repository
Commits	https://github.com/Potentcoin/PotentWallet

Audit Summary

Delivery Date	July23rd, 2021
Method of Audit	Static Analysis, Manual Review
Consultants Engaged	2
Timeline	July 15th, 2021 - July 23rd, 2021

Vulnerability Summary

Total Files (3)	Issues	Vulnerability	Code Smells	Coverage	Duplications
Wallet connect	0	0	1	0%	0%
Web3	0	2	79	0%	1%
Web3j	2	1	40	0%	0%

Files In Scope

All files in:

https://github.com/Potentcoin/PotentWallet/tree/main/Potent_app_report-main/walletconnect

Enums.kt	Passed
EthereumModels.kt	Passed
Exceptions.kt	Passed
JsonRpcModels.kt	Passed
SessionModels.kt	Passed
Extensions.kt	Passed
WCCipher.kt	Passed
WCClient.kt	Passed
WCSession.kt	Passed
JsInjectorClient.java	Passed
JsInjectorResponse.java	Passed
OnEthCallListener.java	Passed
OnGetBalanceListener.java	Passed
OnSetValuesListener.java	Passed
OnSignMessageListener.java	Passed
OnSignPersonalMessageListener.java	Passed
OnSignTransactionListener.java	Passed
OnSignTypedMessageListener.java	Passed
OnVerifyListener.java	Passed
SignCallbackJSInterface.java	Passed
TokenScriptCallbackInterface.java	Passed
UrlHandler.java	Passed
UrlHandlerManager.java	Passed
ValueCallbackJSInterface.java	Passed
Web3TokenView.java	Passed
Web3View.java	Passed
Web3ViewClient.java	Passed
WebViewCookieJar.java	Passed
StructuredData.java	Passed
StructuredDataEncoder.java	Passed

Findings

Issues :

Major Severity Issues:-

This block of commented-out lines of code should be removed.

Potent_app_report-main/web3/JsInjectorClient.java

213 //String injectHeader = "<head><meta name=\"viewport\" content=\"width=device-width, user-scalable=false\" /></head>";

214 String injectHeader = "<head><meta name=\"viewport\" content=\"width=device-width, initial-scale=1, maximum-scale=1, shrink-to-fit=no\" />"; //iOS uses these header settings

Potent_app_report-main/web3/JsInjectorClient.java

220 // the opening of the following is in injectWeb3TokenInit();

221 return injectHeader + style + view + " </body>";

Potent_app_report-main/web3/entity/Web3Transaction.java

16//import com.trustwallet.walletconnect.models.ethereum.WCEthereumTransaction;

17

Potent_app_report-main/web3/entity/Web3Transaction.java

120 //dest.writeLong(gasLimit);

121 dest.writeLong(nonce);

Suggested Solution :

Programmers should not comment out code as it bloats programs and reduces readability.

Add a private constructor to hide the implicit public one.

Potent_app_report-main/web3j/StructuredData.java

```
31public class StructuredData {
32 public static class Entry {
```

Suggested Solution :

Utility classes, which are collections of `static` members, are not meant to be instantiated. Even abstract utility classes, which can be extended...

Replace this use of System.out or System.err by a logger.

Potent_app_report-main/web3/Web3TokenView.java

```
194 System.out.println(data);
195 }
```

Suggested Solution :

When logging a message there are several important requirements which must be fulfilled:

Remove this useless assignment to local variable "stateData".

Potent_app_report-main/web3/Web3View.java + 1 other file

Suggested Solution :

A dead store happens when a local variable is assigned a value that is not read by any subsequent instruction. Calculating or retrieving a value only to then...

Merge this if statement with the enclosing one.

Potent_app_report-main/web3/JsInjectorClient.java + 1 other file

Suggested Solution :

Merging collapsible `if` statements increases the code's readability.

Either remove or fill this block of code.

Potent_app_report-main/web3j/StructuredDataEncoder.java

```
385 | InvocationTargetException ignored) {
386 }
```

Suggested Solution :

Most of the time a block of code is empty when a piece of code is really missing. So such empty block must be either filled or removed.

Constructor has 8 parameters, which is greater than 7 authorized.

Potent_app_report-main/web3/entity/Web3Transaction.java

```
45 public Web3Transaction(
46 Address recipient,
```

Suggested Solution :

A long parameter list can indicate that a new structure should be created to wrap the numerous parameters or that the function is doing too many things.

Remove this expression which always evaluates to "false"

Potent_app_report-main/web3/SignCallbackJSInterface.java
64 if (value.equals("undefined") || value == null) value = "0";
65 if (gasPrice == null) gasPrice = "0";

Suggested Solution :

If a boolean expression doesn't change the evaluation of the condition, then it is entirely unnecessary, and can be removed. If it is gratuitous because it d...

Remove this unused private "WrapWebViewClient" class.

Potent_app_report-main/web3/Web3View.java
290 private class WrapWebViewClient extends WebViewClient {
291 private final Web3ViewClient internalClient;

Suggested Solution :

`private` classes that are never used are dead code: unnecessary, inoperative code that should be removed. Cleaning out dead code decreases the si...

Call "Optional#isPresent()" before accessing the value.

Potent_app_report-main/web3j/StructuredDataEncoder.java
208 dimensions.add(setOfDimensionsInParticularDepth.stream().findFirst().get());
209 }

Suggested Solution :

`Optional` value can hold either a value or not. The value held in the `Optional` can be accessed using the `get()` method, ...

Remove this unused "webView" private field.

Potent_app_report-main/web3/ValueCallbackJSInterface.java
18 private final WebView webView;
19 @NonNull
Potent_app_report-main/web3/Web3TokenView.java
264 private final OnSignTransactionListener innerOnSignTransactionListener = new OnSignTransactionListener() {
265 @Override
Potent_app_report-main/web3/Web3TokenView.java
271 private final OnSignMessageListener innerOnSignMessageListener = new OnSignMessageListener() {
272 @Override
Potent_app_report-main/web3/Web3View.java
267 private final OnVerifyListener innerOnVerifyListener = new OnVerifyListener() {
268 @Override
Potent_app_report-main/web3/Web3View.java
276 private final OnGetBalanceListener innerOnGetBalanceListener = new OnGetBalanceListener() {
277 @Override

Suggested Solution :

If a `private` field is declared but not used in the program, it can be considered dead code and should therefore be removed. This will improve ma...

1 duplicated blocks of code must be removed.

Potent_app_report-main/web3/TokenScriptCallbackInterface.java
1package io.horizontalsystems.bankwallet.web3;
2
Potent_app_report-main/web3/ValueCallbackJSInterface.java
1package io.horizontalsystems.bankwallet.web3;
2

Suggested Solution :
An issue is created on a file as soon as there is at least one block of duplicated code on this file

Define and throw a dedicated exception instead of using a generic one.

Potent_app_report-main/web3j/StructuredDataEncoder.java
183 public List<Integer> getArrayDimensionsFromData(Object data) throws RuntimeException {
184 List<Pair> depthsAndDimensions = getDepthsAndDimensions(data, 0);
Potent_app_report-main/web3j/StructuredDataEncoder.java
203 throw new RuntimeException(
204 String.format(
Potent_app_report-main/web3j/StructuredDataEncoder.java
266 throw new RuntimeException(format);
267 }
Potent_app_report-main/web3j/StructuredDataEncoder.java

274 throw new RuntimeException(format);

275 }

Potent_app_report-main/web3j/StructuredDataEncoder.java

390 throw new RuntimeException(

391 String.format(

Potent_app_report-main/web3j/StructuredDataEncoder.java

442 throws RuntimeException {

443 for (String structName : jsonMessageObject.getTypes().keySet()) {

Potent_app_report-main/web3j/StructuredDataEncoder.java

448 throw new RuntimeException(

449 String.format(

Potent_app_report-main/web3j/StructuredDataEncoder.java

454 throw new RuntimeException(

455 String.format("Invalid Type %s in %s", entry.getType(), structName));

Potent_app_report-main/web3j/StructuredDataEncoder.java

462 throws IOException, RuntimeException {

463 ObjectMapper mapper = new ObjectMapper();

Potent_app_report-main/web3j/StructuredDataEncoder.java

499 private static byte[] convertArgToBytes(String inputValue) throws Exception {

500 String hexValue = inputValue;

Suggested Solution :

Using such generic exceptions as `Error`, `RuntimeException`, `Throwable`, and `Exception` prevents calling metho...

Remove this unused method parameter "code".

Potent_app_report-main/web3/JsInjectorResponse.java

10 JsInjectorResponse(String data, int code, String url, String mime, String charset, boolean isRedirect) {

11 this.data = data;

Suggested Solution :

Unused parameters are misleading. Whatever the values passed to such parameters, the behavior will be the same.

Remove this unused private "isJson" method.

Potent_app_report-main/web3/Web3View.java
373 private static boolean isJson(String value) {
374 try {

Suggested Solution :

`private` methods that are never executed are dead code: unnecessary, inoperative code that should be removed. Cleaning out dead code decreases th...

Rename "type" which hides the field declared at line 9.

Potent_app_report-main/web3/entity/TypedData.java
21 Class<?> type = (Class<?>) in.readSerializable();
22 data = in.readValue(type.getClassLoader());
Potent_app_report-main/web3/entity/Web3Transaction.java
71 String gasPrice = wcTx.getGasPrice() != null ? wcTx.getGasPrice() : "0";
72 String gasLimit = wcTx.getGasLimit() != null ? wcTx.getGasLimit() : "0";
Potent_app_report-main/web3/entity/Web3Transaction.java
72 String gasLimit = wcTx.getGasLimit() != null ? wcTx.getGasLimit() : "0";
73 String nonce = wcTx.getNonce() != null ? wcTx.getNonce() : "";
Potent_app_report-main/web3/entity/Web3Transaction.java
73 String nonce = wcTx.getNonce() != null ? wcTx.getNonce() : "";
74

Suggested Solution :

Overriding or shadowing a variable declared in an outer scope can strongly impact the readability, and therefore the maintainability, of a piece of code. Fur...

Move the contents of this initializer to a standard constructor or to field initializers.

Potent_app_report-main/web3j/StructuredDataEncoder.java
217 {
218 add(data);

Suggested Solution :

Non-static initializers are rarely used, and can be confusing for most developers because they only run when new class instances are created. When possible, ...

Critical Severity Issues

Make sure that using a regular expression is safe here.

Potent_app_report-main/web3j/StructuredDataEncoder.java
55 final Pattern arrayTypePattern = Pattern.compile(arrayTypeRegex);
56
Potent_app_report-main/web3j/StructuredDataEncoder.java
69 final Pattern typePattern = Pattern.compile(typeRegex);
70 // Identifier Regex matches to a valid name, but can't be an array declaration.

Suggested Solution :

Using regular expressions is security-sensitive. It has led in the past to the following vulnerabilities:

Add a nested comment explaining why this method is empty, throw an UnsupportedOperationException or complete the implementation.

Potent_app_report-main/web3/Web3TokenView.java

266 public void onSignTransaction(Web3Transaction transaction, String url) {
267

Potent_app_report-main/web3/Web3TokenView.java

273 public void onSignMessage(EthereumMessage message) {
274

Suggested Solution :

There are several reasons for a method not to have a method body:

This file "JsInjectorClient.java" should be located in "io/horizontalsystems/bankwallet/web3" directory, not in "/tmp/PotentWallet_162677723446330610231568825216 92/Potent_app_report-main/web3".

Potent_app_report-main/web3/JsInjectorClient.java

1package io.horizontalsystems.bankwallet.web3;
2

Potent_app_report-main/web3/JsInjectorResponse.java

1package io.horizontalsystems.bankwallet.web3;
2
Potent_app_report-main/web3/OnEthCallListener.java
1package io.horizontalsystems.bankwallet.web3;
2
Potent_app_report-main/web3/OnGetBalanceListener.java
1package io.horizontalsystems.bankwallet.web3;
2
Potent_app_report-main/web3/OnSetValuesListener.java
1package io.horizontalsystems.bankwallet.web3;
2
Potent_app_report-main/web3/OnSignMessageListener.java
1package io.horizontalsystems.bankwallet.web3;
2
Potent_app_report-main/web3/OnSignPersonalMessageListener.java
1package io.horizontalsystems.bankwallet.web3;
2
Potent_app_report-main/web3/OnSignTransactionListener.java
1package io.horizontalsystems.bankwallet.web3;
2
Potent_app_report-main/web3/OnSignTypedMessageListener.java
1package io.horizontalsystems.bankwallet.web3;
2
Potent_app_report-main/web3/OnVerifyListener.java
1package io.horizontalsystems.bankwallet.web3;
2
Potent_app_report-main/web3/SignCallbackJSInterface.java
1package io.horizontalsystems.bankwallet.web3;
2
Potent_app_report-main/web3/TokenScriptCallbackInterface.java
1package io.horizontalsystems.bankwallet.web3;
2
Potent_app_report-main/web3/UrlHandler.java
1package io.horizontalsystems.bankwallet.web3;
2

Potent_app_report-main/web3/UrlHandlerManager.java

```
1package io.horizontalsystems.bankwallet.web3;
2
```

Potent_app_report-main/web3/ValueCallbackJSInterface.java

```
1package io.horizontalsystems.bankwallet.web3;
2
```

Potent_app_report-main/web3/Web3TokenView.java

```
1package io.horizontalsystems.bankwallet.web3;
2
```

Potent_app_report-main/web3/Web3View.java

```
1package io.horizontalsystems.bankwallet.web3;
2
```

Potent_app_report-main/web3/Web3ViewClient.java

```
1package io.horizontalsystems.bankwallet.web3;
2
```

Potent_app_report-main/web3/WebViewCookieJar.java

```
1package io.horizontalsystems.bankwallet.web3;
2
```

Potent_app_report-main/web3/entity/Address.java

```
1package io.horizontalsystems.bankwallet.web3.entity;
2
```

Potent_app_report-main/web3/entity/FunctionCallback.java

```
1package io.horizontalsystems.bankwallet.web3.entity;
2
```

Potent_app_report-main/web3/entity/PageReadyCallback.java

```
1package io.horizontalsystems.bankwallet.web3.entity;
2
```

Potent_app_report-main/web3/entity/ScriptFunction.java

```
1package io.horizontalsystems.bankwallet.web3.entity;
2
```

Potent_app_report-main/web3/entity/TypedData.java

```
1package io.horizontalsystems.bankwallet.web3.entity;
2
```

Potent_app_report-main/web3/entity/Web3Call.java

1package io.horizontalsystems.bankwallet.web3.entity;
2import org.web3j.protocol.core.DefaultBlockParameter;
Potent_app_report-main/web3/entity/Web3Transaction.java
1package io.horizontalsystems.bankwallet.web3.entity;
2
Potent_app_report-main/web3j/StructuredData.java
13package io.horizontalsystems.bankwallet.web3j;
14
Potent_app_report-main/web3j/StructuredDataEncoder.java
13package io.horizontalsystems.bankwallet.web3j;
14

Suggested Solution :

By convention, a Java class' physical location (source directories) and its logical representation (packages) should be kept in sync. Thus a Java file locate...

Refactor this method to reduce its Cognitive Complexity from 19 to the 15 allowed.

Potent_app_report-main/web3/Web3TokenView.java
107 private void init() {
108 tokenScriptClient = new TokenScriptClient(this);
Potent_app_report-main/web3/Web3ViewClient.java
124 public WebResourceResponse shouldInterceptRequest(WebView view, WebResourceRequest request) {
125 if (request == null) {
Potent_app_report-main/web3j/StructuredDataEncoder.java
282 public byte[] encodeData(String primaryType, HashMap<String, Object> data)
283 throws RuntimeException {

Suggested Solution :

Cognitive Complexity is a measure of how hard the control flow of a method is to understand. Methods with high Cognitive Complexity will be difficult to main...

Define a constant instead of duplicating this literal "text/html" 3 times.

Potent_app_report-main/web3/Web3TokenView.java
182 loadData(error, "text/html", "utf-8");
183 }
Potent_app_report-main/web3/Web3TokenView.java
182 loadData(error, "text/html", "utf-8");
183 }
Potent_app_report-main/web3j/StructuredDataEncoder.java
290 encTypes.add("bytes32");
291 encValues.add(typeHash(primaryType));
Potent_app_report-main/web3j/StructuredDataEncoder.java
421 if (data.get("chainId") != null) {
422 data.put("chainId", ((HashMap<String, Object>)
data.get("chainId")).get("value"));
Potent_app_report-main/web3j/StructuredDataEncoder.java
427 if (data.get("verifyingContract") != null) {
428 data.put(

Suggested Solution :

Duplicated string literals make the process of refactoring error-prone, since you must be sure to update all occurrences.

Minor Severity Issues

Remove the declaration of thrown exception 'java.lang.RuntimeException' which is a runtime exception.

Potent_app_report-main/web3j/StructuredDataEncoder.java
74 public StructuredDataEncoder(String jsonMessageInString) throws IOException, RuntimeException {
75 // Parse String Message into object and validate
Potent_app_report-main/web3j/StructuredDataEncoder.java
183 public List<Integer> getArrayDimensionsFromData(Object data) throws RuntimeException {
184 List<Pair> depthsAndDimensions = getDepthsAndDimensions(data, 0);
Potent_app_report-main/web3j/StructuredDataEncoder.java
283 throws RuntimeException {
284 HashMap<String, List<StructuredData.Entry>> types = jsonMessageObject.getTypes();
Potent_app_report-main/web3j/StructuredDataEncoder.java
402 throws NumberFormatException, NullPointerException {
403 if (value.toString().startsWith("0x")) {
Potent_app_report-main/web3j/StructuredDataEncoder.java
402 throws NumberFormatException, NullPointerException {
403 if (value.toString().startsWith("0x")) {
Potent_app_report-main/web3j/StructuredDataEncoder.java
411 throws RuntimeException {
412 return sha3(encodeData(primaryType, data));
Potent_app_report-main/web3j/StructuredDataEncoder.java
416 public byte[] hashDomain() throws RuntimeException {
417 ObjectMapper oMapper = new ObjectMapper();
Potent_app_report-main/web3j/StructuredDataEncoder.java
442 throws RuntimeException {
443 for (String structName : jsonMessageObject.getTypes().keySet()) {
Potent_app_report-main/web3j/StructuredDataEncoder.java

462 throws IOException, RuntimeException {
463 ObjectMapper mapper = new ObjectMapper();
Potent_app_report-main/web3j/StructuredDataEncoder.java
474 public byte[] getStructuredData() throws RuntimeException {
475
Potent_app_report-main/web3j/StructuredDataEncoder.java
495 public byte[] hashStructuredData() throws RuntimeException {
496 return sha3(getStructuredData());

Suggested Solution :

An exception in a `throws` declaration in Java is superfluous if it is:

Reorder the modifiers to comply with the Java Language Specification.

Potent_app_report-main/web3/JsInjectorClient.java
35 private final static String JS_TAG_TEMPLATE = "<script type=\"text/javascript\">%1$s%2$s</script>";
36

Suggested Solution :

The Java Language Specification recommends listing modifiers in the following order:

The type of the "types" object should be an interface such as "Map" rather than the implementation "HashMap".

Potent_app_report-main/web3j/StructuredData.java
103 @JsonProperty(value = "types") HashMap<String, List<Entry>> types,
104 @JsonProperty(value = "primaryType") String primaryType,
Potent_app_report-main/web3j/StructuredData.java

113 public HashMap<String, List<Entry>> getTypes() {
114 return types;
Potent_app_report-main/web3j/StructuredDataEncoder.java
282 public byte[] encodeData(String primaryType, HashMap<String, Object> data)
283 throws RuntimeException {
Potent_app_report-main/web3j/StructuredDataEncoder.java
410 public byte[] hashMessage(String primaryType, HashMap<String, Object> data)
411 throws RuntimeException {

Suggested Solution :

The purpose of the Java Collections API is to provide a well defined hierarchy of interfaces in order to hide implementation details.

Use isEmpty() to check whether the collection is empty or not.

Potent_app_report-main/web3/Web3ViewClient.java
296 return list.size() > 0;
297 }
Potent_app_report-main/web3j/StructuredDataEncoder.java
91 while (remainingTypes.size() > 0) {
92 String structName = remainingTypes.get(remainingTypes.size() - 1);

Suggested Solution :

Using `Collection.size()` to test for emptiness works, but
using `Collection.isEmpty()` makes the code more readable and can be more pe...

Remove this method to simply inherit it.

Potent_app_report-main/web3/Web3TokenView.java

186 public void setWebChromeClient(WebChromeClient client)
187 {
Potent_app_report-main/web3/Web3View.java
76 public void setWebChromeClient(WebChromeClient client) {
77 super.setWebChromeClient(client);
Potent_app_report-main/web3/Web3View.java
81 public void setWebViewClient(WebViewClient client) {
82 super.setWebViewClient(client);
Potent_app_report-main/web3/Web3View.java
304 public void onPageStarted(WebView view, String url,Bitmap favicon) {
305 super.onPageStarted(view, url, favicon);

Suggested Solution :

Overriding a method just to call the same method from the super class without performing any other actions is useless and misleading. The only time this is j...

Rename this local variable to match the regular expression '^[a-z][a-zA-Z0-9]*$'.

Potent_app_report-main/web3/Web3TokenView.java
313 public String injectJSAtEnd(String view, String JSCode)
314 {

Suggested Solution :

Shared naming conventions allow teams to collaborate effectively. This rule raises an issue when a local variable or function parameter name does not match t...

Remove the "tokenScriptClient" field and declare it as a local variable in the relevant methods.

Potent_app_report-main/web3/Web3TokenView.java

78 private TokenScriptClient tokenScriptClient;

79 private PageReadyCallback assetHolder;

Suggested Solution :

When the value of a private field is always assigned to in a class' methods before being read, then it is not being used to store class information. Therefor...

Use a logger to log this exception.

Potent_app_report-main/web3/SignCallbackJSInterface.java

106 e.printStackTrace();

107 }

Potent_app_report-main/web3/Web3TokenView.java

409 e.printStackTrace();

410 }

Potent_app_report-main/web3j/StructuredDataEncoder.java

247 e.printStackTrace();

248 hashBytes = new byte[0];

Suggested Solution :

`Throwable.printStackTrace(...)` prints a `Throwable` and its stack trace to some stream. By default that stream `System.Err`

Use another way to initialize this instance.

Potent_app_report-main/web3j/StructuredDataEncoder.java

216 return new ArrayList<Object>() {
217 {

Suggested Solution :

Because Double Brace Initialization (DBI) creates an anonymous class with a reference to the instance of the owning object, its use can lead to memory leaks ...

Remove this unused import 'android.text.format.DateUtils'.

Potent_app_report-main/web3/Web3TokenView.java
9import android.text.format.DateUtils;
10import android.util.AttributeSet;
Potent_app_report-main/web3/Web3TokenView.java
47import java.io.IOException;
48import java.io.LineNumberReader;
Potent_app_report-main/web3/Web3View.java
5import android.content.Intent;
6import android.graphics.Bitmap;
Potent_app_report-main/web3/Web3View.java
7import android.net.Uri;
8import android.os.Build;
Potent_app_report-main/web3/Web3ViewClient.java
16import android.webkit.WebSettings;
17import android.webkit.WebView;

Suggested Solution :

The imports part of a file should be handled by the Integrated Development Environment (IDE), not manually by the developer.

Remove this unused "address" local variable.

Potent_app_report-main/web3/SignCallbackJSInterface.java

95 String address = obj.getString("from");

96 String messageData = obj.getString("data");

Potent_app_report-main/web3/Web3View.java

375 JSONObject stateData = new JSONObject(value);

376 return true;

Potent_app_report-main/web3/Web3ViewClient.java

80 boolean result = false;

81 synchronized (lock) {

Suggested Solution :

If a local variable is declared but not used, it is dead code and should be removed. Doing so will improve maintainability because developers will not wonder...

Remove this empty statement.

Potent_app_report-main/web3/entity/PageReadyCallback.java

13 default boolean overridePageLoad(WebView view, String url) { return true; }; //by default, don't allow TokenScript to access any URL

14}

Suggested Solution :

Empty statements, i.e. ;, are usually introduced by mistake, for example because:

Make this final field static too.

Potent_app_report-main/web3j/StructuredDataEncoder.java

54 final String arrayTypeRegex = "^([a-zA-Z_$][a-zA-Z_$0-9]*)((\\[([1-9]\\d*)?\\])+)$";
55 final Pattern arrayTypePattern = Pattern.compile(arrayTypeRegex);
Potent_app_report-main/web3j/StructuredDataEncoder.java
57 final String bytesTypeRegex = "^bytes[0-9][0-9]?$";
58 final Pattern bytesTypePattern = Pattern.compile(bytesTypeRegex);
Potent_app_report-main/web3j/StructuredDataEncoder.java
63 final String arrayDimensionRegex = "\\[([1-9]\\d*)?\\]";
64 final Pattern arrayDimensionPattern = Pattern.compile(arrayDimensionRegex);
Potent_app_report-main/web3j/StructuredDataEncoder.java
68 final String typeRegex = "^[a-zA-Z_$][a-zA-Z_$0-9]*(\\[([1-9]\\d*)*\\])*$";
69 final Pattern typePattern = Pattern.compile(typeRegex);
Potent_app_report-main/web3j/StructuredDataEncoder.java
71 final String identifierRegex = "^[a-zA-Z_$][a-zA-Z_$0-9]*$";
72 final Pattern identifierPattern = Pattern.compile(identifierRegex);

Suggested Solution :

Making a `public` constant just `final` as opposed to `static final` leads to duplicating its value for every instance of th...

Vulnerability

Use a logger to log this exception.

Potent_app_report-main/web3/SignCallbackJSInterface.java

106 e.printStackTrace();

107 }

Potent_app_report-main/web3/Web3TokenView.java

409 e.printStackTrace();

410 }

Potent_app_report-main/web3j/StructuredDataEncoder.java

247 e.printStackTrace();

248 hashBytes = new byte[0];

Suggested Solution :

`Throwable.printStackTrace(...)` prints a `Throwable` and its stack trace to some stream.

By default that stream `System.Err`

Executive Summary

BlockAudit not only shows you the bugs, vulnerabilities, and duplication but also shows the location of the issues and suggestions to solve them.

This helps the developer understand the reason for the issue, its potential downsides as well as the best solution for the same.

BLOCK AUDIT REPORT

Official Website
www.blockaudit.report

E-Mail
team@blockaudit.report

Twitter
https://twitter.com/BlockAudit

Github
https://github.com/blockauditreport