

To: Zencho Holdings Co., Ltd.
Please note that the translation provided above is a direct translation and may not accurately reflect the intended meaning.

**Smartphone application diagnosis.**

**(Android/iOS).**

**Report of Results**

**Olive Hill**

### September 2023.

Mitsui & Co. Secure Direction Corporation.

**Table of Contents**

1. [First of all. 2](#_bookmark0)
2. [Diagnostic Summary 3](#_bookmark1)

[Risk assessment method 3](#_bookmark2)

[Implementation Overview 3](#_bookmark3)

1. [Overall evaluation 4](#_bookmark4)

[Evaluation 4](#_bookmark5)

[Overall evaluation. 4](#_bookmark6)

1. [Detection items 6](#_bookmark7)

[4-1. List of Detected Items 6](#_bookmark8)

[Vulnerability Details 7](#_bookmark9)

1. [Deficiencies in SSL server certificate verification in communication processing. 7](#_bookmark10)
2. [QR code verification failure. 9](#_bookmark12)
3. [Important information is stored in plain text in local files. 11](#_bookmark14)
4. [Important information is stored in plain text in the local file (Cache). 12](#_bookmark17)
5. [Session ID is saved in plain text in local files. 14](#_bookmark19)
6. [Unnecessary log output. 16](#_bookmark20)
7. [Lack of logout function. 18](#_bookmark23)
8. [Unnecessary screen transition. 19](#_bookmark24)
9. [Saving and encapsulating information during development. 21](#_bookmark26)
10. [Important information is stored in plain text in local files. 23](#_bookmark28)
11. [Important information is stored in plain text in the local file (Cache). 24](#_bookmark31)
12. [Session ID is saved in plain text in local files. 27](#_bookmark33)
13. [Lack of logout function. 28](#_bookmark35)
14. [Save snapshots containing important information on the terminal. 29](#_bookmark37)
15. [Unnecessary screen transition. 31](#_bookmark38)
16. [The executable file contains an internal path. 33](#_bookmark41)
17. [Save and include information during development. 35](#_bookmark43)
18. [In conclusion. 36](#_bookmark44)

[Appendix 1. Diagnostic item list 37](#_bookmark45)

[Appendix 2 Danger level determination criteria. 39](#_bookmark46)

# First of all.

Thank you for choosing Mitsui & Co. Secure Direction Corporation (hereinafter referred to as "our company") for your security diagnosis. We have completed the diagnosis and would like to report the results to you.

This report includes explanations of the issues detected during the diagnosis, as well as possible impacts and countermeasures. We hope that you can make use of this information for considering countermeasure policies and implementing them.

Furthermore, this report contains important information regarding vulnerabilities in your company's security. We kindly request that you handle this information appropriately according to its level of importance, and take sufficient care when viewing, copying, etc., to prevent unnecessary dissemination of information. We ask that you handle this report with caution to avoid any problems that may arise if its contents are leaked after delivery. Please note that we cannot be held responsible for any such issues.

# Diagnostic Summary

## Risk assessment method

For each vulnerability detected, considering factors such as the impact on business operations, there are four levels (High, Medium, Low,).

**Info)のリスク評定を行います。具体的な評定基準は巻末の Appendix.2 危険度判定基準の表に示しまし**

Please refer to it. The assessment of the vulnerabilities detected is our own opinion and may include speculation. Please confirm if it is suitable for your company.

## Implementation Overview

#### Diagnostic target

Olive Hill

Android.

File name: app.apk
Version: 2.0.0

MD5 Hash: fd53c720c239af7ebf4c4537d490da80.

[iOS]

File name: app.ipa
Version: 5.0.0

MD5 hash: df112c6d86004dec6eb6d667c83af7da

#### Diagnostic period.

August 14, 2023 to September 1, 2023

#### Diagnostic technique

Manual/Tool-based black box diagnosis.

The list of diagnostic items is referenced in Appendix 1.

#### Diagnostic form

Remote diagnosis via the internet.

#### Use IP address.

113.43.174.224/29

153.150.126.144/29

150.249.228.96/28

114.156.13.240/28

114.156.142.98/32

#### Remarks

-

# Overall evaluation

## Evaluation

**Detected vulnerability with danger level Medium.**

Evaluation Explanation

## Overall evaluation.

**S** Vulnerabilities are not detected and it is in a highly secure state.

1. Detect only vulnerabilities with a Low level of risk.
2. Detected vulnerability with Medium risk level.
3. Detect vulnerability with danger level High.
4. Multiple types of vulnerabilities with a High level of risk were detected.

The criteria for determining the severity of individual vulnerabilities can be found in Appendix.2.

The diagnosis result revealed 1 vulnerability with Medium severity and 2 vulnerabilities with Low severity. Overall assessment.

It will become B.

In Android applications, there is a vulnerability of medium severity called "Insufficient SSL Server Certificate Verification in Communication Processing" that we have confirmed. In the communication detected by WebView, if an unintended server certificate is received, a dialog containing messages such as "SSL Certificate Error, The certificate authority is not trusted. Do you want to continue anyway?" will be displayed, and user intervention to continue or not is required. However, if the user allows communication with a server other than the legitimate one, there is a possibility of leakage or tampering of important information through a man-in-the-middle attack. To establish secure communication, it is recommended to discontinue user selection through the dialog and implement verification of the certificate issuer and hostname.

Furthermore, there are deficiencies in the verification of QR codes used for confirming the waiting order, so any web page can be used by the app.

We are detecting vulnerabilities that can be exploited for phishing and other malicious activities within the application. Please consider implementing measures such as limiting the character types that are loaded to prevent unintended web pages from being displayed and performing input validation.

We are confirming the issue of important information such as the ID and session ID for point migration handled by this application being stored in plaintext on the device for both Android and iOS. Although there is a certain hurdle for attackers to access the relevant information by physically accessing the device, it is recommended to consider measures such as not outputting unnecessary information or encrypting and storing it in order to safely manage the data.

The list and details of all the reported matters detected this time are described in the next chapter. Please review the content and consider the need for countermeasures.

# Detection items

## 4-1. List of Detected Items

The list of detected vulnerabilities is shown below. Please refer to "Appendix.2 Risk Assessment Criteria" for the risk assessment method.

#### Android

|  |  |  |  |
| --- | --- | --- | --- |
| No | Danger level | Name. | Page |
| 1 | **Medium** | Deficiencies in SSL server certificate verification in communication processing. | [7](#_bookmark11) |
| 2 | **low** | QR code verification failure. | [9](#_bookmark13) |
| 3 | **Info** | Important information is stored in plain text in local files. | [11](#_bookmark15) |
| 4 | **Info** | Important information is stored in plain text in local files (Cache) | [12](#_bookmark16) |
| 5 | **Info** | Session ID is stored in plain text in local files | [14](#_bookmark18) |
| 6 | **Info** | Unnecessary log output | [16](#_bookmark21) |
| 7 | **Info** | Lack of logout function | [18](#_bookmark22) |
| 8 | **Info** | Unnecessary screen transitions | [19](#_bookmark25) |
| 9 | **Info** | Saving and including development information | [21](#_bookmark27) |

* **iOS**

|  |  |  |  |
| --- | --- | --- | --- |
| No | Risk level | Name | Page |
| 1 | **Low** | Important information is stored in plain text in local files | [23](#_bookmark29) |
| 2 | **Info** | Important information is stored in plain text in local files (Cache) | [24](#_bookmark30) |
| 3 | **Info** | Session ID is stored in plain text in local files | [27](#_bookmark32) |
| 4 | **Info** | Lack of logout function | [28](#_bookmark34) |
| 5 | **Info** | Saving snapshots containing important information on the device | [29](#_bookmark36) |
| 6 | **Info** | Unnecessary screen transitions | [31](#_bookmark39) |
| 7 | **Info** | Internal paths are included in the executable file | [33](#_bookmark40) |
| 8 | **Info** | Saving and including development information | [35](#_bookmark42) |

(Refer to Appendix.2 for individual vulnerability risk assessment criteria)

## 4-2. Vulnerability Details

There is a possibility that the communication content may be stolen or tampered with by an attacker on the communication path.

Report Number: «ZNSH-app-1»

1. Inadequate SSL server certificate verification in communication processing

**Medium risk level, potential impact on Android**

##### Explanation

There is a flaw in the verification of the server certificate during SSL communication, allowing connections to be established with servers other than legitimate ones.

▼Example: Server certificate and server certificate hostname

In the WebView communication of this application, if an unintended server certificate or server certificate hostname is received, an "SSL Certificate Error, The certificate authority is not trusted."

Dialogs such as "Do you want to continue anyway?" are displayed. However, by allowing the user to proceed with the communication, there is a possibility of a man-in-the-middle attack as any SSL server certificate can be accepted.

The following are part of the FQDNs that are not verifying SSL server certificates when tapping on the "Takeout Order" and "Menu" tabs, which are the main functions of this application's WebView communication.

* [www.olivenooka.jp](http://www.olivenooka.jp/)
* takeout.olivenooka.jp
* ajax.googleapis.com
* [www.googletagmanager.com](http://www.googletagmanager.com/)
* stats.g.doubleclick.net
* analytics.google.com
* [www.google-analytics.com](http://www.google-analytics.com/)
* code.jquery.com

▼Target

Menu tab

Store search tab, search execution, store details

Hamburger menu, frequently asked questions

Hamburger menu, other inquiries, hamburger menu, terms of use

Hamburger menu, privacy policy, home, Olive Hill opens every day at 10:00! Home, concept

Home, menu

Home, takeout menu, home, Uber Eats

Home, Demae-can

Home, part-time staff recruitment!

Home Banner (advertisement), Olive Hill (web page)

Membership card "?" icon For more details, please check the website.

##### Countermeasures

Verify the issuer and hostname of the SSL server certificate.

##### Remarks

-

There is a possibility of exploitation for phishing or other malicious activities through apparent tampering of the page.

Report number: «ZNSH-app-2»

1. QR code verification deficiency

**Low risk, potential impact on Android**

##### Explanation

This application has a feature that allows users to check their waiting status by scanning a QR code for confirmation.

The following is an example of a QR code value used for waiting status confirmation.

[Example of QR code for waiting status confirmation]

However, in order for this application to process the corresponding QR code, the following conditions must be met.

▼Conditions

1. The QR code must contain the string "https://stag.z-navi.com/status-check.html?qr="
2. The length of the string used as the query parameter (qr) in condition (1) must be longer than 21 characters.

However, there is a deficiency in the verification of QR codes for checking waiting status, which allows any web page to be displayed in this application's WebView.

▼Reproduction Example

It is possible for an attacker to create a malicious QR code and guide the victim to scan it, thereby redirecting them to a fraudulent website.

1. The attacker creates a QR code containing the following string:

[Example of a malicious QR code]

1. The attacker guides the victim to scan the QR code created in step (1).
2. After scanning the QR code, the corresponding page is displayed in the WebView because the URL host (www.mbsd.jp) contained in the QR code is accessed.

▼Target

Home tab - Waiting order confirmation

##### Countermeasure

Do not include URLs in the string of the QR code to be read, and limit it to only the values used for waiting order confirmation. Additionally, perform input validation and consider any characters other than the ones used as errors.

##### Remarks

-

There is a possibility of leakage of the ID used for point migration.

Report Number: "ZNSH-app-3"

1. Important information is stored in plaintext in local files

**Impact expected: Android**

##### Explanation

In this application, the ID used for point migration, which is important information, is stored in plaintext in a local file on the device. Therefore, if an attacker physically accesses the device and refers to the file, there is a possibility that the information may be leaked.

▼Case: ID for point migration

The following is the relevant part of the local file where the ID used to migrate points acquired in this application is stored in plaintext.

[Contents (excerpt) of <Application Data>/shared\_prefs/oln\_app\_dev.xml]

<string name="MERGE\_CODE">PPWV2308211457247</string>

<boolean name="CLOSED\_WARNING" value="true" />

▼Target

<Application Data>/shared\_prefs/oln\_app\_dev.xml

##### Countermeasure

Implement one of the following countermeasures:

* Do not store important information on the device
* If storing important information on the device, encrypt it instead of storing it in plaintext
* Delete the corresponding local file when the application is closed, etc.

##### Remarks

In this application, backup is disabled (allowBackup="false") in the AndroidManifest.xml file, so administrator privileges are required to access the file. However, there is a possibility that the attacker may execute root access in order to access the device with administrator privileges after stealing or finding it, so it is recommended to take countermeasures.

Report number: ZNSH-app-17.

1. Important information is stored in plain text in the local file (Cache).

**Level of danger, expected impact on Android.**

There is a possibility of leakage of user's name, phone number, and email address.

##### Explanation.

In this application, important information such as the user's name and email address is stored in plain text as a cache in a local file on the device. Therefore, if an attacker gains physical access to the device and references the file, there is a possibility that the information may be leaked.

Example. User's name, phone number, email address.

The following is the relevant section of a local file where the user's name, phone number, and email address are stored in plain text.

<Application Data>/cache/WebView/Default/HTTP Cache/Cache\_Data/64f90e6996f388a7\_ 0 content (excerpt)]

00003100 20 20 20 20 20 20 20 20

00003110 64 3d 22 73 74 65 70 30

00003120 78 74 30 31 22 3e 4d 69

00003130 6f 75 3c 2f 74 64 3e 0a

20 20 20 3c 74 64 20 69

30 35 5f 30 34 5f 74 65

74 73 75 69 20 54 61 72

20 20 20 20 20 20 20 20

|

<td i|

|d="step005\_04\_te|

|xt01">Mitsui Tar|

|ou</td>. |

...(omitted)...

000031c0 20 20 20 20 20 20 20 20

000031d0 20 69 64 3d 22 73 74 65

000031e0 74 65 78 74 30 32 22 3e

000031f0 36 31 3c 2f 74 64 3e 0a

20 20 20 20 20 3c 74 64

70 30 30 35 5f 30 34 5f

30 33 35 36 34 39 31 39

20 20 20 20 20 20 20 20

|

<td|

| id="step005\_04\_|

|text02">03564919|

|61</td>. |

...(omitted)...

00003290 20 20 20 20 20 20 3c 74

000032a0 65 70 30 30 35 5f 30 34

000032b0 3e 77 65 62 73 65 63 36

000032c0 73 64 2e 6a 70 3c 2f 74

64 20 69 64 3d 22 73 74

5f 74 65 78 74 30 33 22

35 39 40 70 73 2e 6d 62

64 3e 0a 20 20 20 20 20

|

<td id="st|

|ep005\_04\_text03"|

|>websec659@ps.mb|

|sd.jp</td>. |

▼Target

<Application Data>/cache/WebView/Default/HTTP Cache/Cache\_Data/64f90e6996f388a7\_ 0

\*The file name changes each time

##### Countermeasure

Implement one of the following countermeasures.

* Do not save cache data on the device
* Delete the corresponding cache file when the application is closed, etc.

##### Remarks

Administrator privileges are required to access this file. However, it is recommended to take countermeasures as there is a possibility that the attacker may execute rooting in order to access with administrator privileges after stealing or finding the device.

Report Number: ZNSH-app-4

[5] Session ID is stored in plaintext in local files

**Risk Level Info Expected Impact Android**

If a valid session ID is leaked, there is a possibility that an attacker impersonating a legitimate user may illegally use the service.

##### Explanation

In this application, the session ID (Bearer, fueldid) is stored in plaintext in local files on the device. Therefore, if an attacker gains physical access to the device and references the file, there is a possibility that the session ID may be leaked.

▼Case 1: Session ID (Bearer)

The following is the relevant part of the local file where the session ID (Bearer) used between the server (stg.coupons-api.zensho.com) is stored in plaintext.

[Content of <Application Data>/shared\_prefs/oln\_app\_dev.xml (excerpt)]

<string name="BEARER\_TOKEN\_KEY">381703611f77159a62b94f7c1d701a647

1a4dec510953578ed130e2d1d7c9476</string>

<boolean name="IS\_LOGIN" value="true" />

▼Target 1

<Application Data>/shared\_prefs/oln\_app\_dev.xml

▼Case 2: Session ID (fueldid)

The following is the relevant part of the local file where the session ID (fueldid) used when placing a takeout order with the server (takeout.olivenooka.jp) is stored in plaintext.

[<Application Data>/app\_webview/Default/Contents of Cookies (excerpt)]

13338459585612972|takeout.olivenooka.jp||fueldid|S%3Ax62PYOyWLXeMnx\_Nic

C1jv8YeExcTbPk0HXMxA%3D||/|13338466785612972|0|0|13338459585612972|1| 1|1|-1|2|443|0|13338459585613134

13338459579694626|takeout.olivenooka.jp||lat[0]|35.6852807||/|133410515796946

07|0|0|13338459579694626|1|1|1|-1|2|443|0|13338459579694626

AOeKKH-GFZJS\_GMcp07VTgj622sSwmDOOD7QlXyD49hflBemhv6RYrEPekUyS

41wl8AhYlfmzWGvqB5M0bPyJR8fuptoYUgpmfyEjZSzJhlCsG9WdDfNhoHZHfFtca 2JByP7N-1UJIXwSASazwktQC4YzUDpnoswZkhLHQu75ZeYyVgTojogB\_RI1yu9U

Target 2

<Application Data>/app\_webview/Default/Cookies.

##### Countermeasures

Implement one of the following measures.

* Do not save authentication information on the terminal.
* If you want to store authentication information on the terminal, encrypt it instead of storing it in plain text.
* To delete local files that correspond to events such as the termination of the application.

##### Remarks

-

There is a possibility that it could be a clue to analyze the behavior of the application.

Report number: ZNSH-app-5.

Unnecessary log output.

**Level of danger, expected impact on Android.**

##### Explanation.

In this application, the stack trace is output to the log when an error occurs. Therefore, if an attacker physically accesses the terminal and refers to the log, there is a possibility that information that can be a clue to analyze the behavior of the application may be leaked.

Example. Stack trace.

The following is an example of the log output when this application is launched.

[Output log (excerpt)]

W/System.err( 2702): java.lang.NullPointerException: Attempt to invoke interface.

The method 'java.lang.Object[] java.util.Collection.toArray()' is being called on a null object reference.

W/System.err( 2702):

W/System.err( 2702): J(:6)

W/System.err( 2702):

W/System.err( 2702):

W/System.err( 2702):

W/System.err( 2702):

W/System.err( 2702):

W/System.err( 2702): a:942)

W/System.err( 2702): ava:99) W/System.err( 2702):

W/System.err( 2702):

W/System.err( 2702): java:7872) W/System.err( 2702):

W/System.err( 2702):

at java.util.ArrayList.<init>(ArrayList.java:191)

at jp.co.zensho.olivenooka.home.HomeFragment.

at i.a.a.a.q.z$a.J(:2)

at i.a.a.a.q.y$a.d(:2)

at d.a.a.a.a.d.a(:30)

at n.m$b$a.d(Unknown Source:25) at n.a.run(Unknown Source:6)

at android.os.Handler.handleCallback(Handler.jav

at android.os.Handler.dispatchMessage(Handler.j

at android.os.Looper.loopOnce(Looper.java:201) at android.os.Looper.loop(Looper.java:288)

at android.app.ActivityThread.main(ActivityThread.

at java.lang.reflect.Method.invoke(Native Method) at com.android.internal.os.RuntimeInit$MethodAn

dArgsCaller.run(RuntimeInit.java:548)

W/System.err( 2702):

nit.java:936)

at com.android.internal.os.ZygoteInit.main(ZygoteI

▼Target

Entire application

##### Countermeasure

Do not output unnecessary information such as important information and stack traces to logs.

##### Remarks

It is recommended to review the entire application when implementing countermeasures, as similar issues may occur in places other than the reported ones.

Report number: "ZNSH-app-6"

[7] Lack of logout function

**Severity Info Expected impact Android**

If a valid session ID is leaked, there is a possibility that the service may be used illegitimately by an attacker impersonating a legitimate user.

##### Explanation

There is a defect in the logout process implemented in this application.

Although this application has a login function, it does not have a logout function, and users cannot explicitly log out. Therefore, if a session ID is leaked for some reason, it is possible for an attacker to access the server using that session ID and illegitimately use the service by impersonating a legitimate user.

▼Target

Entire application

##### Countermeasure

Implement a logout function to allow users to explicitly invalidate their sessions.

##### Remarks

-

* + There is a possibility that user-entered information will be saved in plain text in a local file.

Report number: "ZNSH-app-7"

[8] Unnecessary screen transitions

**Severity Info Potential impact Android**

* + There is a possibility of accessing malicious websites.

##### Explanation

It is possible to display web pages outside of the service on the WebView of this application.

In this application, you can display any web page on the WebView of this application from links such as SNS accounts displayed on the Olive Hill web page (www.olivenooka.jp). Therefore, if you enter search strings or strings used for authentication on the web page displayed on WebView, there is a possibility that this information will be unintentionally saved in a local file. Also, unlike a browser, the URL of the displayed site is not displayed on the screen in WebView, so users may not notice that they have accessed a malicious site, and there is a possibility of being a victim of phishing, etc.

▼Target

Menu Olive Hill (web page)

Store search Nearby stores / Check stores Olive Hill (web page) Hamburger menu Frequently asked questions

Hamburger menu Other inquiries Hamburger menu Terms of use

Hamburger menu Privacy policy Initial startup Terms of use

New registration Terms of use

New registration Privacy policy

Home Takeout order Official website

Home Banner (advertisement) Olive Hill (web page) Home Olive Hill opens at 10:00 every day!

Home Concept Home Menu

Home Takeout menu Home Uber Eats

Home Demae-can

Home Part-time staff wanted!

Membership card "?" Icon For more details, please check the HP.

##### Countermeasure

Restrict the access destination URL displayed on the WebView of this application. Alternatively, use a browser for unnecessary URLs for this application's service.

##### Remarks

It is recommended to review the entire system as there is a possibility of similar issues occurring even outside the mentioned targets.

* There is a possibility of unauthorized access to the server (development environment) used for development.

Report number: "ZNSH-app-8"

[9] Save and include development information

**Severity Info Potential impact Android**

* There is a possibility of unauthorized use by leaking QR codes for point acquisition.

##### Explanation

The executable file of this application contains hard-coded development IP addresses and QR codes for point acquisition. Therefore, if the executable file is analyzed and the relevant information is leaked, there is a possibility of unauthorized access or misuse.

Example 1. Development IP address.

The following is the relevant section of the file that is believed to contain information about the IP addresses used in development.

The content of `[<APK>/res/xml/security\_config.xml]` (excerpt) is:

<?xml version="1.0" encoding="utf-8"?>

network security configuration

domain configuration clear text traffic permitted.

...(omitted)...

<domain includeSubdomains="true">52.68.217.122

</domain>

<domain includeSubdomains="true">54.64.74.199

</domain>

<domain includeSubdomains="true">54.64.59.18

</domain>

<domain includeSubdomains="true">3.115.201.116

</domain>

</domain-config>

</network-security-config>

Target 1

APK/res/xml/security\_config.xml.

Example 2. QR code for point acquisition.

The following is an example of the relevant part of the class (jp.co.zensho.olivenooka.rankup.scanqrcode.MLKitScanQRCodeRankUpActivity) where a string used as a QR code for point acquisition is hard-coded into the executable file.

jp.co.zensho.olivenooka.rankup.scanqrcode.MLKitScanQRCodeRankUpActivity example (excerpt)

public /\* synthetic \*/ void t0(View view) { l0("RJPQvq+qEwbkHDaIhkB4F8esLND83PxbDz2NtTh+B9fw+Q=OySqH38Pm`.

FDzPkgzgqV/Pw==");

}

In the given value, it is possible that although it was found to be invalid after querying the server (stg.coupons-api.zensho.com) to confirm its validity, it may be used as a valid value in the future.

Target

- APK file (class)

jp.co.zensho.olivenooka.rankup.scanqrcode.MLKitScanQRCodeRankUpActivity jp.co.zensho.olivenooka.rankup.scanqrcode.ScanQRCodeRankUpActivity

##### Countermeasures

Remove unnecessary information and files in the released version of the application.

##### Remarks

-

There is a possibility of leakage of the ID used for point migration.

Report number: "ZNSH-app-9"

[10] Important information is saved in plain text in local files.

**Low risk impact expected on iOS.**

##### Explanation

In this application, the ID used for point migration is saved in plain text in a local file on the device. Therefore, if an attacker physically accesses the device and references the file, there is a possibility that the information may be leaked.

▼Example: ID for point migration

The following is the relevant section of the local file where the ID used to migrate points acquired in this application is saved in plain text.

[Contents (excerpt) of <Application Home>/Preferences/oln\_app\_dev.xml]

<key>merge\_point\_code</key>

<string>PUDJ2308091602017</string>

▼Target

<Application Home>/Preferences/jp.co.zensho.olivenooka.stg.plist

##### Countermeasures

Implement one of the following countermeasures:

* Do not save important information on the device.
* If saving important information on the device, encrypt it instead of saving it in plain text.
* Delete the relevant local file when the application is terminated, etc.
* Save it in the keychain instead of the application area.

##### Remarks

-

Report number: "ZNSH-app-10"

[11] Important information is saved in plain text in local files (Cache).

**Low risk impact expected on iOS.**

* There is a possibility of leakage of important information such as name and phone number.
* If a valid session ID is leaked, there is a possibility that an attacker impersonating a legitimate user may use the service illegally.

##### Explanation

In this application, important information such as name and phone number is saved in plain text as a cache in a local file on the device. Therefore, if an attacker physically accesses the device and references the file, there is a possibility that the information may be leaked.

▼Example 1: Name, phone number, session ID, etc.

The following is the relevant section of the local file where customer information (name, phone number, email address) entered on the takeout lunch web order screen is saved in plain text as a cache. In addition, the session ID (fueldid) used in the relevant process is also saved as a cache.

[Extracted Response]

<div class="info\_block">

<p class="info\_block\_title">Customer Information</p>

<!-- #step005\_04 -->

<div id="step005\_04">

<!-- .info\_block\_table -->

<table class="info\_block\_table">

<tbody><tr>

<th>Name</th>

<td id="step005\_04\_text01">TANAKA Ichirou</td>

</tr>

<tr>

<th>Phone Number</th>

<td id="step005\_04\_text02">08010127251</td>

</tr>

<tr>

<th>Email Address</th>

<td id="step005\_04\_text03">websec181@ps.mbsd.jp</td>

</tr>

</tbody></table>

<!-- /.info\_block\_table -->

</div>

<!-- /#step005\_04 -->

</div>

▼Target 1

<Application Home>/Library/Caches/WebKit/NetworkCache/Version 16/Blobs/75EDDAA05F 3B7731B31F5EDD969475847E93171A

<Application Home>/Library/Caches/WebKit/NetworkCache/Version 16/Records/9B06CFFF 7145CDE8D732B7282CA6E1E8503CDEEF/Resource/20D0D6057CE5AA97D3E270E11D2

0307B2DBAA508-blob

※The directory name and file name under Records will change each time

▼Case 2. Session ID (Bearer)

The following is the relevant part of the local file where the session ID (Bearer) used in communication with the server (map-api.zensho.com, stg.coupons-api.zensho.com) is stored in plain text.

[<Application Home>/Library/Caches/jp.co.zensho.olivenooka.stg/Cache.db-wal content (partial excerpt)]

<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.co m/DTDs/PropertyList-1.0.dtd">

<plist version="1.0">

<dict>

<key>Version</key>

<integer>9</integer>

<key>Array</key>

<array>

<false/>

<dict>

<key>\_CFURLStringType</key>

<integer>15</integer>

<key>\_CFURLString</key>

<string>https://map-api.zensho.com/webapi/now?brand\_id= 39&amp;distance=20000&amp;lat=35.68347720945734&amp;limit=100&amp;lng= 139.7844892793473&amp;offset=0</string>

</dict>

<real>60.000000</real>

<integer>0</integer>

... (omitted) ...

User-Agent

Olivenooka\_Stg/5.0.0 (jp.co.zensho.olivenooka.stg; build:1; iOS 14.6.0) Alamofire/5.6.4

Authorization

Bearer 3113adeb24f830054f14345846a87b1d2186 8235b8a15c3fc85dde5747a972ca

Accept-Language

ja-JP;q=1.0

Accept-Encoding

br;q=1.0, gzip;q=0.9, deflate;q=0.8

</dict>

CFURLRequestNullTokenString

CFURLRequestNullTokenString

</array>

</dict>

</plist>

▼Target 2

<Application Home>/Library/Caches/jp.co.zensho.olivenooka.stg/Cache.db

<Application Home>/Library/Caches/jp.co.zensho.olivenooka.stg/Cache.db-wal

##### Countermeasure

Implement one of the following countermeasures.

* Do not save cache data in the device.
* Delete the corresponding cache file when the application is closed, etc.

##### Remarks

To access the file in question, administrator privileges are required. However, it is also recommended to take precautions as there is a possibility that an attacker may execute JailBreak in order to gain access with administrator privileges after stealing or finding the terminal.

Report number: ZNSH-app-11.

### Session ID is saved in plain text in local files.

##### Expected impact

**Danger level**

**Info**

iOS

There is a possibility that if a valid session ID is leaked, the service may be used improperly by an attacker who impersonates a legitimate user.

##### Explanation.

In this application, the session ID is stored in plain text in a local file on the device. Therefore, if an attacker gains physical access to the device and accesses the file, there is a possibility that the session ID may be leaked.

Example. Session ID (Bearer).

The following is the relevant part of the local file where the session ID (Bearer) used between the servers (stg.coupons-api.zensho.com, map-api.zensho.com) is stored in plain text.

Contents of jp.co.zensho.olivenooka.stg.plist (excerpt)

login token

3113adeb24f830054f14345846a87b1d21868235b8a15c3fc85dde5747a97 2ca

Target

<Application Home>/Library/jp.co.zensho.olivenooka.stg.plist

##### Countermeasures

Implement one of the following measures.

* Do not save authentication information on the terminal.
* If you want to store authentication information on the terminal, encrypt it instead of storing it in plain text.
* To delete local files that correspond to events such as the termination of the application.
* Do not save within the application area, save in the keychain.

##### Remarks

-

Report number: ZNSH-app-12.

[13] Lack of logout function.

**Potential impact of iOS danger level Info.**

There is a possibility that if a valid session ID is leaked, the service may be used improperly by an attacker who impersonates a legitimate user.

##### Explanation.

There is a flaw in the logout process implemented in this application.

This application has a login function implemented, but there is no logout function, so users cannot explicitly log out. Therefore, if the session ID is leaked for some reason, there is a possibility that someone could access the server using that session ID and illegitimately use the service pretending to be a legitimate user.

Target

The entire application.

##### Countermeasures

Implement a logout function so that users can explicitly disable the session.

##### Remarks

-

Report number: ZNSH-app-13

[14] Save snapshots containing important information on the device

**Severity Info, Potential impact iOS**

There is a possibility that the snapshots containing important information may be leaked if the device is stolen, etc., as they are saved on the device.

##### Explanation

When transitioning this application to the background, the snapshot of the screen displayed immediately before the transition is saved on the device. Therefore, if important information such as email addresses or phone numbers was displayed on the screen immediately before transitioning to the background, there is a possibility that this information may be included in the snapshot.

The following are screens where important information is saved as snapshots.

[New Registration]

Email address

[New Registration Email Address Verification] Email address

[New Registration Email Address Verification Register Profile] Gender, Residential area

[Login]

Email address

[Sidebar Menu Others Change Email Address] Email address

[Sidebar Menu Others Customer Information] Gender, Residential area

[Sidebar Menu Others Point Migration] ID for point migration

[Takeout Lunch WEB Order Enter Contact Information] Name, Email address, Phone number

[Takeout Lunch WEB Order Enter Contact Information Confirm Input] Name, Email address, Phone number

▼Target

<Application Home>/Library/SplashBoard/Snapshots/sceneID%3Ajp.co.zensho.olivenooka.s tg-default/E26B4C46-A167-4C54-8A56-F24C2D4C74A1@3x.ktx

<Application Home>/Library/SplashBoard/Snapshots/sceneID%3Ajp.co.zensho.olivenooka.s tg-default/downscaled/0E7BA265-26A1-4372-A96E-2EA3167A6208@3x.ktx

\*The file names will be different each time.

##### Countermeasures

Implement the following countermeasures when transitioning the application to the background.

* Set the hidden property of fields displaying important information to YES.
* Overwrite the entire view with another view.

##### Remarks

It is recommended to review the entire application when implementing countermeasures, as similar issues may occur in other areas not reported.

* + - There is a possibility that user-entered information will be saved in plain text in a local file.

Report number: ZNSH-app-14

[15] Unnecessary screen transitions

**Severity Info, Potential impact iOS**

* + - There is a possibility of accessing malicious websites.

##### Explanation

It is possible to display external web pages in the WebView of this application.

In this application, any web page can be displayed on the WebView of this application from links such as SNS accounts displayed on the Olive Hill web page (www.olivenooka.jp). Therefore, if search strings or authentication strings are entered on the web page displayed in the WebView, there is a possibility that this information will be unintentionally saved in a local file. Also, unlike a browser, the URL of the displayed site is not displayed on the screen in WebView, so users may not notice that they have accessed a malicious site, and there is a possibility of being a victim of phishing, etc.

Target

Home Banner (advertisement) Olive Hill (web page) Home Takeout Order Official Site

Home Olive Hill opens at 10:00 every day! Home Concept

Home Menu

Home Takeout Menu Home Uber Eats

Home Demae-kan

Home Part-time staff recruitment! Menu Olive Hill (web page)

Membership card "?" Icon For more details, please check the HP.

Store search Nearby stores / Check stores Olive Hill (web page) Hamburger menu Frequently Asked Questions

Hamburger menu Other inquiries Hamburger menu Terms of use

Hamburger menu Privacy Policy

##### Countermeasures

Restrict the access destination URL displayed in the WebView of this application. Alternatively, use a browser for unnecessary URLs for this application's service.

##### Remarks

It is recommended to review the entire system as there is a possibility of similar issues occurring even outside the mentioned targets.

Report number: "ZNSH-app-15"

[16] Internal paths are included in the executable file.

**Severity Info Expected impact iOS**

When the application is analyzed, the internal path of the development environment at the time of building may be revealed, and information about the development environment may be leaked.

##### Explanation

In this application, the internal path of the development environment is included in the executable file that is deployed after installation. This internal path may contain the login account of the PC used for building this application, so if the executable file is analyzed by an attacker, account information may be leaked.

The following is the relevant part of the executed file where the internal path of the development environment is output.

[<IPA>/Olivenooka\_Stg.app/Olivenooka\_Stg extracted string (excerpt)]

/Users/ductt7411/Documents/APP/ON/Olivenooka/ViewControllers/Splash/Controller/ONSplashViewController.swift

/Users/ductt7411/Documents/APP/ON/Olivenooka/ViewControllers/Other/Controller/ONOtherViewController.swift

/Users/ductt7411/Documents/APP/ON/Olivenooka/ViewControllers/Common/Controller/BaseViewController.swift

/Users/ductt7411/Documents/APP/ON/Olivenooka/ViewControllers/NewBackup/View/ONGuideBackupViewController.swift

▼Target

<IPA>/Olivenooka\_Stg.app/Olivenooka\_Stg

<IPA>/Frameworks/Alamofire.framework/Alamofire

<IPA>/Frameworks/DeviceKit.framework/DeviceKit

<IPA>/Frameworks/FBLPromises.framework/FBLPromises

<IPA>/Frameworks/FirebaseCore.framework/FirebaseCore

<IPA>/Frameworks/FirebaseCoreDiagnostics.framework/FirebaseCoreDiagnostics

<IPA>/Frameworks/FirebaseCrashlytics.framework/FirebaseCrashlytics

<IPA>/Frameworks/FirebaseInstallations.framework/FirebaseInstallations

<IPA>/Frameworks/FirebaseInstanceID.framework/FirebaseInstanceID

<IPA>/Frameworks/FirebaseMessaging.framework/FirebaseMessaging

<IPA>/Frameworks/Gifu.framework/Gifu

<IPA>/Frameworks/GoogleDataTransport.framework/GoogleDataTransport

<IPA>/Frameworks/GoogleUtilities.framework/GoogleUtilities

<IPA>/Frameworks/ImageSlideshow.framework/ImageSlideshow

<IPA>/Frameworks/IQKeyboardManagerSwift.framework/IQKeyboardManagerSwift

<IPA>/Frameworks/MBProgressHUD.framework/MBProgressHUD

<IPA>/Frameworks/Moya.framework/Moya

<IPA>/Frameworks/nanopb.framework/nanopb

<IPA>/Frameworks/Parchment.framework/Parchment

<IPA>/Frameworks/SDWebImage.framework/SDWebImage

<IPA>/Frameworks/SecureDefaults.framework/SecureDefaults

<IPA>/Frameworks/SnapKit.framework/SnapKit

<IPA>/Frameworks/SwiftDate.framework/SwiftDate

<IPA>/Frameworks/SwiftyJSON.framework/SwiftyJSON

<IPA>/Frameworks/YLProgressBar.framework/YLProgressBar

##### Countermeasure

To exclude the internal path of the development environment from the executable file, implement the following countermeasures.

* Set -DNDEBUG=1 to the other\_c\_flags in Xcode's build options.
* Build in Release mode instead of Debug.
* Comment or control the methods related to assertions during the build.

However, since it may depend on other settings and environments in Xcode, the above settings may not be effective. If countermeasures are difficult, build using an account on a PC that is not affected by the leakage.

##### Remarks

-

Report number: «ZNSH-app-16»

[17] Save and include development information

**Severity Info, Potential impact iOS**

There is a possibility of unauthorized use by leaking the queued QR code that is hard-coded in the executable file.

##### Explanation.

The executable file of this application contains a hard-coded string that is used as a QR code for checking the waiting order. Therefore, if the executable file is analyzed, there is a possibility that the relevant information may be leaked.

Example. QR code for checking waiting order.

The following is the relevant part in the executable file where the string used for QR code confirmation of the waiting order is hard-coded.

Olivenooka\_Stg.app/Olivenooka\_Stg content (excerpt)

qr=ZNV9419000000000000000000000202212091541
English text: qr=ZNV9419000000000000000000000202212091541

When querying the server (stag-api.z-navi.com) to confirm the validity of the above values, an invalid result was obtained, but it is possible that they may be used as valid values in the future.

Target

Olivenooka\_Stg

##### Countermeasures

Remove unnecessary information and files in the released version of the application.

##### Remarks

-

# In conclusion.

The word "cyber attack" is flowing through the mass media as if it were part of everyday life, and the frequency and sophistication of these attacks are increasing day by day. In today's world where not only information devices but everything is connected to the internet, the importance of constantly monitoring information about threats over the internet, taking appropriate measures, and implementing preventive measures is growing day by day.

In 2014, the "Cyber Basic Law" was enacted, and businesses (critical infrastructure operators) that support the foundation of our country and our daily lives are required to implement measures. However, each individual citizen is also required to understand its importance and make efforts.

We hope that this security diagnosis result can be fully utilized to enhance the security of your company's system and to maintain it, given such a background.

Please note that the security assessment conducted this time, which is a black box test, does not guarantee complete reproducibility or coverage of the reported incidents. Also, please be aware that while this report is based on information at the time of the assessment, cyber attack methods are constantly evolving, and there is no guarantee that what is secure today will remain secure tomorrow. To maintain a secure system, we recommend regularly obtaining the latest security information, applying patches, conducting periodic security assessments, and continuously monitoring for unauthorized access.

We deeply appreciate the cooperation of everyone involved in conducting this security diagnosis.

**We accept inquiries regarding the diagnosis results and report contents. Please feel free to contact us at ✉sec-support@d.mbsd.jp if you have any questions or concerns.**

|  |  |  |
| --- | --- | --- |
| **Diagnosis items** | | **Detailed diagnosis content** |
| Application interconnection | | |
|  | Sending and receiving access restriction information | Are we appropriately restricting access from unauthorized applications?  Are there no information leaks or tampering due to receiving unauthorized information?  Are we not sending important information in inappropriate ways? |
| Communication | | |
|  | Protocol | What protocol are we using for communication? |
| Encryption presence. | Are you using encrypted communication when sending and receiving important information? |
| Server certificate verification | Are you verifying the server certificate during SSL/TLS communication? |
| Communication content | Are you sending and receiving important information such as personal information and authentication information? |
| Privacy protection | Are you sending personal information to the server without proper consent? |
| Authentication | | |
|  | Authentication function | Is the authentication function implemented securely? |
| Collaboration function | Is authentication and authorization information not stolen through application collaboration? |
| Logout function | Is the logout function properly implemented? |
| Handling of data within the device | | |
|  | Storage location | Are you not storing important information in shared areas? |
| Access permissions | Are file access permissions properly set? |
| Storage method | Are you encrypting important information when storing it? |
| Storage period | Are you deleting information within the device at appropriate timing? |
| Application files and logs | | |
|  | Presence of unnecessary information | Are there any remaining information related to development environments, test environments, or developers? |
| Presence of unnecessary information output | Are you outputting important information or information that can be clues for analysis? |

**Appendix.1 List of diagnostic items**

[Android/iOS]

|  |  |  |
| --- | --- | --- |
|  |  | Not present |
| Use of functions | | |
|  | Permission settings | Are you not registering unnecessary permissions that the application does not use? |

**Appendix.2 Risk assessment criteria**

[Android/iOS]

|  |  |  |
| --- | --- | --- |
| Severity | Potential damage | Examples |
| **High** | Leakage of important information due to network attacks from external devices | The server started by the application is attacked over the network, resulting in leakage of important information |
| Leakage of information handled by other applications due to attacks from third-party applications. Or, it affects the use of services | Leakage of important information from other applications as a starting point of the application |
| Removal of function restrictions and unauthorized use of the application | Unauthorized use of paid version functions by removing function restrictions of free version applications |
| **Medium** | Leakage of important information due to attacks from third-party applications | The state where important information can be read by other applications |
| Leakage of important information to third-party applications during collaboration | Application implicitly broadcasting and sending important information |
| Leakage of important information due to man-in-the-middle attacks (eavesdropping, etc.) | Sending important information in plain text  Failure to properly verify server certificates during SSL/TLS communication |
| **Low** | Leakage of important information in the device due to attackers who can directly access the device | Storing important information in the device or logs without encryption |
| Displaying unauthorized information on the application due to attacks from third-party applications. | Displaying information received from unauthorized applications in dialogs or notifications. |
| **Info** | Although the potential for direct harm is low, it is desirable to fix this issue. | Hardcoding information that can be used as clues when analyzing the application. |