**金证股份 • <KOCA>**

**安全测试报告**

|  |  |  |  |
| --- | --- | --- | --- |
| **公司名称** | 深圳市金证科技股份有限公司 | **文档编号** |  |
| **文档名称** | KOCA v2.6.0安全测试报告 | **文档版本** | 1.0 |
| **起 草** | 周阳 | **起草日期** | 2021-09-22 |
| **审 批** | <审批人> | **审批日期** | YYYY-MM-DD |

**修订历史**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本号 | 日期 | 状态 | 修订人 | 摘要 |
| 1.0 | 2021/09/22 | C | 周阳 | 创建安全测试报告 |
|  |  |  |  |  |
|  |  |  |  |  |

状态标识：C – Created A - Added M - Modified D - Deleted

**目录**

[**1** **引言** 4](#_Toc79586841)

[**1.1 目的和范围** 4](#_Toc79586842)

[**1.2 背景** 4](#_Toc79586843)

[**2** **安全测试方案** 5](#_Toc79586844)

[**2.1 测试内容** 5](#_Toc79586845)

[**2.2 测试计划** 5](#_Toc79586846)

[**2.3 测试人员及工作量汇总** 5](#_Toc79586847)

[**3** **安全测试过程** 6](#_Toc79586848)

[**3.1 安全流水线测试** 6](#_Toc79586849)

[**3.1.1 第三方依赖安全检测** 6](#_Toc79586850)

[**3.1.2 Fortify扫描** 9](#_Toc79586851)

[**3.2 web安全扫描** 11](#_Toc79586852)

[**3.2.1 web安全扫描结果总结 11**](#_Toc79586853)

[**3.2.2 web安全测试结果分析 12**](#_Toc79586854)

[**3.2.2.1 admin中检出安全漏洞分析 12**](#_Toc79586855)

[**3.2.2.2 studio中检出安全漏洞分析 13**](#_Toc79586856)

[**3.2.2.3 amo中检出安全漏洞分析 13**](#_Toc79586857)

[**4** **修复建议** 13](#_Toc79586858)

1. **引言**

**1.1 目的和范围**

本次安全测试为检测KOCA后台Java代码的安全漏洞和前端门户网站中的攻击点，以及验证KOCA云平台各项功能设计中是否存在安全风险。覆盖模块koca-base、koca-admin、koca-studio、koca-amo、koca-cloud、koca-devops。

**1.2 背景**

KOCA v2.6.0 为KOCA项目最新一个迭代输出， 本报告旨在总结对KOCA各模块安全测试中检出的安全问题，并提出修复建议，以提升平台运行的安全性，被成功入侵的风险降低到最小。

1. **安全测试方案**

**2.1 测试内容**

1. 后台Java代码

Koca-admin、koca-studio、Koca-amo、Koca-cloud、Koca-base、Koca-devops。

1. 前端业务门户

AMO: http://10.202.63.16:4000/#/login?redirect=%2Fhome  
STUDIO：http://10.202.63.7:8000/#/login?redirect=%2Fhome  
ADMIN: http://10.202.63.5:7000/#/login?redirect=%2Fhome

**2.2 测试计划**

（1）使用koca-security-test流水线对后台Java代码中的第三方组件和代码本身进行安全扫描，第三方组件安全性检测使用dependency-check-6.2.2，代码安全性分析使用Fortify20.1.1，对所得检测报告中报出漏洞进行人工验证；

（2）分别使用AppScan10、Awvs13、Xray、Netsparker、OWASP zap对KOCA三个测试环境下的门户网站进行扫描，对工具检出问题进行手动测试，排除误报和漏报；

（3）手动测试其他可能的风险点。

**2.3 测试人员及工作量汇总**

安全测试参加人员：周阳

测试时间：2021-09-14至2021-09-22

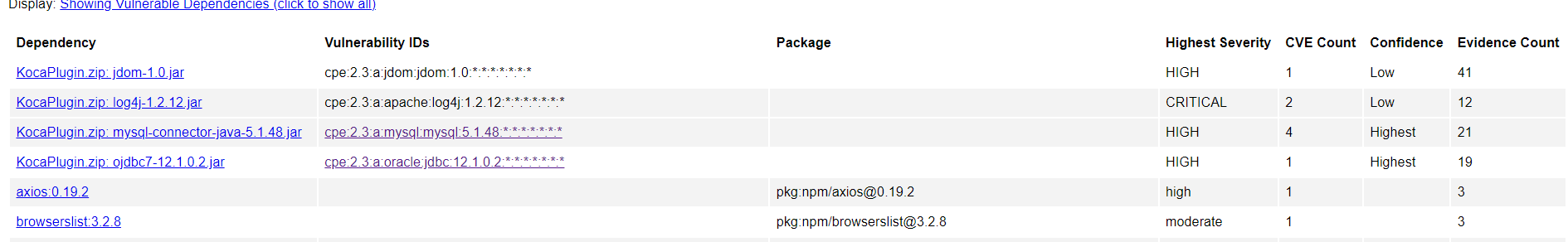
实际工作量如下表所表示：

|  |  |  |
| --- | --- | --- |
| **测试人员** | **估计工作量（日）** | **实际工作量（日）** |
| 周阳 | 5 | 5 |

1. **安全测试过程**

**3.1 安全流水线测试**

**3.1.1 第三方依赖安全检测**



注：标蓝的序号对应的组件暂未完成升级，其余均在2.6做出修复。

后端组件漏洞统计

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 组件名 | 路径 | 漏洞等级 | 修复建议 |
| 1 | log4j-core-2.8.jar | koca-base  kocaPluginV2.5.0-release.zip | CRITICAL | 升级到  log4j-core-2.13.2.jar或以上 |
| 2 | koca-encrypt-1.7.0.jar:jackson-databind-2.10.1.jar | koca-base  koca-encrypt-1.7.0.jar | HIGH | 升级到  jackson-databind-2.11.0或以上 |
| 3 | koca-encrypt-1.7.0.jar:snakeyaml-1.25.jar | koca-base  koca-encrypt-1.7.0.jar | HIGH | 升级到  snakeyaml-1.27.jar或以上 |
| 4 | koca-encrypt-1.7.0.jar:spring-aop-5.2.1.RELEASE.jar | koca-base  koca-encrypt-1.7.0.jar | HIGH | 升级到  spring-aop-5.2.9.RELEASE.jar或以上 |
| 5 | koca-encrypt-1.7.0.jar:spring-core-5.2.1.RELEASE.jar | koca-base  koca-encrypt-1.7.0.jar | HIGH | 升级到  spring-core-5.2.9.RELEASE.jar或以上 |
| 6 | kocaPluginV2.6.0-release.zip: commons-beanutils-1.8.0.jar | kocaPluginV2.6.0-release.zip | HIGH | 升级到  commons-beanutils-1.9.4.jar或以上 |

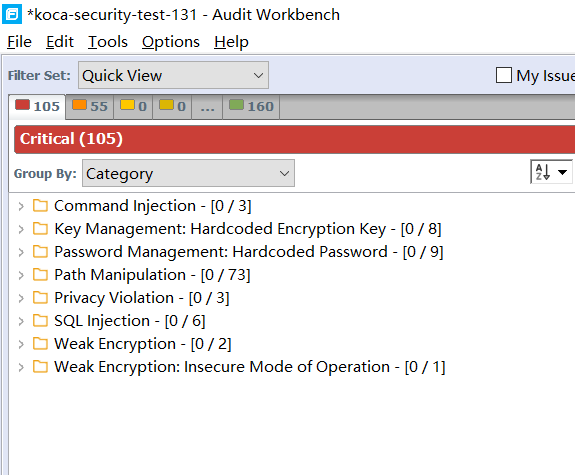
处理结果：log4j-core已升级，koca-encrypt-1.7.0.jar已被删除。

前端组件漏洞统计

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 组件名 | 路径 | 漏洞等级 | 修复建议 |
| 1 | angular-animate.min.js:  1.2.13 | koca-admin  angular\_1.2.13/angular-animate.min.js | medium | 升级到  angular-animate.min.js  1.8.0或以上 |
| 2 | [angular-cookies.min.js](file:///C:\Users\pouo\Desktop\jenkins-koca-code-compile-test-koca-security-test-131.html#l34_092083ea0af70b5ace48365d13523b4c026e8fe1):  1.2.13 | koca-admin  angular\_1.2.13/[angular-cookies.min.js](file:///C:\Users\pouo\Desktop\jenkins-koca-code-compile-test-koca-security-test-131.html#l34_092083ea0af70b5ace48365d13523b4c026e8fe1) | medium | 升级到  angular- cookies.min.js  1.8.0或以上 |
| 3 | angular-resource.min.js:  1.2.13 | koca-admin  angular\_1.2.13/[angular- resource.min.js](file:///C:\Users\pouo\Desktop\jenkins-koca-code-compile-test-koca-security-test-131.html#l34_092083ea0af70b5ace48365d13523b4c026e8fe1) | medium | 升级到  angular- resource..min.js  1.8.0或以上 |
| 4 | angular-route.min.js:  1.2.13 | koca-admin  angular\_1.2.13/[angular- route.min.js](file:///C:\Users\pouo\Desktop\jenkins-koca-code-compile-test-koca-security-test-131.html#l34_092083ea0af70b5ace48365d13523b4c026e8fe1) | medium | 升级到  angular- route.min.js  1.8.0或以上 |
| 5 | angular-sanitize.min.js:  1.2.13 | koca-admin  angular\_1.2.13/[angular- sanitize.min.js](file:///C:\Users\pouo\Desktop\jenkins-koca-code-compile-test-koca-security-test-131.html#l34_092083ea0af70b5ace48365d13523b4c026e8fe1) | medium | 升级到  angular- sanitize.min.js  1.8.0或以上 |
| 6 | angular.min.js: 1.2.13 | koca-admin  angular\_1.2.13/[angular.min.js](file:///C:\Users\pouo\Desktop\jenkins-koca-code-compile-test-koca-security-test-131.html#l34_092083ea0af70b5ace48365d13523b4c026e8fe1) | medium | 升级到  angular.min.js  1.8.0或以上 |
| 7 | axios:0.19.2 | koca-amo  amo-classic/yarn.lock?axios | high | 升级到  axios.0.21.1  或以上 |
| 8 | bootstrap.js:3.1.1 | koca-admin  /bootstrap\_3.1.1/js/bootstrap.js | medium | 升级到  bootstrap:4.1.2或以上 |
| 9 | bootstrap.min.js:3.1.1 | koca-admin  /bootstrap\_3.1.1/js/bootstrap.  min.js | medium | 升级到  bootstrap.min:4.1.2或以上 |
| 10 | browserslist:3.2.8 | koca-amo  amo-classic/yarn.lock?browserslist | moderate | 升级到  browserslist:4.16.5或以上 |
| 11 | css-what:2.1.3、css-what:3.4.2 | koca-amo  amo-msa-basic/yarn.lock?css-what | high | 升级到  css-what:5.0.1或以上 |
| 12 | dns-packet:1.3.1 | koca-amo  /amo-classic/yarn.lock?dns-packet | high | 升级到  dns-packet:5.2.2或以上 |
| 13 | elliptic:6.5.3 | koca-amo  amo-modules/amo-classic/yarn.lock?elliptic | moderate | 升级到  elliptic:6.5.4或以上 |
| 14 | glob-parent:5.1.1 | koca-amo  /amo-basic/yarn.lock?glob-parent | moderate | 升级到  glob-parent:5.1.2或以上 |
| 15 | hosted-git-info:2.8.8 | koca-amo  /amo-classic/yarn.lock?hosted-git-info | moderate | 升级到  hosted-git-info:3.0.8或以上 |
| 16 | ini:1.3.5 | koca-amo  amo-modules/amo-classic/yarn.lock?ini | low | 升级到  ini:1.3.6或以上 |
| 17 | jquery.js:1.11.0 | koca-admin  editor-app/libs/jquery\_1.11.0/jquery.js | medium | 升级到  jquery.js:3.5.0或以上 |
| 18 | jquery.min.js:1.11.0 | koca-admin  editor-app/libs/jquery\_1.11.0/jquery.min.js | medium | 升级到  jquery.min.js:3.5.1或以上 |
| 19 | lodash:4.17.20 | koca-amo  amo-classic/yarn.lock?lodash | high | 升级到  lodash4.17.21或以上 |
| 20 | minimist:0.0.8 | koca-amo  amo-classic/yarn.lock?minimist | low | 升级到  minimist0.2.1,1.2.3或以上 |
| 21 | momentjs.min.js:2.5.1 | koca-admin  /editor-app/libs/momentjs\_2.5.1/momentjs.min.js | low | 升级到momentjs.min.js：2.11.2或以上 |
| 22 | postcss:6.0.23 | koca-amo  /amo-modules/amo-classic/yarn.lock?postcss | moderate | 升级到postcss:8.2.10或以上 |
| 23 | prototype-1.5.1.js、prototype.1.6.0js | koca-admin  /editor-app/libs/prototype-1.5.1.js  /static/wfapp/editor-app/libs/prototype.js | high | 升级到prototype.js:1.6.0.2或以上 |
| 24 | serialize-javascript:2.1.2 | koca-amo  /amo-classic/yarn.lock?serialize-javascript | high | 升级到  serialize-javascript:3.1.0或以上 |
| 25 | ssri:5.3.0 | koca-amo  /amo-modules/amo-classic/yarn.lock?ssri | moderate | 升级到  ssri:6.0.2, 7.1.1, 8.0.1 或以上 |
| 26 | tar:2.2.2 | koca-amo  amo-modules/amo-basic/yarn.lock?tar | high | 升级到  tar:4.4.18, 5.0.10, 6.1.9或以上 |
| 27 | trim-newlines:1.0.0 | koca-amo  /amo-modules/amo-classic/yarn.lock?trim-newlines | high | 升级到  trim-newlines :3.0.1、4.0.1 或以上 |
| 28 | underscore:1.11.0、  underscore:1.12.0 | koca-amo  /amo-modules/amo-classic/yarn.lock?underscore | high | 升级到  underscore：1.12.1 or 1.13.0-2  或以上 |
| 29 | url-parse:1.4.7 | koca-amo  /amo-modules/amo-classic/yarn.lock?url-parse | high | 升级到  url-parse:1.5.0或以上 |
| 30 | wangeditor:3.1.1 | koca-amo  /amo-modules/amo-classic/yarn.lock?wangeditor | high | 暂无安全版本，3.1.1已为最新版 |
| 31 | ws:6.2.1 | koca-amo  /amo-modules/amo-classic/yarn.lock?ws | moderate | 升级到  ws:6.2.2 or 7.4.6或以上 |
| 32 | y18n:3.2.1 | koca-amo  /amo-modules/amo-classic/yarn.lock?y18n | high | 升级到  y18n:3.2.2, 4.0.1, 5.0.5或以上 |

**3.1.2 Fortify扫描**

使用Fortify(规则库为2020年6月)对KOCA中Java代码进行扫描，结果汇总如下图，对其中的High和Critical类漏洞逐一排查。



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 漏洞类型 | 数量 | 排除误报后漏洞数 | 具体分析 |
| 1 | Command Injection（命令注入） | 3 | 0 | 存在于GeneratePg.actionPerformed()  ShowInfoUtils.opendir()的调用中，检查为正常 |
| 2 | Key Management:Hardcoded Enryption Keu(密钥管理：硬编码) | 8 | 1 | koca-base NumberConstants.java  /\*\*       \* 默认密钥       \*/      public static final String KEY = "We@53&es&esT\*7%s"; |
| 3 | Password Mangement:Hardcoded Password密码管理：硬编码） | 9 | 0 | (经证实为测试使用，不携带真实账号，密码，无影响)  koca-base JdbcUtil.java  url = "jdbc:oracle:thin:@127.0.0.1:1521/orcl";  user = "koca";  pwd = "1";  driver = "oracle.jdbc.OracleDriver";  koca-studio JdbcUtil.java,同上  koca-studio PostHttp.java  String pwd = "123456";  Map<String, Object> resultMap = sendGetApi("http://10.202.63.32:8000/auth/randomCode", request); |
| 4 | Privacy Violation(隐私泄露) | 3 | 0 | 均为误报 |
| 5 | Weak Encryption(弱加密) | 2 | 0 | koca-base DesCryptoAlgorithm.java  DES当前被认为是弱加密算法,  DesCryptoAlgorithm未见调用 |
| 6 | Weak Encryption：Insecure Mode of Operation(弱加密) | 1 | 0 | koca-base DesCryptoAlgorithm.java  DES当前被认为是弱加密算法 |
| 7 | SQL Injection(SQL注入) | 6 |  | koca-admin ConnectionWrapper.java  OracleDataProvider.java  StatementWrapper.java  TscConnectionProxy.java  koca-studio SqlQryCheck.java |
| 8 | Path Manipulation(路径篡改) | 73 | 0 | 除非能用通过攻击修改文件操作路径，否则该漏洞无法被利用 |

**3.2 web安全扫描**

## 3.2.1 web安全扫描结果总结

使用AppScan10、Awvs13、Xray、OWAsp zed和Netsparker对koca三个测试环境下的门户网站进行扫描，结果如下：

**（1）业务门户admin（http://10.202.63.5:7000/#/login?redirect=%2Fhome）**

|  |  |  |  |
| --- | --- | --- | --- |
| 检测工具 | 漏洞报告数 | 排除误报剩余漏洞数 | 漏洞记录 |
| AppScan10 | 20 | 1 | 1. CSRF（跨站请求伪造，已修复） 2. nginx1.17.5版本过期，存在漏洞 |
| Awvs13 | 7 | 0 |
| Xray | 0 | 0 |
| OWASP zed | 0 | 0 |
| Netsparker | 5 | 1 |

**（2）开发门户studio（http://10.202.63.7:8000/#/login?redirect=%2Fhome）**

|  |  |  |  |
| --- | --- | --- | --- |
| 检测工具 | 漏洞报告数 | 排除误报剩余漏洞数 | 漏洞记录 |
| AppScan10 | 43 | 1 | 1. CSRF（跨站请求伪造，已修复） 2. Springboot actuator接口信   息暴露  （http://10.202.63.7:8000/  actuator/env） |
| Awvs13 | 8 | 1 |
| Xray | 0 | 0 |
| OWASP zed | 0 | 0 |
| Netsparker | 5 | 1 |

1. **AMO门户amo（http://10.202.63.16:4000/#/login?redirect=%2Fhome）**

|  |  |  |  |
| --- | --- | --- | --- |
| 检测工具 | 工具报告漏洞数 | 排除误报剩余漏洞数 | 漏洞记录 |
| AppScan10 | 128 | 1 | 1. CSRF（跨站请求伪造，已修复） |
| Awvs13 | 8 | 1 |
| Xray | 0 | 0 |
| OWAsp zed | 6 | 0 |
| Netsparker | 4 | 1 |

## 3.2.2 web安全测试结果分析

## 3.2.2.1 admin中检出安全漏洞分析

地址：http://10.202.63.5:7000/

1. CSRF（跨站请求伪造，koca2.6已完成修复）

|  |
| --- |
| 什么是CSRF?  CSRF（Cross-site request forgery）跨站请求伪造：攻击者诱导受害者进入第三方网站，在第三方网站中，向被攻击网站发送跨站请求。利用受害者在被攻击网站已经获取的注册凭证，绕过后台的用户验证，达到冒充用户对被攻击的网站执行某项操作的目的。  一个典型的CSRF攻击有着如下的流程：   * 受害者登录a.com，并保留了登录凭证（Cookie）。 * 攻击者引诱受害者访问了b.com。 * b.com 向 a.com 发送了一个请求：a.com/act=xx。浏览器会默认携带a.com的Cookie。 * a.com接收到请求后，对请求进行验证，并确认是受害者的凭证，误以为是受害者自己发送的请求。 * a.com以受害者的名义执行了act=xx。 * 攻击完成，攻击者在受害者不知情的情况下，冒充受害者，让a.com执行了自己定义的操作。   CSRF Token  前面讲到CSRF的另一个特征是，攻击者无法直接窃取到用户的信息（Cookie，Header，网站内容等），仅仅是冒用Cookie中的信息。  而CSRF攻击之所以能够成功，是因为服务器误把攻击者发送的请求当成了用户自己的请求。那么我们可以要求所有的用户请求都携带一个CSRF攻击者无法获取到的Token。服务器通过校验请求是否携带正确的Token，来把正常的请求和攻击的请求区分开，也可以防范CSRF的攻击。 |

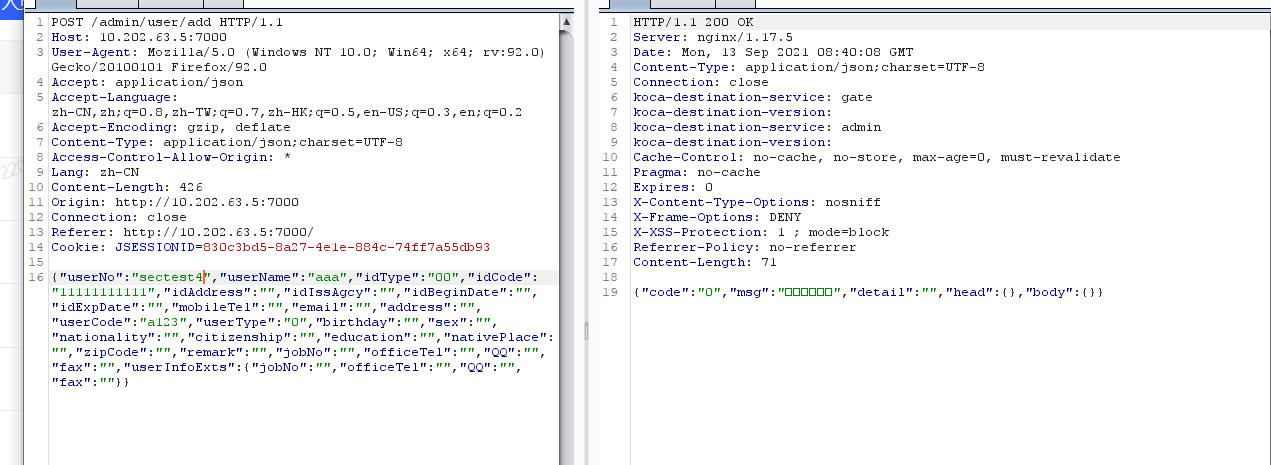
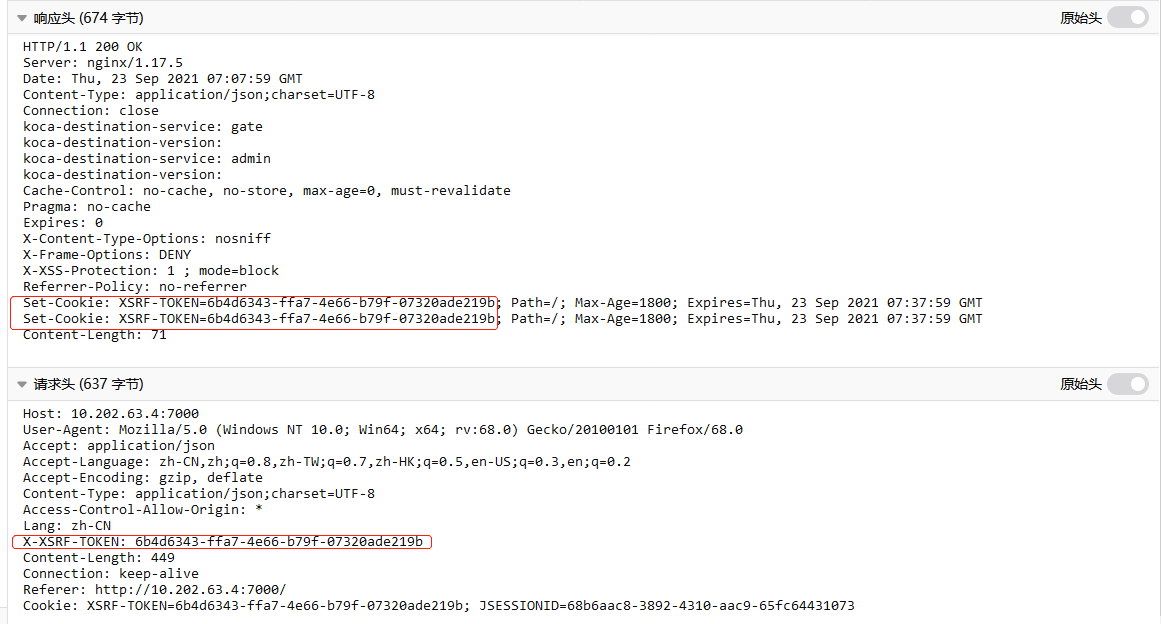


图3-1 admin中人员信息管理添加用户

此前，从添加用户请求和响应中可以看出此前未做针对CSRF的防护。



目前已添加CSRF-TOKEN作为防护。

1. Nginx为1.17.5版本，存在标号为CVE-2021-23017的任意代码执行漏洞，级别为高危。需要将Nginx升级到1.20.1或1.21.0等安全版本或进行打补丁，补丁链接：http://nginx.org/download/patch.2021.resolver.txt；

## 3.2.2.2 studio中检出安全漏洞分析

地址：http://10.202.63.7:8000

（1）springboot actuator接口信息暴露





（2）studio中同样存在CSRF漏洞，koca2.6已完成修复。

## 3.2.2.3 amo中检出安全漏洞分析

amo中同样存在CSRF漏洞，koca2.6已完成修复。

1. **修复建议**
2. 对剩余存在漏洞的第三方组件在后续版本升级到安全的版本；
3. 修复Java代码中密钥、密码硬编码问题；
4. 关闭studio中springboot的actuator的mappings、configprops、env、heapdump、loggers、conditions、beans这7个接口；
5. 在后续版本注意Nginx升级，由当前Nginx1.17.5升级到安全版本1.20.1或1.21.0或以上。