京东外部威胁处理规则

编写人	京东安全应急响应中心
版本号	V5.0
适用范围	通过（http://security.jd.com，以下简称 JSRC）反馈平台收到的所有威胁报告
最后更新时间	2017-09-04
修订记录	V1.0 首发：2013-04-12 V2.0 修改评分标准并提高奖励力度，更新评级规范：2014-07-29 V3.0 修改评分标准并提高奖励力度；更新评级规范：2015-03-11 V4.0 更新评分标准并提高奖励力度；更新评级规范：2015-05-18 V5.0 新增和更新评分标准细则、范围等，新增 FAQ：2017-08-29

目录

基本原则

1) 京东非常重视安全威胁信息并关注安全问题本质，我们承诺，对每一位报告者的问题提供专人跟进、分析和处理，并及时予以答复或反馈。

2) 京东支持合作式的漏洞披露和处理，对于每一位恪守白帽子精神、保护人民群众安全利益、帮助京东提升安全质量的报告者，我们将给予以感谢和回馈。

3) 京东反对和谴责一切以安全检测为由，利用安全漏洞进行破坏、损害用户利益的黑客行为。包括但不限于利用漏洞盗取用户数据及财产、入侵业务系统、恶意传播等行为。

4) 京东认为，每个安全问题的处理及整个安全行业的进步，都离不开各方人士的共同推动与合作，希望企业、安全公司、安全组织和研究者一起加入到“合作式的安全报告披露与处理”中来，为建设良好的互联网安全生态而努力。

威胁反馈与处理流程

【预报告阶段】

报告者访问 JSRC 平台并登录或注册账号。

【报告阶段】

报告者登录京东漏洞反馈平台，提交威胁信息（状态：未处理）

【处理阶段】

1） 一个工作日内 JSRC 工作人员确认报告并跟进评估（状态：漏洞验证/忽略）

2）三个工作日内 JSRC 工作人员处理、给出结论并计分（状态：漏洞确认/漏洞处理）。必要时会与报告者沟通确认，请报告者予以协助。

【修复阶段】

业务部门修复所报告的问题并安排修复上线（状态：已修复）。修复时间根据问题严重程度、修复难度和业务情况而定，一般严重和高危报告会在 24 小时内，中风险 7 个工作日内，低风险 14 个工作日内，客户端和特殊业务因发版及其他限制情况，会根据实际情况做修复。

【*复查阶段*】

可对已修复的问题做复查，若问题仍存在，可再次提交反馈。JSRC 工作人员会对该问题及工单做审查确认，并再次计分或处理。

威胁报告评分标准

JSRC 威胁报告主要包含漏洞报告、情报、通用组件和插件三个报告内容，下面会对每个部分的规则做评分说明。

特殊现金奖励（积分/人民币）：

1 万~50 万：核心业务涉及重要数据和影响巨大的安全问题（可以是漏洞、情报、通用组件或插件），该奖励会不定期做评定和发布。

漏洞报告评分标准

根据报告危害程度分为严重、高危、中危、低危、无（忽略）五个等级，每个等级涵盖的评分标准如下：

严重：【系数：120】

分值　**9~10** 分：

1、直接获取重要系统权限（服务器端权限、客户端权限）的漏洞。包括但不限于：命令注入、远程命令执行、上传获取 WebShell、SQL 注入获取系统权限、缓冲区溢出（包括可利用的 ActiveX 缓冲区溢出）。

2、直接导致重要业务拒绝服务的漏洞。包括通过该远程拒绝服务漏洞直接导致线上应用、系统、服务器无法继续提供服务的漏洞。

3、重要业务的严重逻辑设计缺陷和流程缺陷。包括但不限于任意账号登录和密码修改、任意账号资金消费、订单详细泄露、支付交易流程的漏洞。

4、严重级别的敏感信息泄露。包括但不限于核心 DB 的 SQL 注入漏洞、包含公司、用户敏感数据的接口引发的信息泄露。

高危：【系数：60】

分值　**7~8** 分：

1、直接获取非重要系统权限（服务器端权限、客户端权限）的漏洞。包括但不限于：命令注入、远程命令执行、上传获取 WebShell、SQL 注入获取系统权限、缓冲区溢出（包括可利用的 ActiveX 缓冲区溢出）。

2、越权访问和操作。包括但不限于绕过认证直接访问管理后台可操作、核心业务非授权访问、核心业务后台弱密码，增删查改任意用户信息或状态等重要交互的越权行为。

3、敏感信息泄漏漏洞。包括但不限于源代码压缩包泄漏、包括但不限于可直接利用的敏感数据泄露。

4、可直接盗取公司、用户私密数据、有大范围影响的漏洞。包括重点页面的存储 XSS，普通站点的 SQL 注入。

5、可远程获取客户端权限的漏洞。包括但不限于远程任意命令执行、远程缓冲区溢出及其它逻辑问题导致的客户端漏洞。

中危：【系数：15】

分值　**4~6** 分：

1、普通信息泄露。包括但不限于未涉及敏感数据的 SQL 注入，重要客户端明文密码存储、包含敏感信息的源代码等 log 信息。

2、需交互和前置条件才能获取用户身份信息的漏洞。包括但不限于反射型 XSS、JSON Hijacking、重要业务和交互的 CSRF、一般业务的存储型 XSS。

3、普通的逻辑缺陷和越权。包括但不限于非重要交互和业务的越权行为。

低危：【系数：9】

分值　**1~3** 分：

1、 轻微信息泄露，包括但不限于路径、SVN 信息泄露、PHPinfo、异常和含有敏感字段的调试信息，本地 SQL 注入、日志打印及配置等泄露情况。

2、 只在特定情况之下才能获取用户信息的漏洞，包括但不限于非流行浏览器环境下触发的反射 XSS（包括 DOM 型）、边缘业务的存储 XSS 等。

3、利用有难度但存在安全隐患的漏洞，包括但不限于可引起传播的 Self-XSS，敏感操作但利用条件苛刻的 CSRF，提供有效 PoC 需借助中间人攻击的安全隐患。

无危害（忽略）：不计分

1、 无法利用/无实际危害的漏洞。包括但不限于 Self-XSS、非重要交互的 CSRF、静态文件目录遍历、401 认证钓鱼、内网 IP/域名、无敏感信息的 JSON Hijacking 等。

2、 不能重现的漏洞。包括但不限于经 JSRC 审核者多次确认无法重现的漏洞。

3、 内部已知、正在处理的漏洞，包括但不限于如 Discuz!等已在其他平台公开通用的，白帽子、内部已发现的漏洞。

4、非接收范围或对实际业务无影响的 Bug（严重的 bug 评分确认可参考通用原则）。包括但不限于产品功能缺陷、页面乱码、样式混乱。

5、不接收无实际意义的扫描器结果报告。

6、无证据支持的情况，包括但不限于账号被盗即表示有漏洞。

情报评分标准

一、情报关键线索：

关键词	示例
谁？	可以是具体人或组织的信息，如：联系方式及团体交流渠道（如 QQ 群、社区、YY 等）；
什么场景？	包括但不限于，参考《情报收集场景》，
步骤和流程？	详细的流程步骤，实现该攻击的方式。如该一个有组织的活动秒杀情报：攻击者怎么收集该活动，先做什么后做了什么，怎么实现该秒杀等。

二、情报收集场景参考（包括但不限于）

技术情报	业务情报
服务器被入侵且提供了入侵行为特征等关键线索	对有组织有规模的盗取和售卖京东账号、订单数据等行为提供关键线索
重要业务数据库被脱并提供了数据库名或文件等关键线索	对大批量的刷单、注册小号、套现等行为提供关键线索
支付业务逻辑漏洞、流程绕过等关键线索	对泄露公司内部数据、用户数据等行为提供关键线索
蠕虫传播、流量劫持等提供源链接等关键线索	对有针对性的秒杀、刷积分、绕过现有认证或者流程的事件提供关键线索
用户敏感数据大规模被窃取且提供了攻击代码、漏洞利用工具等关键线索	对业务范围内的活动恶意利用提供关键线索
能够帮助完善防御系统，新型攻击方式、技术等提供详细分析	对正常商品标价错误、删差评、结合内部人员修改正常业务规则等情况提供关键线索

三、情报评分标准

参考《漏洞评分标准》结合实际影响、业务等级和情报线索是否完整综合评分。

通用组件和扫描插件评分标准

一、通用标准

1、 参考《开源软件清单》，但不限于该清单

2、 漏洞必须是中危及以上级别漏洞

3、 需要提供至少 3 个存在此漏洞的网站，以实例证明危害

4、 符合扫描插件编写规范

5、 凡是在 JSRC 提交的漏洞扫描插件，不可在其它平台上提交，如有发现，扣除所有积分

6、 同一漏洞扫描插件有多位白帽子提交，只奖励首个提交的白帽子

7、 对于影响巨大的报告会给予额外的现金奖励（最高额度 50 万 RMB）并且 JSRC 会以威胁报告者的名义向该组件官方发出报告，帮助其改进软件安全性

二、编写规范

1、 推荐使用 Lua，Python，Go 语言编写

2、 不得对目标系统直接或者间接造成损害，如造成宕机、数据删除等严重后果，责任均由插件作者承担

3、 插件误报率不可超过 5%

4、 可参照插件模板编写，具体参考页尾：poc_demo.lua，有任何疑问加 QQ 群：63532669

5、 插件代码质量

i. 程序逻辑清晰

ii. 代码可读性强

iii. 编码规范

iv. 有完善的注释

三、评分标准

通用组件和漏洞扫描插件是根据漏洞影响范围、漏洞危害级别、插件代码质量 3 个标准进行评分。

注：影响范围和危害级别根据业务范围内的现有信息资产进行评级，危害级别参考《漏洞报告评分标准》。

四、开源软件清单

通用软件漏洞奖励计划适用于各种业务范围内使用的通用软件，优先但不限于以下列表：

Web 服务器：Nginx、Tomcat、Apache

操作系统： Linux、IOS、Android、Windows

开源框架/组件：Spring、Struts、OpenSSL

开发语言：JAVA、PHP、Python、C++

数据库系统：MySQL、Redis、Oracle

云/虚拟化：Kubernetes、Qemu 、KVM

分值计算

威胁危害分 x 系数 = 最终分数

例：严重漏洞 10 分 x 系数 120 = 1200 分

注：威胁分值一般均会按照严重、高危、中危、低危四个等级对应分值给予。但也会根据实际影响范围、业务等级和威胁报告本质等因素做综合调整。

业务范围及通用原则

业务范围

（包括但不限于）

京东商城：

.jd.com、.jd.hk、*.jd.ru 、*.jd.id 等;

京东金融：

.jr.jd.com、.baitiao.com *.jdpay.com 等;

京东云：

.jcloud.com、.jcloudec.com 等;

1 号店：

.yhd.com、.yihaodian.com、*.ımall.com 等;

京东到家：

*.jddj.com 等;

通用原则

1、 同一威胁报告（包括情报、通用组件和插件）最早提交者得分，提交网上已公开的报告不计分。

2、 当同一威胁报告（安全漏洞、情报、通用组件和插件）有相似之处，JSRC 工作人员分析发现是由于同一处问题导致时，该情况只有最早提交者得分；

3、 威胁报告提交后在未修复前，主动公开的报告不计分。

4、 同一漏洞导致的多个利用点按照级别最高的奖励执行。

如：同个 JS 引起的多个 XSS 漏洞、同一个发布系统引起的多个页面的 XSS 漏洞、通用框架导致的整站问题等；

5、 对于高质量的威胁报告，如新颖的绕过方式，有一定技术深度但危害较小的报告，JSRC 会综合评估因素给出超出评级规则的奖励来激励报告者。

6、 对于影响业务流程的严重 bug，JSRC 会与业务方沟通确认是否要紧急修复，评分会按照实际影响等综合评定，如：该 bug 不修复，业务流程不能正常进行。

7、 对于非京东商城发布的产品和业务，如京东投资公司、子公司、合资公司等，评分上会有所减少，但会参考实际危害和影响做具体评级操作，其处理和修复不能保证在预定时间内，请报告者理解。

8、 对于京东云租户业务，例如 http://cpse.jcloud.com（虽以 jcloud.com 为域名，但从页面标题及内容可明显辨别为非京东业务），此类威胁报告会按低危处理（但涉及威胁业务范围内的敏感数据，JSRC 工作人员会与业务方沟通后酌情评级处理）。

9、 报告者在进行渗透测试时，如在线上对业务做增删改操作时，请勿直接对正常用户数据做操作，数据添加请在标题或明显字段增加【我是测试】字样，以便于 JSRC 和业务方识别数据真实性。

10、 以漏洞测试为借口,利用漏洞进行损害用户利益、影响业务正常运作、修复前公开、盗取用户数据等行为在溯源发现后将不会计分，同时京东保留采取进一步法律行动的权利；

11、京东集团及京东所属公司员工不得参与漏洞奖励计划

争议解决办法

在威胁处理过程中，如果报告者对处理流程、威胁评定、威胁评分等有异议的，请通过当前报告详细页即时通讯“一键沟通”按钮联系 JSRC 工作人员交流反馈，JSRC 将根据威胁报告者利益优先的原则进行处理。

FAQ

Q: JSRC 平台的 1 积分相当于多少人民币?

A: 截止目前的奖励标准，JSRC 平台 1 积分相当于 5 元人民币。

Q: JSRC 平台礼品兑换发放是不是有固定规则?

A: 是的，每月 2 次兑换，第一次是在每月的 1 号~15 号，第二次在 16 号~月底之间。

Q: 在其他平台提交的京东威胁也有效吗? 与其他安全团体关系如何？

A: 是的，有效。如在其他威胁或漏洞揭报平台提交，也会跟进处理，如该平台有对应规则（虚拟货币、积分等），JSRC 会参考该平台规则给予奖励。JSRC 提倡合作共赢，希望为整个互联网安全生态添砖加瓦，目前 JSRC 已与一些安全团体有了合作，未来会有更多。

Q: 威胁报告在被评分之后，积分是不是会有变动?

A: 是的，少数情况下会有变动。一种是 JSRC 工作人员在评分之后发现评分给少或者给多；另外一种是报告者反馈有异议后，工作人员综合情况做出的变动。JSRC 将尽力减少甚至避免此情况，给出一个客观且无争议的评级。

Q: 威胁报告的确认、处理周期会按照标准来执行吗?

A: 常规情况按照标准执行，但也会存在周期较长的情况，如 JSRC 做活动时、业务大促、客户端漏洞和安全情报等。同时也有可能由于报告者提供的内容不够详细导致确认延迟，请各位理解。因此请报告者尽可能提交详细步骤或 POC，加快工作人员处理速度。

Q: JSRC 有没有先“忽略” 之后偷偷修复情况?

A:绝对不会。提交的“威胁”一旦进入“忽略”状态，工作人员会在备注中说明缘由，JSRC 会根据规则操作每一个报告者提交的威胁，有依有据，客观中肯。当然，中间存在因业务变动导致“威胁”不存在的可能性。但无论如何，JSRC 都不会 “偷偷” 修复。

Q: 相比其他甚至国外 SRC，奖励是否会有更大的优势?

A: 会有，现有状况下，我们会尽最大努力为负责任的威胁报告者争取最大的奖励回馈。

Q: JSRC 究竟什么时候改版?

A: 抱歉，这个需求我们早已意识到但因种种原因导致延迟。不过放心，该需求已在最近的开发计划，让我们拭目以待。

附：扫描插件 Demo

Eg.：poc_demo.lua

```
local http= require("http")
function verify(params)
	local target = params["target"]
	local username = params["username"]
	local password = params["password"]
	local url = target .."manager/html"
	local response, error_message = http.get(url, {
		basicauth={username, password}
	})

	if error_message
	then
		result.vul = false
		result.err = error_message
	else
		if response.status_code == 200 and string.find(response.body,
"<title>/manager</title>")
		then
			result.vul = true
			result.target = target
			result.username = username
			result.password = password
		else
			result.vul = false
			result.err = tostring(response.status_code)
		end
	end
end
```