注意保管，请勿外传

XX测试公司

渗透测试报告

（初测）

XX安全公司

2025年8月

**适用性声明**

本报告仅适用于此次在XX测试公司开展渗透测试时的状况。渗透测试结束之后，因内部环境或不可预知的国内国际政治、经济、法律等社会环境的变化，可能会影响评估结论的有效性。再者，任何控制系统都存在固有的局限性，错误、违反内部控制的情况仍然有可能发生及不被发现。

**版权声明**

本文中的所有信息均为单位内部信息，务请妥善保管，未经XX测试公司作出的书面许可，不得为任何目的、以任何形式或手段（包括电子、机械、复印、录音或其他形式）对本文档的任何部分进行复制、存储、引入检索系统或者传播。

**版本变更记录**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **版本号** | **拟制／修改日期** | **拟制／修改人** | **修改记录** | **批准人** |
| V1.0 | 2025-8-8 | 张三 | 创建文档 | 王五 |

**目 录**

[1 测试概况 2](#_Toc1675725610)

[2 测试计划 3](#_Toc1819780512)

[2.1 测试对象 3](#_Toc588964610)

[2.2 测试人员 3](#_Toc976071247)

[2.3 测试阶段 3](#_Toc201868896)

[2.4 风险等级 3](#_Toc1933856459)

[2.5 测试内容 4](#_Toc160509068)

[2.6 测试流程 7](#_Toc436445244)

[2.7 风险规避 8](#_Toc1678561403)

[3 测试结果详情 10](#_Toc88829582)

[3.1 {{{{{first\_level\_heading}}}}} 10](#_Toc457650009)

[3.1.1 【{{{{{risk\_level}}}}}】{{{{{vul\_name}}}}}（{{{{{is\_fixed}}}}}） 10](#_Toc1584761356)

# 测试概况

我公司依据本次测试内容安排资深安全工程师采用科学的渗透测试手段，于2025年7月1日至2025年7月7日对XX测试公司进行了全面深入的漏洞检测。共计发现6个漏洞，包括1个高危漏洞，2个中危漏洞，3个低危漏洞，各系统漏洞情况如下：

漏洞情况统计表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **序号** | 系统名称 | 子系统名称及访问地址 | 漏洞名称 | 漏洞数量（个） | 漏洞风险等级 |
| 1 | xxxxx | http://xxx.xxx.xxx.xxx | xxxxx | 1 | 高危 |

# 测试计划

## 测试对象

此次渗透测试是根据XX测试公司要求，针对业务系统所进行的一次渗透测试。

测试对象

|  |  |  |  |
| --- | --- | --- | --- |
| **序号** | 系统名称 | 子系统名称及访问地址 | **输出的服务渠道名称（H5页面、SDK包、API接口时填写）** |
| 1 | / | xxx.xxx.xxx.xxx | H5页面 |

## 测试人员

此次项目的渗透测试小组由资深安全服务工程师主要参与，工作人员配合。

项目组成人员如下：

项目成员

|  |  |  |  |
| --- | --- | --- | --- |
| **项目角色** | **姓名** | **职务** | **项目职责** |
| 项目经理 | 王五 | 项目经理 | 项目实施计划、项目实施过程管理。 |
| 渗透工程师 | 李四 | 渗透工程师 | 渗透测试 |

## 测试阶段

根据我公司安全服务的经验，将本次渗透测试项目按照以下阶段进行实施：

测试阶段工作内容

|  |  |
| --- | --- |
| **项目阶段** | **工作内容** |
| 项目启动阶段 | 确定项目组成员、配合需求、  确定工作计划、提交实施方案、获得书面委托及授权。 |
| 风险规避阶段 | 采用调整渗透时间和策略等方式进行风险规避。 |
| 渗透测试阶段 | 对各应用系统进行渗透测试工作。 |
| 数据分析与报告阶段 | 对所有的过程数据进行分析，根据分析结果撰写报告。 |

## 风险等级

风险等级

|  |  |  |
| --- | --- | --- |
| **编号** | **风险等级** | **风险描述** |
| 1 | 高风险 | 可对目标网站造成重大损失，如信息篡改、数据损坏以及远程控制。 |
| 2 | 中风险 | 可获取目标网站敏感信息，并结合其他安全漏洞可能影响目标网站的正常运营。 |
| 3 | 低风险 | 可获取目标网站少量敏感信息，一般不会导致严重的安全事件。 |

## 测试内容

风险等级

| **测试类型** | **测试条目** | **测试描述** |
| --- | --- | --- |
| 信息收集 | 目录爬行/遍历 | 这个阶段将通过浏览、目录爬行的方式捕获/收集应用的资源。 |
| 搜索引擎侦测 | 搜索引擎，比如Google，能够用来发现公开发布的网页应用结构或者错误页面等相关问题。 |
| 应用程序入口探测 | 枚举应用入口和攻击途径是入侵发生之前的预警。这部分枚举完成后，将帮助测试人员找出在应用里面应该重点关注的领域。 |
| Web应用程序指纹探测 | 应用指纹是信息收集的第一步。获取运行网页服务器的版本，让测试人员知道哪些是已知弱点及在测试时使用何种方法恰当。 |
| 应用程序发现 | 本项测试发现以web服务器的网页应用作为目标。本项测试对于发现细节/寻找突破尤为有效，比如发现用于管理的应用脚本，或旧版本的文件/控件，在测试、开发或维护过程中产生的已不用的脚本。 |
| 分析错误代码-信息泄漏 | 在渗透性测试过程中，网页应用可能泄露原本不想被用户看见的信息。错误码等信息能让测试者了解应用程序使用的有关技术和产品。很多情况下，由于异常处理和程序代码的不合理，甚至不需要任何特殊技术或工具，都很容易触发产生错误代码的条件从而产生错误代码导致被攻击者利用。 |
| 配置管理测试 | SSL/TLS测试 | SSL和TLS是两个以加密的方式为传输的信息提供安全隧道的协议，具有保护、加密和身份认证的功能。  这些安全组件在应用中非常关键，因此确保高强度的加密算法和正确地执行非常重要。  本项测试的模块为：SSL版本、算法、密钥长度、数字证书、有效期。 |
| 数据库监听器（DB Listener）测试 | 许多数据库管理员在配置数据库服务器时，没有充分考虑到数据库侦听器组件的安全。如果没有进行安全的配置而使用手动或自动的技术进行侦听，侦听器就可能泄露敏感数据以及配置信息或正在运行的数据库实例信息。泄露的信息对测试者来说通常是有用的，他能将此应用到后续更深入的测试中去。 |
| 基础配置信息测试 | Web应用基础架构由于其内在的复杂性和关联性，一个微小的漏洞就可能对同一服务器上的另一个应用程序产生严重的威胁，甚至破坏整个架构的安全。为了解决这些问题，对配置的管理和已知安全问题进行深入审查尤为重要。 |
| 应用程序配置信息测试 | 通常在应用程序开发和配置中会产生一些没有考虑到的信息，而这些信息暂时被发布后的Web应用程序所隐藏。  这些信息可能从源代码、日志文件或Web服务器的默认错误代码中泄露。 |
| 文件扩展名处理测试 | 通过Web服务器或Web应用程序上的文件扩展名能够识别出目标应用程序使用的技术，例如扩展名JSP与ASP。文件扩展名也可能暴露与该应用程序相连接的其他系统。 |
| 旧文件、备份文件、未引用文件测试 | Web 服务器上存在多余的、可读、可下载的文件，并且用于备份的文件，是信息泄漏的一大源头。因为它们可能包含应用程序或数据库的部分源代码，安装路径以及密码等敏感信息。本项测试验证这些文件是否存在于发布的Web应用系统上。 |
| 应用程序管理接口测试 | 许多应用程序的管理接口通常使用一个公用路径，路径获取后可能面临猜测或暴力破解管理密码的风险。此项测试目的是找到管理接口，并检测是否可以利用它来获取管理员权限。 |
| HTTP请求方法与XST测试 | Web服务器可以配置为多种请求方式，如Get、Post、Put、Delete等，此项测试将鉴定Web服务器是否允许具有潜在危险性的HTTP请求方法，同时鉴定是否存在跨网站追踪攻击（XST）。 |
| 认证测试 | 证书加密通道传输安全性测试 | 本项测试试图分析用户输入Web表单中的数据，如为了登录网站而输入的登录凭据是否使用了安全的传输协议，以免受到攻击。 |
| 用户枚举测试 | 本项测试为了验证是否可能通过与应用程序的认证机制交互（提示信息），收集有效的用户。这项测试好于暴力破解，一旦获取有效的用户名后，就可针对性地进行密码攻击。 |
| 字典猜解测试 | 本项测试鉴定应用系统是否存在默认的用户账户或可猜测的用户名/密码组合（遍历测试）。 |
| 口令暴力破解测试 | 当遍历攻击失败，测试者可尝试使用暴力破解的方式进行验证。暴力破解测试可能可能碰到锁定用户或IP等限制。 |
| 验证绕过测试 | 本项测试尝试以非常规的方式企图绕过身份认证机制，使得应用程序资源失去正常的保护，从而能够在没有认证的情况下访问这些受保护的资源。 |
| 密码重置/找回漏洞测试 | 本项测试鉴定应用程序的“忘记密码”功能是否起到足够的保护，检查应用程序是否允许用户在浏览器中存储密码。 |
| 用户注销缓存漏洞测试 | 检查注销和缓存功能能否得到正确实现。 |
| 多因素认证漏洞测试 | 多因素身份验证将测试以下认证方式的安全性：  一次性密码（OTP）所生成的验证码，  USB加密设备  基于X.509证书的智能卡  通过SMS发送的随机一次性密码  只有合法用户知道的个人信息 |
| 会话管理测试 | 会话管理测试 | 本项测试分析会话管理模式和机制，鉴定发送给客户端浏览器的会话验证码的安全性，鉴定是否能够打破这一机制从而绕过用户会话。如：对Cookie实行反向工程，通过篡改Cookies来劫持会话。 |
| Cookie属性测试 | Cookies通常是恶意用户攻击合法用户的关键途径。本项测试将分析应用程序在分派Cookie时如何采取必要的防护措施，以及这些已正确配置的Cookie属性。 |
| 会话固定测试 | 本项测试鉴定当应用程序在成功验证用户后不再更新Cookie 时，能否找到会话固定漏洞并迫使用户使用攻击者已知的Cookie 。 |
| 会话变量泄漏测试 | 由于会话验证码联系了用户身份和用户会话，它所代表的是保密信息。本项测试鉴定会话验证码是否暴露在漏洞中，并试着追溯会话攻击。 |
| CSRF跨站请求伪造测试 | 跨站伪造请求指在Web应用中，迫使已通过验证的未知用户执行非法请求的方法。本项测试鉴定应用程序是否存在这种漏洞。 |
| 授权测试 | 路径遍历测试 | 本项测试鉴定是否能够找到一种方法来执行路径遍历攻击并获成功得服务器返回的信息。 |
| 授权绕过测试 | 本项测试核实如何对某个角色或特权实施授权模式以便获得保留的功能和资源。 |
| 权限提升测试 | 本项测试确认用户是否可能采用特权提升攻击的方式修改自己在应用程序内部的特权或角色。 |
| 数据验证测试 | 反射型跨站脚本攻击 | 反射型跨站脚本攻击 (XSS) 是非持久性跨站脚本攻击的另一个名称。该攻击不会使用存在漏洞的Web应用程序加载，而使用受害者载入的违规的URI 。  本项测试将确认应用程序对来自用户提交的恶意代码是否进行了存储或反射处理，对各类非法字符进行了严格过滤。 |
| 存储型跨站脚本攻击 | 存储型跨站脚本攻击（XSS）是一种最危险的跨站脚本。允许用户存储数据的Web应用程序都有可能遭受这种类型的攻击。 |
| SQL 注入测试（Oracle Mysql MsSQL Access） | SQL注入测试检测是否有可能将数据注入应用程序中，以便在后端数据库中执行用户定制的SQL查询。如果应用程序在没有合理验证数据的情况下使用用户输入创建SQL查询，那么说明该应用程序存在SQL注入漏洞。成功利用这一类别的漏洞会导致未授权用户访问或操作数据库中的数据。 |
| Code 注入测试 | 代码注入测试检测是否有可能在应用程序中注入稍后由Web服务器执行的代码。 |
| OS Commanding | 本项测试将设法通过HTTP请求在应用程序中注入OS命令。 |
| 缓冲区溢出测试（字符串格式） | 本项测试将检查不同类型的缓冲区溢出漏洞。 |
| Web服务测试 | Web服务信息收集 | 进行Web 服务测试的第一步是确定WS入口点和链接图标。 |
| XML架构测试 | XML需要有合法的格式才能正确地运作。当服务器端进行XML语句分析时，不合规格的XML将会出错。一个解析器需要在整个XML信息中按照序列的方式彻底运行，这样才能评估XML格式是否合格。  XML解析器通常占用较多的CPU资源。某些攻击通过发送大量或者不合规的XML信息来利用这个漏洞。 |
| XML内容级别测试 | 内容级别的攻击对象是Web服务及其使用的应用程序的服务器，包括Web服务器、数据库、应用程序服务器、操作系统等等。内容级别攻击向量包括：1）SQL注入/XPath注入 2）缓存溢出 3）命令注入。 |
| HTTP GET参数/REST测试 | 许多XML应用程序是通过HTTP GET查询传输参数来使用的。在HTTP GET字符串例如，超长的参数（2048字符）、SQL语句/注入（或OS注入参数）中传输恶意内容时，Web服务将会受到攻击。 |
| Naughty SOAP附件 | 本项测试将检测接收附件的Web服务的是否存在漏洞。这类危险存在于当信息附加到服务器和分配到用户的时候。 |
| 重放测试 | 重放攻击的威胁在于攻击者可以伪装成一个合法用户的身份，然后不被察觉的情况下进行一些恶意操作。本项测试将检测Web服务是否存在重放漏洞。 |

## 测试流程

渗透测试服务通过远程利用目标应用系统等安全弱点，模拟真正的黑客入侵攻击方法，以人工渗透为主，以漏洞扫描工具为辅，在保证整个渗透测试过程都在可以控制和调整的范围之内尽可能地获取目标信息系统的管理权限以及敏感信息。

渗透测试服务的主要流程如下：

（1）信息收集

信息收集是指渗透实施前尽可能多地获取目标信息系统相关信息，例如网站注册信息、共享资源、系统版本信息、已知漏洞及弱口令等等。通过对以上信息的收集，发现可利用的安全漏洞，为进一步对目标信息系统进行渗透入侵提供基础。

（2）弱点分析

对收集到的目标信息系统可能存在的可利用安全漏洞或弱点进行分析，并确定渗透方式和步骤实施渗透测试。

（3）获取权限

对目标信息系统渗透成功，获取目标信息系统普通权限。

（4）权限提升

当获取目标信息系统普通管理权限后，利用已知提权类漏洞或特殊渗透方式进行本地提权，获取目标系统远程控制权限。

## 风险规避

渗透测试过程最大的风险在于测试过程中对业务产生影响，为此在实施渗透测试中采取以下措施来减小风险：

（1）双方确认

渗透测试前，双方签署了渗透测试授权书。进行每一阶段的渗透测试前，必须获得同意和授权。对于任何渗透测试对象的变更和测试条件变更也都必须获得双方的同意并达成一致意见，方可执行。

（2）工具选择

为防止造成真正的攻击，在渗透性测试项目中，渗透测试工程师会严格选择测试工具，杜绝因工具选择不当造成的将病毒和木马植入的情况发生。

（3）时间选择

为减轻渗透性测试对用户网络和系统的影响，渗透测试全部安排在不影响正常业务运作的时间段进行。

（4）范围控制

严格保证不对授权范围之外的网络设备、主机和系统进行漏洞检测、攻击性测试，严格按照渗透测试范围内限定的应用系统进行测试。

（5）策略选择

为防止渗透性测试造成网络和系统的服务中断，渗透工程师在渗透性测试中不使用含有拒绝服务的测试策略。

（6）渗透期间保障确认

每天渗透人员测试完毕后，检查确认网站是否正常运行，如发现网站不正常运行，及时与项目联系人取得联系，方便尽快恢复网站系统运行。

# 测试结果详情

## **成都纽博弈公司**

### OA系统

#### 【高危】SQL注入

* 漏洞描述

攻击者通过在应用程序的输入字段中插入恶意SQL代码，当应用程序未正确过滤用户输入时，这些代码会被数据库执行，从而绕过认证或获取敏感数据。

* 风险等级

高危

* 漏洞危害

可能导致数据库信息泄露、数据篡改或服务器被控制

* 漏洞链接地址

请手动填写漏洞链接

* 漏洞证明

请手动填写漏洞证明

* 修复建议

1、修复2221；

2、修复2。

#### 【高危】跨站脚本

* 漏洞描述

攻击者将恶意JavaScript代码注入到网页中，当其他用户访问该页面时，脚本会在他们的浏览器中执行，可以窃取cookie或进行其他恶意操作。

* 风险等级

高危

* 漏洞危害

可能导致用户会话劫持、钓鱼攻击或恶意脚本执行

* 漏洞链接地址

请手动填写漏洞链接

* 漏洞证明

请手动填写漏洞证明

* 修复建议

1、修复1；

2、修复2。

## **成都纽博弈公司汉宝部门**

### 官网

#### 【高危】跨站请求伪造

* 漏洞描述

攻击者诱使用户在已认证的网站上执行非预期的请求，利用用户已建立的会话凭证，如强制用户更改密码或转账。

* 风险等级

高危

* 漏洞危害

可能导致用户在不知情的情况下执行非预期的操作

* 漏洞链接地址

请手动填写漏洞链接

* 漏洞证明

请手动填写漏洞证明

* 修复建议

1、修复1；

2、修复2。

#### 【中危】我的漏洞

* 漏洞描述

的撒的

撒打算大s

阿斯顿撒的撒大

* 风险等级

中危

* 漏洞危害

我的漏洞危害

* 漏洞链接地址

请手动填写漏洞链接

* 漏洞证明

请手动填写漏洞证明

* 修复建议

1、撒撒打算大贺卡上的

2、撒大的撒老师的距离2255