

系统测试技术

- 功能测试
- 信息安全测试
- 性能测试
- 可靠性测试
- 易用性测试
- 兼容性测试
-

功能测试

- 功能测试 (functionality testing) , 又称为正确性测试或一致性测试, 其目的是用以确认软件在指定条件下使用时, 软件产品提供满足明确和隐含要求的功能的能力。
 - 适用性测试, 测试软件为指定的任务和用户目标提供一组合适的功能的能力。
 - 准确性测试, 测试软件提供具有所需精度的正确或相符的结果或效果的能力。
 - 互操作测试, 测试软件与一个或更多的规定系统进行交互的能力。
 - 信息安全测试, 测试软件保护信息和数据的能力, 以使未授权的人员或系统不能阅读或修改这些信息和数据, 而不拒绝授权人员或系统对它们的访问。

对于安全性有较高需求的软件, 则须做专门的信息安全测试。
 - 功能依从性测试, 测试规约中所有的功能都应实现, 而且应是正确的。
- 采用人工测试或采用开源Selenium等测试工具进行自动化测试。

信息安全测试

□ 功能安全性验证

- 测试系统的认证、授权、鉴权和权限控制，确保用户在所授的权限内进行功能的操作和数据的存取。

□ 漏洞检测

- 基于漏洞数据库，通过漏洞扫描工具采用静态和动态手段对系统的安全脆弱性进行检测，发现漏洞。

□ 安全攻击

- 进行威胁建模，找出可以实施渗透攻击的攻击点，进行验证。

信息安全测试

常见的面向Web应用的信息安全测试的测试内容

类型	测试内容
服务器信息测试	运行账号测试；web服务器端口版本测试；HTTP方法测试
文件目录测试	目录遍历测试；文件归档测试；目录列表测试
认证测试	验证码测试；认证错误测试；找回、修改密码测试
会话管理测试	会话超时测试；会话固定测试；会话标识随机性测试
授权管理测试	横向越权测试；纵向越权测试；跨站伪造请求测试
文件上传下载测试	文件上传测试；文件下载测试
信息泄露测试	数据库账号密码测试；客户端源代码测试；异常处理测试
输入数据测试	SQL注入测试；XML注入测试；LDAP注入测试
跨站脚本攻击测试	反射型测试；存储型跨站测试；DOM型跨站测试
逻辑测试	上下文逻辑测试；算术逻辑测试
Webservice测试	Webservice接口测试；Webservice完整性、机密性测试
HTML5测试	CORS测试；Web客户端存储测试；WebWorker安全测试
FLASH安全配置测试	全局配置文件安全测试；浏览器端安全测试
其他测试	Struts2测试；Web部署管理测试；日志审计测试

示例

□ 例如，4S系统的系统测试时

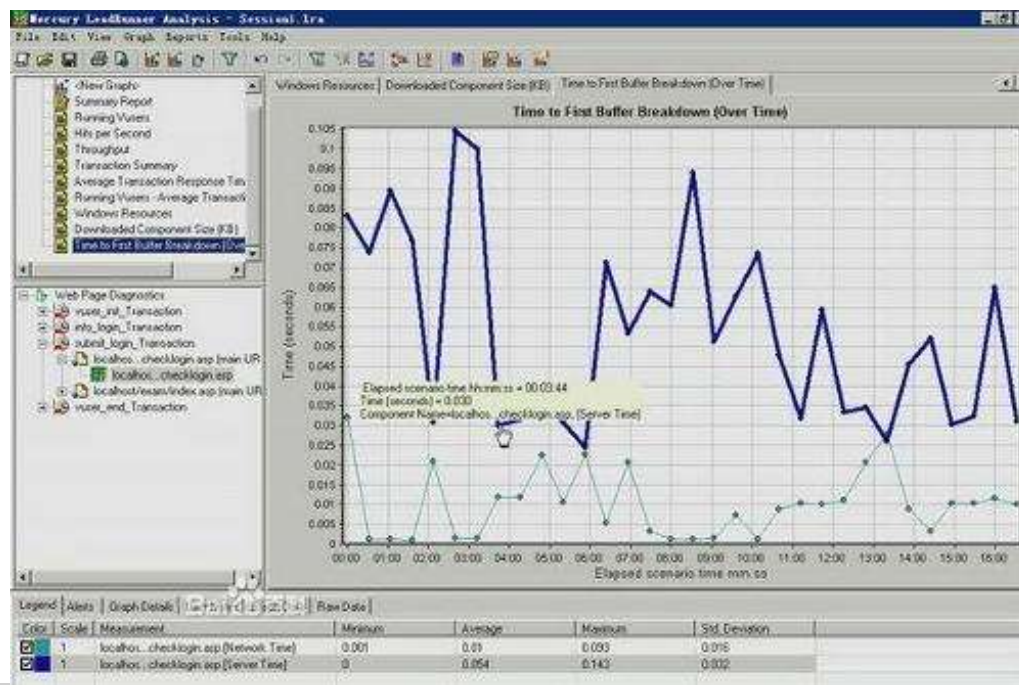
- 4S系统的系统功能测试采用人工测试方式，按其用例模型为每个用例的各个基本流和备选流分别设计了测试用例。
- 检查以上功能是否符合权限控制的要求，即所有授权的动作是否都能做，所有非授权的信息是否都无法看到。
- 测试者扮演一个试图攻击系统的角色，采用各种方式攻击系统：
 - 截获或破译4S系统的用户名和密码；借助于某种软件攻击4S系统；
 - “制服”4S系统，使得别人无法访问；故意引发系统错误，期望在系统恢复过程中侵入系统；
 - 通过浏览非保密的数据，从中找到进入4S系统的钥匙等等。

性能测试

- 性能测试（performance testing）用来测试软件在规定条件下，相对于所用资源的数量，可提供适当性能的能力。
 - 时间特性测试：测试在规定条件下，软件执行其功能时，提供适当的响应和处理时间以及吞吐率的能力。
 - 资源利用性测试：测试在规定条件下，软件执行其功能时，使用合适数量和类别的资源的能力。这些资源包括CPU、内存、网络等。
 - 性能依从性测试：测试软件遵循性能相关的规约、标准或法规的能力。
- 压力测试（stress testing），又称强度测试，是一种超常情况下的性能测试。它需要在超常数量、频率或资源的方式下执行系统，以获得系统对非正常情况下（如大数据量的输入、处理和输出，大并发数等）的承受程度。
- 自动化的性能/压力测试工具：HP的Loadrunner，开源的JMeter等。

示例

- 例如，4S系统的系统测试时，
 - 采用Loadrunner进行自动化性能测试，模拟不同数量的并发用户，在不同的数据量情况下，观察系统的响应时间和资源使用情况。



可靠性测试

- 软件可靠性测试（reliability testing）用以测试在故障发生时，软件产品维持规定的绩效级别的能力。
 - 成熟性测试，测试软件为避免由软件中故障而导致失效的能力。
 - 容错性测试，测试在软件出现故障或者违反其指定接口的情况下，软件维持规定的性能级别的能力。
 - 易恢复性测试，测试在失效发生的情况下，软件重建规定的性能级别并恢复受直接影响的数据的能力。
 - 可靠性的依从性测试，测试软件遵循与可靠性相关的规约、标准或法规的能力。
- 可靠性测试关注故障的避免、预防、容错和恢复，它通过自然方式和故障注入方式来触发和激活系统中的故障来测试。

可靠性和可用性度量

□ 可靠性使用MTBF度量

- MTBF (Mean Time Between Failure) 是平均故障间隔时间, 或称为平均无故障工作时间, 指相邻两次故障之间的平均工作时长

□ 易恢复性使用MTTR度量

- MTTR (Mean Time To Recover) 是平均故障修复时间
- 越短的MTTR意味着在规定条件和规定时间内, 按规定的程序和方法维修时, 系统保持和恢复到规定状态的能力越强

□ 可用性计算: $MTBF/(MTBF+MTTR)$

- 可用性不仅取决于可靠性, 还取决于易恢复性

广义的可靠性包括可靠性和可用性

示例

□ 例如，4S系统的系统测试时，

- 连续运行4S系统100小时，计算系统平均故障间隔时间，以及故障恢复的平均用时；
- 主动制造故障的方法来验证系统的容灾和恢复能力
 - 通过强制系统重启、拔网线，以及制造数据库读写失败和缓存读写失败等各种手段，让4S系统发生故障，然后观察系统是否还能降级运行，是否能及时地恢复运行，数据库中的数据是否因此留下了“脏”数据等。

易用性测试

- 易用性测试 (usability testing) 用以评价用户学习和使用软件（包括用户文档）的难易程度、支持用户任务的有效程度、从用户的错误中恢复的能力。
 - 易理解性测试，测试软件使用户能理解软件是否合适以及如何能将软件用于特定的任务和使用条件的能力。
 - 易学性测试，测试软件使用户能学习其应用的能力。
 - 易操作性测试，测试软件使用户能操作和控制它的能力。
 - 吸引性测试，测试软件吸引用户的能力。
 - 易用依从性测试，测试软件遵循易用相关的规约、标准、风格指南或法规的能力。
- 易用性测试可采用模拟用户的方式进行，也可通过观察用户的操作行为来执行。

示例

- 例如，4S系统的易用性测试时，
 - 测量用户所需的培训时间
 - 观察销售人员完成下订单的任务平均需要多少步骤
 - 测试各界面的风格是否一致
 - 信息显示和反馈是否准确
 - 是否提供在线的支持帮助等等。

兼容性测试

- 兼容性测试用以验证软件系统与其所处的上下文环境的兼容情况，即系统在不同环境下，其功能和非功能质量都能够符合要求。
- 面向以下三个兼容性维度
 - **系统内部兼容**：系统内部各部件之间的兼容性，包括软件和软件(与其他软件、浏览器、操作系统、数据库系统等)、软件和硬件、硬件和硬件之间的兼容性
 - **系统间兼容**：系统与其他系统存在接口互连、功能交互等情况下的配合能力
 - **系统自身兼容**：系统的新老版本间需要保证的功能、操作体验等方面的一致性，包括前向兼容和后向兼容

示例

□ 4S系统的兼容性测试如下：

- 在MS Window和Linux两种操作系统的服务器上，进行系统的首次安装、升级、完整的或自定义的安装，测试是否都能成功安装
- 检查是否能和已安装的杀毒软件等共存
- 在客户端检查是否用Chrome和Firefox浏览器在规定的分辨率下都能正常使用
- 检查是否通过人事管理系统能顺利得到员工信息
- 检查是否能成功卸载