

论 WEB 测试策略

摘要

2007 年 6 月，我参加公司对原有《基于工作流的文档数字安全保护系统》项目升级，担任了开发人员一职，基于该系统可以向外延伸多个系统，涉及到审计人员访问控制，不同用户组的权限管理，权限控制，密钥证书管理，证书验证，文件加解密，文件打包等，要求该系统必须是可靠，稳定性好，在早期的开发过程中不重视，导致后期维护困难，不得不对原有系统进行升级，加强了对软件测试的重视，根据 WEB 系统的特点，将系统测试划分为单元测试，集成测试，系统测试，确认测试，重点讨论了单元测试，功能测试和性能测试采取的措施和策略，2007 年 12 月底，经过以上措施，取得了良好的效果，由于该系统是基于 WEB 方式的与传统的软件测试有很多不同之处，为了保证 WEB 系统的测试充分性，系统在可用性上，安全性方面采取了措施。

正文

随着 Web2.0 时代的到来，信息的流转和共享越来越频繁，企业用户迫切希望能够通过文档的交换来增加信息流动效率、加强协同工作能力。但电子文档具有易复制、易修改和易传播的特征，使它在流转中随时面临被窃取、篡改、非授权访问和非授权分发的威胁，造成电子文档的价值降低甚至丧失，这直接或间接

地制约着文档流转的广泛应用。从目前的系统和文献资料看，现有的保护模型和技术并不能很好地解决流转文档的保护问题，主要是无法有效地为电子文档在整个流转过程中提供灵活的保护；其次，现有的文档保护系统常常忽略与用户当前工作流程的结合，在保护文档的同时也影响了电子文档的正常使用。

公司早期开发的《文档数字安全保护系统》系统已不适应现在的发展要求，之前的系统使用传统的加密解密方式，权限定义简单，无法满足文档流转过程中的权限变化，而且刚开始，文档和用户较少，访问人数有限，响应速度还比较及时，随着规模扩大，时间的积累，数据剧增，导致在查询统计搜索中常常出现页面超时，极大地影响了工作效率。同时早期的开发文档不完整，后期的维护困难，使得系统越来越难以维护。因此，公司决定及时的开发新版本《基于工作流的文档数字版权保护系统》，并增加了打包服务，将数字内容加密并以一定方式封装起来，许可证书生成服务，可视化工作流定义服务和客户使用服务，来适应公司发展的需要。

我作为项目负责人之一，有幸地参与了该项目测试工作，考虑到在加密方法上，JAVA 和 .NET 基本没有太大的差别，但在 WEB 服务安全性上，JAVA 明显比 .NET 落后一些，所以该系统选用了微软的 .NET 开发平台，采用了 C/S 架构设计：表示层，功能层，数据层，表示层负责处理系统与用户或管理员的交互接口，功能层负责系统处理业务逻辑所有的 COM+ 组件组成，数据层存放用户信息，用

户权限信息，密钥证书数据库等关系数据库，该系统业务功能很多，由于该系统是核心系统，基于该系统向外延伸很多系统，比如审计人员系统，新的文件格式打包等等，对系统的性能，可靠性，稳定性，安全性较高。考虑到早期系统失败有一部分因素是早期的开发对软件质量不重视，导致可维护性差，最终造成工作上的损失，因此，在该项目中，我们十分重视软件的测试作，根据系统的具体特点和应用要求，仔细地分析了业务流程和系统应用环境，确定系统的测试要求，并据此编写测试计划，测试用例和测试脚本，将系统的测试划分为单元测试，集成测试，系统测试，确认测试，重点关注了单元测试，功能测试和性能测试，下面就就此作详细介绍。

在单元测试阶段，我们采用测试先行来保证软件质量，采取开发人员自己编写的测试用例，小组人员交叉评审和测试组抽查相结合的策略，以避免测试用例的片面性，同时要求单元测试应紧接在编码编译通过之后，鼓励进行测试先行即先编写测试用例，然后用测试驱动代码实现。

单元测试采用了 Nunit 自动化测试框架，由于本系统是用。NET+C#开发，开发工具采用 VS。NET，Nunit 是当前 C#自动化单元测试的实际标准，。NET 对 Nunit 提供了很好的支持，用它在。NET 类上创建和执行自动的单元测试，这是我们初次使用 Nunit，它的 GUI 界面使用非常方便，一目了然，绿色表示通过，红色表示缺陷，黄色表示忽略，一运行就知道哪里的代码写得不严密，在开发阶

段就避免了很多的错误，发现了很多潜在的危险，减少了后期的大量测试工作，也有效地保证了单元测试的可靠性，为后续测试的顺利奠定了坚实的基础。

对于系统的功能测试，我们主要考虑了页面链接测试，表单提交测试，数据库测试页面的连接测试，采用了 xenu 工具，功能强大的检查网站死链接的软件，只要输入网址就可以检查，它分别列表活链接以及死链接，连转向链接它都分析得一清二楚，支持多线程，可以把检查结果存储成文本文件或网页文件。这基本上不会带给测试人员工作量。

对于表单提交和数据库测试，部分采取了自动测试方法，运用了 Winrunner，测试之前，依据系统需求分析，对每一个交互功能页面设计测试用例，重点关注关键页面的测试用例，如用户登录，用户录入，用户权限申请，查询，更新等页面，对这些页面的测试采取自动录制和手工编写脚本相结合的方式，测试针对提交的操作的完整性来进行，以检验提交给服务器的信息的正确性，同时检查数据库存取时数据的一致性，例如：用户密码要求长度不低于 8 位的非数字字符和数据组合，并区分大小写，测试时，我们以表格驱动的方法，在表格中设置了 8 位以下，8 位及 8 位以上以及的非数字字符，数字字符及其混合型进行测试，另外，如果表单只允许接受某些特定值或使用了默认值，还必须检验指定值和默认值的正确性，测试运行结束后，可以通过 QC 来查看报告，对测试结果进行分析。

对于系统的性能测试，使用了自动化测试方法，采用了 Loadrunner 测试工具，

从连接响应速度和负载两个方面来进行，在连接响应速度测试时，通过测试工具模拟单机访问，我们发现某些组件和页面响应速度较慢，经过分析，我们进行了相应部分的代码优化，SQL 查询优化，采用了数据库连接池技术等，系统性能得到了明显的改善，负责测试时，通过测试工具模拟多用户访问系统，检查系统所能承受的极限访问量，测试中，我们将系统各个部门的日志和监测工具全部打开，在并发用户数逐渐加大的过程中，记录系统在线处理，CPU 负载，内存使用等方面的变化情况，借助这些数据查找并分析系统的性能瓶颈，采取针对性的改善措施。

2007 年 12 月底，《基于工作流的文档数字安全保护系统》开发完毕，得到了领导和同事的一致好评，由于采用了适当的测试方法，测试策略和测试工具，取得了不错的效果，对以后的 WEB 测试起到了很好的作用，有力地保证了软件的质量，但从系统的实践中，我们体会到基于 WEB 的系统测试比传统的软件测试不同，它不但需要检查和验证是否按照设计的要求运行，而且还要测试系统的不同客户的浏览器端的显示是否合适，重要的是，还要从最终用户的角度进行安全性和可用性的测试，然后，INTERNET 和 WEB 媒体的不可预见性使测试基于 WEB 的系统变得困难，同时基于 WEB 的系统比传统的软件有更短的发布周期，测试人员和测试管理人员面临着从测试传统的 C/S 结构和框架环境到测试快速改变的 WEB 应用系统的转变。

为了保证 WEB 系统测试的充分性，除了在性能和功能上的测试，还应在以下方面进行测试：

1、可用性进行测试。比如界面风格，导航是否安排合理，WEB 系统是否需要站点地图，帮助，客户端浏览是否兼容。

2、安全性测试

1) 登录界面上，必须测试有效和无效的用户名和密码，要注意到是否大小写敏感，是否可以不登陆而直接浏览某个页面等。

2) WEB 系统是否有超时限制，也就是用户登录后在一定时间内没有点击任何页面，是否中止会话，重新登录才能正常浏览页面。

3) 为了保证 WEB 应用系统的安全性，日志文件是至关重要的，需要测试相关信息是否写进了日志文件，是否可追踪。

4) 服务器端的脚本常常构成安全漏洞，这些漏洞又常常被黑客利用，所以要测试没有经过授权就不能在服务器端放置和编辑脚本的问题。

我们从实践中体会到，目前，软件开发正在向基于 WEB 的应用发展和转变，对软件测试提出了新的挑战，测试技术还有很多领域和知识点需要我们不断地研究和实践，我们将在该领域中不断地学习，不断地探索。这样，我们的产品质量才能得到更好的保证，产品才能不断地走向成熟，才能在竞争中处于不败之地。