 前段时间，应朋友之邀，我对他们托管的三台服务器的主机进行了测试，发现了JSP网站存在的几个问题。 入侵测试第一步：扫描 扫描是入侵的第一步，它可以让你对即将入侵的目标有一个全面的了解。同时扫描还有可能发现扫描对象的漏洞，为入侵提供一个指导方向。 朋友的两台服务器为Linux，一台为Windows系统，在路由器后面还有一台Cisco PIX 525对三台主机进行保护，只允许外部用户连接不同主机的部分端口，例如80,25,110。 根据检测，Cisco PIX防火墙过滤规则设置比较严密，基本上没有多余端口允许外部用户访问。细致分析后，我发现，目标网络的主机通过地址转换来提供对外访问，内部使用192.168.*.*地址段。 先不考虑那么多，找个扫描软件来看看主机的安全情况。我找来了X-Scan，在外部对这几台主机进行了端口扫描之后，生成了一份关于端口的报表，发现其中有一个Tomcat服务器，解释的自然就是JSP文件了。小知识： Tomcat Web服务器是一款开源的适合于各种平台的免费网络服务器。eBay.com与Dell 计算机等知名网站都采用或者曾经采用Tomcat的container容器执行Servlet 与JSP。 看来，只能通过Web服务进行间接攻击。首先检查TCP 80端口的服务。我发现，新闻搜索的功能是由端口8080提供的，输入http://202.103.*.168:8080/之后，得到了一个系统管理登录页面，简单地测试了一下，输入“ test/test ”作为“ 用户名/口令 ”，似乎认证成功，但实际上并不能进入下一个页面。 专家支招：

对于扫描来说，它很容易暴露我们网站的弱势方面。应对扫描，我们可以架设一个蜜罐来误导扫描者，蜜罐可以让系统伪装成到处是漏洞，从而遮蔽真正存在的漏洞，也可以伪装成没有任何漏洞，让入侵者不知道从何入手。 入侵测试第二步：漏洞尝试 尝试JSP各种已知漏洞，这个是在扫描结果中无法获得任何有效信息指导入侵的情况下，被迫使用的方法。这种方法虽然效果不一定好，但是往往能够起到意想不到的效果，从而让入侵继续下去。 我进行了JSP大小写的测试，因为JSP对大小写是敏感的，Tomcat只会将小写的jsp后缀的文件当作是正常的JSP文件来执行，如果大写了就会引起Tomcat将index.JSP当作是一个可以下载的文件让客户下载，若干测试后，我发现这个方法并不奏效，可能管理员已经在服务器软件的网站上下载了最新的补丁。 我发现大部分的JSP应用程序在当前目录下都会有一个WEB-INF目录，这个目录通常存放的是JavaBeans编译后的class 文件，如果不给这个目录设置正常的权限，所有的class就会曝光。 而采用JAD软件对下载的class文件反编译后，原始的Java文件甚至变量名都不会改变。如果网页制作者开始把数据库的用户名密码都写在了Java代码中，反编译后，说不定还能看到数据库的重要信息。那么，怎么得到这些文件呢？ Tomcat版本的缺省“ /admin ”目录是很容易访问的。输入：http://202.103.*.168/admin/，管理员目录赫然在列。默认情况下，“ User Name ”应该是admin，“ Password ”应该是空，输入用户和密码后，并点击“ Login ”按钮，不能进入，陆续使用了几个比较常见的密码，也无济于事。 默认情况下，Tomcat打开了目录浏览功能，而一般的管理员又很容易忽视这个问题。也就是说，当要求的资源

直接映射到服务器上的一个目录时，由于在目录中缺少缺省的index.jsp等文件，Tomcat将不返回找不到资源的404错误，而是返回HTML格式的目录列表。 想到了这点后，我打开刚才用X-Scan扫描后生成的报表文件，找到“ 安全漏洞及解决方案 ”栏目，看到了几个可能会有CGI漏洞的目录。在地址栏输入其中之一，返回结果如图1所示。 