学生应知信息知识

Web服务器知识手册（五）

狄登峰 主编

目　　录

如何本机调试 CGI

WINDOWS 95/98 操作系统：

1. 安装 Perl 解释器？

Perl 是一种解释执行的语言，所以，要单机运行调试 Perl CGI ，必须先安装 Perl 解释器。

首先下载 Active Perl 5.22e(本站的工具软件栏目中有下载)，一般文件名是 APi5XXe.exe，其中 XX 是版本号。然后运行此程序，默认是安装在 C:\PERL 下，不过为了方便，请最好安装到 C:\USR 目录下，这样写 Perl 解释器的路径就可以直接用 #!/usr/bin/perl 了，可以保持单机环境和网络环境路径一致。

2. 安装 WEB 服务器？

有些网友认为安装了 Perl 解释器后就可通过双击文件运行，这是错误的，因为 Perl 是脚本文件，它只能在浏览器内调用，所以必须安装 WEB 服务器软件。现在 WINDOWS 95/98 下常用的 WEB 服务器有 omniHTTPD 2.06 ， Apache 1.3.12 ， Personal Web Server(PWS)，下面我们就以这三种 WEB 服务器为例，来看看 Perl 环境的架设：

a) omniHTTPD 2.06

安装好后，按照下面三步来修改注册表：

① 运行 RegEdit ，搜寻 : HKEY_LOCAL_MACHINE\System\Currentcontrlset\Services\W3svc\

Parameters\ScriptMap\ 键名

② 然后增加键名："".cgi"，键值："C:\USR\BIN\perl.exe %s %s" 和键名："".pl"，键值："C:\USR\BIN\perl.exe %s %s"

③ 存盘即可。

最后，把主页文件拷到 httpd/htdocs 目录，Perl 程序拷到 httpd/cgi-bin 目录即可。

b) Personal Web Server(PWS)

同 a) 一样修改注册表，然后在 PWS 中把 c:/inetpub/wwwroot/cgi-bin 目录设置为可执行即可。

这种服务器有个缺点，它不支持那些使用了 Unix 进程和函数的 Perl 程序，但也有一个好处，这样的系统同时能支持 ASP。

c) Apache 1.3.12

安装时请注意，安装路径最好装在硬盘根目录 \APACHE 下，不要装在默认的 Program Files\APACHE 下，这样设定绝对路径方便！

装好后，进入 APACHE 下的 CONF 目录，用文本编辑器来编辑 HTTPD.CONF 文件。

① 寻找 ServerName，把前面的 # 号去掉，后面改为你的域名(单机可用 localhost)。

② 寻找 #ScriptAlias /cgi-bin/ "C:/Apache/cgi-bin/"，把前面的 # 号去掉。

③ 寻找 AddHandler cgi-script .cgi ，在后面加上一个空格和 .pl。

④ 寻找 ，下面有一句 AllowOverride, 把后面的参数去掉，改为 All。

⑤ 存盘即可。

最后，把主页文件拷到 apache/htdocs 目录，Perl 程

序拷到 apache/cgi-bin 目录即可。

虽然此种 WEB 服务器最难安装，但它却是支持 Perl 功能最全最完善的，所以本人强烈推荐大家使用此服务器。

WINDOWS NT/2000 操作系统：

1. 安装 Perl 解释器？

方法同上面的 Perl 解释器的安装。

2. 修改注册表

方法同上面的步骤 2 下的 a)中的修改方法。

3. 修改 IIS WEB 服务器

NT/2000 中自带了 IIS WEB 服务器，而且 CGI-BIN 的目录一般已经自动设置好了，在 c:\inetpub\wwwroot\cgi-bin 下，如果发现设置不对的话，可以按照如下办法重新设置：

① 启动 Internet Service Manager。

② 选择 WWW 信息发布服务，双击或使用鼠标右键选择 Service Properties。

③ 在 WWW 信息发布服务属性(Publishing Service Properties)窗口中选择 Directories 子窗口。

④ 用鼠标点击 Add 按钮，打开目录属性(Directory Properties)对话框。

⑤ 在目录属性对话框的第一栏 Directory 用键盘输入需建立目录映射的目录路径(c:\inetpub\wwwroot\cgi－bin)，选中 Visual Directory，在别名(alias)中键入 cgi-bin，并选中 Access 中的 Execute 复选框，最后确定即可。

4. 特别注意事项

如果你的 NT/2000 采用了 NTFS 格式的话，请注

意设置好相应目录的读写属性，一般是在相应目录上添加 Everyone 用户，并设置其可写即可。

这种服务器有个缺点，它不支持那些使用了 Unix 进程和函数的 Perl 程序，但也有一个好处，这样的系统同时能支持 ASP。

测试系统是否安装正确：

输入以下简单程序(请注意第一行，本人假设你的 Perl 解释器是安装在 /usr 下的，如果你安装的是 /perl 下，请把第一行改为 #!/perl/bin/perl)，然后存盘为 test.cgi，放在你的 CGI-BIN 目录下。 <BR

```
#!/usr/bin/perl
print "Content-type:text/html\n\n";
print "http://skyz.yesky.net/";
exit;
#--------
```

运行 WEB 服务器，然后运行浏览器，输入 http://localhost/cgi-bin/test.cgi ，如果浏览器上出现 http://skyz.yesky.net / 则表示你的 Perl 环境已经成功架设好了。

最后，要注意一点，以上单机调试环境均不支持 Unix 下的 flock() 函数，所以如果你的程序运行出现白页的话，请把该程序的所有 flock() 函数都去掉即可。

基于 Apache 的 Web 页面访问权限控制

假设你有一些敏感的信息要放在 Intranet/Internet 上，你首先可能会想到自己开发一个用户身份认证的系

统来保护你的 Web 页面。其实 Apache 本身就自带了限制用户访问 Web 页面的机制，实现起来也不复杂。

本文介绍在 Linux+Apache 上的实现方法：

1、修改 http.conf

假设你想控制/usr/local/apache/htdocs 下各目录的不同访问权限，你可以在与之间加入一行：

AllowOverride All

意思是在/usr/local/apache/htdocs 下不同目录的访问权限由该目录下的.htaccess 文件来控制，而且不同目录的权限策略可互相覆盖。

2、编辑你想要控制的目录下的.htaccess 文件

假设你的 phpmyadmin 目录在/usr/local/apache/htdocs 下，你可以这样在 phpmyadmin 目录下创建一个.htaccess 文件，内容如下：

AuthUserFile /usr/local/apache/pass/pwdPhp

AuthType Basic

AuthName "Database Security Zone"

ErrorDocument 401 /catchErrors/err_401.html

require valid-user

该文件说明了几个问题：

(1) 用户信息存放在/usr/local/apache/pass/pwdPhp 中.

(2) 认证类型为基本型(此外还有一些其他的加密类型).

(3) ErrorDocument 所指向的 html 文件.

(4) 认证方式：用户认证(valid-user)或组认证(valid-group).

3. 生成用户密码文件

有一个用户密码生成程序:htpasswd(在/usr/local/apache/bin 下),它可以加入用户密码信息到指定的文件中,如/usr/local/apache/pass/pwdPhp. 我的用户密码文件内容如下:

admin:a0Hplbj33QjV2

guest:R0BYSO383QjVT

4. 重起 apache deamon

/usr/local/apche/bin/apachectl restart

5. 测试

用 IE 浏览受保护的页面,如 http://ip/phpmyadmin/(可能需要 refresh 几次才能凑效),这时应该出现一个身份认证窗口,你需要输入用户名(admin / guest) 和密码(******)才能访问这个页面。

构建安全的 e-commerce 服务器

一. Background

基于 Internet 的网络经济一直吸引着人们的眼球,随着门户网站的局势已定,现在又涌现出一批以“电子商务”命名的网络公司。相比之下,他们比较冷静和谨慎。在企业级应用上,他们不仅仅满足于协助中小企业上网,更多的是想提供一些电子商务的主打产品:CRM、ERP、SCM 等,或者提供从 IDC 到 ASP 一条龙服务。

但是,就我所经历的情况来看,真正能埋头做产品的公司微乎其微。一是因为投入太大,二是因为很难找到合适的市场定位。做方案和集成,无非是东拼西凑,糊弄初级客户,完全没有自己的东西。那么,目前电子

商务公司除了做一些网站建设和应用项目，还有哪些盈利点？依我所见，由于国内的电子商务环境还不成熟，没有完整的信用体制和支付手段，在这个基础上，许多电子商务活动都是很难开展的。客户的顾虑也很多，仅从安全性上考虑，比如你做 ASP（应用服务提供商），客户很难接受与其他人共享一块硬盘，把数据交给你维护更是忧心忡忡；辛辛苦苦开发出一套系统，放到网上，很轻易的被黑客窃取了源代码，让人心痛；架在一个不堪一击系统上的应用，黑客篡改了页面和数据，都很难向客户交待，影响了自己的声誉和进一步的业务合作。

这就体现出了安全的重要性。是的，安全和黑客技术是比较偏，有些搞软件开发的人甚至对此嗤之以鼻，但是我们不能否认安全在电子商务中的基石作用，不考虑安全和不懂安全的系统分析员来设计开发电子商务系统，最后注定会失败的很惨。

安全是一个很复杂的系统工程，从最初的制度策略的制定，到最后整个系统的 implement，有很多环节。本文仅仅介绍构造一个 e-commerce 服务器，来说明在 Internet 上放置一个可以安全运行的电子商务 WEB 服务器也不是那么的简单。

二．Apache

为什么要选择 Apache？中小企业比较乐于接受较低的系统报价，UNIX 的网管们也可以从技术上替我解释这个问题。是的，相比于漏洞层出不迭的 IIS 来说，Apache 在安全界享有良好的声誉，但是一个默认安装的 Apache 还是不够。

1) 操作系统

Apache 尽管发布了 Windows、Linux、BSD 家族和

其他操作系统的版本，但毫无疑问的是，UNIX 是最好的选择。首先是远程管理上的方便，同时 SSH 提供了远程管理维护的加密通道。在系统性能上，UNIX 类系统更加易于优化配置。

2) 自身的漏洞

尽管 Apache 的内核没有太大的 buffer overflows 和 exploits，但是在 1.3.19 以前的版本有一个 mod_rewrite 漏洞。建议安装最新的版本 1.3.20。

3) 外来的隐患

现在的电子商务网站内容都不是静态，而是动态生成的，所以需要额外的一些模块，如 Java（Jserv）、Perl（mod_perl）、PHP（mod_php）。这些模块给 Apache 引入了安全隐患。如 Windows 平台上的 Apache+PHP 存在目录遍历漏洞，UNIX 平台上，某些版本的 Tomcat 引擎（Java Servlets 和 JSP）也存在目录遍历、甚至泄露.jsp 源代码的漏洞。

Apache 和其它软件产品一样，多多少少存在安全问题。我们不要在嘲笑 IIS 满身窟窿同时，对 Apache 抱着 100%的放心。一般情况下，有两个因素导致软件的不安全性：技术上和配置。如果网管们都能很好的配置服务器，相比之下，软件中的一些 BUG 是很容易解决的。

三．SSL

Internet 是一个开放的系统。大部分的网络通信都是不安全的，就好比传统邮政中的明信片邮寄，恶意用户可以偷看明信片内容、篡改和伪造身份发送。

SSL，即 Secure Socket Layer，是工作在网络层与会话层之间的协议，它在 TCP/IP 和 HTTP 之间增加了一个加密层，主要是使用公开密钥体制和 X.509 数字证书技

术保护信息传输的机密性和完整性，它不能保证信息的不可抵赖性，主要适用于点对点之间的信息传输，常用于 Web Server 方式。

电子商务系统中,最常用的加密协议是 SSL 和 SET。SET 是在应用层，而 SSL 是在会话层，对工作在 HTTP 协议以上的用户而言，加密是透明的。关于 SSL 和 SET 的比较，请参考其他文章。事实上，最容易实现的方案就是采用 SSL，新推出的 TLS 也未被广泛使用。

四．Apache+SSL

好，下面将给出一些实践内容，介绍如何安装一个安全的 Apache SSL Server。首先，必须保证网络和操作系统的安全性：安装了防火墙和路由器并且配置正确，操作系统已打补丁且做了安全优化，系统日志的单独存放等等。

Apache 服务器本身不支持 SSL，我们有很多选择可以完成 Apache/SSL 的合并：（1）Apache-SSL 计划（http://www.apache-ssl.org），它集成了 Apache 服务器和 SSL；(2)第三方的 SSL 补丁,例如 Covalent Networks 的 Covalent SSL (http://www.covalent.com)；(3)mod_ssl，它是通过可动态加载的模块 mod_ssl（http://www.modssl.org）来支持 SSL；(4)基于 Apache 并集成了 SSL 能力的商业 Web 服务器,然而使用这些商业 Web 服务器主要是北美,这是因为在那里 SSL 使用的公开密钥的算法具备专利权，例如 RedHat Secure Server（http://store.redhat.com/commerce/）。

我们选择第三种方法，这样我们就使用 Apache 的最新版本。去三个站点下载以下软件包：

Apache：http://www.apache.org

OpenSSL：http://www.openssl.org

mod_ssl：http://www.modssl.org

下面是安装步骤：

A． 准备

解开 apache、openssl **和** mod_ssl **到**/usr/local/src **目录下。**

B． 编译 Openssl

切换到目录/usr/local/src/openssl-0.9.6：

（1）./Configure linux-elf threads –fPIC –prefix=/usr/local/ssl

（2）make

（3）make test

（4）make install

C． 配置 mod_ssl

进入目录/usr/local/src/mod_ssl-2.8.0-1.3.17 **执行以下命令：**

./configure --with-apache=../apache_1.3.17

D． 配置 Apache

进入目录/usr/local/src/apache_1.3.17：

1. export SSL_BASE=../openssl-0.9.6

2. ./configure \

--prefix=/usr/local/apache \

--enable-module=ssl \

--disable-rule=SSL_COMPAT \

--enable-module=rewrite \

--enable-module=auth-digest \ # use MD5 hashes for HTTP

basic authentication

--enable-module=vhost_alias \ # enable virtual hosts

--enable-module=log_referer \ # enhance logging

--disable-module=userdir \ # not used in e-commerce apps

--disable-module=autoindex \ # do not list directories

3. make

4. make certificate TYPE=dummy

5. make install

6. /src/httpd –l

现在 Apache **已经安装好了，可以通过** httpd –l **来查看安装的模块。**

下面是一些要检查的安全设置：

SSL：

在 httpd.conf **中打开** SSL

Port 80

Listen 80

Listen 443

SSLSessionCache dbm:/usr/local/apache/logs/ssl_scache

SSLSessionCacheTimeout 1200

For increased performance use "SSLMutex sem" instead of the line below

SSLMutex file:/usr/local/apache/logs/ssl_mutex

SSLLog /usr/local/apache/logs/ssl_engine_log

change the log level default from "info" to "warn"

SSLLogLevel warn

SSLOptions +OptRenegotiate

打开虚拟主机的 SSL 支持：

Within the ...

SSLEngine on

Replace with certificate file name

SSLCertificateFile /usr/local/apache/conf/ssl.

cert/

Replace with key file name

SSLKeyFile /usr/local/apache/conf/ssl.key/

SSLVerifyClient none

定制 SSL **的** LOG **格式：**

LogFormat clfa "%h %l %u %t \"%r\" %>s %b\ %{SSL_PROTOCOL}x %{SSL_CIPHER}x \"%{SSL_CLIENT_S_DN_CN}x\""

CustomLog /usr/local/apache/logs/access_log clfa

被保护的目录：

SSLCipherSuite HIGH: MEDIUM

AuthType Digest

AuthName "Beta code testing"

AuthDigestDomain /test/ http://test.my.dom/beta/

AuthDigestFile /usr/local/apache/conf/

digest_pw

Require valid-user

最后的文件检查：

1. SSL **证书和公钥不能存放在** DocumentRoot **下；**

2. SSL **证书和公钥必须被** root **所拥有，**chmod 400 *.crt**；**

3. **移去**/htdocs **和**/cgi-bin **中的所有示例文件；**

4. /htdocs **下的所有文件被** nobody **所拥有。**

如果你不怕配置麻烦，最好把 Apache **放到一个**

chroot 的环境中运行。:)

关于如何生成证书请求包和到 CA 中心去签署，请参考其他文章。目前国内也有很多 CA 中心，如中国电信电子商务安全认证中心（http://www.sinocol.com/），都可以对个人颁发证书。

五．Hardening e-commerce Server

尽管 Apache 安装和配置的都很安全，但是一台具有薄弱口令或者运行着象 wu-ftpd 那样不安全服务的 LINUX 还是很容易被攻破。一般来讲，一台 WEB 服务器仅仅需要的其他服务只有 SSH—远程管理所用。不要安装 x-windows，编译器如 gcc 等应该在系统稳定运行后删去，这样可以避免一些 script-kiddiez 的破坏。

同时，一些包过滤规则（ipfw，ipchains，iptables）应该被应用。这里我们将讨论 Linux 下的 ipchains，假定有以下需求：

1． 服务器有两块网卡

2． 外网卡仅仅允许 80 和 443 端口数据的 incoming

3． 外网卡仅仅允许>1023 端口数据的 outgoing

4． 内网卡仅仅允许 22、80、443 端口的 incoming

5． 内网卡仅仅允许>1023 端口数据的 outgoing。

一般的连接是数据库，oracle 是 1524port，SQL Server 是 1443，SSH 可以加上-P 选项来指定大于 1023 的用户端口。

6． 内网卡允许 ICMP 响应

命令如下（eth0 外网卡，eth1 内网卡）：

```
ipchains -A in-eth0 -p tcp --dport 80 -j ACCEPT
ipchains -A in-eth0 -p tcp --dport 443 -j ACCEPT
ipchains -A in-eth0 -p udp --dport 53 -j ACCEPT
```

ipchains -A in-eth0 -j DENY

ipchains -A out-eth0 -p tcp --dport 1024:65535 -j ACCEPT

ipchains -A out-eth0 -p udp --dport 53 -j ACCEPT

ipchains -A out-eth0 -j DENY

ipchains -A in-eth1 -p tcp --dport 22 -j ACCEPT

ipchains -A in-eth1 -p tcp --dport 80 -j ACCEPT

ipchains -A in-eth1 -p tcp --dport 443 -j ACCEPT

ipchains -A in-eth1 -p udp --dport 53 -j ACCEPT

ipchains -A in-eth1 -p icmp -j ACCEPT

ipchains -A in-eth1 -j DENY

ipchains -A out-eth1 -p tcp --dport 22 -j ACCEPT

ipchains -A out-eth1 -p tcp --dport 1024:65535 -j ACCEPT

ipchains -A out-eth1 -p udp --dport 53 -j ACCEPT

ipchains -A out-eth1 -p icmp -j ACCEPT

ipchains -A out-eth1 -j DENY

剩下的工作就是重新编译系统内核，禁用不需要的模块，可以使某些 rootkits **失效。**

最后，检查 WEB SERVER **上运行着的程序的安全性，有没有缓冲区溢出等安全问题。**

六．参考资料

Improving Apache, by GARY BAHADUR & MIKE SHEMA

SSL: Theory and Practice, Zeus Technology

LASG, i.e. Linux Administrators Security Guide

一个 IP 建多个 Web 站点--主机头名法

由于各种原因,我们有时候需要在一个 IP 地址上建立多个 web 站点,在 IIS5 中,我们可能通过简单的设置达到这个目标.

在 IIS 中,每个 Web 站点都具有唯一的、由三个部分组成的标识，用来接收和响应请求:

(1) IP 地址

(2)端口号

(3)主机头名。

在 IIS 中,在一个 IP 地址上建立多个独立的 web 站点,通常有两种方法,本文以以例子的形式介绍主机头法,使用这种方法可以建立起专业的虚拟主机.

环境:假设沧海公司(呵呵...)用一台 win2000 服务器提供虚拟主机服务,地址是 192.168.1.10.在这台服务器已经安装了 Internet 服务即 IIS5.

现在公司要求网络管理员在服务器上使用一个 IP 为 ABCD 四个公司建立独立的网站,每个网站拥有自己独立的域名.四家网站域名分别为:www.a.com,www.b.com,www.c.com 和 www.d.com.

通过使用主机头，站点只需一个 IP 地址即可维护多个站点。客户可以使用不同的域名访问各自的站点,根本感觉不到这些站点在同一主机上.

具体操作如下:

1.在 win2000 服务器为四家公司建立文件夹,做为 WEB 站点主目录.如下: