

注意，对 ErrorLog 和 TransferLog 命令说明的应该注意的安全问题也适用于 PidFile 命令。

在 httpd.conf 文件中需要配置的最后一命令是服务器的 Internet 主机名：

ServerName [host name]

通常，需要输入诸如 www.yourcompany.com 这样的主机名。然而，确保在这里输入的主机名有正确的域名服务器记录将它指向你的服务器机器。下面讨论的配置文件是 srm.conf 文件。

3.1.2 srm.conf

srm.conf 是资源配置文件。用它来告诉服务器你想在 Web 站点上提供什么资源以及在哪里和如何提供它们。列表 3-2 显示了缺省的 srm.conf 文件。

列表 3-2 由 srm.conf-dist 复制的缺省 srm.conf

```
# With this document, you define the name space that users see
of your http
# server. This file also defines server settings that affect
how requests are
# serviced, and how results should be formatted.

# See the tutorial at http://www.apache.org/for
# more information.

# Originally by Rob McCool; Adapted for Apache

# DocumentRoot: The directory out of which you will serve your
# documents. By default, all requests are taken from this
directory, but
# symbolic links and aliases may be used to point to other
locations.

DocumentRoot /usr/local/etc/httpd/htdocs

# UserDir: The name of the directory which is appended onto a
user's home
# directory if a ~user request is recieved.

UserDir public_html

# DirectoryIndex: Name of the file or files to use as a pre-
written HTML
# directory index. Separate multiple entries with spaces.

DirectoryIndex index.html
```

```
# FancyIndexing is whether you want fancy directory indexing or
standard
```

```
FancyIndexing on
```

```
# AddIcon tells the server which icon to show for different
files or filename
# extensions
```

```
AddIconByEncoding (CMP, /icons/compressed.gif) x-compress
x-gzip
```

```
AddIconByType (TXT, /icons/text.gif) text/*
AddIconByType (IMG, /icons/image2.gif) image/*
AddIconByType (SND, /icons/sound2.gif) audio/*
AddIconByType (VID, /icons/movie.gif) video/*
```

```
AddIcon /icons/binary.gif .bin .exe
AddIcon /icons/binhex.gif .hqx
AddIcon /icons/tar.gif .tar
AddIcon /icons/world2.gif .wrl .wrl .gz .vrm .iv
AddIcon /icons/compressed.gif .Z .z .tgz .gz .zip
AddIcon /icons/a.gif .ps .ai .eps
AddIcon /icons/layout.gif .html .shtml .htm .pdf
AddIcon /icons/text.gif .txt
AddIcon /icons/c.gif .c
AddIcon /icons/p.gif .pl .py
AddIcon /icons/f.gif .for
AddIcon /icons/dvi.gif .dvi
AddIcon /icons/uuencoded.gif .uu
AddIcon /icons/script.gif .conf .sh .shar .csh .ksh .tcl
AddIcon /icons/tex.gif .tex
AddIcon /icons/bomb.gif core
```

```
AddIcon /icons/back.gif ..
AddIcon /icons/hand.right.gif README
AddIcon /icons/folder.gif ^^DIRECTORY^^
AddIcon /icons/blank.gif ^^BLANKICON^^
```

```
# DefaultIcon is which icon to show for files which do not have
an icon
# explicitly set.
```

```
DefaultIcon /icons/unknown.gif
```

```
# AddDescription allows you to place a short description after
```



```
a file in
# server-generated indexes.
# Format: AddDescription "description" filename

# ReadmeName is the name of the README file the server will
look for by
# default. Format: ReadmeName name
#
# The server will first look for name.html, include it if
found, and it will
# then look for name an include it as plaintext if found.
#
# HeaderName is the name of a file which should be prepended to
# directory indexes.

ReadmeName README
HeaderName HEADER

# IndexIgnore is a set of filenames which directory indexing
should ignore
# Format: IndexIgnore name1 name2...
IndexIgnore */.*? *~ *# */HEADER* */README* */RCS

# AccessFileName: The name of the file to look for in each
directory
# for access control information.

AccessFileName .htaccess

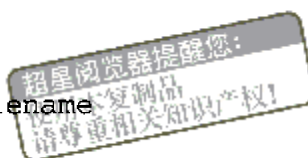
# DefaultType is the default MIME type for document which the
server
# cannot find the type of from filename extensions.

DefaultType text/plain

# AddEncoding allows you to have certain browsers (Mosaic/X
2.1+) uncompress
# information on the fly. Note: Not all browsers support this.

AddEncoding x-compress Z
AddEncoding x-gzip gz

# AddLanguage allows you to specify the language of a document.
You can
# then use content negotiation to give a browser a file in a
language
```



```
# it can understand. Note that the suffix does not have to be
the same
# as the language keyword --those with documents in Polish
(whose
# net-stand language code is pl) may wish to use
"AddLanguage pl .po"
# to avoid the ambiguity with the common suffix for perl
scripts.
```

```
AddLanguage en .en
AddLanguage fr .fr
AddLanguage de .de
AddLanguage da .da
AddLanguage el .el
AddLanguage it .it
```

```
# LanguagePriority allows you to give precedence to some
languages
# in case of a tie during content negotiation.
# Just list the languages in decreasing order of preference.
```

```
LanguagePriority en fr de
```

```
# Redirect allows you to tell clients about documents which
used to exist in
# your server's namespace, but do not anymore. This allows you
to tell the
# clients where to look for the relocated document.
# Format: Redirect fakenam url
```

```
# Aliases: Add here as many aliases as you need (with no
limit). The format is
# Alias fakenam realname
```

```
# Note that if you include a trailing / on fakenam then the
server will
# require it to be present in the URL. So "/icons" isn't
aliased in this
# example.
```

```
# Alias /icons/ /usr/local/etc/httpd/icons/
```

```
# ScriptAlias: This controls which directories contain server
scripts.
# Format: ScriptAlias fakenam realname
```

```
# ScriptAlias /cgi-bin/ /usr/local/etc/httpd/cgi-bin/

# If you want to use server side includes, or CGI outside
# ScriptAliased directories, uncomment the following lines.

# AddType allows you to tweak mime.types without actually
# editing it. or to
# make certain files to be certain types.
# Format: AddType type/subtype ext1

# AddHandler allows you to map certain file extensions to
# "handlers",
# actions unrelated to filetype. These can be either built into
# the server
# or added with the Action command (see below)
# Format: AddHandler action-name ext1

# To use CGI scripts:
# AddHandler cgi-script .cgi

# To use server-parsed HTML files
# AddType text/html .shtml
# AddHandler server-parsed .shtml

# Uncomment the following line to enable Apache's send-asis
# HTTP file
# feature
# AddHandler send-as-is asis

# If you wish to use server-parsed imagemap files, use
# AddHandler imap-file map

# To enable type maps, you might want to use
# AddHandler type-map var

# Action lets you define media types that will execute a script
# whenever
# a matching file is called. This eliminates the need for
# repeated URL
# pathnames for oft-used CGI file processors.
# Format: Action media/type /cgi-script/location
# Format: Action handler-name /cgi-script/location

# MetaDir: specifies the name of the directory in which Apache
# can find
# meta information files. These files contain additional HTTP
```

```

headers
# to include when sending the document

# MetaDir .web

# MetaSuffix: specifies the file name suffix for the file
containing the
# meta information.

# MetaSuffix .meta

# Customizable error response (Apache style)
# these come in three flavors
#
# 1) plain text
# ErrorDocument 500 "The server made a boo boo."
# n.b. the (") marks it as text, it does not get output
#
# 2) local redirects
# ErrorDocument 404 /missing.html
# to redirect to local url /missing.html
# ErrorDocument 404 /cgi-bin/missing_handler.pl
# n.b. can redirect to a script or a document using server-
side-includes.
#
# 3) external redirects
# ErrorDocument 02
http://some.other_server.com/subscription_info.html
#

```

在 `srm.conf` 中使用的配置命令将在下面各节中进行讨论。这些命令用来为主 Web 站点创建目录配置。

3.1.2.1 Web 目录配置

像所有其他 Web 服务器一样, Apache 需要知道将要保存 Web 页面的顶层目录路径。在典型的情况下, 此目录被称为文档根目录。Apache 提供名为 `DocumentRoot` 的命令, 可以用它来指定顶层 Web 目录的路径。

可使用下面的语法, 此命令告诉服务器将所提供的目录作为所有文档的根目录:

```
DocumentRoot [fully qualified path to your Web document
directory]
```

对读者来说, 这是要做出的非常重要的决定。例如, 如果按照下面所示来设置该命令:

```
DocumentRoot /
```

那么, 系统中所有文件都可以被 Web 服务器访问。当然, 通过提供正确的文件权限设置, 可以保护自己的文件, 但是, 将文档根目录设置为系统的实际根目录肯定会带来严

重的安全风险。相反，你需要将文档根目录指向文件系统的一个具体子目录。缺省设置如下所示：

```
DocumentRoot /usr/local/etc/httpd/htdocs
```

然而，更好的选择可能是为你的机构创建一个 Web 目录结构。对于多用户、多域的系统建议使用图 3-3 表示的 Web 目录结构。

正如在图中所看到的，选择创建一个名为 /www 的分区，并且，在它下面为系统中建立的每个 Web 站点都提供相应子目录。例如，/www/www.mycompany.com/ 站点有三个子目录：public、stage 和 development。而这三个子目录中的每一个都还有两个子目录：htdocs 和 cgi-bin。htdocs 是文档根目录，而 cgi-bin 则被用于 CGI 脚本。因此，www.mycompany.com Web 站点的 DocumentRoot 设置是：

```
DocumentRoot /www/www.mycompany.com/public/htdocs
```

此目录结构的优点是它在一个分区 (/www) 下面保存所有的 Web 文档和应用程序。这样就易于备份操作。而且，当指定网络中另一台机器提供 Web 服务的情况下，通过网络文件系统 (NFS)，可以将该分区装配到不同系统上。

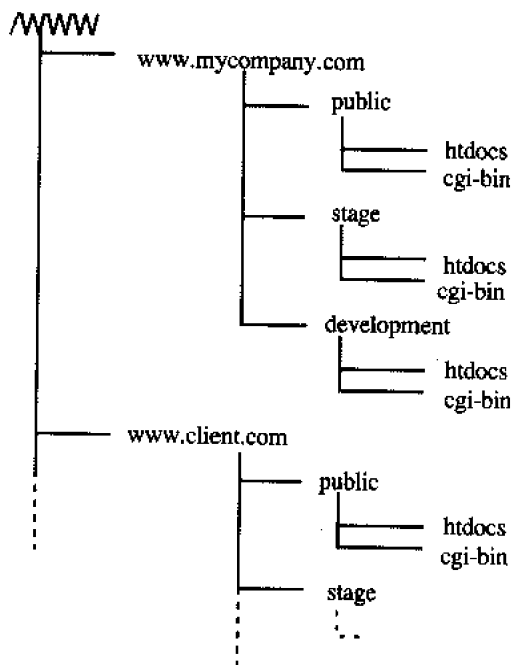



图 3-3 建议的 Web 目录结构



将在第 18 章中了解更多关于经过很好设计的 Web 目录结构。

注意，虽然文档根目录指向特定的目录，但这并不表示 Web 服务器不能访问在文档树之外的各个目录。通过使用符号链接（有正确的文件的权限设置）或使用别名（将在下一章讨论别名），可以很容易地使它做到这一点。从机构和安全的角度看，不推

存使用大量符号链接或别名来访问在文档树之外的文件和目录。不管怎么说，有时在文档树外面保存一定类型的信息是必要的，虽然你需要保持服务器可以经常访问这一目录的内容。

 **Tip** 如果必须增加到文档树之外的其他目录的符号链接，确保在备份文件的时候，指导你的备份程序正确备份符号链接。

3.1.2.2 cgi-bin 目录配置

如果计划运行 CGI 脚本，将需要将这些脚本存储到 DocumentRoot 指定的目录（或它的子目录）以外的一个目录。由于安全的原因，你需要这样做。将脚本保存在文档树中，可能会使某些人能够看到你的脚本，并在其中找到弱点。这就是为什么在图 3-3 中的目录结构中包含名为 cgi-bin 的脚本目录的原因，而该目录位于每个站点的文档树的外面。

要将此目录保存在文档树外面，可以使用 **ScriptAlias** 命令来为该目录创建一个别名，如下所示：

```
ScriptAlias [/alias/] [fully qualified path for script
directory]
```

别名只是任意指定用来存储脚本、小程序以及其他内容的实际目录的名称。CGI 脚本目录的标准别名是 **cgi-bin**；然而，可以用自己喜欢的任何名称来自由地称呼它。缺省设置是：

```
ScriptAlias /cgi-bin/ /usr/local/etc/httpd/cgi-bin/
```

对于在图 3-3 中的 **www.mycompany.com** 站点，它的别名如下所示：

```
ScriptAlias /cgi-bin/ /www/www.mycompany.com/public/cgi-
bin/
```

因此，当要引用在 URL 中的脚本或在文档树中的 HTML 文件时，你应该使用别名，而不是实际文件的位置。例如：

```
http://www.mycompany.com/cgi-gin/finger.cgi
```

将请求在 **/www.mycompany.com/public/cgi-bin/** 目录中的 **finger.cgi** 脚本。考察此命令的另一种方法是：

```
ScriptAlias fakename realname
```

需要讨论的下一条命令是 **UserDir** 命令，它有如下语法：

```
UserDir [directory name]
```

用此命令来告诉 Apache，哪一个目录用作系统中用户的 DocumentRoot。只有在系统中拥有多个用户并允许每个用户有自己的 Web 目录时，此命令才有用。缺省的设置是：

```
UserDir public_html
```

它的含义是，如果将 Web 服务器的名称设置为 **www.yourcompany.com**，并且拥有两个用户（**joe** 和 **jenny**），那么它们的个人 Web 站点 URL 将如表 3-1 中所显示的那样。

表 3-1 用户的个人 Web 目录

URL	Web 目录
http://www.yourcompany.com/~joe	~joe/public_html
http://www.yourcompany.com/~jenny	~jenny/public_html

注意，在 UNIX 系统中，~被扩展为用户的主目录。由 UserDir 命令指定的目录存在于每个用户的主目录中，而 Apache 对这一目录必须有读和执行的权限，以读取在 public_html 目录中的文件和子目录。在 UNIX 系统中，可使用如下命令做到这一点：

```
chown -R <user>.<Apache server's group name>
~<user>/<directory assigned in UserDir>
chmod -R 770 ~<user>/<directory assigned in UserDir>
```

例如，如果用户名是 joe，而 Apache 组被称为 httpd，并且在 UserDir 命令中将用户文档根目录指定为 public_html，那么上述命令应该像下面这样：

```
chown -R joe.httpd ~joe/public_html
chmod -R 2770 ~joe/public_html
```

第一个 chown 命令将 ~joe/public_html 目录（以及在它里面的全部文件和子目录）的所有者变为 joe.httpd。换句话说，它将在 public_html 目录中的全部文件和目录的全部所有权都给予用户 joe 和组 httpd。下一个 chmod 命令将访问权限设置为 2770——换句话说，只有用户(joe)和组(httpd)拥有在 public_html 中的全部读、写和执行的权限及在 public_html 下面的全部文件和子目录。它还确保当在 public_html 中创建新的文件或子目录时，新创建的文件设置了组 ID。这就允许 Web 服务器在没有用户介入的情况下也能访问此新文件。

Tip



如果使用脚本（如在 Linux 系统中的 /usr/sbin/adduser 脚本）来创建用户帐号，你可能想要在此脚本中包含 Web 站点创建过程。只需要添加一个 mkdir 命令来创建一个缺省的 public_html 目录（如果那就是赋给 UserDir 命令的值），以创建 Web 目录。添加 chmod 和 chown 命令，来给予 Web 服务器用户读和执行在此公共目录下的文件和目录的权限。

需要设置的最后一条命令是 DirectoryIndex 命令，它有以下语法：

```
DirectoryIndex [filename1, filename2, filename3, ...]
```

此命令告诉 Apache 服务器应该将哪个文件作为被请求目录的索引。例如，当请求一个 URL，如 www.yourcompany.com/时，Apache 服务器认为此请求将访问 Web 站点的/（文档根）目录。如果 DocumentRoot 命令被设置为：

```
DocumentRoot /www/www.yourcompany.com/public/htdocs
```

那么，Apache 服务器将寻找一个名称为 www/www.yourcompany.com/public/htdocs/index.html 的文件。如果它找到该文件，Apache 通过将该文件内容返回给发出请求的 Web 浏览器来服务该请求。然而，如果 DirectoryIndex 被赋值为 welcome.html，而不是缺省的 index.html，则 Web 服务器将寻找 www/www.yourcompany.com/public/htdocs/welcome.

html。如果该文件不存在，Apache 通过创建动态 HTML 页面来返回目录列表。图 3-4 显示这种情况，在目录中不存在 index.html，服务器为发出请求的浏览器生成了目录列表。

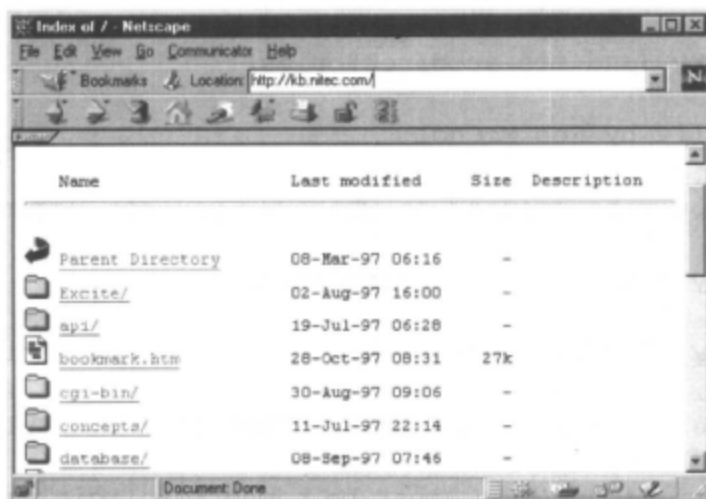


图 3-4 当不存在 index.html 时的动态目录列表

可以在 DirectoryIndex 命令中指定多个索引文件名。例如：

```
DirectoryIndex index.html index.htm welcome.htm
```

就告诉 Web 服务器，它应该检查这三个文件中任一个是否存在，如果找到其中的任一个，就将该文件返回给发出请求的 Web 客户。

注意，列出多个文件作为索引可能会引起两个问题。首先，服务器将不得不对每个目录请求检查大量文件是否存在，这有可能使它比平常慢一些。其次，将多个文件作为索引从机构的角度来看会使站点管理困难一些。然而，如果 Web 站点内容的开发人员使用不同的系统来创建文件，可将 index.html 和 index.htm 都作为索引文件，这样做可能是实际的解决方案。例如，一台 Windows 3.x 机器不能创建扩展名多于三个字符的文件名，因此，一个工作在这样一台机器上的用户可能需要用手工更改他（或她）在 Web 服务器上所有的 index.htm 文件。使用推荐的索引文件名就消除了这种烦恼。

3.1.3 access.conf

需要修改的最后一个配置文件是 access.conf。access.conf 文件被用来设置 Web 站点上诸如文件、目录和脚本项目的访问权限。列表 3-3 显示了缺省的 access.conf 文件。

列表 3-3 由 access.conf-dist 复制的缺省 access.conf

```
# access.conf: Global access configuration
# Online docs at http://www.apache.org/

# This file defines server settings which affect which types of
# services
# are allowed, and in what circumstances.
```