

Apache httpd.conf ファイルの設定項目 (/usr/local/apache/conf/httpd.conf)

注) 以下の設定ファイルは第 1 文字が # の場合はコメントである。

```
##
## httpd.conf -- Apache HTTP server configuration file
##

#
# Based upon the NCSA server configuration files originally by Rob McCool.
#
# This is the main Apache server configuration file.  It contains the
# configuration directives that give the server its instructions.
# See <URL:http://www.apache.org/docs/> for detailed information about
# the directives.
#
# Do NOT simply read the instructions in here without understanding
# what they do.  They're here only as hints or reminders.  If you are unsure
# consult the online docs.  You have been warned.
#
# After this file is processed, the server will look for and process
# /usr/local/apache/conf/srm.conf and then /usr/local/apache/conf/access.conf
# unless you have overridden these with ResourceConfig and/or
# AccessConfig directives here.
#
# The configuration directives are grouped into three basic sections:
# 1. Directives that control the operation of the Apache server process as a
#    whole (the 'global environment').
# 2. Directives that define the parameters of the 'main' or 'default' server,
#    which responds to requests that aren't handled by a virtual host.
#    These directives also provide default values for the settings
#    of all virtual hosts.
# 3. Settings for virtual hosts, which allow Web requests to be sent to
#    different IP addresses or hostnames and have them handled by the
#    same Apache server process.
#
# Configuration and logfile names: If the filenames you specify for many
# of the server's control files begin with "/" (or "drive:/" for Win32), the
# server will use that explicit path.  If the filenames do *not* begin
# with "/", the value of ServerRoot is prepended -- so "logs/foo.log"
# with ServerRoot set to "/usr/local/apache" will be interpreted by the
# server as "/usr/local/apache/logs/foo.log".
#

### Section 1: Global Environment
#
# The directives in this section affect the overall operation of Apache,
# such as the number of concurrent requests it can handle or where it
# can find its configuration files.
#

#
# ServerType is either inetd, or standalone.  Inetd mode is only supported on
# Unix platforms.
#
ServerType standalone

#
# ServerRoot: The top of the directory tree under which the server's
# configuration, error, and log files are kept.
#
# NOTE!  If you intend to place this on an NFS (or otherwise network)
# mounted filesystem then please read the LockFile documentation
# (available at <URL:http://www.apache.org/docs/mod/core.html#lockfile>);
# you will save yourself a lot of trouble.
#
# Do NOT add a slash at the end of the directory path.
#
ServerRoot "/usr/local/apache"

#
# The LockFile directive sets the path to the lockfile used when Apache
```

```

# is compiled with either USE_FCNTL_SERIALIZED_ACCEPT or
# USE_FLOCK_SERIALIZED_ACCEPT. This directive should normally be left at
# its default value. The main reason for changing it is if the logs
# directory is NFS mounted, since the lockfile MUST BE STORED ON A LOCAL
# DISK. The PID of the main server process is automatically appended to
# the filename.
#
#LockFile /usr/local/apache/logs/httpd.lock

#
# PidFile: The file in which the server should record its process
# identification number when it starts.
#
PidFile /usr/local/apache/logs/httpd.pid

#
# ScoreBoardFile: File used to store internal server process information.
# Not all architectures require this. But if yours does (you'll know because
# this file will be created when you run Apache) then you *must* ensure that
# no two invocations of Apache share the same scoreboard file.
#
ScoreBoardFile /usr/local/apache/logs/httpd.scoreboard

#
# In the standard configuration, the server will process this file,
# srm.conf, and access.conf in that order. The latter two files are
# now distributed empty, as it is recommended that all directives
# be kept in a single file for simplicity. The commented-out values
# below are the built-in defaults. You can have the server ignore
# these files altogether by using "/dev/null" (for Unix) or
# "nul" (for Win32) for the arguments to the directives.
#
#ResourceConfig /usr/local/apache/conf/srm.conf
#AccessConfig conf/access.conf

#
# Timeout: The number of seconds before receives and sends time out.
#
Timeout 300

#
# KeepAlive: Whether or not to allow persistent connections (more than
# one request per connection). Set to "Off" to deactivate.
#
KeepAlive On

#
# MaxKeepAliveRequests: The maximum number of requests to allow
# during a persistent connection. Set to 0 to allow an unlimited amount.
# We recommend you leave this number high, for maximum performance.
#
MaxKeepAliveRequests 100

#
# KeepAliveTimeout: Number of seconds to wait for the next request from the
# same client on the same connection.
#
KeepAliveTimeout 15

#
# Server-pool size regulation. Rather than making you guess how many
# server processes you need, Apache dynamically adapts to the load it
# sees --- that is, it tries to maintain enough server processes to
# handle the current load, plus a few spare servers to handle transient
# load spikes (e.g., multiple simultaneous requests from a single
# Netscape browser).
#
# It does this by periodically checking how many servers are waiting
# for a request. If there are fewer than MinSpareServers, it creates
# a new spare. If there are more than MaxSpareServers, some of the
# spares die off. The default values are probably OK for most sites.

```

```

#
MinSpareServers 5
MaxSpareServers 10

#
# Number of servers to start initially --- should be a reasonable ballpark
# figure.
#
StartServers 5

#
# Limit on total number of servers running, i.e., limit on the number
# of clients who can simultaneously connect --- if this limit is ever
# reached, clients will be LOCKED OUT, so it should NOT BE SET TOO LOW.
# It is intended mainly as a brake to keep a runaway server from taking
# the system with it as it spirals down...
#
MaxClients 150

#
# MaxRequestsPerChild: the number of requests each child process is
# allowed to process before the child dies.  The child will exit so
# as to avoid problems after prolonged use when Apache (and maybe the
# libraries it uses) leak memory or other resources.  On most systems, this
# isn't really needed, but a few (such as Solaris) do have notable leaks
# in the libraries.
#
MaxRequestsPerChild 30

#
# Listen: Allows you to bind Apache to specific IP addresses and/or
# ports, in addition to the default. See also the <VirtualHost>
# directive.
#
#Listen 3000
#Listen 12.34.56.78:80

#
# BindAddress: You can support virtual hosts with this option. This directive
# is used to tell the server which IP address to listen to. It can either
# contain "*", an IP address, or a fully qualified Internet domain name.
# See also the <VirtualHost> and Listen directives.
#
#BindAddress *

#
# Dynamic Shared Object (DSO) Support
#
# To be able to use the functionality of a module which was built as a DSO you
# have to place corresponding 'LoadModule' lines at this location so the
# directives contained in it are actually available _before_ they are used.
# Please read the file README.DSO in the Apache 1.3 distribution for more
# details about the DSO mechanism and run 'httpd -l' for the list of already
# built-in (statically linked and thus always available) modules in your httpd
# binary.
#
# Note: The order is which modules are loaded is important.  Don't change
# the order below without expert advice.
#
# Example:
# LoadModule foo_module libexec/mod_foo.so
LoadModule ssl_module libexec/libssl.so
LoadModule jserv_module libexec/mod_jserv.so

#
# ExtendedStatus controls whether Apache will generate "full" status
# information (ExtendedStatus On) or just basic information (ExtendedStatus
# Off) when the "server-status" handler is called. The default is Off.
#
#ExtendedStatus On

```

DSO としてモジュールを組み込む

```

### Section 2: 'Main' server configuration
#
# The directives in this section set up the values used by the 'main'
# server, which responds to any requests that aren't handled by a
# <VirtualHost> definition. These values also provide defaults for
# any <VirtualHost> containers you may define later in the file.
#
# All of these directives may appear inside <VirtualHost> containers,
# in which case these default settings will be overridden for the
# virtual host being defined.
#

#
# If your ServerType directive (set earlier in the 'Global Environment'
# section) is set to "inetd", the next few directives don't have any
# effect since their settings are defined by the inetd configuration.
# Skip ahead to the ServerAdmin directive.
#

#
# Port: The port to which the standalone server listens. For
# ports < 1023, you will need httpd to be run as root initially.
#
Port 80

#
# If you wish httpd to run as a different user or group, you must run
# httpd as root initially and it will switch.
#
# User/Group: The name (or #number) of the user/group to run httpd as.
# . On SCO (ODT 3) use "User nouser" and "Group nogroup".
# . On HP-UX you may not be able to use shared memory as nobody, and the
#   suggested workaround is to create a user www and use that user.
# NOTE that some kernels refuse to setgid(Group) or semctl(IPC_SET)
# when the value of (unsigned)Group is above 60000;
# don't use Group nobody on these systems!
#
User webuser
Group webuser

#
# ServerAdmin: Your address, where problems with the server should be
# e-mailed. This address appears on some server-generated pages, such
# as error documents.
#
ServerAdmin root@mhi173.ctc.ne.jp

#
# ServerName allows you to set a host name which is sent back to clients for
# your server if it's different than the one the program would get (i.e., use
# "www" instead of the host's real name).
#
# Note: You cannot just invent host names and hope they work. The name you
# define here must be a valid DNS name for your host. If you don't understand
# this, ask your network administrator.
# If your host doesn't have a registered DNS name, enter its IP address here.
# You will have to access it by its address (e.g., http://123.45.67.89/)
# anyway, and this will make redirections work in a sensible way.
#
#ServerName mhi173.ctc.ne.jp

#
# 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/apache/htdocs"

#
# Each directory to which Apache has access, can be configured with respect

```

Web サーバの実行者

あらかじめ Linux 上で作成して
おく必要あり

サーバ admin のメールアドレス

サーバの名前指定しない場合は
サーバの IP アドレスからホスト
名を参照する。

```

# to which services and features are allowed and/or disabled in that
# directory (and its subdirectories).
#
# First, we configure the "default" to be a very restrictive set of
# permissions.
#
<Directory />
    Options FollowSymLinks
    AllowOverride None
</Directory>

#
# Note that from this point forward you must specifically allow
# particular features to be enabled - so if something's not working as
# you might expect, make sure that you have specifically enabled it
# below.
#

#
# This should be changed to whatever you set DocumentRoot to.
#
<Directory "/usr/local/apache/htdocs">

#
# This may also be "None", "All", or any combination of "Indexes",
# "Includes", "FollowSymLinks", "ExecCGI", or "MultiViews".
#
# Note that "MultiViews" must be named *explicitly* --- "Options All"
# doesn't give it to you.
#
    Options Indexes FollowSymLinks

#
# This controls which options the .htaccess files in directories can
# override. Can also be "All", or any combination of "Options", "FileInfo",
# "AuthConfig", and "Limit"
#
    AllowOverride None

#
# Controls who can get stuff from this server.
#
    Order allow,deny
    Allow from all
</Directory>

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

#
# Control access to UserDir directories.  The following is an example
# for a site where these directories are restricted to read-only.
#
#<Directory /*/public_html>
#     AllowOverride FileInfo AuthConfig Limit
#     Options MultiViews Indexes SymLinksIfOwnerMatch IncludesNoExec
#     <Limit GET POST OPTIONS PROPFIND>
#         Order allow,deny
#         Allow from all
#     </Limit>
#     <Limit PUT DELETE PATCH PROPPATCH MKCOL COPY MOVE LOCK UNLOCK>
#         Order deny,allow
#         Deny from all
#     </Limit>
#</Directory>

#
# DirectoryIndex: Name of the file or files to use as a pre-written HTML

```

```

# directory index.  Separate multiple entries with spaces.
#
DirectoryIndex index.html

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

#
# The following lines prevent .htaccess files from being viewed by
# Web clients.  Since .htaccess files often contain authorization
# information, access is disallowed for security reasons.  Comment
# these lines out if you want Web visitors to see the contents of
# .htaccess files.  If you change the AccessFileName directive above,
# be sure to make the corresponding changes here.
#
<Files .htaccess>
    Order allow,deny
    Deny from all
</Files>

#
# CacheNegotiatedDocs: By default, Apache sends "Pragma: no-cache" with each
# document that was negotiated on the basis of content. This asks proxy
# servers not to cache the document. Uncommenting the following line disables
# this behavior, and proxies will be allowed to cache the documents.
#
#CacheNegotiatedDocs

#
# UseCanonicalName: (new for 1.3) With this setting turned on, whenever
# Apache needs to construct a self-referencing URL (a URL that refers back
# to the server the response is coming from) it will use ServerName and
# Port to form a "canonical" name. With this setting off, Apache will
# use the hostname:port that the client supplied, when possible. This
# also affects SERVER_NAME and SERVER_PORT in CGI scripts.
#
UseCanonicalName On

#
# TypesConfig describes where the mime.types file (or equivalent) is
# to be found.
#
TypesConfig /usr/local/apache/conf/mime.types

#
# DefaultType is the default MIME type the server will use for a document
# if it cannot otherwise determine one, such as from filename extensions.
# If your server contains mostly text or HTML documents, "text/plain" is
# a good value. If most of your content is binary, such as applications
# or images, you may want to use "application/octet-stream" instead to
# keep browsers from trying to display binary files as though they are
# text.
#
DefaultType text/plain

#
# The mod_mime_magic module allows the server to use various hints from the
# contents of the file itself to determine its type. The MIMEMagicFile
# directive tells the module where the hint definitions are located.
# mod_mime_magic is not part of the default server (you have to add
# it yourself with a LoadModule [see the DSO paragraph in the 'Global
# Environment' section], or recompile the server and include mod_mime_magic
# as part of the configuration), so it's enclosed in an <IfModule> container.
# This means that the MIMEMagicFile directive will only be processed if the
# module is part of the server.
#
<IfModule mod_mime_magic.c>
    MIMEMagicFile /usr/local/apache/conf/magic

```

```
</IfModule>
```

```
#
# HostnameLookups: Log the names of clients or just their IP addresses
# e.g., www.apache.org (on) or 204.62.129.132 (off).
# The default is off because it'd be overall better for the net if people
# had to knowingly turn this feature on, since enabling it means that
# each client request will result in AT LEAST one lookup request to the
# nameserver.
```

```
#
HostnameLookups Off
```

```
#
# ErrorLog: The location of the error log file.
# If you do not specify an ErrorLog directive within a <VirtualHost>
# container, error messages relating to that virtual host will be
# logged here. If you *do* define an error logfile for a <VirtualHost>
# container, that host's errors will be logged there and not here.
```

```
#
#ErrorLog /usr/local/apache/logs/error_log
```

```
#
# LogLevel: Control the number of messages logged to the error_log.
# Possible values include: debug, info, notice, warn, error, crit,
# alert, emerg.
```

```
#
LogLevel warn
```

```
#
# The following directives define some format nicknames for use with
# a CustomLog directive (see below).
```

```
#
LogFormat "%h %l %u %t %>%r%" %>s %b %>{"Referer"}i%" %>{"User-Agent"}i%" combined
LogFormat "%h %l %u %t %>%r%" %>s %b" common
LogFormat "%{Referer}i -> %U" referer
LogFormat "%{User-agent}i" agent
```

```
#
# The location and format of the access logfile (Common Logfile Format).
# If you do not define any access logfiles within a <VirtualHost>
# container, they will be logged here. Contrariwise, if you *do*
# define per-<VirtualHost> access logfiles, transactions will be
# logged therein and *not* in this file.
```

```
#
#CustomLog /usr/local/apache/logs/access_log common
```

```
#
# If you would like to have agent and referer logfiles, uncomment the
# following directives.
```

```
#
#CustomLog /usr/local/apache/logs/referer_log referer
#CustomLog /usr/local/apache/logs/agent_log agent
```

```
#
# If you prefer a single logfile with access, agent, and referer information
# (Combined Logfile Format) you can use the following directive.
```

```
#
#CustomLog /usr/local/apache/logs/access_log combined
```

```
#
# Optionally add a line containing the server version and virtual host
# name to server-generated pages (error documents, FTP directory listings,
# mod_status and mod_info output etc., but not CGI generated documents).
# Set to "Email" to also include a mailto: link to the ServerAdmin.
```

```
# Set to one of: On | Off | Email
```

```
#
ServerSignature On
```

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

```

#
# Note that if you include a trailing / on fakename then the server will
# require it to be present in the URL.  So "/icons" isn't aliased in this
# example, only "/icons/"..
#
Alias /icons/ "/usr/local/apache/icons/"

<Directory "/usr/local/apache/icons">
    Options Indexes MultiViews
    AllowOverride None
    Order allow,deny
    Allow from all
</Directory>

#
# ScriptAlias: This controls which directories contain server scripts.
# ScriptAliases are essentially the same as Aliases, except that
# documents in the realname directory are treated as applications and
# run by the server when requested rather than as documents sent to the client.
# The same rules about trailing "/" apply to ScriptAlias directives as to
# Alias.
#
#ScriptAlias /cgi-bin/ "/usr/local/apache/cgi-bin/"

#
# "/usr/local/apache/cgi-bin" should be changed to whatever your ScriptAliased
# CGI directory exists, if you have that configured.
#
#<Directory "/usr/local/apache/cgi-bin">
#     AllowOverride None
#     Options None
#     Order allow,deny
#     Allow from all
#</Directory>

#
# 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 old-URI new-URL
#

#
# Directives controlling the display of server-generated directory listings.
#

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

#
# AddIcon* directives tell the server which icon to show for different
# files or filename extensions.  These are only displayed for
# FancyIndexed directories.
#
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 .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. These are only displayed for FancyIndexed
# directories.
# Format: AddDescription "description" filename
#
#AddDescription "GZIP compressed document" .gz
#AddDescription "tar archive" .tar
#AddDescription "GZIP compressed tar archive" .tgz

#
# ReadmeName is the name of the README file the server will look for by
# default, and append to directory listings.
#
# HeaderName is the name of a file which should be prepended to
# directory indexes.
#
# The server will first look for name.html and include it if found.
# If name.html doesn't exist, the server will then look for name.txt
# and include it as plaintext if found.
#
ReadmeName README
HeaderName HEADER

#
# IndexIgnore is a set of filenames which directory indexing should ignore
# and not include in the listing. Shell-style wildcarding is permitted.
#
IndexIgnore .?[*] *~ *# HEADER* README* RCS CVS *,v *,t

#
# AddEncoding allows you to have certain browsers (Mosaic/X 2.1+) uncompress
# information on the fly. Note: Not all browsers support this.
# Despite the name similarity, the following Add* directives have nothing
# to do with the FancyIndexing customization directives above.
#
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-standard 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

#
# AddType allows you to tweak mime.types without actually editing it, or to
# make certain files to be certain types.
#
# For example, the PHP3 module (not part of the Apache distribution - see
# http://www.php.net) will typically use:
#
#AddType application/x-httpd-php3 .php3
#AddType application/x-httpd-php3-source .phps

#
# 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)
#
# If you want to use server side includes, or CGI outside
# ScriptAliased directories, uncomment the following lines.
#
# 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.: You can redirect to a script or a document using server-side-includes.
#
# 3) external redirects
#ErrorDocument 402 http://some.other_server.com/subscription_info.html
# N.B.: Many of the environment variables associated with the original
# request will *not* be available to such a script.

#
# The following directives modify normal HTTP response behavior.
# The first directive disables keepalive for Netscape 2.x and browsers that
# spoof it. There are known problems with these browser implementations.
# The second directive is for Microsoft Internet Explorer 4.0b2
# which has a broken HTTP/1.1 implementation and does not properly
# support keepalive when it is used on 301 or 302 (redirect) responses.
#
BrowserMatch "Mozilla/2" nokeepalive
BrowserMatch "MSIE 4.0b2;" nokeepalive downgrade-1.0 force-response-1.0

#
# The following directive disables HTTP/1.1 responses to browsers which
# are in violation of the HTTP/1.0 spec by not being able to grok a
# basic 1.1 response.
#
BrowserMatch "RealPlayer 4.0" force-response-1.0
BrowserMatch "Java/1.0" force-response-1.0
BrowserMatch "JDK/1.0" force-response-1.0

#
# Allow server status reports, with the URL of http://servername/server-status
# Change the ".your_domain.com" to match your domain to enable.
#
#<Location /server-status>
#   SetHandler server-status
#   Order deny,allow
#   Deny from all
#   Allow from .your_domain.com
#</Location>

#
# Allow remote server configuration reports, with the URL of
# http://servername/server-info (requires that mod_info.c be loaded).
# Change the ".your_domain.com" to match your domain to enable.
#
#<Location /server-info>
#   SetHandler server-info
#   Order deny,allow
#   Deny from all
#   Allow from .your_domain.com
#</Location>

#
# There have been reports of people trying to abuse an old bug from pre-1.1
# days. This bug involved a CGI script distributed as a part of Apache.
# By uncommenting these lines you can redirect these attacks to a logging
# script on phf.apache.org. Or, you can record them yourself, using the script

```

```

# support/phf_abuse_log.cgi.
#
#<Location /cgi-bin/phf*>
#   Deny from all
#   ErrorDocument 403 http://phf.apache.org/phf_abuse_log.cgi
#</Location>

#
# Proxy Server directives. Uncomment the following lines to
# enable the proxy server:
#
#<IfModule mod_proxy.c>
#ProxyRequests On
#
#<Directory proxy:*>
#   Order deny,allow
#   Deny from all
#   Allow from .your_domain.com
#</Directory>

#
# Enable/disable the handling of HTTP/1.1 "Via:" headers.
# ("Full" adds the server version; "Block" removes all outgoing Via: headers)
# Set to one of: Off | On | Full | Block
#
#ProxyVia On

#
# To enable the cache as well, edit and uncomment the following lines:
# (no cacheing without CacheRoot)
#
#CacheRoot "/usr/local/apache/proxy"
#CacheSize 5
#CacheGcInterval 4
#CacheMaxExpire 24
#CacheLastModifiedFactor 0.1
#CacheDefaultExpire 1
#NoCache a_domain.com another_domain.edu joes.garage_sale.com

#</IfModule>
# End of proxy directives.

### Section 3: Virtual Hosts
#
# VirtualHost: If you want to maintain multiple domains/hostnames on your
# machine you can setup VirtualHost containers for them.
# Please see the documentation at <URL:http://www.apache.org/docs/vhosts/>
# for further details before you try to setup virtual hosts.
# You may use the command line option '-S' to verify your virtual host
# configuration.

#
# If you want to use name-based virtual hosts you need to define at
# least one IP address (and port number) for them.
#
#NameVirtualHost 12.34.56.78:80
#NameVirtualHost 12.34.56.78

#
# VirtualHost example:
# Almost any Apache directive may go into a VirtualHost container.
#
#<VirtualHost ip.address.of.host.some_domain.com>
#   ServerAdmin webmaster@host.some_domain.com
#   DocumentRoot /www/docs/host.some_domain.com
#   ServerName host.some_domain.com
#   ErrorLog logs/host.some_domain.com-error_log
#   CustomLog logs/host.some_domain.com-access_log common
#</VirtualHost>

#<VirtualHost _default_*>

```

#</VirtualHost>

SSLPassPhraseDialog exec:/usr/local/apache/bin/pp-filter

Listen 80

```
DocumentRoot /home/wg73/share/htdocs
#   ServerAdmin webmaster@host.some_domain.com
#   ServerName host.some_domain.com
```

ScriptAlias /cgi-bin/ "/home/wg73/share/cgi-bin/"

SSLEngine off

```
SSLMutex file:/usr/local/apache/var/run/ssl_mutex
SSLSessionCacheTimeout 300
SSLVerifyClient none
SSLVerifyDepth 10
CustomLog /home/wg73/share/logs/access_log common
CustomLog /home/wg73/share/logs/referer_log referer
CustomLog /home/wg73/share/logs/agent_log agent
ErrorLog /home/wg73/share/logs/error_log
```

```
#from Apache-JServ httpd.conf
ApJServmount /servlets /share
ApJServMountCopy on
```

Listen 443

```
<VirtualHost _default_:443>
DocumentRoot /home/wg73/secure/htdocs
#   ServerAdmin webmaster@host.some_domain.com
#   ServerName host.some_domain.com
ScriptAlias /cgi-bin/ "/home/wg73/secure/cgi-bin/"
```

SSLEngine on

```
SSLCertificateKeyFile /usr/local/apache/etc/ssl/keys/httpd.key
SSLCertificateFile /usr/local/apache/etc/ssl/certs/httpd.pem
CustomLog /home/wg73/secure/logs/ssl_log "%t %{version}c %{cipher}c %{clientcert}c"
CustomLog /home/wg73/secure/logs/ssl_access_log common
CustomLog /home/wg73/secure/logs/ssl_referer_log referer
CustomLog /home/wg73/secure/logs/ssl_agent_log agent
ErrorLog /home/wg73/secure/logs/error_log
```

```
#from Apache-JServ httpd.conf
ApJServmount /servlets /secure
ApJServMountCopy on
```

#</VirtualHost>

Listen 27163

```
<VirtualHost _default_:27163>
DocumentRoot /home/wg73/system/htdocs
#   ServerAdmin webmaster@host.some_domain.com
#   ServerName host.some_domain.com
ScriptAlias /cgi-bin/ "/home/wg73/system/cgi-bin/"
```

SSLEngine off

```
CustomLog /home/wg73/system/logs/system_access_log common
CustomLog /home/wg73/system/logs/system_referer_log referer
CustomLog /home/wg73/system/logs/system_agent_log agent
ErrorLog /home/wg73/system/logs/system_error_log
```

```
#from Apache-JServ httpd.conf
ApJServmount /servlets /system
ApJServMountCopy on
```

パスフレーズの自動入力

ポート 80 の設定

SSL off

ログの設定

Apache-Jserv の設定

ポート 443 の設定

SSL on

認証キーの設定

ログの設定

Apache-Jserv の設定

ポート 27163 の設定
(システム管理者用)

SSL off

ログの設定

Apache-Jserv の設定

```
<Location /servlets>
```

```
    AllowOverride None
    Options None
    Satisfy any
    Order deny,allow
    Deny from all
```

```
    AuthType Basic
    AuthName "System Manage"
    AuthUserFile /usr/local/apache/auth/authuser_system
    AuthGroupFile /dev/null
    require valid-user
```

```
</Location>
```

```
</VirtualHost>
```

```
<Directory "/home/wg73/share/cgi-bin">
```

```
    AllowOverride None
    Options None
    Order allow,deny
    Allow from all
```

```
</Directory>
```

```
<Directory "/home/wg73/secure/cgi-bin">
```

```
    AllowOverride None
    Options None
    Order allow,deny
    Allow from all
```

```
</Directory>
```

```
<Directory "/home/wg73/system/">
```

```
    AllowOverride None
    Options None
    Satisfy any
    Order deny,allow
    Deny from all
```

```
    AuthType Basic
    AuthName "System Manage"
    AuthUserFile /usr/local/apache/auth/authuser_system
    AuthGroupFile /dev/null
    require valid-user
```

```
</Directory>
```

```
<Directory /home/wg73/secure>
```

```
SSLRequireSSL
```

```
</Directory>
```

```
# Include the configuration files needed for mod_jserv
include /usr/local/java/ApacheJServ-1.0/conf/httpd.conf
```

Servlets にベーシック認証を設定する

CGI 用のディレクトリのアクセス権限を指定する。

システム管理者用のディレクトリにベーシック認証を設定する。

SSL 用ディレクトリに SSL を設定する。

Apache-Jserv の httpd.conf を取込む

Apache-Jserv の httpd.conf の設定項目 (/usr/local/Apache-JServ1.0/conf/httpd.conf)

注) 以下の設定ファイルは第 1 文字が # の場合はコメントである。

```
#####
#                               Apache JServ Configuration File                               #
#####

# Note: this file should be appended to or included in httpd.conf

# Tell Apache on win32 to load the Apache JServ communication module
#LoadModule jserv_module modules/ApacheModuleJServ.dll

<IfModule mod_jserv.c>

# Whether Apache must start Apache JServ or not (On=Manual Off=Autostart)
# Syntax: ApJServManual [on/off]
# Default: "Off"
ApJServManual off

# Properties filename for Apache JServ in Automatic Mode.
# In manual mode this directive is ignored
# Syntax: ApJServProperties [filename]
# Default: "./conf/jserv.properties"
#ApJServProperties ./conf/jserv.properties
ApJServProperties /usr/local/java/ApacheJServ-1.0/conf/jserv.properties

# Log file for this module operation relative to Apache root directory.
# Syntax: ApJServLogFile [filename]
# Default: "./logs/mod_jserv.log"
# Note: when set to "DISABLED", the log will be redirected to Apache error log
#ApJServLogFile ./logs/mod_jserv.log
ApJServLogFile /usr/local/apache/logs/mod_jserv.log

# Log Level for this module
# Syntax: ApJServLogLevel [debug | info | notice | warn | error | crit | alert | emerg]
# Default: info (unless compiled w/ JSERV_DEBUG, in which case it's debug)
ApJServLogLevel notice

# Protocol used by this host to connect to Apache JServ
# (see documentation for more details on available protocols)
# Syntax: ApJServDefaultProtocol [name]
# Default: "ajpv11"
ApJServDefaultProtocol ajpv11

# Default host on which Apache JServ is running
# Syntax: ApJServDefaultHost [hostname]
# Default: "localhost"
#ApJServDefaultHost java.apache.org

# Default port that Apache JServ is listening to
# Syntax: ApJServDefaultPort [number]
# Default: protocol-dependant (for ajpv11 protocol this is "8007")
ApJServDefaultPort 8007

# Passes parameter and value to specified protocol.
# Syntax: ApJServProtocolParameter [name] [parameter] [value]
# Default: NONE
# Note: Currently no protocols handle this. Introduced for future protocols.

# Apache JServ secret key file relative to Apache root directory.
# Syntax: ApJServSecretKey [filename]
# Default: "./conf/jserv.secret.key"
# Warning: if authentication is DISABLED, everyone on this machine (not just
# this module) may connect to your servlet engine and execute servlet
# bypassing web server restrictions. See the documentation for more information
ApJServSecretKey DISABLED

# Mount point for Servlet zones
# (see documentation for more information on servlet zones)
# Syntax: ApJServMount [name] [jserv-url]
```

propertis ファイルのパス

ログファイルのパス

```
# Default: NONE
# Note: [name] is the name of the Apache URI path to mount jserv-url on
#       [jserv-url] is something like "protocol://host:port/zone"
#       If protocol, host or port are not specified, the values from
#       "ApJServDefaultProtocol", "ApJServDefaultHost" or "ApJServDefaultPort"
#       will be used.
#       If zone is not specified, the zone name will be the first subdirectory of
#       the called servlet.
# Example: "ApJServMount /servlets /myServlets"
# if user requests "http://host/servlets/TestServlet"
# the servlet "TestServlet" in zone "myServlets" on default host
# thru default protocol on default port will be requested
# Example: "ApJServMount /servlets ajpv11://localhost:8007"
# if user requests "http://host/servlets/myServlets/TestServlet"
# the servlet "TestServlet" in zone "myServlets" will be requested
# Example: "ApJServMount /servlets ajpv11://jserv.mydomain.com:15643/myServlets"
# if user requests "http://host/servlets/TestServlet" the servlet
# "TestServlet" in zone "myServlets" on host "jserv.mydomain.com" using
# "ajpv11" protocol on port "15643" will be executed
#ApJServMount /servlets /root
```

Apache の httpd.conf で設定
するため、本行をコメントに
する。

```
# Whether <VirtualHost> inherits base host mount points or not
# Syntax: ApJServMountCopy [on/off]
# Default: "On"
# Note: This directive is meaningful only when virtual hosts are being used
#ApJServMountCopy on
```

Apache の httpd.conf で設定
するため、本行をコメントに
する。

```
# Executes a servlet passing filename with proper extension in PATH_TRANSLATED
# property of servlet request.
# Syntax: ApJServAction [extension] [servlet-uri]
# Defaults: NONE
# Notes: This is used for external tools.
#ApJServAction .jsp /servlets/nl.nmg.jsp.JSPServlet
#ApJServAction .gsp /servlets/com.bitmechanics.gsp.GspServlet
#ApJServAction .jhtml /servlets/org.apache.servlet.ssi.SSI
#ApJServAction .xml /servlets/org.apache.cocoon.Cocoon
```

```
# Enable the Apache JServ status handler with the URL of
# "http://servername/jserv/" (note the trailing slash!)
# Change the "deny" directive to restrict access to this status page.
<Location /jserv/>
    SetHandler jserv-status
```

```
    order deny,allow
    deny from all
    allow from localhost
</Location>
```

```
##### W A R N I N G #####
# Remember to disable or otherwise protect the execution of the Apache JServ #
# Status Handler (see right above) on a production environment since this may #
# give untrusted users the ability to obtain restricted information on your #
# servlets and their initialization arguments such as JDBC passwords and #
# other important information. The Apache JServ Status Handler should be #
# accessible only by system administrators. #
#####
```

```
</IfModule>
```


Jserv-properties の設定項目 (/usr/local/java/Apache-JServ1.0/conf/jserv.properties)

注) 以下の設定ファイルは第 1 文字が # の場合はコメントである。

```
#####
#                               Apache JServ Configuration File                               #
#####

##### W A R N I N G #####
# Unlike normal Java properties, JServ configurations have some important
# extentions:
#
# 1) commas are used as token separators
# 2) multiple definitions of the same key are concatenated in a
# comma-separated list.
#####

#
# Execution parameters
#####

# The Java Virtual Machine interpreter.
# Syntax: wrapper.bin=[filename]
# Note: specify a full path if the interpreter is not visible in your path.
wrapper.bin=/usr/local/java/jdk117_v3/bin/java

# Arguments passed to Java interpreter (optional)
# Syntax: wrapper.bin.parameters=[string]
# Default: NONE

# Apache JServ entry point class (should not be changed)
# Syntax: wrapper.class=[classname]
# Default: "org.apache.jserv.JServ"

# Arguments passed to main class after the properties filename (not used)
# Syntax: wrapper.class.parameters=[string]
# Default: NONE
# Note: currently not used

# PATH environment value passed to the JVM
# Syntax: wrapper.path=[path]
# Default: "/bin:/usr/bin:/usr/local/bin" for Unix systems
# "c:\¥(windows-dir);\¥(windows-system-dir)" for Win32 systems
# Notes: if more than one line is supplied these will be concatenated using
# ":" or ";" (depending wether Unix or Win32) characters
# Under Win32 (windows-dir) and (windows-system-dir) will be
# automatically evaluated to match your system requirements

# CLASSPATH environment value passed to the JVM
# Syntax: wrapper.classpath=[path]
# Default: NONE (Sun's JDK/JRE already have a default classpath)
# Note: if more than one line is supplied these will be concatenated using
# ":" or ";" (depending wether Unix or Win32) characters. JVM must be
# able to find JSDK and JServ classes and any utility classes used by
# your servlets.
# Note: the classes you want to be automatically reloaded upon modification
# MUST NOT be in this classpath or the classpath of the shell
# you start the Apache from.
wrapper.classpath=/usr/local/java/Apache-JServ/lib/ApacheJServ.jar
wrapper.classpath=/usr/local/java/JSDK2.0/lib/jsdk.jar
wrapper.classpath=/usr/local/java/xml-tr2/xml.jar
wrapper.classpath=/usr/local/java/cos/lib/cos.jar

# An environment name with value passed to the JVM
# Syntax: wrapper.env=[name]=[value]
# Default: NONE on Unix Systems
# SystemDrive and SystemRoot with appropriate values on Win32 systems

# An environment name with value copied from caller to Java Virtual Machine
# Syntax: wrapper.env.copy=[name]
# Default: NONE
```

java インタプリタのパス

CLASSPATH のパス
必要なライブラリを切る

```
# Copies all environment from caller to Java Virtual Machine
# Syntax: wrapper.env.copyall=[true,false]
# Default: false
```

```
# Protocol used for signal handling
# Syntax: wrapper.protocol=[name]
# Default: ajp11
```

```
#
# General parameters
#####
```

```
# Set the port Apache JServ listens to.
# Syntax: port=[1024,65535] (int)
# Default: 8007
port=8007
```

```
#
# Servlet Zones parameters
#####
```

```
# List of servlet zones Apache JServ manages
# Syntax: zones=[servlet zone],[servlet zone]... (Comma separated list of String)
# Default: NONE
zones=share, secure, system
```

zone のリスト
(Apache の httpd.conf で設定)

```
# Configuration file for each servlet zone (one per servlet zone)
# Syntax: [servlet zone name as on the zones list].properties=[full path to configFile] (String)
# Default: NONE
# Note: if the file could not be opened, try using absolute paths.
share.properties=/usr/local/java/ApacheJServ-1.0/conf/share_zone.properties
secure.properties=/usr/local/java/ApacheJServ-1.0/conf/secure_zone.properties
system.properties=/usr/local/java/ApacheJServ-1.0/conf/system_zone.properties
```

zone の properties のパス

```
#
# Security parameters
#####
```

```
# Enable/disable the execution of org.apache.jserv.JServ as a servlet.
# This is disabled by default because it may give informations that should
# be restricted.
# Note that the execution of Apache JServ as a servlet is filtered by the web
# server modules by default so that both sides should be enabled to let this
# service work.
# This service is useful for installation and configuration since it gives
# feedback about the exact configurations Apache JServ is using, but it should
# be disabled when both installation and configuration processes are done.
# Syntax: security.selfservlet=[true,false] (boolean)
# Default: false
# WARNING: disable this in a production environment since may give reserved
# information to untrusted users.
security.selfservlet=true
```

```
# Set the maximum number of socket connections Apache JServ may handle
# simultaneously. Make sure your operating environment has enough file
# descriptors to allow this number.
# Syntax: security.maxConnections=(int)>1
# Default: 50
security.maxConnections=50
```

```
# List of IP addresses allowed to connect to Apache JServ. This is a first
# security filtering to reject possibly unsecure connections and avoid the
# overhead of connection authentication.
# Syntax: security.allowedAddresses=[IP address],[IP Address]... (Comma separated list of IP addresses)
# Default: 127.0.0.1
#security.allowedAddresses=127.0.0.1
```

```
# Enable/disable connection authentication.
# NOTE: unauthenticated connections are a little faster since authentication
# handshake is not performed at connection creation.
# WARNING: authentication is disabled by default because we believe that
```

```

# connection restriction from all IP addresses but localhost reduces your
# time to get Apache JServ to run. If you allow other addresses to connect and
# you don't trust it, you should enable authentication to prevent untrusted
# execution of your servlets. Beware: if authentication is disabled and the
# IP address is allowed, everyone on that machine can execute your servlets!
# Syntax: security.authentication=[true,false] (boolean)
# Default: true
security.authentication=false

# Authentication secret key.
# The secret key is passed as a file that must be kept secure and must
# be exactly the same of those used by clients to authenticate themselves.
# Syntax: security.secretKey=[secret key path and filename] (String)
# Default: NONE
# Note: if the file could not be opened, try using absolute paths.
#security.secretKey=./conf/jserv.secret.key

# Length of the randomly generated challenge string (in bytes) used to
# authenticate connections. 5 is the lowest possible choice to force a safe
# level of security and reduce connection creation overhead.
# Syntax: security.challengeSize=(int)>5
# Default: 5
#security.challengeSize=5

#
# Logging parameters
#####

# Enable/disable Apache JServ logging.
# WARNING: logging is a very expensive operation in terms of performance. You
# should reduced the generated log to a minumum or even disable it if fast
# execution is an issue. Note that if all log channels (see below) are
# enabled, the log may become really big since each servlet request may
# generate many Kb of log. Some log channels are mainly for debugging
# purposes and should be disabled in a production environment.
# Syntax: log=[true,false] (boolean)
# Default: true
log=true

# Set the name of the log file. To avoid possible confusion about
# the location of this file, an absolute pathname is recommended.
# Syntax: log.file=[log path and filename] (String)
# Default: NONE
# Note: if the file could not be opened, try using absolute paths.
log.file=/usr/local/apache/logs/jserv.log

# Enable the timestamp before the log message
# Syntax: log.timestamp=[true,false] (boolean)
# Default: true
log.timestamp=true

# Use the given string as a data format
# (see java.text.SimpleDateFormat for the list of options)
# Syntax: log.dateFormat=(String)
# Default: [dd/MM/yyyy HH:mm:ss:SSS zz]
log.dateFormat=[dd/MM/yyyy HH:mm:ss:SSS zz]

# Since all the messages logged are processed by a thread running with
# minimum priority, it's of vital importance that this thread gets a chance
# to run once in a while. If it doesn't, the log queue overflow occurs,
# usually resulting in the OutOfMemoryError.
#
# To prevent this from happening, two parameters are used: log.queue.maxage
# and log.queue.maxsize. The former defines the maximum time for the logged
# message to stay in the queue, the latter defines maximum number of
# messages in the queue.
#
# If one of those conditions becomes true (age > maxage || size > maxsize),
# the log message stating that fact is generated and the log queue is
# flushed in the separate thread.
#

```

ログファイルのパス

```
# If you ever see such a message, either your system doesn't live up to its
# expectations or you have a runaway loop (probably, but not necessarily,
# generating a lot of log messages).
#
# WARNING: Default values are lousy, you probably want to tweak them and
# report the results back to the development team.

# Syntax: log.queue.maxage = [milliseconds]
# Default: 5000
log.queue.maxage = 5000

# Syntax: log.queue.maxsize = [integer]
# Default: 1000
log.queue.maxsize = 1000

# Enable/disable channels, each logging different actions.
# Syntax: log.channel.[channel name]=[true,false] (boolean)
# Default: false
log.channel.init=true
log.channel.terminate=true
log.channel.serviceRequest=true
log.channel.authentication=true
log.channel.signal=true
log.channel.exceptionTracing=true
log.channel.servletLog=true

# These channels are mainly for internal debugging purposes.
#log.channel.requestData=true
#log.channel.responseHeaders=true
#log.channel.servletManager=true
#log.channel.singleThreadModel=true
#log.channel.queueStatus=true
```

*****zone.properties の設定項目 (/usr/local/java/Apache-JServ1.0/conf/*****zone.properties)
 注) 以下の設定ファイルは第 1 文字が # の場合はコメントである。

```
#####
#                               Servlet Zone Configuration File                               #
#####

##### W A R N I N G #####
#
# Unlike normal Java properties, JServ configurations have some important
# extentions:
#
# 1) commas are used as token separators
# 2) multiple definitions of the same key are concatenated in a
#    comma-separated list.
#
#####

# List of Repositories
#####

# The list of servlet repositories controlled by this servlet zone
# Syntax: repositories=[repository],[repository]...
# Default: NONE
# Note: The classes you want to be reloaded upon modification should be put
#       here.

repositories=/home/wg73/system/servlets,/usr/local/java/xml-
tr2/xml.jar,/usr/local/java/cos/lib/cos.jar,/home/wg73/pgm

# Classloader parameters
#####

# Enable servlet class autoreloading.
# Syntax: autoreload.classes=[true,false] (boolean)
# Default: true
autoreload.classes=true

# Enable servlet resourced autoreloading (properties and other loaded resources)
# Syntax: autoreload.file=[true,false] (boolean)
# Default: true
autoreload.file=true

# Set the number of millisecond to wait before giving up on initializing a servlet.
# (a timeout of zero means no timeout)
# Syntax: init.timeout=(long)>0
# Default: 10000 (10 secs)
init.timeout=10000

# Set the number of millisecond to wait before giving up on destroying a servlet.
# (a timeout of zero means no timeout)
# Syntax: destroy.timeout=(long)>0
# Default: 10000 (10 secs)
destroy.timeout=10000

# Set the number of millisecond to wait before invalidating an unused session.
# Syntax: session.timeout=(long)>0
# Default: 1800000 (30 mins)
session.timeout=1800000

# Set how frequently (milliseconds) to check for timed-out sessions.
# Syntax: session.checkFrequency=(long)>0
# Default: 30000 (30 secs)
session.checkFrequency=30000

# SingleThreadModel Servlets parameters
#####
```

servlet のパス system のときはこのとおり secure のときは repositories=/home/wg73/secure/servlets share のときは repositories=/home/wg73/share/servlets

```

# Set the initial capacity of the STM servlets pool.
# Syntax: singleThreadModelServlet.initialCapacity=(int)>1
# Default: 5
singleThreadModelServlet.initialCapacity=5

# Set the number of servlet instances should be added to the pool if found empty.
# Syntax: singleThreadModelServlet.incrementCapacity=(int)>1
# Default: 5
singleThreadModelServlet.incrementCapacity=5

# Set the maximum capacity of the STM pool
# Syntax: singleThreadModelServlet.maximumCapacity=(int)>1
# Default: 10
singleThreadModelServlet.maximumCapacity=10

##### S E R V L E T   P A R A M E T E R S #####

##### N O T E #####
# When "classname" is specified, it means a Java dot-formatter full class name
# without the ".class". For example, a class with source file named
# "Dummy.java" with a package name "org.fool" is defined as "org.fool.Dummy".
#
# Since each servlet may have lots of private initialization data, Apache JServ
# allows you to store those servlet initArgs in a separate file. To do this,
# simply do not set any initArgs in this file: Apache JServ will then look for
# a file named "[servlet classname].initargs" in the same directory of that
# class. Note that this may work with even class archives.
#####

# Startup Servlets
#####

# Comma or space delimited list of servlets to launch on startup.
# This can either be a class name or alias.
# Syntax: servlets.startup=[classname or alias],[classname or alias],...
# Default: NONE
# servlets.startup=hello,snoop,org.fool.Dummy

# Servlet Aliases
#####

# This defines aliases from which servlets can be invoked.
# Each alias give a new instance of the servlet. This means that if a servlet
# is invoked both by class name and by alias name, it will result in _TWO_
# instances of the servlet being created.
# Syntax: servlet.[alias].code=[classname] (String)
# Default: NONE
# servlet.snoop.code=SnoopServlet
# servlet.hello.code=org.fool.Dummy

# Global Init Parameters
#####

# Parameters passed here are given to each of servlets. You should put
# configuration information that is common to all servlets.
#
# The value of the property is a comma delimited list of "name=value" pairs
# that are accessible to the servlet via the method getInitParameter()
# in ServletConfig.
# Syntax: servlets.default.initArgs=[name]=[value],[name]=[value],...
# Default: NONE
# servlets.default.initArgs=common.to.everybody=Hi everybody!

# Servlet Init Parameters
#####

# These properties define init parameters for each servlet that is invoked
# by its classname.

```

```
# Syntax: servlet.[classname].initArgs=[name]=[value],[name]=[value],...
# Default: NONE
# servlet.org.fool.Dummy.initArgs=message=I'm a dummy servlet

# Aliased Servlet Init Parameters
#####

# These properties define init parameters for each servlet that is invoked
# by its alias.
# Syntax: servlet.[alias].initArgs=[name]=[value],[name]=[value],...
# Default: NONE
# servlet.snoop.initArgs=message=I'm a snoop servlet
# servlet.hello.initArgs=message=I say hello world to everyone
```

A p a c h e用ユーザディレクトリ

Port	ディレクトリ名	ディレクトリパス
443	ドキュメントディレクトリ	/home/wg73/secure/htdocs/
	ダウンロード用ディレクトリ	/home/wg73/secure/htdocs/download/
	アップロード用ディレクトリ	/home/wg73/secure/htdocs/upload/
	C G Iディレクトリ	/home/wg73/secure/cgi-bin/
	Servletディレクトリ	/home/wg73/secure/servlets/
	掲示板用ディレクトリ	/home/wg73/secure/htdocs/board/
	ログディレクトリ	/home/wg73/secure/logs/
80	ドキュメントディレクトリ	/home/wg73/share/htdocs/
	C G Iディレクトリ	/home/wg73/share/cgi-bin/
	Servletディレクトリ	/home/wg73/share/servlets/
	ログディレクトリ	/home/wg73/share/logs/
27163	ドキュメントディレクトリ	/home/wg73/system/htdocs/
	C G Iディレクトリ	/home/wg73/system/cgi-bin/
	Servletディレクトリ	/home/wg73/system/servlets/
	ログディレクトリ	/home/wg73/system/logs/