Apache Quick Reference Card

Revision 1.6 for Apache version 1.3.6

Andrew Ford **refcards.com**™

This card summarizes information in the Apache documentation, which can be found at http://www.apache.org/docs/.

Each directive is listed with its arguments. Defaults are given where appropriate in parentheses at the end of the line, followed by an indication of the status of the directive (C – core, B – base, E – extension) and the contexts in which the directive is valid (c – in a configuration file, v – within a virtual host section, d – within a directory section, a – in an access control file).

Options and directives that are new for Apache 1.3 are marked **1.3**, while those that have changed are marked **⊙**.

Command line options

```
httpd [-d directory] [-f file] [-C directive]
      [-c directive] [-L] [-l] [-S] [-v] [-V] [-X]
```

-d directory	initial value for ServerRoot	
-D parameter	define parameter for <IfDefine>	
-f file	configuration file (conf/httpd.conf)	
-C directive	process directive before reading config files	**1.3**
-c directive	process directive after reading config files	**1.3**
-L	list available configuration directives and exit	⊙
-l	list compiled-in modules and exit	⊙
-S	show *vhost* settings	
-t	test syntax of configuration files and exit	
-v	print version and build date, and exit	
-V	show compilation settings and exit	**1.3**
-X	single-processs debug mode	

General configuration directives

Directive	Default	Status
DocumentRoot *directory*	(/usr/local/apache/htdocs)	C cv
ServerRoot *directory*	(/usr/local/apache)	C c
AccessConfig *filename*	(conf/access.conf)	C cv
ResourceConfig *filename*	(conf/srm.conf)	C cv
Include *filename*		C cvd **1.3**
TypesConfig *filename*	(conf/mime.types)	B c
PidFile *filename*	(logs/httpd.pid)	C c
LockFile *filename*	(logs/accept.lock)	C c
ScoreBoardFile *file*	(logs/apache_status)	C c
ServerAdmin *email-addr*		C cv
ServerType { inetd|standalone }	(standalone)	C c
ServerName *name*	(*determined automatically*)	C cv
ServerAlias *fqdn* ...		C v
ServerSignature { On|Off|Email }	(Off)	C cvda **1.3**
ServerPath *pathname*		C v
ServerTokens { Min[imal]|OS|Full }	(Full)	C c **1.3**
BindAddress { *|IP-addr|fqdn }	(*)	C c
Listen [IP-addr:]port-number		C c
Port *portno*	(80)	C c
User { username|#userid }	(#-1)	C cv
Group { group-name|#group-id }	(#-1)	C cv
NameVirtualHost addr[:port]		C c **1.3**
UseCanonicalName { On|Off }	(On)	C cvda **1.3**
CoreDumpDirectory *directory*	(*as ServerRoot*)	C c **1.3**
IdentityCheck { On|Off }	(Off)	C cvd

Directive	Status
ClearModuleList	C c
AddModule *module* ...	C c
AddModuleInfo *module text*	E cv **1.3**
ExtendedStatus { On|Off }	(Off) E c **1.3**

Sectional directives

Directive	Status
<Directory { *pattern* | ~ *regex* } > ... </Directory>	C cv ⊙
<DirectoryMatch *regex*> ... </DirectoryMatch>	C cv **1.3**
<Files { *pattern*|~ *regex* } > ... </Files>	C cvda ⊙
<FilesMatch *regex*> ... </FilesMatch>	C cvda **1.3**
<Location { *pattern*|~ *regex* } > ... </Location>	C cv ⊙
<LocationMatch *regex*> ... </LocationMatch>	C cv **1.3**
<VirtualHost *pattern*> ... </VirtualHost>	C c
<IfDefine [!]*parameter*> ... </IfDefine>	C cvda **1.3**
<IfModule [!]*module*> ... </IfModule>	C cvda
<Limit *method*> ... </Limit>	C cvda
<LimitExcept *method*> ... </LimitExcept>	C cvda

Logging directives

Directive	Default	Status
ErrorLog { *file*|syslog[:*facility*] }	(logs/error_log)	C cv ⊙
CustomLog *file-or-pipe* { *format-string*|*format-name* } [env=[!]*var*]		B cv ⊙
TransferLog *file-or-pipe*	(logs/access_log)	B cv
HostnameLookups { On|Off|Double }	(Off)	C cvd ⊙
RefererIgnore *hostname* ...		E cv
LogFormat *format-string* [*format-name*]	("%h %l %u %t \"%r\" %s %b")	B cv ⊙
LogLevel {emerg|alert|crit|error|warn|notice| info|debug }	(error)	C cv **1.3**

format-string can contain literal text and format specifiers:

%a	Remote IP address
%b	Number of bytes sent (excluding headers)
%{*var*}e	Environment variable value
%f	Filename
%h	Remote host
%{*hdr*}i	Incoming header value
%l	Remote logname (from identd)
%{*label*}n	Labelled note from another module **1.3**
%{*hdr*}o	Outgoing header value
%p	Canonical port number of server
%P	Process ID of child server process
%r	First line of request
%s	Status
%t	Time in Common Log Format (CLF) format
%{*format*}t	Time (*format* is strftime format)
%T	Time taken to serve the request (in seconds)
%u	Remote username
%U	URL requested
%v	Canonical ServerName of server
%V	Server name as per UseCanonicalName setting ⊙

Performance tuning directives

Directive	Default	Status
Timeout *secs*	(300)	C c
KeepAlive { On|Off }	(On)	C c
KeepAliveTimeout *seconds*	(15)	C c
MaxKeepAliveRequests *number*	(100)	C c
MaxClients *number*	(256)	C c
MaxRequestsPerChild *number*		C c
ExcessRequestsPerChild *number*		E c **1.3**
MaxSpareServers *number*	(10)	C c
MinSpareServers *number*	(5)	C c
StartServers *number*	(5)	C c
ListenBacklog *number*	(511)	C c **1.3**
LimitRequestBody *nbytes*		C cvda **1.3**
LimitRequestFields *number*	(100)	C c **1.3**
LimitRequestFieldSize *nbytes*	(8190)	C c **1.3**
LimitRequestLine *nbytes*	(8190)	C c **1.3**
SendBufferSize *bytes*		C c
RLimitCPU { *secs*|max } [{ *secs*|max }]		C cvda
RLimitMEM { *bytes*|max } [{ *bytes*|max }]		C cvda
RLimitNPROC { *nprocs*|max } [{ *nprocs*|max }]		C cvda
ThreadsPerChild *number*	(50)	C c **1.3**

File typing directives

Directive	Default	Status
DefaultType *MIME-type*	(text/html)	C cvda
ForceType *MIME-type*		B da
AddEncoding *MIME-encoding extension* ...		B cvda
AddLanguage *MIME-language extension* ...		B cvda
AddType *MIME-type extension* ...		B cvda
DefaultLanguage *MIME-language*	(no default language)	C cvda **1.3**
LanguagePriority *MIME-language* ...		B cvda
XBitHack { On|Off|Full }	(Off)	B cvda
MetaFiles { On|Off }	(Off)	E cvda **1.3**
MetaDir *dir-name*	(.web)	E cvda ⊙
MetaSuffix *suffix*	(.meta)	E cvda ⊙
MimeMagicFile *filename*		E cv **1.3**
ContentDigest { On|Off }	(Off)	C cvda
ExpiresActive { On|Off }		E cvda
ExpiresBytype *MIME-type expiry-code*		E cvda
ExpiresDefault *expiry-code*		E cvda
Header { set|append|add|unset } *header value*		E cvda

Mapping directives

Directive	Default	Status
Action { *MIME-type*|*handler-name* } *script*		B cvda
Alias *url-path real-path*		B cv
AliasMatch *url-regex path-replacement*		B cv **1.3**
ErrorDocument *error-code* { *string*|*url* }		C cvda
UserDir { *directory*|DISABLED [*user* ...]| ENABLED *user* ... }	(public_html)	B cv ⊙
Redirect [*status*] *url-path url*		B cvda
RedirectMatch [*status*] *url-regex dest-url*		B cv **1.3**
RedirectPermanent *url-path url-dest*		B cvda
RedirectTemp *url-path url-dest*		B cvda
Script *method script*		B cvd
ScriptAlias *url-path script-dir*		B cv
ScriptAliasMatch *url-regex script-dir*		B cv **1.3**
AddHandler *handler-name extension* ...		B cvda
RemoveHandler *extension* ...		B da **1.3**
SetHandler *handler-name*		B da
ImapBase { map|referer|*url* }	(http://*server*)	B cvda
ImapDefault { error|nocontent|referer|menu| *url* }	(nocontent)	B cvda
ImapMenu { none|formatted|semiformatted|unformatted }		B cvda
RewriteBase *url*		E cvda
RewriteCond *test-string cond-pattern* [*flags*]		E cvda
RewriteEngine { On|Off }	(Off)	E cvda
RewriteLock *filename*		E cv **1.3**
RewriteLog *filename*		E cv
RewriteLogLevel *digit*		E cv

Scripting directives

Directive	Default	Context
RewriteMap *map-name* *map-type*:*map-source*		E cv [1.3]
RewriteOptions *option*		E cvda
RewriteRule *pattern* *substitution* [*flags*]		E cvda [1.3]

Scripting directives

Directive	Default	Context
BrowserMatch *regexp* *var*[=*value*] ...		B cv
BrowserMatchNoCase *regexp* *var*[=*value*] ...		B cv
PassEnv *varname* ...		B cv
SetEnv *varname* *value*		B cv
SetEnvIf *attribute* *regex* *var*[=*value*] ...		B cv [1.3]
SetEnvIfNoCase *attribute* *regex* *var*[=*value*] ...		B cv [1.3]
UnsetEnv *varname* ...		B cv
ScriptLog *filename*		B c
ScriptLogBuffer *size-bytes*	(1024)	B c
ScriptLogLength *length-bytes*	(10385760)	B c

Directory indexing directives

Directive	Default	Context
DirectoryIndex *file* ...	(index.html)	B cvda
ReadmeName *filename*		B cvda
HeaderName *filename*		B cvda
IndexIgnore *pattern* ...		B cvda
IndexOrderDefault {Ascending|Descending} {Name|Date|Size|Description}		B cvda [1.3]
DefaultIcon *url*		B cvda
AddAlt "*string*" *file* ...		B cvda
AddAltByEncoding "*string*" *MIME-encoding* ...		B cvda
AddAltByType "*string*" *MIME-type* ...		B cvda
AddIcon *icon* *file* ...		B cvda
AddIconByType *icon* *MIME-type* ...		B cvda
AddIconByEncoding *icon* *MIME-encoding* ...		B cvda
AddDescription "*description*" *file-pattern*		B cvda
IndexOptions {+*option*|-*option*} ...	(*no options*)	B cvda [1.3]

Valid index options are: FancyIndexing, IconsHeight=*pixels*, IconsWidth=*pixels*, NameWidth=*n*, IconsAreLinks, ScanHTMLTitles, SupressColumnSorting, SupressDescription, SupressHTMLPreamble, SupressLastModified, SupressSize

Access control directives

Directive	Default	Context
AccessFileName *filename* ...	(.htaccess)	C cv [1.3]
AllowOverride *directive-group* ...	(All)	C d

Valid directive groups are: None, All, AuthConfig, FileInfo, Indexes, Limit, Options

Directive	Default	Context
Options {+*feature*|-*feature*} ...	(All)	C cvda

Valid options are: None, All, ExecCGI, FollowSymLinks, Includes, IncludesNOEXEC, Indexes, MultiViews, SymLinksIfOwnerMatch

Directive	Default	Context
AuthName *realm-name*		C da
AuthType *type*		C da
AuthUserFile *filename*		E da
AuthGroupFile *filename*		E da
AuthAuthoritative {On|Off}	(On)	B da
allow from {all|*host* ...|env=*var*}		B da
deny from {all|*host* ...|env=*var*}		B da
order {allow,deny|deny,allow|mutual-failure}	(deny,allow)	B da
Require {user *user* ...|group *group* ...|valid-user}		C da
Satisfy {All|Any}	(All)	C da

Proxy/cache directives

Directive	Default	Context
NoProxy {*host*|*domain*|*ip-addr*|*subnet*} ...		E cv [1.3]
AllowCONNECT *port* ...	(443 563)	E cv [1.3]
ProxyBlock {*|*host*|*domain*} ...		E cv
ProxyDomain *domain-name*		E cv [1.3]
ProxyPass *path* *url*		E cv
ProxyPassReverse *path* *url*		E cv [1.3]
ProxyReceiveBufferSize *number-bytes*		E cv [1.3]
ProxyRemote *match* *remote-server*		E cv
ProxyRequests {On|Off}	(Off)	E cv
ProxyVia {On|Off|Full|Block}	(Off)	E cv [1.3]
CacheDefaultExpire *hours*	(1)	E cv
CacheDirLength *number*	(1)	E cv
CacheDirLevels *number*	(3)	E cv
CacheForceCompletion *percentage-complete*	(90)	E cv [1.3]
CacheGcInterval *hours*		E cv
CacheLastModifiedFactor *factor*	(0.1)	E cv
CacheMaxExpire *hours*	(24)	E cv
CacheRoot *directory*		E cv
CacheSize *kb*	(5)	E cv
NoCache {*|*host*|*domain*} ...		E cv

Common Gateway Interface (CGI) variables

Variable	Description
AUTH_TYPE	Authentication method used
CONTENT_LENGTH	Length of input stream data
CONTENT_TYPE	Internet media type of input stream
GATEWAY_INTERFACE	CGI version
PATH_INFO	URL part after script identifier
PATH_TRANSLATED	PATH_INFO translated into filesystem
QUERY_STRING	Query string from URL (if present)
REMOTE_ADDR	IP address of client
REMOTE_HOST	DNS name of client
REMOTE_IDENT	Remote user id (unreliable)
REMOTE_USER	Name of the authenticated user
REQUEST_METHOD	HTTP request method
SCRIPT_NAME	Virtual path of the script
SERVER_NAME	Host-name of the server
SERVER_PORT	Port number of server
SERVER_PROTOCOL	Name and version of the protocol
SERVER_SOFTWARE	Server software name and version
HTTP_ACCEPT	MIME types the browser will accept
HTTP_REFERER	URL of referring document
HTTP_USER_AGENT	Client software identification
MOD_PERL	Defined if running under mod_perl

Server Side Includes (SSI)

Format is: `<!--#element attr=value ... -->`

Elements are:

Element	Description
config	controls parsing; attributes: errmsg, sizefmt (one of bytes or abbrev), timefmt (strftime format)
echo	print the value of a variable; attribute var (can be a CGI variable or one of: DATE_GMT, DATE_LOCAL, DOCUMENT_NAME, DOCUMENT_URI or LAST_MODIFIED)
set	set the value of a variable; attributes: var, value
printenv	print a list of all variables and their values
exec	execute a shell command or CGI script; attribute: cmd or cgi (IncludesNOEXEC option disables exec)
fsize	print size of file; attribute: file or virtual
flastmod	print file modification time; attribute file or virtual
include	include another document; attribute file or virtual
if	conditional; attribute: expr
elif	conditional; attribute: expr
else	conditional
endif	conditional

HTTP 1.1 status codes

Code	Meaning	Code	Meaning
100	Continue	404	Not found
101	Switching protocols	405	Method not allowed
200	OK	406	Not acceptable
201	Created	407	Proxy authentication required
202	Accepted	408	Request timeout
203	Non-authoritative information	409	Conflict
204	No content	410	Gone
205	Reset content	411	Length required
206	Partial content	412	Precondition failed
300	Multiple choices	413	Request entity too large
301	Moved permanently	414	Request-URI too long
302	Moved temporarily	415	Unsupported media type
303	See other	500	Internal server error
304	Not modified	501	Not implemented
305	Use proxy	502	Bad gateway
400	Bad request	503	Service unavailable
401	Unauthorized	504	Gateway timeout
402	Payment required	505	HTTP version not supported
403	Forbidden		

Apache Web Resources

URL	Description
http://www.apache.org	Apache home page
http://modules.apache.org	Apache Module Registry
http://www.apache-ssl.org	Apache-SSL home page
http://www.modssl.org	mod_ssl home page
http://java.apache.org	The Java-Apache Project
http://perl.apache.org	The Apache/Perl Integration Project
http://www.php.net	PHP: Hypertext Processor
http://www.apacheweek.com	Apache Week
http://www.netcraft.com/survey	Netcraft Server Survey
http://www.refcards.com	Quick Reference Cards