Rocky Enterprise Linux 9.2 Manual Pages on command 'ab.1'

$ man ab.1

AB(1)   ab   AB(1)

NAME

    ab - Apache HTTP server benchmarking tool

SYNOPSIS

    ab [-A auth-username:password] [-b windowsize] [-B local-address] [-c concurrency]

    [-C cookie-name=value] [-d] [-e csv-file] [-E client-certificate file] [-f proto?

    col] [-g gnuplot-file] [-h] [-H custom-header] [-i] [-k] [-l] [-m HTTP-

    method] [-n requests] [-p POST-file] [-P proxy-auth-username:password] [-q] [-r

    ] [-s timeout] [-S] [-t timelimit] [-T content-type] [-u PUT-file] [-v ver?

    bosity] [-V] [-w] [-x <table>-attributes] [-X proxy[:port]] [-y <tr>-attributes]

    [-z <td>-attributes] [-Z ciphersuite] [http[s]://]hostname[:port]/path

SUMMARY

    ab is a tool for benchmarking your Apache Hypertext Transfer Protocol (HTTP) server. It is

    designed to give you an impression of how your current Apache installation performs. This

    especially shows you how many requests per second your Apache installation is capable of

    serving.

OPTIONS

    -A auth-username:password

        Supply BASIC Authentication credentials to the server. The username and password

        are separated by a single : and sent on the wire base64 encoded. The string is sent

        regardless of whether the server needs it (i.e., has sent an 401 authentication

        needed).

    -b windowsize

Size of TCP send/receive buffer, in bytes.

-B local-address

Address to bind to when making outgoing connections.

-c concurrency

Number of multiple requests to perform at a time. Default is one request at a time.

-C cookie-name=value

Add a Cookie: line to the request. The argument is typically in the form of a name=value pair. This field is repeatable.

-d Do not display the "percentage served within XX [ms] table". (legacy support).

-e csv-file

Write a Comma separated value (CSV) file which contains for each percentage (from 1% to 100%) the time (in milliseconds) it took to serve that percentage of the re? quests. This is usually more useful than the 'gnuplot' file; as the results are al? ready 'binned'.

-E client-certificate-file

When connecting to an SSL website, use the provided client certificate in PEM for? mat to authenticate with the server. The file is expected to contain the client certificate, followed by intermediate certificates, followed by the private key. Available in 2.4.36 and later.

-f protocol

Specify SSL/TLS protocol (SSL2, SSL3, TLS1, TLS1.1, TLS1.2, or ALL). TLS1.1 and TLS1.2 support available in 2.4.4 and later.

-g gnuplot-file

Write all measured values out as a 'gnuplot' or TSV (Tab separate values) file. This file can easily be imported into packages like Gnuplot, IDL, Mathematica, Igor or even Excel. The labels are on the first line of the file.

-h Display usage information.

-H custom-header

Append extra headers to the request. The argument is typically in the form of a valid header line, containing a colon-separated field-value pair (i.e., "Accept-En? coding: zip/zop;8bit").

-i Do HEAD requests instead of GET.

-k Enable the HTTP KeepAlive feature, i.e., perform multiple requests within one HTTP

session. Default is no KeepAlive.

-l   Do not report errors if the length of the responses is not constant. This can be

useful for dynamic pages. Available in 2.4.7 and later.

-m HTTP-method

Custom HTTP method for the requests. Available in 2.4.10 and later.

-n requests

Number of requests to perform for the benchmarking session. The default is to just

perform a single request which usually leads to non-representative benchmarking re?

sults.

-p POST-file

File containing data to POST. Remember to also set -T.

-P proxy-auth-username:password

Supply BASIC Authentication credentials to a proxy en-route. The username and pass?

word are separated by a single : and sent on the wire base64 encoded. The string is

sent regardless of whether the proxy needs it (i.e., has sent an 407 proxy authen?

tication needed).

-q   When processing more than 150 requests, ab outputs a progress count on stderr every

10% or 100 requests or so. The -q flag will suppress these messages.

-r   Don't exit on socket receive errors.

-s timeout

Maximum number of seconds to wait before the socket times out. Default is 30 sec?

onds. Available in 2.4.4 and later.

-S   Do not display the median and standard deviation values, nor display the warn?

ing/error messages when the average and median are more than one or two times the

standard deviation apart. And default to the min/avg/max values. (legacy support).

-t timelimit

Maximum number of seconds to spend for benchmarking. This implies a -n 50000 inter?

nally. Use this to benchmark the server within a fixed total amount of time. Per

default there is no timelimit.

-T content-type

Content-type header to use for POST/PUT data, eg. application/x-www-form-urlen?

coded. Default is text/plain.

-u PUT-file

File containing data to PUT. Remember to also set -T.

-v verbosity

Set verbosity level - 4 and above prints information on headers, 3 and above prints

response codes (404, 200, etc.), 2 and above prints warnings and info.

-V   Display version number and exit.

-w   Print out results in HTML tables. Default table is two columns wide, with a white

background.

-x <table>-attributes

String to use as attributes for <table>. Attributes are inserted <table here >.

-X proxy[:port]

Use a proxy server for the requests.

-y <tr>-attributes

String to use as attributes for <tr>.

-z <td>-attributes

String to use as attributes for <td>.

-Z ciphersuite

Specify SSL/TLS cipher suite (See openssl ciphers)

OUTPUT

The following list describes the values returned by ab:

Server Software

The value, if any, returned in the server HTTP header of the first successful re?

sponse. This includes all characters in the header from beginning to the point a

character with decimal value of 32 (most notably: a space or CR/LF) is detected.

Server Hostname

The DNS or IP address given on the command line

Server Port

The port to which ab is connecting. If no port is given on the command line, this

will default to 80 for http and 443 for https.

SSL/TLS Protocol

The protocol parameters negotiated between the client and server. This will only be

printed if SSL is used.

Document Path

The request URI parsed from the command line string.

Document Length

This is the size in bytes of the first successfully returned document. If the docu‐

ment length changes during testing, the response is considered an error.

Concurrency Level

The number of concurrent clients used during the test

Time taken for tests

This is the time taken from the moment the first socket connection is created to

the moment the last response is received

Complete requests

The number of successful responses received

Failed requests

The number of requests that were considered a failure. If the number is greater

than zero, another line will be printed showing the number of requests that failed

due to connecting, reading, incorrect content length, or exceptions.

Write errors

The number of errors that failed during write (broken pipe).

Non-2xx responses

The number of responses that were not in the 200 series of response codes. If all

responses were 200, this field is not printed.

Keep-Alive requests

The number of connections that resulted in Keep-Alive requests

Total body sent

If configured to send data as part of the test, this is the total number of bytes

sent during the tests. This field is omitted if the test did not include a body to

send.

Total transferred

The total number of bytes received from the server. This number is essentially the

number of bytes sent over the wire.

HTML transferred

The total number of document bytes received from the server. This number excludes

bytes received in HTTP headers

Requests per second

This is the number of requests per second. This value is the result of dividing the

number of requests by the total time taken

Time per request

The average time spent per request. The first value is calculated with the formula concurrency * timetaken * 1000 / done while the second value is calculated with the formula timetaken * 1000 / done

Transfer rate

The rate of transfer as calculated by the formula totalread / 1024 / timetaken

BUGS

There are various statically declared buffers of fixed length. Combined with the lazy parsing of the command line arguments, the response headers from the server and other ex? ternal inputs, this might bite you.

It does not implement HTTP/1.x fully; only accepts some 'expected' forms of responses. The rather heavy use of strstr(3) shows up top in profile, which might indicate a performance problem; i.e., you would measure the ab performance rather than the server's.

Apache HTTP Server   2018-10-10   AB(1)