Rocky Enterprise Linux 9.2 Manual Pages on command 'gnutls-cli.1'

$ man gnutls-cli.1

gnutls-cli(1) User Commands gnutls-cli(1)

NAME

gnutls-cli - GnuTLS client

SYNOPSIS

gnutls-cli [-flags] [-flag [value]] [--option-name[[=|]value]] [host?

name]

Operands and options may be intermixed. They will be reordered.

DESCRIPTION

Simple client program to set up a TLS connection to some other com?

puter. It sets up a TLS connection and forwards data from the standard

input to the secured socket and vice versa.

OPTIONS

-d num, --debug=num

Enable debugging. This option takes an integer number as its

argument. The value of num is constrained to being:

in the range 0 through 9999

Specifies the debug level.

-V, --verbose

More verbose output.

--tofu, --no-tofu

Enable trust on first use authentication. The no-tofu form will disable the option.

This option will, in addition to certificate authentication, perform authentication based on previously seen public keys, a model similar to SSH authentication. Note that when tofu is specified (PKI) and DANE authentication will become advisory to assist the public key acceptance process.

--strict-tofu, --no-strict-tofu

Fail to connect if a certificate is unknown or a known certifi? cate has changed. The no-strict-tofu form will disable the op? tion.

This option will perform authentication as with option --tofu; however, no questions shall be asked whatsoever, neither to ac? cept an unknown certificate nor a changed one.

--dane, --no-dane

Enable DANE certificate verification (DNSSEC). The no-dane form will disable the option.

This option will, in addition to certificate authentication us? ing the trusted CAs, verify the server certificates using on the DANE information available via DNSSEC.

--local-dns, --no-local-dns

Use the local DNS server for DNSSEC resolving. The no-local-dns form will disable the option.

This option will use the local DNS server for DNSSEC. This is disabled by default due to many servers not allowing DNSSEC.

--ca-verification, --no-ca-verification

Enable CA certificate verification. The no-ca-verification form will disable the option. This option is enabled by default.

This option can be used to enable or disable CA certificate ver? ification. It is to be used with the --dane or --tofu options.

--ocsp, --no-ocsp

Enable OCSP certificate verification. The no-ocsp form will disable the option.

This option will enable verification of the peer's certificate using ocsp

-r, --resume

Establish a session and resume.

Connect, establish a session, reconnect and resume.

--earlydata=str

Send early data on resumption from the specified file.

-e, --rehandshake

Establish a session and rehandshake.

Connect, establish a session and rehandshake immediately.

--sni-hostname=str

Server's hostname for server name indication extension.

Set explicitly the server name used in the TLS server name indi? cation extension. That is useful when testing with servers setup on different DNS name than the intended. If not specified, the provided hostname is used. Even with this option server certifi? cate verification still uses the hostname passed on the main commandline. Use --verify-hostname to change this.

--verify-hostname=str

Server's hostname to use for validation.

Set explicitly the server name to be used when validating the server's certificate.

-s, --starttls

Connect, establish a plain session and start TLS.

The TLS session will be initiated when EOF or a SIGALRM is re? ceived.

--app-proto

This is an alias for the --starttls-proto option.

--starttls-proto=str

The application protocol to be used to obtain the server's cer? tificate (https, ftp, smtp, imap, ldap, xmpp, lmtp, pop3, nntp,

sieve, postgres). This option must not appear in combination with any of the following options: starttls.

Specify the application layer protocol for STARTTLS. If the pro? tocol is supported, gnutls-cli will proceed to the TLS negotia? tion.

-u, --udp

Use DTLS (datagram TLS) over UDP.

--mtu=num

Set MTU for datagram TLS. This option takes an integer number as its argument. The value of num is constrained to being:

in the range 0 through 17000

--crlf Send CR LF instead of LF.

--fastopen

Enable TCP Fast Open.

--x509fmtder

Use DER format for certificates to read from.

--print-cert

Print peer's certificate in PEM format.

--save-cert=str

Save the peer's certificate chain in the specified file in PEM format.

--save-ocsp=str

Save the peer's OCSP status response in the provided file. This option must not appear in combination with any of the following options: save-ocsp-multi.

--save-ocsp-multi=str

Save all OCSP responses provided by the peer in this file. This option must not appear in combination with any of the following options: save-ocsp.

The file will contain a list of PEM encoded OCSP status re? sponses if any were provided by the peer, starting with the one for the peer's server certificate.

--save-server-trace=str

Save the server-side TLS message trace in the provided file.

--save-client-trace=str

Save the client-side TLS message trace in the provided file.

--dh-bits=num

The minimum number of bits allowed for DH. This option takes an integer number as its argument.

This option sets the minimum number of bits allowed for a Diffie-Hellman key exchange. You may want to lower the default value if the peer sends a weak prime and you get an connection error with unacceptable prime.

--priority=str

Priorities string.

TLS algorithms and protocols to enable. You can use predefined sets of ciphersuites such as PERFORMANCE, NORMAL, PFS, SE? CURE128, SECURE256. The default is NORMAL.

Check the GnuTLS manual on section ?Priority strings? for more information on the allowed keywords

--x509cafile=str

Certificate file or PKCS #11 URL to use.

--x509crlfile=file

CRL file to use.

--x509keyfile=str

X.509 key file or PKCS #11 URL to use.

--x509certfile=str

X.509 Certificate file or PKCS #11 URL to use. This option must appear in combination with the following options: x509keyfile.

--rawpkkeyfile=str

Private key file (PKCS #8 or PKCS #12) or PKCS #11 URL to use.

In order to instruct the application to negotiate raw public keys one must enable the respective certificate types via the priority strings (i.e. CTYPE-CLI-* and CTYPE-SRV-* flags).

Check the GnuTLS manual on section ?Priority strings? for more information on how to set certificate types.

--rawpkfile=str

Raw public-key file to use. This option must appear in combina?

tion with the following options: rawpkkeyfile.

In order to instruct the application to negotiate raw public

keys one must enable the respective certificate types via the

priority strings (i.e. CTYPE-CLI-* and CTYPE-SRV-* flags).

Check the GnuTLS manual on section ?Priority strings? for

more information on how to set certificate types.

--srpusername=str

SRP username to use.

--srppasswd=str

SRP password to use.

--pskusername=str

PSK username to use.

--pskkey=str

PSK key (in hex) to use.

-p str, --port=str

The port or service to connect to.

--insecure

Don't abort program if server certificate can't be validated.

--verify-allow-broken

Allow broken algorithms, such as MD5 for certificate verifica?

tion.

--ranges

Use length-hiding padding to prevent traffic analysis.

When possible (e.g., when using CBC ciphersuites), use

length-hiding padding to prevent traffic analysis.

NOTE: THIS OPTION IS DEPRECATED

--benchmark-ciphers

Benchmark individual ciphers.

By default the benchmarked ciphers will utilize any capabilities

of the local CPU to improve performance. To test against the raw

software implementation set the environment variable

GNUTLS_CPUID_OVERRIDE to 0x1.

--benchmark-tls-kx

Benchmark TLS key exchange methods.

--benchmark-tls-ciphers

Benchmark TLS ciphers.

By default the benchmarked ciphers will utilize any capabilities of the local CPU to improve performance. To test against the raw software implementation set the environment variable GNUTLS_CPUID_OVERRIDE to 0x1.

-l, --list

Print a list of the supported algorithms and modes. This option must not appear in combination with any of the following op? tions: port.

Print a list of the supported algorithms and modes. If a prior? ity string is given then only the enabled ciphersuites are shown.

--priority-list

Print a list of the supported priority strings.

Print a list of the supported priority strings. The ciphersuites corresponding to each priority string can be examined using -l -p.

--noticket

Don't allow session tickets.

Disable the request of receiving of session tickets under TLS1.2 or earlier

--srtp-profiles=str

Offer SRTP profiles.

--alpn=str

Application layer protocol. This option may appear an unlimited number of times.

This option will set and enable the Application Layer Protocol Negotiation (ALPN) in the TLS protocol.

--compress-cert=str

Compress certificate. This option may appear an unlimited num?
ber of times.
This option sets a supported compression method for certificate
compression.

-b, --heartbeat

Activate heartbeat support.

--recordsize=num

The maximum record size to advertise. This option takes an in?
teger number as its argument. The value of num is constrained
to being:
in the range 0 through 4096

--disable-sni

Do not send a Server Name Indication (SNI).

--disable-extensions

Disable all the TLS extensions.
This option disables all TLS extensions. Deprecated option. Use
the priority string.
NOTE: THIS OPTION IS DEPRECATED

--single-key-share

Send a single key share under TLS1.3.
This option switches the default mode of sending multiple key
shares, to send a single one (the top one).

--post-handshake-auth

Enable post-handshake authentication under TLS1.3.
This option enables post-handshake authentication when under
TLS1.3.

--inline-commands

Inline commands of the form ^<cmd>^.
Enable inline commands of the form ^<cmd>^. The inline commands
are expected to be in a line by themselves. The available com?
mands are: resume, rekey1 (local rekey), rekey (rekey on both
peers) and renegotiate.

--inline-commands-prefix=str

Change the default delimiter for inline commands.

Change the default delimiter (^) used for inline commands. The delimiter is expected to be a single US-ASCII character (octets 0 - 127). This option is only relevant if inline commands are enabled via the inline-commands option

--provider=file

Specify the PKCS #11 provider library.

This will override the default options in /etc/gnutls/pkcs11.conf

--fips140-mode

Reports the status of the FIPS140-2 mode in gnutls library.

--list-config

Reports the configuration of the library.

--logfile=str

Redirect informational messages to a specific file.

Redirect informational messages to a specific file. The file may be /dev/null also to make the gnutls client quiet to use it in piped server connections where only the server communication may appear on stdout.

--keymatexport=str

Label used for exporting keying material.

--keymatexportsize=num

Size of the exported keying material. This option takes an in? teger number as its argument.

--waitresumption

Block waiting for the resumption data under TLS1.3.

This option makes the client to block waiting for the resumption data under TLS1.3. The option has effect only when --resume is provided.

--ca-auto-retrieve, --no-ca-auto-retrieve

Enable automatic retrieval of missing CA certificates. The no-ca-auto-retrieve form will disable the option.

This option enables the client to automatically retrieve the

missing intermediate CA certificates in the certificate chain,

based on the Authority Information Access (AIA) extension.

-v arg, --version=arg

Output version of program and exit. The default mode is `v', a

simple version. The `c' mode will print copyright information

and `n' will print the full copyright notice.

-h, --help

Display usage information and exit.

-!, --more-help

Pass the extended usage information through a pager.

EXAMPLES

Connecting using PSK authentication

To connect to a server using PSK authentication, you need to enable the

choice of PSK by using a cipher priority parameter such as in the exam?

ple below.

$ ./gnutls-cli -p 5556 localhost --pskusername psk_identity --pskkey 88f3824b3e5659f52d00e959bacab954b6540344 --priority NORMAL:-KX-ALL:+ECDHE-PSK:+DHE-PSK:+PSK

Resolving 'localhost'...

Connecting to '127.0.0.1:5556'...

- PSK authentication.

- Version: TLS1.1

- Key Exchange: PSK

- Cipher: AES-128-CBC

- MAC: SHA1

- Compression: NULL

- Handshake was completed

- Simple Client Mode:

By keeping the --pskusername parameter and removing the --pskkey param?

eter, it will query only for the password during the handshake.

Connecting using raw public-key authentication

To connect to a server using raw public-key authentication, you need to

enable the option to negotiate raw public-keys via the priority strings

such as in the example below.

$ ./gnutls-cli -p 5556 localhost --priority NORMAL:-CTYPE-CLI-ALL:+CTYPE-CLI-RAWPK --rawpkkeyfile cli.key.pem --rawpkfile cli.rawpk.pem

Processed 1 client raw public key pair...

Resolving 'localhost'...

Connecting to '127.0.0.1:5556'...

- Successfully sent 1 certificate(s) to server.

- Server has requested a certificate.

- Certificate type: X.509

- Got a certificate list of 1 certificates.

- Certificate[0] info:

- skipped

- Description: (TLS1.3-Raw Public Key-X.509)-(ECDHE-SECP256R1)-(RSA-PSS-RSAE-SHA256)-(AES-256-GCM)

- Options:

- Handshake was completed

- Simple Client Mode:

Connecting to STARTTLS services

You could also use the client to connect to services with starttls ca?

pability.

$ gnutls-cli --starttls-proto smtp --port 25 localhost

Listing ciphersuites in a priority string

To list the ciphersuites in a priority string:

$ ./gnutls-cli --priority SECURE192 -l

Cipher suites for SECURE192

TLS_ECDHE_ECDSA_AES_256_CBC_SHA384 0xc0, 0x24 TLS1.2

TLS_ECDHE_ECDSA_AES_256_GCM_SHA384 0xc0, 0x2e TLS1.2

TLS_ECDHE_RSA_AES_256_GCM_SHA384 0xc0, 0x30 TLS1.2

TLS_DHE_RSA_AES_256_CBC_SHA256 0x00, 0x6b TLS1.2

TLS_DHE_DSS_AES_256_CBC_SHA256 0x00, 0x6a TLS1.2

TLS_RSA_AES_256_CBC_SHA256 0x00, 0x3d TLS1.2

Certificate types: CTYPE-X.509

Protocols: VERS-TLS1.2, VERS-TLS1.1, VERS-TLS1.0, VERS-SSL3.0, VERS-DTLS1.0

Compression: COMP-NULL

Elliptic curves: CURVE-SECP384R1, CURVE-SECP521R1

PK-signatures: SIGN-RSA-SHA384, SIGN-ECDSA-SHA384, SIGN-RSA-SHA512, SIGN-ECDSA-SHA512

Connecting using a PKCS #11 token

To connect to a server using a certificate and a private key present in

a PKCS #11 token you need to substitute the PKCS 11 URLs in the

x509certfile and x509keyfile parameters.

Those can be found using "p11tool --list-tokens" and then listing all

the objects in the needed token, and using the appropriate.

```
$ p11tool --list-tokens
Token 0:
    URL: pkcs11:model=PKCS15;manufacturer=MyMan;serial=1234;token=Test
    Label: Test
    Manufacturer: EnterSafe
    Model: PKCS15
    Serial: 1234
$ p11tool --login --list-certs "pkcs11:model=PKCS15;manufacturer=MyMan;serial=1234;token=Test"
Object 0:
    URL: pkcs11:model=PKCS15;manufacturer=MyMan;serial=1234;token=Test;object=client;type=cert
    Type: X.509 Certificate
    Label: client
    ID: 2a:97:0d:58:d1:51:3c:23:07:ae:4e:0d:72:26:03:7d:99:06:02:6a
$ MYCERT="pkcs11:model=PKCS15;manufacturer=MyMan;serial=1234;token=Test;object=client;type=cert"
$ MYKEY="pkcs11:model=PKCS15;manufacturer=MyMan;serial=1234;token=Test;object=client;type=private"
$ export MYCERT MYKEY
$ gnutls-cli www.example.com --x509keyfile $MYKEY --x509certfile $MYCERT
```

Notice that the private key only differs from the certificate in the

type.

EXIT STATUS

One of the following exit values will be returned:

0 (EXIT_SUCCESS)

Successful program execution.

1 (EXIT_FAILURE)

The operation failed or the command syntax was not valid.

SEE ALSO

gnutls-cli-debug(1), gnutls-serv(1)

AUTHORS

COPYRIGHT

BUGS

Please send bug reports to: bugs@gnutls.org