Red Hat Enterprise Linux Release 9.2 Manual Pages on 'xmlsec1.1' command

$ man xmlsec1.1

XMLSEC1(1)   User Commands   XMLSEC1(1)

NAME

xmlsec1 - sign, verify, encrypt and decrypt XML documents

SYNOPSIS

xmlsec <command> [<options>] [<files>]

DESCRIPTION

xmlsec is a command line tool for signing, verifying, encrypting and

decrypting XML documents. The allowed <command> values are:

--help display this help information and exit

--help-all

display help information for all commands/options and exit

--help-<cmd>

display help information for command <cmd> and exit

--version

print version information and exit

--keys keys XML file manipulation

--sign sign data and output XML document

--verify

verify signed document

--sign-tmpl

create and sign dynamicaly generated signature template

--encrypt

encrypt data and output XML document

--decrypt

decrypt data from XML document

OPTIONS

--ignore-manifests

do not process <dsig:Manifest> elements

--store-references

store and print the result of <dsig:Reference/> element process?
ing just before calculating digest

--store-signatures

store and print the result of <dsig:Signature> processing just
before calculating signature

--enabled-reference-uris <list>

comma separated list of of the following values: "empty",
"same-doc", "local","remote" to restrict possible URI attribute
values for the <dsig:Reference> element

--enable-visa3d-hack

enables Visa3D protocol specific hack for URI attributes pro?
cessing when we are trying not to use XPath/XPointer engine;
this is a hack and I don't know what else might be broken in
your application when you use it (also check "--id-attr" option
because you might need it)

--binary-data <file>

binary <file> to encrypt

--xml-data <file>

XML <file> to encrypt

--enabled-cipher-reference-uris <list>

comma separated list of of the following values: "empty",
"same-doc", "local","remote" to restrict possible URI attribute
values for the <enc:CipherReference> element

--session-key <keyKlass>-<keySize>

generate new session <keyKlass> key of <keySize> bits size (for
example, "--session des-192" generates a new 192 bits DES key
for DES3 encryption)

--output <filename>

write result document to file <filename>

--print-debug

print debug information to stdout

--print-xml-debug

print debug information to stdout in xml format

--dtd-file <file>

load the specified file as the DTD

--node-id <id>

set the operation start point to the node with given <id>

--node-name [<namespace-uri>:]<name>

set the operation start point to the first node with given <name> and <namespace> URI

--node-xpath <expr>

set the operation start point to the first node selected by the specified XPath expression

--id-attr[:<attr-name>] [<node-namespace-uri>:]<node-name>

adds attributes <attr-name> (default value "id") from all nodes with<node-name> and namespace <node-namespace-uri> to the list of known ID attributes; this is a hack and if you can use DTD or schema to declare ID attributes instead (see "--dtd-file" op?tion), I don't know what else might be broken in your applica?tion when you use this hack

--enabled-key-data <list>

comma separated list of enabled key data (list of registered key data klasses is available with "--list-key-data" command); by default, all registered key data are enabled

--enabled-retrieval-uris <list>

comma separated list of of the following values: "empty", "same-doc", "local","remote" to restrict possible URI attribute values for the <dsig:RetrievalMethod> element.

--gen-key[:<name>] <keyKlass>-<keySize>

generate new <keyKlass> key of <keySize> bits size, set the key name to <name> and add the result to keys manager (for example, "--gen:mykey rsa-1024" generates a new 1024 bits RSA key and sets it's name to "mykey")

--keys-file <file>

load keys from XML file

--privkey-pem[:<name>] <file>[,<cafile>[,<cafile>[...]]]

load private key from PEM file and certificates that verify this key

--privkey-der[:<name>] <file>[,<cafile>[,<cafile>[...]]]

load private key from DER file and certificates that verify this key

--pkcs8-pem[:<name>] <file>[,<cafile>[,<cafile>[...]]]

load private key from PKCS8 PEM file and PEM certificates that verify this key

--pkcs8-der[:<name>] <file>[,<cafile>[,<cafile>[...]]]

load private key from PKCS8 DER file and DER certificates that verify this key

--pubkey-pem[:<name>] <file>

load public key from PEM file

--pubkey-der[:<name>] <file>

load public key from DER file

--aeskey[:<name>] <file>

load AES key from binary file <file>

--deskey[:<name>] <file>

load DES key from binary file <file>

--hmackey[:<name>] <file>

load HMAC key from binary file <file>

--pwd <password>

the password to use for reading keys and certs

--pkcs12[:<name>] <file>

load load private key from pkcs12 file <file>

--pkcs12-persist

persist loaded private key

--pubkey-cert-pem[:<name>] <file>

load public key from PEM cert file

--pubkey-cert-der[:<name>] <file>

load public key from DER cert file

--trusted-pem <file>

load trusted (root) certificate from PEM file <file>

--untrusted-pem <file>

load untrusted certificate from PEM file <file>

--trusted-der <file>

load trusted (root) certificate from DER file <file>

--untrusted-der <file>

load untrusted certificate from DER file <file>

--verification-time <time>

the local time in "YYYY-MM-DD HH:MM:SS" format used certificates

verification

--depth <number>

maximum certificates chain depth

--X509-skip-strict-checks

skip strict checking of X509 data

--insecure

do not verify certificates

--crypto <name>

the name of the crypto engine to use from the following list:

openssl, mscrypto, nss, gnutls, gcrypt (if no crypto engine is

specified then the default one is used)

--crypto-config <path>

path to crypto engine configuration

--repeat <number>

repeat the operation <number> times

--disable-error-msgs

do not print xmlsec error messages

--print-crypto-error-msgs

print errors stack at the end

--help

print help information about the command

--xxe

enable External Entity resolution. WARNING: this may allow the reading of arbitrary files and URLs, controlled by the input XML document. Use with caution!

--url-map:<url> <file>

maps a given <url> to the given <file> for loading external re? sources

AUTHOR

Written by Aleksey Sanin <aleksey@aleksey.com>.

REPORTING BUGS

Report bugs to http://www.aleksey.com/xmlsec/bugs.html

COPYRIGHT

Copyright ? 2002-2016 Aleksey Sanin <aleksey@aleksey.com>. All Rights Reserved..

This is free software: see the source for copying information.

xmlsec1 1.2.29 (openssl) October 2019 XMLSEC1(1)