NAME

ldns-signzone - sign a zonefile with DNSSEC data

SYNOPSIS

ldns-signzone [*OPTIONS*] *ZONEFILE* KEY [KEY [KEY] ...]

DESCRIPTION

ldns-signzone is used to generate a DNSSEC signed zone. When run it will create a new zonefile that contains RRSIG and NSEC resource records, as specified in RFC 4033, RFC 4034 and RFC 4035.

Keys must be specified by their base name (i.e. without .private). If the DNSKEY that belongs to the key in the .private file is not present in the zone, it will be read from the file <base name>.key. If that file does not exist, the DNSKEY value will be generated from the private key.

Multiple keys can be specified, Key Signing Keys are used as such when they are either already present in the zone, or specified in a .key file, and have the KSK bit set.

OPTIONS

-b Augments the zone and the RR's with extra comment texts for a more readable layout, easier to debug. DS records will have a bubblebabble version of the data in the comment text, NSEC3 records will have the unhashed owner names in the comment text.

Without this option, only DNSKEY RR's will have their Key Tag annotated in the comment text.

-d Normally, if the DNSKEY RR for a key that is used to sign the zone is not found in the zone file, it will be read from .key, or derived from the private key (in that order). This option turns that feature off, so that only the signatures are added to the zone.

-e *date*

Set expiration date of the signatures to this date, the format can be YYYYMMDD[hhmmss], or a timestamp.

-f *file*

Use this file to store the signed zone in (default <originalfile>.signed)

-i *date*

Set inception date of the signatures to this date, the format can be YYYYMMDD[hhmmss], or a timestamp.

-o *origin*

Use this as the origin of the zone

-u set SOA serial to the number of seconds since 1-1-1970

-v Print the version and exit

-z *[scheme:]hash*

Calculate the zone's digest and add those as ZONEMD RRs. The (optional) 'scheme' must be 'simple' (or 1) and 'hash' should be 'sha384' (or 1) or 'sha512' (or 2). This option can be given more than once.

-Z Allow ZONEMDs to be added without signing

-A Sign the DNSKEY record with all keys. By default it is signed with a minimal number of keys, to keep the response size for the DNSKEY query small, and only the SEP keys that are passed are used. If there are no SEP keys, the DNSKEY RRset is signed with the non-SEP keys. This option turns off the default and all keys are used to sign the DNSKEY RRset.

-U Sign with every unique algorithm in the provided keys. The DNSKEY set is signed with all the SEP keys, plus all the non-SEP keys that have an algorithm that was not presen in the SEP key set.

-E *name*

Use the EVP cryptographic engine with the given name for signing. This can have some extra options; see ENGINE OPTIONS for more information.

-K *algorithm-id,key-id*

Use the key ‘key-id’ as the signing key for algorithm ‘algorithm-id’ as a Key Signing Key (KSK). This option is used when you use an OpenSSL engine, see ENGINE OPTIONS for more information.

-k *algorithm-id,key-id*
Use the key ‘key-id’ as the signing key for algorithm ‘algorithm-id’ as a Zone Signing Key (ZSK). This option is used when you use an OpenSSL engine, see ENGINE OPTIONS for more information.

-n Use NSEC3 instead of NSEC.

If you use NSEC3, you can specify the following extra options:

-a *algorithm*
Algorithm used to create the hashed NSEC3 owner names

-p Opt-out. All NSEC3 records in the zone will have the Opt-out flag set. After signing, you can add insecure delegations to the signed zone.

-s *string*
Salt

-t *number*
Number of hash iterations

ENGINE OPTIONS

You can modify the possible engines, if supported, by setting an OpenSSL configuration file. This is done through the environment variable OPENSSL_CONF.

The key options (-k and -K) work as follows: you specify a DNSSEC algorithm (using its symbolic name, for instance, RSASHA256 or its numeric identifier, for instance, 8), followed by a comma and a key identifier (white space is not allowed between the algorithm and the comma and between the

comma and the key identifier).

The key identifier can be any of the following:

<id>
<slot>:<id>
id_<id>
slot_<slot>-id_<id>
label_<label>
slot_<slot>-label_<label>

Where '<id>' is the PKCS #11 key identifier in hexadecimal notation, '<label>' is the PKCS #11 human-readable label, and '<slot>' is the slot number where the token is present.

More recent versions of OpenSSL engines may support the PKCS #11 URI scheme (RFC 7512), please consult your engine's documentation.

If not already present, a DNSKEY RR is generated from the key data, and added to the zone.

EXAMPLES

ldns-signzone nlnetlabs.nl Knlnetlabs.nl.+005+12273

Sign the zone in the file 'nlnetlabs.nl' with the key in the files 'Knlnetlabs.nl.+005+12273.private'. If the DNSKEY is not present in the zone, use the key in the file 'Knlnetlabs.nl.+005+12273.key'. If that is not present, generate one with default values from 'Knlnetlabs.nl.+005+12273.private'.

AUTHORS

Written by the ldns team as an example for ldns usage.
Portions of engine support by Vadim Penzin <vadim@penzin.net>.

REPORTING BUGS

Report bugs to <ldns-team@nlnetlabs.nl>.

COPYRIGHT