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

C:\>man nsupdate.1

NSUPDATE(1) BIND9 NSUPDATE(1)

NAME

nsupdate - Dynamic DNS update utility

SYNOPSIS

nsupdate [-d] [-D] [-i] [-L level] [[-g] | [-o] | [-l] | [-y [hmac:]keyname:secret]

| [-k keyfile]] [-t timeout] [-u udptimeout] [-r udpretries] [-v] [-T]

[-P] [-V] [[-4] | [-6]] [filename]

DESCRIPTION

nsupdate is used to submit Dynamic DNS Update requests as defined in RFC 2136 to a name server. This allows resource records to be added or removed from a zone without manually editing the zone file. A single update request can contain requests to add or remove more than one resource record.

Zones that are under dynamic control via nsupdate or a DHCP server should not be edited by hand. Manual edits could conflict with dynamic updates and cause data to be lost.

The resource records that are dynamically added or removed with nsupdate have to be in the same zone. Requests are sent to the zone's master server. This is identified by the MNAME field of the zone's SOA record.

Transaction signatures can be used to authenticate the Dynamic DNS updates. These use the TSIG resource record type described in RFC 2845 or the SIG(0) record described in RFC 2535 and RFC 2931 or GSS-TSIG as described in RFC 3645.

TSIG relies on a shared secret that should only be known to nsupdate and the name

server. For instance, suitable key and server statements would be added to /etc/named.conf so that the name server can associate the appropriate secret key and algorithm with the IP address of the client application that will be using TSIG authentication. You can use ddns-confgen to generate suitable configuration fragments. nsupdate uses the -y or -k options to provide the TSIG shared secret. These options are mutually exclusive.

SIG(0) uses public key cryptography. To use a SIG(0) key, the public key must be stored in a KEY record in a zone served by the name server.

GSS-TSIG uses Kerberos credentials. Standard GSS-TSIG mode is switched on with the -g flag. A non-standards-compliant variant of GSS-TSIG used by Windows 2000 can be switched on with the -o flag.

OPTIONS

-4

Use IPv4 only.

-6

Use IPv6 only.

-d

Debug mode. This provides tracing information about the update requests that are made and the replies received from the name server.

-D

Extra debug mode.

-i

Force interactive mode, even when standard input is not a terminal.

-k keyfile

The file containing the TSIG authentication key. Keyfiles may be in two formats: a single file containing a named.conf-format key statement, which may be generated automatically by ddns-confgen, or a pair of files whose names are of the format K{name}.+157.+{random}.key and K{name}.+157.+{random}.private, which can be generated by dnssec-keygen. The -k may also be used to specify a SIG(0) key used to authenticate Dynamic DNS update requests. In this case, the key specified is not an HMAC-MD5 key.

-l

Local-host only mode. This sets the server address to localhost (disabling the

server so that the server address cannot be overridden). Connections to the local server will use a TSIG key found in /var/run/named/session.key, which is automatically generated by named if any local master zone has set update-policy to local. The location of this key file can be overridden with the -k option.

-L level

Set the logging debug level. If zero, logging is disabled.

-p port

Set the port to use for connections to a name server. The default is 53.

-P

Print the list of private BIND-specific resource record types whose format is understood by nsupdate. See also the -T option.

-r udpretries

The number of UDP retries. The default is 3. If zero, only one update request will be made.

-t timeout

The maximum time an update request can take before it is aborted. The default is 300 seconds. Zero can be used to disable the timeout.

-T

Print the list of IANA standard resource record types whose format is understood by nsupdate. nsupdate will exit after the lists are printed. The -T option can be combined with the -P option.

Other types can be entered using "TYPEXXXXX" where "XXXXX" is the decimal value of the type with no leading zeros. The rdata, if present, will be parsed using the UNKNOWN rdata format, (<backslash> <hash> <space> <length> <space> <hexstring>).

-u udptimeout

The UDP retry interval. The default is 3 seconds. If zero, the interval will be computed from the timeout interval and number of UDP retries.

-v

Use TCP even for small update requests. By default, nsupdate uses UDP to send update requests to the name server unless they are too large to fit in a UDP request in which case TCP will be used. TCP may be preferable when a batch of update requests is made.

-V

Print the version number and exit.

-y [hmac:]keyname:secret

Literal TSIG authentication key. keyname is the name of the key, and secret is the base64 encoded shared secret. hmac is the name of the key algorithm; valid choices are hmac-md5, hmac-sha1, hmac-sha224, hmac-sha256, hmac-sha384, or hmac-sha512. If hmac is not specified, the default is hmac-md5 or if MD5 was disabled hmac-sha256.

NOTE: Use of the -y option is discouraged because the shared secret is supplied as a command line argument in clear text. This may be visible in the output from ps(1) or in a history file maintained by the user's shell.

INPUT FORMAT

nsupdate reads input from filename or standard input. Each command is supplied on exactly one line of input. Some commands are for administrative purposes. The others are either update instructions or prerequisite checks on the contents of the zone. These checks set conditions that some name or set of resource records (RRset) either exists or is absent from the zone. These conditions must be met if the entire update request is to succeed. Updates will be rejected if the tests for the prerequisite conditions fail.

Every update request consists of zero or more prerequisites and zero or more updates. This allows a suitably authenticated update request to proceed if some specified resource records are present or missing from the zone. A blank input line (or the send command) causes the accumulated commands to be sent as one Dynamic DNS update request to the name server.

The command formats and their meaning are as follows:

server {servername} [port]

Sends all dynamic update requests to the name server servername. When no server statement is provided, nsupdate will send updates to the master server of the correct zone. The MNAME field of that zone's SOA record will identify the master server for that zone. port is the port number on servername where the dynamic update requests get sent. If no port number is specified, the default DNS port number of 53 is used.

local {address} [port]

Sends all dynamic update requests using the local address. When no local statement is provided, nsupdate will send updates using an address and port chosen by the system. port can additionally be used to make requests come from a specific port. If no port number is specified, the system will assign one.

zone {zonename}

Specifies that all updates are to be made to the zone zonename. If no zone statement is provided, nsupdate will attempt determine the correct zone to update based on the rest of the input.

class {classname}

Specify the default class. If no class is specified, the default class is IN.

ttl {seconds}

Specify the default time to live for records to be added. The value none will clear the default ttl.

key [hmac:] {keyname} {secret}

Specifies that all updates are to be TSIG-signed using the keynamesecret pair. If hmac is specified, then it sets the signing algorithm in use; the default is hmac-md5 or if MD5 was disabled hmac-sha256. The key command overrides any key specified on the command line via -y or -k.

gsstsig

Use GSS-TSIG to sign the updated. This is equivalent to specifying -g on the command line.

oldgsstsig

Use the Windows 2000 version of GSS-TSIG to sign the updated. This is equivalent to specifying -o on the command line.

realm {[realm_name]}

When using GSS-TSIG use realm_name rather than the default realm in krb5.conf. If no realm is specified the saved realm is cleared.

check-names {[yes_or_no]}

Turn on or off check-names processing on records to be added. Check-names has no effect on prerequisites or records to be deleted. By default check-names processing is on. If check-names processing fails the record will not be added to the UPDATE message.

[prereq] nxdomain {domain-name}

Requires that no resource record of any type exists with name domain-name.

[prereq] yxdomain {domain-name}

Requires that domain-name exists (has as at least one resource record, of any type).

[prereq] nxrrset {domain-name} [class] {type}

Requires that no resource record exists of the specified type, class and domain-name. If class is omitted, IN (internet) is assumed.

[prereq] yxrrset {domain-name} [class] {type}

This requires that a resource record of the specified type, class and domain-name must exist. If class is omitted, IN (internet) is assumed.

[prereq] yxrrset {domain-name} [class] {type} {data...}

The data from each set of prerequisites of this form sharing a common type, class, and domain-name are combined to form a set of RRs. This set of RRs must exactly match the set of RRs existing in the zone at the given type, class, and domain-name. The data are written in the standard text representation of the resource record's RDATA.

[update] del[ete] {domain-name} [ttl] [class] [type [data...]]

Deletes any resource records named domain-name. If type and data is provided, only matching resource records will be removed. The internet class is assumed if class is not supplied. The ttl is ignored, and is only allowed for compatibility.

[update] add {domain-name} {ttl} [class] {type} {data...}

Adds a new resource record with the specified ttl, class and data.

show

Displays the current message, containing all of the prerequisites and updates specified since the last send.

send

Sends the current message. This is equivalent to entering a blank line.

answer

Displays the answer.

debug

Turn on debugging.

version

Print version number.

help

Print a list of commands.

Lines beginning with a semicolon are comments and are ignored.

EXAMPLES

The examples below show how nsupdate could be used to insert and delete resource records from the example.com zone. Notice that the input in each example contains a trailing blank line so that a group of commands are sent as one dynamic update request to the master name server for example.com.

```
# nsupdate
> update delete oldhost.example.com A
> update add newhost.example.com 86400 A 172.16.1.1
> send
```

Any A records for oldhost.example.com are deleted. And an A record for newhost.example.com with IP address 172.16.1.1 is added. The newly-added record has a 1 day TTL (86400 seconds).

```
# nsupdate
> prereq nxdomain nickname.example.com
> update add nickname.example.com 86400 CNAME somehost.example.com
> send
```

The prerequisite condition gets the name server to check that there are no resource records of any type for nickname.example.com. If there are, the update request fails. If this name does not exist, a CNAME for it is added. This ensures that when the CNAME is added, it cannot conflict with the long-standing rule in RFC 1034 that a name must not exist as any other record type if it exists as a CNAME. (The rule has been updated for DNSSEC in RFC 2535 to allow CNAMEs to have RRSIG, DNSKEY and NSEC records.)

FILES

/etc/resolv.conf

used to identify default name server

/var/run/named/session.key

sets the default TSIG key for use in local-only mode

K{name}.+157.+{random}.key

base-64 encoding of HMAC-MD5 key created by dnssec-keygen(8).

K{name}.+157.+{random}.private

base-64 encoding of HMAC-MD5 key created by dnssec-keygen(8).

SEE ALSO

RFC 2136, RFC 3007, RFC 2104, RFC 2845, RFC 1034, RFC 2535, RFC 2931, named(8), ddns-confgen(8), dnssec-keygen(8).

BUGS

The TSIG key is redundantly stored in two separate files. This is a consequence of nsupdate using the DST library for its cryptographic operations, and may change in future releases.

AUTHOR

Internet Systems Consortium, Inc.

COPYRIGHT