

DNS

- [Introduction](#)
 - [DNS configuration](#)
 - [DNS Cache](#)
 - [DNS Static](#)
- [DNS over HTTPS \(DoH\)](#)
 - [Known compatible/incompatible DoH services](#)
- [Adlist](#)
- [Configuration examples:](#)
 - [URL based adlist:](#)
 - [Locally hosted adlist:](#)

Introduction

Domain Name System (DNS) usually refers to the Phonebook of the Internet. In other words, DNS is a database that links strings (known as hostnames), such as www.mikrotik.com to a specific IP address, such as 159.148.147.196.

A MikroTik router with a DNS feature enabled can be set as a DNS cache for any DNS-compliant client. Moreover, the MikroTik router can be specified as a primary DNS server under its DHCP server settings. When the remote requests are enabled, the MikroTik router responds to TCP and UDP DNS requests on port 53.

When both static and dynamic servers are set, static server entries are preferred, however, it does not indicate that a static server will always be used (for example, previously query was received from a dynamic server, but static was added later, then a dynamic entry will be preferred).



When DNS server *allow-remote-requests* are used make sure that you limit access to your server over TCP and UDP protocol port 53 only for known hosts.

There are several options on how you can manage DNS functionality on your LAN - use public DNS, use the router as a cache, or do not interfere with DNS configuration. Let us take as an example the following setup: Internet service provider (ISP) → Gateway (GW) → Local area network (LAN). The GW is RouterOS based device with the default configuration:

- You do not configure any DNS servers on the "GW" DHCP server network configuration - the device will forward the DNS server IP address configuration received from `ISP` to `LAN` devices;
- You configure DNS servers on the "GW" DHCP server network configuration - the device will give configured DNS servers to `LAN` devices (also " /ip dns set allow-remote-requests=yes" *must* be enabled);
- "dns-none" configured under DNS servers on "GW" DHCP server network configuration - the device will not forward any of the **dynamic** DNS servers to `LAN` devices;

DNS configuration

DNS facility is used to provide domain name resolution for the router itself as well as for the clients connected to it.

Property	Description
allow-remote-requests (<i>yes no</i> ; Default: no)	Specifies whether to allow router usage as a DNS cache for remote clients. Otherwise, only the router itself will use DNS configuration.
cache-max-ttl (<i>time</i> ; Default: 1w)	Maximum time-to-live for cache records. In other words, cache records will expire unconditionally after cache-max-TTL time. Shorter TTLs received from DNS servers are respected.
cache-size (<i>integer</i> [64..4294967295]; Default: 2048)	Specifies the size of the DNS cache in KiB.
max-concurrent-queries (<i>integer</i> ; Default: 100)	Specifies how many concurrent queries are allowed.
max-concurrent-tcp-sessions (<i>integer</i> ; Default: 20)	Specifies how many concurrent TCP sessions are allowed.

max-udp-packet-size (<i>integer [50..65507]; Default: 4096</i>)	Maximum size of allowed UDP packet.
query-server-timeout (<i>time; Default: 2s</i>)	Specifies how long to wait for a query response from a server.
query-total-timeout (<i>time; Default: 10s</i>)	Specifies how long to wait for query response in total. Note that this setting must be configured taking into account "query-server-timeout" and the number of used DNS servers.
servers (<i>list of IPv4/IPv6 addresses ; Default: </i>)	List of DNS server IPv4/IPv6 addresses
cache-used (<i>integer</i>)	Shows the currently used cache size in KiB
dynamic-server (<i>IPv4/IPv6 list</i>)	List of dynamically added DNS servers from different services, for example, DHCP.
doh-max-concurrent-queries (<i>integer; Default: 50</i>)	Specifies how many DoH concurrent queries are allowed.
doh-max-server-connections (<i>integer; Default: 5</i>)	Specifies how many concurrent connections to the DoH server are allowed.
doh-timeout (<i>time; Default: 5s</i>)	Specifies how long to wait for query response from the DoH server.
use-doh-server (<i>string; Default: </i>)	Specified which DoH server must be used for DNS queries. DoH functionality overrides "servers" usage if specified. The server must be specified with an "https://" prefix.
verify-doh-cert (<i>yes no; Default: no</i>)	Specifies whether to validate the DoH server, when one is being used. Will use the "/certificate" list in order to verify server validity.

```
[admin@MikroTik] > ip dns print
      servers:
dynamic-servers: 10.155.0.1
  use-doh-server:
verify-doh-cert: no
doh-max-server-connections: 5
doh-max-concurrent-queries: 50
      doh-timeout: 5s
allow-remote-requests: yes
max-udp-packet-size: 4096
query-server-timeout: 2s
query-total-timeout: 10s
max-concurrent-queries: 100
max-concurrent-tcp-sessions: 20
      cache-size: 2048KiB
    cache-max-ttl: 1d
      cache-used: 48KiB
```

Dynamic DNS servers are obtained from different facilities available in RouterOS, for example, DHCP client, VPN client, IPv6 Router Advertisements, etc.

Servers are processed in a queue order - static servers as an ordered list, dynamic servers as an ordered list. When DNS cache has to send a request to the server, it tries servers one by one until one of them responds. After that this server is used for all types of DNS requests. Same server is used for any types of DNS requests, for example, A and AAAA types. If you use only dynamic servers, then the DNS returned results can change after reboot, because servers can be loaded into IP/DNS settings in a different order due to a different speeds on how they are received from facilities mentioned above.

If at some point the server which was being used becomes unavailable and can not provide DNS answers, then the DNS cache restarts the DNS server lookup process and goes through the list of specified servers once more.

DNS Cache

This menu provides two lists with DNS records stored on the server:

- "ip dns cache": this menu provides a list with cache DNS entries that RouterOS cache can reply with to client requests ;
- "ip dns cache all": This menu provides a complete list with all cached DNS records stored including also, for example, PTR records.

✓ You can empty the DNS cache with the command: "/ip dns cache flush".

DNS Static

The MikroTik RouterOS DNS cache has an additional embedded DNS server feature that allows you to configure multiple types of DNS entries that can be used by the DNS clients using the router as their DNS server. This feature can also be used to provide false DNS information to your network clients. For example, resolving any DNS request for a certain set of domains (or for the whole Internet) to your own page.

```
[admin@MikroTik] /ip dns static add name=www.mikrotik.com address=10.0.0.1
```

The server is also capable of resolving DNS requests based on POSIX basic regular expressions so that multiple requests can be matched with the same entry. In case an entry does not conform with DNS naming standards, it is considered a regular expression. The list is ordered and checked from top to bottom. Regular expressions are checked first, then the plain records.

Use regex to match DNS requests:

```
[admin@MikroTik] /ip dns static add regexp="[*mikrotik*]" address=10.0.0.2
```


If DNS static entries list matches the requested domain name, then the router will assume that this router is responsible for any type of DNS request for the particular name. For example, if there is only an "A" record in the list, but the router receives an "AAAA" request, then it will reply with an "A" record from the static list and will query the upstream server for the "AAAA" record. If a record exists, then the reply will be forwarded, if not, then the router will reply with an "ok" DNS reply without any records in it. If you want to override domain name records from the upstream server with unusable records, then you can, for example, add a static entry for the particular domain name and specify a dummy IPv6 address for it "::ffff".


List all of the configured DNS entries as an ordered list:


```
[admin@MikroTik] /ip/dns/static/print
Columns: NAME, REGEXP, ADDRESS, TTL
# NAME          REGEXP          ADDRESS      TTL
0 www.mikrotik.com      10.0.0.1      1d
1 [*mikrotik*]  10.0.0.2      1d
```

Property	Description
address (IPv4/IPv6)	The address that will be used for "A" or "AAAA" type records.
cname (string)	Alias name for a domain name.
forward-to	The IP address of a domain name server to which a particular DNS request must be forwarded.
mx-exchange (string)	The domain name of the MX server.
name (string)	Domain name.
srv-port (integer; Default: 0)	The TCP or UDP port on which the service is to be found.
srv-target	The canonical hostname of the machine providing the service ends in a dot.
text (string)	Textual information about the domain name.
type (A AAAA CNAME FWD MX NS N XDOMAIN SRV TXT ; Default: A)	Type of the DNS record.
address-list (string)	Name of the Firewall address list to which address must be dynamically added when some request matches the entry. Entry will be removed from the address list when TTL expires.
comment (string)	Comment about the domain name record.
disabled (yes no; Default: yes)	Whether the DNS record is active.
match-subdomain (yes no; Default: no)	Whether the record will match requests for subdomains.

mx-preference (<i>integer</i> , Default: 0)	Preference of the particular MX record.
ns (<i>string</i>)	Name of the authoritative domain name server for the particular record.
regexp (POSIX regex)	Regular expression against which domain names should be verified.
srv-priority (<i>integer</i> , Default: 0)	Priority of the particular SRV record.
srv-weight (<i>integer</i> , Default: 0)	Weight of the particular SRV record.
ttl (<i>time</i> , Default: 24h)	Maximum time-to-live for cached records.


 For each static A and AAAA record, in cache automatically is added a PTR record.

 Regexp is case-sensitive, but DNS requests are not case sensitive, RouterOS converts DNS names to lowercase before matching any static entries. You should write regex only with lowercase letters. Regular expression matching is significantly slower than plain text entries, so it is advised to minimize the number of regular expression rules and optimize the expressions themselves.


 Be careful when you configure regex through mixed user interfaces - CLI and GUI. Adding the entry itself might require escape characters when added from CLI. It is recommended to add an entry and the execute print command in order to verify that regex was not changed during addition.

DNS over HTTPS (DoH)

Starting from RouterOS version v6.47 it is possible to use DNS over HTTPS (DoH). DoH uses HTTPS protocol to send and receive DNS requests for better data integrity. The main goal is to provide privacy by eliminating "man-in-the-middle" attacks (MITM).

 Currently, DoH is not compatible with FWD-type static entries, in order to utilize FWD entries, DoH must not be configured.

Watch our [video about this feature](#).

 It is strongly recommended to import the root CA certificate of the DoH server you have chosen to use for increased security. We strongly suggest not using third-party download links for certificate fetching. Use the Certificate Authority's own website.

There are various ways to find out what root CA certificate is necessary. The easiest way is by using your WEB browser, navigating to the DoH site, and checking the security of the website. You can download the certificate straight from the browser or fetch the certificate from a trusted source.

Download the certificate, upload it to your router and import it:

```
/certificate import file-name=CertificateFileName
```

Configure the DoH server:

```
/ip dns set use-doh-server=DoH_Server_Query_URL verify-doh-cert=yes
```

Note that you need at least one regular DNS server configured for the router to resolve the DoH hostname itself. If you do not have any dynamical or static DNS server configured, add a static DNS entry for the DoH server domain name like this:

```
/ip dns set servers=1.1.1.1
```

If you do not have any dynamical or static DNS server configured, add a static DNS entry for the DoH server domain name like this:

```
/ip dns static add address=IP_Address name=Domain_Name
```



RouterOS prioritizes DoH over the DNS server if both are configured on the device.



If `/certificate/settings/set crt-use` is set to *yes*, RouterOS will check CRL for each certificate in a certificate chain, therefore, an entire certificate chain should be installed into a device - starting from Root CA, intermediate CA (if there are such), and certificate that is used for specific service.

For example, Google DoH, Cloudflare, and OpenDNS full chain contain three certificates, NextDNS has four certificates.

Known compatible/incompatible DoH services

Compatible DoH services:

- Cloudflare
- Google
- NextDNS
- OpenDNS

Incompatible DoH services:

- Mullvad
- Yandex

Adlist

Adlist is an integral component of network-level ad blocking, comprising a curated collection of domain names known for serving advertisements. This feature operates by utilizing Domain Name System (DNS) resolution to intercept requests to these domains. When a client device queries a DNS server for a domain listed on the adlist, the DNS resolution process is altered. Instead of returning the actual IP address of the ad-serving domain, the DNS server responds with the IP address 0.0.0.0. This effectively null-routes the request, as 0.0.0.0 is a non-routable meta-address used to denote an invalid, unknown, or non-applicable target. By redirecting ad-related requests in this manner, the adlist feature ensures that advertisement content is not loaded, enhancing network performance and improving the user experience by reducing unwanted ad traffic.



Before configuring, increase the DNS cache as it's used to store adlist entries. If limit is reached and error in DNS,error topic is printed "*adlist read: max cache size reached*"

Property	Description
url	Used to specify the URL of an adlist.
ssl-verify	Specifies whether to validate the server's SSL certificate when connecting to an online resource. Will use the <code>/certificate</code> list in order to verify server validity.
file	Used to specify a local file path from which to read adlist data
pause	Temporarily pause the use of all adlist.

Configuration examples:

URL based adlist:

```
/ip/dns/adlist add url=https://raw.githubusercontent.com/StevenBlack/hosts/master/hosts ssl-verify=no
```

To see how many domain names are present and matched, you can run:

```
/ip/dns/adlist/print
Flags: X - disabled
0 url="https://raw.githubusercontent.com/StevenBlack/hosts/master/hosts" ssl-verify=no match-count=122 name-
count=164769
```

Locally hosted adlist:

To create your adlist, you can create a Txt file with the domains. Example:

```
0.0.0.0 example1.com
0.0.0.0 eu1.example.com
0.0.0.0 ex.com
0.0.0.0 com.example.com
```



You can create the txt file on your PC, but it is also possible to create it in RouterOS, with following commands

"/file/add name=host.txt", and then you can run "file/edit host.txt contents" after adding entries, press "ctrl o" to save the entries.

To add file to adlist :

```
/ip/dns/adlist/add file=host.txt match-count=0 name-count=4
```



You can verify that file is formatted correctly with "/ip/dns/adlist/print" ,the results will show how many hostnames you have added, the hostname format must match the format given in previous example.