DNS Records

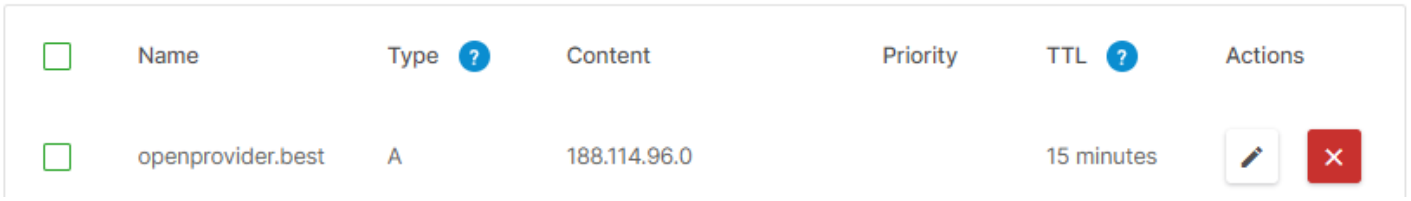
The following DNS record types are available in the DNS Management of Openprovider; below the table, each DNS record is explained in more detail:

Record type	Description	Value	Priority
A	Refers to an IP address	Valid IPv4 address	-
AAAA	Refers to an IPv6 address	Valid IPv6 address	-
CAA	Defines which CAs are allowed to issue SSL certificates	Issuing information	
CNAME	Alias for a subdomain	Valid hostname	-
MX	Defines a mailserver	Existing A-record	Low value means high priority
NS	Defines the nameservers	Valid nameservers	-
SOA	Important information about the DNS zone	This records contains information about: - the master nameserver - e-mailaddress of the zone responsible - some synchronization fields	-
SPF (deprecated)	Authorizes e-mail messages; see TXT record for details.	Sender Policy Framework definition	-
SRV	Allows for discovery of services	Weight, port and target	-
TXT	Allows human-readable text (max. 255 characters)	Free value; also used for definition of SPF, DKIM and DMARC records.	-
TLSA record	Associates a domain with its TLS/SSL certificate for secure authentication	Usage, Selector, Matching type and Certificate association data	
Wildcard	General information about using wildcard DNS records.		
Unsupported DNS record types	General information about DNS records not (yet) supported by Openprovider.		

A Record

An A record is used to point a logical domain name, like openprovider.com, to the IP address of Openprovider's hosting server, "76.76.21.21".

The value of this record must be a Valid IPv4 address, see the following example:

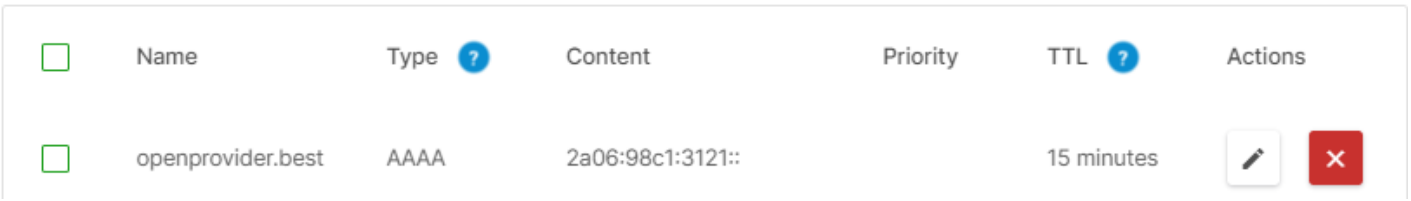


AAAA

This record has the same function as an A record but while the A record points to a IPv4 address the AAAA record points to a IPv6 address for a given host.

The value of this record must be valid IPv6 address.

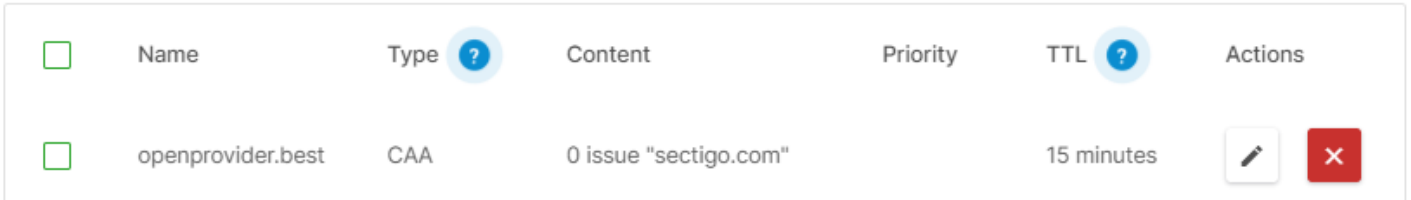
This is what an AAAA record looks like:



CAA

Certification Authority Authorization, or CAA record allows you to declare specific CA's permitted to issue an SSL certificate for your domain.

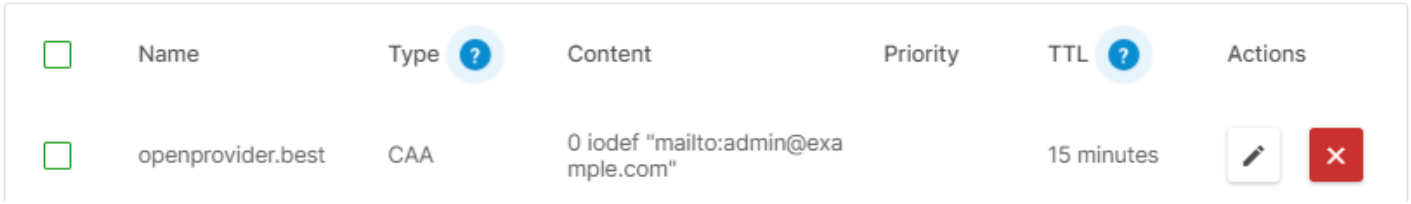
The record’s syntax is fairly simple. Here's an example of a valid CAA record in Openprovider:



Let's take a look at some of the values more closely:

- **www.domain.tld** - your domain name.
- **0** - Issuer critical value flag. According to RFC 6844, this value should always be set to 0 to be compliant with future extensions to CAA.
- **issue** - Permits a specified CA to issue a certificate for the domain. The value has two valid states: **issue**, which allows non-wildcard certificates to be issued, and **issuewild** - permits the issuance of wildcard certificates.
- **"sectigo.com"** - the domain name of the CA you're granting permission to issue a certificate. You may allow a different CA or multiple CAs to issue certificates in the same way.

Moreover, you can create an additional "**incident reporting**" record, which will send iodef format reports of any issued or requested certificates that violate your CAA policy to a specified email or URL. Please refer to the example below:

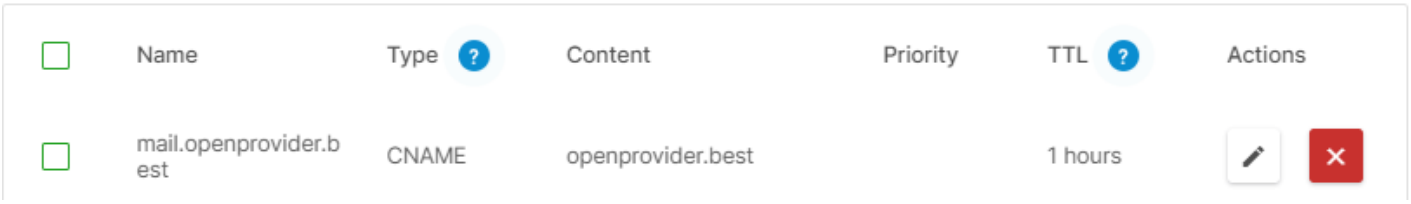


CNAME

CNAME stands for *Canonical Name* and is an alias for a subdomain. With such a record you can point a subdomain (like www) to the DNS entry of the domain.

The value of such a record must be a valid hostname.

Here is an example of a CNAME record in our system:



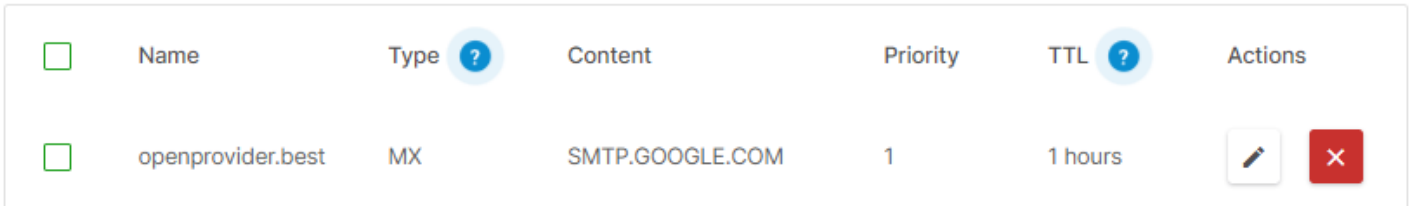
One thing to keep in mind when using CNAME records, a CNAME record gets priority over other records, which means that other records using the same hostname, will not be read. This can cause issues if you have MX or other records using the same hostname as the CNAME record. Learn more here.

Aliases to the apex are not supported at the moment.

MX

A *Mail eXchanger* record (**MX record**) is a type of resource record in the Domain Name System that specifies a mail server responsible for accepting email messages on behalf of a recipient's domain, and a preference value used to prioritize mail delivery if multiple mail servers are available.

Here you can see what an MX record looks like:



Note: MX records with a priority of "NULL" are not supported in our DNS Zones.

NS

Nameserver (NS) records define the authoritative nameservers for the domain. In the Openprovider DNS management, the NS records are created automatically.
It is not possible to assign nameservers for subdomains.

SOA

This records contains important information about the DNS zone. These records are used to determine how your zone propagates to the secondary nameservers. The SOA record includes the following details:

- The primary name server for the domain.
- The responsible party for the domain.
- A timestamp that changes if you update the domain.
- The number of seconds before the zone should be refreshed.

Hereby you can find an example of a SOA Record:

openprovider.ac	SOA	ns1.openprovider.nl dns@openprovider.eu 2016071503	1 day

In the Openprovider DNS management, the SOA record is created automatically. More information about Openprovider SOA record and its expire value can be found here.

SPF

Note that the record type "**SPF**" is deprecated; SPF records should be defined in TXT records. More extensive information about SPF records is available on our special SPF page.

SRV

SRV records are used in Internet Telephony to define where a SIP service may be found. An SRV record typically defines a symbolic name and the transport protocol used as part of the domain name and defines the priority, weight, port, and target for the service in the record content. It is possible to enter **SRV records** in the DNS panel of Openprovider. The general structure of such a record is the following:

- The **name** of the record is the *service* followed by the *protocol*, for example, **_sip._tls**. No domain name is added: Openprovider does this automatically
- The **value** contains the fields for *weight*, *port,* and *target*, each separated by a space. Example: **1 443 sipdir.domain.com**
- The **priority** is entered into the *priority* field
- The **TTL** is entered into the *TTL* field

Some more examples:

Name	Type	Content	Priority	TTL
_service._protocol	SRV	weight port target	priority	ttl
_sip._tls	SRV	1 443 sipdir.online.lync.com	100	86400
_sipfederationtls._tcp	SRV	1 5061 sipfed.online.lync.com	100	86400

TXT

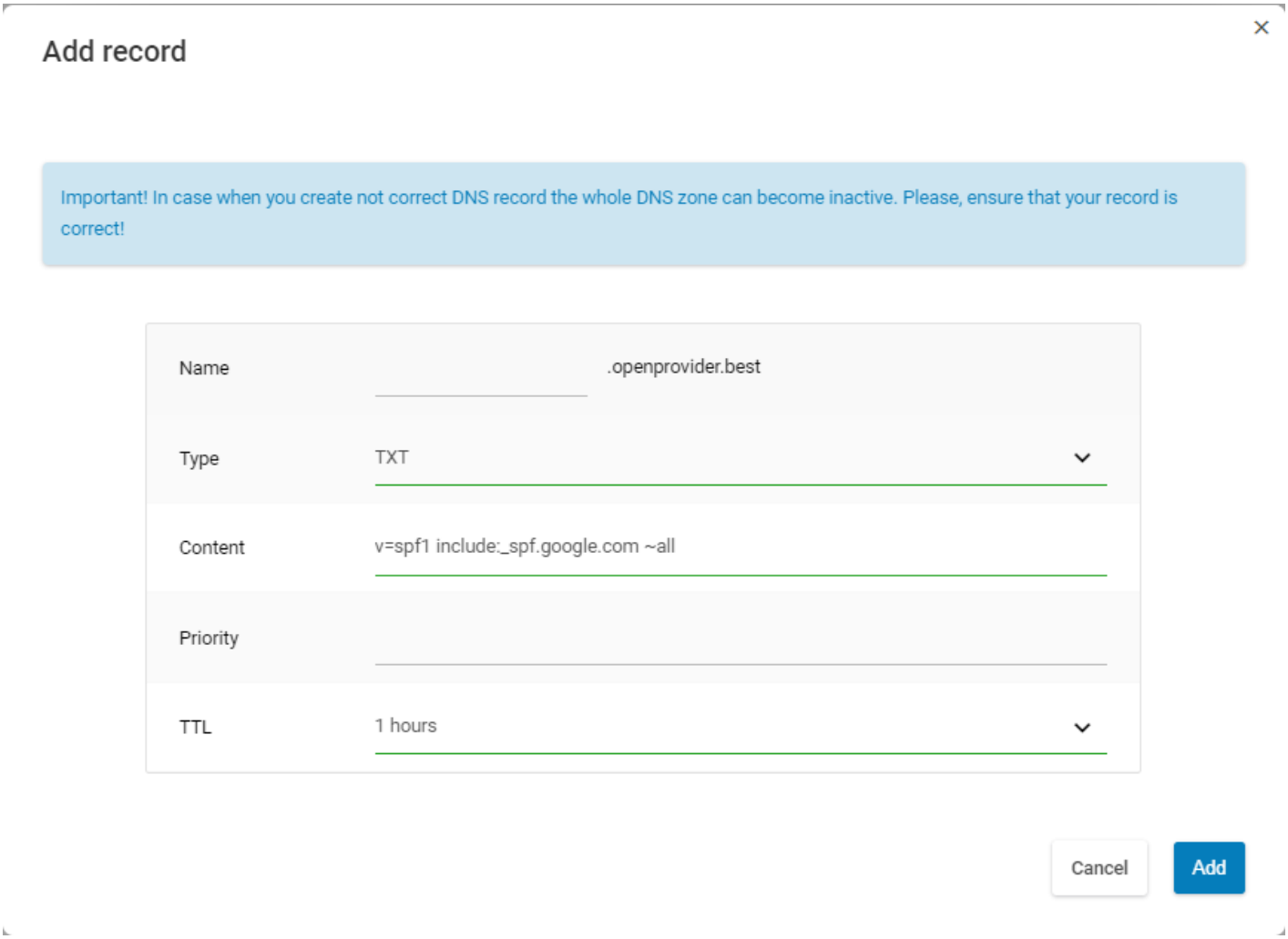
A TXT record (short for text record) provides text information to sources outside your domain. It is a type of resource record in the Domain Name System (DNS) used to provide the ability to associate some arbitrary and unformatted text with a host or other name, such as human-readable information about a server, network, data center, and other accounting information.

Hereby an example:

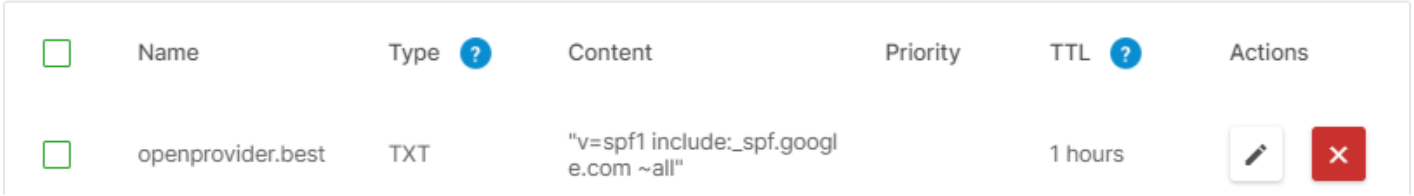
openprovider.nl	TXT	google-site-verification=lSg4lF4ftYLoWGFPAKcu04_x6Wzk07-fFhSi1Re-AyQ	15 minutes

TXT records are often used in e-mail security, for storing SPF, DKIM, and/or DMARC DNS records:

Below you find an example of how to create an SPF record in the root zone of a domain.



The (SPF) TXT record will look in the DNS zone as:



It is **required** to surround TXT records with quotes, otherwise, the entire zone can have issues to resolve correctly. It's even required when creating a TXT record to define SPF data; if not surrounded by quotes, the behavior may be different from what you expect.

If you want to add a large TXT record you may use quotes to split it into smaller pieces within one record like the one below:

```
name IN TXT ( “v=DKIM1; g=*; k=rsa; “
“p=MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEA1Z4F”
“JEMHjJDuBmt25zvYFVejlARZGt1L8f0s1+rLxIPYkfCogQi+Y8”
“oLEg9vvEKnLx9aogZzuNt6j4Sty3LgXxaIwHnMqk0LldbA/mh3”
“wLZb16Wc6btXHON0o3uDipxqGK2iRLTvcgAnNDegseOS+i0aJE”
“nNSl663ywRBp/QKezhUC7cnbqR/H8dz8pEOjeawNN3nexdHGsk”
“+RaafYvCFvU+70CQORcsk+mxb74SwGT2CGHWxVywQA9yrV+sYk”
“JpxaufZLo6xp0Z7RZmbf1eGlCAdhkEy+KYQpQkw2Cdl7iKIK4+”
“17gr+XZOrfFLJ5IwpVK/a19m3BLxADf0Kh3oZwIDAQAB” )
```

Wildcards

This section does not describe a special type of record but covers the concept of a wildcard ("*", a star or asterisk) in a DNS record, like

```
*.example.com    A    1.2.3.4
```

The above example says that any subdomain under example.com (*www.example.com*, *mail.example.com*, *whatever.example.com*) should be treated as an A record to the IP address 1.2.3.4. This is useful to quickly catch all possible subdomains to one specific IP address.

There is one important exception when using wildcards: the wildcard will *not* match if another record exists with the same subdomain. For example, if there is a record *default.**_domainkey*** *.example.com*, the wildcard **.example.com* will not match ***_domainkey****.example.com* anymore.

All details about wildcards in a DNS record can be found in section 4.3.3 of RFC1034.

TLSA records

A TLSA record is a DNS record (type TLSA) that binds a TLS certificate or public key to a specific service on a specific host and port, usually protected by DNSSEC. TLSA record is supported by Openprovider DNS service, but **not** by Sectigo Premium DNS.

To add a TLSA record, you need:

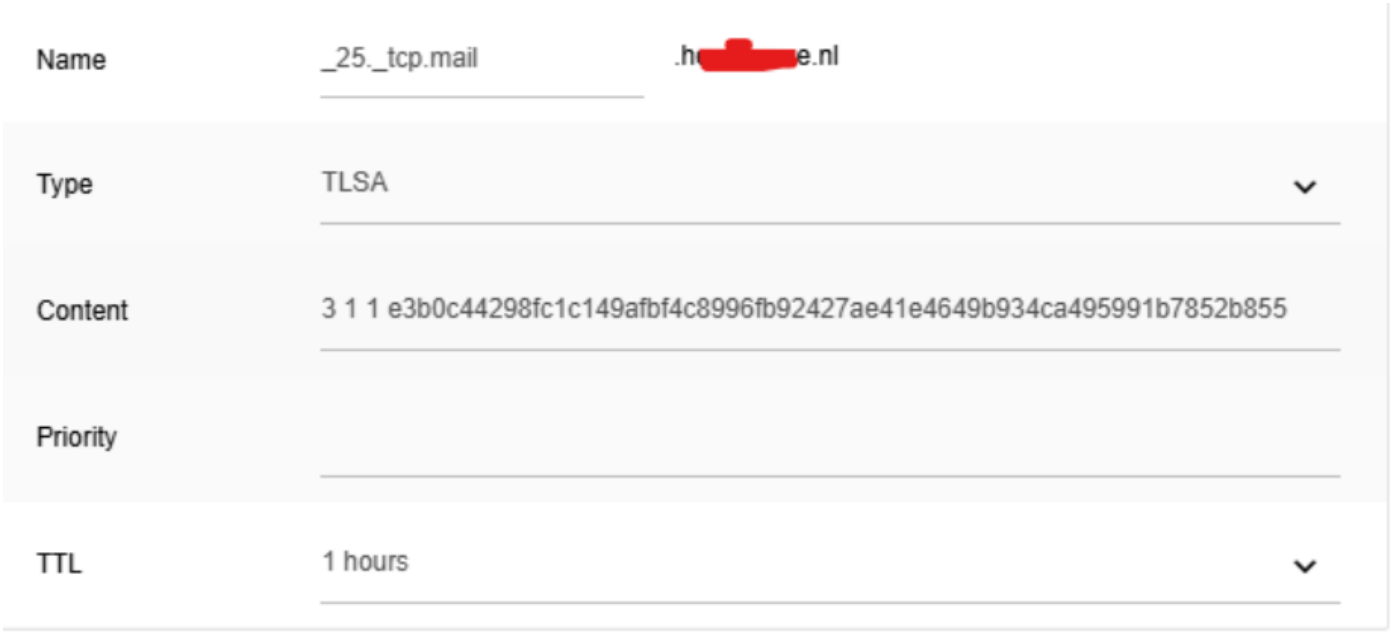
Name: Identifies the service. The format is always `_port._protocol.hostname`

Type: TLSA

Content: Made up of 4 parts.

- Usage - DANE end-entity
- Selector - Public key (SubjectPublicKeyInfo) of the certificate (DER encoded)
- Matching type - How the TLSA record stores the certificate (eg : SHA-256)
- Certificate association data - Hash of your certificate’s public key

Please refer the screen-shot below for ane example:



In the above example:

port = 25
protocol = tcp
hostname = mail.h*****e.nl (domain name is auto filled)
Type = TLSA
Usage = 3
Selector = 1
Matching type =1
Certificate association data =
e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855

Note: The above are example values. When adding a TLSA record for your domains, please ensure to use real values specific to the host, port and certificate.

DNS Round-robin

By managing the Domain Name System’s (DNS) answers to address requests from client computers according to an acceptable statistical model, round-robin DNS is a technique of load distribution, load balancing, or fault-tolerance delivering multiple, redundant Internet Protocol service hosts, e.g., Web server, FTP servers.

Drawbacks :

- Although simple to set up, round-robin DNS has several disadvantages, including record caching in the DNS hierarchy and client-side address caching and reuse, which can be difficult to manage when used together.
- For service availability, round-robin DNS should not be depended on entirely.
- If a service at one of the addresses on the list goes down, the DNS will continue to give out that address, and clients will try to contact the broken service.
- **Openprovider does not actively support DNS Round-robin.**

Unsupported record types

This section contains basic information about record types that are not supported in the Openprovider DNS management. This section is included for reference.

ALIAS/ANAME

An ANAME record is like a CNAME record but at the root level. An ANAME record bypasses the problems with CNAME records at the root level, so it can be used to redirect a domain to another domain. However, ANAME records are still in draft specification and as a result, are not supported by Openprovider.

NAPTR

NAPTR records are most commonly used for applications in Internet telephony, for example, in the mapping of servers and user addresses in the Session Initiation Protocol (SIP). The combination of NAPTR records with Service Records (SRV) allows the chaining of multiple records to form complex rewrite rules that produce new domain labels or uniform resource identifiers (URIs). For this moment Openprovider does not support NAPTR records. Because of the different structure of NAPTR records it's **for this moment** not supported by Openprovider.

PTR

PTR records are used for the Reverse DNS (Domain Name System) lookup. Using the IP address you can get the associated domain/hostname. An **A record** should exist for every **PTR record**. The usage of a reverse DNS setup for a mail server is a good solution. A hosting provider can add the record to the IP block.

You can't create a PTR record for the IP address in your domain name DNS zone. The PTR record needs to be created in the rDNS zone by the owner of the netblock that encompasses your IP address (i.e. your ISP).

So, this **cannot** be added to your DNS zone at Openprovider.

Use of '@' symbol

Some DNS providers allow the use of the `@` symbol to represent the root domain (e.g., example.com).

However, our system does not support this notation. Instead, please leave the "Name" field blank when editing a DNS zone.

Revision #2
Created 2026-08-07 14:27:26 UTC by Sarath
Updated 2026-08-07 15:45:03 UTC by Sarath