Setting server-wide DNS templates

This page explains how to set server-wide DNS templates that can be used by any new DNS Zone added to the system.

To manage DNS templates, click the DNS templates button in the **System Templates** area. The server global DNS templates are available to all clients that have not set up their own DNS templates.

In the **Server Global DNS Templates** management page, you can add new server-wide DNS templates, edit and delete existing templates.

Creating a new sever global DNS template

To create a global DNS Template, follow the steps below:

1. Enter a name in the **Template name** text box.
2. From the drop-down list, select the template type:
    - Forward for forward DNS zones
    - Reverse for reverse IPV4 DNS zones
    - Reverse IPV6 for reverse IPV6 zones
    - E.164 for E.164 DNS zones
3. From the drop-down list, select the template's availability:
    - Owned template
    - Wide template, which can be used by all clients.
4. Click the **OK** button.

Wherever you want the domain name to be automatically replaced by the name of the newly created domain, enter **[domain]** in the **domain name** field.

In order to have an IP address automatically replaced, use the **[ip]** tag.

A new page will open where you need to define the DNS records and Template IPs. This is where you can view the list of DNS records included in the template. The following details are available:

- **Host:** The host name or IP address of every DNS record;
- **Record type**: The type of the DNS record;
- **Value**: Depending on the record type, this field displays an IP address, an alias, a name server, a host name, or a text;
- **M**: If you click the **Modify** icon, you can edit the details of the corresponding DNS record.

To add a new DNS record to the server global DNS template, select the **Record type** in the **New DNS Record** area and then click **Add**.

To remove a DNS record from the template, select the corresponding checkbox and click the Remove Selected link. You can delete several DNS records at the same time. DNS Manager will ask you to confirm this operation before permanently deleting the records.

Adding records to a DNS template

To add a record to a DNS template, click the **Add DNS records** button in the template's management page. In the **Add new record** page that opens, select the record type and configure it taking into account the explanations below.

DNS Manager also accepts internationalized domain names (IDN) - Internet domain names that contain non-ASCII characters.

The following types of DNS records are available:

Record Type	Details

IP Address (A)	Maps a hostname to a 32-bit IPv4 address. Type A rules have the following format: `hostname. IN A XXX.XXX.XXX.XXX` where: • `XXX.XXX.XXX.XXX` is the IP address for the hostname. • `hostname.` is the zone name or one of its subdomains. Examples: ``` domain.com. IN A 1.2.3.4 subdomain.domain.com. IN A 1.2.3.4 domain.com. IN A [IP] ``` Click **here** for more info on this type of record.
AAAA Record (AAAA)	Maps a hostname to a 128-bit IPv6 address. AAAA rules have the following format: `hostname. IN AAAA:AAAA:AAAA:AAAA:AAAA:AAAA:AAAA:AAAA` where: • AAA:AAAA:AAAA:AAAA:AAAA:AAAA:AAAA:AAAA is the IPv6 address for the hostname. • hostname. is the zone name or one of its subdomains. Examples: ``` domain.com. IN AAAA abcd:1234:ffff:0:12:3:ab1:aa subdomain.domain.com. IN AAAA abcd:1234:ffff:0:12:3:ab1:aa ``` Click **here** for more info on this type of record.
Certification Authority Authorization (CAA)	Specifies one or more Certification Authorities (CAs) authorized to issue certificates for that domain. CAA rules have the following format: `hostname. IN CAA flags tag value` where: • flags is an unsigned integer between 0 and 255. It is currently used to represent the *critical* flag, that has a specific meaning per RFC. • tag is an ASCII string that represents the identifier of the property represented by the record • value is the value associated with the tag. The CAA record consists of a flags byte and a tag-value pair referred to as a 'property'. Multiple properties may be associated with the same domain name by publishing multiple CAA records at that domain name. There are 3 available tags: • `issue`: explicity authorizes a single certificate authority to issue a certificate (any type) for the hostname. • `issuewild`: explicity authorizes a single certificate authority to issue a wildcard certificate (and only wildcard) for the hostname. • `iodef`: specifies a URL to which a certificate authority may report policy violations. Examples: ``` example.com. IN CAA 0 issue ";" example.com. CAA 0 issue "letsencrypt.org" example.com. CAA 0 issuewild "comodoca.com" example.com. CAA 0 iodef "mailto:example@example.com" ```

Alias for record (CNAME)	Canonical name record is an alias (or nickname) of one name to another. The A record to which the alias points can be either local or remote - on a foreign name server. This is useful when running multiple services (like an FTP and a webserver) from a single IP address. Each service can then have its own entry in DNS (like ftp.example.com. and www.example.com.). It is also used when running multiple HTTP servers, with different names, on the same physical host. CNAME rules have the following format: `hostname. IN CNAME servername.` where: • `hostname.` is the zone name or one of its subdomains • `servername.` is a fully qualified domain name (FQDN) either inside or outside the zone. Examples: ``` ftp.domain.com. IN CNAME inside.domain.com. ftp1.domain.com IN CNAME outside.zone.com. k1._domainkey.domain.com IN CNAME dkim.zone.com. ``` RFC 1034 states: “If a CNAME record is present at a node, no other data should be present; this ensures that the data for a canonical name and its aliases cannot be different.” In order for these requirements to be met in DNS Manager, the value specified in the **Zone alias name** field of the CNAME record cannot be set for the **DNS Zone name** filed in NS, A, AAAA, SRV, CNAME and TXT records or for the **Zone email** field in an MX record. Click **here** for more info on this type of record.
Nameserver (NS)	Maps a domain name to a list of DNS servers authoritative for that domain. Delegations depend on NS records. NS rules have the following format: `hostname. IN NS servername.` where: • `hostname.` is the zone name or one of its subdomains • `servername.` is a domain name which specifies an authoritative host for the specified hostname. Examples: ``` domain.com. IN NS ns1.example.com. domain.com. IN NS ns2.example.com. ``` The NS records of $ORIGIN are displayed in bold characters. DNS Manager allows to choose in the interface which is the primary nameserver on a zone (required for some local TLDs). In order to set up an NS record as primary check Make primary when you add /edit the desired NS record. For BIND to take a DNS zone into consideration, at least one NS record must be defined for $ORIGIN in the respective zone. For best practice, it is recommended to have at least two NS records defined for each public domain. Click **here** for more info on this type of record.

Mail exchanger (MX)	Maps a domain name to a list of mail exchange servers for that domain. MX rules have the following format: `hostname. IN MX preference servername.` where: • `hostname.` is the zone name or one of its subdomains • `preference` indicates the hostname's priority. The lower the preference, the higher the priority. This parameter accepts values between 0 and 50. • servername. is a fully qualified domain name (FQDN) inside the zone Examples: ``` mail.domain.com. IN MX 10 domain.com. webmail.domain.com. IN MX 5 domain.com. ``` Click **here** for more info on this type of record.
Text record (TXT)	Allows an administrator to insert arbitrary text into a DNS record. This has been used to implement new functions with DNS support without allocating new record types. For example, this record is used to implement the Sender Policy Framework and DomainKeys specifications. TXT rules have the following format: `hostname. IN TXT "Text information"` where: • hostname. is the zone name or one of its subdomains • "Text information" can be any type of string including strings generated by SPF Rules Examples: ``` domain.com. IN TXT "k=rsa; p=MEwwDQYerwqEWwE" subdomain.domain.com. IN TXT "this is a test" ``` Click **here** for more info on this type of record.
Service Record (SRV)	Specifies the location of the server(s) for a specific protocol and domain. SRV rules have the following format: `_Service._Protocol.Hostname. IN SRV TTL Priority Weight Port Target` where: • `Service` is the symbolic name of the desired service. You can find a list of the available services at http://www.dns-sd.org/ServiceTypes.html. • `Protocol` is the protocol of the desired service. This is usually TCP or UDP, but 4PSA DNS Manager supports all the protocols listed here http://www.iana.org/assignments/protocol-numbers. • `Hostname.` is the domain name for which the record is valid. • `TTL` is the standard DNS time to live field. If there is no TTL specified for the record, the TTL value for the zone will be employed. • `Priority` is the priority of the target host. The lower the value, the higher the priority level. • `Weight` indicates a relative weight between records with the same priority. • `Port` is the port on which the service is to be found. • `Target` is the domain name of the target host. The `Target` parameter can not be an alias (CNAME). When `Target` is set to `.` the service is unavailable. Examples: ``` _service._tcp.domain.com. IN SRV 0 1 9 subdomain.domain.com. *._tcp.domain.com. IN SRV 0 0 0 . ``` on TCP protocol Click **here** for more info on this type of record.

NAPTR record (NAPTR)	Naming Authority Pointers. NAPTR rules have the following format: `order preference services flag regexp replacement` where: • order indicates the order in which records are to be processed when a query returns multiple NAPTR records • preference indicates the processing order for multiple records with identical order • services indicate the resolution protocol and resolution services employed when applying a rewrite according to the regexp or replacement field • flag is a modifier that affects the next DNS lookup • regexp is the primary field used for rewrite rules • replacement is a secondary field used for rewrite rules Examples: ``` domain.com. IN NAPTR 100 10 "u" "sip+E2U" "!^.*$! sip:information@foo.se!i" . subdomain.domain.com. IN NAPTR 102 10 "u" "smtp+E2U" "!^.*$! mailto:information@foo.se!i" . ```

Record Type	Details
Nameserver (NS)	Specifies a host which should be authoritative for the specified class. For class C reverse zones, 4PSA DNS Manager accepts NS records for $ORIGIN and supports classless delegation records, as described in RFC 2317, chapter 4. The NS records of $ORIGIN are displayed in bold characters. DNS Manager allows to choose in the interface which is the primary nameserver on a zone (required for some local TLDs). In order to set up an NS record as primary check Make primary when you add /edit the desired NS record. For BIND to take a DNS zone into consideration, at least one NS record must be defined for $ORIGIN in the respective zone. For best practice, it is recommended to have at least two NS records defined for each public domain. Class A and B zones support NS records for $ORIGIN and inferior class zones and do not support classless delegation records. For class C reverse zones, 4PSA DNS Manager automatically generates CNAME records that correspond to the NS records created for classless delegation records. If the Automatically generate CNAME records for delegated subnets checkbox is selected, then the CNAME records will be automatically generated. This checkbox is available only for NS records with a subnet mask lower than 24 (having a numeric value higher than 24). NS rules have the following format: `ip_part.host_ip_addr.in-addr.arpa. IN NS servername.` where: • `host_ip_addr.in-addr.arpa.` is the zone name. • `ip_part` is the IP section that completes the IP address when prepended to host_ip_addr (for class A, B and D zones, and for class C $ORIGIN NS). • for classless delegation records, `ip_part` is the IP section that completes the IP address when prepended to host_ip_addr including the subnet mask. • `servername.` is a domain name which specifies an authoritative host for the specified zone. Examples: ``` 1.2.3.in-addr.arpa. IN NS ns2.server.com. 1.2.3.in-addr.arpa. IN NS ns3.server.com. 0/29.1.2.3.in-addr.arpa. IN NS example.com. ```

Reverse record (PTR)	Maps an IPv4 address to the canonical name for that host. Setting up a PTR record for a hostname in the in-addr.arpa. domain that corresponds to an IP address implements reverse DNS lookup for that address. PTR rules have the following format: `IPaddress IN PTR hostname.` where: • `IPaddress` is the IPv4 address in the IN-ADDR.ARPA. domain • `hostname.` is the corresponding location in the domain name space Examples: `5.1.2.3.in-addr.arpa. IN PTR test.com.` Click **here** for more info on this type of record.
Alias for record (CNAME)	A canonical name record is an alias of one name to another. According to RFC 2317, CNAME records are only supported in C class reverse zones. CNAME rules have the following format: `ip_part.network.host_ip_addr.in-addr.arpa. IN CNAME ip_part.host_ip_addr.in-addr.arpa.` where: • `ip_part` is the IP section that completes the IP address when prepended to host_ip_addr • `network` is the subnet mask • `host_ip_addr.in-addr.arpa.` is the zone name Examples: `0.1.2.3.in-addr.arpa. IN CNAME 0.0/29.1.2.3.in-addr.arpa.` `1.1.2.3.in-addr.arpa. IN CNAME 1.0/29.1.2.3.in-addr.arpa.` `...` `7.1.2.3.in-addr.arpa. IN CNAME 7.0/29.1.2.3.in-addr.arpa.` Click **here** for more info on this type of record.
Text record (TXT)	Allows an administrator to insert arbitrary text into a DNS record. This has been used to implement new functions with DNS support without allocating new record types. For example, this record is used to implement the Sender Policy Framework and DomainKeys specifications. TXT rules have the following format: `ip_part.host_ip_addr.in-addr.arpa. IN TXT "Text information"` where: • `ip_part` is the IP section that completes the IP address when prepended to host_ip_addr • `host_ip_addr.in-addr.arpa.` is the zone name • `"Text information"` can be any type of string Examples: `4.1.2.3.in-addr.arpa. IN TXT "This is a test"` Click **here** for more info on this type of record.
Record Type	**Details**

Nameserver (NS)	Specifies a host which should be authoritative for the chosen class. The NS records can be defined only for $ORIGIN. The NS rules have the following format: `ipv6_part.host_ipv6_addr.IP6.ARPA. IN NS servername.` where: • `ipv6_part` is the IP section that completes the IP address when prepended to host_ipv6_addr. • `host_ipv6_addr.IP6.ARPA.` is the zone name. • `servername.` is a domain name which specifies an authoritative host for the defined zone. For example: `1.0.2.3.4.5.6.7.8.9.0.1.2.3.4.5.6.7.8.9.0.A.B.C.D.E.F.IP6.ARPA. IN NS example.com.` `5.5.1.3.2.1.0.2.3.4.5.6.7.8.9.0.1.2.3.4.5.6.7.8.9.0.A.B.C.D.E.F.IP6.ARPA. IN NS example.com.` For more information about this record type, see RFC4291.
Reverse record (PTR)	This record type maps an IPv6 address to the canonical name for that host. Setting up a PTR record for a hostname in the IP6.ARPA. domain that corresponds to an IPv6 address implements reverse DNS lookup for that address. The PTR rules have the following format: IPv6_address IN PTR hostname. where: • `IPv6_address` is the IPv6 address in the IP6.ARPA. domain • `hostname.` is the corresponding location in the domain name space For example: `8.b.d.0.1.0.0.2.IP6.ARPA. IN PTR test.com.` `1.1.1.1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.8.B.D.0.1.0.0.2.IP6.ARPA. IN PTR test.com.` `*.1.1.1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.8.B.D.0.1.0.0.2.IP6.ARPA. IN PTR test.com.` For more information about this record type, see RFC4291.
Record Type	**Details**
Nameserver (NS)	Maps a domain name to a list of DNS servers authoritative for that domain. Delegations depend on NS records. NS rules have the following format: `hostname. IN NS servername.` where: • hostname. is the zone name or one of its subdomains • servername. is a domain name which specifies an authoritative host for the specified hostname. Examples: `1.2.e164.arpa. IN NS ns1.example.com.` `1.2.e164.arpa. IN NS ns2.example.com.` `5.1.2.e164.arpa. IN NS ns1.example.com.` The NS records of $ORIGIN are displayed in bold characters. DNS Manager allows to choose in the interface which is the primary nameserver on a zone (required for some local TLDs). In order to set up an NS record as primary check Make primary when you add /edit the desired NS record. For BIND to take a DNS zone into consideration, at least one NS record must be defined for $ORIGIN in the respective zone. For best practice, it is recommended to have at least two NS records defined for each public domain. Click **here** for more info on this type of record.

NAPTR record (NAPTR)	Naming Authority Pointers. NAPTR rules have the following format: `order preference services flag regexp replacement` where: • order indicates the order in which records are to be processed when a query returns multiple NAPTR records • preference indicates the processing order for multiple records with identical order • services indicate the resolution protocol and resolution services employed when applying a rewrite according to the regexp or replacement field • flag is a modifier that affects the next DNS lookup • regexp is the primary field used for rewrite rules • replacement is a secondary field used for rewrite rules Examples: `1.2.e164.arpa. IN NAPTR 100 10 "u" "sip+E2U" "!^.*$! sip:information@foo.se!i" .` `1.2.e164.arpa. IN NAPTR 102 10 "u" "smtp+E2U" "!^.*$! mailto:information@foo.se!i" .` Click **here** for more info on this type of record.

Managing a Template's IPs

To access the **Template's IP Management** page, select the template in the list and then click the **Template IPs** button. This page displays the template's IPs and lets you add new IPs.

The existing IPS are listed with the following details:

- **T**: Refers to the IP address type. It can be **master** or **allow transfer**.
- **IP address**: If the IPs list is too long, you may use the Search to find a specific IP more quickly.
- **Search**: Enter the criteria in the text box and click the Search button.

To add an IP to the template, follow the steps below:

1. Go to the **Add Template IPs** area and enter the IP in the appropriate field.
2. Select one of the following options:
    - **Add the following master IP**: for master IPs assigned to slave zones. You can add multiple IPs by pressing the button.
    - **Add the following allow transfer IP to master zones**: for allow transfer IPs assigned to master zones. You can add multiple IPs by pressing the button.
3. Press **OK** when you're done.

You can use both IPv4 and IPv6 addresses for transfer.

Setting the template's availability

To set the template's availability, you have the following options:

- **Owned templates** (icon), which cannot be used by other clients.
- **Wide templates** (icon), which can also be used by other clients.

To switch from owned to wide templates, press their corresponding icon.