



Name Server (NS) Records in Avi DNS

Avi Technical Reference (v18.1)

Copyright © 2018

Name Server (NS) Records in Avi DNS

[view online](#)

As of release 17.1.2, Avi DNS supports name server (NS) records as part of the static DNS records (Avi GSLB does not support NS records).

A DNS Name Server (NS) record specifies the domain name of the name server servicing a particular domain. For example, an NS record with a time-to-live (TTL) of 1100 seconds, and for the `com` domain serviced by the name server `a.gtld-servers.net`, would be defined as below:

```
com. 1100 IN NS a.gtld-servers.net.
```

For the NS records, the Avi DNS can behave both as an authoritative server as well as a non-authoritative server. An NS record can be configured on Avi DNS for two modes of the DNS domain, delegated and non-delegated.

Delegated Domains or Zone Cuts

If the desired behavior is to have the Avi DNS act as a non-authoritative server for the domain of the NS record, then the NS record must be configured with the option of delegation set to `True`. The delegation indicates that the domain name for the NS record is delegated to another authoritative name server. That is, the domain name for the NS record is a zone cut, with the domains within this domain being owned by another name server. For example, `xyz.foo.com` is a zone cut for the domain `foo.com`, whereby domains within `xyz.foo.com` (aka `*.xyz.foo.com`) is managed by another name server than the one managing `xyz.foo.com`.

When a domain name is configured to be a zone cut, it is mandatory to configure the IP address of the corresponding name servers in the NS record associated with this domain name. The IP address is mandatory when the name server's domain can result in a recursive DNS query to this domain name:

1. DNS query for NS record for domain: `foo.com`
2. Response contains name server: `ns1.avi.com`
3. Since domain name `ns1.foo.com` must be resolved, it results in an A query for `ns1.foo.com`, resulting in a response with NS record for `ns1.foo.com` (but no A record, thereby leading to NS request for `ns1.foo.com` again and again), leading to an indefinite recursion.

To avoid this recursion, the FQDN for the name server in the NS record is also configured with the A record (the IP address). This A record is called the glue record for the NS record. Refer to the below example.

```
static_dns_records {
    fqdn: "avi.com"
    type: DNS_RECORD_NS
    delegated: true
    wildcard_match: true
    ttl: 1100
    ns {
        nsname: "ns1.avi.com"
        ip_address {
            addr: "192.5.6.30"
        }
    }
}
```

```

        type: V4
    }
}
ns {
    nsname: "ns2.avi.net"
}
}

```

The above configuration for a static NS record is for the domain `?avi.com?`, which has NS records pointing to two name servers. Thus, any DNS query with the subdomain `?avi.com?` that comes to this server will be served the NS records in the AUTHORITY SECTION of the DNS response. This response will cause a recursive name resolver to direct the DNS query to either of the two name servers. Thus, a query for A record for `www.avi.com` will get a response of two NS records in the DNS response along with the glue record for `ns1.avi.com` in the ADDITIONAL SECTION of the response. The recursive resolver will then send the same DNS query to the name server `ns1.avi.com` or `ns2.avi.net`.

NOTE: If the desired behavior is to catch all queries for domain names containing `?avi.com?` as a subdomain, then you should set the wildcard match field to `True`, otherwise only queries for `?avi.com?` will get directed to the name servers. When wildcard matching is set for `?avi.com?` any query of the form `.avi.com` (e.g. `foo.avi.com`, `abc.xyz.avi.com`, etc.) will get directed to the configured name servers. When wildcard matching is not set, then query to domains of the form `.avi.com` may or may not succeed depending on whether these FQDNs are configured or not in the DNS VS.

Shown below is a `dig` request for

- a record (A, AAAA, CNAME, SRV, etc.)
- to a domain containing the delegated domain `?com?` as a subdomain,
- with wildcard matching set, 13 name servers (from `a.gtld-servers.net` to `m.gtld-servers.net`), and
- corresponding glue records.

The requested domain `sales.avi.com` has `com` as a subdomain.

```

; <<>> DiG 9.10.3-P4-Ubuntu <<>> @10.90.91.62 sales.avi.com A
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 57472
;; flags: qr aa rd; QUERY: 1, ANSWER: 0, AUTHORITY: 13, ADDITIONAL: 13
;; WARNING: recursion requested but not available

;; QUESTION SECTION:
;sales.avi.com.      IN      A

;; AUTHORITY SECTION:
com.                1100   IN      NS      a.gtld-servers.net.
com.                1100   IN      NS      b.gtld-servers.net.
com.                1100   IN      NS      c.gtld-servers.net.
com.                1100   IN      NS      d.gtld-servers.net.
com.                1100   IN      NS      e.gtld-servers.net.
com.                1100   IN      NS      f.gtld-servers.net.
com.                1100   IN      NS      g.gtld-servers.net.

```

```

com.          1100   IN     NS     h.gtld-servers.net.
com.          1100   IN     NS     i.gtld-servers.net.
com.          1100   IN     NS     j.gtld-servers.net.
com.          1100   IN     NS     k.gtld-servers.net.
com.          1100   IN     NS     l.gtld-servers.net.
com.          1100   IN     NS     m.gtld-servers.net.

;; ADDITIONAL SECTION:
a.gtld-servers.net. 1100   IN     A      192.5.6.30
b.gtld-servers.net. 1100   IN     A      192.33.14.30
c.gtld-servers.net. 1100   IN     A      192.26.92.30
d.gtld-servers.net. 1100   IN     A      192.31.80.30
e.gtld-servers.net. 1100   IN     A      192.12.94.30
f.gtld-servers.net. 1100   IN     A      192.35.51.30
g.gtld-servers.net. 1100   IN     A      192.42.93.30
h.gtld-servers.net. 1100   IN     A      192.54.112.30
i.gtld-servers.net. 1100   IN     A      192.43.172.30
j.gtld-servers.net. 1100   IN     A      192.48.79.30
k.gtld-servers.net. 1100   IN     A      192.52.178.30
l.gtld-servers.net. 1100   IN     A      192.41.162.30
m.gtld-servers.net. 1100   IN     A      192.42.93.30

;; Query time: 0 msec
;; SERVER: 10.90.91.62#53(10.90.91.62)
;; WHEN: Fri May 26 14:55:20 PDT 2017
;; MSG SIZE rcvd: 469

```

Non-delegated Domains

If the desired behavior is to have the Avi DNS act as an authoritative server for the NS record, then the NS record must not be configured with the option of delegation set to `True`. The delegation indicates that the domain name for the NS record is delegated to another authoritative name server. Refer to the below example.

```

static_dns_records {
    fqdn: "google.com"
    type: DNS_RECORD_NS
    wildcard_match: true
    ttl: 1100
    ns {
        nsname: "ns1.google.com"
        ip_address {
            addr: "192.5.6.30"
            type: V4
        }
    }
    ns {
        nsname: "ns2.google.com"
        ip_address {
            addr: "192.33.14.30"
            type: V4
        }
    }
}

```

```

    }
  }
  ns {
    nsname: "ns3.google.com"
    ip_address {
      addr: "192.26.92.30"
      type: V4
    }
  }
  ns {
    nsname: "ns4.google.com"
    ip_address {
      addr: "192.31.80.30"
      type: V4
    }
  }
}
}
}

```

The above configuration for a static NS record is for the domain ?google.com,? which has NS records pointing to four name servers. For these NS records, Avi DNS is authoritative name server. Thus, any DNS NS query with the domain ?google.com? that comes to this server will be served the NS records in the ANSWER SECTION of the DNS response and the A records for the name server IP addresses are served in the ADDITIONAL SECTION.

NOTE: If the wildcard match field is set to *True* but the domain is not delegated, then any DNS query which matches the non-delegated domain will get an NXDOMAIN response.

Shown below is a dig query request of

- any type (A, NS, CNAME, SRV, etc.)
- to the domain ?google.com,?
- with wildcard matching on, and
- 4 name servers (from ns1.google.com to ns4.google.com).

```

; <<>> DiG 9.10.3-P4-Ubuntu <<>> @10.90.91.63 google.com A
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->HEADER<<- opcode: QUERY, status: NOERROR, id: 39235
;; flags: qr aa rd; QUERY: 1, ANSWER: 4, AUTHORITY: 0, ADDITIONAL: 4
;; WARNING: recursion requested but not available

;; QUESTION SECTION:
;google.com.                IN      A

;; ANSWER SECTION:
google.com.                 1100    IN      NS      ns1.google.com.
google.com.                 1100    IN      NS      ns2.google.com.
google.com.                 1100    IN      NS      ns3.google.com.
google.com.                 1100    IN      NS      ns4.google.com.

```

```
;; ADDITIONAL SECTION:
ns1.google.com.      1100   IN     A      192.5.6.30
ns2.google.com.      1100   IN     A      192.33.14.30
ns3.google.com.      1100   IN     A      192.26.92.30
ns4.google.com.      1100   IN     A      192.31.80.30

;; Query time: 0 msec
;; SERVER: 10.90.91.63#53(10.90.91.63)
;; WHEN: Fri May 26 15:49:35 PDT 2017
;; MSG SIZE rcvd: 164
```

A dig query request of any type (A, NS, CNAME, SRV, etc.) to a domain containing the domain `?google.com?`, with wildcard matching on, 4 name servers (from `ns1.google.com` to `ns2.google.com`) and corresponding A records will always elicit an `NXDOMAIN` response:

```
; <<>> DiG 9.10.3-P4-Ubuntu <<>> @10.90.91.63 mail.google.com A
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->HEADER<<- opcode: QUERY, status: NXDOMAIN, id: 42986
;; flags: qr aa rd; QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 0
;; WARNING: recursion requested but not available

;; QUESTION SECTION:
;mail.google.com.      IN     A

;; Query time: 0 msec
;; SERVER: 10.90.91.63#53(10.90.91.63)
;; WHEN: Fri May 26 16:56:25 PDT 2017
;; MSG SIZE rcvd: 33
```