Network Working Group
Request for Comments: 2694
Category: Informational

P. Srisuresh
Consultant
G. Tsirtsis
BT Laboratories
P. Akkiraju
Cisco Systems
A. Heffernan
Juniper Networks

September 1999

DNS extensions to Network Address Translators (DNS_ALG)

Status of this Memo

Copyright Notice

Abstract

Domain Name Service (DNS) provides name to address mapping within a routing class (ex: IP). Network Address Translators (NATs) attempt to provide transparent routing between hosts in disparate address realms of the same routing class. Typically, NATs exist at the border of a stub domain, hiding private addresses from external addresses. This document identifies the need for DNS extensions to NATs and outlines how a DNS Application Level Gateway (DNS_ALG) can meet the need. DNS_ALG modifies payload transparently to alter address mapping of hosts as DNS packets cross one address realm into another. The document also illustrates the operation of DNS_ALG with specific examples.

1. Introduction

Network Address Translators (NATs) are often used when network's internal IP addresses cannot be used outside the network either for privacy reasons or because they are invalid for use outside the network.

Ideally speaking, a host name uniquely identifies a host and its address is used to locate routes to the host. However, host name and address are often not distinguished and used interchangeably by applications. Applications embed IP address instead of host name in

payload. Examples would be e-mails that specify their MX server address (ex: user@666.42.7.11) instead of server name (ex: user@private.com) as sender ID; HTML files that include IP address instead of names in URLs, etc. Use of IP address in place of host name in payload represents a problem as the packet traverses a NAT device because NATs alter network and transport headers to suit an address realm, but not payload.

DNS provides Name to address mapping. Whereas, NAT performs address translation (in network and transport headers) in datagrams traversing between private and external address realms. DNS Application Level Gateway (DNS_ALG) outlined in this document helps translate Name-to-Private-Address mapping in DNS payloads into Name-to-external-address mapping and vice versa using state information available on NAT.

A Network Address Port Translator (NAPT) performs address and Transport level port translations (i.e, TCP, UDP ports and ICMP query IDs). DNS name mapping granularity, however, is limited to IP addresses and does not extend to transport level identifiers. As a result, the DNS_ALG processing for an NAPT configuration is simplified in that all host addresses in private network are bound to a single external address. The DNS name lookup for private hosts (from external hosts) do not mandate fresh private-external address binding, as all private hosts are bound to a single pre-defined external address. However, reverse name lookups for the NAPT external address will not map to any of the private hosts and will simply map to the NAPT router. Suffices to say, the processing requirements for a DNS_ALG supporting NAPT configuration are a mere subset of Basic NAT. Hence, the discussion in the remainder of the document will focus mainly on Basic NAT, Bi-directional NAT and Twice NAT configurations, with no specific reference to NAPT setup.

Definitions for DNS and related terms may be found in [Ref 3] and [Ref 4]. Definitions for NAT related terms may be found in [Ref 1].

2. Requirement for DNS extensions

There are many ways to ensure that a host name is mapped to an address relevant within an address realm. In the following sections, we will identify where DNS extensions would be needed.

Typically, organizations have two types of authoritative name servers. Internal authoritative name servers identify all (or majority of) corporate resources within the organization. Only a portion of these hosts are allowed to be accessed by the external world. The remaining hosts and their names are unique to the private network. Hosts visible to the external world and the authoritative

name server that maps their names to network addresses are often configured within a DMZ (De-Militarized Zone) in front of a firewall. We will refer the hosts and name servers within DMZ as DMZ hosts and DMZ name servers respectively. DMZ host names are end-to-end unique in that their FQDNs do not overlap with any end node that communicates with it.

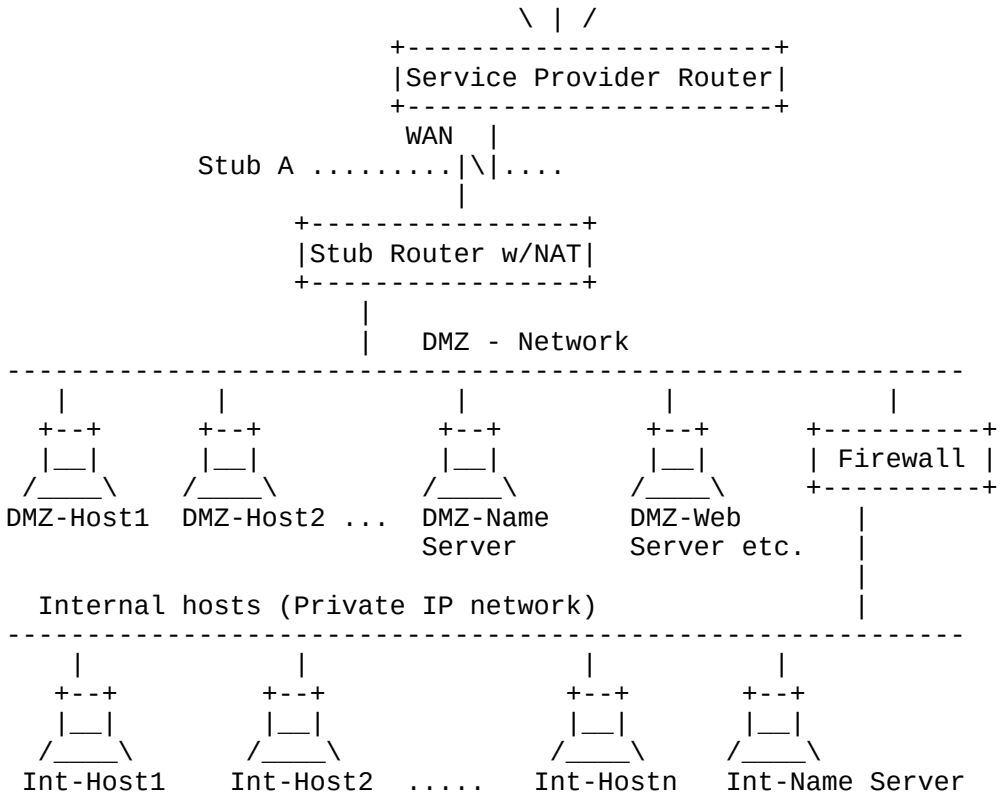


Figure 1: DMZ network configuration of a private Network.

Figure 1 above illustrates configuration of a private network which includes a DMZ. Actual configurations may vary. Internal name servers are accessed by users within the private network only. Internal DNS queries and responses do not cross the private network boundary. DMZ name servers and DMZ hosts on the other hand are end-to-end unique and could be accessed by external as well as internal hosts. Throughout this document, our focus will be limited to DMZ hosts and DMZ name servers and will not include internal hosts and internal name servers, unless they happen to be same.

2.1. DMZ hosts assigned static external addresses on NAT

Take the case where DMZ hosts are assigned static external addresses on the NAT device. Note, all hosts within private domain, including the DMZ hosts are identified by their private addresses. Static mapping on the NAT device allows the DMZ hosts to be identified by their public addresses in the external domain.

2.1.1. Private networks with no DMZ name servers

Take the case where a private network has no DMZ name server for itself. If the private network is connected to a single service provider for external connectivity, the DMZ hosts may be listed by their external addresses in the authoritative name servers of the service provider within their forward and in-add.arpa reverse zones.

If the network is connected to multiple service providers, the DMZ host names may be listed by their external address(es) within the authoritative name servers of each of the service providers. This is particularly significant in the case of in-addr.arpa reverse zones, as the private network may be assigned different address prefixes by the service providers.

In both cases, externally generated DNS lookups will not reach the private network. A large number of NAT based private domains pursue this option to have their DMZ hosts listed by their external addresses on service provider’s name servers.

2.1.2. Private networks with DMZ name servers

Take the case where a private network opts to keep an authoritative DMZ name server for the zone within the network itself. If the network is connected to a single service provider, the DMZ name server may be configured to obviate DNS payload interceptions as follows. The hosts in DMZ name server must be mapped to their statically assigned external addresses and the internal name server must be configured to bypass the DMZ name server for queries concerning external hosts. This scheme ensures that DMZ name servers are set for exclusive access to external hosts alone (not even to the DMZ hosts) and hence can be configured with external addresses only.

The above scheme requires careful administrative planning to ensure that DMZ name servers are not contacted by the private hosts directly or indirectly (through the internal name servers). Using DNS-ALG would obviate the administrative ordeals with this approach.

2.2. DMZ hosts assigned external addresses dynamically on NAT

Take the case where DMZ hosts in a private network are assigned external addresses dynamically by NAT. While the addresses issued to these hosts are fixed within the private network, their externally known addresses are ephemeral, as determined by NAT. In such a scenario, it is mandatory for the private organization to have a DMZ name server in order to allow access to DMZ hosts by their name.

The DMZ name server would be configured with private addresses for DMZ hosts. DNS Application Level Gateway (DNS_ALG) residing on NAT device will intercept the DNS packets directed to or from the DMZ name server(s) and perform transparent payload translations so that a DMZ host name has the right address mapping within each address realm (i.e., private or external).

3. Interactions between NAT and DNS_ALG

This document operates on the paradigm that interconnecting address realms may have overlapping address space. But, names of hosts within interconnected realms must be end-to-end unique in order for them to be accessed by all hosts. In other words, there cannot be an overlap of FQDNs between end nodes communicating with each other. The following diagram illustrates how a DNS packet traversing a NAT device (with DNS_ALG) is subject to header and payload translations. A DNS packet can be a TCP or UDP packet with the source or destination port set to 53. NAT would translate the IP and TCP/UDP headers of the DNS packet and notify DNS-ALG to perform DNS payload changes. DNS-ALG would interact with NAT and use NAT state information to modify payload, as necessary.

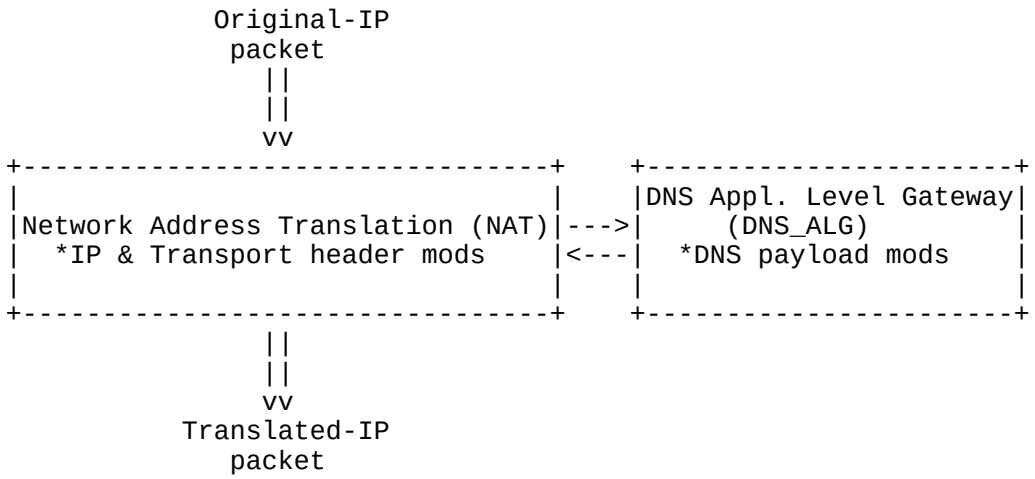


Figure 2: NAT & DNS-ALG in the translation path of DNS packets

3.1. Address Binding considerations

We will make a distinction between "Temporary Address Binding" and "Committed Address Binding" in NATs. This distinction becomes necessary because the DNS_ALG will allow external users to create state on NAT, and thus the potential for denial-of-service attacks. Temporary address binding is the phase in which an address binding is reserved without any NAT sessions using the binding. Committed address binding is the phase in which there exists at least one NAT session using the binding between the external and private addresses. Both types of bindings are used by DNS_ALG to modify DNS payloads. NAT uses only the committed address bindings to modify the IP and Transport headers of datagrams pertaining to NAT sessions.

For statically mapped addresses, the above distinction is not relevant. For dynamically mapped addresses, temporary address binding often precedes committed binding. Temporary binding occurs when DMZ name server is queried for a name lookup. Name query is likely a pre-cursor to a real session between query originator and the queried host. The temporary binding becomes committed only when NAT sees the first packet of a session between query initiator and queried host.

A configurable parameter, "Bind-holdout time" may be defined for dynamic address assignments as the maximum period of time for which a temporary address binding is held active without transitioning into a committed binding. With each use of temporary binding by DNS_ALG (to modify DNS payload), this Bind-holdout period is renewed. A default Bind-holdout time of a couple of minutes might suffice for most DNS-ALG implementations. Note, it is possible for a committed address

binding to occur without ever having to be preceded by a temporary binding. Lastly, when NAT is ready to unbind a committed address binding, the binding is transitioned into a temporary binding and kept in that phase for an additional Bind-holdout period. The binding is freed only upon expiry of Bind-holdout time. The Bind-holdout time preceding the committed-address-binding and the address-unbinding are required to ensure that end hosts have sufficient time in which to initiate a data session subsequent to a name lookup.

For example, say a private network with address prefix 10/8 is mapped to 198.76.29/24. When an external hosts makes a DNS query to host7, bearing address 10.0.0.7, the DMZ name server within private network responds with an A type RR for host7 as:

```
host7  A  10.0.0.7
```

DNS_ALG would intercept the response packet and if 10.0.0.7 is not assigned an external address already, it would request NAT to create a temporary address binding with an external address and start Bind-holdout timer to age the binding. Say, the assigned external address is 198.76.29.1. DNS-ALG would use this temporary binding to modify the RR in DNS response, replacing 10.0.0.7 with its external address and reply with:

```
host7  A  198.76.29.1
```

When query initiator receives DNS response, only the assigned external address is seen. Within a short period (presumably before the bind-holdout time expires), the query initiator would initiate a session with host7. When NAT notices the start of new session directed to 198.76.29.1, NAT would terminate Bind-holdout timer and transition the temporary binding between 198.76.29.1 and 10.0.0.7 into a committed binding.

To minimize denial of service attacks, where a malicious user keeps attempting name resolutions, without ever initiating a connection, NAT would have to monitor temporary address bindings that have not transitioned into committed bindings. There could be a limit on the number of temporary bindings and attempts to generate additional temporary bindings could be simply rejected. There may be other heuristic solutions to counter this type of malicious attacks.

We will consider bi-directional NAT to illustrate the use of temporary binding by DNS_ALG in the following sub-sections, even though the concept is applicable to other flavors of NATs as well.

3.2. Incoming queries

In order to initiate incoming sessions, an external host obtains the V4 address of the DMZ-host it is trying to connect to by making a DNS request. This request constitutes prelude to the start of a potential new session.

The external host resolver makes a name lookup for the DMZ host through its DNS server. When the DNS server does not have a record of IPv4 address attached to this name, the lookup query is redirected at some point to the Primary/Backup DNS server (i.e., in DMZ) of the private stub domain.

Enroute to DMZ name server, DNS_ALG would intercept the datagram and modify the query as follows.

a) For Host name to Host address query requests:
   Make no change to the DNS payload.

b) For Host address to Host name queries: Replace the external V4 address octets (in reverse order) preceding the string "IN-ADDR.ARPA" with the corresponding private V4 address, if such an address binding exists already. However, if a binding does not exist, the DNS_ALG would simply respond (as a name server would) with a response code (RCODE) of 5 (REFUSED to respond due to policy reasons) and set ANCOUNT, NSCOUNT and ARCOUT to 0 in the header section of the response.

In the opposite direction, as DNS response traverses from the DNS server in private network, DNS_ALG would once again intercept the packet and modify as follows.

a) For a host name to host address query requests, replace the private address sent by DMZ name server with a public address internally assigned by the NAT router. If a public address is not previously assigned to the host’s private address, NAT would assign one at this time.

b) For host address to host name queries, replace the private address octets preceding the string "IN-ADDR.ARPA" in response RRs with their external address assignments. There is a chance here that by the time the DMZ name server replies, the bind-holdout timer in NAT for the address in question has expired. In such a case, DNS_ALG would simply drop the reply. The sender will have to resend the query (as would happen when a router enroute drops the response).

For static address assignments, the TTL value supplied in the original RR will be left unchanged. For dynamic address assignments, DNS_ALG would modify the TTL value on DNS resource records (RRs) to be 0, implying that the RRs should only be used for transaction in progress, and not be cached. For compatibility with broken implementations, TTL of 1 might in practice work better.

Clearly, setting TTL to be 0 will create more traffic than if the addresses were static, because name-to-address mapping is not cached. Specifically, network based applications will be required to use names rather than addresses for identifying peer nodes and must use DNS for every name resolution, as name-to-address mapping cannot be shared from the previously run applications.

In addition, NAT would be requested to initiate a bind-holdout timer following the assignment. If no session is initiated to the private host within the Bind-holdout time period, NAT would terminate the temporary binding.

3.3. Outgoing Queries

For Basic and bi-directional NATs, there is no need to distinguish between temporary and committed bindings for outgoing queries. This is because, DNS_ALG does not modify the DNS packets directed to or from external name servers (used during outbound sessions), unlike the inbound DNS sessions.

Say, a private host needs to communicate with an external host. The DNS query goes to the internal name server (if there exists one) and from there to the appropriate authoritative/cache name server outside the private domain. The reply follows the same route but neither the query nor the response are subject to DNS_ALG translations.

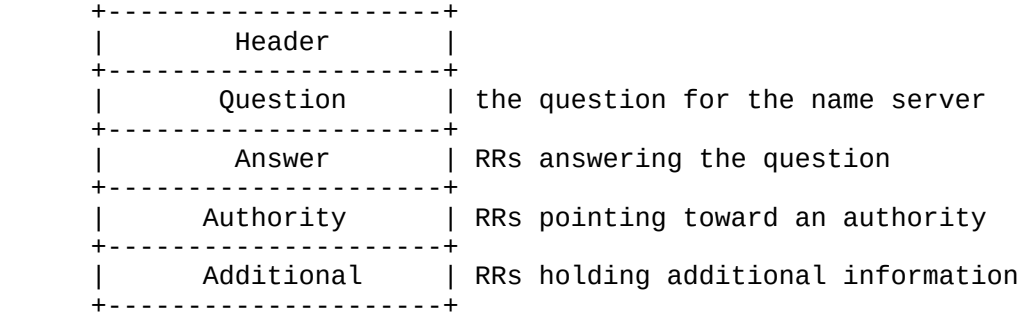
This however will not be the case with address isolated twice NAT private and external domains. In such a case, NAT would intercept all DNS packets and make address modifications to payload as discussed in the previous section. Temporary Private to external address bindings are created when responses are sent by private DNS servers and temporary external to private address bindings are created when responses are sent by external DNS servers.

4. DNS payload modifications by DNS-ALG

Typically, UDP is employed as the transport mechanism for DNS queries and responses and TCP for Zone refresh activities. In either case, name servers are accessed using a well-known DNS server port 53 (decimal) and all DNS payloads have the following format of data [Ref

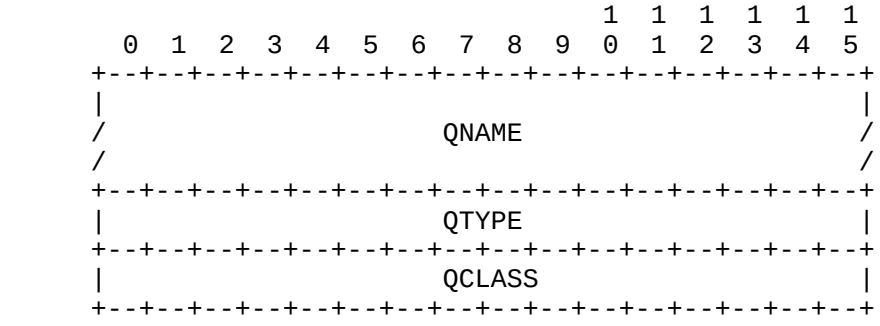
4]. While NAT is responsible for the translation of IP and TCP/UDP headers of a DNS packet, DNS-ALG is responsible for updating the DNS payload.

The header section within the DNS payload is always present and includes fields specifying which of the remaining sections are present. The header identifies if the message is a query or a response. No changes are required to be made by DNS-ALG to the Header section. DNS_ALG would parse only the DNS payloads whose QCLASS is set to IN (IP class).



4.1. Question section

The question section contains QDCOUNT (usually 1) entries, as specified in Header section, with each of the entries in the following format:



4.1.1. PTR type Queries

DNS_ALG must identify all names, whose FQDNs (i.e., Fully Qualified Domain Names) fall within IN-ADDR.ARPA domain and replace the address octets (in reverse order) preceding the string "IN-ADDR.ARPA" with the corresponding assigned address octets in reverse order, only if the address binding is active on the NAT router. If the address

preceding the string "IN-ADDR.ARPA" falls within the NAT address map, but does not have at least a temporary address binding, DNS_ALG would simply simply respond back (as a DNS name server would) with a response code (RCODE) of 5 (REFUSED to respond due to policy reasons) and set ANCOUNT, NSCOUNT and ARCOUT to 0 in the header section of the response.

Note that the above form of host address to host name type queries will likely yield different results at different times, depending upon address bind status in NAT at a given time.

For example, a resolver that wanted to find out the hostname corresponding to address 198.76.29.1 (externally) would pursue a query of the form:

QTYPE = PTR, QCLASS = IN, QNAME = 1.29.76.198.IN-ADDR.ARPA.

DNS_ALG would intervene and if the address 198.76.29.1 is internally mapped to a private address of 10.0.0.1, modify the query as below and forward to DMZ name server within private network.

QTYPE = PTR, QCLASS = IN, QNAME = 1.0.0.10.IN-ADDR.ARPA

Presumably, the DMZ name server is the authoritative name server for 10.IN-ADDR.ARPA zone and will respond with an RR of the following form in answer section. DNS_ALG translations of the response RRs will be considered in a following section.

1.0.0.10.IN-ADDR.ARPA PTR host1.fooboo_org.provider_domain

An example of Inverse translation is e-mail programs using inverse translation to trace e-mail originating hosts for spam prevention. Verify if the address from which the e-mail was sent does indeed belong to the same domain name the sender claims in sender ID.

Query modifications of this nature will likely change the length of DNS payload. As a result, the corresponding IP and TCP/UDP header checksums must be updated. In case of TCP based queries, the sequence number deltas must be tracked by NAT so that the delta can be applied to subsequent sequence numbers in datagrams in the same direction and acknowledgement numbers in datagrams in the opposite direction. In case of UDP based queries, message sizes are restricted to 512 bytes (not counting the IP or UDP headers). Longer messages must be truncated and the TC bit should be set in the header.

Lastly, any compressed domain names using pointers to represent common domain denominations must be updated to reflect new pointers with the right offset, if the original domain name had to be translated by NAT.

4.1.2. A, MX, NS and SOA type Queries

For these queries, DNS_ALG would not modify any of the fields in the query section, not even the name field.

4.1.3. AXFR type Queries

AXFR is a special zone transfer type query. Zone transfers from private address realm must be avoided for address assignments that are not static. Typically, TCP is used for AXFR requests.

When changes are made to a zone, they must be distributed to all name servers. The general model of automatic zone transfer or refreshing is that one of the name servers is the master or primary for the zone. Changes are coordinated at the primary, typically by editing a master file for the zone. After editing, the administrator signals the master server to load the new zone. The other non-master or secondary servers for the zone periodically check the SERIAL field of the SOA for the zone for changes (at some polling intervals) and obtain new zone copies when changes have been made.

Zone transfer is usually from primary to backup name servers. In the case of NAT supported private networks, primary and backup servers are advised to be located in the same private domain (say, private.zone) so zone transfer is not across the domain and DNS_ALG support for zone transfer is not an issue. In the case a secondary name server is located outside the private domain, zone transfers must not be permitted for non-static address assignments. Primary and secondary servers are required to be within the same private domain because all references to data in the zone had to be captured. With dynamic address assignments and bindings, it is impossible to translate the axfr data to be up-to-date. Hence, if a secondary server for private.zone were to be located external to the domain, it would contain bad data. Note, however, the requirement outlined here is not in confirmence with RFC 2182, which recommends primary and secondary servers to be placed at topologically and geographically dispersed locations on the Internet.

During zone transfers, DNS_ALG must examine all A type records and replace the original address octets with their statically assigned address octets. DNS_ALG could also examine if there is an attempt to

transfer records for hosts that are not assigned static addresses and drop those records alone or drop the whole transfer. This would minimize misconfiguration and human errors.

4.1.4. Dynamic Updates to the DNS.

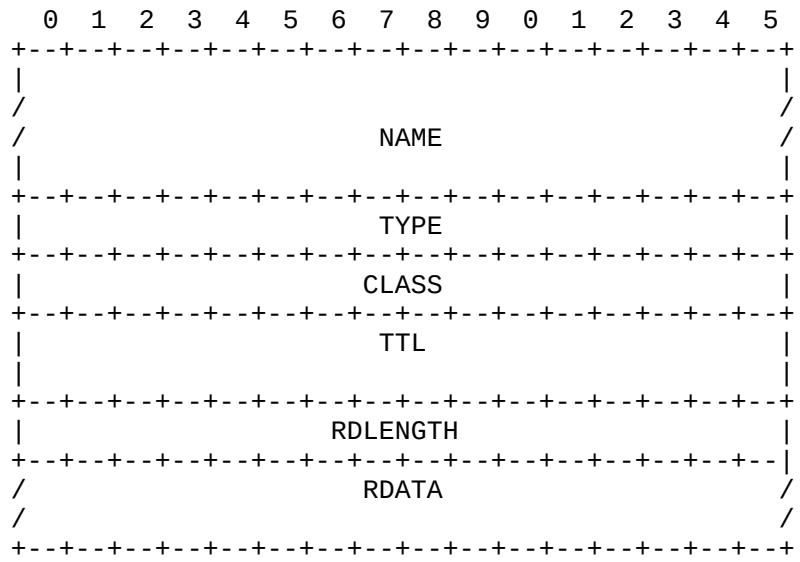
An authoritative name server can have dynamic updates from the nodes within the zone without intervention from NAT and DNS-ALG, so long as one avoids spreading a DNS zone across address realms. We recommend keeping a DNS zone within the same realm it is responsible for. By doing this, DNS update traffic will not cross address realms and hence will not be subject to consideration by DNS-ALG.

Further, if dynamic updates do cross address realms, and the updates must always be secured via DNSSEC, then such updates are clearly out of scope for DNS-ALG (as described in section 7).

4.2. Resource records in all other sections

The answer, authority, and additional sections all share the same format, with a variable number of resource records. The number of RRs specific to each of the sections may be found in the corresponding count fields in DNS header. Each resource record has the following format:

The TTL value supplied in the original RRs will be left unchanged for static address assignments. For dynamic address assignments, DNS_ALG will modify the TTL to be 0, so the RRs are used just for the transaction in progress, and not cached. RFC 2181 requires all RRs in an RRset (RRs with the same name, class and type, but with different RDATA) to have the same TTL. So if the TTL of an RR is set to 0, all other RRs within the same RRset will also be adjusted by the DNS-ALG to be 0.



4.2.1. PTR type RRs

The considerations specified in the Question section is equally valid with names for PTR type RRs. Private address preceding the string "IN-ADDR.ARPA" (in reverse order of octets) must be replaced by its external address assignment (in reverse order), if a binding exists. The remaining fields for this RR remain unchanged.

4.2.2. A type RRs

The RDATA for A records is a 4-byte IP address. DNS_ALG would simply replace the original address in RDATA with its externally assigned IP address, if it succeeded in finding an address binding. Successful address translation should cause no changes to payload length. Only the transport header checksum would need updating. In case of failure to find an address binding, DNS_ALG would have to drop the record and decrement the corresponding COUNT field in the header section.

4.2.3. CNAME, MX, NS and SOA type RRs

No changes required to be made by DNS_ALG for these RRs, as the RDATA does not contain any IP addresses. The host names within the RDATA remain unchanged between realms.

5. Illustration of DNS_ALG in conjunction with Bi-directional NAT

The following diagram illustrates the operation of DNS_ALG in a a bi-directional NAT router. We will illustrate by walking through how name lookup and reverse name lookup queries are processed.

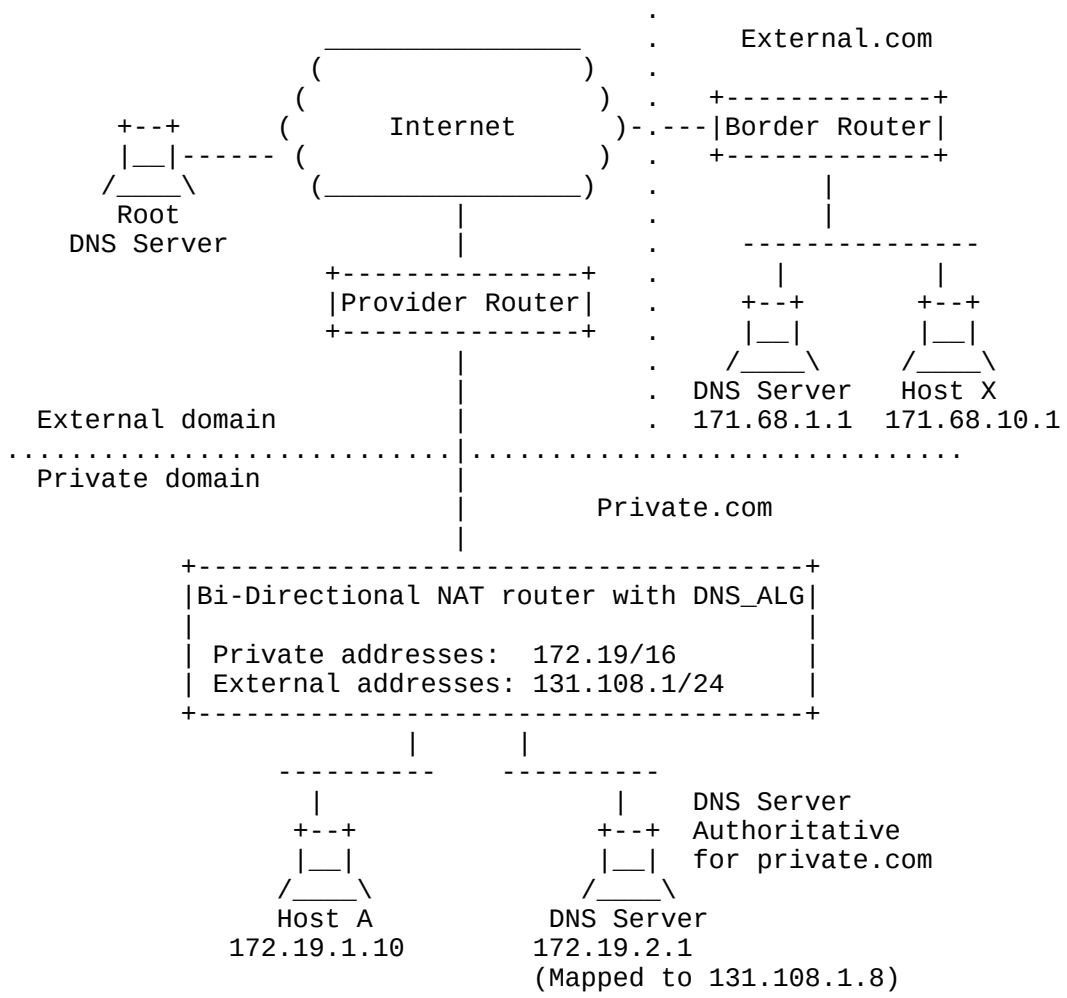


Figure 3: DNS-ALG operation in Bi-Directional NAT setup

The above diagram depicts a scenario where a company private.com using private address space 172.19/16 connects to the Internet using bi-directional NAT. DNS_ALG is embedded in the NAT device to make necessary DNS payload changes. NAT is configured to translate the private addresses space into an external address block of

131.108.1/24. NAT is also configured with a static translation for private.com's DNS server, so it can be referred in the external domain by a valid address.

The company external.com is located in the external domain, using a registered address block of 171.68/16. Also shown in the topology is a root DNS server.

Following simplifications are made to the above configuration:

* private.com is not multihomed and all traffic to the external space transits a single NAT.

* The DNS server for private.com is authoritative for the private.com domain and points to the root server for all other DNS resolutions. The same is true for the DNS server in external.com.

* The internal name servers for private.com and external.com are same as their DMZ name servers. The DNS servers for these domains are configured with addresses private to the organization.

* The name resolvers on host nodes do not have recursion available on them and desire recursive service from servers. All name servers are assumed to be able to provide recursive service.

5.1. Outgoing Name-lookup queries

Say, Host A in private.com needs to perform a name lookup for host X in external.com to initiate a session with X. This would proceed as follows.

1. Host A sends a UDP based name lookup query (A record) for "X.External.Com" to its local DNS server.

2. Local DNS server sends the query to the root server enroute NAT. NAT would change the IP and UDP headers to reflect DNS server's statically assigned external address. DNS_ALG will make no changes to the payload.

3. The root server, in turn, refers the local DNS server to query the DNS server for External.com. This referal transits the NAT enroute to the local DNS server. NAT would simply translate the IP and UDP headers of the incoming packet to reflect DNS server's private address. No changes to the payload by DNS_ALG.

4. Private.com DNS server will now send the query to the DNS server for external.com, once again, enroute NAT. Just as with the query to root, The NAT router would change the IP and UDP headers to reflect the DNS server's statically assigned external address. And, DNS_ALG will make no changes to the payload.

5. The DNS server for external.com replies with the IP address 171.68.10.1. This reply also transits the NAT. NAT would translate the IP and UDP headers of the incoming packet to reflect DNS server's private address. Once again, no changes to the payload by DNS_ALG.

6. The DNS server in Private.com replies to host A.

When Host A finds the address of Host X, A initiates a session with host X, using a destination IP address of 171.68.10.1. This datagram and any others that follow in this session will be translated as usual by NAT.

Note, DNS_ALG does not change the payload for DNS packets in either direction.

5.2. Reverse name lookups originated from private domain

This scenario builds on the previous case by having host A in Private.com perform a reverse name lookup on 171.68.10.1, which is host X's global address. Following is a sequence of events.

1. Host A sends a UDP based inverse name lookup query (PTR record) for "1.10.68.171.IN-ADDR.ARPA." to its local DNS server.

2. Local DNS server sends the query to the root server enroute NAT. As before, NAT would change the IP and UDP headers to reflect DNS server's statically assigned external address. DNS_ALG will make no changes to the payload.

3. The root server, in turn, refers the local DNS server to query the DNS server for External.com. This referal transits the NAT enroute to the local DNS server. NAT would simply translate the IP and UDP headers of the incoming packet to reflect DNS server's private address. No changes to the payload by DNS_ALG.

4. Private.com DNS server will now send the query to the DNS server for external.com, once again, enroute NAT. Just as with the query to root, The NAT router would change the IP and UDP headers to reflect the DNS server's statically assigned external address. And, DNS_ALG will make no changes to the payload.

5. The DNS server for external.com replies with the host name of "X.External.Com.". This reply also transits the NAT. NAT would translate the IP and UDP headers of the incoming packet to reflect DNS server's private address. Once again, no changes to the payload by DNS_ALG.

6. The DNS server in Private.com replies to host A.

Note, DNS_ALG does not change the payload in either direction.

5.3. Incoming Name-lookup queries

This time, host X in external.com wishes to initiate a session with host A in Private.com. Below are the sequence of events that take place.

1. Host X sends a UDP based name lookup query (A record) for "A.Private.com" to its local DNS server.

2. Local DNS server in External.com sends the query to root server.

3. The root server, in turn, refers the DNS server in External.com to query the DNS server for private.com,

4. External.com DNS server will now send the query to the DNS server for Private.com. This query traverses the NAT router. NAT would change the IP and UDP headers of the packet to reflect the DNS server's private address. DNS_ALG will make no changes to the payload.

5. The DNS server for Private.com replies with the IP address 172.19.1.10 for host A. This reply also transits the NAT. NAT would translate the IP and UDP headers of the outgoing packet from the DNS server.

   DNS_ALG will request NAT to (a) setup a temporary binding for Host A (172.19.1.10) with an external address and (b) initiate Bind-holdout timer. When NAT successfully sets up a temporary binding with an external address (say, 131.108.1.12), DNS_ALG would modify the payload to replace A's private address with its external assigned address and set the Cache timeout to 0.

6. The server in External.com replies to host X

When Host X finds the address of Host A, X initiates a session with A, using a destination IP address of 131.108.1.12. This datagram and any others that follow in this session will be translated as usual by the NAT.

Note, DNS_ALG changes only the response packets from the DNS server for Private domain.

5.4. Reverse name lookups originated from external domain

This scenario builds on the previous case (section 5.3) by having host X in External.com perform a reverse name lookup on 131.108.1.12, which is host A's assigned external address. The following sequence of events take place.

1. Host X sends a UDP based inverse name lookup query (PTR record) for "12.1.108.131.IN-ADDR.ARPA." to its local DNS server.

2. Local DNS server in External.com sends the query to the root server.

3. The root server, in turn, refers the local DNS server to query the DNS server for Private.com.

4. External.com DNS server will now send the query to the DNS server for Private.com. This query traverses the NAT router. NAT would change the IP and UDP headers to reflect the DNS server's private address.

   DNS_ALG will enquire NAT for the private address associated with the external address of 131.108.1.12 and modify the payload, replacing 131.108.1.12 with the private address of 172.19.1.10.

5. The DNS server for Private.com replies with the host name of "A.Private.Com.". This reply also transits the NAT. NAT would translate the IP and UDP headers of the incoming packet to reflect DNS server's private address.

   Once again, DNS_ALG will enquire NAT for the assigned external address associated with the private address of 172.19.1.10 and modify the payload, replacing 172.19.1.10 with the assigned external address of 131.108.1.12.

6. The DNS server in External.com replies to host X.

Note, DNS_ALG changes the query as well as response packets from DNS server for Private domain.

6. Illustration of DNS_ALG in conjunction with Twice-NAT

The following diagram illustrates the operation of DNS_ALG in a Twice NAT router. As before, we will illustrate by walking through how name lookup and reverse name lookup queries are processed.

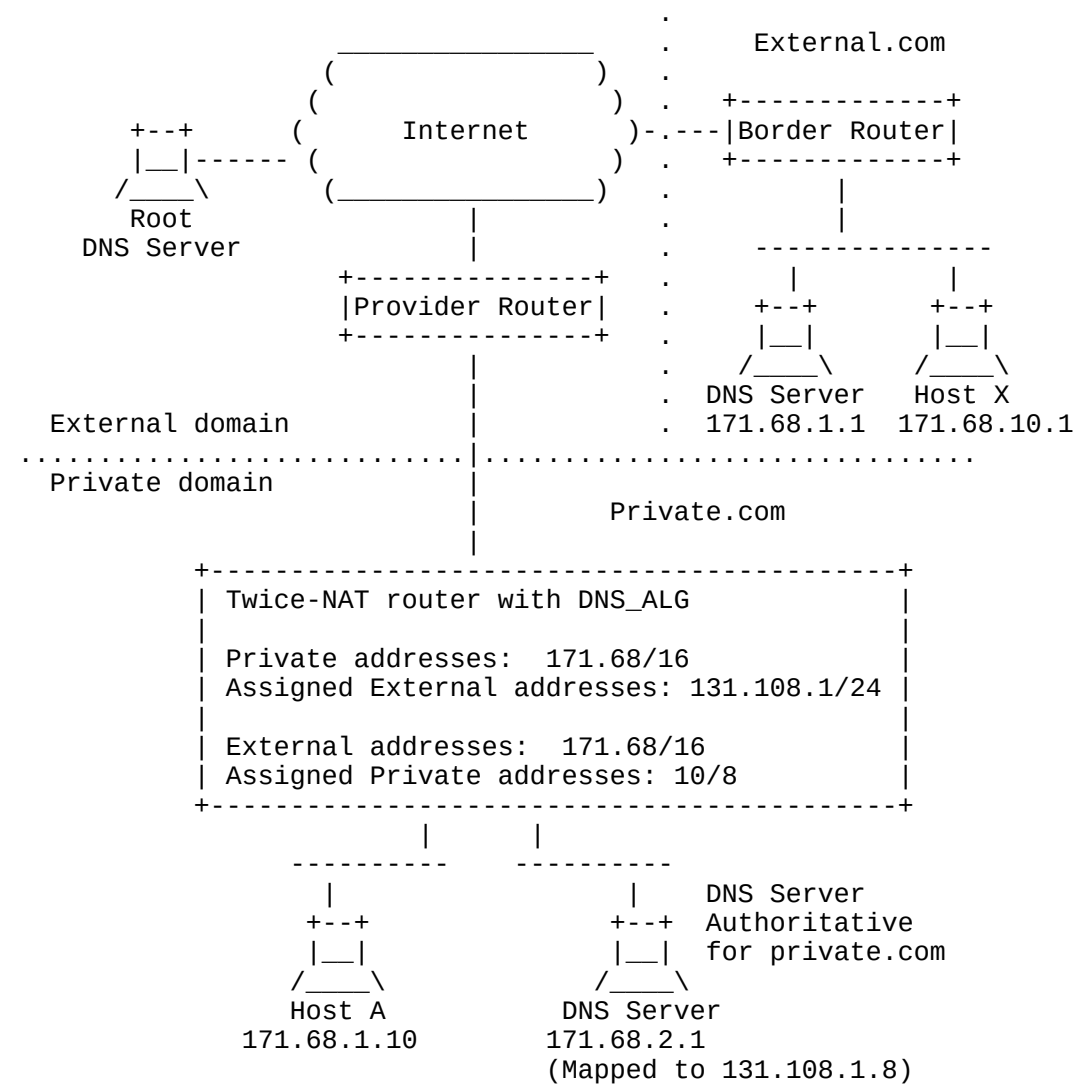


Figure 4: DNS-ALG operation in Twice-NAT setup

In this scenario, hosts in private.com were not numbered from the RFC 1918 reserved 172.19/16 space, but rather were numbered with the globally-routable 171.68/16 network, the same as external.com. Not only does private.com need translation service for its own host addresses, but it also needs translation service if any of those hosts are to be able to exchange datagrams with hosts in external.com. Twice-NAT accommodates the transition by translating the overlapping address space used in external.com with a unique

address block (10/8) from RFC 1918 address space. Routes are set up within the private domain to direct datagrams destined for the address block 10/8 through Twice-NAT device to the external global network space.

Simplifications and assumptions made in section 5.0 will be valid here as well.

6.1. Outgoing Name-lookup queries

Say, Host A in private.com needs to perform a name lookup for host X in external.com (host X has a FQDN of X.external.com), to find its address. This would would proceed as follows.

1. Host A sends a UDP based name lookup query (A record) for "X.External.Com" to its local DNS server.

2. Local DNS server sends the query to the root server enroute NAT. NAT would change the IP and UDP headers to reflect DNS server's statically assigned external address. DNS_ALG will make no changes to the payload.

3. The root server, in turn, refers the local DNS server to query the DNS server for External.com. This referal transits the NAT enroute to the local DNS server. NAT would simply translate the IP and UDP headers of the incoming packet to reflect DNS server's private address.

   DNS_ALG will request NAT for an assigned private address for the referral server and replace the external address with its assigned private address in the payload.

4. Private.com DNS server will now send the query to the DNS server for external.com, using its assigned private address, via NAT. This time, NAT would change the IP and UDP headers to reflect the External addresses of the DNS servers. I.e., Private.com DNS server's IP address is changed to its assigned external address and External.Com DNS server's assigned Private address is changed to its external address.

   DNS_ALG will make no changes to the payload.

5. The DNS server for external.com replies with the IP address 171.68.10.1. This reply also transits the NAT. NAT would once again translate the IP and UDP headers of the incoming to reflect the private addresses of the DNS servers. I.e., Private.com DNS

server's IP address is changed to its private address and External.Com DNS server's external address is changed to its assigned Private address.

DNS_ALG will request NAT to (a) set up a temporary binding for Host X (171.68.10.1) with a private address and (b) initiate Bind-holdout timer. When NAT successfully sets up temporary binding with a private address (say, 10.0.0.254), DNS_ALG would modify the payload to replace X's external address with its assigned private address and set the Cache timeout to 0.

6. The DNS server in Private.com replies to host A.

When Host A finds the address of Host X, A initiates a session with host X, using a destination IP address of 10.0.0.254. This datagram and any others that follow in this session will be translated as usual by Twice NAT.

Note, the DNS_ALG has had to change payload in both directions.

6.2. Reverse name lookups originated from private domain

This scenario builds on the previous case by having host A in Private.com perform a reverse name lookup on 10.0.0.254, which is host X's assigned private address. Following is a sequence of events.

1. Host A sends a UDP based inverse name lookup query (PTR record) for "254.0.0.10.IN-ADDR.ARPA." to its local DNS server.

2. Local DNS server sends the query to the root server enroute NAT. As before, NAT would change the IP and UDP headers to reflect DNS server's statically assigned external address.

   DNS_ALG will translate the private assigned address 10.0.0.254 with its external address 171.68.10.1.

3. The root server, in turn, refers the local DNS server to query the DNS server for External.com. This referal transits the NAT enroute to the local DNS server. NAT would simply translate the IP and UDP headers of the incoming packet to reflect DNS server's private address.

   As with the original query, DNS_ALG will translate the private assigned address 10.0.0.254 with its external address 171.68.10.1. In addition, DNS_ALG will replace the external address of the referal server (i.e., the DNS server for External.com) with its assigned private address in the payload.

4. Private.com DNS server will now send the query to the DNS server for external.com, using its statically assigned private address, via NAT. This time, NAT would change the IP and UDP headers to reflect the External addresses of the DNS servers. I.e., Private.com DNS server's IP address is changed to its assigned external address and External.Com DNS server's assigned Private address is changed to its external address.

   As with the original query, DNS_ALG will translate the private assigned address 10.0.0.254 with its external address 171.68.10.1.

5. The DNS server for external.com replies with the FQDN of "X.External.Com.". This reply also transits the NAT. NAT would once again translate the IP and UDP headers of the incoming to reflect the private addresses of the DNS servers. I.e., Private.com DNS server's IP address is changed to its private address and External.Com DNS server's external address is changed to its assigned Private address.

   Once again, DNS_ALG will translate the query section, replacing the external address 171.68.10.1 with its assigned private address of 10.0.0.254

6. The DNS server in Private.com replies to host A.

Note, the DNS_ALG has had to change payload in both directions.

6.3. Incoming Name-lookup queries

This time, host X in external.com wishes to initiate a session with host A in Private.com. Below are the sequence of events that take place.

1. Host X sends a UDP based name lookup query (A record) for "A.Private.com" to its local DNS server.

2. Local DNS server in External.com sends the query to root server.

3. The root server, in turn, refers the DNS server in External.com to query the DNS server for private.com,

4. External.com DNS server will now send the query to the DNS server for Private.com. This query traverses the NAT router. NAT would change the IP and UDP headers to reflect the private addresses of the DNS servers. I.e., Private.com DNS server's IP address is changed to its private address and External.Com DNS server's external address is changed to assigned Private address.

DNS_ALG will make no changes to the payload.

5. The DNS server for Private.com replies with the IP address 171.68.1.10 for host A. This reply also transits the NAT. NAT would once again translate the IP and UDP headers of the incoming to reflect the external addresses of the DNS servers. I.e., Private.com DNS server's IP address is changed to its assigned external address and External.Com DNS server's assigned private address is changed to its external address.

   DNS_ALG will request NAT to (a) set up temporary binding for Host A (171.68.1.10) with an external address and (b) initiate Bind-holdout timer. When NAT succeeds in finding an external address (say, 131.108.1.12) to temporarily bind to host A, DNS_ALG would modify the payload to replace A's private address with its external assigned address and set the Cache timeout to 0.

6. The server in External.com replies to host X

When Host X finds the address of Host A, X initiates a session with A, using a destination IP address of 131.108.1.12. This datagram and any others that follow in this session will be translated as usual by the NAT.

Note, DNS_ALG changes only the response packets from the DNS server for Private domain.

6.4. Reverse name lookups originated from external domain

This scenario builds on the previous case (section 6.3) by having host X in External.com perform a reverse name lookup on 131.108.1.12, which is host A's assigned external address. The following sequence of events take place.

1. Host X sends a UDP based inverse name lookup query (PTR record) for "12.1.108.131.IN-ADDR.ARPA." to its local DNS server.

2. Local DNS server in External.com sends the query to the root server.

3. The root server, in turn, refers the local DNS server to query the DNS server for Private.com.

4. External.com DNS server will now send the query to the DNS server for Private.com. This query traverses the NAT router. NAT would change the IP and UDP headers to reflect the private addresses of the DNS servers. I.e., Private.com DNS server's IP address is changed to its private address and External.Com DNS server's external address is changed to assigned Private address.

   DNS_ALG will enquire NAT for the private address associated with the external address of 131.108.1.12 and modify the payload, replacing 131.108.1.12 with the private address of 171.68.1.10.

5. The DNS server for Private.com replies with the host name of "A.Private.Com.". This reply also transits the NAT. NAT would once again translate the IP and UDP headers of the incoming to reflect the external addresses of the DNS servers. I.e., Private.com DNS server's IP address is changed to its assigned external address and External.Com DNS server's assigned private address is changed to its external address.

   Once again, DNS_ALG will enquire NAT for the assigned external address associated with the private address of 172.19.1.10 and modify the payload, replacing 171.68.1.10 with the assigned external address of 131.108.1.12.

6. The DNS server in External.com replies to host X.

Note, DNS_ALG changes the query as well as response packets from DNS server for Private domain.

7. DNS-ALG limitations and Future Work

NAT increases the probability of mis-addressing. For example, same local address may be bound to different public address at different times and vice versa. As a result, hosts that cache the name to address mapping for longer periods than the NAT router is configured to hold the map are likely to misaddress their sessions. Note, this is mainly an issue with bad host implementations that hold DNS records longer than the TTL in them allows and is not directly attributable to the mechanism described here.

DNS_ALG cannot support secure DNS name servers in the private domain. I.e., Signed replies from an authoritative DNS name server in the DMZ to queries originating from the external world will be broken by the DNS-ALG. At best, DNS-ALG would be able to transform secure dnssec data into unprotected data. End-node demanding DNS replies to be signed may reject replies that have been tampered with by DNS_ALG. Since, the DNS server does not have a way to find where the queries come from (i.e., internal or external), it will most likely have to

resort to the common denomination of today's insecure DNS. Both are serious limitations to DNS_ALG. Zone transfers between DNS-SEC servers is also impacted the same way, if the transfer crosses address realms.

The good news, however, is that only end-nodes in DMZ pay the price for the above limitation in a traditional NAT (or, a bi-directional NAT), as external end-nodes may not access internal hosts due to DNS replies not being secure. However, for outgoing sessions (from private network) in a bi-directional NAT setup, the DNS queries can be signed and securely accepted by DMZ and other internal hosts since DNS_ALG does not intercept outgoing DNS queries and incoming replies. Lastly, zone transfers between DNS-SEC servers within the same private network are not impacted.

Clearly, with DNS SEC deployment in DNS servers and end-host resolvers, the scheme suggested in this document will not work.

8. Security Considerations

If DNS packets are encrypted/authenticated per DNSSEC, then DNS_ALG will fail because it won't be able to perform payload modifications. Alternately, if packets must be preserved in an address realm, DNS_ALG will need to hold the secret key to decrypt/verify payload before forwarding packets to a different realm. For example, if DNS-ALG, NAT and IPsec gateway (providing secure tunneling service) are resident on the same device, DNS-ALG will have access to the IPsec security association keys. The preceding section, "DNS-ALG limitations and Future Work" has coverage on DNS_ALG security considerations.

Further, with DNS-ALG, there is a possibility of denial of service attack from a malicious user, as outlined in section 3.1. Section 3.1 suggests some ways to counter this attack.

REFERENCES

[1] Srisuresh, P. and M. Holdrege, "IP Network Address Translator (NAT) Terminology and Considerations", RFC 2663, August 1999.

[2] Egevang, K. and P. Francis, "The IP Network Address Translator (NAT)", RFC 1631, May 1994.

[3] Rekhter, Y., Moskowitz, B., Karrenberg, D., de Groot, G. and E. Lear, "Address Allocation for Private Internets", BCP 5, RFC 1918, February 1996.

[4] Mockapetris, P., "Domain Names - Concepts and Facilities", STD 13, RFC 1034, November 1987.

[5] Mockapetris, P., "Domain Names - Implementation and Specification", STD 13, RFC 1035, November 1987.

[6] Reynolds J. and J. Postel, "Assigned Numbers", STD 2, RFC 1700, October 1994.

[7] Braden, R., "Requirements for Internet Hosts -- Communication Layers", STD 3, RFC 1122, October 1989.

[8] Braden, R., "Requirements for Internet Hosts -- Application and Support", STD 3, RFC 1123, October 1989.

[9] Baker, F., "Requirements for IP Version 4 Routers", RFC 1812, June 1995.

[10] Carpenter, B., Crowcroft, J. and Y. Rekhter, "IPv4 Address Behaviour Today", RFC 2101, February 1997.

[11] Eastlake, D., "Domain Name System Security Extensions", RFC 2535, March 1999.

[12] Vixie, P., Thompson, S., Rekhter Y. and J. Bound, "Dynamic Updates in the Domain Name System (DNS UPDATE)", RFC 2136, April 1997.

[13] Eastlake, D., "Secure Domain Name System Dynamic Update", RFC 2137, April 1997.

[14] Elz R. and R. Bush, "Clarifications to the DNS specification", RFC 2181, July 1997.

[15] Elz, R., R. Bush, Bradner S. and M. Patton, "Selection and Operation of Secondary DNS Servers", RFC 2182, July 1997.

Authors' Addresses

Pyda Srisuresh
849 Erie Circle
Milpitas, CA 95035
U.S.A.

Phone: +1 (408) 263-7527
EMail: srisuresh@yahoo.com

George Tsirtsis
Internet Transport Group
B29 Room 129
BT Laboratories
Martlesham Heath
IPSWICH
Suffolk IP5 3RE
England

Phone: +44 1473 640756
Fax: +44 1473 640709
EMail: george@gideon.bt.co.uk

Praveen Akkiraju
cisco Systems
170 West Tasman Drive
San Jose, CA 95134 USA

Phone: +1 (408) 526-5066
EMail: spa@cisco.com

Andy Heffernan
Juniper Networks, Inc.
385 Ravensdale Drive.
Mountain View, CA 94043 USA

Phone: +1 (650) 526-8037
Fax: +1 (650) 526-8001
EMail: ahh@juniper.net

Full Copyright Statement

Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.