DHC Working Group M. Stapp
Internet-Draft Cisco Systems, Inc.
Expires: May 2, 2003 November 1, 2002

Resolution of DNS Name Conflicts Among DHCP Clients
<draft-ietf-dhc-ddns-resolution-05.txt>

Status of this Memo

Copyright Notice

Abstract

DHCP provides a powerful mechanism for IP host configuration. However, the configuration capability provided by DHCP does not include updating DNS(RFC1034[1], RFC1035[2]), and specifically updating the name to address and address to name mappings maintained in the DNS.

The "Client FQDN Option"[3] specifies the client FQDN option, through which DHCP clients and servers can exchange information about client FQDNs. This document describes techniques for the resolution of DNS name conflicts among DHCP clients.

Table of Contents

1. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119[4].

2. Introduction

"The Client FQDN Option"[3] includes a description of the operation of DHCP[5] clients and servers that use the client FQDN option. Through the use of the client FQDN option, DHCP clients and servers can negotiate the client's FQDN and the allocation of responsibility for updating the DHCP client's A RR. This document identifies situations in which conflicts in the use of FQDNs may arise among DHCP clients, and describes a strategy for the use of the DHCID DNS resource record[6] in resolving those conflicts.

In any case, whether a site permits all, some, or no DHCP servers and clients to perform DNS updates into the zones that it controls is entirely a matter of local administrative policy. This document does not require any specific administrative policy, and does not propose one. The range of possible policies is very broad, from sites where only the DHCP servers have been given credentials that the DNS servers will accept, to sites where each individual DHCP client has been configured with credentials that allow the client to modify its own domain name. Compliant implementations MAY support some or all of these possibilities. Furthermore, this specification applies only to DHCP client and server processes: it does not apply to other processes that initiate DNS updates.

3. Issues with DNS Update in DHCP Environments

There are two DNS update situations that require special consideration in DHCP environments: cases where more than one DHCP client has been configured with the same FQDN, and cases where more than one DHCP server has been given authority to perform DNS updates in a zone. In these cases, it is possible for DNS records to be modified in inconsistent ways unless the updaters have a mechanism that allows them to detect anomolous situations. If DNS updaters can detect these situations, site administrators can configure the updaters' behavior so that the site's policies can be enforced. We use the term "Name Conflict" to refer to cases where more than one DHCP client wishes to be associated with a single FQDN. This specification describes a mechanism designed to allow updaters to detect these situations, and suggests that DHCP implementations use this mechanism by default.

3.1 Client Mis-Configuration

At many (though not all) sites, administrators wish to maintain a one-to-one relationship between active DHCP clients and domain names, and to maintain consistency between a host's A and PTR RRs. Hosts that are not represented in the DNS, or hosts which inadvertently share an FQDN with another host may encounter inconsistent behavior or may not be able to obtain access to network resources. Whether each DHCP client is configured with a domain name by its administrator or whether the DHCP server is configured to distribute the clients' names, the consistency of the DNS data is entirely dependent on the accuracy of the configuration procedure. Sites that deploy Secure DNS[9] may configure credentials for each host and its assigned name in a way that is more error-resistant, but this level of pre-configuration is still rare in DHCP environments.

Consider an example in which two DHCP clients in the "org.nil" network are both configured with the name "foo". The clients are permitted to perform their own DNS updates. The first client, client A, is configured via DHCP. It adds an A RR to "foo.org.nil", and its DHCP server adds a PTR RR corresponding to its IP address lease. When the second client, client B, boots, it is also configured via DHCP, and it also begins to update "foo.org.nil".

At this point, the "org.nil" administrators may wish to establish some policy about DHCP clients' DNS names. If the policy is that each client that boots should replace any existing A RR that matches its name, Client B can proceed, though Client A may encounter problems. In this example, Client B replaces the A RR associated with "foo.org.nil". Client A must have some way to recognize that the RR associated with "foo.org.nil" now contains information for Client B, so that it can avoid modifying the RR. When Client A's lease expires, for example, it should not remove an RR that reflects Client B's DHCP lease.

If the policy is that the first DHCP client with a given name should be the only client associated with that name, Client B needs to be able to determine that it is not the client associated with "foo.org.nil". It could be that Client A booted first, and that Client B should choose another name. Or it could be that B has booted on a new subnet, and received a new lease. It must either retain persistent state about the last lease it held (in addition to its current lease) or it must have some other way to detect that it was the last updater of "foo.org.nil" in order to implement the site's policy.

3.2 Multiple DHCP Servers

At many sites, the difficulties with distributing DNS update credentials to all of the DHCP clients lead to the desire for the DHCP servers to perform A RR updates on behalf of their clients. If a single DHCP server managed all of the DHCP clients at a site, it could maintain some database of the DNS names that it was managing, and check that database before initiating a DNS update for a client. Such a database is necessarily proprietary, however, and that approach does not work once more than one DHCP server is deployed.

Consider an example in which DHCP Client A boots, obtains a DHCP lease from Server S1, presenting the hostname "foo" in a Client FQDN option[3] in its DHCPREQUEST message. Server S1 updates its domain name, "foo.org.nil", adding an A RR that matches Client A's lease. The client then moves to another subnet, served by Server S2. When Client A boots on the new subnet, Server S2 will issue it a new lease, and will attempt to add an A RR matching the new lease to "foo.org.nil". At this point, without some communication mechanism which S2 can use to ask S1 (and every other DHCP server that updates the zone) about the client, S2 has no way to know whether Client A is currently associated with the domain name, or whether A is a different client configured with the same hostname. If the servers cannot distinguish between these situations, they cannot enforce the site's naming policies.

4. Use of the DHCID RR

A solution to both of these problems is for the updater (a DHCP client or DHCP server) to be able to determine which DHCP client has been associated with a DNS name, in order to offer administrators the opportunity to configure updater behavior.

For this purpose, a DHCID RR, specified in [6], is used to associate client identification information with a DNS name and the A or PTR RR associated with that name. When either a client or server adds an A or PTR RR for a client, it also adds a DHCID RR that specifies a unique client identity, based on data from the client's DHCPREQUEST message. In this model, only one A RR is associated with a given DNS name at a time.

By associating this ownership information with each DNS name, cooperating DNS updaters may determine whether their client is currently associated with a particular DNS name and implement the appropriately configured administrative policy. In addition, DHCP clients which currently have domain names may move from one DHCP server to another without losing their DNS names.

The specific algorithms utilizing the DHCID RR to signal client

ownership are explained below. The algorithms only work in the case where the updating entities all cooperate -- this approach is advisory only and is not a substitute for DNS security, nor is it replaced by DNS security.

5. DNS RR TTLs

RRs associated with DHCP clients may be more volatile than statically configured RRs. DHCP clients and servers that perform dynamic updates should attempt to specify resource record TTLs which reflect this volatility, in order to minimize the possibility that there will be stale records in resolvers' caches.

A reasonable basis for RR TTLs is the lease duration itself. The RR TTL on a DNS record added for with a DHCP lease SHOULD NOT exceed 1/3 of the lease time, and SHOULD be at least 10 minutes. We recognize that individual administrators will have varying requirements: DHCP servers and clients SHOULD allow administrators to configure TTLs, either as an absolute time interval or as a percentage of the lease time. In general, the TTLs or RRs added as a result of DHCP lease activity SHOULD be less than the initial lease time.

6. Procedures for performing DNS updates

6.1 Adding A RRs to DNS

When a DHCP client or server intends to update an A RR, it first prepares a DNS UPDATE query that includes as a prerequisite the assertion that the name does not exist. The update section of the query attempts to add the new name and its IP address mapping (an A RR), and the DHCID RR with its unique client-identity.

If this update operation succeeds, the updater can conclude that it has added a new name whose only RRs are the A and DHCID RR records. The A RR update is now complete (and a client updater is finished, while a server might proceed to perform a PTR RR update).

If the first update operation fails with YXDOMAIN, the updater can conclude that the intended name is in use. The updater then attempts to confirm that the DNS name is not being used by some other host. The updater prepares a second UPDATE query in which the prerequisite is that the desired name has attached to it a DHCID RR whose contents match the client identity. The update section of this query deletes the existing A records on the name, and adds the A record that matches the DHCP binding and the DHCID RR with the client identity.

If this query succeeds, the updater can conclude that the current

client was the last client associated with the domain name, and that the name now contains the updated A RR. The A RR update is now complete (and a client updater is finished, while a server would then proceed to perform a PTR RR update).

If the second query fails with NXRRSET, the updater must conclude that the client's desired name is in use by another host. At this juncture, the updater can decide (based on some administrative configuration outside of the scope of this document) whether to let the existing owner of the name keep that name, and to (possibly) perform some name disambiguation operation on behalf of the current client, or to replace the RRs on the name with RRs that represent the current client. If the configured policy allows replacement of existing records, the updater submits a query that deletes the existing A RR and the existing DHCID RR, adding A and DHCID RRs that represent the IP address and client-identity of the new client.

DISCUSSION:
The updating entity may be configured to allow the existing DNS records on the domain name to remain unchanged, and to perform disambiguation on the name of the current client in order to attempt to generate a similar but unique name for the current client. In this case, once another candidate name has been generated, the updater should restart the process of adding an A RR as specified in this section.

6.2 Adding PTR RR Entries to DNS

The DHCP server submits a DNS query that deletes all of the PTR RRs associated with the lease IP address, and adds a PTR RR whose data is the client's (possibly disambiguated) host name. The server MAY also add a DHCID RR as specified in Section 4.

6.3 Removing Entries from DNS

The most important consideration in removing DNS entries is be sure that an entity removing a DNS entry is only removing an entry that it added, or for which an administrator has explicitly assigned it responsibility.

When a lease expires or a DHCP client issues a DHCPRELEASE request, the DHCP server SHOULD delete the PTR RR that matches the DHCP binding, if one was successfully added. The server's update query SHOULD assert that the name in the PTR record matches the name of the client whose lease has expired or been released.

The entity chosen to handle the A record for this client (either the client or the server) SHOULD delete the A record that was added when

the lease was made to the client.

In order to perform this delete, the updater prepares an UPDATE query that contains two prerequisites. The first prerequisite asserts that the DHCID RR exists whose data is the client identity described in Section 4. The second prerequisite asserts that the data in the A RR contains the IP address of the lease that has expired or been released.

If the query fails, the updater MUST NOT delete the DNS name. It may be that the client whose lease on has expired has moved to another network and obtained a lease from a different server, which has caused the client's A RR to be replaced. It may also be that some other client has been configured with a name that matches the name of the DHCP client, and the policy was that the last client to specify the name would get the name. In these cases, the DHCID RR will no longer match the updater's notion of the client-identity of the host pointed to by the DNS name.

6.4 Updating Other RRs

The procedures described in this document only cover updates to the A and PTR RRs. Updating other types of RRs is outside the scope of this document.

7. Security Considerations

Unauthenticated updates to the DNS can lead to tremendous confusion, through malicious attack or through inadvertent misconfiguration. Administrators should be wary of permitting unsecured DNS updates to zones that are exposed to the global Internet. Both DHCP clients and servers SHOULD use some form of update request authentication (e.g., TSIG[12]) when performing DNS updates.

Whether a DHCP client may be responsible for updating an FQDN to IP address mapping, or whether this is the responsibility of the DHCP server is a site-local matter. The choice between the two alternatives may be based on the security model that is used with the Dynamic DNS Update protocol (e.g., only a client may have sufficient credentials to perform updates to the FQDN to IP address mapping for its FQDN).

Whether a DHCP server is always responsible for updating the FQDN to IP address mapping (in addition to updating the IP to FQDN mapping), regardless of the wishes of an individual DHCP client, is also a site-local matter. The choice between the two alternatives may be based on the security model that is being used with dynamic DNS updates. In cases where a DHCP server is performing DNS updates on behalf of a client, the DHCP server should be sure of the DNS name

to use for the client, and of the identity of the client.

Currently, it is difficult for DHCP servers to develop much confidence in the identities of their clients, given the absence of entity authentication from the DHCP protocol itself. There are many ways for a DHCP server to develop a DNS name to use for a client, but only in certain relatively rare circumstances will the DHCP server know for certain the identity of the client. If DHCP Authentication[13] becomes widely deployed this may become more customary.

One example of a situation that offers some extra assurances is one where the DHCP client is connected to a network through an MCNS cable modem, and the CMTS (head-end) of the cable modem ensures that MAC address spoofing simply does not occur. Another example of a configuration that might be trusted is one where clients obtain network access via a network access server using PPP. The NAS itself might be obtaining IP addresses via DHCP, encoding a client identification into the DHCP client-id option. In this case, the network access server as well as the DHCP server might be operating within a trusted environment, in which case the DHCP server could be configured to trust that the user authentication and authorization processing of the remote access server was sufficient, and would therefore trust the client identification encoded within the DHCP client-id.

8. Acknowledgements

Many thanks to Mark Beyer, Jim Bound, Ralph Droms, Robert Elz, Peter Ford, Olafur Gudmundsson, Edie Gunter, Andreas Gustafsson, R. Barr Hibbs, Kim Kinnear, Stuart Kwan, Ted Lemon, Ed Lewis, Michael Lewis, Josh Littlefield, Michael Patton, Glenn Stump, and Bernie Volz for their review and comments.

References

[1] Mockapetris, P., "Domain names - Concepts and Facilities", RFC 1034, Nov 1987.

[2] Mockapetris, P., "Domain names - Implementation and Specification", RFC 1035, Nov 1987.

[3] Stapp, M. and Y. Rekhter, "The DHCP Client FQDN Option (draft-ietf-dhc-fqdn-option-*.txt)", March 2001.

[4] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", RFC 2119, March 1997.

[5] Droms, R., "Dynamic Host Configuration Protocol", RFC 2131,

March 1997.

[6] Stapp, M., Gustafsson, A. and T. Lemon, "A DNS RR for Encoding DHCP Information (draft-ietf-dnsext-dhcid-rr-*)", March 2001.

[7] Marine, A., Reynolds, J. and G. Malkin, "FYI on Questions and Answers to Commonly asked ``New Internet User'' Questions", RFC 1594, March 1994.

[8] Vixie, P., Thomson, S., Rekhter, Y. and J. Bound, "Dynamic Updates in the Domain Name System", RFC 2136, April 1997.

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

[10] Wellington, B., "Secure Domain Name System (DNS) Dynamic Update", RFC 3007, November 2000.

[11] Rivest, R., "The MD5 Message Digest Algorithm", RFC 1321, April 1992.

[12] Vixie, P., Gudmundsson, O., Eastlake, D. and B. Wellington, "Secret Key Transaction Authentication for DNS (TSIG)", RFC 2845, May 2000.

[13] Droms, R. and W. Arbaugh, "Authentication for DHCP Messages", RFC 3118, June 2001.

Author's Address

Mark Stapp
Cisco Systems, Inc.
250 Apollo Dr.
Chelmsford, MA 01824
USA

Phone: 978.244.8498
EMail: mjs@cisco.com

Full Copyright Statement

Acknowledgement

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