DHC A. Kostur
Internet-Draft Incognito
Intended status: Standards Track July 04, 2014
Expires: January 03, 2015

Load Balancing for DHCPv6
draft-ietf-dhc-dhcpv6-load-balancing-02

Abstract

This document proposes a method of algorithmic load balancing for IPv6 Dynamic Host Configuration Protocol (DHCPv6) traffic. It enables multiple, cooperating servers to decide which one should service a client, without necessarily exchanging any information between the servers. The server selection is based on the servers hashing client DHCP Unique Identifiers (DUIDs) when multiple DHCPv6 servers are available to service DHCPv6 clients. The proposed technique provides for efficient server selection when multiple DHCPv6 servers offer services on a network without requiring any changes to existing DHCPv6 clients. This algorithm is an extension of an already defined and proven algorithm used for DHCPv4, as described in RFC 3074.

Status of this Memo

Copyright Notice

Table of Contents

1. Introduction

This document is intended to extend the algorithm described in DHC Load Balancing Algorithm [RFC3074] to apply to DHCPv6 [RFC3315] traffic. Most of the terminology and procedures are identical to the ones specified in RFC 3074. As a short summary: servers which are participating in load balancing calculate hash values for the Service Transaction ID (STID) based on client-specific values (the client DUID for DHCPv6, the Client ID or CHADDR field for DHCPv4) for each incoming UDP packet. This hash is then used to select a hash bucket. Servers are assigned to service particular buckets.

Load balancing is not the same as failover, as load balancing is not attempting to address any redundancy concerns [RFC6853]. Load balancing does not attempt to address the issues of configuration or data synchronization between DHCPv6 servers. However, load balancing

may be desirable in a failover set of servers in order to reduce the load on the servers in normal operations, and certain desirable behaviors can occur if load balancing is aware that data synchronization is occurring.

1.1. Requirements Language

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 [RFC2119].

2. Background and External Requirements

The requirements for DHCPv6 are substantially the same as for DHCPv4, replacing DHCPDISCOVER with SOLICIT, DHCPREQUEST with REQUEST, CONFIRM, RENEW, or REBIND (as appropriate), etc.

3. Operation

A DHCPv6 server performing this load balancing will operate in substantially the same manner as if it were a DHCPv4 server load balancing an incoming DHCPv4/BOOTP packet with the following differences.

Load balancing only applies to incoming client-originated UDP DHCPv6 messages. RELAY-FORWs are processed based on the content of the most encapsulated packet (ie: the client-originated DHCPv6 message). Future message types will have to be considered as they are proposed as to how they may be load balanced.

LEASEQUERY [RFC5007] messages MUST NOT be load balanced. Devices which are sending LEASEQUERY packets will have to be sending those packets to all of the load balanced DHCPv6 servers, and the LEASEQUERY specification already considers what the device should do when it receives responses to the LEASEQUERY from multiple sources.

DHCPV4-QUERY [I-D.ietf-dhc-dhcpv4-over-dhcpv6] messages SHOULD NOT be load balanced, but SHOULD be subject to DHCPv4 load balancing, if the server supports it.

ADVERTISE, REPLY, RELAY-REPL, LEASEQUERY-REPLY, and RECONFIGURE-REPLY [RFC6977] are messages which should not be received by a DHCPv6 server and thus are not considered in this document.

3.1. Messages with a Server Identifier

Messages which contain a Server Identifier to direct that message to a specific server SHOULD be processed as if load balancing were not in play, with the exception of RENEWs.

3.2. RENEWs with the DHCPv6 servers sharing lease information

A DHCPv6 server receiving a RENEW with the server's Server Identifier specified MAY choose to ignore the request if the load balancing algorithm decides that this server should not process this message. Let us assume the following sequence of events:

1. There is a pair of DHCPv6 servers that are known to be exchanging lease information with each other

2. The first server fails and is no longer servicing DHCPv6 clients

3. Some number of DHCPv6 clients are bound to the second DHCPv6 server (whether by performing a SOLCIT-ADVERTISE-REQUEST-REPLY sequence, or by REBINDing to the second server)

4. The first server is restored to service and is able to service DHCPv6 clients

At this point, a disproportionate set of DHCPv6 clients are now bound to the second DHCPv6 server. If the second DHCPv6 server is permitted to ignore the RENEW even though the Server Identifier would indicate that it should respond, then the clients which should be answered by the first server will get no response to the RENEW that contains the second server's Server Identifier and will perform the normal retry mechanisms. At some point the client will transition into the REBIND state and will attempt to REBIND. That REBIND will not have a Server Identifier and will be received by both DHCPv6 servers. Since the servers were exchanging lease information, the first DHCPv6 server would have sufficient information to be able to REPLY to the client to extend the lease and those clients would now be bound to the first DHCPv6 server again. Over time this would result in the DHCPv6 population being rebalanced.

3.3. RENEWs with the DHCPv6 servers not sharing lease information

A DHCPv6 server receiving a RENEW with the server's Server Identifier specified SHOULD be processed as if load balancing were not in play. If the server ignored these RENEWs, the requesting device would eventually transition to REBIND, and the other servers may not have any lease information to answer the REBIND with, forcing the client to eventually drop its lease and start again from SOLICIT.

3.4. Selecting the STID

DHCPv6 servers MUST use the client's DUID in its entirety as the STID. This is different than RFC 3074 which limited the STID to 16 bytes.

An INFORMATION-REQUEST may have no client DUID in the message. Calculate the hash as if a 0-length DUID were supplied, effectively assigning those messages to hash bucket 0.

3.5. Replacing the secs field

A DHCPv6 server providing the capability of Delayed Service SHOULD use the value in the OPTION_ELAPSED_TIME wherever RFC 3074 makes reference to the secs field.

4. Acknowledgements

Thanks to Bernie Volz, Steve Gonczi, Ted Lemon, and Rob Stevens as this document heavily borrows from their previous work on RFC 3074, as well as Bernie and Tomek Mrugalski's additional comments during the discussions.

5. IANA Considerations

This memo includes no request to IANA.

6. Security Considerations

This proposal in and by itself provides no security, nor does it impact existing security. Servers using this algorithm are responsible for ensuring that if the contents of the hash bucket assignments are transmitted over the network as part of the process of configuring any server, that message be secured against tampering, since tampering with the HBA could result in denial of service for some or all clients.

If the hash bucket assignments are not configured such that each bucket is assigned to one and only one DHCP server, this results in some clients that will be either completely ignored by the DHCP servers (if no server is configured to answer that hash bucket), or get multiple responses (if more than one server is configured to answer that hash bucket).

7. References

7.1. Normative References

[I-D.ietf-dhc-dhcpv4-over-dhcpv6]
Sun, Q., Cui, Y., Siodelski, M., Krishnan, S. and I. Farrer, "DHCPv4 over DHCPv6 Transport", Internet-Draft draft-ietf-dhc-dhcpv4-over-dhcpv6-05, February 2014.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC3074] Volz, B., Gonczi, S., Lemon, T. and R. Stevens, "DHC Load Balancing Algorithm", RFC 3074, February 2001.

[RFC3315] Droms, R., Bound, J., Volz, B., Lemon, T., Perkins, C. and M. Carney, "Dynamic Host Configuration Protocol for IPv6

(DHCPv6)", RFC 3315, July 2003.

[RFC5007] Brzozowski, J., Kinnear, K., Volz, B. and S. Zeng, "DHCPv6 Leasequery", RFC 5007, September 2007.

[RFC6977] Boucadair, M. and X. Pougnard, "Triggering DHCPv6 Reconfiguration from Relay Agents", RFC 6977, July 2013.

7.2. Informative References

[RFC6853] Brzozowski, J., Tremblay, J., Chen, J. and T. Mrugalski, "DHCPv6 Redundancy Deployment Considerations", BCP 180, RFC 6853, February 2013.

Author's Address

Andre Kostur
Incognito Software Inc.
Suite 500 - 375 Water St.
Vancouver, BC V6B 5C6
CA

Phone: +1 604 678 2864
Email: akostur@incognito.com