DHC WG
Internet-Draft
Intended status: Informational
Expires: August 15, 2013

B. Rajtar
Hrvatski Telekom
I. Farrer
Deutsche Telekom AG

February 11, 2013

Provisioning IPv4 Configuration Over IPv6 Only Networks
draft-rajtar-dhc-v4configuration-00

Abstract

As IPv6 becomes more widely deployed, some service providers are taking the approach of deploying IPv6 only networks, without dual-stack functionality for IPv4. However, access to IPv4 based services is still an ongoing requirement and approaches such as IPv4-in-IPv6 softwire tunnels are being developed to meet this need.

In order to provision end-user's hosts with the necessary IPv4 configuration, a number of different mechanisms have been proposed. This memo discusses the benefits and drawbacks of each and recommend a single approach to use for the basis for future work.

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].

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on August 15, 2013.

Copyright Notice

Table of Contents

1. Introduction

A service provider with an IPv6-only core network must also be able to provide customers with access to the Internet and other services over IPv4. Softwire based IPv4-in-IPv6 tunneling mechanisms are an obvious example of this, such as the ones described in: [I-D.cui-softwire-b4-translated-ds-lite], [I-D.ietf-softwire-map] and [I-D.bfmk-softwire-unified-cpe].

A general trend here is to distribute NAT functionality and IPv4 address sharing from the centralized tunnel concentrator to the CPE in order to achieve better scalability. This results in a number of configuration parameters needing to be provisioned to the CPE such as the external public IPv4 address and a restricted port-range to use for NAT.

In order to configure customer's devices for softwire function, a dynamic provisioning mechanism is necessary. In IPv4 only networks, DHCPv4 has often been used to provide configuration, but in an IPv6 only network, DHCPv4 messages cannot be transported.

This document compares three different approaches which have been proposed for resolving this problem.

2. Approaches for Configuring IPv4 Parameters

In order to resolve the problem described above, the following approaches for transporting IPv4 configuration parameters have been suggested:

1. Adapt DHCPv4 format messages to be transported over IPv6 as described in [I-D.ietf-dhc-dhcpv4-over-ipv6]. For brevity, this is referred to as DHCPv4o6.

2. Extend DHCPv6 with new options for IPv4 configuration, such as

[I-D.mdt-softwire-map-dhcp-option] describes.

3. Use DHCPv6 as above for external IPv4 address and source port configuration. Use DHCPv4 over IPv4 messages within an IPv6 softwire for configuring additional parameters. For brevity, this is referred to as DHCPv4oSW.

At the time of writing, working examples of the first two approaches have been developed and successfully tested in several different operators networks. The third approach is still only theoretical.

Each of these approaches are described in more detail underneath.

2.1. DHCPv4o6 Based Provisioning - Functional Overview

In order to receive IPv4 configuration parameters, IPv4-only clients initiate and exchange DHCPv4 messages with the DHCPv4 server. In order adapt this to an IPv6-only network, an existing DHCPv4 client implements a 'Client Relay' (CRA) function, which takes DHCPv4 messages and puts them into UDPv6 and IPv6.

As the mechanism involves unicast based communications, the IPv6 address of the server must be provisioned to the client. A new DHCPv6 option has been defined for this purpose.

The DHCPv4o6 server must either provide an IPv6 interface to the client, or an intermediary 'Transport Relay Agent' device can act as the gateway between the IPv4 and IPv6 domains.

The DHCPv4o6 server needs to be extended to support the new functionality, such as storing the IPv6 address of DHCPv4o6 clients.

This approach currently uses functional elements for ingress and egress of the IPv6-only transport domain--the CRA on the host and the TRA or TSV on the server. As a result, this approach has sometimes been referred to as a tunneling approach. However, relay agent encapsulation is not a tunnel, since it carries only DHCP traffic; it would be more accurate to describe it as an encapsulation.

It is worth noting that there is no technical reason for using relay encapsulation for DHCPv4o6; this approach was taken because the authors of the draft originally imagined that it might be used to

provide configuration information for an unmodified DHCPv4 client. However, this turns out not to be a viable approach: in order for this to work, there would have to be IPv4 routing on the local link to which the client is connected. In that case, there's no need for DHCPv4o6.

Given that this is the case, there is no technical reason why DHCPv4o6 can't simply use the IPv6 transport directly, without any relay encapsulation. This would greatly simplify the specification and the implementation, and would still address the requirements stated in this document.

This solution is described in detail in [I-D.ietf-dhc-dhcpv4-over-ipv6].

2.2. DHCPv6 Based Provisioning - Functional Overview

In this approach, DHCPv6 would be extended with new DHCPv6 options for configuring IPv4 based functions.

An example of this approach is described in [I-D.mdt-softwire-map-dhcp-option], where a DHCPv6 message is used to convey parameters necessary for IPv4 in IPv6 softwire configuration.

2.3. DHCPv4oSW Based Provisioning - Functional Overview

In this approach, the configuration of IPv4 address and source ports (if required) is carried out as described in section 2.2 above. Additional IPv4 configuration parameters are then provisioned using a DHCPv4 messages transported within IPv6 in the softwire in the same manner as any other IPv4 based traffic.

On receipt at the tunnel concentrator (e.g. MAP Border Router or a Lightweight 4over6 lwAFTR), the DHCPv4 message removed from the softwire and forwarded to the DHCPv4 server in the same way as any other IPv4 packet is handled.

As the client is already configured with its external IPv4 address and source ports, the messages exchanged between the DHCPv4 client and server would be strictly DHCPINFORM/DHCPACK messages, for the configuration of additional IPv4 parameters.

For this approach to function, a mechanism for the DHCPv4 client to learn the IPv4 address of the DHCPv4 server is needed. This could be done by defining a well-known IPv4 address for the DHCPv4 server, implementing a DHCPv4 relay function within the tunnel concentrator or other configuration methods.

From a transport perspective, the key difference between this method and DHCPv4o6 (described above) is that here, the DHCPv4 message is put into UDPv4 and IPv4 and then put into the IPv6 softwire, instead of directly placing the DHCPv4 message into UDPv6 and IPv6.

3. Comparison of the Three Approaches

The following section of the document provides the pros and cons of the approaches.

3.1. DHCPv4o6 Based Provisioning

3.1.1. Pros

1. Once implemented, all existing DHCPv4 options will be be available with no further ongoing development work necessary.

2. IPv4 and IPv6 based provisioning can be separated from each other if required, allowing flexibility in network design.

3. Easy to implement through minor adaptation of existing DHCPv4 client/server code.

4. Simple, in that no additional functional elements are necessary except the DHCPv4o6 client and server. The Transport Relay Agent is completely optional.

3.1.2. Cons

1. More complex, in that there are more new functional elements within the architecture than are necessary in DHCPv6 based provisioning.

2. A new DHCPv6 option is necessary in order to provision the IPv6 address of the DHCPv4 server to the end device.

3. DHCPv4 clients needs to be updated to implement the IPv6 encapsulation and decapsulation function.

4. The DHCPv4 server needs to be updated to implement new DHCPv4o6 functionality.

3.2. DHCPv6 Based Provisioning

3.2.1. Pros

1. Simpler, in that no additional functional elements are required except the DHCPv6 client and server.

2. A single protocol is used to deliver configuration information for IPv4 and IPv6.

3.2.2. Cons

1. Any required DHCPv4 options must be ported to DHCPv6, which will require a large amount of re-development work. All functional elements in the DHCPv6 implementation (clients, servers, relays) would need to be updated for each change.

2. Means that DHCPv4 'legacy' options, which will be of decreasing relevance in the future will remain in DHCPv6 for the lifetime of the protocol.

3. Each time that a DHCPv4 option is ported to DHCPv6, all clients and servers would need to be updated to implement the new option.

4. Does not provide an architecture for keeping IPv4 and IPv6 domains separated.

3.3. DHCPv4oSW Based Provisioning

3.3.1. Pros

1. Once implemented, all existing DHCPv4 options will be be available with no further ongoing development work necessary.

2. Uses the existing DHCPv4 and DHCPv6 architectures in order to

provide IPv4 configuration in an IPv6 only environment.

3. DHCPv4 and DHCPv6 based provisioning can be separated from each other if required, allowing flexibility in network design.

3.3.2. Cons

1. More complex, in that there are more new functional elements within the architecture than are necessary in DHCPv6 based provisioning.

2. IPv4 over IPv6 softwire approaches which distribute NAT to the CPE and allow for IP address sharing (MAP-E & LW4o6) forbid the use of reserved TCP/UDP ports (e.g. 0-1024). Every DHCPv4 client sharing the same address needs to have a UDP listener running on UDP port 68. To resolve this would require significant rework to either the softwire mechanisms and/or the DHCPv4 client implementation.

3. From the current specification, DHCPINFORM is not suitable for use over a softwire. Additional work, such as the development of 'shims' would be necessary

4. The current DHCPINFORM specification has a number of unclear points, such as those described in [I-D.ietf-dhc-dhcpinform-clarify]. Substantial work would be required to resolve this.

5. Links the deployment of IPv4 configuration over IPv6 to a softwire implementation (e.g. requiring a softwire concentrator to act as a DHCPv4 relay). Whilst softwires are the only application for this functionality at the moment, this may not always be the case.

6. A new mechanism must be defined in order to provide the DHCPv4 client with the IPv4 address of the DHCPv4 server so that unicast DHCPINFORM messages can be sent.

4. Conclusion

Whilst all of the approaches described here will require some development work in order to realise, it is clear from the above analysis that the most sustainable approaches capitalise on existing DHCPv4 implementations and wrap these within IPv6. The main rationale for this is that it enables all of DHCPv4's existing options to be migrated for use over IPv6 in a single step.

The alternative would require ongoing development work, re-implementing existing DHCPv4 functionality in DHCPv6. This will result in having legacy DHCPv4 options in DHCPv6, which will no longer be useful once IPv4 is completely abandoned.

Of the two methods which support the reuse of existing DHCPv4 implementations (DHCPv4o6 and DHCPv4oSW), DHCPv4oSW may seem to be the simpler of the two solutions. However, as outlined in section 3.3.2 above, there are some significant problems with its design and implementation that would require substantial development and standardisation work to resolve. Although DHCPv4o6 also requires additional development work, the technical barriers for realisation are much simpler to resolve.

Therefore, the DHC working group recommends that DHCPv4o6 is the best underlying approach for provisioning IPv4 parameters over an IPv6 only network. It further recommends that the modifications described in section 2.1 are made to simplify the current DHCPv4o6 draft.

5. IANA Considerations

This document makes no request of IANA.

Note to RFC Editor: this section may be removed on publication as an RFC.

6. Security Considerations

7. Acknowledgements

8. References

8.1. Normative References

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

8.2. Informative References

[I-D.bfmk-softwire-unified-cpe]
Boucadair, M. and I. Farrer, "Unified IPv4-in-IPv6 Softwire CPE", draft-bfmk-softwire-unified-cpe-02 (work in progress), January 2013.

[I-D.cui-softwire-b4-translated-ds-lite]
Cui, Y., Sun, Q., Boucadair, M., Tsou, T., Lee, Y., and I. Farrer, "Lightweight 4over6: An Extension to the DS-Lite Architecture", draft-cui-softwire-b4-translated-ds-lite-09 (work in progress), October 2012.

[I-D.ietf-dhc-dhcpinform-clarify]
Hankins, D., "Dynamic Host Configuration Protocol DHCPINFORM Message Clarifications", draft-ietf-dhc-dhcpinform-clarify-06 (work in progress), October 2011.

[I-D.ietf-dhc-dhcpv4-over-ipv6]
Cui, Y., Wu, P., Wu, J., and T. Lemon, "DHCPv4 over IPv6 Transport", draft-ietf-dhc-dhcpv4-over-ipv6-05 (work in progress), September 2012.

[I-D.ietf-softwire-map]
Troan, O., Dec, W., Li, X., Bao, C., Matsushima, S., and T. Murakami, "Mapping of Address and Port with Encapsulation (MAP)", draft-ietf-softwire-map-04 (work in progress), February 2013.

[I-D.mdt-softwire-map-dhcp-option]
Mrugalski, T., Troan, O., Bao, C., and W. Dec, "DHCPv6 Options for Mapping of Address and Port", draft-mdt-softwire-map-dhcp-option-03 (work in progress), July 2012.

Authors' Addresses

Branimir Rajtar
Hrvatski Telekom
Zagreb
Croatia

Email: branimir.rajtar@t.ht.hr

Ian Farrer
Deutsche Telekom AG
Bonn
Germany

Email: ian.farrer@telekom.de