Independent Submission
Request for Comments: 7393
Category: Informational
ISSN: 2070-1721

X. Deng
M. Boucadair
France Telecom
Q. Zhao
Beijing University of Posts and Telecommunications
J. Huang
C. Zhou
Huawei Technologies

November 2014

Using the Port Control Protocol (PCP) to Update Dynamic DNS

Abstract

This document focuses on the problems encountered when using dynamic DNS in address-sharing contexts (e.g., Dual-Stack Lite (DS-Lite) and Network Address and Protocol Translation from IPv6 Clients to IPv4 Servers (NAT64)) during IPv6 transition. Both issues and possible solutions are documented in this memo.

Status of This Memo

This document is not an Internet Standards Track specification; it is published for informational purposes.

This is a contribution to the RFC Series, independently of any other RFC stream. The RFC Editor has chosen to publish this document at its discretion and makes no statement about its value for implementation or deployment. Documents approved for publication by the RFC Editor are not a candidate for any level of Internet Standard; see Section 2 of RFC 5741.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at http://www.rfc-editor.org/info/rfc7393.

Table of Contents

1. Introduction

1.1. Problem Statement

Dynamic DNS (DDNS) is a widely deployed service to facilitate hosting servers (e.g., access to a webcam, HTTP server, FTP server, etc.) at customers' premises. There are a number of providers that offer a DDNS service, working in a client and server mode, which mostly use web-form-based communication. DDNS clients are generally implemented in the user's router or computer; once changes are detected to its assigned IP address, an update message is automatically sent to the DDNS server. The communication between the DDNS client and the DDNS server is not standardized, varying from one provider to another, although a few standard web-based methods of updating have emerged over time.

In address-sharing contexts, well-known port numbers (e.g., port 80) won't be available for every user [RFC6269]. As such, the DDNS client will have to register the IP address and/or the external port(s) on which the service is listening. Also, the DDNS client has to report any change of this IP address and/or the external port(s). It will also require the ability to configure corresponding port forwarding on Carrier-Grade NAT (CGN) [RFC6888] devices so that incoming communications initiated from the Internet can be routed to the appropriate server behind the CGN.

Issues encountered in address sharing are documented in [RFC6269]. This document focuses on the problems encountered when using dynamic DNS in address-sharing contexts (e.g., DS-Lite [RFC6333] and NAT64 [RFC6146]). The main challenges are listed below:

Announce and discover an alternate service port: The DDNS service must be able to maintain an alternative port number instead of the default port number.

Allow for incoming connections: Appropriate means to instantiate port mappings in the address-sharing device must be supported.

Detect changes and trigger DDNS updates: The DDNS client must be triggered by the change of the external IP address and the port number. Concretely, upon change of the external IP address (and/or external port number), the DDNS client must refresh the DNS records; otherwise, the server won't be reachable from outside. This issue is exacerbated in the DS-Lite context because no public IPv4 address is assigned to the Customer Premises Equipment (CPE).

1.2. Scope and Goals

This document describes some candidate solutions to resolve the aforementioned issues with a particular focus on DS-Lite. These solutions may also be valid for other address-sharing schemes.

This document sketches deployment considerations based on the Port Control Protocol (PCP) [RFC6887]. Note that DDNS may be considered as an implementation of the rendezvous service mentioned in [RFC6887].

Indeed, after creating an explicit mapping for incoming connections using PCP, it is necessary to inform remote hosts about the IP address, protocol, and port number for the incoming connection to reach the services hosted behind a DS-Lite CGN. This is usually done in an application-specific manner. For example, a machine hosting a game server might use a rendezvous server specific to that game (or specific to that game developer), a SIP phone would use a SIP proxy, a client using DNS-Based Service Discovery [RFC6763] would use DNS Update [RFC2136][RFC3007], etc. PCP does not provide this rendezvous function.

The rendezvous function may support IPv4, IPv6, or both. Depending on that support and the application's support of IPv4 or IPv6, the PCP client may need an IPv4 mapping, an IPv6 mapping, or both. An example illustrating how the DDNS server may implement such a service notification functionality if necessary is provided in Section 3.

This document does not specify any protocol extension but instead focuses on the elaboration of the problem space and illustrates how existing tools can be reused to solve the problem for some deployment contexts. Particularly, this document requires no changes to PCP or dynamic updates in the standard domain name system [RFC2136]; rather, it is an operational document to make the current DDNS service providers aware of the impacts and issues that IPv6 transitioning and IPv4 address sharing will bring to them, and it gives solutions to address the forthcoming issues. The current DDNS service providers usually employ a web-based form to maintain DDNS service registration and updates.

Generic deployment considerations for DS-Lite, including Basic Bridging BroadBand (B4) remote management and IPv4 connectivity check, can be found in [RFC6908]. This document complements [RFC6908] with deployment considerations related to rendezvous service maintenance. Additional PCP-related deployment considerations are available at [PCP-DEPLOYMENT].

Solutions relying on DNS-Based Service Discovery [RFC6763] or Apple's Back to My Mac (BTMM) Service [RFC6281] are not considered in this document. Moreover, this document does not assume that DDNS service relies on [RFC2136].

IPv4 addresses used in the examples are derived from the IPv4 block reserved for documentation in [RFC6890]. DNS name examples follow [RFC2606].

2. Solution Space

2.1. Locate a Service Port

As listed below, at least two solutions can be used to associate a port number with a service:

1. Use service URIs (e.g., FTP, SIP, HTTP) that embed an explicit port number. Indeed, the Uniform Resource Identifier (URI) defined in [RFC3986] allows the port number to be carried in the syntax (e.g., mydomain.example:15687).

2. Use SRV records [RFC2782]. Unfortunately, the majority of browsers do not support this record type.

The DDNS client and DDNS server are to be updated so that an alternate port number is signaled and stored by the DDNS server. Requesting remote hosts will be then notified with the IP address and port number to reach the server.

2.2. Create Explicit Mappings for Incoming Connections

PCP is used to install the appropriate mapping(s) in the CGN so that incoming packets can be delivered to the appropriate server.

2.3. Detect Changes

In a network as described in Figure 1, a DDNS client/PCP client can be running on either a CPE or the host that is hosting some services itself. There are several possible ways to address the problems stated in Section 1.1:

1. If the DDNS client is enabled, the host periodically issues (e.g., 60 minutes) PCP MAP requests (e.g., messages 1 and 2 in Figure 1) with short lifetimes (e.g., 30s) for the purpose of inquiring an external IP address and setting. If the purpose is to detect any change to the external port, the host must issue a

PCP mapping to install for the internal server. Upon change of the external IP address, the DDNS client updates the records accordingly (e.g., message 3 in Figure 1).

2. If the DDNS client is enabled, it checks the local mapping table maintained by the PCP client. This process is repeated periodically (e.g., 5 minutes, 30 minutes, 60 minutes). If there is no PCP mapping created by the PCP client, it issues a PCP MAP request (e.g., messages 1 and 2 in Figure 1) for the purpose of inquiring an external IP address and setting up port forwarding mappings for incoming connections. Upon change of the external IP address, the DDNS client updates the records in the DDNS server, e.g., message 3 in Figure 1.

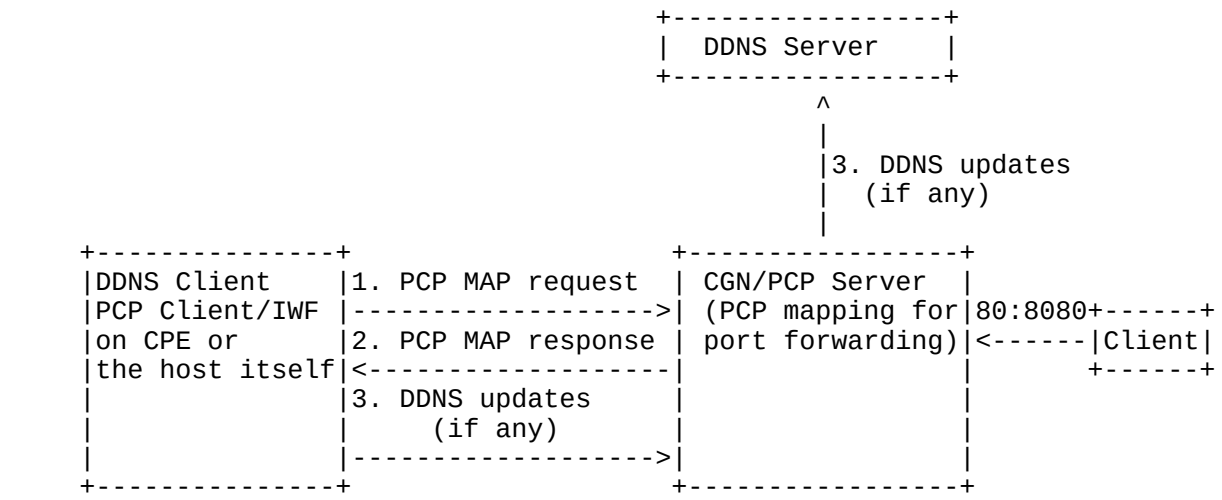

Figure 1: Flow Chart

3. Some Deployment Solutions

3.1. Reference Topology

Figure 2 illustrates the topology used for the deployment solutions elaborated in the following subsections.

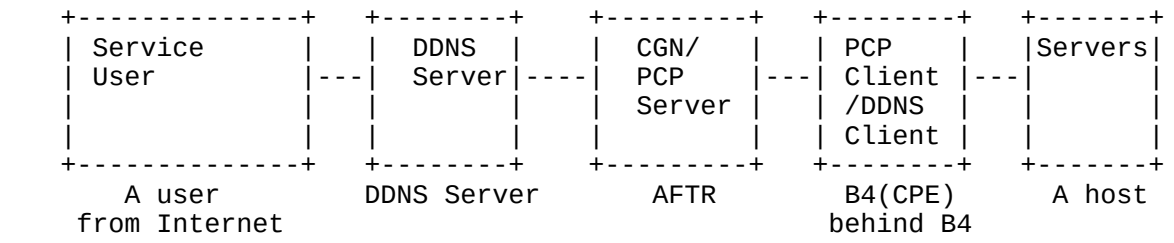

Figure 2: Implementation Topology

Figure 2 involves the following entities:

- Servers: Refers to the servers that are deployed in the DS-Lite network, or more generally, an IP address-sharing environment. They are usually running on a host that has been assigned with a private IPv4 address. Having created a proper mapping via PCP in the Address Family Transition Router (AFTR), these services have been made available to Internet users. The services may provide web, FTP, SIP, and other services though these may not be able to be seen as using a well-known port from the outside anymore, in the IP address-sharing context.

- B4(CPE): An endpoint of an IPv4-in-IPv6 tunnel [RFC6333]. A PCP client together with a DDNS client are running on it. After a PCP client establishes a mapping on the AFTR, an end user may register its domain name and its external IPv4 address plus port number to its DDNS service provider (DDNS server), manually or automatically by a DDNS client. Later, likewise, end users may manually announce or let the DDNS client automatically announce IP address and/or port changes to the DDNS server.

- AFTR: Responsible for maintaining mappings between an IPv6 address, the internal IPv4 address plus internal port, and the external IPv4 address plus port [RFC6333].

- DDNS server: Maintains a table that associates a registered domain name and a registered host’s external IPv4 address/port number pair. When being notified of IP address and port number changes from a DDNS client, the DDNS server announces the updates to DNS servers on behalf of the end user. [RFC2136] and [RFC3007] may be

used by DDNS servers to send updates to DNS servers. In many current practices, a DDNS service provider usually announces its own IP address as the registered domain names of end users. When HTTP requests reach the DDNS server, they may employ URL Forwarding or HTTP 301 redirection to redirect the request to a proper registered end user by looking up the maintained link table.

- o Service users: Refers to users who want to access services behind an IP address-sharing network. They issue standard DNS requests to locate the services, which will lead them to a DDNS server, provided that the requested services have been registered to a DDNS service provider. The DDNS server will then handle the rest in the same way as described before.

3.2. For Web Service

Current DDNS server implementations typically assume that the end servers host web servers on the default 80 port. In the DS-Lite context, they will have to take into account that external ports assigned by the AFTR may be any number other than 80, in order to maintain proper mapping between domain names and the external IP plus port. If a proper mapping is maintained, the HTTP request would be redirected to the AFTR, which serves the specific end host that is running the servers.

Figure 3 depicts how messages are handled in order to be delivered to the right server.

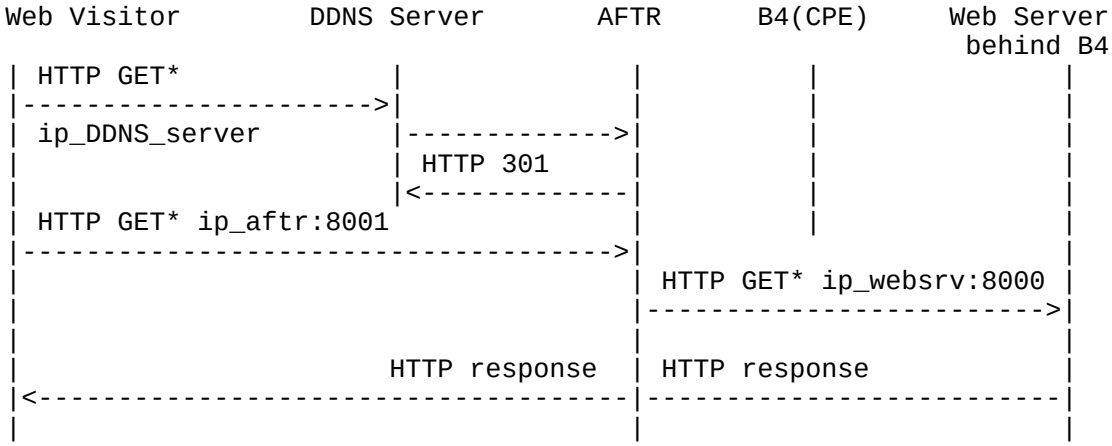

Figure 3: HTTP Service Messages

When a web user sends out an HTTP GET message to the DDNS server after a standard DNS query, the DDNS server redirects the request to a registered web server, in this case, by responding with an HTTP 301 message. Then, the HTTP GET message will be sent out to the AFTR, which will in turn find the proper hosts behind it. For simplicity, messages among AFTR, B4, and the web server behind B4 are not shown completely; for communications among those nodes, refer to [RFC6333].

3.3. For Non-web Service

For non-web services, as mentioned in Section 2, other means will be needed to inform the users about the service information.

[RFC6763] includes an example of a DNS-based solution that allows an application running in the end user’s device to retrieve service-related information via DNS SRV/TXT records and list available services. In a scenario where such an application is not applicable, the following provides another solution for a third party, e.g., a DDNS service provider, to disclose services to Internet users.

A web portal can be used to list available services. A DDNS server maintains a web portal for each user’s Fully Qualified Domain Name (FQDN), which provides service links to users. Figure 4 assumes "websrv.example.com" is a user’s FQDN provided by a DDNS service provider.

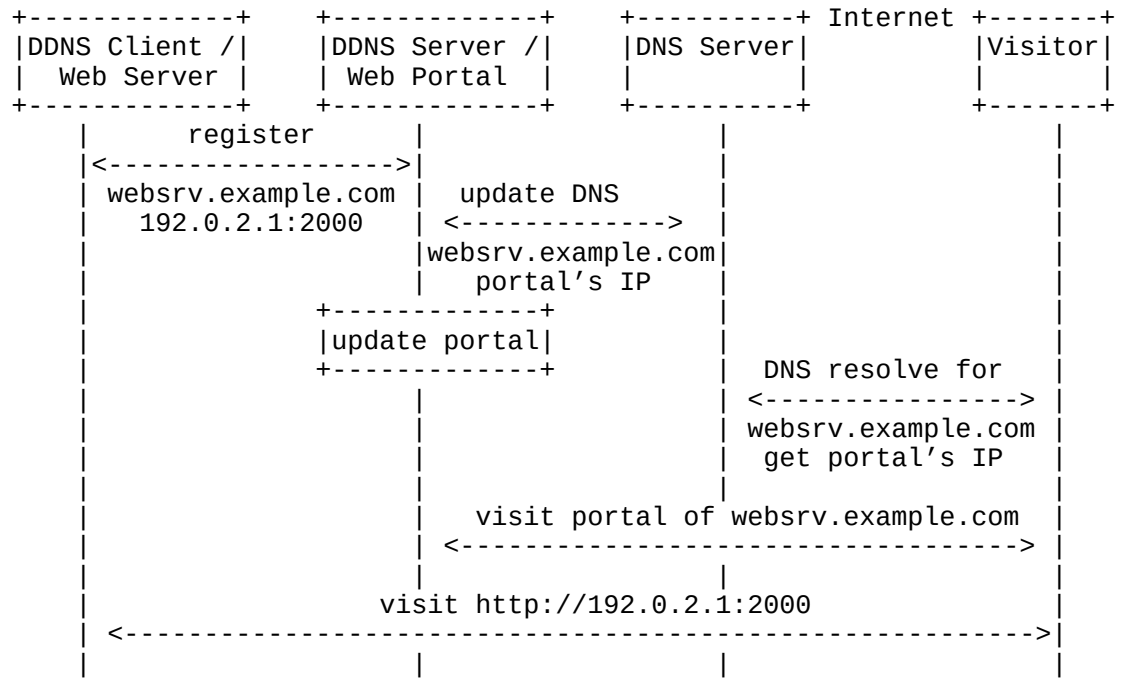

Figure 4: Update Web Portal

The DDNS client registers the server's information to the DDNS server, including the public IP address and port obtained via PCP, the user's FQDN, and other necessary information. The DDNS server also behaves as a portal server; it registers its IP address, port number, and the user's FQDN to the DNS system so that visitors can access the web portal.

A DDNS server also maintains a web portal for each user's FQDN and updates the portal according to registered information from the DDNS client. When a visitor accesses "websrv.example.com", a DNS query will resolve the portal server's address and port number, and the visitor will see the portal and the available services.

Figure 5: An Example of a Web Portal

As shown in Figure 5, the web portal shows the service URLs that are available to be accessed. Multiple services are accessible per a user's FQDN.

Some applications that are not HTTP based can also be delivered using this solution. When a user clicks on a link, the registered application in the client OS will be invoked to handle the link. How this can be achieved is out of the scope of this document.

4. Security Considerations

This document does not introduce a new protocol, nor does it specify protocol extensions. Security-related considerations related to PCP [RFC6887] and DS-Lite [RFC6333] should be taken into account.

The protocol between the DDNS client and DDNS server is proprietary in most cases; some extensions may be necessary, which is up to the DDNS operators. These operators should enforce security-related policies in order to keep illegitimate users from altering records installed by legitimate users or installing fake records that would attract illegitimate traffic. Means to protect the DDNS server against Denial of Service (DoS) should be enabled. Note that these considerations are not specific to address-sharing contexts but are valid for DDNS services in general.

5. References

5.1. Normative References

[RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, January 2005, <http://www.rfc-editor.org/info/rfc3986>.

[RFC6333] Durand, A., Droms, R., Woodyatt, J., and Y. Lee, "Dual-Stack Lite Broadband Deployments Following IPv4 Exhaustion", RFC 6333, August 2011, <http://www.rfc-editor.org/info/rfc6333>.

[RFC6887] Wing, D., Cheshire, S., Boucadair, M., Penno, R., and P. Selkirk, "Port Control Protocol (PCP)", RFC 6887, April 2013, <http://www.rfc-editor.org/info/rfc6887>.

5.2. Informative References

[PCP-DEPLOYMENT] Boucadair, M., "Port Control Protocol (PCP) Deployment Models", Work in Progress, draft-boucadair-pcp-deployment-cases-03, July 2014.

[RFC2136] Vixie, P., Thomson, S., Rekhter, Y., and J. Bound, "Dynamic Updates in the Domain Name System (DNS UPDATE)", RFC 2136, April 1997, <http://www.rfc-editor.org/info/rfc2136>.

[RFC2606] Eastlake, D. and A. Panitz, "Reserved Top Level DNS Names", BCP 32, RFC 2606, June 1999, <http://www.rfc-editor.org/info/rfc2606>.

[RFC2782] Gulbrandsen, A., Vixie, P., and L. Esibov, "A DNS RR for specifying the location of services (DNS SRV)", RFC 2782, February 2000, <http://www.rfc-editor.org/info/rfc2782>.

[RFC3007] Wellington, B., "Secure Domain Name System (DNS) Dynamic Update", RFC 3007, November 2000, <http://www.rfc-editor.org/info/rfc3007>.

[RFC6146] Bagnulo, M., Matthews, P., and I. van Beijnum, "Stateful NAT64: Network Address and Protocol Translation from IPv6 Clients to IPv4 Servers", RFC 6146, April 2011, <http://www.rfc-editor.org/info/rfc6146>.

[RFC6269] Ford, M., Boucadair, M., Durand, A., Levis, P., and P. Roberts, "Issues with IP Address Sharing", RFC 6269, June 2011, <http://www.rfc-editor.org/info/rfc6269>.

[RFC6281] Cheshire, S., Zhu, Z., Wakikawa, R., and L. Zhang, "Understanding Apple's Back to My Mac (BTMM) Service", RFC 6281, June 2011, <http://www.rfc-editor.org/info/rfc6281>.

[RFC6763] Cheshire, S. and M. Krochmal, "DNS-Based Service Discovery", RFC 6763, February 2013, <http://www.rfc-editor.org/info/rfc6763>.

[RFC6888] Perreault, S., Yamagata, I., Miyakawa, S., Nakagawa, A., and H. Ashida, "Common Requirements for Carrier-Grade NATs (CGNs)", BCP 127, RFC 6888, April 2013, <http://www.rfc-editor.org/info/rfc6888>.

[RFC6890] Cotton, M., Vegoda, L., Bonica, R., and B. Haberman, "Special-Purpose IP Address Registries", BCP 153, RFC 6890, April 2013, <http://www.rfc-editor.org/info/rfc6890>.

[RFC6908] Lee, Y., Maglione, R., Williams, C., Jacquenet, C., and M. Boucadair, "Deployment Considerations for Dual-Stack Lite", RFC 6908, March 2013, <http://www.rfc-editor.org/info/rfc6908>.

Acknowledgements

Thanks to Stuart Cheshire for bringing up DNS-Based Service Discovery (SD) and [RFC6281], which covers a DNS-based SD scenario and gives an example of how the application is a means for a solution to address dynamic DNS updates; in this case, Apple's BTMM can be achieved.

Many thanks to D. Wing, D. Thaler, and J. Abley for their comments.

Contributors

The following individuals contributed text to the document:

Xiaohong Huang
Beijing University of Posts and Telecommunications, China
EMail: huangxh@bupt.edu.cn

Yan Ma
Beijing University of Posts and Telecommunications, China
EMail: mayan@bupt.edu.cn

Authors' Addresses

Xiaohong Deng

EMail: dxhbupt@gmail.com

Mohamed Boucadair
France Telecom
Rennes 35000
France

EMail: mohamed.boucadair@orange.com

Qin Zhao
Beijing University of Posts and Telecommunications
China

EMail: zhaoqin.bupt@gmail.com

James Huang
Huawei Technologies
China

EMail: james.huang@huawei.com

Cathy Zhou
Huawei Technologies
China

EMail: cathy.zhou@huawei.com