Network Working Group C. Huitema
Internet-Draft Microsoft
Intended status: Standards Track T. Mrugalski
Expires: April 4, 2016 ISC
S. Krishnan
Ericsson
October 2, 2015

Anonymity profile for DHCP clients
draft-ietf-dhc-anonymity-profile-04.txt

Abstract

Some DHCP options carry unique identifiers. These identifiers can enable device tracking even if the device administrator takes care of randomizing other potential identifications like link-layer addresses or IPv6 addresses. The anonymity profile is designed for clients that wish to remain anonymous to the visited network. The profile provides guidelines on the composition of DHCP or DHCPv6 requests, designed to minimize disclosure of identifying information. This draft updates RFC4361.

Table of Contents

1. Introduction 3
1.1. Requirements 3
2. Application domain 3
2.1. MAC address Randomization hypotheses 4
2.2. MAC address Randomization and DHCP 5
2.3. Radio fingerprinting 5
2.4. Operating system fingerprinting 6
2.5. No anonymity profile identification 6
2.6. Using the anonymity profiles 7
2.7. What about privacy for DHCP servers 8
3. Anonymity profile for DHCPv4 8
3.1. Avoiding fingerprinting 9
3.2. Client IP address field 9
3.3. Requested IP address option 10
3.4. Client hardware address field 10
3.5. Client Identifier Option 11
3.6. Parameter Request List Option 12
3.7. Host Name Option 12
3.8. Client FQDN Option 13
3.9. UUID/GUID-based Client Identifier Option 13
3.10. User and Vendor Class DHCP options 14
4. Anonymity profile for DHCPv6 14
4.1. Option encoding and avoiding fingerprinting 15
4.2. Do not send Confirm messages, unless really sure where . 15
4.3. Client Identifier DHCPv6 Option 16
4.3.1. Anonymous Information-Request 16
4.4. Server Identifier Option 16
4.5. Address assignment options 17
4.5.1. Obtain temporary addresses 17
4.5.2. Prefix delegation 18
4.6. Option Request Option 18
4.6.1. Previous option values 18
4.7. Authentication Option 19
4.8. User and Vendor Class DHCPv6 options 19
4.9. Client FQDN Option 19
5. Operational Considerations 19
6. Security Considerations 20
7. IANA Considerations 20
8. Acknowledgments 20

9. Changes from previous versions 20
10. References 22
10.1. Normative References 22
10.2. Informative References 23
Authors' Addresses 24

1. Introduction

Reports surfaced recently of systems that would monitor the wireless connections of passengers at Canadian airports [CNBC]. We can assume that these are either fragments or trial runs of a wider system that would attempt to monitor Internet users as they roam through wireless access points and other temporary network attachments. We can also assume that privacy conscious users will attempt to evade this monitoring, for example by ensuring that low level identifiers such as link-layer addresses are "randomized," so that the devices do not broadcast a unique identifier in every location that they visit.

Of course, link layer "MAC" addresses are not the only way to identify a device. As soon as it connects to a remote network, the device may use DHCP and DHCPv6 to obtain network parameters. The analysis of DHCP and DHCPv6 options shows that parameters of these protocols can reveal identifiers of the device, negating the benefits of link-layer address randomization. This is documented in detail in [I-D.ietf-dhc-dhcp-privacy] and [I-D.ietf-dhc-dhcpv6-privacy]. The natural reaction is to restrict the number and values of such parameters in order to minimize disclosure.

In the absence of a common standard, different system developers are likely to implement this minimization of disclosure in different ways. Monitoring entities could then use the differences to identify the software version running on the device. The proposed anonymity profile provides a common standard that minimizes information disclosure, including the disclosure of implementation identifiers.

1.1. Requirements

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 [RFC2119].

2. Application domain

Mobile nodes can be tracked using multiple identifiers, the most prominent being link-layer addresses, a.k.a. MAC addresses. For example, when devices use Wi-Fi connectivity, they place the MAC address in the header of all the packets that they transmit. Standard implementation of Wi-Fi use unique 48 bit link-layer

addresses, assigned to the devices according to procedures defined by IEEE 802. Even when the Wi-Fi packets are encrypted, the portion of the header containing the addresses will be sent in clear text. Tracking devices can "listen to the airwaves" to find out what devices are transmitting near them.

We can easily imagine that the MAC addresses can be correlated with other data, e.g., clear text names and cookies, to build a registry linking MAC addresses to the identity of devices' owners. Once that correlation is done, tracking the MAC address is sufficient to track individual people, even when all application data sent from the devices is encrypted. link-layer addresses can also be correlated with IP addresses of devices, negating potential privacy benefits of IPv6 "privacy" addresses. Privacy advocates have reasons to be concerned.

The obvious solution is to "randomize" the MAC address. Before connecting to a particular network, the device replaces the MAC address with a randomly drawn 48 bit value. Link-layer address randomization was successfully tried at the IETF in Honolulu in November 2014 [IETFMACRandom]. However, we have to consider the linkage between link-layer addresses, DHCP identifiers and IP addresses.

2.1. MAC address Randomization hypotheses

There is not yet an established standard for randomizing link-layer addresses. Various prototypes have tried different strategies, such as:

Per connection: Configure a random link-layer address at the time of connecting to a network, e.g. to specific Wi-Fi SSID, and keep it for the duration of the connection.

Per network: Same as "per connection," but always use the same link-layer address for the same network -- different of course from the addresses used in other networks.

Time interval: Change the link-layer address at regular time intervals.

In practice, there are many reasons to keep the link-layer address constant for the duration of a link-layer connection, as in the "per connection" or "per network" variants. On Wi-Fi networks, changing the link-layer address requires dropping the existing Wi-Fi connection and then re-establishing it, which implies repeating the connection process and associated procedures. The IP addresses will change, which means that all required TCP connections will have to be

re-established. If the network access is provided through a NAT, changing IP address also means that the NAT traversal procedures will have to be restarted. This means a lot of disruption. At the same time, an observer on the network will easily notice that a station left, another came in just after that, and that the new one appears to be communicating with pretty much the same set of IP addresses as the old one. This provides for easy correlation.

The anonymity profile pretty much assumes that the link-layer address randomization follows the "per connection" or "per network" strategies, or a variant of the "time interval" strategy in which the interval has about the same duration as the average connection.

2.2. MAC address Randomization and DHCP

From a privacy point of view, it is clear that link-layer address, IP address and DHCP identifier shall evolve in synchrony. For example, if the link-layer address changes and the DHCP identifier stays constant, then it is really easy to correlate old and new link-layer addresses, either by listening to DHCP traffic or by observing that the IP address remains constant, since it is tied to the DHCP identifier. Conversely, if the DHCP identifier changes but the link-layer address remains constant, the old and new identifiers and addresses can be correlated by listening to L2 traffic. The procedures documented in the following sections construct DHCP identifiers from the current link-layer address, automatically providing for this synchronization.

The proposed anonymity profiles solve this synchronization issues by deriving most identifiers from the link-layer address, and generally by making sure that DHCP parameter values do not remain constant after an address change.

2.3. Radio fingerprinting

MAC address randomization solves the trivial monitoring problem in which someone just uses a Wi-Fi scanner and records the MAC addresses seen on the air. DHCP anonymity solves the more elaborated scenario in which someone monitor link-layer addresses and identities used in DHCP at the access point or DHCP server. But these are not the only ways to track a mobile device.

Radio fingerprinting is a process that identifies a radio transmitter by the unique "fingerprint" of its signal transmission, i.e., the tiny differences caused by minute imperfections of the radio transmission hardware. This can be applied to diverse types of radios, including Wi-Fi as described for example in [WiFiRadioFingerprinting]. No amount of link-layer address

randomization will protect against such techniques. Protections may exist, but they are outside the scope of the present document.

On the other hand, we should not renounce randomization just because radio fingerprinting exists. The radio fingerprinting techniques are harder to deploy than just recording link-layer addresses with a scanner. They can only track devices for which the fingerprint are known, and thus have a narrower scope of application than mass monitoring of addresses and DHCP parameters.

2.4. Operating system fingerprinting

When a standard like DHCP allows for multiple options, different implementers will make different choices for the options that they support or the values they chose for the options. Conversely, monitoring the options and values present in DHCP messages reveals these differences and allows for "operating system fingerprinting," i.e., finding the type and version of software that a particular device is running. Finding these versions provides some information about the device identity, and thus goes against the goal of anonymity.

The design of the anonymity profiles attempts to minimize the number of options and the choice of values, in order to reduce the possibilities of operating system fingerprinting.

2.5. No anonymity profile identification

Reviewers of the anonymity profiles have sometimes suggested adding an option to explicitly identify the profiles as "using the anonymity option." One suggestion is that if the client wishes to remain anonymous, it would be good if the client told the server about that in case the server is willing to co-operate. Another possibility would be to use specific privacy-oriented construct, such as for example a new type of DUID for a temporary DUID that would be changing over time.

This is not workable in a large number of cases as it is possible that the network operator (or other entities that have access to the operator's network) might be actively participating in surveillance and anti-privacy, willingly or not. Declaring a preference for anonymity is a bit like walking around with a Guy Fawkes mask. When anonymity is required, it is generally not a good idea to stick out of the crowd. Simply revealing the desire for privacy, could cause the attacker to react by triggering additional surveillance or monitoring mechanisms. Therefore we feel that it is preferable to not disclose one's desire for privacy.

This preference leads to some important implications. In particular, we make an effort to make the mitigation techniques difficult to distinguish from regular client behaviors, if at all possible.

2.6. Using the anonymity profiles

There are downsides to randomizing link-layer addresses and DHCP identifiers. By definition, randomization will break management procedures that rely on tracking link-layer addresses. Even if this is not too much of a concern, we have to be worried about the frequency of link-layer address randomization. Suppose for example that many devices would get new random link-layer addresses at short intervals, maybe every few minutes. This would generate new DHCP requests in rapid succession, with a high risk of exhausting DHCPv4 address pools. Even with IPv6, there would still be a risk of increased neighbor discovery traffic, and bloating of various address tables. Implementers will have to be cautious when programming devices to use randomized MAC addresses. They will have to carefully chose the frequency with which such addresses will be renewed.

This document only provides guidelines for using DHCP when clients care about privacy and servers do not object. We assume that the request for anonymity is materialized by the assignment of a randomized link-layer address to the network interface. Once that decision is made, the following guidelines will avoid leakage of identity in DHCP parameters or in assigned addresses.

There may be rare situations where the clients want anonymity to attackers but not to their DHCP server. These clients should still use link-layer address randomization to hide from observers, and some form of encrypted communication to the DHCP server. This scenario is out of scope for this document.

To preserve anonymity, the clients need to not use stable values for the client identifiers. This is clearly a tradeoff, because a stable client identifier guarantees that the client will receive consistent parameters over time. An example is given in [RFC7618], where the client identifier is used to guarantee that the same client will always get the same combination of IP address and port range. Static clients benefit most from stable parameters, and can often be already identified by physical connection layer parameters. These static clients will normally not use the anonymity profile. Mobile clients, in contrast, have the option of using the anonymity profile in conjunction with [RFC7618] if they are more concerned with privacy protection than with stable parameters.

2.7. What about privacy for DHCP servers

This document only provides recommendations for DHCP clients. The main target are DHCP clients used in mobile devices. Such devices are a tempting target for various monitoring systems, and providing them with a simple anonymity solution is urgent. We can argue that some mobile devices embed DHCP servers, and that providing solutions for such devices is also quite important. Two plausible examples would be a DHCP server for a car network, or a DHCP server for a mobile hot spot. However, mobile servers get a lot of privacy protection through the use of access control and link layer encryption. Servers may disclose information to clients through DHCP, but they normally only do that to clients that have passed the link-layer access control and have been authorized to use the network services. This arguably makes solving the server problem less urgent than solving the client problem.

Server privacy issues are presented in [I-D.ietf-dhc-dhcp-privacy] and [I-D.ietf-dhc-dhcpv6-privacy]. Mitigation of these issues is left to further study.

3. Anonymity profile for DHCPv4

Clients using the DHCPv4 anonymity profile limit the disclosure of information by controlling the header parameters and by limiting the number and values of options. The number of options depend on the specific DHCP message:

DHCPDISCOVER: The anonymized DHCPDISCOVER messages MUST contain the Message Type, MAY contain the Client Identifier, and MAY contain the Parameter Request List options. It SHOULD NOT contain any other option.

DHCPREQUEST: The anonymized DHCPREQUEST messages MUST contain the Message Type, MAY contain the Client Identifier, and MAY contain the Parameter Request List options. If the message is in response to a DHCPOFFER, it MUST contain the corresponding Server Identifier option and the Requested IP address. If the message is not in response to a DHCPOFFER, it MAY contain a Requested IP address as explained in Section 3.3. It SHOULD NOT contain any other option.

DHCPDECLINE: The anonymized DHCPDECLINE messages MUST contain the Message Type, Server Identifier, and Requested IP address options, and MAY contain the Client Identifier options.

DHCPRELEASE: The anonymized DHCPRELEASE messages MUST contain the Message Type and Server Identifier options, and MAY contain the Client Identifier option.

DHCPINFORM: The anonymized DHCPINFORM messages MUST contain the Message Type, and MAY contain the Client Identifier and Parameter Request List options. It SHOULD NOT contain any other option.

Header fields and option values SHOULD be set in accordance with the DHCP specification, but some header fields and option values SHOULD be constructed per the following guidelines.

The inclusion of HostName and FQDN options in DHCPDISCOVER, DHCPREQUEST or DHCPINFORM messages is discussed in Section 3.7 and Section 3.8.

3.1. Avoiding fingerprinting

There are many choices for implementing DHCPv4 messages. Clients can choose to transmit a specific set of options, pick particular encoding for these options, and transmit options in different orders. These choices can be use to fingerprint the client.

The following sections provide guidance on the encoding of options and fields within the packets. However, this guidance alone may not be sufficient to prevent fingerprinting from revealing information, such as the device type, vendor type or OS type and in some cases specific version, or from revealing whether the client is using the anonymity profile.

The client willing to protect its privacy SHOULD limit the subset of options sent in messages to the subset listed in the following sections.

The client willing to protect its privacy SHOULD randomize options order before sending any DHCPv4 message. If this random ordering cannot be implemented, the client MAY arrange options by increasing order of option codes.

3.2. Client IP address field

Four bytes in the header of the DHCP messages carry the "Client IP address" (ciaddr) as defined in [RFC2131]. In DHCP, this field is used by the clients to indicate the address that they used previously, so that as much as possible the server can allocate them the same address.

There is very little privacy implication of sending this address in the DHCP messages, except in one case, when connecting to a different network than the last network connected. If the DHCP client somehow repeated the address used in a previous network attachment, monitoring services might use the information to tie the two network locations. DHCP clients should ensure that the field is cleared when they know that the network attachment has changed, and in particular if the link layer address is reset by the device's administrator.

The clients using the anonymity profile MUST NOT include in the message a Client IP Address that has been obtained with a different link-layer address.

3.3. Requested IP address option

The Requested IP address option id defined in [RFC2132] with code 50. It allows the client to request that a particular IP address be assigned. The option is mandatory in some protocol messages per [RFC2131], for example when a client selects to use an address offered by a server. However, this option is not mandatory in the DHCPDISCOVER message. It is simply a convenience, an attempt to regain the same IP address that was used in a previous connection. Doing so entails the risk of disclosing an IP address used by the client at a previous location, or with a different link-layer address.

When using the anonymity profile, clients SHOULD NOT use the Requested IP address option in DHCPDISCOVER messages. They MUST use the option when mandated by the DHCP protocol, for example in DHCPREQUEST messages.

There are scenarios in which a client connecting to a network remembers previously allocated address, i.e. is in the INIT-REBOOT state. In that state, the client that is concerned with privacy SHOULD perform a complete four way handshake starting with DHCPDISCOVER to obtain a new address lease. If the client can ascertain that this is exactly the same network to which it was previously connected, and if the link layer address did not change, the client MAY issue a DHCPREQUEST to try reclaim the current address.

3.4. Client hardware address field

Sixteen bytes in the header of the DHCP messages carry the "Client hardware address" (chaddr) as defined in [RFC2131]. The presence of this address is necessary for the proper operation of the DHCP service.

Hardware addresses, called "link layer address" in many RFCs, can be used to uniquely identify a device, especially if they follow the IEEE 802 recommendations. These unique identifiers can be used by monitoring services to track the location of the device and its user. The only plausible defense is to somehow reset the hardware address to a random value when visiting an untrusted location, before transmitting anything at that location with the hardware address. If the hardware address is reset to a new value, or randomized, the DHCP client SHOULD use the new randomized value in the DHCP messages.

3.5. Client Identifier Option

The client identifier option is defined in [RFC2132] with option code 61. It is discussed in detail in [RFC4361]. The purpose of the client identifier option is to identify the client in a manner independent of the link layer address. This is particularly useful if the DHCP server is expected to assign the same address to the client after a network attachment is swapped and the link layer address changes. It is also useful when the same node issues requests through several interfaces, and expects the DHCP server to provide consistent configuration data over multiple interfaces.

The considerations for hardware independence and strong client identity have an adverse effect on the privacy of mobile clients, because the hardware-independent unique identifier obviously enables very efficient tracking of the client's movements. One option would be to not transmit this option at all, but this may affect interoperability and will definitely mark the client as requesting anonymity, exposing it to the risks mentioned in Section 2.5.

The recommendations in [RFC4361] are very strong, stating for example that "DHCPv4 clients MUST NOT use client identifiers based solely on layer two addresses that are hard-wired to the layer two device (e.g., the Ethernet MAC address)." These strong recommendations are in fact a tradeoff between ease of management and privacy, and the tradeoff should depend on the circumstances.

In contradiction to [RFC4361], when using the anonymity profile, DHCP clients MUST use client identifiers based solely on the link layer address that will be used in the underlying connection. This will ensure that the DHCP client identifier does not leak any information that is not already available to entities monitoring the network connection. It will also ensure that a strategy of randomizing the link layer address will not be nullified by DHCP options.

There are usages of DHCP where the underlying connection is a point to point link, in which case there is no link layer address available to construct a non-revealing identifier. If anonymity is desired in

such networks, the client SHOULD pick a random identifier that is unique to the current link, using for example a combination of a local secret and an identifier of the connection.

3.6. Parameter Request List Option

The Parameter Request List (PRL) option is defined in [RFC2132] with option code 61. It list the parameters requested from the server by the client. Different implementations request different parameters.[RFC2132] specifies that "the client MAY list the options in order of preference." It practice, this means that different client implementations will request different parameters, in different orders.

The choice of option numbers and the specific ordering of option numbers in the Parameter Request List can be used to fingerprint the client. This may not reveal the identity of a client, but may provide additional information, such as the device type, vendor type or OS type and in some cases specific version.

The client willing to protect its privacy SHOULD only request a minimal number of options in PRL, and SHOULD also randomly shuffle the option codes order in PRL. If this random ordering cannot be implemented, the client MAY order option codes order in PRL by increasing order of option codes.

3.7. Host Name Option

The Host Name option is defined in [RFC2132] with option code 12. Depending on implementations, the option value can carry either a fully qualified domain name such as "node1984.example.com," or a simple host name such as "node1984." The host name is commonly used by the DHCP server to identify the host, and also to automatically update the address of the host in local name services.

Fully qualified domain names are obviously unique identifiers, but even simple host names can provide a significant amount of information on the identity of the device. They are typically chosen to be unique in the context where the device is most often used. If that context is wide enough, in a large company or in a big university, the host name will be a pretty good identifier of the device. Monitoring services could use that information in conjunction with traffic analysis and quickly derive the identity of the device's owner.

When using the anonymity profile, DHCP clients SHOULD NOT send the host name option. If they chose to send the option, DHCP clients

MUST always send a non-qualified host name instead of a fully qualified domain name, and MUST obfuscate the host name value.

There are many ways to obfuscate a host name. The construction rules SOULD guarantee that a different host name is generated each time the link-layer address changes and that the obfuscated host name will not reveal the underlying link layer address. The construction SHOULD generate names that are unique enough to minimize collisions in the local link. Clients MAY use the following algorithm: compute a secure hash of a local secret and of the link layer address that will be used in the underlying connection, and then use the hexadecimal representation of the first 6 bytes of the hash as the obfuscated host name.

There is a potential downside to having a specific name pattern for hosts that require anonymity, as explained in Section 2.5. For this reason, the above algorithm is just a suggestion.

3.8. Client FQDN Option

The Client FQDN option is defined in [RFC4702] with option code 81. The option allows the DHCP clients to advertise to the DHCP server their fully qualified domain name (FQDN) such as "mobile.example.com." This would allow the DHCP server to update in the DNS the PTR record for the IP address allocated to the client. Depending on circumstances, either the DHCP client or the DHCP server could update in the DNS the A record for the FQDN of the client.

Obviously, this option uniquely identifies the client, exposing it to the DHCP server or to anyone listening to DHCP traffic. In fact, if the DNS record is updated, the location of the client becomes visible to anyone with DNS lookup capabilities.

When using the anonymity profile, DHCP clients SHOULD NOT include the Client FQDN option in their DHCP requests. Alternatively, they MAY include a special purpose FQDN using the same hostname as in the Host Name Option, with a suffix matching the connection-specific DNS suffix being advertised by that DHCP server. Having a name in the DNS allows working with legacy systems that require one to be there, e.g., by verifying a forward and reverse lookup succeeds with the same result.

3.9. UUID/GUID-based Client Identifier Option

The UUID/GUID-based Client Machine Identifier option is defined in [RFC4578], with option code 97. The option is part of a set of options for Intel Preboot eXecution Environment (PXE). The purpose of the PXE system is to perform management functions on a device

before its main OS is operational. The Client Machine Identifier carries a 16-octet Globally Unique Identifier (GUID), which uniquely identifies the device.

The PXE system is clearly designed for devices operating in a controlled environment, and its functions are not meant to be used by mobile nodes visiting untrusted networks. If only for privacy reasons, nodes visiting untrusted networks MUST disable the PXE functions, and MUST NOT send the corresponding options.

3.10. User and Vendor Class DHCP options

Vendor identifying options are defined in [RFC2132] and [RFC3925]. When using the anonymity profile, DHCP clients SHOULD NOT use the Vendor Specific Information option (code 43), the Vendor Class Identifier Option (60), the Vendor Class option (code 124), or the Vendor Specific Information option (code 125) as these options potentially reveal identifying information.

4. Anonymity profile for DHCPv6

DHCPv6 is typically used by clients in one of two scenarios: stateful and stateless configuration. In the stateful scenario, clients use a combination of SOLICIT, REQUEST, CONFIRM, RENEW, REBIND and RELEASE messages to obtain addresses, and manage these addresses.

In the stateless scenario, clients configure addresses using a combination of client managed identifiers and router-advertised prefixes, without involving the DHCPv6 services. Different ways of constructing these prefixes have different implications on privacy, which are discussed in [I-D.ietf-6man-default-iids] and [I-D.ietf-6man-ipv6-address-generation-privacy]. In the stateless scenario, clients use DHCPv6 to obtain network configuration parameters, through the INFORMATION-REQUEST message.

The choice between the stateful and stateless scenario depends on flag and prefix options published by the "Router Advertisement" messages of local routers, as specified in [RFC4861]. When these options enable stateless address configuration hosts using the anonymity profile SHOULD choose it over stateful address configuration, because stateless configuration requires fewer information disclosures than stateful configuration.

When using the anonymity profile, DHCPv6 clients carefully select DHCPv6 options used in the various messages that they send. The list of options that are mandatory or optional for each message is specified in [RFC3315]. Some of these options have specific

implications on anonymity. The following sections provide guidance on the choice of option values when using the anonymity profile.

4.1. Option encoding and avoiding fingerprinting

There are many choices for implementing DHCPv6 messages. As explained in Section 3.1, these choices can be use to fingerprint the client.

The following sections provide guidance on the encoding of options. However, this guidance alone may not be sufficient to prevent fingerprinting from revealing information, such as the device type, vendor type or OS type and in some cases specific version, or from revealing whether the client is using the anonymity profile.

The client willing to protect its privacy SHOULD limit the subset of options sent in messages to the subset listed in the following sections.

The client willing to protect its privacy SHOULD randomize options order before sending any DHCPv6 message. If this random ordering cannot be implemented, the client MAY arrange options by increasing order of option codes.

4.2. Do not send Confirm messages, unless really sure where

[RFC3315] requires clients to send a Confirm message when they attach to a new link to verify whether the addressing and configuration information they previously received is still valid. This requirement was relaxed in [I-D.ietf-dhc-rfc3315bis]. When these clients send Confirm messages, they include any IAs assigned to the interface that may have moved to a new link, along with the addresses associated with those IAs. By examining the addresses in the Confirm message an attacker can trivially identify the previous point(s) of attachment.

Clients interested in protecting their privacy SHOULD NOT send Confirm messages and instead directly try to acquire addresses on the new link. However, not sending confirm messages can result in connectivity hiatus in some scenarios, e.g. roaming between two access points in the same wireless network. DHCPv6 clients that can verify that the previous link and the current link are part of the same network MAY send Confirm messages while still protecting their privacy.

4.3. Client Identifier DHCPv6 Option

The client identifier option is defined in [RFC3315] with option code 1. The purpose of the client identifier option is to identify the client to the server. The content of the option is a DHCP Unique Identifier (DUID). One of the primary privacy concerns is that a client is disclosing a stable identifier (the DUID) that can be use for tracking and profiling. Three DUID formats are specified in [RFC3315]: Link-layer address plus time, Vendor-assigned unique ID based on Enterprise Number, Link-layer address. A fourth type, DUID-UUID is defined in [RFC6355].

When using the anonymity profile in conjunction with randomized link-layer addresses, DHCPv6 clients MUST use the DUID format number 3, Link-layer address. The value of the Link-layer address should be that currently assigned to the interface.

When using the anonymity profile without the benefit of randomized link-layer addresses, clients that want to protect their privacy SHOULD generate a new randomized DUID-LLT every time they attach to a new link or detect a possible link change event. The exact details are left up to implementors, but there are several factors should be taken into consideration. The DUID type SHOULD be set to 1 (DUID-LLT). Hardware type SHOULD be set appropriately to the hardware type. Time MAY be set to current time, but this will reveal the fact that the DUID is newly generated. Implementors interested in hiding this fact MAY use a time stamp from the past. e.g. a random timestamp from the previous year could be a good value.

4.3.1. Anonymous Information-Request

According to [RFC3315], a DHCPv6 client typically includes its client identifier in most of the messages it sends. There is one exception, however. Client is allowed to omit its client identifier when sending Information-Request.

When using stateless DHCPv6, clients wanting to protect their privacy SHOULD NOT include client identifiers in their Information-Request messages. This will prevent the server from specifying client-specific options if it is configured to do so, but the need for anonymity precludes such options anyway.

4.4. Server Identifier Option

When using the anonymity profile, DHCPv6 clients SHOULD use the Server Identifier Option (code 2) as specified in [RFC3315]. Clients MUST only include server identifier values that were received with

the current link-layer address, because reuse of old values discloses information that can be used to identify the client.

4.5. Address assignment options

When using the anonymity profile, DHCPv6 clients might have to use SOLICIT or REQUEST messages to obtain IPv6 addresses through the DHCP server. Clients interested in privacy SHOULD request addresses using the Identity Association for Non-temporary Addresses Option (IA_NA, code 3).

The IA_NA option includes an IAID parameter that identifies a unique identity association for the interface for which the Address is requested. Clients interested in protecting their privacy MUST ensure that the IAID does not enable client identification. They also need to conform to the requirement of [RFC3315] that the IAID for that IA MUST be consistent across restarts of the DHCP client. We interpret that as requiring that the IAID MUST be constant for the association, as long as the link layer Address remains constant.

Clients MAY meet the privacy, uniqueness and stability requirement of the IAID using by constructing it as the combination of one byte encoding the interface number in the system, and three bytes of the link layer address.

The clients MAY use the IA Address Option (code 5) but need to balance the potential advantage of "address continuity" versus the potential risk of "previous address disclosure." A potential solution is to remove all stored addresses when a link-layer address changes, and to only use the IA Address option with addresses that have been explicitly assigned through the current link-layer address.

4.5.1. Obtain temporary addresses

[RFC3315] defines a special container (IA_TA, code 4) for requesting temporary addresses. This is a good mechanism in principle, but there are a number of issues associated with it. First, this is not a widely used feature, so clients depending solely on temporary addresses may lock themselves out of service. Secondly, [RFC3315] does not specify any lifetime or lease length for temporary addresses. Therefore support for renewing temporary addresses may vary between client implementations, including not being supported at all. Finally, by requesting temporary addresses a client reveals its desire for privacy and potentially risks countermeasures as described in Section 2.5.

Because of these Clients interested in their privacy SHOULD NOT use IA_TA.

The addresses obtained according to Section 4.5 are temporary in nature, and will be discarded when the link layer address is changed. They thus meet most of the use cases of the temporary addresses defined in [RFC4941]. Clients interested in their privacy should not publish their IPv6 addresses in the DNS or otherwise associate them with name services, and thus do not normally need two classes of addresses, one public, one temporary.

The use of mechanisms to allocate several IPv6 addresses to a client while preserving privacy is for further study.

4.5.2. Prefix delegation

The interaction between prefix delegation and anonymity require further study. For now, the simple solution is to avoid using prefix delegation when striving for anonymity. When using the anonymity profiles, clients SHOULD NOT use IA_PD, the prefix delegation form of address assignment.

4.6. Option Request Option

The Option Request Option (ORO) is defined in [RFC3315] with option code 6. It specifies the options that the client is requesting from the server. The choice of requested options and the order of encoding of these options in the ORO can be used to fingerprint the client.

The client willing to protect its privacy SHOULD only request a minimal subset of options and SHOULD randomly shuffle the option codes order in ORO. If this random ordering cannot be implemented, the client MAY order option codes in ORO by increasing order of option codes.

4.6.1. Previous option values

According to [RFC3315], the client that includes an Option Request Option in a Solicit or Request message MAY additionally include instances of those options that are identified in the Option Request option, with data values as hints to the server about parameter values the client would like to have returned.

When using the anonymity profile, clients SHOULD NOT include such instances of options because old values might be used to identify the client.

4.7. Authentication Option

The purpose of the Authentication option (code 11) is to authenticate the identity of clients and servers and the contents of DHCP messages. As such, the option can be used to identify the client, and is incompatible with the stated goal of "client anonymity." DHCPv6 clients that use the anonymity profile SHOULD NOT use the authentication option. They MAY use it if they recognize that they are operating in a trusted environment, e.g., in a work place network.

4.8. User and Vendor Class DHCPv6 options

When using the anonymity profile, DHCPv6 clients SHOULD NOT use the User Class option (code 15) or the Vendor Class option (code 16), as these options potentially reveal identifying information.

4.9. Client FQDN Option

The Client FQDN option is defined in [RFC4704] with option code 29. The option allows the DHCP clients to advertise to the DHCP server their fully qualified domain name (FQDN) such as "mobile.example.com." When using the anonymity profile, DHCPv6 clients SHOULD NOT include the Client FQDN option in their DHCPv6 messages because it identifies the client. As explained in Section 3.8 they MAY use a local-only FQDN by combining a host name derived from the link layer address and a suffix advertised by the local DHCP server.

5. Operational Considerations

The anonymity profile has the effect of hiding the client identity from the DHCP server. This is not always desirable. Some DHCP servers provide facilities like publishing names and addresses in the DNS, or ensuring that returning clients get reassigned the same address.

Clients using the anonymity profile may be consuming more resources. For example when they change link-layer address and request for a new IP, the old one is still marked as leased by the server.

Implementers SHOULD provide a way for clients to control when the anonymity profile is used, and when standard behavior is preferred. Implementers MAY implement this control by tying use of the anonymity profile to that of link-layer address randomization.

6. Security Considerations

The use of the anonymity profile does not change the security considerations of the DHCPv4 or DHCPv6 protocols.

7. IANA Considerations

This draft does not require any IANA action.

8. Acknowledgments

The inspiration for this draft came from discussions in the Perpass mailing list. Several people provided feedback on this draft, notably Noel Anderson, Lorenzo Colitti, Stephen Farrell, Nick Grifka, Tushar Gupta, Gabriel Montenegro, Marcin Siodelski, Dave Thaler, Bernie Volz, and Jun Wu.

9. Changes from previous versions

The RFC Editor must ensure that this section is removed prior to RFC publication.

Changes from draft-00 to draft-01:

1. In Section 2.6, added guidance when using [RFC7618].

2. In Section 3.5, added guidance for case when no link layer address is available.

3. In Section 3.7, changed the recommended mechanism for obfuscating host names, in order to avoid reveal the underlying link layer address.

4. In Section 4.2, added an exception to the "should not send confirm" recommendation, to account for the "good" use of Confirm when roaming between access points on the same network.

Changes from draft-01 to draft-02:

1. In Section 3, checked the requirements of parameters in messages to ensure consistency with the main text.

2. In Section 3.5, added guidance for case when no link layer address is available.

3. In Section 3.7, specified that clients SHOULD NOT send the option, and that the optional obfuscation mechanism is just a suggestion.

4. Updated the text in Section 4.5.1 for temporary IPv6 address allocation.

5. Rewrote Section 5 on operational requirements for clarity.

Changes from draft-02 to draft-03:

1. Removed the update of [RFC4361] since we are specifying when to use that RFC, but are not recommending any specific change.

2. Fixed a number of typos and nits.

3. In Section 2.7, specified that mitigation of server privacy issues is left for further study.

4. Moved the guidance on avoiding fingerprinting to Section 3.1 and Section 4.1.

5. In Section 3.5, added text explaining why the client identifier option should still be sent, even when anonymity is desired.

6. Added Section 3.6 specifying the random ordering of requested option codes in the PRL parameter, with an alternative option for strict ordering.

7. Changed the requirement in Section 4.6 to allow "increasing order of option codes" as an alternative to "randomized order of options".

8. In Section 4.5.1 revised the language stating lack of support for renewing temporary addresses, as RFC 3315 does in fact specify a mechanism for doing so.

Changes from draft-03 to draft-04 address comments received during Working Group Last Call:

1. In Section 3, tightened the normative language and the use of message codes.

2. In Section 3.3, clarified the reference to the INIT-REBOOT scenario.

3. Revised the writing of Section 4.5 for greater clarity.

10. References

10.1. Normative References

[I-D.ietf-dhc-rfc3315bis]
Mrugalski, T., Siodelski, M., Volz, B., Yourtchenko, A., Richardson, M., Jiang, S., and T. Lemon, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6) bis", draft-ietf-dhc-rfc3315bis-01 (work in progress), July 2015.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <http://www.rfc-editor.org/info/rfc2119>.

[RFC2131] Droms, R., "Dynamic Host Configuration Protocol", RFC 2131, DOI 10.17487/RFC2131, March 1997, <http://www.rfc-editor.org/info/rfc2131>.

[RFC2132] Alexander, S. and R. Droms, "DHCP Options and BOOTP Vendor Extensions", RFC 2132, DOI 10.17487/RFC2132, March 1997, <http://www.rfc-editor.org/info/rfc2132>.

[RFC3315] Droms, R., Ed., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, DOI 10.17487/RFC3315, July 2003, <http://www.rfc-editor.org/info/rfc3315>.

[RFC3925] Littlefield, J., "Vendor-Identifying Vendor Options for Dynamic Host Configuration Protocol version 4 (DHCPv4)", RFC 3925, DOI 10.17487/RFC3925, October 2004, <http://www.rfc-editor.org/info/rfc3925>.

[RFC4361] Lemon, T. and B. Sommerfeld, "Node-specific Client Identifiers for Dynamic Host Configuration Protocol Version Four (DHCPv4)", RFC 4361, DOI 10.17487/RFC4361, February 2006, <http://www.rfc-editor.org/info/rfc4361>.

[RFC4702] Stapp, M., Volz, B., and Y. Rekhter, "The Dynamic Host Configuration Protocol (DHCP) Client Fully Qualified Domain Name (FQDN) Option", RFC 4702, DOI 10.17487/RFC4702, October 2006, <http://www.rfc-editor.org/info/rfc4702>.

[RFC4704] Volz, B., "The Dynamic Host Configuration Protocol for IPv6 (DHCPv6) Client Fully Qualified Domain Name (FQDN) Option", RFC 4704, DOI 10.17487/RFC4704, October 2006, <http://www.rfc-editor.org/info/rfc4704>.

[RFC4861] Narten, T., Nordmark, E., Simpson, W., and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861, DOI 10.17487/RFC4861, September 2007, <http://www.rfc-editor.org/info/rfc4861>.

[RFC4941] Narten, T., Draves, R., and S. Krishnan, "Privacy Extensions for Stateless Address Autoconfiguration in IPv6", RFC 4941, DOI 10.17487/RFC4941, September 2007, <http://www.rfc-editor.org/info/rfc4941>.

10.2. Informative References

[CNBC] Weston, G., Greenwald, G., and R. Gallagher, "CBC News: CSEC used airport Wi-Fi to track Canadian travellers", Jan 2014, <http://www.cbc.ca/news/politics/csec-used-airport-wi-fi-to-track-canadian-travellers-edward-snowden-documents-1.2517881>.

[I-D.ietf-6man-default-iids] Gont, F., Cooper, A., Thaler, D., and S. LIU, "Recommendation on Stable IPv6 Interface Identifiers", draft-ietf-6man-default-iids-07 (work in progress), August 2015.

[I-D.ietf-6man-ipv6-address-generation-privacy] Cooper, A., Gont, F., and D. Thaler, "Privacy Considerations for IPv6 Address Generation Mechanisms", draft-ietf-6man-ipv6-address-generation-privacy-08 (work in progress), September 2015.

[I-D.ietf-dhc-dhcp-privacy] Jiang, S., Krishnan, S., and T. Mrugalski, "Privacy considerations for DHCPv4", draft-ietf-dhc-dhcp-privacy-01 (work in progress), August 2015.

[I-D.ietf-dhc-dhcpv6-privacy] Krishnan, S., Mrugalski, T., and S. Jiang, "Privacy considerations for DHCPv6", draft-ietf-dhc-dhcpv6-privacy-01 (work in progress), August 2015.

[IETFMACRandom] Zuniga, JC., "MAC Privacy", November 2014, <http://www.ietf.org/blog/2014/11/mac-privacy/>.

[RFC4578] Johnston, M. and S. Venaas, Ed., "Dynamic Host Configuration Protocol (DHCP) Options for the Intel Preboot eXecution Environment (PXE)", RFC 4578, DOI 10.17487/RFC4578, November 2006, <http://www.rfc-editor.org/info/rfc4578>.

[RFC6355] Narten, T. and J. Johnson, "Definition of the UUID-Based DHCPv6 Unique Identifier (DUID-UUID)", RFC 6355, DOI 10.17487/RFC6355, August 2011, <http://www.rfc-editor.org/info/rfc6355>.

[RFC7618] Cui, Y., Sun, Q., Farrer, I., Lee, Y., Sun, Q., and M. Boucadair, "Dynamic Allocation of Shared IPv4 Addresses", RFC 7618, DOI 10.17487/RFC7618, August 2015, <http://www.rfc-editor.org/info/rfc7618>.

[WiFiRadioFingerprinting] Brik, V., Banerjee, S., Gruteser, M., and S. Oh, "Wireless Device Identification with Radiometric Signatures", September 2008, <http://www.winlab.rutgers.edu/~gruteser/papers/brik_paradis.pdf>.

Authors' Addresses

Christian Huitema
Microsoft
Redmond, WA 98052
U.S.A.

Email: huitema@microsoft.com

Tomek Mrugalski
Internet Systems Consortium, Inc.
950 Charter Street
Redwood City, CA 94063
USA

Email: tomasz.mrugalski@gmail.com

Suresh Krishnan
Ericsson
8400 Decarie Blvd.
Town of Mount Royal, QC
Canada

Phone: +1 514 345 7900 x42871
Email: suresh.krishnan@ericsson.com