DHC
Internet-Draft
Expires: January 2004

T. Lemon
Nominum, Inc.
M. Richardson
SSW
June 22, 2003

DHCP RSA/DSA Authentication using DNS KEY records
draft-ietf-dhc-auth-sigzero-00.txt

Abstract

This document defines a method for using a KEY record in the DNS belonging to a particular host to authenticate that host's DHCP transactions.

1. Introduction

Authentication for DHCP Messages [RFC3118], defines a mechanism by which DHCP messages can be signed, and it also defines two protocols for signing such methods. This document adds a new

protocol to RFC3118 that can be used to sign DHCP a message using the private half of a public/private key pair, and to verify the signature on the message using the public half, which stored in the DNS, in a KEY resource record stored on the signing host's domain name.

It is important to note that this document specifies an authentication mechanism, not an authorization mechanism. The ability to prove that a host knows the secret half of the public key associated with its name can be very useful, but in itself doesn't necessarily mean that it should or should not be trusted in any particular way.

However, a mechanism whereby a host can prove that it knows the private key associated with a hostname can be useful in at least two ways. First, DHCP clients can ask DHCP servers to set up PTR records on their behalf. The client provides the name to be stored in the PTR record. With this authentication mechanism, the client, by signing its messages with its private key, proves that it has a right to use the name associated with the public half of the key.

The second use that we envision is that both the client's and server's domain names provide a token that can be used in an authorization database - a handle whereby a relationship between the client and server can be documented. On the part of the server, this can be a way to determine whether or not the client is entitled to be allocated resources by the DHCP server, and to what resources it is entitled, in cases where different clients may be allocated different resources.

On the part of a client, while the mere fact that a DHCP server has signed its message with a key that turns out to be valid does not mean that the DHCP server is trustworthy, the signature does provide a potentially helpful audit trail back to the source of the invalid DHCP message in the case that it indeed turns out to be invalid.

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 "Key words for use in RFCs to Indicate Requirement Levels" [RFC2119].

2. Authentication option for dns-sigzero protocol with RSA.

The new algorithm type dns-sigzero for the DHCP option is defined as follows:

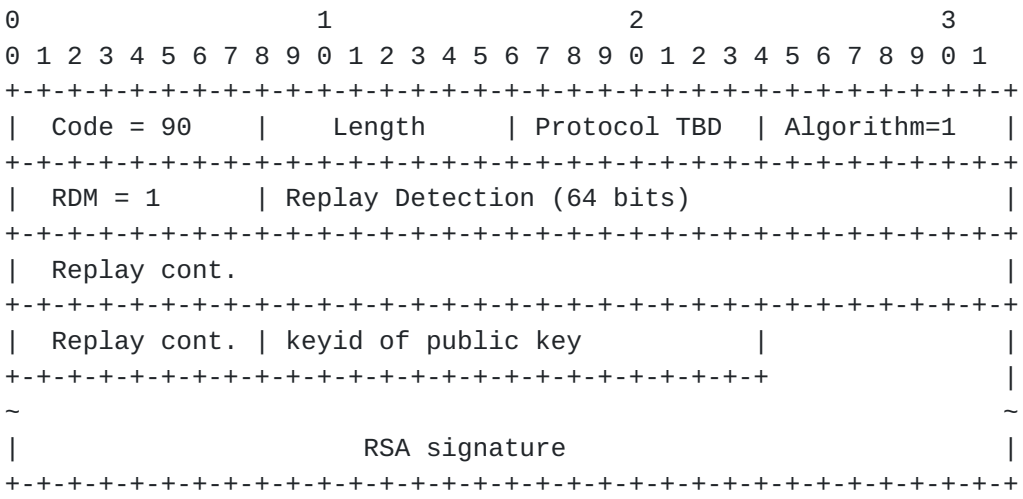

Figure 1: RSA authentication option for DHCP

This section defines the contents of the Authentication Information field of this payload. The RSA signature algorithm is defined in PKCS RSA Cryptography Specifications Version 2.1 [RFC3447]. A more concise definition is provided in RSA/MD5 KEYs and SIGs in the Domain Name System [RFC2537]. The RSA signature is defined with MD5 as the hash algorithm.

2. Authentication option for dns-sigzero protocol with DSA.

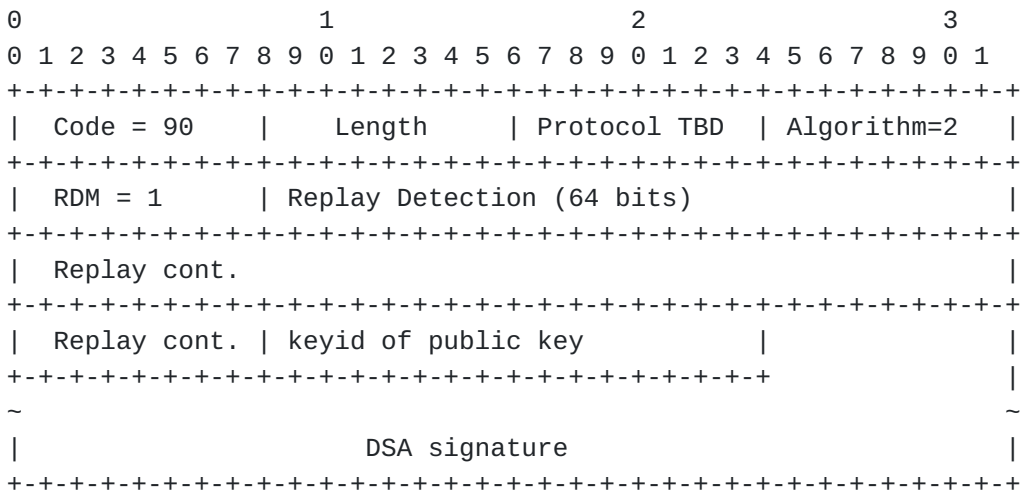

Figure 2: DSA authentication option for DHCP

4. Model of operation

4.1 Changes in processing for DHCP clients

Section 4.4 of Dynamic Host Configuration Protocol [RFC2131], defines how the DHCP client processes messages and makes state transitions.

When a DHCP client uses the DHCP authentication option with the dns-sigzero protocol, it MUST send a DHCP client FQDN option, and the domain name contained in that option must be fully-qualified. The domain name MUST be a name that exists in the DNS and has a KEY resource record associated with it, and the KEY record's type MUST be DNSSEC. The private key that the client uses to sign its messages MUST be the private half of a public/private key pair. The public half of that key pair must be stored in the KEY resource record associated with the name specified in the FQDN option.

4.2 Client INIT state changes

The client MUST include a parameter request list option that includes the DHCP authentication option.

4.3 Client SELECTING state changes

Many DHCP clients simply accept the first DHCPOFFER that they

receive, despite the fact that RFC2131 allows them to wait to collect more than one offer. DHCP clients that implement the dns-sigzero authentication protocol MUST wait for a reasonable period of time after receipt of the first response, and MUST collect additional responses. The client SHOULD validate every response that it receives that is signed.

If the client receives any DHCPOFFER message that is signed with a key that it has been configured to consider valid, the client should discard any DHCPOFFER messages that it has received that do not meet this standard. If it receives more than one such message, it should choose between them as specified in RFC2131.

If no preferred offer is seen, then the client MUST select among the offers in a non-deterministic manner (ideally, random). This step is important so that a client that has once been deceived into binding to the wrong DHCP server will have a chance to select a different server.

A client SHOULD NOT assume that offers that do not include valid and verifiable signature options are exclusively preferred. There may be no DHCP security on the network in question, and attackers could keep the client from ever selecting the "real", unauthenticated server.

Note that this behavior differs from that described in point 2 of section 5.5.1 of Authentication For DHCP Messages [RFC3118]. This is because a client may not be able to determine the authenticity of the offer until after it has connected to the network.

4.4 Client REQUESTING state changes

If the DHCP client is responding to a message from a DHCP server that is signed using the dns-sigzero protocol, the DHCP client MUST sign its response using the dns-sigzero protocol, using the same key and FQDN that it sent in its DHCPDISCOVER message.

In this case, when the client receives a DHCPACK from the server, if the DHCPACK is not signed, or is signed with a different keyid than was used to sign the DHCPOFFER, or if the server identifier is different, the DHCP client MUST drop the DHCPACK message and return to the INIT state.

Otherwise, instead of making the transition into the BOUND state, the DHCP client SHOULD make a transition into a new state defined in this document, the PROVISIONALLY-BOUND state.

4.5 The PROVISIONALLY-BOUND state

The PROVISIONALLY-BOUND state is operationally similar to the BOUND state. The timers should be recorded as with the BOUND state. Additional DHCP offers received should be discarded. The DHCP client MUST configure the IP stack and DNS server IP addresses as if it were entering the BOUND state. It SHOULD NOT configure IP addresses for any other servers, nor should it start any services that would normally be started on entry into the BOUND state.

Upon entering this state, after performing the partial configuration, the client MUST authenticate the DHCPACK message. To do this, if it does not already have the public key of the DHCP server, it must look it up.

The client MUST look up the KEY resource record (subtype DNSSEC) associated with the name provided by the server in its DHCP server name option. The DNS lookup MUST be done using DNSSEC.

If the DHCPACK can not be authenticated (either because the KEY can not be retrieved, the DNSSEC does not authenticate the key, or integrity check on the message fails), then the lease MUST be discarded. The client MUST unconfigure the network and return to the INIT state.

If the DHCP client is able to authenticate the DHCPACK, then it MUST make the transition to the BOUND state.

4.6 Client BOUND state changes

There is a new transition from the Provisionally BOUND state.

The only change in behaviour of this state is that when lease renewal occurs, the DHCPREQUEST SHOULD be signed. This is done even if the lease was not originally acquired through a signature, as it MAY be that the server will adopt security in the interum.

4.7 Client RENEWING state changes

There is a new transition to the Provisionally BOUND state.

If a DHCPACK is received that has a DHCP Authentication option in it, then the client transitions to the Provisionally BOUND state rather than directly back to the BOUND state.

The DHCPREQUEST SHOULD be signed using the DHCP authentication

option, as with the one sent by state SELECTING.

4.8 Client REBINDING state changes

The system will transition to Provisionally BOUND upon receipt of a DHCPACK that contains a DHCP authentication option, and a DHCP Server name option.

The broadcast DHCPREQUEST SHOULD be signed using authentication option, as with the one sent by state SELECTING.

4.9 Changes in processing for DHCP servers.

The Dynamic Host Configuration Protocol [RFC2131] describes how DHCP servers process DHCP messages. This document makes some changes to how messages are processed in the case where an Authentication option is provided that uses the dns-sigzero authentication protocol type.

When a DHCP server receives a message from a DHCP client that uses the DHCP authentication option with the dns-sigzero protocol, and it needs to respond to the DHCP client, it SHOULD respond with a signed message using the the dns-sigzero protocol. The DHCP server does not use the Client FQDN option to indicate where the client should look up its KEY record - any Client FQDN option the DHCP server sends contains the client's FQDN information. Instead, the IP address that the DHCP server sends in the server-identifier option MUST have a valid PTR record in the DNS. This PTR record MUST point to an FQDN that has a KEY record, and the KEY record MUST have a type of DNSSEC. The key stored in the KEY record is the public half of a public/private key pair. The DHCP server MUST use the private half of that key pair to sign the message that it sends to the DHCP client.

4.9.1 DHCP DISCOVER processing changes

Upon receipt of a DHCPDISCOVER that includes an Authentication option for the DHCP sigzero authentication protocol, the DHCP server MUST verify that the DHCPDISCOVER includes a client FQDN option, and that it is fully qualified. If the message does not contain a client FQDN option containing an FQDN, the DHCP server MUST drop the packet without further processing.

Otherwise, the server SHOULD do a secure DNS lookup on the provided FQDN, looking for a KEY resource record (sub-type DNSSEC). Having found a valid KEY (with the matching keyid), the server MAY verify the signature at this point. The server may defer authentication at

this step, for example if it is above a specified load threshold.

If appropriate authentication material is not found, then the DHCPDISCOVER SHOULD be treated as if it were not signed.

Otherwise, the DHCP server SHOULD be validate the signature on the message. If the signature cannot be validated, the DHCP server SHOULD log an audit entry that includes the keyid and FQDN that were specified used, and the signatures as computed and as specified in the signature on the message. The server should then drop the message without further processing.

If the DHCP server is able to successfully validate the signature, it SHOULD process the DHCPDISCOVER message as specified in [RFC2131].

4.9.2 DHCPREQUEST processing changes

Upon receipt of a DHCPREQUEST that includes an Authentication option for the DHCP sigzero authentication protocol, the DHCP server MUST verify that the DHCPREQUEST includes a client FQDN option, and that it is fully qualified. If the message does not contain a client FQDN option containing an FQDN, the DHCP server MUST drop the packet without further processing.

If the server does not have a cached copy of the KEY associated with the supplied FQDN as a result of some previous transaction (e.g., the DHCPDISCOVER/DHCPOFFER transaction), it MUST look up the record again, as described above.

The DHCP server SHOULD be validate the signature on the message at this point. If the signature cannot be validated, the DHCP server SHOULD log an audit entry that includes the keyid and FQDN that were specified used, and the signatures as computed and as specified in the signature on the message. The server should then drop the message without further processing.

If the DHCP server is able to successfully validate the signature, it SHOULD process the DHCPREQUEST message as specified in [RFC2131].

4.9.3 DHCPDECLINE processing changes

When the DHCP server receives an unsigned DHCPDECLINE message for a transaction where the preceding DHCPREQUEST message was signed using an Authentication option for the DHCP sigzero authentication protocol, the DHCP server SHOULD drop the DHCPDECLINE without

further processing.

Upon receipt of a DHCPDECLINE that includes an Authentication option for the DHCP sigzero authentication protocol, if the DHCPDECLINE does not include a client FQDN option that contains an FQDN, the DHCP server SHOULD drop the DHCPDECLINE packet without further processing. If the specified FQDN is not the same as the FQDN used to acquire the address being declined, the DHCP server SHOULD drop the DHCPDECLINE packet without further processing.

Otherwise, the DHCP server SHOULD retrieve the KEY record associated with the specified FQDN either from its cache or through a secure DNS lookup, and SHOULD validate the signature on the message.

If the signature cannot be validated, the DHCP server SHOULD log an audit entry that includes the keyid and FQDN that were specified used, and the signatures as computed and as specified in the signature on the message. The server should then drop the message without further processing.

If the DHCP server is able to successfully validate the signature, it SHOULD process the DHCPDECLINE message as specified in [RFC2131].

4.9.4 Annotated exchange between client and server

5. Security Considerations

This draft provides a new security mechanism for the DHCP protocol, which may in many cases provide enhanced security. However, like many security mechanisms, the work required to verify public key signatures and trace back through DNSSEC trees is substantial, and can be used against the DHCP server in a denial of service attack. It is also theoretically possible to use this against a DHCP client in a denial of service attack, although this may be somewhat more difficult.

6. IANA Considerations

Section 2 defines a new protocol code, as described in [RFC3118]. The value for this code is TBD. Sections 2 and three also describe algorithm codes specific to protocol 2. The algorithm codes are 1 for RSA and 2 for DSA.

7. Acknowledgments

This draft is the very belated result of a conversation with Randy Bush, with some kibbitzing from Johan Ihrens, Olaf Kolkman and a few others at a DHCP/DNS workshop in Amsterdam in January of 2001. The bulk of the idea came from Randy Bush, who was trying to come up with a better way for DHCP clients and servers to do DNS updates.

Normative references

[RFC3118] Droms, R., Editor, Arbaugh, W. and Editor, "Authentication for DHCP Messages", June 2001.

[RFC2131] Droms, R., "Dynamic Host Configuration Protocol", March 1997.

[RFC2132] Alexander, S. and R. Droms, "DHCP Options and BOOTP Vendor Extensions", RFC 2132, March 1997.

[RFC2537] Eastlake, D., "RSA/MD5 KEYs and SIGs in the Domain Name System (DNS)", March 1999.

[RFC3447] Jonsson, J. and B. Kaliski, "Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1", February 2003.

[DHCFQDN] Stapp, M. and Y. Rekhter, "The DHCP Client FQDN Option", ID (draft-ietf-dhc-fqdn-option-05.txt), November 2002.

Authors' Addresses

Ted Lemon
Nominum, Inc.
2385 Bay Road
Redwood City, CA 94063
USA

EMail: mellon@nominum.com

Michael C. Richardson
Sandelman Software Works
470 Dawson Avenue
Ottawa, ON K1Z 5V7

CA

EMail: mcr@sandelman.ottawa.on.ca
URI: http://www.sandelman.ottawa.on.ca/

Full Copyright Statement

Acknowledgement

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