dnsmasq

Environmental Information

Ubuntu22.04

dnsmasq-2.91

Network card configuration

```Bash
root@5bfa19bcb2f8:~/projects/dnsmasq-2.91/src# ifconfig
dhcp-test: flags=195<UP,BROADCAST,RUNNING,NOARP>  mtu 1500
        inet 192.168.100.1  netmask 255.255.255.0  broadcast 0.0.0.0
        inet6 fe80::ece6:eff:feb6:c837  prefixlen 64  scopeid 0x20<link>
        ether ee:e6:0e:b6:c8:37  txqueuelen 1000  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 51536  bytes 17403433 (17.4 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 172.17.0.5  netmask 255.255.0.0  broadcast 172.17.255.255
        ether 36:a7:3d:b9:4e:eb  txqueuelen 0  (Ethernet)
        RX packets 9487524  bytes 9496935139 (9.4 GB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 9167813  bytes 7466589563 (7.4 GB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 296956686  bytes 59034980140 (59.0 GB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 296956686  bytes 59034980140 (59.0 GB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

vethc: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet6 fe80::1c1b:beff:fe48:d192  prefixlen 64  scopeid 0x20<link>
        ether 1e:1b:be:48:d1:92  txqueuelen 1000  (Ethernet)
        RX packets 594  bytes 83566 (83.5 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 165  bytes 19025 (19.0 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

veths: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet6 fe80::100d:39ff:feea:5599  prefixlen 64  scopeid 0x20<link>
        inet6 2001:db8::1  prefixlen 64  scopeid 0x0<global>
        ether 12:0d:39:ea:55:99  txqueuelen 1000  (Ethernet)
        RX packets 165  bytes 19025 (19.0 KB)
```

```
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 594  bytes 83566 (83.5 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

dnsmasq.conf

```yaml
interface=dhcp-test
bind-interfaces

port=0

log-dhcp
log-queries

dhcp-range=2001:db8::100, 2001:db8::200, 64, 12h

dhcp-option=option6:dns-server,[2001:db8::1]

dhcp-option=option6:domain-search,example.local

dhcp-option=option6:ntp-server,[2001:db8::1]

# 启用路由器通告（RA）
enable-ra

# RA参数配置
ra-param=dhcp-test,high,0,3600

# 静态地址分配示例（可选）
# dhcp-host=id:00:01:02:03:04:05,[2001:db8::50],client1

# 状态文件位置
dhcp-leasefile=/var/lib/dhcp/dnsmasq.leases

# PID文件
pid-file=/var/run/dnsmasq.pid
```

tshark –i lo –Y "dhcpv6" –O dhcpv6

./dnsmasq ––no–daemon ––log–queries –C ./dnsmasq.conf

Violation Report

1. Server Fails to Discard Unicast Solicit Message

```markdown
RFC 3315 & RFC8415
15. Message Validation
A server MUST discard any Solicit, Confirm, Rebind or Information-request messages
it receives with a unicast destination address.
```

According to RFC 3315 & RFC8415, Section 15, "A server **MUST** discard any Solicit... messages it receives with a unicast destination address." The `Solicit` message is intended for discovering available DHCPv6 servers and therefore must be sent to a multicast address.

Our test confirms that `dnsmasq` processes and responds to a `Solicit` message sent directly to its unicast address.

Reproduce

Action: A `Solicit` message was sent directly to the server's unicast address (`2001:db8::1`) on port 547.

Observed Behavior: The `tshark` capture clearly shows the server at `2001:db8::1` receiving the unicast `Solicit` message (Frame 218) and subsequently responding with an `Advertise` message (Frame 239).

Expected Behavior: The server **MUST** have silently discarded the unicast `Solicit` message and sent no response.

Bash

```
echo "0155acc30001000e000100010000000eee60eb6c8370003000c0000000100000000000000000006000400170018" | xxd -r -p | nc -6u 2001:db8::1 547
```

The server's `Advertise` response is definitive proof of the violation.

Bash

```
root@5bfa19bcb2f8:~/projects/dnsmasq-2.91/src# tshark -i lo -Y "dhcpv6" -O dhcpv6
Running as user "root" and group "root". This could be dangerous.
Capturing on 'Loopback: lo'
 ** (tshark:4182782) 14:38:54.972521 [Main MESSAGE] -- Capture started.
 ** (tshark:4182782) 14:38:54.972564 [Main MESSAGE] -- File: "/tmp/wireshark_loFTGC82.pcapng"
Frame 218: 108 bytes on wire (864 bits), 108 bytes captured (864 bits) on interface lo, id 0
Ethernet II, Src: 00:00:00_00:00:00 (00:00:00:00:00:00), Dst: 00:00:00_00:00:00 (00:00:00:00:00:00)
Internet Protocol Version 6, Src: 2001:db8::1, Dst: 2001:db8::1
User Datagram Protocol, Src Port: 59785, Dst Port: 547
DHCPv6
    Message type: Solicit (1)
    Transaction ID: 0x55acc3
    Client Identifier
        Option: Client Identifier (1)
        Length: 14
        DUID: 000100010000000eee60eb6c837
        DUID Type: link-layer address plus time (1)
        Hardware type: Ethernet (1)
        DUID Time: Jan  1, 2000 00:00:00.000000000 UTC
        Link-layer address: ee:e6:0e:b6:c8:37
    Identity Association for Non-temporary Address
        Option: Identity Association for Non-temporary Address (3)
        Length: 12
```

```
        IAID: 00000001
        T1: 0
        T2: 0
    Option Request
        Option: Option Request (6)
        Length: 4
        Requested Option code: DNS recursive name server (23)
        Requested Option code: Domain Search List (24)

Frame 239: 203 bytes on wire (1624 bits), 203 bytes captured (1624 bits) on interface lo, id 0
Ethernet II, Src: 00:00:00_00:00:00 (00:00:00:00:00:00), Dst: 00:00:00_00:00:00 (00:00:00:00:00:00)
Internet Protocol Version 6, Src: 2001:db8::1, Dst: 2001:db8::1
User Datagram Protocol, Src Port: 547, Dst Port: 546
DHCPv6
    Message type: Advertise (2)
    Transaction ID: 0x55acc3
    Client Identifier
        Option: Client Identifier (1)
        Length: 14
        DUID: 000100010000000eee60eb6c837
        DUID Type: link-layer address plus time (1)
        Hardware type: Ethernet (1)
        DUID Time: Jan  1, 2000 00:00:00.000000000 UTC
        Link-layer address: ee:e6:0e:b6:c8:37
    Server Identifier
        Option: Server Identifier (2)
        Length: 14
        DUID: 000100012feeadfb36a73db94eeb
        DUID Type: link-layer address plus time (1)
        Hardware type: Ethernet (1)
        DUID Time: Jun 25, 2025 12:54:19.000000000 UTC
        Link-layer address: 36:a7:3d:b9:4e:eb
    Identity Association for Non-temporary Address
        Option: Identity Association for Non-temporary Address (3)
        Length: 40
        IAID: 00000001
        T1: 21600
        T2: 37800
        IA Address
            Option: IA Address (5)
            Length: 24
            IPv6 address: 2001:db8::102
            Preferred lifetime: 43200
            Valid lifetime: 43200
    Status code
        Option: Status code (13)
        Length: 9
        Status Code: Success (0)
        Status Message: success
    Preference
```

```
        Option: Preference (7)
        Length: 1
        Pref-value: 0
    Domain Search List
        Option: Domain Search List (24)
        Length: 15
        Domain name suffix search list
            List entry: example.local.
    DNS recursive name server
        Option: DNS recursive name server (23)
        Length: 16
         1 DNS server address: 2001:db8::1

Frame 240: 251 bytes on wire (2008 bits), 251 bytes captured (2008 bits) on inter
face lo, id 0
Ethernet II, Src: 00:00:00_00:00:00 (00:00:00:00:00:00), Dst: 00:00:00_00:00:00
(00:00:00:00:00:00)
Internet Protocol Version 6, Src: 2001:db8::1, Dst: 2001:db8::1
Internet Control Message Protocol v6
DHCPv6
    Message type: Advertise (2)
    Transaction ID: 0x55acc3
    Client Identifier
        Option: Client Identifier (1)
        Length: 14
        DUID: 0001000100000000eee60eb6c837
        DUID Type: link-layer address plus time (1)
        Hardware type: Ethernet (1)
        DUID Time: Jan  1, 2000 00:00:00.000000000 UTC
        Link-layer address: ee:e6:0e:b6:c8:37
    Server Identifier
        Option: Server Identifier (2)
        Length: 14
        DUID: 000100012feeadfb36a73db94eeb
        DUID Type: link-layer address plus time (1)
        Hardware type: Ethernet (1)
        DUID Time: Jun 25, 2025 12:54:19.000000000 UTC
        Link-layer address: 36:a7:3d:b9:4e:eb
    Identity Association for Non-temporary Address
        Option: Identity Association for Non-temporary Address (3)
        Length: 40
        IAID: 00000001
        T1: 21600
        T2: 37800
        IA Address
            Option: IA Address (5)
            Length: 24
            IPv6 address: 2001:db8::102
            Preferred lifetime: 43200
            Valid lifetime: 43200
    Status code
        Option: Status code (13)
```

```
        Length: 9
        Status Code: Success (0)
        Status Message: success
    Preference
        Option: Preference (7)
        Length: 1
        Pref-value: 0
    Domain Search List
        Option: Domain Search List (24)
        Length: 15
        Domain name suffix search list
            List entry: example.local.
    DNS recursive name server
        Option: DNS recursive name server (23)
        Length: 16
```

3. Unconditional New Binding Creation on `Rebind`, Leading to Address Pool Exhaustion

According to RFC 8415, Section 18.2.2:

Markdown

```
RFC 8415
18.3.5. Receipt of Rebind Messages
Therefore, the server SHOULD only create new bindings during processing of a Rebind message if the server is configured to respond with a Reply message to a Solicit message containing the Rapid Commit option.

```

Our analysis of the `dnsmasq` source code indicates that the `dhcp6_no_relay` function, which handles the `Rebind` message, does not adhere to this guideline. When the function receives a `Rebind` for a lease that it does not currently have on record, it proceeds to call `lease6_allocate` to create a new binding for the client unconditionally. The implementation is missing the required check of the server's Rapid Commit configuration before creating this new binding.

This means an attacker does not need to go through the normal `Solicit->Advertise->Request->Reply` (SARR) exchange; they can directly trigger a new lease creation simply by sending a `Rebind` message for a lease that does not exist.

Reproduce:

Bash

```
root@5bfa19bcb2f8:~/projects/dnsmasq-2.91/src# echo "06dd9a2c0001000e00010001685d3c4f001122334455000300280000000100000000000000000005001820010db80000000000000000000001230000003c00000078" | xxd -r -p | nc -6u -q1 2001:db8::1 547
```

Markdown

```
Frame 17713: 128 bytes on wire (1024 bits), 128 bytes captured (1024 bits) on interface lo, id 0
Ethernet II, Src: 00:00:00_00:00:00 (00:00:00:00:00:00), Dst: 00:00:00_00:00:00 (00:00:00:00:00:00)
```

```
Internet Protocol Version 6, Src: 2001:db8::1, Dst: 2001:db8::1
User Datagram Protocol, Src Port: 37101, Dst Port: 547
DHCPv6
    Message type: Rebind (6)
    Transaction ID: 0xdd9a2c
    Client Identifier
        Option: Client Identifier (1)
        Length: 14
        DUID: 00010001685d3c4f001122334455
        DUID Type: link-layer address plus time (1)
        Hardware type: Ethernet (1)
        DUID Time: Jun 26, 2055 12:25:51.000000000 UTC
        Link-layer address: 00:11:22:33:44:55
    Identity Association for Non-temporary Address
        Option: Identity Association for Non-temporary Address (3)
        Length: 40
        IAID: 00000001
        T1: 0
        T2: 0
        IA Address
            Option: IA Address (5)
            Length: 24
            IPv6 address: 2001:db8::123
            Preferred lifetime: 60
            Valid lifetime: 120

Frame 17813: 209 bytes on wire (1672 bits), 209 bytes captured (1672 bits) on inte
rface lo, id 0
Ethernet II, Src: 00:00:00_00:00:00 (00:00:00:00:00:00), Dst: 00:00:00_00:00:00 (0
0:00:00:00:00:00)
Internet Protocol Version 6, Src: 2001:db8::1, Dst: 2001:db8::1
User Datagram Protocol, Src Port: 547, Dst Port: 546
DHCPv6
    Message type: Reply (7)
    Transaction ID: 0xdd9a2c
    Client Identifier
        Option: Client Identifier (1)
        Length: 14
        DUID: 00010001685d3c4f001122334455
        DUID Type: link-layer address plus time (1)
        Hardware type: Ethernet (1)
        DUID Time: Jun 26, 2055 12:25:51.000000000 UTC
        Link-layer address: 00:11:22:33:44:55
    Server Identifier
        Option: Server Identifier (2)
        Length: 14
        DUID: 00010001300a4cd936a73db94eeb
        DUID Type: link-layer address plus time (1)
        Hardware type: Ethernet (1)
        DUID Time: Jul 16, 2025 11:43:21.000000000 UTC
        Link-layer address: 36:a7:3d:b9:4e:eb
    Identity Association for Non-temporary Address
```

```
        Option: Identity Association for Non-temporary Address (3)
        Length: 40
        IAID: 00000001
        T1: 54
        T2: 99
        IA Address
            Option: IA Address (5)
            Length: 24
            IPv6 address: 2001:db8::123
            Preferred lifetime: 120
            Valid lifetime: 120
    NTP Server
        Option: NTP Server (56)
        Length: 20
        NTP Server Address
            Suboption: NTP Server Address (1)
            Length: 16
            NTP Server Address: 2001:db8::1
    Domain Search List
        Option: Domain Search List (24)
        Length: 15
        Domain name suffix search list
            List entry: example.local.
    DNS recursive name server
        Option: DNS recursive name server (23)
        Length: 16
         1 DNS server address: 2001:db8::1
```

Bash

```
root@5bfa19bcb2f8:~/projects/dnsmasq-2.91/src# cat /var/lib/dhcp/dnsmasq.leases
duid 00:01:00:01:30:0a:4c:d9:36:a7:3d:b9:4e:eb
1752667777 1 2001:db8::123 * *   # ->[new]<-
1752710104 1 2001:db8::113 * 00:01:00:01:68:5d:0f:51:00:11:22:33:44:55
1752709993 1 2001:db8::1e4 * 00:01:00:01:68:5d:0f:41:00:11:22:33:44:55
root@5bfa19bcb2f8:~/projects/dnsmasq-2.91/src#
```

An attacker can repeatedly send `Rebind` messages with spoofed, unique Client Identifiers (DUIDs). For each malicious `Rebind` packet, `dnsmasq` will incorrectly and unconditionally create a new lease, reserving an IPv6 address from its pool. This process can be repeated until the server's entire address pool is depleted.

Once the address pool is exhausted, legitimate devices on the network will be unable to obtain new leases or renew existing ones, effectively cutting them off from network access and causing a widespread service outage.

4. Unconditional Lease Creation Without Configuration Check

Markdown

```
RFC 8415
18.3.5.  Receipt of Rebind Messages
```

```
3  If the server is configured to create new bindings as a result of processing Rebind
   messages (also see the note below about the Rapid Commit option (Section 21.14)), t
   he server SHOULD create a binding and return the IA with allocated leases with life
   times and, if applicable, T1/T2 values and other information requested by the clien
   t.
```

We have observed that when processing Rebind messages, the code unconditionally creates a new lease if an existing lease is not found, without verifying whether the server is configured to allow the creation of new bindings in response to Rebind messages. Consequently, we have classified this behavior as non-compliant. Nevertheless, while this appears to be a minor issue, we have retained the classification to ensure completeness.

5. Violation: Ignoring Zero Link-Address Field

Markdown

```
1  rfc8415
2  13.1.  Selecting Addresses for Assignment to an IA_NA
3  According to [RFC6221], the server MUST ignore any link-address field whose value i
   s zero.
```

We have discovered that when handling DHCPv6 relay-forward messages, the code copies the link address field (which may be zero) into the status structure without checking for a zero value. This violates Section 3.1 of RFC6221, which requires that servers must ignore any link address fields with a value of zero.

Bash

```
1  root@5bfa19bcb2f8:~/projects/dnsmasq-2.91/src# echo "0c000000000000000000000000000000
   000000fe8000000000000000001122fffe3344660009002601a5342e0001000e000100010000000000112
   233446600030000c0000000100000000000000000" | xxd -r -p | nc -6u -q1 2001:db8::1 547ec
   ho "0c000000000000000000000000000000000fe8000000000000001122fffe3344660009002601a
   5342e0001000e000100010000000001122334466000300000c0000000100000000000000000" | xxd -r
   -p | nc -6u -q1 2001:db8::1 547
```

Markdown

```
1  Frame 25386: 138 bytes on wire (1104 bits), 138 bytes captured (1104 bits) on inte
   rface lo, id 0
2  Ethernet II, Src: 00:00:00_00:00:00 (00:00:00:00:00:00), Dst: 00:00:00_00:00:00 (0
   0:00:00:00:00:00)
3  Internet Protocol Version 6, Src: 2001:db8::1, Dst: 2001:db8::1
4  User Datagram Protocol, Src Port: 57034, Dst Port: 547
5  DHCPv6
6      Message type: Relay-forw (12)
7      Hopcount: 0
8      Link address: ::
9      Peer address: fe80::11:22ff:fe33:4466
10     Relay Message
11         Option: Relay Message (9)
12         Length: 38
13         DHCPv6
14             Message type: Solicit (1)
15             Transaction ID: 0xa5342e
```

```
16              Client Identifier
17                  Option: Client Identifier (1)
18                  Length: 14
19                  DUID: 000100010000000000001122334466
20                  DUID Type: link-layer address plus time (1)
21                  Hardware type: Ethernet (1)
22                  DUID Time: Jan  1, 2000 00:00:00.000000000 UTC
23                  Link-layer address: 00:11:22:33:44:66
24              Identity Association for Non-temporary Address
25                  Option: Identity Association for Non-temporary Address (3)
26                  Length: 12
27                  IAID: 00000001
28                  T1: 0
```

As expected, the program should have discarded such messages instead of continuing to process them, but the log:

Markdown

```
dnsmasq-dhcp: no address range available for DHCPv6 request from relay at ::
dnsmasq-dhcp: RTR-ADVERT(veths) 2001:db8::
```

6. Violation in Handling Relay-forward Messages with Hop-Count Limit

Markdown

```
rfc8415
19.1.2.  Relaying a Message from a Relay Agent
If the message received by the relay agent is a Relay-forward message and the hop-count value in the message is greater than or equal to HOP_COUNT_LIMIT, the relay agent discards the received message.

rfc3315
20.1.2. Relaying a Message from a Relay Agent
If the message received by the relay agent is a Relay-forward message and the hop-count in the message is greater than or equal to HOP_COUNT_LIMIT, the relay agent discards the received message.
```

This issue is small but really very interesting.

The code fails to discard Relay-forward messages when hop-count equals HOP_COUNT_LIMIT (32). The rule requires discarding if hop-count >= 32, but the code only checks for hopcount > 32.

```
2183    /* RFC 3315 HOP_COUNT_LIMIT */
2184    if (hopcount > 32 || !(header = put_opt6(NULL, 34)))
2185      return 1;
2186
```

7. Incorrect Use of Interface-Id Option in DHCPv6 Messages

纯文本

```
rfc8415
21.18.  Interface-Id Option
This option MUST NOT appear in any message except a Relay-forward or Relay-reply message.

rfc3315
22.18. Interface-Id Option
This option MUST NOT appear in any message except a Relay-Forward or Relay-Reply message.
```

According to the description of this rule, the Interface–Id Option must not appear in any message other than Relay–forward or Relay–reply. However, when we constructed such messages, the program did not check or reject them.

Bash

```
root@5bfa19bcb2f8:~/projects/dnsmasq-2.91/src# echo "010030390001000e00010001000000000112233445500030000c0000000100000000000000000120014746573742d696e746572666163652d3132333435" | xxd -r -p | nc -6u -q1 2001:db8::1 547
```

Python

```
Frame 30586: 124 bytes on wire (992 bits), 124 bytes captured (992 bits) on interface lo, id 0
Ethernet II, Src: 00:00:00_00:00:00 (00:00:00:00:00:00), Dst: 00:00:00_00:00:00 (00:00:00:00:00:00)
Internet Protocol Version 6, Src: 2001:db8::1, Dst: 2001:db8::1
User Datagram Protocol, Src Port: 34285, Dst Port: 547
DHCPv6
    Message type: Solicit (1)
    Transaction ID: 0x003039
    Client Identifier
        Option: Client Identifier (1)
        Length: 14
        DUID: 0001000100000000001122334455
        DUID Type: link-layer address plus time (1)
        Hardware type: Ethernet (1)
        DUID Time: Jan  1, 2000 00:00:00.000000000 UTC
        Link-layer address: 00:11:22:33:44:55
    Identity Association for Non-temporary Address
        Option: Identity Association for Non-temporary Address (3)
        Length: 12
        IAID: 00000001
        T1: 0
        T2: 0
    Interface-Id
        Option: Interface-Id (18)
        Length: 20
        Interface-ID: 746573742d696e746572666163652d3132333435

Frame 30613: 227 bytes on wire (1816 bits), 227 bytes captured (1816 bits) on interface lo, id 0
Ethernet II, Src: 00:00:00_00:00:00 (00:00:00:00:00:00), Dst: 00:00:00_00:00:00 (0
```

```
0:00:00:00:00:00)
Internet Protocol Version 6, Src: 2001:db8::1, Dst: 2001:db8::1
User Datagram Protocol, Src Port: 547, Dst Port: 546
DHCPv6
    Message type: Advertise (2)
    Transaction ID: 0x003039
    Client Identifier
        Option: Client Identifier (1)
        Length: 14
        DUID: 0001000100000000001122334455
        DUID Type: link-layer address plus time (1)
        Hardware type: Ethernet (1)
        DUID Time: Jan  1, 2000 00:00:00.000000000 UTC
        Link-layer address: 00:11:22:33:44:55
    Server Identifier
        Option: Server Identifier (2)
        Length: 14
        DUID: 00010001300a4cd936a73db94eeb
        DUID Type: link-layer address plus time (1)
        Hardware type: Ethernet (1)
        DUID Time: Jul 16, 2025 11:43:21.000000000 UTC
        Link-layer address: 36:a7:3d:b9:4e:eb
    Identity Association for Non-temporary Address
        Option: Identity Association for Non-temporary Address (3)
        Length: 40
        IAID: 00000001
        T1: 21600
        T2: 37800
        IA Address
            Option: IA Address (5)
            Length: 24
            IPv6 address: 2001:db8::1da
            Preferred lifetime: 43200
            Valid lifetime: 43200
    Status code
        Option: Status code (13)
        Length: 9
        Status Code: Success (0)
        Status Message: success
    Preference
        Option: Preference (7)
        Length: 1
        Pref-value: 0
    NTP Server
        Option: NTP Server (56)
        Length: 20
        NTP Server Address
            Suboption: NTP Server Address (1)
            Length: 16
            NTP Server Address: 2001:db8::1
    Domain Search List
        Option: Domain Search List (24)
```

```
80          Length: 15
81          Domain name suffix search list
82              List entry: example.local.
83      DNS recursive name server
84          Option: DNS recursive name server (23)
85          Length: 16
86           1 DNS server address: 2001:db8::1
```

8. Incorrect Handling of NoAddrsAvail Status Code for IA Option

Markdown

```
rfc8415
18.3.2.  Receipt of Request Messages
If the server does not send the NotOnLink status code but it cannot assign any IP addresses to an IA, the server MUST return the IA option in the Reply message with no addresses in the IA and a Status Code option containing status code NoAddrsAvail in the IA.
```

Our analysis reveals that when there are no available addresses in the IA (Identity Association) within a DHCPv6 REQUEST, the server adds a top-level status code option (NoAddrsAvail) instead of embedding it within the IA option as required. According to the specifications, the IA option must include the indication of no available addresses and embed the status code option (NoAddrsAvail) inside the IA itself.

To verify this issue, we first exhausted the address pool of the DHCPv6 server and then sent the request packet.

Bash

```
echo "030030390001000e000100010000000000011223344550002000e000100012feeadfb36a73db94eeb0003000c0000000100000000000000000000500182001 0db8000000000000000000000010000000e1000001c20" | xxd -r -p | nc -6u -q1 2001:db8::1 547
```

Markdown

```
root@5bfa19bcb2f8:~/projects/dnsmasq-2.91/src# tshark -i lo -Y "dhcpv6" -O dhcpv6
Running as user "root" and group "root". This could be dangerous.
Capturing on 'Loopback: lo'
 ** (tshark:1593669) 06:27:22.152591 [Main MESSAGE] -- Capture started.
 ** (tshark:1593669) 06:27:22.152641 [Main MESSAGE] -- File: "/tmp/wireshark_loUPVR82.pcapng"
Frame 84: 146 bytes on wire (1168 bits), 146 bytes captured (1168 bits) on interface lo, id 0
Ethernet II, Src: 00:00:00_00:00:00 (00:00:00:00:00:00), Dst: 00:00:00_00:00:00 (00:00:00:00:00:00)
Internet Protocol Version 6, Src: 2001:db8::1, Dst: 2001:db8::1
User Datagram Protocol, Src Port: 56518, Dst Port: 547
DHCPv6
    Message type: Request (3)
    Transaction ID: 0x003039
    Client Identifier
        Option: Client Identifier (1)
```

```
        Length: 14
        DUID: 000100010000000000001122334455
        DUID Type: link-layer address plus time (1)
        Hardware type: Ethernet (1)
        DUID Time: Jan  1, 2000 00:00:00.000000000 UTC
        Link-layer address: 00:11:22:33:44:55
    Server Identifier
        Option: Server Identifier (2)
        Length: 14
        DUID: 000100012feeadfb36a73db94eeb
        DUID Type: link-layer address plus time (1)
        Hardware type: Ethernet (1)
        DUID Time: Jun 25, 2025 12:54:19.000000000 UTC
        Link-layer address: 36:a7:3d:b9:4e:eb
    Identity Association for Non-temporary Address
        Option: Identity Association for Non-temporary Address (3)
        Length: 12
        IAID: 00000001
        T1: 0
        T2: 0
    IA Address
        Option: IA Address (5)
        Length: 24
        IPv6 address: 2001:db8::100
        Preferred lifetime: 3600
        Valid lifetime: 7200

Frame 122: 224 bytes on wire (1792 bits), 224 bytes captured (1792 bits) on inter
face lo, id 0
Ethernet II, Src: 00:00:00_00:00:00 (00:00:00:00:00:00), Dst: 00:00:00_00:00:00
(00:00:00:00:00:00)
Internet Protocol Version 6, Src: 2001:db8::1, Dst: 2001:db8::1
User Datagram Protocol, Src Port: 547, Dst Port: 546
DHCPv6
    Message type: Reply (7)
    Transaction ID: 0x003039
    Client Identifier
        Option: Client Identifier (1)
        Length: 14
        DUID: 000100010000000000001122334455
        DUID Type: link-layer address plus time (1)
        Hardware type: Ethernet (1)
        DUID Time: Jan  1, 2000 00:00:00.000000000 UTC
        Link-layer address: 00:11:22:33:44:55
    Server Identifier
        Option: Server Identifier (2)
        Length: 14
        DUID: 000100012feeadfb36a73db94eeb
        DUID Type: link-layer address plus time (1)
        Hardware type: Ethernet (1)
        DUID Time: Jun 25, 2025 12:54:19.000000000 UTC
        Link-layer address: 36:a7:3d:b9:4e:eb
```

```
    Identity Association for Non-temporary Address
        Option: Identity Association for Non-temporary Address (3)
        Length: 37
        IAID: 00000001
        T1: infinity
        T2: infinity
        Status code
            Option: Status code (13)
            Length: 21
            Status Code: NoAddrAvail (2)
            Status Message: address unavailable
    Status code
        Option: Status code (13)
        Length: 9
        Status Code: Success (0)
        Status Message: success
    Preference
        Option: Preference (7)
        Length: 1
        Pref-value: 0
    NTP Server
        Option: NTP Server (56)
        Length: 20
        NTP Server Address
            Suboption: NTP Server Address (1)
            Length: 16
            NTP Server Address: 2001:db8::1
    Domain Search List
        Option: Domain Search List (24)
        Length: 15
        Domain name suffix search list
            List entry: example.local.
    DNS recursive name server
        Option: DNS recursive name server (23)
        Length: 16
```

9. Missing Requested Options When No Addresses Are Available

纯文本

```
rfc8415
18.3.9.  Creation of Advertise Messages
The server MUST include options in the Advertise message containing configuration p
arameters for all of the options identified in the Option Request option (see Secti
on 21.7) in the Solicit message that the server has been configured to return to th
e client.
```

We found that the server does not include requested options in the Advertise message when no addresses are available. The rule mandates that the server MUST include all requested options

(from the Option Request option in the Solicit) that it is configured to return, regardless of address assignment status.

We constructed a Solicit message explicitly requesting the DNS recursive name server (Option 23) and the domain search list (Option 24). However, in the Advertise message, these options were not included; only the client identifier, server identifier, and the status code "NoAddrAvail" were returned. This indicates that the server was unable to allocate an address and did not provide the requested option content.

Bash

```
root@5bfa19bcb2f8:~/projects/dnsmasq-2.91/src# echo "010030390001000e00010001000000000011223344550003000c0000000100000000000000000000600040017001
8" | xxd -r -p | nc -6u -q1 2001:db8::1 547
```

Markdown

```
root@5bfa19bcb2f8:~/projects/dnsmasq-2.91/src# tshark -i lo -Y "dhcpv6" -O dhcpv6
Running as user "root" and group "root". This could be dangerous.
Capturing on 'Loopback: lo'
 ** (tshark:1613859) 06:40:21.568514 [Main MESSAGE] -- Capture started.
 ** (tshark:1613859) 06:40:21.568556 [Main MESSAGE] -- File: "/tmp/wireshark_lo77FI82.pcapng"
Frame 60: 108 bytes on wire (864 bits), 108 bytes captured (864 bits) on interface lo, id 0
Ethernet II, Src: 00:00:00_00:00:00 (00:00:00:00:00:00), Dst: 00:00:00_00:00:00 (00:00:00:00:00:00)
Internet Protocol Version 6, Src: 2001:db8::1, Dst: 2001:db8::1
User Datagram Protocol, Src Port: 51651, Dst Port: 547
DHCPv6
    Message type: Solicit (1)
    Transaction ID: 0x003039
    Client Identifier
        Option: Client Identifier (1)
        Length: 14
        DUID: 0001000100000000001122334455
        DUID Type: link-layer address plus time (1)
        Hardware type: Ethernet (1)
        DUID Time: Jan  1, 2000 00:00:00.000000000 UTC
        Link-layer address: 00:11:22:33:44:55
    Identity Association for Non-temporary Address
        Option: Identity Association for Non-temporary Address (3)
        Length: 12
        IAID: 00000001
        T1: 0
        T2: 0
    Option Request
        Option: Option Request (6)
        Length: 4
        Requested Option code: DNS recursive name server (23)
        Requested Option code: Domain Search List (24)

Frame 106: 130 bytes on wire (1040 bits), 130 bytes captured (1040 bits) on interface lo, id 0
```

```
Ethernet II, Src: 00:00:00_00:00:00 (00:00:00:00:00:00), Dst: 00:00:00_00:00:00 (00:00:00:00:00:00)
Internet Protocol Version 6, Src: 2001:db8::1, Dst: 2001:db8::1
User Datagram Protocol, Src Port: 547, Dst Port: 546
DHCPv6
    Message type: Advertise (2)
    Transaction ID: 0x003039
    Client Identifier
        Option: Client Identifier (1)
        Length: 14
        DUID: 0001000100000000001122334455
        DUID Type: link-layer address plus time (1)
        Hardware type: Ethernet (1)
        DUID Time: Jan  1, 2000 00:00:00.000000000 UTC
        Link-layer address: 00:11:22:33:44:55
    Server Identifier
        Option: Server Identifier (2)
        Length: 14
        DUID: 000100012feeadfb36a73db94eeb
        DUID Type: link-layer address plus time (1)
        Hardware type: Ethernet (1)
        DUID Time: Jun 25, 2025 12:54:19.000000000 UTC
        Link-layer address: 36:a7:3d:b9:4e:eb
    Status code
        Option: Status code (13)
        Length: 24
        Status Code: NoAddrAvail (2)
        Status Message: no addresses available
```

10. Violation of DHCPv6 IA_PD Handling in Advertise Messages

Markdown

```
rfc8415
18.3.9.  Creation of Advertise Messages
The server MUST include IA options in the Advertise message containing any addresses and/or delegated prefixes that would be assigned to IAs contained in the Solicit message from the client.

rfc3315
17.2.2. Creation and Transmission of Advertise Messages
If the Solicit message from the client included one or more IA options, the server MUST include IA options in the Advertise message containing any addresses that would be assigned to IAs contained in the Solicit message from the client.
```

We found that dnsmasq does not handle the IA_PD option in request messages. This is because only IA_NA and IA_TA are processed in the check_ia function, which explicitly filters out the IA_PD option (OPTION6_IA_PD). As a result, when constructing the Advertise message, the IA_PD option in the request message is ignored, which violates relevant rules.

C

```c
static int check_ia(struct state *state, void *opt, void **endp, void **ia_option)
{
  state->ia_type = opt6_type(opt);
  *ia_option = NULL;

  if (state->ia_type != OPTION6_IA_NA && state->ia_type != OPTION6_IA_TA)
    return 0;

  if (state->ia_type == OPTION6_IA_NA && opt6_len(opt) < 12)
    return 0;

  if (state->ia_type == OPTION6_IA_TA && opt6_len(opt) < 4)
    return 0;

  *endp = opt6_ptr(opt, opt6_len(opt));
  state->iaid = opt6_uint(opt, 0, 4);
  *ia_option = opt6_find(opt6_ptr(opt, state->ia_type == OPTION6_IA_NA ? 12 : 4), *endp, OPTION6_IAADDR, 24);

  return 1;
}
```

We constructed a Solicit message that includes an IA_PD option, but the server did not return an Advertise that contains IA_PD.

Bash

```
root@5bfa19bcb2f8:~/projects/dnsmasq-2.91/src# echo "011718020001000e00010001000000
00eee60eb6c8370019000c0000000100000000000000000" | xxd -r -p | nc -6u -q1 2001:db8::
1 547
```

纯文本

```
Frame 40917: 100 bytes on wire (800 bits), 100 bytes captured (800 bits) on interf
ace lo, id 0
Ethernet II, Src: 00:00:00_00:00:00 (00:00:00:00:00:00), Dst: 00:00:00_00:00:00 (0
0:00:00:00:00:00)
Internet Protocol Version 6, Src: 2001:db8::1, Dst: 2001:db8::1
User Datagram Protocol, Src Port: 45731, Dst Port: 547
DHCPv6
    Message type: Solicit (1)
    Transaction ID: 0x171802
    Client Identifier
        Option: Client Identifier (1)
        Length: 14
        DUID: 000100010000000eee60eb6c837
        DUID Type: link-layer address plus time (1)
        Hardware type: Ethernet (1)
        DUID Time: Jan  1, 2000 00:00:00.000000000 UTC
        Link-layer address: ee:e6:0e:b6:c8:37
    Identity Association for Prefix Delegation
        Option: Identity Association for Prefix Delegation (25)
        Length: 12
        IAID: 00000001

```

```
        T1: 0
        T2: 0

Frame 40948: 130 bytes on wire (1040 bits), 130 bytes captured (1040 bits) on interface lo, id 0
Ethernet II, Src: 00:00:00_00:00:00 (00:00:00:00:00:00), Dst: 00:00:00_00:00:00 (00:00:00:00:00:00)
Internet Protocol Version 6, Src: 2001:db8::1, Dst: 2001:db8::1
User Datagram Protocol, Src Port: 547, Dst Port: 546
DHCPv6
    Message type: Advertise (2)
    Transaction ID: 0x171802
    Client Identifier
        Option: Client Identifier (1)
        Length: 14
        DUID: 000100010000000eee60eb6c837
        DUID Type: link-layer address plus time (1)
        Hardware type: Ethernet (1)
        DUID Time: Jan  1, 2000 00:00:00.000000000 UTC
        Link-layer address: ee:e6:0e:b6:c8:37
    Server Identifier
        Option: Server Identifier (2)
        Length: 14
        DUID: 00010001300a4cd936a73db94eeb
        DUID Type: link-layer address plus time (1)
        Hardware type: Ethernet (1)
        DUID Time: Jul 16, 2025 11:43:21.000000000 UTC
        Link-layer address: 36:a7:3d:b9:4e:eb
    Status code
        Option: Status code (13)
        Length: 24
        Status Code: NoAddrAvail (2)
```

11. DHCPv6 Relay Agent Fails to Include Mandatory `Interface-Id` Option

Markdown

```
rfc3315
20.1.1. Relaying a Message from a Client
If the relay agent cannot use the address in the link-address field to identify the interface through which the response to the client will be relayed, the relay agent MUST include an Interface-Id option (see Section 22.18) in the Relay-forward message.

rfc8415
19.1.1.  Relaying a Message from a Client
If the relay agent cannot use the address in the link-address field to identify the interface through which the response to the client will be relayed, the relay agent
```

```
    MUST include an Interface-Id option (see Section 21.18) in the Relay-forward messag
```

According to RFC 3315/8415, if a relay agent cannot guarantee that the `link-address` it uses will be sufficient for the DHCPv6 server to determine the correct return path, it **MUST** include an `Interface-Id` option to provide an unambiguous identifier for the client's interface.

Our code audit of the `relay_upstream6` function (in `src/rfc3315.c`) reveals the following flaw:

1. The function unconditionally copies a configured local address (`relay->local.addr6`) into the `link-address` field of the `Relay-forward` message. **Please see rfc3315.c line 2209**.
2. Crucially, it **lacks any validation logic** to check if this `link-address` (which could be the unspecified address `::` or an address shared by multiple interfaces) is actually sufficient to uniquely identify the interface on which the client's message was received.
3. Then, the function **never adds the `OPTION6_INTERFACE_ID` option** , even in scenarios where it is mandatory.

Thus, we consider it may be a violation.

We do not have dynamic validation due to environment configuration.

13. Missing Validation for Mandatory Options in DHCPv6 Option Request Option (ORO) Processing

Markdown

```
rfc8415
21.7.  Option Request Option
For certain message types, some option codes MUST be included in the Option Request
option; see Table 4 for details.
```

Bash

```
root@5bfa19bcb2f8:~/projects/dnsmasq-2.91/src# echo "031616160001000e00010001000000
00001122334455000200 0e00010001300a4cd936a73db94eeb0008000200000003002800000001000000
70800000a8c0005001899990db80badf00d0000000000001234000 00e1000001c20000600040017002
c" | xxd -r -p | nc -6u -q1 2001:db8
```

We specified LQ_QUERY in the Option Request of the Request message, but according to the provisions of Table 4, this is prohibited. Therefore, it can be determined that there is a lack of corresponding check logic in dnsmasq.

Markdown

```
+---------+--------------------------+------------------+-----------+
|  Option | Option Name ("OPTION"    | Client ORO (1)   | Singleton |
|         | prefix removed)          |                  | Option    |
+---------+--------------------------+------------------+-----------+
|      44 | LQ_QUERY                 | No               | Yes       |
```

Markdown

```
Frame 59557: 160 bytes on wire (1280 bits), 160 bytes captured (1280 bits) on inte
rface lo, id 0

```

```
Ethernet II, Src: 00:00:00_00:00:00 (00:00:00:00:00:00), Dst: 00:00:00_00:00:00 (0
0:00:00:00:00:00)
Internet Protocol Version 6, Src: 2001:db8::1, Dst: 2001:db8::1
User Datagram Protocol, Src Port: 52054, Dst Port: 547
DHCPv6
    Message type: Request (3)
    Transaction ID: 0x161616
    Client Identifier
        Option: Client Identifier (1)
        Length: 14
        DUID: 0001000100000000001122334455
        DUID Type: link-layer address plus time (1)
        Hardware type: Ethernet (1)
        DUID Time: Jan  1, 2000 00:00:00.000000000 UTC
        Link-layer address: 00:11:22:33:44:55
    Server Identifier
        Option: Server Identifier (2)
        Length: 14
        DUID: 00010001300a4cd936a73db94eeb
        DUID Type: link-layer address plus time (1)
        Hardware type: Ethernet (1)
        DUID Time: Jul 16, 2025 11:43:21.000000000 UTC
        Link-layer address: 36:a7:3d:b9:4e:eb
    Elapsed time
        Option: Elapsed time (8)
        Length: 2
        Elapsed time: 0ms
    Identity Association for Non-temporary Address
        Option: Identity Association for Non-temporary Address (3)
        Length: 40
        IAID: 00000001
        T1: 1800
        T2: 2700
        IA Address
            Option: IA Address (5)
            Length: 24
            IPv6 address: 9999:db8:bad:f00d::1234
            Preferred lifetime: 3600
            Valid lifetime: 7200
    Option Request
        Option: Option Request (6)
        Length: 4
        Requested Option code: DNS recursive name server (23)
        Requested Option code: Leasequery Query (44)

Frame 59576: 183 bytes on wire (1464 bits), 183 bytes captured (1464 bits) on inte
rface lo, id 0
Ethernet II, Src: 00:00:00_00:00:00 (00:00:00:00:00:00), Dst: 00:00:00_00:00:00 (0
0:00:00:00:00:00)
Internet Protocol Version 6, Src: 2001:db8::1, Dst: 2001:db8::1
User Datagram Protocol, Src Port: 547, Dst Port: 546
DHCPv6
```

```
    Message type: Reply (7)
    Transaction ID: 0x161616
    Client Identifier
        Option: Client Identifier (1)
        Length: 14
        DUID: 000100010000000000001122334455
        DUID Type: link-layer address plus time (1)
        Hardware type: Ethernet (1)
        DUID Time: Jan  1, 2000 00:00:00.000000000 UTC
        Link-layer address: 00:11:22:33:44:55
    Server Identifier
        Option: Server Identifier (2)
        Length: 14
        DUID: 00010001300a4cd936a73db94eeb
        DUID Type: link-layer address plus time (1)
        Hardware type: Ethernet (1)
        DUID Time: Jul 16, 2025 11:43:21.000000000 UTC
        Link-layer address: 36:a7:3d:b9:4e:eb
    Identity Association for Non-temporary Address
        Option: Identity Association for Non-temporary Address (3)
        Length: 29
        IAID: 00000001
        T1: infinity
        T2: infinity
        Status code
            Option: Status code (13)
            Length: 13
            Status Code: NotOnLink (4)
            Status Message: not on link
    Status code
        Option: Status code (13)
        Length: 24
        Status Code: NoAddrAvail (2)
        Status Message: no addresses available
    DNS recursive name server
        Option: DNS recursive name server (23)
        Length: 16
```

12/14/15/16/17

We also noticed some mandatory rules related to the client side. Theoretically, dnsmasq, as the server, should have corresponding error message checking logic, but this might be controversial or lead to unnecessary mandatory checks. Therefore, we classify them under the last item as an optional fix.

> The following has been verified: the server lacks the logic to check for violations of the corresponding messages.

```
Markdown
```

```
rfc8415
A DHCP client MUST include the INF_MAX_RT option code in any Option Request option (see Section 21.7) it sends in an Information-request message.

A DHCP client MUST request the Information Refresh Time option in the Option Request option (see Section 21.7) when sending Information-request messages.

A client MUST include an Elapsed Time option in messages to indicate how long the client has been trying to complete a DHCP message exchange.

A client MUST NOT request the Information Refresh Time option in the Option Request option in any other messages.

A client MUST include an Option Request option in a Solicit, Request, Renew, Rebind, or Information-request message to inform the server about options the client wants the server to send to the client.
```