Relay Agent Encapsulation for DHCPv4

Ted Lemon, Nominum
Hui Deng, Huawei

Requirements

- Relay options from more than one relay
- Support for Layer 2 relay agents
- Support for legacy relay agents

Options

- Relax limit on option 82
- Add new option 82-like option
- Copy RFC3315-style encapsulation

We chose encapsulation

- Cleaner model
- Doesn't conflict with option concatenation
- Less wasted space

It's complicated

- Current draft is 20 pages
- Reasons for design choices non-obvious
- So...

More than you ever wanted to know about option encapsulation

...

Basic model

- New DHCP message types: RELAYFORWARD and RELAYREPLY
- Only encapsulate options buffer, not fixed fields, and don't encapsulate magic cookie
- Each message has two parts: relay segment and encapsulation segment
- When encapsulating, relay segment and encapsulated segment become new encapsulated segment, new relay segment is added

Relay segment

- All options are relay suboptions
- dhcp-message-type relay suboption
- encapsulation information option
- gateway and relay ip address options
- other relay agent information options

dhcp-message-type suboption

- This is just the same as the dhcp-message-type option, but we need to reserve the code point in the relay space so it doesn't get allocated to something else
- Valid message types are RELAYFORWARD and RELAYREPLY
- Server doesn't know this is an encapsulated message until it sees this option

Encapsulation Information Option

- Length of relay segment
- Length of encapsulation segment
- Number of omitted Pad options
- End present flag
- RFC3119 conveniently ends signature calculation at first End option

Notice the implication

- DHCP server must look at the first option in the buffer to see if this is an encapsulation
- Otherwise concatenation might occur before server divides relay and encapsulation segments
- So dhcp-message-type has to be packed first
- Non-conforming DHCP servers had better not receive RELAYFORWARD messages

Gateway IP address option

- IP address from giaddr, when we get a packet from a legacy agent

- Required to reconstruct the packet for the legacy relay agent

- Only present in RELAYFORWARD encapsulation immediately following legacy relay agent

Relay IP address option

- Relay IP address option is added by layer three relay agents only.
- Should be what would go in giaddr in a legacy-style relay packet

Other agent options

- Any other agent options that are appropriate can also appear in the relay segment.
- How the server handles them is left somewhat open
- By default, if multiple relays send the same option, the one from the relay closest to the server is chosen (discuss?)

Encapsulation Segment

- Follows last option in Relay Segment
- Regular DHCP message: End and Pad options are eliminated, but remembered for later signature validation
- If it's a RELAYFORWARD or RELAYREPLY, or originates with the server, this isn't necessary
- All options are DHCP options
- Could contain Relay Agent Information option + suboptions

Packet direction

- A BOOTREQUEST message is a message going toward the server
- A BOOTREPLY message is a message going toward the client
- Not guaranteed that we won't receive RELAYFORWARD packets on a server-facing interface, so can't use interface to decide
- Message type doesn't work because we don't want to have to enumerate future message types

Encapsulating in the relay

- By default, agents MUST NOT encapsulate!
- If configured to encapsulate, MUST encapsulate, even if no relay options to send

Decapsulating

- When the server receives a RELAYFORWARD, it decapsulates into a nested data structure, parsing the relay segment of each encapsulation until it's down to the inner message, which it parses normally.

- It MUST then re-encapsulate on the way out, using this data structure (does the draft say this?)

Encapsulating in the server

- Using the data structure from the decapsulation, do a new encapsulation
- Options in encapsulation mostly the same
- Destination address options are different--server sends reply to first relay IP address
- Destination address options only appear in relay segments destined for layer 3 relay agents
- Not sure the draft has all this correct.

Legacy agents

- Only one is supported in any relay sequence
- First relay following legacy agent on the way to the server has to do special handling in both directions
- Again, not sure the draft gets this right

Known errata

- Option 82 in relay space not required, as stated in draft
- Actually, that's the only one I know of, but the specification needs more eyes