IPv6 DOTS Signal Option
draft-francois-dots-ipv6-signal-option-01

Jérôme François, Inria, jerome.francois@inria.fr
Abdelkader Lahmadi, Université de Lorraine,
abdelkader.lahmadi@loria.fr
Marco Davids, SIDN Labs, marco.davids@sidn.nl
Giovane Moura, SIDN Labs, giovane.moura@sidn.nl

IETF 98 Chicago
Key idea

- **Objective:** signal DDoS attacks from a DOTS client (detection) to DOTS server (mitigation)

- Regular signalling paths for delivering DOTS signals might be also affected by the DDoS → Adding an auxiliary mechanism for signaling (does not substitute)

- **Use IPv6 Hop-by-Hop Option Header [RFC2460]**
  - Embed the information into pre-existing packet
  - Signaling information is embedded into outgoing IPv6 packets
  - In an opportunistic manner (not all packets, not only those outgoing to the DOTS server... but some well chosen)
  - The DOTS client initiate this process, intermediate capable routers can store the information and embed it into other packets
Option processing (Example)

The client tries to initialize the regular signaling

The client initializes the Hop-by-hop based signaling

→ outgoing IPv6 are selected for marking

Non-capable routers ignore the option and forward the packets

The client continues the marking

When arriving at capable agents (gateways, routers), embedded information is stored

The gateway tries to initialize the regular signaling

The capable router having saved the information embeds it again in other IPv6 packets

C Client
S Server
G Gateway
R Capable Router
● Non-capable Router
⇒ Forwarded IPv6 packet
Option processing (Example)

- The client tries to initialize the regular signaling
- The client initializes the Hop-by-hop based signaling → outgoing IPv6 are selected for *marking*
Option processing (Example)

- The client tries to initialize the regular signaling
- The client initializes the Hop-by-hop based signaling → outgoing IPv6 are selected for *marking*
- Non-capable routers ignore the option and forward the packets
- The client continues the *marking*

---

**Diagram:**

- **C** Client
- **S** Server
- **G** Gateway
- **R** Capable Router
- **Ø** Non-capable Router
- ➡️ Forwarded IPv6 packet

---

Regular signaling
Option processing (Example)

- The client tries to initialize the regular signaling
- The client initializes the Hop-by-hop based signaling → outgoing IPv6 are selected for *marking*
- Non-capable routers ignore the option and forward the packets
- The client continues the *marking*
- When arriving at capable agents (gateways, routers), embedded information is stored

![Diagram showing IPv6 packet flow](image)
Option processing (Example)

- The client tries to initialize the regular signaling
- The client initializes the Hop-by-hop based signaling → outgoing IPv6 are selected for *marking*
- Non-capable routers ignore the option and forward the packets
- The client continues the *marking*
- When arriving at capable agents (gateways, routers), embedded information is stored
- The gateway tries to initialize the regular signaling
- The capable router having saved the information embeds it again in other IPv6 packets

![Diagram of network with nodes labeled C (Client), S (Server), G (Gateway), R (Router), and arrows indicating the flow of packets marked for forwarding. The diagram shows the process of option processing.]
Option processing

- Selection of packets is rule-based to only consider a subset
- A sequence of rules where each is defined by
  - 1st level: a filter on IPv6 header to be matched
  - 2nd level: a ratio of previously matched packets
  - + a timeout
- When a rule expires (timeout) the next one is applied
- Rules are manually configured
- Recommendation: firsr rules should select more packets
  (taking benefit of the first instant before loosing connectivity)

1: all outgoing IPv6 packets with a 10 second timeout
2: all outgoing IPv6 packets with a ratio of 10% and a 1 minute timeout
3: all outgoing multicast IPv6 packets with a ratio of 10% and a 1 minute timeout
4: all outgoing anycast IPv6 packets with a ratio of 10% and a 5 minute timeout
5: all outgoing IPv6 packets heading to the DOTS server with a ratio of 100% and a one hour timeout
Option encoding

- TLV-encoded in the IPv6 header

```
+-------------------------------+
| Option type | Option Data Len | DOTS Signal Attribute[1] |
+-------------------------------+
| DOTS Signal Attribute[2]     | ...              | DOTS Signal Attribute[n] |
+-------------------------------+
```

- DOTS attributes
  - from draft-reddy-dots-transport
  - + a specific TTL value to avoid embedding the information into new packets indefinitely
  - + address and port of the DOTS server to reach (+ flags)
  - a mix between TLV and fixed-length fields

```
| Attribute type  | value |
+-----------------+-------+
| policy-id       | 0     |
| target-ip       | 1     |
| target-port     | 2     |
| target-protocol | 3     |
| lifetime        | 4     |
```
Example

0 7 15 22 31

+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Next header   | Hdr Ext Len=6 | TTL=128   | Flags=IPv4,TCP|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| host=192.0.2.1 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   port=443     | A. type=policy| Att Data Len=2|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| 143            | Attr. type=ip| Att Data Len=4|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| 192.0.2.20     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Attr. type=ip  | Att Data Len=16|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| 2001:db8:6401::1 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Attr. type=port| Att Data Len=2|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| 8080           | Attr. type=port| Att Data Len=2|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| 443            | Attr. type=proto| Att Data Len=2|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| TCP            | Attr. type=lt  | Att Data Len=2|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| 600            | 1              | Opt Data Len=0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
Deployment considerations 1/2

- IPv6 extension headers are often rate-limited or dropped entirely
  - One reason is the overhead of processing
  - Our proposed option is only used under a DDoS attack and performance might be so already degraded
  - Keep limited the use to the intra-domain use case
- Modification to IP layers implementations
  - capable routers: need to extract store and embed signaling information
  - clients: need to create the specific option header to be embedded then
  - servers and gateways: all DOTS signaling information contained in IPv6 headers has to transmitted to the application layer
Deployment considerations 2/2

- Need an interface for modifying/listening IPv6 packets
  - use of Hop-by-Hop option for applications → header violation
  - advanced socket API (RFC3542)
- Header insertion issue (rfc2460bis)
  - considered as harmful
  - potential solution by encapsulating into new packets
  - keep the use limited to routers under the same authority and make transparent packet modifications → fits well the intra-domain use case
Security considerations

- Forged option headers from non legitimate sources to entail additional processing on routers
  - Source-based filtering to discard those since we know which sources can emit such IPv6 packets
  - The option can be signed by the clients and verified by the servers and gateways (intermediate capable routers do not for efficiency reason → exclude TTL from the signature calculation)
- Replay attack from a compromised router to inject more packets
  - Thanks to the id and TTL, other agents will not consider the header