DOTS
Interop test report

IETF 104 Hackathon

Kaname Nishizuka/NTT Communications

Jon Shallow/Self

Hackathon History

Hackathon	What we did	Signal Channel	Data Channel	Participants
IETF99	Implementation of OSS (go-dots)	✔		NTT
IETF100	1st Interoperability Test	✔		NTT, NCC Group, Huawei
IETF101	2nd Interoperability Test	✔		NTT, NCC Group, Huawei
IETF102	3rd Interoperability Test - The first data-channel interop	✔	✔	NTT, NCC Group, Huawei
IETF103	4th Interoperability Test - with attack and protection demo	✔	✔	NTT, NCC Group, Huawei
IETF104	5th Interoperability Test - control-filtering via signal-channel	✔	✔	NTT, NCC Group, Huawei, China Mobile

Hackathon Plan

- Drafts
  - draft-ietf-dots-signal-channel-30 (latest)
  - draft-ietf-dots-data-channel-27 (latest)
  - draft-nishizuka-dots-signal-control-filtering-05 <- NEW!!

- Interoperability test and PoC of signal-control-filtering

Test target

2 independent implementations:

- go-dots (https://github.com/nttdots/go-dots)
  - Tested as a client/server
  - Insertion of ACLs on routers
  - BGP route injection(Traffic redirection and RTBH)
  - BGP flowspec
- DDoS Secure (NCC Group)
  - Tested as a client/server/gateway
  - Inline protection as a DDoS Mitigation System (DMS)

Tested functions in both peace-time and attack-time:

- signal channel:
  - session configuration, mitigation request, CoAP ping, observe
  - signal-control-filtering<-NEW
- data-channel:
  - registration of client/alias/filtering rules

Interop test result(1/2)

- We are on the last corner for the first publication of DOTS spec! (~90% test coverage)

✅ Done beforeIETF103
✅ Done in internal intreop testing before IETF104
IETF104 Hakcathon

				# Interoperability Testing Results			
# DOTS Signal Channel Features implementation status				client: godots	client: ncc		
#	feature	ncc*	godots*	server: ncc	server: godots	issue to WG	memo
1	Session Configuration	✅	✅	✅	✅		
2	Mitigation Request	✅	✅	✅	✅		
3	CoAP Ping	✅	✅ *	✅	✅		* Added DTLS session resiliency after heartbeat loss
4-1	Observe (Mitigation)	✅	✅	✅	✅		
4-2	Observe (Session Config)	✅	✅	✅	✅		
5	Efficacy Update	✅	✅	✅	✅		
6	Request Confliction Handling	✅	✅	✅	✅		
7	Confliction Notify	✅	✅	✅	✅		
8	Deadman's Trigger	✅	✅				
9	Block-wise Transfer	✅	✅	✅	✅		
10	Gateway Function	✅					
11	Redirection	✅					
12	Happy Eyeballs	✅					
	* supporting both PKI and PSK						

block-wise transfer [RFC7959] worked. (there was no significant issue

				client: godots	client: ncc		
# DOTS Data Channel Features implementation status							
#	feature	ncc	godots	server: ncc	server: godots	issue to WG	memo
1	Register DOTS clients	✅	✅	✅	✅		
2	Register Alias	✅	✅	✅	✅ *(bug)		* handling of implict protocol number
3	Register Filtering Rules	✅	✅	✅	✅		
4	Capabilities	✅	✅	✅	✅		
5	Gateway Function	✅					

Interop test result(2/2)

- signal-control-filtering was tested between 2 independent implementations
  - Codes are updated during(and after) the hackathon intensively
  - questions found will be raised to WG

# DOTS Control Filtering Implementation Status					client: godots	client: ncc		
#	feature	ncc	godots		server: ncc	server: godots	issue to WG	memo
1	Control Filtering	✅	✅		✅	✅ *	questions raised	Fixed at IETF104 hackathon oppotunity *coding of activation-type(enum)

Result Sheet:
https://docs.google.com/spreadsheets/d/1CoRZa6havsL2r_EO0ax1ukJnAbxDFb7AnF97yNEnfXE/edit?usp=sharing

PoC of signal-control-filtering

- Successful run-through of PoC demo (2 use cases in the latest draft)
- Conflict Handling (3.2.2.1)
  - accept ACL: activate-when-mitigating->deactivate
- Activate Rate-Limit or Drop Filters (3.2.2.3)
  - rate-limit or drop ACL: deactivate->immediate

Hackathon Wrap Up

Team members:
Kaname Nishizuka (NTTCom)
Yasuaki Morita (Lepidum)
Jon Shallow (Self)
Liang 'Frank' Xia (Huawei)

First timers @ IETF/Hackathon:
4 new members!! contributed to OSS project

https://github.com/nttdots/go-dots

A brief summary of a signal-control-filtering usecase

- Activate Rate-Limit or Drop Filters (3.2.2.3)
  1. Install an accept-list that rate-limits all (or a part thereof) traffic during 'idle' time (via Data-Channel) with "activation-type": "deactivate"
  2. a DDoS attack is detected by the DOTS client
  3. the DOTS client sends a mitigation request to its DOTS server
  4. For some reason (e.g., the DOTS server, or the mitigator, is lacking a capability or capacity), the DOTS client is still receiving the attack
  5. Then the DOTS client can change activation-type of the filter to "immediate" by signal-control-filtering which will ensure that you can proceed to next actions…

#1: Questions about signal-control-filtering

```
Header: PUT (Code=0.03)
Uri-Path: ".well-known"
Uri-Path: "dots"
Uri-Path: "mitigate"
Uri-Path: "cuid=0opPisZqo4SLv64TLPXrxA"
Uri-Path: "mid=85"
Content-Format: "application/dots+cbor"
{
  "ietf-dots-signal-channel:mitigation-scope": {
    "scope": [
      {
        "target-prefix": [
          "2001:db8:123::/48"
        ],
        "acl-list": [
          {
            "acl-name": "my-ratelimit-list",
            "activation-type": "activate"
          }
        ]
        "lifetime": 3600
      }
    ]
  }
}
```

mitigation request

turn on/off ACL

Figure 9: DOTS Signal Channel Mitigation Request to Activate a Rate-Limit Filter

#1: Thoughts about signal-control-filtering

- Thoughts
  1. A request including acl-list will 1) trigger a mitigation request (with target-*) and 2) turn on/off ACL
  2. GET response of that mid will be treated as if it is a mitigation request only (already noted in the draft) (no inclusion of acl-list)
  3. The change to ACL is declarative. DELETE of the mid or expiration of lifetime will not affect the final state of the ACL

#1: Questions about signal-control-filtering

- Questions
  1. Should a mitigation request create a mitigation before doing a PUT + acl-list [{acl-name, activation-type}] against the active mitigation, or is a 'PUT + acl-list [{acl-name, activation-type}]' allowed to create a new mitigation?
     - [note]If signal-control-filtering always accompany with a mitigation request, it can be always treated as it's in attack-time.
  2. Should the response to a GET (or Observed GET) include the acl-list [{acl-name, activation-type}] if the PUT included it?
  3. Does the response to the PUT (the echoed back response payload of the PUT representation https://tools.ietf.org/html/rfc7252#section-5.9.1.1) include the acl-list (if defined) or not?
  4. Is the activation change to the ACL a permanent change to the ACL (so a GET on the data channel returns the new type)?
  5. Does the activation change to the ACL count as an ACL refresh (pending-lifetime update)?
  6. Is CBOR activation –type comprehension-required or comprehension-optional?

#2: Data Channel Implicit protocol number

- ACL with tcp port number leaf
  - "tcp":{"destination-port":{"lower-port":443}}
- Should it be treated as "protocol":6 even if no protocol number was specified in the ipv4 or ipv6 section?
- https://tools.ietf.org/html/rfc8519#section-4.4 examples do not include ipv4/ipv6 protocol definitions for tcp/udp examples

#3: (D)TLS session lifetime

- 4.7 Heartbeat Mechanism

  After the maximum 'missing-hb-allowed' threshold is reached, the DOTS client SHOULD try to resume the (D)TLS session. The DOTS client SHOULD send mitigation requests over the current DOTS signal channel session, and in parallel, for example, try to resume the (D)TLS session or use 0-RTT mode in DTLS 1.3 to piggyback the mitigation request in the ClientHello message.

  – From the view point of DOTS server, when to expire the old (D)TLS session is implementation specific
  – Do you have any ideas?

Takeaways

- Most of issues about core specification is addressed so far.
- signal-control-filtering spec gives a useful switch for turning on/off of ACL via signal-channel.
  - It’s proven to work!
- We’re attracting new players in this field.

Questions

Or

Comments?

Thank You