Controlling Filtering Rules Using DOTS Signal Channel

draft-nishizuka-dots-signal-control-filtering-05

IETF#104 Prague, March 2019

K. Nishizuka, M. Boucadair, T. Reddy,
T. Nagata

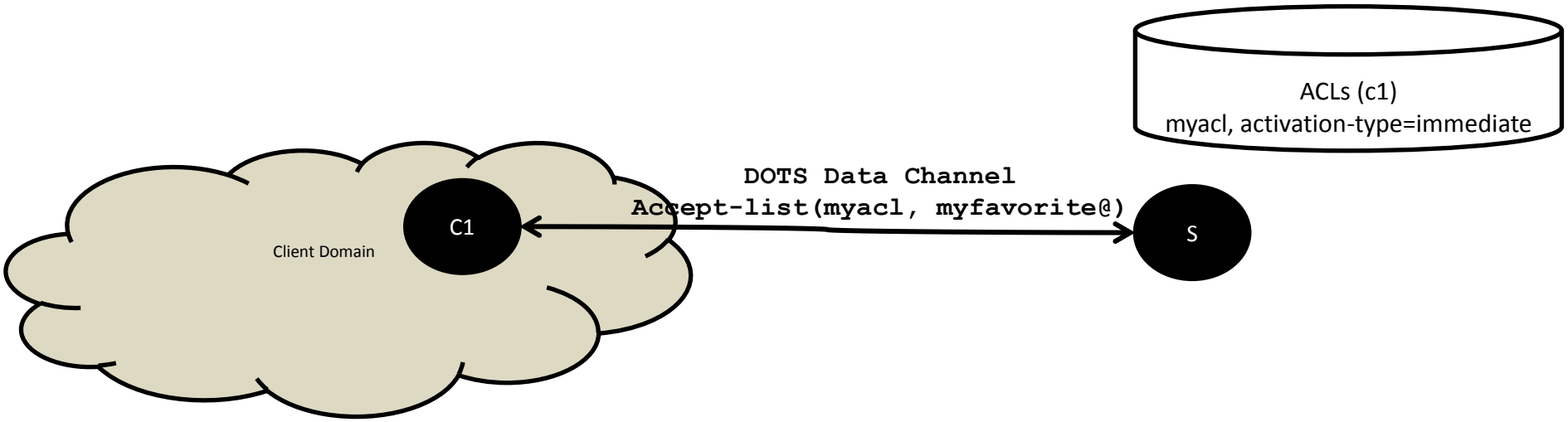
A Reminder

```
module: ietf-dots-data-channel
      +--rw dots-data
         +--rw dots-client* [cuid]
         |  +--rw cuid              string
         |  +--rw cdid?             string
         |  +--rw aliases
         |  |  ...
         |  +--rw acls
         |     +--rw acl* [name]
         |        +--rw name                  string
         |        +--rw type?                 ietf-acl:acl-type
         |        +--rw activation-type?      activation-type
```

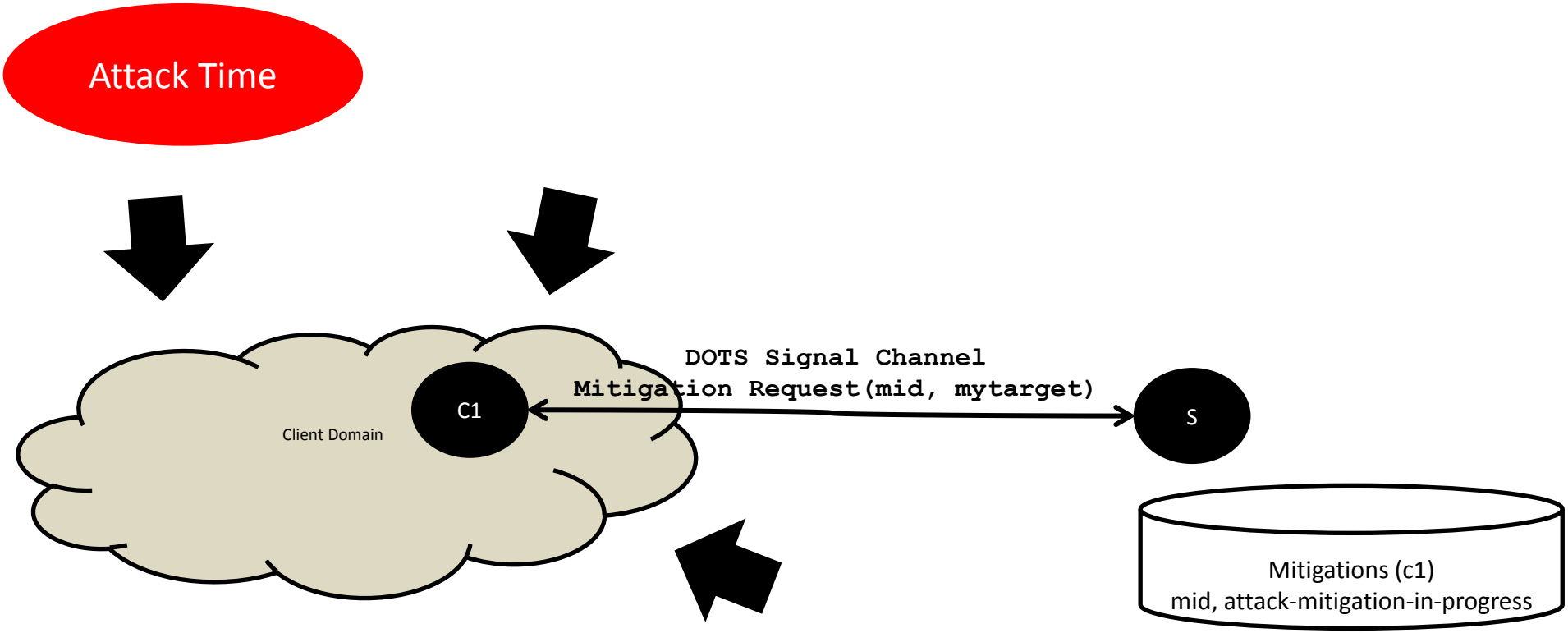
1. activate-when-mitigating
2. Immediate
3. deactivate

The Initial Problem

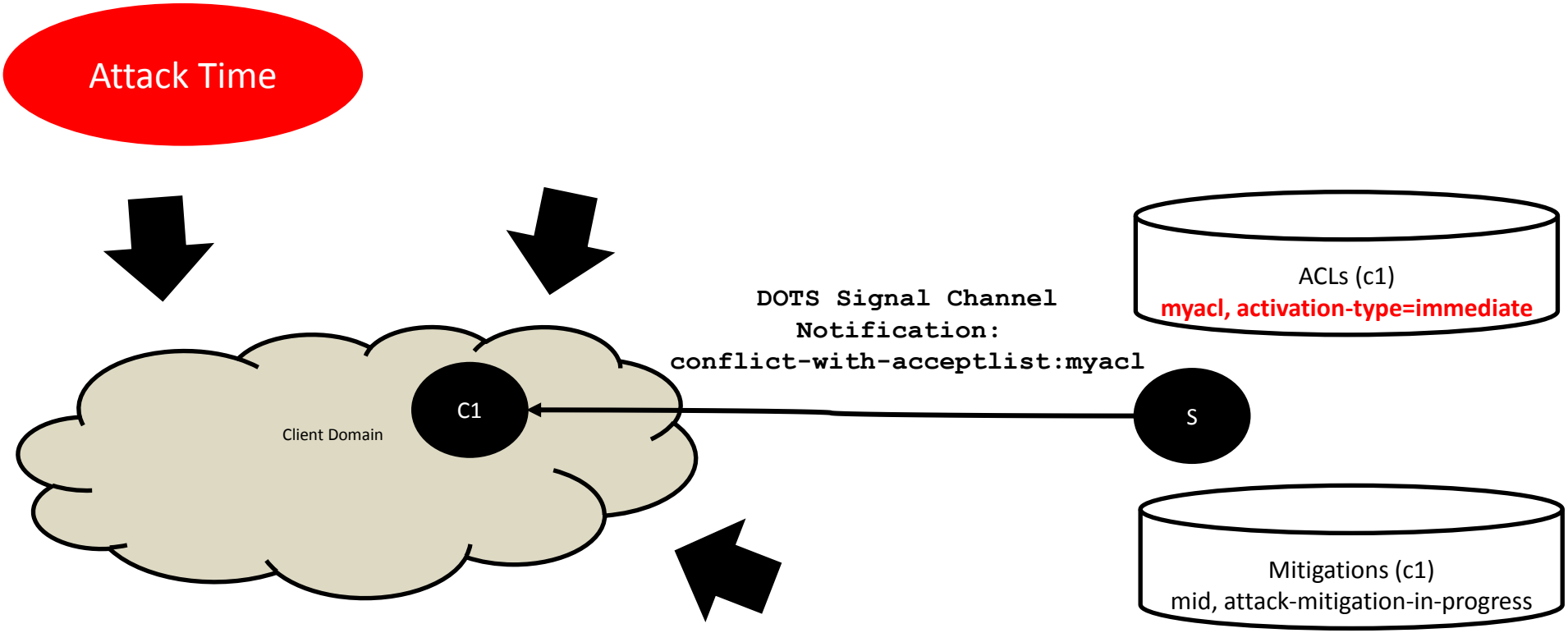
Idle Time



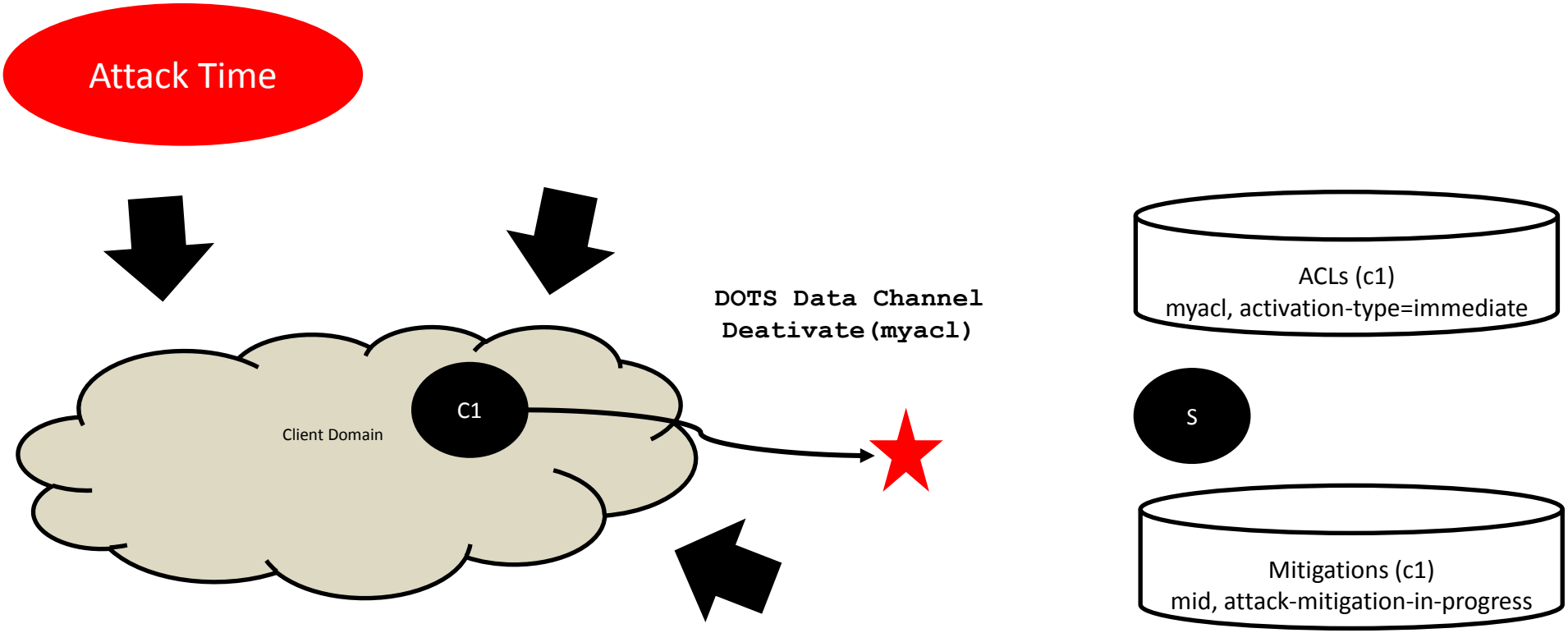
The Initial Problem



The Initial Problem



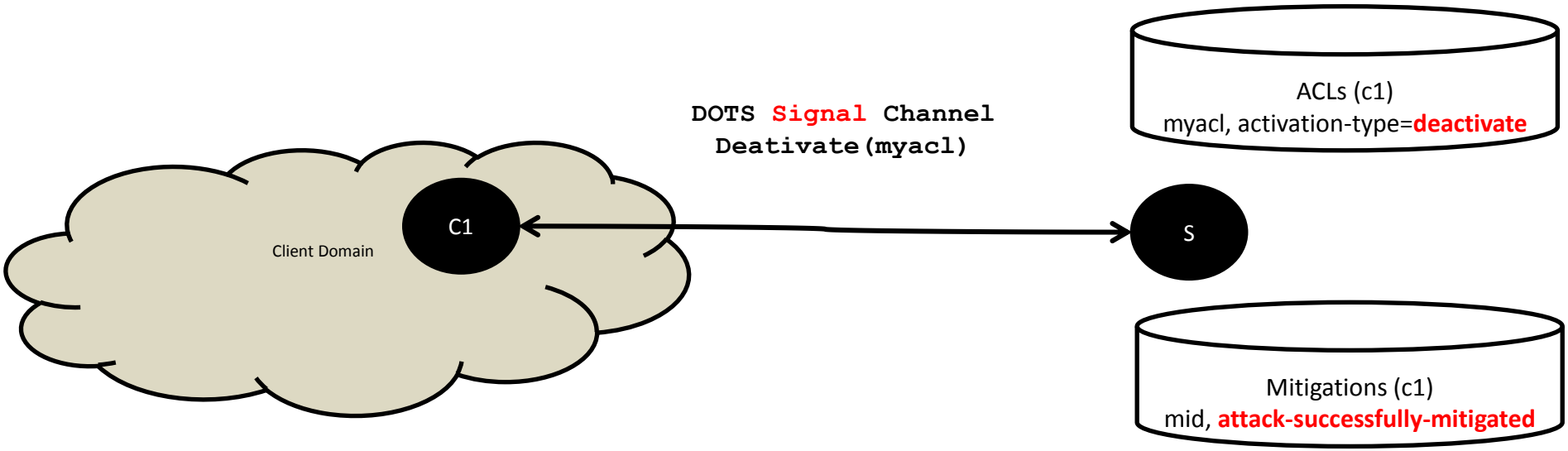
The Initial Problem



- The use of the data channel during attack time is not an option
- The signal channel does not allow to control ACLs

The Solution

Attack Time



Allow for the Signal Channel to get control on filters:

```
augment /ietf-signal:dots-signal/ietf-signal:message-type
        /ietf-signal:mitigation-scope/ietf-signal:scope:
  +--rw acl-list* [acl-name] {control-filtering}?
     +--rw acl-name
     |    -> /ietf-data:dots-data/dots-client/acls/acl/name
     +--rw activation-type?   activation-type
```

Already defined as comprehension-required parameters in [I-D.ietf-dots-signal-channel]

New parameter

Other Use Cases

- Activate (preconfigured) ACLs during mitigation, e.g.,
  - Enforce a rate-limit/drop-filter if the Mitigator is lacking capacity or capability

Some Recommendations

- It is RECOMMENDED for a DOTS client to subscribe to asynchronous notifications of the attack mitigation
- A DOTS client MUST NOT use the filtering control over DOTS signal channel if no attack (mitigation) is active
- ACL-related clauses are not included in a PUT request used to send an efficacy update, GET responses, and DELETE requests

What's Next?

- The bug was initially detected during IETF#103 interop
  - The proposed solution is tested during IETF#104
- All received comments were addressed (many thanks to the reviewers)
- Request WG Adoption