CLI-DOS: Collaborative Counteraction against Denial of Service in the Internet of Things

Status, Main Results and Plan

Syafiq Al Atiiq[1] Christian Gehrmann[1]

[1]Dept. of Electrical and Information Technology
LTH Lund University

Sec4Factory Workshop, October 2019

Table of Contents

1. Background
2. Current Approach
3. Proposal
4. Status

Table of Contents

1. Background
2. Current Approach
3. Proposal
4. Status

Background

- IoT devices are especially exposed to the Denial of Service attack.

Background

- IoT devices are especially exposed to the Denial of Service attack.
- It is even harder to defend DoS in IoT environment, due to:
  - Limited resource (CPU and memory).
  - Transmission capabilities.
  - Limited battery.

Background

- IoT devices are especially exposed to the Denial of Service attack.
- It is even harder to defend DoS in IoT environment, due to:
  - Limited resource (CPU and memory).
  - Transmission capabilities.
  - Limited battery.
- Denial of Service (DoS) in IoT
  - Flooding a server hosts with messages.
  - Exhaust server resources (e.g. bandwidth, processing, energy)
  - Less reactive or even unable to serve legitimate requests

Background

- IoT devices are especially exposed to the Denial of Service attack.
- It is even harder to defend DoS in IoT environment, due to:
  - Limited resource (CPU and memory).
  - Transmission capabilities.
  - Limited battery.
- Denial of Service (DoS) in IoT
  - Flooding a server hosts with messages.
  - Exhaust server resources (e.g. bandwidth, processing, energy)
  - Less reactive or even unable to serve legitimate requests
- Countermeasure categories: router-based and host-based.

Table of Contents

1 Background

2 Current Approach

3 Proposal

4 Status

Current Approach

Existing mechanisms only either:

- Make it more difficult to perform massive scale DoS attacks, or...

Current Approach

Existing mechanisms only either:

- Make it more difficult to perform massive scale DoS attacks, or...
- Completely shut down the connection from outside world to save energy

Table of Contents

1. Background
2. Current Approach
3. Proposal
4. Status

CLI-DOS

A Collaborative Counteraction against Denial of Service.

CLI-DOS

A Collaborative Counteraction against Denial of Service.
The rationale:

- Reducing the impact of DoS attacks against a victim device, and
- Offloading a computational expensive filtering at the IoT unit to a much more powerful gateway, while at the same time...

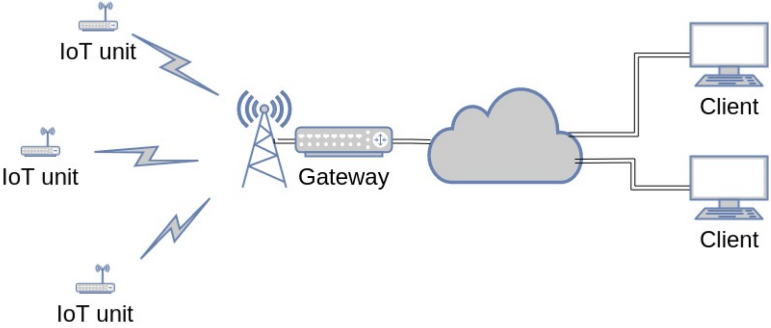
CLI-DOS

A Collaborative Counteraction against Denial of Service.

The rationale:

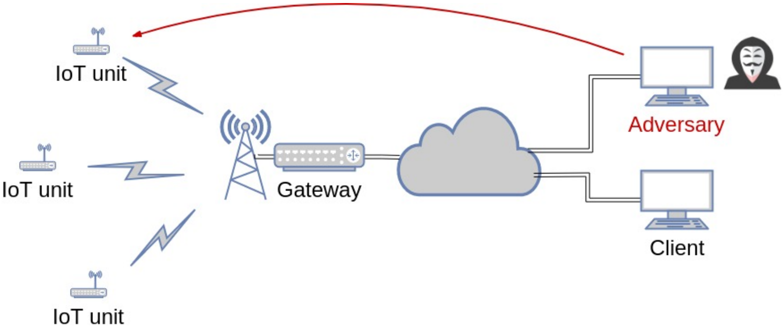
- Reducing the impact of DoS attacks against a victim device, and
- Offloading a computational expensive filtering at the IoT unit to a much more powerful gateway, while at the same time...
- Allowing legitimate request to be served to the best possible extent

CLI-DOS(Scenario)



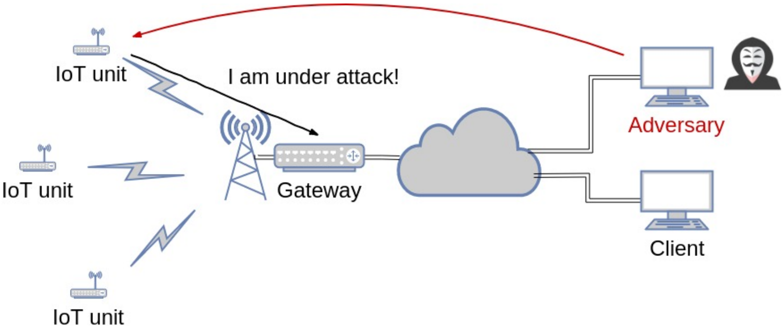
A resource-constrained, wireless IoT-units are connected and able to serve requests.

CLI-DOS



An adversary repeatedly send bogus messages to the IoT units, trying to induce the device to worthlessly commit resources.

CLI-DOS

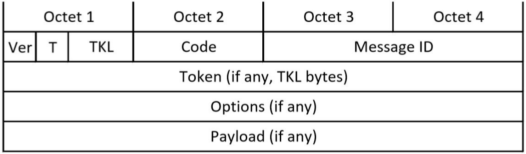


The IoT unit send a request to the gateway to start a security filtering mechanism, i.e. block ranges of IP, block all traffic but CoAP, etc. The request also contains time S as a sleeping period (victim server shutting down the radio communication).

CLI-DOS

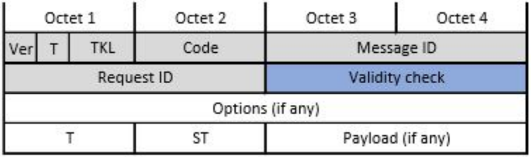
To distinguish the valid and invalid messages, the victim uses a short MAC (Message Authentication Code)[1]embedded in token field of the CoAP.

Default CoAP packet format:



Octet 1			Octet 2	Octet 3	Octet 4
Ver	T	TKL	Code	Message ID	
Token (if any, TKL bytes)					
Options (if any)					
Payload (if any)					

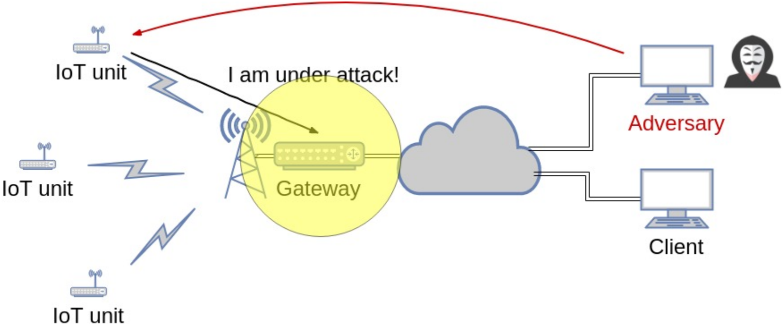
CoAP with short MAC:



Octet 1			Octet 2	Octet 3	Octet 4
Ver	T	TKL	Code	Message ID	
Request ID				Validity check	
Options (if any)					
T			ST	Payload (if any)	

[1] C. Gehrmann, M. Tiloca and R. Höglund, "SMACK: Short message authentication check against battery exhaustion in the Internet of Things"

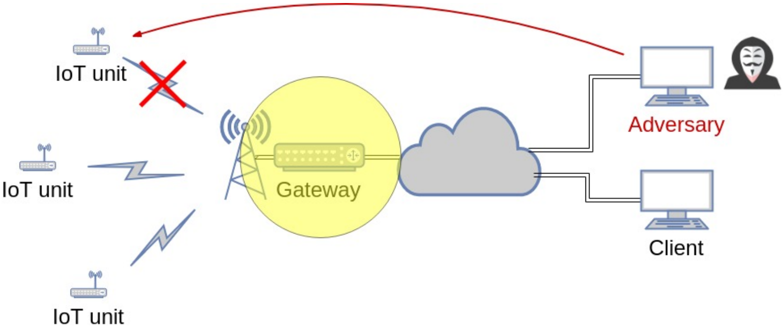
CLI-DOS



The gateway now:

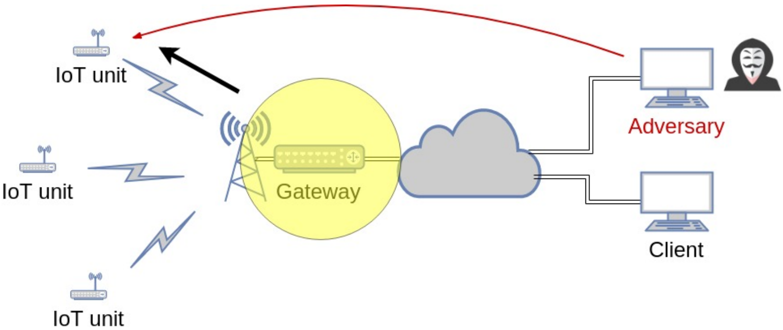
- Proceses the received information to calculate a new filtering rule.
- Measure the number of valid/invalid packets targetting the victim.
- Store the valid packet in internal buffer.

CLI-DOS



The IoT unit goes into sleep mode and only comes into active mode after agreed sleep period S.

CLI-DOS



After sleep period S is over, if invalid messages are below threshold, the gateway forwards all the packets from the buffer. Otherwise, the gateway updates its filtering rule to a more restrictive setting without forwarding any packets to the IoT unit. If it still continue, it sends a general warning to the system responsible.

Table of Contents

1. Background

2. Current Approach

3. Proposal

4. Status

Status

- Implemented as an additional module for Contiki-NG.

Status

- Implemented as an additional module for Contiki-NG.
- Hardware for testing: Zolertia Firefly rev.A.

Status

- Implemented as an additional module for Contiki-NG.
- Hardware for testing: Zolertia Firefly rev.A.
- Experimental evaluation on a local testbed.

Experimental Evaluation (On progress)

Measured values:

- Round Trip Time.
- Energy usage.

Variation of:

- Attack intensity
- Number of clients
- Gateway policies

The End

Thank you! Questions?