IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

SECURITY ISSUES IN THE INTERNET OF PERSONS, THINGS AND SERVICES

Marco LOBE KOME

Joint work with

Nora Cuppens-Boulahia
Frédéric Cuppens
Vincent Frey

SUMMARY

Contributions

3 contributions:

- DIscovery and REgistration Protocol (DIRE)
- Constrained Application Protocol 2 (CoAP2.0)
- Detection and Response to Data Exfiltration Attack

DIRE

DIscovery and REgistration Protocol, For Device and Person Identity Management in IoT

International Conference on Information Systems Security (ICISS), Bombai, Inde. December 2017

DIRE

Problem statement

- The number of Internet-connected devices exceeded the human population in 2010 (Cisco study *[ciscostudy]*)
- The limited resources available on constrained devices makes it difficult to apply high strength encryption and signature algorithms
- Users are sharing human identity credentials with their unsecured devices, which makes those credentials exposed.
- The behavior of connected devices depends highly on manufacturers *[fitbit]*

[*ciscostudy*] D.Evans, “”the internet of things”:how the next evolution of the internet is chang-ing everything,” Whitepaper, Cisco Internet Business Solutions Group (IBSG), 2011.
[fitbit] thenextweb, “Fitbit users are unwittingly sharing details of their sex lives with the world.,” 2013.

- OAuth 2.0 Internet of Things (IoT) Client Credentials Grant [1]
  Do not consider the multi user-agent factor
- User Managed Access (UMA) [2], Federated Identity and Access Management (FIAM) for IoT [3]
  The notion of device identity is missing
- IoT OAuth based Authorization Service architecture (IoT-OAS) [4]
  This solution stresses more on access control management.

[1] H. Tschofenig, “The OAuth 2.0 Internet of Things (IoT) Client Credentials Grant.”
[2] “User Managed Access - Kantara Initiative.”
[3] P. Fremantle, B. Aziz, J. Kopecky, and P. Scott, “Federated identity and access management for the internet of things,” in Secure Internet of Things (SIoT), 2014 International Workshop on, pp. 10–17, IEEE, 2014.
[4] S. Cirani, M. Picone, P. Gonizzi, L. Veltri, and G. Ferrari, “Iot-oas: an oauth-based authorization service architecture for secure services in iot scenarios,” IEEE Sensors Journal, vol. 15, no. 2, pp. 1224–1234, 2015.

DIRE

Thing Description Document (TDD)

```
{
"properties": {
  "thing_id":"5E:FF:56:A2:AF:15",
  "name":"Connected flower pot",
  "description":"This is the description of the connected pot",
  "last-modified":"2016-07-20",
  "capabilities":[
    "temperature",
    "moisture",
    "luminosity"
  ]
},
"services":[
  "api":"connected_flower.raml",
  "intents":[
    "send-mail",
    "social-network-broadcast",
  ],
  "scopes":[
    "profile",
    "contact"
  ]
]
}
```

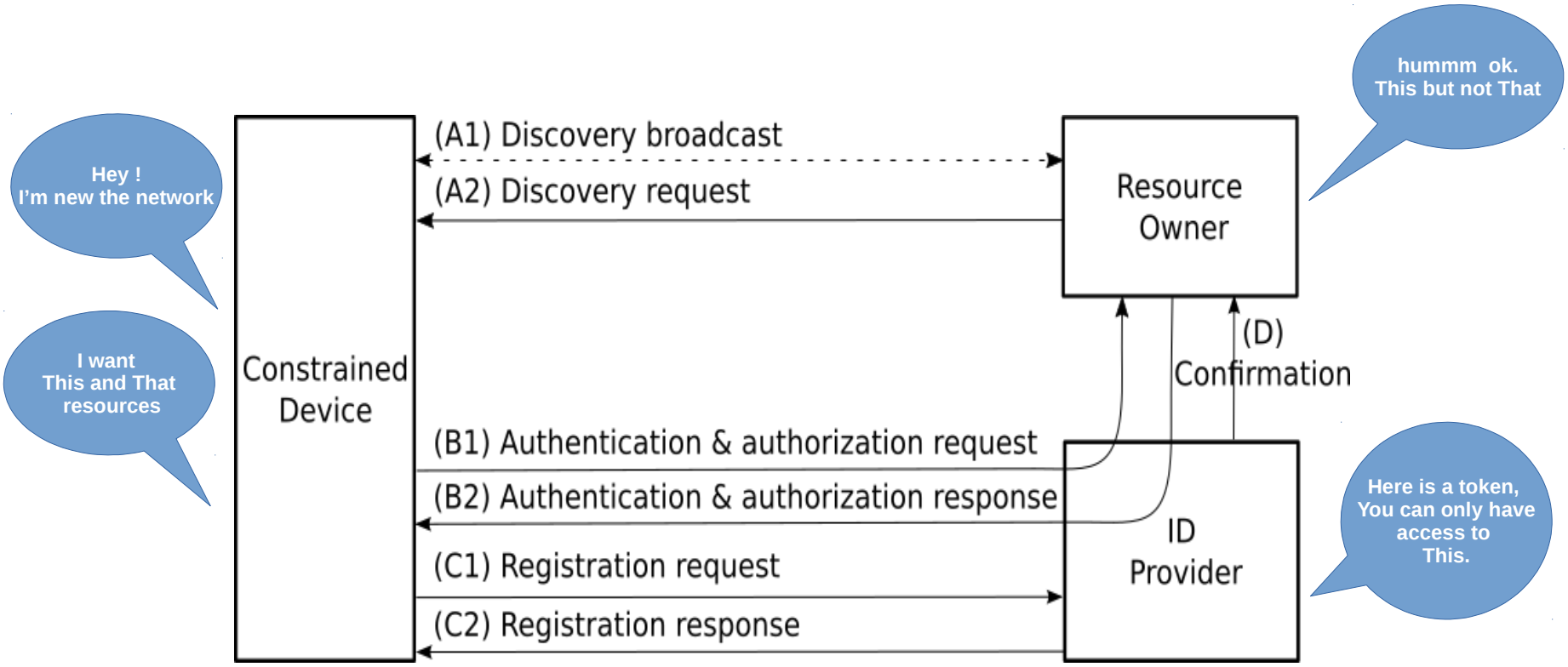
Properties and *Services* objects can be filled with additional information for a richer discovery and registration experience.

Inspired by the Simurgh framework **[simurgh]**

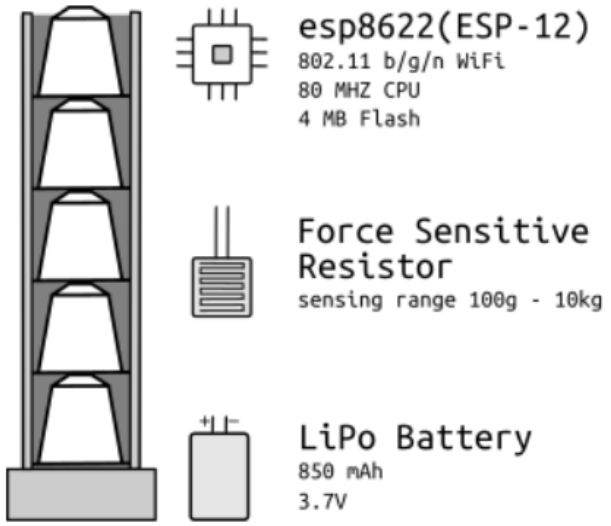
[simurgh] F. Khodadadi, A. V. Dastjerdi, and R. Buyya, “Simurgh: A framework for effective discovery, programming, and integration of services exposed in IoT,” in Recent Advances in Internet of Things (RIoT), 2015 International Conference on, pp. 1–6, IEEE, 2015.

DIRE

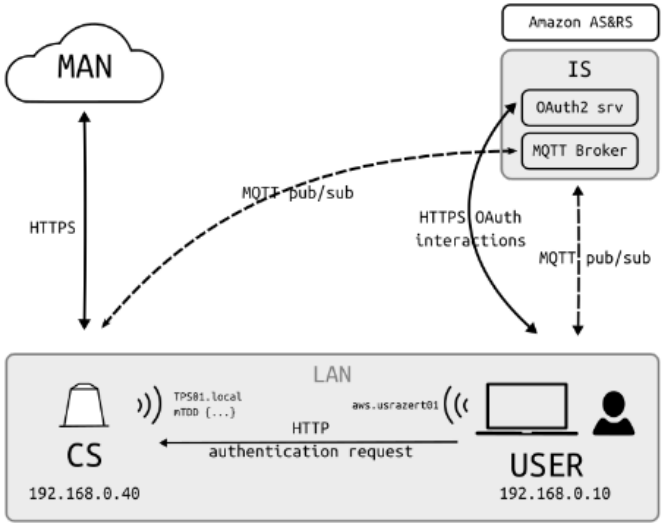
Overall architecture



- Auto refill example



(a) Coffee Supplier (CS)

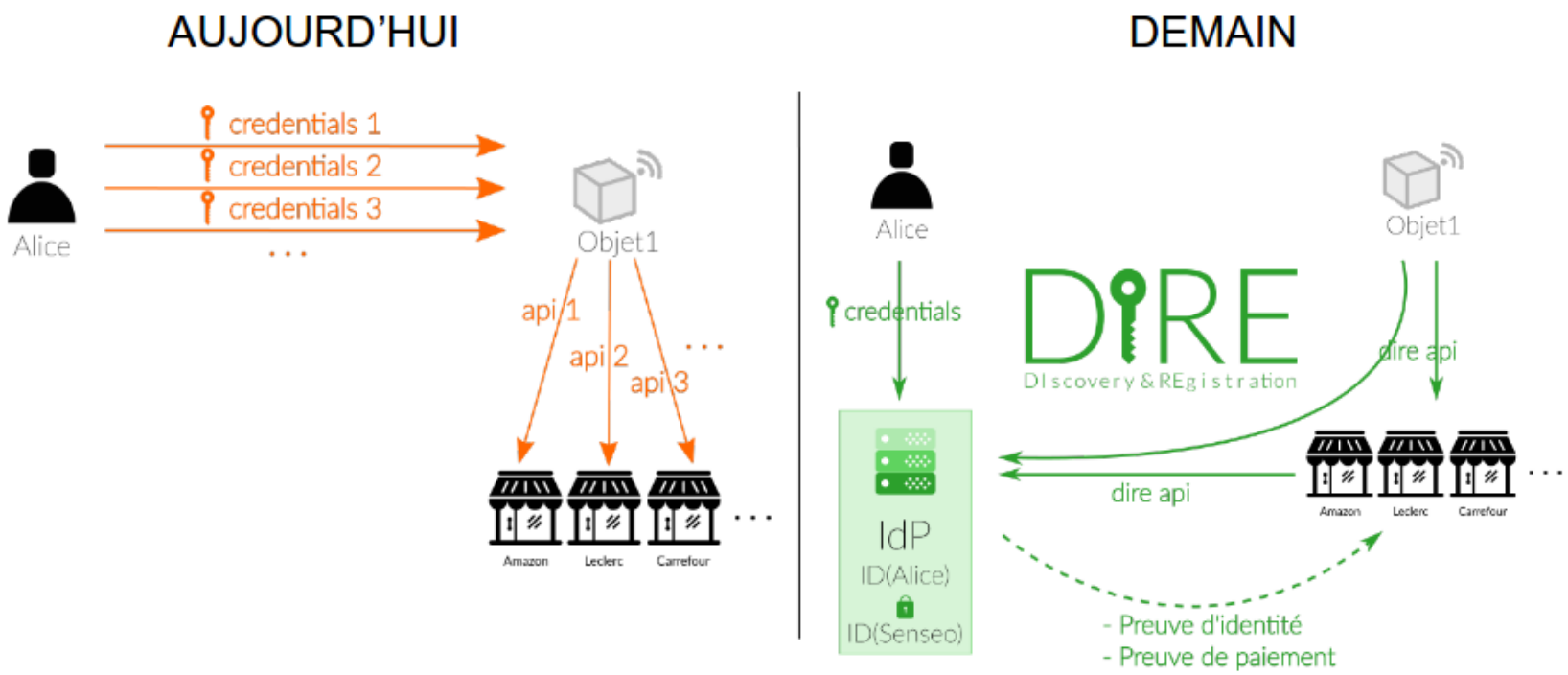


(b) Implementation architecture

DIRE

Implementation

- Use case with Orange as the IdP



- Non-repudiation of the device: The user and the device are clearly differentiated
- The system is deterministic, each error message is notified to the owner
- Discovering and registring a device take less than 10s
- 45 % of memory used by the firmware.
- The respect of integrity, anonymity and confidentiality properties.
- The protocol can be better implemented using CoAP

CoAP2.0

CoAP Enhancement For a Better IoT Centric Protocol

International Conference on Internet of Things: Systems, Management and Security (IoTSMS), Valencia, Spain. October 2018

COAP2.0

- Problem statement

3 main behaviors to fulfill in an IoT centric protocol :

- **Advertisement**
- **Notification**
- **Synchronous** and **asynchronous communications**

COAP2.0

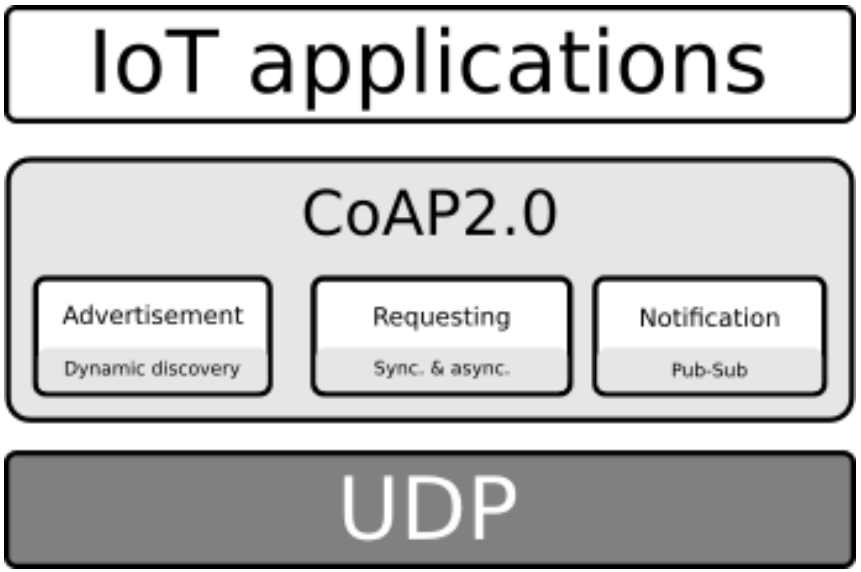
- Existing solutions

Protocol	Advertising	Sync & Async	Notification
mDNS	++	--	--
MQTT	--	+-	++
CoAP	+-	++	+-

++ Built for that
+- can be done but not efficient
- - inexistant functionality

COAP2.0

- Our solution : general overview



COAP2.0

- Our solution : Advertisement

Field	Description	Length/Bits
QNAME	The hostname to resolve or just ".local"	variable
QTYPE	Set to 'COAP'	16
U-R	Set to '1'	1
QCLASS	Set to 'IN' for Internet	15
QAUTHZ	The authorization token	Variable
FILTER	list of paths to resolve	Variable

Whatever the value of U-R, if QAUTHZ is not null then the response will be unicast for obvious security purposes.

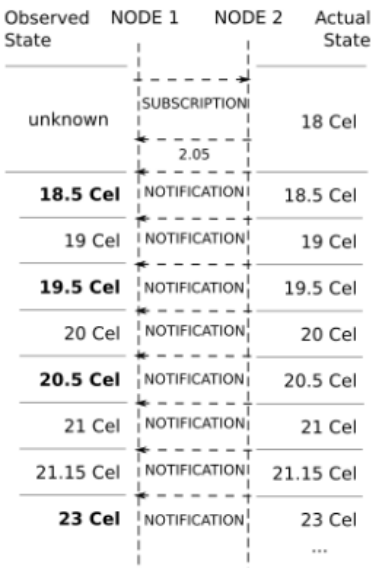
a) Coap2 discovery request

```
1 Type:     COAP
2 Hostname: coffee-machineAZ012.local
3 TTL:      5 minutes
4 Addr:     192.168.0.3
5 Resource: /sensors/temp
6 Resource: /sensors/light
7 Resource: <etc>
8 Text:     additionnal information 1
9 Text:     <etc>
```

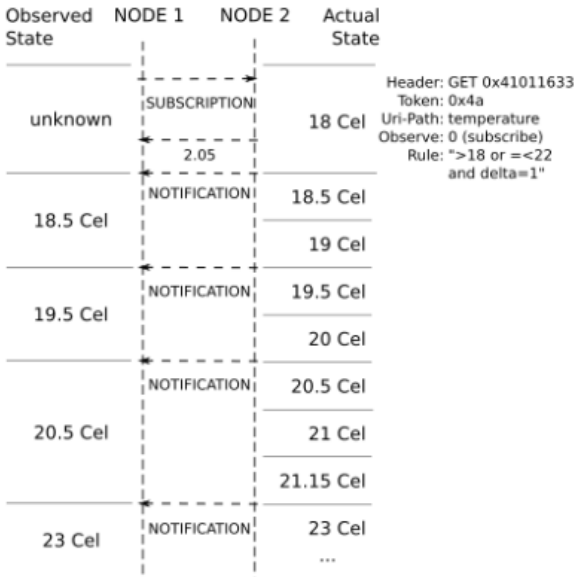
b) Coap2 discovery response

COAP2.0

- Our solution : Notification



a) CoAP notification



b) CoAP2.0 notification

COAP2.0

Results and conclusions

- Discovery requieres only 0.4 % of available resources and is constent whatever the number of node on the network
- Smart discovery and notification whith a reduced impact on the memory.
- Compliance with traditional security mecanism (TLS over UDP, JWT, ...)
- Develop filter with SQL syntax as a future work

Detection and Response to Data Exfiltration

from Internet of Things Android Devices

World Computer Congress (WCC), Poznan, Pologne. September 2018

RESPONSE TO DATA EXFILTRATION

- The context

- The user privacy threat is growing along with the number of IoT devices. 30 billion connected objects are expected by 2020 [ieee].
- Hackers aim to exfiltrate personal data stored in the IoT devices such as smartphones through USB port.
- Quang et al. demonstrates how to use its adversary model to covertly exfiltrate data from Android devices. [quang]
- Christian et al. investigate how an attacker could abuse of a command line tool distributed with iTunes to exfiltrate data from a paired iOS device [christian]

[*ieee*] Amy Nordrum, “Popular Internet of Things Forecast of 50 Billion Devices by 2020 Is Outdated”, Whitepaper, https://spectrum.ieee.org/, 2016.
[quang] Quang et al., “Exfiltrating data from Android devices”, Computers & Security 48, 74-91, 2015.
[christian] Christian et al, “Data exfiltration from Internet of Things devices: iOS devices as case studies”, IEEE Internet of Things Journal 48, 524-535, 2017.

RESPONSE TO DATA EXFILTRATION

- Existing solutions

Some existing security tools in Android systems focus on detection of the sensitive data leakage :

- TaintDroid **[1]** enables real-time analysis of Android applications behaviours seeking for misbehaving ones.
- ScanDroid **[2]** checks whether data are flowing according to the permissions granted by the user and stored on the Android Manifest.
- Combine dynamic and static Taint to trace back data flow and detect the sensitive information leakage **[3]**.

[1] Enck et al, “TaintDroid: an information-flow tracking system for realtime privacy monitoring on smartphones”
[2] Fuchs et al, “Scandroid: Automated security certification of android”
[3] Graa et al, “Tracking explicit and control flows in Java and native Android apps code”, in ICISSP 2016, 2016, pp. 307–316

RESPONSE TO DATA EXFILTRATION

- Security goals

Let us consider ***m*** as the application data message, sent over channel ***c*** between an honest client ***C*** and an honest server ***S***

- **Secrecy** The message ***m*** is kept confidential from the attacker ***E***
- **Integrity** The message ***m*** can be seen but cannot be modified by ***E***
- **Authentication** via :
  - **Injective agreement** This property holds if each event from run n is different from events from run n + 1.
  - **Integrity of the message *m*** The authentication property is satisfied if the injective agreement holds and if the message ”m” has not been modified.

RESPONSE TO DATA EXFILTRATION

- Attack Model

RESPONSE TO DATA EXFILTRATION

- The protocol

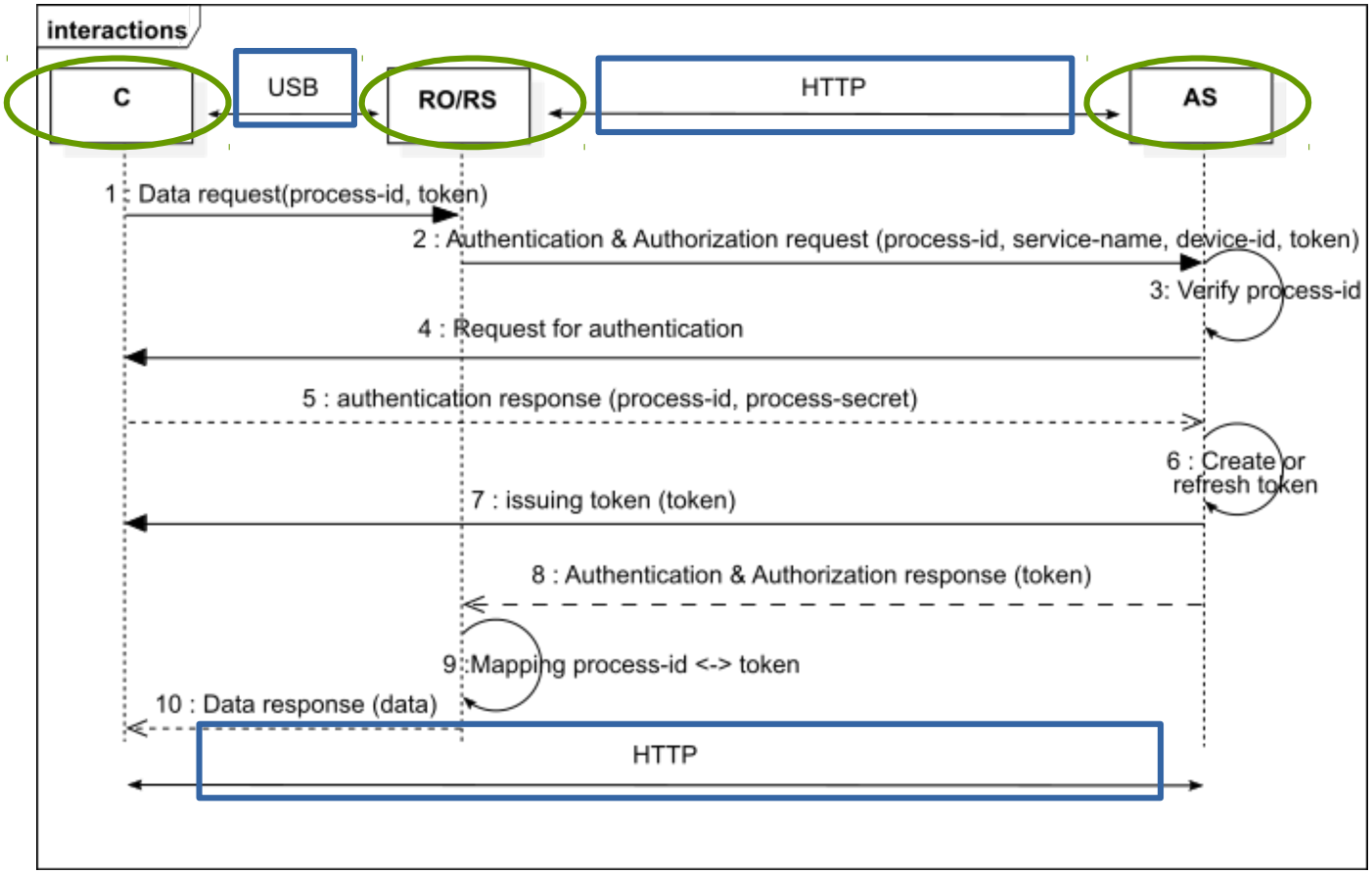


Figure 1: Overall architecture.

RESPONSE TO DATA EXFILTRATION

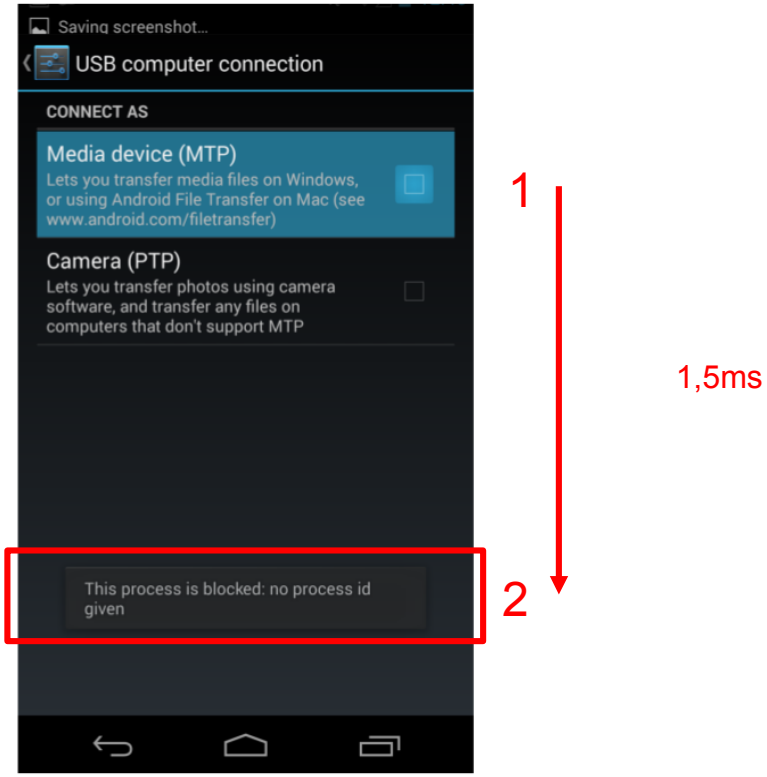
- Authentication Server

Cases	id	token	App. state	AS responses
1	unregistered	All cases	All cases	Error : unregistered
2	registered	null	authenticated	New token delivered
3	registered	null	Not authenticated	Data exfiltration attack
4	registered	Is modified	All cases	Data exfiltration attack
5	registered	Not modified & valid	All cases	Access allowed
6	registered	Not modified & not valid	All cases	Proceed to authentication

RESPONSE TO DATA EXFILTRATION

- Results : Implementation on Android 4

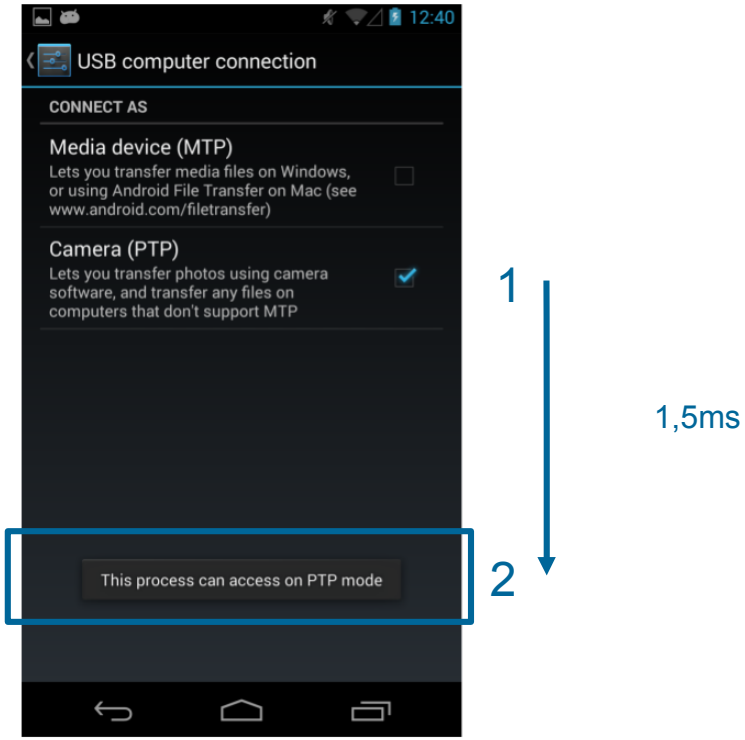
Cases	id	token	App. state	AS responses
1	unregistered	All cases	All cases	Error : unregistered



RESPONSE TO DATA EXFILTRATION

- Results : Implementation on Android 4

Cases	id	token	App. state	AS responses
2	registered	null	authenticated	New token delivered
5	registered	Not modified & valid	All cases	Access allowed



RESPONSE TO DATA EXFILTRATION

- Results : Detect outdated tokens

Cases	id	token	App. state	AS responses
6	registered	Not modified & not valid	All cases	Proceed to authentication

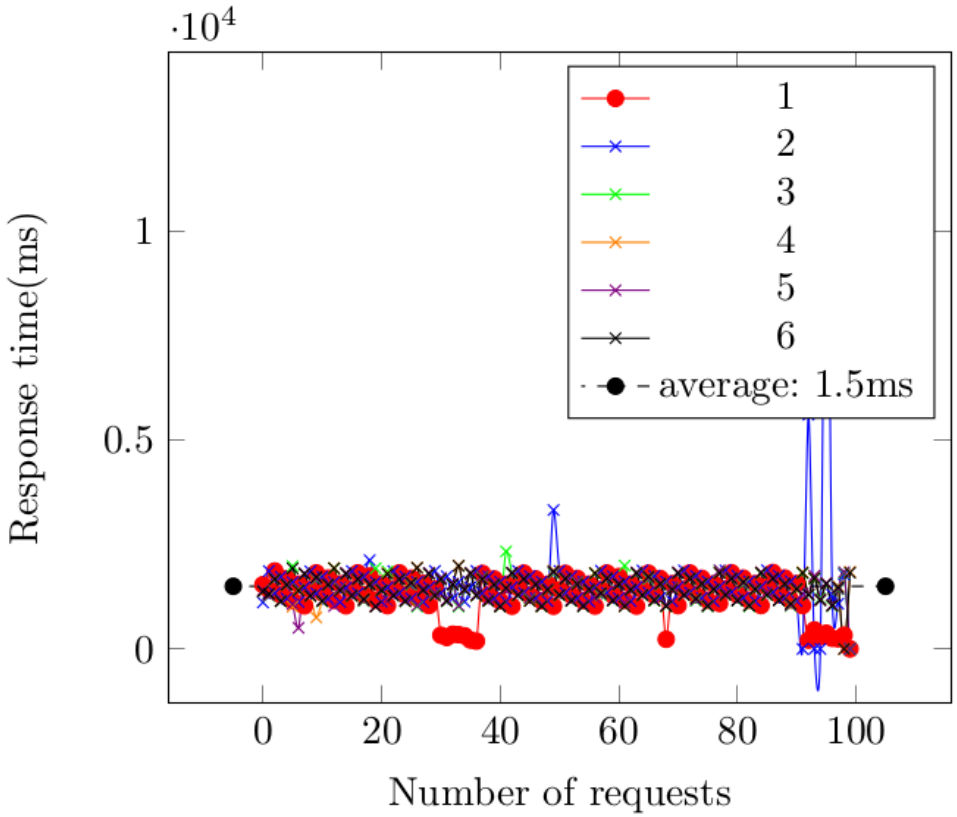
```
10.0.2.2 - - [02/Feb/2018 14:05:53] "POST /mtp/123/123456789 HTTP/1.1"
200 -
> The process 123456789 wants to get access to 123 in mtp mode
> Verifying process_id : 123456789
process 123456789 registered
Token decode result : {u'iss': u'123', u'rec': u'123456789', u'sub':
u'mtp', u'exp': u'201802020910'}
This token is no more valid. Starting 123456789 authentication...
```

Figure 9: case 6, the token is no more valid. The application needs to be authenticated again

RESPONSE TO DATA EXFILTRATION

- Results : Performance evaluation

- 2,5 % overhead thanks to Caffeine Mark **[caff1][caff2]**



[caff1] Fuchs et al, "Scandroid: Automated security certification of android"
[caff2] http://www.benchmarkhq.ru/cm30/

SECURITY EVALUATION

- The Model

Intruder model: Dolev-Yao.

We assume that the intruder cannot break the cryptographic construction used to make secure channels (HTTPS).

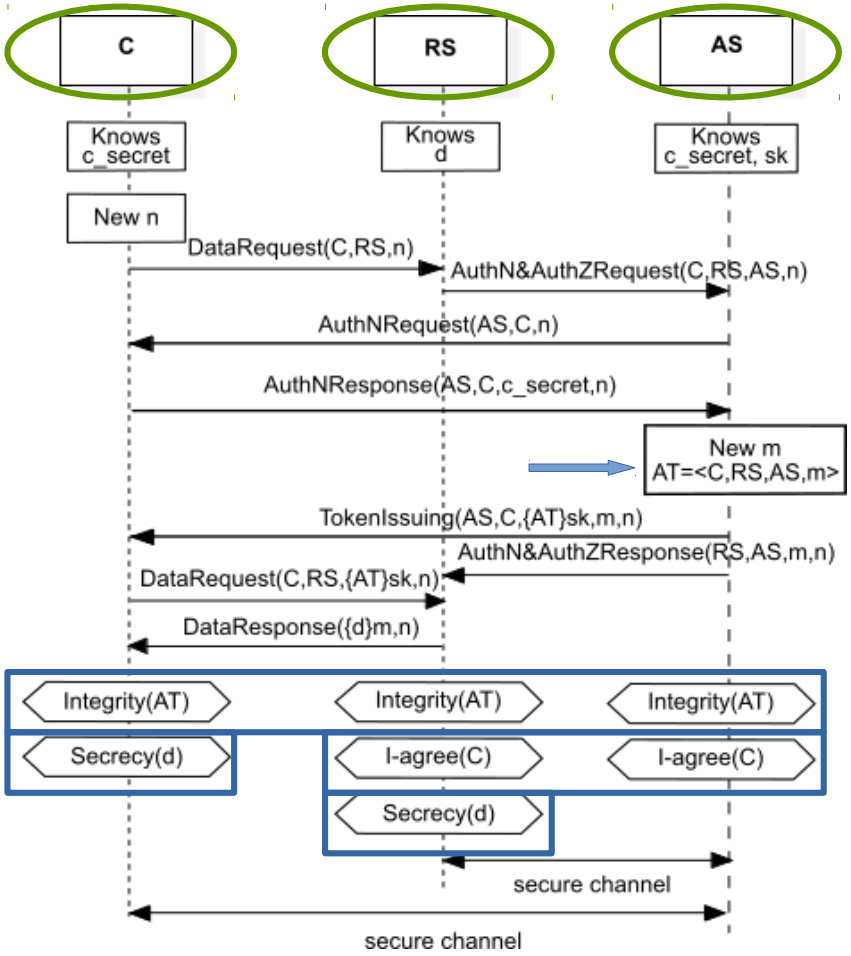


Figure 6: Proverif security model of the protocol **[proverif]**

[proverif] proverif.org

RESPONSE TO DATA EXFILTRATION

Conclusion & future work

Conclusion

- Security improvement with fine-grained filtering
- A priori attack detection
- Detect attacks in an acceptable amount of time (1.56 ms on average).
- Acceptable overhead execution on Android system (2.5 %).

Future work

- Implement the solution on Android 8

IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

THANK YOU !