An Analytics Framework to Detect Compromised IoT Devices using Mobility Behavior

Mukesh Taneja
Cisco Systems
Bangalore, India
tanejamukesh9@gmail.com

***Abstract*— Certain security mechanisms assume that the end device is secured. In an IoT network, the IoT device itself could be compromised. An attacker could steal the device, gain access to it and use this for more damaging attacks. I propose an analytical framework where I specify certain mobility behavior indicators that are computed at network nodes and optionally at IoT devices. These are communicated to an analytics server using lightweight protocol enhancements specified here. IoT user specifies expected behavior using these indicators. Analytics server analyzes expected and observed values of these indicators and informs if it detects some unusual activity.**

Keywords*— *Wireless Sensor Networks, IP Networks, Security Attacks, Mobility, Analytics.

I. INTRODUCTION

A network architecture where a device used for Machine to Machine (M2M) or Internet of Things (IoT) purpose communicates with an IoT Gateway (IoT GW) via wireline or wireless technologies is shown in Figure 1. An IoT GW as shown in Figure 2, could be an IP router with wireless capability to communicate with IoT devices. IoT (M2M) devices could communicate with IoT Gateway using Cellular (2G / 3G / LTE) [13], WLAN [14], IEEE802.15.4 [15], Ethernet or some other technologies. This IoT GW could use wireless or Ethernet based backhaul to communicate with other network nodes. Some of these devices may support single access technology while some (such as video cameras for surveillance purpose) may support multiple radio access technologies. An IoT service operator may deploy IoT service platforms to provide certain IoT capabilities to IoT users. An IoT user, such as a utility company or a cab company or a logistic company, to access IoT devices (and associated services) using IoT service platform.

Several IoT protocols have been specified or are getting specified in various organizations. Extensible Messaging and Presence Protocol (XMPP) [11], Hyper Text Transfer Protocol (HTTP) [9], Manufacturing Message Specification (MMS) [7], Open Smart Grid Protocol (OSGP) [8] and some other IoT protocols are being targeted for multiple IoT segments. IETF Constrained RESTful Environment (CORE) working group [3] is providing a framework for resource constrained application intended to run on IP networks. Representational State Transfer (REST) is an architecture style for designing network applications. As part of this work, Constrained Application Protocol (CoAP), is being specified. CoAP is an implementation of RESTful architecture and can be used for end-to-end communication in an IoT network. Loosely speaking, it is a lightweight version of HTTP for constrained networks though it offers new capabilities that are needed for IoT applications. It runs over UDP and uses Datagram Transport Layer Security (DTLS) [2] to protect application layer traffic. IoT Service Platform (on behalf of an IoT user such as a utility company) contacts an IoT device (such as a smart meter or a sensor device) using its Uniform resource Identifier (URI) [12]. This URI is translated to IP address of the device. An end-to-end protocol stack for an IoT network that uses IEEE802.15.4 devices and where IoT GW uses Long Term Evolution (LTE) based backhaul is shown in Figure 3. An IoT GW, as shown in Figure 2, could run a proxy for some of these IoT protocols, intercept data being exchanged for IoT purposes and provide value added services.

Several network security mechanisms assume that the (user) device is secured. In IoT / M2M space, device itself could be compromised. A device is considered compromised if an attacker gains control or access to the device itself after it is deployed. In an invasive attack [5], attacker physically breaks into hardware by modifying its hardware structure. In a non-invasive attack [5], data is taken from device without any sort of structural modifications done on the device. For example, interface between Subscriber Identity Module (SIM) card and device System on Chip (SoC) may not be secure in some cases and an attacker could take advantage of that. It is also possible that an attacker could steal the device itself and use this as a stepping stone for more damaging attacks.

I define certain mobility related behavioral indicators for IoT (M2M) devices in this paper and provide mechanisms to collect those at an analytics server. In this framework, IoT service platform to provide expected values of these indicators to Analytics server. Observed values of these indicators are collected via different network nodes, such as Wireless LAN Controller (WLC), IoT GW, LTE/3G/WiFi Access Point (AP), Radio Network Controller (RNC) and Mobility Management Entity (MME). Observed values of these indicators are also collected (optionally) from IoT / M2M devices. Analytics server analyzes expected and observed behavioral indicators, and detects if an IoT (M2M) device is acting in an unusual manner. It can detect compromised IoT devices for the

Disclaimer: Opinions expressed in this paper are author's own personal opinions and do not represent his employer's view in anyway.

scenarios where mobility behavior of IoT device changes after a security attack or IoT device is detected in an area where it is not supposed to be. Resource allocation mechanisms at different network nodes (and macro level device tracking mechanisms) can also make use of information available in the framework specified here.

This paper is organized as follows. I specify mobility related behavioral indicators and mechanisms to collect these indicators in section II. I define an analytical framework to detect possibility of security attacks on IoT devices in section III and conclude this paper in section IV.

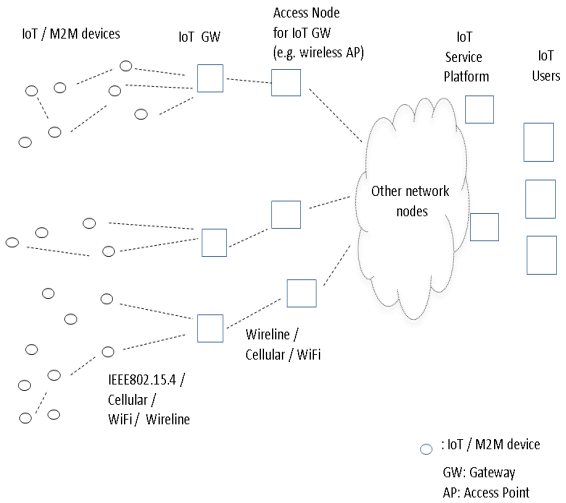


Figure 1: An IoT Network

II. MOBILITY BEHAVIOR INDICATORS

A. Mobility Behavioral Indicators

I specify following mobility behavioral indicators in this paper:

- AP id with which an IoT device is associated along with the time interval for association. This is captured by wireless controllers or gateways (such as WLC, MME, RNC, IoT GW) and reported to Analytics server. Radio Access Technology (RAT) used for each such association is also provided. Here AP is used in a generic sense and could be supporting Cellular (such as LTE, 3G, 2G) [13], WiFi [14] or even IEEE802.15.4 [15] based access technologies. In the case of 802.15.4, this AP may be acting as a coordinator node for a mesh of 802.15.4 devices and this coordinator could be part of IoT GW itself. In a 3GPP networks, an AP gets to know about presence of a device in its area when that device tries to establish Radio Resource Control (RRC)[13] connection with that AP. MME / RNC[13] also get to know about it during 3GPP Attach process. If a GPS is available at AP or location information is configured at AP, this also provides some information about the location / region of device (as it is in coverage area of this AP).
- Fraction of time device has been moving during past certain duration. To keep overhead low in the network, information is not sent frequently and is sent in a compact format indicating low, medium, high and very high values. A new CoAP Option is added that can be used to piggyback this information on existing CoAP messages.

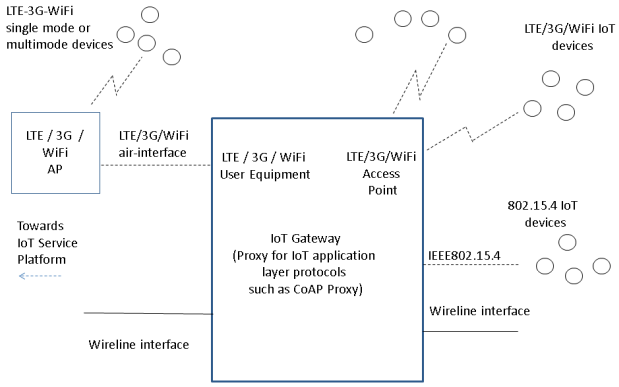


Figure 2: An IoT GW with wireless / wireline interfaces

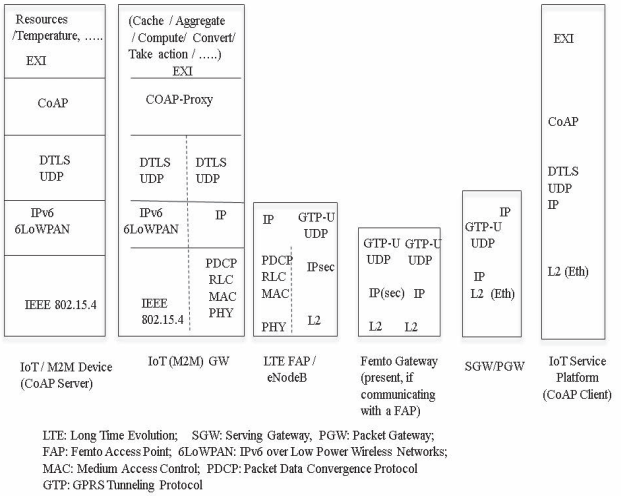


Figure 3: End-to-End Protocol Stacks in an IoT Network

- Number of transitions made by IoT device in certain time interval. As devices moves, stops, moves again,

that information is captured by this mobility indicator. To keep load low in the network, a 2 bit field is used to convey this information indicating low, medium, high and very high values. A new CoAP Option is added that can be used to piggyback this information on existing CoAP messages.

B. *Mobility Behavioral Indicators Detected and Reported by Wireless / Network Controller Nodes and IoT GW*

A network assisted mechanism to capture mobility related information at Analytics server is specified here. A wireless controller or gateway (such as WLC, RNC, MME, IoT GW.) communicates to Analytics server when an IoT device associates (and/or communicates) with an AP (or coordinator node such as for IEEE802.15.4 based networks) that is served by this controller or gateway. It also communicates information about previously associated AP (or coordinator node) as shown in Figure 4. Information about previously associated AP is available at wireless controller (or gateway) if that other AP was also using the same wireless controller (e.g. in the case of inter-AP intra-wireless controller handover). In this figure, IoT device d1 associates with AP_k that is using radio access technology R_k at time t_k for k = 1, ….n+2. As multimode IoT devices could use different radio access technologies (such as LTE, 3G, 2G, WiFi etc.) at different time to associate with APs (or coordinator nodes), that information is also captured.

Wireless controller concatenates this information over a period of time and communicates using a compact packet structure to analytics server as shown in Figure 5. I use CoAP as shown in Figure 6 to communicate these behavioral indicators from wireless controllers (or gateways) to analytics server. Analytics server running CoAP Client to register with wireless controller or gateway (such as WLC / RNC / MME / IoT GW) running CoAP Server for receiving access node change related events.

C. *Mobility Behavioral Indicators Captured and Reported by IoT Devices*

I capture the following additional information related to mobility pattern of IoT devices:

- Fraction of time IoT device was moving in the past certain duration, frac-moving as indicated in Figure 7, and
- Mobility transitions of IoT device in the past certain duration, mob-transitions, as indicated in Figure 7

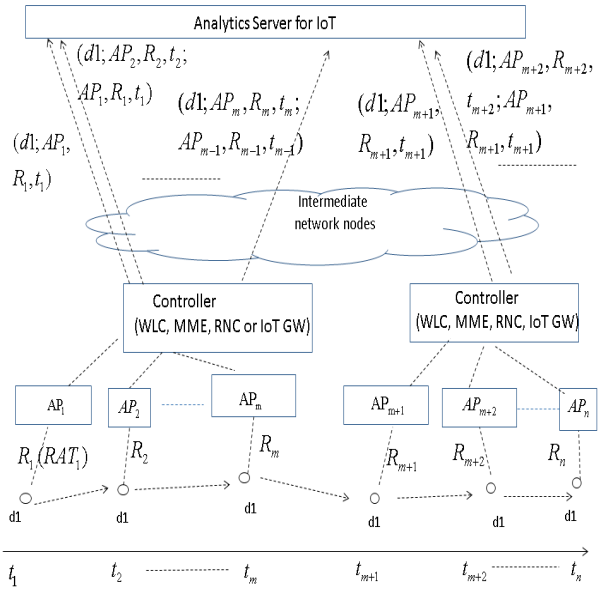
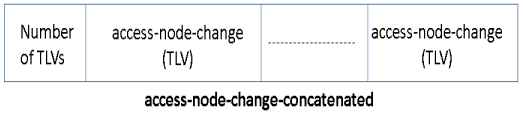


Figure 4: Wireless Controllers (such as WLC, MME, RNC, ….) detecting and reporting mobility indicators

Mobility Indicators provided by wireless controllers	Type	Length of value field	Value
access-node-change (inter-AP, intra-Controller handover)	00000001	Length	IoT device id, (AP id, time when it associated with this AP) for current and previously associated AP
access-node-change (inter-AP, inter-Controller handover)	00000010	Length	IoT device id, (AP id, time when it associated this AP) for current AP only



TLV: Type, Length and Value pair

Figure 5: "access-node-change-concatenated" communicated from wireless controllers to analytics server

These mobility indicators are computed as shown in Figure 8. IoT devices to (optionally) compute and provide these indicators to analytics server. This information is sent infrequently to keep load in the network and overhead on IoT (M2M) devices low. Instead of sending absolute value of these indicators, 2-bit field for each of the indicator, indicating low (00), medium (01), high (10) and very high (11), as shown in Figure 9, is sent. CoAP can be used to convey this

information from IoT device to analytics server using CoAP Observe / Notification mode as shown in Figure 10.

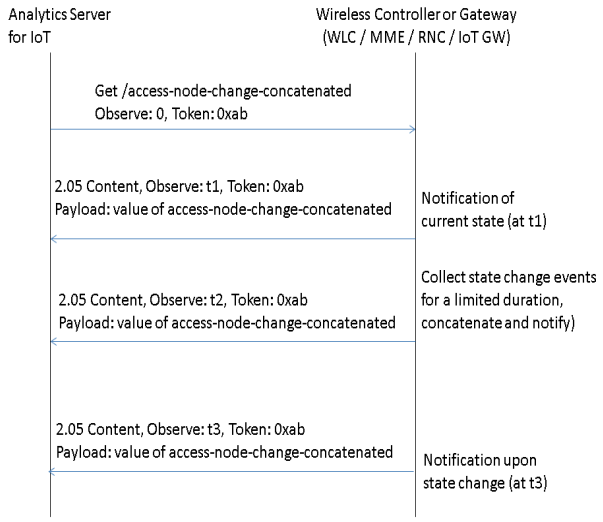


Figure 6: Use of CoAP Observe / Notification

I also provide an alternate method by adding new CoAP Option [13] for frac-moving-indicator and mob-transition-indicator as shown in Figure 11. In this case, mobility indicator can be piggybacked over other CoAP messages. It is also possible for IoT GW to aggregate these mobility events from IoT devices and then communicate these to analytics server. Analytics server to analyze this information and indicators obtained via network assisted methods as specified earlier as explained in the next section.

Mobility indicators from device (i.e. mobility-indicators-device), is an optional field but an IoT operator can always make a subset of this mandatory for devices owned and controlled by it.

III. PROCESSING AT ANALYTICS SERVER

A. Expected Behavior of IoT Devices

In this framework, I ask IoT user that wants to take advantage of solutions provided by us, to specify expected mobility behavior of its IoT devices to Analytics server as shown in Figure 12. I don't make it mandatory for IoT user to specify all the mobility indicators specified in previous sections of this papers. IoT user could specify one, two or all the three mobility indicators specified here for each and every device.

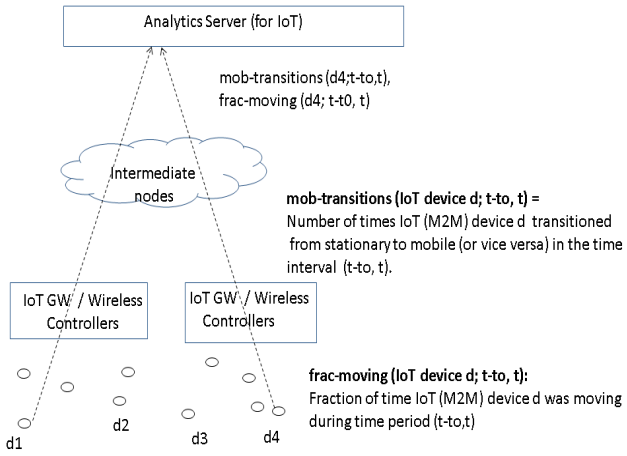


Figure 7: Observed mobility indicators communicated by IoT devices to analytics server

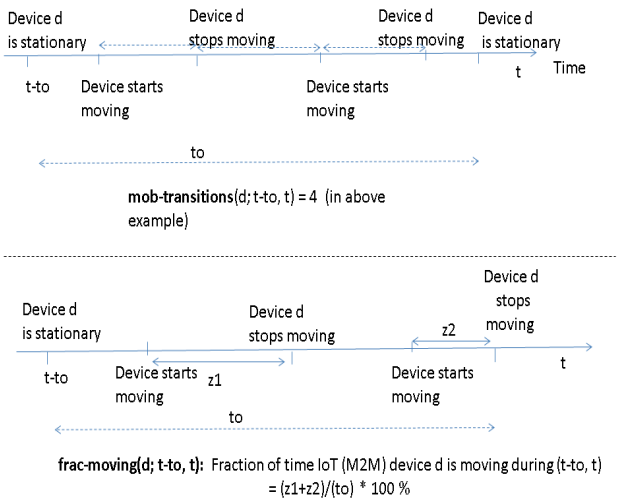


Figure 8: Computation of mobility indicators

B. Overall Processing at Analytics Server

Analytics server gets observed mobility indicators as specified in previous sections and shown in Figure 4 and Figure 7. A given IoT system may support all or of a subset of these mobility indicators. It uses mobility indicator, access-node-change that was communicated by wireless controller to analytics server as shown in Figure 4, to compute number (and list) of APs visited by IoT device d during any time interval (t_1, t_n) as shown in Figure 13. Expected and observed mobility behavioral indicators are listed in Table 1.

As shown in Figure 14 and Figure 15, analytics server to compare expected and observed / captured indicators. If difference of any of these indicators is above a threshold or if it detects some unusual activity, it can indicate this to analytics server and collect data for some more time. It could also run some additional tests on the device. Eventually, it could take corrective action like blocking access (or limiting access) to a device that is not behaving in its usual manner and is suspected to be compromised.

frac-moving-indicator:
00 if 0 <= frac-moving < 25%
01 if 25% <= frac-moving < 50%,
10 if 50% <= frac-moving < 75%,
11 if 75% <= frac-moving <= 100%

mob-transition-indicator:
00 if 0 <= mob-transitions < n1
01 if n1 <= mob-transitions < n2,
10 if n2 <= mob-transitions < n3,
11 if n3 <= mob-transitions <= n4

n1, n2, n3, n4: Configurable parameters (integer values)

mobility-indicators-device:

xy	z	frac-moving-indicator	mob-transition-indicator	padding	time interval / offset (z)
2 bits	1 bit	2 bits	2 bits	1 bits	

xy:11, frac-moving-indicator and mob-transition-indicator are present
xy=10, only frac-moving-indicator is present and valid in mobility-indicator-device;
xy=01, only mob-transition-indicator is valid in mobility-indicators-device

z=0, time interval / offset is specified as offset from the last reported time
(i.e. measurements done in continuation)
z=1, time interval / offset is specified as (starting time for measurement, offset from this)

Figure 9: Compact representation of mobility indicators (mobility-indicators-device)

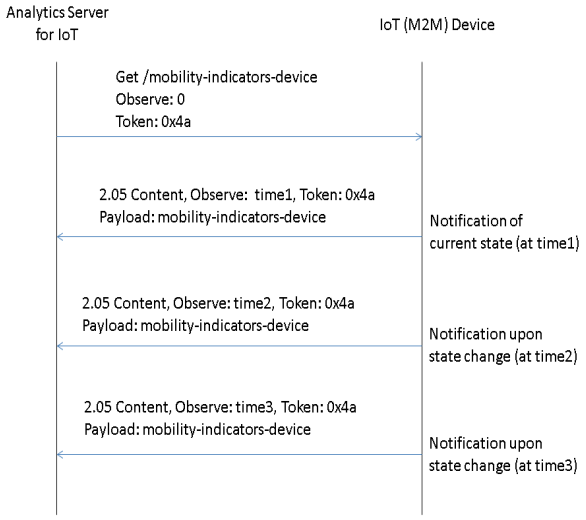
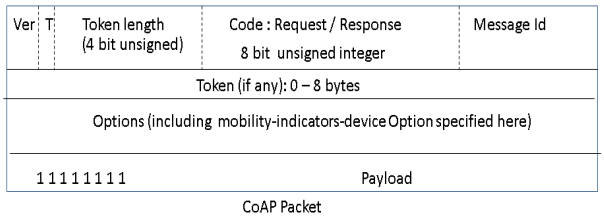


Figure 10: Communication of observed mobility indicators from IoT devices to Analytics server



T: 2 bits to indicate type of message: Confirmable (0), Non-Confirmable (1), Ack (2), Reset (3)

New CoAP option is added here for mobility-indicators-device

Option Number	C	U	N	R	Name	Format	Length	Default
Option number for "mobility-indicators-device"				X	mobility -indicators -device	Integer	1 byte	

C: Critical, U: UnSafe, N: No-Cache-Key, R: Repeatable

Figure 11: New CoAP Options for observed mobility-indicators-device – An alternate method

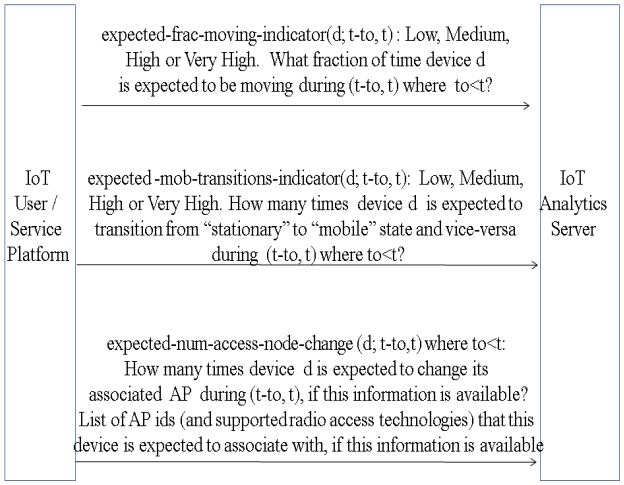


Figure 12: Expected Behavioral Indicators

IV CONCLUSIONS

I have proposed an analytics framework that can be used to detect IoT (M2M) devices that have been compromised in an IoT network. I have defined mobility behavior and related indicators to detect if an IoT device is acting in some unusual manner. I have also proposed lightweight protocol enhancements for this purpose. No one mechanism can detect compromised devices in all scenarios. This method detects compromised IoT devices for scenarios where mobility behavior of device has changed as specified in this paper. This

analytical framework can also potentially be used to improve efficiency of resource management and mobility mechanisms in IoT networks.

$$(t_1, t_n) = \{(t_1, t_2), (t_2, t_3), ..., (t_a, t_b), ..., (t_m, t_{m+1}), ..., (t_{n-1}, t_n)\}$$

$$I(d; t_a, t_b) = \begin{cases} 1, \text{if } access-node-change(d; \text{AP}_a, t_a; AP_{a-1}, t_{a-1}) \text{ or} \\ \qquad access-node-change(d; \text{AP}_a, t_a) \text{ is received} \\ 0, otherwise \end{cases}$$

Number of times access node (i.e. AP) changes for device d during (t_1, t_n):

$$\text{Observed num-access-node-change}\ (d; t_1, t_n) = \sum_{(a,b)} I(d; t_a, t_b), \text{ such that } t_1 \le t_a \le t_b \le t_n,$$
$$1 \le a \le n-1,\ 2 \le b \le n$$

List of AP ids visited by device d during any given time interval also follows from access-node-change

Figure 13: Observed number of APs visited by an IoT device d during a time interval (t_1, t_n)

Expected mobility behavioral indicators	Observed mobility behavioral indicator
expected-num-access-node-change	num-access-node-change (device, time interval)
expected-mob-transitions-indicator	mob-transitions-indicator (device, time interval)
expected-frac-moving-indicator	frac-moving-indicator (device, time interval)
Expected list of AP ids (along with supported RATs) with which device could associate in a given interval	List of AP ids, along with RAT used (by IoT device) to associate with these APs in a given time interval

Table 1: Expected and Observed mobility indicators.

REFERENCES

[1] Al-Sakib Khan Pathan, Hyung-Woo Lee, Choong Seon Hong, "Security in Wireless Sensor Networks," Issues and Challenges, Proceedings of 8th IEEE ICACT 2006, Volume II, February 20-22, Phoenix Park, Korea, 2006, pp. 1043-1048

[2] E. Rescorla, N. Modadugu, "Datagram Transport Layer Security (DTLS) version 1.2," IETF RFC 6347.

[3] IETF Constrained RESTful Environment (core) working group

[4] IEC website (www dot iec dot ch)

[5] Javier Lopez, Rodrigo Roman and Cristina Alcaraz, "Analysis of Security Threats, Requirements, Technologies and Standards in Wireless Sensor Networks," FOSAD 2007/2008/2009, LNCS 5705, pp. 289-338, 2009, Springer Verlag Berlin Heidelberg 2009.

[6] Message Queue Telemetry Transport (MQTT) (www dot mqtt dot org)

[7] Manufacturing Messaging Specification (MMS), ISO 9506

[8] Open Smart Grid Protocol (OSGP), ETSI Group Specification OSG 001

[9] R. Fielding et al, "Hypertext Transfer Protocol – HTTP / 1.1," IETF RFC 2616.

[10] T. Berners-Lee, R. Fielding, L. Masinter, "Uniform Resource Identifiers (URI): Generic Syntax," IETF RFC 2396.

[11] XMPP (Extensible Messaging and Presence Protocol), xmpp.org

[12] Z. Shelby, K. Hartke, C. Bormann, B. Frank, "Constrained Application Protocol (CoAP)", IETF draft-ietf-core-coap-18.txt

[13] 3GPP Specifications

[14] IEEE802.11 Working Group

[15] IEEE802.15.4 Working Group

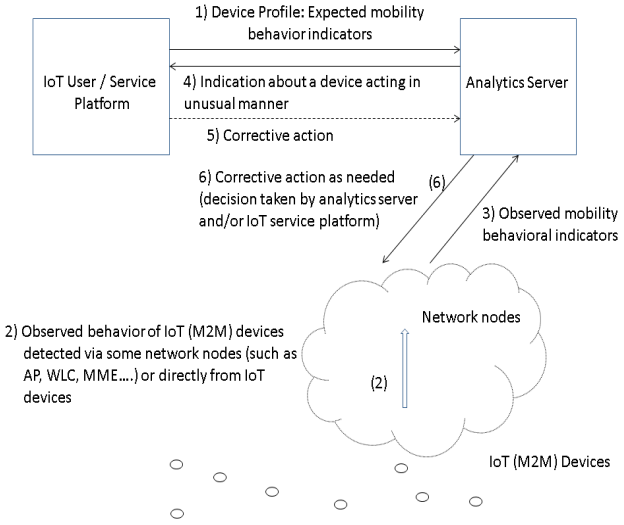


Figure 14: Expected and Observed Mobility Indicators: Overall Processing

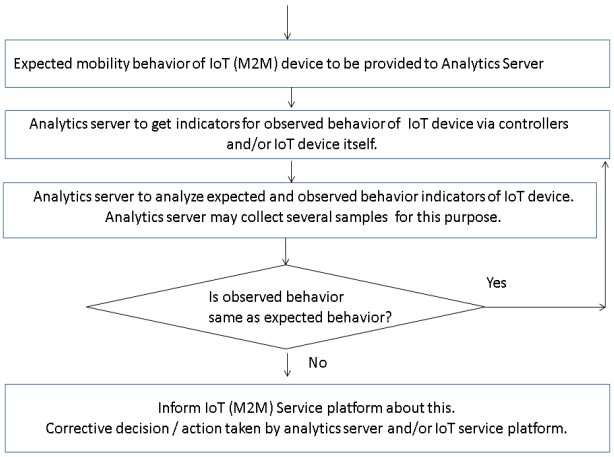


Figure 15: Flow chart