# Internet of Things Security - Multilayered Method For End to End Data Communications Over Cellular Networks

Craig Lee
AT&T Labs - Ecosystems & Innovation
Internet of Things Foundry
Plano, Texas, United States of America
Craig.Lee@att.com

Andrea Fumagalli
Open Networking Adv. Res. (OpNeAR) Lab
The University of Texas at Dallas
Richardson, Texas, United States of America
andreaf@utdallas.edu

**Abstract — The aim of this paper is to put forth a multilayered method for securing data transport from a cellular connected Internet of Things device to a host through a cellular network. This method employs many interlocking security elements – described in this paper – that when implemented in their totality provide a highly secure connectivity solution.**

***Keywords— Internet of Things, Security***

## I. Introduction

With the explosive growth of Internet of Things (IoT) solutions comes the greater concern over security issues associated with the plurality of devices being connected. It is projected that the growth in Internet of Things connected devices will exceed 20 Billion devices by 2020. Many of these solutions will be leveraging and utilizing cellular connection connectivity to interconnect.

Poorly architected cellular connectivity can open the solution to potential security issues. A highly secured architectural solution requires a multilayered security approach encompassing the overall architectural design for connectivity, spanning from the edge device all the way up to the destination host for processing, storage, and further use.

Alternatives to cellular connectivity include solutions that may utilize Ethernet or wireless Wi-Fi and depend upon the public Internet for transport of data from the edge device to the host. These alternative solutions present a number of security vulnerabilities. The autonomous nature of an IoT solution – i.e., a machine communicating autonomously with a backend host, which by definition does not require human interaction nor control – provides the justification for remote monitoring & tracking of any number of useful IoT solutions. Due to the nature of the data transport not being constantly monitored or initiated by human input, an intrusion may occur without a human noticing as they might with a wireless handheld not performing as expected. In fact, several recent security attacks have occurred whereby a vulnerability associated with Wi-Fi over the public Internet was used as a way of getting into the stream of data and taking control or modifying the functionality of the IoT device.

The multilayered methodology described here within provides a secure wireless connection utilizing packetized data as found in 3G and above cellular carrier technology to establish a robust connection between the IoT device and the backend host for bidirectional communications. This methodology consists of a number of interlocking functional elements, which are discussed in turn throughout this paper. They include:

- SIM-based authentication and key agreement,
- Radio access network encryption,
- Custom access point name (APN),
- Private non-routable TCP/IP addressing,
- Non-split tunnel routing schema,
- Point to point data transport to host,
- Destination host router monitoring,
- No direct device to device communications,
- SIM toolkit IMEI validation and alerting, and
- PIN locking of SIM

## II. Architectural System Description

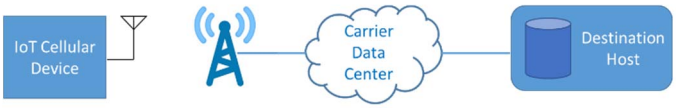


Fig. 1. Simplified end to end architecture for connecting IoT wireless devices to a destination host.

Figure 1 shows a highly simplified diagram of the architecture described in this paper. The IoT cellular device is connected to the local serving cell tower via an encrypted radio access network. The local tower is securely connected to the home carrier's mobility data center. The destination host is connected via a point to point encrypted link.

Two inherent benefits of this architecture are 1) leveraging standards-based elements so that the solution may be applied to off-the-shelf IoT devices and 2) end-to-end security, which is achieved without a costly over-the-top data encryption from the

978-1-5386-4980-0/19/$31.00 ©2019 IEEE

device to the host. While encrypting the data stream prior to leaving the edge device and unencrypting at the host does provide an extra layer of security, it typically comes at the cost of higher data throughput which in the cellular space is metered per byte or kilobyte. The architecture described in this paper provides desired security without incurring increased data payload for security encryption. For many classes of Internet of Things applications, this solution allows for TCP or UDP data to be sent in clear text, yet the secure architecture provides a suite of satisfactory security interlocking functional elements that properly combined can prevent unauthorized access to the data channel.

## III. The Interlocking Functional Elements

Each security functional element available in the architecture in Fig. 1 is briefly described in this section.

### A. SIM-based authentication and Key Agreement

The first critical element of the secure architecture is the subscriber identifier module (SIM) [1]. The SIM's basic function is to protect authentication keys from being compromised. The SIM consists of a microprocessor that incorporates a number of hardware protection technologies to prevent compromise through chemical decomposition, x-ray or any number of attempts to reverse engineer. Additionally, protection procedures are applied to the I/O pins of the SIM to prevent forced external anomalies from rendering the SIM vulnerable to compromise. For instance, inducing a higher or lower voltage on the TX and RX pins in reference to the supply and ground in an effort to cause the I/O circuitry to latch up or go into an unintended state is monitored by the SIM I/O circuitry and will disable the SIM if these conditions exist. The SIM is also protected against anomalous clocking and input data [2].

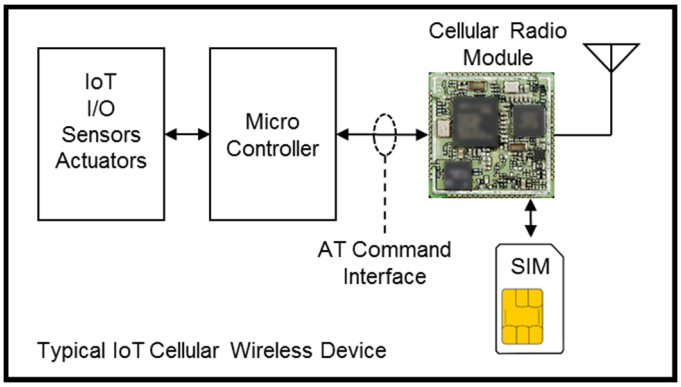


Fig. 2. Typical IoT wireless device components highlighting SIM connectivity to the Radio Module and the AT command interface where radio chip set probe information is exchanged.

As shown in Fig. 2, the connectivity to the SIM is only through the cellular radio module [2] preventing the onboard microprocessor from directly accessing the SIM. Any communications with the SIM is performed solely by the radio stack layer built into the radio module.

When a cellular device, in this case an Internet of Things device, is powered up the radio is automatically programed to scan the available radio band and catalog possible cell carriers that it may be able to attach to [3][1]. This process is set to prefer a carrier tower that first matches its own carrier's mobile network code (MNC). This is accomplished by searching the radio bands looking for broadcast codes that match the SIM's International Mobile Subscriber identity (IMSI) code. If a matching carrier is not located, namely the SIM *home carrier*, the radio scans for other *serving carriers* and compare them to an updatable list of preferred roaming partners.

3GPP specification 11.11 [1] describes the authentication mechanism and cipher key generation utilized by the SIM and carrier network to authenticate the SIM and device.

The authentication and key agreement mechanism diagrammed in Fig. 3, utilizes a secret key K which is known only to the SIM and the Internet of Things home carrier [4].

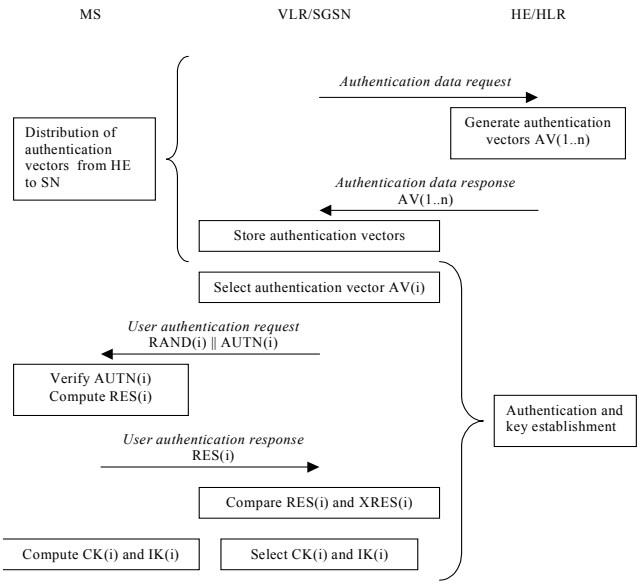


Fig. 3. Authentication and Key Agreement diagram [4]

The SIM provides a very secure method to authenticate an IoT device even before the data link is established. The SIM also plays a role in several of the other security functional elements as articulated next.

[1] There are many methodologies in place, not addressed in this paper, to ensure that the Internet of Things device uses the desired radio carrier to attach to.

### B. Radio access network encryption

An additional security element as part of the overall data transport security methodology is that the radio link layer between the IoT device radio module and the cellular tower is 128-bit encrypted, utilizing the key from the previous section, as part of the standard GSM protocol for 3G and above transmissions [4].

### C. Custom access point name (APN)

Once the Internet of Things device has authenticated with the serving cell carrier, the microprocessor within the Internet of Things device can initiate a data session for TCP/IP transport.

The command set is an extension of the Hayes modem command set a.k.a. AT+. The microprocessor submits an AT command to the radio module passing along specific variables including the access point name (APN) that the application wishes to connect with [10]. In consumer phones, the APN is a generic name common to all phones that the home carrier uses to route consumer data from handsets and tablets out to the public Internet. These data packets are typically routed through a port address translation (PAT) and then through a stateful firewall out to the public Internet. While this common practice works well for consumer based solutions, which need access to the broader Internet, it introduces a security vulnerability point in an Internet of Things architecture.

The secure IoT architecture presented herein makes use of a custom APN assigned to each of the enterprise customers deploying IoT devices. This custom APN is unique to the enterprise which has provisioning and management capabilities of IoT devices. The custom APN allows such enterprise (and only such enterprise) to give an IoT device permission to access that Enterprise's custom APN for the data transport.

The use of a unique custom APN offers the following additional security mechanism. While a malicious entity may discover or guess an enterprise's custom APN name, the action of merely requesting that APN through the AT command would not allow for provisioning and authentication of that device to use such custom APN. The mechanism for ensuring that the IoT device requesting a custom APN has permission to use that custom APN for its data transport builds upon the authentication transactions previously described in Section III-A.

At this juncture in the establishment of data service, the serving element within the home carrier's network is the Gateway GPRS Support Node (GGSN) or Packet Data Network Gateway (PDN-GW). The GGSN/PDN-GW acts as the mediator for establishing the data session and characteristics over the connection spanning from the IoT device over the radio network into the mobility data center and into the GGSN/PDN-GW element. The outbound direction from the GGSN/PDN-GW out of the mobility data center is also mapped and controlled by parameters within the custom APN construct whereby the routing of the TCP/IP data packets instead of going to the public Internet (a.k.a. consumer experience) would go from the mobility data center out through a point to point connection to the destination host directly (receiving and collecting the IoT device's provided data). More details can be found in sections below.

### D. Private non-routable TCP/IP addressing

One of the key secure methodologies utilized in the selection and set up of a custom APN construct is the use of private IP addressing, typically a class B 10.x range. The GGSN/PDN-GW assigns the class B IP address during the data connection set up between the IoT radio module and the mobility data center. The dynamically assigned private IP address is selected from the available range of IP addresses as defined by the APN construct.

The 10.x non-routable IP address that is assigned to the radio link is preserved in the packet along the path between the IoT radio module and the GGSN/PDN-GW avoiding any Network Address Translation (NAT) or Port Address Translation (PAT) or any other type of conversion into a public routable IP address. Because of the nature of the overarching architecture, the private IP addressing schema does not route to the public Internet. Given the non-routable nature of the IP addressing that is preserved end-to-end, even if a malicious packet were to make its way into this secure pipe or one of the IP packets were to "escape" the pipe, they would be immediately dropped by the first router hop due to their non-routable IP address. The inherent security that comes with the non-routable IP addressing schema and the fact that a custom APN can only be established and provisioned by the intended enterprise customer delivers a highly secure solution that prevents data traffic from being maliciously intercepted or interjected.

The use of non-routable IP addressing does not prevent the application from accessing information available through the public Internet. For instance, an IoT device in a commercial water sprinkler may have to query an external informational source such as weather information. The IoT controller of the commercial water sprinkler may request the weather forecast for determining if it needs to water on a given day. That IP packet being generated by the microprocessor of the IoT controller as a request outbound would have a destination of a National Oceanic and Atmospheric Administration (NOAA) URL that provides a machine-readable weather feed. The secure schema architecture as articulated in this paper would deliver that public destination addressed packet through the system to the customer's router at their data center. That public IP addressed packet could then be proxied routed out to the public Internet after passing through the enterprise customers router and firewall.

### E. Non-split tunnel routing schema

Poorly architected solutions often provision an APN to split the data packet tunnel. In other words, while the 10.x private data packets are directly sent to the enterprise customers as previously described, the public destined packets are directly

routed from the mobility data center off to the public Internet. While this is a possible architecture, it is one that breaks this security methodology because now an IoT device has a path to the public Internet that is not under the control of the enterprise customer.

### F. Point to point data transport between cellular carrier and host

The point-to-point connection between the mobility data center and the destination host may take the form of an IPsec VPN tunnel, or MPLS, or frame relay, or any number of landline secure point-to-point connectivity solutions that might may be offered by the outbound side of the carrier's service. Most prevalent is IPsec VPN generally utilizing Cisco firewall VPN equipment at both the carrier and the host data sites. By combining the two functional elements provided by the custom APN and IPsec VPN tunnel, it is possible to establish a closed secure pipe starting at the radio module within the IoT device all the way through the tower and the carriers mobility data center transiting through the GGSN/PDN-GW, routed out through the mobility data center via a custom off-the-Internet secure pipe to the enterprise customers host terminating at the IPsec VPN router.

### G. No direct device to device commuinication

It is interesting to note here that a commonly used term for IoT is Machine to Machine (M2M), creating the misconception that IoT devices talk directly with one another. The methodology as described in this paper prohibits direct device-to-device communications via the carrier or customer's host router. In fact, IoT device communications are only routed to the application layer managing the IoT solutions within the backend host. If the solution requires and exchange of data between IoT device A and IoT device B, this goal is achieved through the application layer at the backend host server, as opposed to a direct data packet exchange between the two devices. While it is technically possible to configure an APN to 'hairpin' device-to-device routing through the carrier's data center, this approach breaks the security methodology described herein. If IoT devices were allowed to communicate directly with one another through the carrier's data center, this procedure would not leave a record or footprint in the customer's router or backend host system. In other words, the devices may be chatting back and forth without the enterprise customer having record of or seeing the M2M device traffic, thus being unable to probe the traffic for malicious behavior.

### H. Destination host router monitoring

Another intangible benefit of this secure routing schema is that all data packets to and from the IoT device passes through the enterprise customer's router. Deep packet inspection at the customer's router can detect in real-time that abnormal data behavior is occurring from the IoT devices, which may indicate fraudulent or malicious activity. Automated alarming is then used to trigger the provisioning system to disable the IoT device and alert technical intervention. This would be impossible if the publicly routed data packets were tunneled off at the carrier site and only those packets with the 10.x would be passing through the customer's enterprise router. By delivering every packet through the customer enterprise router, the customer's host data center has a holistic view of all traffic coming from and going to the IoT devices and can forensically detect if fraudulent behavior is or has occurred.

### I. SIM toolkit IMEI validation and alerting

Once the IoT device has established a secure connection between its radio module and its backend serving host, additional security is garnered in the form of physical protection. One method already described is the physical hardware security included in the SIM element to protect the encrypted keys held within. A second hardware feature is the unique serial number that is contained within the radio chipset. This is referred to as the IMEI or international mobile equipment identifier. Each manufacturer of wireless devices after completing the required PTCRB approval testing is assigned its own unique range of IMEI's for its product. The IMEI serial number is 15 digits long (16 in the IMEI software version), with six of the digits containing the unique serial number and the preceding eight digits identify the manufacturer/product code referred to as Type Allocation Code (TAC). Hence an IMEI is capable of identifying up to one million unique devices. These IMEI numbers are registered in a searchable database for members of the PTCRB body. Cellular carriers can readily identify the make and model of an IoT device from the unique IMEI code that is stored within the radio chipset. Part of the records that are exchanged between the IoT device radio module and the serving carrier during authentication is the pairing of the IMEI and the International Mobile Subscriber Identity (IMSI), which is the serial number of the SIM. The carrier can use both codes to authenticate, allow or deny connectivity services if fraudulent behavior is detected with any of these elements, and determine whether the device is a trusted device that has undergone rigorous certifications by the carrier and the industry to achieve its unique IMEI code sequence.

There is another unique security feature offered by the SIM. Being a microprocessor unto itself, which has executable code space, the SIM executes a series of security mechanism programs that are only known to the carrier. The methodology in this paper describes a sequence within the SIM programming that upon power up requests the IMEI of the radio module to which is directly connected. [2] The SIM being a secure storage location either has received a copy of its allowed connected device's IMEI or discovers it upon first power up. This allowable IMEI(s) are held in nonvolatile memory within the SIM and upon subsequent power ups the SIM makes the same request of the radio module it is attached to and compares the delivered IMEI to the stored IMEI within the SIM. If the two match, then the SIM assumes that it is in the equipment that it

was intended to be in as the equipment serial number matches the securely held permissioned serial number of the hardware.

If the SIM detects a different IMEI, it can assume that malicious behavior has occurred and that someone has removed the physical SIM from the trusted device and inserted into a possible malicious device. The SIM alerts the host mobility data center that a mismatch has occurred upon power up and the carrier can then take immediate steps to disable the SIM or prevent dataflow from occurring until the issue is investigated. This notification is extended to the enterprise customer's business methodologies providing visibility into whether a SIM has been moved from one IoT device to another.

### C. PIN LOCKING OF SIM

The standard PIN locking functionality of the SIM is utilized by the IoT device as a security method [5]. The device manufacturer creates a secure hashing algorithm within the device processor's firmware that is keyed from a hardware serial number, for instance the IMEI, that renders a unique 4-digit number. At time of assembly, the SIM associated with the device is programed into the locked state with the unique 4 digit code as the unlock key. At power up or reset, the SIM requests the unlock code via the radio interface to the device processor. The firmware runs the hashing algorithm to produce the 4 digit unlock code and passes to the SIM. If the code matches the SIM's then the SIM is enabled for operation. If the pin does not match after 3 attempts, the SIM is rendered unusable or blocked. This solution prevents a SIM from being removed from an IoT device and inserted into a consumer phone. The phone UI will request a pin that the malicious person would not know and after only 3 failed attempts, the SIM is no longer functional. There is an unblocking sequence using an 8-digit code known only to the device provider that can unblock a SIM and if that is incorrectly administered 10 times, the SIM becomes permanently inoperable [5].

## IV. CONCLUSION

This paper has set forth a multi-tiered solution for securely establishing end-to-end TCP/IP based Internet of Things communications over UMTS/LTE cellular-based networks. This methodology consists of standards based interlocking functional elements deployed in a securely architected carrier network providing a secure end to end communications channel for Internet of Things devices and applications.

## REFERENCES


[1] 3GPP TS 11.113rd Generation Partnership Project; Technical Specification Group Terminals Specification of the Subscriber Identity Module - Mobile Equipment (SIM - ME) interface

[2] 3GPP TS 22.022 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Personalisation of Mobile Equipment (ME); Mobile functionality specification

[3] 3GPP TS 23.122 3rd Generation Partnership Project; Technical Specification Group Core Network and Terminals; Non-Access-Stratum (NAS) functions related to Mobile Station (MS) in idle mode

[4] 3GPP TS 33.102 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; 3G Security; Security architecture

[5] 3GPP TS 21.111 3rd Generation Partnership Project; Technical Specification Group Core Network and Terminals; Universal Subscriber Identity Module (USIM) Application Toolkit (USAT)