Privacy and Security: Two Prominent Aspects of the Internet of Things

Mohammad Ali Jazayeri and Steve H. L. Liang

1 University of Calgary, smajazay@ucalgary.ca
2 University of Calgary, steve.liang@ucalgary.ca

Abstract

The *Internet of Things* (*IoT*) is a new paradigm referring to uniquely identifiable objects and their virtual representations in the Internet. Sensors, actuators, embedded computers and HTTP protocols would be beneficial to make physical things and their information accessible in the digital world. Two of the main issues of the IoT which have not been toughed seriously are *privacy* and *security* of information. These two aspects are major concerns for the IoT due to unauthorized access to personal devices and their information. As the resource discovery is handled by search engines or catalog services, mechanisms to preserve data privacy should be implemented in those resource discovery services. We also consider five solutions including *government rules*, *user agreement*, *role-based access control*, *anonymity* for measured phenomena, and *location obfuscation*. On the other hand, for the security issue, we consider Transport Layer Security (TLS), Secure Socket Layer (SSL), and Transport Control Protocol (TCP) in the communication layer of our proposed architecture. Moreover, in the service layer, we implement access control algorithms to restrict queries, and public key encryption (RSA) to guarantee data integrity.

Background and Relevance

The Internet of Things (IoT) is a new paradigm referring to *uniquely* identifiable objects and their virtual representations in the Internet. The basic concept of the IoT is the ubiquitous existence of various things or objects that can communicate and cooperate with each other in order to achieve shared goals (Atzori, et al. 2010).

Bormann et al. (2012) analyzed and categorized IoT objects into three categories: *class-0* devices (i.e., impossibly limited devices), *class-1* devices (i.e., devices with about 10 Kbytes of RAM and 100 Kbytes of code space), and *class-2* devices (i.e., devices with about 50 Kbytes of RAM and 250 Kbytes of code space). Bormann et al. (2012) argued that the class-0 devices need extra help to communicate with other devices; the class-1 devices cannot easily communicate with other devices or applications through traditional XML-data representations and protocols; and the class-2 devices should be able to communicate with the traditional transfer protocols and data encodings. Since class-1 devices are relatively inexpensive and small size, they would be a good candidate for the IoT. Thus, we initially develop a tiny web service for a class-1 IoT device, which makes the IoT object *self-describable* and *self-contained* in order to describe and advertise both itself and its capabilities. Since in the IoT, daily devices would be globally accessible through the World Wide Web, so the security and privacy preservation are highlighted.

The privacy means the data will never be disclosed to unauthorized users (Li, et al., 2010). In the IoT, this concern comes from various approaches of privacy such as device itself, device location, sensor metadata, and sensor observations. On the other hand, security means protecting information and information systems from unauthorized access, use, disruption, modification, recording or destruction (Ralph, 1990). By defining IoT as a *data-centric* application, enough level of security and privacy for data integrity and confidentiality are highly required.

Methods and Data

1. Architecture

1.1 Network Architecture

For the decentralized environment such as the IoT, resource discovery is always an issue. In our case, each device has a tiny web service, which allows users to directly connect to. However, users still need to know the service's Internet location (i. e., URL).

In order to address the resource discovery issue, we propose the sensor registry service (SRS). The SRS is similar to search engines and catalog services (Open Geospatial Consortium, 2010), which stores the metadata of web services and allows users to search services with criteria on metadata.

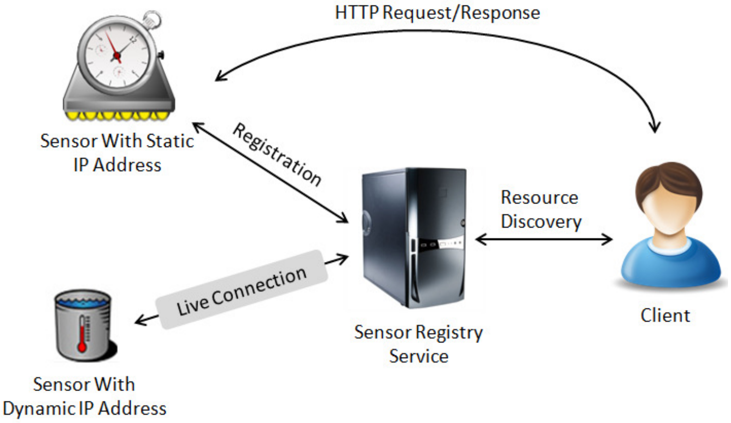


Figure 1- Network architecture diagram

The overall resource discovery process is shown in Figure 1. We can simply develop the strategies existing in computer security on the SRS, because it has enough computational resources. This attempt can preserve sufficient level of security and privacy for the dynamic IP sensors, as those sensors interact with only one client (SRS) in reality. Therefore, we should emphasize the security and privacy strategies on the static IP sensors that are communicating independently to the whole Internet nodes.

1.2 Device Architecture

Figure 2 depicts the architecture of an IoT device including *communication layer*, *service layer*, and *sensor layer*. In this research, we equip the communication layer with *Transmission Control Protocol* (*TCP*) in packet transportation, *Transport Layer Security* (TLS), and *Secure Socket Layer* (SSL) in session management. In addition, the service layer handles the business logic of the device web server. This layer consists of three modules: *request validator unit*, *response engine*, and *sensor data repository*. When a message is delivered to a device, the request validator unit processes its content. Then, the response engine prepares the required content from the permanent memory (e.g., a predefined text file like sensor metadata), or from the sensor data repository (e.g., sensor readings), and forwards it to the communication layer.

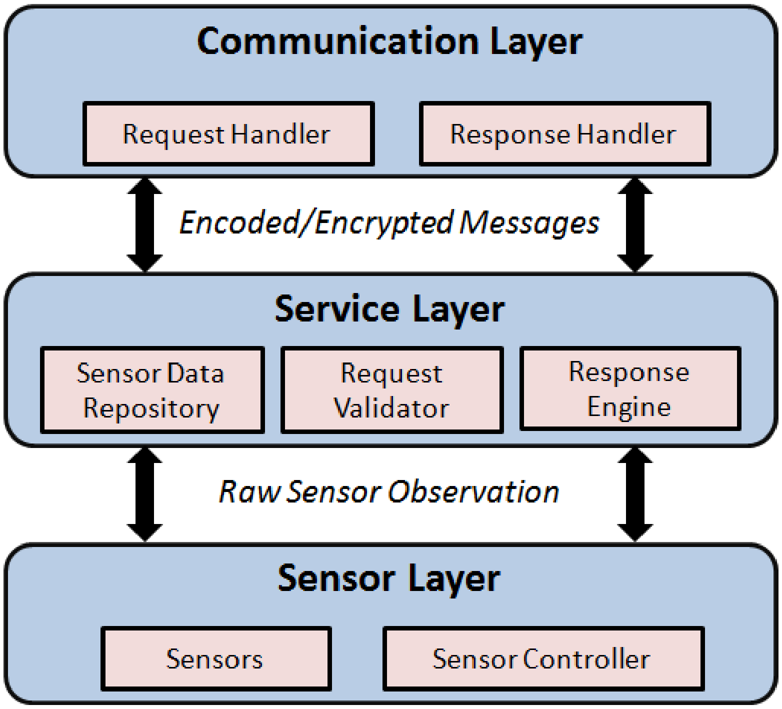


Figure 2- Device architecture diagram

2. Implementation

2.1 Development Platform

In this project, we choose a microcontroller as a development platform, named Netduino Plus (Figure 3). The board features a 32-bit Atmel microcontroller with 48 MHz speed, 28 Kbytes main memory (i.e., RAM), and 64 Kbytes code storage. In this case, Netduino Plus belongs to the class-1 device category.

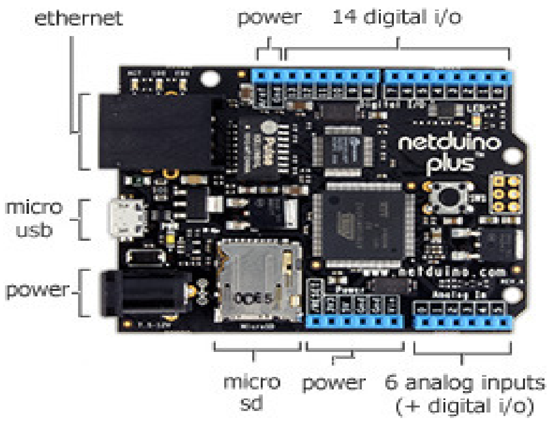


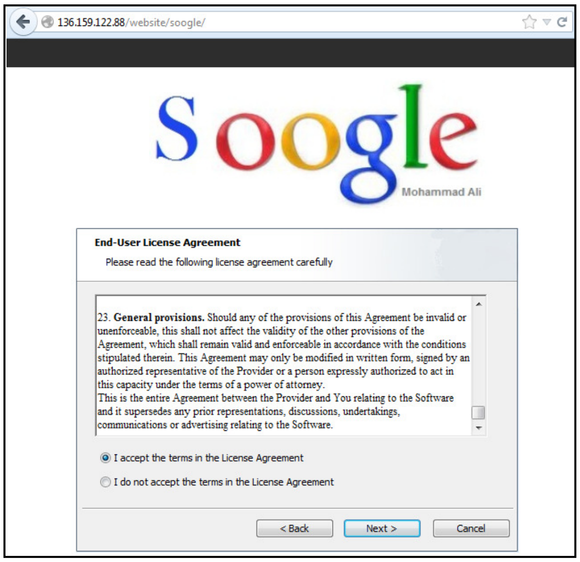
Figure 3- Netduino Plus (http://www.netduino.com/)

2.2 Privacy Concerns and Strategies

As we said, the dynamic IP sensors are kept safe since the sensor registry service acts as a gateway that validates the requests. Furthermore, for the static IP sensors, their owners are able to restrict the sensor service advertisement through the sensor registry service.

To enhance location privacy, Duckham and Kulik (2006) listed four general methods: *regulatory strategies*, *privacy policies*, *anonymity* and *obfuscation*. Here, we categorize them into two groups and add one more method introduced by Ferraiolo et al. (1995).

Non-computational methods originally look at the privacy like a law and agreement. By using *regulatory strategies*, we should ask government to interact with IoT researchers to define several rules on the misuse of personal information and devices. Furthermore, IoT developers should generate trust-based agreements between device owner and whoever is connecting to the IoT device (Figure 4).



End-User License Agreement

Please read the following license agreement carefully

23. **General provisions.** Should any of the provisions of this Agreement be invalid or unenforceable, this shall not affect the validity of the other provisions of the Agreement, which shall remain valid and enforceable in accordance with the conditions stipulated therein. This Agreement may only be modified in written form, signed by an authorized representative of the Provider or a person expressly authorized to act in this capacity under the terms of a power of attorney.
This is the entire Agreement between the Provider and You relating to the Software and it supersedes any prior representations, discussions, undertakings, communications or advertising relating to the Software.

● I accept the terms in the License Agreement

○ I do not accept the terms in the License Agreement

Figure 4- Trust-based agreement on sensor registry service

On the other hand, computational methods have been already utilized in computer information privacy (Bakken, et al., 2004). *Anonymity* is one of them that uses a pseudonym and creates ambiguity by grouping with other attributes. In our implementation, we considered anonymity for observed phenomena and unit of measurement. Although it is a good way to fool the attackers, this approach does not perform well in some cases that the number of attributes are not enough. However, *obfuscation* reduces data quality by considering uncertainty. In our system, we apply obfuscation technique for the latitude and longitude of the sensor's location. In addition to these techniques, we also enable the device service with *Role-Based Access Control* (*RBAC*).

One of the most challenging privacy concerns is *distance bounding* which restricts clients located in a specific region. To implement this, we save a table on the Netduino memory containing a mapping between IP address ranges to country names. Thus, whenever a user connects to the device, the user's IP address, and consequently country of that IP are checked. If an owner requires higher spatial resolution (e.g., city), we should replace the Netduino Plus with a hardware providing more memory capacity, or we should place a gateway in between to determine the route (Sachin, et al., 2004) . Generally, we can achieve our goal using RBAC method, even by much higher resolution like room, floor, and building access.

2.3 Security Strategies on the service layer

We enable the service layer to authenticate the clients in the request validator unit. Although the class-1 devices are not capable to contain a database server, we simply record the user information (e.g. username, password, and access level) on the micro SD. In our system, we consider three access roles: 1 (admin), 2 (authorized user), and 3 (unknown user). When a user signs up on a device, his/her role is set to 3 by default till the device owner validates the user as an authorized user.

Apart from RBAC strategy, the noises and eavesdropping hackers can still deduce endanger the data integrity. To prevent from these attacks, we apply digital signature mechanism using a famous *public key encryption* called *RSA* (Rivest, et al., 1978). The RSA algorithm involves three steps: *key generation*, *encryption* and *decryption*. Since the RSA explanation is out of the scope of this paper, we only talks about a simple implementation of RSA (i.e., SRSA) on our Netduino Plus. For the public and private key generation step, we randomly choose 2 prime numbers from a predefined set {2, 3, 5, ..., 97}. Then, we upload the private key on the sensor registry service for our credible users. Before sending critical data to the user, we encrypt the data with the public key. On the other hand, the user can simply use the sensor registry service to decrypt the message. In spite of data integrity by SRSA, data privacy is also maintained.

Results

To evaluate the performance of our implementation, the following experiments were accomplished:

- *Privacy Quantification*: There is not yet a standard way to quantify privacy as Krumm (2009) claimed. Since location can be specified as a single coordinate, one way to measure location privacy is by how much an attacker might know about this coordinate. In our system, the real sensor's location was 51.054, -144.066, respectively denoting latitude and longitude. However, the location that we showed to the client was between 50.054 to 52.054 for latitude and -143.066 to -145.066 for longitude.

- *Security Assessment*: To evaluate the security of the tiny web service, we simulated a common attack, namely *Denial-of-Service* (i.e., *DoS*). This threat is an attempt to make a machine or network resource slow or unavailable. Therefore, we implemented DoS in the below scenarios:

  a. DoS uses bandwidth: we simulated it by sending frequent packets (every 300 ms) to the tiny web service. The device worked properly since appropriate strategies have already been considered in the light-weight stack of the network card of Netduino Plus.

  b. DoS uses memory: similarly, we implemented this attack by sending a large packet (80 KB) to the device. Fortunately, we could pull down the server because its main memory was only 25 KB. To overcome this attack in the future, Netduino plus was tasked to read the first byte (content size) of the request before reading the whole content.

  c. DoS uses disk space: if a user frequently registers into the tiny web service, the permanent memory is quickly occupied. To prevent this attack, we restricted the user registrations by recording the connected IP.

Figure 5 shows the performance of the secure web service on Netduino Plus. The left-side one is a simple web service generated by the existing C# libraries, and the middle one is a tiny web service which is developed from scratch without any specific C# libraries. Based on this chart, the security and privacy mechanisms only occupied around 14 KB (21%) of the code storage which demonstrates the code efficiency.

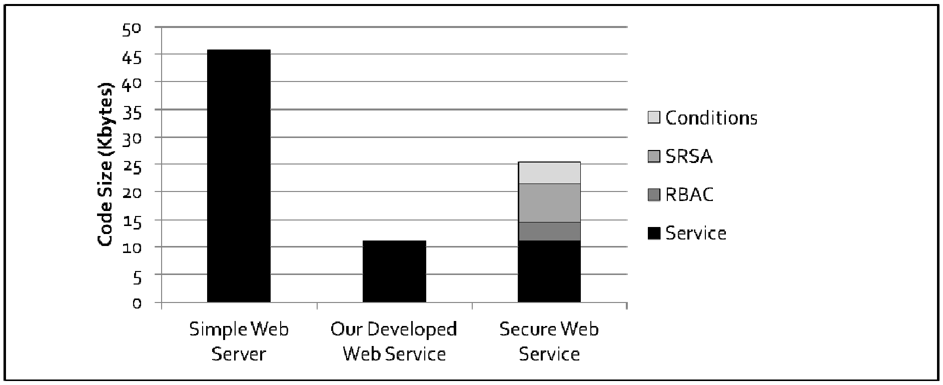


Figure 5- Developed web servers on Netduino PlusConclusions

Conclustions

In this research, we efficiently implemented existing security and privacy mechanisms on a class-1 IoT device. As a result, we presented possible secure connection and favorable privacy for the IoT objects, so we can encourage more people to integrate their devices to the IoT. However, we also observed some potential issues on our system.

One immediate issue arises from the nature of the IoT objects that should be connected to the Internet wirelessly. Since the wireless networking shares a physical medium to transfer data, a great number of noises may cause packet transmissions.

Moreover, one of the most challenging issues points to power supply. In this paper, we focused on the IoT devices that have unlimited power resource. This assumption really depends on the use cases and the different deployment environments. However, some sensor nodes will be battery-operated, so energy is perhaps the greatest constraint for the IoT devices.

As we mentioned, there are several remaining issues in the proposed system. Therefore, one of our future directions is to explore the potential solutions to address these issues.

References

Atzori L., Antonio I., and Giacomo M. "The internet of things: A survey." *Computer Networks* 54.15 (2010): 2787-2805.

Bakken, David E., et al. "Data obfuscation: anonymity and desensitization of usable data sets." *Security & Privacy, IEEE* 2.6 (2004): 34-41.

Bormann, C., Castellani, A.P., Shelby, Z, 2012. "CoAP: An Application Protocol for Billions of Tiny Internet Nodes", *IEEE Internet Computing*, Volume: 16 , Issue: 2, Page(s): 62-67.

Duckham, M. and L. Kulik, "Location privacy and location-aware computing," in *Dynamic & Mobile GIS: Investigating Change in Space and Time*, J. Drummond, et al., Editors. 2006, CRC Press: Boca Raton, FL USA. p. 34-51.

Ferraiolo, David, Janet Cugini, and D. Richard Kuhn. "Role-based access control (RBAC): Features and motivations." *Proceedings of 11th Annual Computer Security Application Conference*. sn, 1995.

Krumm, John. "A survey of computational location privacy." *Personal and Ubiquitous Computing* 13.6 (2009): 391-399.

Li, Ming, Wenjing Lou, and Kui Ren. "Data security and privacy in wireless body area networks." *Wireless Communications, IEEE* 17.1 (2010): 51-58.

Open Geospatial Consortium, 2007, OpenGIS® Catalogue Services Specification, retrieved from: http://portal.opengeospatial.org/files/?artifact_id=20555.

Rivest, Ronald L., Adi Shamir, and Len Adleman. "A method for obtaining digital signatures and public-key cryptosystems." *Communications of the ACM* 21.2 (1978): 120-126.

Ralph C. Merkle, , A Certified Digital Signature, In Gilles Brassard, ed., *Advances in Cryptology – CRYPTO '89*, vol. 435 of Lecture Notes in Computer Science, pp. 218–238, Spring Verlag, 1990.

Sachin G., Krishnakumar A. S., and P. Krishnan. "Infrastructure-based location estimation in WLAN networks." *IEEE Wireless Communications and Networking Conference (WCNC 2004)*. Vol. 1. 2004.