International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 06 Issue: 03 | Mar 2019 www.irjet.net p-ISSN: 2395-0072

Enhancing Security Features for IOT Devices Connected to Cloud Services through IAM

Anandhavalli D[1], Gladstone Abraham R[2], Sugadev D[3]

[1]Assisstant Professor, Velammal College of Engineering & Technology, Madurai
[2&3]UG Students, Velammal College of Engineering & Technology, Madurai

---

Abstract - *In the World of Internet of things, all tasks are automated and are integrated together with the help of some hosted Cloud Application. Lots of devices are getting connected to these Applications with different tasks, each with its own complexities and criticality. Not all the devices getting connected to the Application can be allowed to Access the resources of the Cloud Application. Internet being vast with lots of anonymous users, leads the security of the Application in jeopardy. This leads us to a position where, we have to authenticate every device's identity and authorize the access to its resources. It's very complicated to handle identity of every single device connecting to the application, there are lots of existing standards used with validating users can help us with it. The solution I am proposing is with OAuth2.0(Open Authorization) standard has a grant type named client credentials, which authenticates the device or any component with a Client Id, Client Secret, X509 Certificates and JWT(Java script object notation Web Token). In addition to that, the further authorization of the devices which requests services can be permitted based on Permissions and Policies which will be defined in a Simple IAM implementation.*

Key Words: Cloud Computing, Security, OAuth, Certification, IAM

1. INTRODUCTION

Under IoT, one of the root problems, in the context of cyber security assurance, is the lack of a rigorous notion of "Identity" in the Internet-of-Things (IDoT). In traditional systems and networks, multi-factor authentication is often used to define and recognize the "Identity" of a user (IDoU). Typically, three categories of information are involved. They are knowledge, procession, and inherence, which corresponds to the logics, built in the device, the resources allocated, and context in which the device is used.

IoT devices are primarily deployed in most commonly accessible public locations that lead to a greater security threat for IoT. However multi-factor authentication approach is much more complex and challenging. This is due to the new difficulties and challenges in defining and composing identity for IoT objects. In the following sections, we will first analyze the different information categories that can possibly serve as identifiers to composite identity for IoT objects and the other complex issues, when managing this information in the IoT network.

Leveraging the ideas from the "Identity" of a user (IDoU) in traditional systems and network, to be used as the information stack for "Identity" in the IoT (IDoT). In this information stack, there are four categories: inheritance, association, knowledge and context. The first information category in the stack is the "inheritance". Just like the biometrics identifiers (such as fingerprints and retina) of human, researchers are exploring similar type of information that is inherited from the IoT object hardware. The result is the PUF (physical unclonable function), which is defined as a physical entity that is embodied in a physical structure and is easy to evaluate, but hard to predict even for an attacker with physical access, or practically impossible to duplicate even given the exact manufacturing process that produced it.

Very often, it depends on the uniqueness of their physical microstructure and manufacturing process. A typical example is the Silicon PUF that is embedded into an integrated circuit. When the PUF is queried with a challenge or physical stimulus, it will return an unpredictable (but repeatable) response that depends on both stimulus and the unique object-specific physical characteristics of the object containing the PUF.

These "inheritance" information categories are very attractive to aid the definition and construction of IDoT. However, as expected, it is not as flexible as other information categories because it depends on the chip/hardware manufacturers. Furthermore, since PUF can be very noisy, precautions will be needed to ensure that the expected requirements for the function can be achieved. Currently, it is only used in applications with high security requirements.

The second information category in the stack is the "association". Unlike the "procession" information category for IDoU, it is not easy for an IoT object to process something external such as hardware token. However, under some specific situations or for some specific IoT objects such as personal wearables, it is common for the IoT objects to be associated (or linked) to a given personal gateway such as smart phone so that data will only be sent to the data cloud store through the predefined smart phone.

The third information category in the stack is the "knowledge". Similar to the second information category, the kind and amount of information that the IoT object can know is limited when compared to the case of IDoU. One typical

example of this information type is the IMEI (International Mobile Equipment Identity) of the mobile phone. But changing IMEI of a mobile phone is not as trivial as changing the password, in short if the owner of a given IoT network claims to change the IMEI of all the IoT objects that he/she deployed. Recently, one new research direction that people are investigating under this information category is to use the historical sensed data that a given IoT object has captured to define/construct its dynamic "Identity". However, this is still in the early stage of research.

The last information category in the stack is the "context". Unlike in the situation of IDoU where this information category is not used so often, this category attracts a lot of attention in IoT security. Normally, IoT sensors are deployed in groups that are related to each other. By studying the monitored behavior profile of different members within the same group and comparing it against the expected behavior profile, certain aspects of IDoT can be derived. Note that unlike the first three categories that come from the same IoT object, this information category is likely to derive from multiple inter-related IoT objects. The precision and quality of information in this category is relative lower than the other three information categories.

2. RELATED WORK

From the last section, it is clear that using the proposed information stack to define IDoT is indeed a new challenge, as compared to that for IDoU [1]. Due to the limited information availability in the "association" and "knowledge" categories, together with the inflexibility of the category "inheritance" and the imprecision of the category "context", risk-based authentication using multi-factors would definitely be the preferred option. And the category "context" will likely be the information target for IDoT researchers to explore. On top of the challenges to use multi-factors from the proposed information stack to define and construct IDoT, there are at least two additional issues in IoT that further complicates the management of IDoT [2].

The first issue is related to the ownership and user identity relationship of an IoT object. At any time t, every IoT object should have an device owner, but might have one or more users authorized by the device owner [5]. The relationship among the IoT object, owner, and users might also change with respect to time in its lifecycle. In addition to it, each IoT object might capture or sense from one or more data sources simultaneously. All these complicate the IDoT for authentication and other subsequent processes, including authorization and governance, in particular when the upper information categories such as "context" are used to define IDoT [3].

The second issue is related to the management of identifies and namespace of IoT objects. On the Internet, each resource has an URI (Uniform Resource Identifier). There is also DNS (Domain Name System) that maps URI to its current resource IP address; and this DNS is managed by the organization Internet Assigned Numbers Authority (IANA). With this namespace and identifier mapping framework, the dynamics of identifiers such as IP address of an URI can be hidden and communication between URIs becomes much easier [4].

However, in the IoT space, due to the wide variety of already existing mapping solutions from different manufacturers, defining this kind of unified identity framework will not be easy, at least not in the near future. Obviously, this will have negative impacts on IDoT when the information category "context" is used. It also affects the practicability of edge computing on IoT security [8].

3. PROPOSED WORK

Security being the prior concern of any application, the highest priority in deployment phase is to deploy the security services. The application is deployed in a cloud server, with the Node JS server hosted with HTTPS implementation with a Root Central Authority Signed Certificate matching its sub domain name. The various steps involved in the deployment phase includes.,

1. Set up the Application Server
2. Planning and project setup
3. Setting up the IOT device.
4. Testing and Evaluation
5. Model Deployment

3.1. System Architecture

The OAuth2.0 Specification for Client Credentials, The devices are considered as a client to the authorization server. And the Authorization Server issues the access token for the devices and the Resource Server uses the access token to validate the devices with the Resource Owner.

The Authorization Server must be configured with the device's credentials like Client id and Secret, X509 certificates, JSON Web Token. So that, the device gets it's own identity when connecting to the application assigned to it.

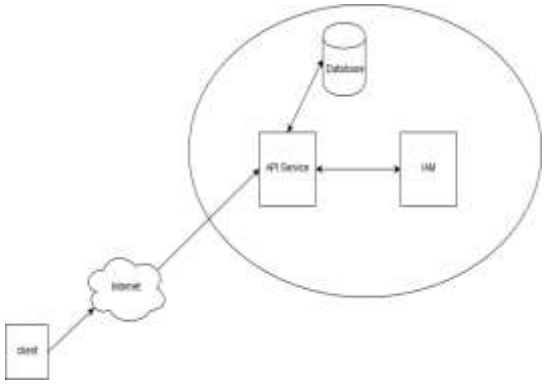

Fig:1 Base Architecture for the Device Credential

Device Roles are created to the devices based on the different categories of the tasks the devices perform. And then the roles are mapped to the devices such the permissions and policies can be formulated based on the need. The Policies defined here helps, to set different authorization levels for the roles created here. The Resource Server also acts as a client to the Authorization Server, so for accessing specific resources a client, the role mapped to the resource must be mapped with the client requesting it too.

The Resource Server must also be configured with the credentials such that the trust with the Authentication Server.

```
{
    "realm": "TesterRealm",
    "auth-server-url": "http://139.59.66.151:8080/auth",
    "ssl-required": "external",
    "resource": "ApiService",
    "verify-token-audience": true,
    "credentials": {
      "secret": "6b9d4964-f1fb-4f18-a996-6f6fd115f789"
    },
    "confidential-port": 0,
    "policy-enforcer": {}
  }
keycloak.json (END)
```

For Proof of Concept purpose, Client Id and Client Secret is used. Policy Enforcers can be implemented based on the different groups of devices/users and Realms. The Inter Realm communications can also be handled. This can also support the Single Sign On option if the Authentication Server handles more than one Resource Servers, then the same access token can be accessed across the Entire Application of Services.

The device requesting it must also be configured with its own credentials, which is used to get the Access token from the Authentication Server.

3.2. Sequence Diagram

A scenario is a specific sequence of action elements between the client and the API service and with the identity and access management. The access token is fetched from the IAM and then utilized in the further steps to access the authorized resources.

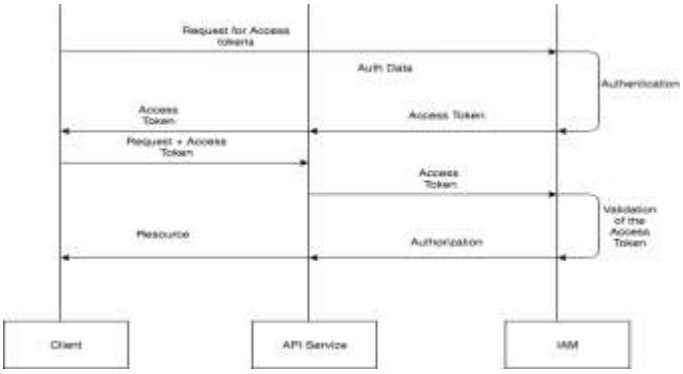

Fig:2 Sequence diagram for Client Authentication

The device connects to the Authorization Server with the device credentials and gets access token that is mapped with the roles of the device. The device requests the resources from the resource server with the access token in its header as mentioned in the OAuth Specification.

```
GET /resource HTTP/1.1
Host: server.example.com
Authorization: Bearer mF_9.B5f-4.1JqM
```

3.3. Setting Up the Identity and Access Management

After the completion of X509 Certificate Creation, Setting Up the Express JS Application Server and Setting Up the IOT Device, the main configuration part is done on the IAM since all the user management and the roles assigned to them are well organized in it. The roles define the levels of authorization. The resource owner usually defines the roles which can access resources.

1. Create a new Realm: A new realm for this project is to be created, such that all the components will be clients to this realm. The realm created has various configuration settings shown in fig 3. Since we are using the X509 certificates, login settings can be switched to SSL only.

2. Create the client entries for the Application Server and the IOT device. The credentials are configured based on the options chosen. The common authentication credentials include the JWT, X509 certificates, the client id and client secret. From the installation tab get the configuration file for both the clients, this helps in configuring the clients to the IAM server.

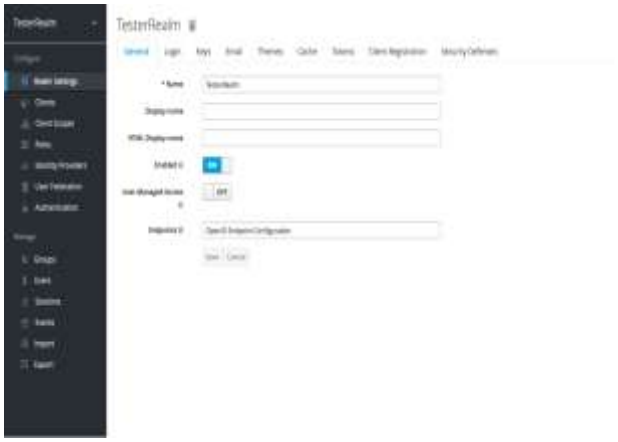

Fig:3 Create New realm

3. Creating a new Role for the client such that the client is has the roles and the various clients can be mapped to it using the service accounts tab.
4. Role Mapping is done in the IAM such that the clients can interact with each other.
5. Groups can be used to group users, when more than one client or users belong to the same role then the users can be a part of the group and the role can be mapped to the group created.
6. Client credentials used for authentication can be specified in the Credentials tab.

4. RESULTS

The Resource Server uses the access token to validate the device connecting to it, without the access token, resource server redirects to the authentication server to get a access token. When the token is mapped to role that is not authorized to access the server then, the resource is not allowed to be accessed.

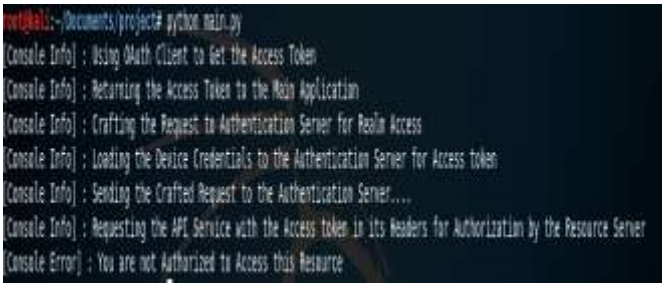

Fig:4 Validation (Sample for Rejection)

On Success it just returns the resource back. Since the device is authorized by the Resource Owner to access the resource.

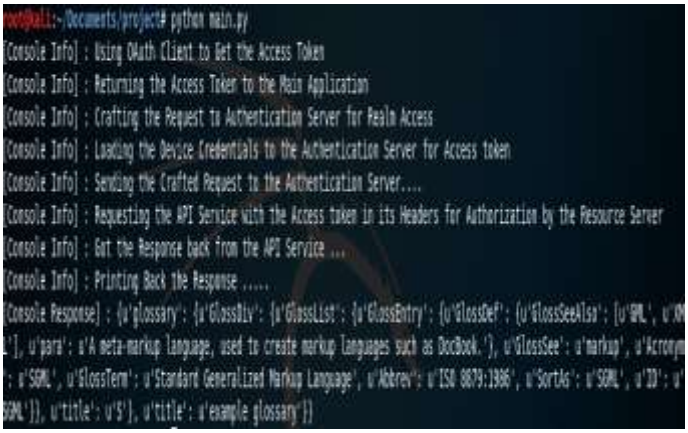

Fig:5 Validation (Sample for Acceptance)

5. Conclusion

The Security concerns with respect to the IOT in the cloud era, mainly focuses on the Identity management and Authorization levels handling for different resources. This paper helps to improve the levels of security by enhancing the grant type solving the above issue. By this flow, the device acts as user and each device is mapped to specific roles. Devices involving human automation are identified and mapped to various authorization levels of accessing the resources. In future, The Authentication made here can be further enhanced by additional levels of security. The sample model must be developed as an application with multiple sub domains and the inter communication between the components. Now the implemented model for authorization is Subject-Push model instead Resource-Pulling model must be implemented.

6. References

[1] Granjal, J., Monteiro, E., Silva, J.S.: Security for the internet of things: a survey of existing protocols and open research issues. IEEE Commun. Surv. Tutorials 17(3), 1294–1312 (2015)

[2] Zhao, K., Ge, L.: A survey on the internet of things security. In: Proceedings of Ninth IEEE International Conference on Computational Intelligence and Security (2013)

[3] Zhao, K., Ge, L.: A survey on the internet of things security. In: Proceedings of Ninth IEEE International Conference on Computational Intelligence and Security (2013)

[4] Deepak H. Sharmaa *, Dr. C. A. Dhoteb , Manish M. Poteyc: Identity and Access Management as Security-as-a-Service from Clouds. In: 7th International Conference on Communication, Computing and Virtualization (2016)

[5] I. Indua, P.M. Rubesh Ananda, Vidhyacharan Bhaskar :Identity and access management in cloud environment: Mechanisms and challenges In : Engineering Science and Technology, an International Journal 21 (2018)

[6] S. Eludiora, A user identity management protocol for cloud computingparadigm, Int. J. Commun. Netw. Syst. Sci. 4 (2011) 152–163,

[7] S. Subashini, V. Kavitha, A survey on security issues in service delivery modelsof cloud computing, J. Netw. Comput. Appl. 34 (2011)

[8] Shuai Zhang, Shufen Zhang, Xuebin Chen, XiuzhenHuo, "Cloud Computing Research and Development Trend", 2010 Second InternationalConference on Future Networks, IEEE 2010

[9] Deepak Sharma, Dr. C A. Dhote, Manish Potey, "Security-as-a-Service from Clouds: A comprehensive Analysis", IJCA Volume 67-Number 3, April 2013

[10] ChunduruAnilkumar, Sumathy:Security strategies for cloud identity management In : International Journal of Engineering and Technology 7(2018)