Tackling the prominent threats in Cloud Computing using Fog Computing in IoT

PRASHANTH PEDDABBU (UFID - 29989728), YUVARAJ SRIPATHI (UFID - 14677313) - Group 2

I. Abstract

Internet of Things (IoT) brings significant challenges with respect to data handling for the data generated by the dense network of connected devices. Many of the connected devices are resource-constrained sensors and actuators and the others contain reasonable computation abilities which communicate the data to the Cloud over the internet for enhanced processing and efficient storage purposes. There exists a set of prominent threats to Cloud computing paradigm and the usage of Cloud to handle IoT data make the Cloud module more taxing. On considering the ramifications of prominent threats to Cloud computing along with the overheads of dealing with IoT data, we introduce an approach to mitigate the same by using Fog computing. Our proposal makes use of a layered architecture for easing the communication between end devices and Cloud module and also provide solutions for tackling the prominent threats in Cloud computing using Fog computing in IoT environment.

II. Introduction

Cloud computing paradigm attributes to the computing model for providing on-demand service such as access to resources, storage, servers, etc. The bigger picture proves that the exploitation of Cloud services has made things much easier especially in the field of IoT. The IoT environment setup that uses Cloud services faces the threats and vulnerabilities that exist in the cloud computing system. Large-scale Distributed Denial of Service (DDoS) which are executed by the set of compromised IoT end devices (convert them to Botnets) whose credentials are not protected [1] highlights the importance to be shown towards securing them. The credentials related data of the end devices in IoT setup are present at Cloud and if an attacker gains access to such crucial data then, the possibility of DDoS attack only increases. If we want to continue to make use of Cloud services for efficient data handling of IoT-related data over the internet, an intermediate layer between the Cloud and the end devices in IoT is required to secure the IoT related data as well as the end devices in the IoT network - “Fog Computing”.

III. Related Work

As the research continues to optimize the efficacy of using Cloud computing in the implementation of IoT in a large-scale environment, the challenges associated with it is also increasing. In the first place, the use of cloud computing has its own threats and vulnerabilities [2] then there are other security threats and limitations in using Cloud computing for IoT infrastructure [3].

Fog computing gained interest in the network community, mainly because of its features that enable to tackle the challenges that prevails in the IoT infrastructure using Cloud computing. Research is proving the effectiveness in application of Fog computing in various scenarios like in the field of mobility of the connected devices [4], utilizing its applications in the field of automation [5], optimizing the performance of protocols [6], a better heterogeneous connectivity in the network [7]. Fog computing is also being applied in real life scenarios in IoT like – Traffic Management, smart grid network and health care [8]–[10] and more.

IV. Prominent threats in Cloud Computing

Cloud computing is the type of Computing which depends on sharing computing resources instead of having local servers or personal devices to handle applications by using the internet to access any software which is actually running on someone else’s data center/hardware.

The Cloud services are categorized into following three ways,

- **IaaS (Infrastructure as a Service)**
  Provides virtualized computing resources for computation and storage purposes. Example - Amazon Web Service, Windows Azure.

- **SaaS (Software as a Service)**
  Provides a software distribution model for third party vendors to host their private applications and make them available to interested users over the internet. Example - Oracle, SAP.

- **PaaS (Platform as a Service)**
  Provides software and hardware tools as a service to users for the development of Web-based applications. Example - Google App Engine, Heroku.

Also, the deployment of cloud services to users is again classified into the following categories.

- **Public Cloud**
  Open for public use and effective for application hosting

for a large-scale set of users.

- **Private Cloud**
  Provides services only for a closed set of users like for an organization. The services are crafted as per the organizations' requirement.

- **Community Cloud**
  Used by a group of organizations working on a common goal.

- **Hybrid Cloud**
  A combination of Public and Private Cloud deployments considering the set of users.

Based on the Cloud services and its deployment strategies, there are 7 prominent threats that are hindering the complete switching to Cloud from local service [11]. They are,

1) Abuse and Nefarious use of Cloud Computing
2) Insecure Interfaces and APIs
3) Malicious Insiders
4) Shared Technology Issues
5) Data Loss or Leakage
6) Accounts or service Hijacking
7) Unknown Risk Profile

Since the amount of data generated in a current IoT environment is huge and there is a dire requirement for accessing high computation ability services and storage centers, the IoT devices focus towards the Cloud solutions for the same.

Though there are some solutions to the above mentioned prominent threats to Cloud computing, which are being implemented [12], these solutions do not consider the overheads that occur from using Cloud computing in IoT paradigm. The overheads are mainly in the form of data congestion at Cloud-level due to the number of connected devices which generate data at a very high rate, security issues in the maintenance of the profiles of each device that communicates with the Cloud and high latency in data processing for latency sensitive IoT applications.

The set of prominent threats is applicable even for the usage of Cloud computing in IoT environment. Each of the threats causes further repercussions of issues and vulnerabilities on both ends, i.e. Cloud and end devices in IoT.

Let us assume that the IoT scenario where there are many heterogeneous sets of devices (like sensors, actuators, smart phone, smart vehicle, etc.) that communicates with its corresponding Cloud module for transmitting the generated data that needs to be processed at Cloud and respond with corresponding useful information back to the end devices. Also, the necessary data needs to be stored on Cloud in a secure manner for future access.

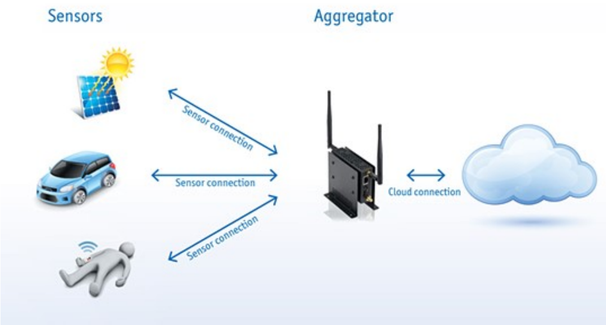

Figure 1. IoT - End devices communicating with Cloud module via the internet

Applicability of the prominent threats in Cloud computing in IoT environment,

1) **Abuse and Nefarious use of Cloud Computing**
   On a public network, there exists a vulnerability of an attacker gaining access to crucial information that attributes to end devices at Cloud-level or in the communication link (devices like sensors and actuators does not have enough resources to make use of advanced cryptographic methodologies for securing the transmission link with the Cloud). If the attacker is successful in gaining this information, the attacker can make use of the same and take control over the corresponding end device in the Edge module and thereby can make use of the compromised device's resources. This is basically the creation of Botnets that work for the attacker. A simple malware upload to the Cloud module by an attacker can infiltrate its way to these end devices so that the attacker can exploit the connected devices' resources for malicious activities.

2) **Insecure Interfaces and APIs**
   As mentioned in the environment setup, there is a heterogeneous set of end devices that is connected and communicating with Cloud to work towards a common goal. It is possible that each of the devices is using its own proprietary programming interface and thereby there is a requirement of providing a layer of abstraction at either end to deal with the diverse set of interfaces. This layer adds to the latency while processing the data and also puts forward the issue of decision making with respect to programming policies to be followed. These decisions are ambiguous since the decision needs to consider the policies of several diverse interfaces supported by the system. This causes in dissimilar ways of handling the data communicated with each of the heterogeneous set of end devices.

3) **Malicious Insiders**
   There is no transparency as to who is accessing and

implementing the policies at Cloud level. So, it is possible that there can be an insider attack due to access policy changes intentionally or unintentionally which is a vulnerability of the IoT related data stored on the Cloud (crucial data like Credit card details, health record details, etc. is exposed) and for the very connectivity of the end devices with the Cloud (Possible Denial of Service due to policy changes). Since not all end devices are capable of securing the data that is sent to Cloud, the data can be compromised.

4) **Shared Technology Issues**
The underlying components that make the cloud resources (like CPU caches, GPU, etc.) are not designed to provide for strong isolation properties for a multi-tenant architecture. So, a third party hypervisor acts as a mediator to facilitate the same. But these are not cent percent reliable as there are vulnerabilities which provide inappropriate levels of control for higher level software over the underlying components. Let us consider the issue of handling race conditions while computing with the data generated from end devices, it is possible that the latency sensitive data (related to accident prevention from a Road side Unit) is not given priority due to lack of resource for that particular process (resource might be scheduled to some other trivial process due to the specified vulnerability).

5) **Data Loss or Leakage**
The communication channel between the end devices and Cloud module is over the internet. Since there are many devices that generate and transmit a lot of data to Cloud at a given time, there is a possibility network congestion due to high data traffic. Due to this, the probability of packets being dropped is high. It causes major impact from application perspective when there is packet loss of data related to sensitive application information (like temperature warnings, abrupt rate fluctuations in smart grid, etc.).

Not all end devices encrypt their respective data while transmitting to Cloud due to resource constraints (like sensors and actuators), so there is a vulnerability of attackers performing Man-in-the-Middle and Eavesdropping techniques. On gaining knowledge of crucial information, cascading failures/attacks can be triggered by the attackers (Replay attack, DoS, etc.).

Also, there exists a vulnerability of losing IoT related the data which is stored in the Cloud (when Cloud infrastructure resources are compromised).

6) **Accounts or service Hijacking**
With the usage of Cloud, attacks like Phishing and other fraudulent activities are known vulnerabilities to the end devices and users. If the attackers are able to elicit the required information by such methods, then it is possible that they take control of the end devices in its operation or they have complete access to the device's data and mimic the same with some modifications and send to the Cloud (Man-in-the-Middle). The corresponding computation on such modified data can cause serious damages in the form of output (like causing power line failure in smart grid due to the wrong rate alteration detection, changing the traffic signals based on malicious inputs, etc.).

7) **Unknown Risk Profile**
The Cloud module implements security by obscurity. Since there are a many Cloud providers each with their own set of security rules and policies (with their own vulnerabilities) and heterogeneous set of end devices that communicate with Cloud, there exists an ambiguity while dealing with new threats and modeling policies for the same to prevent the same in the future across the Cloud. In this process, there will be many version changes, code updates, rule implementations and if the corresponding information is not synchronized with the end devices, then there is always a possibility of the existence of an unknown vulnerability (which might affect the end application output and operation of end devices).

V. Introduction to Fog Computing

In the next few years, there will be a humongous growth of "smart devices" which will communicate and share data at a large scale via the internet. Many of these smart things will be part of much larger systems which require heavy computation and storage capacity for their generated data. Since the data from Things is growing huge, moving this big data from network edge (end networking systems or devices) to network core (mainly cloud computing servers) and vice versa that is from the network core to the network edge, will naturally present a valid question of where should those resources be placed without compromising on security and privacy in the pile of internet of things.

Are there applications in the IoT Space that would require offloading some computation tasks onto other devices, but the mere fact of having the compute and storage resources, confined in a centralized data center (cloud systems) makes the overall platform unfeasible? Are there any vulnerabilities in cloud IoT space which are susceptible to attacks and a few of those notable threats can be mitigated using a different model? The answer to this question is 'Yes'. To address many of these questions, there is a need for the new computing model, which is termed as "Fog computing" [13].

Fog computing is an emerging research field that aims at providing satisfactory services to customer needs in between the "Cloud" and "Ground" space. In the recent times, the term "Fog" is labeled to depict the eventual evolution where

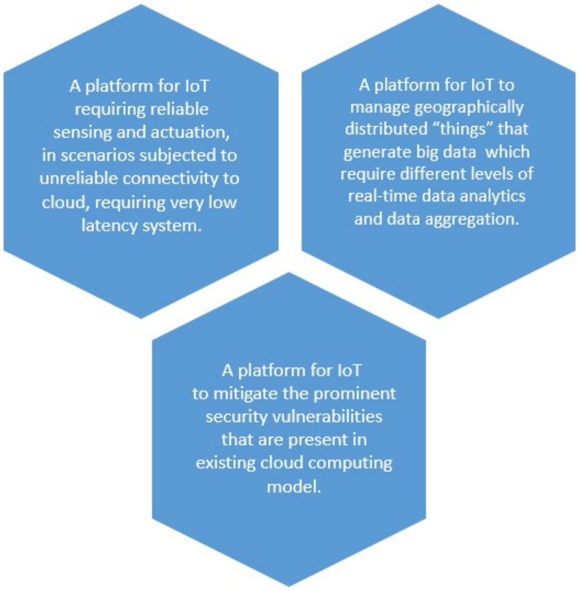
A platform for IoT requiring reliable sensing and actuation, in scenarios subjected to unreliable connectivity to cloud, requiring very low latency system.
A platform for IoT to manage geographically distributed "things" that generate big data which require different levels of real-time data analytics and data aggregation.
A platform for IoT to mitigate the prominent security vulnerabilities that are present in existing cloud computing model.

Figure 2. Some of the key requirements for designing and building and adaptable and scalable IoT platform

the cloud is migrating to the edge of the network in which routers themselves and other networking devices may act as a virtualization infrastructure for offering more scalable solutions in relatively less turnaround time when compared to the existing cloud infrastructures. Cisco's view of the Fog [14] depicted it as a new paradigm that executes generic application logic on resources throughout the network, including dedicated computing nodes or systems and routers.

Fog is composed of all the smart sensing/computing devices that are around us, tied together. Typically, it contains a small data center, which is located close to the things in IoT. However, there is an unavoidable "interplay between the Fog and the Cloud" [15] in the Internet of Things in coming years.

The concept of fog computing adds a whole new dimension to the IoT model for meeting the customers' needs such as reliable and high-speed connection, off-loading network core traffic since it is expected to move to the network edge, highly scalable system, easy for management and infrastructure reuse. Fog computing will support a wide range of IoT applications, device to device data sharing, vehicular systems, wearable cognitive assistance, smart surveillance applications, etc.

To emphasize its difference from cloud computing, placing intelligence in the network makes fog computing resources to perform low-latency processing near the edge while latency-tolerant, large-scope aggregation can still be efficiently performed on powerful resources in the core of the network. Data center resources may still be used with the fog computing, but they do not constitute the entire picture.

Since the presence of cloud computing is at the network core, a variety set of opportunities is created for security breaches while accessing the cloud infrastructure. These attacks have different aims starting from eavesdropping, to complete system failure. It is vital that these attacks are identified clearly and their mitigation techniques.

VI. KEY ATTRIBUTES OF FOG COMPUTING

- **Heterogeneity**
  Fog computing being a virtualized platform forming a layer between the core module and the actual set of connected devices, it provides for computing, storage and networking services for the other two layers. Also, the Fog layer acts as an abstraction layer for either side modules, thereby limiting the overheads of dealing with the heterogeneous set of connected devices, software and platform at the core module. The Fog module can take care of the providing an interface that acts as a wrapper for each of the different services and APIs provides the application developers a conducive environment to deploy their applications in the Core module and among the end devices.

- **Geographical distribution**
  Fog computing paradigm shifts the focus of core module's centralized architecture to more of a distributed one. This is targeted for deploying the applications that are more distributive and rely on data obtained from a much larger set of connected devices. Example - Video streaming service for the set of moving vehicles in each area.

- **Edge location**
  Much of the Fog components that are used for processing and transmitting data are placed in a closer proximity to the devices of interest compared to the resources in the core module. This provides for servicing latency sensitive applications within the same network. Example - gaming, augmented reality, etc.

- **Support for mobility**
  Fog computing provides for dynamic allocation and deallocation of near the edge resources as per the application and service requirement at that point of time. This allows the applications to support mobile devices also as long as the devices are under the coverage of the Fog module.

- **Real-time service**
  Fog computing based applications provide for real-time interactions unlike the batch processing feature of the core module. This gives a better service experience in terms of lesser latency and maximum usage of the resources available.

- **Large-scale networks**
  In order to monitor the sensors in a large scale network and process data generated from the same and make decisions in real time, Fog computing provides for the infrastructure to do such activities instead of depending on the unreliable (latency sensitive) core module. Example - Smart grid network computations to prevent cascading failure.

VII. Tackling the Cloud Computing threats using Fog Computing

The proposed set of solutions holds good in IoT environment that is implemented based on the below mentioned three layered architecture.

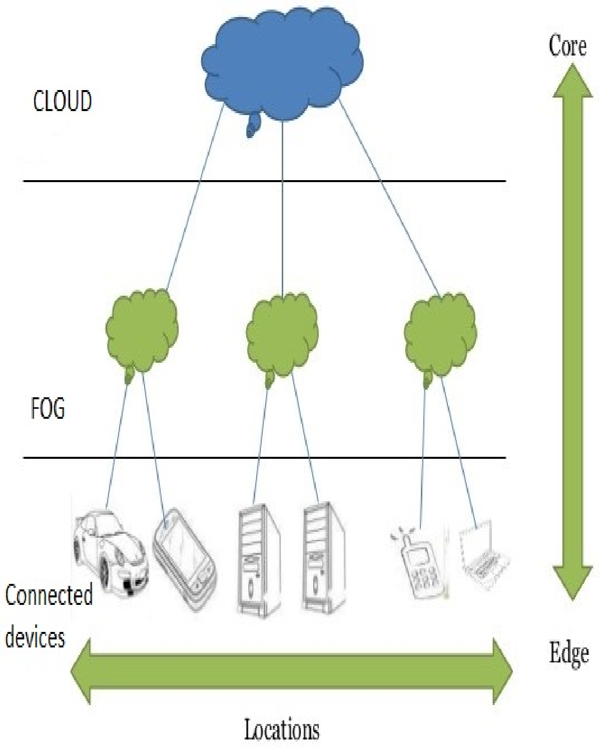

Figure 3. Layered architecture of IoT environment with Fog module

1) **Cloud module**
   Contains the cloud computing resources which the user/devices in the layer below are authorized to use. Also, known as the core module. These resources are meant mainly for efficient large-scale data storage and to perform resource heavy computations on a large set of data.

2) **Fog module**
   Contains dedicated systems that has reasonably good amount of storage capacity and computational abilities (Very less compared to cloud module). It can be any device like a Smart router, dedicated server, a Smart phone, Road Side Unit (RSU), etc.

3) **Edge module**
   Contains the network of physical devices (can also be set of mobile devices) that either generates useful data or it helps in transmitting the same to other devices. Devices can be set of sensors, actuator, RFID, camera, Smart Vehicle, etc. It is assumed that these sets of devices and their corresponding Fog nodes are secured by using a private network.

Tackling the threats using the Fog computing paradigm

1) **Abuse and Nefarious Use of Cloud Computing**
   If malicious users try to access the data to which they are not authorized in the cloud, Fog module should be able to detect such anomalous behavior and prevent the data being accessed. The basic step here is to mitigate the damage caused by stolen data from the cloud by reducing 'the actual value' of the data. For this, a system that detects the unusual behavior of the user and provides bogus/fake data, such that the malicious user does not get suspicious about the possible attack detection is introduced [16]. It is one of the efficient methodologies to secure the Cloud module, but it has a lot of overheads when the same is used for securing Cloud which has to communicate with a many heterogeneous end devices in IoT environment. This implementation can be moved down to the Fog module to filter out the attackers and provide them with the fake data until the attacker can be identified at the edge layer itself. The very access to the cloud level by the attacker is thereby prevented. There are two sub-modules required for such a defense as shown in Figure 4,

   - **User Behaviour Profiling**
     The system is able to detect the legitimate users who are accessing their respective data and services in the Cloud by performing a certain set of computations based on the users' past activities - user profiling [17]. This requires access to volumetric information corresponding to the user. The Fog nodes can access this information from the Cloud if it is not available at Fog level and can perform user activity comparisons at the edge level and try to detect the fraudulent users (Since all the traffic from all the physical devices in the IoT network to the Cloud are routed via the Fog nodes).

   - **Decoy System**
     There is a necessity for each user to provide fake data which basically serve as honeypots (like fake

information about passwords, bank details, medical record, etc.) which will be provided in the form of fake data, also known as decoy information to the malicious users who try to access data belonging to the actual user. The Fog nodes can download corresponding decoy data from the cloud whenever a new user/device wants to access the data in the Cloud and cache it for identifying the corresponding user/device. Once the user is found to be legitimate the decoy data can be deleted from the corresponding Fog node. The Fog nodes on monitoring access to such decoy information detects the occurrence of an attack.

Each of the decoy related information contains a keyed-Hash Message Authentication Code (HMAC) hidden in the header section [18]. The HMAC is computed over the file's contents using a key unique to each user. On computing the HMAC of the each of the file contents accessed by the user, it is compared with the decoy related HMAC data in the header of the same message. If both the HMAC matches, then, the user is not legitimate and corresponding alarms are issued. Access of decoy related data by a legitimate user can be detected by implementing a challenge-response based protocol to ensure the legitimacy of the user.

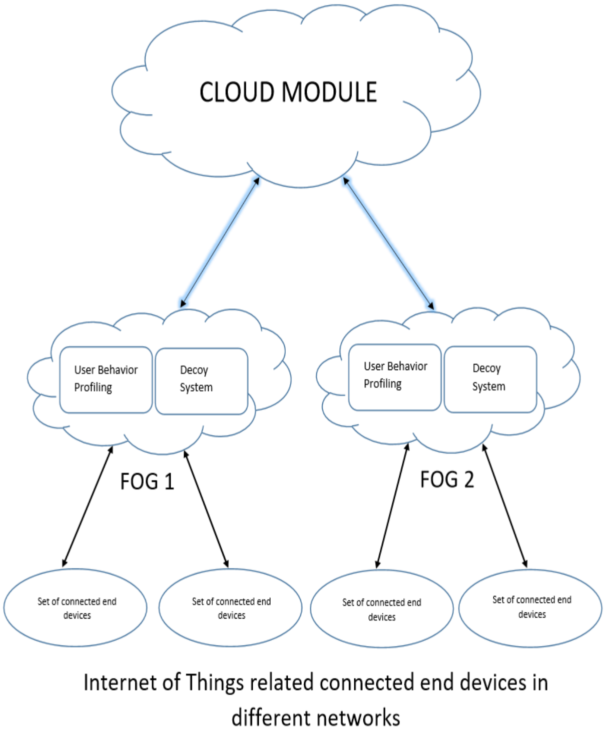
CLOUD MODULE
User Behavior Profiling
Decoy System
FOG 1
User Behavior Profiling
Decoy System
FOG 2
Set of connected end devices
Set of connected end devices
Set of connected end devices
Set of connected end devices
Internet of Things related connected end devices in different networks

Figure 4. Fog implementation for avoiding Abuse and Nefarious use of Cloud computing

The combination of successful results of search anomaly detection and access to decoy information, gives a strong proof of maleficence existence in the network (An insider attack in the private network). On off-loading these activities to Fog, quick decisions can be taken on attack detection and since user profile statistics and the decoy related information for each of the users is downloaded into the Fog nodes, the data access in Cloud by an attacker is mitigated. Since the Fog nodes are strategically placed and scale dynamically, depending on the number of cloud users/physical devices, the corresponding amount of user profile statistics and each user's decoy data needs to be cached in Fog nodes. Policy based decisions should also be implemented related to the time duration of these downloaded data that should persist in the cache of Fog nodes based on the credibility of each of the corresponding user profile.

2) **Insecure Interfaces and APIs**

As per our assumed architecture of the Fog module between the Cloud module and the set of connected end devices, all the communication to the Cloud happens via the Fog module and vice-versa. Only the Fog nodes that communicate with the Cloud module needs to oblige the different interfaces and APIs of the Cloud module. In the Fog module, we will assume that there will be two layers of abstraction for dealing with the data obtained from both the upper and bottom layers as depicted in Figure 5. The lower abstraction layer in the Fog module deals with providing an abstraction layer for formatting the data received from the set of heterogeneous connected end devices while the upper layer of abstraction deals with the formatting the data received from the Cloud channel.

By providing such layers of abstraction, the Fog module deals with the overheads of a heterogeneous set of APIs and interfaces, so that the application developers can focus more on handling the data on both sides. In order to showcase the flexibility of having such abstraction in the Fog module that is present between the Cloud module and the set of connected devices, an infrastructure is proposed [4] that consists of a set of APIs and event handlers that can be used by the application developers without being overwhelmed by the heterogeneous nature of the connected devices' interfaces and Cloud module's proprietary set of interfaces and mechanisms.

3) **Malicious insiders**

The risk of data compromise in the Cloud module due to the loopholes in the security policies is mitigated by the introduction of the Fog layer. The IoT related data are now sent over to the Cloud module, be it for storage or processing purpose, they are in an encrypted format. The encryption and the decryption of the data is taken care at the Fog level. Even if the malicious insiders gain access to these encrypted data, they are basically

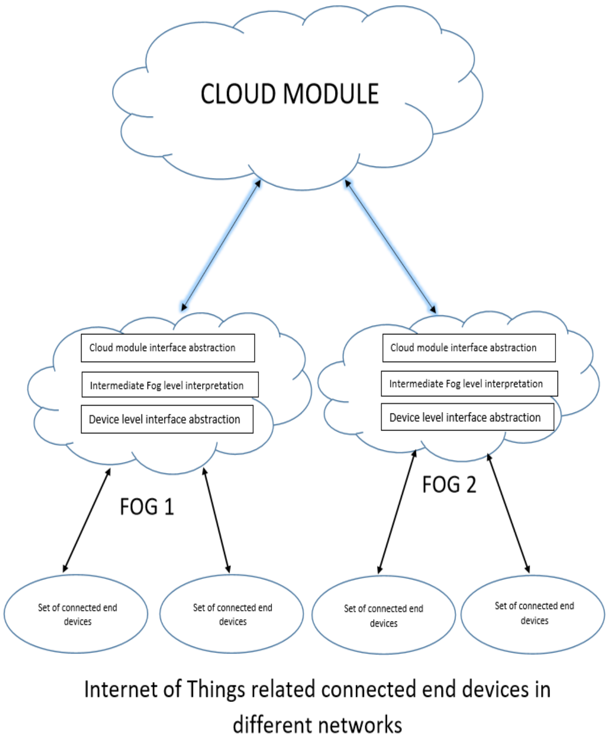
CLOUD MODULE
Cloud module interface abstraction
Intermediate Fog level interpretation
Device level interface abstraction
FOG 1
Cloud module interface abstraction
Intermediate Fog level interpretation
Device level interface abstraction
FOG 2
Set of connected end devices
Set of connected end devices
Set of connected end devices
Set of connected end devices
Internet of Things related connected end devices in different networks

Figure 5. Implementing abstraction layers for Cloud module and device level interfaces in the Fog layer

in the form of cipher text and hence is of no use to the attacker. The Fog modules (Fog nodes) possess reasonably a better set of resources, hence they can perform the operations of encryption and decryption of the data before sending the same to the Cloud module (the same could not be performed by the end devices due to the limitation of resources and lack of complex computational abilities).

The Fog module as per the architecture assumed, contains sufficient storage and computation capacity, thereby all the sensitive information can be contained within the Fog layer and the end devices and not letting them over to the Cloud. This greatly averts the sensitive data compromise (Example - Banking information, smart grid rate alteration, etc.).

4) **Shared Technology Issues**
The Fog module in the proposed architecture consists of more than one Fog node. Similar to the resource allocation in Cloud module, the resources are allocated in the Fog module (collaborative effect of many Fog nodes) depending upon the process and the set of data associated with it. Only the set of data and processes that cannot be handled at the Fog level is transferred to the Cloud module and only corresponding resources is required at the Cloud level.

All the sensitive data and data related to latency sensitive applications are handled at the Fog layer itself. This allows for better scheduling of services at the Cloud layer since delays in scheduling of the resources can be tolerated.

The Fog layer will sense the sudden bursts in the request for resources and Fog layer can request for similar scaled up resources in the Cloud layer in a proactive manner so that there would not be any occurrences of the bottleneck situation at the Cloud level for the IoT related data awaiting the resource allocation (This is a trivial case since Cloud modules contains enough resources to satiate the computing of the data transferred by the Fog module).

5) **Data Loss or Leakage**
As per the resources present in the Fog module in our assumed IoT environment, the Fog nodes have better computational abilities than that of the actual edge device like sensors or actuators. Strong policies can be implemented in these Fog nodes as to what type of data must be sent to the cloud and vice versa. Since we can make the Fog nodes as the decision makers, sensitive data obtained from the end devices (like bank details, accident prevention data, alarm data, etc.) can be computed in the Fog nodes itself and the resulting data can be used for further processing within the Fog and Edge module. For some of the processes, it is necessary to integrate the freshly generated data or processed results with the cloud. For this, many of the industrial standard cryptographic solutions can be implemented.

The overhead for the end devices (like sensors, actuators, RFID, etc.) for encrypting the data and sending the cipher to the Cloud is abolished because the Fog nodes do this operation on behalf of them. Since the Fog nodes contains set of rules and policies related to data that needs to be computed internally and data that needs to be sent to Cloud, only those data which are being sent to Cloud needs to be encrypted (Lower level layers need not contain any cryptography implementations since they are in a Private network). If the data is just to be stored on Cloud for later use, the data is encrypted by the Fog nodes and the encryption keys are in the possession only with the Fog nodes, the data stored in the cloud is just the blocks of cipher text stored in a logical manner. If the data is being sent from Fog to Cloud for further processing, then, a pre-agreed cryptographic technique is used for encrypting and decrypting data on both ends and the key exchanges for the same happens in a secure mode like using Diffie-Hellman key exchange [19]. Also, the Fog and Cloud modules can generate a different set of keys for every new session depending on the sensitivity of the data and environment to reduce the probability of key compromise (Man-in-the-middle, Eavesdropping, etc.).

6) **Accounts or service Hijacking**
By introducing the Fog layer, all the credentials and identities of the set of connected end devices are maintained within the Fog layer. Since these credentials are not let out to the internet or the Cloud module, it becomes almost impossible for an attacker to take control of the end devices, thereby, we are mitigating the probability of the end devices becoming a part of botnet network.

Better and sophisticated authentication methods can be used and also different authentication methods can be used on either side layers to the Fog module, thereby inducing authorization policies based on the user/device rather than the service. For example, for every new connection set up between a Fog level service and Cloud level service, a two-factor authentication [20] system or Public Key Infrastructure (PKI) [21] can be implemented and for the authentication of the end devices and the Fog module, an identity based authentication (due to the memory constrained end devices) mechanism like OAuth [22], Single Sign on (SSO) [23] can be implemented and the Fog nodes can deal with the maintenance of the identity token distribution for supporting communication among the end devices under the same Fog module.

7) **Unknown Risk Profile**
Dealing with Unknown Risk Profile as described in the threats to Cloud computing is one of the entities that exists for any system implementation. But we can always mitigate the effect of such a defect/attack on our IoT related data by implementing strong security policies and consistent updates of the software and hardware (Fog nodes and end devices). Prioritize the computation and storage of sensitive IoT data and device credentials at the Fog level and following latest standard of cryptographic standards with respect to the communication of data to the Cloud module.

All the end devices and the Fog nodes should contain the information about the architecture in which they are present like in which layer do they belong and which are its immediate Fog nodes and on any security revocations, update themselves accordingly. Also, the Cloud module should disclose as much of the information related to security policies to its users for better understanding and following of the same.

VIII. Conclusion

The rapid development in the field of Internet of Things and Cloud computing comes with the cost of a higher number of security threats. The challenge is to provide privacy for the shared personal data. This paper focuses on proposing an enhanced model of protecting the Cloud using the Fog computing paradigm. By exploiting the characteristics of the Fog computing and the architecture of current IoT network, it is possible to mitigate the vulnerabilities of the Cloud computing from the existing set of security threats.

References

[1] "http://dyn.com/blog/dyn-statement-on-10212016-ddos-attack."

[2] B. Grobauer, T. Walloschek, and E. Stocker, "Understanding cloud computing vulnerabilities," *IEEE Security Privacy*, vol. 9, no. 2, pp. 50–57, March 2011.

[3] T. Bhattasali, R. Chaki, and N. Chaki, "Secure and trusted cloud of things," in *2013 Annual IEEE India Conference (INDICON)*, Dec 2013, pp. 1–6.

[4] K. Hong, D. Lillethun, U. Ramachandran, B. Ottenwalder, and B. Kold, "Mobile fog: A programming model for large scale applications on the internet of things," in *Second ACM SIGCOMM workshop on Mobile cloud computing*, 2013.

[5] H. P. Breivold and K. Sandström, "Internet of things for industrial automation – challenges and technical solutions," in *2015 IEEE International Conference on Data Science and Data Intensive Systems*, Dec 2015, pp. 532–539.

[6] M. Slabicki and K. Grochla, "Performance evaluation of coap, snmp and netconf protocols in fog computing architecture," in *NOMS 2016 - 2016 IEEE/IFIP Network Operations and Management Symposium*, April 2016, pp. 1315–1319.

[7] S. Cirani, G. Ferrari, N. Iotti, and M. Picone, "The iot hub: a fog node for seamless management of heterogeneous connected smart objects," in *Sensing, Communication, and Networking - Workshops (SECON Workshops), 2015 12th Annual IEEE International Conference on*, June 2015, pp. 1–6.

[8] C. A. R. L. Brennand, F. D. da Cunha, G. Maia, E. Cerqueira, A. A. F. Loureiro, and L. A. Villas, "Fox: A traffic management system of computer-based vehicles fog," in *2016 IEEE Symposium on Computers and Communication (ISCC)*, June 2016, pp. 982–987.

[9] Y. Yan and W. Su, "A fog computing solution for advanced metering infrastructure," in *2016 IEEE/PES Transmission and Distribution Conference and Exposition (T D)*, May 2016, pp. 1–4.

[10] R. Craciunescu, A. Mihovska, M. Mihaylov, S. Kyriazakos, R. Prasad, and S. Halunga, "Implementation of fog computing for reliable e-health applications," in *2015 49th Asilomar Conference on Signals, Systems and Computers*, Nov 2015, pp. 459–463.

[11] M. Irfan, M. Usman, Y. Zhuang, and S. Fong, "A critical review of security threats in cloud computing," in *Computational and Business Intelligence (ISCBI), 2015 3rd International Symposium on*, Dec 2015, pp. 105–111.

[12] N. Sharma and M. S. Dr, "An analysis of prominent security threats in cloud computing," in *NIU Journal of Science and Technology, ISSN: 23479787, Vol2, Issue 1*, 2014.

[13] M. Yannuzzi, R. Milito, R. Serral-Gracià, D. Montero, and M. Nemirovsky, "Key ingredients in an iot recipe: Fog computing, cloud computing, and more fog computing," in *2014 IEEE 19th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)*, Dec 2014, pp. 325–329.

[14] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the internet of things," in *Cisco Systems Inc. Fog computing and its role in the internet of things. In Workshop on Mobile cloud Computing, MCC*, 2012.

[15] F. Bonomi, Milito, Rodolfo, P. Natarajan, Zhu, and Jiang, "Fog computing: A platform for internet of things and analytics," in *Big Data and Internet of Things: A Roadmap for Smart Environments*, 2014.

[16] S. J. Stolfo, M. B. Salem, and A. D. Keromytis, "Fog computing: Mitigating insider data theft attacks in the cloud," in *Security and Privacy Workshops (SPW), 2012 IEEE Symposium on*, May 2012, pp. 125–128.

[17] Sahil, S. Sood, S. Mehmi, and S. Dogra, "Artificial intelligence for designing user profiling system for cloud computing security: Experiment," in *2015 International Conference on Advances in Computer Engineering and Applications*, March 2015, pp. 51–58.

[18] H. E. Michail, A. P. Kakarountas, A. Milidonis, and C. E. Goutis, "Efficient implementation of the keyed-hash message authentication code (hmac) using the sha-1 hash function," in *Proceedings of the 2004 11th IEEE International Conference on Electronics, Circuits and Systems, 2004. ICECS 2004.*, Dec 2004, pp. 567–570.

[19] W. Diffie and M. Hellman, "New directions in cryptography," *IEEE Transactions on Information Theory*, vol. 22, no. 6, pp. 644–654, Nov 1976.

[20] F. Wang, Y. Zhang, Y. Xu, L. Wu, and B. Diao, "A dos-resilient enhanced two-factor user authentication scheme in wireless sensor networks," in *2014 International Conference on Computing, Networking and Communications (ICNC)*, Feb 2014, pp. 1096–1102.

[21] I. Ijaz, A. Aslam, B. Bukhari, R. Javed, and S. Anees, "Securing cloud infrastructure through pki," in *Fifth International Conference on Computing, Communications and Networking Technologies (ICCCNT)*, July 2014, pp. 1–6.

[22] S. Emerson, Y. K. Choi, D. Y. Hwang, K. S. Kim, and K. H. Kim, "An oauth based authentication mechanism for iot networks," in *2015 International Conference on Information and Communication Technology Convergence (ICTC)*, Oct 2015, pp. 1072–1074.

[23] A. Witkovski, A. Santin, V. Abreu, and J. Marynowski, "An idm and key-based authentication method for providing single sign-on in iot," in *2015 IEEE Global Communications Conference (GLOBECOM)*, Dec 2015, pp. 1–6.