IJIREEICE

ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526

International Journal of Innovative Research in
Electrical, Electronics, Instrumentation and Control Engineering

ISO 3297:2007 Certified

Vol. 5, Issue 2, February 2017

# A Review on Internet of Things - Protocols, Issues

**Mohd. Abdul Sattar[1], Mohammed Anwaruddin[2], Mohd. Anas Ali[3]**

Associate Professor & Head, Dept of ECE, Nawab Shah Alam Khan College of Engineering &Technology, Hyderabad, India[1]

Assistant Professor, Dept of ECE, Nawab Shah Alam Khan College of Engineering & Technology, Hyderabad, India[2]

Assistant Professor, Dept of ECE, Nawab Shah Alam Khan College of Engineering & Technology, Hyderabad, India[3]

**Abstract:** The "Internet of Things" (IoT) concept is used to define or reference systems that rely on the autonomous communication of a group of physical objects. The applications areas of the IoT are numerous, including: smart homes, smart cities and industrial automation. IoT systems often provide great benefits to numerous industries and society as a whole. Many of the IoT systems and technologies are relatively novel. The aim of this paper is to provide the last and most innovative contributions concerning the Protocol, Technology, Application, Architecture & Issues of interest in IoT solutions that involve interconnected smart things that interoperate with the objective of solving problems, provide functionality or optimize tasks.

**Keywords:** IoT, Protocol, Technology, RFID, NFC.

## I. INTRODUCTION

The Internet of Things (IoT) becomes an attractive research topic, in which the real entity in physical world becomes virtual entity in cyber world, and both physical and digital entities are enhanced with sensing, processing, and self- adapting capabilities to perform interaction through special addressing scheme. Along with the combination of Internet and modern sensor technologies such as Radio Frequency Identification (RFID), Near Field Communication (NFC), and Wireless Sensor and Actuator Networks (WSAN), IoT itself is suffering from more rigorous security challenges. Several issues in terms of system architecture, standard, and human involvement are subsequently raised. The following security problems seem to be intense speculations, such as how to design appropriate security framework for things' intelligent applications? What is advanced security technology applied into mass data processing? How to maintain a balance between things' high security requirements and supporting infrastructures' hardware limitation? And how human society securely participates in both cyber and physical worlds with inter- connection? Such significant obstacles influence the development of the future IoT, along with the exposure of mass data which causes various potential vulnerabilities from robust adversaries. Besides, resource restrictions including heterogeneous networks and sensor nodes, communication channels/interfaces, bandwidth, storage, and energy, may also induce unique model design. Towards the general IoT, studies on its architecture model, standard, communication protocol, and network management have been researched. Towards the particular IoT security, there are several open issues such as cryptographic algorithms, authentication protocols, access control, trust/privacy, and governance frameworks. Several researches mainly focus on specific communication techniques (e.g., WLAN, RFID), detailed cryptographic mechanisms (e.g., key management), and practical applications (e.g., supply chain management, multimedia traffic).Meanwhile, the security frameworks in traditional networks can also provide merits for IoT security protection. However, security issue towards the future IoT is not a simple technically tough problem, but a multidimensional topic which combines the information security, network security, infrastructure security, and management security. Most existent schemes provide solutions for special communication techniques or applications, which may lack universality for the complicated system. Thus, we will establish an integrated security architecture to promote universal security consideration for the future IoT. In the paper, we focus on a typical future IoT architecture (short for U2IoT) Protocol, Technologies, Application, Issues, Security, Services which comprises two subsystems that Unit IoT and Ubiquitous IoT. In the U2IoT model, conceptions of mankind neural system and social organization framework are introduced for the future IoT. Thereafter, we propose a systematic security architecture (named IPM) by integrating the awareness and interactivity of cyber world, physical world, and human social into the U2IoT model. Meanwhile, the proposed IPM is presented with embedded interactions among information, physical, and management. Specifically, 1) information security model with the considerations for basic and advanced security

IJIREEICE

ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526

International Journal of Innovative Research in
Electrical, Electronics, Instrumentation and Control Engineering

ISO 3297:2007 Certified

Vol. 5, Issue 2, February 2017

requirements that are mapped into the security layer to deal with sensing, networking, application, and social attribution; 2) physical security including external con-text and inherent infrastructure are inspired by artificial immune, and it ensures that the things should be adapt-able to dynamic semantic contexts with innate and adaptive immunities against malicious attacks; 3) management security provides recommended strategies for hierarchical classified scenes with rationality and compatibility. IPM realizes the unison of cyber world, physical world and human social to guarantee security and privacy for U2IoT. The remainder of the paper is organized as follows. In next Section, we illustrate the existent U2IoT model, and propose the security architecture (IPM). The main features of IPM referring to information security, physical security, and management security are given in subsequent Section. Finally, Section one draws a conclusion. The Internet can be described as a ubiquitous infrastructure that has evolved from being a technology for connecting people and places to a technology connecting things. The future is the Internet of Things (IoT), which aims to unify everything in our world under a common infrastructure, giving us not only control of the things around us, but also keeping us informed of the state of the things around us. One of the main problems with IoT is that it is so vast and such a broad concept that there is no proposed, uniform architecture. In order for the idea of IoT to work, it must consist of an assortment of sensor, network, communications and computing technologies, amongst others. But when you start putting together different types of technologies, the problem of interoperability arises. One proposed solution is to adopt the standards of the services-oriented architecture (SOA) deployed in business software systems. Another takes a similar approach, suggesting the integration of Web Services into sensor network with the use of IoT optimized gateways, which would bridge the gap between the network and the terminal. In general, it may be beneficial to incorporate a number of the technologies of IoT with the use of ser-vices that can act as the bridge between each of these technologies and the applications that developers wish to implement in IoT.

This paper breaks down four main categories of services according to technical features, as proposed and described by[3].In categorizing IoT services, we aim to provide application developers a starting point, giving them something to build upon so that they know the types of services that are available. This will allow them to focus more on the application instead of designing the services and architectures required to sup-port their IoT application. The IoT envisions hundreds or thousands of end-devices with sensing, actuating, processing and communication capabilities able to be connected to the Internet. These devices can be directly connected using cellular technologies such as 2G/3G/Long Term Evolution and beyond (5G) or they can be connected through a gateway, forming a local area network, to get connection to the Internet. The latter is the case where the end-devices usually form Machine to Machine (M2M) networks using various radio technologies, such as Zigbee (based on the IEEE 802.15.4 Standard), Wi-Fi (based on the IEEE 802.11 Standard), 6LowPAN over Zigbee (IPv6 over Low Power Personal Area Networks), or Bluetooth (based on the IEEE 802.15.1).Regardless the specific wireless technology used to deploy the M2M network, all the end-devices should make their data available to the Internet. This can be achieved either by sending the information to a proprietary web server accessible from the Internet or by employing the cloud. Besides acting as remote data bases, M2M clouds also offer the following key services:

1. They offer Application Programming Interfaces (API) with built-in functions for end-users, thus providing the option to monitor and control end-devices remotely from a client device2. They act as asynchronous intermediate nodes between the end-devices and final applications running on devices such as smart phones, tablets or desktops. Our paper focuses on the protocols that handle the communication between the gateways, the public Internet, and the final applications. They are application layer protocols that are used to update online servers with the latest end-device values but also to carry commands from applications to the end-device actuators. The rest of the paper is organized as follows. Next Section describes our research motivation whereas each of the other sections is dedicated to a specific application layer protocol. At the first part of each section we introduce each application layer protocol, we present its usage, we discuss the reliability and security features it offers and we then compare its suitability for the IoT with other application layer protocols. Finally, in one Section, we present overall conclusions based on the previous sections and we provide further research areas. The IoT is a term used for a huge wave of innovation originated in industries, but currently heading to urban centers, in-home environments, and individuals.

## II. PROTOCOLS

Our main motivation was to create an IoT test-bed where to test communications protocols and also innovative applications that could be applied to a gamut of scenarios. While searching for the appropriate application layer protocols to use, we found out that while comparisons can be found between two protocols, there is no paper over viewing all the possible alternatives with pros and cons. The main motivation of this paper is to fill this gap and provide a brief yet accurate description of the key protocols that are being used today to implement the IoT. More specifically, we will discuss on the following list of protocols being used alternatively or jointly to solve different needs of the communication between machines:

1) **CoAP**: Constrained Application Protocol.
2) **MQTT**: Message Queue Telemetry Transport.

IJIREEICE

ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526

International Journal of Innovative Research in
Electrical, Electronics, Instrumentation and Control Engineering

ISO 3297:2007 Certified

Vol. 5, Issue 2, February 2017

3) **XMPP**: Extensible Messaging and Presence Protocol.
4) **RESTFUL Services**: Representational State Transfer.
5) **AMQP**: Advanced Message Queuing Protocol
6) **Websockets**.

**2.1CoAP**

The Constrained Application Protocol (CoAP) is a synchronous request/response application layer protocol that was designed by the Internet Engineering Task Force (IETF) to target constrained-recourse devices. It was designed by using a subset of the HTTP methods making it interoperable with HTTP. CoAP runs over UDP to keep the overall implementation lightweight. It uses the HTTP commands GET, POST, PUT, and DELETE to provide resource-oriented interactions in a client-server architecture. CoAP is a request/response protocol that utilizes both synchronous and asynchronous responses. The reason for designing a UDP-based application layer protocol to manage the resources is to remove the TCP overhead and reduce bandwidth requirements. Additionally, CoAP supports unicast as well as multicast, as opposed to TCP, which is by its nature not multicast-oriented. Running on the unreliable UDP, CoAP integrated its own mechanisms for achieving reliability. Two bits in the header of each packet state the type of message and the required Quality of Service (QoS) level. There are 4 message types:

1. Con_rmable: A request message that requires an acknowledgement (ACK). The response can be sent either synchronously (within the ACK) or if it needs more computational time, it can be sent asynchronously with a separate message.
2. Non-Con_rmable: A message that does not need to be acknowledged.
3. Acknowledgment: It confirms the reception of a confirmable message.
4. Reset: It confirms the reception of a message that could not be processed.

There is also a simple Stop-and-Wait retransmission mechanism for confirmable message sand a 16-bit header field in each CoAP packet called Message ID which is unique and used for detecting duplicates. CoAPC HTTP Mapping enables CoAP clients to access resources on HTTP servers through a reverse proxy that translates the HTTP Status codes to the Response codes of CoAP. Even though CoAP was created for the IoT and for M2M communications, it does not include any built-in security features.

**2.2 Message Queue Telemetry Transport (MQTT)**:

It was released by IBM and targets lightweight M2M communications. It is an asynchronous publish/subscribe protocol that runs on top of the TCP stack. Publish/subscribe protocols meet better the IoT requirements than request/response since clients do not have to request updates thus, the network bandwidth is decreasing and the need for using computational resources is dropping. In MQTT there is a broker (server) that contains topics. Each client can be a publisher that sends information to the broker at a specific topic or/and a subscriber that receives automatic messages every time there is a new update in a topic he is subscribed. The MQTT protocol is designed to use bandwidth and battery usage sparingly, which is why, for example, it is currently used by Facebook Messenger. MQTT ensures reliability by providing the option of three QoS levels:

1. Fire and forget: A message is sent once and no acknowledgement is required.
2. Delivered at least once: A message is sent at least once and an acknowledgement is required.
3. Delivered exactly once: A four-way handshake mechanism is used to ensure the message is delivered exactly one time. Even though MQTT runs on TCP, it is designed to have low overhead compared to other TCP-based application layer protocols. Moreover, the publish/subscribe architecture that it used, is more suitable for the IoT than request/response of CoAP, for example, because messages do need to be responded. This means lower network bandwidth and less message processing that actually extends the lifetime of battery-run devices. To ensure security, MQTT brokers may require username/password authentication which is handled by TLS/SSL (Secure Sockets Layer), i.e., the same security protocols that ensure privacy for HTTP transactions all over the Internet. By comparing MQTT with the aforementioned CoAP, it is possible to see that the UDP-based CoAP has lower overhead than the TCP-based MQTT. However, due to the lack of TCPs retransmission mechanisms, packet loss is more likely to happen when using CoAP. According to a recent research study, MQTT experiences lower delays that CoAP for low packet losses, but CoAP generates less extra traffic for ensuring reliability. However, results can vary depending on the network conditions. Additionally packet loss and delays depend on the QoS of the messages. In both protocols, packet loss degrades and delays increase when the QoS level is higher.

**2.3 The Extensible Messaging and Presence Protocol (XMPP)**

It was designed for chatting and message exchanging. It was standardized by the IETF over a decade ago, thus being a well-proven protocol that has been used widely all over the Internet. However, being an old protocol, it falls short to provide the required services for some of the new arising data applications. For this reason, last year, Google stopped supporting the XMPP standard due to the lack of worldwide support. However, lately XMPP has re-gained a lot of attention as a communication protocol suitable for the IoT. XMPP runs over TCP and provides publish/subscribe (asynchronous)and also request/response (synchronous) messaging systems. It is designed for near real-time communications and thus, it supports small

IJIREEICE

ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526

International Journal of Innovative Research in
Electrical, Electronics, Instrumentation and Control Engineering

ISO 3297:2007 Certified

Vol. 5, Issue 2, February 2017

message footprint and low latency message exchange. As the name explicitly states, XMPP is extensible and allows the specification of XMPP Extension Protocols (XEP) that increase its functionality. XMPP has TLS/SSL security built in the core of the specification. However, it does not provide QoS options that make it impractical for M2M communications. Only the inherited mechanisms of TCP ensure reliability. XMPP supports the publish/subscribe architecture that is more suitable for the IoT contrast to CoAPs request/response approach. Furthermore, it is an already established protocol that is supported all over the Internet as a plus with regard to the relatively new MQTT. However, XMPP uses XML messages (eXtensible Markup Language) that create additional overhead due to unnecessary tags and require XML parsing that needs additional computational ability which increases power consumption.

**2.4 RESTful Services**

The Representational State Transfer (REST) is not really a protocol but an architectural style. It was first introduced by Roy Fielding in 2000, and it is being widely used ever since. REST uses the HTTP methods GET, POST, PUT, and DELETE to provide a resource oriented messaging system where all actions can be performed simply by using the synchronous request/response HTTP commands. It uses the built-in accept header of HTTP to indicate the format of the data that it contains. The content type can be XML or JSON (JavaScript Object Notation) and depends on the HTTP server and its configuration. REST is already an important part of the IoT because it is supported by all the commercial M2M cloud platforms. Moreover it can be implemented in smartphone and tablet applications easily because it only requires an HTTP library which is available for all the Operative Systems (OS) distributions. The features of HTTP can be completely utilized in the REST architecture including cashing, authentication, and content typenegotiation. RESTful services use the secure and reliable HTTP which is the proven worldwide Internet language. It can make use of TLS/SSL for security. However, today most commercial M2M platforms do not support HTTPS requests. Instead, they provide unique authentication keys that need to be in the header of each request to achieve some level of security. Even though REST is already used widely in commercial M2M platforms, it is unlikely that it will become a dominant protocol due to not being easily implementable. It uses HTTP which means no compatibility with constrained-communication devices. This leaves its use for final applications. Given the current tendency for applications running on smartphones, tablets and pads, the additional overhead associated to request/response protocols affect battery usage, as it also does the continuous polling or long polling for values especially when there are no new updates and the overhead becomes useless. Issues that can be avoided if a publish/subscribe protocol is used such as MQTT or XMPP. CoAP on the other hand, which is the lightweight version of REST, bears the same disadvantages of the request/response architecture. However it is designed to run over UDP making it capable of being used be constrained resource devices, counter to REST.

**2.5 Advanced Message Queuing Protocol (AMQP):**

The Advanced Message Queuing Protocol (AMQP) is a protocol that arose from the financial industry. It can utilize different transport protocols but it assumes an underlying reliable transport protocol such as TCP AMQP provides asynchronous publish/subscribe communication with messaging. Its main advantage is its store-and-forward feature that ensures reliability even after network disruptions. It ensures reliability with the following message-delivery guarantees:

1. At most once: means that a message is sent once either if it is delivered or not.
2. At least once: means that a message will be definitely delivered one time, possibly more.
3. Exactly once: means that a message will be delivered only one time. Security is handled with the use of the TLS/SSL protocols over TCP. Recent research has shown that AMQP has low success rate at low bandwidths, but it increases as bandwidth increases. Another study shows that comparing AMQP6with the aforementioned REST, AMQP can send a larger amount of messages per second. Additionally, it has been reported that an AMQP environment with 2,000 users spread across five continents can process 300 million messages per day. Furthermore, JPMorgan which is an American banking and financial services company uses AMQP to send 1 billion messages per day.

**2.6 WEBSOCKET:**

The Websocket protocol was developed as part of the HTML 5 initiative to facilitate communications channels over TCP. Websocket is neither a request/response nor publish/subscribe protocol. In Websocket a client initializes a handshake with a server to establish aWebsocket session. The handshake itself is similar to HTTP so that web servers can handle Websocket sessions as well as HTTP connections through the same port. However, what comes after the handshake does not conform to the HTTP rules. The session can be terminated when it is no longer needed from either the server or the client side. Websocket was created to reduce the Internet communication overhead while providing real-time full-duplex communications. There is also a Websocket sub-protocol called Websocket Application Messaging Protocol (WAMP) that provides publish/subscribe messaging systems. Websocket runs over the reliable TCP and implements no reliability mechanisms by its own. If needed, the sessions can be secured using the Websocket over TLS/SSL. During the session, Websocket messages have only 2 bytes of overhead. As reported by relevant studies, the HTTP polling (in REST) repeats header information when the data transmission rate increases,

IJIREEICE

ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526

International Journal of Innovative Research in
Electrical, Electronics, Instrumentation and Control Engineering

ISO 3297:2007 Certified

Vol. 5, Issue 2, February 2017

thus increasing latency. Websocket is estimated to provide a three-to-one reduction in latency against the half-duplex HTTP polling. Websocket is not designed for resource constrained devices as the previous protocols and its client/server based architecture does not suit IoT applications. However it is designed for real-time communication, it is secure, it minimizes overhead and with the use of WAMPit can provide efficient messaging systems. Thus, it can compete any other protocol running over TCP.

## III. ISSUES

In scenario, the effect of IoT can be seen in all technical areas. It helps in smart communication between objects but several issues are there to be addressed before the worldwide implementation of IoT. In this section, we identify some important issues related to addressing, routing protocol, security and privacy, standardization issue and congestion and overload issue.

### 3.1Addressing and networking issue:

Each and every device connected in the network has a unique address by which it can be identified. As the IoT is gaining grounds in scenario, the demand for these unique address increases at a very fast rate. There are very limited number of address available in IPv4 addressing and will soon reach zero as it identifies each node through a 4-byte address. To handle the ever increasing demand of unique address, one require IPv6 addressing scheme to full fill the requirement. IPv6 addresses are expressed by means of 128 bits and, therefore, it is possible to define 1038 addresses, which should be enough to identify any object. Another important issue is regarding networking i.e. which protocol is to be used to send the data from source to destination. In traditional internet, the protocol utilized at the transport layer for reliable communications is the Transmission Control Protocol (TCP) (CERF;DALAL; SUNSHINE, 1974). It is clear that TCP is insufficient for the IoT because we need to set-up a connection first in case of TCP, but most of communication in IoT is a very short communication. So, considerable time will be wasted in the connection setup. One more issue with TCP is congestion control, TCP is responsible for end-to-end congestion control, but incise of IoT the amount of data transfer is very small, so TCP congestion control is useless. As a consequence, TCP cannot be used efficiently for the end-to-end transmission control in the IoT. Till now, no solutions have been proposed and, therefore, research is required in this area.

### 3.2 Routing protocol issue in V2V communication:

Routing is a very important aspect in the field of V2V communication as it is a type of distributed processing with a great number of nodes and a constrained and highly variable network topology. There are two basic ways by which one can route the data from source to destination. The first one is source routing: in this all the information like how to get from source to destination is collected on the source and then stored in the packets to be send, and the job of the intermediate node is to read this information and route the packet according to it towards the destination. Second one is hop-to-hop routing: in this routing technique, node has information only about the next node; the work of intermediate node is a bit complex as they know the destination address only, not the whole route to get towards the destination (KUMAR;KUMAR; KADIAN, 2011). This hop-to-hop is more efficient as in this we can choose the best next hop according to the topology. The architecture of routing in V2V communication is the same as the architecture of routing in other connectionless networks. Routing is the backbone of the network. There are lots of protocols present there like Geographical Source Routing (JERBI et al., 2009) which is hop-to-hop routing. This routing is based on the topology information given by global positioning system; frequently changes in topology causes route oscillation and path instability. In On-Demand Routing protocol (DAS; PERKINS; ROYER, 2000) node attempts to discover a route to the destination when it has a packet to send. In this protocol, flooding method is used to discover the route which creates the congestion in the network as it sends the packet to all the nodes for route discovery. There are various other routing technique like Greedy Perimeter Stateless Routing (GPSR)(KARP; KUNG, 2000), Dynamic MANET on Demand (DYMO) (CHAKERES; PERKINS, 2006),etc., but each one has its shortcoming. The key challenge is to design a protocol which will improve reliability of protocols and reduce delivery delay time and number of packet transmission. To make VANET a reality, lots of research is needed as each one of the existing protocol has some drawbacks as explained above. The driver behavior should also be concerned in designing the routing protocols.

### 3.3 Privacy and security issue:

The IoT is extremely vulnerable to attacks as its components spend most of the time unattended, so it became very easy to attack them. Apart from this, one more thing is that, most of the communication is wireless which makes snooping very easy. This is probably one of the biggest concerns for consumers when it comes to IoT. For instance, in NFC enabled devices, the device not only works as a credit card but also the key to your house, it will also contain the personal information of the owner. If a smartphone is stolen, the thief move's the phone over a card reader at a store to make a purchase (NFC, 2013). To avoid this, smartphone owners must protect their phone with strict password protection, so hacker is not able to come out with the correct password. More specifically, the major problems related to security concern authentication and data integrity. Authentication is required before making a connection between the two devices to prevent data theft. The infrastructure is required for the authentication as we generally have to exchange some public and private keys through the node. Solutions like

IJIREEICE

ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526

International Journal of Innovative Research in
Electrical, Electronics, Instrumentation and Control Engineering

ISO 3297:2007 Certified

Vol. 5, Issue 2, February 2017

cryptography and key management have been proposed in the recent past (e.g., (KAVITHA; SRIDHARAN, 2010),(ESCHENAUER; GLIGOR, 2002a)), but none of them will prevent from the man-in-the-middle attack and proxy attack problem.

Data integrity prevents any modification in the data by middle man; it ensures that the data received at the receiver node is in the unaltered form as send by the sender. Solutions have been proposed like Keyed-Hash Message Authentication Code (HMAC) scheme (ESCHENAUER;GLIGOR, 2002b), to protect the data against the attack but still new research is required in the field of security and privacy.

### 3.4 Standardization issue:

Standards are required to allow global interoperability. As the term Internet of Things is gaining popularity, the more and more number of devices is activated daily. To ensure the proper functionality of these devices, there should be certain standards we have to follow to provide proper service to the client. As the platform on which these IoT devices works is not the same in all cases, so it became more necessary to define certain standards to make those devices compatible with the others. EPCglobal (Electronic Product Code) (EPCGLOBAL, 2013),as well as ISO (International Organization for Standardization), offers a family of standards, and they are gaining popularity in the wireless sensor area.

### 3.5 Congestion and overload issue:

Congestion is occurred due to simultaneous messages from several devices that can lead to peak load situation and may have a tremendous impact on the network (3GPP, 2010). This affects the performance of the network, and may lead to failure of the network if the network is overloaded. This situation is mainly seen in M2M and V2V communication, and it can be solved with the help of emerging technologies like LTE-advanced or existing technologies like LTE high bandwidth networks (TALEB; KUNZ, 2012).

The congestion situation also occurred because of malfunction of server or application; so to avoid this one has to design an application in such a way that can handle maximum load with minimum failure. Overload issue can be solved with the help of time controlled features, i.e., allow connection to the network only at a certain time periods, defined by the network operator. Only in this time period, the devices are allowed to connect, devices are not able to connect to the network in the forbidden time period. The other solution is by rejecting the connection request by specific network nodes, particularly from those that are causing congestion and shall have no impact on the traffic (TALEB; KUNZ, 2012). This will help in managing the overall load of the network by rejecting the nodes which are creating the congestion.

## IV. CONCLUSION

In this paper, a security architecture IPM is discussed and analysed for U2IoT model. The main purpose is to establish integrated security architecture with considerations on cyber-physical-social world. The proposed IPM comprises three essential security perspectives (i.e., information, physical, and management), in which three-dimensional information security model introduces social layer, and intelligence and compatibility for security consideration; artificial immunity is applied to describe physical security; and a series of social strategies are recommended to achieve management security.

## ACKNOWLEDGMENT

We express our sincere gratitude to P**rof. Mohd. Muzaffar Ahmad,** Electronics & Communication Engineering Department, Nawab Shah Alam Khan College of Engineering & Technology, Hyderabad, for extending his valuable insight for completion this work.

## REFERENCES

[1] Tasos Kaukalias and Periklis Chatzimisios, Internet of Things (IoT) C Enabling technologies, applications and open issues, Encyclopedia of Information Science and Technology(3rd Ed.), IGI Global Press, 2014.

[2] Periklis Chatzimisios, Industry Forum & Exhibition Panel on Internet of Humans and Machines, IEEE Global Communications Conference (Globecom 2013), Atlanta, USA, December 2013.

[3] Angelo P. Castellani, Mattia Gheda, Nicola Bui, Michele Rossi, Michele Zorzi, Web Services for the Internet of Things through CoAP and EXI, IEEE International Conference on Communications Workshops (ICC), 5-9 June 2011, pp. 1-6.

[4] Sye Loong Keoh, Sandeep S. Kumar, Hannes Tschofenig, Securing the Internet of Things: A Standardization Perspective, Internet of Things Journal IEEE (Volume:1,Issue: 3), June 2014, pp. 265-275.

[5] Maria Rita Palattella, Nicola Accettura, Xavier Vilajosana, Thomas Watteyne, Luigi Alfredo Grieco, Gennaro Boggia, Mischa Dohler, Standardized Protocol Stack for the 8Internet of (Important) Things, Communications Surveys & Tutorials IEEE 15(3),2013, pp. 1389-1406.

[6] Thamer A. Alghamdi, Aboubaker Lasebae, Mahdi Aiash, Security Analysis of the Constrained Application Protocol in the Internet of Things, Second International Conference on Future Generation Communication Technology (FGCT), 12-14 Nov.2013, pp. 163-168.

[7] Shahid Raza, Hossein Shafagh, Kasun Hewage, Ren Hummen, Thiemo Voigt, Lithe:Lightweight Secure CoAP for the Internet of Things, Sensors Journal, IEEE 13(10),Oct. 2013, pp. 3711-3720.

[8] Shinho Lee, Hyeonwoo Kim, Dong-kweon Hong, Hongtaek Ju, Correlation Analysis of MQTT Loss and Delay According to QoS Level, International Conference on Information Networking (ICOIN), 28-30 Jan. 2013, pp. 714-717.

[9] http://mqtt.org/2011/08/mqtt-used-by-facebook-messenger, cited 28 Jul 2014.

[10] Dinesh Thangavel, Xiaoping Ma, Alvin Valera, Hwee-Xian Tan, Colin Keng-Yan Tan, Performance Evaluation of MQTT and CoAP via a Common Middleware, IEEE Ninth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 21-24 April 2014, pp. 1-6.

[11] http://www.zdnet.com/google-moves-away-from-the-xmpp-open messagingstandard-7000015918/, cited 28 Jul 2014.

IJIREEICE

ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526

International Journal of Innovative Research in
Electrical, Electronics, Instrumentation and Control Engineering

ISO 3297:2007 Certified

Vol. 5, Issue 2, February 2017

[12] Sven Bendel, Thomas Springer, Daniel Schuster, Alexander Schill, Ralf Ackermann, Michael Ameling, A Service Infrastructure for the Internet of Things based on XMPP,IEEE International Conference on Pervasive Computing and Communications Workshops(PERCOM Workshops), 18-22 March 2013, pp. 385-388.
[13] Michael Kirsche, Ronny Klauck, Unify to Bridge Gaps: Bringing XMPP into the Internet of Things, IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops), 19-23 March 2012, pp. 455-458.
[14] Roy Thomas Fielding, Architectural Styles and the Design of Network-based Software Architectures, PhD thesis, University of California, Irvine, USA, 2000.
[15] Bipin Upadhyaya, Ying Zou, Hua Xiao, Joanna Ng, Alex Lau, Migration of SOAP based Services to RESTful Services, 13th IEEE International Symposium on Web Systems Evolution (WSE), 30 Sept. 2011, pp. 105-114.
[16] http://en.wikipedia.org/wiki/Advanced Message Queuing Protocol, cited 28 Jul 2014.
[17] Frank T. Johnson, Trude H. Bloebaum, Morten Avlesen, Skage Spjelkavik, Bjørn Vik, Evaluation of Transport Protocols for Web Services, Military Communications and Information Systems Conference (MCC), 7-9 Oct. 2013, pp. 1-6.
[18] Joel L. Fernandes, Ivo C. Lopes, Joel J. P. C. Rodrigues, Sana Ullah, Performance Evaluation of RESTful Web Services and AMQP Protocol, Fifth International Conference on Ubiquitous and Future Networks (ICUFN), 2-5 July 2013, pp. 810-815.9
[19] http://www.amqp.org/about/examples, cited 28 Jul 2014.
[20] http://en.wikipedia.org/wiki/WebSocket, cited 28 Jul 2014.
[21] Victoria Pimentel, Bradford G. Nickerson, Communicating and Displaying Real-Time Data with WebSocket, Internet Computing IEEE 16(4), July-Aug. 2012, pp. 45-53.

## BIOGRAPHIES

**Mohd Abdul Sattar**, received B. Tech. Degree in Electronics and Communication Engineering from National Institute of Technology (NIT), Warangal and M.Tech. in Embedded Systems from JNTUH. He is an Associate Professor & Head of the Dept. of ECE in Nawab Shah Alam Khan College of Engineering & Technology, Malakpet, Hyderabad. He is also a member of IEEE.

**Mohammed Anwaruddin**, received B.Tech. Degree in Electronics and Communication Engineering from JSN College of Engineering & Technology and M.Tech. in Digital Electronics & Communication Systems from JNTU College of Engineering, Anantapur. He is an Assistant Professor in Dept. of ECE at Nawab Shah Alam Khan College of Engineering & Technology, Malakpet, Hyderabad.

**Mohd Anas Ali**, received B.Tech. Degree in Electronics and Communication Engineering from Pujya Shri Madhavanji College of Engineering & Technology affiliated to JNTU Hyderabad in 2013 & M. Tech. degree in Embedded System from Nawab Shah Alam Khan College of Engineering & Technology. He is presently working at Nawab Shah Alam Khan College of Engineering & Technology, Malakpet, Hyderabad.