High-level architectural design of management system for the Internet of Underwater Things

Khamdamboy Urunov
Special Communication Research Center,
Kookmin University
Seoul, South Korea
hamdamboy@kookmin.ac.kr

Soo-Young Shin
Special Communication Research Center,
Kookmin University
Seoul, South Korea
sy-shin@kookmin.ac.kr

Jeong-Il Namgung
Special Communication Research Center,
Kookmin University
Seoul, South Korea
greenji@kookmin.ac.kr

Soo-Hyun Park
School of Software, Kookmin University,
Seoul, South Korea
shpark21@kookmin.ac.kr

Abstract**—** High-level seamless interconnecting network community is able to make a service for using sufficiently network resources. In order to integrate services and devices discovery opportunity for using the Internet of Things (IoT) possibility. Such as a number of integration in IoT devices and services, they are requiring available management system. Indeed, Network Management System (NMS) is the ability of monitoring, managing and optimizing of the network. In order to processing management system which is efficiently optimizing resource consumptions and reducing the cost of services. The NMS is an application layer protocol and set of system components. This paper represents management system role and integrating the terrestrial system to the constraint environment. Based on high-level system architecture model designing, and using management mechanism for the Internet of Underwater Things (IoUT). Moreover, topological discovery algorithm helps to improve management services and adjusting an underwater environment.

Keywords — IoT, NMS, IoUT, Underwater-NMS(U-NMS), Managment Information Base (MIB), Simple Network Management Protocol (SNMP), underwater SNMP (u-SNMP)

I. INTRODUCTION

The Internet is one of a big market for integrating technologies and services. This opportunity can use the process the management system, it acts as functions and elements based on reading and writes different state variable of the network. Such as a hardware power, a device configuration, a traffic routing [1], and a multi-device tunneling. Such of the way the management system can provide devices and networks sufficiently integration. Basically, management is a local or remotely configured management devices based on software [2]. In this case, management network supports the main role of the data modeling for using the system configuration. Another option is adjusting proceed of the workstation and other related hardware elements. Thus, NMS is set to [3] application and obtain system devices and networks included management. The NMS acts as a small program to remotely configure network devices whenever to adjust proceeding of a workstation with the management system. According to research of the management system, the paper relies on a constrained management and integration of the unconstrained network. Especially, the constrained network is an underwater environment as a U-NMS and this stands for Underwater – Network Management System (U-NMS). The U-NMS is a small software program for adjusting each individual device in the underwater network [4]. The U-NMS is an application layer protocol and the ability of managing system devices. Fig.1 illustrates constrained management system models and that is a different point of the constraint management from legacy NMS. The main components and functions pointed out following structural elements. The first one is a Functions, which consists of FCAPSC using first letter abbreviation (shown Fig.1, Functions). The green color rows are constrained management components.

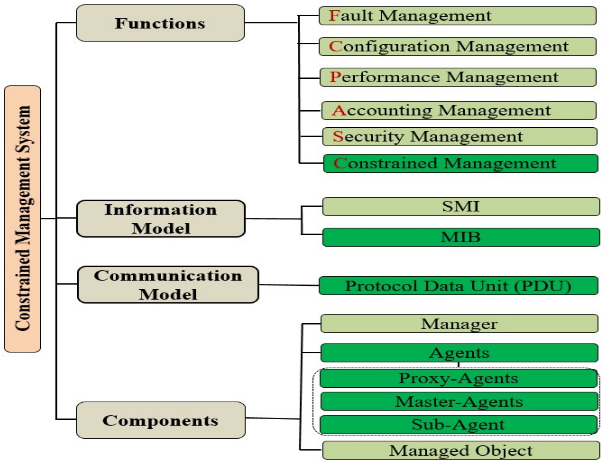


Fig.1 Constrained Management System

Those types of Functions are required lightweight and integrated system elements. FCAPSC is one of the examples.

Fault management – that requires easy detecting, correlating network problem to keep the network running effectively, recovering, handling, and filtering management.

Configuration management – this system can support constrained device (like underwater) configurations and dynamic changing operations based on versions. That should be an easy task to manage or to configure in-network facility.

Performance management – this system can support detecting poor response times observed and reconfiguring in the network to alleviate the challenges.

978-1-5386-4646-5/18/$31.00 ©2018 IEEE

Accounting management – this can support network resource for using users, which can charge based on the tariff.

Security management – this follows confidentiality and authentication in the system and using secure mechanism.

Constrained management – system supports constrained environments software and protocols. That is supposed to be lightweight environmental components, etc.

The second is concerning SMI and MIB module. This stands for Structure Information Base (SMI), for designing Managed Object (MO). The advantage point is lightweight MIB integrating to underwater acoustic communication. The third one is a Communication Model, and Protocol Data Unit (PDU). Those are the main body of a message format and consist of several parts as a header, PDU, and reserve field. The important factor is reducing the number of message and header value. Commonly SNMP (Simple Network Management Protocol) [5] can use around 4 KB message size. This message size so high level for implementing the constrained environment and expecting value is 10 byte for the underwater real testbed. The last one is a Component, this concerns the manager (server side, in Fig.2 Datacenter) and agents. IoT and IoUT devices always support agent resource.

II. Motivation and Related works

The legacy network management is so widely using and more available in the terrestrial area. Likewise, impossible to use directly most of the management components in the constrained network. Because of that, the system requires a lightweight and easy integrating management system techniques. Initial motivation of this paper is able to make underwater management system and deploying the steps in the system. According to the researching underwater management is related protocols and MO binding techniques. The basic structure of SNMP relies on manager and agent capable of interconnection. Indeed, SNMP consists of three versions and using different operations. SNMPv1 [5] is not security concerned and efficient. It has been designed to be an interim solution. SNMPv2 [6] aggregates three new more operations (GetBulk, Notification, Inform) and authentication of the Message source. The placing access control on MIBs. The method of the trap as similar another version of PDU value. The final version 3 of SNMP [7] is the addition of security and administrative capabilities. Moreover, inform and report new operations are working well. Even characteristic of message format SNMP v1 is Version, Community, and PDU (value and type), Request ID, Error status, Error index, Variable binding. SNMP consists of measurement that is a 4 byte.

TABLE-1. COMPARISON OF RELATED TECHNIQUES

Categories	SNMP	NETCONF	CWMP
Standard	IETF	IETF [8]	DLS Forum
Protocol	Application layer		
Transport	UDP	SSH/TCP	TCP
Encoding	ASN.1 with BER	YANG/SOAP	XML/SOAP
MO	MIB	Topo Objects	Parameters
Number of MO in MIB	Few < (10,000)	Few < (10,000)	Many > (10,000)
MIB	SMI	YANG	GMDO
Identify MO	OID	Unique ID	Parameter's name
How to know MO	Pre-defined	Pre-defined	Dynamic

SNMPv2 also similar to version 1, and SNMPv3 has scoped PDU value is higher than another version of SNMP. There is Context Engine ID, Context Name, PDU control Field, and Variable binding. The constraint management needs to the lightweight of SNMP or other related protocol. In the underwater, it should be u-SNMP [9] is a lightweight version for underwater management protocol. The main different point is reducing message size and number of messaging to the agent and manager. We made lightweight uMIB to underwater each individual devices. There is another candidate technique for the IoT management system using constrict network. That is a Low PAN Network management (LNMP) [10] protocol over UDP/IPv6. Table-1 defined several protocol techniques for using related protocol compression. CPE WAN management protocol (CWMP) as a Technical Report-069 (TR-069) [11] is using TCP/IP protocol suite. By the more, Network Configuration (NETCONF) is standard protocol in the IETF and using YANG associated data modeling language. When compare SNMP and NETCONF, they are using different data modeling. YANG is namespace-stmt (URI) for addressing system. SMI (in SNMP) is Module-Identity (OID). The underwater network is not stainable connecting manager and agent based on acoustic communication, that needs to a management system for deploy and using a lightweight software usage dynamically.

III. Management system usage for IoUT

The management system relies on manager and agent trust relationship with the community. When the manager and agent are applications in the same community group, those can communicate with each other easily when communication tunnel is stainable. There are two kinds of modeling group as the environment as a terrestrial and an underwater.

A. *System structure and devices integration*

When the processing of deployment to the real sector of a communication network, that should be devices characteristic of Full Functional Device (FFD) and Reduce Functional Device (RFD). Both functional devices required lightweight management system techniques. However, the current management system has several problems for managing IoT and IoUT. Additionally, dynamically changing devices positions and number of interconnecting management systems. FFD can upload manager or master agent (more responsible and functional elements than a subagent) and RFD can be deployed agent (subagent). Those devices operate on these layers in any self-organizing [16] application network. This device required even less memory and small ROM, RAM (see Fig.2). The major reason is most of the sensor node or IoUT devices as an RFD, which deployed the agent. If no more limitation processing of the management system that should be servers including databases as FFD. The paper defined devices discovery mechanism and algorithm. However, there are several challenges and problems concerning the main factors of building more reliable and an efficient underwater network management solution. The current practice of constraint network management has several challenges:

- Dislocate management of the network elements

- Heavy reliance on the low-level interfaces with constraint devices
- Seamless interconnection, knowing that each constraint devices are complex
- Terrestrial current network protocols are not available to connect constraint devices

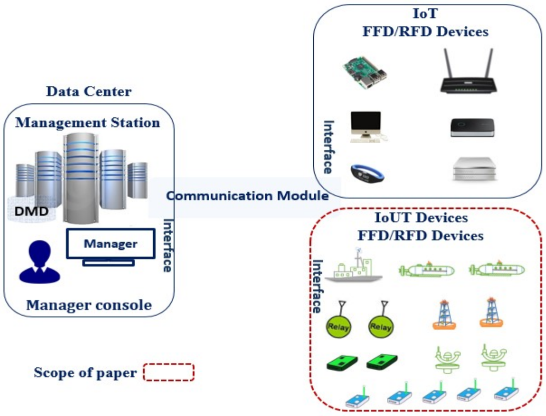


Fig.2 Scope of integration IoT/IoUT and Data Management Center

When considering the first problem, many components are involved in the hop-by-hop path of the underwater application. Initially, applications are running on the servers in Management Center (see Fig.2) and management station can send traffic to other servers or IoT/IoUT workstations. The second is configuring network infrastructure and operating in the hardware. That is also related to power management, to communication topology setup. The traffic engineering for routing the application through the network infrastructure. The third one is impossible seamless interconnection devices in the management system. The underwater case, there might be more disconnection, disruption, and losing data etc. Indeed, according to the problems, the paper defined several approaches:

- The system protocols must be lightweight and easy integrate low-level interface, and devices
- The system devices can support heterogeneous network interface and lightweight MIB modules
- Design of architectural management system for developing underwater management
- Requiring of FCAPSC (Fault, Configure, Account, Performance, Security, Constrained) for using functional elements
- Managing an overwhelming flow of low-quality alarms, fault management process
- Discovery mechanism based on an algorithm

B. Constraint management system requirements for the IoUT

Management system requires general and functional requirements. That should be as the following structure:

a) The management system has required a manager and lightweight agents for the underwater communication.

b) In U-NMS, FCAPSC is required for the solid management of each device in the underwater environment.

c) The U-NMS architecture must be able to scale with a large number of underwater devices.

d) The U-NMS requires underwater device configurations and dynamic changing position. The U-NMS should easily support the task managing changes in network configuration.

e) The U-NMS enabler can require lightweight MO for each individual Unique OID.

f) The U-NMS enabler requires Object ID and identity lightweight variable bindings.

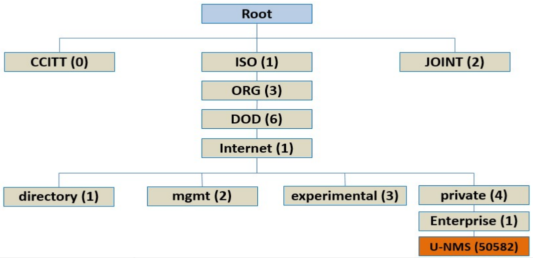
g) The U-NMS enabler requires a device to discover or another method of booting managed devices.

h) In U-NMS enabler requires the network scheduler for reducing the network traffic.

Indeed, it is also important for all managed devices to be accessible using a single user interface.

C. MIB structure

MIB is a core element for controlling system device and network. The MIB is a virtual database which is including a MO, OID, and values. MIB module is concerning 100 standardized specified MO via SMI. However, MIB module always extended and any organization or institute can join own a private MIB to a standard one. The Fig.3 illustrated MIB structure and identifier table on it.



Internet	**OBJECT IDENTIFIER :: = {ISO ORG(3) DOD(6) 1}**
directory	OBJECT IDENTIFIER :: = {Internet 1}
mgmt	OBJECT IDENTIFIER :: = {Internet 2}
experimental	OBJECT IDENTIFIER :: = {Internet 3}
private	OBJECT IDENTIFIER :: = {Internet 4}
Enterprise	OBJECT IDENTIFIER :: = {Internet private(4) 1}
U-NMS	OBJECT IDENTIFIER :: = {Internet private(4) Enterprise (1) 50582}

Fig.3 Management Information Base structure

As before defined U-NMS, which joined after Enterprise (1) value and it has unique number U-NMS (50582) from IANA PEN [12]. The language of SNMP relies on SMI, ASN.1, and BER. That stands for SMI and specifies the format used for defining MOs that are accessed via the SNMP. The next important factor is ASN.1 (Abstract Syntax Notation One) used to define the format of SNMP messages and MO. They are using an unambiguous data description format. The last one is BER (Basic Encoding Rules) used to encode the SNMP message into a format suitable for transmission across a network. The main idea is transmitted data is self-identifying and using data type (ASN.1defined type), length of the data byte, and value of data which encoded according to ASN.1 standard.

IV. HIGH-LEVEL ARCHITECTURE MODEL FOR IOUT

This part of the paper includes detailed information on management components and high-level architectural model integration. Indeed, candidate protocols are related to design structure model of high-level architecture. SNMP and NETCONF candidate techniques.

A. Architectural model for IoUT

The initial step of this part, defining all of the related management components. Especially, underwater with agent structure. The Fig.4 shows high-level architecture module components and integration API structure. Those following structures have management components:

Data Center – the ability of monitoring, collecting data, and request and response possibility.

Manager console – someone is an administrator for adjusting the whole of the system community.

Manager - one of the high-level performance devices which have the capability of querying any managed device – via polling.

Agent – that is the small piece of source code, that is running on every underwater device.

Device Management Database (DMD) – regularly stores and contains a standard set of statistical. The MIB file also included there.

MIB - collections of definitions, which is the property of the MO within the device to be managed.

Object ID (OID) - Uniquely identify managed objects in a MIB hierarchy.

MO – represents characteristics of a managed device.

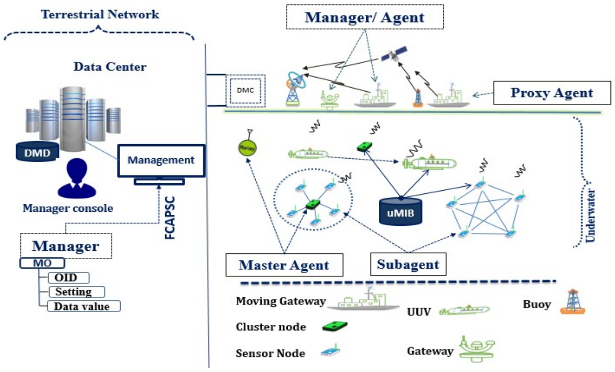


Fig.4 High-level architectural model for IoUT

Additionally, Data management center is adjusting manager possibility and functions. FCAPSC function is starting there. Surface case devices (Gateway, moving Gateway, and Buoy) are supporting manager or agents. That depends on topological mechanism and structure. Especially, an agent has three types of the management system. Those are proxy, master, and subagent. A proxy agent is a middle process of the management system and unmanaged devices, allowing management by proxy. Master agent is a small piece of software installed on Cluster head, UUV and others Master agent can collect data from a subagent. A subagent is a small piece of program adjusting system deploying IoUT devices. Considering the system protocol SNMP based community, devices used to identify the group nodes and sensors. The initial step, the management system can use the basic security of SNMP. The Fig.5 illustrates more detailed information concerning the management system elements.

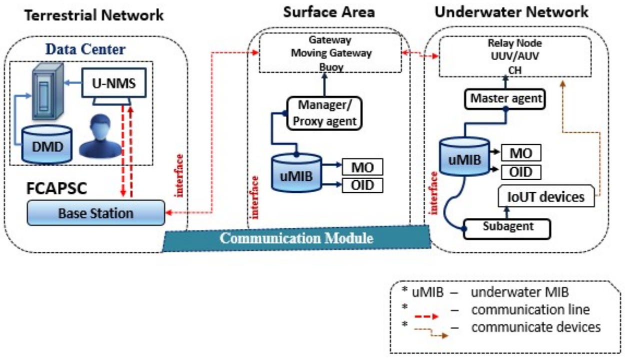


Fig.5 Conceptual model of management system design for IoUT

Indeed, uMIB has attached each underwater devices and all underwater devices MO, other data value included. Each functions as FCAPSC and using the whole of the management system.

B. Topological discovery algorithm

This algorithm is discovery mechanism of a topological model of underwater devices. Two main address system are included this algorithm (Application layer management OID and Network layer IP). Before explaining this algorithm we should define several terms and definitions. Table-2 represents discovery algorithm terms.

TABLE-2. DISCOVERY ALGORITHM TERMS

Terms	Definition
Waiting_list	For waiting for activation in the system.
IPGatewayTable	IP address lists with the active gateway and other related devices.
Gateway type	Devices category is included MIB library
IPGatewayMask	Mask address of the gateway
IPDeviceNextHop	Finding and integrating next device
IPDeviceDest	Devices destination address
Alive_list	The device is still active in the system
OID_list	The management system is deployed in devices
IP_list	Devices already included the system

There are three steps for defining discovery algorithm method.

B1. Algorithm process and specific terms

As before explained, a gateway is an underwater device (see Fig.4) on the surface. Here is the initial point of discovery mechanism which is adding gateway queue in searched Waiting_list. Address of the IP needs to the analysis of the current gateway. Management system case, SNMP is reading the current gateway's DB information. Most of IP address and other MO information invoked to uMIB library in the IPGatewayTable data. Indeed, traverse each record in the table, when GatewayType = direct (directly connected), it will IPGatewayTable and IPGatewayMask according to bitwise and operation. Here is important to directly connected subnet address information, and added the subnet to subnet queue. The gateway and subnet added to connect queues at the same time [13]. Gateway and subnet are relationships added to connect queue when GatewayType = indirect. That is the important rule for Gateway Type value. That can find the next hop devices directly connected device IPDeviceNextHop, it not discovered, add it to the gateway queue and added IPDeviceDest to be found device queue Waiting_list. When Waiting_list all IP were processed to generate the network topological structure.

B2. 6LowPAN topology discovery algorithm.

IPv6 over Low power Wireless Personal Area Network (6LoWPAN) is the first and most commonly used standard in this category. 6LowPAN is network layer protocol for using constrained devices and network [14]. In the majority of the network can support at the layer topology discovery, mainly based on ping, traceroute tool for topology discovery. The basic idea is to use the ping command to determine the active devices IP address. Those ways to define devices active or unmanaged in the system. Surely, the gateway analyzes the relationship between the IP devices connection information and then testing multiple access devices. Afterward, that should send 6LowPAN packet to get the network address and subnet mask address of the gateway interface.

B3. Structure of algorithm process.

The message of ping the address space of all the IP addresses, the IP address will be added to the response to active queue Alive_list, to determine the active device, the traceroute trace Alive_list each IP address. Moreover, OID_list is adjusting devices based on the management system. Finally, analyzing gateway interface, IP address and subnet IP address which is getting the relationship in the connection queue. The Fig.6 illustrated algorithm steps for discovery devices. The initial step is generated the Alive_list and sending Ping message all devices in the system. If system devices are getting pending queue (initialized to be the search for the entire range of IP addresses), response (NO) to remove an IP address and this device is not managed the system. Add the device to IPGatewayTable and sending managed devise step. However, checking the response is (YES) active device in the system and get the IP address belongs to the subnet address. Then the subnet address of the default device will search its device address to the gateway queue to be Waiting_list. After completing this process, search for the device to be out of a gateway queue Waiting_list, determine whether the gateway supports SNMP protocol. The Fig.7 depicted Topological discovery algorithm-b for using management system part of it.

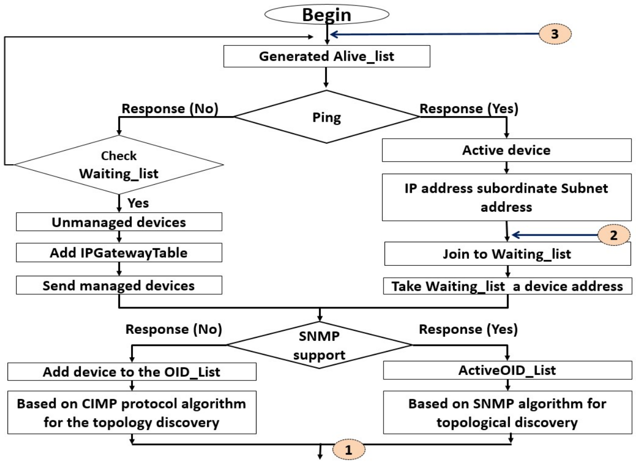


Fig.6 Topological discovery algorithm -a

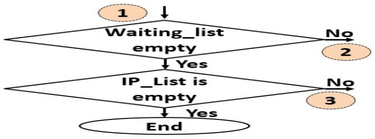


Fig.7 Topological discovery algorithm -b

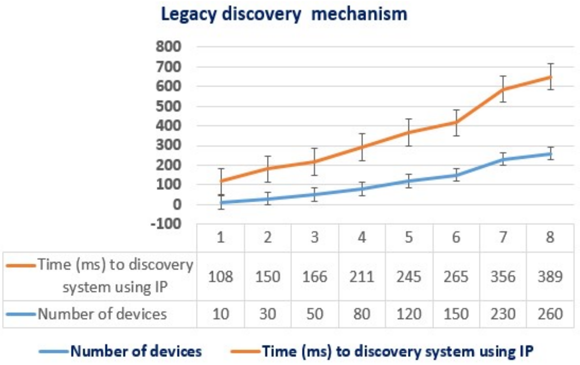
Their suggestion idea is two kinds of devices list as an already management in the MIB, that included devices and no any management system value. If supported, the gateway has been found to support added to the SNMP protocol. If the device cannot support SNMP, add the device to the OID_List and using CIMP-based protocol network layer topology discovery algorithm. if devices supported management system, in that case, added to the device ActiveOID_List a using SNMP algorithm. Indeed, for using the 6LowPAN protocol based network layer topology discovery algorithm and SNMP for management topology discovery, they are checking the device queue Waiting_list be searched is empty, if empty, check IP_list queue. Otherwise, return again to take a looping queue from Waiting_list devices. Table-3 represents the conceptual calculation of discovery mechanism.

TABLE-3. DISCOVERY ALGORITHM FOR THE NUMBER OF DEVICES ACTIVITY AND USING MANAGEMENT

Number of devices	Discovery System Using IP (Alive Activated) Time	Management system discovery Time (ms)
10	108	94
30	150	122
50	166	138
80	211	200
120	245	210
150	265	215
230	356	323
260	389	355

Those numbers chosen randomly and calculated connection time to a system and using two possible way. The first one is legacy system algorithm (without management) and the second one is using management discovery. When the

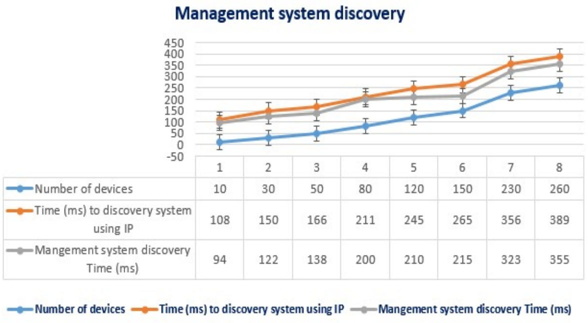
IP_list queue is empty, the process is ended. Finally, this algorithm controls all possible devices and nodes interconnection to management system discovery.



	1	2	3	4	5	6	7	8
Time (ms) to discovery system using IP	108	150	166	211	245	265	356	389
Number of devices	10	30	50	80	120	150	230	260

Fig.8 Legacy discovery mechanism

The Fig.8 represents a number of devices and discovery time represents (more time without management process). We design Fig.8 based on Table-3 second column data values. If many devices are not active, that should be more time for integrating or problem to the connection. This legacy system mechanism getting more time based on only (Ping message right side YES response) in the Fig.6/Fig7.



	1	2	3	4	5	6	7	8
Number of devices	10	30	50	80	120	150	230	260
Time (ms) to discovery system using IP	108	150	166	211	245	265	356	389
Mangement system discovery Time (ms)	94	122	138	200	210	215	323	355

Fig.9 Management system discovery mechanism

Additionally, according to the algorithm if adding a mechanism of management discovery mechanism for improving devices capability for interconnecting to the system. In that case, the third column value illustrated Fig.9, that management system discovery mechanism in a number of devices and time. The management system is regularly checking devices status and discovery possibility as well. Less spending time more useful to saving energy and battery usage.

CONCLUSION

The achievement of the paper relies on several results. The initial step this paper represented legacy constrained management system functional elements. There are four main components as allocated constrained IoT management system (see Fig.1). The U-NMS is complex to configuration system elements (manager and agent) which process of deployment. There are several problem statements and key factors of management system techniques for the IoUT. In such of the case, some problem statements. Not easy installing agent software to embedded devices and that is making a private uMIB also so complex. When a duplicate of the OID in a management system. By the more, one of the constrained environment is the underwater network. More originality and usability of the system is U-NMS. The U-NMS became reality (Fig.4) and represent technical specification each device. Indeed, this suggestion is concerning related techniques in constrained network and IoT management steps based on discovery algorithm. The architectural model of legacy NMS and a related technology IoUT integrated by the proxy agent. The main elements of management are MIB hierarchic tree and using the IANA PEN for uniqueness. In researching process, we are implementing U-NMS system to a real field.

ACKNOWLEDGMENT

The work is a part of the results of the research “Development of the wide-band underwater mobile communication systems” supported by Ministry of Oceans and Fisheries, Korea.

REFERENCES

[1] Sun P. Integrating Network Management For Cloud Computing Services – Princeton University, **2015**.

[2] Sheng Z. et al. Lightweight management of resource-constrained sensor devices in Internet-of-Things //IEEE Internet of Things Journal. – T. 2. – №. 5. – C. 402-411, **2015**.

[3] Mauro, Douglas, and Kevin Schmidt, ”Essential SNMP”, Help for System and Network Administrators. O'Reilly Media, Inc, **2009**.

[4] Urunov Khamdamboy, et al. "Analysis of the Network Management System with Constrained Underwater Devices." Proceedings of Symposium of the Korean Institute of Communications and Information Sciences: 500-501, **2017**.

[5] Yang S. H. Internet of things //Wireless Sensor Networks. – Springer London, – C. 247 -261, **2014**

[6] The case, Jeffrey D., et al. Simple network management protocol (SNMP). No. RFC 1157, **1990.**

[7] Case J. et al. Coexistence between version 1 & 2 of the Internet-standard Network Management Framework. –№. RFC 1452, **1993.**

[8] Case J. et al. Introduction and applicability statements for internet-standard management framework. – №. RFC 3410, **2002**.

[9] Enns R. et al. RFC-6241 //Network Configuration Protocol (NETCONF). – **2011**.

[10] Khamdamboy Urunov, Soo-Young Shin, Soo-Hyun Park and Kwan Yi “u-SNMP for the Internet of Underwater Things”. SERSC. International Journal of Control and Automation (IJCA)(pp. 199-216), October **2017**.

[11] Mukhtar, H., Kang-Myo, K., Chaudhry, S. A., Akbar, A. H., Ki-Hyung, K., & Yoo, S. W. LNMP-Management architecture for IPv6 based low-power wireless Personal Area Networks (6LoWPAN). In Network Operations and Management Symposium, 2008. NOMS 2008. IEEE (pp. 417-424). IEEE, 2008, April.

[12] Hillen B. A. G. et al. Remote management of mobile devices with broadband forum's TR-069 //Telecommunications Network Strategy and Planning Symposium. Networks 2008. The 13th International. – IEEE, 2008. – C. 1-7, **2008**.

[13] “PRIVATE ENTERPRISE NUMBERS” online, Available: https://www.iana.org/assignments/enterprise-numbers/enterprise-numbers, Febrary **2018**.

[14] Ge J. X., Xiao W. Y. Network layer network topology discovery algorithm research //Applied Mechanics and Materials. – Trans Tech Publications T. 380. – C. 1327-1332, **2013**.

[15] Salman T., Jain R. Networking Protocols and Standards for the Internet of Things //Internet of Things and Data Analytics Handbook. – 2015. – C. 215-238, **2015**.

[16] Kim H., Cho H. S. SOUNET: Self-organized underwater wireless sensor network //Sensors. – 2017. – T. 17. – №. 2. – C. 0283, **2017**.