A Policy Based Management Framework for Machine to Machine Networks and Services

Rossi Kamal, Muhammad Shoaib Siddiqui, Haw Rim and Choong Seon Hong
Department of Computer Engineering
Kyung Hee University
Email: (rossi,shoaib)@networking.khu.ac.kr and (rhaw,cshong)@khu.ac.kr

***Abstract*—With the penetration of smart-devices and intelligent home-appliances, M2M (machine to machine) communication has opened the door of new prospects for us. The success of such M2M networks depends on its human involvements at the least possible level. In this context, policy based management has huge potentials to be used for M2M communication. This paper presents such a policy based management framework for M2M networks and services. We have considered two M2M communication platforms namely smart-devices and sensor network in our system.**

I. Introduction

[1]

With the rapid advancement, technology is replacing direct human interaction with machine to machine (M2M) communication and humans are merely becoming the consumer of these systems. M2M network [1] has a high promising future. With advents of autonomous organization and management, conventional networks are fast replaced by machine-to-machine M2M networks. Penetration of the Internet enabled smart-devices is growing fast and as a consequence, we will see the widespread use of intelligent home-appliances in the future. Sensor network [2] has also same prospect as it is deployable in ubiquitous environment easily anywhere, anytime. One of the major characteristics behind the success of such M2M platforms is the least human involvement. With the penetration of smart-devices and intelligent home-appliances, M2M (machine to machine) communication has opened the door of new prospects for us. The success of such M2M networks depends on its human involvements at the least possible level. In this context, policy based management has huge potentials to be used for M2M communication. Policy based management [3] can be a better solution for the M2M networks as predefined policies in the system can response automatically to the M2M service requests. These policies are defined by system manager or a system expert. However, policies cannot be predefined for each and every event which can occur within a system. Therefore, a framework is required which monitors the M2M environment, analyzes its monitored data and represents it to the policy manager, along with aiding tools such as policy editor, to define and orchestrate new policies for the system. The policy compiler then compiles the new set of policies and integrates them into the system. This paper presents such a policy based management framework for M2M networks and services. We have considered two M2M communication platforms namely smart-devices and sensor network in our system. The major components for our policy based management framework for M2M networks and services are policy compiler, editor, adapter, analyzer and a repository. Sensor motes and smart-devices are our target M2M devices those are connected to the policy server by an adapter. We propose an event-condition-action based policy information model, as shown in Fig. 2. Policy specification languages are also presented along with the proposed framework. We also provide the implementation aspects of our system with an implementation scenario for the better understanding of the audience. The paper is articulated as follows. Section II provides the proposed policy based management framework with its major components. Section III discusses the implementation aspects along with an implementation scenario. Section IV provides some related work in the field of policy based network management and M2M network and Service management. Finally, section V concludes our paper.

[1]This research was supported by Future-based Technology Development Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education, Science and Technology (No.20100020728).
*Dr. CS Hong is the corresponding author

II. Proposed System

Policy based network management is a condition-action response mechanism to provide an automatic response to conditions in the network according to pre-defined policies. We have proposed such a policy based management framework for M2M networks.

A. Proposed Architecture

Proposed architecture (Fig. 1) consists of major modules namely policy compiler, editor, adapter, decision point, analyzer, controller and policy repository. An M2M policy information model runs behind the overall system. M2M specific policies are written in policy specification languages with XML and Java/C++.

B. M2M Policy Information Model

M2M policy information model (Fig. 2) is an event-condition-action based policy information model [3]. It is an extension of CIM policy model [4] . As in CIM, all policies are derived from ManagedElement. M2MPolicy defines a general policy in proposed policy based system. In this model,

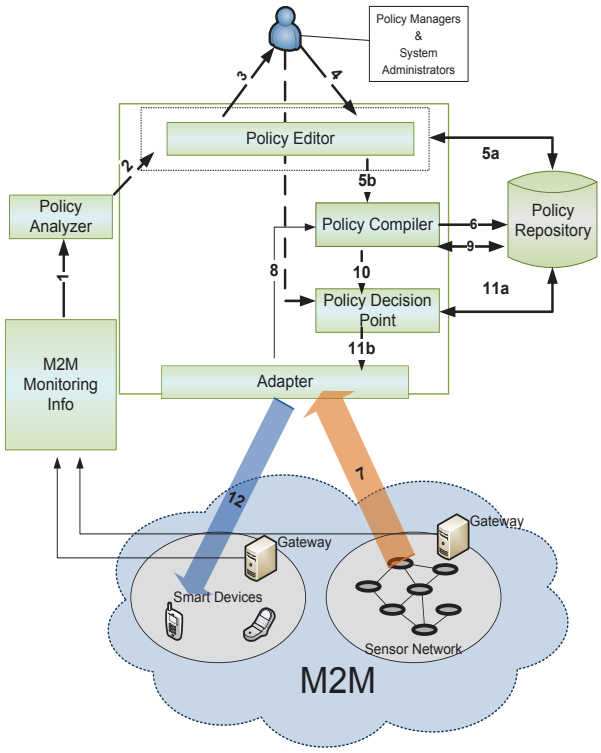

Fig. 1. Workflow in the proposed system

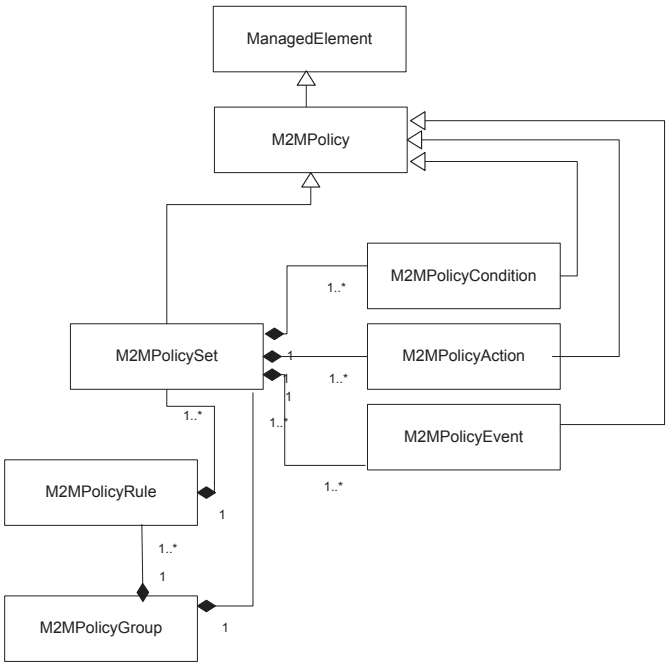

Fig. 2. Proposed M2M Policy Information Model

any policy is a combination of event, condition and action (represented by M2MPolicyEvent, M2MPolicyCondition and M2MPolicyAction respectively). Event, condition or action can be compound also. We can aggregate policies of the same type for different M2M network and service domains. M2MPolicyRule aggregates similar policy sets. Similarly, M2MpolicyGroup aggregates policy rules for similar M2M network or service domain.

C. Policy Specification Languages

Proposed system uses policy specification languages in different stages of policy compilation. Policies are written through the policy editor. It is then translated to XML (Fig. 3) format by a translation module. At last, it is interpreted to Java / C++ object by interpretation module.

D. Policy Editor

Policy editor sub-system is used to create, edit and update policies in the system. It consists of modules namely policy language definition, user viewing, policy evaluation and dictionary.

1) Policy Language Definition: Policy language definition module allows policy manager to define or edit policies.

2) User Viewing: Policy manager views existing policies and events through user viewing module. He also views M2M monitoring information through this module.

3) Policy Evaluation: With Policy evaluation module, a policy manager can evaluate performance of policies through this module. Policy manager can analyze defined policies and the outcome of it to evaluate policy performance.

Human Understandable Policies

Event	Condition			Action		
	HeartRate	Activity	Age	Status	Sample Rate	Call for
HeartRateVeryHigh	X > 170	Lying	Y > 40	Critical	1/10 sec	Intensive care
HeartRateHigh	100 < X <170	Lying	Y > 40	Prone	1/2 min	Precaution
HeartRateIncreasing	70 < X <100	Lying	Y > 40	Normal	1/5 min	n/a
HeartRateHigh	100 < X <170	Running	Y < 40	Normal	1/10 min	n/a
HeartRateIncreasing	70 < X <100	Running	Y > 40	Prone	½ min	Precaution

Intermediate language (XML)

```
< event name="HeartRateHigh"></event>
< condition>
   <condition_argument1 >if Heart rate &gt;100 && lt;170 then
   </condition_arguement1>
   <condition_argument2 name="Activity" value="Lying">
   </condition_arguement2>
   <condition_argument3 >if Age &gt;40 then
   </condition_arguement3>
</condition>
< action>
   <action_argument1 name="Status" value="Prone">
   </action_arguement1>
   <action_argument2 name="Sample Rate" value="1/2 min">
   </action_arguement1>
   <action_argument3 name="Call for " value="Precaution" >
   </action_arguement3>
</action>
```

Fig. 3. Policy specification languages in proposed system. Policy is translated to an intermediate language (XML) format. At last, XML is interpreted to low level language (Java).

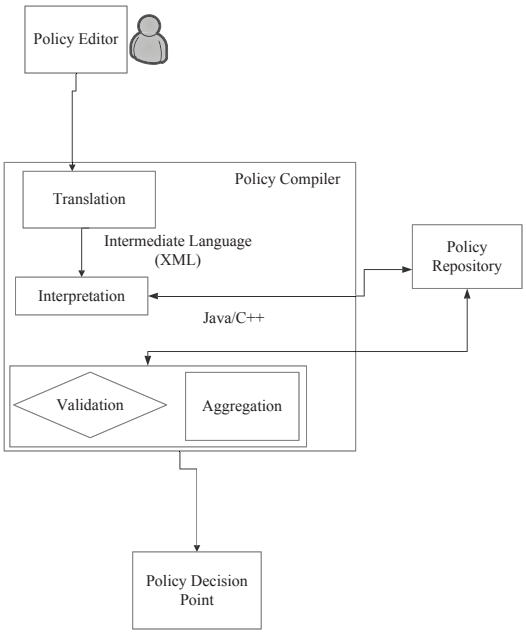

Fig. 4. Translation, Interpretation, Validation and Aggregation in the Policy Compiler

4) Dictionary: Dictionary works as a help file for policy manager. It also provides keywords for policy creation and editing

E. Policy Compiler

Policy compiler (Fig. 4) consists of sub-modules namely translation, interpretation, validation and aggregation.

1) Translation: Translation module converts high level language to intermediate level language. It takes input from policy editor and gives output to policy interpretation module.

2) Interpretation: Interpretation module converts intermediate language to low level language. It takes input from policy translation module and gives output to validation module.

3) Validation: Validation module helps whether defined policy is semantically valid or not.

4) Aggregation: This module combines existing policies to create a compound policy.It also combines policies in a group and saves it to repository.

F. Policy Repository

Policy repository preserves the policies, conditions, actions or events in its database. It also stores M2M network and service information. A policy compiler stores newly created policies in policy repository. Compiler also uses existing policies (from policy repository)at the time of policy aggregation and validation. When M2M device wants access to the policy based system, its authentication and service information are checked at policy repository. If it is a valid M2M network domain and its requested service is available, corresponding policies are enforced on it.

G. Adapter

Adapter provides abstraction for the physical devices attached to the M2M network and the policy server. Heterogeneous devices such as different sensor networks (nodes and gateway and smart devices (smart-phone, smart-tab) etc. are connected to the policy server. Each device provides different interfacing mechanism; adapter is responsible to provide communication among all the devices with the policy server.

H. Policy Decision Point

PDP is a logical entity that makes policy decisions for the network elements specifying which policies should be deployed at which points. Policy decision point gives compiled policies to M2M device through an adapter. It takes input from policy compiler and also connects with policy repository to decide on a M2M service-instance.

I. Policy Enforcement Point

PEP is the logical entity which executes the policy at the network element and verifies that a set of policies and its actions have been successfully performed on specific targets.

J. Policy Controller

Policy controller is responsible for coordinating the actions of the policy server. It coordinates between the internal components of the policy server. The coordination between PDP and PEPs is also managed by the policy controller.

K. Policy Analyzer

Policy Analyzer is used to perform analysis on the monitored data and feedbacks to analyze the effects of certain policies. It provides the evaluation of the policies and help in creating better policies.In the future, it is targeted to provide autonomic policy updating using decision support systems and expert systems.

III. Implementation

A. Functionality

Our policy based management system works for two M2M domains (sensor network and smart-devices). Fig.1 represents overall workflow in the proposed system. M2M monitoring information about available sensors or smart-device is sent to policy analyzer (step 1). Policy analyzer is connected to the policy editor (step 2). Then, a policy manager through user viewing module monitors current M2M network status(step 3). He then creates policies (step 4) for that specific M2M network or service domain. Policies are translated and interpreted in the policy compiler (step 5b) with the interaction of policy repository (step 5a). Written policies are stored in the policy repository (step 6).

Sensor nodes /smart devices send sensed information and service request to the proposed system through adapter(step 7). Adapter then contacts with policy compiler (step 8). Policy compiler then checks what policies are available for that specific M2M network and service domain (step 9). It then deploys appropriate policy through policy decision point (step

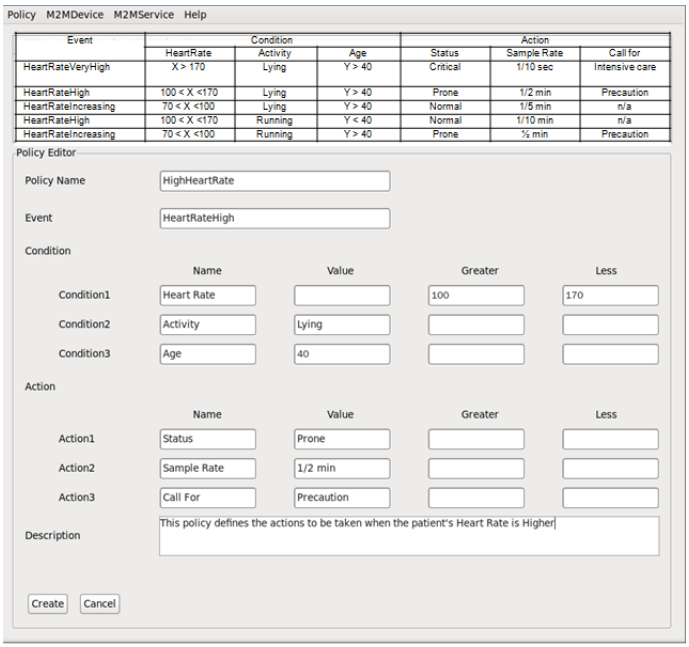

Fig. 5. Screenshot of the policy editor

10) to the policy adapter (step 11). Sensor nodes/ smart devices get the requested service through adapter (step 12).

B. Implementation Scenario

We have implemented our proposed based management framework in a ubiquitous healthcare scenario. Depending on the M2M monitoring information (from available smart-devices, wearable sensor nodes and their network traffic), policies are written and preserved in policy repository of the system. Wearable sensor nodes at patients send sensed information through the adapter to the system. The system immediately responses with appropriate actions (according to corresponding policies). This action reaches the smartphone of remote nurse who is in charge of that patient.

C. Development Environment

Proposed system is developed in C++/Java in Linux operating system. Policy specification languages are defined in XML, Java and C++. Policy repository is developed with MySQL database. nesC over TinyOS is used in sensor motes. Android SDK (version 2.5) is used in smart-phones. Fig. 8 shows the screenshot of developed policy editor in our system. Policy manager writes a policy (policy name and definition, event name, condition name and parameters, action name and parameters) through this policy editor.

IV. Related Works

Research efforts have been made for the development of policy based management systems and M2M networks separately. Ponder2 [5] provides the policy based approach for autonomous management of body sensor network. It uses SMC (self managed cell) that consists of an autonomous set of hardware and software components representing an administrative domain. It focuses on ubiquitous system with only body sensor networks. Ponder2 uses an event-condition-action based policy information model. PonderTalk language has been considered as a standard high level policy specification language. Connected objects [6] solves M2M application development challenges (variations in device capabilities, scalability and flexibility etc). This M2M service platform provides application programming interfaces (APIs) for various M2M resources namely data aggregation, security handler, context information, etc. It only considers enterprise application domain. But, its object and edge are conceptually similar to our M2M network and service domain respectively.

Research on policy information model is a key issue in policy based network management. Policy core information model (PCIM) [7] existed several years for network oriented policies like network access control or traffic differentiate etc. CIM [8] policy model was influenced by PCIM and its extension PCIM-E. CIM policy model was condition-action based. But, in M2M networks, policy evaluation is done asynchronously for the occurence of external or internal events. Neither PCIM or CIM policy model has support for event based M2M networks.

V. Conclusion and Future Works

We have proposed a policy based management framework for two M2M networks with smart-devices and sensor nodes. Proposed M2M policy information model runs behind the overall functionality of this system. We have considered our system in a ubiquitous healthcare scenario with a patient with biosensors and a remote nurse with a smart-phone.

References

[1] Geng Wu; Talwar, S.; Johnsson, K.; Himayat, N.; Johnson, K.D., *M2M: From mobile to embedded internet*, Communications Magazine, IEEE , vol.49, no.4, pp.36-43, April 2011.

[2] Akyildiz, I.F.; Weilian Su; Sankarasubramaniam, Y.; Cayirci, E., *survey on sensor networks*, Communications Magazine, IEEE , vol.40, no.8, pp. 102- 114, Aug 2002

[3] Strassner, J.C, *Policy-Based Network Management- Solution for the Next Generation*, Elsevier, 2004

[4] Westerinen, A.; Schott, J., *Implementation of the CIM Policy Model using PONDER*, Distributed Systems and Networks, 2004. POLICY 2004. Proceedings. Fifth IEEE International Workshop on , vol., no., pp. 207-210, 7-9 June 2004.

[5] Twidle, K.; Lupu, E.; Dulay, N.; Sloman, M., *Ponder2 - A Policy Environment for Autonomous Pervasive Systems," Policies for Distributed Systems and Networks*, Policies for Distributed Systems and Networks, 2008. POLICY 2008. IEEE Workshop on , vol., no., pp.245-246, 2-4 June 2008.

[6] Herstad, A.; Nersveen, E.; Samset, H.; Storsveen, A.; Svaet, S.; Husa, K.E., *Connected objects: Building a service platform for M2M*, Intelligence in Next Generation Networks, 2009. ICIN 2009. 13th International Conference on , vol., no., pp.1-4, 26-29 Oct. 2009.

[7] B. Moore, E. Elleson, J. Strasser, A. Weterinen, *Policy Core Information Model*, IETF RFC 3060, February 2001

[8] Distributed Management Task Force (DMTF), *Common Information Model (CIM) Specification*, URL:http://www.dmtf.org.

[9] R. Kamal, M. S. Siddiqui, R. Haw and C. S. Hong, *Towards Policy Compiler and Policy Editor for M2M Networks & Services*, The 21th Joint Conference on Communications,May, 2011