Flexible Configuration of IP Networks Through Policies and Mobile Agents

Kun Yang†, Alex Galis†, Kerry Jean† and Telma Mota‡

† University College London, Department of Electronic and Electrical Engineering, Torrington Place London WC1E 7JE UK
{kyang | agalis | kjean}@ee.ucl.ac.uk

‡ Portugal Telecomm Inovacao, SA. Rua Eng. José Ferreira Pinto Basto 3810-106 AVEIRO, Portugal
Email: telma@ptinovacao.pt

Abstract: With the magnificent expansion of network, it is getting more imperative to seek a means to deploy the network management tasks in a fast, unique and automated way. This paper proposes to use the integration of mobile agent technology (MAT) and Policy-based Network Management (PBNM) to achieve this flexibility and automation. PBNM, as a newly introduced but widely welcomed technology in the Internet world, can take over the overall management of IP network with MAT enabling the implementation of PBNM system. The work presented in this paper has been developed in the framework of the EU IST Project MANTRIP. A scenario for solving a practical network management challenge, i.e., inter-domain IP Virtual Private Network, is implemented, which shows how the integration of mobile agent technology and PBNM could be used to overcome many management problems inherent in traditional management approaches.

1. Introduction

With the magnificent expansion of network, both in the types of network elements and the software components to manage them, driven by the increasing requirement of different services, it is getting more imperative to seek a means that can deploy network management tasks in a fast, ubiquitous and automated way. A great deal of effort has been made, among which, CORBA, SOAP (Simple Object Access Protocol) and COPS (Common Open Policy Services) are widely accepted and are currently used in some of commercial products. But all of these solutions are based on traditionally client/server model therefore, at least theoretically, lack flexibility and have lower performance. Whereas *Mobile Agent Technology (MAT)*, typical representative of mobile code technology, provides a more promising means to achieve this goal.

The mobile agent paradigm [1] intends to bring an increased performance and flexibility to distributed systems by promoting "autonomous code migration" (mobile code moving between places) instead of traditional RPC (remote procedure call). With code migration, the actual code or script moves from place to place and executes locally, achieving lower latency, little need for remote interactions and highly flexible control. Mobile agents can easily represent one of the roles involved in the network management, such as service provider, connectivity provider, resource or end-user, and act on their behalf, based on established policies.

PBNM technology is very suitable for setting up the overall management architecture for large-scaled networks [2]. In comparison with previous traditional network management approaches such as TMN or TINA-C [3], PBNM offers a more flexible management solution for a specific application tailored to a customer. Nevertheless, the current PBNM architecture can only address issues that can be translated to fixed configuration settings and complex interactions cannot be easily implemented in the current PBNM architecture. For example, QoS issues often needs complex interactions between relevant network management components. Moreover, according to the policy framework, policies are defined or modified by an administrative tool and the intervention of human is always required. These features of current PBNM system greatly confine its wider and ubiquitous application. MAT can solve many of the problems inherent in current PBNM. Therefore the integration of MAT and PBNM is used for the novel network management.

The paper is organized as follows. Based on the analysis of this section, Section 2 presents the fully MAT-supported IP network management architecture designed within MANTRIP [4]. Section 3 details the design and implementation of a generic network management architecture that are fundamentally based on policy management and technically enabled by mobile agents. The features covered in this management architecture is further integrated and verified in a practical IP network management scenario, inter-domain IP VPN (Virtual Private Network) configuration, which contributes to Section 4. Section 5 concludes the whole paper.

2. MANTRIP Network Management Architecture

This paper aims to present the work done in the QoS Configuration and Auditing Application, which architecture is depicted in Figure 1, following the overall MANTRIP Network Management System (NMS) architecture, although only the components and resources used for QoS and IP VPN are shown.

The major objective of the MANTRIP project resides in providing a set of novel management applications for managing IP based networks. Those applications are aimed to bringing management intelligence to the managed

resources through *Mobile Agent Technology (MAT*). Amongst a long list of mobile agent platforms developed, Grasshopper [5] is most popular and is adopted in this paper. Grasshopper has been designed in conformance with the first mobile agent industry standard given by OMG (Object Management Group), namely the Mobile Agent System Interoperability Facility (*MASIF*) [6].

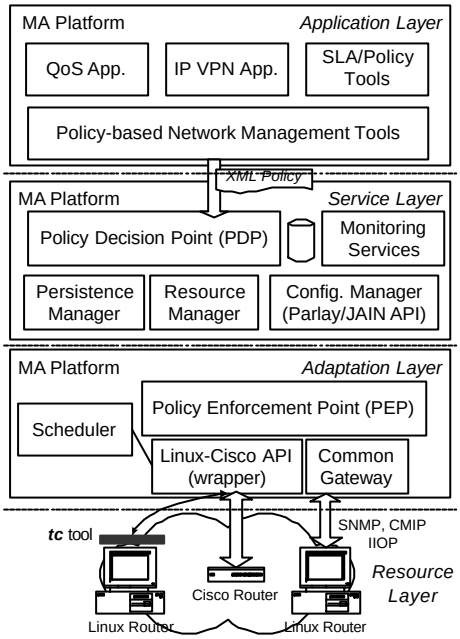

Figure 1: Policy and MA based MANTRIP NMS Architecture

The MANTRIP NMS has four layers as follows. *Application layer* includes the MANTRIP management user applications. *Service layer* contains the MANTRIP management services (e.g. Parlay/JAIN API) that may be used by either the MANTRIP applications or some other third party applications. *Adaptation layer* is responsible for hiding the protocol details from the service layer. It contains the protocol adapters and/or the resource wrappers. *Resource layer* contains the managed/controlled MANTRIP resources.

As shown in Figure 1, the components of PBNM system spread in different layers and contribute the core of this management architecture.

3. PBNM System Architecture enabled by Mobile Agent Technology

PBNM technology is very suitable for setting up the overall management architecture for large-scaled networks. In order to deploy PBNM technology, a standardization process should be followed to ensure the interoperability between equipment from different vendors and, furthermore, PBNM systems themselves from different developers. Both the Internet Engineering Task Force (IETF) [7] and the Distributed Management Task Force (DMTF) [8] are currently working for the definition of standards for Policy Based Network Management. The work done by IETF is more related to the network management and the IETF policy architecture has covered almost all the components required by a PBNM system, therefore, the IETF policy architecture gains more popularity and is adopted in MANTRIP NMS. The architecture and its components are shown in Figure 2, which provides detailed design of Policy Framework proposed by IETF.

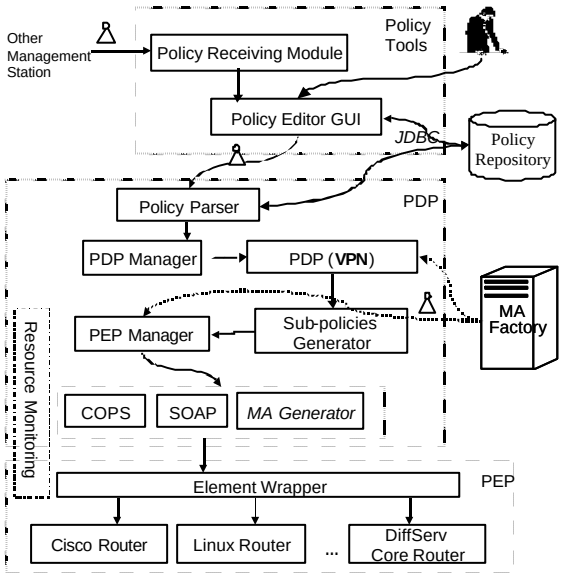

Figure 2: PBNM Architecture enabled by Mobile Agents

3.1 Policy Tools

Policy Tools provide the mechanism to receive policies from other management stations. With these tools the administrator can also define new policies for the system so as to guide the behaviour of agents, edit existing ones, or simply view the policies that exist in the policy repository. The input of policies may be done in a high level language or in a visualisation way, which offers abstractions to the administrator. But the specification of policies is in XML (eXtensible Mark-up Language). In this case, the tool also performs a translation of these high level policies to a set of policy representation that can be interpreted by PDP (Policy Decision Point). The extension of policy core information model (PCIM) defined by IETF, in the form of schema if XML is used, is used to express the syntax of policy.

Policy Tools mainly include Policy Receiving Module and Policy Editor GUI. *Policy Receiving Module* is implemented as a stationary agent for receiving XML-based policies transported by mobile agent. These policies are given by upper management station. Policies can also be inputted directly by network administrator with the assistance of Policy Editor GUI. *Policy Editor GUI* provides a user-friendly way for administrator to input some simple information such as source and destination host names, select the QoS parameter required by user such as gold or sliver or bronze, etc., then it can generate XML-based policy automatically and store it into the Policy Repository.

3.2 Policy Repository

The *policy repository* is used for the storage of policies, after they have been defined and validated by the policy management tool. The general framework of IETF does not require a specific implementation for the policy repository, or the repository access protocol. In this paper, relational database management system, *PostgreSQL*, is used for policy database, which is connected to Policy Tools and Policy Decision Point (PDP) via JDBC.

3.3 Policy Decision Point (PDP)

PDP is the component that retrieves policies from repository, parses them thus evaluates them and eventually sends the necessary commands to the policy target. Additionally, the PDP performs a local conflict check, checking only those devices that are controlled by the specific PDP. The PDP also checks if the resources needed for a specific policy are available in all the controlled devices.

The main role of *PDP manager* is to co-ordinate the different PDPs to support integrated scenarios, and resolve possible conflict. PDP manager can also dynamically download PDP code according to the availability of the code. PDP manager also serves as the coordinator of PDP and Policy Parser. It uses Policy Parser to read policy from policy DB and to parse policy with the help of XML parser. The existence of PDP manager makes the whole policy management system extensible to contain other future PDP.

PDP Module together with *Sub-policies Generator* can translate higher-level policy into sub-domain level policies, with the information from monitoring service. After receiving the policies in XML file, the PDP Module extracts from the XML file the sub-domain level policies. Then, it needs to decide when this policy should be applied by looking at the conditions of the policy thus deciding whether it needs any information to make a decision. If so, it will ask Monitoring Service to register the condition to be monitored. Otherwise, it asks the Resource Monitoring Service module if there are enough resources to apply this policy. If the answer is positive, the policy will be passed to PEP Manager Module to be fulfilled. All the policies are based on fixed schema, so that they are understandable by different levels.

The *Resource Monitoring* module receives the registration of resource monitoring according to the requirement of policies and make sure that all the resources registered can be monitored. If the necessary metering code (daemon) is not currently instantiated, the resource monitoring service module will try to make a query to a specific resource monitoring code base so that the corresponding monitoring daemon code, in the form of mobile agent, can be downloaded and run itself. Monitoring Service component just enquiries the Resource Monitoring Daemon to get the necessary information. Resource monitoring functionality also exists in PEP to monitor the enforcement result of the policy.

According to the given policy, *PEP Manager* can select which kind of PEP enforcement protocols, such as COPS (Common Open Policy Service), SOAP (Simple Object Access Protocol), together with its parameters to be used to fulfil the policy. Due to the inherent limitation of Client/Server structure used by both COPS and SOAP, this paper evaluates the use of mobile agent technology for enforcing policy in order to gain more flexibility and automation. But COPS and SOAP mechanisms are also implemented to make the policy enforcement scale to more network elements.

Based on the parameters given by PEP Manager, *Mobile Agent Generator* can automatically generate the corresponding mobile agents, which can migrate themselves to the specific PEPs to enforce the policy.

3.4 Policy Enforcement Point (PEP)

The policy target is the managed device, where the policy is finally enforced. There is a need of a transport protocol for communication between PDP and PEP, so that administrator can send policy rules or configuration information to the target, or read configuration and state information from the device. There is no specific requirement of protocol for this operation. Again, mobile agents take the responsibility for policy enforcement. In order for mobile agents to communicate with different controlled elements, such as Linux routers, Cisco Routers or other computing resources, the corresponding element wrappers are provided, in the form of stationary agents.

4. Case Study: Inter-domain IP VPN using Mobile Agent enabled PBNM

Based on the policy and mobile agent based network management architecture given above, this section presents a case study to evaluate this architecture, i.e., inter-domain IP VPN provisioning guided by policies and fulfilled by mobile agents. Inter-domain communication is also a challenging research field in network management. This paper provides, as a case study, a solution to inter-domain communication by introducing the integration of mobile agent technology and policy-based network management technology. Mobile agent plays a very important role since the most essential components in PBNM, such as PDP and PEP, are in the form of mobile agents that can move themselves to the required place dynamically and begin work with guidance from policies. Other non-movable components in PBNM architecture, such as policy receiving module, are in the form of

stationary agents waiting for the communication with coming mobile agents. Mobile agents are also responsible for transporting XML-based policy across multiple domains.

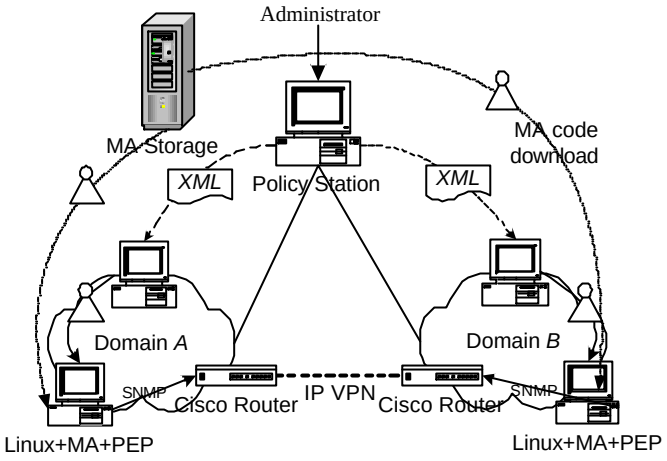

Figure 3 Inter-domain IP VPN using MA-enabled PBNM

This scenario is implemented in the test-bed provided by Portugal Telecomm with its own products inside, aiming to enable full use of mobile agent technology in real commercial world, as it is also one of important motivation of MANTRIP project.

The whole scenario is depicted in Figure 3. Network administrator uses Policy Management Station to manage the underlying network environment (including two domains with one physical router and one Linux machine next to Cisco router at each domain) by giving policies, which are further translated into XML files and transported to relevant sub-domain PBNM stations using mobile agents. In this scenario, two mobile agents are generated at the same time, each going to one domain. Let's take one mobile agent for example. After the mobile agent arrives at the sub-domain management station, the mobile agent communicates with the stationary agent waiting at the sub-domain management station. Based on the policy carried by this mobile agent, the sub-domain PDP manager can download the proper PDP, which is in the form of mobile agent, to make the policy decision. After this, the selected or/and generated policies are handed to PEP manager, which requires the availability of PEP code, e.g., for new IP tunnel configuring, according to the requirement given in XML file. The PEP, also in the form of mobile agent, moves itself to the Linux machine, on which it uses SNMP (Simple Network Management Protocol) to configure the physical router to set up one end of IP VPN tunnel. Same process takes place at the other domain to bring up the other end of the tunnel; therefore the whole tunnel is set up.

5. Conclusions

As shown in the above case study, after administrator inputted the requirement, the whole procedure processed automatically. Administrator doesn't need to care about the information specific to the sub-domain. The whole system scales to the changing of the application requirement automatically thanks to the inherent mobility of mobile agent and its intelligence empowered by policies. The scalability of this network management system also enables itself to be seamlessly integrated into other network management system.

The combination of PBNM and MAT enables the achievement of many advantages, such as, the automated and rapid deployment of new services, the customisation of existing service features, the scalability and cost reduction in network and service management. All these mean a ubiquitous network management architecture that will further promote the appearance of new services and business opportunities.

Acknowledgements

This paper describes part of the work undertaken in the context of the EU IST projects MANTRIP. The IST programme is partially funded by the Commission of the European Union.

References

1. A. Bieszczad, T. White and B. Pagurek. Mobile Agents for Network Management. IEEE Communications Surveys, 1998
2. N. Damianou, N. Dulay, E. Lupu, and M Sloman. The Ponder Specification Language. Workshop on Policies for Distributed Systems and Networks (Policy2001), HP Labs Bristol, 29-31 Jan 2001.
3. M. Hall, G. Nilson and P. Prozeller. Management Architecture Specification. TINA-C (www.tinac.com), Dec., 1993.
4. MANTRIP project website: http://www.solinet.com/mantrip
5. Grasshopper web site: http://www.grasshopper.de
6. D. Milojicic et al., "MASIF: The OMG mobile agent system interoperability facility" in Proc. 2nd Int. Workshop Mobile Agents: Springer-Verlag, Lecture Notes in Computer Science, vol. 1477, Sept. 1998.
7. IETF Policy workgroup, http://www.ietf.org/html.charters/policy-charter.html
8. Distributed Management Task Force (DMTF), www.dmtf.org

Routing Policy Configuration. Overview of Routing Policies. Licensing Requirements and Limitations for Routing Policy. Configuring an IP Prefix List.Â To improve BGP network security, you can configure BGP authentication and GTSM on the BGP network. Usage Scenario. By performing authentication for BGP peer connections and configuring BGP GTSM, you can improve BGP network security.Â GTSM prevents attacks through TTL detection. An attacker simulates real BGP packets and sends the packets in a large quantity to the router. After receiving the packets, an interface board of the router directly sends the packets to the BGP module of the control plane if the interface board finds that the packets are sent by the local router, without checking the validity of the packets. Mobile IP is a communication protocol (created by extending Internet Protocol, IP) that allows the users to move from one network to another with the same IP address. It ensures that the communication will continue without userâ€™s sessions or connections being dropped. Terminologies: Mobile Node (MN): It is the hand-held communication device that the user caries e.g. Cell phone. Home Network: It is a network to which the mobile node originally belongs to as per its assigned IP address (home address). Home Agent (HA): It is a router in home network to which the mobile node was originally connected. Home Address: It is the permanent IP address assigned to the mobile node (within its home network). Mobile Agent Network Management Policy Decision Point Simple Network Management Protocol Policy Enforcement Point. These keywords were added by machine and not by the authors. This process is experimental and the keywords may be updated as the learning algorithm improves.Â Yang K., Galis A., Guo X., Liu D. (2003) Rule-Driven Mobile Intelligent Agents for Real-Time Configuration of IP Networks. In: Palade V., Howlett R.J., Jain L. (eds) Knowledge-Based Intelligent Information and Engineering Systems. KES 2003. The cellular mobile network has evolved so much the last decades with improved coverage, speed and reliability. Therefore it is now possible to use the.Â Therefore it is now possible to use the 3G/4G cellular mobile network as a reliable backup-up connection of your main connection line. Cisco has several router devices that have either an embedded 3G/4G modem or a standalone HWIC (High-speed Wan Interface Card) that that can be attached to a modular router. The Mobile IP Home Agent Policy Routing feature allows policy routing for mobile nodes based on the NAI configuration. ISPs can use this feature to route traffic originating from different sets of users, as identified by the NAI realm name, through different Internet connections across the policy routers. When the mobile node registers, entries are added dynamically in the access list pointed to by the route map and the route map is applied to the tunnel interface.