I.J. Education and Management Engineering **2012, 7, 50-60**
Published Online July 2012 in MECS (http://www.mecs-press.net)
DOI: 10.5815/ijeme.2012.07.08

Available online at http://www.mecs-press.net/ijeme

Policy Model in the Desktop Management System

Zhao Fang[a], Liu Yin[b]

[a] *College of Information Science and Technology, Beijing Forestry University, Beijing, China*
[b] *College of Information Science and Technology, Beijing Forestry University, Beijing, China*

Abstract

By studying the policy and desktop management systems theories, referencing the Internet Engineering Task Force (IETF) policy model, this paper proposed a policy model that can be applied in specific desktop management system. It mainly explains the whole system framework and its implementation mechanisms, and it discusses the problems and solutions that the policy model uses in the desktop management system.

Index Terms: Policy; Policy Model; Policy-Based Desktop Management; Access Control

1. Introduction

With the rapid development of computer technology and widely used Internet applications, there have been an increasing number of computer equipments in the Enterprises. These computers usually run a variety of applications. Therefore, their stability, reliability, security and performance not only directly affect the efficiency of operations, but also have an important influence on both business efficiency and the administration cost of the whole enterprise. In addition, with the refinement of enterprise sectors, different department has their own requirements for desktop environment, in order to ensure the security of internal resources, classified restriction for authority to access the content of each local computer is needed. Such as stopping clients from any access to system volume, closing the USB port and CD-ROM drive for users of that department. The above problems can be solved with the help of a crucial content from the "Desktop Management System"---the access control of local resource, it will enable each enterprise to have a different desktop environment that meet their own needs.

Policy is the aggregation of a group of regulation used to describe the operation can be executed under which cases. With the fast development of network, Policy-based "network management" theory has been proposed. It can realize the demands on the network management system [1-2]. This policy can be dynamically changed, which can dynamically change the system's behavior and strategies. The Internet Engineering Task Force (IETF) and other organizations for standardization, research institutions and network equipment manufacturers have begun to engage in Policy management research and have published the relevant drafts [3-

Corresponding author:
E-mail address: [a]fangzhao@bjfu.edu.cn ; [b]Liuyin.tiger@gmail.com

4]. Although IETF Policy Management Model is now a commonly used Policy Management Model (PMM), the introduction of Policy and policy model to the desktop management system will bring some new problems.

2. Related Studies

Desktop management systems and Policy have made some achievements in their respective areas.

- Chen Zhifeng presents a Web-based desktop management system, in which they use a one-server model to manage multiple computers [5]. It describes and analyzes the system design and implementation techniques. However, due to the lack of the introduction of Policy and policy model in the system, it can only manage the desktop in accordance with the fixed rules therefore his system can not cope with the dynamic changes on the management rules.
- Li Jinping and Gao Dongjie proposed a policy-based software platform for network management system [6]. They use XML language to define, store and access the policy rules and uses CORBA technology to achieve distributed network management.
- Li Qinghai and Zhang Deyun proposed a distributed policy of role-based management practices [7]. The paper proposed high-level role-based policy management practices based on the IETF policy framework. Its domain concept enables the combination of the roles and domains, and upon that, the expression of the role is simplified. However, it is the policy model used in the network management system; improvements need to be made if wanting to apply the policy mode to the desktop management system.

In conclusion, the Related Studies are mainly on the policy model or on the desktop management system. Others are to combine together the policy model and the network management. However, researches related to the policy model applied to the desktop management system to solve problems caused by the introduction of policy model remains untouched upon.

3. Basic policy Model

3.1. Definition of policy

Generally speaking, the definition of policy mainly has the following forms:

- RFC3198 defines the policy with two compatible viewpoints [8]: The policy is the target, path or method of behavior, used for guiding or deciding current and future decisions. The policy is a set of rules, used for managing and controlling the access of resources.
- The policy is the norm used to define system behavior rule. It comes from management goals and it is persistent and descriptive [9].
- The policy is the behavior that influences subject and target. It describes the relation of subject and target [10].

3.2. *IETF Policy Management Model*

The Policy Management Model [10] that IETF puts forward is extensively accepted; most policy-based network management systems are based on it.

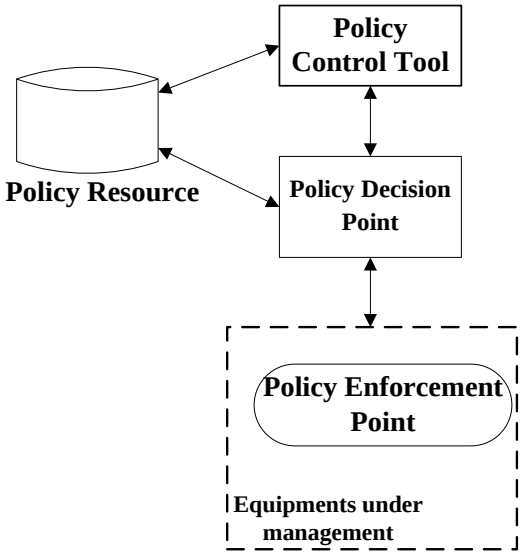

Fig. 1. The IETF Policy Management Model

The Fig. 1 shows that the Policy Management Model which IETF proposed should have a minimum of four basic functional components:

1) Policy control tool

The policy control tool is an interface for administrators to edit the policy. It is able to refine the natural language form entered by administrator to many policy rules and put the policies into the database by appropriate format. The policy control tool is also used to combine the decision policy equipment and the equipment which apply the policies. Furthermore, it supervises and controls the operation of the whole system. In addition, it provides a simple verification mechanism that examines the latent policy conflicts.

2) Policy Resource

Policy resource is a store facility which saves policy and other related information, such as relational database and directory service.

3) Policy Decision Point (PDP)

The Policy Decision Point is a logical decision-making entity; decisions are made according to the policy rules and status of net service. It explains, launches and executes polices and rules. For instance, it can receive requests and the policy conditions sent by Policy Enforcement Point (PEP) and find a matching policy in the policy resource, and then returns the result or action to the PEP.

4) Policy Enforcement Point (PEP)

The Policy Enforcement Point is a logical entity which enforces policy decisions. It is responsible for sending policy decision requests to PDP and providing policy conditions; and translating the returned decisions to operating configuration orders relevant to the specific equipment for execution.

3.3. The execution process of IETF PolicyManagement Model

1) Stage of the development policy

First, the administrator enters the policies by using policy control tool. Second, the policy control tool refines the policies to specific rules. Finally, put these rules into the policy resource.

2) Stage of implementation policy

A request will be sent to PDP when the PEP monitored an event. Then, the PDP seek from policy resource and determine whether there is a policy for this event. If so, deciding how to respond. And send back to PEP. PEP receives and executes the decision and responds to the event in the end.

3.4. *The deficiencies of IETF Policy Management Model used in the desktop management system*

1) The problem of Policy description

Policy expressions have the form as if<conditions> then<actions>. But IETF Policy Management Model has not determined a specific way to describe the policy. The difference of selected policy representation and storage leads to the failure of realization on policy management in large-scale.

2) The problem of policy conflict

When administrator edits the policy, a number of different policies established on the same resource because of man made error, and these policies provide the resources with different and even conflicting requirements of access control. That leads to the problem of policy conflict.

3) The problem of net interruption

Because of the dissimilarity of desktop management and network management, the network management relies on the network; the paralysis of network will definitely lead to the paralysis of the entire management system. However the desktop management is different, when the network run into an interrupt, the desktop terminal is able to continue to work, and manage its respective processes and events by the policy. For instance, a certain department of a company has no permission to access the system disk. This policy is stored in the policy resource online. It needs PEP (where the computer terminal) to submit the request on the network, and transmit the final decision from PDP to the PEP on net. And the occurrence of network interruption will make desktop management system unable to respond to events based on the policy. That will enable the personnel in that department to operate the computer without going through the limit of policy and results in the failure of desktop management in the end.

4) The problem of transmission speed

If we choose to use the IETF Policy Management Model based on the request / response distribution model, the interaction of PEP and PDP would fully relying on the network. Then, one time PEP request, from the beginning to the end requires several times of transmission online. In the desktop management, however, high speed network transmission is required for its responding mainly to the event on the desktop. For instance, the user would have no access to the system disk if there is no authorization policy. But the policy is from network. At this point, if the network transmission speed is slow, it will greatly affect the user experience.

4. Policy Model in the Desktop Management System

4.1. Frame the way of policy expression

1) Policy

Policy is consisted of user or user group, rules, IP restrictions and time limits. It completely defines the access controlling requirements of the resource, namely the definition of "which person" (user / user group) "under what conditions" (time limit, IP restrictions) "allow / ban access to what resources" (rules).

Policy = Users + Rules + Conditions {include IP restrictions, Time limits, etc.}

2) Rule

The rules define access and control rights of specific resources. A complete rule is consisted of Security Domain, Resource, Action and Time limits.

Rule= Security Domain + Resource + Action + Time limits

3) Policy Domain

Policy domain is a logical division for the management task of security resources. It is a container which can put user, user group, rule, security domain and policy into it. Each policy domain can include several security domains and each policy domain can have administrator itself.

4) Security Domain

The security domain is a set of designated resources. It is usually an aggregation of sources which has the same certification requirements and access control needs.

4.2. *Make priority principles of policy*

In order to improve the problem of policy conflict, a series of priority principles are made, to ensure correct execution of the policy.

1) Unbound protection principle

For Security Domain or Rule that is not bound to user, user group, we explained that as unfinished work of administrator's configuration. In such case, when administrator's safety intent is not clear, for security reasons, we deny all the people to access these resources which are under the protection of Security Domain or the activated rules.

2) Rejection first principle

For the same resource, the rejected rules for access are prior to the allowed rules. For example, there are two rules in the policy and each of them has set the permissions to the same resource. One allows the user to access this resource. The other denies it. In this case, the access operation will be rejected according to the Rejection first principle.

3) Accurate match first principle

When there are more than one rules relating to the resources which is requested, the most accurate matching rule effect first. For example, existing Rule1 bans users from accessing the path "C:*.exe". However, Rule2 permits users to access the path "C:\Documents and Settings\1.exe" In this case, the path "C:\Documents and Settings\1.exe" can be accessed for it's the accurate match.

4) primacy of priority principles

The descending order of policy priority is that:

- Unbound Security Domain protection principle.
- Unbound rule protection principle.
- Rejection first principle.
- Accurate match first principle.

4.3. The improved policy model in the desktop management system

As mentioned above, flaws of IETF Policy Management Model in the desktop management system are found. In this paper, based on the basic model, we made necessary improvement for its shortage and designed the improved model in the desktop management system. Fig. 2

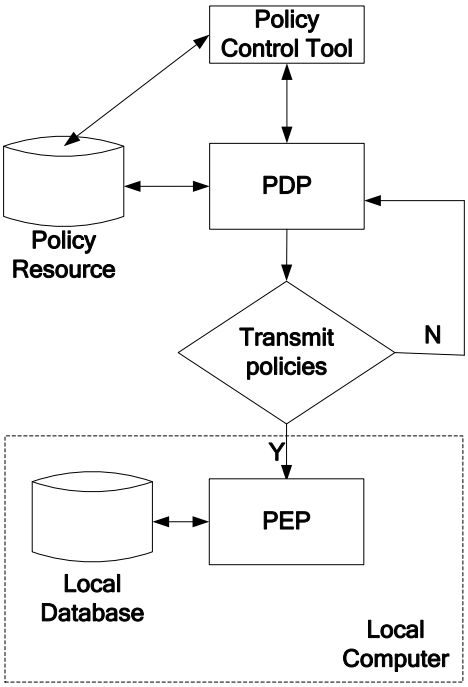

Fig. 2. The improved policy model in the desktop management system

Compared with the IETF Policy Management Model, we added a local policy resource database. There are two databases. One is the policy resource database (policy resource for short) as IETF Policy Management Model. The other is the local policy resource database (local database for short). After administrator's editing the policy, they put it into the policy resource. The PDP should decide, according to whether it has new policy or any undated policies, if it needs to download the policy to local database. If necessary, store the policy need to be downloaded in local database or updated the policy in the local database by PEP.

Thus, as to control the various operations of local computer, when PEP decides whether to execute a policy, it will not only rely on PDP getting the corresponding responses by seeking policy resource and send back to PEP for execution, it can search the policy in the local database directly. And the function of PDP is kept. The policy implemented by PEP, in fact, determined by PDP as usual. The only change is to put the policy which PDP decided to execute into the local database and wait for PEP to seek and execute it. This will improve two problems of IETF Policy Management Model in desktop management system:

- PEP can visit the local database without using the network request / response mode. When the network is interrupted, PEP can still find and respond to the policy in local database, and control the operation of the local computer. In addition, if administrator has changed the policy in policy resource during the interruption, the local database will be immediately updated after the network restoration, so as to make sure PEP use the latest policy to control the local computer. We solve the problem of network interrupt by this means.
- We add the local database to the model, and it only saves the policy which is contained this local computer in the domain. When the policy was added or was changed, the local database gets updated immediately. Because PEP can execute the policy without sending and receiving the requests online and reduce unnecessary searching consumption. So it greatly increased the speed of response policy and improved the problem of transmission speed.

5. System Implementation Technology

5.1. The overall logical organization of system

Based on the above policy model which we proposed, we designed the overall logical structure of the system. Fig. 3

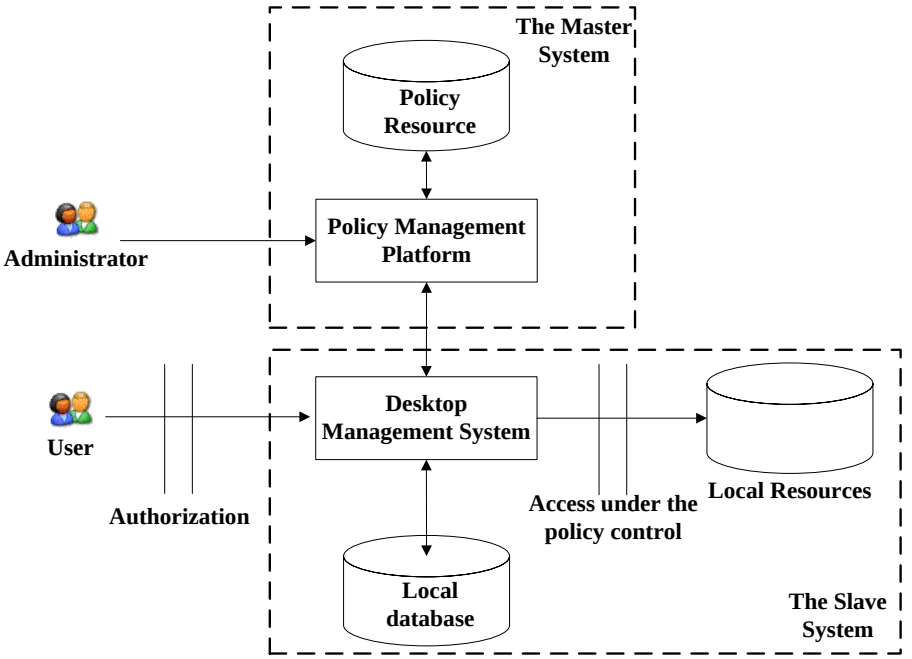

Fig. 3. The overall logical organization of system

The Overall structure includes the master and the slave system.

The master system is mainly used by administrator to edit policy and put the policy into the policy resource.

The slave system is mainly used to, according as policy; achieve the purpose of controlling access of the local computer through interaction of desktop system and local database.

Through the interaction of the master and the slave system, new policy and updated policy are put into local database. Furthermore, the slave system also can interact with master to respond to the information of local computer such as Process status, CPU status, and memory usage and so on.

5.2. System realization

As shown in Fig. 4, this is the realization of policy model which is improved and designed by this paper in desktop system.

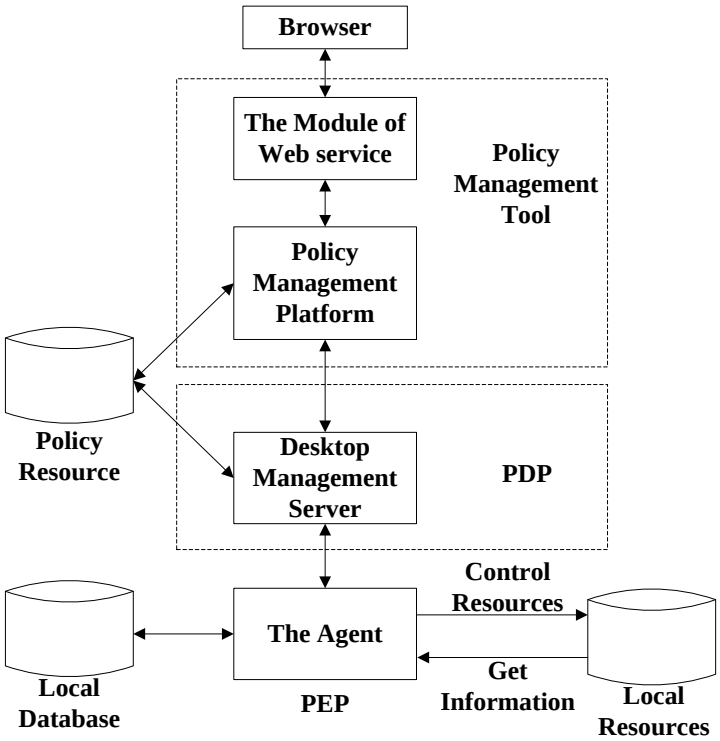

Fig. 3. The realization of improved policy model in the desktop management system

The whole system can be divided into two parts the policy management part and the desktop management part. And some technologies involved in this system are as follow: C++, Java, JSP, Servlet, XML and some scripting languages.

Each function module in the Fig. 4 is described below.

1)The Module of Web service

This module is designed to provide the web service. It has compiling system to interpret and execute the Java and Servlet code. This module mainly used to store the JSP or Servlet programs from the Policy Management Platform. The module of web service is used to provide the graphical user interface of the Policy Management Platform. Users can operate the Policy Management Platform by Browsing, adding, modifying, deleting and searching the policy directly.

2)Policy Management Platform

Policy Management Platform and the Web service constitute the module of policy management tool in policy model. The policy management platform mainly includes the following functions.

a) Editing policy

Administrator edits the policy in accordance with the way that the policy is expressed as mentioned above. It specifically includes the Policy Domain management (the operate includes: adding, modifying and deleting and so on), the User management (the operate includes: adding, modifying and deleting and setting the authority and so on), the User Group management (the operate includes: adding a user group, adding or deleting a user from user group and so on), the Security Domain management (the operate includes: adding, modifying and deleting and so on), the Rule management (the operate includes: adding, modifying and deleting and so on), the

Policy management (the operate includes: adding, modifying, deleting, bonding user or user group, bonding rules and so on). Finally, the policy which is edited by administrator will be put into the policy resource.

b) Interact with policy resource

The Policy Management Platform needs to interact with policy resource in order to realize operations such as adding, modifying, deleting, and search and so on. The administrator can execute associated operations directly via Web browser.

c) Interact with desktop management server

There are two different interactions between The Policy Management Platform and the desktop management server.

- When the administrator adds a new policy or updates a policy by using the Policy Management Platform, it should send a message to desktop management server, and wait for its next operation.
- When the administrator wants to find out information of local computer such as Process, Memory and so on, it should send a message to desktop management server, and wait for its next operation.

3)Policy Resource and Local Database

The policy resource in improved model remains the same in IETF Policy Management Model. It is used to store the policies, the relevant policy data such as policy's rules, user group, rule domain, Security Domain, the information of local computer and so on.

The local database, as already stated in our previous paragraph, only store the policies in the local computer domain.

There are several ways to implement the policy resource and local database. It can be LDAP directory service or the relational database system. The paper used the RDS because of the local database needs to add, update and delete the data in time. The RDS is much more efficient than the LDAP in terms of writing speed. Meanwhile, we can make the Index of database to increase the RDS's reading efficiency.

4)Desktop management server

The Desktop management server is corresponding to the PDP in the IETF Policy Management Model, It mainly includes the following functions.

a) Interact with Policy Management Platform

There are two different interactions between the desktop management server and The Policy Management Platform.

- When the Policy Management Platform sends a message to desktop management server and informs the server that the policy has been updated, the desktop management server response to this message and go to the next step.
- It responses to the request for the local computer information submitted by The Policy Management Platform. And it sends the local information to The Policy Management Platform.

b) Interact with policy resource

After the desktop management server received the message of policy update, it searches the policy resource in time. And finally decides whether it needs to transmit the policy to local domain. If the transmit is necessary, it will gather the policies need to be updated in local domain and then send policies to the Agent in local.

c) Interact with the Agent

There are two parts of the interaction between the Desktop management server and the Agent.

- It transmits the policies need to be updated to Agent.
- Obtain the local information reported by Agent.

5) The Agent

The Agent is the Core Module of the desktop management system. It is a permanent process program running on the local computer. All of the access control on local computers is achieved through the Agent. It is corresponding to PEP in IETF Policy Management Model and the Agent mainly includes the following functions.

a) Monitoring the local information and operation

In order to control the local computer, The Agent needs to inject into the process permanently and monitor the local information and operation. For instance, if there is a policy when the local computer's CPU usage exceeds a certain limit it will alarm. Then the Agent, through monitoring, is able to detect the situation of CPU and respond to it according to the policy.

b) Blocking operation

The definition of Blocking Operation is when a user submits a request to the operating system, it is blocked by Agent. In order to control access to the local computer, the blocking operation is necessary. For instance, a user group does not have permission to access the C drive. When the user submits a request to OS for visiting the C drive, the Agent should hold up this request.

c) Interact with local database

There are two parts of the interaction between the Agent and the local database.

- Adding or updating the policy to local database that downloaded from desktop server.
- Searching the policy information from local database and executing in local computer.

d) Interact with the desktop management server

There are two parts of the interaction between the Agent and the Desktop management server.

- The Agent receives the policies transmitted from desktop management server.
- The Agent received the request sent from desktop management server (such as the request of getting the local information), and then return the result to desktop management server.

6. Conclusion

This paper improved the IETF Policy Management Model and applied it to the desktop management system. It has the following features: a) formulate the way of policy expression. b) Improved the problem of policy conflict by making the priority of policy. c) Improved the problem of net interruption and transmission speed by using the local database. Because of putting the policy model into the desktop management system is a relatively new attempt, some of problems need continued study.

References

[1] Verma Dinesh C, Calo Seraphin, Amiri Khalil, "Policy-based management of content distribution networks,"IEEE network, vol. 16, pp. 34-39,2002

[2] Morris Sloman,Jorge Lobo, "Policies for distributed systems and networks," International Workshop on Policy, Bristol, UK, 2001

[3] Braden R., Zhang L., Berson S., Herzog S., S. Jamin, Resource ReSerVation Protocol (RSVP)Version 1 Functional Specification, RFC 2205, September 1997.

[4] Herzog S.,RSVP Extensions for Policy Control, RFC 2750, January 2000.

[5] Chen-Zhifeng "The Solution of Desktop Management System Based on Network," Application Research of Computers (in Chinese),2001,(9),pp.21-22.

[6] Li-Jinping, Gao-Dongjie."The Research and Design of Policy-based Network Management Software Platform," COMPUTER ENGINEERING AND APPLICATIONS (in Chinese),2002,(12),pp.177-179.
[7] Li-Qinghai, Zhang-Deyun, Duan-Zhongxing, Sun-Zhaohui."Design and Implementation of Specification for Role-Based Policy Management of Distributed System," JOURNAL OF XI'AN JIAOTONG UNIVERSITY (in Chinese),2004,38(6),pp.566-570.
[8] A.Westerinen, J. Schnizlein, J. Strassner "Terminology for Policy based Management," IETF RFC 3198, 2001
[9] J. Moffett and M. Sloman, "Policy Hierarchies for Distributed Systems Management," IEEE Journal on Selected Areas in Communication,1993,11(9),pp.1404-1414.
[10] R. Yavatkar, D. Pendarakis, R. Guerin, "A Framework for Policy-based Admission Control," IETF RFC 2753, 2000