POLICY BASED SYSTEMS MANAGEMENT
A NEW SOLUTION FOR MANAGING ACADEMIC COMPUTER SYSTEMS

Author: Kevin Sullivan

Middler Year Writing Requirement
ENG 1125 – 5:15PM Section
Ms. Amy Kaufman
May 29, 2003

Spring 2003

College of Computer and Information Science, Northeastern University
360 Huntington Avenue, Boston MA, 02115
Table of Contents

ABSTRACT ................................................................................................................................................................. 3

1. THE STATE OF SYSTEM ADMINISTRATION ........................................................................................................ 4

2. SYSTEMS MANAGEMENT PROBLEMS .................................................................................................................. 5
   2.1 ACCESS CONTROL ............................................................................................................................................... 5
   2.2 SYSTEM CONFIGURATION .................................................................................................................................. 6
   2.3 STORAGE MANAGEMENT .................................................................................................................................. 6
   2.4 PROBLEM BACKGROUND ..................................................................................................................................... 6

3. POLICY BASED MANAGEMENT PRIMER .............................................................................................................. 9
   3.1 POLICY LANGUAGE ............................................................................................................................................... 9
   3.2 POLICY INFRASTRUCTURE .................................................................................................................................. 10
   3.3 IMPACT ON SYSTEMS ......................................................................................................................................... 10

4. ADVANTAGES AND DISADVANTAGES ....................................................................................................................... 11

5. IMPLEMENTING A POLICY BASED SYSTEM ........................................................................................................... 13
   5.1 DESKTOP CONFIGURATION MANAGEMENT ................................................................................................. 13
   5.2 SERVER CONFIGURATION MANAGEMENT ..................................................................................................... 13
   5.3 STORAGE MANAGEMENT SYSTEM .................................................................................................................... 14
   5.4 ACCESS CONTROL SYSTEM ................................................................................................................................ 14

6. CONCLUSION ............................................................................................................................................................ 16

7. REFERENCES ............................................................................................................................................................ 17

ANNOTATED BIBLIOGRAPHY ................................................................................................................................... 18

DOCUMENT PROPERTIES ........................................................................................................................................ 20
Abstract

In academic environments, computer systems are being utilized for nearly every imaginable task from research to administration. As the number of these systems grows, managing each becomes increasingly difficult. This paper proposes a system that will aid in the configuration and administration of a large number of systems. A detailed problem analysis is followed by a concrete set of recommendations stemming from commercially available products. The core of the recommendation is a policy-based management system that enables administrators to write high-level policies which are automatically transformed into system specific rules. System policies are the key to relating business rules to technology rules. This paper covers previous research in the area of policy based management including infrastructure and policy syntax. While still a developing research topic, the policy based initiative is an effective means to manage multiple computer systems in a complex environment.
1. The State of System Administration

In the field of systems management there are a few givens. Systems will break on a Friday afternoon, and you will not be able to find the documentation that you need when you need it. Such problems are unnecessary and should be fully avoidable. A number of years ago, a team of system administrators had a disastrous experience resulting from a lack of good systems management policies and procedures. The team was in the midst of rolling out new software images to desktop computer, rushing to meet organizational Year 2000 compliance deadlines. This team consisted of a group of administrators with a wide range in experience, from interns to senior level engineers. The group was always considered successful as they kept the systems running with minimal downtime for many years. In the few months leading up to this event the computing infrastructure had grown tremendously as a result of departmental consolidations. Systems had been patched together and still continued to function normally.

One morning that summer a few users reported that their email was responding abnormally slow and they could not open some messages. The administrators thought this was strange but nothing completely out of the ordinary. As the morning went on, more and more users were reporting the same situation. An administrator thought it would be wise to check on the mail server in a hunt for clues. The server itself was also responding very slowly. The team took the usual steps to try to clear up a problem, a few minutes later the problems has manifested again. This troubleshooting continued for the next 27 hours. A large group of experts, including on-site consultants and product support specialists from Microsoft was assembled to resolve the problem. By start of business the next day the users had their email functioning again, though some had permanently lost a sizable amount of their saved mail.

These types of problems are not uncommon in systems management. There is always the possibility that a system will break, but minimizing those chances is key part of the administrator’s job. The problem cited here resulted from a lack of adherence to a few basic policies and procedures. The root cause of the problem was that the email server was being used for another task that competed with the mail database for disk space. That server should not have been used for that task, yet access was still granted. The problem most detrimental to the users was the loss of some of their saved messages. Recent backups of the mail database were useless because the disk layout of the server was improperly configured. A standard convention for arranging disk drives on mail servers was available yet never followed when this server was built. The once well respected team had lost their good standing with many customers due to carelessness and lack of policy that led to data loss for their customers.

A popular argument is that system administrators lack the appropriate tools to effectively and efficiently manage their systems. Computer systems should be managed by specific enforceable policies. Policies can dictate all areas of system administration including access control and system configuration. The Department of Economics at the Massachusetts Institute of Technology can surely benefit from this systems management technique. The Economics Department runs dozens of high-end computer systems to aid in faculty and student research. Because of the system’s heterogeneous nature an abstracted policy framework would greatly increase the efficiency and reliability of these systems. The department’s systems group is well positioned to implement this technology. This paper will examine the problems that rise from a lack of policy and recommend the implementation of a policy based management system to aid in systems management.
2. Systems Management Problems

System administration takes many forms in different environments. There are organizations where a group of experts is responsible for all computing services, companies who have one person work as a part time administrator, and other shops where each employee is responsible for administering his or her own computers. No matter which model is employed, organizational computing policies abound. Most organizations see computer systems as a means to accomplish an end goal. Too often computer systems hinder that goal.

Business rules and polices are not completely compatible with computer operating systems. As organizations move to more complex and heterogeneous computing environments it becomes more difficult to actively enforce policies within computer systems. There is too much dependence on external devices for ensuring policy compliance, often an employee manually checking for compliance. Verma repeatedly states that the fundamental systems management problem lies in the translation of business level policies to technology level policies [1]. Many software developers provide some degree of policy support in their products, yet it is not generic enough to allow for the complex situations that arise in academic computing departments.

The Economics department has managed computer systems for many years. An entire team is now devoted towards managing desktop systems and a number of high end research servers. Over the last few years the development of systems management techniques has not kept pace with the explosive growth in the number of supported systems. As a result, serious problems have emerged that affect both the administrators and end users alike.

Nearly every aspect of a system administrator’s job has a problem that can be solved or a process that can be made more efficient. In the department there are three especially problematic tasks. The first is access control, the process of determining which departmental resources a particular user will have access to. System configuration is another area that has been extremely problematic for the department. Finally, storage management, the allocation of storage resources to users and groups is a highly political subject and has been full of problems in the past.

2.1 Access Control

Dealing with access control requests is a daily occurrence for members of the systems team. This type of task involves a great deal of interaction with the user community as well. Currently each user’s account contains a list of the resources, or servers, that they are allowed to use. Each server must be configured to check the existence of the account attribute and deny access in its absence. This process requires configuration changes to every account and every resource (server). Recently during a major system upgrade, several of our systems changed names. This required that an administrator manually edit each user’s account to remove access to the old machine and add access to the newly renamed computer. A task such as this could take the better part of a working day to complete. Many of the department’s implicit policies regard access control. The policies include general rules on server access, account creation, and password settings. Since these documents are not defined anywhere or checked by any computer system they are open to interpretation any time an access control operation takes place.
2.2 System Configuration

System configuration may be the number one bullet point on an administrator’s job description. Members of the team must often configure both desktop and server class computers. The closest to a policy the department has ever had was a checklist of software packages that should be installed on all new desktop computers. This document was seldom followed and did not address any specific for configuration options and difference depending on what type of user the computer was for. A policy for server configuration has never existed and each server is configured differently today. This creates a problem when users work on multiple servers and are confused because of the subtle yet disruptive differences. The team has come under scrutiny in the past because a software package wasn’t installed or backup jobs were not setup. This lack of consistency also creates problems when troubleshooting computer problems. A troubleshooter comes to depend on having certain tools available on a system or knowing from experience where particular data files are located. That person loses these benefits when working with an improperly configured machine. In a large environment system configuration is a serious duty and needs to be properly undertaken or the results can be disastrous.

2.3 Storage Management

Another time consuming administrative task facing the department is the allocation of storage space on the department’s file server. Almost two years ago the department centralized all of its storage in a central system that is accessible to all other systems via a network connection. Since the space on this server is used for so many different purposes, it is difficult to represent the department’s policy decisions with the software that manages the server. The primary uses of the storage space are for students to store their files and to give faculty space to support their research projects. Problems commonly occur at the intersection of these two applications; students may work with various faculty on their research and will require storage space for that work. Traditional network operating systems do not provide native support for these collaborative situations. The current rule involves a loosely defined policy that is interpreted differently by each administrator. Any time a change is to be made an administrator must translate the users request and current policy into a structure that can be represented in the server. This often contradicts existing policy and leads to user confusion when their coworkers have seemingly been held to a different policy.

2.4 Problem Background

These problems have recently become a very acute problem in the department due to the rapid expansion of computing services offered to students and faculty. Since the systems group manages resources for a wide variety of individuals and purposes, it has become increasingly difficult to support the systems in accordance with policies established by the department. There are numerous areas of administration that can suffer from a lack of policy. Among them are server access, desktop restrictions, storage allocation, and document management. Generally, there is little support in standard software for setting and enforcing policies. Currently, none of the systems in the department have built in support for a policy system such as the one recommended in this paper.

Administrators and users are almost equally affected by this problem. Manually enforcing policies is a very time consuming endeavor for system administrators. The potential fallout from
a breach of policy would be disastrous. The lack of a clear, universally enforced policy often pits users against administrators. System administrators are often perceived to be standing in the way of a user’s work. In Verma's paper, he defines policies is a constant struggle between two forces: user independence and the efficiency of the organization or support group [2]. Mark Leary, a systems administrator at the MIT Economics Department believes that it is possible to achieve perfect equilibrium between the two forces. In a personal interview, he remarked that “end I do think user freedom and control is more important because if a person cannot get his work done using the technology and tools provided to him, then those tools become useless.” These comments echo perfectly the goal that system administrators strive towards. However, it has been my experience that most users do not understand how much work goes into the smooth operation of multiple systems. It is not enough to have a policy when its enforcement is questionable. I foresee that users would be less likely to question a policy when they see that it is unambiguously enforced.

The effects of this problem on the department have been twofold. First, there has been a huge loss of productivity among the system administrators. Many hours are lost in the process of manually setting and enforcing policy decisions. Any time a new user account is created; many departmental policies need to be manually imposed on that account. Rules for group membership, storage limits, and remote access must be set with each account. Breaches of policy are even more time consuming. It is often very difficult to curb a user’s behavior once they act outside of policy. For example, if a user has been consuming 100 gigabytes of storage and the policy states that they are entitled to 50, it will be very difficult for them to reduce their disk usage. As shown above, this dilemma affects users as well. It is customary that when a community will be governed by rules, those rules should be clearly published and universally enforced. Today policies are stashed away in a drawer or only remembered by a handful of employees. Enforcement is sporadic at best and depends on manual intervention by administrators on a constant basis. This is a problem that will continue to grow if left alone. The complexity of the problem increases with every system that is added to the network. The systems group was created in an effort to centralize computing knowledge and offer a resource for support within the department. If the group can not keep up with the effective management of its growing system then I foresee members of the department choosing again to administer their own computer systems.

The systems group has the power to address this problem. They are the primary point of reference for all computing purchases in the department. The members of the group are skilled enough to understand and implement a system which would more effectively manage the computing resources of the department. Addressing the problem means actively evaluating new hardware and software purchases for compliance to policy standards. The benefits presented in this paper should justify any funding needed.

The biggest reason that this problem has yet to be solved is that to an outsider it generally seems as if things are working properly. The department’s systems are generally functional and users can work without incident. This observation is very misleading. A common problem that system administrators face is that they are deemed almost inessential until something breaks; when there is an outage the blame is almost always placed on the administrators. Systems groups may have trouble obtaining funding for preventive maintenance projects such as this. Often a group will not receive necessary funding until a catastrophe has already struck. This recommendation comes at a critical time for the systems group. In previous years it was possible, though not terribly efficient to manage each system manually. This took many man hours and
results in dozens of inconsistencies among the systems. Also, in previous years, undertaking a major project like this would seem too costly for the limited benefits. Now the department is seeing unprecedented growth of its computing systems. Faculty have multiple supported computers, graduate students bring laptops to the department and researchers order new computational servers at a greater pace than ever before. The department is reaching the critical point where systems simply can not be effectively managed by hand. The benefits of any project to simplify and expedite administrative tasks would outweigh any costs at this point.

Solving this problem, like any other that the department faces, will be a costly endeavor. There will be a great paradigm shift in terms of how most administration problems are handled. Almost every facet of administration will be abstracted from the specific problem and this will require a change in thought processes for many team members. There is no reason to believe that this project will lead to any loss of jobs; it will only free the team from the most tedious parts of their jobs. There will be more time available to work on the more challenging projects that have previously been put off due to time constraints. There will be some additional overhead in solving a problem such as this. The infrastructure to support the solution will require upfront hardware purchases but they scale much better than having to hire additional employees as the server population grows.

Managing systems through policies is just one in a long line of possible solutions to this pressing problem. When you break down many of the other available solutions they too are based on policies. However, it does not seem that the developers of those solutions realized that policies were the backbone of the solution. This proposal addresses policies as the fundamental object to solving many of today’s systems administration problems.
3. Policy Based Management Primer

There are two methodologies to policy management. In both cases policies are written by administrators. These policies can either be enforced by the administrators or automatically by a computer system. In the past, system policies have been manually enforced by the administrators themselves. Hogan and Limoncelli, in their tome of system administration, define three roles in the area of policy management. There is the policy writer, the policy navigator, and the policy enforcer [3]. A policy writer is someone who acts as an intermediary between business and technology. This person understands enough of each to make the position valuable. A policy navigator is a person who understands what the policies are and why they are important. The person in this position helps users accomplish their task while staying in line with policy. This role is crucial to avoid hindering user’s work with policies. Finally, the policy enforcer is charged with saying “no” when someone wants to do something that is against policy [3]. The enforcer can stop an activity when it is deemed to violate policy. This person is in the difficult position of seemingly getting in the way of progress. These roles are currently undefined in the Economics department. One administrator may assume the role when necessary but there is little continuity in these roles.

Policy and policy enforcement is not new to system administration. However, administrators are now faced with more complex policies. Small organizations that could benefit from uniform and enforceable polices are not able to dedicate three positions to policy management as outlined above. A more efficient system has been under development by internet standards bodies. This system allows for most of the policy management duties to be carried out by computer systems rather than administrators. The components of such a system are outlined below.

3.1 Policy Language

Most polices will be represented in a textual form. The many various policies that would be involved in systems management do not conform to a particular language or standard. Therefore a very extensible format for expressing these policies must be found. There are two competing ways to represent; for humans the best way to represent a policy would be natural language, while for computers the best choice would be a formal specification language [1]. The two options break down further into either descriptive or logical policy documents. A descriptive policy document will describe precisely what a user or system is allowed to do, for example where to print or what software can be installed. A logical policy document is a set of rules and actions. When defining a prototype language, Li, et al describe the document at follows: “when an event occurs, if some condition is true then execute the assigned action [4]. A mix of both these approaches would be necessary to manage the plethora of devices and services found in large networks. The systems group is already familiar with the language types as they are found in much of the web content and configuration files that the department’s systems utilize.
3.2 Policy Infrastructure

There are four main components to a Policy Based Management system. An administrator starts by using a policy management tool to define policies. A policy tool is a graphical application that lets the administrator manage multiple policies. This tool can be likened to an Integrated Development Environment (IDE) for programmers. A policy repository is a data store that holds information about all the currently available policies. In order to ensure interoperability across products from different vendors, information stored in the repository must correspond to an information model specified by the Policy Framework Working Group [1]. These two entities represent the endpoints of the infrastructure necessary to support policy based management.

On the internals of the system area policy enforcement point (PEP) and a policy decision point (PDP). The PEP is a device or computer system that is capable of applying and executing policies. A PDP is responsible for interpreting each policy in context and passing the appropriate decision down to the PEP.

3.3 Impact on Systems

Most modern systems are well suited towards policy based initiatives. Many commercial products already support policy in some form. Unfortunately in today’s competitive environment, these systems are far from interoperable. Martinez and Brunner, et al. have developed an implementation of the IETF policy framework. Each of the aforementioned infrastructure components has been implemented using protocols and services that are widely available. The impact of this research is that these intangible ideas have been partially implemented albeit in a laboratory environment. The Martinez and Brunner paper specifically examines the “PDP has a mid-level manager which generates concrete device configurations from a set of more abstract policy rules” [5]. This component is the heart of policy based systems management. With this piece in place, the goal of defining abstract high-level policies and having them translated to system specific configurations is fully realized. The simple network management protocol (SNMP) is used to set the appropriate configurations. Nearly every networked system has some support for SNMP configuration management. Though this is a good mechanism for proof of concept, SNMP is insecure and is not the preferred management technique for many modern operating systems.
4. Advantages and Disadvantages

A system that is capable of managing nearly all aspect of departmental computing poses serious advantages for a department that implements such a system. Many of the immediate advantages from this system center on consistency of systems and labor efficiency. These are the two major problems with managing systems using the current method, so a system that seeks to address those needs is advantageous.

Consistency is highly sought among systems management. Supporting a large number of systems leads to differing uses for systems and eventually discrepancies between configurations. Systems managed by policy would have a specified configuration that would be enforced by the policy system. This is a benefit to two stakeholder groups. Administrators benefit because all systems can be managed using similar tools and all files and applications would be located in the same place across system. This helps users of the system as well since they can seamlessly transition from one system to another. Commonly users will be confused because something that worked for them on one system does not have the same behavior on another system. These types of problems represent a large portion of the help desk requests a systems team would receive. Eliminating unnecessary inconsistencies would go a long way towards improving general usability of systems.

Secondly, implementing a policy based management system would reduce the maintenance costs of the departments systems. Being able to manage groups of systems as one entity saves countless hours of configuration time. Beyond that, a reduction in support requests will occur as users will no longer be confused by different setups and configurations among systems. Lowering the total cost of ownership (TCO) is extremely advantageous to the department. With a lower TCO, the department is able to support more systems than was previously possible. Faculty and researchers whose requests for more processing systems were once denied can now be accommodated with efficiently managed systems.

Critics have found few faults with the adoption of this policy based system. Among them are financial constraints and employee training and retooling. This proposal will require significant investments in hardware, software and development time. Given the economic climate an overly expensive solution will not be accepted by the decision makers in the department. However, annual budgets for maintaining systems should cover any costs incurred. Hardware such a UNIX and Windows servers will need to be dedicated to the policy management infrastructure. To manage Windows systems additional software and licenses have to be purchased but that investment will cover numerous other applications that the department would already need to purchase. Development work may pose a time constraint for the department but a coop employee would be able to engineer most parts of the solution. While there is a sizable cost to this proposal it in no way negates the benefits to be realized by the department’s users and administrators.

Boardman and Saperia claim that “automated policy-based management is supposed to fix it so that networks can be run by fewer – and less highly skilled – individuals.” [6] In the case of systems management I do not find this statement to be true. Systems management involves a constant and sometimes growing number of users. Implementing policy based management will not decrease the number of supported users. A policy based system will allow administrators to do their jobs more easily and with greater efficiency. This proposal will require slight retooling on behalf of the administrators. For the members of the Economics Department Systems group
this is an easily surmountable obstacle. Administrators will perform task in a more logical goal oriented manner. A modicum of experience with object-programming will be required in order to understand and write the policies for systems. While the system will drastically change the efficiency and reliability of the department’s computer systems there is no threat to job security or the notion of employees being replaced by automated systems. This system will be merely a tool to aid administrators.

This is a system where the advantages clearly outweigh the disadvantages. Both main disadvantages are a cost that would come with any effort to increase efficiency. Again, both can be overcome. Users and administrators alike stand to gain substantially from this proposal. The specific recommendations will detail how to best achieve the benefits of this system and to overcome and negative aspects of the implementation.
5. Implementing a Policy Based System

I am proposing a system that will, through modular components, achieve the goal of policy based systems management. At this time, there is no one product, out of the box or otherwise that will serve as a panacea for all systems management problems. Here I propose implementing a set of solutions that will position the department for complete policy based systems management at the time that a full solution is available. These recommendations will cover both UNIX and Windows system configuration, along with a design for developing storage and account management solutions.

5.1 Desktop Configuration Management

Desktop computers running the Windows operating system can be managed using tools included in the standard operating system. This suite of tools is known as Group Policy. Microsoft states that group policy can be used to “Administrators use Group Policy to specify options for managed configurations for groups of computers and users.” [7] This aligns with the goals of a policy based management system. I recommend that the department use the features of Group Policy to deploy and maintain software on desktop computers. This process consists of three steps.

1. Repackage software in a format understandable by the group policy applications. The most widely used format is Microsoft Installer (MSI). MSI packages can be created using Winstall LE from Veritas. Use the Winstall software to repackage each software title and store it on a network accessible location.

2. Create organizational units (OU) that reflect logical groups of users and computers. Organizational units are logical grouping that can define the structure of users on a computer system. Users and computer can be grouped by job task of physical locations. Use Active Directory Users and Computers on a Windows 2000 server to arrange the resources.

3. Create group policy files that link software packages to OUs. This is also done within Active Directory Users and Computers. On the software deployment tab add any packages that you wish to be deployed to a group of users or computers. These steps define an enforceable policy across an entire Windows domain. Each domain controller serves as both a PDP and PEP. Policies can be set by administrators from their own workstations using a standard tool and storing the policy data in a central repository. One drawback to this solution is that policies are not viewable by users though they should be translated to lay terms and posted to an intranet site for that purpose.

5.2 Server Configuration Management

With many faculty, students and researcher using central processing servers to manipulate data for their research, the stable and consistent configuration of servers becomes just as important as desktop configuration. There are a variety of open source packages available for Linux system configuration management. One of the oldest and most reliable packages available
is cfengine by Mark Burgess [8]. Cfengine holds true to the policy based initiative by allowing administrators to define intended configuration in a central policy file, with PEPs and PDPs on each target host. I propose that cfengine be employed to manage configuration files and software distribution on all of the departments Linux hosts. Basic system configurations such as Network File Service (NFS) mounts, directory service configurations, and logging, would be identical across all machines. The only difference that would occur between servers are for special purposes such as a web server or mail server, and to comply to licensing restrictions, such as a limit to the number of computers that a particular package can be installed on.

5.3 Storage Management System

I propose that the department develop of extensible storage management system to handle the complex needs of academia. This would effectively act as a middleware solution between current file server software and a set of storage policies.

1. Adopt an XML schema for defining storage quotas. The extensibility of XML allows the department to make any additional provisions that that may be needed. Sloman touts XML as a widely accepted document format that has been successfully used in other policy implementations [9]. This will provide a uniform way to author all quota policies.
2. Program a script that will transform XML policy documents into a format that the file server can understand. Currently, the NetApp file server expects NFS style quota files. An XML transform file can do this generically for all input policy files.
3. Optionally, develop a front end for creating and editing policy files. The XML files are human readable as will be thoughtfully commented, but an easy to use interface will be indispensable when pitching this solution to higher management.

This portion of the solution is crucial to repairing the relationship between users and administrators. Storage policies will be accessible to all in a clearly readable format. All changes and modification will occur in the XML policy file rather than the lower level quota file. This allows for extra information to be stored about each entry. This high level XML quota file should be made publicly available so that all users are aware of their storage privileges and limitations.

5.4 Access Control System

In order to solve the access control dilemma I am recommending that the department develop an extensible account management system. The current problem is that user account privileges can not be easily updated. To that end, I recommend the following action steps.

1. Create a set of additional attributes to be associated with each user system. This should include expiration dates, the sponsor of the account, and a multi-valued entry to enumerate any access rights.
2. Define a resource access policy in XML that maps users and groups to resources. This policy will be used centrally instead of storing all of the information in each users account. For example, if a server’s name changes, it will only need to be
changed in the policy file as opposed to previous time when each user’s account needed to be edited.

3. The access control policies should be stored in a centralized directory server. Chadwick and Otenko present a model that focuses on hierarchical role based access control [10]. Storing user accounts in a tree fashion complies with this model. The departments existing directory server can serve as a policy repository for this function.

This is another aspect of the solution that will be best developed in house. Web based development provides a strong jumping off point to interface design for this type of system. As Mark Leary noted in the interview, “access [control] is the most important place to have a strong policy structure in place, as it has the greatest security implications,” accordingly this solution should be given priority and designed especially carefully.
6. Conclusion

The Economics department now finds itself at a crucial crossroads. The demand for computing systems is growing at a rapid pace. Students, faculty and staff are using computers for nearly every facet of their jobs. Researchers are requesting more and more computing power to keep their growing staff productive. Management of these systems becomes exponentially more difficult as they grow in number. The department can no longer afford to hire additional staff to support a growing system population as they had done before.

Implementation of a policy based management system would allow current administrators to do their jobs more efficiently and effectively while decreasing the need for additional staffing at a time when economic conditions preclude that. System stability would rise, user complaints would plummet and administrators could be working at their most efficient levels. Implementing a system such as this would propel the department towards becoming a model environment for other academic computing departments. Many systems groups strive for what is known as “5 nines reliability”, that is 99.999% uptime. This is a lofty goal that could easily be achieved with a quality management solution such as a policy based system. Few groups today are able to pride themselves on the benefits that this system would deliver.

On the other hand, grim consequences could follow if the department does not act soon to rectify this situation. Without a solid management solution desktop systems will fall out of date and server class machines will become less and less reliable. Administrators would simply not be able to deliver the quality of service that they would like to and this will be detrimental to the end users.

Like any other project the systems group embarks on, this will have significant costs in terms of hardware, software and development time. This solution seeks to present great cost savings to the department in future years. The methods and solutions presented seek to deliver the best cost benefit in line with the goals and objectives of the department.
7. References


Annotated Bibliography


Chadwick and Otenko present a model for managing user privileges in computer systems. The model is based on standard, readily available technologies. They present a rich information medium to dictate policies related to access control. This article presents the language used to author policies as well as a detailed overview of the infrastructure to enforce the policies on a wide variety of systems.

This paper will be instrumental in the access control section of my research. Access control comprises a large portion of the duties that system administrators are charged with. I plan to use this paper to propose an system independent model for access control.

David Chadwick is recognized internationally as an expert in directory services, an extensible phone book for the internet. He has also authored an internet standard on directory services. Chadwick is currently a professor at Salford University in England where he manages a number of research projects related to access control and directory services.


This paper surveys the domains of security policy and management policy. Details and applications of both are discussed. The final section of this paper reviews currently available commercial tools for policy management and enforcement. “In policy based networking most of the tool support comes from industry… [3]” From this list of tools, I would like to choose one or two that would be both useful and feasible for my department.

Morris Sloman is a Professor of Distributed Systems Management at Imperial College in London. He is a member of the Policy Research Group and has chaired several conferences and workshops regarding policy based network management.


In this article, the authors discuss an algorithmic implementation of system management policies. A formal language is presented in which complex policies can be represented unambiguously. This system is based on first order logic and uses syntax similar to Prolog [2]. This paper also details the execution model for policy enforcement and an infrastructure for distributing policies across managed systems.
I plan to use this article to demonstrate how policies for the department would be written. Based on the figures in this article I would be able to produce sample polices congruent with the language presented here. Also, the information about the management framework will be helpful in modeling the system. The authors are researchers in the Computer Department at the Institute of Computer Engineering located in Nanjing, China.


In this paper, the authors discuss an implementation of policy based management that is based on a recommendation by an internet standards body. This recommendation covers creating, storing, evaluating and enforcing policies [4]. Two prototype implementations are discussed in the course of the paper. I will use the information found in this paper to evaluate commercial products on how well each conforms to the established standards.

Marcus Brunner is a senior researcher at Network Laboratories in Heidelberg, Germany. He has earned a PhD from the Swiss Federal Institute of Technology. He has worked with the Internet Engineering Task Force on the development of standards for policy based management and lists Network and Service Management as his primary research interest.


This article discusses the complete spectrum of policy based management. The process of transforming business requirements to computer system code is fully documented. The latter half of the paper focuses on the infrastructure required to support policy based management. The author presents an excellent example in which he applies these concepts to managing a service level agreement (SLA), a contract specifying performance metrics.

Since this article is so thorough in its coverage of policy based management it has become the main resource for my research. I will use this information to support my recommendations for policy based systems. Also the section on translating business level policies to technology level polices is central to the topic and will be very useful in convincing my audience of the benefits of policy based management.

Dinesh Verma holds a Ph.D. in Computer Science from the University of California, Berkeley. He is the research manager of the Edge Networking group at the IBM Watson Research Center. His major research interest in policy based networking. He is also a member of the IEEE.
Document Properties