Automatic Enforcement of Location Aware User Based Network Access Control Policies

TUGKAN TUGLULAR
Department of Computer Engineering
Izmir Institute of Technology
Gulbahce Koyu, Urla, Izmir
TURKEY

Abstract: Multiple interconnected network segments distributed across various locations, such as corporate networks, where users or employees constantly travel among segments and require to access servers, need to have network access control mechanisms that are able to adapt to these location changes. The idea of a firewall changing or adapting its rules depending on the location of users is presented by an architecture in this paper. This architecture proposes deployment of a policy server at the management level and policy agents at the firewall level, so that policy-driven network security management is enabled by specifying location aware user based network access control policies at the network security management and enforcing them at the managed firewalls. The architecture presented in this paper utilizes user VPN connection event triggers for dynamic policy configuration and automated policy deployment to firewalls. Location aware user based network access control policies, which are management level policies, are implemented using XACML. A network level policy is usually a configuration, or policy, file local to the firewall. The policy agent incorporated into the firewall performs the mapping from management level policy to firewall policy.

Key-Words: Access Control, Network Security Policies, Firewalls, Location Awareness, XACML, Firewall Policy Agents.

1 Introduction

Maintaining security on their networks is critical for all corporations. One primary tool that every network needs is access control – the ability to carefully define and enforce which users have what type of access to specific applications, data and services [10]. When the network is contained within a single building, the problem is generally handled by a firewall. For corporate networks, which involve multiple interconnected segments distributed across various locations, the problem is hard and beyond node level configuration. To manage this kind of heterogeneous, distributed, and dynamic networks, system administrators want to focus on policy level management instead of taking their time for node level configuration [22]. Therefore, utilization of policy-based network management systems for the large scale networks is suggested [6].

Policy-based management has become a promising solution for managing enterprise-wide networks and distributed systems. These are typically large-scale systems which require management solutions that are both self-adapting and that dynamically change the behavior of the managed system [5]. The main motivation for the recent interest in policy-based services, networks and security systems is to support dynamic adaptability of behavior by changing policy without recoding or stopping the system [5]. This implies that new mechanisms, which dynamically update the policy rules interpreted by distributed entities to modify their behavior, should be integrated to existing architectures. The ability to dynamically change, distribute, and enforce these policies is the key to robust corporate network security management [18].

In corporate network environments, where locations of users change frequently, it is desired to have network access control mechanisms that are able to adapt to these changes. This will provide users with more seamless, easy use services. The paradigm of tailoring applications, services, communication and connectivity to the user's current situation and needs, is referred to as context awareness [8]. The goal of location aware user based network access control approach is to develop an architecture that enables the network access control policy to be adapted to user context information, specifically to the location of the user.

This paper addresses the challenge of managing and enforcing a corporate's network access control policies with respect to user and his/her location. An important aspect of this architecture is that network security policy management and enforcement are externalized from applications. This reduces effort and complexity related to maintaining and enforcing organizational policy, especially when information and related policies are distributed across multiple network segments [3].

This paper presents a network access control approach based on the XACML [15] standards, which will be used for expressing access control policies based on authorization attributes and statements. Similar approaches exist in the literature [12], [11], and [7]. One of the reasons to choose XACML as the access control policy language for the architecture is that there is an implementation to support the XACML specification, known as Sun java XACML implementation [19]. It is possible to extend this open-source implementation to adapt it to context-aware access control [8]. In this paper, introduction of XACML implementation is followed by the demonstration of the way to extend it for location aware user based network access control usage.

For the enforcement of location aware user based network access control policies, the focus will be on firewalls as the network access control mechanisms. The challenge is for firewalls to know the right reconfiguration so that the appropriate security policies are upheld preventing illegitimate users from gaining access [4]. The architecture presented in this paper also focuses on the configuration management of firewalls so that organizational network security policies can be enforced. Building a management layer for firewalls using policies enhances its value by making the management of the firewall technologies highly programmable and easily adaptable to different environment and security requirements [21]. Relevant network access control policies will be disseminated to the firewall policy agents so that they implement them on their firewalls [17].

The rest of this paper is structured as follows. Section 2 provides an overview of related work. Section 3 describes the proposed architecture for the enforcement of location aware user based network access control policies. Finally, the paper concludes with some remarks and future directions.

2 Related Work

Yoshioka et. al. proposed a policy based automatic network configuration mechanism that makes the configuration of segmented networks inconsistency-free [22]. In their proposed mechanism, a policy describes the types of events and configuration messages a network element is to accept, and that policy is registered on a central policy server by each network element in the initialization phase of the element. After the initialization, all events and messages are sent to the policy server, and the server forwards them to appropriate network elements considering the registered policy so that they can reconfigure themselves automatically. There is another work on policy based configuration of network elements using SNMPv3 [14]. In their work, Omari et. al. aimed to integrate the policy concept into the SNMPv3 framework. They proposed a set of rules to map authorization policies to the VACM (View Based Access Control Model) standardized as part of the SNMPv3 management framework.

Various frameworks, systems and architectures have been proposed providing context aware access control and/or policy enforcement. Some of them are closely related to the work presented in this paper. Lymberopoulos et. al. presented a framework for specifying policies for the management of network services in their paper [13]. Their framework supports automated policy deployment and flexible event triggers to permit dynamic policy configuration. In their paper, they focused on solutions for dynamic adaptation of policies in response to changes within the managed environment. Kapsalis et. al. proposed a context-aware, access control architecture, in order to support fine-grained authorizations for the provision of e-services, based on an end-to-end web services infrastructure. In their proposal, access permissions to distributed web services are controlled through an intermediary server, based on a role-based access control model, which incorporates dynamic context information, in the form of context constraints [9]. Han surveyed the recent researches about context awareness and context-aware security. Based on the survey results, the requirements of CASPEr system are analyzed and using the knowledge from analysis and up-to-date technologies the CASPEr model is designed. Furthermore, this model is applied on access control to let this security mechanism to be context aware [8].

Lorch et. al. presented XACML, a standard access control language, as one component of a distributed and inter-operable authorization framework in their paper. Several emerging systems which incorporate XACML are discussed and using these discussions they illustrate how authorization can be deployed in distributed, decentralized systems [11]. Giordano et. al. proposed a visual language hierarchy and a tool for specifying and implementing access and security policies according to the RBAC model. They represented those policies in XACML [7]. Tuglular et. al. suggested a method to represent firewall policies using XACML for the platform independent management of distributed firewalls [20].

3 Architecture

Traditional security systems lack adaptive security policies and enforcement mechanisms [16]. In the non-adaptive setting the set of policies is chosen in advance, before the server connection request is received. The adaptive policy enforcement architecture presented in this paper selects the appropriate policies during the course of VPN connection operation based on the current location of the user. This architecture employs a Policy Server (PS) and a number of Policy Agents (PA). Each domain has a policy server to adapt network access policies and each firewall in the corporate network has a policy agent to modify its policy. The whole architectural picture is given in Fig.1.

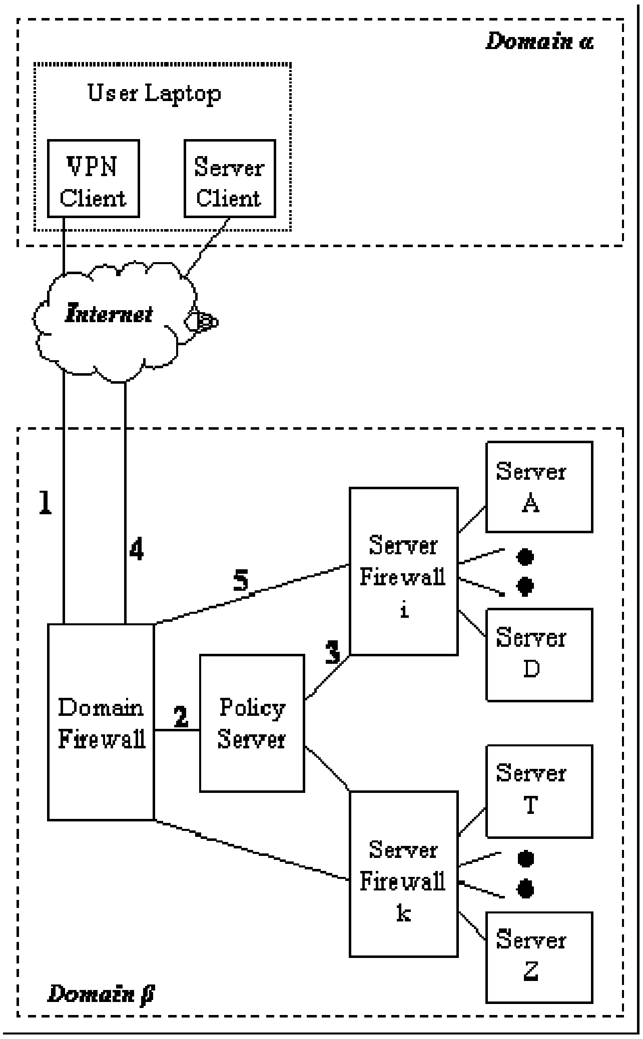


Fig.1 Location aware user based network access control architecture

The main principle of this architecture can be defined as centralized policy decision and distributed reconfiguration. In this semi-distributed architecture, the domain firewall initiates firewall configuration requests, then policy server takes network access control policy decisions and communicates with firewall agents to distribute necessary policy changes so that firewalls are reconfigured automatically. On-demand configuration requires the following algorithm [2], which assumes pre-determined policy is already deployed on firewalls. All the steps are performed automatically:

i. adapt policy to current user location
ii. deploy adapted policy
iii. enforce adapted policy on firewall(s)
iv. monitor user location changes
v. return to step *i* when a change is detected

The above algorithm becomes operational on the proposed architecture as follows. First, VPN client opens a VPN connection to the domain firewall, so that domain firewall becomes aware of the users location and informs the policy server as shown as step 2 in Fig.1. Depending on the user access rights, policy server modifies high level network access control policy and deploys this adapted policy to corresponding firewalls in step 3. The policy agent residing on the corresponding firewall maps high level policy to machine dependent low level policy, which is actually the firewall policy. Starting from that point on, server client application, such as a web browser, can connect to servers that the user has access rights. When the VPN connection is broken the domain firewall notifies the policy server, so that it replaces the adapted policy with the original network access control policy and deploys to the corresponding firewalls.

In the core of this architecture lies the policy server which manages user location based access control policies. The following components, which are shown in Fig.2, are identified within the policy server: Configuration Deployment Point (CDP), Policy Decision Point (PDP), User access rights Policy Information Point (UPIP), Network access control Policy Information Point (NPIP), and Topology Information Point (TIP). CDP is responsible for enforcing the policy decisions through configuration change obligations. PDP is the core of the policy server. It evaluates applicable policies, i.e. user access control policies, and renders policy decisions against network topology information to formulate new network access control policy. Policy and topology information are retrieved from UPIP, NPIP and TIP.

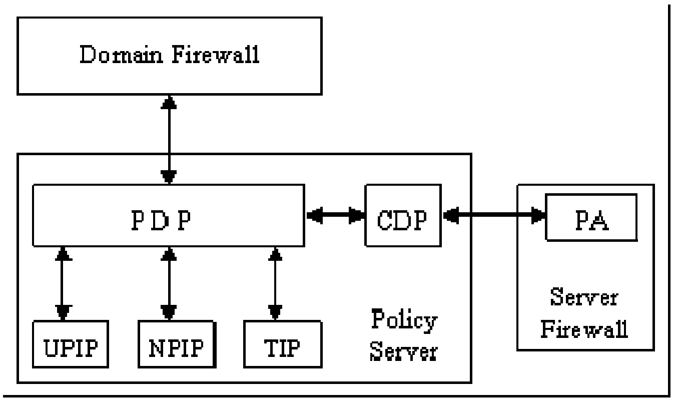


Fig.2 Policy server components

All the information points that exist within the policy server store their information in XML format. UPIP holds user access rights with respect to servers, for which the proposed DTD (Document Type Definition) is shown in Fig.3. It simply represents which user can access which servers.

```
<?XML version=”1.0”?>
<!DOCTYPE Access_Rights [
    <!ELEMENT Access_Rights (User, Server*)>
    <!ELEMENT User>
    <!ATTLIST User VPN_Client_ID CDATA #REQUIRED>
    <!ELEMENT Server>
    <!ATTLIST Server IP_Address CDATA #REQUIRED
                                    Port CDATA #REQUIRED>
]>
```

Fig.3 A DTD for user access rights representation

TIP stores topology information mapping which firewall protects which servers. Therefore, a topology element consists of one or more firewall elements and a firewall element consists of one or more server elements. A firewall element has only an IP_Address (the IP address of the firewall) attribute, whereas a server element has two attributes: IP_Address (the IP address of the server) and the Port (the port number it serves from). To represent such information, a DTD for topology representation is proposed here and presented in Fig.4.

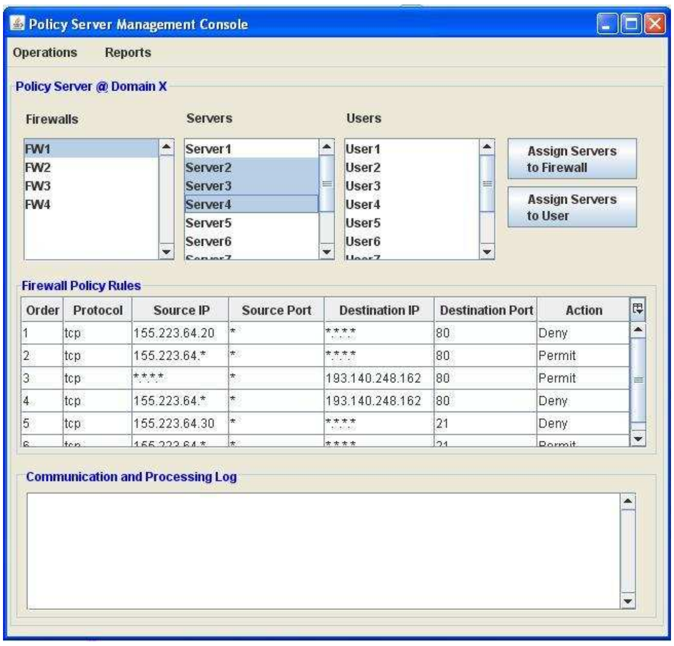
```
<?XML version=”1.0”?>
<!DOCTYPE Topology [
    <!ELEMENT Topology (Firewall*)>
    <!ELEMENT Firewall (Server*)>
    <!ATTLIST Firewall IP_Address CDATA #REQUIRED>
    <!ELEMENT Server>
    <!ATTLIST Server IP_Address CDATA #REQUIRED
                                   Port CDATA #REQUIRED>
]>
```

Fig.4 A DTD for topology representation

NPIP is responsible for holding network access control policies. Tuglular et. al. developed a method to represent high level firewall policies using XACML [20]. XACML is suitable as policy description language because it is able to represent relations among firewall policies and to embed other information related to a firewall into its policy such as; firewall IP address, agent identification, implementation independent representation of policy rules and rule combining algorithms. The same method is used here for NPIP to store firewall policies. The XACML PolicySet element holds all the policies of firewalls deployed in the domain. The XACML Policy element encapsulates all the rules of a firewall policy and the XACML Rule element represents a single rule of a firewall policy. The order and action fields of a policy rule are represented as the attributes of XACML Rule element. The protocol information is stored in the XACML Rule Condition element. The src_ip and src_port information are held in a Subject element of the XACML Rule Target element. Similarly, the dst_ip and dst_port information are held in a Resource element of the XACML Rule Target element. The implementation independent representation of a rule is stored in the XACML Rule Description element.

Policy agent plays an important role in reconfiguring firewall according to the policy decision made by the policy server. In this paper, firewall policies are adapted by using one of the ways explained by Lymberopoulos et. al. [13], which is the adaptation by dynamically changing the rules of a firewall policy to specify new configuration for the run-time of firewall. This policy adaptation process is sometimes referred as mapping high level policies to low level policies in the literature [1], [14]. In the proposed architecture, high level policies represented in XACML are mapped to machine dependent firewall rules with the help of policy agents.

The security officer or system administrator responsible for policy management needs a management console, where she/he can administer firewalls, servers and users and their interrelationships. A prototype implementation of policy server management console is shown in Fig.5. This console is designed in such a way that when an item from firewalls, servers, or users categories is selected, the console shows interrelated items from other categories. For instance, as seen in Fig.5 firewall FW1 protects servers Server2, Server3, and Server4. When a server item is chosen, the firewall that protects that server and user(s) that has access rights to that server will be highlighted.



Order	Protocol	Source IP	Source Port	Destination IP	Destination Port	Action
1	tcp	155.223.64.20	*	*.*.*.*	80	Deny
2	tcp	155.223.64.*	*	*.*.*.*	80	Permit
3	tcp	*.*.*.*	*	193.140.248.162	80	Permit
4	tcp	155.223.64.*	*	193.140.248.162	80	Deny
5	tcp	155.223.64.30	*	*.*.*.*	21	Deny
6	tcp	155.223.64.*	*	*.*.*.*	21	Permit

Fig.5 A prototype implementation of policy server management console

Same management console can be used to assign servers to a firewall and to assign servers to a user via the buttons on the right handside of the console. The prototype also includes a panel for firewall policy rules. When a firewall is selected, its rules can be seen in this panel. The last panel implemented in the prototype is the communication and processing log panel, which displays category item operations (i.e. adding or deleting a server), assignment operations (i.e. assigning a server to a user), firewall rule operations (i.e. updating a rule), and policy mapping operations (i.e. modifying a machine dependent firewall policy or configuration file).

4 Conclusion

This paper describes a location aware user based network access control architecture which provides automatic reconfiguration of firewalls so that a corporate user requesting a VPN connection from any domain will have network access control mechanisms updated with respect to his/her server access rights when the VPN connection procedure is completed. The proposed architecture utilizes network access control and user access rights policies as well as network topology information to make a policy decision and then this decision is mapped to firewall policies through firewall agents. The whole process is transparent to both users and servers. The requirements and the constituent parts of the proposed architecture are explained. A generic scenario is given to present the use of the architecture.

The proposed architecture has a lot of features that requires further development. One important future work is to include a secure policy exchange protocol between the policy server and policy agents that provides authentication, data integrity, confidentiality and access control. Moreover, policy decisions made on location awareness can be extended to encapsulate not only access to servers but also type of access to servers. Location awareness can be incorporated with temporal awareness, so that they both can effect policy decisions.

References:

[1] A. Apvrille and M. Pourzandi. XML distributed security policy for clusters. In Computers & Security, Volume 23, Issue 8, December 2004.

[2] F. Barrère, A. Benzekri, F. Grasset, R. Laborde, and B. Nasser. Automated inter-domain security policy generation. In Proceedings of the 11th Annual Workshop of HP OpenView University Association, June 2004, Paris, France.

[3] P. Charles and B. Daly. Rule Based Infrastructure: A Design and Runtime System for Enabling XML Schema Driven Applications. available at http://rosetta.sims.berkeley.edu:8084/docs/RuleBasedInfrastructure.html

[4] J. Burns, A. Cheng, P. Gurung, S. Rajagopalan, P. Rao, D. Rosenbluth, A.V. Surendran, and D.M. Martin, Jr. Automatic Management of Network Security Policy. In Proceedings of DARPA Information Survivability Conference & Exposition II, Anaheim, CA, USA, 2001.

[5] N. Damianou, A. Bandara, M. Sloman, and E. Lupu. A Survey of Policy Specification Approaches. April 2002. available at http://www.doc.ic.ac.uk/~mss/MSSPubs.html

[6] P. Flegkas, P. Trimintzios, G. Pavlou, and A. Liotta. Design and implementation of a policy-based resource management architecture. In Proceedings of IFIP/IEEE Eighth International Symposium on Integrated Network Management. 2003.

[7] M. Giordano, G. Polese, G. Scanniello, and G. Tortora. Visual modelling of role-based security policies in distributed multimedia applications. In Proceedings of IEEE Sixth International Symposium on Multimedia Software Engineering. 2004.

[8] Y. Han. Context Aware Security Policy Enforcement: CASPEr. Master's Thesis. Technische Universiteit Eindhoven. 2005.

[9] V. Kapsalis, D. Karelis, L. Hadellis, and G. Papadopoulos. A context-aware access control framework for e-service provision. In Proceedings of (ICIT 2005) IEEE International Conference on Industrial Technology, 2005.

[10] R. Kay. Analysis: XACML. 31/12/2003. available at http://www.computerworld.com.au/index.php?id=997396262&fp=16&fpid=0

[11] M. Lorch, S. Proctor, R. Lepro, D. Kafura, and S. Shah. First experiences using XACML for access control in distributed systems. In Proceedings of the 2003 ACM workshop on XML security. Fairfax, Virginia. 2003.

[12] G. Lopez, A.F. Gomez, R. Marin, and O. Canovas. A Network Access Control Approach based on the AAA Architecture and Authorization Attributes. In Proceedings of 19th IEEE International Parallel and Distributed Processing Symposium. 2005.

[13] L. Lymberopoulos, E. Lupu, and M. Sloman. An Adaptive Policy-Based Framework for Network Services Management. In Journal of Network and Systems Management, Volume 11, Issue 3, September 2003.

[14] S. Omari, R. Boutaba, and O. Cherkaoui. Policies in SNMPv3-based Management. In Proceedings of the Sixth IFIP/IEEE International Symposium on Integrated Network Management (Distributed Management for the Networked Millennium), Boston, MA, USA. 1999.

[15] OASIS eXtensible Access Control Markup Language (XACML). available at http://www.oasis-open.org/committees/xacml/

[16] T. Ryutov and C. Neuman. The Specification and Enforcement of Advanced Security Policies. In Proceedings of Third International Workshop on Policies for Distributed Systems and Networks. 2002.

[17] M. Sloman and E. Lupu. Security and management policy specification. In IEEE Network, Volume 16, Issue 2, March, 2002.

[18] T.J. Smith and L. Ramakrishnan. Joint Policy Management And Auditing In Virtual Organizations. In Proceedings of Fourth International Workshop on Grid Computing. 2003.

[19] Sun's XACML Implementation, available at http://sunxacml.sourceforge.net/

[20] T. Tuglular, F. Cetin, O. Yarimtepe, and G. Gercek. Firewall Configuration Management Using XACML Policies. To be presented in 13th International Telecommunications Network Strategy and Planning Symposium, Sep. 28 – Oct. 2, 2008, Budapest, Hungary.

[21] A. Virmani, J. Lobo, and M. Kohli. Netmon, network management for the SARAS softswitch. In Proceedings of (NOMS 2000) IEEE/IFIP Network Operations and Management Symposium, 2000.

[22] T. Yoshioka, T. Igakura, and T. Tonouchi. Policy-based Automatic Configuration of Network Elements in Separate Segments. In Proceedings of (APNOMS 2003) Asia-Pacific Network Operations and Management Symposium. Oct. 2003.