Universal Journal of Electrical and Electronic Engineering 6(3): 79-93, 2019 http://www.hrpub.org
DOI: 10.13189/ujeee.2019.060301

Dynamic Role-based User Service Authority Control and Management on Cloud Computing

Yejin Kwon[1], Jerry H. Seo[1], Young Bom Park[2,*]

[1]Institute of Supercomputing and Networking, Korea Institute of Science and Technology Information (KISTI), South Korea
[2]Department of Software Engineering, Dankook University, South Korea

Abstract The demands for computing resource allocation, user management, and various contents management are increasing with the proliferation of cloud computing. A cloud service system has to provide the system services that are combined or shared based on the web environment, unlike a single application in the past. It has also evolved to appropriately allocate and combine service resources according to the needs and needs of various users to provide a user-dependent web environment. There is an increasing need for software convergence services in the form of software. In response to these needs, a variety of customized services are provided to a user who promises an optional service environment. Those services can be used in combination according to a user's purpose and needs, instead of providing an environment in which a user purchases and uses software locally. In particular, because cloud computing provides the process that a user gain accessing authority to the system which has been granted a service access right for a certain period of time, and including a process of expire the corresponding right is completed, a various authority access control method is designed. In this paper, we define the roles of users who access cloud computing service, and manage user rights according to each role, provide and expire appropriate service resources according to user's privilege and session information. Moreover we analyzed the procedures for constructing an algorithm for service ripple effects according to the procedure of granting service resources to each user and accessing web services and constructing a new role.

Keywords Role Based Access Control, Authority, Role Distribution, Access Control List, Dynamic Role, Static Role

1. Introduction

The demands for computing resource allocation, user management, and various contents management are increasing with the proliferation of cloud computing. A cloud service system has to provide the system services that are combined or shared based on the web environment, unlike a single application in the past. It has also evolved to appropriately allocate and combine service resources according to the needs and needs of various users to provide a user-dependent web environment. There is an increasing need for software convergence services in the form of software. In response to these needs, a variety of customized services are provided to a user who promises an optional service environment. Those services can be used in combination according to a user's purpose and needs, instead of providing an environment in which a user purchases and uses software locally.

Recently, most of the software is provided in a dynamic form in the cloud computing environment on the convergence of the needs of various users and the specialized fields and the individual dependent system. Unlike a user purchases software locally the past, there is an increasing need for various types of software convergence services that are customized to the user. Instead of providing the software, which provides an optional service environment in which the service can be used according to the user's purpose and need. In particular, various methods of accessing and controlling the system have been devised, as the user is assigned the service access authority which is an accessing the system for a certain period of time, and after the deadline the corresponding authority is expired [12].

In order to provide appropriate services to users, it is necessary to manage an access authority to access each service, and a system that manages the authority is necessary. Currently, various service-based architecture systems are provided with the goal of providing the necessary services to the users. Nevertheless, it is also important to give proper service access authority to the users and to analyze the effect of the service authority recovery and the whole system. Also, in the enterprise system architecture, research has been conducted to organize the user's authority and group according to the

business model and to grant authority of the system accordingly. Therefore, it is an important issue in the service-based system to provide an organized system through a process of granting proper authority to a user and recalling it. The group or organization that accesses the system has various types of services and privileges required according to each role, because it provides appropriate services and solely manage the security issues of the system by granting rights according to the role. It could be a method to enhance the competitiveness of software in order to increase the efficiency of the system and to provide user-specific services [1] [8] [13].

In this paper, we define the roles of users who access cloud computing service, manage user authorities according to each role, and provide and expire appropriate service resources according to user's privilege and session information. Thus, analyze the service ripple effects according to the procedure of each user accessing cloud computing service, granting service resource, and expiring. We analyze the effect of each user on the other users and the influence of other users on the service according to the correlation between the authority to access the cloud computing service that defined by the web portal and the role of the user who allocated the corresponding authority. Moreover, we developed an algorithm to analyze the cloud computing system and ripple effect of roles. Each service includes a function to control or share resources to be provided to users.

2. Related Work

2.1. Role Based Service Access Control (RBAC)

Role Based Access Control (RBAC) operates access control lists (ACLs) for service requests to access the database, and grants or denies access to use requests. It is a system that provides a security service that uses the authority granted to a user as a policy. Role-based access control algorithms have been used in security policies to control transactions that access information and grant roles and authorities to users [3]. In other words, research has been conducted to define various user's hierarchical layers and privileges according to the role and scope of the user, by limiting authority which the access to the information according to be mining user's role in the system that required security access control. User role based access control system has been used in various industries requiring security of equipment and system [3], and has been applied to systematically manage group or organization hierarchically in various applications.

At the first, the role-based access control system is constructed based on a simple relationship between users, roles, and permissions, and controls access to the system by granting roles to users and defining permissions according to roles. As the role-based access control system continues to be studied, complex algorithms and layered roles for access rights based on roles and relationships have begun to be constructed as the demands of roles that can be accessed by the system become diversified and complex [8] [13]. Recently, various analysis techniques such as behavior analysis and scenario analysis [14] have been applied, and research has been conducted to provide a complex and large-scale service that gives various roles to users and controls access rights have. In addition, research on the security aspects of how to assign appropriate roles to users and how to distribute and maintain the assigned roles is underway [6] [7].

Figure 1 shows the entire structure of RBAC applied to a virtual enterprise system. In order to create an interaction system between multiple geographically separated enterprises, the most important thing is to share the right authority with each other's systems or to share resources by sharing new rights. The process to verify the rights of users is belonging to different enterprises that require each enterprise resource and verify that they have appropriate authority, the authority of each enterprise member is stored in the database.

In this way, the RBAC system is applied maintains a list of access rights for user's requesting resources in the database or file repository, verifies that the user who has requested the proper authority has the right to receive the request, so that it can be used.

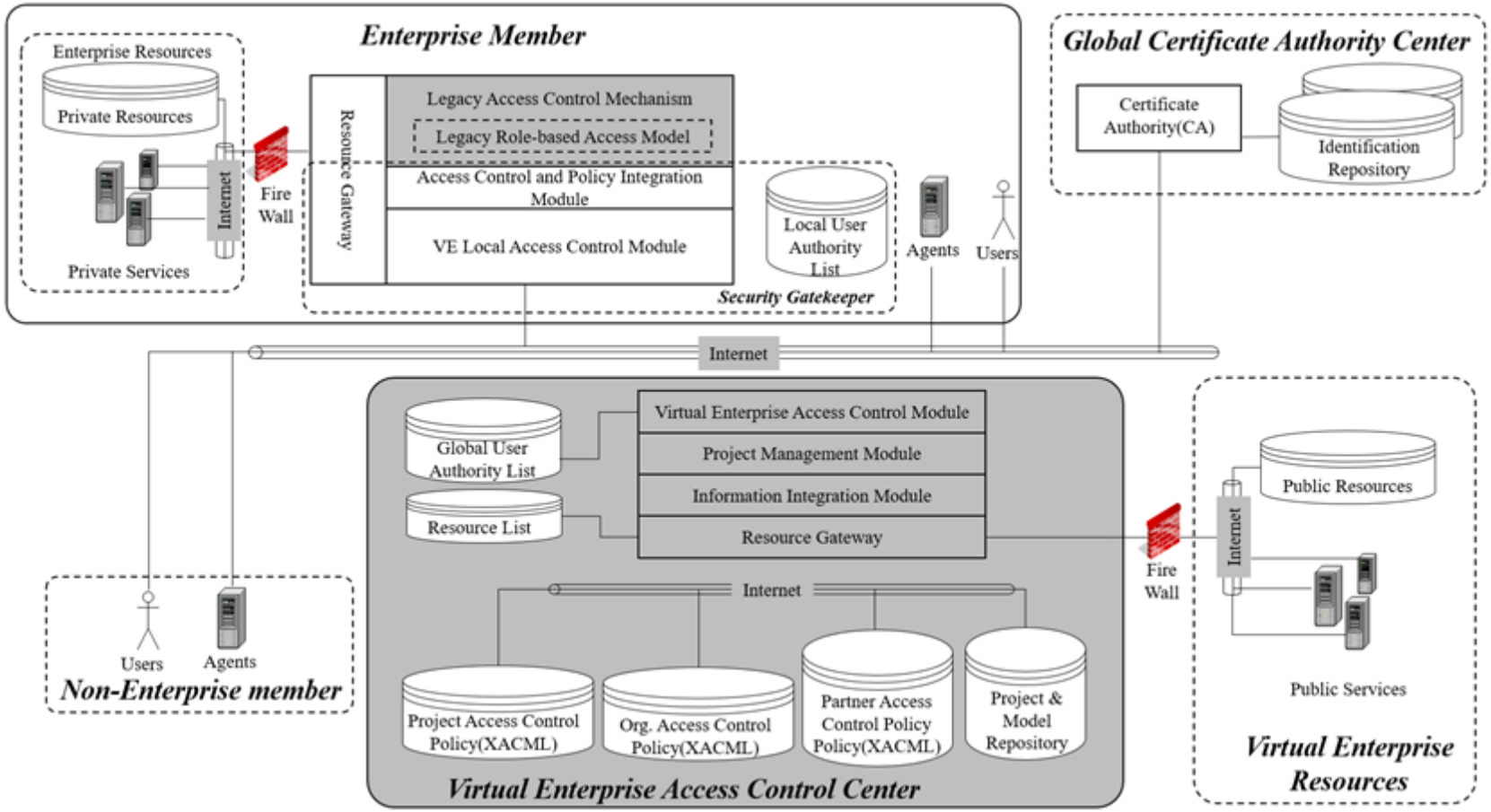

Figure 1. RBCA System based on Virtual Enterprise Architecture [8]

Table 1. Cloud Computing Definition

Cloud Computing Model		
Essential Characteristics	*Service models*	*Deployment model*
• On-demand self-service • Broad nework access • Resource pooling • Rapid elasticity • Measured service	• Software as a Service(SaaS) • Platform as a Service(PaaS) • Infrastructure as a Service(IAAS)	• Private • Community • Public • Hybrid

2.2. Cloud Computing and Service

Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. This cloud model is composed of five essential characteristics, three service models, and four deployment models [15].

Service-Oriented Architecture (SOA) is the fundamental architectural model that supports the overall paradigm of services computing from architecture perspective. SOA is a paradigm for organizing a set of capabilities, often distributed across the network and possibly under the control of different ownership domains. The organized capabilities can be used to provide solutions to business problems. Service is a useful concept if it describes the boundary between a consumer understandable value proposition and a technical implementation. The value proposition will be a matter of perspective for the consumer of the service and the technical implementation will be the responsibility of the provider for which they too will have a perspective. Consumers may wish to know some detail about the performance of the service to satisfy themselves as interested parties in a transaction but this does not affect the principle here. The use of the term technical is used in a loose sense and should not be taken as defining the level at which a service operates [16] [17].

To achieve the real-time functionality and the required QoS level, the infrastructure operation is separated in two phases: the offline, where the application and Application Service Components (ASCs) are prepared (i.e. development, modeling, etc.) and the online, where the resources are negotiated and reserved and the application is initialized and operates [18].

In this paper, we propose a dynamic role based user service authorization control system that provides appropriate services to users consuming services and the necessary services in real-time according to user's application environment and requirements. This system needs to consider into various authority that constructed by the service provider management and services in providing the service to the user.

3. Role Based Service Distribution System

In the cloud web portal system, it is the service authorization system according to the user role that provides the service of the system according to the role given to the user. And that system supports the user who accesses the session to use the service resource of the given authority.

3.1. User, Role and Authentication

Table 2 defines the service for the user who accesses the web portal service and has a proper authentication.

The authority to access the service is given differently according to the role given to the user. One user can have several roles, and the authority to use the services according to the role and user is given. A service to be given to a user defines a service to be provided to a user through an exclusive OR between a role assigned to each user and a period during which the corresponding role is maintained. Each user can be provided with a service including the granted role, the period in which the session is maintained, or the period during which the user can access the service. The authority granted to the user is maintained in the form of a token, and when the user requests the service, the service resource authorization is granted through the authority analysis for the user.

Table 2. Role Based Service Definition

Equation	Definition
$U = \{u \mid u_k \in U,\ 0 \le k \le \lvert U \rvert\}$	Subject who request to achieve service and get authentication
$R = \{r \mid r_i \in R,\ 0 \le i \le \lvert R \rvert\}$	The authentication which can access to service utility
$SS = \{s \mid s_j \in SS,\ 0 \le j \le \lvert SS \rvert\}$	A session in which a user accesses and maintains a web portal, and authorizes the user while each session is maintained
$P\_token = \{p(u_k, r_i) \mid p_m \in U \times R,\ 0 \le m \le \lvert U \times R \rvert\}$	The rights assignment according to the role of users and users who access the web portal
$S(u) = \begin{cases} \sum_{\exists(u,r)} f(t) \oplus p(u, r_i) \\ r_i \in R \end{cases}$	The function that provides server resources Utility for user's convenience
$authS(u) = \begin{cases} \bigcup_{f(U \to SS)}^{f(SS \to R)} S(u, P\_token) \\ P\_token \in P_{static} \cdot P_{dynamic} \end{cases}$	The authority granted to the user is maintained in the form of a token

3.2. System Architecture

The system architecture describes that distributes the service to the users through the role of the user accessing the service, the access right according to each role, and the session holding time as shown Figure 2.

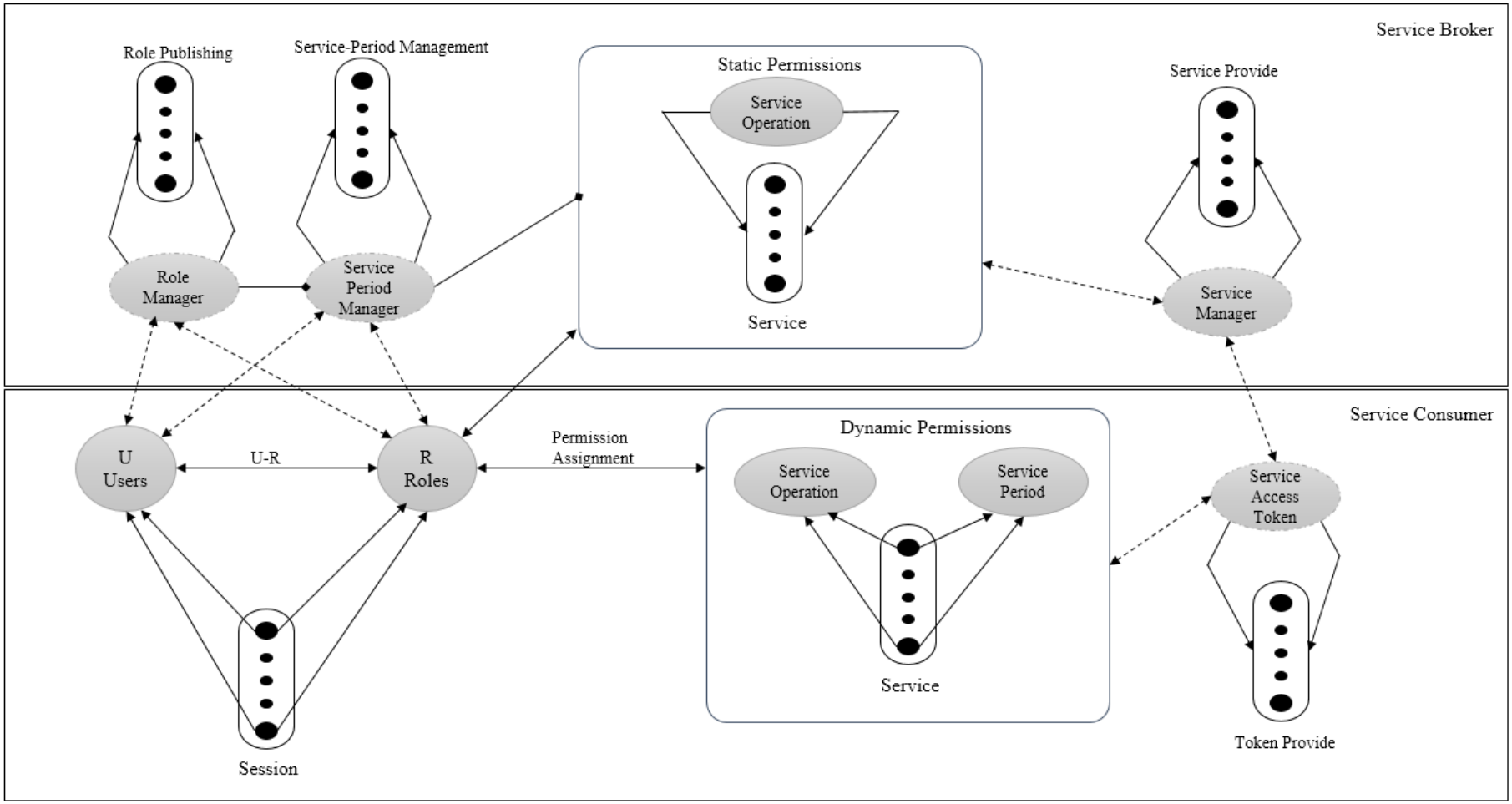

Figure 2. System Architecture

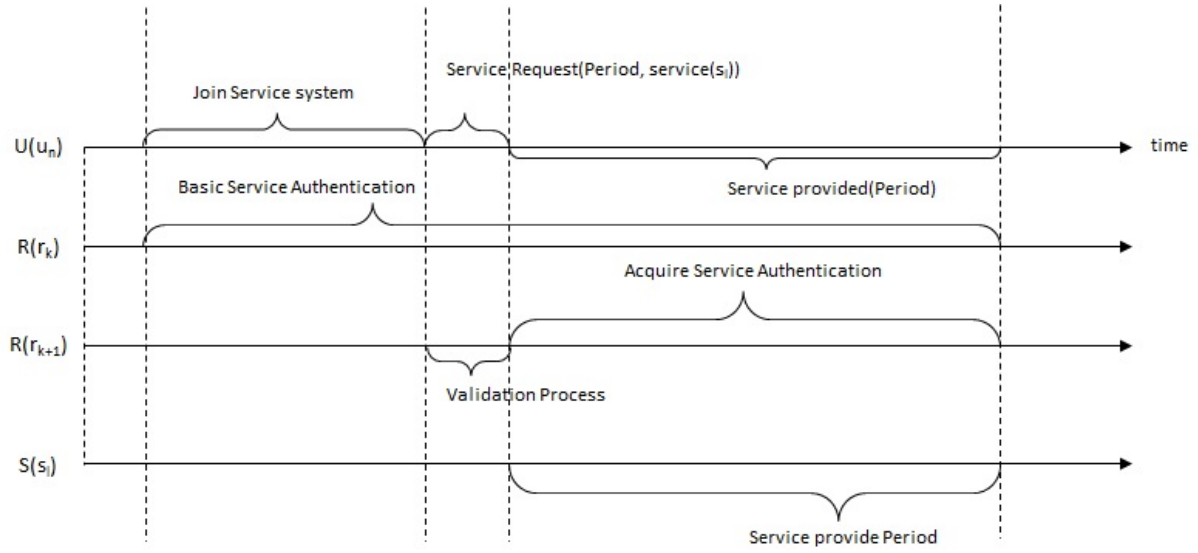

Figure 3. Role based Service Distribution System on Time Line

A user connected to the system can use the service after being authorized according to the role that can access the service resource. A user's role can be classified into a static role that does not change when a user first accesses the system, and a dynamic role that receives a different privilege according to a certain period. Accordingly, service access rights are divided into a static service and a dynamic service, and each user has a service access right according to a given role. That is, the user has a static role as a basis, and can be temporarily granted or authorized to access the service for a certain period of time. Also, if a user who has access to the service system has the authority to grant a role to another user, the user can grant authority to another user for a certain period of time and expire it again. Figure 3 is a graphical representation of changes in service entitlement over time.

When a user accesses the role-based service distribution system, the user is given a restricted role (Static Role) fixedly, and can receive service resources according to roles that assigned to the user. In addition, if a user requests authentication for using a new service while maintain on the current assigned service, that user could be granted or denied their request according to the administrator or the system policy having the authority to grant the service. When the authentication of the lastly requested service is completed, the user is given a role for this requested service for a certain period of time and can receive the service according to the role. User receives a role (Dynamic Role) temporarily granted to the user when the service period ends, user's dynamic role is expired and the service to the user is also stopped. These types of services distribution system is suitable for an environment where operation environment that user authentication and withdrawal are frequently have. For instance, web-based service provision system could be utilized in an interactive changing, or that have an enterprise-based systems which change the types of organization.

3.3. System Design and Construction

The service access control system based on the user role is developed on web based application and consists of each service portlet. The administrator who can temporarily grant the authority to the user has the control right to grant a dynamic role to the user, so that the service resource can be accessed and utilized by the user who assigned a dynamic role. Basically, the service resources that a user can access are limited. When accessing a web portal and logging in and requesting a new resource, service control of a user's request can be done on a portlet basis.

3.3.1. The Service Access Control based on Static Role

When a user accesses the system, the role that is basically provided can be defined as a static role. Therefore, the distribution of the user's static role is defined and maintained as a role that is provided as a basis in the existing defined role-based access control system. The User service control by static role in the service control system can be expressed as shown in Figure 4.

Fig 4 Service Access Control System on Static Role

As shown in Figure 4, when the user accesses the system and logs in, the system authenticates the role internally. Basically, various services can be used depending on the hierarchy of the user's role in addition to the privileges provided by the system. This role is stored and maintained in the role-based service access control system according to the user. Initially, the entire system administrator gives a new role or controls the service access request of the user. In the case of this system, the service resource means access control to the portlet, and the role and authority of the general user are given when the user logs in to the system.

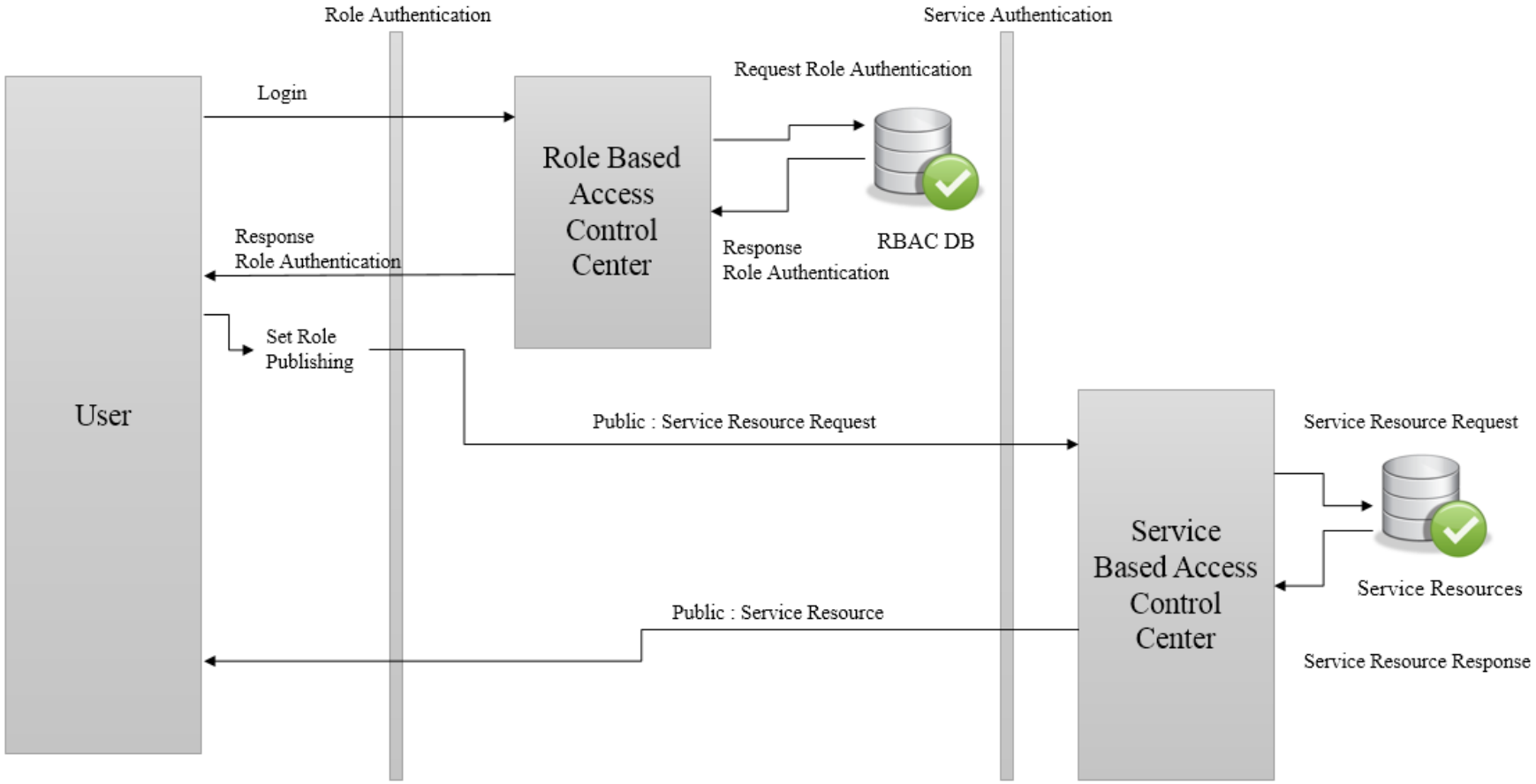

Figure 4. Service Access Control System on Static Role

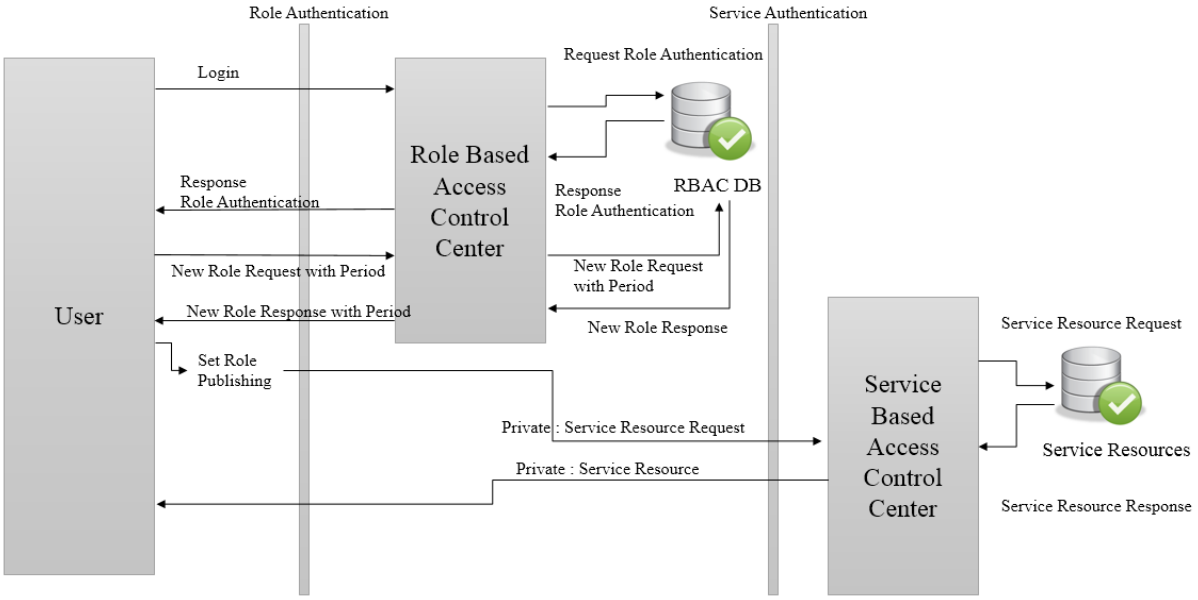

Figure 5. Service Access Control System on Dynamic Role

3.3.2. The Service Access Control Based on Dynamic Role

A service access control system, explains in Figure 5, receives a request to control services based on that role, and provides resources of a new role. This system accepts or denied their request following the security policy while a user is already logged in to the system, or that assigns a dynamic role by another user who has the authority to grant a new role.

A user is authenticated whether the service that user requesting has an authority of the administrator or can automatically assign the role according to the system internal policy when a new role is assigned to a user and a corresponding service is requested. The authentication process according to the dynamic role is similar to the service control system according to the static role in terms of the role given to the user. However, it is difference with that it is not provided in initial and when the service period is expired. Therefore, in order to grant a new role to a user must be maintained the information that a subject to be assigned a new role, a user to be assigned the role, an access service resource, and a service access period. Those information is maintained in a dynamic user based access control system, and operated similar method with the static user based access control system.

The dynamic rights control system according to the user role is constructed to dynamically control that the necessary services for the user and check the necessary permission in real time and to provide the necessary services effectively to the users.

3.3.3. The Service Control Process

When a user accesses a role-based service distribution system, a user is given a static role assigned fixedly. The system could be distributed service resources corresponding to a role assigned to the user. This process was shown in solid line in Figure 6.

If a user requests an authentication to use a new service while using the service, the user is granted or denied the request according to the administrator or system policy that has the authority to grant the service. When the authentication which newly requested service is completed, the user is granted the role including requested service for a certain period of time, and obtained the authority to access the service. This process that dynamically grants services according to their authority describes in order by dashed lines in Figure 6.

A user who wishes to be granted a dynamic role goes through a process of requesting to user who is currently assigned the role being managing to the role. The user who wants to be granted a dynamic role sends a request that is currently granted a dynamic role and requests permission for that role. The user who receives the request that grants the dynamic role goes through the authorization check. A user with permission who assigned dynamic role is temporarily granted a dynamic role and obtains access to the service resource.

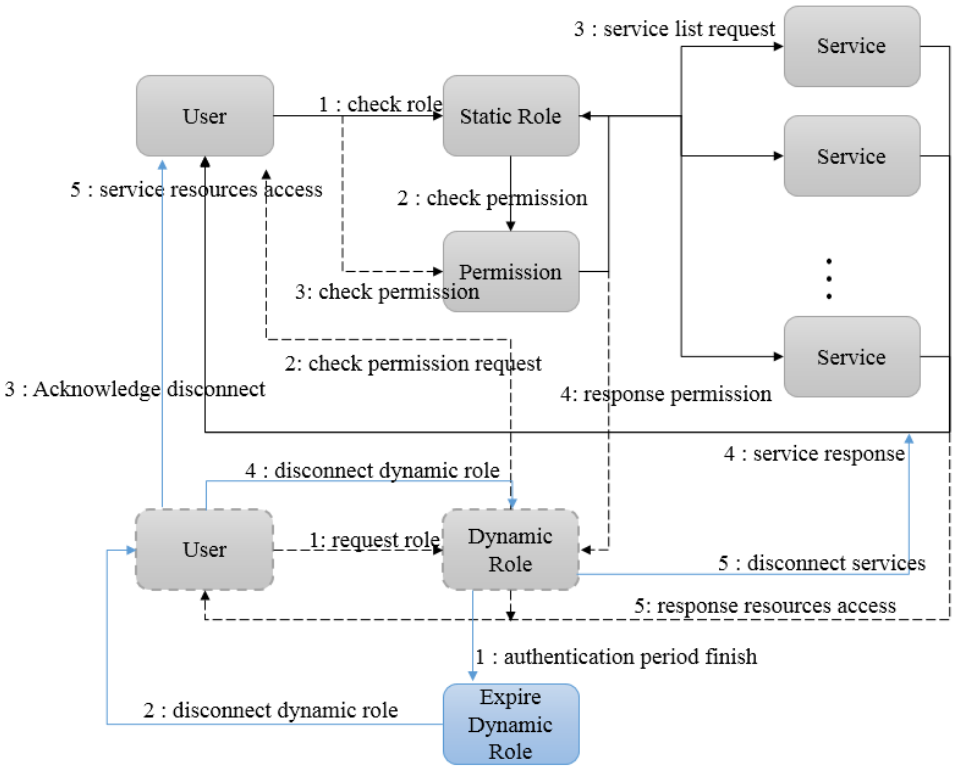

Figure 6. Service Process Control System on Dynamic Role

When a user accesses the system, the role that is basically provided can be defined as a static role. Therefore, the distribution of the

When the authentication process of the newly requested service is completed, the user is given a role for the newly requested service for a certain period of time, and can receive the service resources according to the role. In addition, when the service period ends, the user withdraws a role (Dynamic Role) temporarily granted to the user, and the service to the user is also stopped. A user who is granted a dynamic role will use some sort of token authentication. Therefore, as shown by the solid blue line in Figure 6, the user who has been granted the dynamic role has the authorization period for the role.

4. Analyze Dynamic RBAC System

Lots of systems have been proposed for accessing services or requesting service according to the role assigned to the user. Nevertheless, it has not been proposed deeply such that analyzes the role of the user and how to access the service, dynamically assigns the role to the user, analyzes the privileges and services granted to the role. In this paper, we analyze the correlation of services given to each user by role, and analyze the ripple effect of service in assigning role to each user.

The process of accessing and requesting services from a user with a static role is to provide the same services as an existing cloud-based Web system. When a user accesses the system, it analyzes the role of the connected user and provides a service that has a role is not changed. Therefore, the effect of the service spread to the user can be analyzed if the process of analyzing the service according to the role of the existing user is performed.

In addition, it does not deviate much from the existing service category. So that the part of the service which new combination of services, or effects of new form of services is excluded on role based ripple effect analysis algorithm of this paper.

4.1. Analysis Ripple Effects of User Service by Dynamic Roles

Assigning the role on dynamic role based system means that granting access to those services to the user or a whole team by temporarily for a specific period time while a user participates in a specific team, group, project, etc., and receives service.

Granting a dynamic role to a user goes through a process that responds to the user's request. First, when a user requests a new service, the user verifies whether the system requests an appropriate service internally. Second, the user service request authentication process is performed through an internal algorithm of how the ripple effect on the service providing role will affect the existing system.

$$
\begin{aligned}
&function\ analysisRippleEffectofRole\left(S''_{dynamic}\right)\{\\
&\quad r' \leftarrow \emptyset, R' \leftarrow \emptyset\\
&\quad probarbility\ \mathrm{pro1} = 1, pro2 = 0, pro3 = 1;\\
&\quad R' \leftarrow find\ \forall r' has\{\ P_{token}, s\}\\
&\quad for\ \{P_{token}, s'(u')\}\ in\ S''_{dynamic}\\
&\qquad authS(u) = \bigcup_{f(u' \to s'(u'))}^{f(s'(u') \to R')} S(u, P_{token})\\
&\quad end\ for\\
&\quad pro1\ \leftarrow P(authS(u)|R'|U \times R)\\
&\quad for\ r' in\ R'\\
&\qquad for\ S''_{dynamic} = \{\ P_{token}, s\}\ in\ r'\\
&\qquad\quad if\ \neg\ empty\ r'(s)\\
&\qquad\qquad if\ \neg \mathrm{P}(\{P_{token}, s\}|r'(s)) = 1\\
&\qquad\qquad pro2 \leftarrow \sum_{\emptyset}^{|r'(s)|} \mathrm{P}(\{P_{token}, s\}|r'(s))\\
&\qquad\qquad end\ if\\
&\qquad\quad end\ if\\
&\qquad end\ for\\
&\qquad pro3 = \prod_{\exists r'}^{|R'|} pro2\\
&\quad end\ for\\
&\quad if\ pro3 > 1\\
&\qquad pro1 = pro1 \times pro2\\
&\quad end\ if\\
&\quad probarbility \leftarrow \max\{pro1, pro3\}\\
&\quad return\ probarbility\\
&\}
\end{aligned}
$$

Algorithm 1. Analyze Ripple Effect of Dynamic Role

Algorithm 1 analyzes the ripple effect of service which user requested on the internal dynamic role based control system. It analyzes the effect to the overall system structure and the influence of other users using the same service. Analyzes the ripple effect on the current system based on the service requested by the user, and transmits the result to the user control client application. First, the impact on the overall system is measured by a probabilistic calculation based on the role of the service. The dynamic service requested by the user is searched for a role including the requested service. If the role exists, a probabilistic review of the user role is performed on the entire web service system. Second, the analysis of ripple effect on the requested service by the user is analyzed based on the role of using the requested service. It calculates the probability of the service used in the role, and analyzes the ripple effects based on the sum of the overall probability of the requested service. Finally, we analyze the role-based probability and the service probability, and then, analyze the results of the system-based overall ripple effect. The weight of the system ripple effect is examined, and the decision, whether a new dynamic role is created and assigned to the user, is made through an internal algorithm.

Whether to grant or deny new authority through appropriate analysis of the services is determined by the weight values set internally by the system. It analyzes whether the service requested by the user will have an effect on the user who uses the service in the whole system, and what kind of ripple effect will be caused in the role used in the existing system including the requested service. The process of granting the new dynamic authority is defined through the whole algorithm. Therefore, the system is designed and implemented to follow the process of authorization for new service request analysis and the ripple effect on the whole system. Users who are using the same service have been informed about the new role. We have tried to reduce the ripple effects on the process of creating new relationships in the existing authority control system.

The role provided by the implemented system is shown in the figure below. The service that can be provided by the role has a user authentication service (Permission Service) for authenticating another user. There is also a service resource provided to the user. A service unit includes a transaction up to the user's request and result confirmation.

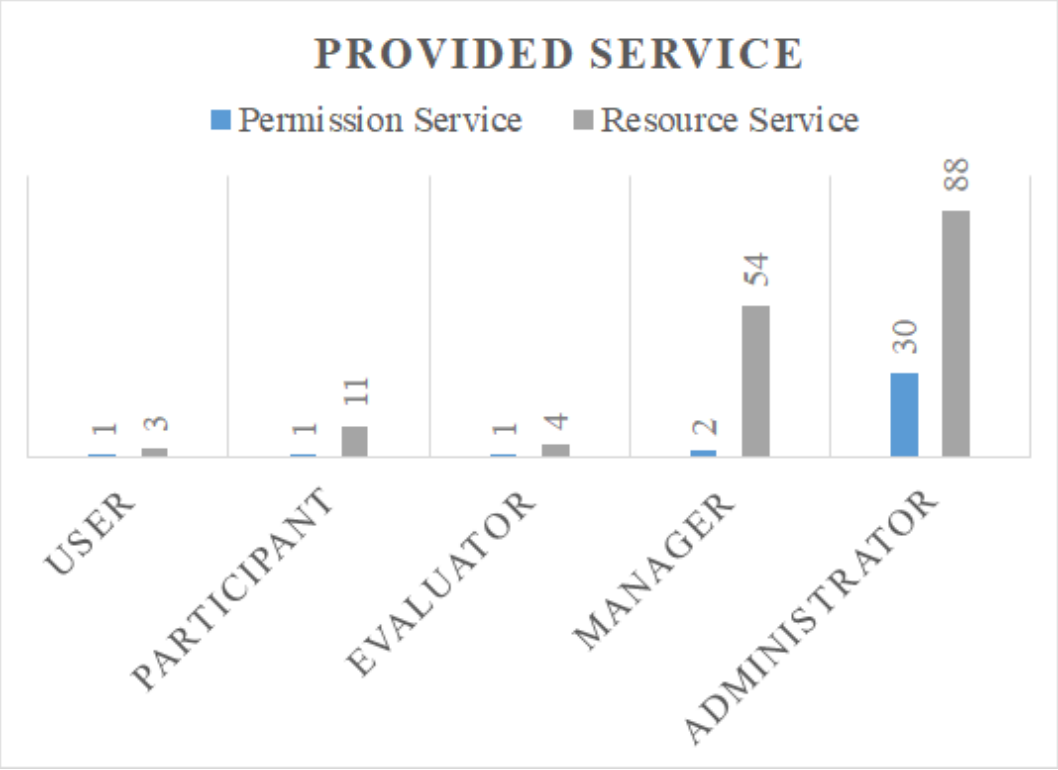

Figure 7. Provided Service on System

When each user makes a new service request, it checks whether the role (user, participant, evaluator, manager, administrator) having the service has the corresponding service, and then, generates a dynamic role through user authentication.

4.2. System Analyze

The dynamic authentication control system checks according to the user role dynamically controls the service required by the user, and the necessary permission in real time. After that, it was constructed to provide effective service to users. The dynamic authentication control system is provided for a team, which is a static role defined internally in the system, it is as shown in the following figure 8.

As shown in Figure 8, the static role consists of Team Member and Team Manager. In order to configure each team, the user is assigned to the team, the user is assigned to the role of the team member, and several team members are managed by the team manager. There has been a demand from users who temporarily participate in a team for a certain period and leave the process on the static role based control system. In this study, we assigned the role dynamically to the user that includes grant of the service use is given to the user for a certain period of time. Table 3 shows the system usage when using the dynamic authentication control system of user role based service.

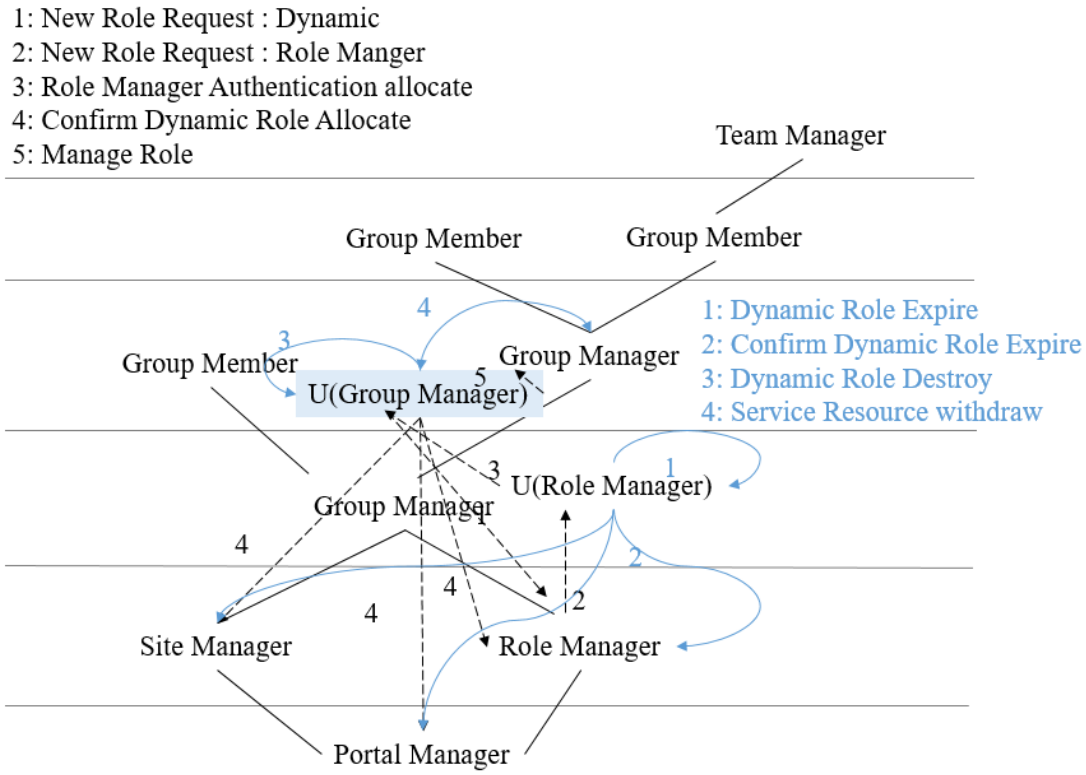

Figure 8. Dynamic Team Role Management

Table 3. System Usage

RBCA	Role	Permission	Service	Memory (MB)
Static	60	1920	112	31.35
Dynamic	53	2833	260	49.934375

Table 3 compares the system usage of the existing role-based service access control system (Static RBCA) and the dynamic rights-based service access control system (Dynamic RBCA). There are no significant differences in the number of actual roles of the dynamic role based service access control system and the existing static role based service access control system. Though, permissions and services have been increased, because users have been subdivided into access authority, and subdivide services according to the corresponding access authority. Likewise, as the authority and services have increased, the memory usage for the authority and services has increased.

Table 4 defines the procedures for providing services to users. In this paper, we propose a dynamic role based control system based on a static role based control system. A static role control system carries out a process of creating a new role, authority and service and allocating it to a user when a user needs a new role. In the dynamic role service control system, the process of creating and assigning roles, privileges, and services required by users is merged and used. The size of the overall procedure has been reduced in the dynamic service role control system, but the internal algorithm for assigning and analyzing the roles required internally is further complicated.

Table 4. System Usage

RBCA	Process No.	Process Step
Static	6	$P_{createRole}$, $P_{createPermisson}$, $P_{createService}$, $P_{assignRole}$, $P_{assignPermisson}$, $P_{assignService}$
Dynamic	3	$P_{createRole} \cup P_{assign}$, $P_{createPermission} \cup P_{assign}$, $P_{createService} \cup P_{assign}$

Described at Figure 9, the each user who has participant, manager, evaluator, and an administrator role has requested the service for a certain period. The users are allowed to participate in a particular team, group, project, etc., and using the service provided may temporarily grant the role to the user and provide the authority to access the service. It means to provide authority to access service resources.

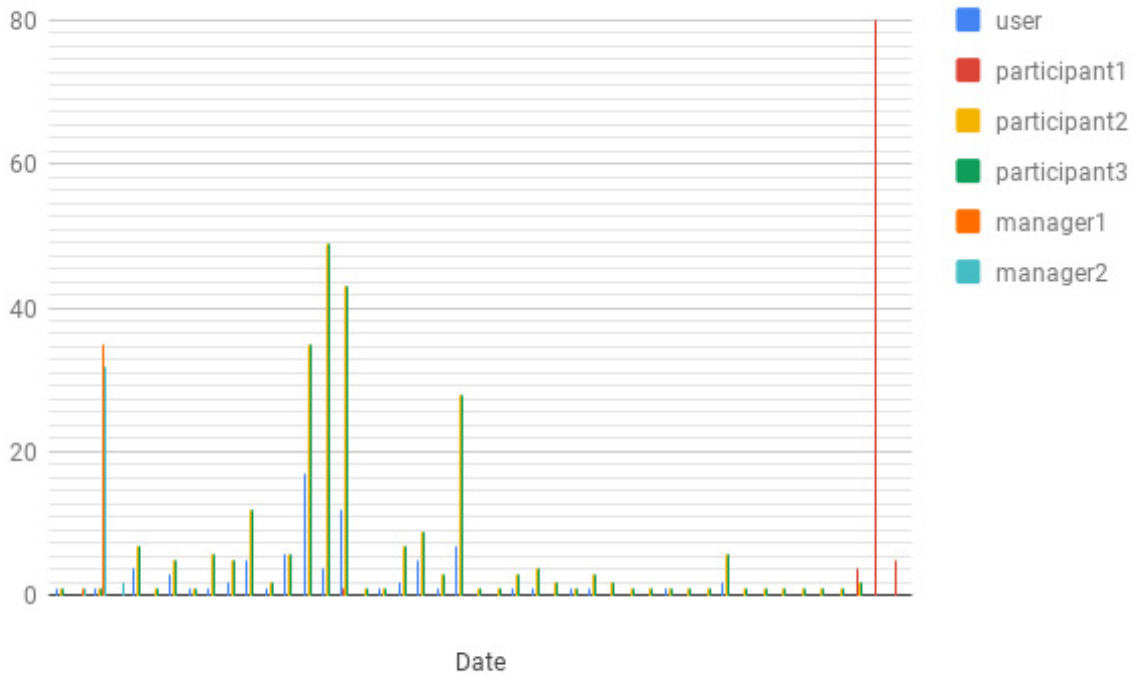

Figure 9. Service Request and Response on RBAC System

In the graph above, the horizontal axis represents a specific period. For each period, an end user can form a team to access the system. The domain of Figure 9 graph includes specialized service with role. Administrators and evaluators who have access to the created team can access share or modify the specified services. The dynamic role based access control system is constructed to perform a dynamic role during this specified period, and each granted role is automatically destroyed at the end of the above period.

5. Conclusions

So far, we have defined the dynamic configuration and control of services according to user roles, and defined procedures for dynamically assigning and retrieving roles according to the needs of various users. In the privilege control system according to the existing user role, the process of granting the appropriate role defined to the user and the authority and access control for the service according to the privilege granted to the user have been defined. Predefined roles are defined with respect to the service access privileges that allow users to access system resources, and each role is assigned to the user. When the user accesses the system, it analyzes the authority to access, and use the service resource according to the role assigned to the user, and provides the appropriate service resource to the user.

In this study, we analyzed various resource services provided to users in web based cloud computing system. We define procedures for dynamically constructing, providing and retrieving appropriate service systems for each user through the process of controlling the roles and service entitlements required for each user. We have issued minimum service access control tokens through service segmentation process based on various user needs and user roles. We defined the process of updating the access control system by creating the dynamic role of the user in the direction of minimizing the ripple effect by analyzing the system influence of the existing static role control system.

Future research is that analyze users' service usage using dynamic privileges defined by users and service usage according to existing static roles. We are going to compare and analyze the service resource access according to the roles of the user who uses the new user role and the user who has the existing static role. We will also develop new effects analysis and algorithms by comparing the amount of system resources used in the algorithm, the internal system load, and the amount of data required in granting new roles to existing role-based service access control systems. Next, compare the usage of dynamically requested service and service privilege, and provide prediction results on the requirements of the new user and the service providing effect of the system. As a goal, develop a calculation algorithm for the service resource used by the dynamic rights control algorithm and the service prediction probability for user service prediction.

Acknowledgements

This work was funded by the EDISON program through the National Research Foundation (NRF) of Korea funded by the Ministry of Science, ICT & Future Planning (NRF-2011-0020576).

REFERENCES

[1] Ferraiolo, D., Kuhn, D. R., & Chandramouli, R. (2003). Role-based access control. Artech House.

[2] Nash, M. J., & Poland, K. R. (1990, May). Some conundrums concerning separation of duty. In Research in Security and Privacy, 1990. Proceedings. 1990 IEEE Computer Society Symposium on (pp. 201-207). IEEE.

[3] Sandhu, R. S., Coyne, E. J., Feinstein, H. L., & Youman, C. E. (1996). Role-based access control models. Computer, 29(2), 38-47.

[4] Sandhu, R., Ferraiolo, D., & Kuhn, R. (2000, July). The NIST model for role-based access control: towards a unified standard. In ACM workshop on Role-based access control (Vol. 2000, pp. 1-11).

[5] Bacon, J., Eyers, D. M., Singh, J., & Pietzuch, P. R. (2008, July). Access control in publish/subscribe systems. In Proceedings of the second international conference on Distributed event-based systems (pp. 23-34). ACM.

[6] Xu, Z., & Stoller, S. D. (2012, June). Algorithms for mining meaningful roles. In Proceedings of the 17th ACM symposium on Access Control Models and Technologies (pp. 57-66). ACM.

[7] Wang, J., & Osborn, S. L. (2004, June). A role-based approach to access control for XML databases. In Proceedings of the ninth ACM symposium on Access control models and technologies (pp. 70-77). ACM.

[8] Chen, T. Y., Chen, Y. M., Chu, H. C., & Wang, C. B. (2007). Development of an access control model, system architecture and approaches for resource sharing in virtual enterprise. Computers in Industry, 58(1), 57-73.

[9] NICKLOUS, M.; HEPPER, Stefan. JSR 286: Portlet specification 2.0. Standard available (retrieved 2011-06-25) at http://www. jcp. org/en/jsr/detail, 2008.

[10] HEPPER, Stefan. JSR 286: Java portlet specification version 2.0. Java Community Process, 2008.

[11] SPECIFICATIONS, Introducing Java Portlet. JSR 168 and JSR 286. Nettiartikkeli lokakuu, 2008.

[12] REZGUI, Yacine. Role-based service-oriented implementation of a virtual enterprise: A case study in the construction sector. Computers in Industry, 2007, 58.1: 74-86.

[13] KIM, Tae-Young, et al. A modeling framework for agile and interoperable virtual enterprises. Computers in industry, 2006, 57.3: 204-217.

[14] STREMBECK, Mark. Scenario-driven role engineering. IEEE Security & Privacy, 2010, 8.1.

[15] MELL, Peter, et al. The NIST definition of cloud computing. 2011

[16] Zhang, L. J., Zhang, J., & Cai, H., Service-oriented architecture. Services Computing, pp. 89-113, 2007.

[17] Laskey, K. B., & Laskey, K., Service oriented architecture. Wiley Interdisciplinary Reviews: Computational Statistics, vol. 1(1), pp. 101-105, 2009.

[18] Kyriazis, Dimosthenis, et al., A real-time service oriented infrastructure. GSTF Journal on Computing (JoC) 1.2, 2018.