한국시뮬레이션학회 논문지
Vol. 15, No. 4, pp. 21-28 (2006. 12)

모바일 그리드 어플리케이션을 위한 커널 기반 관리방안에 대한 연구

김태경[1†] · 서희석[2]

A Study on the Kernel-Based Management for Mobile Grid Application

Kim Tae-Kyung · Seo Hee-Seok

ABSTRACT

In this paper, we introduce the mobile grid architecture using kernel-based managements. Mobile grid paradigm is used to allow mobile users to access distributed resources in a transparent, secure and effective way. Each mobile grid user requests job submission, searches for suitable resources to run job, executes the job to the allocated system resources, and receives the results of the requested job. There are a number of challenging issues when taking mobile environment into account, such as limited performance of mobile devices and mobility of mobile device. To resolve these issues, we design and implement the kernel-based management system for mobile grid application to overcome the limited resources of mobile devices. Also, we show the efficiency of the suggested architecture using the performance evaluation.

Key words : Mobile Grid, Management, Kernel, Performance Evaluation

요 약

본 논문에서는 커널기반 관리 방식의 모바일 그리드의 구조를 제시하였다. 모바일 그리드 패러다임은 모바일 사용자들로 하여금 효과적이고 안전하면서 투명하게 분산된 자원에 접근할 수 있도록 한다. 각각의 모바일 사용자가 작업의 수행을 요청하면, 작업을 수행하기에 적합한 자원을 찾고, 할당된 시스템에서 작업을 수행하며, 수행된 결과를 작업의 요청자에게 전달하게 된다. 그러나 모바일 그리드 환경에서는 한정된 모바일 장비의 성능과 모바일 장비의 이동성 등의 제한점들을 고려해야 한다. 그래서 본 논문에서는 모바일 장비의 한정된 자원의 제한성을 극복하기 위해서 커널기반의 모바일 그리드 어플리케이션을 위한 관리 시스템을 설계 및 구현하였으며, 성능평가를 통하여 제안한 구조의 효율성을 제시하였다.

주요어 : 모바일 그리드, 관리, 커널, 성능평가

1. Introduction

Adapting wireless devices to communicate to Grid networks enables an extensive rich set of capabilities which devices can utilize as a consumer or even as a publisher for other peer device utilization. Modern computer applications are increasingly concerned with utilizing distributed resources, like processors and storage. The Grid is a computing and data management infrastructure that will provide the electronic basis for a global society in business, government, research, science, and entertainment. Grids integrate networking, communication, computation, and information to provide a virtual platform for computation and data management in the same way that the Internet integrates resources to form a virtual platform for information. Namely, Grid infrastructure will provide us

2006년 7월 10일 접수, 2006년 11월 15일 채택
1) 서일대학교
2) 한국기술교육대학교
주 저 자 : 김태경
교신저자 : 서희석
E-mail; tkkim@seoil.ac.kr, histone@kut.ac.kr

with the ability to dynamically link together resources as an ensemble to support the execution of large-scale, resource-intensive, and distributed applications[1]. Grid network assumes the systems are placed in the designated space and managed by administrators to provide stable resources. This assumption is challenged by the wireless mobile environment.

Mobile computing[2] is another distinct paradigm of traditional distributed system, considering mobility, portability and wireless communications. Mobile users might be the future users of mobile grid network. The mobile computing environment is broad and varied in the technologies and capabilities that are available. There are many challenges when we consider mobile devices as one of grid computing resources or interfaces. For examples, this is not constant bandwidth as is typical of wired network; wireless devices are characterized by intermittent connectivity and faults due to the increased impact of noise and signal degradation. Issues such as the duration and frequency of network connections, as well as mobility, must be addressed. Moreover, these disconnects may be voluntary or involuntary, leaving it up to the Grid system software to behave gracefully even without knowledge of these frequent network outages[3]. To solve these limitations and constraints, we suggest the mobile grid architecture using kernel-based management. Kernel-based management can offer reduced processing power in reduced time.

In this paper, we focus on suggesting the architecture of mobile grid which can provide the grid service efficiently. The theme of mobile communication has been anytime, anywhere access. In section2, related works are described. Section 3 presents the architecture of mobile grid and implementation of mobile grid management system using kernel based method. In section 4, we show the efficiency of mobile grid system and kernel based management system. Section 5 summarizes our work and identifies future research directions.

2. Related Works

This section introduces the mobile grid systems. Research issues like enabling mobile devices to perform Grid operations or be a part of the clustered grid network are being currently explored.

Foster and Iamnitchi [4] have identified problematic similarities between P2P and Grid computing. They argue that Grid computing has added to the notion of a persistent, standards-based service infrastructure while P2P computing has addressed resource sharing in the face of unreliable networks and consumer devices. The success of the SETI@home project[5] proves that large P2P systems can effectively be deployed in a high performance environment by aggregating the unused cycles of desktop PCs. In a P2P system, users are free to join and leave the network at any time. Thomas Phan et al.[6] favored proxy based clustered system architecture with favorable deployment, interoperability, scalability, adaptivity, and fault-tolerance characteristics as well as an economic model to stimulate future research in this emerging field. Dan C. M. et al. [7] introduced ad hoc grids as hierarchy of mobile devices with different computing and communication capabilities. They describe the generic architecture of an ad hoc grid and outline the communication architecture. Also, discuss briefly power consumption models for different types of activities and sketch an agent-based power management system for ad hoc grids. M. Fukuda et al.[8] proposed a mobile agent based middleware that mainly benefits remote desktop computer users as a mutual computing resource exchange. The project combines the enhancement of a mobile agent platform and the behavioral design of mobile agents to keep allocating computing resources available to a user job. J. Hwang et al.[9] present middleware architecture capable of integrating mobile devices with existing grid platforms to conduct peer-to -peer operations. They include a list of application scenarios to which their middleware architecture can be put to maximum use. D. Bruneo et al.[10] analyzed the issues related to mobile users exploiting the potential of Grid computing. They presented mobile agent based architecture, which can manage the transparency of the access to the distributed services on one hand, and the user's mobility on the other hand. Also, they investigated the performance offered from different communication paradigms through Petri net models.

These systems'design principle overlaps with our proposed architecture in terms of using mobile devices however the originality of our approach lies in use of kernel based management technology to manage the mobile grid system in constrained resource environments

3. Designs and Implementation of Mobile Grid Architecture

Mobile users introduce new challenges in the Grid areas. Mobile terminals have different nature, ranging from powerful laptops to mobile phones with different capabilities in terms of maximum transmission power, energy availability, mobility patterns, and QoS requirements. This means that the Grid can rely on some terminals for full job generation and/or result presentation. But other ones can hardly perform any task without draining their batteries. And there is a variety

Table 1. A four-level model of grid network system

	Products	mobility	Characteristics
Level 1	Desktop computer Server	Not support	High speed network access Sufficient secondary storage Sufficient electronic power
Level 2	Laptop computer	Support	Somewhat secondary storage Long life battery
Level 3	PDA	Support	Little or no secondary storage Middle life battery
Level 4	Mobile phone	support	Little or no secondary storage short life battery

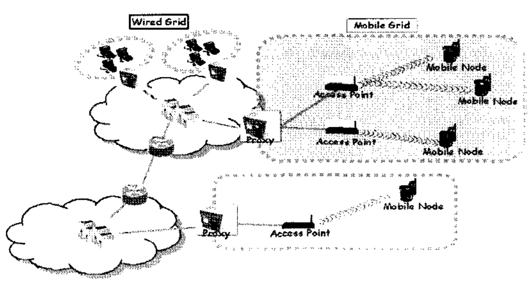


Figure 1. Mobile Grid Architecture

of user interfaces. PDAs can execute programs, but mobile phones are limited to specific application clients. Mobile devices can be separated into 4 types [7] according to the I/O bandwidth, battery power, CPU cycles and memory.

We designed the mobile grid architecture considering all kinds of mobile devices. Level-1 systems use common grid architecture and services. This level systems use GT-3 (Globus Toolkit 3) middleware to run the grid applications. Level-2 systems do not install the GT-3 middleware and should manage the security and reliability problems of mobility. Level-1 and Level-2 systems can implement complex behavior and computation intensive tasks. Level-3 systems are suffering from the insufficient power and storage. So they can't have enough power to run computation intensive tasks. Level-4 type devices can deploy only simple reactive agents. While these agents do not have the power to implement formal model of communication, they can still communicate using very simple messages and relying on the emerging properties of the system to achieve a higher level goal.[7] Level-1 and Level-2 systems do not install "Globus" [11] requires substantial memory and storage requirements, which eliminates possibilities for wireless devices to run a Globus API.

3.1 Design of Mobile Grid System

As shown in Figure 1, we design to use grid service based on proxy based system[6] in wired and wireless networks. The architecture of mobile grid can be separated into two main sub systems such as a proxy and mobile node systems.

■ **Proxy system**

The function of a proxy system can be described below: First, the user authenticator processes the authentication procedure using the mobile device user's information from the user database. The resource information database contains the information regarding the currently connected mobile devices; the job information database contains the information of jobs and the full path name for job execution and distribution. The job management system receives job requests from the mobile devices and composes the job distribution policy using the job scheduler. The job management system distributes the jobs to mobile devices using the MPICH module. When the sub jobs are collected by the MPICH module, it returns the results to the mobile devices. The communicator is used for socket communication between the mobile device and the proxy systems.

■ **Mobile device system**

The function of mobile device system can be illustrated below: The user interface supports the various kinds of information encompassing the creation of a mobile grid user's job process to the completion of a specific job. The super daemon performs the communication functions with the communicator of the proxy system. The job execution module executes the functions of distributed computing for the mobile grid application. The system state management module provides the resource information to the user and the proxy system. The resource manager collects the information regarding the system and available resources, more specifically the mobile devices utilizing the system kernel. The node controller controls the functions of mobile devices. The MGP (Mobile Grid Protocol) module is the communication module used in the

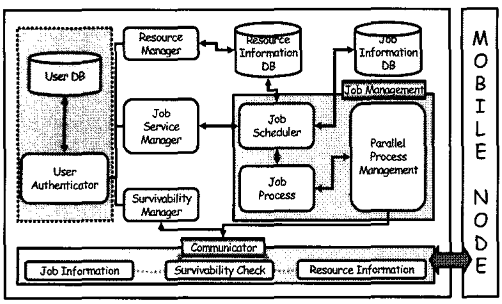


Figure 2. Design of proxy system

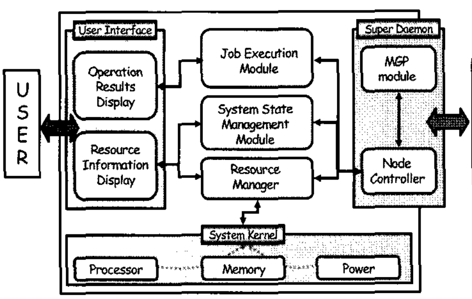


Figure 3. Design of mobile device system

mobile grid.

The detailed operation of MGP module is shown in Figure 4. The main function of MGP can be classified like this:

■ Security function: security function processes mutual authentication and authorization between proxy system and mobile device to check the proper user or authorized system resources

■ Resource management function : proxy system must check the available resources of mobile devices to run distributed application and network connectivity between proxy system and mobile devices

■ Communication function: MPICH module is used to distribute mobile grid application to the available resources of mobile devices and to obtain the results of the requested sub jobs of mobile grid application.

3.2 Design of Linux Kernel Based Information Collector

Kernel based Information provider collects the information of network and system resources which is automatically accumulated in kernel layer, calculate the useful information such as CPU (user, nice, system, idle), memory, network information (bandwidth, jitter, delay, loss, and end-to-end performance) etc. This information provider collects the information of each protocol stack using the suggested protocol stack hooking layer in kernel.

Kernel based network information gathering mechanism is shown in Figure 5. Each process of information gathering mechanism is like these :

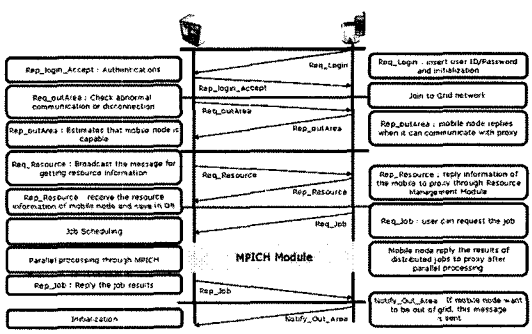


Figure 4. Mobile Grid Communication Module (MGP)

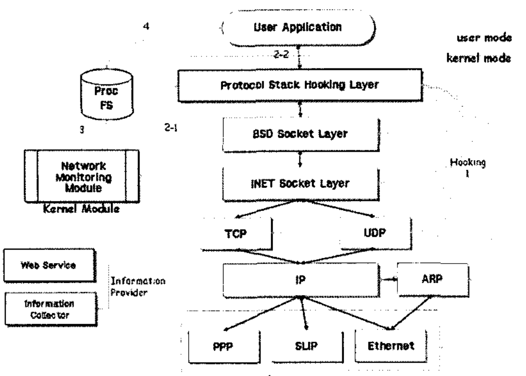


Figure 5. Kernel based network information gathering mechanism

▪ **Hooking**

It replaces the existing protocol stack logic that gathers network related information in the abstract with the logic that gathers network related information in detail. (ex : End-to-End bandwidth)

▪ **Information gathering using kernel module**

It gathers information from protocol stack hooking layer. Protocol stack hooking layer hooks each protocol stack and stores network related information after processing into user-readable format.

▪ **Information gathering using kernel memory interface**

An interface is supported that user can access to the kernel data through it. (ex: Kernel Virtual Memory interface library (KVM))

▪ **Data accumulating**

Kernel module stores data into the file-system that can be used by user at user level. By using ProcFS (Process information pseudo-File system) we can reduce the load that should be occurred by using real file systems.

▪ **Information processing**

User application reads network monitoring parameters from ProcFS and processes them as network parameter for Grid applications.

Protocol stack hooking layer uses Netfilter Layer that is supported on the Linux kernel2.4.X to 2.6.X. Netfilter layer supports to hook in the protocol stack by using user supplemental functions. It does not modify the protocol stack code, so it can process information that kernel uses without modification of original kernel data. Kernel module is based on the "ip_conntrack" kernel module supplied by Netfilter layer. Some codes are added and modified to gather and process user specific network parameter in detail. Information Collector daemon processes the network related information in the ProcFS and it encodes gathered information with XML scheme and send to the Web Service[12] application. Messaging Protocol uses XML (eXtensible Markup Language) and SOAP (Simple Object Access Protocol) [13] to communicate with each service.

The communication message protocol for kernel based management system is shown in Figure 6.

3.3 Implementation of Linux Kernel Based Management System for Mobile Grid

We implemented the mobile grid system with 4 mobile nodes (PDA) and a server system. The implementation environment and service platform are depicted below:

▪ **Proxy System**

CPU: Intel Pentium III 1GHz, RAM: SDram 256MB, O.S: Redhat Linux 7.3, Kernel 2.4.18-3, Compiler: gcc version 2.96, Database: mysql Ver 11.16 Distrib 3.23.49

▪ **Mobile Device (PDA) System**

Hardware: Sharp Zaurus SL-6000/C860 (CPU 400 MHz), CDE (Cross Development Environment): Linux Kernel, binutils, gcc, glibc, Qt/Embedded - Cross Development Kit

▪ **Service Platform**

JAVA WSDP (Web Services Developer Pack) JAXM (Java API for XML Messaging) / JAVA

We can identify the normal running status of web service using the web browser for monitoring the mobile device as shown in Figure 7.

The implemented proxy and mobile device systems is shown in Figure 8.The proxy system shows the log file of job processing. The mobile node GUI shows the login, resource information, and job request user interface.

The implemented kernel based mobile grid management system is shown in Figure 9. This Figure shows the system, network information of available mobile

```
<?xml version="1.0" encoding="UTF-8"?>
<Message_root>
<NodeInfo>
        <IP>IP Address</IP>
        <OS>Operating System</OS>
        <IFSpeed>Interface Speed</IFSpeed>
</NodeInfo>
<SystemParam>
        <CPU>
                <user>user</user>
                <sys>system</sys>
                <nice>nice</nice>
                <idle>idle</idle>
        </CPU>
        <MEM>
                <av>average</av>
                <used>used</used>
                <free>free</free>
                <shrd>shared</shrd>
                <buff>buffer</buff>
        </MEM>
</SystemParam>
<NetworkParam>
        <Traffic>
                <in>incoming bytes/sec</in>
                <out>outgoing bytes/sec</out>
        </Traffic>
        <Bandwidth>
                <prot>protocol</prot>
                <src>source address</src>
                <dst>destination address</dst>
                <sport>source port</sport>
                <dport>destination port</dport>
                <inpkt>inpacket</inpkt>
        ..................................................
```

Figure 6. Example of Communication Protocol for Kernel based Management

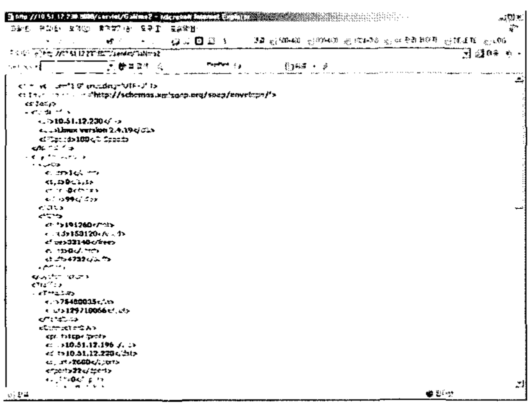


Figure 7. Checking the web service working with web browser

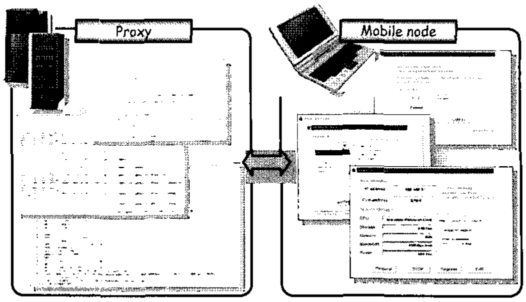


Figure 8. Implemented Mobile Grid System

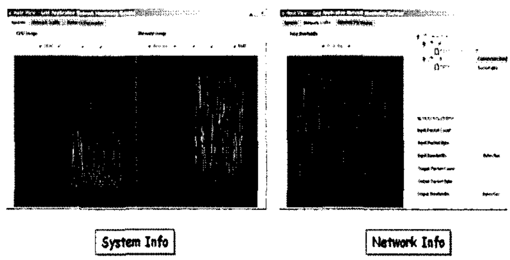


Figure 9. Implemented kernel based mobile grid management system

nodes.

This kernel based mobile grid management system can also provide the statistics information of system and network usage resources. This system can provide the usage information of system and network resources to the each grid application by measuring the real time traffic and systems.

4. Performance Evaluation of Kernel Based Management System

At first, we evaluate the performance of mobile grid system concerning to the job response time. In this simulation, 4 mobile nodes were used to calculate the prime numbers (primem.c). As shown in Figure 10, we executed the mobile grid application on 1 mobile node and calculate the job response time. Then we executed the mobile grid application on from 2 to 4 mobile nodes and calculated the job response time. As the numbers of participating mobile nodes were increased,

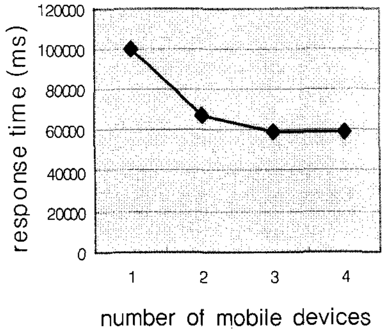


Figure 10. Response Time of Mobile Grid Application when using 1 ~ 4 mobile nodes

the job response time was decreased.

To show the efficiency of kernel based mobile grid management system, we compare the performance of monitoring load of kernel based method with that of Libpcap based method. Libpcap is the Packet Capture library which provides a high level interface to packet capture systems and provides a portable framework for low-level network monitoring. It can provide the information of TCP, UDP, and IP header.

The process of packet capturing and traffic analysis uses the mobile system resources such as CPU and memory etc. Mobile node has constrained processing power so the overload of management process can cause the job failure due to lack of resources. In case of performance estimation methods on kernel layer use a little system resources by using the accumulated information in OS. Therefore, this method does not use the system resources to capture packets and analyze the data. General monitoring system gathers the management information using system call but kernel based methods don't necessary to interrupt system calls.

The performance evaluation concerning to the CPU usage is shown in Table 2.

Kernel based monitoring system use a less CPU resource than Libpcap based method. Therefore we can efficiently manage the mobile device using the kernel based monitoring methods. Also the elapsed time of kernel based method for monitoring the mobile device was shorter than that of the Libpcap based method.

Table 2. Occupied CPU load for kernel based and Libpcap based methods

Monitoring method	Occupied load for CPU
Libpcap based method	31.8 %
Kernel based method	0.2 %

5. Conclusion and Future Works

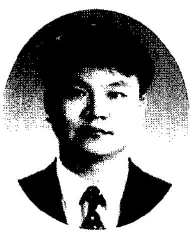
Grid infrastructure provides us with the ability to dynamically link resources together in a shared architecture to enable the implementation and sustained support of large-scale, resource-intensive, distributed applications. The mobile Grid architecture creates complicated issues when used over networks because of the requirement of providing services to users over multiple networks, regardless of location. In this paper, we design and implement the mobile grid system and kernel based monitoring system to reliably provide the grid service anywhere, anytime, and any place. As shown in chapter 4, mobile grid can provide grid service efficiently. Also using the kernel based monitoring system, mobile grid can overcome the limited resource of mobile grid. This system can provide the resource usage of each grid application to allocate the proper network and system resource.

We intend to adapt the AAA protocol into mobile grid system to provide security service efficiently using context transfer technology. Also we will research the job migration in mobile grid environments.

References

1. F. Berman, G. Fox, T. Hey, The Grid: past, present, future, Grid Computing - Making the Global Infrastructure a Reality, Wiley and Sons.
2. M. Satyanarayanan. "Fundamental Challenges in Mobile Computing," In proceedings of the fifteenth annual ACM Symposium on Principles of Distributed Computing, Philadelphia, Pennsylvania, 1996.
3. B. Clarke, M. Humphery, "Beyond the "Device as Portal": Meeting the Requirements of Wireless and Mobile Devices in the Legion Grid Computing System," Proceedings of the International Parallel and Distributed

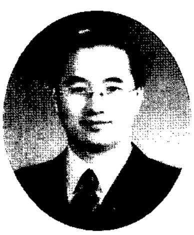
Processing Symposium (IPDPS'02), 2002.
4. I. C. Foster and A. Imanitchi, "On Death, Taxes and the Convergence of Peer-to-Peer and Grid Computing", 2nd International Workshop on Peer-to-Peer Systems.
5. SETI@home. http://setiathome.ssl.berkeley. edu, March 2001.
6. Thomas Phan, Lloyd Huang, and Chris Dulan, "Challenge: Integrating Mobile Wireless Devices Into the Computational Grid", MOBICOM 02, September 23~26, Atlanta Georgia, USA.
7. D. C. Marinescu, G. M. Marinescu, Y. Ji, and L. Boloni, "Ad Hoc Grids: Communication and computing in a Power Constrained Environment," Workshop on Energy-fficient Wireless Communications and Networks (EWCN) 2003, Phoenix, USA, 2003.
8. M. Fukuda, Y. Tanaka, N. Suzuki, L. F. Bic, "A Mobile- gent-Based PC Grid," Autonomic Computing Workshop Fifth Annual International Workshop on Active Middleware Services (AMS'03), Seattle, Washington, June 2003.
9. J. Hwang, P. Aravamudham, "Proxy-based Middleware Services for Peer-to-Peer Computing in Virtually Clustered Wireless Grid Networks," International Conference on Computer, Communication and Control Technologies (CCCT'03), Orlando, Florida, July 2003.
10. D. Bruneo, M. Scarpa, A. Zaia, and A. Puliafito, "Communication Paradigms for Mobile Grid Users," The 3rd International Symposium on Cluster Computing and the Grid (CCGRID2003), 2003.
11. I. Foster, K. Kasselman, The Globus Project: A status report, IPPS/SPDP'98 Heterogeneous Computing Workshop S.4-18 (1998).
12. Web Services: W3C Architecture Domain, http://www.w3.org/2002/ws.
13. SOAP Framework: W3C Simple Object Access Protocol Version 1.2, June 2003, http://www.w3.org/TR/SOAP.



김 태 경 (tkkim@seoil.ac.kr)

1996년 기아정보시스템 사원
1997년 단국대학교 수학교육과 (이학사)
2001년 성균관대학교 전기전자및컴퓨터공학과 (공학석사)
2003년~2004년 (주) 스페이스센싱 선임연구원
2005년 성균관대학교 전기전자및컴퓨터공학과 (공학박사)
2005년~2006년 성균관대학교 정보보호 인증기술연구센터 박사후연구원
2006년~현재 서일대학 정보기술계열 정보전자전공 교수

관심분야 : 네트워크보안, 그리드 네트워크, USN



서 희 석 (histone@kut.ac.kr)

2000년 성균관대학교 산업공학과 (공학사)
2002년 성균관대학교 전기전자및컴퓨터공학과 (공학석사)
2004년 (주)정보감리평가원 선임연구원
2005년 성균관대학교 전기전자및컴퓨터공학과 (공학박사)
2005년~현재 한국기술교육대학교 인터넷미디어공학부 정보보호전공 교수

관심분야 : 네트워크보안, 보안 시뮬레이션, USN