

실시간 무선 원격제어 시스템의 설계 및 구현

이 문 구, *박 성 원
김포대학 컴퓨터계열, *(주) NCERTI
전화 : 031-999-4660 / 핸드폰 : 019-386-4407

Design and Implementation of Real-time Remote Control System

Moon Ku Lee, Seong Won Park
Dept. of Computer Science, Kimpo College
E-mail : ycon0330@kimpo.ac.kr

Abstract

Existing system management software solutions show limitations in time and space, as well as problems such as uncertain error messages, and also difficulty providing swift assistance or real time emergency support. Therefore, a wireless remote control system has been designed and implemented in this thesis, which is capable of managing and monitoring remote systems using mobile communication devices (Mobile Phone, PDA, Smart Phone, Webpad) for instantaneous control. The implemented real time wireless remote control system provides remote server management functions, error or event message functions, log record functions, authentication function via mobile devices.

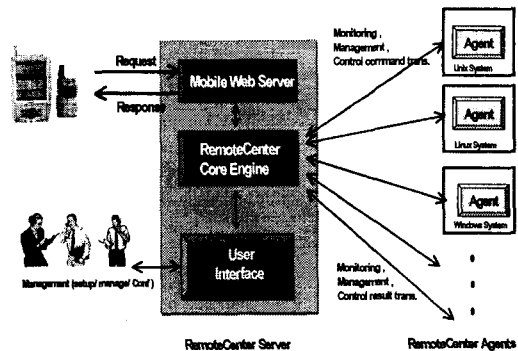
I. 서론

기존의 웹 기반인 유선상의 시스템 관리 소프트웨어 솔루션들은 시간적, 공간적 제약뿐만 아니라 오류사항에 대한 통보가 불확실하거나 신속한 지원요구 및 실시간 긴급조치가 어렵다는 문제점들을 갖는다. 따라서 본 논문에서는 모바일 통신기기(휴대폰, PDA, Smart Phone, Webpad)를 이용하여 원격지의 시스템을 관리 및 모니터링하고 즉각적으로 제어할 수 있는 실시간 무선 원격제어 시스템을 설계하고 구현하였다. 또한 구현된 실시간 무선원격 제어 시스템은 최근에

이슈가 되고 있는 사용자 인증문제와 시스템의 보안문제를 해결하고자 모바일 웹 서버와 원격제어서버의 코어 엔진 모듈에서 상호간의 인증과정이 실시되도록 하였다. 때문에 인증되지 않은 제 3자가 시스템에 접근할 수 없도록 하였으며, 실제로 접근이 허용된 사용자라도 대상 시스템을 제어하기 위해서는 접근권한 허용을 위하여 원격지 서버와 대상 시스템간의 인증과정을 실행하도록 하였다.

II. 제안하는 무선 원격제어 시스템

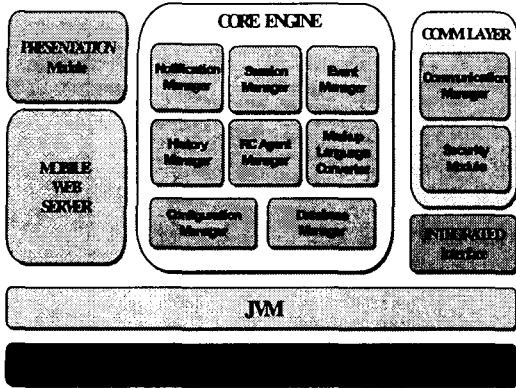
2.1 시스템의 구성 요소



[그림 1] 무선 원격제어 시스템의 구조도
제안하는 무선 원격제어 시스템은 크게 모바일 장비, 원격지 서버 그리고 대상 시스템으로 구성된다.

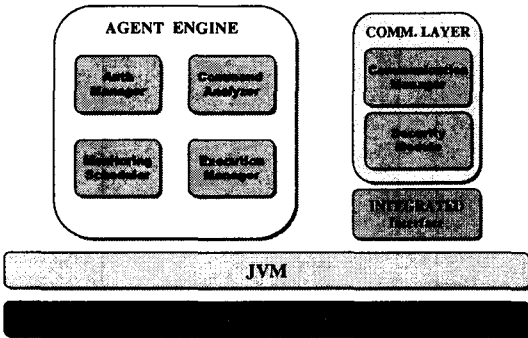
2.2 원격지 제어서버의 구조와 모듈별 주요기능
1) 모바일 웹 서버(Web server)는 모바일 단말기 상에서 원격지 시스템의 모니터링 결과와 실행명령에 대한

출력결과를 모바일 단말기의 기종에 적절한 형태로 출력해 주는 기능을 제공하기 위하여 표현 모듈(Presentation Module)을 탑재한다.



[그림 2] 원격지 제어서버의 구조

2) 코어 엔진(Core Engine)은 다음의 모듈들로 구성된다.



[그림 3] 원격지 서버 에이전트의 구조

- 통보 관리(Notification Manager) 모듈은 원격지 시스템의 오류상태 발생시 관리자 또는 지정된 사용자에게 이메일 또는 단문메시지 형태로 통보하는 기능을 제공한다.
- 세션 관리자(Session Manager) 모듈은 모바일 단말기를 통하여 접속한 사용자의 세션 관리 기능을 제공한다.
- 이벤트 관리자(Event Manager) 모듈은 원격지 시스템에서 발생한 각종 이벤트를 분석하여 데이터베이스에 기록과 동시에 통보여부를 판단하여 통보 관리자(Notification Manager)에게 전달하는 기능을 제공한다.
- 기록 관리자(History Manager) 모듈은 모바일 단말기를 통하여 원격지 시스템에 제어기능 수행 시 이에

대한 각종 기록(Log File) 정보를 관리하는 기능을 제공한다.

- 에이전트 관리자(Agent Manager) 모듈은 제어서버를 통하여 관리대상이 되는 원격지 시스템 에이전트들의 상태 등을 체크하는 기능을 제공한다.
- 마크업 언어 변환기(Markup Language Converter) 모듈은 접속한 모바일 단말기의 종류를 자동 판별하여 해당단말기에서 수용 가능한 마크업 언어(mHTML, WML, HDML 등)로 자동 변환하여 주는 기능을 제공한다.
- 환경구성 관리자(Configuration Manager) 모듈은 사용자 등록, 원격지 시스템 정보 등록, 모니터링 항목의 임계값 설정 등 각종 구성 정보를 등록 관리하는 기능을 제공한다.
- 데이터베이스 관리자(Database Manager) 모듈은 제어서버에서 사용하는 원격 데이터베이스 관리 시스템에 접속하여, 각종 SQL 질의(Query)를 담당하는 기능을 제공한다.
- 통신 관리자(Communication Manager) 모듈은 원격지 제어서버와 대상 시스템간의 통신 기능을 담당하며, 원격지 서버로부터 전송된 에이전트와 대상 시스템간의 통신에서 SSL(Secure Socket Layer)방식으로 통신이 이루어진다.
- 보안 모듈(Security Module)은 단말기와 제어서버 그리고 원격지 서버 에이전트와 대상 시스템간의 통신에서 인증 및 접근제어 등 보안 기능을 제공 및 관리한다[1].

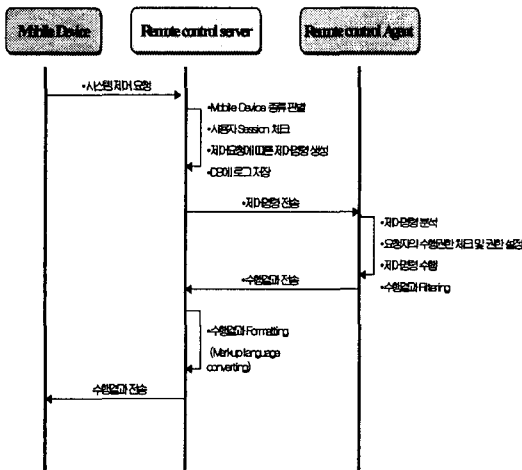
3) 응용 프로토콜 인터페이스(Application Protocol Interface) 모듈은 다른 유선상의 시스템 관리 소프트웨어 솔루션들과의 연동을 위한 인터페이스 기능을 제공한다.

III. 무선원격제어시스템의 알고리즘

3.1 모바일 원격제어시스템의 실행

모바일 장비와 원격지 제어 서버 그리고 에이전트가 상호 동작하는 과정은 [그림 4]와 같다. 원격지 제어 서버의 웹 서버는 JSP 엔진기능이 제공되며, 모바일 장비의 종류를 판별하고, 모바일 장비(예, PDA)로부터 시스템 제어요청(request)을 받아서 원격지 제어서버의 엔진으로 처리를 넘기고, 그 결과를 응답(response) 하는 부분을 담당한다[5][6]. 이때, 모바일 장비의 종류를 판별하고, 사용자의 세션을 체크한 후 제어 요청에 따른 제어 명령을 생성하면서 데이터베이스 연결모듈을

통하여 각종 데이터베이스 정보를 질의 및 업데이트 모듈로 처리한 후 데이터베이스에 로그 파일을 저장한 후, 제어 명령을 에이전트로 전송된다. 원격지 제어 서버의 에이전트 엔진은 제어명령을 분석하고 요청자의 수행권한 체크 및 권한 설정을 하기위하여 정보를 파싱(parsing)하여 정보 추출, 객체에 저장하는 페이지 퍼메팅(Page formatting), 원격지 에이전트의 alive 상태 체크, 이벤트 발생시 전자메일과 유선관리시스템 전송, 사용자 세션 관리, 기타 데이터의 분석 및 가공을 위한 대부분의 로직 그리고 사용자 인터페이스 모듈등과 같은 역할을 담당한다. 원격지 에이전트모듈에서 각 에이전트들은 각 원격지 서버의 노드(host)들에 설치되며, 이는 원격지 콘솔과 통신, 모니터링 데이터의 추출역할 그리고 제어명령을 분석하고, 요청자의 수행권한 체크 및 권한 설정, 제어명령 수행 그리고 수행결과를 필터링하여 수행결과를 원격지 서버에 전송하도록 명령어 라인 인터페이스(Command Line Interface)를 사용한다.



[그림 4] 모바일 원격 시스템 제어의 실행과정

모바일 장비의 특성상 기존의 세션 관리 기법으로는 세션 관리가 힘들다. 따라서 사용자 세션 관리 부분이 별도로 구축되어야 하며, 세션 관리를 위하여 [그림 5]와 같이 제어 서버의 사용자 세션 관리를 위한 주요 알고리즘이 구현되어야 한다. 세션 ID에 해당하는 키 값은 사용자가 제어서버에 접속한 최종시간을 기준으로 하며, 이 값은 모바일 장비와 제어서버 간에 통신 시 파라미터로 전송되어 모바일 세션을 관리하도록 한다[6].

[그림 6]은 원격제어서버와 대상시스템의 통신부분 알고리즘이다. 시스템 제어에 있어 접근권한 및 명령

어 수행권한 등을 체크하기 위해서는 반드시 사용자 ID와 패스워드 정보가 필요하며, 이에 대한 정보의 교환은 통신을 통해 이루어진다. 따라서 통신시 데이터의 암호화는 필수적인 부분으로 SSL 프로토콜을 이용하여 통신이 이루어지고, 대상 시스템과 제어명령 및 명령어 수행결과를 주고받는 방법(method)의 구현이 필요하며, 이 방법을 통해 원격제어 에이전트와 데이터통신이 이루어진다.

```

// Package import
.....
public class RCServer{
.....
/* 호스트명과 포트번호를 이용하여 대상서버에 접근할 RCServer객체를 만든다. */
public RCServer(String hostName, int port)
{
    try{
        // SSL protocol 암호화 설정
        .....
        SSLSocketFactory ssf = context.getSocketFactory();

        // SSL Socket Connection 설정
        rcSocket = ssf.createSocket(InetAddress.getByName(hostName), port);
        out = new PrintWriter(rcSocket.getOutputStream(), true);
        in = new BufferedReader(new InputStreamReader(rcSocket.getInputStream()));
    } catch (Exception) { /* Exception 처리 */ }
}

public void sendCommand(String cmd)
{
    /* 대상서버로 제어 명령어를 전송한다. */
}

public String readOutput()
{
    /* 대상서버로부터 전송된 수행결과를 수신하고, 전송이 종료되면 자원 해제 */
    return out.Line;
}
}
    
```

[그림 5] 제어서버와 대상시스템과의 통신부분 알고리즘

```

public class RCCpuStatus {
.....
// 그래프용 벡터 생성
Vector usrVt = new Vector();
public RCCpuStatus(String cpuStr)
{
    StringTokenizer st = new StringTokenizer(cpuStr);
    // Tokenizing 하여 attribute 변수에 set
}

public void gatherData(String cpuStr)
{
    // 그래프 또는 데이터의 평균값을 구하기 위하여 String을 tokenizing 하여
    // data gathering
    .....
    // Tokenizing
    StringTokenizer st = new StringTokenizer(cpuStr);

    i_usr = Integer.parseInt(st.nextToken());
    .....
    // 그래프 데이터 값을 벡터에 저장
    usrVt.add(new Integer(i_usr));
}

public void calcAvg()
{
    // 축적된 모니터링 데이터에 대한 평균값 계산
}
.....
}
    
```

[그림 6] 제어서버의 수행결과 포맷팅(Formatting)주요 알고리즘

제어명령의 수행결과는 기본적으로 대상시스템에서 stdout으로 출력되는 텍스트 스트링이 대부분이다. 따라서 이 수행결과를 모바일 장비에 맞는 형태로 포맷팅하기 위해서는 대부분의 경우 [그림 7]의 알고리즘에서와 같이 스트링에 대한 구현(implementation)으로 구성된다.

```
// 사용자 RunTime 에서 수행을 위한 Attribute 선언
private Runtime sysRunTime = Runtime.getRuntime();
private Process sysProc = null;
private InputStream is;
private InputStream errorIs;
private InputStreamReader isr;
private InputStreamReader errorIsr;
private BufferedReader br;
private BufferedReader errorBr;

// 제어명령 수행 및 stdout 정상 수행 출력결과 Setting
sysProc = sysRunTime.exec(command);
is = sysProc.getInputStream();
isr = new InputStreamReader(is);
br = new BufferedReader(isr);
// stdout 수행 error 출력결과 Setting
errorIs = sysProc.getErrorStream();
errorIsr = new InputStreamReader(errorIs);
errorBr = new BufferedReader(errorIsr);

String outputLine = null;

while( (outputLine=br.readLine()) != null){
    // 제어서버로 정상 수행 출력결과 전송
}
while( (outputLine=errorBr.readLine()) != null){
    // 제어서버로 수행 error 출력결과 전송
}
```

[그림 7] 원격지 에이전트의 주요 알고리즘

[그림 8]은 제어명령 수행 및 수행결과 처리를 위한 원격지 에이전트의 주요 알고리즘이다. 제어명령 수행 후 stdout 으로 출력되는 출력결과를 제어서버에 전송해 주는 알고리즘이다. stdout으로 출력되는 제어명령은 동기 명령(Sync command)으로 분류하며, 비동기 명령(Async command)은 (예, daemon 수행과 같은 경우) 제어명령 수행 후, 수행이 완료되었다는 메시지만을 전송한다. 제어명령은 자바의 런타임 객체를 이용하여 시스템의 런타임 환경에서 명령을 수행하도록 한다.

제어명령을 처리하는 방식에는 스크립트 파일을 생성하여 수행하는 방식이 있으며, 스크립트 파일 생성 알고리즘은 [그림 9]와 같다. 스크립트 파일 생성 시 주의할 점은 대상시스템에 로그인한 사용자의 환경과 권한으로 생성시켜주기 위하여 사용자 환경변수 설정, 임시 스크립트 파일의 권한 및 퍼미션(Permission) 설정을 해주어야 하며, 수행 시 해당 사용자의 권한으

로 수행시켜야 한다[10]. 이러한 처리를 해주어야만, reboot과 같은 명령의 수행을 일반 사용자가 수행 할 수 없게 된다.

```
public String makeScript(String id, String command)
{
    // 1. 시스템의 password 파일의 정보를 이용하여 다음의 정보를 추출한다.
    // UID, GID, HOME Directory, 사용하는 SHELL
    // 2. 임시로 사용할 Temporary script File을 생성한다.
    sFile = File.createTempFile("rcs", ".sh");
    String shellFile = tmpDir + sFile.getName();
    fw = new FileWriter(shellFile);
    // 3. Temporary script File에 수행할 script 를 write 한다.
    // 3-1. 사용자 Runtime Shell 환경을 추출하여 환경변수 Setting
    // 3-2. 수행할 command 를 Setting
    // 4. 사용자의 UID, GID 정보를 추출하여 script 파일의 권한 및 Permission 변경
}
```

[그림 8] 원격지 에이전트의 스크립트생성을 위한 주요 알고리즘

IV. 결론

본 연구는 모바일 단말기로 이동 중에도 원격지의 시스템을 실시간으로 모니터링 할 뿐만 아니라 문제가 발생되면 즉각 모바일 단말기로 장애통보를 해주거나, 원격지의 시스템에 바로 접근해서 문제를 해결 할 수 있는 원격제어 기능을 제공할 수 있는 실시간 무선 원격제어 시스템을 설계 및 구현하였다. 그리고 모바일 장비 사용자가 원격지 서버에 접속하기 위하여 강력한 인증과정을 수행하도록 인증기능을 실행하고, 원격지 서버로부터 인증된 사용자라도 대상 서버에 접속하여 명령어를 원격에서 입력하려면 명령어 사용에 대한 권한을 부여 받도록 하였다.

참고문헌

- [1] Carles Arehart, Nirmal Chidambaram, etc. "Professional WAP" Wrox. 2000.
- [2] Douglas Comer and David Stevens, "Internetworking with TCP/IP Vols I, II and III", Prentice-Hall. 1991.
- [3] Dr. Mikael Sjodin 2002 "Remote Monitoring and Control Using Mobile Phone"
- [4] Ericsson Enterprise AB 2002 EN/LZT 102 3511 RC "The path to the Mobile Enterprise"
- [5] James F. Kurose and Keith W. Ross, "Computer Networking", Addison Wesley 2nd-Edition, 2002.
- [6] Kaveh Pahlavan and Prashant Krishnamurthy, "Principles of Wireless Networks", Prentice-Hall.