

논문 2021-1-9 <http://dx.doi.org/10.29056/jsav.2021.06.09>

리눅스 기반 프로그래밍 언어의 온라인 학습 시스템 구성에 관한 연구

전호익*, 이현창**†

A study on the On-line Teaching system for Linux-based Programming Language

Ho-Ik Jun*, Hyun-Chang Lee**†

요 약

본 논문에서는 리눅스 기반 프로그래밍 언어를 온라인으로 실습할 수 있는 시스템의 구성 방법을 제시하였다. 제시한 시스템은 리눅스 운영체제의 가장 큰 특징인 웹-서버 기능을 활용하며, 방화벽이나 기타 보안적 제약을 받지 않으면서 telnet과 FTP 기능을 모사하여 실제 리눅스 콘솔과 유사한 실습이 가능하다. 이를 위해 웹 상에서 프로그래밍 도구가 가져야 할 기능적 요소들을 분석하고 이를 구현하기 위한 알고리즘을 정립하였으며, 특히 사용자의 실수에 의한 오류 메시지 등이 실제 telnet 화면과 동일한 형태로 나타날 수 있는 방법을 구현하였다. 구현된 학습 시스템을 학생들을 대상으로 수업에 활용한 결과 온라인으로 리눅스 프로그래밍 언어의 실습이 가능함은 물론 교수자가 학습자들의 모든 사항들을 직접 점검하고 지도할 수 있어 학습자의 만족도가 오프라인 수업 때와 유사한 결과를 확인하였다.

Abstract

In this paper, a system configuration method that can practice Linux-based programming language online is presented. The proposed system utilizes the web-server function, which is the biggest feature of the Linux operating system, and simulates the telnet and FTP functions without firewalls or other security restrictions, so that it is possible to practice similar to the actual Linux console. To do this, we analyzed the functional elements that a programming tool should have on the web and established an algorithm to implement it. In particular, a method was implemented in which an error message caused by a user's mistake can appear in the same form as the actual telnet screen. As a result of using the implemented learning system in the class for students, it is possible to practice the Linux programming language online, as well as the instructor can directly check and guide all the learners, so the learner's satisfaction is similar to that of the offline class was confirmed.

한글키워드 : 리눅스, 웹 서버, 온라인 실습, 프로그래밍, 학습 도구

keywords : Linux, Web-server, On-line experiment, programming, teaching tool

* 해전대학교 전기과

** 공주대학교 정보통신공학부

† 교신저자: 이현창(email: hcleee@kongju.ac.kr)

접수일자: 2021.06.02. 심사완료: 2021.06.19.

게재확정: 2021.06.20.

1. 서 론

예전에는 사이버 교육과 같은 원격교육, 온라

인 교육의 수요가 일부 있어왔으나, 최근의 계속된 코로나 상황에 의해 온라인 교육의 수요와 시장 규모는 물론 그 중요성 또한 매우 커졌다. 과거에 실시되던 온라인 교육은 이론적인 내용이나 인문학 등과 같이 실습이 없는 분야를 주요 대상으로 진행했으나, 최근 코로나의 비대면 상황에 의해 실습과목들의 온라인화 문제가 크게 대두되었다. 이 중 소프트웨어 분야의 실습은 대부분 교수자와 학습자 사이에 동일한 소프트웨어의 설치에 의해 원격리에서도 실습이 가능하지만, 이 경우 교수자가 학습자의 소스를 확인하기 위해, 또는 학습자의 소스에 오류가 발생한 경우 별도의 수단을 이용해 교수자에게 전달하고, 교수자는 이를 분석해 다시 학습자에게 전달해야 하며, 많은 학과목과 수강생을 담당하는 교육자에게는 과도한 업무 부하로 인해 현실적으로 실현 불가능한 방법이 될 수밖에 없다. 학습자가 환경을 갖추기 어려운 실습, 예를 들어 리눅스 기반 서버 프로그래밍 등과 같은 경우에는 모든 학습자가 동일한 환경을 갖추기 어려우므로 이에 관한 여러 가지 연구가 이루어졌다.

Kang[1] 등은 온라인 교육 시 학생들이 학습에 참여하는데 영향을 미치는 다양한 요소들에 대해 분석하였고, Choi[2] 등은 컴퓨터 프로그래밍 언어실습을 위한 가상학습 콘텐츠 개발을 연구하고 이를 온라인 학습시스템으로 응용하는 연구를 하였으며, Jun[3] 등은 리눅스 기반 서버 시스템의 특징을 이용해 온라인 원격실습을 진행할 때의 문제점 및 갖춰야 할 조건들을 분석하였다.

본 논문에서는 이러한 다양한 연구와 분석들을 바탕으로 리눅스 기반 프로그래밍 실습을 원격으로 진행함은 물론 교수자가 학습자의 소스를 쉽게 접근해 관리와 확인이 가능한 시스템을 제시하고 이를 구성해 효과를 입증하고자 한다.

2. 온라인 학습 시스템의 제약점

2.1 학습자의 리눅스 시스템

온라인을 이용해 실습이 진행되려면 교수자와 학습자 사이에 동일한 환경이 구성되어야 한다. 그러나 일반적으로 학습자들이 사용하는 퍼스널 컴퓨터의 경우 대부분 윈도 운영체제를 이용하므로 비록 교수자가 리눅스 서버를 구비하고 있다 할지라도 학습자는 이러한 환경이 구성되지 못함에 따라 단순히 교수자가 조작하고 그 결과를 보여주는 수준의 학습만 이루어질 수밖에 없다.

이에 대한 방안으로서 학습자의 컴퓨터에 Docker나 가상 머신(Virtual Machine) 소프트웨어를 설치해 활용할 수 있으나[4,5], 실제 학습에 적용한 경우 학습자 컴퓨터의 과부하 문제는 물론 학습자는 리눅스 시스템에 관한 학습 노력에 비해 가상 머신을 학습하고 운영하는 것이 더 어려운 것이 현실이다. 특히 이러한 학습의 경우 학습자가 자신의 PC에서 실습하는 것이므로 교수자가 직접 이를 확인해 지도할 수 없으며, 어떤 오류가 발생한 경우 학습자가 메일이나 기타 수단을 이용해 교수자에게 소스를 전달하고 학습자는 이에 대한 답변을 메일이나 기타 수단을 이용해 전달해야 하므로 수많은 학과목 및 학습자를 상대하는 교수자 입장에서는 그 번잡성에 의해 적용이 거의 불가능하다.

2.2 FTP, Telnet의 사용 불가

교수자가 학습자의 소스를 관리하고 확인할 수 있도록 리눅스 기반 서버 시스템 프로그래밍 시스템을 구성하는 경우 학습자의 소스 파일 등을 교환할 FTP와 이를 컴파일 하기 위한 Telnet의 사용이 필수적이다[6]. 그러나 서버가 설치된 학교와 같은 공간은 http 포트 이외의 다른 포트가 방화벽으로 차단되고, 보안 처리된 FTP, Telnet을 이용하는 경우에도 포트의 개방 문제를 비롯해 보안 키 관리 등 부수적인 복잡한 문제점이 발생해 전문적인 지식을 갖추지 못한 경우 매

우 큰 위험에 노출될 수 있어 적용이 어렵다.

3. 온라인 리눅스 학습 시스템의 구성

이상에서 고찰한 온라인 학습 시스템의 각종 제약점들을 극복하기 위해 다음과 같은 요건을 갖춘 온라인 학습 시스템의 구성이 필요하다.

첫째, 교수자가 학습자의 소스와 문제점을 쉽게 확인할 수 있도록 서버에 모든 학습자 소스를 집중시키고, 이 서버에서 컴파일과 실행을 진행한다. 둘째, 학습자가 방화벽 등의 보안문제 없이 리눅스 서버로 접속하기 위해 http 포트를 활용한다. 셋째, http 포트를 경유한 홈-페이지 형태의 학습화면을 활용함에 있어서 FTP, telnet과 유사한 환경과 기능을 수행한다. 넷째, 각 학습자가 자신의 고유한 ID와 패스워드를 이용해 진입하고 이 때 각 학습자의 고유 영역이 존재하며, 이 영역을 대상으로 제어한다. 다섯째, telnet을 이용한 제어 시 실제 리눅스 시스템의 동작과 반응 상태를 학습자에게 전달한다. 여섯째, 학습자가 불순한 의도로, 또는 시스템을 파손할 우려가 있는 telnet 조작을 감지해 이를 미연에 방지한다. 일곱째, 학습자와 교수자가 동일 시간에 서버에 존재하지 않더라도 상호 의사교환을 진행할 수 있는 간이 게시판이 각 사용자마다 존재한다.

이에 따라 각 사용자마다 보존되어야 할 데이터는 표 1과 같이 정리될 수 있다.

3.1 학습자의 시스템 접속

실습용 서버의 경우 학습자의 자유로운 실습을 위해 보안이 해제상태로 운영되고, 로그-인 과정도 보안된 것을 사용할 경우 보안 키 획득과 관리가 복잡하므로 로그-인 화면으로 진입하는 특별 주소를 설정하고 이 주소를 학습자에게 알려준다면 일반인에게는 그림 1(a)와 같이 로그-

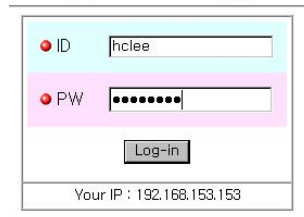
표 1. 사용자용 보존 데이터
Table 1. Retention data for users

구분	기능
연결 키	비 연결성 통신인 http를 이용해 지속적인 연결을 유지하기 위한 확인 데이터로서 통신 연결 시 query로 전달 및 확인.
학습자 작업 디렉토리 위치	학습자가 현재 작업을 진행하는 디렉토리의 위치. cd 명령 등에 의해 이동한 경우 이후 접속 시 이 위치를 지속한다.
화면 데이터	학습자에게 표시될 리눅스 터미널 화면 상태 보존 버퍼.
게시판 데이터	학습자와 교수자 사이에 주고받는 각종 메시지들을 보관.
History	학습자가 입력했던 커맨드 보관



(a) 초기 공개 화면 예

● CGI 실습 시스템 ●



(b) 비밀 접속 페이지 화면 예

그림 1. 접속 페이지의 예

Fig. 1. Example of an access page

인 폼 자체가 표시되지 않아 보안이 강화될 수 있다.

즉 로그-인 폼으로 진입하는 자체가 일종의 비밀번호 역활을 하며, 로그-인 시점에서도 그림 1(b)와 같이 다시 한번 비밀번호를 입력한다[7].

학습자의 접속사항은 각 학습자마다 작업 디렉토리와 작업 상황이 다르므로 학습자를 구분하기 위한 ID가 필요하며, 학습자가 타 학습자로 진입해 파일을 다룰 경우 타 학습자는 매우 큰 혼란에 빠질 수 있으므로 이를 방지하기 위해 간단한 비밀번호, 예를 들어 학습자의 학번 뒤 4자리 등을 입력하도록 구성한다.

3.2 FTP 기능의 구현

FTP 기능은 파일의 put과 get의 2가지 형태로 구분할 수 있다. 파일의 송신은 그림 2(a), (b)에 나타난 바와 같이 HTML의 FORM 태그를 활용해 파일을 서버로 전송하고, action으로 지정된 서버 프로그램은 이를 수신해 사용자 디렉토리에 파일로 저장한다.

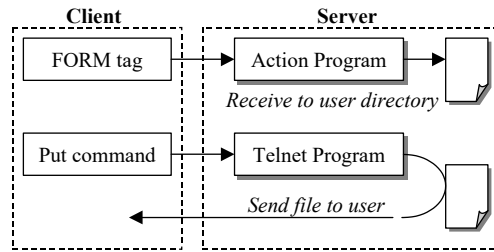
파일의 수신은 이후에 다룰 telnet 커맨드 입력창에 put 커맨드를 입력하면 telnet 명령 처리기에서 서버로부터 해당 파일을 전송해 그림 2(c)와 같이 사용자의 브라우저에서 직접 저장하도록 구성한다. 이상의 기능에 의해 필요로 하는 파일을 사용자와 서버 사이에서 주고받을 수 있다.

3.3 Telnet 기능의 구현

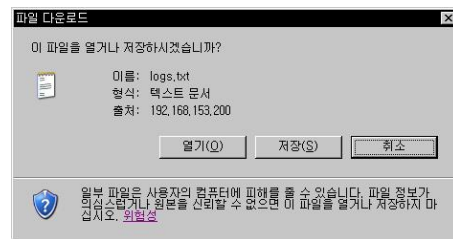
Telnet 기능은 커맨드를 입력할 수 있는 입력창과 콘솔 터미널의 출력을 그대로 보여줄 수 있는 표시 창의 형태로 그림 3과 같이 IFRAME 태그를 이용해 콘솔과 최대한 유사한 형태로 구성한다.



(a) Get form의 예



(b) 파일 전송 과정



(c) Put 커맨드 이후의 반응

그림 2. 파일 전송 구현 방법

Fig. 2. Method to implement file transfer



그림 3. Telnet 입력 및 출력 창

Fig. 3. Telnet input and output window

Telnet 기능은 커맨드 입력 시 그림 4에 나타난 바와 같은 플로차트로 처리한다.

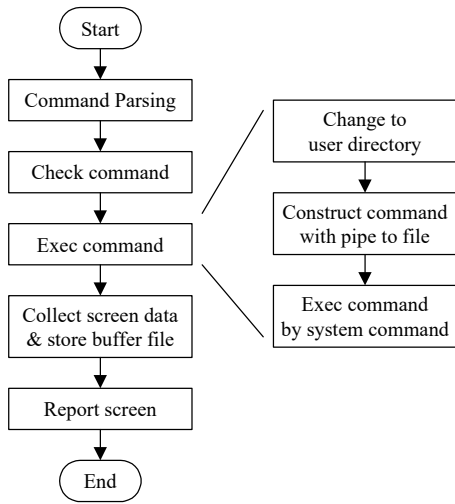


그림 4. Telnet 처리 플로차트
Fig. 4. Telnet processing flow chart

그림에서 사용자가 입력한 커맨드 내용을 분석해 커맨드와 파라미터들을 분리하고, 커맨드 종류를 분석해 시스템에 지장을 줄 가능성이 있는 명령들은 권한이 없는 것으로 처리하고, 그 외에는 이를 실행하도록 허가한다. 이후 명령 실행 시 cd 명령인 경우는 사용자의 작업 디렉토리 위치에서 벗어나지 않는 범위에서 조정해 파일에 저장하며, 그 이외의 명령들은 시스템 shell에 의해 사용자의 디렉토리로 이동 후 사용자의 명령어를 조립하며, 이 때 그 결과를 pipe를 통해 [8] 화면 버퍼로 향하도록 구성한다. 이와 같이 구성된 명령어 구문을 시스템 shell에 의해 http 데몬의 권한으로 실행하고, 이때 화면 버퍼에 저장된 내용들을 사용자 화면 데이터 버퍼 파일에 정리해 저장한다.

사용자 화면 데이터를 보존하는 데이터 버퍼 파일은 그림 5와 같이 콘솔 크기와 동일한 형태로 80컬럼, 24라인의 레코드로 구성하며, 현재 커맨드 라인 위치를 지시하는 포인터를 이용해 전체적으로 환형 큐(circular queue)를 형성한다. 이와 같이 구성된 버퍼에서 현재 입력된 커맨드는

커맨드 라인 포인터가 지시한 위치에 저장하고, 이후 포인터를 증가하면서 pipe에 의해 캡처된 화면 데이터를 순차적으로 저장한다.

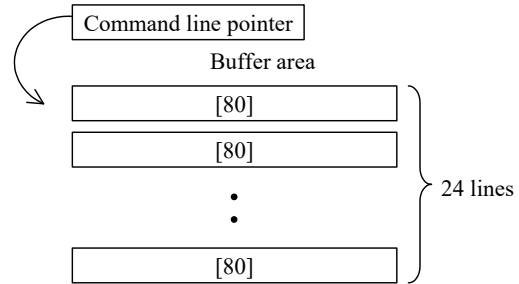


그림 5. 화면 데이터 버퍼의 구성
Fig. 5. Configuration of the screen data buffer

모든 데이터가 저장되면 현재의 라인 포인터 다음 열부터 콘솔 화면의 표시 첫 줄로서 전체를 표시하며, 이 경우 그림 6과 같이 커맨드 실행 전의 내용들과 함께 현재 실행된 결과화면이 저장되며 발생한다.

특히 이와 같은 pipe에 의한 화면 전달 방식은 리눅스 시스템에서 발생하는 오류가 그림 6과 같이 그대로 표시됨으로 인해, 발생한 오류를 학습자가 체험하고 수정하는 학습이 가능하다.

```

[hclee@ok hclee]$
[hclee@ok hclee]$
[hclee@ok hclee]$gcc -o env.cgi enf.c
gcc: enf.c: No such file or directory
gcc: no input files
[hclee@ok hclee]$
  
```

그림 6. 화면 표시의 예
Fig. 6. An example of a screen display

또한 실제 리눅스 시스템에서 자주 이용되는 history 기능은 그림 7에 나타난 바와 같이 커맨드 라인 하단에 선택 박스의 형태로 표시하여 이전에 입력했던 명령들을 반복해 사용할 수 있다.

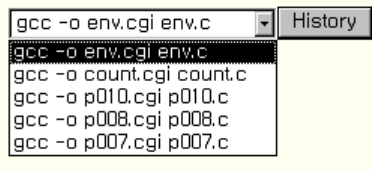


그림 7. History 기능의 구현 예
 Fig. 7. An example implementation of the history function

4. 결론

본 논문에서는 리눅스 기반 프로그래밍 언어를 온라인으로 실습할 수 있는 시스템의 구성 방법을 제시하였다. 제시한 시스템은 리눅스 운영체제의 가장 큰 특징인 웹-서버 기능을 활용하여 방화벽이나 기타 보안적 문제를 해결하며 웹 상에서 telnet과 FTP 기능을 모사하도록 구현하였다. 이를 위해 웹 상에서 프로그래밍 도구가 가져야 할 기능적 요소들을 분석하고 이를 구현하기 위한 알고리즘을 정립하였다. 이상에서 분석된 기능들을 리눅스 서버에 구축하여 학생들을 대상으로 수업에 활용한 결과 원격에서 온라인으로 리눅스 프로그래밍 언어의 실습이 가능함은 물론 교수가 학습자의 소스와 실행결과에 자유롭게 접근할 수 있어 수시로 점검과 문제해결이 가능함을 확인하였고, 그 결과 대면수업이었던 2019년도 강의평가 90.44점에 비해 본 방법을 적용한 2020년도에는 92.43점으로 학습자의 만족도가 오프라인 수업과 유사한 효과를 확인하였다.

본 연구는 2020년도 해전대학교 교내 학술연구
 비지원에 의해 연구되었음

참고 문헌

[1] Jae-shin Kang, Tae Yim Kang, “Analysis

of Content Components that Influence Learning Participation in Online Lectures for Multimedia Practice Subjects”, The Korea Society of Design Culture, Vol.26, No.4, pp.19-32, Dec. 2020.
 DOI : 10.18208/ksdc.2020.26.4.19

[2] Kwan-sun Choi, Heung-goo Jun, “A Study on the Development of Virtual Education Contents for Computer Programming Laboratory”, Korean Institute of Information Technology, Vol.2, No.2, pp.61-68, Sep. 2004.
 UCI : I410-ECN-0101-2009-566-016509357

[3] Ho-ik Jun, Hyun-chang Lee, Kyu-Tae Lee, Seo-Ik Kang, Won-Shik Na, “Ayalysis of Online Teaching Tool Configuration Requirements for Linux-based Software Programming Language”, 32th Conference of Korea Software Assessment and Valuation Society, pp.61-62, July 15, 2020.

[4] Min-ho Lee, Young-ik Eom, “Analysis of Copy-on-Write Overhead on Virtual Machine Disk”, Conference of the Korean Institute of Information Scientists and Engineers, pp.1454-1455, Dec. 2018.

[5] Yu-jin Jang, Tae-hyung Lee, Young-Ik Eom, “Performance Analysis based on CPU Limit of Virtual Machine”, Conference of the Korean Institute of Information Scientists and Engineers, pp.2089-2090, Dec. 2018.

[6] Kebi 캠퍼스, 웹구축 - Linux vs Windows2000, 이한디지털리, July 2001.
 ISBN : 89-89345-29-4

[7] 이현창, 이종언, CGI로 C-언어 정복, 상학당, Feb. 2012. ISBN : 978-89-85437-81-3

[8] Sun-ja Kim, Ik-soon Kim, Oh-seok Kwon, “User-level Pipes for Multiprocessors”, The Korean Institute of Information Scientists and Engineers, Vol. 19, No. 3, pp.128-134, Mar. 2013. ISSN : 2383-6296

저 자 소 개



전 호익(Ho-ik Jun)

1984.2 단국대 전자공학과 공학사
1986.8 단국대 전자공학과 공학석사
1997.8 단국대 전자공학과 공학박사
1992.10 - 현재 해전대학교 전기과 부교수
<주관심분야> 멀티미디어 회로, 전동기제어, 마이크로프로세서



이 현창(Hyun-Chang Lee)

1986.2 단국대 전자공학과 공학사
1989.8 단국대 전자공학과 공학석사
1996.2 단국대 전자공학과 공학박사
1996.3-2004 국립 천안공업대학
정보통신과 부교수
2005.3-현재 국립 공주대학교 공과대학
정보통신공학부 교수
<주관심분야> 멀티미디어 회로, 전동기제어, 마이크로프로세서, 임베디드소프트웨어