

내장형 시스템용 웹 서버 개발

강선례*, 신동하**

*상명대학교 일반대학원 컴퓨터학과

**상명대학교 소프트웨어학부

e-mail : srkang604@smu.ac.kr

Development of A Web Server For Embedded Systems

Sunrye Kang*, Dongha Shin**

*Dept. of Computer Science, Sangmyung University

** Division of Computer Software, Sangmyung University

요 약

내장형 시스템용 웹 서버는 일반 웹 서버와는 매우 다른 요구 사항을 가지고 있다. 예를 들어 작은 메모리의 사용, 특수 목적으로 정의된 정적 및 동적 페이지의 제공, 웹 서버 내부에 기본 시스템 및 네트워크 관리 기능의 내장, 내장형 응용 프로그램과 웹 서버와의 유연한 연결 그리고 다양한 내장형 프로세서의 지원 등의 요구 사항이 있다. 본 논문은 이러한 내장형 시스템용 웹 서버의 요구 사항을 기술하고, 이들 요구 사항을 바탕으로 우리 고유의 내장형 웹 서버인 "웹박스(Webbox)"를 설계하고 내장형 리눅스 개발 환경에서 이를 구현하는 방법을 기술한다. 현재 내장형 웹 서버 웹박스의 알파 버전이 개발되어 시험 중이며 최종 결과물이 완성되면 공개 소프트웨어로 오픈할 예정이다.

1. 서론

내장형 시스템용 웹 서버는 일반 웹 서버와는 매우 다른 요구 사항을 가지고 있다. 예를 들어 작은 메모리의 사용, 특수 목적으로 정의된 정적 및 동적 페이지의 제공, 웹 서버 내부에 기본 시스템 및 네트워크 관리 기능의 내장, 내장형 응용 프로그램과 웹 서버와의 유연한 연결 그리고 다양한 내장형 프로세서의 지원 등의 요구 사항이 있다. 본 논문은 내장형 시스템용 웹 서버 개발을 위해 필요한 요구 사항을 기술하고, 이들 요구 사항을 바탕으로 우리가 개발한 내장형 시스템용 서버인 "웹박스(Webbox)"의 구조, 설계 그리고 구현에 대하여 기술한다.

본 논문의 2 절에서는 내장형 시스템에 적합한 웹 서버의 요구 사항을 기술한다. 3 절에서는 일반적인 웹 기반 내장형 시스템 구조와 우리가 개발한 내장형 시스템용 웹 서버인 웹박스를 탑재한 내장형 시스템 구조를 설명한다. 4 절에서는 2 절에서 기술한 내장형 웹 서버의 요구 사항을 바탕으로 웹박스의 설계에 대해서 기술한다. 5 절에서는 4 절에서 기술한 웹박스의 설계부분을 C 언어를 사용하여 구현한 것을

기술한다. 마지막으로 6 절에서는 논문의 결론 및 앞으로 연구 계획을 기술한다.

2. 내장형 웹 서버의 요구 사항

내장형 시스템용 웹 서버는 전형적인 일반 웹 서버와는 다른 요구 사항을 가지고 있다. 본 절에서는 작은 메모리 사용, 특수 목적으로 정의된 정적 및 동적 페이지의 제공, 웹 서버 내부에 기본 시스템 및 네트워크 관리 기능의 내장, 내장형 응용 프로그램과 웹 서버와의 유연한 연결 그리고 다양한 내장형 프로세서의 지원 등의 요구 사항에 대해 기술한다.

2.1 작은 메모리 사용

내장형 시스템은 일반 시스템에서 보다 매우 작은 용량의 주 메모리를 사용하고, 디스크 저장 장치 대신 작은 용량의 플래시 메모리를 사용한다 [1][2][4].

2.2 웹 페이지 제공

내장형 시스템용 웹 서버는 정적, 동적 콘텐츠를 웹을 통해 제공한다. 특수목적으로 정의된 내장형 시스템의 정적 웹 페이지는 한번 만들어지면 수정되거나 추가되지 않는다. 내장형 시스템용 웹 서버는 일반적인 웹 서버에 비해서 정적 웹 페이지 보다는 동적 웹 페이지를 더 많이 서비스 한다[4].

2.3 시스템 및 네트워크 관리 기능

일반 시스템은 시스템 관리자가 시스템에 연결된 콘솔을 통하여 시스템의 이상 유무를 판단하여 적절한 조치를 취할 수 있다. 하지만 대부분의 내장형 시스템은 콘솔이 시스템에 연결되어 있지 않은 경우가 대부분이며 시스템이 설치된 장소가 사람이 접근하기 쉬운 장소가 아니기 때문에 인터넷으로 시스템에 접근하여 시스템을 관리할 수 있는 기능을 제공하는 것은 매우 중요하다[4].

2.4 내장형 응용 프로그램과의 연결

대부분의 내장형 시스템은 내장형 응용 프로그램에 접근하기 위해서 별도의 소프트웨어를 설치하거나 개발하여야 한다. 이런 불편함을 줄이기 위해서 웹 서버가 웹 클라이언트를 통해 플랫폼 독립적이고 표준화된 GUI 를 사용자에게 제공하는 것 또한 중요하다[4][6].

2.5 다양한 프로세스 수용

다양한 종류의 내장형 시스템을 위한 웹 서버는 내장형 시스템에 쉽게 이식이 가능해야 한다. 이식성을 높이기 위해 내장형 웹 서버는 표준 C 언어의 범위에서 프로그래밍 하고 POSIX 표준 시스템 호출을 사용하여야 한다. 그리고 소켓 프로그래밍을 위하여 표준화된 인터페이스를 사용한다. 이를 만족하면 대부분의 내장형 프로세서에 내장형 웹 서버의 이식이 가능하다.

3. 구조

일반적인 웹 기반 내장형 시스템 구조와 우리가 개발한 내장형 시스템용 웹 서버인 웹박스를 탑재한 내장형 시스템 구조를 살펴보자.

시스템 및 네트워크 관리 라이브러리	웹 서버	내장형 응용 프로그램
리눅스 커널		
하드웨어		

그림 1 일반적인 웹 기반 내장형 시스템 구조

그림 1 은 일반적인 웹 기반 내장형 시스템 구조[5]이다. 기본적인 내장형 시스템 위에 서로 다른 응용 프로그램인 시스템 및 네트워크 관리 라이브러리, 웹 서버, 내장형 응용 프로그램을 서로 연결 시키거나 독자적으로 수행시킨다. 이러한 구조는 각각의 프로그램에 접근하기 위해 별도의 프로그램의 설치나 개발을 해야 하는 불편함이 있다.

그림 2 는 그림 1 의 구조의 불편함을 보안하는 것으로, 시스템 및 네트워크 관리 라이브러리, 웹 서버 그리고 내장형 응용 프로그램을 하나로 통합한 웹박스를 탑재한 내장형 시스템 구조[5]를 보여준다.

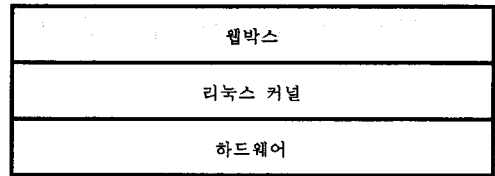


그림 2 웹박스를 탑재한 내장형 시스템 구조

4. 설계

웹박스의 주요 구성 요소는 웹 컴파일러[4][6], HTTP 엔진 그리고 시스템 및 네트워크 관리 라이브러리 이다.

그림 3 은 우리가 개발한 내장형 시스템용 웹 서버인 웹박스의 생성절차[4]를 보인다.

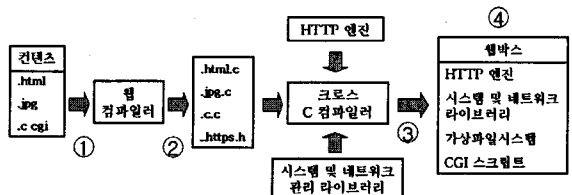


그림 3 웹박스의 생성 절차

- ① 정적, 동적 웹 페이지 .html, .jpg, .cgi 를 웹 컴파일러로 컴파일한다.
- ② 웹 컴파일러는 각각의 파일들에 대해서 .c 파일과 .x 파일을 생성하고, .x 파일들을 이용하여 다시 _https.h 를 생성한다.
- ③ 웹 컴파일러에서 생성된 파일들과 HTTP 엔진, 시스템 및 네트워크 관리 라이브러리를 크로스 C 컴파일한다.
- ④ HTTP 엔진, 가상 파일 시스템, CGI 스크립트, 시스템 및 네트워크 관리, 내장형 응용을 포함한 웹박스가 생성된다.

4.1 웹 컴파일러

웹 컴파일러는 .html, .txt, .jpg, .gif 과 같은 정적 웹 콘텐츠 파일과 동적 콘텐츠 파일인 CGI 스크립트를 읽어서 C 프로그램 파일을 생성한다.

이렇게 생성된 C 프로그램 파일들은 가상 파일 시스템[4][6]에 저장된다. .html 과 .txt 같은 정적 콘텐츠는 가상 파일 시스템의 크기를 줄이기 위해 압축된 상태로 저장된다. CGI 와 같은 동적 콘텐츠는 웹박스의 수행 파일 안에서 함수로 수행될 수 있도록 원본 파일이 약간 수정된다.

4.2 HTTP 엔진

HTTP 엔진은 HTTP/1.0[9]의 일부를 구현한 프로그램으로 클라이언트가 웹 콘텐츠를 요청할 때, 그 콘텐츠를 찾아 전달하는 기능을 한다. 만약 정적 콘텐츠이면 HTTP 엔진은 가상 파일 시스템에서 그 파일을 찾는다. 가상 파일 시스템에 압축된 상태로 저장되어 있는 정적 콘텐츠 파일들은 압축이 풀린 상태로 클라이언트에게 전달된다. CGI 와 같은 동적 페이지에 대한 요청이 들어오면 HTTP 엔진은 그 동적 파일에 상응하는 C 함수를 호출하게 된다. 내장형 웹 서버는 특성상 클라이언트로부터 전송되는 정보의 양이 비교적 작아서 HTTP GET method 를 사용하는 것이 메모리 관리에 효율적이다[3].

4.3 시스템 및 네트워크 관리 라이브러리

동적 웹 콘텐츠와 같은 형식으로 구현된 시스템 및 네트워크 관리 라이브러리는 클라이언트에게 내장형 시스템의 관리를 웹상에서 가능하게 한다. 현재 개발한 웹박스 알파버전은 웹박스 안에 기본 시스템 및 네트워크 관리에 필요한 라이브러리를 함수로 구현한다.

5. 구현

현재 내장형 웹 서버는 Intel x86 RedHat 리눅스 8.0 에서 개발하고 32bits Intel SA-1110(206MHz) CPU, 32 SDRAM, 16MB Flash memory 를 사용하는 내장형 시스템 개발 도구인 Hyper104 에서 수행 시험 중이다.

5.1 웹 컴파일러

웹 컴파일러는 .html, .txt, .jpg, .gif, .c 파일을 읽고 .c 파일과 파일 정보를 가진 .x 파일을 생성하고, 읽은 파일들의 특성과 위치 정보를 가진 _https.h 라는 가상 파일 시스템 테이블을 만든다. shell 로 작성된 웹 컴파일러는 .c 와 .x 파일을 만드는 wc-a2cx.c 와 .x 파일을 이용하여 가상 파일 시스템 테이블을 만드는 wc-x2h.c 를 차례로 호출한다. 그림 4 는 wc-a2cx.c 가 호출되어 .c, .x 파일이 생성되는 과정을 보이고 있다. index.html 은 정적 파일이고 shell.c 는 동적파일이다.

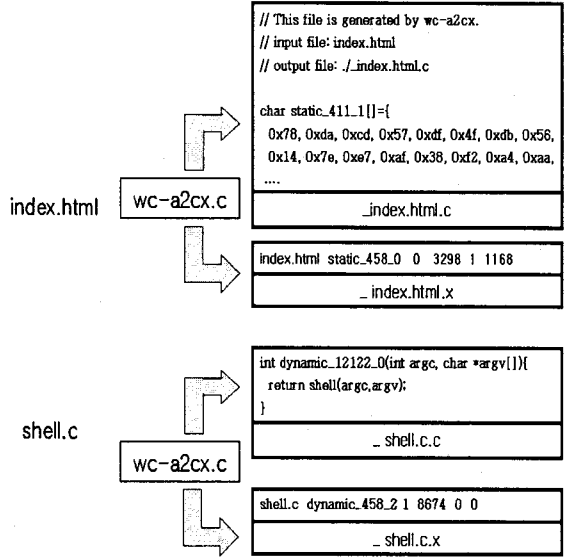


그림 4 wc-a2cx.c 에 의해 생성된 .c, .x 파일

그림 5 에서 wc-x2h.c 는 wc-a2cx.c 에 의해 생성된 .x 파일을 읽어서 가상 파일 시스템 테이블인 file_array[]를 가지고 있는 _https.h 를 보여준다.

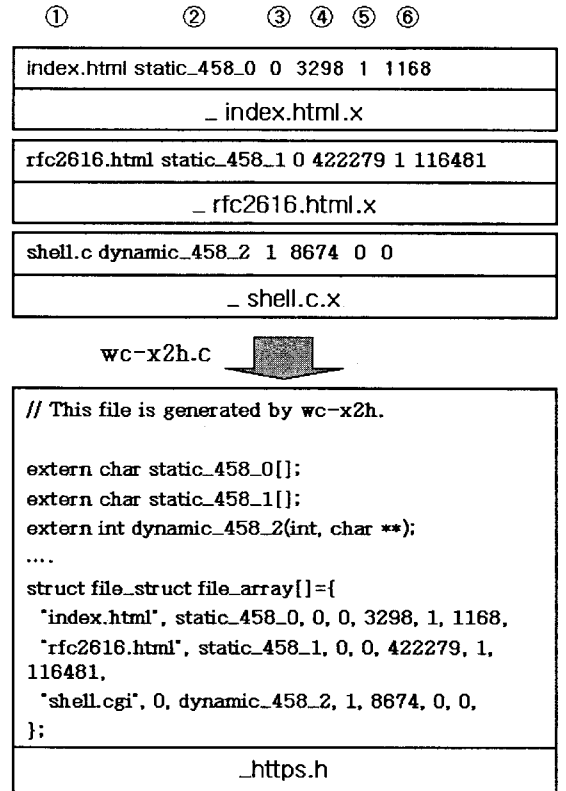


그림 5 wc-x2h.c 에 의해 생성된 _https.h

- ① 파일의 경로명
- ② 변수이름
- ③ 정적 동적 파일 구분(정적=0, 동적=1)
- ④ 파일의 크기
- ⑤ 압축유무(압축=1, 압축이 아닌 경우=0)
- ⑥ 압축후 파일의 크기

5.2 HTTP 엔진

HTTP 엔진은 웹 서버의 주된 프로그램으로 클라이언트에게 자원을 제공해주는 역할을 한다. HTTP 엔진에는 중요한 함수 2개가 있다. 첫째 함수는 `int search_file(char*)`이고 두번째 함수는 `void process_file(char*, int)`이다.

`int search_file(char*)` 함수는 인수로 자원의 이름을 받아서 가상 파일 시스템 테이블인 `file_array[]` 인덱스를 찾아 그 인덱스를 `void process_file(char*, int)`에 두번째 인수로 준다. `void process_file(char*, int)`이 첫번째 인수로 자원의 이름을 받고 두번째 인수로 `file_array[]` 인덱스를 받아서 클라이언트에게 자원을 제공한다.

5.3 시스템 및 네트워크 관리 라이브러리

C 함수로 구현된 시스템 및 네트워크 관리 라이브러리는 내장형 웹 서버의 프로세스로 수행되는 것이 아니라 함수로 구현된다. 웹박스에 포함된 시스템 및 네트워크 관리 라이브러리로는 `ls`, `cd`, `cat`, `cp`, `mv`, `cd`, `date`, `rm`, `df`, `du`, `reboot` 등을 구현하였고, `ifconfig`, `route` 등이 구현중이다.

6. 결론

우리는 본 논문에서 내장형 시스템용 웹 서버의 요구 사항을 기술하고, 이를 바탕으로 내장형 시스템용 웹 서버를 설계하고 구현하였다. 현재 일반적인 웹 기반 내장형 시스템에서 웹 서비스를 제공하기 위해 발생하는 불편함을 보완하여 웹박스를 개발하였다.

이런 불편함을 보완하여 개발한 웹박스는 몇 가지 장점을 가진다. 첫째, 웹박스 내에는 가상 파일 시스템이 있다. 따라서 웹 콘텐츠를 제공하는 파일들의 정보를 제공하기 위한 별도의 파일 시스템을 제공할 필요가 없어진다. 둘째, 웹박스는 웹 서버인 HTTP 엔진 크기와 시스템 및 네트워크 관리 라이브러리의 크기를 합하면 약 27k 이다. 이것은 작은 메모리를 사용해야 하는 내장형 시스템의 요구 사항에 충족하는 큰 장점이 된다. 셋째, 웹박스를 내장형 시스템에 탑재함으로써 웹 서버 기능과 시스템 및 네트워크 관리 기능이 가능해진다. 시스템 및 네트워크 관리 기능으로 관리자는 웹 기반에서 내장형 시스템의 정보를 제공 받거나 내장형 시스템 자체 관리가 가능해진다.

넷째, 내장형 응용을 관리하기 위한 별도의 프로그램이 없이 인터넷으로 연결된 웹 브라우저를 통해 통합 GUI 를 제공한다. 이것은 웹 브라우저만 있으면 내장형 시스템에서 제공하는 웹 콘텐츠를 제공 받을 수 있으며, 내장형 시스템의 관리까지 전반적인 기능을 할 수 있다는 것이다.

앞으로의 연구에서 우리는 웹박스의 기능을 좀 더 보완하고, 내장형 시스템용 웹 서버 평가를 위한 다양한 방법으로 내장형 시스템용 웹 서버인 웹박스를 평가할 것이다. 최종 결과물이 완성되면 공개 소프트웨어로 오픈할 예정이다.

참고문헌

- [1] Ian Agranat, Embedded Web Servers in Network Devices, *Communication Systems Design*, 30/36, 1998.
- [2] Ian Agranat, Engineering Web Technologies for Embedded Applications, *IEEE Internet Computing*, Vol. 2, No. 3, 40-45, 1998.
- [3] Manfred Bathelt, Ulrich Gall, Bernd Hindle and Christian Kurzke, Accessing Embedded System via WWW: the ProWeb Toolset, *Computer Networks and ISDN Systems*, 29, 1065-1073, 1997.
- [4] 신동하, 내장형 시스템용 웹 서버, *자연과학연구*, 9/1, 상명대학교, 2002.12.
- [5] Woochul Lee and Dongha Shin, Webbox for Embedded Linux Systems, *The 2003 International Conference on Embedded Systems and Applications*, 2003. 6.
- [6] Hong-Taek Ju, Mi-Joung Choi and James W. Hong, An Efficient and Lightweight Embedded Web Server for Web-based Network Element Management, *International Journal of Network Management*, 2000:10, 261- 275, 2000.
- [7] 오민정, 임성락, 기기 제어를 위한 저가의 초소형 임베디드 웹 서버, *정보처리학회논문지 A*, 제 9-A 권 제 1 호, 2002.3.
- [8] 김채규, 김홍남, 임채덕, 유비쿼터스 시대를 향한 임베디드 소프트웨어 발전 방향 및 개발 전략, *Telecommunications Review*, Vol.13, No.1, 2003.
- [9] W3C, RFC-1945: Hypertext Transfer Protocol HTTP/1.0, HTTP Working Group, <http://www.w3.org/Protocols/rfc1945/rfc1945>, 1996.