

논문 2019-1-11 <http://dx.doi.org/10.29056/jsav.2019.06.11>

# 리눅스 기반 서버의 소프트-RAID 시스템용 원격 상태 표시 소프트웨어의 구성에 관한 연구

나원식\*, 이현창\*\*†

## A study on the Construction of Remote Status Display Software for Soft-RAID system of Linux based Server

Won-Shik Na\*, Hyun-Chang Lee\*\*†

### 요 약

본 논문에서는 Linux OS로 구성된 서버에서 사용되는 소프트-RAID 시스템에서, 저장 장치에서 발견된 오류들을 원격에서 직관적으로 확인할 수 있는 방법을 제시하였다. 이를 위해 리눅스 OS의 소프트-RAID 방식에서 오류 보고 방법의 원리 및 문제점을 분석하고, 저장장치들의 상태를 인터넷 홈페이지를 통해 원격으로 확인할 수 있는 방법을 제시하였다. 제시한 방법은 인터넷 홈페이지에 그림을 표시하는 방법으로 구성되어 홈페이지 작성 시 자유로운 배치가 가능하고, 표시할 그림 데이터는 외부 파일의 형태로 구성되므로 그림의 편집 및 교체가 매우 편리한 장점을 가진다. 제시한 방법의 효과를 확인하기 위해 리눅스 서버 시스템에서 소프트웨어를 구성해 실험한 결과, 홈페이지 구성에 큰 추가사항 없이 원격에서 각 저장장치들의 상태를 직관적이면서 쉽게 점검할 수 있음을 확인하였다.

### Abstract

In this paper, we propose a method to remotely intuitively identify faults found in storage devices in soft-RAID used in a server system composed of Linux. To do this, we analyze the principle and problem of fault reporting method in the soft-RAID system of Linux OS and suggest the state of storage devices in remote Internet Home-page. The proposed method consists of a method of displaying images on the Internet home-page, so that it can be arranged freely when creating a home-page, and the image data is composed of external files, so it is very convenient to edit and replace images. In order to verify the effectiveness of the proposed method, we have confirmed that the state of each storage device can be confirmed at a long distance without any major addition to the Home-page configuration.

**한글키워드** : 소프트-레이드, 저장장치, 리눅스, 유닉스, 레이드-시스템

**keywords** : soft-raid, linux, unix, storage fault, raid system

\* 남서울대학교 컴퓨터소프트웨어학과

\*\* 공주대학교 정보통신공학부

† 교신저자: 이현창(email: hlee@kongju.ac.kr)

접수일자: 2019.06.02. 심사완료: 2019.06.17.

게재확정: 2019.06.20.

## 1. 서론

최근에는 컴퓨터 기술과 반도체 소자기술의 발전에 힘입어 매우 작은 규모임에도 고성능이

구현되는 다양한 형태의 컴퓨터들이 개발되었으며, 이에 따라 과거에는 개인이 운영할 수 없었던 서버 시스템이 초소형으로 가능해져 PC와 더불어 서버의 설치 및 운영이 급속히 증가하고 있다. 이로 인해 인터넷, IP-TV[1]와 같은 대형 서버를 중심으로 제공되던 서비스들이 IoT[2], 블록체인[3], 빅데이터[4] 등과 같은 서비스들로 그 영역이 크게 확대되고 있으며, 이러한 서버의 운영이 증가하고, 특히 저장할 데이터 용량이 크게 증가함에 따라 데이터를 안전하게 저장할 수 있는 저장장치의 안정성 유지가 더욱 중요하게 부각되고 있다.

서버 시스템에서 데이터의 안정성을 유지하기 위한 수단으로서 RAID(Redundant Array of Independent Disks)[5] 시스템이 주로 사용되는데, 본격적인 RAID 시스템의 경우 엄청난 설치 비용과 고도의 운영능력 확보가 요구되므로 개인 서버나 소형 서버 시스템에 적용하는 것은 어려움이 있다. 이에 따라 리눅스 OS에서는 소프트웨어에 의해 RAID 시스템을 구현한 소프트웨어-RAID[6] 기능을 내장하고 있으며, 이는 데이터의 안정성 유지에 큰 효과를 얻을 수 있다. 그러나 이는 기본적으로 소프트웨어에 의해 운영되는 시스템이므로 하드웨어에 의한 RAID 시스템에 비해 한계점이 존재하는데, 특히 저장장치에서 오류 발생 시 이를 관리자에게 빠르고 쉽게 알릴 수 없다는 점이 가장 큰 제약점이다.

이에 따라 Lee[7] 등은 리눅스 OS를 사용하는 서버에서 소프트웨어-RAID에서 오류가 발생된 상황을 관리자에게 빠르고 쉽게 알리기 위해 필요로 하는 구성 요소들에 관해 고찰한 바 있으며, 소프트웨어-RAID 시스템 개발 팀에서도 이의 개선책으로 오류 보고를 관리자의 이메일로 전송하도록 구성하였다[8]. 이 방법은 개발 초기에는 매우 효과적으로 관리자에게 오류보고를 할 수 있는 수단이었으나, 이러한 오류보고 메일로 위장해 각종 해킹

프로그램이나 랜섬웨어 등을 전송함으로써 인해 오히려 관리자의 보안이 취약해지는 폐단이 발생함에 따라 관리자는 보고 메일이 오더라도 열람하지 않고 무시하므로 큰 효과를 얻지 못하고 있다.

본 논문에서는 소프트웨어-RAID 시스템에서 가장 큰 제약점인 저장 장치의 상태 감시 문제를 관리자가 간단한 방법을 통해 직관적이면서도 원격에서 확인이 가능한 방법을 제시하고자 한다.

## 2. RAID 시스템

### 2.1 RAID 시스템의 동작원리 및 구성

RAID 시스템은 서버에서 저장장치의 데이터 안정성을 확보하기 위한 하드웨어 장치로서, 그 중 RAID Level-1(mirroring)[9]을 예로 들면, 2대의 동일한 저장장치를 설치하고 데이터를 기록할 때 2개의 저장장치에 동일한 데이터를 동시에 기록하며, 비록 용량은 1대 분량 밖에 확보되지 않지만 만약 한 대의 저장장치에서 이상이 발생한 경우 이를 LED 표시장치나 알람소리 등을 통해 관리자에게 알려 장치를 교체하도록 요구한다. 새롭게 저장장치가 교체되면 RAID 시스템에서는 자동으로 새로운 장치에 정상적인 장치의 내용을 그대로 복사해 두 장치들이 항상 동일한 상태가 유지되도록 한다. 이 방법은 2개의 장치가 동시에 파손되는 경우는 거의 없으므로 관리자가 신속히 대응하는 경우 데이터 안정성에 매우 유리한 시스템이 될 수 있다.

### 2.2 소프트웨어-RAID 시스템

소프트웨어-RAID는 하드웨어 RAID 시스템과 마찬가지로 2개의 동일한 저장장치를 설치한 후 이들을 1개의 저장장치로 관리하고, 서버에서 데이터를 기록할 때 RAID 관리 데몬(daemon)에 의

해 2개의 저장장치에 동일한 데이터를 기록하며, 이 때 단일로 구성된 저장장치나 하드웨어 RAID 시스템에 비해 2배의 시간이 소요되지만 하드웨어 RAID 시스템과 동일한 데이터 안정성을 확보할 수 있다.

소프트-RAID는 하드웨어 RAID 시스템과 달리 소프트웨어에 의해 구현되는 것이므로 LED 등과 같은 하드웨어를 이용해 오류 보고를 할 수 없으므로, 특정한 파일에 저장장치들의 상태를 수시로 기록하며, 서버 관리자는 이 파일을 열람함으로써 저장장치들의 상태를 확인할 수 있다.

소프트-RAID는 기본적으로 소프트웨어에 의해 운영되는 시스템이므로 몇 가지 제약점이 발생하는데, 그 중 가장 큰 제약점은 구성 저장장치에 이상이 발생했을 때 이를 관리자에게 빠르고 쉽게 알릴 수 없다는 것이다. 서버 시스템을 구성하는 저장장치에서 이상이 발생한 경우 시스템 관리자가 이를 최대한 빠른 시간에 인식하고 조치해야 데이터의 안정성이 확보되므로 하드웨어 RAID 시스템에서는 오류가 발생하면 경보와 함께 어떤 저장장치에 문제가 발생했는지 경고 램프 등을 통해 표시한다. 이에 비해 소프트웨어의 경우 관리자가 서버에 접속해 상태파일을 확인해야 이상을 인지할 수 있으므로, 장치에 이상이 발생한 사실을 관리자에게 빠르고 쉽게 알릴 수 없어 보수할 타이밍을 놓치는 경우가 자주 발생한다.

### 3. 소프트웨어-RAID의 보고 방법 개선

#### 3.1 홈페이지를 이용한 원격 확인

이상에서 고찰한 소프트웨어-RAID의 문제점에서, 서버의 콘솔 장치를 통해 로그-인하는 등 복잡한 과정 없이 확인하며, 특히 원격에서 수시로 확인할 수 있는 방법으로는 서버에서 발생하는 인터넷 홈페이지를 이용하는 것을 고려할 수 있다.

서버 시스템이 본래 웹-서버용이라면 기본적으로 홈페이지가 발생되며, 다른 용도의 서버라 할지라도 홈페이지 발생기능은 리눅스 OS에서 기본적인 기능이므로 홈페이지를 가동해 이를 통해 쉽게 확인할 수 있도록 구성한다.

#### 3.2 홈페이지에 기능 추가 방법

인터넷 홈페이지를 통해 소프트웨어-RAID의 장치 상태를 표시하도록 구성할 때, 홈페이지 자체에서 기능을 구현할 경우 홈페이지의 구성이 복잡해지고 홈페이지 채편성 시 보고 기능 또한 재구성되어야 한다. 따라서 홈페이지에서 소프트웨어-RAID 상태 표시 프로그램을 호출하는 방법으로 구성해 홈페이지의 고유기능과 소프트웨어-RAID 상태표시 기능을 분리한다. 홈페이지에서 소프트웨어-RAID 상태표시 프로그램을 간단히 호출하기 위해 홈페이지 코드 중의 원하는 곳에 <IMG SRC=""> 태그(tag)를 이용해 그림으로 표시하도록 구성한다[10].

#### 3.3 표시용 그림의 외부파일 구성

소프트-RAID 상태표시 프로그램에서 그림의 형태로 상태를 나타내도록 구성할 때, 그림 데이터를 프로그램 내부에 포함시킬 경우 홈페이지 배경색이나 디자인과 어울리도록 변경하려면 프로그램을 변경해야 하므로, 그림 데이터를 외부파일의 형태로 유지하고 이를 프로그램에서 가져오는(fetch) 형태로 구성한다.

이상에서 살펴본 사항들을 고려해 그림 1에 나타난 바와 같은 흐름으로 처리를 진행한다.

그림 1에서 인터넷 홈페이지를 구성하는 내용에 기록된 <IMG SRC=""> 태그에서 그림 이름 대신 상태표시 프로그램을 호출한다(①). 호출된 프로그램은 소프트웨어-RAID의 상태 보고파일을 분석해(②) 장치의 상태에 해당하는 그림을 가져와

(3) 이를 홈페이지로 회신하면(4) 홈페이지에 서는 이 그림을 홈페이지 상에 표시한다(5).

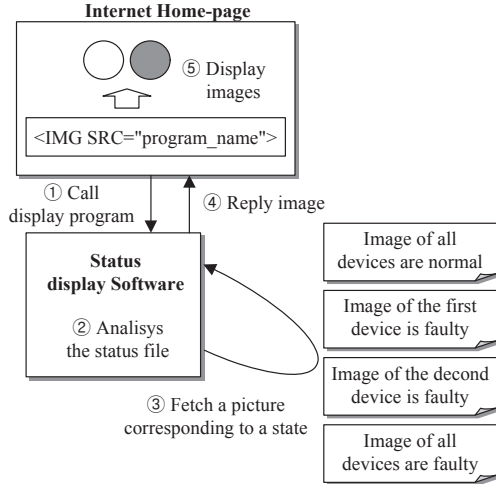


그림 1. 처리의 흐름도  
Fig. 1. Flow diagram of processing

발생한 장치는 under-bar로 표시된다. 이 파일의 내용을 해석해 그림 3과 같은 과정을 거쳐 상태를 보고하도록 구성하였다.

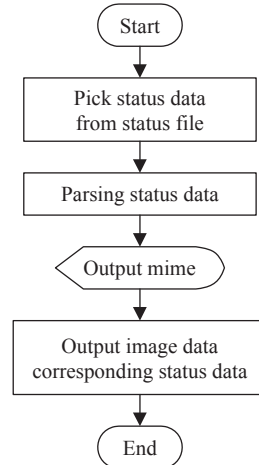


그림 3. 소프트웨어 플로차트  
Fig. 3. Software flowchart

## 4. 실험 및 결과

### 4.1 상태 보고 프로그램의 구성

소프트-RAID는 특정한 파일(mdstat 파일)에 저장장치들의 상태를 기록해 나타내며, 이의 예를 그림 2에 나타내었다.

```

Personalities : [raid1]
read_ahead 1024 sectors
md0 : active raid1 hdc1[0] hdd1[1]
      117218176 blocks [2/2] [UU]
unused devices: <none>
    
```

그림 2. 상태 보고파일의 예  
Fig. 2. Example of status report file

그림 2에서 [UU]라 표시된 부분이 저장 장치들의 가동상태를 표시한 것으로서, 두 저장 장치가 모두 정상일 경우 [UU]로 표시되고, 이상이

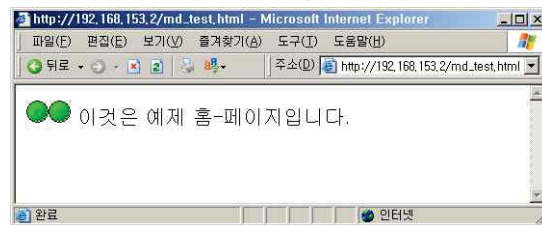
### 4.2 실험 결과

제시한 방법에 의해 작성된 mdstat.cgi 프로그램을 홈페이지에서 호출해 실행한 결과 예를 그림 4에 나타내었다.

```

<HTML>
<BODY>
  <IMG SRC="/cgi-bin/mdstat.cgi">
  이것은 예제 홈-페이지 입니다.
</BODY>
</HTML>
    
```

(a) Home-page code



(b) Result of execution

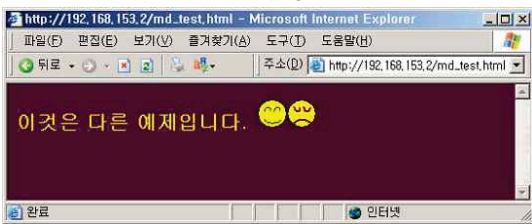
그림 4. 실행 예-1  
Fig. 4. Example of execution-1

그림 4(a)의 홈페이지 구성 코드 중 <IMG SRC=""> 태그를 삽입해 상태표시 프로그램을 호출하도록 구성하고, 이 홈페이지를 호출해 그림 4(b)와 같은 결과화면을 얻었다. 그림의 표시에서는 모두 녹색으로서 두 저장장치가 정상임을 나타내고 있다.

그림 5는 홈페이지를 다른 형태로 구성하고, 상태표시용 그림도 다른 형태로 구성한 예로서, 만약 홈페이지의 배경색이나 디자인이 변경되었을 경우 그림 5(b)와 같이 다른 그림의 형태로 구성할 수 있다. 이 예에서 1번 장치는 정상이지만 2번 장치에 이상이 발생한 것을 스마일 그림의 형태로 나타내었다. 즉, 홈페이지의 배경색이나 홈페이지의 분위기 및 디자인에 따라 그림 과일을 교체함으로써 일반 사용자들 입장에서는 어색하지 않으면서도 관리자 입장에서는 항상 서버의 상태를 확인할 수 있다.

```
<HTML>
  <BODY BGCOLOR="#600030">
    <FONT COLOR="Yellow">
      <B>이것은 다른 예제입니다.
    <IMG SRC="/cgi-bin/mdstat.cgi"></B>
    </FONT>
  </BODY>
</HTML>
```

(a) Home-page code



(b) Result of execution

그림 5. 실행 예-2

Fig. 5. Example of execution-2

## 5. 결론

본 논문에서는 Linux OS로 구성된 서버에서 사용되는 소프트-RAID 시스템에서, 저장 장치에서 발견된 오류들을 원격에서 직관적으로 확인할 수 있는 방법을 제시하였다. 이를 위해 리눅스 OS의 소프트-RAID 방식에서 오류 보고 방법의 원리 및 문제점을 분석하고, 저장장치들의 상태를 인터넷 홈페이지를 통해 원격으로 확인할 수 있는 방법을 제시하고 소프트웨어를 구성해 실험하였으며, 그 결과 다음과 같은 특징을 확인하였다. 첫째, 복잡한 로그-인 과정 없이 수시로 서버의 소프트-RAID 상태를 확인할 수 있어 시스템에 이상이 발생한 경우 관리자가 빠른 시간에 이에 대한 조치를 취함으로써 서버의 데이터 안정성이 크게 증진될 수 있다. 둘째, 홈페이지를 통해 표시하므로 인터넷이나 와이파이가 되는 곳이면 어디서나 원격에서 서버의 상태를 확인할 수 있다. 셋째, 상태표시용 그림을 외부 파일의 형태로 유지함으로써 홈페이지의 디자인에 맞는 그림으로 매우 쉽게 교체할 수 있다. 넷째, 홈페이지 태그를 이용해 표시하므로 홈페이지의 어느 위치에나 자유롭게 정보를 위치시킬 수 있다.

이상과 같은 특징들에 의해 관리자가 상태확인이 어려웠던 소프트-RAID 시스템의 단점이 해결되어 소프트-RAID의 사용과 신뢰성이 크게 향상될 수 있을 것으로 보이며, 향후 하드웨어적으로 LED나 알람 등의 수단을 이용해 직접 표시할 수 있는 방법까지 추가로 연구된다면 더욱 유용한 시스템이 될 수 있을 것으로 사료된다.

참고 문헌

[1] Donald E. Smith, "IP TV Bandwidth Demand: Multicast and Channel Surfing", 26th. IEEE INFOCOM 2007 proceedings, pp.2546-2550, May 29, 2007. DOI: 10.1109/INFCOM.2007.318

[2] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): A Vision, architectural elements, and future directions", Future Generation Computer Systems, Vol. 29, No. 7, pp.1645-1660, Sep. 2013. DOI: 10.1016/j.future.2013.01.010

[3] Melanie Swan, "Blockchain : Blueprint for a New Economy", O'Reilly Media, pp.1-6, Feb. 2015. ISBN: 978-1-493-92049-7

[4] Andrew McAfee and Erik Brynjolfsson, "Big Data : The Management Revolution", Harvard Business School, Oct. 2012. <http://tarjomefa.com/wp-content/uploads/2017/04/6539-English-TarjomeFa-1.pdf>.

[5] D. D.E. Long, Bruce R. Montague, and Luis Cabrera, "SWIF/RAID: A DISTRIBUTED RAID SYSTEM", University of California, pp.333-359, 1994. [http://www.usenix.org/legacy/publications/compsystems/1994/sum\\_long.pdf](http://www.usenix.org/legacy/publications/compsystems/1994/sum_long.pdf)

[6] J. Østergaard, E. Bueso, "The Software-RAID How To", Mar. 2010. <https://www.tldp.org/HOWTO/Software-RAID-HOWTO.html>

[7] Hyun-Chang Lee, Kyu-Tae Lee, Seo-Ik Kang, Sung-Yeol Kwon, and Min-Young Kim, "A Study on the Displaying status of Soft-RAID System in the Linux Server", 29'th Korea Software Assessment and Valuation Society Conference, pp.151-152, Nov. 2018.

[8] SoftRAID, SoftRAID Monitor, Other World Computing, 2018. [https://www.softraid.com/pages/features/softraid\\_monitor.html](https://www.softraid.com/pages/features/softraid_monitor.html).

[9] D. A. Patterson, G. Gibson, and R. H. Katz, "A case for redundant arrays of inexpensive disks(RAID)", '88 Proceedings of the 1988 ACM SIGMOD, pp.109-116, Jun. 1988. DOI:10.1145/50202.50214

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

저자 소개



나원식(Won-Sik Na)

2005.8 경희대학교 컴퓨터공학과 박사  
 2001.3 - 2003.2 (주) 성신점유 전산실장  
 2006.3-현재 남서울대학교  
 컴퓨터소프트웨어학과 부교수  
 <주관심분야> 네트워크 보안, 정보보호, 의료정보, 멀티미디어, U-computing



이현창(Hyun-Chang Lee)

1986.2 단국대 전자공학과 학사  
 1989.8 단국대 전자공학과 석사  
 1996.2 단국대 전자공학과 박사  
 1996.3-2004 국립 천안공업대학  
 정보통신과 부교수  
 2005.3-현재 국립 공주대학교 공과대학  
 정보통신공학부 교수  
 <주관심분야> 멀티미디어 회로, 전동기 제어회로, 마이크로프로세서, 임베디드 소프트웨어