

해상 전투 체계의 효율적 관리를 위한 SNMP 기반의 통합 관리 시스템 설계 및 구현

Design and Implementation of an SNMP-based Integrated Management System
for Efficient Management of Naval Combat System

김은희*
Kim, Eun-Hee

최미정*
Choi, Mi-Jung

홍원기*
Hong, James W.K.

조명섭**
Cho, Myeong-Seob

천성은**
Chun, Sung-Eun

이용욱**
Lee, Yong-Ook

ABSTRACT

The existing management system of Naval combat system does not provide efficient management methods since an integrating management mechanism does not exist. This paper examines the existing Naval combat system management techniques and summarizes their shortcomings. To solve the drawbacks, we apply the Simple Network Management Protocol(SNMP) mechanism and propose a more efficient management method. Based on SNMP, we analyze the requirements of management system from the perspective of an agent and a manager. We design the management system and also define the management information as SNMP Management Information Base(MIB). We implement a prototype based on our proposed design and verify our prototype by applying the system to the management of selected network devices in a real environment.

주요기술용어(주제어) : Naval Combat System, SNMP, Integrated Management System, Manager/Agent, MIB

1. 머리말

기존 국내 개발 해상 전투체계 관리는 각 장비별로 proprietary한 방법으로 이루어지고 있다. 직접 개발

한 장비에 대해서는 관리 메시지 포맷을 별도로 정의하여 장비 상태를 감시하며 랜 스위치, 프린터와 같은 구매 장비는 각 업체에서 제공하는 Web 기반의 UI를 사용하여 개별적으로 관리하고 있다. 이렇듯, 현 해상 전투체계 관리는 개별적 관리 체계를 통해 관리되므로 효율적이고 통합적인 관리 방법을 제공하지 못하고 있다. 또한, 현 전투 체계에서도 네트워크 연결을 통한 기능이 강화되면서 네트워크 관리가 중요한 요소가 되고 있다. 그러나 전투 체계의 네트워크 관리에 있어서 네트워크 성능 측정 기능과 네트워

† 2008년 2월 26일 접수~2008년 4월 4일 게재승인

* 포항공과대학교 컴퓨터공학과(POSTECH)

** 삼성탈레스(주) 시스템2팀(Samsung Thales Company)

주저자 이메일 : della@postech.ac.kr

크 장비를 감시하기 위한 장비 관련 정보 등이 부족하다는 점, 그리고 네트워크 장비를 reboot 하거나 shutdown 시킬 수 있는 기능이 없다는 점이 현 전투 체계관리의 한계점으로 지적되고 있다. 본 논문에서는 현 해상 전투체계 관리 기법의 부족한 점을 구체적으로 알아보고, 기존 관리 체계에 네트워크 관리 표준인 SNMP를 적용하여 그 부족한 점을 해결하고, 더 효율적으로 관리할 수 있는 방법을 제시하고자 한다.

SNMP는 네트워크 관리를 위한 인터넷 표준 프레임워크로 간단하면서도 가볍다. 또한 기존의 상용 네트워크 장비는 기본적으로 SNMP 에이전트를 탑재하여 관리 기능을 제공하므로 새롭게 개발하는 해상 전투 체계내의 시스템에만 SNMP 에이전트를 개발하여 탑재하면 상용 제품을 포함한 해상 전투 체계내의 모든 시스템에 대해 현 전투 체계 관리의 한계점으로 지적되고 있는 네트워크 성능 정보 및 reboot 등의 제어 action까지 포함하여 SNMP 기반으로 효율적으로 통합 관리할 수 있게 된다.

본 논문의 구성은 다음과 같다. 2장에서는 관련연구로 SNMP 개요와 SNMP 관리를 해상 전투체계에 도입한 사례에 대해 간단히 설명한다. 3장에서는 기존 해상 전투체계 관리 기법을 분석하고 기존 관리 체계의 부족한 점을 제시하는 동시에 SNMP를 이용하여 부족한 점을 해결하는 방법을 제시한다. 4장에서는 SNMP 기반 해상 전투체계 관리 요구사항을 각각 매니저와 에이전트로 나누어 분석하며, 관리 대상의 관리 정보를 SNMP MIB으로 정의한다. 5장에서는 분석한 요구사항을 토대로 매니저와 에이전트의 구조를 제시한다. 6장에서는 5장에서 설계한 내용을 바탕으로 프로토타입 개발에 대해서 설명한다. 프로토타입 구현을 위한 관리 대상 선정 및 관리 대상에 따른 관리 항목에 대하여 살펴보고 관리 각 기능을 설명한다. 마지막으로 7장에서는 연구 내용을 정리하고 SNMP 기반의 해상 전투체제로 나가기 위해 추후에 진행되어야 할 연구방향에 대해서 논의한다.

2. 관련연구

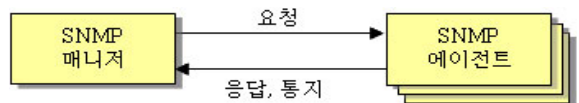
이 장에서는 SNMP 개요와 SNMP 관리를 해상

전투 체계에 도입한 사례에 대해서 설명한다.

가. SNMP 개요

네트워크 관리는 안정적이면서 안전하고 효율적인 네트워크 환경을 제공하기 위해서 네트워크 상에 존재하는 다양한 자원들을 모니터링하고 제어하는 것을 말한다. 네트워크 관리 프로토콜로 대표적인 것은 TCP/IP 네트워크에서 사용되는 SNMP^[1]이다. 인터넷의 관리 체제인 SNMP는 IETF(Internet Engineering Task Force)^[2]에 의해 표준화가 이루어지고 있으며 네트워크 관리를 위한 관리 구조, 관리 정보의 구조, 관리 프로토콜 등에 대한 표준화 작업을 진행하고 있다.

SNMP 네트워크 관리 구조는 그림 1과 같이 매니저/에이전트 구조로 되어있다. 관리 대상 장비에는 SNMP 에이전트가 탑재되어 관리 대상 장비의 정보를 수집하며, SNMP 매니저는 네트워크 관리자가 장비를 관리할 수 있도록 에이전트로부터 정보를 수집하고 처리하는 기능을 수행한다. 에이전트의 정보를 매니저가 수집하는 방법은 폴링(Polling)과 트랩(Trap)의 두 가지 방법이 있다. 폴링은 매니저의 요청에 대해 에이전트가 응답하는 방법이고, 트랩은 에이전트가 매니저의 요청 없이 발생된 이벤트에 대해서 매니저에게 통보하는 방법이다.



[그림 1] SNMP 네트워크 관리 구조

나. SNMP 관리를 해상 전투체계에 도입한 사례

SNMP 관리를 해상 전투 체계에 도입한 예는 영국, 네덜란드, 미국에서 찾아볼 수 있다. 영국은 2006년에 진수한 Type 45 차기구축함의 관리에 SNMP 기반의 관리 기법을 적용하여, 장비에 대한 가용/비가용 상태를 중점적으로 감시하는 장애 관리를 대체하였으며, 네덜란드의 경우 TACTICOS 전투관리체계에 SNMP를 이용한 전투체계 네트워크 관리를 새로이 도입하고 있다. 미국은 2004년 미해군이 OA(Open Architecture)의 기능적 요소와 기술적 요소를 발표

하였으며, OA 컴퓨팅 환경 기술과 표준의 14가지 중에서 Resource Management에 적용 가능한 표준으로 SNMP를 규정하여 적용하였다.

3. 기존 해상 전투체계 관리 기법 분석

이 장에서는 먼저 기존 관리 체계의 기능을 살펴보고, 기존 관리 체계의 문제점과 SNMP를 적용하여 문제점을 해결하는 방안을 제시한다.

가. 기존 해상 전투체계 관리 기능 소개

기존 해상 전투체계 관리 기능은 모드관리, 체계 감시 및 통제, 운용자 관리, 체계 파라미터 관리 등으로 구성된다. 그 중에서 본 논문에서 초점을 맞춰 SNMP 기반 관리로 적용 하고자 하는 기능은 체계 감시 및 통제 기능이다.

체계 감시 및 통제 기능은 운용자가 지휘무장통제 체계(CFCS)를 구성하는 노드들에 대해서 상태를 실시간으로 감시할 수 있게 한다. 장비상태는 가용, 기능저하, 비가용의 3가지 상태를 가지며 가용은 정상적으로 장비를 운용할 수 있는 상태, 기능저하는 장비의 일부 기능이 사용 불가능하나 운용은 가능한 상태, 비가용은 장비를 운용할 수 없는 상태를 의미한다.

나. 기존 해상 전투체계 관리의 문제점과 해결방안

기존 해상 전투체계 관리 기능의 문제점 및 SNMP 기반의 관리를 통한 해결방안은 아래와 같다.

첫째, 기존 해상 전투체계 관리에서는 통합적인 관리 방법 및 interface가 제공되지 않는다. 개발 장비에 대해서는 다기능 콘솔의 전용 감시화면에서, 상용 장비에 대해서는 별도의 Web 기반의 UI를 통해서 관리가 이루어진다. 따라서 운용자는 각 장비를 관리하기 위하여 여러 개의 UI에 접속하여 관리해야 하므로 관리가 번거로우며 통합된 관리 View를 제공받지 못하고 있다. 이러한 단점은 SNMP 기반의 관리 프레임워크를 통하여 통합된 관리 방법을 제공함으로써 해결 가능하며 또한 통합된 UI도 제공 가능하다.

둘째, 기존 전투체계 관리는 성능 관리기능이 미흡하다. 네트워크 인터페이스별 성능 정보와 각 장비와

랜 스위치 관련된 성능 정보(CPU, Memory)를 알 수 없었다. 이는 SNMP MIB-II^[3]의 Interfaces 그룹에서 제공하는 트래픽의 통계 정보와 인터페이스별 네트워크 트래픽 정보를 얻음으로써 해결할 수 있다. 또한 장비 성능 정보는 HOST-Resources MIB^[4]의 CPU 및 Memory 사용률 모니터링을 통하여 제공받을 수 있다.

셋째, 기존 전투체계 관리는 네트워크 장비 관련 정보를 제공하는 기능이 미흡하다. 각 SBC(Single Board Computer) Board는 CPU와 Memory 정보를 외부 장비 연결(노트북)을 통해서만 알 수 있으며, 장비 감시 단계에서는 가용/비가용의 감시가 가능하지만 LRU(Line Replaceable Unit) 단위의 정보는 알 수 없다. 이는 각 SBC Board에 SNMP 에이전트를 구현하여 탑재시키고 관리 정보에 대해 추가적인 extension MIB을 정의하여 제공함으로써 해결 할 수 있다.

마지막으로, 기존 전투체계 관리는 네트워크 장비를 reboot하거나 shutdown시키는 등의 제어를 수행할 수 있는 방법이 마련되어 있지 않다. 이에 대한 해결방법으로는 reboot과 shutdown에 해당하는 MIB 값을 정의하여, 해당하는 MIB 값의 변화에 따라 control action을 호출하여 관리 제어를 수행할 수 있게 한다. MIB 값의 변화에 따라 control action을 호출하여 관리 제어를 수행 하는 것을 side effect 효과라 하는데, 이를 통하여 장비의 reboot, shutdown 기능을 제공할 수 있다.

4. SNMP 기반 해상 전투체계 관리 요구사항

3장에서 기존 해상 전투체계 관리 기능의 문제점과 해결책을 살펴보았다. 이를 바탕으로 SNMP 기반의 해상 전투체계 매니저와 에이전트의 요구사항을 분석한 내용은 다음과 같다.

가. 해상 전투체계 SNMP 매니저 요구사항

해상 전투체계 SNMP 매니저 요구사항을 기능적인 요구사항과 비기능적인 요구사항으로 나누어 분석한다.

1) 기능적 요구사항

가) 장애 관리

장애 관리 기능으로 첫째, 실시간 장애 표시 및 알람 설정 기능이 제공되어야 한다. 관리하는 장비에 장애가 발생한 경우, Map을 통해 장비의 아이콘 색을 변경시켜 실시간으로 장애를 표시해야 한다. 둘째, 임계값(Threshold)에 따른 이벤트 발생 설정 및 기록 관리 기능이 제공되어야 한다.

나) 구성 관리

구성 관리 기능으로 첫째, 관리 장비의 자동 검색 및 수동 추가, 삭제 기능이 제공되어야 한다. 둘째, Map을 통한 효과적인 장비 관리 기능이 제공되어야 한다. 셋째, 장비의 IP 주소, 시스템 이름, 장소, 관리자 정보 등 관리 장비의 구성 정보를 설정 및 변경할 수 있는 기능이 제공되어야 한다.

다) 성능 관리

성능 관리 기능으로 첫째, 서버의 성능 관리정보에 해당하는 CPU, Memory, Disk 사용률 등을 모니터링 하는 기능을 제공해야 한다. 둘째, 네트워크 인터페이스의 성능 관리 정보에 해당하는 트래픽양, 에러량 등을 모니터링하고 분석하는 기능을 제공해야 한다. 셋째, 서버 및 네트워크 장비의 성능을 모니터링 한 결과를 그래프로 보여주는 기능을 제공해야 한다.

2) 비기능적 요구사항

해상 전투체계 SNMP 매니저의 비기능적인 요구사항은 다음과 같다. 첫째, 해상 전투체계 SNMP 매니저는 쉽고 편리하며 기존 관리 시스템의 UI와 비슷하여 사용자에게 친숙한 사용자 인터페이스를 제공해야 한다. 둘째, 해상 전투체계 관리에 있어서 중요한 정보는 실시간으로 운용자에게 보고되어야 한다. 예를 들어 장비 상태의 주요 변경에 대해서는 3초 내에 화면에 갱신되는 것을 보장해야 한다.

셋째, 관리 대상 및 네트워크에 부하를 주지 않고 관리 기능이 수행되어야 한다. 넷째, fault-tolerant한 관리 기능이 제공되어야 한다. 관리 서버의 장애 발생시, 관리 기능을 끊임없이 지속적으로 제공하기 위하여 제 2관리 서버가 제 1관리 서버를 대신하여 모

든 관리를 인계(Takeover)받는 방안이 마련되어야 한다. 마지막으로, 관리 정보의 백업 기능이 제공되어야 한다.

나. 해상 전투체계 SNMP 에이전트 요구사항

본 장에서는 해상 전투체계 SNMP 에이전트 요구사항을 기능적인 요구사항과 비기능적인 요구사항으로 나누어 분석한다.

1) 기능적 요구사항

해상 전투체계 SNMP 에이전트의 기능적인 요구사항으로 첫째, SMIV2^[5]와 SNMPv1 프로토콜(get/set/getnext/trap operations)을 지원해야 한다. 둘째, 각 장비별 관리 정보 값을 매니저에게 제공하기 위하여 실제 관리 대상의 resource에서 수집한 값을 MIB 변수에 채워 넣는 backend interface 기능이 제공되어야 한다. 셋째, 기존 해상 전투 관리 체계가 제공하는 모든 관리 정보와 관리 기능을 제공해야 한다. 그 이외에 side effect 효과를 통하여 장비를 shutdown/reboot 시키는 기능을 제공해야 한다. 마지막으로, SNMP 에이전트의 구성 정보인 community name, port, trap port, trap host IP 등을 설정하는 기능이 제공되어야 한다.

2) 비기능적 요구사항

해상 전투체계 SNMP 에이전트의 비기능적인 요구사항으로 SNMP 에이전트는 첫째, 내장되는 장비의 다른 일들을 방해하지 않을 정도로 lightweight 하게 구현되어야 한다. 둘째, 추후에 필요한 관리 정보와 기능의 추가시 쉽게 확장 가능하게 구조화되어야 한다. 마지막으로, 어떤 새로운 장비에도 탑재되기 위해 유연하게 모듈화되어 설계되고 구현되어야 한다.

5. 해상 전투체계 관리 시스템 설계

본장에서는 4장에서 분석한 요구사항을 바탕으로 해상 전투체계 관리정보를 MIB으로 정의하였으며 해상 전투체계 관리 시스템을 위한 매니저와 에이전트 구조를 제시한다.

가. 해상 전투체계 관리정보 분석 및 MIB 정의

기존 해상 전투체계에서 관리하는 관리정보를 SNMP 기반의 관리체계로 관리하기 위해서는 관리정보를 MIB으로 정의해야 한다. 관리정보는 크게 시스템 정보, 네트워크 정보, 기타 장비 정보로 나누어 표 1, 표 2, 표 3으로 정리하였다. 기존에 정의된 표준 MIB 과 CISCO 등 각 장비 업체에서 정의한 enterprise MIB은 새로운 정의없이 기존 MIB을 참조(Reference) 하여 사용한다.

[표 1] 시스템 관리 정보

관리대상 정보	값	MIB
장비명/호스트명	Hostname	RFC1213-MIB
IP	ip address	RFC1213-MIB
OS/Version		RFC1213-MIB
장비의 기술적 연결상태	Online/Offline	STC-MIB
장비 운용가용성	Available/ Degraded/ Not Available	STC-MIB
SBC Board 상태	Normal/Fault	STC-MIB
Ethernet Card 상태	Normal/Fault	STC-MIB
PMC 422 Card 상태	Normal/Fault	STC-MIB
Discrete Card 상태	Normal/Fault	STC-MIB
Synchro Card 상태	Normal/Fault	STC-MIB
SCRAMNet Card 상태	Normal/Fault	STC-MIB
Shutdown	On/Off	STC-MIB
장비 CPU 사용률 - Linux - Windows NT(XP)	%	UCD-SNMP-MIB HOST-RESOURCES-MIB
장비 Memory 사용률	%, KB	HOST-RESOURCES-MIB
장비 Disk 사용률	%, GB	HOST-RESOURCES-MIB

[표 2] 네트워크 관리 정보

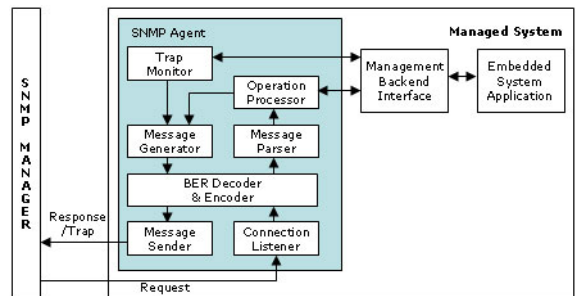
관리대상 정보	값	MIB
OS/Version		RFC1213-MIB
CPU 사용률	%	OLD-CISCO-CPU-MIB
Memory 사용률	%, MB	CISCO-MEMORY-POOL-MIB
Peak/AVG Traffic	Bytes, %	CISCO-STACK-MIB
Restart	On/Off	STC-MIB
포트별 Tx/Rx Traffic	Bytes	RFC1213-MIB
포트별 Tx/Rx Traffic Error	Bytes	RFC1213-MIB

[표 3] 프린터(기타장비)의 관리 정보

관리대상 정보	값	MIB
CPU 사용률	%	HOST-RESOURCES-MIB
Memory 사용률	%, MB	HOST-RESOURCES-MIB
총 인쇄매수 LifeCount, PowerOnCount	매	PRINTER-MIB
토너상태(Toner Empty/Toner Low/Almost Empty)		PRINTER-MIB

나. 해상 전투체계 SNMP 에이전트 설계

SNMP 에이전트는 그림 2와 같은 구조를 가지며, 각 장비에 탑재되어 관리 정보를 제공하는 역할을 한다.



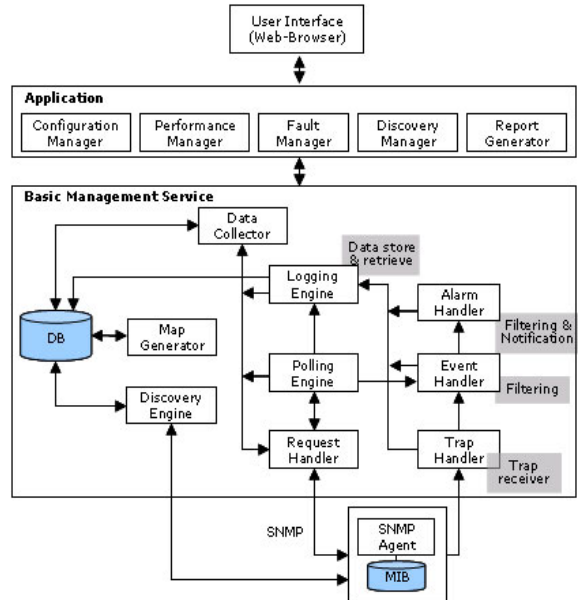
[그림 2] 에이전트의 구조

- Connection Listener : Default로 161번 port를 열어두고 SNMP 매니저로부터 요청 메시지를 기다린다.
 - BER Decoder & Encoder : 수신/응답 메시지를 디코딩/인코딩해서 Message Parser/Message Sender로 전달한다.
 - Message Parser : 디코딩한 메시지를 파싱하여 오퍼레이션과 OID(Object Identifier) 값을 분석한다.
 - Operation Processor : Message Parser로부터 받은 메시지 처리에 따라 해당 OID에 관리 동작(Get/Set)을 수행한다.
 - Trap Monitor : Trap으로 정의된 정보에 대해서 값이 변경되면 SNMP 매니저에게 알리도록 실시간으로 모니터링 한다.
 - Message Generator : Trap Monitor와 Operation Processor의 요청 시 매니저에게 전송할 응답메시지를 생성한다.
 - Message Sender : 인코딩된 메시지를 매니저에게 전송한다.
 - Management Backend Interface : 관리 정보를 MIB값에 채워 넣기 위해서 실제 resource로부터 값을 가져오는 역할을 수행한다.
 - Embedded System Application : 장비의 embedded OS를 포함한 시스템 응용 부분이다.
- 예를 들어, SNMP 매니저가 SNMP 에이전트에게 장비명에 대한 Get 오퍼레이션을 요청할 때, 이 메시지는 Connection Listener가 받게 되고 BER 인코딩된 메시지를 디코딩하여 Message Parser에게 넘겨주게 된다. 파서는 메시지의 오퍼레이션 종류와 OID를 파싱하여 Operation Processor에게 넘겨주면 Operation Processor는 Backend Interface 모듈을 통하여 실제 장비의 장비명을 가져다주게 된다. 이 때 Message Generator는 Get 명령에 대한 Response 메시지를 생성하여 BER Encoder에게 넘기게 되고 BER Encoder는 메시지를 인코딩하여 Message Sender 모듈을 통해 매니저에게 보내게 된다.

다. 해상 전투체계 SNMP 매니저 설계

해상 전투체계 SNMP 매니저는 그림 3과 같은 구

조를 가지고 있다. Basic Management Service의 각 모듈은 아래와 같은 기능을 수행한다.



[그림 3] 매니저의 구조

- Data Collector : 각 Application이 요청하는 데이터를 SNMP 에이전트로부터 수집하여 Application으로 전달한다.
- Polling Engine : 주기적으로 모니터링해야 하는 관리정보에 대해 각 정보의 주기에 따라 Request Handler에게 요청한다.
- Request Handler : SNMP 에이전트에게 SNMP Get/Set 메시지를 보내고 응답받아 전달한다.
- Logging Engine : 주기적인 모니터링 정보와 이벤트, 알람 정보에 대해 정보의 발생 시간을 포함한 내용을 DB에 저장한다.
- Map Generator : Discovery Engine에서 찾은 장비에 대해 장비간의 연결 정보를 포함한 토폴로지 Map을 생성한다.
- Trap Handler : SNMP 에이전트에서 발생한 Trap을 받아 처리한다.
- Event Handler : Trap Handler로부터 이벤트 데이터를 받아서 처리하고 Logging Engine을 통해 저장한다.

- Alarm Handler : Event Handler로부터 받은 이벤트를 처리하여 알람을 생성하고 저장하며 Application에 통보한다.

사용자가 UI를 통해서 현재 네트워크의 In/Out 트래픽 양을 보고 싶으면 매니저의 Application 내의 Performance Manager를 통하여 요청을 보내게 된다. 이 때 Performance Manager는 네트워크 트래픽에 대한 요청을 Basic Management Service내의 Data Collector에게 보내게 되고 이 요청은 Request Handler에게 넘겨진다. Request Handler는 SNMP Get 메시지를 생성하여 해당 SNMP 에이전트에게 보내게 되고 이 메시지에 대한 응답을 SNMP 에이전트로부터 받아서 요청한 반대의 순서로 사용자 UI에 전달하게 된다.

라. 연동단 관리를 위한 STC-MIB

그림 4는 관리 대상 장비 중 연동단 장비의 관리를 위해 관리 정보를 선정하여 MIB을 정의한 내용이다. 8개의 관리 객체로 구성되어 있으며, 각각의 관리 정보는 다음과 같다. 'icuStatus'는 연동단의 현 상태가 offline인지 online인지 알려주는 정보이다. 'sbcBdStatus', 'etherNetCardStatus', 'pmc422CardStatus'는 각각 SBC 보드, EtherNet 카드, PMC 422 카드의 상태가 fault인지 normal인지를 알기 위해 정의된 정보이며, 'restart'는 재부팅을 위해 정의한 것으로 control action 기능을 side effect 효과를 통해 얻을 수 있다. 다음으로, 'icuCpuBusy', 'icUMemoryUsed', 'icUStorageSize'는 각각 CPU, MEMORY, DISK의 사용률을 얻기 위해 정의한 정보이다.

6. 관리 시스템 프로토타입 개발

이 장에서는 5장의 설계 구조를 바탕으로 관리자가 웹 인터페이스를 통해 네트워크를 효율적으로 관리할 수 있는 에이전트와 매니저의 프로토타입 개발에 대해서 설명하도록 한다. 먼저, 프로토타입 구현을 위한 관리 대상 및 각 관리 항목에 대해서 살펴보고, 관리 기능에 대해서 UI를 기준으로 설명하도록 한다.

```

-- STC-MIB-icu.my
-- 연동단 관련 관리 정보: ICU 기술적 상태, SBC 보드 상태, Ethernet 카드 상태, PMC
422 카드 상태, 재부팅을 위한 설정값, 성능 모니터링을 위한 CPU 사용률, memory 사용
률, storage 사용률

STC-MIBv2 DEFINITIONS ::= BEGIN
IMPORTS
    enterprises                FROM RFC1155-SMI
    OBJECT-TYPE                FROM SNMPv2-SMI;
--
-- 1.3.6.1.4.1.24623
stc OBJECT IDENTIFIER ::= { enterprises 24623 }
-- define System Management Information for STC
--
-- 1.3.6.1.4.1.24623.1
icu OBJECT IDENTIFIER ::= { stc 1 }
--
-- 1.3.6.1.4.1.24623.1.1
icuStatus OBJECT-TYPE
    SYNTAX INTEGER {
        offline(0), online(1) }
    MAX-ACCESS STATUS
    DESCRIPTION
        "Status of the ICU"
    ::= { icu 1 }
--
-- 1.3.6.1.4.1.24623.1.2
sbcBdStatus OBJECT-TYPE
    SYNTAX INTEGER {
        fault(0), normal(1) }
    ::= { icu 2 }
--
-- 1.3.6.1.4.1.24623.1.3
etherNetCardStatus OBJECT-TYPE
    SYNTAX INTEGER {
        fault(0), normal(1) }
    ::= { icu 3 }
--
-- 1.3.6.1.4.1.24623.1.4
pmc422CardStatus OBJECT-TYPE
    SYNTAX INTEGER {
        fault(0), normal(1) }
    ::= { icu 4 }
--
-- 1.3.6.1.4.1.24623.1.5
restart OBJECT-TYPE
    SYNTAX INTEGER {
        off(0), on(1) }
    ::= { icu 5 }
--
-- 1.3.6.1.4.1.24623.1.6
icuCpuBusy OBJECT-TYPE
    SYNTAX OCTET STRING (SIZE (0.5))
    ::= { icu 6 }
--
-- 1.3.6.1.4.1.24623.1.7
icUMemoryUsed OBJECT-TYPE
    SYNTAX OCTET STRING (SIZE (0.5))
    ::= { icu 7 }
--
-- 1.3.6.1.4.1.24623.1.8
icUStorageSize OBJECT-TYPE
    SYNTAX OCTET STRING
    ::= { icu 8 }
END
    
```

[그림 4] 연동단 관리를 위한 Extension MIB

가. 관리 대상

기존 시스템에서 실제 사용되는 네트워크 장비들 중 대표적인 장비들로 SNMP 매니저가 설치되는 윈도우 기반의 관리콘솔, 데이터를 저장하는 자료서버, 스위치, 프린터, 연동단의 5대 장비를 관리대상 장비

로 선정하여 시험을 위한 Test Bed를 구축하였다. 스위치와 프린터에는 기본적으로 각 vendor에서 제공하는 SNMP 에이전트가 자동 실행되며, 윈도우 기반의 관리콘솔과 자료서버의 경우는 윈도우 구성요소에서 관리 및 모니터링 도구를 선택하여 설치하여 실행함으로써 SNMP 에이전트를 실행시키게 된다. 연동단의 경우는 직접 SNMP 에이전트를 개발하여 탑재시킴으로써 관리 기능을 수행한다. 각 장비별 관리 정보는 5장에서 정의한 것을 사용한다.

나. 에이전트

연동단의 경우, 그림 4에서 정의한 확장 MIB을 포함한 SNMP 에이전트를 VxWorks^[6] OS에서 Wind River의 SNMP 에이전트 개발 tool kit을 이용하여 개발하였다. 먼저 그림 4의 연동단 관리 extension MIB을 정의한 후 MIB2C stub 생성 컴파일러를 사용하여 c stub 코드를 자동으로 생성한다. 연동단의 경우 관리 항목은 그림 4처럼 8가지 항목이고, 이 중에서 read-only 값에 대해서는 해당 registry 값을 back-end interface를 호출하여 값을 가져다 보여지게 된다. reboot과 같은 제어 값에 대해서는 값이 1로 설정되었을 때 내부적으로 reboot() 함수를 호출하여 실제로 연동단 장비가 재부팅되는 제어 action을 취하게 구현하였다. 에이전트의 프로토타입 구현에 있어서 trap 부분은 제외되었다.

다. 매니저

1) 개발 환경

개발환경으로는 하드웨어는 CPU 2.6GHz, 1GB 메모리를 사용했으며, 소프트웨어 환경은 윈도우 XP 운영체제하에 톱캣 서버와 오라클 데이터베이스를 이용하여 자바 언어로 구현했으며, AdventNet에서 제공하는 SNMP API^[7]를 사용하였다.

2) 매니저 기능

가) 로그인

웹 브라우저를 실행 시켜서 해당 관리 시스템의 URL을 입력하면 사용자 이름과 패스워드를 넣는 로그인 화면이 나타난다. 사용자는 크게 모든 설정을 포함한 관리 기능을 수행할 수 있는 Super Admin과 관

리 정보에 대한 열람만 가능한 일반 사용자로 나뉜다.

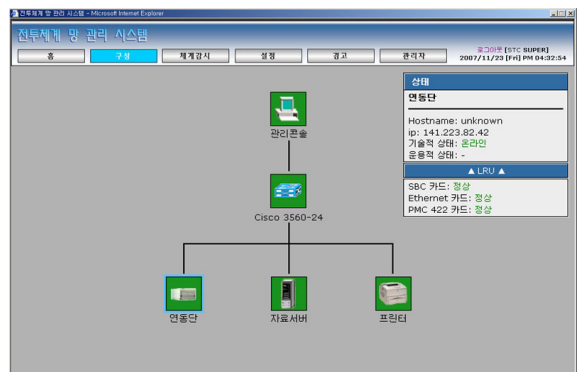
나) 홈

로그인 성공시, 접속하는 첫화면으로 전투체계 망 관리 시스템의 기능에 대해 각 메뉴별로 간단한 기능에 대해서 설명한다.

다) 구성

메뉴에서 구성을 선택시, 그림 5와 같은 대표 관리 장비 5대로 구성된 고정된 맵을 볼 수 있다. 장비의 online/offline의 상태에 따라 녹색/적색으로 구분되며, 현재 화면은 모두 정상(online) 상태이다. 각각의 장비 그림을 선택하면 장비의 상태 정보가 오른쪽 화면 윗단에 나타나게 되며 장비의 종류, 장비의 이름, IP 주소, 기술적 상태를 제공한다. 관리콘솔과 자료서버의 경우에는 기술적 상태뿐 아니라 운용적 상태 정보도 제공한다. 또한, 연동단의 경우 LRU단의 SBC, Ethernet, PMC 422 카드의 상태의 normal/fault 상태 정보도 확인할 수 있다.

프로토타입 구현에 있어서 trap 처리 부분은 제외되어서 장비의 상태 변화에 대한 알람메시지는 주기적인 모니터링을 통하여 이루어진다. 장비가 online에서 offline 상태로 바뀌게 되면 알람이 발생하면서 장비의 상태가 장비 상태 모니터링 주기(최소 약 3초) 이내에 적색으로 전시된다.



[그림 5] 구성 화면

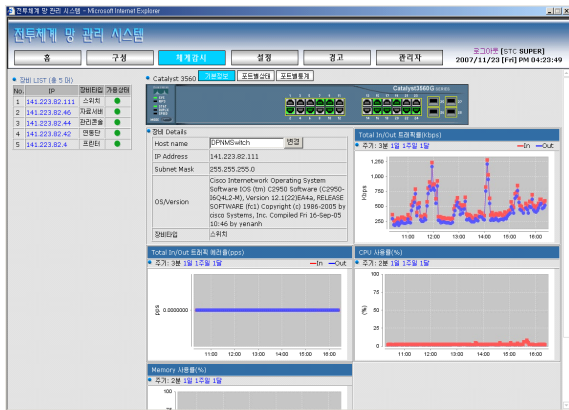
라) 체계감시

체계감시 메뉴를 선택하면 왼쪽에는 장비 리스트로

장비의 IP 주소 및 장비 상태가 전시된다. 왼쪽의 장비 리스트에서 장비 클릭시에 해당되는 장비의 상세 정보 및 성능정보가 오른쪽 화면에 전시된다. 장비마다 선정된 관리항목이 다르므로 각각에 대해서 알아 본다.

① 스위치

그림 6은 장비 리스트에서 스위치를 선택한 화면으로 스위치의 기본정보 화면에 해당한다. 가장 위쪽은 실제 스위치의 그림에 해당하며 스위치 포트별 사용여부가 녹색/회색으로 구분하여 나타난다.



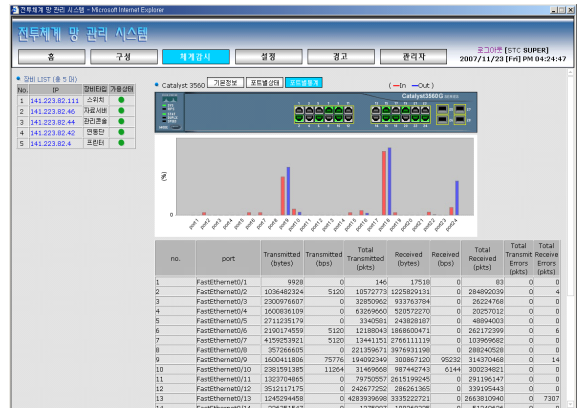
[그림 6] 체계감시 중 스위치의 기본정보

그 아래쪽에 장비의 상세 정보로 Host name, IP Address, Subnet Mask, OS/version과 장비타입을 볼 수 있으며, 성능정보로 Total In/Out 트래픽률, Total In/Out 트래픽 에러율, CPU 사용률, Memory 사용률을 최근 6시간 단위가 그래프로 전시된다. 각 그래프는 각 성능의 주기가 표시되며 1일, 1주일, 1달 링크를 누르게 되면 새창이 뜨며 최근 1일의 정보가 각 시각별 단위로, 최근 1주일의 정보가 1일 단위로, 최근 1달(30일)의 정보가 1일 단위로 전시된다.

스위치 장비 그림 위쪽에 있는 '포트별상태' 버튼을 선택하면 스위치의 각 포트별 상태정보를 볼 수 있다.

그림 7은 '포트별통계'를 선택한 경우의 화면으로 각 포트별 통계 정보를 보여준다. 그 아래 표는 장비의 각 포트별 In/Out 트래픽양과 In/Out 트래픽 에러량을 나타낸다. 표 중에서 Transmitted(bps), Received

(bps)은 각 포트별 1분동안의 In/Out 트래픽양을 계산하여 숫자로 나타내며, 막대 그래프는 이 값을 각 포트별 bandwidth로 나누어 포트별 이용률(%)을 나타낸다. 각 포트별 값에서 적색 막대 그래프가 1분동안 In 트래픽 이용률이며, 청색이 Out 트래픽 이용률을 나타낸다.



[그림 7] 체계감시 중 스위치의 포트별 통계 정보

② 자료서버/관리콘솔

윈도우 기반의 자료서버와 관리콘솔은 장비의 상세 정보로 Host name, IP Address, Subnet Mask, OS/version과 장비타입을 볼 수 있으며 성능정보로 CPU, Memory, Disk 사용률을 그래프로 제공한다.

③ 프린터

장비 리스트에서 프린터를 선택한 경우에 상세정보로 Host name, IP Address, Subnet Mask, OS/version과 장비타입을 제공하며, 추가적으로 총 인쇄매수정보로 LifeCount, PowerOnCount를 제공하며, 토너 상태 정보로 토너가 충분한 경우에는 'good'을 표시하며, 토너량에 따라 Toner Empty/Toner Low/Almost Empty를 표시하게 된다.

④ 연동단

장비 리스트에서 연동단을 선택한 경우, 상세정보로 Host name, IP Address, Subnet Mask, OS/version과 장비타입을 제공하며, 추가적으로 재부팅 기능이 제공된다. 재부팅 버튼을 누르게 되면 정의한

MIB 값이 변경되어 그 값을 모니터링 하고 있는 에이전트에서 값 변화를 인지하여 재부팅 기능을 수행하게 된다.

마) 설정

메뉴에서 설정을 선택하면 관리되는 장비 리스트를 볼 수 있으며 장비 리스트에서 더 이상 관리가 불필요한 장비를 선택하여 삭제할 수 있다. 또한, 새로운 장비를 추가할 수 있는 기능이 있으며 각 장비별로 주기적인 모니터링을 위한 폴링 주기를 설정하는 기능도 수행한다.

바) 경고

메뉴에서 경고를 선택하면 각 장비별 임계값을 설정할 수 있으며 여기서 설정한 임계값을 넘게 되면 관리자에게 경고창을 띄워 알리게 된다. 또한, 이 화면에서 알람 리스트를 기본적으로 30일간을 보여주며 기간별, 장비별, 알람 메시지 별로 검색하여 필터링하는 것이 가능하다.



[그림 8] 경고 메시지 예

장비의 연결상태가 offline 상태가 되었을 때와 각 장비별로 설정한 임계값을 넘게 되면 그림 8과 같은 경고창을 띄워 알리게 된다. 경고창을 통해 알람 발생시간, 장비명, IP 주소, 알람메시지를 알 수 있다. 그림 8은 연동단(ICU)의 기술적인 연결상태가 offline 상태로 되어 발생한 경고창에 해당한다.

사) 관리자

Super Admin의 관리자 화면에서는 ‘일반 사용자 추가/삭제’, ‘Super Admin 사용자 정보 변경’, ‘백업’ 기능을 수행할 수 있다.

7. 맺음말 및 향후과제

본 논문에서는 효율적인 SNMP 기반의 해상 전투 체계 관리 시스템 설계를 위해서 기존 전투체계의 관리 기법을 분석하여 관리에 있어 부족한 점을 정리하고 해결방안을 제시했다. 현 전투 체계관리의 한계는 네트워크 성능 측정 기능과 네트워크 장비를 감시하기 위한 장비 관련 정보 등이 부족하다는 점, 그리고 네트워크 장비를 reboot 하거나 shutdown 시킬 수 있는 기능이 없다는 점을 들 수 있다. 그 부족한 점을 네트워크 관리 표준인 SNMP 기반으로 관리하여 해결할 수 있는 방안을 제시했으며 이를 바탕으로 SNMP 기반 관리의 요구사항을 에이전트와 매니저 입장에서 분석하였다. 또한, 분석된 요구사항을 바탕으로 에이전트와 매니저로 구성된 관리 시스템을 설계했으며 관리 대상에 대한 관리 정보를 SNMP MIB으로 정의하였다.

제시한 시스템을 토대로 SNMP 기반의 전투 체계 관리 시스템의 프로토타입을 개발하여 해상전투체계 관리를 위해 선정된 5대의 실제 사용되는 장비들로 구축된 Test Bed에서 테스트를 수행함으로써 네트워크 성능 관리 정보의 부재와 reboot 등의 제어 action의 부재에 대한 기존 전투체계의 부족한 점을 SNMP 기반의 관리를 통해 해결할 수 있음을 보였다.

향후 과제로 다음과 같은 연구가 계속 진행 되어야 한다. 먼저, 향후 5대 장비 이외의 다른 장비에도 SNMP 에이전트를 탑재하여 SNMP 매니저가 모든 관리 대상 장비에 대해 SNMP 기반 관리를 수행하도록 한다. 다음으로, 현재는 주기적인 모니터링을 통한 장비의 알람이 발생하도록 되어 있다. 향후에는 SNMP 에이전트가 탑재된 장비가 문제가 발생하면 SNMP 매니저에게 notify하는 trap방식이 추가되어야 한다. 또한, 전체 시스템을 SNMP 기반으로 관리하기 위해서는 기존 proprietary 방법으로 관리되던 장비에 대해서도 SNMP 기반으로 통합 관리할 수 있는 방안이 마련되어야 한다. 마지막으로 구현한 시스템이 SNMP를 통하여 관리 기능을 수행함에 있어서 적은 네트워크 부하를 발생하는지 여부를 실험을 통하여 검증할 필요가 있다.

참 고 문 헌

- [1] William Stallings, "SNMP, SNMPv2, SNMPv3 and RMON 1 and 2", Third Edition, Addison-Wesley, 1999.
- [2] The Internet Engineering Task Force(IETF), "IETF Home Page", <http://www.ietf.org>, Refer Apr. 2008.
- [3] K. McCloghrie, M. Rose, "Management Information Base for Network Management of TCP/IP-based internets : MIB-II", IETF, RFC1213, March 1991.
- [4] S. Waldbusser, P. Grillo, "Host Resources MIB", IETF, RFC 2790, March 2000.
- [5] M. Rose, K. McCloghrie, "Structure and Identification of Management Information for TCP/IP-based Internets", IETF, RFC 1155, May 1990.
- [6] Wind River Corps, "VxWorks", <http://www.windriver.com/vxworks/>, Refer Apr. 2008.
- [7] AdventNet Inc, "AdventNet API", <http://snmp.adventnet.com/help/snmpapi/snmpv1/>, Refer Apr. 2008.