韓國컴퓨터情報學會 論文誌
第15卷 第8號, 2010. 8.

2010-15-8-3-2

가상 네트워크 망으로부터 문제점 식별 및 진단에 관한 연구

김 정 수*

A Study on Problem Identification and Diagnosis from Virtual Network

Jeong-Su Kim*

요 약

ISP 사업자는 NMS을 이용하여 자사망에 대한 원활한 서비스를 목적으로 지속적인 모니터링을 하고 있다. NMS는 ISP 사업자의 네트워크 망을 구축한 이후, 이 NMS로 자사 네트워크 망에 대한 장애 이벤트를 분석하여 잠재적인 장애 원인에 대한 진단을 수행한다. 그러나 만약 ISP 사업자가 네트워크 망을 구축하기 이전 가상 네트워크 망모형 설계 및 네트워크 망에 대한 문제점 식별과 진단이 가능한 소프트웨어가 있다면 어떨까라는 의문의 동기로 본 연구를 시작하게 되었다. 따라서 본 논문에서는 가상의 네트워크 망 모형을 통한 장애 원인 및 진단이 가능한 소프트웨어 기반의 시뮬레이션 툴인 NetDoctor를 소개한다. 이 NetDoctor를 이용하여 가상의 네트워크 망을 구성한 후 고의적으로 장애를 발생시키고 그 장애에 대한 식별이 가능한지를 분석하였다. 실험 결과, NetDoctor는 가상 망모형에 대한 문제점 식별과 진단이 가능한 것을 알 수 있었다. 국내 NetDoctor를 이용한 연구 사례가 미흡한 점을 가만해 볼 때 본 연구를 통하여 가상 망모형에 대한 문제점 식별과 진단의 기초 연구가 되길 바란다.

Abstract

Various services such as IPTV, VoIP, multimedia over IP, on-line payment, on-line game, etc. were made possible due to the rapid advance of the network. In order to provide secure and seamless services over the network, the Internet service providers are performing continuous network monitoring using NMS. The main function of NMS is to perform a diagnosis to identify the potential causes of failure from event messages. In this paper, a simulation tool, named as NetDoctor, is presented which is capable of identifying and diagnosing the potential problems in the virtual network, before the network model is constructed. In NetDoctor, a series of various and artificial failure is imposed on the virtual network, and it was analyzed if NetDoctor could identify the problems. The experimental results on virtual network show that the developed tool is very effective in identifying and diagnosing the problems. The presented simulation tool can be used in the design of robust network.

▸ Keyword : Virtual Network, Problem Identification, Diagnosis

---

• 제1저자 : 김정수
• 투고일 : 2010. 05. 30, 심사일 : 2010. 06. 15, 게재확정일 : 2010. 06. 17.
* 광운대학교 경영정보학과 경영정보학박사

Ⅰ. 서 론

최근 유무선 통신 네트워크의 빠른 진화로 IPTV, VoIP, 영상 통화, 온라인 결재, 온라인 게임 등 다양한 서비스를 앤드 유저가 유무선 단말기로 제공받을 수 있는 편한 세상이 되었다. 이러한 서비스를 위하여 ISP(Internet Service Provider) 사업자는 NMS(Network Management System)을 이용하여 네트워크 망에 장애 발생 시 알람, 경고, 측정된 매트릭 등과 같은 운영 이벤트 데이터를 분석하여 잠재적인 장애 원인에 대한 진단을 수행한다. 만약 ISP 사업자의 NMS가 존재함에도 불구하고 인터넷 상의 진단 실패가 발생할 경우 복구하기 위한 많은 시간 소비와 진단 원인을 규명하기란 결코 쉬운 작업이 아닐 것이다. 그 이유로 거대한 네트워크 망은 수많은 관리자 도메인으로 구성되었으며 이러한 도메인을 통해 다양한 네트워크 장비들로 구축되었기 때문이다. 일반적으로 ISP 사업자는 네트워크 망을 구축한 이후 NMS로 자사 네트워크 망을 수시적으로 감시한다. 그러나 만약 ISP 사업자의 네트워크 망을 구축하기 이전 가상 네트워크 망모형을 설계할 수 있고 설계된 네트워크 망모형에 대한 네트워크 망의 문제점 식별과 진단이 가능한 소프트웨어가 있다면 어떨까? 만약 이러한 소프트웨어를 ISP 사업자가 보유하고 있다면 네트워크 망 구축 중 구성 실패를 줄일 수 있고 고가의 네트워크 장비에 대한 효율적인 배치로 시설 투자비용을 절감할 수 있을 것으로 판단된다. 따라서 본 논문은 가상의 네트워크 망모형을 통한 장애 원인 및 진단이 가능한 소프트웨어 기반인 시뮬레이션 툴을 소개한다. 이 시뮬레이션 툴을 이용하여 가상의 네트워크 망을 구성하고 고의적으로 장애를 발생 시킨 후 장애에 대한 원인이 파악할 수 있는지를 분석하였다. 본 연구의 기여점은 국내 NetDoctor를 이용한 연구 사례가 미흡한 점을 가만해 볼 때 본 연구를 통하여 거대한 네트워크 망 구성에 대한 에러 원인 추적과 진단이 가능하다는 점을 실험을 통하여 증명되었다.

본 논문의 구성은 다음과 같다. 2장 관련연구에서는 크게 세가지로 분류하여 정리하였다. 첫째, 과거 문헌을 기반으로 네트워크 장애 진단에 대한 사례연구 분석, 둘째, 가상의 네트워크 망에 대한 모델링과 시뮬레이션이 가능한 실험 툴 조사, 끝으로 채택한 시뮬레이션 툴에 대한 기능을 간략히 소개하였다. 3장은 NetDoctor 정의와 워크플로우, NetDoctor에서 제공되는 다양한 룰(Rule) 패턴, 룰 수행 프로세스를 기술하였다. 4장에서는 NetDoctor를 이용한 가상의 네트워크 장애 식별과 진단을 실험하였다. 5장은 NetDoctor 실험에 따른 실험 요약과 마지막 6장에서 결론으로 본 연구를 종결(終結)한다.

Ⅱ. 관련연구

2.1 네트워크 장애 진단

현재 NMS에서 직면하고 있는 주요 도전은 NMS로 장애 측정이 가능해야 한다. NMS는 미해결된 이벤트를 캐시에서 관리되어야 하고 캐시 내의 루트 이벤트로 구성되어야 한다. 이벤트의 의존성을 억제하기 위하여 모든 캐시 이벤트에 대한 입력 이벤트와 연관된 룰-기반 메커니즘을 사용하고 있다. 장애에 의해 수집된 이벤트는 장애 노드와 관련된 토폴로지로 노드의 일정한 크기의 서브셋[2]으로 매우 작게 생성되어진다. 따라서 NMS상의 각 이벤트 도착은 이벤트와 연관된 선형 복잡성 분야, 캐시 내의 이벤트에 대한 일정한 크기의 서브셋으로 분리되어 점차 작아진다. 직관적으로 선형 복잡성은 암호와 활용된 네트워크 토폴로지 데이터 구조를 요구한다. 이와 같은 연구를 위하여 META(Monitoring network Events with Topology Assistance) 프레임워크로 구성하였고 스케일러블한 네트워크 장애 진단을 실행하기 위한 토폴로지 이벤트 패턴 방식을 이용한 첫 번째 제안된 연구이다[1]. 이와 같은 META 주요 아키텍처는 그림 1과 같다.

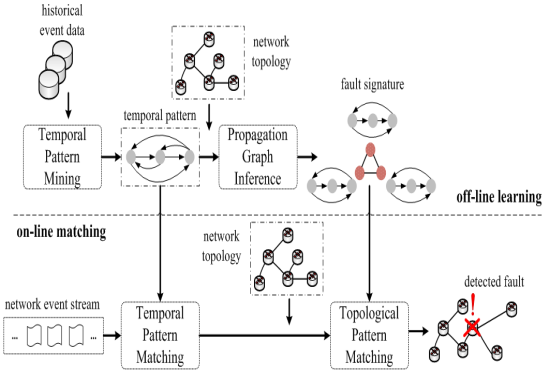

그림 1. META 주요 아키텍처 설명
Fig 1. Illustration of The Main Architecture of META

1) 이러한 서브셋 크기는 분포 정도에 의존함. 네트워크는 네트워크 크기에 독립적이어야 함

표 1. 네트워크 문제점 식별과 진단에 대한 관련연구
Table. 1 A Literature for Network Problem Identification and Diagnosis

연구자	연구 내용
Ting Wang et al. [2009]	• 연구 동기: 장애가 발생했을 때 노드 상의 토폴로지 관계뿐만 아니라 네트워크 장애에 의해 생성된 이벤트를 평가하기 위하여 인덱스 장애 표시의 새로운 클래스를 소개 • 연구목적: 새로운 시공간 인덱스 구조와 노이즈 히스토리 이벤트 데이터로부터 장애 신호를 추출하기 위한 효율적인 학습 알고리즘을 제안 • 기여점: 사회와 정보 네트워크를 포함한 다른 많은 유형에서도 네트워크 중요성에 대한 잠재적인 애플리케이션을 지적하고 본 연구 방법의 효력을 탐색하기 위하여 광범위한 실험 연구 결과를 제공한 것임
Chun Yuan et al. [2006]	• 연구 동기: 인간 중재에 의존한 전형적인 검사 방법으로 문제를 해결하기 위하여 부정확한 결과 산출과 비효율적인 프로세스로 구성됨. 이러한 원인이 사용자 불만족에 대한 중요한 요인으로 제공함 • 연구 목적: 구조적으로 존재하는 애매모한 텍스트 설명 대신에 문제를 해결하기 위한 상호 관계를 구축한 시스템 이벤트 추적과 같은 시스템 행동 정보사용을 제안 • 기여점: 루트 원인 인식에 따른 정확성을 고려한 분석이 가능함
Atsuo TACHIBANA et al. [2006]	• 연구 동기: 인터넷 상의 충돌은 종종 발생하며 특히 충돌 패스 상의 영역 위치는 네트워크 성능 감소 또는 ISP 사업자에게 치명적인 결과를 초래함 • 연구 목적: 본 연구는 네트워크 단층 접근을 이용한 다중 목적지로부터 적합한 패스 상의 단방향 종단간 패킷 손실 측정에 의해 충돌 위치 식별에 대한 구조적인 방법을 제안함. 따라서 일본 상용 인터넷 상의 다중 패스로 오랜 기간 실험을 통하여 측정된 패킷 손실을 알 수 있도록 제공함 • 기여점: 실험 결과 제안한 방법은 충돌 세그먼트 위치를 정확하게 파악할 수 있었음
Anukool Lakhina et al. [2004]	• 연구 동기: 이상 현상을 위한 진단은 네트워크 운영자와 엔드 유저 모두에게 크리티컬함. 노이즈 데이터, 하이-디멘젼의 거대한 양에 대한 이상 현상 패턴 해석이 가능해야 하며 이것을 추출할 수 있어야 하기 때문에 이상 현상 진단이 어려움 • 연구 목적: 따라서 본 연구는 이상 현상을 진단하기 위한 일반적인 방법을 제안함. 이 방법은 이상적인 네트워크 제어와 일반적인 하부 공간을 분리한 네트워크 트래픽 측정 집합으로 하이-디멘젼 공간 분리를 기본으로 함. 이러한 분리로 기본적인 컴포넌트 분석을 효율적인 수행이 가능토록 제공함 • 기여점: 링크로부터 간단한 트래픽 측정으로 이상 현상 연구와 방법을 제시함 - 네트워크 트래픽 이상 현상을 감지하기 위한 일반적인 접근과 "이상 현상 볼륨" 분리하기 위한 간단한 링크 트래픽 통계 방법의 애플리케이션 제안 - 두 개의 서로 다른 백본 네트워크 상에 수집된 실제 데이터를 사용하여 이 방법을 검증
David A. Maltz et al. [2004]	• 연구 동기: 네트워크 라우팅 설계는 프로토콜 구성을 표현한 것임. 라우팅 설계는 본래부터 어렵고 네트워크 설계 작업시 매우 중요한 부분으로 차지함. 이와 같은 작업의 상황을 향상시키기 위한 방법이 무엇일까라는 동기로부터 본 연구를 시작하는 계기가 됨 • 연구 목적: 운영 네트워크 내에 사용된 라우팅 프로토콜이 무엇인지를 세부적인 실험으로 표현. 내부/외부 게이트웨이 프로토콜의 운송 모델을 제시하고 구조상에 사용된 메커니즘의 다양한 셋을 기술하기엔 불충분함. 따라서 본 연구는 라우팅 설계를 리버스 엔진 기반의 white-box 접근법으로 강점과 약점을 토론하고 네트워크 행동과 설계를 새롭게 이해할 수 있도록 패스를 개방함 • 기여점: 중/대형 네트워크(예: 서비스 제공자의 백본 네트워크, 글로벌 기업 네트워크, 기업 등)로부터 31개의 네트워크 영역으로 분류하여 8,035개 구성 파일을 세부적으로 분석함. 이와 같은 작업의 시도는 이전 연구에는 존재하지 않으며 스케일 역시 다른 연구와 비교할 수가 없는 점이 본 연구에 기여점임
Ratul Mahajan et al. [2003]	• 연구 동기: ISP 사업자의 장애 컴포넌트 매핑은 ISP 사업자 역할임. 앤드 유저는 장애가 발생할 경우 ISP 사업자는 앤드 유저에게 빠른 대응이 가능토록 제공해야 함 • 연구 목적: 사용자 레벨 인터넷 패스 진단을 위한 아키텍처와 현재 인터넷 내의 진단 패스 관련에 대한 구조적인 툴을 소개 - 전자는 모든 라우터 상의 패스를 통해 수집된 패킷을 추적하여 분석 - 후자는 제안한 진단 툴인 Tulip은 진단 재요구, 손실, 인터넷 패스를 통한 큐잉으로 운영 중인 네트워크 문제점을 진단 • 기여점: 제안한 진단 툴인 Tulip으로 인터넷 내의 잠재적인 문제점에 대한 불충분한 진단을 효율적으로 치료할 수 있음. 그러나 Tulip 진단 툴은 라우터 상의 라운드 트립 측정 기반의 구조적인 단점으로 포워드와 리버스 패스에 대한 분석은 아직 미흡한 실정임

과거 많은 문헌에서는 네트워크 장애식별과 진단에 대한 중요성을 강조했으며 네트워크 망에서의 이상 현상에 대한 빠른 인지(認知)로 ISP 사업자는 장애 식별이 가능해야 한다 [1][2][3][4][5][6]. 이에 대한 대표적인 연구 문헌은 〈표 1〉과 같다.

네트워크상의 이상 현상은 네트워크 트래픽 레벨 내의 변화된 신호와 사용되지 않은 트래픽을 말한다. 이러한 네트워크 내의 트래픽 이상 현상에 대한 이해는 매우 중요한 문제이다.

이상 현상은 악의성 있는 장애 또는 악의성이 아닌 장애 두 가지로 분류되며 이상 현상 종류로 DoS(Denial of Service)공격, 잘못된 라우터 구성, BGP(Border Gateway Protocol) 정책 변경 결과 등을 들 수 있다. Lakhina는 트래픽 특성에 대

한 많은 문헌이 있음에도 불구하고 트래픽 이상 현상을 가볍게 이해하고 있다고 주장했다. 그 이유로 첫째, 이상 현상 식별은 정교한 모니터링 구조를 요구한다. 그러나 불행히도 대부분의 ISP 사업자는 간단한 트래픽 측정으로 데이터를 수집(예: SNMP(Simple Network Management Protocol))을 이용한 평균 트래픽)한다. 보다 진보된 ISP 사업자는 에지 링크 상의 플로우 카운터를 수집하여 분석하는 실정이다. 둘째, 트래픽 이상 현상에 대한 이해 결핍성 때문이다. 즉, 실시간 이상 현상을 감지하기 위하여 충분히 빠른 프로세싱 측정 툴을 보유하고 있지 않다. 그러므로 ISP 사업자는 장애가 발생한 후 주요 장애 원인(예: 웜 바이러스, DoS 공격)을 알아차린다. 이 또한 장애 원인을 추적하는 동안 장애 감지를 할 수 없다는 점이 일반적이다. 끝으로 네트워크 트래픽 성격은 하이-디멘젼과 노이즈로 이것을 트래픽 통계로부터 이상 현상에 대한 의미 있는 정보를 추출하기란 쉽지 않기 때문이다[4].

조사된 문헌처럼 네트워크 장애는 ISP 사업자뿐만 아니라 앤드 유저에게도 치명적인 결과를 초래한다. ISP 사업자는 네트워크 내의 트래픽 이상 현상을 좀 더 신중히 이해해야 할 것이며 네트워크 장애에 대한 빠른 진단으로 앤드 유저에게 끊김없는 양질의 서비스를 제공해야 할 것이다. 아울러 이와 같은 서비스 제공을 위하여 ISP 사업자는 실시간 이상 현상을 감지할 수 있는 측정 툴을 보유하여 네트워크 장애에 대한 빠른 진단이 가능해야 할 것이다.

2.2 실험툴

거대한 네트워크 망에 대한 모델링과 시뮬레이션이 가능한 툴로 NS2[1)]와 OPNET, NCTUns[2)], NetDAS[3)] 등이 존재한다[7][8][9]. 이와 같은 시뮬레이션 중 세계적으로 많은 네트워크 연구자들이 연구 목적으로 사용되는 시뮬레이션 툴로 NS2와 OPNET를 들 수 있으며 대표적인 장단점은 〈표 2〉와 같다.

표 2. 시뮬레이션 장단점
Table. 2 Simulation's Strengths and Weaknesses

제품명	장점	단점
NS2	• Open Source • TCP의 시뮬레이션을 위해 라우팅, 멀티캐스트 프로토콜, 유무선 네트워킹 등 많은 기능을 제공	• 가상으로 네트워크 환경 구성이 어려움 • 다양한 네트워크 장비 컴포넌트 제공의 한계 • 트래픽의 다양한 분포적 특성을 반영하기 어렵고 제한된 분포의 트래픽만 생성할 수 있음 • 드롭된 대상의 패킷 덤프와 패킷 에러 플래그 표시 중 하나만 링크 레벨단의 에러 또는 패킷 손실을 에러 모델에서 지원하나 링크, 노드, 보안, 트래픽 등 다양한 유형의 네트워크 장애에 대한 진단 식별이 어려움
OPNET	• 상업용 소스 • GUI 구성이 쉬움 • 다양한 모듈(들)과 표준네트워크 장비컴포넌트를 제공	• 고가의 소프트웨어
NCTUns	• 상업용 소스 • GUI 구성이 쉬움	• VoIP를 직접적으로 지원하지 않으며 PSTN(Public Switched Telephone Network)과의 연동도 가능하지 않음 • 네트워크 장애에 대한 진단 식별이 어려움
NetDAS	• 국내 대학에서 자바로 개발한 시뮬레이션 툴로 확장성 문제에 관한 한 구현상의 제한없음 • 역동적인 트래픽 흐름, 라우팅 등의 전송 메커니즘과 내부 동작을 가시화하여 보여줌 • 시뮬레이션시 VoIP 게이트웨이와 게이트키퍼 장비 등을 분석할 수 있음	• 시뮬레이션 수행 시간이 오래 걸림 • 네트워크 장애에 대한 진단 식별이 어려움

1) NS(Network Simulator)2
- TCL의 확장된 스크립트 언어로 쓰여진 NS2 모듈을 이용하여 시뮬레이션 수행
- TCP 기반에서 FTP나 텔넷 트래픽을 생성하며 UDP의 CBR에서 지수나 파레토 트래픽을 생성

2) NCTUns(the National Chial Tung University network simulator)
- 커널 안의 실제 TCP/IP 프로토콜 스택을 사용하여 시뮬레이션 결과를 생성
- NCTUns는 시뮬레이션에서 TCPdump 프로그램의 트래픽 트레이스를 사용한 방식으로 시뮬레이션을 수행
- 시뮬레이션 수행 중 노드 상의 애플리케이션 프로그램을 런칭하거나 노드 인터페이스를 통한 패킷 캡처를 위해 TCPdump를 실행할 수 있음

3) NetDAS(Network Design and Analysis System)
- 직관적인 인터페이스와 간단한 설정 과정으로 네트워크를 설계할 수 있으며 시뮬레이션 수행 후, 그 결과를 분석할 수 있음
- NetDAS는 사용자의 요구 수준을 반영하는 수학적인 확률 분포로 트래픽을 생성하여 시뮬레이션을 수행
- 설계된 네트워크는 실제 장비와 동일한 방식으로 동작하게 되며 시뮬레이션 과정을 애니메이션으로 가시화하여 시각적인 효과를 높이고 수행 결과 보고서 제공함으로써 분석이 용이

위의 〈표 2〉처럼 소프트웨어 기반의 여러 시뮬레이션 툴이 존재하지만 대부분 다양한 유형의 네트워크 장애 및 진단 식별이 불가능한 반면 OPNET은 이러한 네트워크 장애 및 진단 식별이 가능한 것으로 조사되었다. 따라서 실제 망에서 앤드 유저가 체감한 패킷을 Import 가능한 가상 네트워크 환경 구성, 네트워크 모델링을 보다 편리하게 구성 가능한 GUI환경, 마지막으로 최적화된 가상네트워크 환경에 대한 장애식별이 가능한 OPNET사의 Modeler(버전 14.5)로 선정하였다.

Modeler는 네트워크 설계자 또는 QoS(Quality of Service) 전문가들이 애용(愛用)하는 상용 제품으로 연구 목적인 대학교 및 연구소에서 사용되며 국방부의 차세대 정보통신망 최적화 설계 분야에서도 이를 이용하였다[10]. 뿐만 아니라 기업을 위한 네트워크 컨설팅 업체에서도 주로 사용하고 있다. 아울러 Modeler는 여러 개의 모듈로 구성되어 있고 각 모듈마다 기능이 다르므로 연구자는 필요한 모듈만 가지고 사용하면 된다. 보다 구체적인 Modeler의 특징은 아래 〈표 3〉와 같이 요약할 수 있다.

표 3. Modeler 특성 요약
Table. 3 Summary of Modeler Characteristics

툴명	특성
OPNET 사의 Modeler	• 지능적인 네트워크 관리 소프트웨어(예: 네트워크 장비, 네트워크 프로토콜, 애플리케이션, 서버 운용에 대한 전문적인 지식을 수집) • 종단간 애플리케이션 성능 문제 및 네트워크 문제를 진단하고 해결책 제시(네트워크 오류 사전 파악) • 운용 전 새로운 애플리케이션을 분석 • 성능을 저하시키는 운영상의 구성 문제를 파악 • 네트워크와 응용프로그램에 가장 적합한 서버를 시뮬레이션을 통하여 제공 • 장애처리, 운영상의 분석, 엔지니어링, 설계&계획 등이 가능 • 모듈: ACE, ACE Decode Module, Flow Analysis, NetDoctor, Wireless, etc.

2.3 시뮬레이션

실제 종단간 네트워크 환경을 구성하기엔 ISP 사업자가 아닌 이상 불가능하다. 그 이유로 실제 종단간 네트워크 환경을 구성하기 위해서는 수많은 구간의 네트워크 장비와 링크 등을 구성해야 하기 때문에 많은 투자비용이 예상되기 때문에 본 연구는 가상 네트워크 망을 구성하여 해당 네트워크 장비 구성을 고의적으로 장애를 발생시킨 후 이에 대한 문제점 식별 및 진단 가능여부를 확인하기 위하여 시뮬레이션을 도입하였다. 시뮬레이션이란 실세계의 문제점을 컴퓨터 내에서 가상으로 모형화하는 것을 의미한다. 시뮬레이션 측정 툴은 시스템의 현재 그리고 과거 행동의 특성을 가지고 있다. 그러나 이것은 서로 다른 워크로드 또는 서로 다른 하드웨어와 소프트웨어 구성으로 미래 행동을 예측할 수 있어야 한다. 이러한 이유로 시스템에 대한 분석적 Queueing 모델을 개발한다. 시스템의 복잡성, 인터프리터 결과와 속도, 시스템 오버 시간 또는 실세계 프로세스로부터 동작 한계점 때문에 시뮬레이션을 선택한다[11]. 즉, 시스템, 엔티티들 사이의 심벌 관계, 로직, 수학(Mathematical) 등을 가정한 집합으로 개발된 것을 말한다. 이러한 시뮬레이션 연구영역이 보다 다양화되고 복잡하게 발점됨에 따라, 인공지능의 지능형 에이전트기법을 도입하여 해결하는 연구가 활성화 되고 있는 추세이다[12]. 부가적으로 시뮬레이션을 통해 개발된 소프트웨어 또는 측정 장비로부터 측정된 측정결과값을 시뮬레이션의 분석 결과값과 검증하기 위한 방법으로도 사용된다. 그 이유는 시뮬레이션에서 사용된 장비 컴포넌트는 각 통신제조사의 표준 스팩을 준수하여 개발된 모듈이고 분석된 결과값 또한 검증된 수식 알고리즘으로 산출된 결과값이기 때문이다.

이미 앞 절에서 언급한 OPNET사의 Modeler내의 모듈인 NetDoctor로 가상망으로부터 장애 식별 및 진단에 대한 시뮬레이션을 수행하였다. NetDoctor 특성은 아래와 같이 요약할 수 있다.

- Import된 네트워크 모델 검증
- 네트워크 감사(監査)
- 구성된 정책에 대한 디바이스 구성파일 감사
- 포트 스캔 분석으로 네트워크 보안 감사
- 네트워크 보안 표준 준수
- 네트워크 문제가 발생된 위치 공지
- 문제점 식별

시뮬레이션 유형의 다른 종류로 DES(Discrete Event Simulation)와 Hybrid Simulation 은 이전 많은 문헌에 소개되었으므로 구체적인 언급은 지면관계상 생략하기로 한다[13]. 본 연구는 다음과 같은 시뮬레이션을 수행하였다. 첫째, 종단간 네트워크 토폴로지 형상을 위해 가상 네트워크 라우터를 만들고 각 노드까지 링크를 연결하여 구성한다. 둘째, 가상의 종단간 네트워크 토폴로지 형상으로부터 고의적인 장애를 발생시킨 후 이에 대한 문제점 식별과 진단이 가능한지를 확인하였다.

III. 가상 네트워크 모형

3.1 NetDoctor

NetDoctor는 구성된 네트워크 환경에 대한 네트워크 성능 분석, 네트워크 구성 재검증, 다양한 룰4) 기반 엔진을 적용할 수 있을 뿐만 아니라 네트워크상의 보안, 성능, 유용성 영향을 줄 수 있는 잠재적인 문제점을 추출하기 위하여 NetDoctor를 사용한다. 포괄적인 룰들은 규칙 룰을 만들거나 또는 제공된 룰을 사용할 수 있다. NetDoctor는 구성 환경내의 비용 문제점이 발생하기 이전 구성된 네트워크 내의 에러를 올바르게 잡을 수 있다. NetDoctor의 워크플로우는 그림 2와 같다.

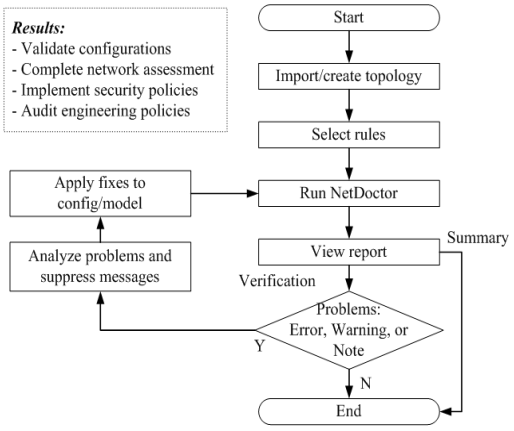

그림 2. NetDoctor 워크플로우
Fig 2. NetDoctor Workflow

네트워크 진단자가 네트워크 구성 장비에 대한 Import/create 토폴로지를 생성한다. 가상 망구성의 모형이 완료되었다면 Configure/Run NetDoctor에서 룰을 선택한다. 룰 구성 선택 완료 후, NetDoctor를 실행한다. NetDoctor 실행이 끝나면 NetDoctor 보고서가 생성되고 생성된 NetDoctor 보고서에 에러가 발생하지 않으면 종료된다. 만약 생성된 NetDoctor에 에러가 발생되면 문제점을 분석한 후 가상망으로부터 구성/모델을 다시 재설정한다. 구성/모델에 대한 재설정이 완료되면 다시 NetDoctor를 실행하고 생성된 NetDoctor 보고서의 에러가 없어진 것을 확인한 후 종료한다.

3.2 Rule Pattern

룰은 오픈 소스인 Python으로 작성되었다. 따라서 네트워크 진단자가 룰을 직접 만들 수도 있고 커스트마이징된 룰을 사용할 수 있다. 룰은 데이터 사용과 네트워크 오브젝트로부터 구성된 데이터를 수집한다. 구성된 네트워크 요소 또는 구성된 네트워크 정책이 올바르게 구성되어야만 한다. 룰은 검증 룰과 요약 룰로 분리할 수 있다. 첫째, 검증 룰은 네트워크 내의 기술된 구성 정보를 전달해 준다. 검증 룰의 한 예로 중복된 IP 주소를 설정했다면 네트워크 모델 내의 임의의 중복 IP 주소 설정이라는 정보를 알려준다. 둘째, 요약 룰은 네트워크 내의 일반적인 정보를 전달해 준다. 요약 룰의 한 예로 OSPF(Open Shortest Path First) 영역으로 네트워크 모델 내의 모든 OSPF 영역에 대한 정보를 알려준다. 검증 룰과 요약 룰에 대한 카탈로그로 일반 유형(예: 장치, 링크, 디멘드, 패스 등)과 IP(예: IP 인터페이스, 라우터, IP 서브넷 피어 등)로 분류된다. 일반적으로 검증 룰은 NetDoctor 보고서에 에러, 경고, 주석, 패스 등을 생성해 주며 요약 룰은 파이 차트와 바 그래프와 같은 이미지를 제공해 준다. 이와 같은 룰에 대한 설명을 룰 카탈로그로 분류할 수 있으며 이를 도식화한 것이 아래 그림 3과 같다.

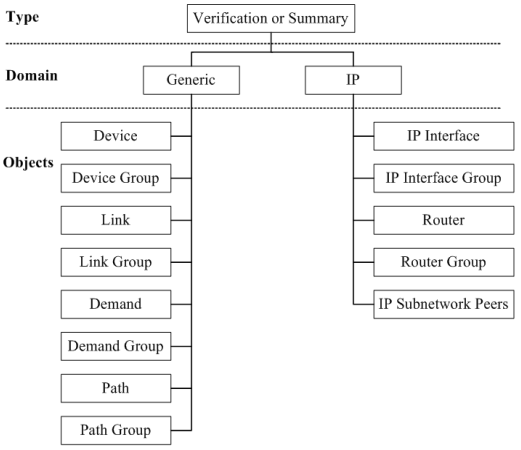

그림 3. 룰 카탈로그
Fig 3. Rule Categories

룰에 대한 분류는 크게 45개로 분류되며 세부적인 기능은 656개로 분류된다. 〈표 4〉은 45개 룰에 대한 세부 룰은 656개의 방대한 분량으로 인하여 대표 룰 정의로만 한정하여 기술

4) Rule이란 구성데이터를 실행한다는 의미임. 이러한 구성데이터는 제작된 네트워크로부터 Import된 디바이스 구성들의 집합을 말함

하였다[14].

표 4. NetDoctor Rule Suites
Table. 4 NetDoctor Rule Suites

룰 종류	상세 기술
AAA	Cisco IOS(originally Internetwork Operating System) 수행 장치를 활성화 해줌. 이 장치는 모든 가상 라인과 인터페이스에 대한 디폴트 인증 방식으로 자동 인식 제공
ATM	각 DTL 경로 구성이 올바르게 설정됐는지를 체크
Administration	네트워크 내에 장치들이 그 장치 내에 있는 운영체제의 기술된 버전으로 실행했는지를 체크
BGP	근접한 EBGP가 EBGP-multihop 비활성화로 구성됐는지를 체크
DLSw+	두 개의 라우터 사이의 구성에 대한 피어 관계 순서를 각 라우터로 반드시 기술됐는지를 체크
EIGRP[5]	다중 EIGRP 라우팅 도메인을 체크
EIGRP (Advanced)	EIGRP 룰로부터 잘못된 매트릭 계수가 있는지를 체크
Firewalls	ACL(Access Control List) 레퍼런스 상에 정의되지 않은 오브젝트 그룹을 체크
HAIPE	보안 정책으로 모든 트래픽을 정상적으로 저지하는지를 체크
HSRP	인터페이스 트랙킹을 선점없이 활성화됐는지를 체크
HTTP	인터페이스 트랙킹과 선제공격은 그룹 내의 라우터에 일관되게 세팅되었는지를 체크
IGRP	주요 네트워크 접촉 상태가 정상적인지를 체크
IP Address	IP 주소 인터페이스가 올바르게 설정되지 않았을 경우 체크
IP Multicast	IGMP[6] 접근 그룹에 정의되지 않은 ACL이 있는지를 체크
IP Routing	EIGRP, IGRP, OSPF, Peer Interface 등에 대한 IP 경로가 일치되지 않은 경우 이를 체크
IPSec	암호화된 모든 트래픽이 라우터로 보내졌는지를 체크
IPX	논리적으로 연결된 인터페이스가 올바르게 설정되었는지를 체크
ISIS	인터페이스 패스워드가 잘못 대칭되었는지를 체크
Kerberos	서버가 네트워크 내의 모든 장치들로 구성되었는지를 체크
Link Aggregation	정의되지 않은 채널 그룹 내의 라우터 인터페이스가 있는지를 체크
MPLS	LSP[7]에 사용된 패스가 진입 라우터 상에 올바른 패스인지를 체크
MPLS VPNs	라우터상의 VRF[8] 인터페이스가 정의되어 있지 않은 맵이 있는지를 체크
NAT	같은 ACL로 인터페이스 상에 다중 해석되어 사용했는지를 체크
NTP	장치의 모든 활동 IP 인터페이스가 비활성화로 설정되어있지 않은지를 체크
OSPF	ABR(Area Border Router)이 백본 영역에 연결되었는지를 체크
OSPF (Advanced)	백본 영역에 연결되지 않을 경우 라우팅 문제 발생소지가 있으므로 이를 체크
Organizational Policies	장치 구성 파일이 정상적인지를 체크
Policy-Based Routing	장치상의 정책과 관련된 매칭 그룹이 해당 장치 내에 올바르게 정의되었는지를 체크
QoS	인터페이스 상에 적용된 QoS 프로파일이 라우터에 정상적으로 정의되었는지를 체크
RADIUS	장치상의 올바르지 않은 근원지 인터페이스를 체크
RIP	• 주요 네트워크가 접촉 상태가 정상적인지를 체크 • IP 통신망의 경로 지정에 대하여 정상적인지를 체크
RSRB	피어(Peer) 정의가 대칭적이지 않은 것을 체크
Route Maps and ACLs	호스트 비트가 고정 필터 구성으로 설정되었는지를 체크
SNMP	정의되지 않은 ACL을 체크
Security	언블록된 포트의 인터페이스를 체크
Services	Bootp 서버가 활성화가 됐는지를 체크
Spanning Tree	일치하지 않은 포트 비용 체크
Static Routing	작성된 경로가 구성된 디폴트 네트워크상의 라우터로 구성되었는지를 체크
System Logging	장치로부터 시스템 로깅 서버의 최대 수 이상으로 로깅 메시지가 시스템에 보내졌는지를 체크
TACACS+	서버 상의 경로가 설정되어 있지 않은 것이 있는지를 체크
Tunnel Interfaces	GRE[9] 터널 키가 잘못 매칭된 것을 추출
VLANs	트렁크 포트 사이의 캡슐화가 잘못 매칭된 것을 추출
VRRP	가상 라우터 그룹 내의 라우터 통지 간격이 일정한지를 체크
Voice over IP	네트워크 내의 모든 보이스 게이트웨이에 대한 구성을 체크
WLAN	중복 접근 포인터 지역이 있는지를 체크

5) EIGRP(Enhanced Interior Gateway Routing Protocol): 미국 Cisco System사가 개발한 자율 시스템 내의 경로 제어 통신 규약
6) IGMP(Internet Group Management Protocol): IP 멀티캐스트를 실현하기 위한 통신 규약
7) LSP(Lable Switched Path): MPLS는 IP 패킷의 플로우를 MPLS 네트워크상에 미리 정해진 경로를 따라 전달하는 역할을 담당함. 이 경로를 LSP라 부름
8) VRF(Virtual Routing Forwarding): 각각의 VPN 라우팅 테이블
9) GRE(Generic Routing Encapsulation): 네트워크를 통한 보안 터널을 형성하기 위해 IP 위에 암호 루틴을 삽입해 만들어지는 암호화 과정

3.3 Rule Execution Process

이 장은 앞 절의 룰을 어떤 방식으로 어떻게 수행하는지에 대한 룰 실행 프로세스를 조사해 보기로 한다. 룰 실행 프로세스는 오브젝트 종류에 대한 작성된 룰로 실행 프로세스 기술은 크게 세가지로 분류된다.

- 단일 오브젝트 룰
- 오브젝트 그룹 룰
- IP 서브네트워크 피어 룰

단일 오브젝트 룰

단일 오브젝트 동작 룰은 IP 인터페이스, 일반적인 디바이스, 테스트 기능과 필터 기능 사용 등을 말한다. 단일 오브젝트 룰의 몇몇 예로 BGP-잘못된 근접 주소로 BGP-근접 주소가 로컬 인터페이스 주소인 것, OSPF-ABR 백본 지역에 연결되지 않은 것을 들 수 있다. 아래 그림 4는 단일 오브젝트 룰 수행 프로세스 수행을 도식화한 것이다.

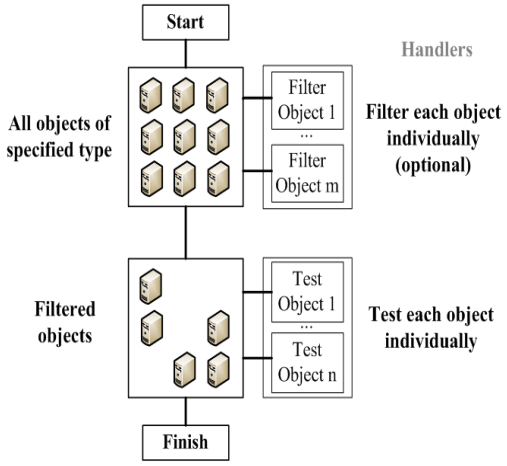

그림 4. 단일 오브젝트 룰 수행 프로세스
Fig 4. Single Object Rule Exception Process

분석 엔진은 각 오브젝트 시간에 대한 필터 기능을 포함하고 네트워크 내의 기술된 종류의 모든 오브젝트로 주어진다. 필터 기능은 오브젝트 허용 또는 거절을 위하여 분석 엔진에게 질의한다. 만약 룰이 필터 기능을 가지고 있지 않는다면 모든 오브젝트로 처리된 분석 엔진을 허용하기 때문에 이러한 이유로 필터 기능을 가진 것이다. 또한 분석 엔진은 각 오브젝트를 위한 테스트 기능을 호출하면 테스트 핸들러에 의해 개별적인 오브젝트를 테스트하고 종료한다.

오브젝트 그룹 룰

오브젝트 그룹 룰은 오브젝트 그룹(예: IP 인터페이스 그룹, 디바이스 그룹)을 동작시키는 것이다. 오브젝트 그룹 룰은 테스트 기능과 필터 기능을 추가하여 그룹 기능을 요구한다. 즉, 그룹 기능은 어떤 오브젝트가 그룹 내에 포함된 것인지를 기술한 것이다. 예를 들면 오브젝트 그룹 룰은 OSPF-중복 라우터 ID, EIGRP-불일치 활동 타이머 등을 들 수 있다. 아래 그림 5는 오브젝트 그룹 룰의 수행 프로세스 과정을 보여준 것이다.

분석 엔진은 각 오브젝트에 대한 필터 기능을 호출하고 네트워크상에 기술된 종류의 모든 오브젝트를 가져온다. 필터링 프로세스는 앞 그림 4처럼 단일 오브젝트 룰과 동일하다. 필터 오브젝트 집합은 그룹 기능을 통과하여 그룹핑 가능 그룹에 대한 모든 오브젝트와 관련시킨다. 그룹핑 가능 그룹은 오브젝트 주요 쌍 매핑으로 만들어지며 각 값은 관련된 오브젝트 리스트이고 대응된 키는 그룹에 유일하게 식별할 수 있다. 이러한 매핑은 분석 엔진 뒷단에서 통과된다. 모든 오브젝트 그룹 룰은 그룹핑 기능을 요구하며 이러한 그룹핑 기능이 단일 그룹 내의 전체 오브젝트로 형성된다. 분석 엔진을 통해 각 그룹에 대한 테스트 기능을 호출하고 전체 그룹을 테스트한 후 종료한다.

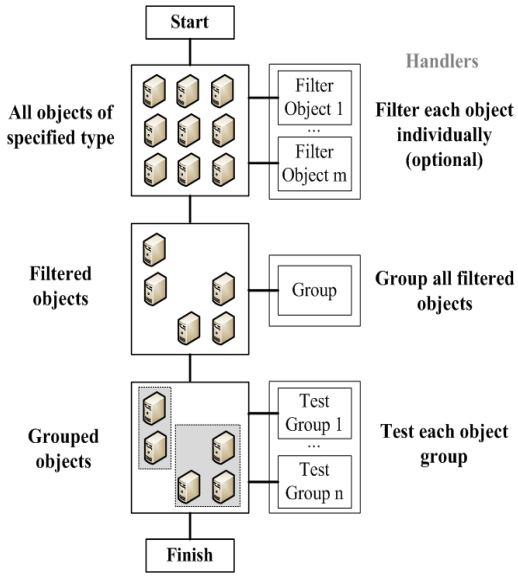

그림 5. 오브젝트 그룹 룰 수행 프로세스
Fig 5. Object Group Rule Exception Process

IP 서브네트워크 피어 룰

IP 서브네트워크 피어 룰은 오브젝트 그룹 룰의 특별한 종류로 분석 엔진이 내부적인 그룹핑으로 실행된다. IP 서브네트워크 피어 룰은 단지 IP 인터페이스만 동작한다. 예를 들면 IP 라우팅-불일치 매트릭 컴포넌트, IP 라우팅-불일치 라우팅 프로토콜 등을 들 수 있다. 아래 그림 6은 IP 서브네트워크 피어 룰의 수행 프로세스 과정을 설명한 것이다.

분석 엔진은 오브젝트 그룹 룰과 동일하며 필터링 프로세스는 단일 오브젝트 룰과 같다. 필터링 된 IP 인터페이스의 자동화된 분석 엔진은 물리적인 연결과 IP 서브넷 주소를 기반으로 하며 그룹핑 기능으로 분할하지는 않는다. IP 서브네트워크 피어의 각 그룹은 테스트 기능을 통과한 후, 테스트 기능에서 리포트 상의 문제점과 전체 그룹을 테스트하고 종료한다. 테스트 기능은 오브젝트 그룹 룰과 같지 않고 IP 서브네트워크 피어 룰로 그룹 또는 그룹을 식별하기 위한 유일한 키도 갖지 않는 차이점이 있다.

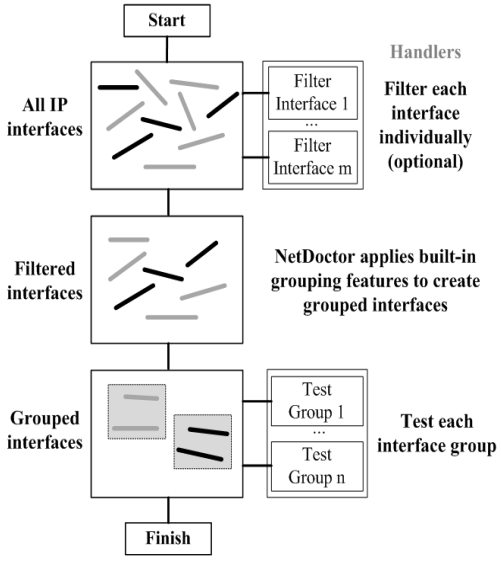

그림 6. IP 서브네트워크 피어 룰 수행 프로세스
Fig 6. IP Subnetwork Peers Rule Execution Process

참고로 그림 4~6내에 있는 핸들러(Handler)는 분석 엔진을 의미하며 테스트 핸들러, 그룹 핸들러, 필터 핸들러, 파라미터 핸들러, 프롤로그 핸들러, 에필로그 핸들러로 분류되며 네트워크 진단자 또는 룰 개발자가 Python으로 핸들러를 작성하여 이를 사용한다.

NetDoctor는 이와 같이 세가지 유형으로 룰 실행 프로세스가 처리되어 가상 네트워크 망에 대한 문제점 식별 및 진단을 수행한다.

Ⅳ. 가상 네트워크상의 문제점 식별 및 진단

NetDoctor 문제점 식별 및 진단을 위하여 두가지 실험 유형으로 가상 망모형을 설계했다. 첫 번째 실험인 IP Addressing은 별첨에 있는 그림 7 왼쪽과 같다. 맨 마지막 노드에 고의적으로 IP Addressing을 잘못 기입 후 실험하였다. 두 번째 실험은 IP Routing으로 별첨에 있는 그림 7 오른쪽과 같다. 즉, Kwangju-Core/Busan-Core에 고의적으로 IP Routing을 잘못 기입 후 실험을 수행하였다. 이와 같이 IP Addressing과 IP Routing에 대한 고의적으로 잘못 구성한 지점은 별첨에 있는 그림 7에 점선으로 표시된 부분이다.

실험 수행 이전 이와 같은 가상 망모형을 만들기 위하여 다음과 같은 순서로 구성해야 한다. 첫째, 망을 설계하기 위한 방법으로 라우터에 대한 cfg 확장 파일을 Import하는 방법과 OPNET에서 제공된 표준 장비를 새로 생성하여 구성하는 방법 중 전자를 선택했다. 즉, Seoul-Core/Dajun-Edge/Kwangju-Core/Busan-Core를 각각 cfg(MRTG에서 수집된 값을 cfg라 함) 확장 파일로 구성한다. 라우터 샘플 cfg 내용은 아래 〈표 5〉과 같다.

표 5. 라우터 샘플 cfg 파일
Table. 5 Router Sample cfg File

```
1: !
2: version 12.1
3: service timestamps debug uptime
4: service timestamps log uptime
5: no service password-encryption
6: !
7: hostname Seoul-Core
..... 중간 생략 .....
8: interface Loopback0
9:   ip address 192.168.0.117 255.255.255.255
10: !
11: interface Serial2/2
12:  description Link to Daejun-Edge
13:  bandwidth 8192
14:  ip address 192.168.0.57 255.255.255.252
15:  ip route-cache flow
16: !
..... 중간 생략 .....
17: line vty 0 4
18:  password 0.cisco
19:  login
20: !
21: end
```

둘째, 각 cfg 확장 파일이 구성되면 Import from Device Configuration으로 cfg 확장 파일을 불러온다. 셋째, cfg 확장 파일 로딩이 끝나면 Core/Edge 라우터간의 링크가 자동 설정되어진다. 만약 링크가 연결되지 않을 경우 Import Assistant로 연결할 수 있다. 끝으로, 링크 설정 완료 후 Core/Edge 라우터간의 단말 장치 추가가 완료되면 Core/Edge 라우터간 링크 설정을 수작업으로 연결해 주면 된다. 참고로 단말 장치는 ethernet_firewall, server(s), mobile_server, mobile_client, pc, pc group 등으로 구성하였다.

이와 같은 순서로 구성이 완료되면 별첨에 있는 그림 7과 같은 종단간 망모형으로 구성된다. 망모형을 구성한 후 IP Addressing과 IP Routing 두가지 실험에 대한 NetDoctor로 문제점 식별과 진단이 가능한지를 확인한다. 실험을 위하여 45개의 룰 구성 중 IP Addressing과 IP Routing을 활성화한 후 시뮬레이션 분석 결과, 서버 IP Addressing과 라우터 IP Routing에 대한 잘못 지정된 위치를 정확히 추적 가능하였다. 이러한 정보는 NetDoctor 보고서에 기술되었으며 기술된 NetDoctor 분석 결과는 아래와 같다.

- Case 1. IP Addressing
  - B_client_group, B_client_group_0, K_client _group과 Kwangju-Core 사이의 IP중복입력
  - DB Server와 ethernet_firewall간 IP중복입력
  - Web Server와 ethernet_firewall간 IP중복입력
- Case 2. IP Routing
  - Busan-Core와 Kwangju-Core 사이의 IP Routing 잘못 구성

NetDoctor 보고서에서 지정한 에러를 가상 망구성 상에 올바르게 재설정한 후, 시뮬레이션 수행 결과 4개 에러는 모두 올바르게 설정되었다고 NetDoctor 보고서로 분석되었다. 이에 대한 시뮬레이션 분석 결과보고서로 별첨에 있는 그림 8과 같다. IP Addressing과 IP Routing의 고의적인 구성에 대한 시뮬레이션 진단 결과는 별첨에 있는 그림 8의 상단과 같으며 IP Addressing과 IP Routing에 대한 에러 지점을 다시 재구성한 후, 시뮬레이션을 수행한 결과는 별첨에 있는 그림 8의 하단과 같다.

V. 실험 요약

NetDoctor는 네트워크 구성이 올바르게 되었는지 재검토가 가능하며 만약 링크/노드간 에러가 발생했을 경우 에러를 감소하기 위하여 네트워크를 재구성한 다음 이전 보고서 결과와 재구성한 네트워크 보고서를 비교한 후 어느 정도 에러가 감소했는지를 비교 분석할 수 있다. 가상의 종단간 망구성이 아닌 실제 망구성에서 이와 같은 에러가 발생했다면 네트워크 대혼란이 발생했을 것이다. 그러나 실제 망구성을 NetDoctor로 가상 망구성으로 설계한 후 분석한다면 사전에 미리 에러를 막을 수 있어 네트워크 대혼란을 저지할 수 있을 것으로 판단된다. 이처럼 네트워크상의 이상 현상에 대한 장애 식별 및 진단은 중요하다. 만약 가상망이 아닌 실제 ISP 사업자 망에서 장애가 발생했다면 ISP 사업자 및 앤드 유저에게 크리티컬한 치명적인 결과를 초래했을 것이다.

NetDoctor 실험을 위하여 Core/Edge 라우터를 구성한 후 종단간에 단말 장치와 링크를 연결하여 구성하였다. 이렇게 가상으로 라우터/서버 IP Addressing과 IP Routing을 잘못 구성한 후 NetDoctor로 문제점 식별과 진단이 가능한지를 관찰하였다. 실험 결과, 고의적으로 잘못 구성한 라우터/서버 IP Addressing과 IP Routing에 대한 잘못된 영역의 위치를 정확히 추적할 수 있었다. 실험에 사용했던 45개의 룰 구성 중 IP Addressing과 IP Routing으로 망구성에 대한 에러 위치를 정확히 발견할 수 있었던 것처럼 그 이외의 룰 구성으로 보다 다양한 망구성에 대한 에러를 발견할 수 있다는 의미가 내포되어 있음을 알 수 있었다.

본 실험을 통하여 국내 NetDoctor를 이용한 연구 사례가 미흡한 점을 가만해 볼 때 가상의 망구성에 대한 에러 원인 추적과 진단이 가능하다는 점을 파악할 수 있었다.

VI. 결론

본 논문은 만약 ISP 사업자가 네트워크 망을 구축하기 이전 가상 네트워크 망모형 설계 및 네트워크 망에 대한 문제점 식별과 진단이 가능한 소프트웨어가 있다면 어떨까라는 의문의 동기로부터 시작된 것이다. 따라서 가상 네트워크 망모형 설계 및 네트워크 망에 대한 문제점 식별과 진단이 가능한 소프트웨어인 NetDoctor를 소개하고 이를 이용한 실험을 수행하였다. 실험결과, 고의적으로 잘못 구성한 라우터/서버 IP Addressing과 IP Routing에 대한 문제점 식별을 정확히 파악할 수 있었다.

본 연구 진행 중 한계는 다음과 같이 요약할 수 있다. ISP 사업자는 네트워크 망에 대한 정보를 여러 가지 이유로 네트워크 망을 개방하지 않는다. 만약 ISP 사업자의 네트워크 망에 대한 구성된 디바이스 구성에 대한 정보를 알 수 있다면 실제 망 운영에 대한 진단이 가능했을 것으로 판단된다. 이와

같은 이유로 본 연구는 가상으로 디바이스를 구성한 후 구성된 가상의 망모형으로 실험을 수행한 것이다.

향후 연구는 다음과 같다. 첫째, NetDoctor의 룰에 대한 보다 세부적인 실험이 필요할 것이다. 본 연구에서는 NetDoctor에서 제공된 45개의 룰 중 IP Addressing과 IP Routing 두 개의 룰을 사용했지만 네트워크 진단자가 새로운 룰을 만들어 가상 망구성에 그 룰을 적용한 후 가상망에 대한 진단 여부를 적용한다. 이와 같은 연구는 국제 표준 기관의 관련 표준 문서를 이해한 후 Modeler에 제공되지 않는 룰을 새롭게 생성해야하기 때문에 중요한 연구가 될 것이다. 둘째, ISP 서비스 상품 중 와이브로를 이용한 멀티미디어 데이터, 온라인 결제 등과 같은 서비스를 고객 관점의 실제 체감 서비스 품질을 측정한 후 그 패킷을 Modeler에 Import한다. Import된 가상 망모형을 기반으로 다양한 Background Traffic을 발생시킨 후 이때 수용 가능한 사용자 수에 대한 트래픽 예측 연구가 필요할 것이다.

참고문헌

〔1〕 T. Wang, M. Srivatsa, D. Agrawal, L. Liu, "Learning, Indexing, and Diagnosing Network Faults," Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2009.

〔2〕 C. Yuan, N. Lao, J. R. Wen, J. Li, Z. Zhang, Y. M. Wang, W. Y. Ma, "Automated Known Problem Diagnosis with Event Traces," In EuroSys, 2006.

〔3〕 A. Tachibana, S. Ano, T. Hasegawa, M. Tsuru, Y. Oie, "Locating Congested Segments over the Internet Based on Multiple End-to-end Path Measurements," IEICE TRANS. COMMUN., Vol. E89-B, No. 4, April 2006.

〔4〕 A. Lakhina, M. Crovella, C. Diot, "Diagnosing Network-Wide Traffic Anomalies," Proceedings of the 2004 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, 2004.

〔5〕 D. A. Maltz, G. Xie, J. Zhan, H. Zhang et al, "Routing Design in Operational Networks: A Look from the Inside," Proceedings of the 2004 Conference on Applications, Technologies, Architectures and Protocols for Computer Communications, 2004.

〔6〕 R. Mahaian, N. Spring, D. Wetherall, T. Anderson, "User-level Internet Path Diagnosis," In ACM SOSP, 2003.

〔7〕 이선명, "네트워크 시뮬레이션 시스템에서의 트래픽 생성기," 석사학위논문, 창원대학교, 2003.

〔8〕 최재원, 전의수, 이현주, 김정애, 이선명, 김현주, 김태수, 이광휘, "네트워크 분석을 위한 시뮬레이션 시스템 설계," 2002년도 한국통신학회 하계종합학술발표회, Vol. 25, 2002년 7월.

〔9〕 최재원, 이광휘, "VoIP 시뮬레이션을 지원하는 네트워크 설계 및 분석 도구의 구현," 전자공학회논문지, Vol. 42, No. 1, 2005년 1월

〔10〕 한국전자통신연구원, "차세대 국방정보통신망 최적화 설계 연구," 국방부 프로젝트, 2004.

〔11〕 J. Dilley, R. Friedrich, T. Jin, J. Rolia, "Measurement Tools and Modeling Techniques for Evaluating Web Server Performance," Lecture Notes in Computer Science, pp. 155-168, 1996.

〔12〕 우종우, 김대령, "멀티 에이전트 기반의 지능형 시뮬레이션 도구의 개발," 한국컴퓨터정보학회 논문지, Vol. 12, No. 6, pp. 21-30, 2007년 12월.

〔13〕 김정수, "전자상거래 시스템의 서비스 품질 측정과 예측에 관한 연구," 박사학위논문, 광운대학교, 2005.

〔14〕 OPNET, "NetDoctor," *Modeler Product Documentation* Release 14.5.

저 자 소 개

김 정 수

2005년 8월: 광운대학교 경영정보학과 (경영정보학박사)

2010년 7월: (사)한국IT비즈니스진흥협회 '10년도 표준기술력향상사업 지원

현재: 메타빌드(주) R&D연구원

관심분야: QoS, 시뮬레이션 분석, NMS etc.

〈별첨〉

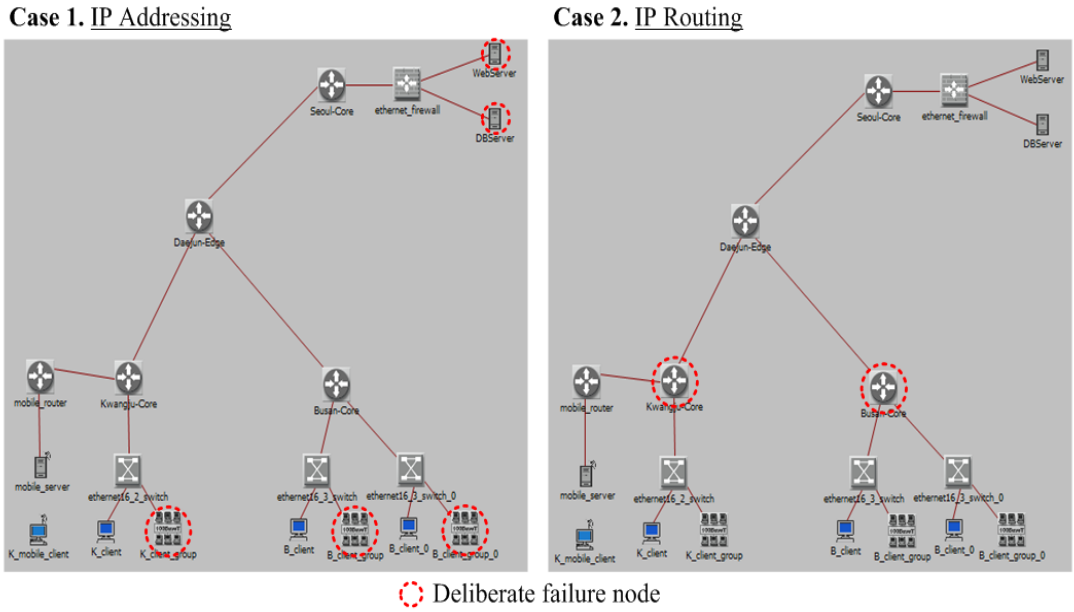

그림 7. 가상 네트워크 모형
Fig 7. Models for Virtual Network

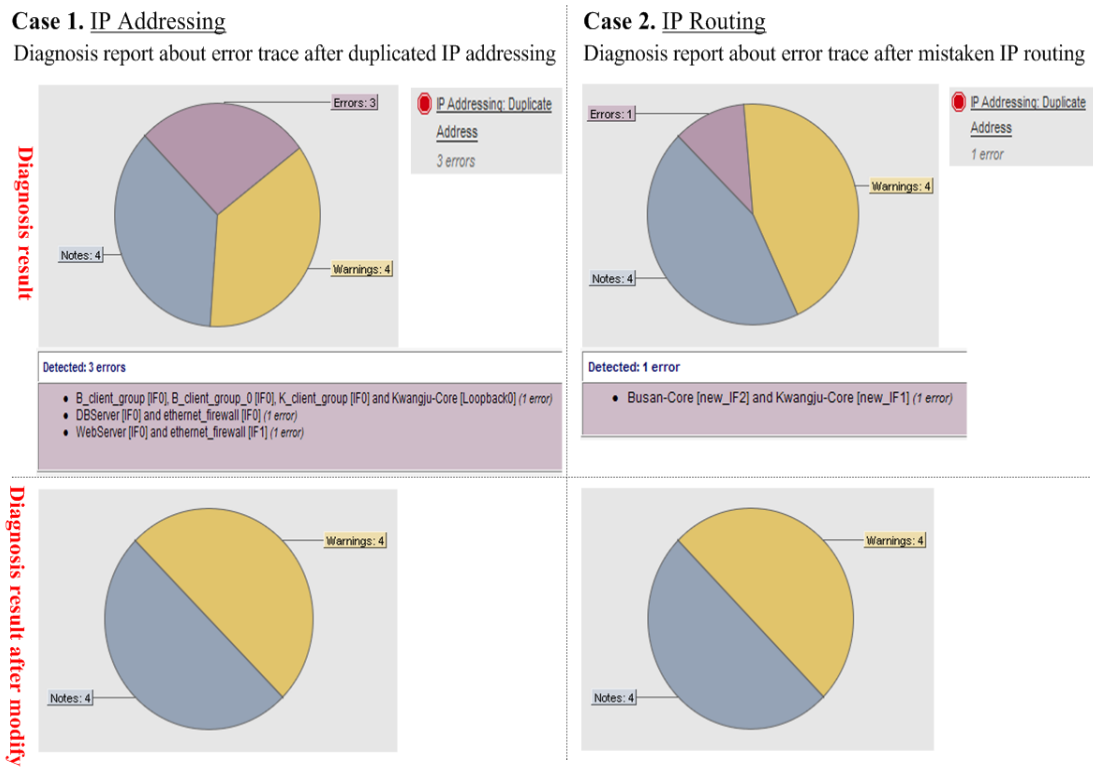

그림 8. NetDoctor 분석 결과
Fig 8. Analysis Result using NetDoctor