응용 레벨 트래픽 분류를 위한 시그니쳐 생성 시스템 및 검증 네트워크의 개발

박준상, 박진완, 윤성호, 오영석, 김명섭
고려대학교 컴퓨터정보학과
{runtoyou, pakjw84, hiption, 840105, tmskim}@korea.ac.kr

Development of signature Generation system and Verification Network for Application Level Traffic classification

Jun-Sang Park, Jin-Wan Park, Sung-Ho Yoon, Young-Seok Oh, Myung-Sup Kim
Dept. of Computer Information Science, Korea University

요 약

네트워크 트래픽 모니터링과 분석은 엔터프라이즈 네트워크의 효율적인 운영과 안정적 서비스를 제공하기 위한 필수적인 요소이다. 다양한 트래픽 분석 방법 중 시그니쳐 기반의 분석 방법은 가장 높은 분석률을 보이지만 모든 시그니쳐를 수작업으로 추출하기 때문에 응용프로그램의 변화와 출현에 유연하게 대응하지 못한다. 따라서 본 논문에서는 응용프로그램 시그니쳐 생성 과정의 단점을 보완할 수 있는 시그니쳐 자동 생성 시스템을 제안한다. 응용프로그램 시그니쳐는 페이로드 내의 고유한 바이트 시퀀스로 정의하며 응용프로그램이 발생시키는 모든 트래픽을 대상으로 추출한다. 또한 생성 시스템의 실효성을 증명할 수 있는 검증 시스템 및 검증 네트워크를 제시한다.

1. 서론

과거의 인터넷은 Well-Known Port 기반의 HTTP, Telnet, E-mail, FTP, NNTP 의 응용들이 대부분의 인터넷 트래픽을 차지하고 있었기 때문에 IANA[1]에 정의된 port 정보 기반의 분석으로 신뢰성이 높은 분석 결과를 도출할 수 있었다. 하지만 스트리밍 응용프로그램 및 passive FTP 와 같이 하나의 응용 프로그램이 둘 이상의 세션을 형성하고, 이들 중 데이터 세션의 port 가 동적으로 생성됨에 따라 port 기반의 분석은 더 이상 높은 신뢰성을 제공할 수 없게 되었다. 이를 보완하기 위한 방법으로 응용계층 프로토콜 내용을 참조하여 동적으로 생성되는 port 정보를 얻어내어 분석하는 방법인 Mmdump [2], SM-MON [3]에서 소개되었다. 그러나 이 방법은 RTSP, MMS, SIP 와 같이 응용 프로토콜이 공개되었거나 알려진 응용 트래픽의 분석에만 사용 가능하고 비공개 응용 프로토콜이 대다수 포함된 전체 인터넷 트래픽에 적용할 수 없다는 문제점이 있다. 이와 같은 문제점을 해결하기 위해 시그니쳐 기반 분석 방법[4]이 제시되었다. 이 방법은 시그니쳐가 확인된 응용에 대해서는 정확한 분석이 가능하다는 장점을 갖지만, 모든 응용별로 시그니쳐를 수작업으로 찾아야만 하고, 찾아진 시그니쳐가 응용프로그램의 변화에 유연하게 대처하지 못하는 문제점을 보인다. 또한 다양한 연구에서 90%이상의 높은 분석률을 주장하고 있지만 검증을 위한 테스트 데이터의 범위가 제한적이며, 가정에 기반한 검증 결과를 보이고 있어 실제 네트워크에 적용하였을 때 신뢰성을 보장하지 못하는 문제점을 보인다.

따라서 본 논문에서는 높은 정확성을 보장하는 응용프로그램별 시그니쳐를 생성하고, 시그니쳐의 생성 효율을 높여 응용프로그램의 지속적인 업그레이드에 대처하기 위한 시그니쳐 자동 생성 시스템을 제안하며, 생성 시스템의 실효성과 생성된 시그니쳐의 정확성을 증명할 수 있는 검증 시스템 및 검증네트워크의 구축 방법을 제안한다.

본 논문은 다음과 같은 순서로 구성된다. 2 장에서는 응용프로그램 시그니쳐 생성을 위한 고려사항에 대해 살펴본다. 3 장에서는 시그니쳐 생성 시스템에 대한 설명을 다룬다. 4 장에서는 검증 시스템과 검증 네트워크에 대해 설명하고 이를 바탕으로 시그니쳐 생성 시스템의 실효성을 증명한다. 5 장에서는 결론을 맺고 향후 연구에 대하여 기술한다.

2. 응용프로그램 시그니쳐 정의 및 고려사항

다양한 연구에서 시그니쳐 기반의 분류 방법을 제안하고 시그니쳐의 정확성을 주장하고 있지만 시그니쳐의 정의, 생성 범위, 생성 단위에 따라 시그니쳐 생성 방법, 분류 알고리즘, 분류 결과는 다른 형태로 나타난다. 분류 결과는 네트워크 관리자의 분류 목적에 따라 실효성이 달라지기 때문에 트래픽 분류 목적에

* 이 논문은 2007 년 정부(교육인적자원부)의 재원으로 한국학술진흥재단의 지원을 받아 수행된 연구임.(KRF-2007-331-D00387)

따라 생성 범위 및 단위를 분명하게 결정되어야 한다.

2.1 응용 프로그램 시그니쳐 정의

응용프로그램 시그니쳐란 응용프로그램 별로 그들만이 사용하는, 다른 응용들과 구분되는, 전체 응용프로그램의 트래픽으로부터 해당 응용프로그램을 분류할 수 있는 고유한 패턴으로 정의된다. 본 연구에서 응용프로그램 시그니쳐를 패킷의 페이로드로부터 추출하며 해당 응용프로그램의 플로우 내에서 변하지 않은 바이트의 시퀀스로 정의한다.

2.2 응용 프로그램 시그니쳐 고려사항

I. 시그니쳐의 생성 범위

응용프로그램에 의해서 발생되는 모든 트래픽을 대상으로 한다. 응용프로그램이 통합화되고 복잡해지면서 응용프로그램을 설치, 갱신하기 위한 트래픽과 주요 기능을 제공하기 위한 보조적인 트래픽의 양이 증가하고 있기 때문에 이에 대한 분석이 필수적으로 요구되고 있다.

II. Application vs. Protocol

본 논문에서는 프로세스를 단위로 분류 가능한 시그니쳐를 추출하여 1 차적으로 분류하며, 각 프로세스의 집합으로 구성되는 응용프로그램으로 2 차 분류한다. 응용 레벨 프로토콜을 기준으로 트래픽을 분류하는 경우 성격이 다른 다양한 응용프로그램이 하나의 프로토콜로 통합되어 분류되는 문제점이 발생한다. 이러한 트래픽 분류 결과는 서비스의 통합화, 응용 레벨 트래픽의 복잡성을 고려했을 때 네트워크 관리자에게 효과적인 정보를 제공하기 어렵다.

III.Packet vs. Stream

응용프로그램의 시그니쳐는 패킷을 기준으로 생성하는 방법과 플로우 전체의 스트림에서 생성하는 방법으로 나눌 수 있다. 본 연구에서는 응용 프로그램의 분류율과 분류시스템의 부하를 고려하여 패킷을 기준으로 시그니쳐를 생성하는 방법으로 응용프로그램 시그니쳐를 생성한다.

플로우의 스트림에서 생성된 시그니쳐에 기반한 분류 시스템은 트래픽의 분류를 위해 모든 패킷을 스트림 형태로 재조합하는 과정이 요구된다. 때문에 분류 시스템의 부가적인 프로세싱 과정과 저장 공간이 필요하다. 또한 패킷의 손실, 비대칭 라우팅으로 인해 플로우의 스트림이 완전하게 구성되지 못하면 분류가 불가능하게 되는 문제점이 발생한다.

IV.단방향 vs. 양방향

단 방향의 시그니쳐를 적용하는 분류 시스템에서 시그니쳐를 포함하고 있는 패킷이 손실되거나 비대칭 라우팅에 의한 패킷 전송이 발생하는 경우 미분류의 원인으로 작용할 수 있다. 따라서 분류율을 향상시키고 견고한 분류 시스템을 구축하기 위해서는 양 방향에서 시그니쳐를 생성해야 한다.

3. 응용프로그램 시그니쳐 생성 시스템

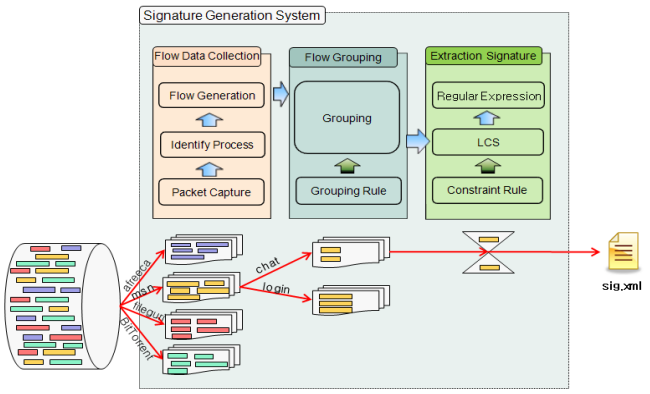
2 장에서 설명한 고려사항을 바탕으로 시그니쳐 자동 생성 시스템의 구성을 각 단계 별로 설명한다.

3.1 자동 생성 시스템의 정의

본 논문에서 제안하는 시그니쳐 생성 시스템은 시그니쳐 생성을 위한 데이터 수집과 추출로 구분된다.

기존의 데이터 수집 방법은 시그니쳐 생성을 위한 응용프로그램의 특정 기능을 독립적으로 반복 수행하여 데이터를 수집한다. 이와 같은 방법은 목적에 맞는 정확한 데이터 수집이 어렵기 때문에 시그니쳐의 신뢰도를 보장할 수 없다. 따라서 본 논문에서는 사용자의 컴퓨터에서 발생하는 모든 응용프로그램의 트래픽을 발생 시킨 프로세스를 기준으로 분류하고, 패킷의 페이로드를 포함한 플로우 단위로 수집한다.

시그니쳐를 추출하기 위해 네트워크 관리자는 수집되어진 데이터를 바탕으로 데이터를 의미적으로 판단하여 시그니쳐를 추출한다. 이러한 방법은 해당 응용프로그램에서 사용하는 응용 레벨 프로토콜에 대한 사전 지식을 요구하게 된다. 공개되지 않은 응용 레벨 프로토콜을 사용하는 응용프로그램의 수가 증가하고, 응용 프로그램의 변화가 잦은 추세를 고려했을 때 많은 시간이 소비되는 비효율적인 방법이다. 이러한 문제를 해결하기 위한 시그니쳐 추출 시스템은 LCS(longest Common Subsequence) 알고리즘을 기반으로 패킷의 페이로드로부터 동일한 바이트의 시퀀스를 추출한다. 그림 1 은 시그니쳐 생성 시스템의 모듈 구성과 패킷 수집부터 시그니쳐 생성까지의 일련의 과정을 보여주고 있다.



(그림 1) 시그니쳐 생성 시스템 구조

3.2 플로우 데이터 수집 시스템

플로우 데이터 수집 시스템의 입력 데이터는 시그니쳐를 생성하고자 하는 응용프로그램의 트래픽을 포함한 사용자의 컴퓨터에서 발생한 모든 트래픽이다. 트래픽은 플로우 형태로 저장되며 해당 플로우를 발생 시킨 프로세스 정보와 맵핑하여 프로세스를 기준으로 플로우 데이터가 저장된다. 플로우는 수 많은 패킷으로 구성되는데 모든 패킷을 대상으로 시그니쳐를 추출할 필요가 없는 것은 [5]에서 증명하였다. 조사하는 패킷의 개수를 결정하는 것은 LCS 기반의 시그니쳐 추출 시스템의 효율성에 직접적인 영향을 미치기 때문에 본 시스템은 아래와 같은 조건에 의해서 플로우 데이터를 제한적으로 수집한다.

- 페이로드가 존재하는 초기 10 개의 패킷
- FIN, RST 플래그를 포함하는 패킷의 발생
- 30 분 이상 지연된 플로우

3.3 플로우 그룹핑 시스템

응용프로그램의 트래픽은 다양한 동작 형태로 발생하기 때문에 플로우를 동작별로 그룹하는 과정을 거치지 않고 LCS 기반의 시그니쳐 추출 시스템의 입력으로 사용한다면 부정확한 시그니쳐가 생성되거나 시그니쳐가 생성되지 않을 수 있다. 따라서 플로우 그룹핑을 통해서 LCS 기반의 시그니쳐 추출 시스템의 정확한 입력데이터를 생성하는 시스템이 요구된다. 플로우 그룹핑 시스템은 전송계층 프로토콜을 기준으로 다른 그룹 규칙이 적용된다. TCP 는 In/Outbound, Web 포트 순으로 그룹되며, In/Outbound 트래픽으로 그룹 후 고정 포트의 유무에 따라 동적 포트는 플로우의 첫번째, 두번째 패킷의 크기를 통해 그룹된다. 고정 포트의 경우 포트를 통한 그룹 후 첫번째와 두번째 패킷의 크기를 통해 재그룹한다. 고정 포트는 특정 동작을 수행하는 단위로 사용되지만 하나의 포트를 통해 다양한 기능을 제공하는 응용프로그램이 존재하기 때문에 패킷 크기의 분포를 통한 세분화된 그룹이 요구된다. TCP 와 달리 UDP 는 대부분의 응용프로그램에서 특정 동작만을 수행하기 위해 발생되기 때문에 하나의 그룹으로 시그니쳐의 생성이 가능하다.

3.4 LCS 기반 시그니쳐 생성 시스템

LCS(Longest Common Subsequence) 문제를 해결하기 위한 알고리즘은 여러 가지가 존재한다. 그 중 Brute force 알고리즘은 $\Theta(n2^m)$ 시간복잡도와 추가적인 저장 공간을 요구하는 알고리즘으로 입력 데이터가 크고 복잡한 응용프로그램의 트래픽에서 LCS 를 추출하기에 부적합한 알고리즘이다. 따라서 본 연구에서는 LCS 문제를 해결하기 위해 추가적인 저장 공간만을 요하고 상수 시간에 문제를 해결할 수 있는 Dynamic program 방법을 선택하였다.

기본적인 LCS 알고리즘의 결과는 한가지의 경우로 추출된다. 하지만 실제 LCS 의 해답은 다양한 경우의 수로 나타날 수 있으며, 입력 스트링의 순서에 따라 그 결과가 다르게 나타난다. 따라서 LCS 로 추출될 수 있는 모든 경우의 수를 고려해야 정확한 응용프로그램 시그니쳐의 추출이 가능하다. 아래의 예는 이와 같은 문제점을 간단한 입력 데이터를 통해 보여 주고 있다.

$X = <A, B, C, B, D, A, B>$

$Y = <B, D, C, A, B, A>$

X 와 Y 의 LCS 결과는 $<B,C,B,A>$로 나타나며, Y 와 X 의 LCS 결과는 $<B,A,D,B>$로 나타난다. 이는 서로 다른 크기의 LCS 테이블을 생성하고 다른 경로를 통해 LCS 의 결과가 얻어지기 때문이다. 또한 X 와 Y 두개의 입력 스트링으로부터 추출 가능한 LCS 는 $<B,C,B,A>$, $<B,C,A,B>$, $<B,D,A,B>$으로 다양한 경우의 수로 나타난다. 이와 같은 문제를 해결하고 최적의 응용프로그램의 시그니쳐를 추출하기 위해서는 두 가지 단계의 추가적인 알고리즘이 요구된다.

첫째, 기대되는 모든 경우를 추출하여 최적의 시그니쳐를 선택하는 단계로 다음과 같은 규칙이 순서대로 적용된다.

I. 길이가 가장 긴 substring 이 존재하는 경우
II. Packet 의 페이로드 시작 부분을 포함하는 경우

둘째, 첫번째 단계에서 선택된 시그니쳐에서 시그니쳐로서 변별력을 갖지 못하는 substring 을 제거한다.

I. Substring 의 최소 길이는 2 바이트로 제한
II. Common substring 제거

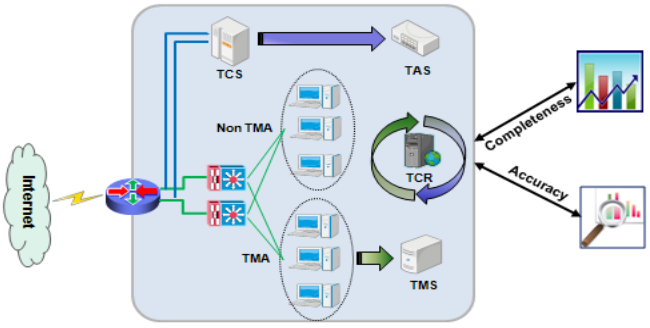
이러한 과정을 통해 추출된 시그니쳐는 표준화된 문자열 기술 방법인 정규 표현식으로 표현되어 시그니쳐 기반 분석 시스템의 입력데이터로 제공된다.

4. 검증방법 및 결과 분석

기존의 다양한 연구에서 트래픽 분류 방법론을 제안하고 분류 결과를 검증하고 있지만, 분류 기준이 불명확하고 가정에 기반한 트래이스를 대상으로 결과를 검증하고 있다. 이러한 검증 방법은 정확한 ground truth 를 제공하지 못하기 때문에 검증 결과의 신뢰성을 보장할 수 없다. 이러한 문제점을 해결하고 시그니쳐 생성 시스템의 타당성을 증명하기 위해 학내망에서 발생하는 모든 트래픽을 대상으로 실시간 검증 시스템 및 네트워크를 구축하였다.

4.1 검증 시스템 및 검증 네트워크 구축

실시간 검증 네트워크는 그림 2 와 같이 트래픽 수집 시스템(TCS), 트래픽 분류 시스템(TAS), 트래픽 측정 에이전트(TMA), 트래픽 측정 서버(TMS), 트래픽 분류 리포터(TCR)로 구성된다. 검증 시스템은 고속 링크의 대용량 트래픽을 실시간으로 처리하고, 다양한 트래픽 분류 방법의 동시 적용을 위해 분산 시스템 환경으로 구성된다. 따라서 다양한 분류 방법의 성능 비교와 분류 방법의 타당성을 증명할 수 있는 정보를 제공한다.



(그림 2) 검증 시스템 및 네트워크 구성

분류 결과의 정확성을 검증하기 위해 TMA(Traffic Measurement Agent)를 기반으로 Ground Truth 트래픽을 수집한다. TMA 는 학내망의 단말 호스트에 설치되며 소켓 정보를 기반으로 하여 Process name, IP, port, protocol, path 등의 정보를 생성한다. TMA 가 설치된 호스트에서 열려진 소켓을 주기적으로 검사하여 TMS 로 TMA 정보를 전송하고 TMS 는 각 호스트로부터 전달받은 TMA 정보를 통합하여 분류 시스템의 분류 결과의 ground truth 를 제공한다.

TCR 은 시그니쳐 기반 분류 시스템의 분류 결과와 TMS 정보를 비교하여 전체 트래픽, 응용프로그램 단

위 검증 결과를 Web 을 통해 실시간으로 제공한다. TCR 에서는 제공하는 정보는 표 1 과 같다.

<표 1> 분류 알고리즘 검증 요소

<table>
<tr><td>검증 항목</td><td>검증 시간</td><td>검증 단위</td><td colspan="6">검증 요소</td></tr>
<tr><td>Accuracy</td><td rowspan="2">min
hour
day</td><td rowspan="2">flow
byte
packet</td><td>TP</td><td>TN</td><td>FN</td><td>FP</td><td>Precision</td><td>Recall</td></tr>
<tr><td>Comple-teness</td><td colspan="6">Amount</td></tr>
</table>

4.2 트래픽 분류 결과

Coverage, Accuracy, Completeness 는 시그니쳐의 실효성 및 정확성을 평가하기 위한 Metric 으로 사용된다. 실험 결과는 2009 년 02 월 01 일 0 시부터 2009 년 02 월 07 일 24 시까지의 연속적인 트래이스를 바탕으로 결과를 보여주고 있다.

ⅰ. Coverage : 분류 가능한 응용의 개수

시그니쳐 생성 시스템을 이용하여 학내망에서 발생하는 응용프로그램을 대상으로 537 개의 시그니쳐를 추출하였다. 표 2 는 분류 가능한 응용의 수를 보인다.

<표 2> Coverage 결과

분류 단위	분류 대상 응용 개수
응용프로그램	81
공개된 프로토콜	13
프로세스	153

ⅱ. Completeness : 전체 트래픽 중 분류되어진 양

표 3 은 실험 기간 동안 학내망 전체에서 발생된 트래픽중 분류되어진 트래픽의 양을 나타내고 있다. 미 분류 트래픽은 시그니쳐의 높은 신뢰도를 고려했을 때 대부분 Coverage 에 포함되지 않는 응용프로그램에 의해서 발생되는 트래픽으로 분석되었다. Completeness 는 분류 대상 네트워크에서 발생되는 응용프로그램의 종류에 따라 다른 결과로 나타나게 된다. 트래픽 분류 시스템의 신뢰도를 보장하기 위해서는 일정 수준 이상의 Completeness 가 요구되며 이를 위해서는 Coverage 의 향상이 필수적임을 알 수 있다.

<표 3> Completeness 결과

	Flow	Packet	Byte
Total	41275724	3473298129	210TB
Completeness	86.06%	76.75%	75.72%

ⅲ. Accuracy : 분류의 정확도

아래의 표 4 는 시그니쳐의 정확성을 평가하는 Acccuracy 에 대한 측정 결과를 나타낸다.

<표 4> Accuracy 측정 결과

Accuracy Range	Flow	Packet	Byte
Classification	96.63%	98.35%	98.34%
Answer	97.63%	91.80%	91.24%

Accuracy 의 측정 결과는 그 측정 범위에 따라 다른 의미를 갖는다. Classification 범위는 시그니쳐에 의해서 분류된 트래픽으로 제한하며 오 분류되는 트래픽의 비율을 나타낸다. Answer 범위는 Coverage 외의 응용프로그램의 트래픽을 포함한 결과로 Classification 범위보다 낮은 Accuracy 를 보인다. 이는 오 분류와 미 분류되는 트래픽의 비율을 나타낸다. 오 분류의 발생 원인은 특정 응용프로그램의 시그니쳐에 의해서 Internet Explorer 의 트래픽을 분류하는 것으로 분석되었다. 이는 특정 응용프로그램에 종속되어 부가적으로 발생하는 Internet Explorer 의 트래픽과의 충돌이 발생하기 때문이다. 미 분류의 원인은 아래의 세가지 원인으로 분석된다.

- 패킷의 페이로드가 존재하지 않는 트래픽
- 시그니쳐를 추출할 수 없는 트래픽
- Coverage 에 포함되지 않은 응용프로그램의 트래픽

5. 결론 및 향후 과제

본 논문에서는 시그니쳐 생성의 효율을 높이고 응용프로그램의 지속적인 업그레이드에 대처하기 위한 시그니쳐 자동생성 시스템을 개발하였다. 또한 생성된 시그니쳐를 바탕으로 인터넷 트래픽을 분류하는 프로토타입 시스템의 개발하였으며, 이를 바탕으로 분석 결과의 정확성을 보장 할 수 있는 검증 시스템 및 검증 네트워크를 구축하여 응용프로그램 시그니쳐 자동 생성 시스템의 실효성을 증명하였다.

응용프로그램의 변화와 출현에 대해 네트워크 관리자가 수동적으로 인식하고 대처하기에는 많은 인력과 시간이 요구된다. 이러한 문제점을 해결하기 위해 검증 네트워크에 데이터 수집, 시스니쳐 추출, 시그니쳐 갱신 시스템을 통합하고 시그니쳐를 데이터베이스화 하여 주기적으로 시그니쳐의 갱신이 이루어지는 시스템을 구축할 계획이다.

참고문헌

[1] IANA port number list, IANA, http://www.iana.org/assignments/port-numbers.

[2] Jacobus van der Merwe, Ramon Caceres, Yang-hua Chu, and Cormac Sreenan "mmdump- A Tool for Monitoring Internet Multimedia Traffic," ACM Computer Communication Review, 30(4), October 2000.

[3] Hun-Jeong Kang, Myung-Sup Kim, and James Won-Ki Hong, "Streaming Media and Multimedia Conferencing Traffic Analysis Using Payload Examination," ETRI Journal, Vol.26, No.3, Jun. 2004, pp.203-217.

[4] TS Choi, CH Kim, SH Yoon, JS Park, HS Chung, BJ Lee, HH Kim, and TS Jeong, "Rate-based Internet Accounting System Using Application-aware Traffic Measurement," Proc. of 2003 Asia-Pacific Network Operations and Management Symposium (APNOMS 2003), Fukuoka, Japan, October 1-3, 2003, pp.404-415.

[5] Byung-Chul Park, Young J. Won, Myung-Sup Kim, James W. Hong, "Towards Automated Application Signature Generation for Traffic Identification," Proc. of the IEEE/IFIP Network Operations and Management Symposium (NOMS) 2008, Salvador, Bahia, Brazil, Apri. 7-11, 2008, 160-167