

**[솔루션 기능 테스트]**  
**Pintrus X-Traffic**  
**기능 테스트 절차서**

**한국소프트웨어진흥원**  
**공개SW기술지원센터**

## <Revision 정보>

일자	VERSION	변경내역	작성자
2007.11. 20	0.1	초기 작성	이규경

# 목 차

1. 문서 개요 .....	4
가. 문서의 목적 .....	4
나. 본 문서의 사용방법 .....	4
2. 테스트 절차 내역 .....	5
가. Pintrus X-Traffic 설치 .....	5
나. Pintrus X-Traffic 기능 테스트 .....	14
 <그림 차례>	
그림 1 트래픽사용량 분석1 .....	15
그림 2 트래픽사용량 분석2 .....	15
그림 3 트래픽사용량 분석(MRTG) .....	16
그림 4 인터페이스에 대한 회선 사용량 데이터 조회 .....	16
그림 5 인터페이스에 일/주/월/년 그래프 조회 .....	17
그림 6 네트워크 구간 조회1 .....	17
그림 7 네트워크 구간 조회2 .....	18
그림 8 네트워크 구간 조회3 .....	18
그림 9 BPS TOP N 사용량 1 .....	20
그림 10 PPS TOP N 사용량 1 .....	21
그림 11 BPS TOP N 사용량 2 .....	21
그림 12 PPS TOP N 사용량 2 .....	22
그림 13 트래픽 구간 사용 검색 .....	22
그림 14 트래픽 1시간 간격 데이터 검색 .....	23
그림 15 트래픽 1일 간격 데이터 검색 .....	23

## 1. 문서 개요

본 문서는 Flow 기반의 TMS 솔루션인 Pintrus X-Traffic을 Asianux3.0 OS (kernel 2.6.18)에서 호환성 및 기능성 검증을 중심으로 테스트 하였으며, 관련 솔루션 업체의 참고자료 활용을 위해 제작되었다.

### 가. 문서의 목적

다음과 같은 세부적인 목적을 달성하기 위하여 작성되었다.

- Flow 기반의 TMS 솔루션인 Pintrus X-Traffic과 Asianux3.0 OS (kernel 2.6.18) 기능성 테스트 절차.
- 기타 리눅스 OS(CentOS 5.0, Redhat5.0) 간의 정합성 테스트 수행의 절차서로 사용.
- 진행 중 문제 발생 사항과 각각의 진행사항 기술.
- X-Traffic 솔루션 도입을 검토하는 업체의 참고자료로 제공.
- 공개SW 지원 솔루션 확보 확대.

### 나. 본 문서의 사용방법

다음과 같은 방법으로 사용할 수 있다.

- Asianux3.0 OS (kernel 2.6.18) 기반하에 X-Traffic의 설치 절차 및 결과를 확인한다.
- Asianux3.0 OS (kernel 2.6.18) 기반하에 X-Traffic의 중요 기능 확인 절차 및 결과를 확인한다.
- 기타 리눅스 OS(CentOS 5.0, Redhat5.0)간의 정합성 테스트 절차서로 사용한다.
- X-Traffic 도입 검토시 참고 자료로 사용한다.

## 2. 테스트 절차 내역

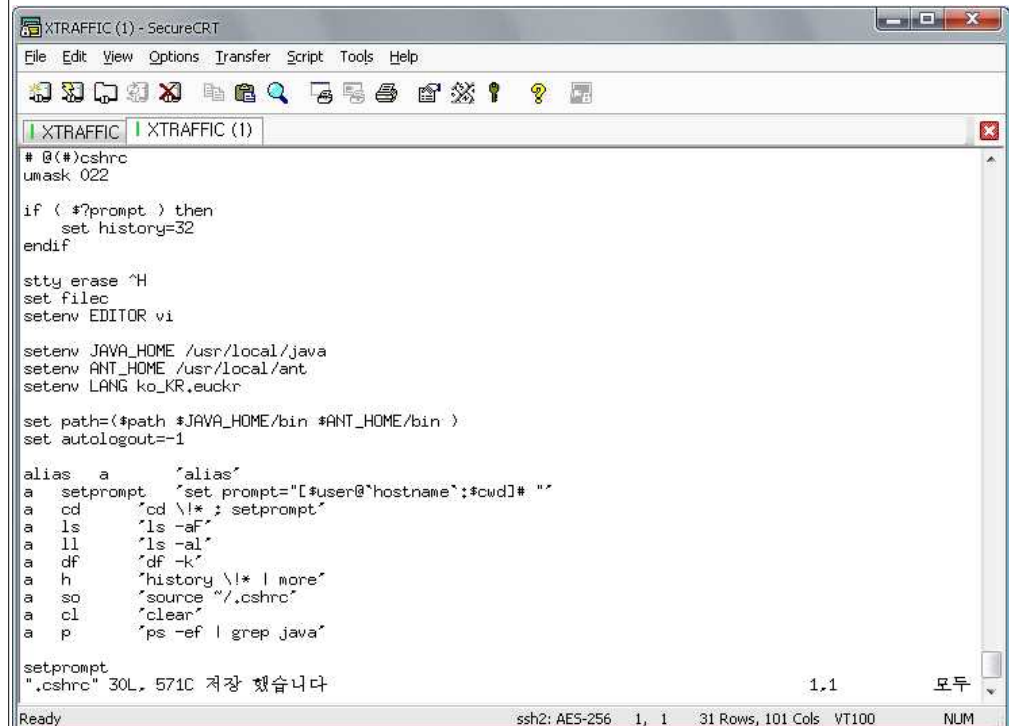
### 가. Pintrus X-Traffic 설치

단계	항목/시험/결과																								
1	시험항목	Pintrus X-Traffic 설치																							
	시험절차	1. 설치																							
	시험결과	<p>1. Pintrus X-Traffic 설치 요소</p> <p>Pintrus X-Traffic은 Java와 .NET 기반의 프로그램이다.</p> <p>서버는 Java로 작성되었으며 데이터베이스는 MySQL 5.0 버전을 지원하며 Asianux3.0 OS (kernel 2.6.18)에 배포본으로 JDK1.6과 MySQL5.0을 갖고 있으며 이를 사용하면 된다.</p> <table border="1"> <thead> <tr> <th>구성요소</th><th>종류 (버전)</th><th>비고</th></tr> </thead> <tbody> <tr> <td>데이터베이스</td><td>MySQL (5.0 이상)</td><td></td></tr> <tr> <td>필수 소프트웨어</td><td>Java (SDK 5.0 이상)</td><td>권장 버전 SDK 6</td></tr> <tr> <td>구성정보 엔진</td><td>SCMEngine (Version 1.0)</td><td></td></tr> <tr> <td>이력정보 엔진</td><td>PHMEngine (Version 1.0)</td><td></td></tr> <tr> <td>분석 엔진</td><td>TAEngine (Version 1.0)</td><td></td></tr> <tr> <td>Flow 수집엔진</td><td>FCEngine (Version 1.0)</td><td></td></tr> <tr> <td>SNMP 수집엔진</td><td>SCEngine (Version 1.0)</td><td></td></tr> </tbody> </table> <p>2. 환경 설정</p> <p>X-Traffic을 설치할 계정 생성과 생성된 계정의 환경변수를 설정한다.</p> <p>1) 그룹, 사용자 생성</p> <p>xtraffic으로 그룹과 사용자를 생성하며 사용하는 셸은csh이다.</p> <pre>\$&gt;groupadd xtraffic \$&gt;useradd xtraffic -g xtraffic -s /bin/csh</pre> <p>2) 사용자 환경 설정</p> <p>Csh의 경우는 사용자 홈 디렉토리에 .cshrc파일 형태로 사용자 환경변수를 설정한다.</p> <p>무조건 csh일 필요는 없으며 bash 셸, ksh 셸을 사용하여도 된다. X-Traffic에서는 csh를 사용하는 것으로 간주되어 작성되었다.</p> <p>Bash 셸을 사용할 경우 사용자 홈 디렉토리에 .bashrc파일이나 .bash_profile파일 형태로 사용자 환경변수를 설정한다.</p> <p>환경변수 지정 명령어 또한 다르며 setenv대신 export 명령을 사용한다.</p> <p>필수 환경변수는 JAVA_HOME, LANG, PATH이다.</p>	구성요소	종류 (버전)	비고	데이터베이스	MySQL (5.0 이상)		필수 소프트웨어	Java (SDK 5.0 이상)	권장 버전 SDK 6	구성정보 엔진	SCMEngine (Version 1.0)		이력정보 엔진	PHMEngine (Version 1.0)		분석 엔진	TAEngine (Version 1.0)		Flow 수집엔진	FCEngine (Version 1.0)		SNMP 수집엔진	SCEngine (Version 1.0)
구성요소	종류 (버전)	비고																							
데이터베이스	MySQL (5.0 이상)																								
필수 소프트웨어	Java (SDK 5.0 이상)	권장 버전 SDK 6																							
구성정보 엔진	SCMEngine (Version 1.0)																								
이력정보 엔진	PHMEngine (Version 1.0)																								
분석 엔진	TAEngine (Version 1.0)																								
Flow 수집엔진	FCEngine (Version 1.0)																								
SNMP 수집엔진	SCEngine (Version 1.0)																								

ANT\_HOME은 부가 프로그램인 ant를 설치하였을 경우 설정하면 된다.  
이외 나머지 환경변수는 사용자 편의를 위한 것들이다.

사용자 환경변수중 LANG값이 현재 버전의 리눅스들은 한국어 지원으로 설치할 경우 대부분 ko\_KR.UTF-8로 지정된다.  
telnet이나 ssh 터미널 프로그램에서 UTF-8 타입을 지원하지 않는 경우가 있기에 사용에 불편함이 있다.  
다국어지원 설정파일인 /etc/sysconfig/i18n파일에 설정된 LANG값을 바꾸어주는 방법도 있으며 이는 서버장비 전체에 변경이 적용된다.

\$>vi .cshrc



```
# @(#)cshrc
umask 022

if ( $?prompt ) then
    set history=32
endif

stty erase ^H
set filec
setenv EDITOR vi

setenv JAVA_HOME /usr/local/java
setenv ANT_HOME /usr/local/ant
setenv LANG ko_KR.euckr

set path=( $path $JAVA_HOME/bin $ANT_HOME/bin )
set autologout=-1

alias a      "alias"
a setprompt  "set prompt='[#user@`hostname`:$cwd]# '"
a cd         "cd \!* ; setprompt"
a ls         "ls -aF"
a ll         "ls -al"
a df         "df -k"
a h          "history \!* | more"
a so         "source ~/.cshrc"
a cl         "clear"
a p          "ps -ef | grep java"

setprompt
".cshrc" 30L, 571C 저장 했습니다

1.1      모두
```

\$> source ~/.cshrc

### 3. 구성정보 엔진 설치

X-Traffic에서 사용하는 구성정보의 조회와 설정시 데이터베이스와 다른 엔진들 그리고 클라이언트의 통신에 가교역할을 한다.

#### 1) 설치 절차

##### (1) 설치 소프트웨어 준비

- 가. X-Traffic에서 제공하는 SCMEngine의 배포버전을 사용한다.
- 나. 배포버전은 zip파일 압축 형태로 배포된다.

##### (2) Xtraffic 사용자로 설치

- 가. Unzip 명령어로 배포버전 파일의 압축을 해제한다.
- 나. SCMEngine 디렉토리가 생성되며 다음과 같은 디렉토리

구조를 갖는다.

- Bin : 실행 파일
- Build : 배포 파일
- Conf : 설정 파일

- Lib : 사용 라이브러리 파일
  - Ext : 사용 확장 라이브러리 파일(SDK5.0에서 미 지원하는 SDK6.0 라이브러리)
  - Log : 로그 파일
  - As\_info : AS 정보 자동 업데이트 데이터 파일
- 다. 데이터베이스 연결정보 설정
- Conf 디렉토리의 scme\_db.xml파일을 열어 데이터베이스 연결 정보를 수정한다.
  - 접속 Host 정보, 계정 정보

\$> vi scme\_db.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE properties SYSTEM "http://java.sun.com/dtd/properties.dtd">
<properties>
<comment>Engine Configuration Propertis XML</comment>
<entry key="db.repositoryName">SCM</entry>
<entry key="db.kind">1</entry>
<entry key="db.host">127.0.0.1</entry>
<entry key="db.port">3306</entry>
<entry key="db.dbName">XTRAFFIC</entry>
<entry key="db.userName">xtraffic</entry>
<entry key="db.password">2x1a</entry>
<entry key="db.usePool">>true</entry>
<entry key="db.initConn">1</entry>
<entry key="db.maxActiveConn">255</entry>
<entry key="db.maxInactiveConn">5</entry>
<entry key="db.connExpireTime">300</entry>
<entry key="db.connWaitTime">3</entry>
<entry key="db.sleepTime">60</entry>
</properties>
```

- 라. 구성정보 엔진 구동
- Bin 디렉토리의 확장자 sh파일에 실행권한을 부여한다.
  - 설치된 Java VM이 32bit이면 scme.sh를 64bit이면 scme64.sh를 사용하여 start 옵션으로 실행한다.

\$> scme64.sh start

- 마. 구성정보 엔진 종료
- 설치된 Java VM이 32bit이면 scme.sh를 64bit이면 scme64.sh를 사용하여 stop -force 옵션으로 실행한다.

\$> scme64.sh stop -force

#### 4. 이력정보 엔진 설치

X-Traffic에서 처리되는 분석정보, 이력정보, 감지정보 등을 데이터베이스에 저장, 조회, 삭제 등의 하는 다른 엔진들과 클라이언트와의 가교역할을 한다.

##### 1) 설치 절차

##### (1) 설치 소프트웨어 준비

- 가. X-Traffic에서 제공하는 PHMEngine의 배포버전을 사용한다.
- 나. 배포버전은 zip파일 압축 형태로 배포된다.

##### (2) Xtraffic 사용자로 설치

- 가. Unzip 명령어로 배포버전 파일의 압축을 해제한다.
- 나. PHMEngine 디렉토리가 생성되며 다음과 같은 디렉토리

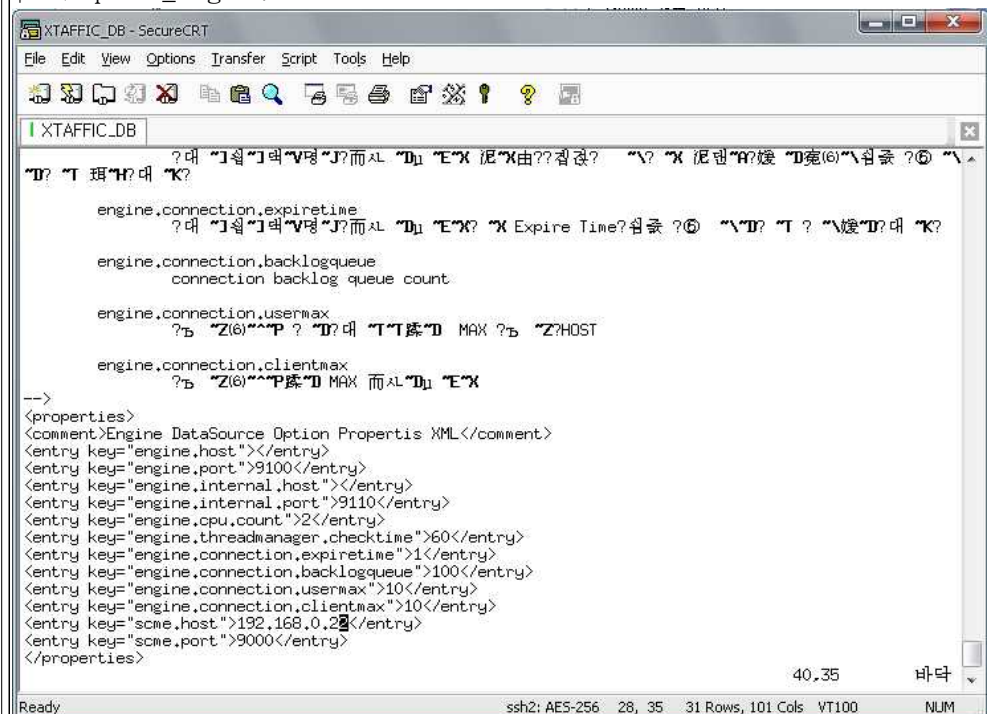
구조를 갖는다.

- Bin : 실행 파일
- Build : 배포 파일
- Conf : 설정 파일
- Lib : 사용 라이브러리 파일
- Ext : 사용 확장 라이브러리 파일(SDK5.0에서 미 지원하는

SDK6.0 라이브러리)

- Log : 로그 파일
- 다. 구성정보 엔진 연결정보 설정
- Conf 디렉토리의 phme\_engine.xml파일을 열어 구성정보 엔진 연결정보를 수정한다.

\$> vi phme\_engine.xml



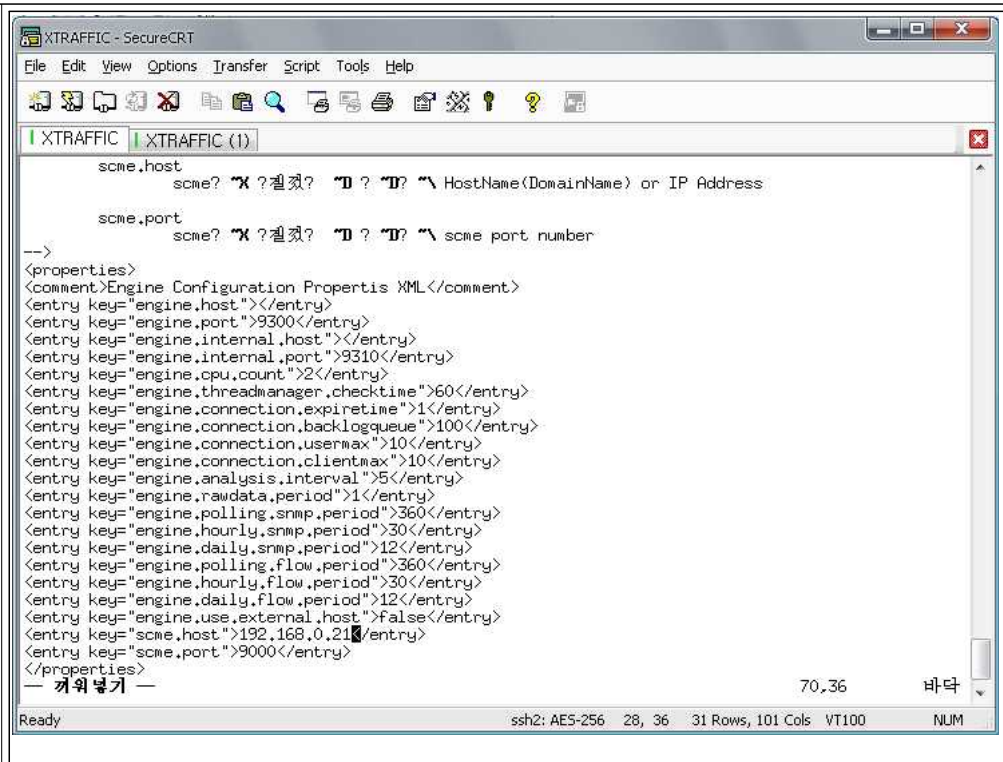
```

engine.connection.expiretime
engine.connection.backlogqueue
engine.connection.usermax
engine.connection.clientmax

-->
<properties>
<comment>Engine DataSource Option Propertis XML</comment>
<entry key="engine.host"></entry>
<entry key="engine.port">9100</entry>
<entry key="engine.internal.host"></entry>
<entry key="engine.internal.port">9110</entry>
<entry key="engine.cpu.count">2</entry>
<entry key="engine.threadmanager.checktime">60</entry>
<entry key="engine.connection.expiretime">1</entry>
<entry key="engine.connection.backlogqueue">100</entry>
<entry key="engine.connection.usermax">10</entry>
<entry key="engine.connection.clientmax">10</entry>
<entry key="scme.host">192.168.0.2</entry>
<entry key="scme.port">9000</entry>
</properties>
    
```



		<p>라. 이력정보 엔진 구동</p> <ul style="list-style-type: none"> <li>● Bin 디렉토리의 확장자 sh파일에 실행권한을 부여한다.</li> <li>● 설치된 Java VM이 32bit이면 phme.sh를 64bit이면 phme64.sh를 사용하여 start 옵션으로 실행한다.</li> </ul> <pre>\$&gt; phme64.sh start</pre> <p>마. 이력정보 엔진 종료</p> <ul style="list-style-type: none"> <li>● 설치된 Java VM이 32bit이면 phme.sh를 64bit이면 phme64.sh를 사용하여 stop -force 옵션으로 실행한다.</li> </ul> <pre>\$&gt; phme64.sh stop -force</pre> <p>5. 분석 엔진 설치</p> <p>X-Traffic 수집서버에서 수집된 트래픽 데이터를 분석, 가공하여 데이터베이스화 하며 데이터 축약, Baseline 분석 등의 기능을 수행한다.</p> <p>1) 설치 절차</p> <p>(1) 설치소프트웨어 준비</p> <p>가. X-Traffic에서 제공하는 TAEengine의 배포버전을 사용한다.</p> <p>나. 배포버전은 zip파일 압축 형태로 배포된다.</p> <p>(2) Xtraffic 사용자로 설치</p> <p>가. Unzip 명령어로 배포버전 파일의 압축을 해제한다.</p> <p>나. TAEengine 디렉토리가 생성되며 다음과 같은 디렉토리 구조를 갖는다.</p> <ul style="list-style-type: none"> <li>● Bin : 실행 파일</li> <li>● Build : 배포 파일</li> <li>● Conf : 설정 파일</li> <li>● Lib : 사용 라이브러리 파일</li> <li>● Ext : 사용 확장 라이브러리 파일(SDK5.0에서 미 지원하는 SDK6.0 라이브러리)</li> <li>● Log : 로그 파일</li> </ul> <p>다. 구성정보 엔진 연결정보 설정</p> <ul style="list-style-type: none"> <li>● Conf 디렉토리의 ta_engine.xml파일을 열어 구성정보 엔진 연결정보를 수정한다.</li> </ul> <pre>\$&gt; vi ta_engine.xml</pre>
--	--	---



```

XTRAFFIC - SecureCRT
File Edit View Options Transfer Script Tools Help

XTRAFFIC XTRAFFIC (1)

sme.host
sme? "X" ?질렀? "D" ? "D" "\ HostName(DomainName) or IP Address

sme.port
sme? "X" ?질렀? "D" ? "D" "\ sme port number

-->
<properties>
<comment>Engine Configuration Properties XML</comment>
<entry key="engine.host"></entry>
<entry key="engine.port">9300</entry>
<entry key="engine.internal.host"></entry>
<entry key="engine.internal.port">9310</entry>
<entry key="engine.cpu.count">2</entry>
<entry key="engine.threadmanager.checktime">60</entry>
<entry key="engine.connection.expiretime">1</entry>
<entry key="engine.connection.backlogqueue">100</entry>
<entry key="engine.connection.usermax">10</entry>
<entry key="engine.connection.clientmax">10</entry>
<entry key="engine.analysis.interval">5</entry>
<entry key="engine.rawdata.period">1</entry>
<entry key="engine.polling.snmp.period">360</entry>
<entry key="engine.hourly.snmp.period">30</entry>
<entry key="engine.daily.snmp.period">12</entry>
<entry key="engine.polling.flow.period">360</entry>
<entry key="engine.hourly.flow.period">30</entry>
<entry key="engine.daily.flow.period">12</entry>
<entry key="engine.use.external.host">false</entry>
<entry key="sme.host">192.168.0.21</entry>
<entry key="sme.port">9000</entry>
</properties>
-- 끼워넣기 --
Ready ssh2: AES-256 28, 36 31 Rows, 101 Cols VT100 NUM
  
```

라. 분석 엔진 구동

- Bin 디렉토리의 확장자 sh파일에 실행권한을 부여한다.
- 설치된 Java VM이 32bit이면 tae.sh를 64bit이면 tae64.sh를

사용하여 start 옵션으로 실행한다.

```
$> tae64.sh start
```

마. 분석 엔진 종료

- 설치된 Java VM이 32bit이면 tae.sh를 64bit이면 tae64.sh를 사용하여 stop -force 옵션으로 실행한다.

```
$> tae64.sh stop -force
```

## 6. Flow 수집엔진 설치

네트워크 장비나 Flow Probe등에서 제공하는 sFlow, Netflow, cFlow등을 수집하여 1차 분석을 수행한다.

### 1) 설치절차

#### (1) 설치 소프트웨어 준비

- 가. X-Traffic에서 제공하는 FCEngine의 배포버전을 사용한다.
- 나. 배포버전은 zip파일 압축 형태로 배포된다.

#### (2) Xtraffic 사용자로 설치

- 가. Unzip 명령어로 배포버전 파일의 압축을 해제한다.
- 나. FCEngine 디렉토리가 생성되며 다음과 같은 디렉토리 구조를

찾는다.

- Bin : 실행 파일
- Build : 배포 파일
- Conf : 설정 파일
- Lib : 사용 라이브러리 파일
- Ext : 사용 확장 라이브러리 파일(SDK5.0에서 미 지원하는 SDK6.0 라이브러리)

다.

- Log : 로그 파일
- Dump : Raw Flow 데이터 덤프 파일
- Analyzed : 1차 분석된 Flow 데이터 덤프 파일

구성정보 엔진 연결정보 설정

- Conf 디렉토리의 fc\_engine.xml파일을 열어 구성정보 엔진 연결정보를 수정한다.

\$> vi fc\_engine.xml

```

engine.analysis.interval
  raw dump file? "X" "D" "I" "E" "N" "T" "E" "R" "N" "A" "L" "I" "Z" "E" "D" "D" "U" "M" "P" "F" "I" "L" "E"

engine.rawdata.period
  Raw Flow "D" "U" "M" "P" "F" "I" "L" "E" "S" "I" "Z" "E" "D" "D" "U" "M" "P" "F" "I" "L" "E"

scme.host
  scme? "X" "D" "U" "M" "P" "F" "I" "L" "E" "S" "I" "Z" "E" "D" "D" "U" "M" "P" "F" "I" "L" "E"
  scme? "X" "D" "U" "M" "P" "F" "I" "L" "E" "S" "I" "Z" "E" "D" "D" "U" "M" "P" "F" "I" "L" "E"

scme.port
  scme? "X" "D" "U" "M" "P" "F" "I" "L" "E" "S" "I" "Z" "E" "D" "D" "U" "M" "P" "F" "I" "L" "E"
  scme? "X" "D" "U" "M" "P" "F" "I" "L" "E" "S" "I" "Z" "E" "D" "D" "U" "M" "P" "F" "I" "L" "E"

-->
<properties>
<comment>Engine Configuration Properties XML</comment>
<entry key="engine.host"></entry>
<entry key="engine.port">9600</entry>
<entry key="engine.internal.host"></entry>
<entry key="engine.internal.port">9610</entry>
<entry key="engine.cpu.count">2</entry>
<entry key="engine.threadmanager.checktime">60</entry>
<entry key="engine.connection.expirytime">1</entry>
<entry key="engine.connection.backlogqueue">100</entry>
<entry key="engine.connection.usermax">10</entry>
<entry key="engine.connection.clientmax">10</entry>
<entry key="engine.analysis.interval">5</entry>
<entry key="engine.rawdata.period">1</entry>
<entry key="engine.dump.buffer.size">8192000</entry>
<entry key="scme.host">192.168.0.22</entry>
<entry key="scme.port">9000</entry>
</properties>
  
```

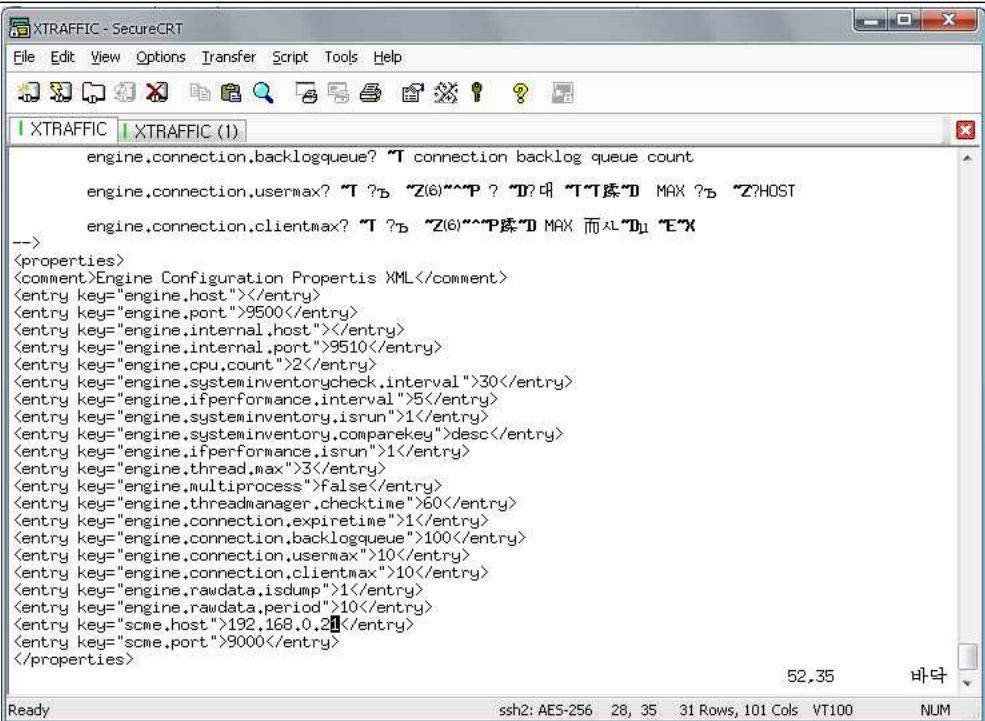
라. Flow수집 엔진 구동

- Bin 디렉토리의 확장자 sh파일에 실행권한을 부여한다.
- 설치된 Java VM이 32bit이면 fce.sh를 64bit이면 fce64.sh를

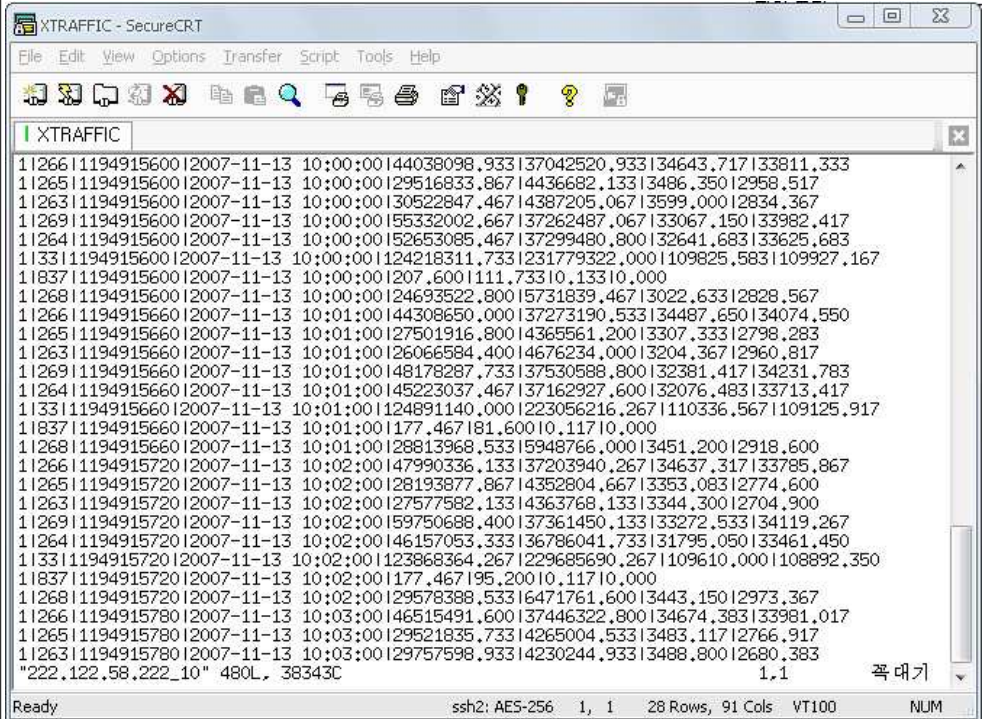
사용하여 start 옵션으로 실행한다.

\$> fce64.sh start

		<p>마. Flow수집 엔진 종료</p> <ul style="list-style-type: none"> <li>● 설치된 Java VM이 32bit이면 fce.sh를 64bit이면 fce64.sh를 사용하여 stop -force 옵션으로 실행한다.</li> </ul> <pre>\$&gt; fce64.sh stop -force</pre> <p>7. SNMP 수집엔진 설치</p> <p>분석 대상 네트워크 장비에서 SNMP를 이용 장비 구성정보와 성능정보를 수집하는 기능을 수행한다.</p> <p>1) 설치 절차</p> <p>(1) 설치 소프트웨어 준비</p> <p>가. X-Traffic에서 제공하는 SCEngine의 배포버전을 사용한다.</p> <p>나. 배포버전은 zip파일 압축 형태로 배포된다.</p> <p>(2) Xtraffic 사용자로 설치</p> <p>가. Unzip 명령어로 배포버전 파일의 압축을 해제한다.</p> <p>나. SCEngine 디렉토리가 생성되며 다음과 같은 디렉토리 구조를 갖는다.</p> <ul style="list-style-type: none"> <li>● Bin : 실행 파일</li> <li>● Build : 배포 파일</li> <li>● Conf : 설정 파일</li> <li>● Lib : 사용 라이브러리 파일</li> <li>● Ext : 사용 확장 라이브러리 파일(SDK5.0에서 미 지원하는 SDK6.0 라이브러리)</li> <li>● Log : 로그 파일</li> <li>● Analyzed : 수집 주기로 수집된 SNMP 성능데이터를 분석 주기 데이터로 가공한 데이터 파일</li> </ul> <p>다. 구성정보 엔진 연결정보 설정</p> <ul style="list-style-type: none"> <li>● Conf 디렉토리의 sc_engine.xml파일을 열어 구성정보 엔진 연결정보를 수정한다.</li> </ul> <pre>\$&gt; vi sc_engine.xml</pre>
--	--	---

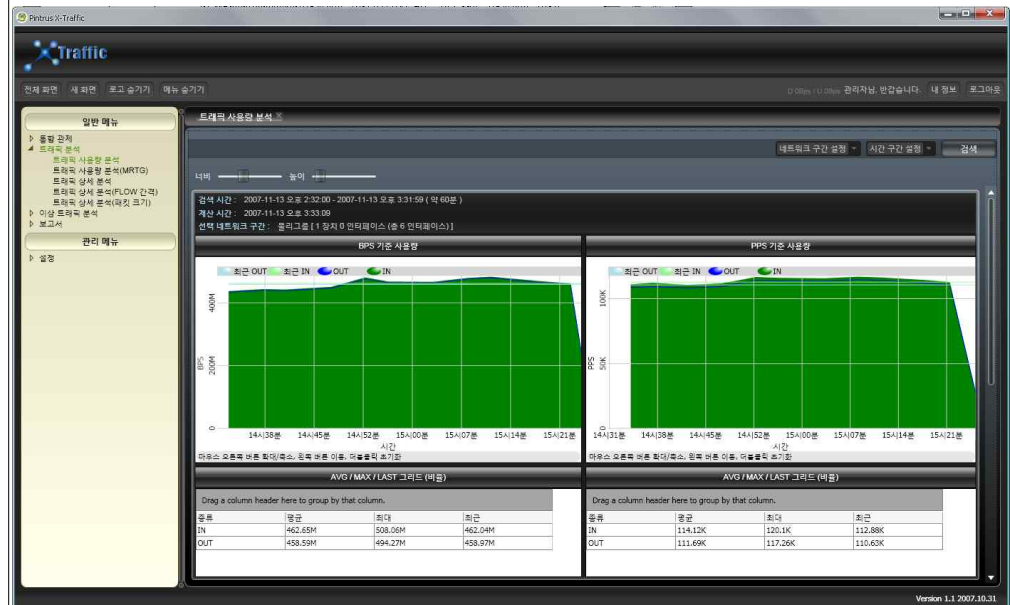
		 <p>라. SNMP수집 엔진 구동</p> <ul style="list-style-type: none"> <li>● Bin 디렉토리의 확장자 sh파일에 실행권한을 부여한다.</li> <li>● 설치된 Java VM이 32bit이면 sce.sh를 64bit이면 sce64.sh를 사용하여 start 옵션으로 실행한다.</li> </ul> <pre>\$&gt; sce64.sh start</pre> <p>마. SNMP수집 엔진 종료</p> <ul style="list-style-type: none"> <li>● 설치된 Java VM이 32bit이면 sce.sh를 64bit이면 sce64.sh를 사용하여 stop -force 옵션으로 실행한다.</li> </ul> <pre>\$&gt; sce64.sh stop -force</pre>
비	고	

## 나. Pintrus X-Traffic 기능 테스트

단계	항목/시험/결과	
	시험항목	SNMP 데이터 수집/분석
	시험절차	1. SNMP 데이터 수집 2. SNMP 데이터 분석 3. SNMP 데이터 MRTG 분석 4. SNMP 데이터 축약
1	시험결과	1. SNMP 수집 엔진이 설치된 서버에 telnet 또는 ssh로 접속한다.  SEngine/analyzed/[대상장비 IP주소]디렉토리로 이동하여 [대상장비 IP주소]_[시간] 형태의 파일을 열어본다. 정보는 “ ”로 다음과 같이 구분되어 저장된다. "대상장비 인터페이스 시간 시간 BPS_IN BPS_OUT PPS_IN PPS_OUT"

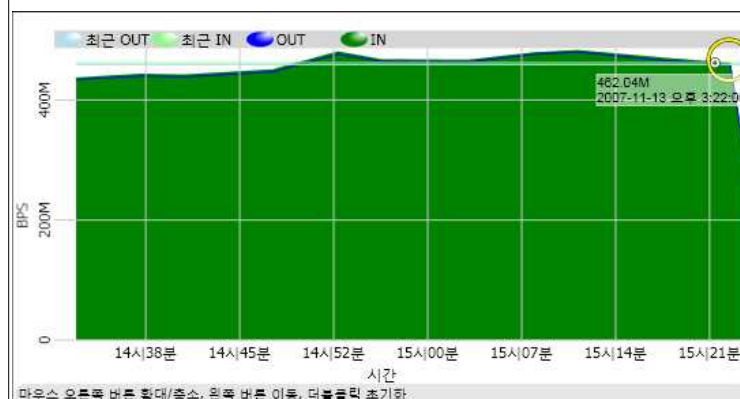


2. X-Traffic Client를 실행하여 좌측 일반 메뉴에 있는 트래픽 분석>>트래픽 사용량 분석 메뉴를 선택한 후 네트워크 구간을 전체로 시간 구간을 최근 1시간으로 설정한 후 검색 버튼을 클릭한다.



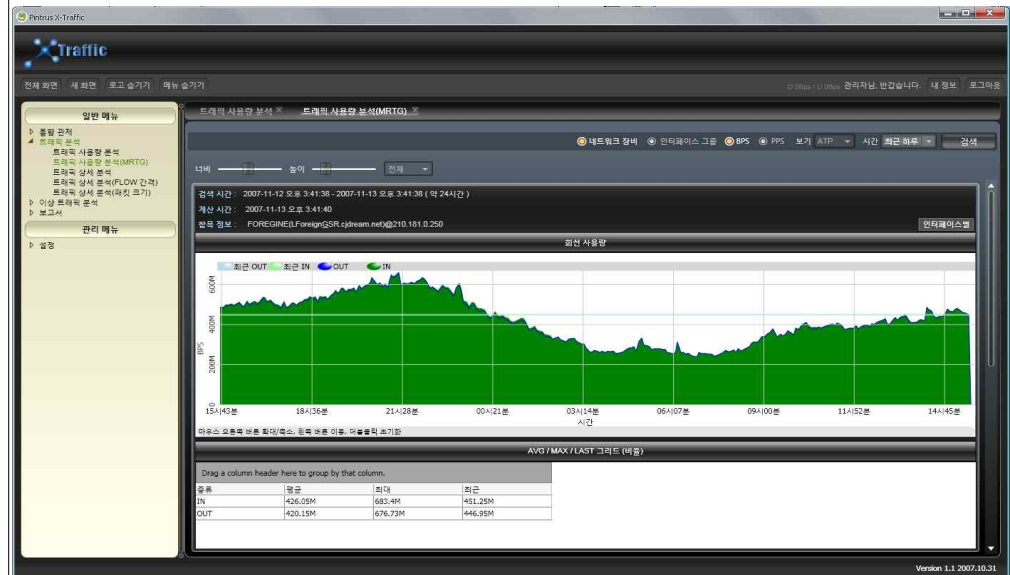
<그림 1 트래픽사용량 분석1>

그래프에 마우스 커서를 가져가면 다음 그림과 같이 노란색 원형이 나타나며 시간정보와 사용량 정보가 표시된다.



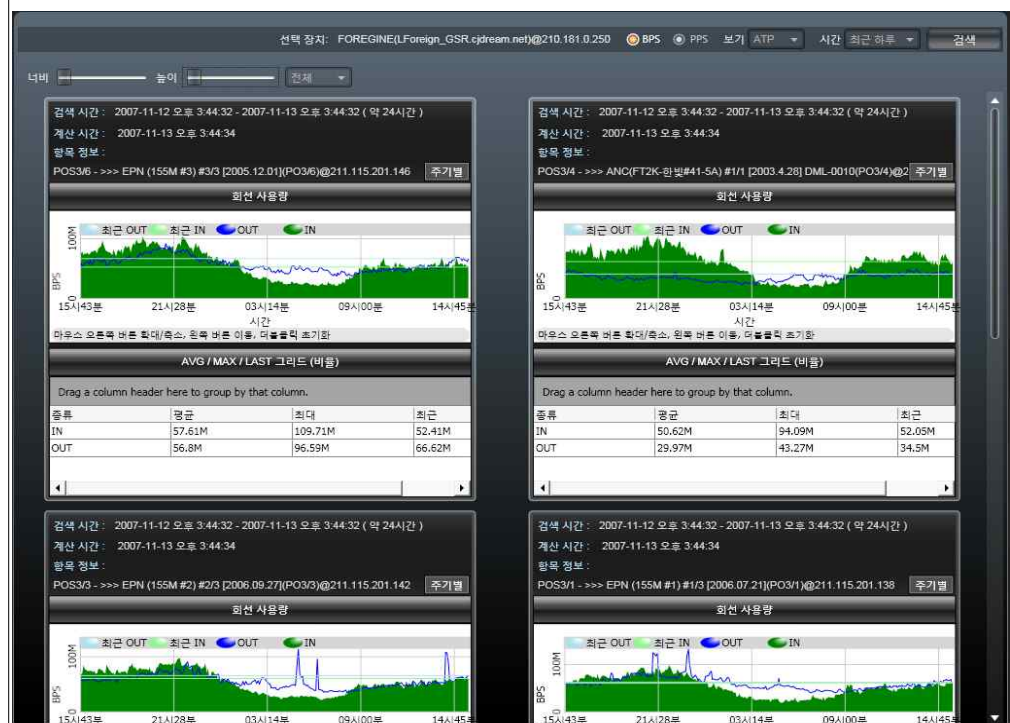
<그림 2 트래픽사용량 분석2>

3. X-Traffic Client를 실행하여 좌측 일반 메뉴에 있는 트래픽 분석>>트래픽 사용량 분석(MRTG) 메뉴를 선택한 후 검색 버튼을 클릭한다.



<그림 3 트래픽사용량 분석(MRTG)>

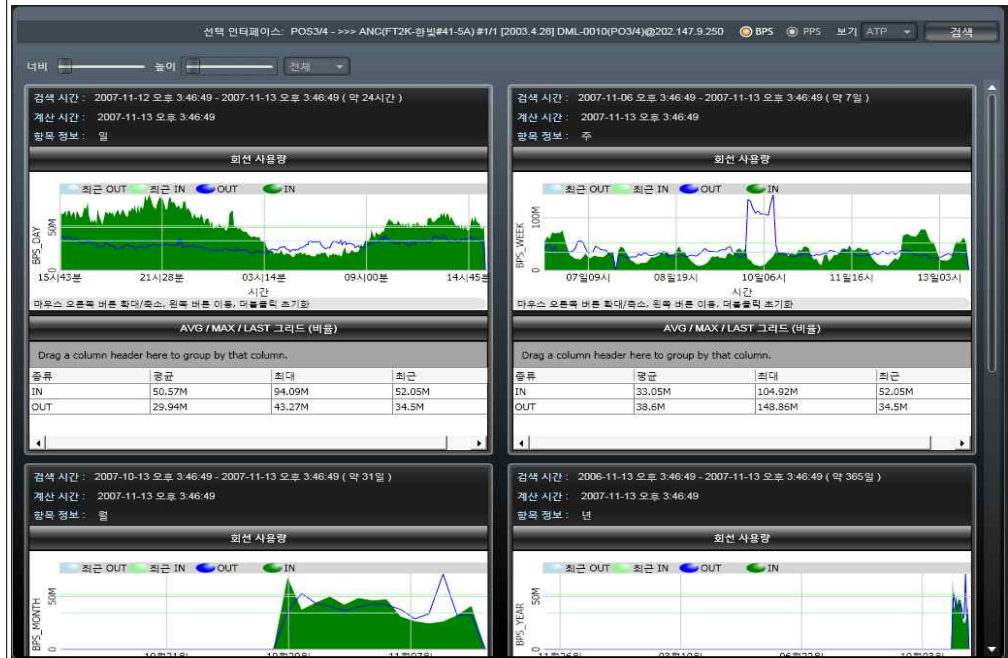
그래프 우측 상단에 위치한 인터페이스별 버튼을 클릭하면 장비에 속한 모든 인터페이스에 대한 회선 사용량 데이터를 조회할 수 있다.



<그림 4 인터페이스에 대한 회선 사용량 데이터 조회>



그래프 우측 상단에 위치한 주기별 버튼을 클릭하면 해당 인터페이스에 대한 일/주/월/년 그래프를 조회할 수 있다.



<그림 5 인터페이스에 일/주/월/년 그래프 조회>

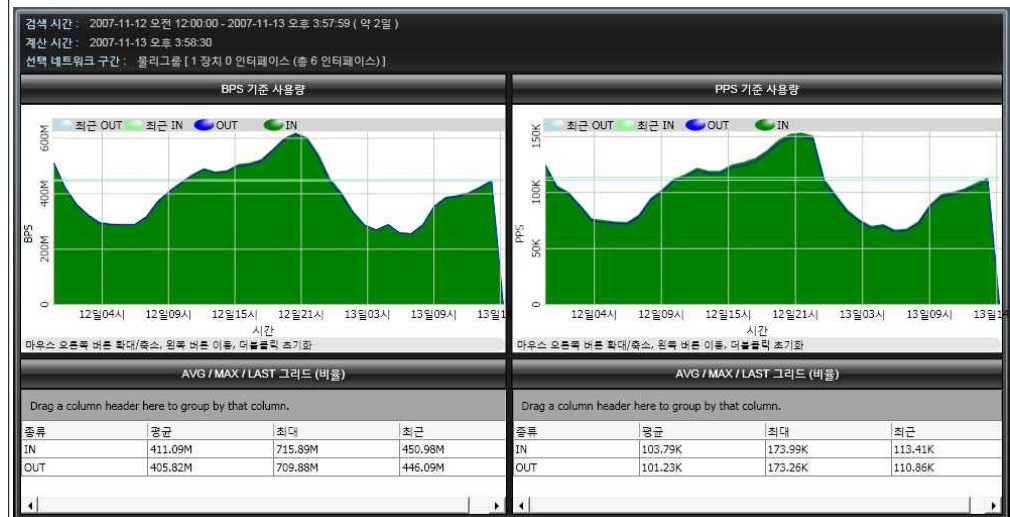
4. X-Traffic Client를 실행하여 좌측 일반 메뉴에 있는 트래픽 분석>>트래픽 사용량 분석 메뉴를 선택한 후 네트워크 구간을 전체로 시간을 30시간 전, 10일 전으로 각각 선택하여 조회한다.

현재시간에서 최근 30시간 이전 데이터



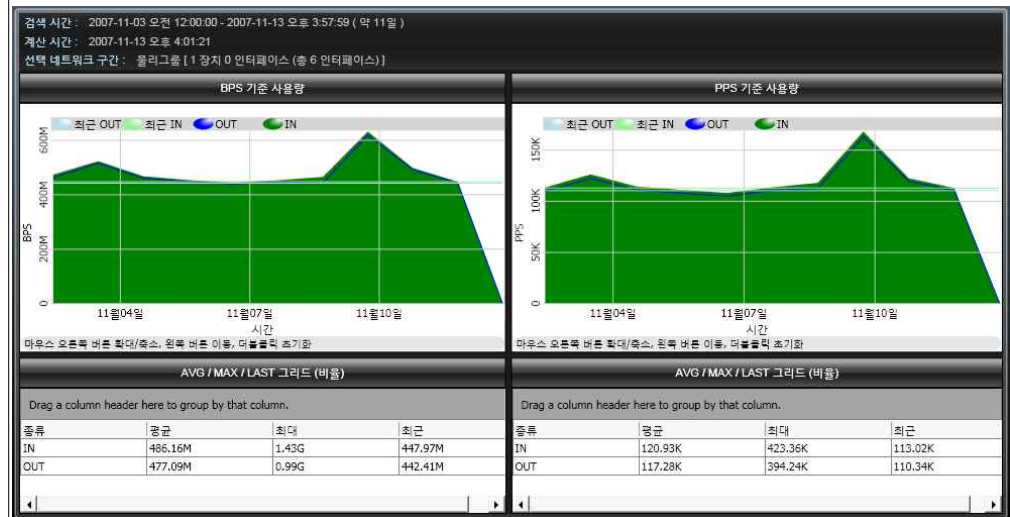
<그림 6 네트워크 구간 조회1>

최근 30시간 이후부터 최근 10일 이전 데이터



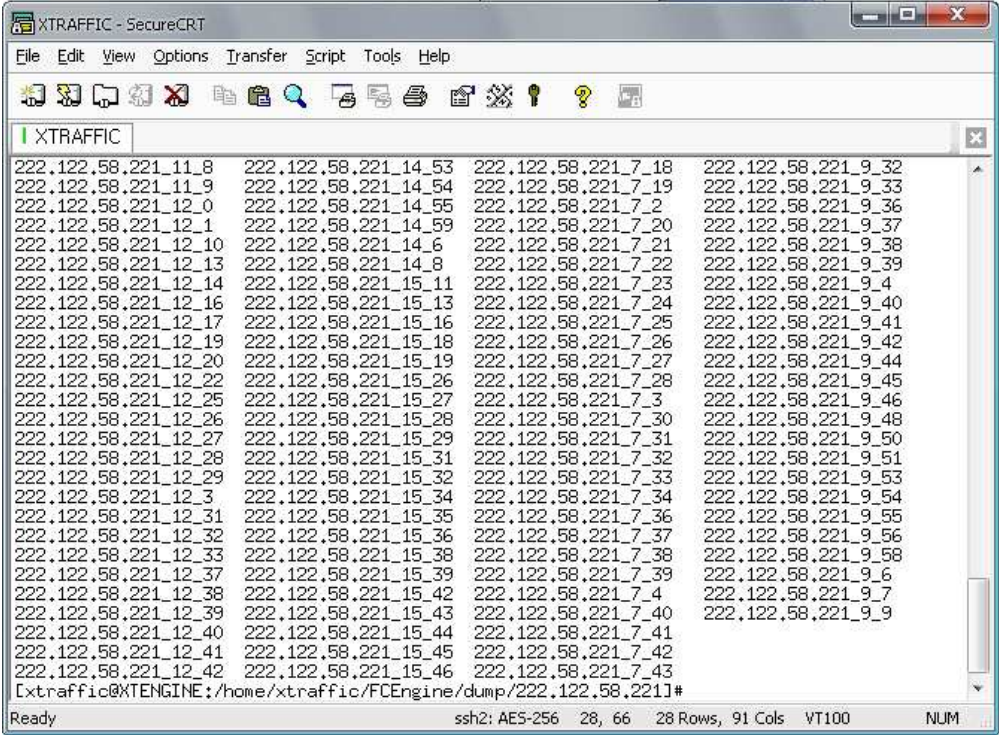
<그림 7 네트워크 구간 조회2>

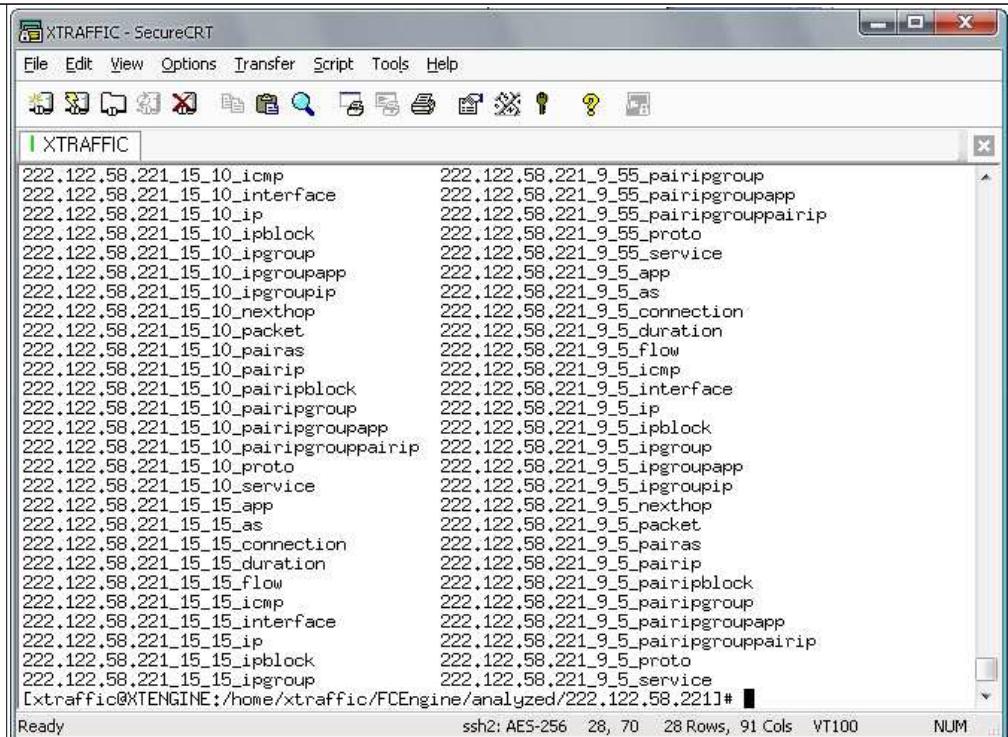
최근 10일 이후부터 데이터 최대 유지기간까지의 데이터



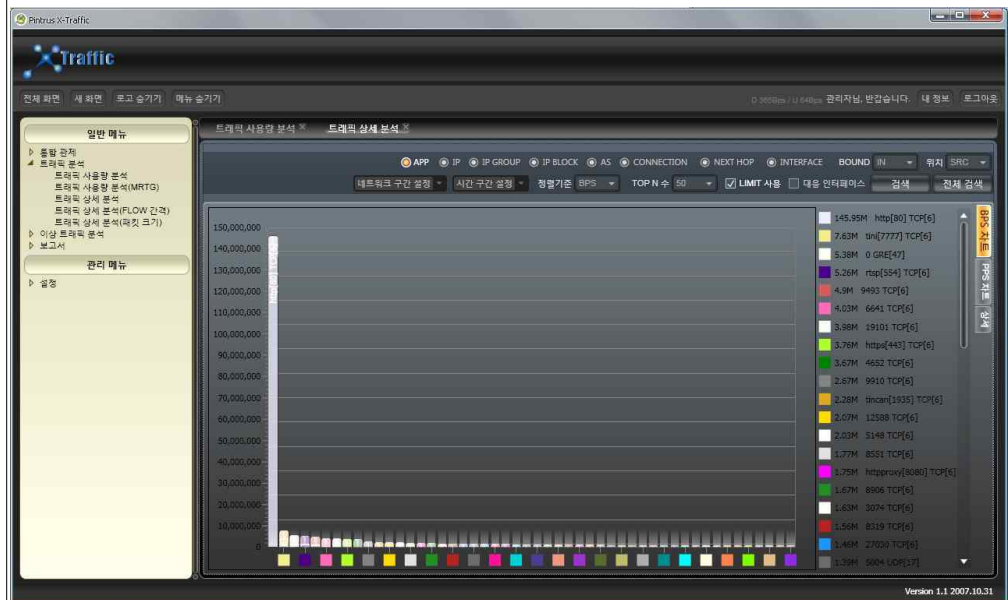
<그림 8 네트워크 구간 조회3>

비 고

단계	항목/시험/결과	
	시험항목	Flow 데이터 수집/분석
	시험절차	1. Flow 데이터 수집 2. Application 데이터 분석 3. IP 데이터 분석 4. Flow 데이터 축약
1	시험결과	<p>1. Flow 수집 엔진이 설치된 서버에 telnet 또는 ssh로 접속한다.</p>  <p>FCEngine/dump/[대상장비 IP주소]디렉토리로 이동하여 [대상장비 IP주소]_[시간]_[분] 형태의 파일을 확인한다. 이는 Flow UDP Packet을 파일로 저장한것이다.</p> <p>FCEngine/analyzed[대상장비 IP주소]디렉토리로 이동하여 [대상장비 IP주소]_[시간]_[분] 형태의 파일을 확인한다. 이는 dump 디렉토리에 저장된 Raw Flow 데이터를 분석하여 분석항목별로 저장한것 이다.</p> <p>2. X-Traffic Client를 실행하여 좌측 일반 메뉴에 있는 트래픽 분석&gt;&gt;트래픽 상세 분석 메뉴를 선택한 후 App Radio 버튼을 클릭하고 네트워크 구간을 전체, 시간 구간을 1시간으로 설정한 후 검색버튼을 클릭한다.</p>



BPS TOP N 사용량



<그림 9 BPS TOP N 사용량 1>

PPS TOP N 사용량

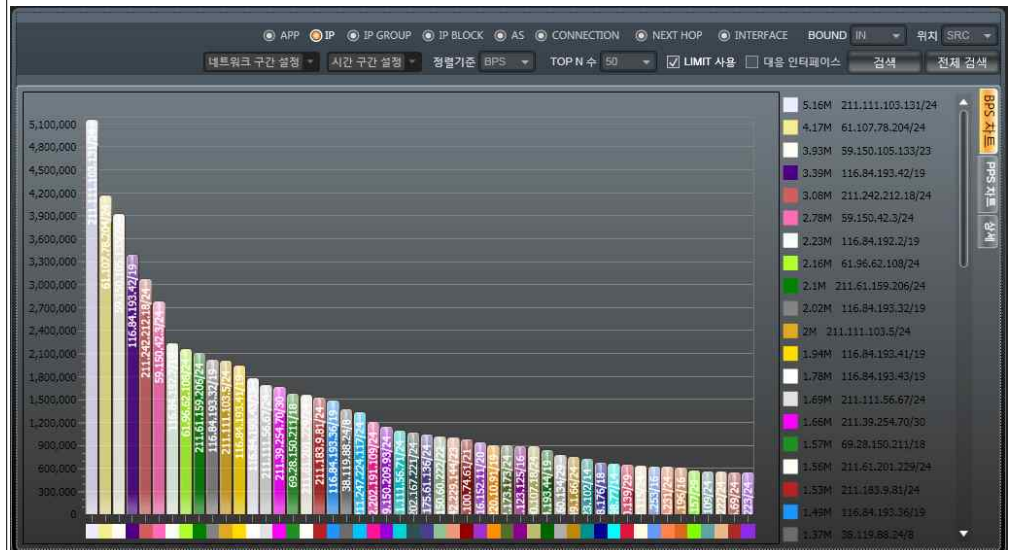




<그림 10 PPS TOP N 사용량 1>

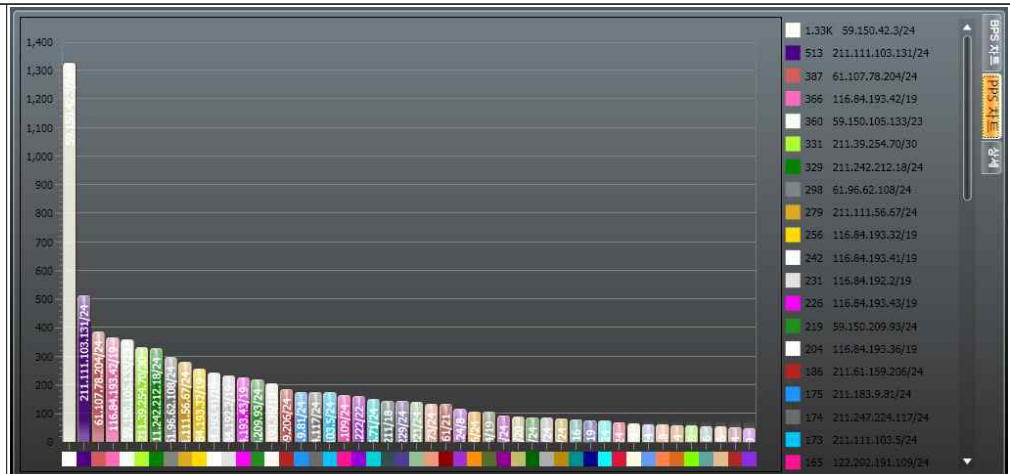
3. X-Traffic Client를 실행하여 좌측 일반 메뉴에 있는 트래픽 분석>>트래픽 상세 분석 메뉴를 선택한 후 IP Radio 버튼을 클릭하고 네트워크 구간을 전체, 시간 구간을 1시간으로 설정한 후 검색버튼을 클릭한다.

BPS TOP N 사용량



<그림 11 BPS TOP N 사용량 2>

PPS TOP N 사용량



<그림 12 PPS TOP N 사용량 2>

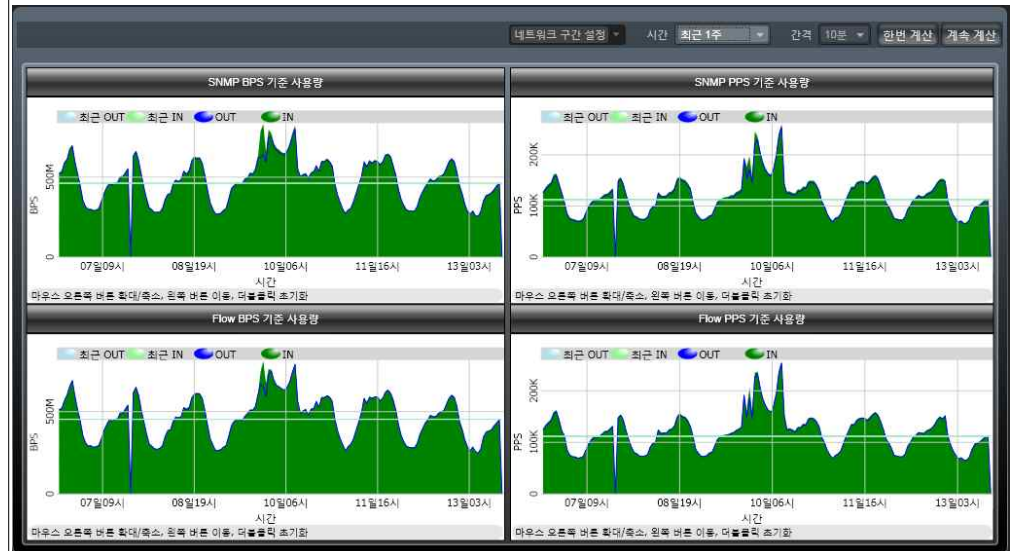
4. X-Traffic Client를 실행하여 좌측 일반 메뉴에 있는 통합 관제>>요약(그래프) 트래픽 사용량 메뉴를 선택한 후 네트워크 구간을 전체, 시간 구간을 1시간으로 설정한 후 검색버튼을 클릭한다.



<그림 13 트래픽 구간 사용 검색>

데이터 축약은 SNMP 데이터와 동일하게 수집주기 데이터 30시간, 1시간 간격 데이터 10일 1일 간격 데이터 최장 유지기간까지(통상 1년)로 이루어진다. 시간선택을 최근 1주와 최근 1달로 선택하여 각각 축약 데이터를 확인한다.

#### 1시간 간격 데이터



<그림 14 트래픽 1시간 간격 데이터 검색>

#### 1일 간격 데이터



<그림 15 트래픽 1일 간격 데이터 검색>

비 고