Automated Application Signature Generation for Traffic Identification

Young J. Won, Seong-Chul Hong,
Byung-Chul Park, and James W. Hong

Distributed Processing and Network Management Lab.
Dept. of Computer Science and Engineering
POSTECH, Korea
{yjwon, jwkhong}@postech.ac.kr

Aug. 16, 2008

Outline

- Introduction on DPNM, POSTECH
- Our Experience on Measurement
- Automated Signature Generation
- Conclusion

POSTECH Since 1986

- Founded by POSCO – 2nd largest iron and steel manufacturer in the world
  - 3000 students, 230 faculty members, 800 researchers
- Distributed Processing and Network Management Lab. (http://dpnm.postech.ac.kr) since 1995
  - 6 PhD students, 3 MS students, 1 researcher as of 2008

Recent Industry Projects

Projects Regarding Traffic Measurement & Analysis Only

- **Korea Telecom (KT)**
  - BGP threats & ISP relations (2008~)
  - Bundled service traffic analysis (2007)
  - Application-level traffic classification (2006)
  - High-speed network monitoring system (2005)
- **POSCO**
  - Industrial control networks fault detection & prediction (2008~)
  - Remote monitoring & fault analysis in industrial control network n etworks (2007)
- **Government**
  - CASFI (2008)
  - High-speed traffic monitoring & audit systems (2004~2005)
- **Others**
  - nTelia – Traffic analysis of mobile data networks (2006)

POSTECH's Experiences in Traffic Measurement & Analysis

- Traffic Monitoring Systems
- Enterprise Networks
- Mobile Data Networks
- Industrial Control Networks
- IPTV Traffic

Traffic Monitoring Systems

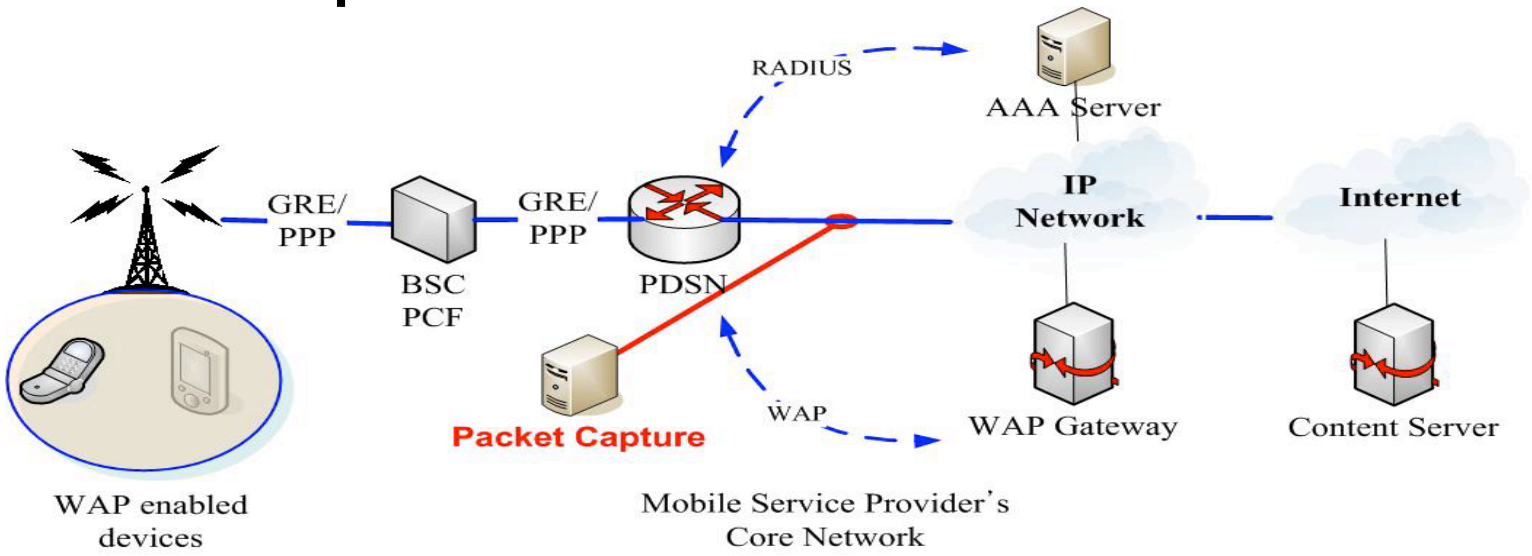
- **MRTG+ (1997)**
  - Extension of MRTG, LIVE visualization of traffic
- **WebTrafMon-I & II (1998, 2000)**
  - Passive traffic monitoring system (up to 100 Mbps)
  - Distributed architecture
- **NGMON (2002~)**
  - Next Generation Network MONitoring and Analysis System
  - Targeting 1-10 Gbps or higher networks
  - Traffic classification, security attack detection & host analysis

Enterprise Networks

- Campus Networks
  - Characteristics analysis of Internet traffic from the perspecti ve of flows [ComCom '06]
  - Application-level traffic monitoring & analysis [ETRI '05]
- Korea Internet eXchange (2004)
- Participating DITL packet collection (2007, 2008)
- Analysis Categories
  - Flow size / duration / packet distribution / size distribution / f lash flows / volume pattern / flow occurrence period / port n umber distribution and more
  - Flow & Packet-based analysis
  - Focusing on traffic classification & its applications

Mobile Data Networks

- **Investigating the unique and unusual traffic characteristics reflecting the user and data service patterns [PAM '07]**
  - **Previous works are limited to small scale measurement study between the selected end hosts**
  - **They focused on TCP or performance factors rather than understanding the user behavior and the root cause for such phenomenon**



Industrial Control Networks

- **Industrial Control Networks (ICN)?**
  - Robust communications between controlling and controlled devices in a manufacturing environment
    - Building, Factory, and Process Automation
  - Mission critical process & Non-fault tolerable networks
  - Emergence of Industrial Ethernet → Ethernet/IP-based
    - EtherNet/IP, PROFINET, TCnet, Vnet/IP, EPA, RAPIEnet
  - Real-world ICN test bed: POSCO

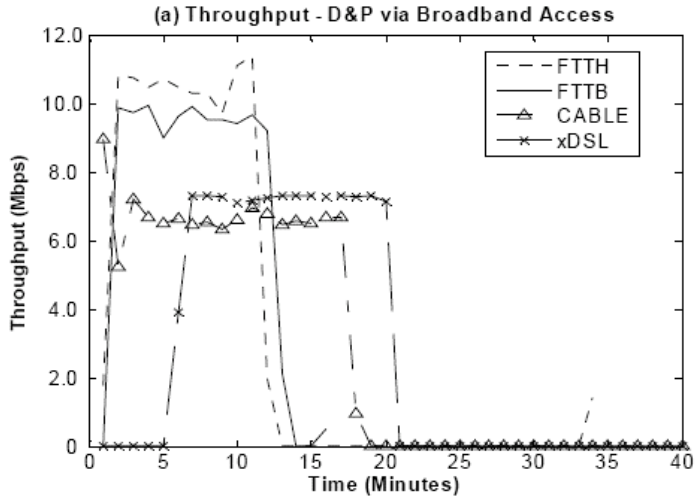
- **Problems?**
  - The cost of network malfunctioning is severe.
  - ICN fault diagnosis techniques require different standards.
    - due to differences of traffic nature

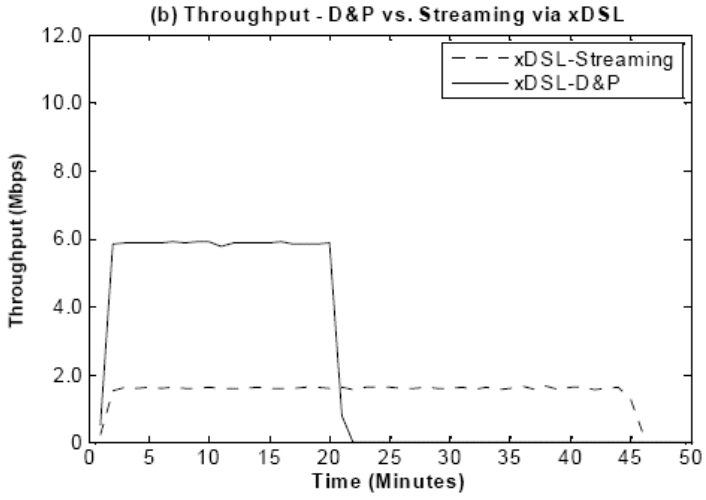
- **Papers**
  - Traffic characteristics [APNOMS '07]
  - Fault detection and analysis system [ComMag '08]

IPTV Traffic

- **Investigation of combinational traffic models for TPS components**
  - **Bandwidth demand models, Traffic impact analysis**

- **Commercial IPTV traffic measurements [ComMag '08]**
  - **End-user IPTV traffic measurements of residential broadband access networks**
    - **IPTV STB over ADSL, Cable, FTTB, and FTTH**





Automated Signature Generation for Traff ic Identification

Traffic Classification

- Classification has been done based on: [Szabo '08]
  - Port
  - Signature
  - Connection pattern
  - Statistics
  - Information theory
  - Combined classification method

- Signature-based method often is used as ground truth for validation
  - We focus on obtaining accurate signatures

Motivation

- Desire for obtaining accurate, non-bias, and less time-con suming signatures
  - No systematic approach for signature extraction
  - Avoiding tedious and exhaustive search for signatures
  - Dealing with thousands of applications (e.g., P2P)
- Validation requirements
  - Cross validation with classification algorithms themselves
  - Relying on signature eventually for ground truth
- No concrete set of signatures
  - Proposing a sharing data set for signature list
  - Industry: Ipoque, Sandvine, Procera, and etc.
- An extra question in mind
  - What about encrypted traffic applications?

Related Work

- **POSTECH's work on classification**
  - Flow Relationship Mapping (FRM) [M.Kim, '04]
  - Hybrid approach between flow relations and signature matching [Won '06]
  - ML-based attempts - papers in Korean
- **P2P traffic identification using signature**
  - Packet inspection [Gummandi '03, Karagiannis '04]
  - Protocol analysis [Sen '04]
    - Accurate but only for open protocols
- **Automated worm signature generation [Kim '04, Singh '04, Singh '05]**
  - Sliding-window algorithms [Scheirer '05]

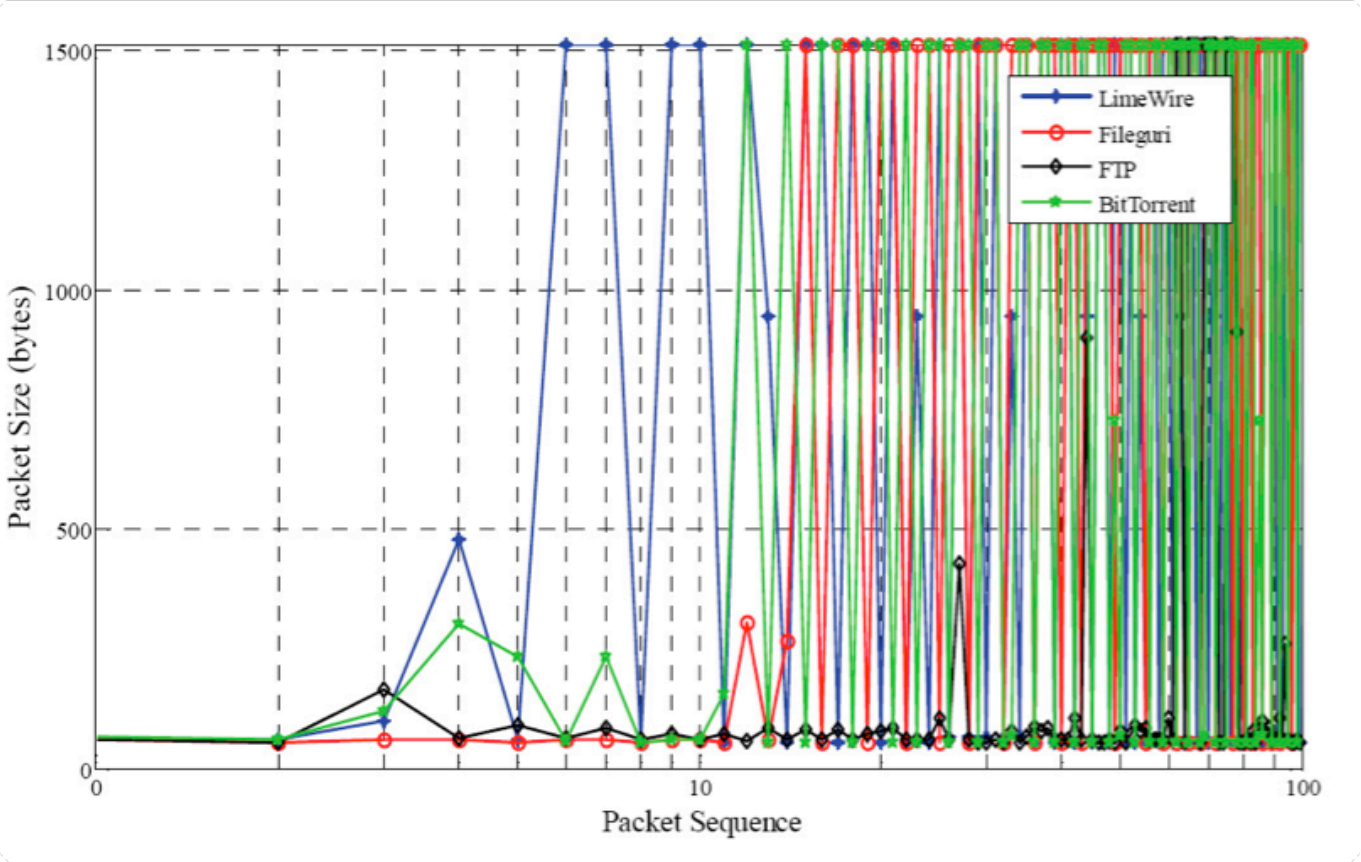
LASER

- We proposed a LCS-based Application Signature ExtRaction technique - LASER [NOMS '08]
  - Longest Common Subsequence algorithm [Cormen '01]
  - Avoiding exhaustive search for signatures
  - Extracting candidate signature for later analysis

Constraints of LASER (1/2)

- **Number of packets per flow**
  - A concrete signature exists in the initial few packets of the flow [Sen '04]
  - Tentative packet grouping
- **Minimum substring length**
  - Signature is simply a sequence of substrings
  - Length of substring reflect the significance as a signature
  - To avoid trivial signatures
    - e.g. '/' in HTTP protocol
- **Packet size**
  - Size differs due to purpose of the packets (signaling or download)
  - Packet size in a close range infers higher chance for valid signatures

Constraints of LASER (2/2)



- **Example: LimeWire**
  - **Signaling - avg. 390bytes, Downloading - 1460bytes**
  - **Avoiding unnecessary packet comparisons**
  - **Reducing garbage characters from the generated signature**

LASER Pseudocode

```
1: procedure Signature_Generation ()
2:   Flow_Pool {F1[]…Fx[]} ← Sanitized_packet_collector
3:    F1[] ← Iterate, packet dump for Flow 1
4:    F2[] ← Iterate, packet dump for Flow 2
5:    while i from 0 to #_packet_constraint do
6:      while j from 0 to #_packet_constraint do
7:        if |F1[i].packet_size - F2[j].packet_size| < threshold
8:          result_LCS ← LASER (F1[i], F2[j])
9:          LCS_Pool {} ← Append result_LCS, end if
10:       j++, end while
11:   i++, end while
12:   S ←select the longest from LCS_Pool
13:   while i from 0 to number of rest flows of Flow_Pool do
14:     Fi ← select one from the rest of Flow_Pool
15:     result_LCS ← LASER (S, Fi)
16:     S ← select the longest from result_LCS
17:     i++, end while, end while
18:   return S
```

```
19: procedure LASER (PacketA[1…m], PacketB[1…n])
20:   PacketA [m…1] ← Reverse byte stream
21:   PacketB [n…1] ← Reverse byte stream
22:   Matrix [m][n]
23:   while i from 0 to m do
24:     while j from 0 n do
25:       if i = 0 or j = 0, then Matrix [i][j] ← 0
26:       else if PacketA [i] = PacketB [j], then
27:         Matrix [i][j] ← 'Diagonal'
28:       else if Matrix[i][j] != p[i][j-1], then
29:         Matrix[i][j] ← 'Up'
30:       else Matrix[i][j] ← 'Left', end while
31:   end while
32:   i ← m-1; j ← n-1 //Tracking
33:   while Matrix[i][j] != 0 do
34:     if Matrix[i][j] = 'Left', then j--
35:       else if Matrix[i][j] = 'Up', then i--
36:       else if Matrix[i][j] = 'Diagonal', then do
37:         Substring ← Append PacketA[i]
38:     if Matrix[i-1][j-1] != 'Diagonal', then
39:       Substring ← Append special break point character (e.g.'/')
40:       i--; j--, end while
41:   while tokenizing substring based on break point do
42:   if token_length > minimum_substring_length_constraint
43:   then, result_LCS ← Append token_substring, end while
44: return result_LCS
```

Applying Constraints

```
3:    F1[] ← Iterate, packet dump for Flow 1
4:    F2[] ← Iterate, packet dump for Flow 2
5:    while i from 0 to #_packet_constraint do
6:      while j from 0 to #_packet_constraint do
7:        if |F1[i].packet_size - F2[j].packet_size| < threshold
8:          result_LCS ← LASER (F1[i], F2[j])
```

- Number of packets per flow constraint
- Packet size constraint
- F1 and F2 are used as input to LASER

Refining Process

```
12:  S ←select the longest from LCS_Pool
13:  while i from 0 to number of rest flows of Flow_Pool do
14:    Fi ← select one from the rest of Flow_Pool
15:    result_LCS ← LASER (S, Fi)
16:    S ← select the longest from result_LCS
17:    i++, end while, end while
```

- **Simply put,**

Candidate_signature_1 = Signature (Flow 1, Flow 2)
Candidate_signature_2 = Signature (Flow 3, Candidate_signature_1)
…
Candidate signature_n = Signature (Flow n+1, Candidate_signature_n-1)

If Candidate_signature_n = Candidate signature_n-1
For the certain iteration counts then
Candidate_signature_n is the final signature

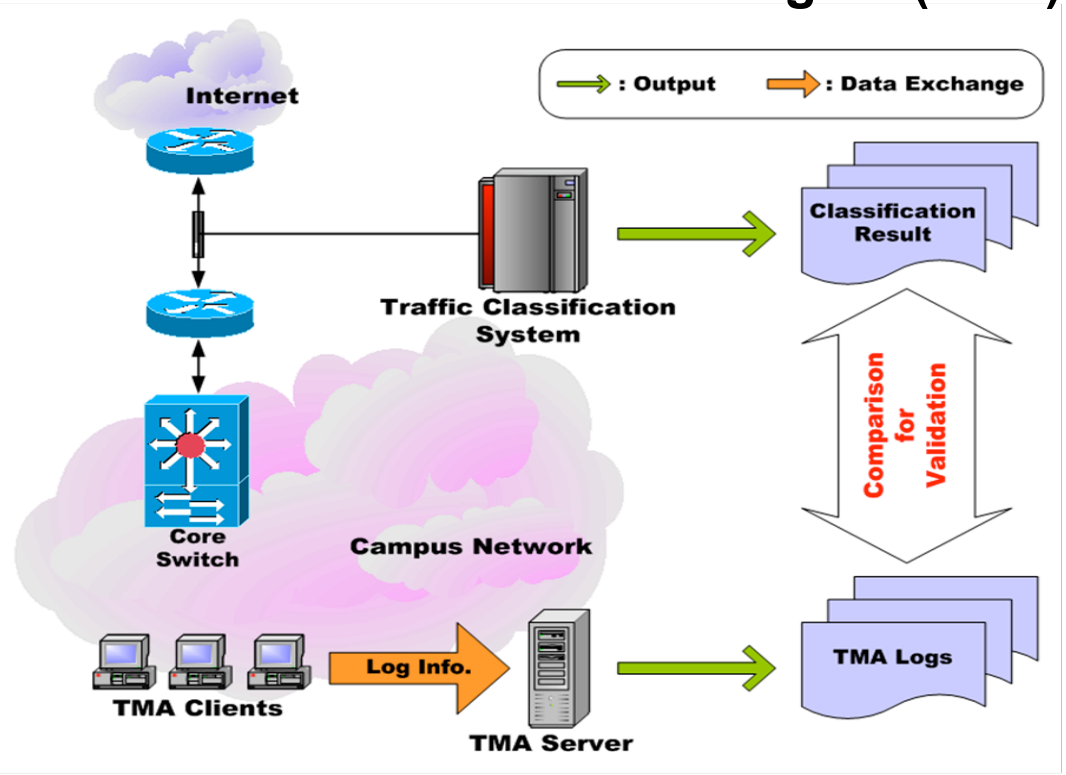
Signatures by LASER

LimeWire	Sequence of 10 substrings - ”LimeWire”, ”Content-Type:”, ”Content-Length:”, ”X-Gnutella-Content-URN”, ”run:sha:1”, ”XAlt”, ”X-Falt”, ”X-Create-Time:”, ”X-Features:”, ”X-Thex-URI”
BitTorrent	Sequence of 1 substring- “0x13BitTorrent protocol”
Fileguri	Sequence of 6 substrings- “HTTP”, “Freechal P2P”, “User-Type:”, “P2PErrorCode:”, “Content-Length:”, “Content-Type:”, “Last-Modified”

- **Choice of P2P applications for early evaluation**
- **Signature extraction from encrypted traffic: Skype v3.0**
  - **No signature was found yet**
  - **The signatures of v1.5 and v2.0 [Ehlert ’06] were not valid anymore**

Classification with Absolute Ground Truth

- **Validation approaches**
  - Cross match with known signatures
  - Cross validation with other classification method
  - Cross validation with ground truth set
- **Agent-based log collection**
  - Traffic Measurement Agent (TMA)



Choice of Classification Algorithm

VS

Absolute Ground Truth

- Text log data with process name, timestamp, IP, port information
- 100% accuracy

Automated Signature Generation System

- **LASER agent**
  - Signature extraction of on-going application in PC
  - Reporting to the collecting server periodically
  - MSDN functions for process id and name look up
  - Winpcap for packet dump
  - Low CPU load (<5%) and memory consumption

- **Collection server**
  - Aggregating signatures according to process name
  - Filtering process – Applying the LASER algorithm among the collected signatures
    - Removing garbage characters/terms
    - Finding common set among possible candidates

- **Open Signature List: http://dpnm.postech.ac.kr/signature**
  - LASER agent program is available.
  - Providing over 80 pre-searched signatures by exhaustive search and in related literatures
  - Providing a list of automatically generated signatures for comparison

Concluding Remarks

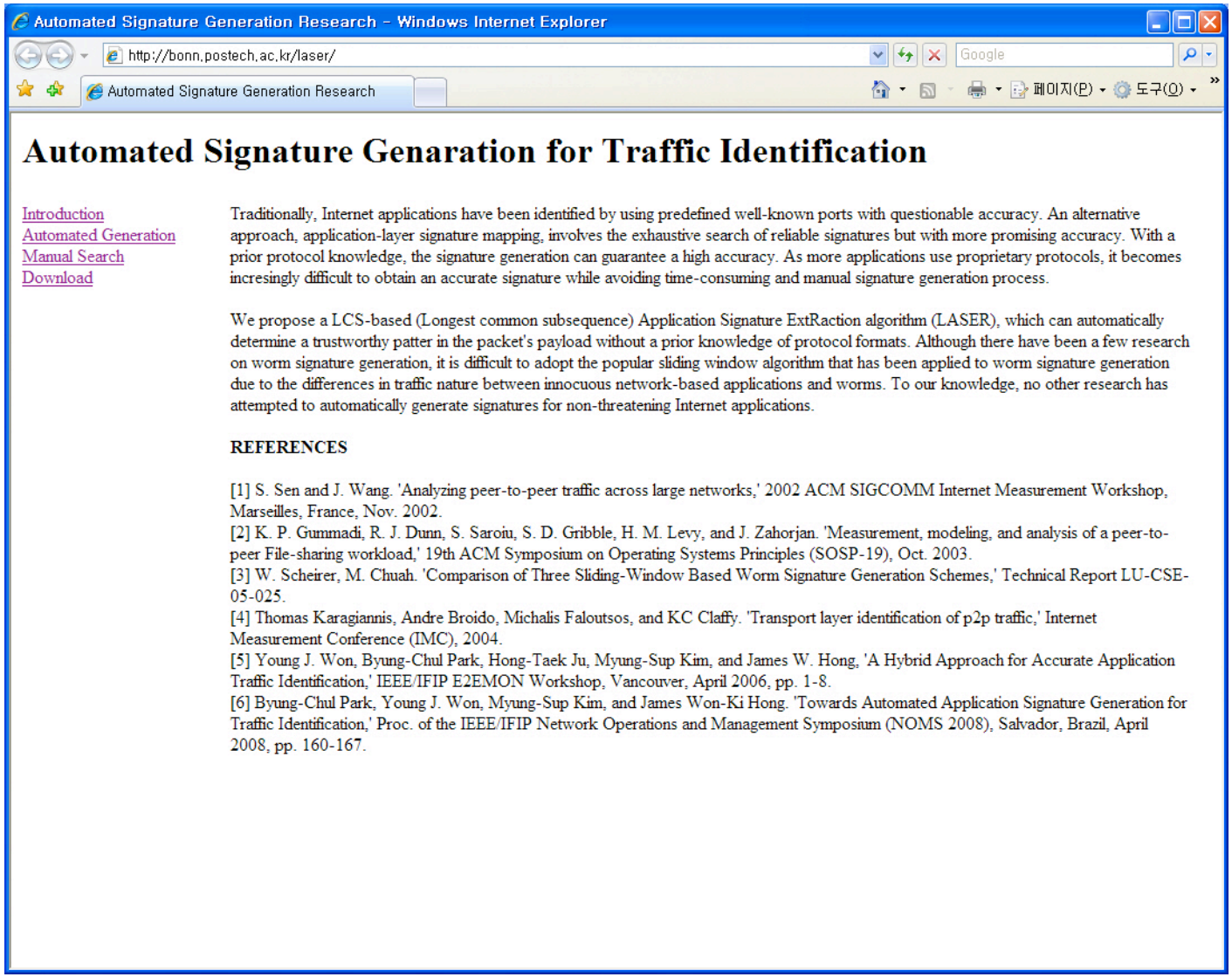
- We have shown
  - POSTECH's efforts on traffic monitoring and analysis
  - Automated signature generation algorithm
- We propose a open repository for signatures
- Future Work
  - Automated rule discovery system
    - Containing not just signatures, but pattern information
  - A new approach to cope with encryption or tunneling tra ffic
  - Signatures for WiMAX applications (Wibro in Pohang)
  - Certifying signatures

Ground Truth vs. LASER

Application	TMA Log (MB)	Classification Result (MB)	False Negative (%)	False Positive (%)
LimeWire	1223.36	1120.35	8.42	0
BitTorrent	4190.07	3754.30	10.40	0
Fileguri	3189.61	3177.17	0.39	0
Others	12482.69	13033.91	-	-
Total			-	-
Overall Accuracy	97.39 %			

- **Accuracy analysis against signature-based classification algorithms**
  - **LASER algorithm achieves 97% accuracy**
- **0% FP: Restricted signature format**
  - **HTTP traffic was not classified as LimeWire or Fileguri**
  - **Cause of FN: HTTP traffic, packets containing flags only**

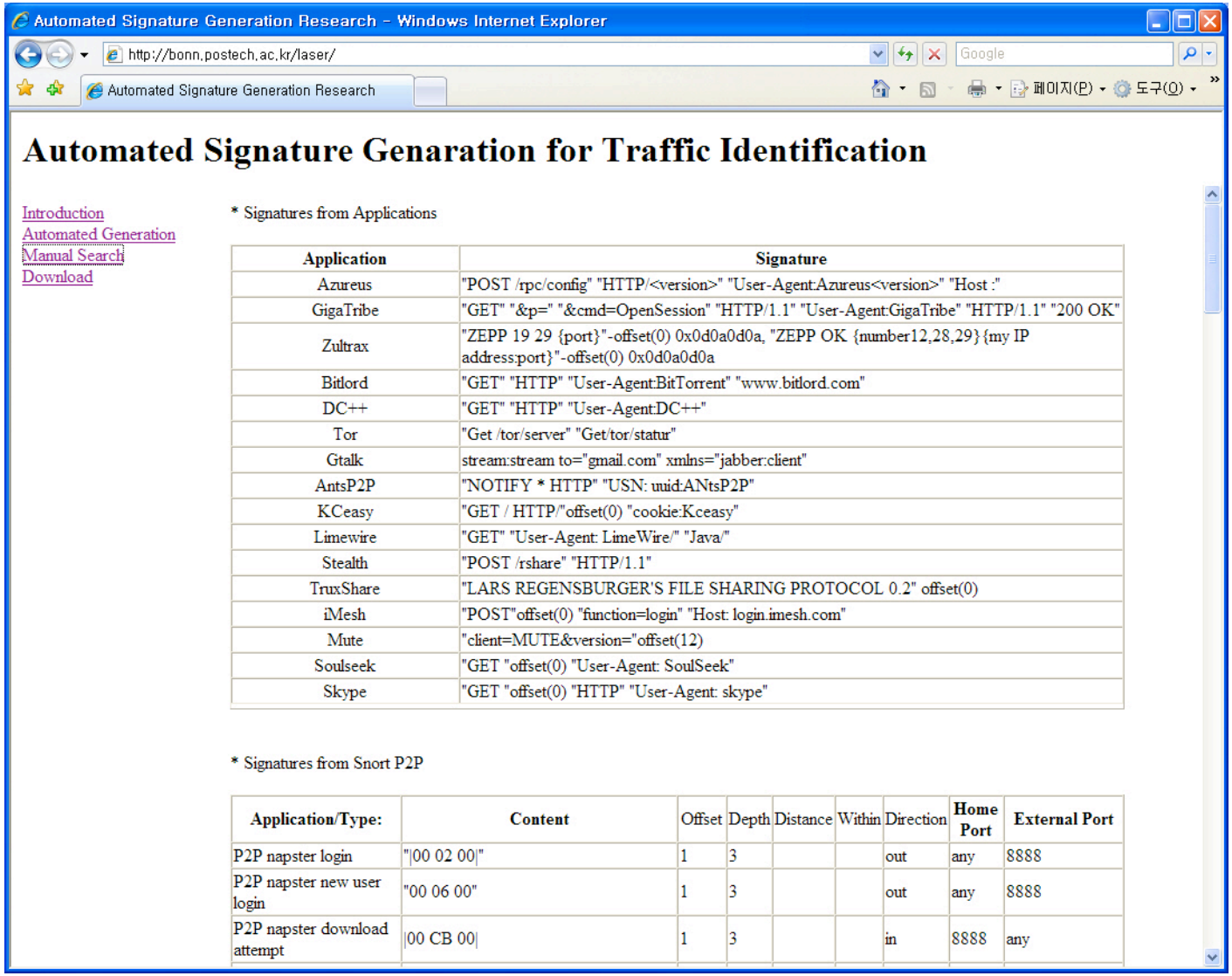
Screenshots (1/3)



Screenshots (2/3)

	HTTP User-Agent GigaTribe Host login gigatribe com Cache-Control no-cache Cookie referer www google utma utmz utmccn organic utmcsr google utmctr gigatribe utmcmd organic whitehat CreationDate	out	2008-08-11 18:03:01
MsnMsgr.Exe	GET messengertabs png HTTP Accept If-Modified-Since Tue Apr GMT If-None-Match User-Agent Mozilla compatible MSIE Windows InfoPath NET CLR Windows Live Messenger Host akimages msn Connection Keep-Alive Cookie MUID EDE	in	2008-08-08 02:24:23
	GET messengertabs png HTTP Accept If-Modified-Since Tue Apr GMT If-None-Match User-Agent Mozilla compatible MSIE Windows InfoPath NET CLR Windows Live Messenger Host akimages msn Connection Keep-Alive Cookie MUID EDE	out	2008-08-08 02:24:23
NATEONMain.exe	GET onlinead SKT T-olympic swf HTTP Accept Accept-Language ko-KR Referer http adimg nate com img swf x-flash-version UA-CPU Accept-Encoding gzip deflate User-Agent Mozilla compatible MSIE Windows InfoPath NET CLR Host img wise aircross com Connection Keep-Alive Cookie EMSID	in	2008-08-07 17:39:34
	GET onlinead SKT T-olympic swf HTTP Accept Accept-Language ko-KR Referer http adimg nate com img swf x-flash-version UA-CPU Accept-Encoding gzip deflate User-Agent Mozilla compatible MSIE Windows InfoPath NET CLR Host img wise aircross com Connection Keep-Alive Cookie EMSID	out	2008-08-07 17:39:34
OUTLOOK.EXE		in	
	+OK POP3 TIMS server ready @postech.ac.kr	out	2008-03-03 19:12:08
Skype.exe		in	2008-08-09 02:51:20
		out	2008-08-09 02:51:20
tor.exe		in	2008-08-12 11:18:49
	www net www net mmQ	out	2008-08-12 11:18:49
Wow.exe	WoW niW RKok WHITEHAT	in	2008-08-11 04:05:22
	WoW niW RKok WHITEHAT	out	2008-08-11 04:05:22
Zultrax.Exe	GET uri-res urn sha NWQGXIA UKUJOGY VKU VDFLT HTTP Node User-Agent LimeWire Connection Keep-Alive Range bytes	in	2008-08-07 00:37:10
	GET uri-res urn sha NWQGXIA UKUJOGY VKU VDFLT HTTP Node User-Agent LimeWire Connection Keep-Alive Range bytes	out	2008-08-07 00:37:10

Screenshots (3/3)



Application	Signature
Azureus	"POST /rpc/config" "HTTP/<version>" "User-Agent:Azureus<version>" "Host :"
GigaTribe	"GET" "&p=" "&cmd=OpenSession" "HTTP/1.1" "User-Agent:GigaTribe" "HTTP/1.1" "200 OK"
Zultrax	"ZEPP 19 29 {port}"-offset(0) 0x0d0a0d0a, "ZEPP OK {number12,28,29}{my IP address:port}"-offset(0) 0x0d0a0d0a
Bitlord	"GET" "HTTP" "User-Agent:BitTorrent" "www.bitlord.com"
DC++	"GET" "HTTP" "User-Agent:DC++"
Tor	"Get /tor/server" "Get/tor/statur"
Gtalk	stream:stream to="gmail.com" xmlns="jabber:client"
AntsP2P	"NOTIFY * HTTP" "USN: uuid:ANtsP2P"
KCeasy	"GET / HTTP/"offset(0) "cookie:Kceasy"
Limewire	"GET" "User-Agent: LimeWire/" "Java/"
Stealth	"POST /rshare" "HTTP/1.1"
TruxShare	"LARS REGENSBURGER'S FILE SHARING PROTOCOL 0.2" offset(0)
iMesh	"POST"offset(0) "function=login" "Host: login.imesh.com"
Mute	"client=MUTE&version="offset(12)
Soulseek	"GET "offset(0) "User-Agent: SoulSeek"
Skype	"GET "offset(0) "HTTP" "User-Agent: skype"

Application/Type:	Content	Offset	Depth	Distance	Within	Direction	Home Port	External Port
P2P napster login	"|00 02 00|"	1	3			out	any	8888
P2P napster new user login	"00 06 00"	1	3			out	any	8888
P2P napster download attempt	|00 CB 00|	1	3			in	8888	any