

Network Security Incident Analysis System for Detecting Large-scale Internet Attacks

Dr. Kenji Rikitake

Security Advancement Group

NICT, Japan

September 6, 2005

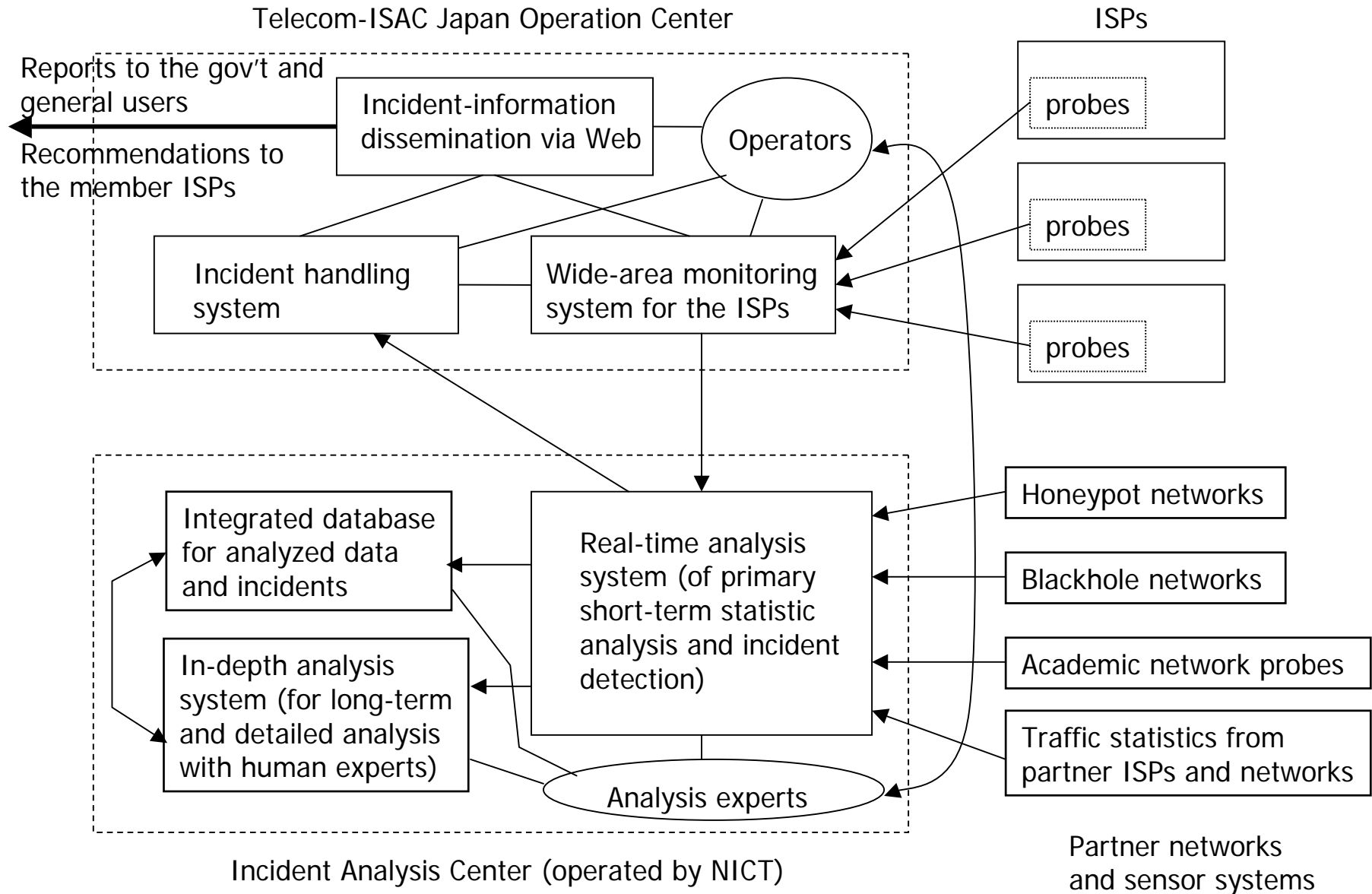
Our goals

- Collaborative monitoring, centralized network security incident analysis and handling among Japanese Internet Service Providers (ISPs), including:
 - real-time analysis for early-warning trends
 - in-depth analysis for detecting new threats
 - recommendation to the ISPs and users
- Protecting National IT infrastructure

Our partners

- Telecom-ISAC Japan
 - Wide-area monitoring with probes on ISPs
 - Incident handling with contingency plans
 - Clearing house of incident info for ISPs
- Internet Security research communities
 - Academic network administrators
 - Virus and malware analysis experts
 - Datamining and statistics experts

Our project and Telecom-ISAC



Roles of our analysis center

- Real-time monitoring
 - from various kinds of network providers
 - from various types of information sources
 - for detecting precursors ASAP
- Real-time (automated) analysis
- In-depth analysis (with the experts)
- Archiving events for future analyses

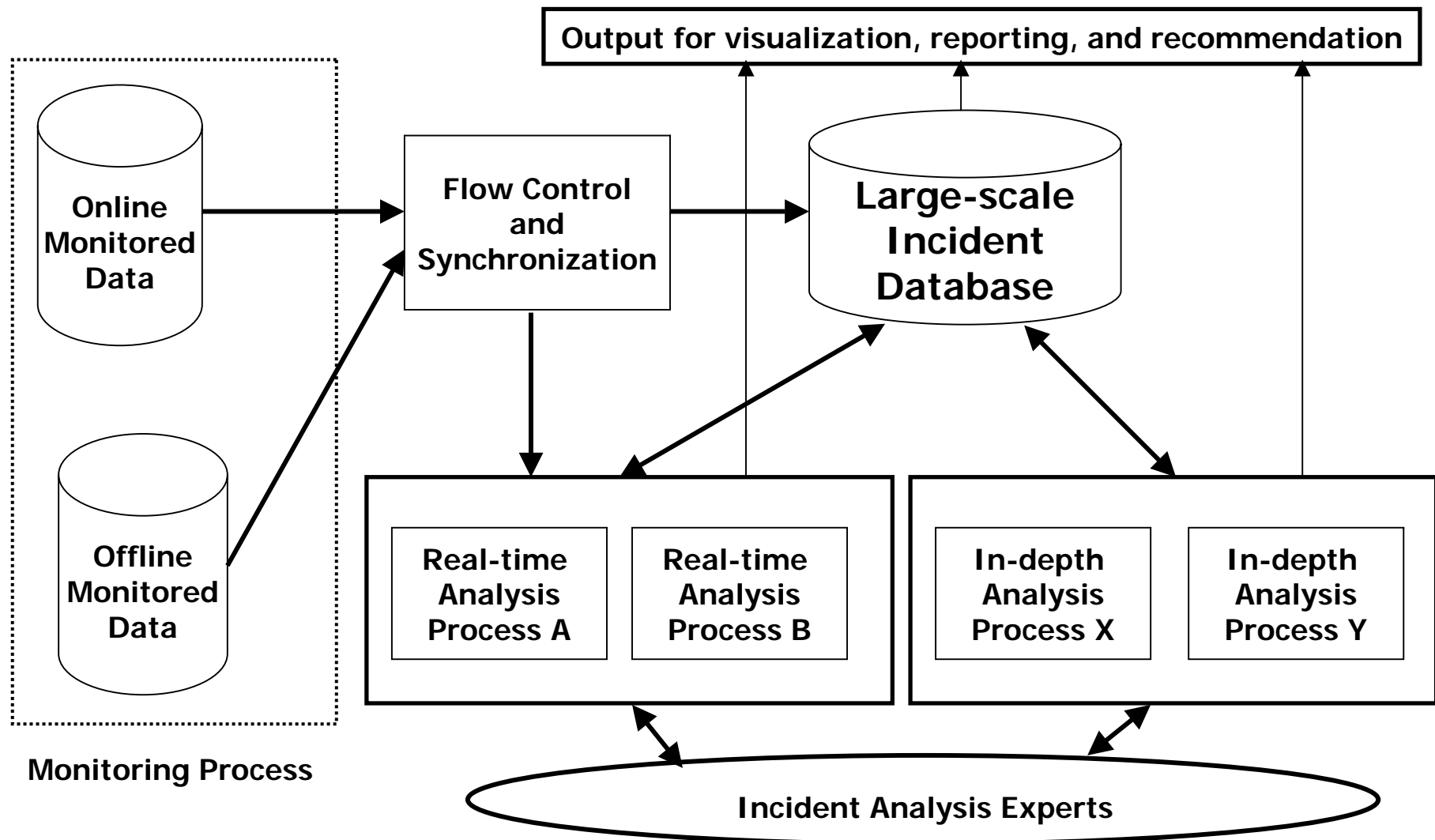
Required functions (1/2)

- Flow control and synchronization
 - of different types of monitored data
 - of different time resolutions and frames
- Parallel analysis of multiple algorithms
 - for finding out clues of new incident trends
 - such as virus or DDoS attack breakouts
- Visualization by multiple methods
 - for helping the experts to find anomalies

Required functions (2/2)

- Large-scale incident database storage
 - for archiving massive (tera-to-petabyte) amount of incident-related data
 - for fast retrieval by the experts and the in-depth analyzing tools
 - for storing non-realtime large statistic data
- Workbench for in-depth analysis
 - behavioral analysis of quarantined viruses

Configuration schematics of the Incident Analysis System

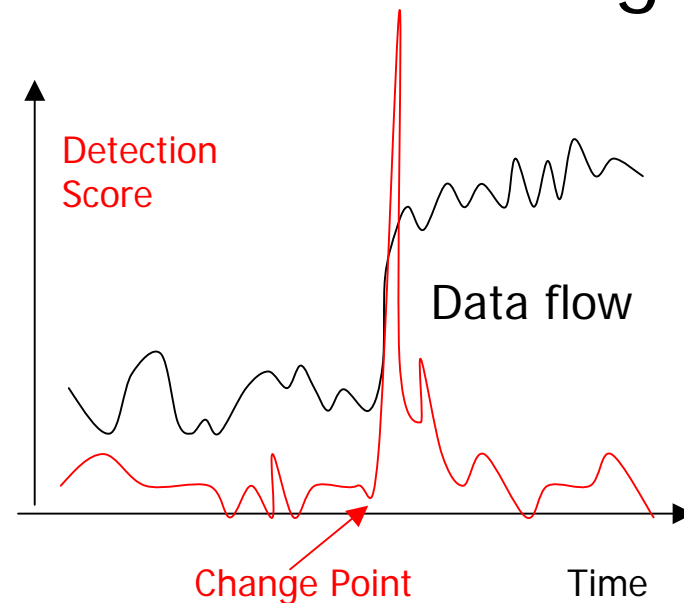


Monitoring networks and the probes

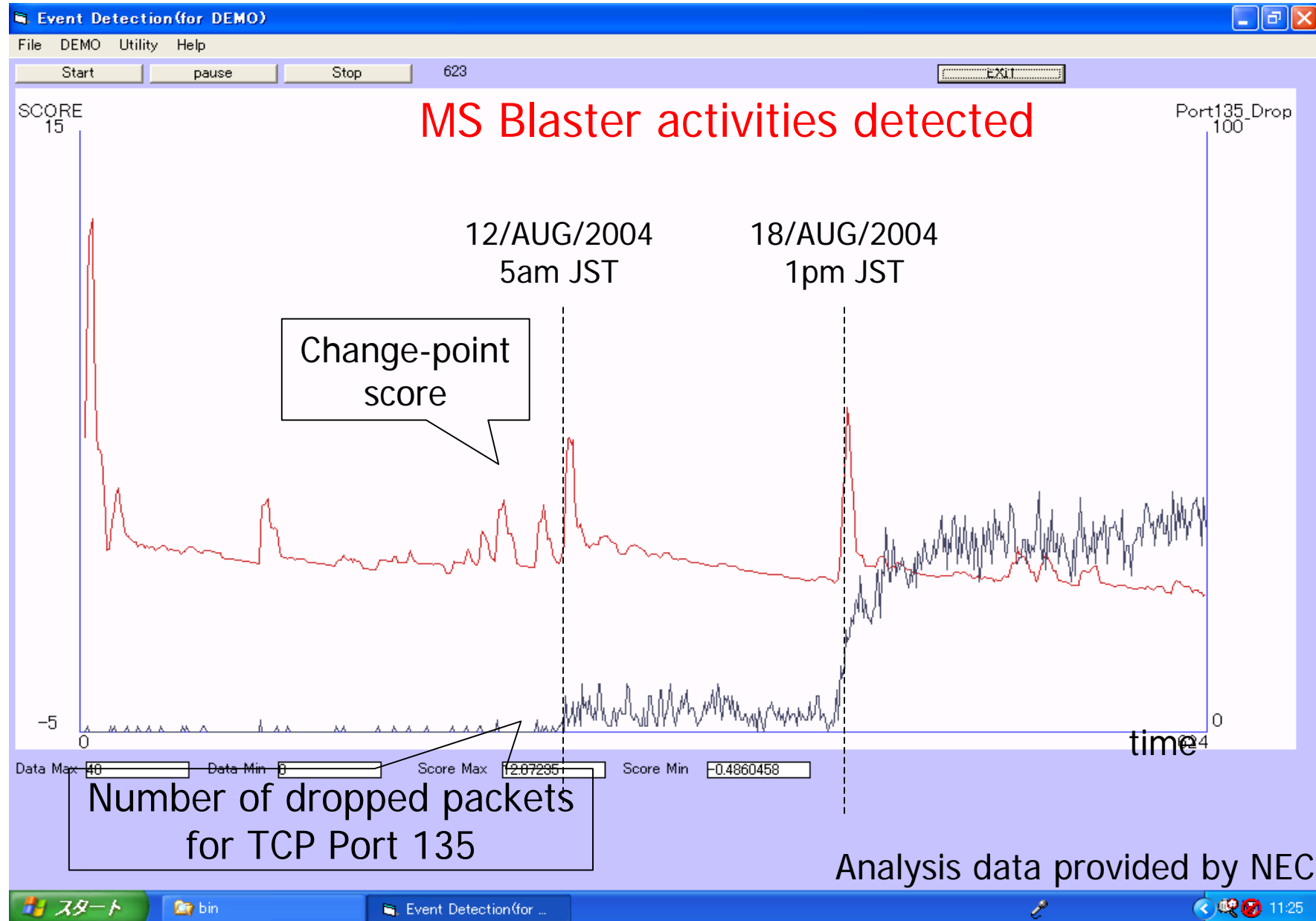
- Monitoring methods
 - Capturing packets (raw and digested)
 - Blackhole networks
 - responding only to ICMP echo requests
 - no actual hosts – only attack packets coming
 - TCP first-client-packet monitor
 - sending a dummy ACK to a SYN request
 - Effective to obtain HTTP methods for attacks
- Traffic/alert logs (syslog, IDS logs)

A real-time analysis method example: change-point detection

- Detecting timing of rapid change of a time-variant data flow
- Faster than repetitive statistical testings
 - Fast real-time learning
 - Adaptive to long-term change
 - Fast detection
 - Low false-alarm rate
 - Applicable to DDoS by detecting rapid quantitative change of traffics



A change-point analysis example



Other candidate algorithms for the real-time traffic analysis

- Rare-ratio analysis
 - determining how *rare* an event is, by using the standard/Gaussian distribution model
- Differential analysis
 - comparing event rate difference between short-term and long-term time frames
- Those analyses are effective for comparing logs of multiple IDSes of different network traffic characteristics

An example of in-depth analysis: DDoS attacks on a well-known site

- The virus generates simultaneous HTTP requests on specific days of month
- The attacked site can no longer serve normal HTTP requests
- In-depth analysis performed by our engineers
 - Using actual traffics captured at the victim server
 - With cooperation of Telecom-ISAC and OCN (ISP of NTT in Japan) twice on August 2004 and August 2005

In-depth DDoS analysis summary (1/2)

- Preprocessing
 - per-minute log of captured data
 - making digests of per-minute logs
 - discarding unrelated payload contents
 - preserving necessary data for analysis
 - reducing the amount of data to process
 - making access history of hosts
 - for each IP source address

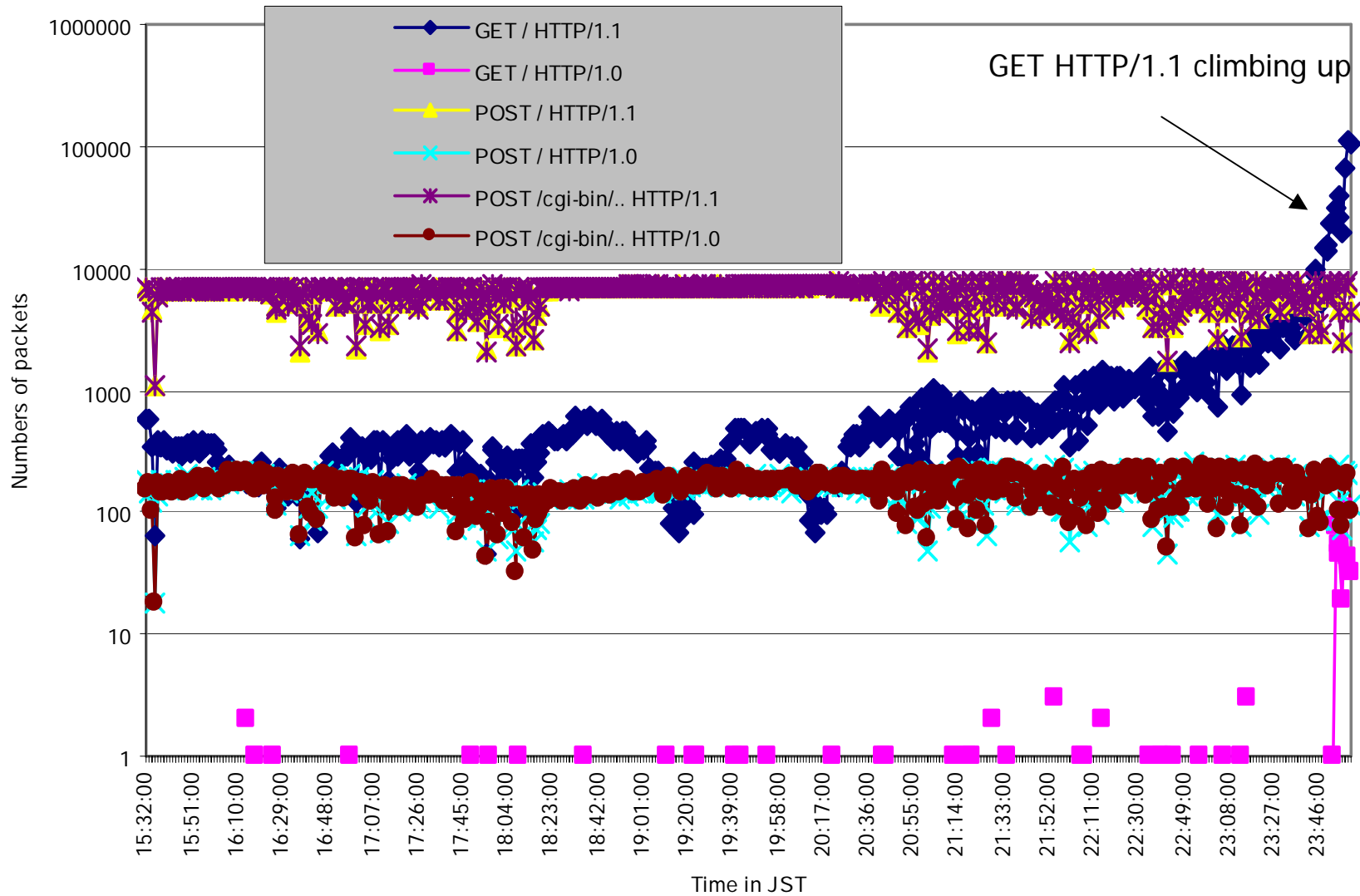
In-depth DDoS analysis summary (2/2)

- Making per-host attack activity ranking
 - based on the history of each host
 - using numbers of transmitted bytes, packets, HTTP requests, and session connection time
- Profiling based on HTTP methods
 - per-hour summary for each method sent
- Passive operating system estimation
 - using TCP signatures (p0f)

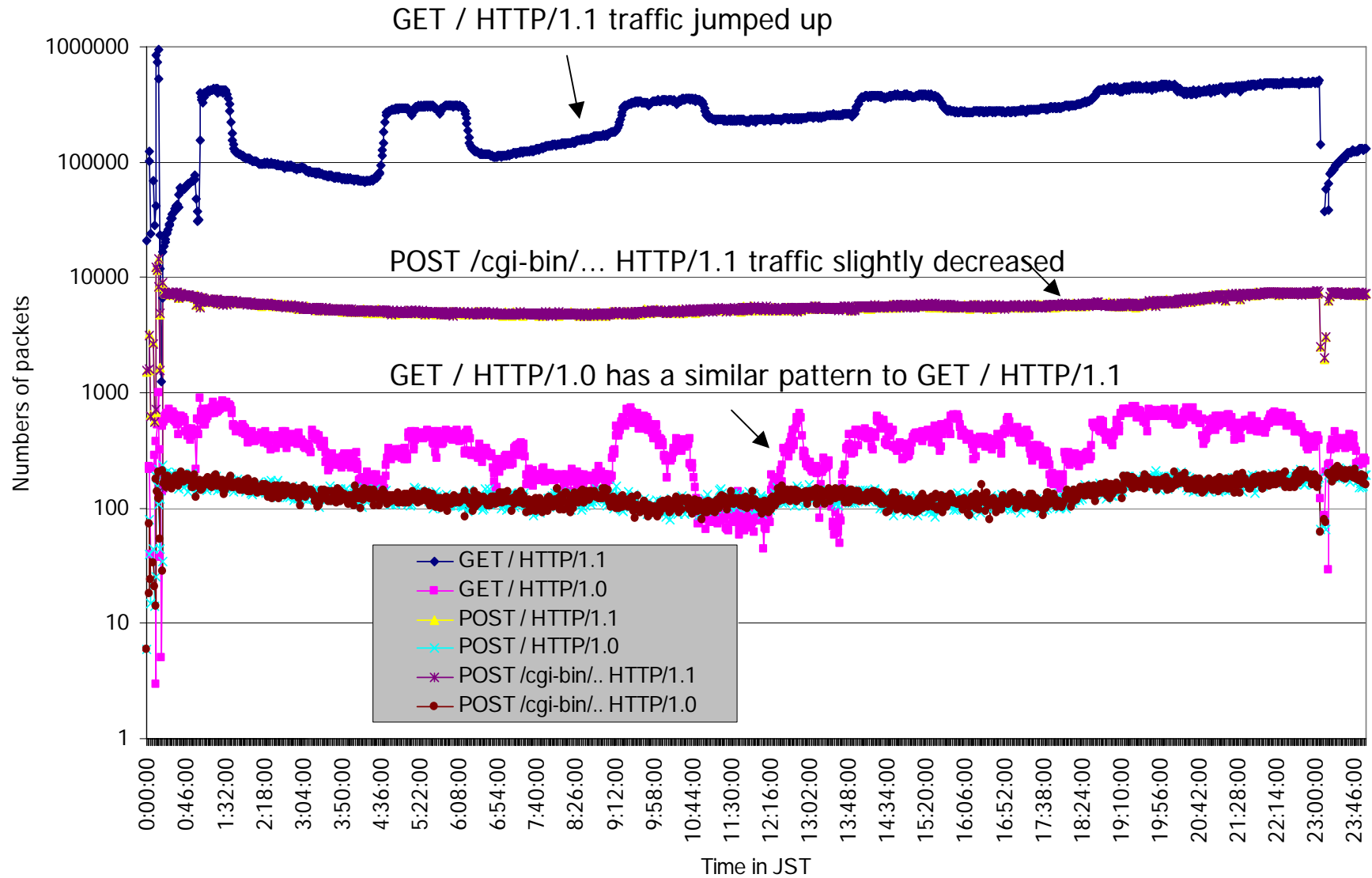
Digested log values and fields of each DDoS attacking packets

- | | | |
|----------------------------|---------------------------|----------------------------|
| + TCP | + UDP | + ICMP |
| - UNIX time() value | - UNIX time() value | - UNIX time() value |
| - Packet length | - Source IP address | - Source IP address |
| - Source IP address | - Destination IP address | - Destination IP address |
| - Destination IP address | - IP header flags | - IP header flags |
| - IP header flags | - "U" for identifying UDP | - "I" for identifying ICMP |
| - TCP header length | - Source port number | - Type |
| - "T" for identifying TCP | - Destination port number | - Code |
| - Source port number | - UDP payload length | - ICMP payload length |
| - Destination port number | | |
| - Sequence number | | |
| - Ack number | | |
| - TCP flags | | |
| - TCP payload length | | |
| - HTTP method (if existed) | | |

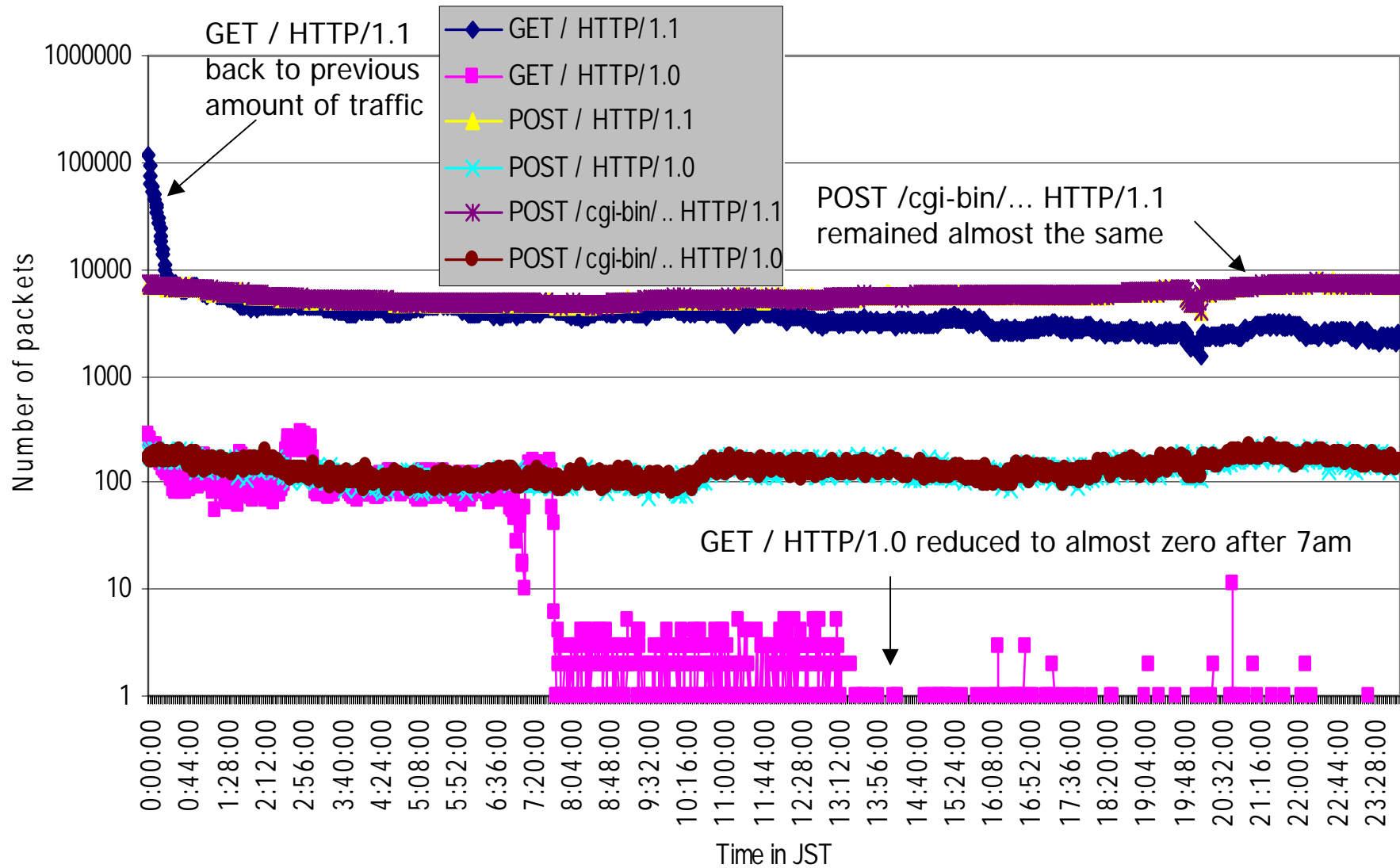
DDoS activity of July 31, 2005



DDoS activity of August 1, 2005



DDoS activity of August 2, 2005



Operating systems estimated for the DDoS attacking hosts

(The DDoS virus has been known as Windows-specific)

Windows 2000 SP4, XP SP1
Windows 2000 SP2+, XP SP1 (seldom 98 4.10.2222)
Windows XP Pro SP1, 2000 SP3
Windows XP Pro SP1, 2000 SP3 (NAT!)
Windows XP/2000 [GENERIC]
Windows 3.11 (Tucows) (firewall!)
OpenBSD 3.0 <i>{note: this is probably a Web proxy server OS}</i>
Windows XP/2000 (RFC1323 no tstamp) [GENERIC]
Windows 2000 SP4, XP SP1 (firewall!)
Windows XP (RFC1323, w+) [GENERIC]

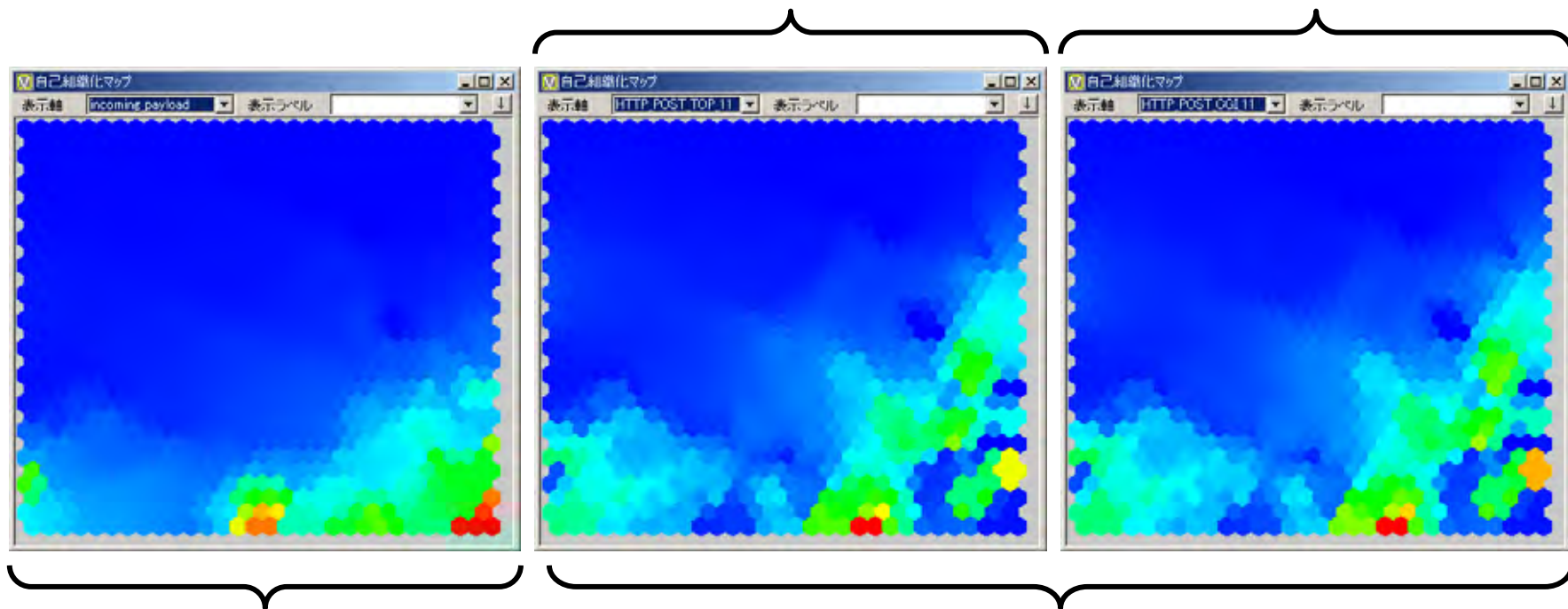
Trends observed from the monitored DDoS activities

- Increased on the day 1 of the month
 - two GET activities
- Steady traffics
 - two POST HTTP/1.1 activities
 - two POST HTTP/1.0 activities
- While the above three trend groups were the same as in 2004, detailed traffic time variance have been changed

Another candidate algorithms for in-depth analysis and visualization: self-organizing maps

- SOMs are effective to detect similarities between different datasets
- The meaning of the resulting figures is non-trivial, though

similar patterns for / and /cgi... POST methods



similarity detected on incoming TCP packets and HTTP POST methods

Schedule and things to do

- Research towards data integration needed
 - More expertise and **research works** needed to understand the relationship between data trends and actual incidents happening on the networks
 - More information sources needed
 - We need to be careful on the legal requirements and rights of the network users (i.e., privacy of traffics)
- Schedule
 - December 2005: 1st beta-version demo of Incident Analysis Center System
 - Production-level operation on 2007