High Speed Search Using Tarari Content Processor in Digital Forensics

By

Jooyoung Lee, Sungkyong Un, Dowon Hong

Presented At

The Digital Forensic Research Conference

DFRWS 2008 USA Baltimore, MD (Aug 11th - 13th)

High-speed Search using Tarari Content Processor in Digital Forensics

Jooyoung Lee

joolee@etri.re.kr

IT R&D Global Leader
ETRI

Contents

Overview (1/3)

ETRI

- Requirement for high-tech tools against high-tech crimes has been increasing steadily

- "Speed" is one of the hot issue in DF
  - 500GB HDD costing about $0.18/GB
  - Recent technology implementing areal density of 1 Tb/in^2
  - Plan to commercialize 4 TB HDD for desktop PC by 2011 (Hitach GST)

- It means
  - 14 hours to search1 TB of data with normally used forensic tools
  - "Size" is a serious problem in DF

Overview (2/3)

ETRI

- **Hardware forensic tools on the market**
  - **Evidence cloning, password cracking aiming to acceleration**

	Forensic Tool	Manufacture	Feature
	HardCopyII Shadow 2	Voom technology	- H/W based imaging tools with writing protect - Up to 5.5 GB/min (ATA Drive)
	Instant Recall		- 2nd generation instant recovery tool
	TACC1441	Tableau	- Accelerating password recovery - Attacks for algorithms WinRar, PGP, Winzip by a factor of 6-30 times with PRTK
	T35e		- Write Blocker
	OmniClone Sonix	Logicube	- Hard drive duplication system - at peek rate of 3.5 GB/min (SATA Drive)

Overview (3/3)

ETRI

- **Forensic Search Tools**
  - **Main requirements**
    - **to present all the matching results without missing when an investigator gives a query**
    - **Forensic search needs more time than traditional search because it has to perform bitwise operations on the whole disk in the physical level**
  - **Traditional approach to forensic search**
    - **Stream based search using bitwise comparison**
    - **Index-based search**
    - **Search based on distributed processing using multiple systems**

Our approach

ETRI

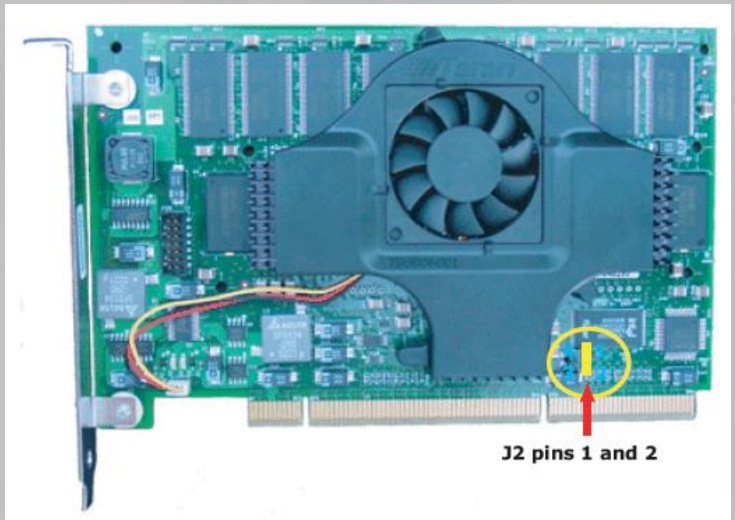
- Design and develop a high-speed search engine with a Tarari CP

- Goals
  - To get high-speed in forensic search
  - To be practical and scalable method
  - To apply hardware-based approach to the field of forensic search and analysis
  - To meet domestic requirements
    - support document files by domestic word processors
    - support Korean and English Language in the documents

- Evaluation
  - Compare performance and advantages to those of a popular forensic tool on the market – Encase

Tarari Content Processor

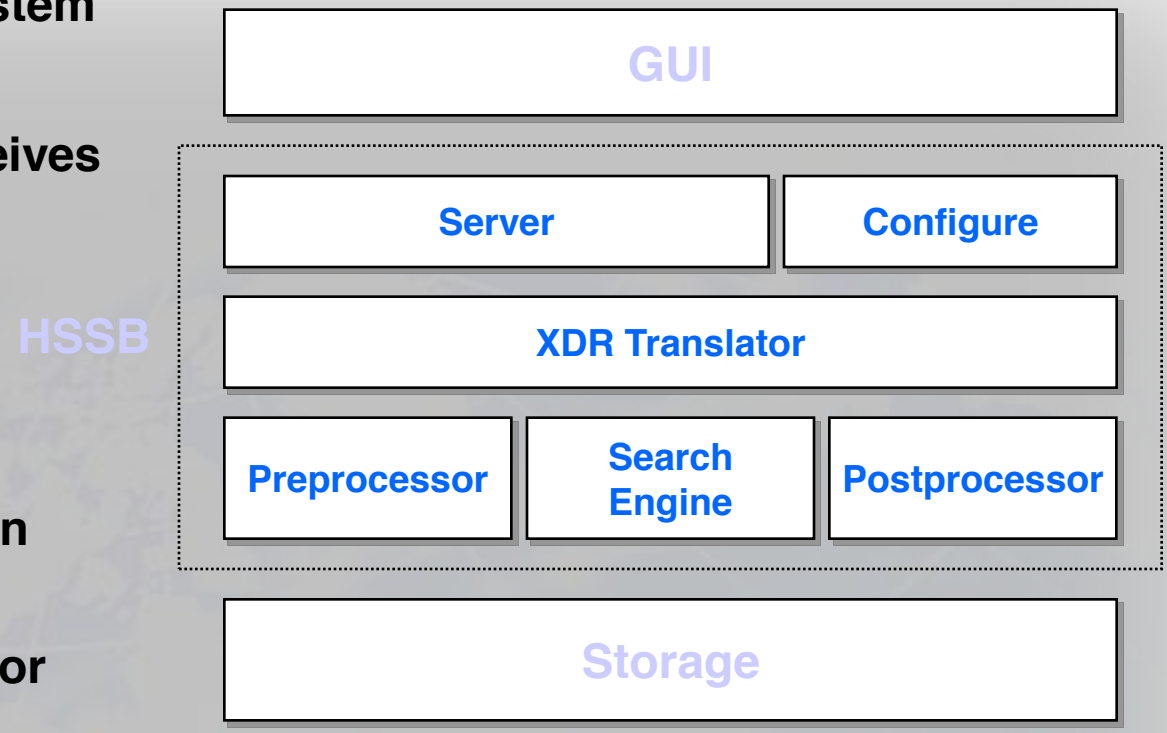
- Allows a user to develop applications that exploit Tarari RegEx Agent which provides an arbitrary content identification and characterization
- Enables applications to analyze fixed or variable patterns in a data stream at speeds up to 1 Gb/s
- Applications
  - Intrusion detection and prevention
  - Anti-SPAM
  - Content filtering
  - MIME and XML parsing
  - Anti-virus
  - Real time message routing
  - Protocol emulation/modeling



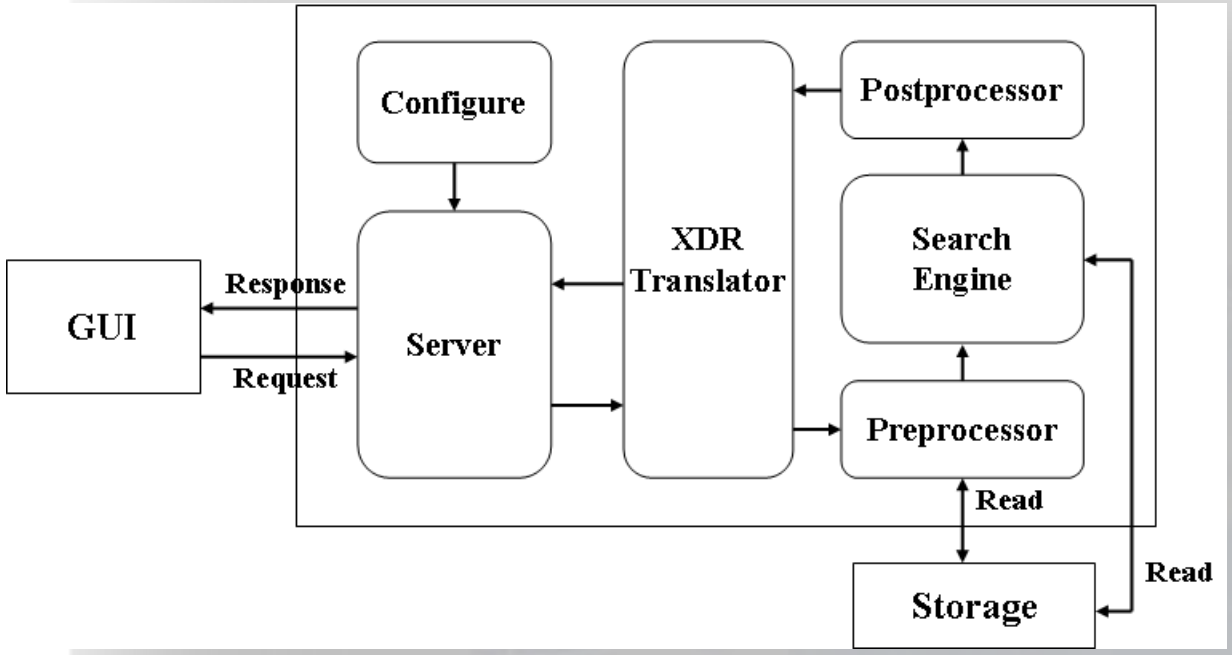
Tarari Content Processing Platform (CPP)

Architecture

- **Client**
  - Installed on a Windows system
  - Presents GUI to a user
  - Sends commands and receives its results
- **HSSB**
  - Server on a Linux system
  - XDR Translator
    - Network communication module
  - Preprocessor/Postprocessor
  - Search Engine
    - Search using Tarari board
- **Storage**
  - NAS connected with NFS



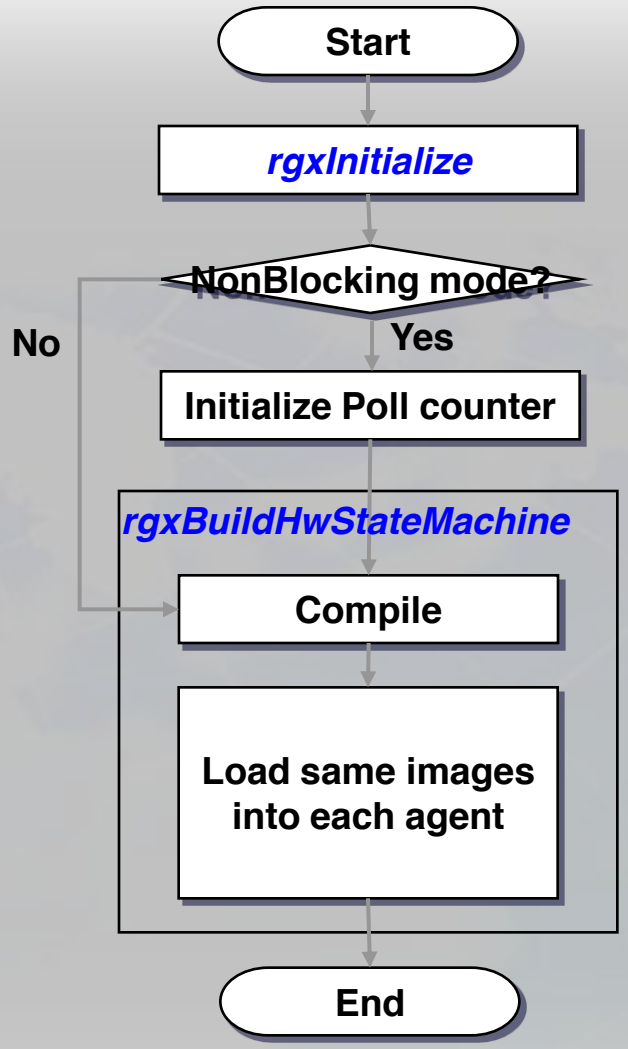
Processing Flow



- **Internal process of search engine**
  - Initialization
  - Compilation
  - Loading
  - Scan

Programming Model Used

- **Load balancing process**
  - Loads keyword(s) or regular expressions to agents
  - 4 agents used
  - Automatic load balancing model used
  - Before loading, the keyword(s) must be compiled into Tarari image by a compiler



Programming Model Used

ETRI

- **Scanning Process**
  - **The forensic image is scanned for keywords by the agents**
  - **A single threaded model used**
  - **When the jobs are completed,**
    - **Searched pattern**
    - **Starting point**
    - **End point**

Scanning Algorithm

```
initialize
getHWConfiguration
read in rexFile
compileAndSave
loadImageToAgent
initializeThread
for total job {
   read in dataFile
   scanNonBlock
}
while(!JobListCompleted) {
   If (jobCompleted) {
      getResults
      printResults
      freeJob
   }
}
freeJobList
deinitializeThread
shutdown
```

System Setup

ETRI

❖ HSSB

Platform	Description
CPU	Intel Xeon 5149 2.33Ghz
Memory	1GB DDR2 667Mhz ECC
Disk	500GB 7.2K rpm SATA
Interface	PCI-X slot
Pattern Matching Board	Tarari Grand Prix 3200
OS	Linux Fedora Core 6
Compiler	gcc

❖ GUI Module

Platform	Description
CPU	Intel Core™2 2.4Ghz
Memory	3GB DDR2
OS	Microsoft Windows XP Professional SP2
Compiler	Visual Studio 2005

Evaluation 1 (1/2)

ETRI

- **Objective**
  - To measure time to take for searching keywords

- **1 GB forensic image made with *dd* command of Linux**

- **Keywords**
  - Single keyword
    - “홍길동”(Korean)
  - Multiple keywords
    - “홍길동”(Korean), “Searching”, “암호기술연구팀”(Korean), “forensic”
  - A Regular expression
    - [0-9][0-9][0-1][0-9][0-3][0-9] *- *[0-4][0-9]][0-9][0-9][0-9][0-9][0-9]

Evaluation 1 (2/2)

ETRI

MB/s(Hit)

	Single Keyword	Multiple Keywords	Regular Expression
Proposed	100.84(18)	97.03(823)	102.58(70)
EnCase	20.14(18)	17.41(711)	17.12(0)

- **Search speed for keywords using the proposed method is faster over 5 times than that of EnCase**
- **The number in the parenthesis indicates the hit number of keywords**
  - **EnCase finds fewer patterns**
  - **It is caused by the fact that EnCase could not extract texts in a structured format by a domestic word processor, Hangul**

Evaluation 2 (1/2)

ETRI

- **Objective**
  - To measure speeds according to size variation of forensic images

- **4 forensic images made with *dd* command of Linux**
  - 274 MBytes
  - 552 MBytes
  - 1.1 GBytes
  - 2.03 Gbytes

- **Keywords**
  - “홍길동”(Korean)
  - [0-9][0-9][0-1][0-9][0-3][0-9] *- *[0-4][0-9]][0-9][0-9][0-9][0-9][0-9]

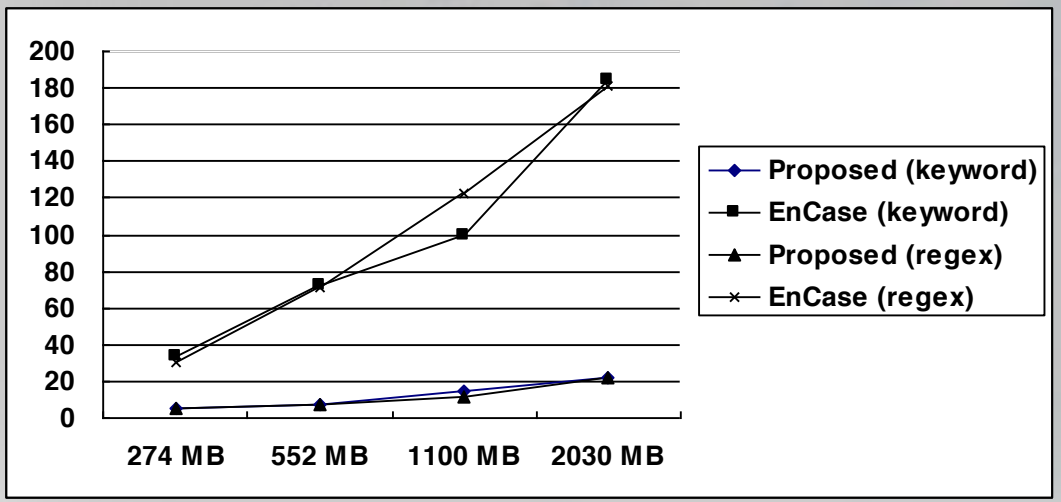
Evaluation 2 (2/2)

ETRI

- **This result shows the proposed method is so scalable that we can apply it to a very large scale of evidence practically**

(sec)

	274 MB	552 MB	1100 MB	2030 MB
Proposed (keyword)	6	7	15	22
EnCase (keyword)	33	72	100	184
Proposed (regex)	5	7	11	22
EnCase (regex)	30	71	122	181



Summary

- High-speed search in physical level
  - Search a string in ADS (Alternative Data Stream) and hidden files
  - Support searching at 100 MB/sec
- Supported file formats
  - MS Office
  - PDF
  - HWP (Domestic word processor popularly used)
  - …
- Encoding
  - ASCII, Unicode, UTF-7, UTF-8
- Query keyword format
  - Text in Korean and English
  - Regular Expressions

Conclusion

- We have proposed a forensic searching method using hardware as a solution to those trends and requirements
- Our results show that search using a Tarari board can be performed over 5 times faster than tools currently on the market
  - same results with even a set of regular expression
- It is feasible and practical approach for getting high speed in search and analysis of digital forensics

Further Works

- **Problem**
  - Over-analysis or misanalysis requiring the investigators to spend time for filtering unnecessary data
- **To research methods to decrease over-analysis or misanalysis rate, keeping recall ratio 100%**
  - Presenting relatively fittest information to the investigator's intention in the front parts of the result list
  - But, required a way to evaluate the satisfaction degree of the investigators
- **Web-based GUI**
  - Allow investigators an access to HSSB remotely for convenience