ForeIndex

Framework for Storage and Indexing of Forensic Data

Marcelo Antonio da Silva
Brazilian Federal Police

Romualdo Pereira
Brazilian Space Agency
Schedule

- Brazilian Federal Police / Brasília University
- Demand for Storage and Indexing in Forensics
- ForeIndex – Workflow
- ForeIndex – Architecture
- Case Study
Brazilian Federal Police

- Forensic Computer Crime Unit – Brasília/DF

ForeIndex – Framework for Distributed Indexing of Forensic Data
Partnership with Universities:

- Brasília University
Forensic Computer Crime Unit - 2010:

- Specialists made 9050 reports;
- Analysis of around 4.6 PB of data on cybercrimes;
- Some cases with seizures hundreds of computers;
- Necessity to analyse data correlated of different medias of the same case.
Demands for Storage and Indexing

- Some cases result in seizure of hundreds of medias;

- Modern forensics tools provide many artifacts for each media analyzed;

- In some cases this is only data, not knowledge;

- Demand to triage and analyze correlated data.
Demands for Storage and Indexing

- Storage and Indexing as the bottleneck in this process;

Case Study:
- 2,274,796 files (482 GB);
- OS: Windows 7 / openSUSE 11.4 (Linux 2.6)
- Hardware: Intel Core-2 Quad, 2.66 GHz, 4 GB RAM
- Average Time to Copy: **12 hours** (NTFS, Ext4)
- Average Time to Index: **26 hours** (Forensics Tools)
ForeIndex

- Framework for storage and indexing distributed of Forensic Data;

- Utilized in 2 cases of the forensics process:
  - After of the data acquisition phase (triage):
    - Minimize the amount to data to analyse.
  - After of the analysis phase (reviewers):
    - Enabling analysis of correlated evidence.
ForeIndex – Frameword for Distributed Indexing of Forensic Data
ForeIndex – WorkFlow (Tools)

- **Data Providing:**
  - Sleuth Kit Scripts;
  - Files of another forensic analysis process.

- **Search Process:**
  - Apache Solr;
  - JSP and Servlets (Smart GWT);
  - Jasper Reports.
ForeIndex – WorkFlow (Tools)

- ForeIndex:
  - Copy and Index Phases;
  - Hadoop Distributed FileSystem (HDFS)
  - Hadoop MapReduce;
  - Lucene Indexer;
  - Tika.
ForeIndex - Architecture

ForeIndex – Frameword for Distributed Indexing of Forensic Data
ForeIndex – Architecture

HDFS Architecture

- Client
  - Metadata Operations
    - Name Node
      - Metadata
        - [Path, File Name, Replicas, Blocks Locations...]
        - /usr/foreindex/data, arg1, 3, ...
  - Block Operations
    - Data Nodes - Rack 1
      - Replica
    - Data Nodes - Rack 2
      - Replica

ForeIndex – Framework for Distributed Indexing of Forensic Data
ForeIndex – Architecture

HDFS Architecture

- HDFS – Features:
  - Streaming data access;
  - Commodity hardware;
  - Namenode and Datanodes;
  - Data Replication;
  - Data Blocks;
  - Data disk failure, heartbeats, re-replication
ForeIndex – Architecture

MapReduce

- MapReduce - Google™:
  - Parallel programming model for data processing;
  - Processing in 2 phases: Map Phase, Reduce Phase;
  - Commodity hardware, fault-tolerant manner;
  - Data input splitted for map tasks processing;
  - Maps output organized and processed for reduce tasks;
  - Maps and Reduces tasks are scheduled and monitored;
  - Compute nodes and datanodes typically are the same.
ForeIndex – Architecture

**MapReduce**

- **Input Data (HDFS)**
- **Map Task**
- **Reduce Task**
- **Output Data (HDFS)**

**Splits**
- [key-1, value-1]
- **Map**
- [key-2, value-2]

**Organize (Sort)**
- [by key-2]

**[key-2, [value-2, value-2, ...]**
- **Merge**
- [key-3, value-3]

ForeIndex – Framework for Distributed Indexing of Forensic Data
ForeIndex – Architecture

MapReduce – Example (WordCount)

File-1: Hello World Bye World
File-2: Hello Hadoop Bye Hadoop

Splits
[File-1, words]
Map
[Hello, 1]
[World, 2]
[Bye, 1]

Organize (Sort)
[by key-2 - Word]

[Hello, [1, 1]]
Merge
[Hello, 2]

Reduce Task

Map Task

Input Data
(HDFS)

Map Task

Output Data
(HDFS)

Splits
[File-2, words]
Map
[Hello, 1]
[Hadoop, 2]
[Bye, 1]

[Hello, 2]
[World, 2]
[Hadoop, 2]
[Bye, 2]
ForeIndex – Architecture

MapReduce – Hadoop Architecture
Hadoop MapReduce – Features:
- HDFS Block Size – Input Split Size;
- Data Locality;
- Job Manage and Monitoring;
- MapReduce functions in many languages;
- Many data types and formats;
- Counters, Sorter, Joins.
ForeIndex - Architecture

ForeIndex – Framework for Distributed Indexing of Forensic Data
Copy Process - Requirements:

- Many files to process;
- Many types of files to process;
- File size less than block size (in average);
- Block Size in HDFS is similar to Cluster Size in NTFS;
- Majority of files can’t be split to be parsed.
ForeIndex – Copy Process

ForeIndex – Framework for Distributed Indexing of Forensic Data
ForeIndex – Copy Process

ForeIndex – Frameword for Distributed Indexing of Forensic Data
ForeIndex – Copy Process

- Features:
  - Distributed copy process;
  - One or more files contained in FileSet <SequenceFile>;
  - FileSet at least with the size of HDFS Block;
  - Namenode more efficiently used;
  - File is not splitted (good for parsing);
  - Other benefits in indexing process.
ForeIndex – Index Process

Distributed File System and Distributed Index Process

- Console Manager
  - Start Index Process
  - Manage HDFS

- ForeIndex Driver
  - Manage Index Job

- Name Node
  - Manage HDFS

- Job Tracker
  - Manage Job

- Data Node
  - Task Tracker

- Index Storage
  - Indexes

ForeIndex – Framework for Distributed Indexing of Forensic Data
ForeIndex – Index Process
ForeIndex – Index Process

- **Features:**
  - Distributed index process;
  - Use Lucene and Tika;
  - Input data are SequenceFiles (FileSet);
  - Sequence Files and Data Locality;
  - Pipeline for read, parse and index the files;
  - The index slices are a functional index;
  - The index slices can be merged.
ForeIndex – Case Study

- **Standalone Test:**
  - 2,274,796 files (482 GB);
  - OS: Windows 7 / openSUSE 11.4 (Linux 2.6)
  - Hardware: Intel Core-2 Quad, 2.66 GHz, 4 GB RAM
  - Average Time to Copy: **12 hours** (NTFS, Ext4)
  - Average Time to Index: **26 hours** (Forensics Tools)
  - Time to Copy in Forensic Cloner: 02:40 (hh:mm)
ForeIndex – Case Study

- ForeIndex Test (2.274.796 files)

  - Configuration:
    - 2.274.796 files (482 GB);
    - Files format: .txt, .xls(s), .xls, .doc(x), .rtf, .msg
    - OS: openSUSE 11.4 (Linux 2.6)
    - Hardware: Intel Core-2 Quad, 2.66 GHz, 4 GB RAM
    - Cluster: 12 Machines (1 Namenode, 1 Job Tracker, 10 Workers [Datanode, TaskTracker]);
    - HDFS Block Size: 64 MB;
    - Local Area Network: 1 Gbps;
ForeIndex – Case Study

- ForeIndex Test (2,274,796 files)
  - Copy Process:
    - Data Source: 2 HDDs SATA-II (no RAID);
    - 4 Maps per Worker = 40 Maps;
    - SequenceFiles created = 206,799;
    - Time to copy = 03:25 (hh:mm)

- Time to Copy in Forensic Cloner: 02:40 (hh:mm)
- Time to Copy in Standalone Test: 12:00 (hh:mm)
ForeIndex – Case Study

- ForeIndex Test (2,274,796 files)
  - Copy Process:
    - Data Source: 2 HDDs SATA-II (RAID-1);
    - 4 Maps per Worker = 40 Maps;
    - SequenceFiles created = 206,799;
    - Time to copy = 01:50 (hh:mm)

- Time to Copy in Forensic Cloner: 02:40 (hh:mm)
- Time to Copy in Standalone Test: 12:00 (hh:mm)
ForeIndex – Case Study

- ForeIndex Test (2,274,796 files)
  - Index Process:
    - 30 Maps, 10 Reducers;
    - SequenceFiles processed = 206,799;
  - Time to Index in Standalone Test = 26:00 (hh:mm)
  - Time to Index in ForeIndex Cluster = 00:25 (hh:mm)
Questions?

Marcelo Antonio da Silva
Brazilian Federal Police
marcelosilva.mas@dpf.gov.br
ForeIndex

Framework for Storage and Indexing of Forensic Data

Marcelo Antonio da Silva
Brazilian Federal Police

Romualdo Pereira
Brazilian Space Agency