Fast Disk Analysis with Random Sampling

Simson L. Garfinkel
Associate Professor, Naval Postgraduate School
CENIC 2010 — March 9, 2010 - 5:06pm
http://domex.nps.edu/deep/
NPS is the Navy’s Research University.

Location: Monterey, CA
Campus Size: 627 acres

Students: 1500
- US Military (All 5 services)
- US Civilian (Scholarship for Service & SMART)
- Foreign Military (30 countries)

Digital Evaluation and Exploitation:
- Research computer forensics.
- Develop “corpora” for use in research & education.
- Identify limitations of current tools & opportunities for improvement.
- http://domex.nps.edu/deep/

“The views expressed in this presentation are those of the author and do not necessarily reflect those of the Department of Defense or the US Government.”
This talk is about a breakthrough in digital forensics.

How to analyze a 1TB drive in 2 minutes.
Instant Drive Analysis with Statistical Sampling
What if US agents encounter a hard drive at a border crossing?

Or a search turns up a room filled with servers?
It takes 3.5 hours to read the contents of a 1TB drive. What can you learn in 1 minute?

<table>
<thead>
<tr>
<th>Minutes</th>
<th>208</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max Data Read</td>
<td>1 TB</td>
<td>4.8 GB</td>
</tr>
</tbody>
</table>

4.8 GB (0.48%) is a tiny fraction of the disk. But 4.8 GB is a lot of data!
Hypothesis: The contents of the disk can be predicted by identifying the contents of randomly chosen sectors.

US elections can be predicted by sampling a few thousand households:

Hard drive contents can be predicted by sampling a few thousand sectors:

The challenge is identifying likely voters.

The challenge is *identifying the content* of the sampled sectors.
We use random sampling; any other approach could be exploited by an adversary.

But sampling has an important limitation...
Data on hard drives divides into three categories:

- **Resident Data**
  - user files
  - email messages
  - [temporary files]

- **Deleted Data**

- **No Data**
  - blank sectors
Sampling can distinguish between "zero" and data. It can't distinguish between resident and deleted.
Let's simplify the problem. Can we use statistical sampling to verify wiping?

I bought 2000 hard drives between 1998 and 2006. Most of were not properly wiped.

![Bar chart showing data in the file system (level 0), data not in the file system (level 2 and 3), and no data (blocks cleared).]
It should be easy to use random sampling to distinguish a properly cleared disk from one that isn't.

Let’s try reading 10,000 random sectors and see what happens….
We read 10,000 randomly-chosen sectors … and they are all blank
We read 10,000 randomly-chosen sectors … and they are all blank
We read 10,000 randomly-chosen sectors … and they are all blank
We read 10,000 randomly-chosen sectors … and they are all blank.

Chances are good that they are all blank.
Random sampling cannot find a disk with a single sector.

If the disk has 2,000,000,000 blank sectors (0 with data)
  - The sample is identical to the population

If the disk has 1,999,999,999 blank sectors (1 with data)
  - The sample is representative of the population.
  - We will only find that 1 sector using exhaustive search.
What about non-uniform distributions?

If the disk has 1,000,000,000 blank sectors (1,000,000,000 with data)

- The sampled frequency should match the distribution.
- *This is why we use random sampling.*

If the disk has 10,000 blank sectors (1,999,990,000 with data)

— and all these are the sectors that we read???

- We are incredibly unlucky.
- *Somebody has hacked our random number generator!*
Rephrase the problem.
Not a blank disk; a disk with less than 10MB of data.

Sectors on disk: 2,000,000,000 (1TB)
Sectors with data: 20,000 (10 MB)

Chose one sector. Odds of missing the data:
- \( \frac{(2,000,000,000 - 20,000)}{(2,000,000,000)} = 0.99999 \)
- You are very likely to miss one of 20,000 sectors if you pick just one.

Chose a second sector. Odds of missing the data on both tries:
- \( 0.99999 \times \frac{(1,999,999,999-20,000)}{(1,999,999,999)} = .99998 \)
- You are still very likely to miss one of 20,000 sectors if you pick two.

But what if you pick 1000? Or 10,000? Or 100,000?
The more sectors picked, the less likely you are to miss all of the sectors that have non-NULL data.

\[ P(X = 0) = \prod_{i=1}^{n} \frac{((N - (i - 1)) - M)}{(N - (i - 1))} \]  

(5)

<table>
<thead>
<tr>
<th>Sampled sectors</th>
<th>Probability of not finding data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.99999</td>
</tr>
<tr>
<td>100</td>
<td>0.99900</td>
</tr>
<tr>
<td>1000</td>
<td>0.99005</td>
</tr>
<tr>
<td>10,000</td>
<td>0.90484</td>
</tr>
<tr>
<td>100,000</td>
<td>0.36787</td>
</tr>
<tr>
<td>200,000</td>
<td>0.13532</td>
</tr>
<tr>
<td>300,000</td>
<td>0.04978</td>
</tr>
<tr>
<td>400,000</td>
<td>0.01831</td>
</tr>
<tr>
<td>500,000</td>
<td>0.00673</td>
</tr>
</tbody>
</table>

**Table 1:** Probability of not finding any of 10MB of data on a 1TB hard drive for a given number of randomly sampled sectors. Smaller probabilities indicate higher accuracy.

<table>
<thead>
<tr>
<th>Non-null data Sectors</th>
<th>Bytes</th>
<th>Probability of not finding data with 10,000 sampled sectors</th>
</tr>
</thead>
<tbody>
<tr>
<td>20,000</td>
<td>10 MB</td>
<td>0.90484</td>
</tr>
<tr>
<td>100,000</td>
<td>50 MB</td>
<td>0.60652</td>
</tr>
<tr>
<td>200,000</td>
<td>100 MB</td>
<td>0.36786</td>
</tr>
<tr>
<td>300,000</td>
<td>150 MB</td>
<td>0.22310</td>
</tr>
<tr>
<td>400,000</td>
<td>200 MB</td>
<td>0.13531</td>
</tr>
<tr>
<td>500,000</td>
<td>250 MB</td>
<td>0.08206</td>
</tr>
<tr>
<td>600,000</td>
<td>300 MB</td>
<td>0.04976</td>
</tr>
<tr>
<td>700,000</td>
<td>350 MB</td>
<td>0.03018</td>
</tr>
<tr>
<td>1,000,000</td>
<td>500 MB</td>
<td>0.00673</td>
</tr>
</tbody>
</table>

**Table 2:** Probability of not finding various amounts of data when sampling 10,000 disk sectors randomly. Smaller probabilities indicate higher accuracy.
Part 2: Can we classify files based on a sector?

A file 30K consists of 60 sectors:

![Diagram showing file structure]

Many file types have characteristics headers and footer:

<table>
<thead>
<tr>
<th>File Type</th>
<th>Header</th>
<th>Footer</th>
</tr>
</thead>
<tbody>
<tr>
<td>HTML</td>
<td><code>&lt;html&gt;</code></td>
<td><code>&lt;/html&gt;</code></td>
</tr>
<tr>
<td>JPEG</td>
<td><code>&lt;FF&gt;&lt;D8&gt;&lt;FF&gt;&lt;E0&gt;&lt;00&gt;&lt;10&gt;JFIF&lt;00&gt;</code></td>
<td><code>&lt;FF&gt;&lt;D9&gt;</code></td>
</tr>
<tr>
<td>ZIP</td>
<td><code>PK&lt;03&gt;</code></td>
<td><code>&lt;00&gt;&lt;00&gt;&lt;00&gt;&lt;00&gt;</code></td>
</tr>
</tbody>
</table>

But what about the file in the middle?
JPEGs:
Most FFs are followed by 00 due to “byte stuffing.”
This works!
We identify the *content* of a 160GB iPod in 118 seconds.

Identifiable:
- Blank sectors
- JPEGs
- Encrypted data
- HTML

Report:
- Audio Data Reported by iTunes: 2.42GB
- MP3 files reported by file system: 2.39GB
- Estimated MP3 usage:
  - 2.71GB (1.70%) with 5,000 random samples
  - 2.49GB (1.56%) with 10,000 random samples

Sampling took 118 seconds.
Work to date:

Publications:


Work in progress:

- Alex Nelson (PhD Candidate, UCSC) summer project
- Using “Hamming,” our 1100-core cluster for novel SD algorithms.
- Similarity Metric
In summary:
Statistical disk sampling is a new, powerful technique

We can:

- Rapidly determine if a disk was properly wiped.
- Identify the percentage of:
  - JPEGs
  - MPEGs
  - Compressed Data
  - Encrypted or Random Data

- Possible Applications:
  - Healthcare
  - End-of-life auditing
  - Privacy Protection
  - Boarder Crossing

Questions?

slgarfin@nps.edu
http://domex.nps.edu/deep