



## System for the Proactive, Continuous, and Efficient Collection of Digital Forensic Evidence

*By*

**Clay Shields, Ophir Frieder and Mark Maloof**

*Presented At*

The Digital Forensic Research Conference

**DFRWS 2011 USA** New Orleans, LA (Aug 1<sup>st</sup> - 3<sup>rd</sup>)

DFRWS is dedicated to the sharing of knowledge and ideas about digital forensics research. Ever since it organized the first open workshop devoted to digital forensics in 2001, DFRWS continues to bring academics and practitioners together in an informal environment. As a non-profit, volunteer organization, DFRWS sponsors technical working groups, annual conferences and challenges to help drive the direction of research and development.

**<http://dfrws.org>**



# A System for Proactive, Continuous, and Efficient Collection of Digital Evidence

Clay Shields

Ophir Frieder

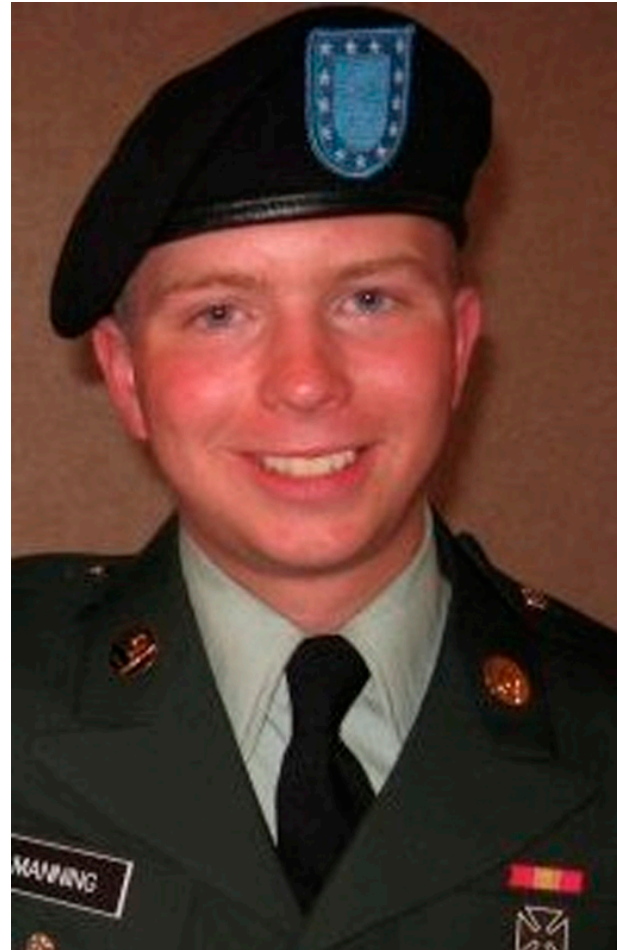
Mark Maloof

Department of Computer Science  
Georgetown University



# Motivation

- How do you find Bradley Manning?  
(Assuming Adrian Lamo doesn't turn him in)
  - Very large network
  - Some documents from that network
- Who had access?
- Who released them?





# Scalable Internal Investigations

- You own the equipment in advance
  - Can plan for investigations in advance
- However
  - Forensic tools were developed for situations where equipment was seized
    - Assume no prior access to equipment
  - Added client-server model allows remote investigation
    - Some agent capabilities
- Huge amounts of information distributed across many machines
  - Where have we seen this?



# PROOFS

## An Information Retrieval Approach

- Google for forensic examiners
  - Save information in advance to make life easier later
  - Centralize it for easy searching
- Parse and record data about file contents
  - When files are unlinked or closed
  - Store information in a scalable manner
- Allows investigation over four axes
  - Time
  - User ID
  - System ID
  - **File contents**



# Forensic Document Signatures

- The information stored is a document *signature*
  - Store in a central database for ease of searching
    - Local storage temporarily when needed
  - Metadata about the file
    - Path, size, owner, MAC dates
  - One or more file *fingerprints*
    - Computed from file content
      - Outside In or similar can be used to extract text
    - Unlike hashes can match across edits
    - Can match across file types



# Fingerprint Creation

- Use a training set of documents
  - Documents that are similar to those sought
  - General documents in correct language
- Extract statistically important terms

$$idf_T = \log \frac{|\# D|}{1 + |\# D_T|}$$

- Create a dictionary of terms within a range of IDFs
  - Low IDFs too common
  - High IDFs too distinct



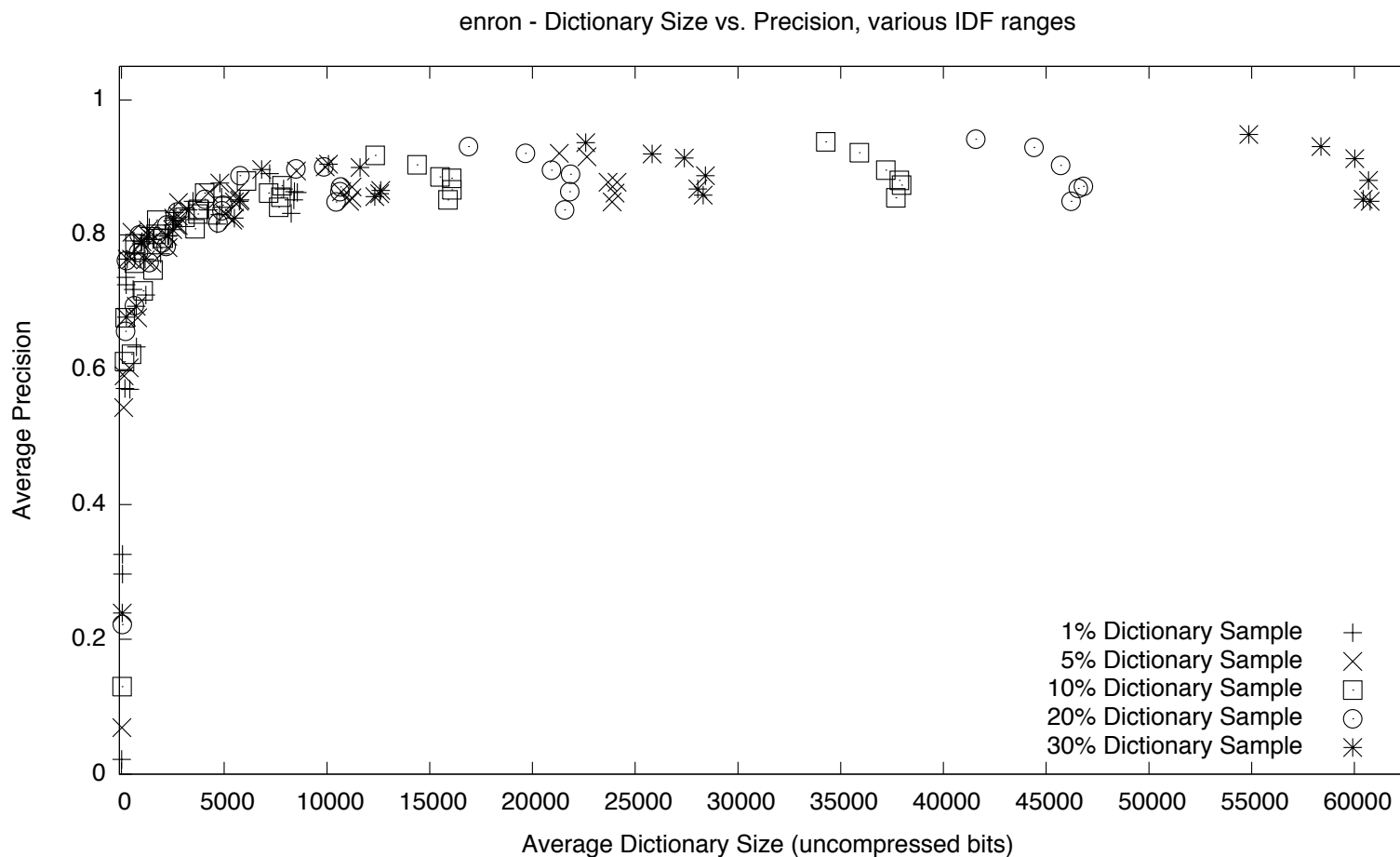
# Bit Vector Fingerprints

- A Bit Vector fingerprint shows which dictionary terms were present in a document
  - Process document
  - For each term in document in dictionary, mark that position
- Generally sparse, highly compressible
- Can add terms to end of vector over time
  - Allows for different dictionary versions
- Robust matching using cosine similarity
  - Parameter allows tradeoff of accuracy



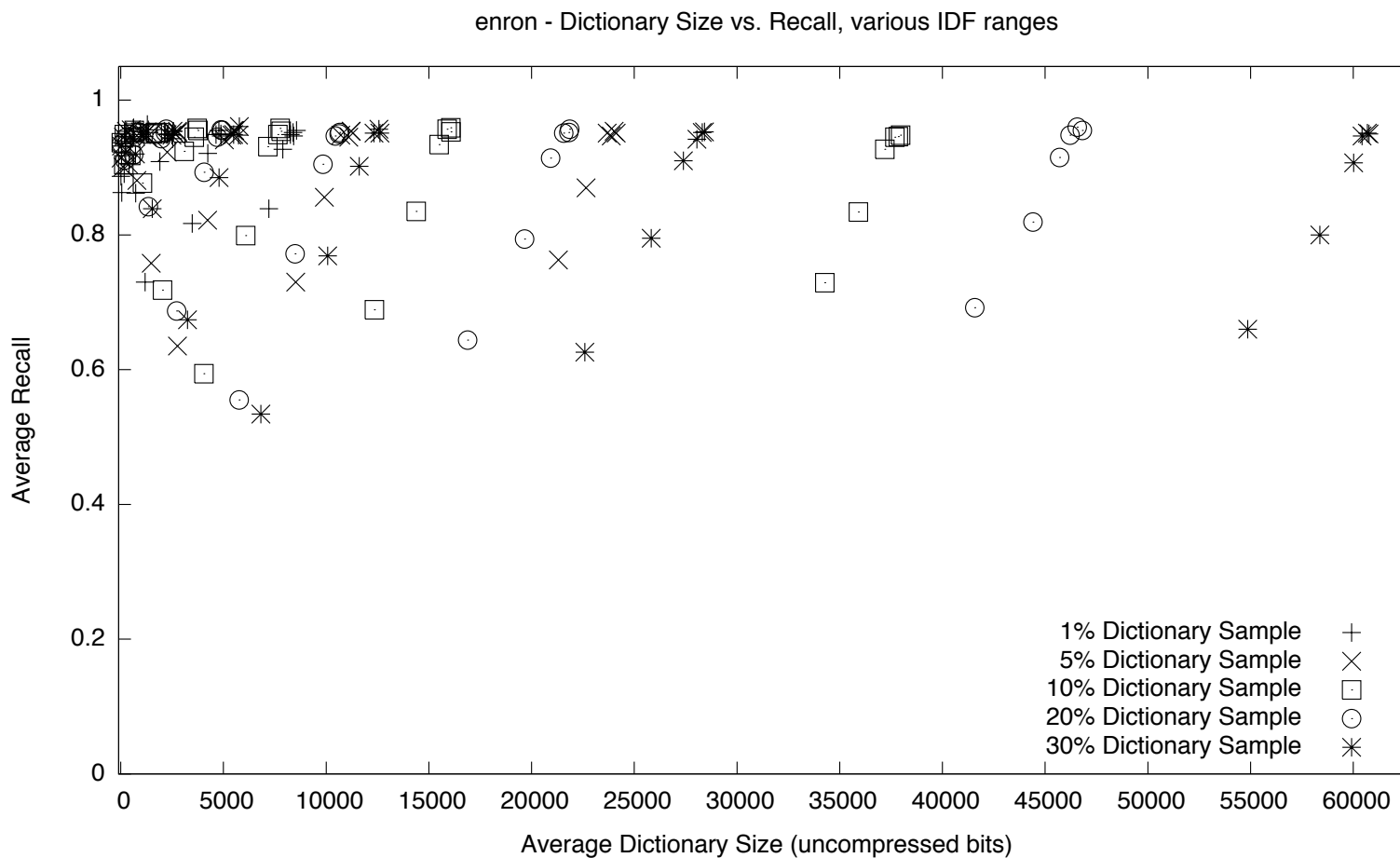


# Bit Vector Size vs. Performance Precision





# Bit Vector Size vs. Performance Recall





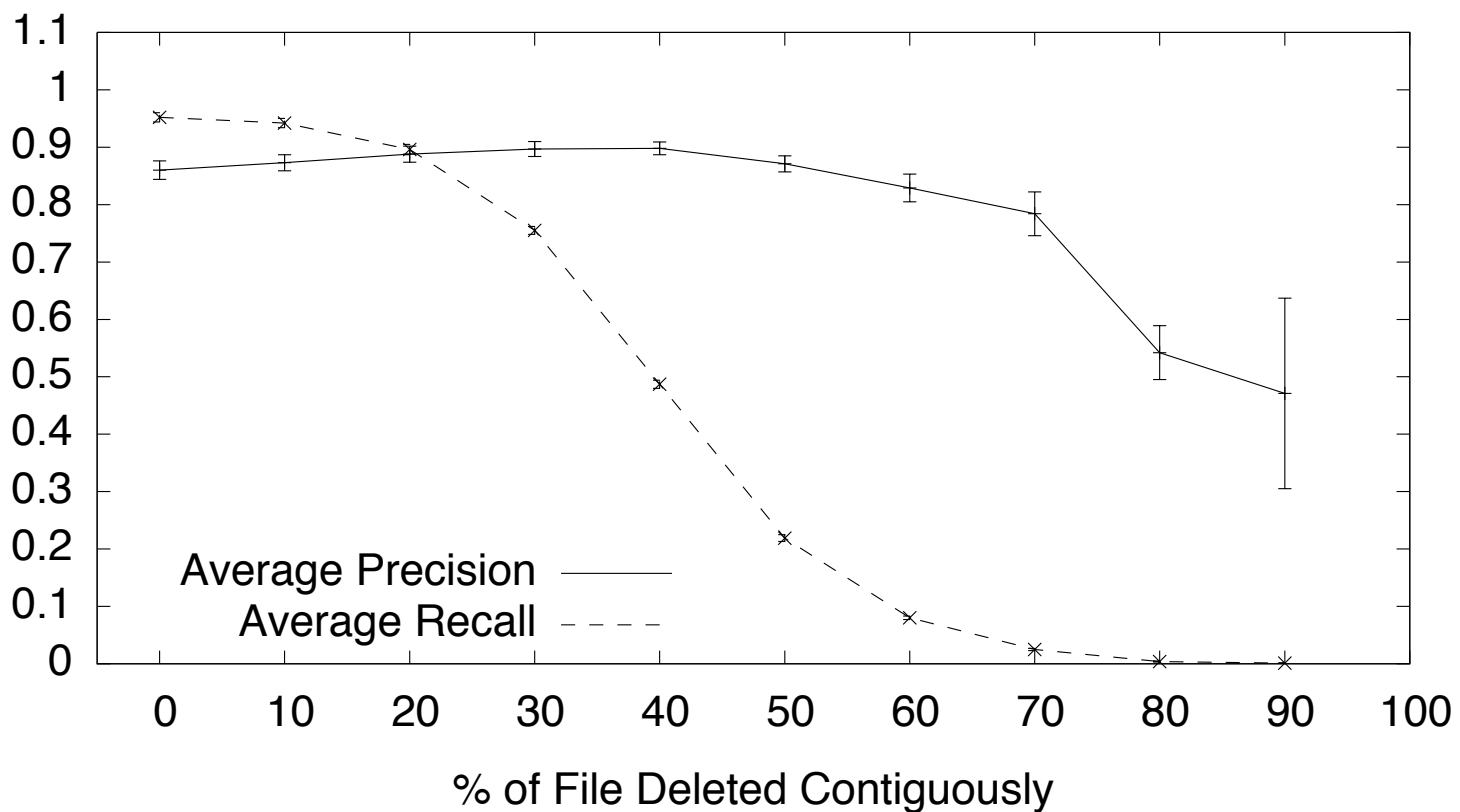
# Performance with Errors

- Forensic recovery often finds file fragments
- Text extraction is prone to errors
  - Formatting
  - OCR
- Simulate these errors in our testing
  - Delete sections, tokens, characters
  - Insert tokens, characters, whitespace
  - Change token, character
  - Automated edits for size



# Bit Vector Fingerprint Robustness

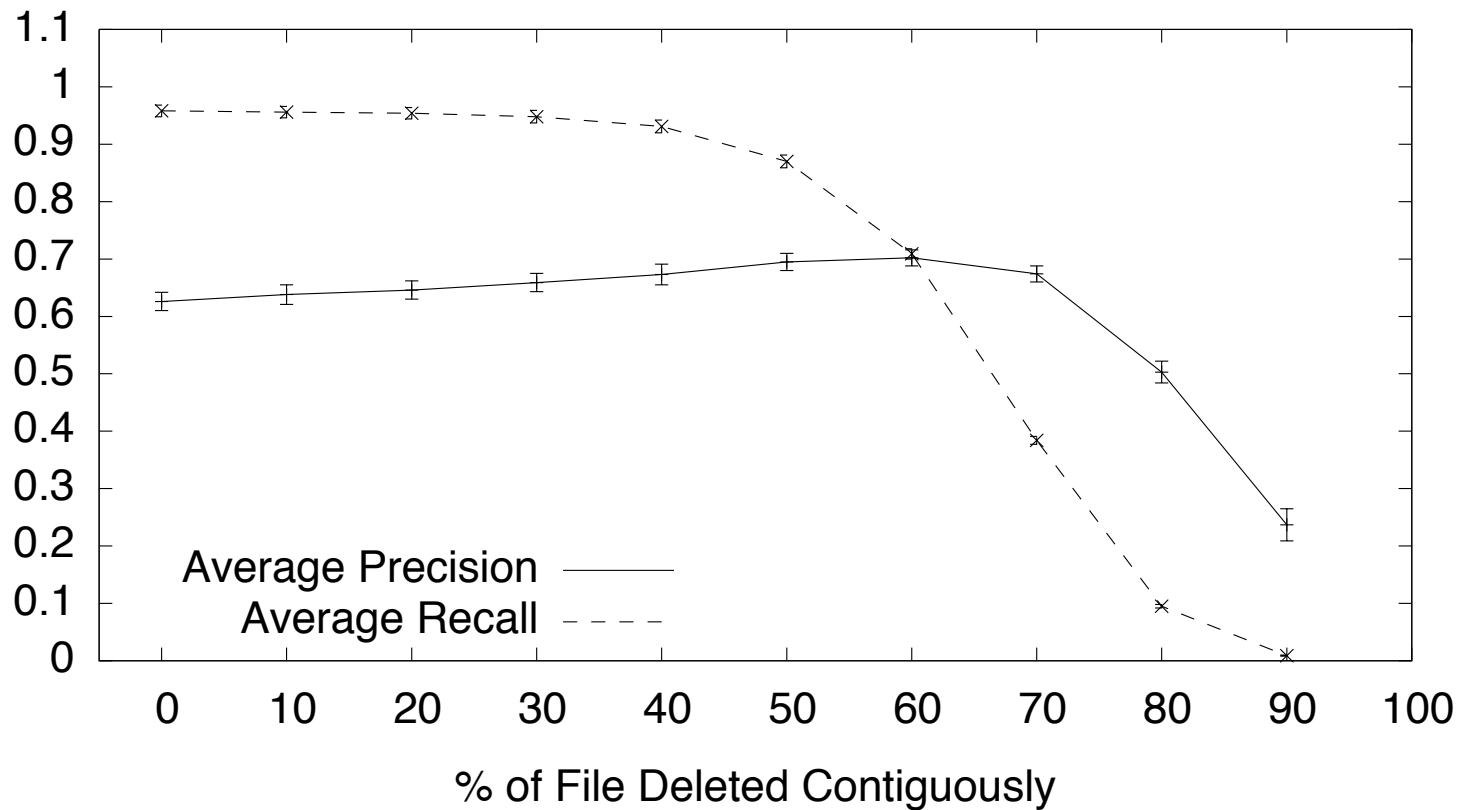
Enron Dataset, Matcher Setting 80, 95% CI Error Bars





# Bit Vector Fingerprint Robustness

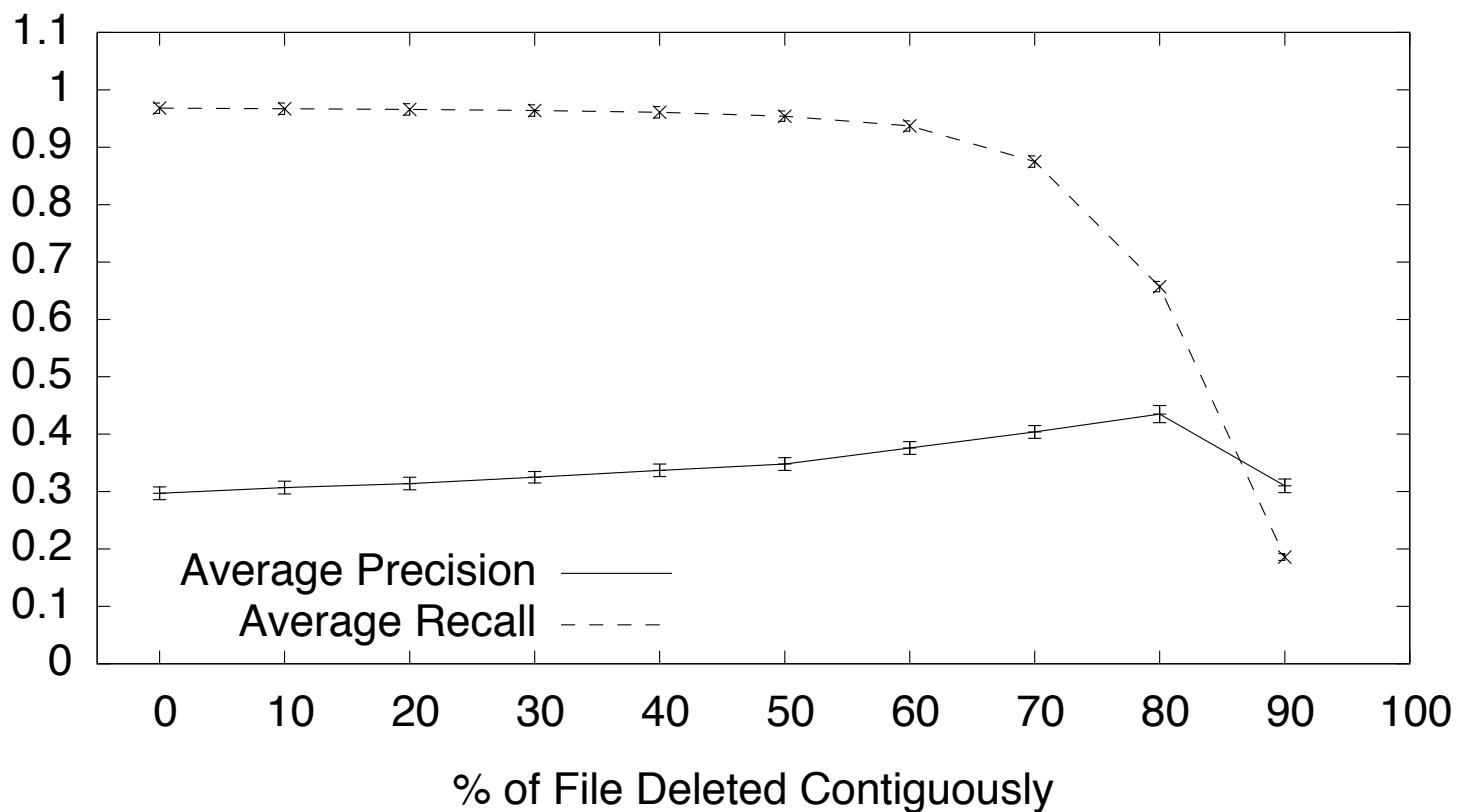
Enron Dataset, Matcher Setting 60, 95% CI Error Bars





# Bit Vector Fingerprint Robustness

Enron Dataset, Matcher Setting 40, 95% CI Error Bars





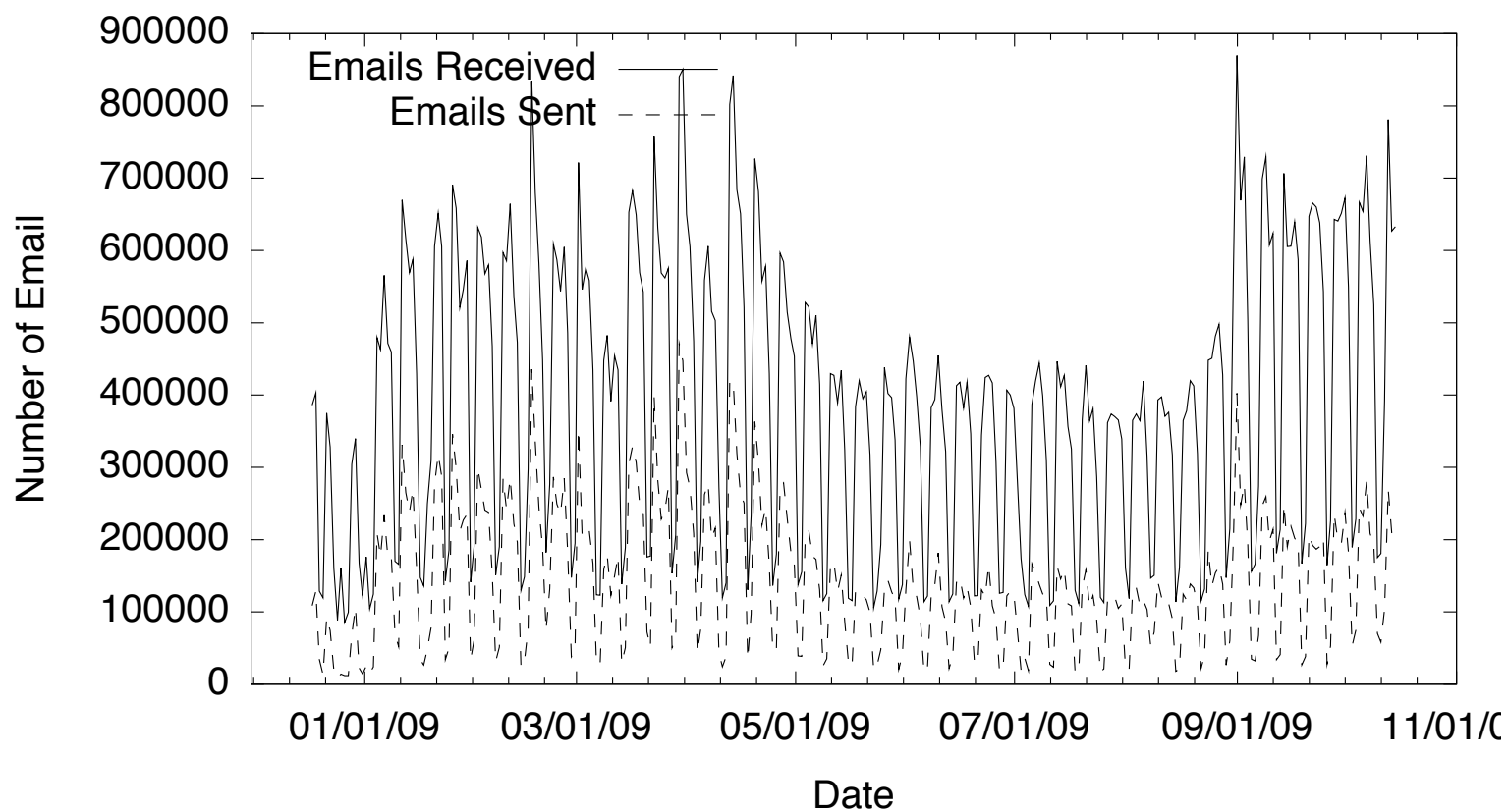
# Overhead

- Two concerns:
  - Storage
  - CPU usage
- Trace driven simulation to determine feasibility
  - Email traces from Georgetown University
    - ~8800 users
  - SOS file system traces from Harvard server
    - Older, but public



# Email Overhead

Emails Processed by Georgetown Mail Server, Dec. 2008 - Oct. 2009

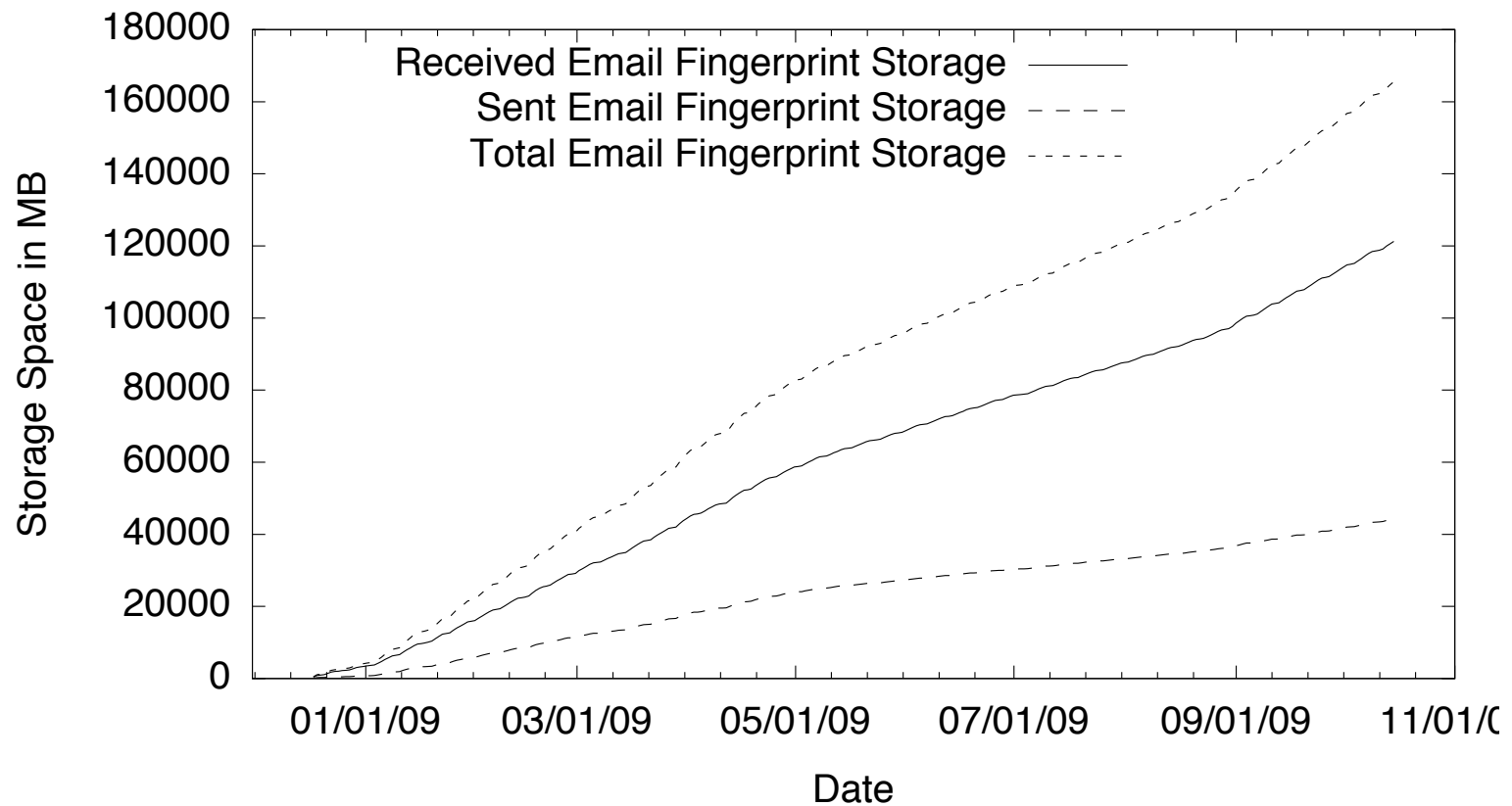






# Email Overhead

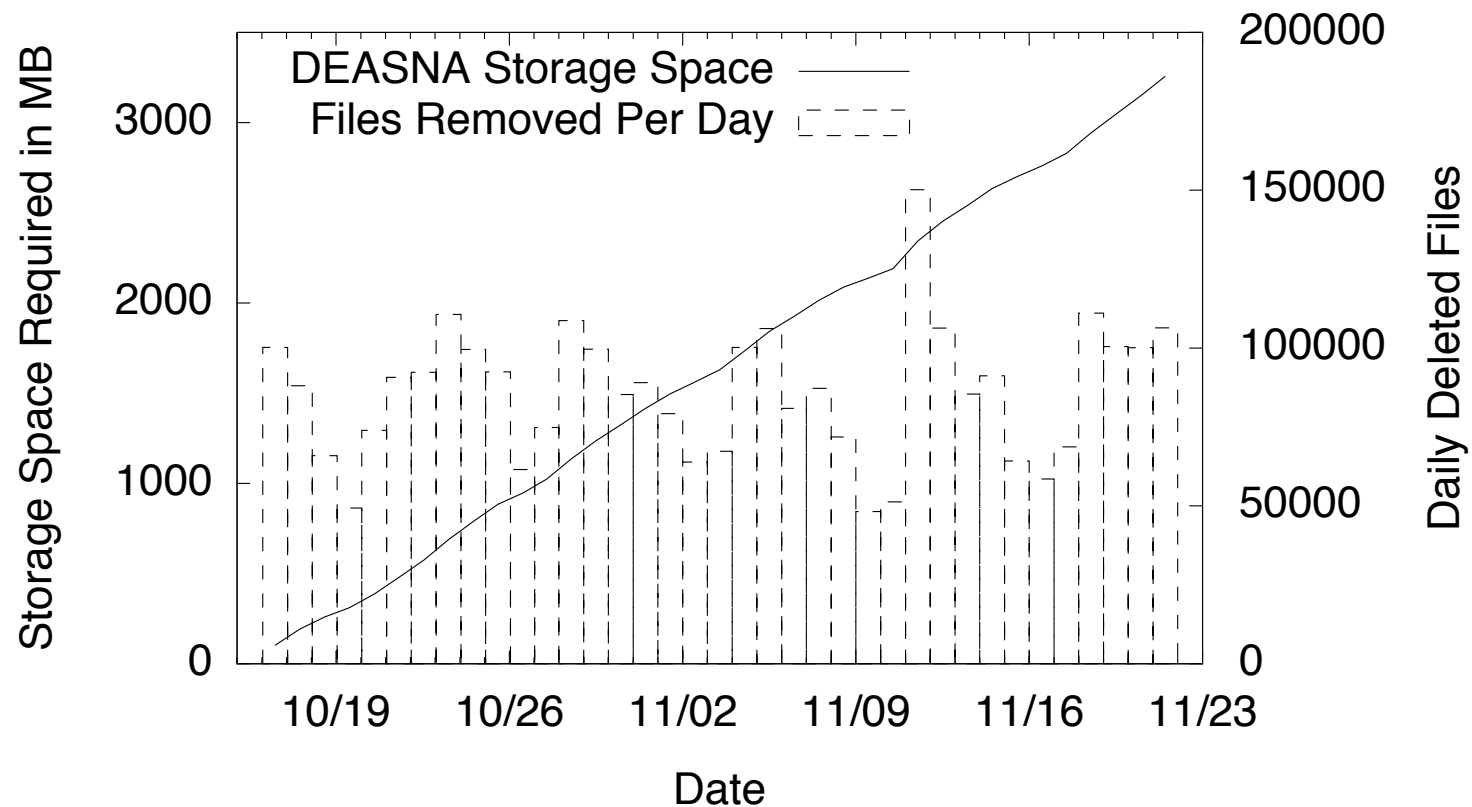
Cumulative Fingerprint Storage Required with 1,024 B Email Signatures





# Server Storage Overhead

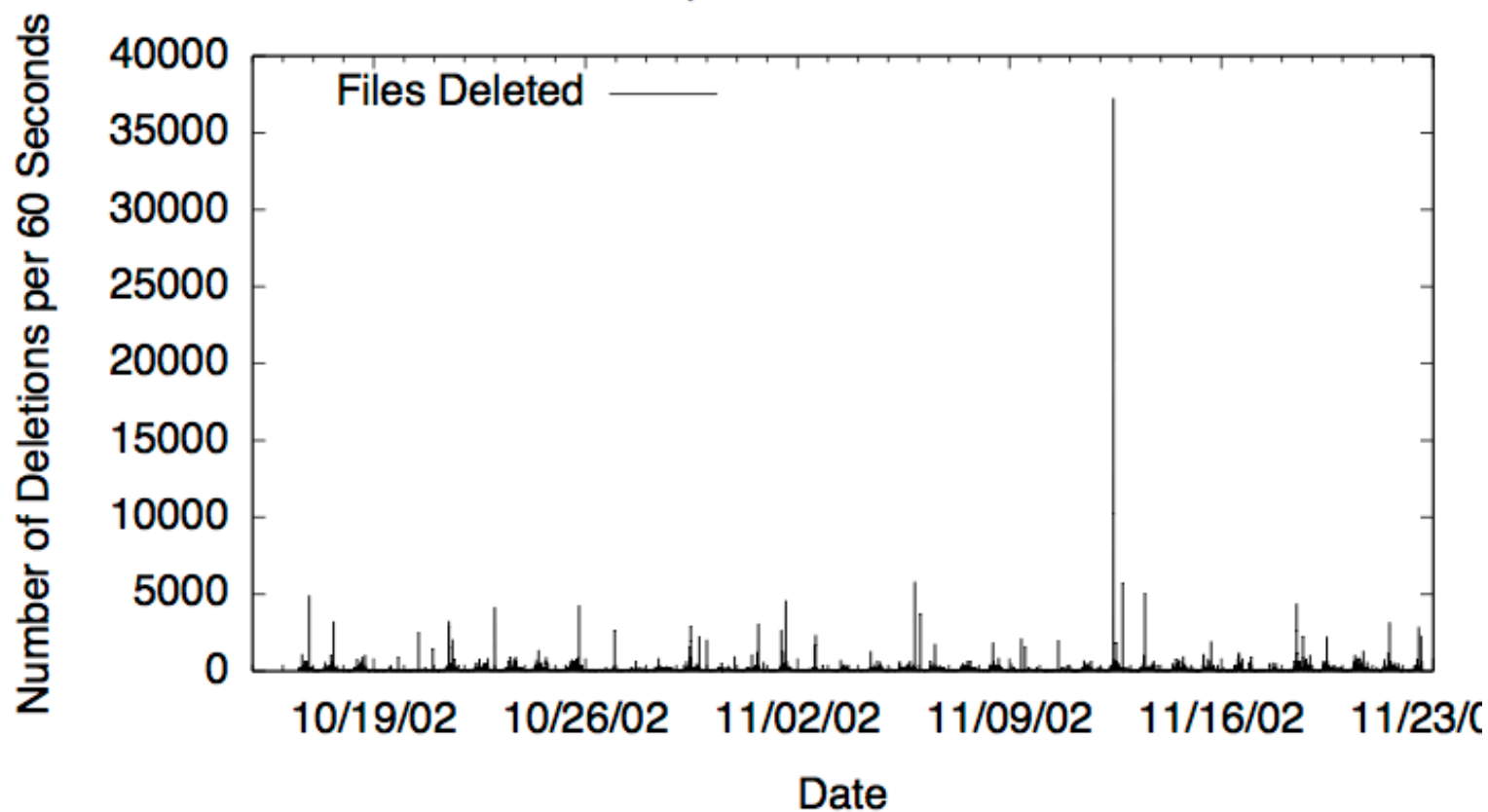
Cumulative Storage Space for DEASNA with 1,024 B Signatures





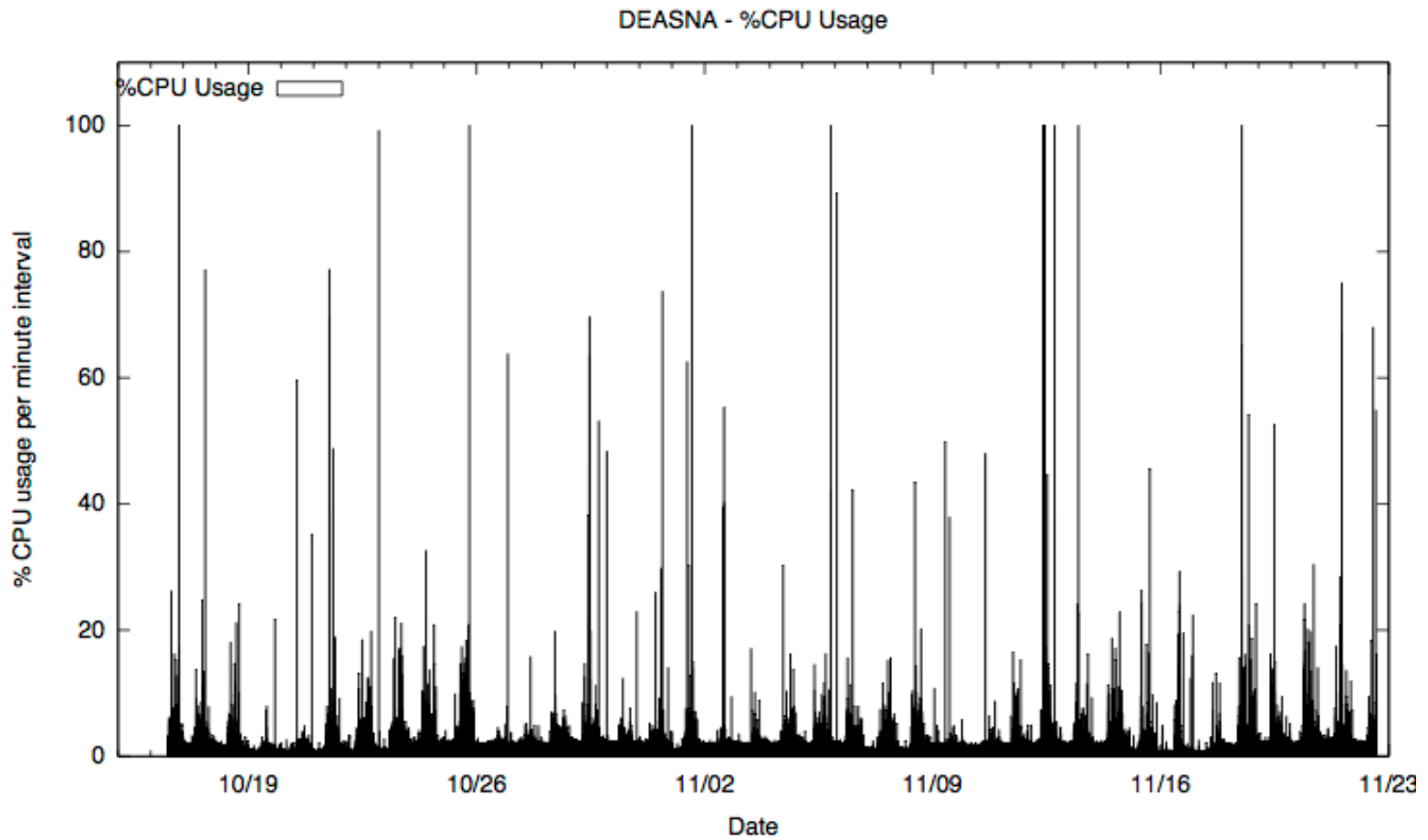
# Server Activity

DEASNA - Files Deleted per Minute, Oct. 16, 2002 - Nov. 22, 2002



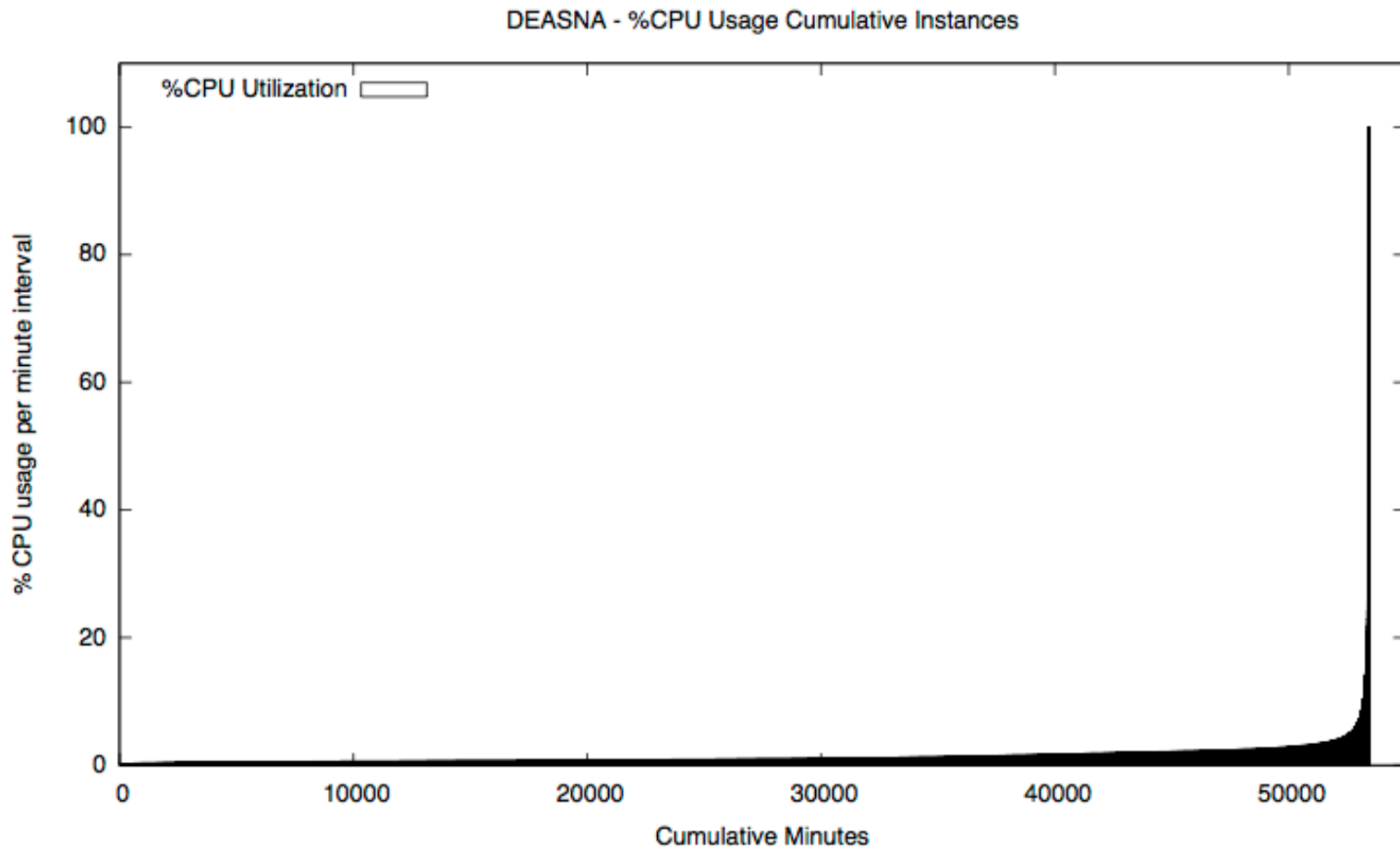


# Server CPU Activity





# Server CPU Activity





# Supported Investigations

- Leaked Documents
  - Given a recovered document, find all users that have ever held a copy
- Misuse Investigations
  - Determine what files an employee was copying or accessing
  - Determine email correspondence, web access
- Keyword Search Overwritten Files
  - Identify which systems to preserve when so required
- Intrusion Response
  - Given a file that was used in an intrusion, find all systems that had that file
- Examination support
  - Identify fragment sources
- Lost equipment review
  - What was on that laptop left in the taxi?



# Continuing Work

- Use fingerprints as an alternative to hashes in large data sets
  - Fingerprint documents by section
- OS Hooks
  - Process files as they are modified or deleted
- Fast fingerprint matching
  - Cosine matching is not suitable for Bloom Filters
- Create signatures for non-text files
  - Images, audio, video, executables, source code



# Summary

- PROOFS allows for efficient proactive collection
  - Google for forensic examiners
  - Make investigations faster, cheaper and more accurate
- Fingerprints have other uses as well
  - Recognizing files in large data sets





# A System for Proactive, Continuous, and Efficient Collection of Digital Evidence

Clay Shields

Ophir Frieder

Mark Maloof

Department of Computer Science  
Georgetown University