EviPlant:
An Efficient Digital Forensic Challenge Creation, Manipulation and Distribution Solution

MARK SCANLON, XIAOYU DU, DAVID LILLIS

Forensics and Security Research Group
School of Computer Science
University College Dublin

Agenda

- Existing Approaches to Challenge Creation
- Motivation for this Work
- Design Considerations
- EviPlant
- Future Work

Digital Forensic Challenges

- Digital Forensics Education and Training
- Proficiency Testing
- Forensic Software Tool Testing and Validation

Necessary Challenge Characteristics

- Answer Keys
  - These are solutions to the challenges presented.
- Realistic Wear and Depth
  - The system being investigated should contain regular usage surrounding email, web browsing, application installations, file creation and deletion, and downloaded content.
- Realistic Background Data/Noise
  - The injection of “incriminating” data should not be obviously the only non-OS/non-application data stored on the disk.
- Sharing and Redistribution

Woods K, Lee C, Garfinkel S, Dittrich D, Russel A, Kearton K. Creating Realistic Corpora for Forensic and Security Education. In: Proceedings of the ADFSL Conference on Digital Forensics, Security and Law. 2011:123–134.

Existing Approaches: Manual

- Typically, this involves an instructor creates a disk image that contains specific evidence for students to find.
- There is no requirement to wait for interesting activity to occur in a natural setting, as the instructor is free to perform/emulate any actions that are desired.
- However, creating these images is a very time-consuming task, particularly given the requirement to ideally provide realistic wear and depth.
- This has the advantage that the precise evidence is known to the instructor and can be used for evaluation purposes.

Existing Approaches: Honeypots

- A *honeypot* involves connecting a computer to a network with the express intention of it being attacked and compromised
- By recording the activities of attackers, interesting disk images can be created.
- However, the majority of attacks are automated, and the quantity of images that feature manual attacks for students to study is low.

Existing Approaches: Second Hand Equipment

- This approach results in valuable data on naturally occurring phenomena on disks,
- One drawback of this approach is that it does not necessarily include materials relating to real crimes that could be used for training purposes
- As the data is generated by real users, privacy law (which greatly varies by jurisdiction) must be taken into account, particularly when redistributing images

Existing Approaches: Automated Scripting

- This approach attempts to leverage the advantages of manual creation, while expediting the process.
- Randomness can be introduced to the scripted process, e.g., Forensig^2
- Instructor based selection of automated actions, e.g, ForGe

Moch, C., Freiling, F.C. The forensic image generator generator (Forensig^2). In: Fifth International Conference on IT Security Incident Management and IT Forensics, 2009. IMF'09. IEEE; 78-93; 2009.

Visti, H., Tohill, S., Douglas, P., 2015. Automatic creation of computer forensic test images. In: Computational Forensics. Lecture Notes in Computer Science, vol 8915, Springer, pp. 163-175; 2015.

Motivation for this Work

- In order to provide realistic data for training, each educational institution creates their own emulated "incriminating" digital data source for investigation, e.g., disk images, network traffic logs, mobile device images, etc.
- Emulating accurate and useful digital evidence for use in the classroom is an overly arduous task.
- Currently this process typically requires days or weeks of experts' time (professors, training personnel) in creating viable digital traces to be discovered during the practical investigation training.
- This project aims to greatly reduce this wasted time through the development of a methodology and technical standard for the automated "planting" of digital evidence in a range of device images for a variety of purposes.

UCD DUBLIN

EviPlant

Design Considerations

- Ease of Creation
- Efficient Distribution
- Efficient Injection
- Operating System Compatibility
- Mobile Compatibility

Premise for EviPlant

- Base images are created for all necessary operating system
- Actions are performed to emulate necessary activity
  - Bare with me!
- The resultant images are “diffed” against the base images to create injectable evidence packages

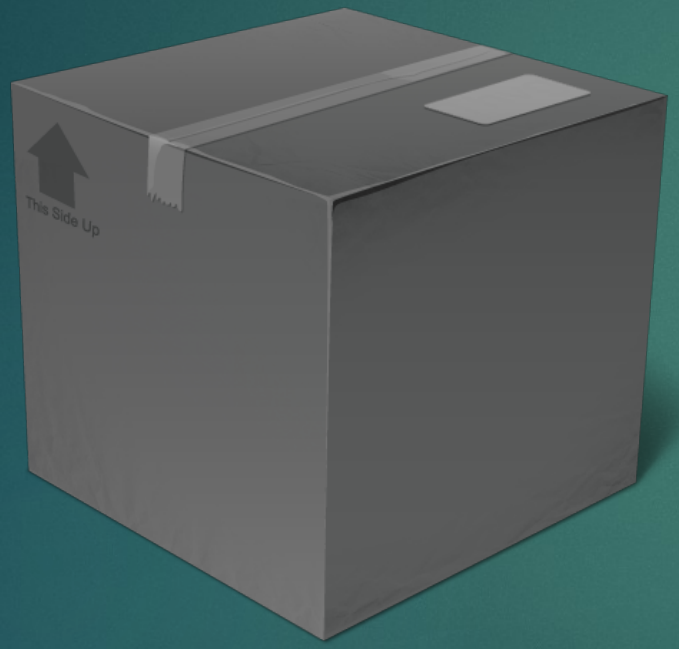
Premise: Diffing Snapshots

- Base images are created for numerous operating systems
  - These base images are booted to perform tasks
- Diffing Engine compares base image with used images and changes are identified
  - Files
  - Settings
  - History/Logs
  - Installations/Uninstallations
- Changes stored in an Evidence Package

Injectable Evidence Packages

- Contains the files and associated metadata for insertion into standard base OS images
  - Effectively a grouping of the artefacts and metadata from the aforementioned Deduplication Project
- Packages creatablefor wear and depth, background data, personas, case types, etc.

Types of Packages



Black Box

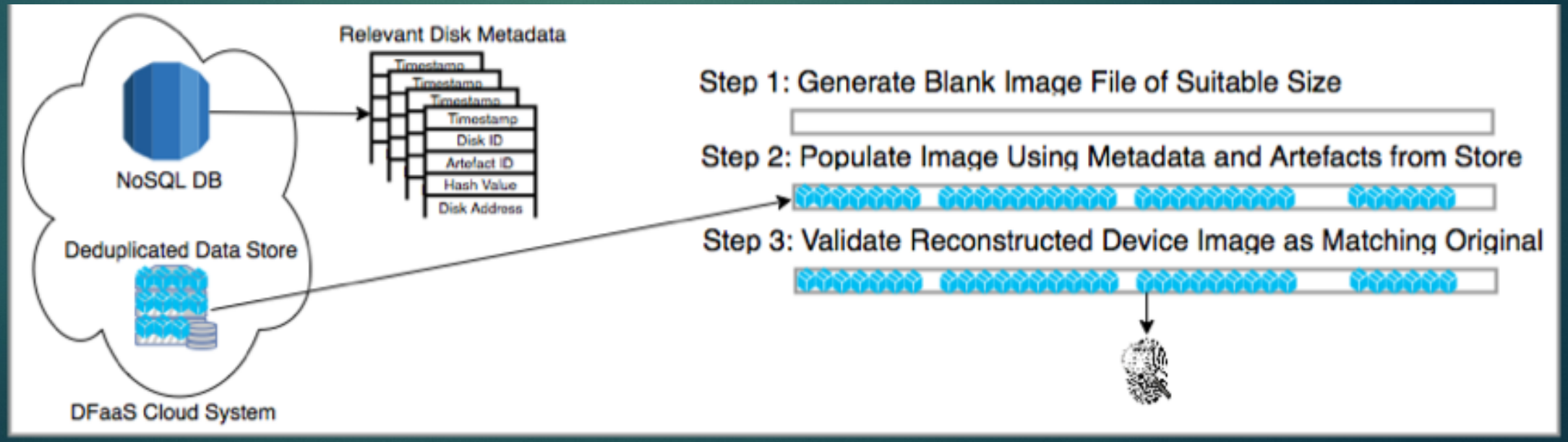
Reverse Engineered

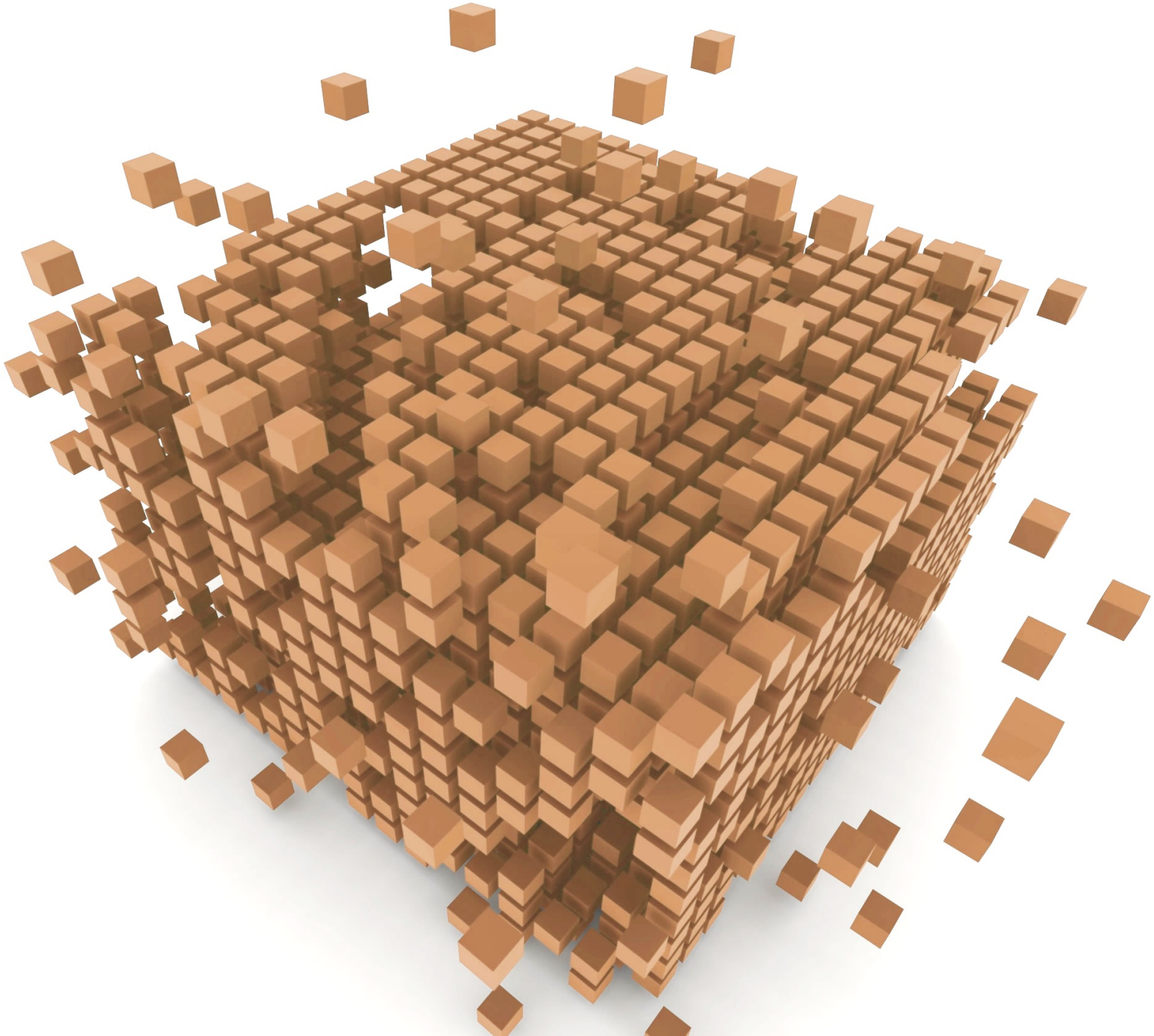
Reverse Engineered Packages

- Ability to manipulate package contents (files and metadata)
  - Facilitates timeline creation/modification
- Ability to create nested “story” packages
  - Dependencies possible on existing packages

Evidence Planting Phase

- Evidence injection phase involves:
  - Artefact “Planting”
  - Timeline creation
  - Metadata manipulation (for example SQLite inserts, Registry Edits, Application settings/data replacement etc.)





Prototype

- Built in Python using the `pytsk` library
  - Compatibility with NTFS, FAT, ExFAT, UFS 1, UFS 2, EXT2, EXT3, EXT4, HFS, ISO 9660, and YAFFS2
- Base images created for Windows 7 and newer, and Ubuntu
- Various user activities were emulated
- Image were diffed and evidence package created
- Analytically sound reconstruction
- Planted evidence was recoverable using standard tools

Educational Benefits

- **Automated Practice Exercises** - The ability to create exercises for students on-the-fly will allow students to practice their skills on many different exercises as opposed to being limited to the few disk images made available to them.
- **Helps to Eliminate Plagiarism** - Custom generated, practical digital forensic challenge exercises eliminates the possibility of students engaging in plagiarism of results for known, freely available corpora. After building a sufficiently large corpus of evidence packages, it's possible that a unique disk image could be automatically created per student with each offering their own challenges for the students while achieving the learning outcomes of the current topic.

Educational Benefits

- **Assessment** – The creation of a different challenge for each student in a class can enable a laboratory based practical assessment to count towards their final grades. Multiple students could take this test simultaneously in the same room as a bespoke challenge could be set for each.
- **“Replay” case challenges** – Previously analyzed cases could be “replayed” easily providing valuable real-world training data. These cases come with ready-made answer sets.

Other Potential Benefits

- Proficiency Testing
  - In a controlled environment (i.e. equipment, tools, time), standardised in-house proficiency testing could take place.
- Automated DF Tool Testing and Validation
- Point-in-time Reconstruction
- Malware Lifecycle Analysis

Future Work

- Reversing of comprehensive list of artefacts
- Artefact Collision Resolution Policy
- Physical Injection

MARK.SCANLON@UCD.IE
WWW.MARKSCANLON.CO
YES, .CO
@MRKSCN