Arizona State University

CSE 469: Computer and Network Forensics

Topic 2: Evidence Acquisition

Dr. Jaejong Baek | Spring 2020

Acquisition

- First step in the forensic process:
  - Copy the evidence/data without altering or damaging the original data or scene.
  - Can you think of a circumstance where analyzing the original would be impossible?
- Must be done concurrently with Authentication:
  - Prove that the recovered evidence/data is the same as the original data.
  - Why?

Purpose of Acquisition

- Imagine this scenario:
  - While examining some files on a hard drive, the examiner forgets to turn on the write-blocker and some file attributes change.
  - Examiner argues that "none of the files impacting the case were affected."
- Consequences:
  - Defense may claim the evidence is no longer trustworthy and should not be ***admitted***.
  - Perhaps some important files *were* changed, but the examiner has no way to know for sure.

Forensic examiners can't risk compromising the evidence (changing it without meaning to). It could be the difference between proving someone's innocence *or* guilt!

So... we work on a *copy* of the evidence instead.

- Write-blocker
  - HW Write-blocker

  - SW Write-blocker: Encase, FTK-imager
    - Windows 10: regedit

      \HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\
      New > Key > type “storageDevicePolicies”
      New > Edit DWORD(32-bit) Value > type “WriteProtect”
      Modify > Value, 0 -> 1

Purpose of Authentication

- Acquired copy of evidence provides protection for the original.
- Authentication proves the copy is ***exactly the same*** as the original.
- How can you prove two digital things are exactly the same?
  - Compare every single bit.
  - OR…
  - Compute a cryptographic hash of both.

Arizona State University

Hash Functions / Message Digests

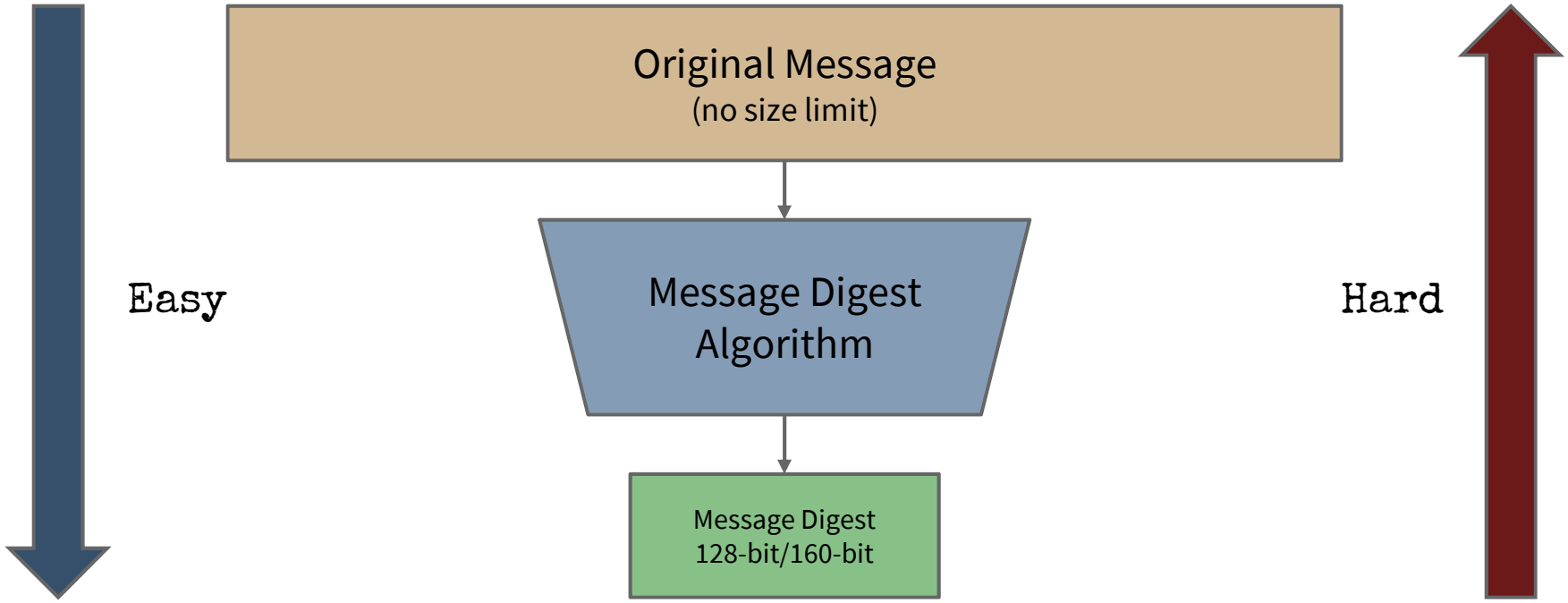
Message Digests

- Also called *cryptographic hash functions*
- Purposes:
  1. Uniquely identify data using the data itself as the source
     - Better than an index or a random number because others can generate the same identification using just the data
     - Should be easy to generate for any input (message)
  2. Infeasible to find data that will generate a specific digest
     - Can't process the hash in reverse
  3. Infeasible to find two messages that will generate the same digest
  4. The digest changes if the data changes

Called a "collision"

- Usually based on "lossy" computations (**compression**)
  - Data encoding method that uses inexact approximations and partial data discarding to represent the content.

Message Digests



Hash Function: Unique fixed length

20-letter password

MD5 → 128 bit digest

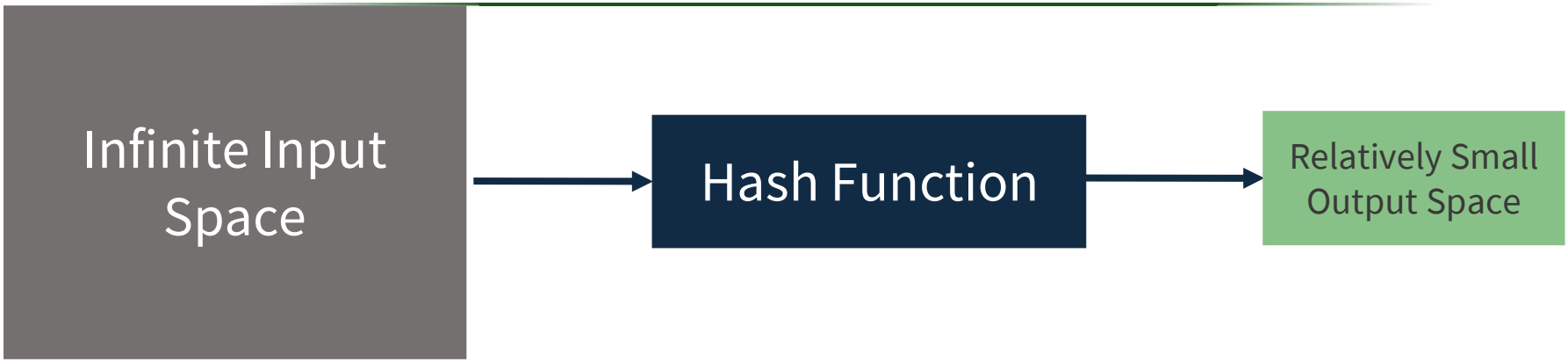
20MB PDF file

MD5 → 128 bit digest

1TB Hard Disk

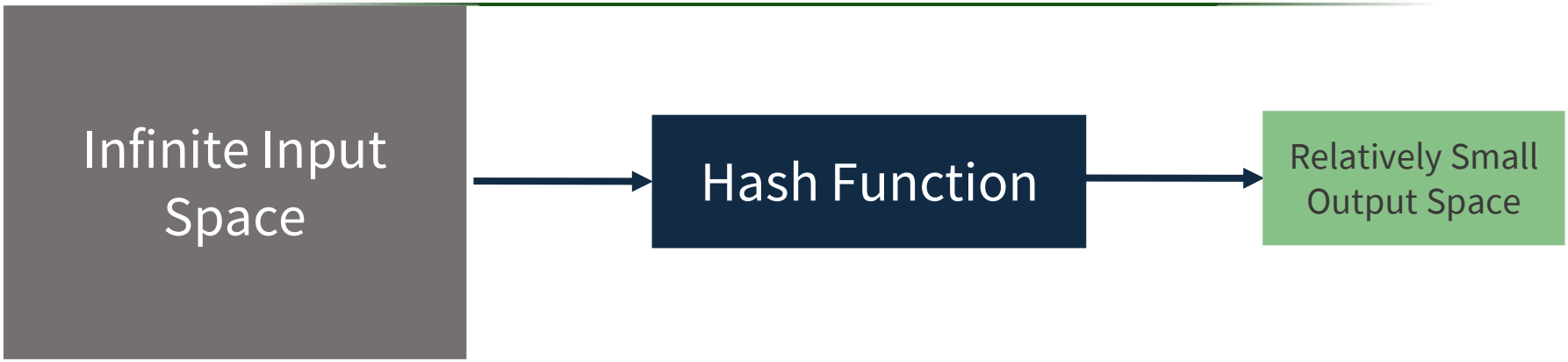
MD5 → 128 bit digest

Hash Function: One-Way



- One-way function: It is impossible to calculate m from $H(m)$

Hash Function: Collision



- Some pairs of inputs will be mapped to the same hash value. This is called a collision.
- The **pigeonhole principle** states that if n items are put into m containers, with $n > m$, then at least one container must contain more than one item.

Message Digest Algorithms

- MD5 (Ron Rivest, 1991)
  - 128-bit digest
  - Simple, compact, and fast
  - Has collision problems - $2^{20.96}$ instead of 2^{64} as expected
- SHA-1 (NIST, 1995)
  - 160-bit digest
  - Similar to MD5
- SHA-2 (Family, 2001)
  - Includes SHA-256 and SHA-512
  - 256-bit or 512-bit digest
  - Only has theoretical attacks at present
- SHA-3 (2006)
  - 1600-bit digest
  - Not meant to replace SHA-2, only provide a strong alternative
  - Became a FIPS standard when approved on August 5, 2015
    - Federal Information Processing Standard - Maintained by NIST

Note: The MD5 and SHA-1 algorithms have been broken and are no longer recommended for use for anything important. (2013)

See https://en.wikipedia.org/wiki/Message_digest

- Online hash calculator
  - https://www.tools4noobs.com/online_tools/hash/
- crackstation.net
  - https://crackstation.net/

Desirable Properties of Hash Functions

- Consider a hash function *H*:
  - Performance: Easy to compute *H(m)*
  - Preimage resistant: Given a hash value *h*, it's computationally infeasible to find an *n* that *H(n)=h*
  - 2nd preimage (weak collision) resistant: Given *m*, it's computationally infeasible to find *m'* such that *H(m')=H(m)* and *m'!=m*
  - Strong collision resistant: Computationally infeasible to find *m1*, *m2* such that *H(m1)=H(m2)*

Arizona State University

Acquisition Types and Methods

Acquisition Types

- Static (or dead) acquisitions
  - System is turned off
  - ***Preferred method*** of acquisition (basic, common)
  - Limits the data available
    - No RAM data
    - No way to decrypt
    - Nonvolatile data source
- Live acquisitions
  - System is still running
  - Data still available in RAM (Volatile source)
  - Crucial if the storage is ***encrypted*** - only way to recover the key to decrypt the data
  - Including Remote acquisition via network

Three Acquisition Methods

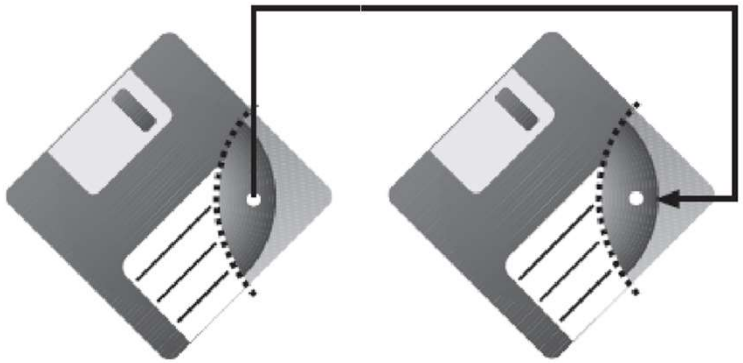
Ordered from the least amount of data collected to the most:

1. Logical Acquisition
   - Captures only ***specific files*** of interest to the case or specific ***types of files***.
   - Example: Email investigation - `.pst` and `.ost` files.
   - Focus: Filesystem (relies on filesystem to list files correctly)
2. Sparse Acquisition
   - Same as logical, but includes fragments of ***unallocated*** (deleted) data.
   - Focus: Partition or Volume
3. Bit-stream Copy or Acquisition
   - ***Exact copy*** (bit for bit) of the entire device; also called a ***forensic copy***.
   - Includes deleted files, fragments, etc.
   - Focus: Disk or other storage medium.

NOTE: A logical or sparse acquisition may be more appropriate if *time is limited* or if the *original storage isn't accessible*, such as in web or cloud forensic cases.

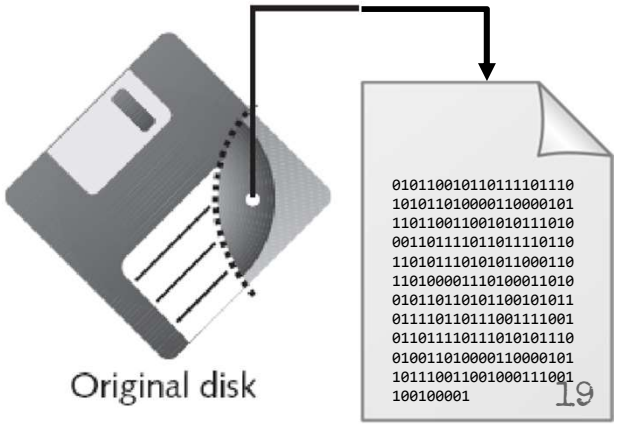
More on Bit-Stream Acquisitions (1)

- Two types of bit-stream copies:
  1. Bit-stream disk-to-disk
     - Contents of evidence written to a storage device that exactly matches the make and model of the original: a ***literal duplicate*** of the original.
     - Only used when something about the storage device itself is important.



More on Bit-Stream Acquisitions (2)

- Two types of bit-stream copies:
  2. Bit-stream disk-to-image file
     - All bits from the evidence are copied to a file: a ***virtual duplicate*** of the original.
     - More common method than disk-to-disk.
     - Referred to as an “image” or “image file”.
     - File is the exact size of the original evidence.

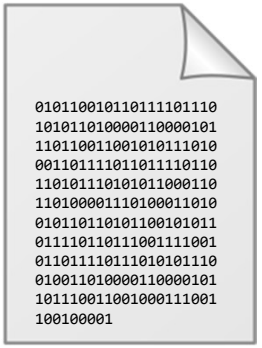


Arizona State University

Evidence Formats

Raw

- Bit-stream image file
- Advantages
  - Fast (but uncompressed) data transfers.
  - Can ignore minor data read errors on source drive.
  - “Universal” format - not specific to any tool.
- Disadvantages
  - Requires as much storage as original disk or data.
  - Tools might not collect marginal (bad) sectors.



Proprietary Formats

- Features:
  - Compressed image files.
  - Split an image into smaller segments.
  - Integrate metadata into the image file.
- Disadvantages:
  - Inability to share an image between different tools.
  - File size limitation for each segmented volume.
- Unofficial standard: Expert Witness
  - Files end in `.e01`, `.e02`, `.e03`, etc.

Advanced Forensics Format

- Developed by Dr. Simson L. Garfinkel
- Design goals
  - Provide compressed or uncompressed image files.
  - No size restriction for disk-to-image files.
  - Provide space in the image file or segmented files for metadata.
  - Simple design with extensibility.
  - ***Open source*** for multiple platforms and OSs - no vendor lock-in.
  - Internal consistency checks for self-authentication.
- File extensions
  - *.afd for segmented image files.
  - *.afm for AFF metadata.

Arizona State University

Acquisition Tools

Acquisition in Linux

- Preparing a target drive for acquisition in Linux
  - Linux distributions can create Microsoft FAT and NTFS partition tables.
  - `fdisk` command lists, creates, deletes, and verifies partitions in Linux.
  - `mkfs.msdos` command formats a FAT file system from Linux.

- Acquiring data with dd in Linux
  - `dd` ("data dump") command
    - Can read and write from media device and data file.
    - Creates raw format file that most computer forensics analysis tools can read.

dd

- dd command

```
> dd if=[source] of=[destination] [block size]
   [block #] + [0|1] records in
   [block #] + [0|1] records out
```

Examples:

```
> sudo fdisk –l | grep –i dev
> dd if=/dev/sdb of=./usb.dd
> dd if=/dev/hda of=/mnt/hda.dd bs=2k | md5sum
> dd if=/dev/sda | split –b 650m – image_sda
```

dcfldd

- Enhanced version of GNU dd with features useful for forensics and security
  - Hashing on-the-fly
  - Status output
  - Flexible disk wipes
  - Image/wipe Verify
  - Multiple outputs
  - Split output
  - Piped output and logs

```
$ sudo apt install dcfldd
$ sudo dcldd if=/dev/sdb of=usb.dd hash=sha1 hashlog=usb.log
$ More usb.log
```

ProDiscover Basic

- Proprietary Acquisition Format
  - Image file will be split into segments of 650MB
  - Creates image files with an .eve extension, a log file (.log extension), and a special inventory file (.pds extension)
- Raw Acquisition Format
  - Select the UNIX style dd format in the Image Format list box
  - Raw acquisition saves only the image data and hash value

https://www.prodiscover.com/

AccessData FTK Imager

- Included on AccessData Forensic Toolkit
- Makes disk-to-image copies of evidence drives
  - At logical partition and physical drive level.
  - Can segment the image file.
- Evidence drive must have a hardware write-blocking device
  - Or the USB write-protection Registry feature enabled.
- FTK Imager can't acquire drive's host protected area