Friedrich-Alexander-Universität
Technische Fakultät

DiskForge: Timestomping on Disk Images for Educational Benefit

Niclas Pohl Lena L. Voigt Christopher Hargreaves Christofer Fein Felix Freiling

IMF 2025 | September 16, 2025

Motivation

- Realistic disk images are essential for digital forensics training.
- Developing exercises is often done manually
  - Time consuming
  - Error Prone
- Flexible tools are needed to customize exercises and control artifacts.
- Automation can enhance scalability and reduce instructor workload.

Contributions

- **DiskForge:** open-source, extensible framework for disk image manipulation
- Implemented timestomping for:
  1. File system metadata
  2. Log files (e.g., syslog)
  3. SQLite databases
- Evaluated against existing manipulation approaches
- Discussed detection of forgeries and implicit traces

Background

File System & Database Timestamps

- Ext4 Inode Timestamps:
    - Accessed, Modified, Changed and Created
    - Support sub-second precision

	Format	**Example**
Ext4 Inode	Integer / Unix Timestamp	1758113100
SQLiteDB	TEXT / ISO-8601	2025-09-17T12:45:00.000Z
	REAL / Julian Date	2460936.03125
	INTEGER / Unix Timestamp	1758113100

Background
Logs & Mounting Artifacts

Syslog Entries:

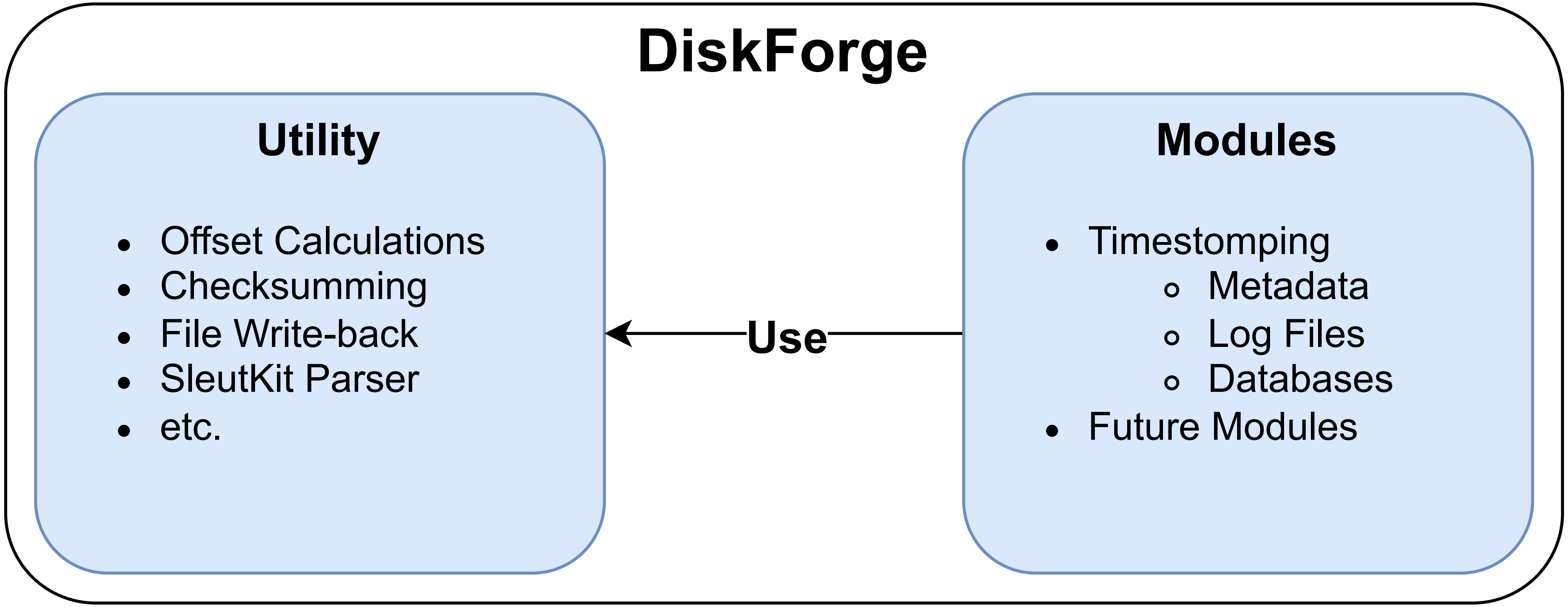
```
MONTH DAY TIME HOSTNAME PROCESS [PID]: MESSAGE
Jun 18 23:01:45 host kernel: [    0.000000] SMBIOS ...
Jun 18 23:01:45 host systemd[1]: Starting system logging service...
```

Mounting Artifacts:

- Last Mounted on
- Mount Path
- etc.

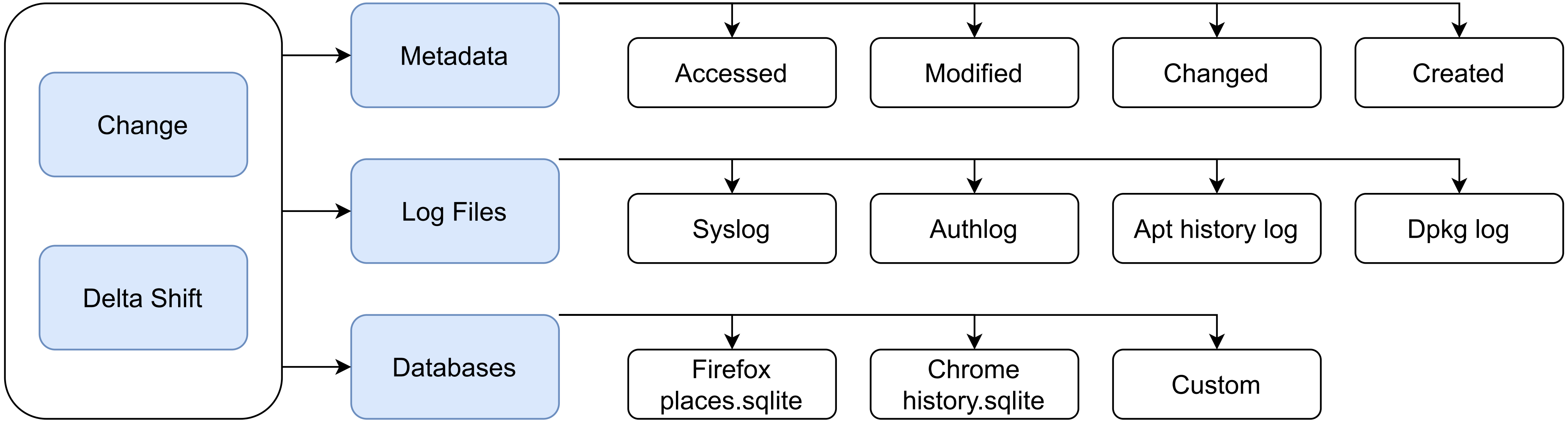
DiskForge Framework

Structure - Utility & Modules



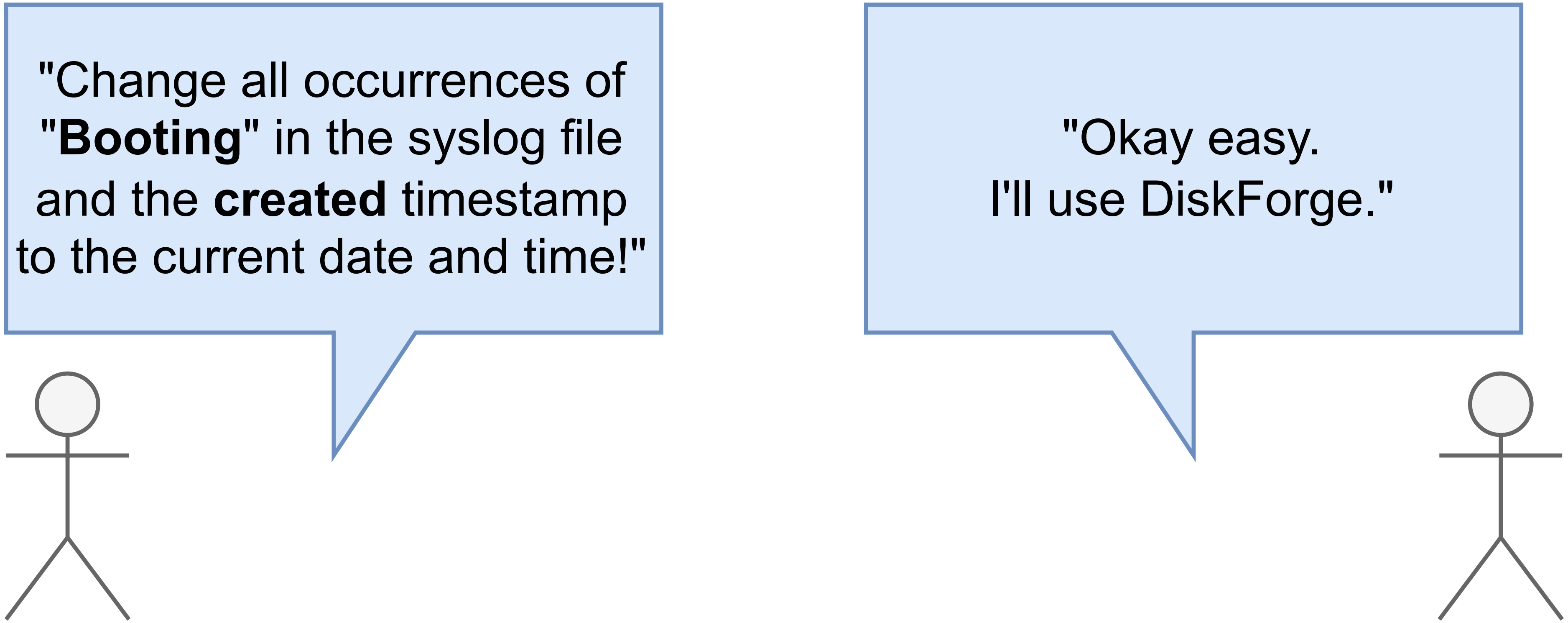
DiskForge Framework

Structure - Change & Delta Shift



DiskForge Framework

Example Use Case



"Change all occurrences of "**Booting**" in the syslog file and the **created** timestamp to the current date and time!"

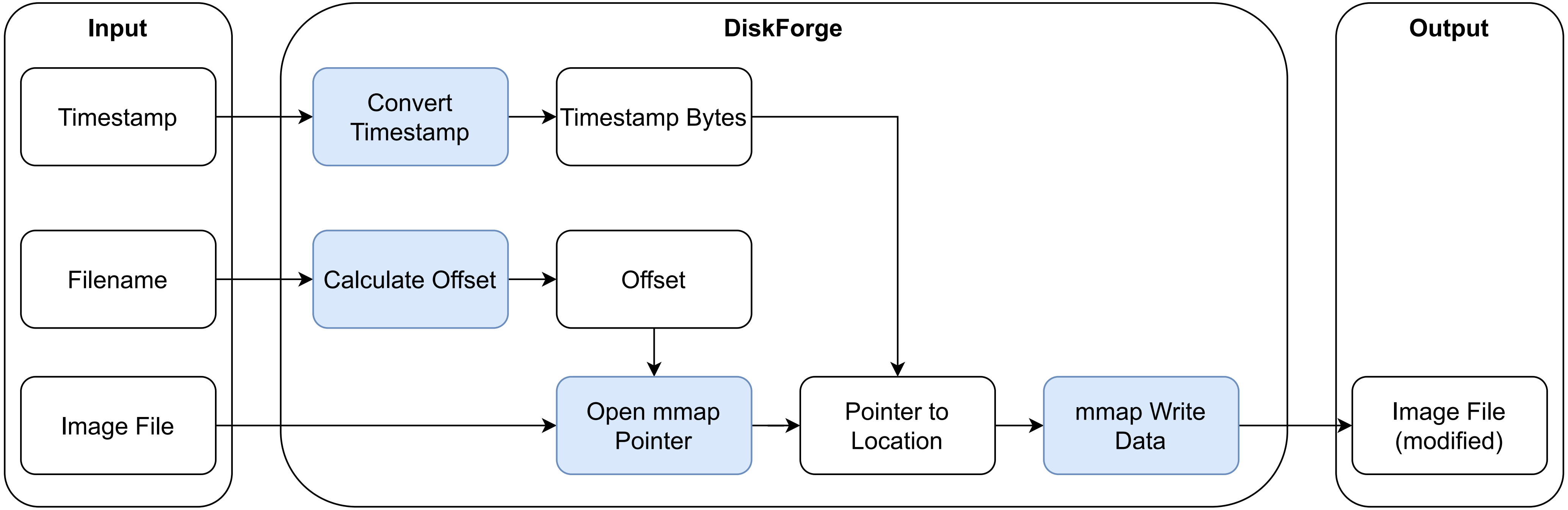
"Okay easy.
I'll use DiskForge."

DiskForge Framework

Example Use Case - Demonstration

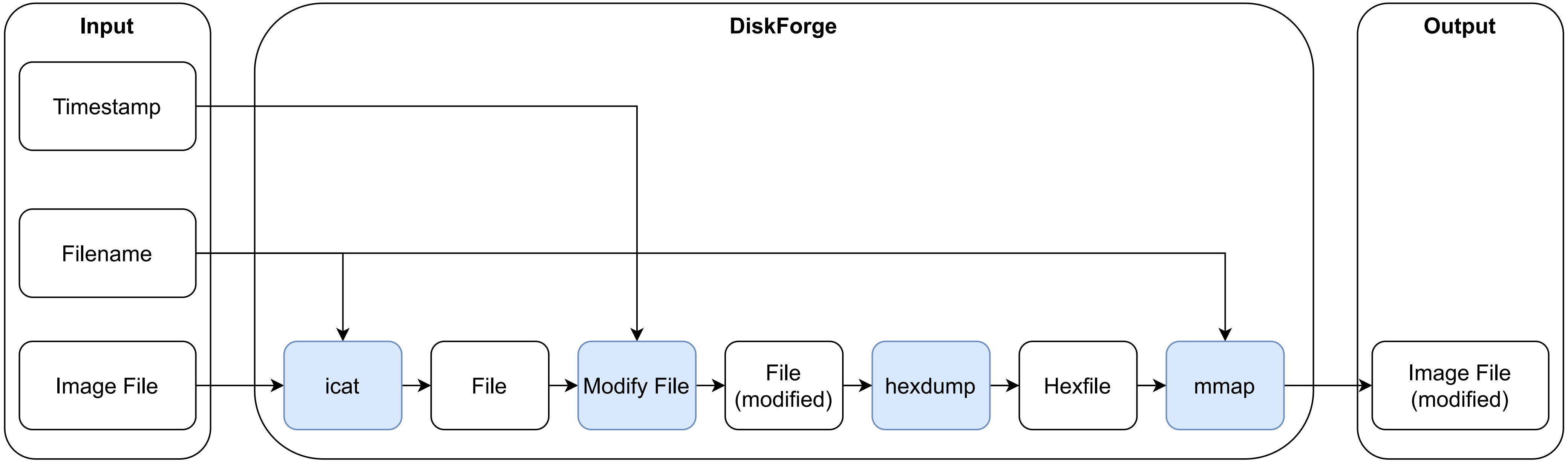
DiskForge Framework

Example Use Case - Workflow Metadata Manipulation



DiskForge Framework

Example Use Case - Workflow File Manipulation



Evaluation

		Changed Bytes	Changed Blocks
Metadata			
	Touch	13,167	13
	Debugfs	36	1
	DiskForge	36	1
Log Files			
	Gedit	839,757	254
	Sublime Text	1,591,386	418
	DiskForge	736,213	189
Databases			
	SQLite Browser	174,458	75
	DiskForge	42	3

Evaluation

		Changed Bytes	Changed Blocks
Metadata			
	Touch	13,167	13
	Debugfs	**36**	**1**
	DiskForge	**36**	**1**
Log Files			
	Gedit	839,757	254
	Sublime Text	1,591,386	418
	DiskForge	736,213	189
Databases			
	SQLite Browser	174,458	75
	DiskForge	42	3

Evaluation

		Changed Bytes	Changed Blocks
Metadata			
	Touch	13,167	13
	Debugfs	36	1
	DiskForge	36	1
Log Files			
	Gedit	**839,757**	**254**
	Sublime Text	**1,591,386**	**418**
	DiskForge	**736,213**	**189**
Databases			
	SQLite Browser	174,458	75
	DiskForge	42	3

Discussion

Limitations of DiskForge

- Currently supports only **ext4** file systems
- **File size constraints**: modifications must fit within allocated space
- **Zero-byte overwrites**: shortened files filled with 0x00, unusual in text files and detectable
- No support for **multi-file log archives** (e.g., syslog, syslog.1, syslog.2.gz)
  - Events may end up in the wrong file chronologically
- Lacks **event correlation** and **plausibility checks**
- Ethical safeguard: writes a **watermark** into the ext4 superblock

Discussion

Implicit Traces of Manipulation

- File timestamps vs. parent directory timestamps
- Log file continuity:
  - Sequential order across rotated logs (syslog, syslog.1, ...)
  - Multi-event chains must remain connected
- Extra Time Information in Logs
- Inconsistencies across **database entries and values**
  - e.g., `visit_count`, `last_visit_date`, and IDs in SQLite

Discussion

Implicit Traces of Manipulation

- File timestamps vs. parent directory timestamps
- Log file continuity:
  - Sequential order across rotated logs (syslog, syslog.1, ...)
  - Multi-event chains must remain connected
- Extra Time Information in Logs
- Inconsistencies across **database entries and values**
  - e.g., `visit_count`, `last_visit_date`, and IDs in SQLite

→ *Even when explicit traces are avoided, these inconsistencies can reveal manipulation.*

Conclusion

Closing Thoughts

It is extremely hard to create a perfect forgery

Even with full control over every bit and byte...

Conclusion

Closing Thoughts

It is extremely hard to create a perfect forgery

Even with full control over every bit and byte...

But without cross-checking and correlation, forensic analysis can still be misled.

Q&A

Further Information

Diskforge: https://github.com/NiclasPohl/DiskForge
Paper: https://dl.acm.org/doi/10.1145/3748265
Thesis: https://github.com/NiclasPohl/Timestomping-on-Disk-Images
Contact: niclas.pohl@outlook.de