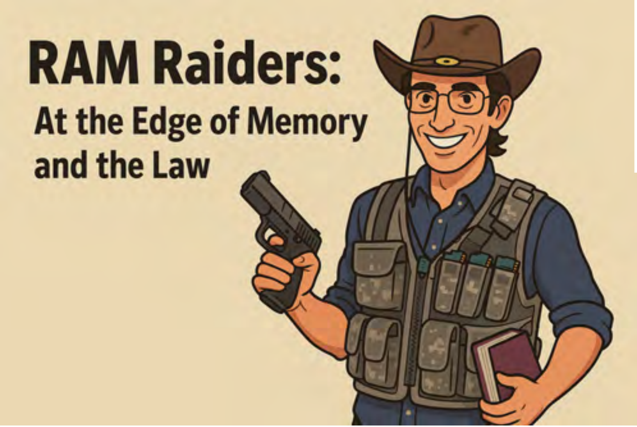


RAM Raiders: At the Edge of Memory and the Law

(PDF slides)

Ricardo J. Rodríguez

Universidad de Zaragoza

14th Int. Conf. on IT Security Incident Management & IT Forensics (IMF 2025)

Albstadt, Germany

All cartoons shown in this presentation were generated using ChatGPT

$whoami

(DFRWS EU 2024 RODEO)

- **Associate Professor at the University of Zaragoza**
- **Research lines**:
  - Program binary analysis
  - Digital forensics
  - System security
  - Formal methods applied to cybersecurity
- Speaker and trainer at different infosec conferences (NcN, HackLU, RootedCON, STIC CCN-CERT, HIP, MalCON, HITB...)

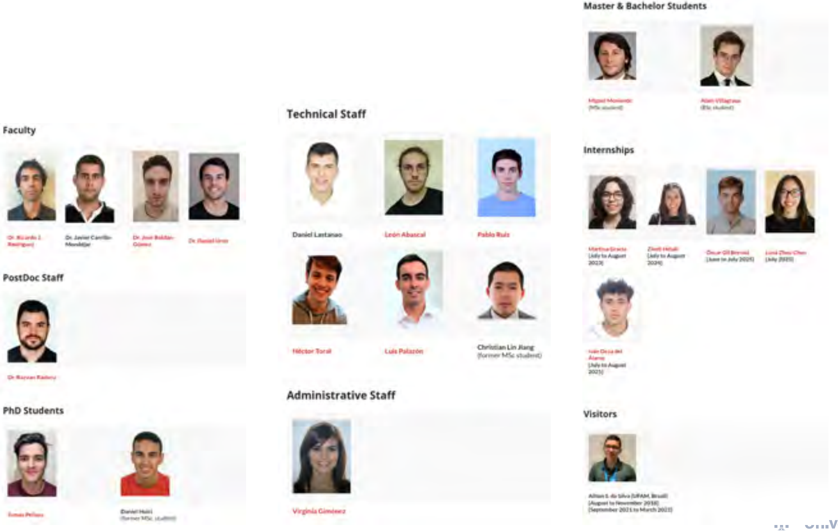
$whoami

(DFRWS EU 2024 RODEO)

- **Associate Professor at the University of Zaragoza**
- **Research lines**:
  - Program binary analysis
  - Digital forensics
  - System security
  - Formal methods applied to cybersecurity
- Speaker and trainer at different infosec conferences (NcN, HackLU, RootedCON, STIC CCN-CERT, HIP, MalCON, HITB. . .)
- **Research team** – *we make really good stuff!* ☺
  - `https://reversea.me` / `https://t.me/reverseame`

~~$whoami~~ $whoarewe

https://reversea.me/index.php/people/



Faculty

Dr. Ricardo J. Rodríguez

Dr. Javier Carrillo-Mondéjar

Dr. José Roldán-Gómez

Dr. Daniel Uroz

PostDoc Staff

Dr. Razvan Raducu

PhD Students

Tomás Pelayo

Daniel Huici
(former MSc student)

Technical Staff

Daniel Lastanao

León Abascal

Pablo Ruiz

Héctor Toral

Luis Palazón

Christian Lin Jiang
(former MSc student)

Administrative Staff

Virginia Giménez

Master & Bachelor Students

Miguel Menéndez
(MSc student)

Alain Villagrasa
(BSc student)

Internships

Martina Gracia
(July to August 2023)

Ziedi Hélali
(July to August 2024)

Óscar Gil Bernal
(June to July 2025)

Lucía Zhou Chen
(July 2025)

Iván Ocaña del Álamo
(July to August 2021)

Visitors

Ailton S. da Silva (UFAM, Brasil)
(August to November 2018)
(September 2021 to March 2022)

Agenda

1 Introduction

2 Current Challenges

3 From RAM to Courtroom

4 Conclusions

5 Future Directions

Agenda

1 Introduction

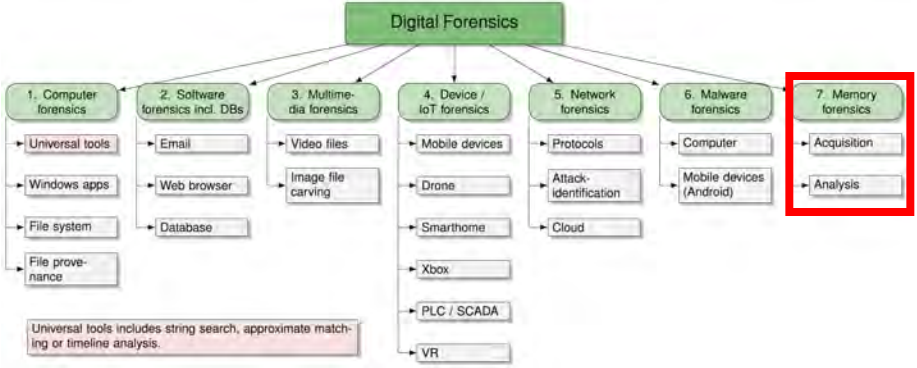
2 Current Challenges

3 From RAM to Courtroom

4 Conclusions

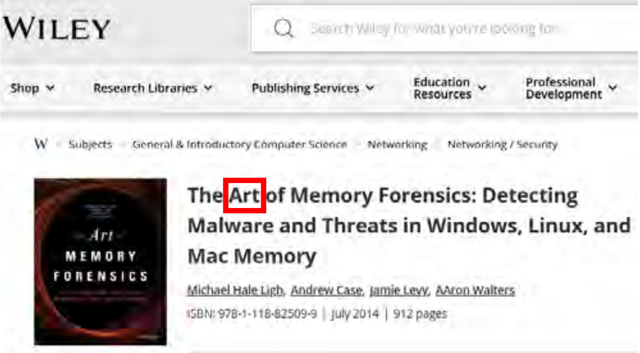
5 Future Directions

Introduction



Credits: *Digital forensic tools: Recent advances and enhancing the status quo*. Wu et al., doi: 10.1016/j.fsidi.2020.300999

Introduction



WILEY

Search Wiley for what you're looking for

Shop Research Libraries Publishing Services Education Resources Professional Development

W Subjects General & Introductory Computer Science Networking Networking / Security

The Art of Memory Forensics: Detecting Malware and Threats in Windows, Linux, and Mac Memory

Michael Hale Ligh, Andrew Case, Jamie Levy, AAron Walters

ISBN: 978-1-118-82509-9 | July 2014 | 912 pages

Credits: *The Art of Memory Forensics*, Ligh et al., Wiley, 2014. ISBN 978-1-118-82509-9.

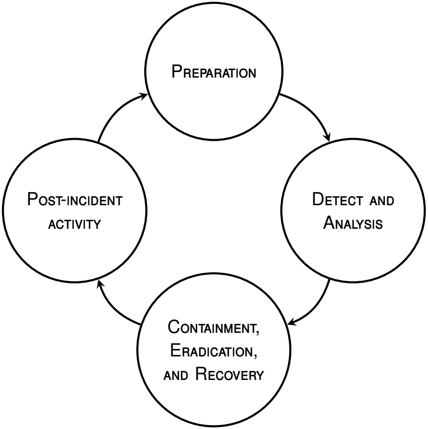
Introduction

Analysis of volatile memory to uncover evidence of system activity

- **Runtime insights**: Good complement to disk and network forensics
- Provides a snapshot of **"what was happening" at a specific moment**
- **Structured analysis of data** – *think in* `strings` *with steroids*

Introduction

Incident Response and Memory Forensics

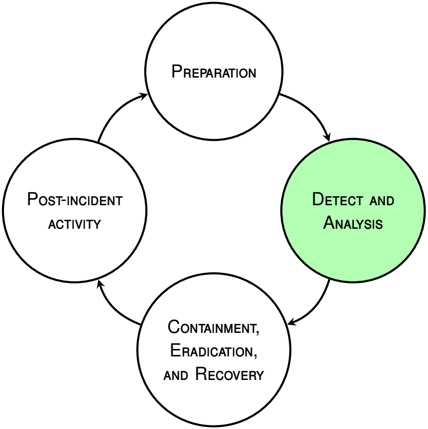


- **Live evidence**: RAM can be the only place where forensic artifacts exist
- **Runtime snapshot**: Shows what was really happening at runtime
- **Time sensitivity**: Evidence is volatile – *you must act fast*

(as defined by NIST)

Introduction

Incident Response and Memory Forensics

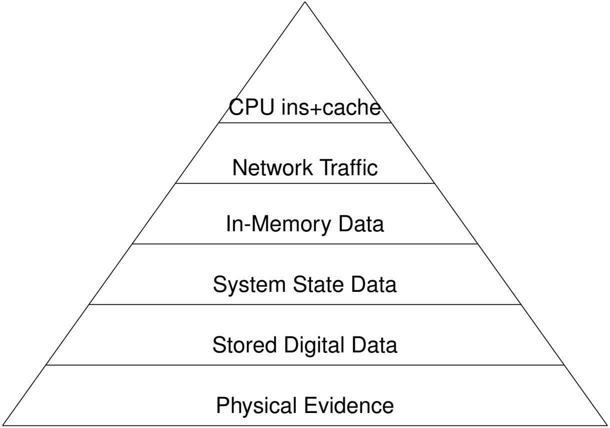


- **Live evidence**: RAM can be the only place where forensic artifacts exist
- **Runtime snapshot**: Shows what was really happening at runtime
- **Time sensitivity**: Evidence is volatile – *you must act fast*

(as defined by NIST)

Introduction

Evidence volatility



Introduction

What can we get in a memory dump?

- Running processes and hidden code (including injected code)
- Encryption keys and credentials (captured before they vanish)
- Command histories and user activity (e.g., shells, chats, clipboard data)
- Loaded drivers, DLLs, kernel structures (even malicious ones when hiding)
- Network connections in memory (active sessions, sockets)
- Open files and related data
- ...

Introduction

Why Memory Forensics Matters Today?

Modern malware often hides in memory only

Introduction

Why Memory Forensics Matters Today?

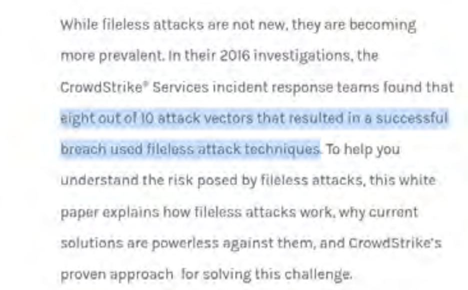
Modern malware often hides in memory only

- **Fileless attacks**
- **In-memory payloads**
- **Reflective DLL injection**
- **Process hollowing**

Disk-based analysis is no longer enough

Introduction

Why Memory Forensics Matters Today?

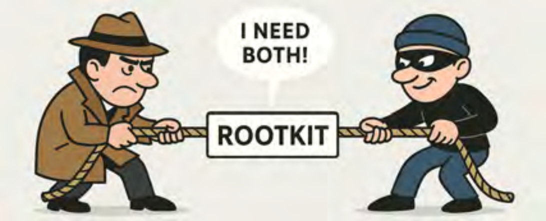


While fileless attacks are not new, they are becoming more prevalent. In their 2016 investigations, the CrowdStrike® Services incident response teams found that eight out of 10 attack vectors that resulted in a successful breach used fileless attack techniques. To help you understand the risk posed by fileless attacks, this white paper explains how fileless attacks work, why current solutions are powerless against them, and CrowdStrike's proven approach for solving this challenge.

Credits: *Who Needs Malware? How Adversaries use Fileless Attacks to Evade Your Security (CrowdStrike, 2017)*

Introduction

Why Memory Forensics Matters Today? – The Rootkit Paradox



A rootkit must reveal itself to control the system, but simultaneously must remain hidden to avoid detection

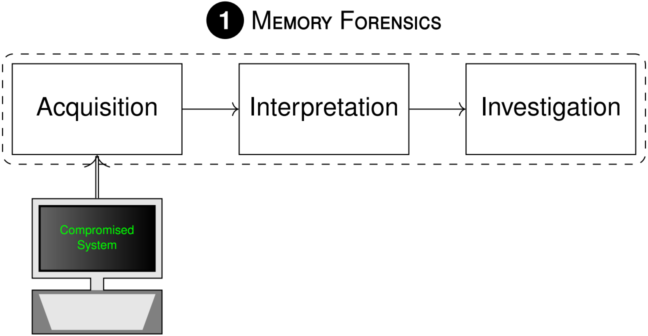
Further reading: *Exploiting the Rootkit Paradox with Windows Memory Analysis*, J. D .Kornblum, Int. J. Digit. Evid. 5(1), 2006.

Introduction

Why Memory Forensics Matters Today?

- **Live system visibility**
- **Operating System and application state analysis**
- **Credential and secret recovery**
- **Detection of covert channels**
- **Reverse engineering of evasive techniques**
- **Debugging complex software systems**

Introduction



Agenda

1 Introduction

2 Current Challenges

3 From RAM to Courtroom

4 Conclusions

5 Future Directions

Current Challenges

Snapshots ≠ atomic *(and that's normal)*

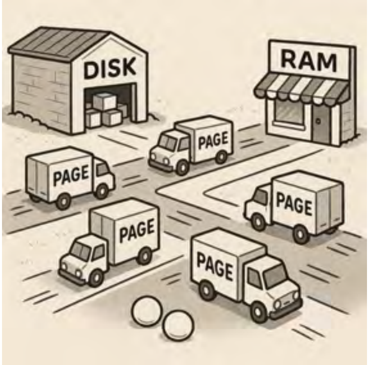
- Correctness, integrity, atomicity
  - Vömel and Freiling (DIIN, 2011)
- Page smearing, non-resident pages
  - Case & Richard (DIIN, 2017)
- Casual inconsistences and VAD-tree mismatches are common
  - Pagani et al. (ACM TOPS, 2019)
  - Rzepka et al. (DTRAP, 2024)
- Inconsistencies are **difficult to detect**

Current Challenges

Your tool, your choice

- **Different extraction tools**
  - Ruff (SSTIC 2017/JCVHT, 2008)
  - Latzo et al. (DIIN, 2019)
- Tool-dependent impact
  - Sylve et al. (DIIN, 2012)
  - Rzepka et al. (DFRWS EU/FSIDI, 2025)
- Anti-forensics
  - Baier & Knauer (IMF, 2014)
  - Zhang et al. (TIFS, 2018)
- Cross-version support
  - Oliveri et al. (NDSS, 2023)
  - Maggio et al. (DFRWS USA/FSIDI, 2021)
- **Benchmarking difficulties**

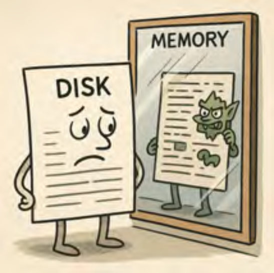
Current Challenges



Missing context

- Paging analysis
  - Richard & Case (DFRWS USA/DIIN, 2014)
  - Gruhn (IMF, 2015)
- Incomplete coverage
  - White et al. (DFRWS USA/DIIN, 2012)
  - Martín-Pérez & Rodríguez (ICDF2C, 2021)
- Process user-related metadata
  - Uroz et al. (to appear, 2026)
- **Risks of incomplete/inaccurate data**

Current Challenges



Trust in memory ≠ trust on disk

- Reconstruct before you trust (virtual vs physical layouts, relocations)
  - White et al. (DFRWS USA/FSIDI, 2013)
  - Martínez-Pérez et al. (COSE, 2021)
- Digital signatures are not enough
  - Uroz & Rodríguez (DFRWS EU/FSIDI, 2020)
- Shared pages reconstruction issues
  - Fernández-Álvarez & Rodríguez (DFRWS EU/FSIDI, 2023)
- **Untrusted evidence**

Current Challenges

Attach provenance to every claim

- “What this page is?”
  - Dolan-Gavitt (DFRWS USA/DIIN, 2007)
  - Van Baar et al. (DFRWS USA/DIIN, 2008)
  - Butler & Murdock (BH USA, 2011)
  - Cohen (DIIN, 2017)
  - Parida & Das (IJIS, 2020)
- Residual data
  - Dolan-Gavitt (DFRWS USA/DIIN, 2008)
  - Schuster (DFRWS USA/DIIN, 2008)
  - Beverly et al. (DFRWS USA/DIIN, 2011)
- App-level/OS inconsistencies
  - Kumar & Karabiyik (ISNCC, 2021)
  - Oliveri et al. (DFRWS EU/FSIDI, 2025)
- **Otherwise, it’s just hex**

Current Challenges

Big-brother view

- In-app data (and secrets!)
  - Maartmann-Moe et al. (DFRWS USA/DIIN, 2009)
  - Manna et al. (DFRWS USA/FSIDI, 2021)
  - Fernández-Álvarez & Rodríguez (DFRWS EU/FSIDI, 2022)
  - Ali et al. (DFRWS USA/FSIDI, 2025)
  - Ali et al. (DFRWS USA/FSIDI, 2025)
  - Abascal & Rodríguez (to appear, 2026)
- Code tampering
  - Pék et al. (ESORICS, 2016)
  - Block & Dewald (DFRWS USA/DIIN, 2019)
  - Case et al. (COSE, 2020)
  - Block (DFRWS USA/FSIDI, 2023)
  - *(I have few ideas here, contact me for further discussion* ☺*)*
- Cross-view correlations
  - Vömel & Lenz (IMF, 2013)
  - Aghaeikheirabady et al. (ICTCK, 2014)
  - Nagy (DFRWS USA/FSIDI, 2025)
- **Tool-specific, version-drift approaches**

Current Challenges

Emerging frontiers

- Scalability and DFaaS
  - Van Baar et al. (DFRWS EU/DIIN, 2014)
  - Van Beek at al. (DIIN, 2015)
  - Van Beek et al. (FSIDI, 2020)
  - Huici et al. (DFRWS USA/FSIDI, 2025)
- Confidential computing
  - Halderman et al. (CACM, 2009)
- Heterogeneous and emerging platforms (IoT, Cloud, drones, health devices, etc.)
  - Hay & Nance (ACM SIGOPS, 2008)
  - Nance et al. (ARES, 2009)
  - Hua & Zhang (CSMA, 2015)
  - Pichan et al. (DIIN, 2015)
  - Stoyanova et al. (COMST, 2019)
- **Scale-ready pipelines, new strategies and approaches**

Agenda

1 Introduction

2 Current Challenges

3 From RAM to Courtroom

4 Conclusions

5 Future Directions

From RAM to Courtroom

From challenges to claims

What we *know*	**How we make it *defensible***
Snapshots ≠ atomic	Report findings as *consistent with* measured conditions; log load & timings
Tool/runtime affect evidence	Justify tool choice; prefer fast capture; take confirmatory dump when feasible
Missing context	Capture RAM + pagefile/hiber; preserve PTE/PFN; record OS build/kernel
Trust in memory ≠ trust on disk	Reconstruct modules before trust; attach VAD→file provenance to each claim
Residual data and inconsistencies	Filter benign hooks; use cross-views
Big-brother view	Anonymize when possible; limit scope to case-relevant data; record tool versions
Emerging frontiers	Anticipate volatility; validate AI-assisted analysis; keep methods transparent

From RAM to Courtroom

Acquisition ⇒ Authority & necessity

Snapshots ≠ atomic; tool/runtime matter; anti-forensics possible

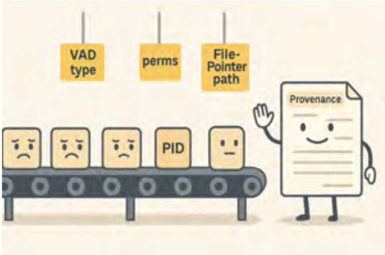
What we say (and document)

- **Necessity.** Volatile artifacts (keys, runtime state) likely to be lost on shutdown
- **Authority.** Warrant/consent/exigent rationale recorded on the form
- **Context.** Start/stop times, runtime, CPU/I/O load, OS build, kernel
- **Scope.** Process-scoped where lawful; why broader scope was/was not needed
- **Adjuncts.** Pagefile/hiber captured (or reason why not)

Universidad Zaragoza

From RAM to Courtroom

Interpretation ⇒ Explainability & Provenance



Trust in memory ≠ trust on disk; attach provenance to claims

What turns bytes into testimony

- **Reconstruct before trust:** handle relocs/shared pages/partials
- **Attach provenance:** VAD type (private/mapped), perms, FilePointer path, owning process. . .
- **Context-aware hits:** PFN→VA→process→file for any IoC
- **Missed moments:** infer via paging-state symptoms (PTE/PDE anomalies)
- **Plain-language gloss:** one sentence that a lawyer can read into the record

From RAM to Courtroom

Investigation ⇒ Validation & Repeatability

Hooks/tampering vs. benign;

version drift; cross-view needs

How we prove reliability

- **Exact-byte diffs** against Image Section Object; benign-hook allowlist (e.g., Chromium/Office/.NET)
- **Cross-view** checks for hidden KMs; independent sources must agree
- **Intermediate outputs** saved (maps, logs, diffs) to audit trail
- **Dual-tool confirmation** on key artifacts
- **Parser drift policy:** record build IDs; re-validate layouts; note tool versions

From RAM to Courtroom

Investigation ⇒ Proportionality & Minimization

Whole RAM vs. targeted processes/VADs

How we narrow and explain scope

- Default to **process-scoped** analysis where lawful; justify any expansion
- Record **VAD→file provenance** for every reported allocation
- Purpose limitation, retention window, and access controls stated in report
- Explicit list of *deliberately not examined* regions (and why)

Universidad Zaragoza

From RAM to Courtroom

Investigation ⇒ Be ready for the **now**

Big memory, confidential computing, and diverse platforms reshape acquisition & analysis

- **Scalability & performance.** Streaming acquisition; parallel triage; chunk hashing
- **Confidential computing.** SEV/TDX/CCA/SGX ⇒ blind spots; attestation logs, policy docs, and provider APIs as *adjunct evidence*
- **Heterogeneous targets.** Cloud/VM/container/serverless; mobile/IoT with secure enclaves and DMAs; OS-agnostic profiles
- **Automation with accountability.** Assisted-ML ⇒ reproducibility, explainability and drift under control
- **At scale governance.** DFaaS: multi-tenant isolation, chain of custody audits

Agenda

1 Introduction

2 Current Challenges

3 From RAM to Courtroom

4 Conclusions

5 Future Directions

Conclusions

Three takeaways

What we learned

1. Snapshots ≠ atomic ⇒ log context; **report as *consistent with* conditions**
2. Attach provenance: **VAD** → **file** → **process** (reconstruct before you trust)
3. *Defensibility by design*: **exact byte diffs** or **independent cross-views**, plus validation & minimization

Conclusions

Admissibility by design

Law-ready language

- **Necessity:** Volatile artifacts were likely to be lost on shutdown; live capture was required
- **Limits:** RAM captures are not perfectly atomic; findings are *consistent with* measured conditions
- **Provenance:** For each item we report VAD type, perms, owning process, and original file path
- **Integrity:** Key artifacts were independently confirmed; byte-level diffs and logs are preserved
- **Scope:** Analysis was limited to processes/mappings relevant to the mandate; unrelated regions were not examined

Agenda

1 Introduction

2 Current Challenges

3 From RAM to Courtroom

4 Conclusions

5 Future Directions

Future Directions

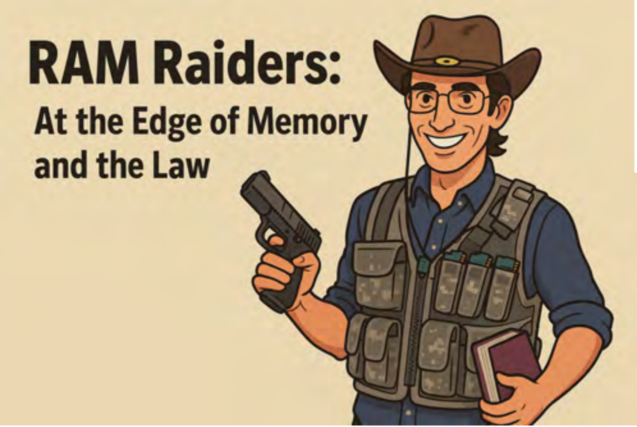
- **Technical**: Smarter, faster, and more reliable memory forensic tools
  - Handle multi-terabyte dumps, streaming acquisition
  - Combine memory with disk/network/cloud forensics
  - Scale memory forensic soundness to new platforms

Future Directions

- **Technical**: Smarter, faster, and more reliable memory forensic tools
  - Handle multi-terabyte dumps, streaming acquisition
  - Combine memory with disk/network/cloud forensics
  - Scale memory forensic soundness to new platforms
- **Legal**: Clear frameworks for the admissibility of volatile evidence
  - Lack of harmonization/methodologies for volatile evidence
  - Chain of custody in volatile contexts
  - Train judges/lawyers to understand memory forensic experts

Future Directions

- **Technical**: Smarter, faster, and more reliable memory forensic tools
  - Handle multi-terabyte dumps, streaming acquisition
  - Combine memory with disk/network/cloud forensics
  - Scale memory forensic soundness to new platforms
- **Legal**: Clear frameworks for the admissibility of volatile evidence
  - Lack of harmonization/methodologies for volatile evidence
  - Chain of custody in volatile contexts
  - Train judges/lawyers to understand memory forensic experts
- **Community**: Researchers + practitioners + policymakers
  - Shared (sanitized) datasets
  - Cross-disciplinary workshops
  - Education and training



(PDF slides)

Ricardo J. Rodríguez
Universidad de Zaragoza

14th Int. Conf. on IT Security Incident Management & IT Forensics (IMF 2025)

Albstadt, Germany

All cartoons shown in this presentation were generated using ChatGPT