ZILLNER
IT SECURITY

Memory Forensics using Virtual Machine Introspection for Cloud Computing

Tobias Zillner, BSc MSc MSc

ABOUT ME

- Tobias Zillner, BSc MSc MSc
  - Vienna, Austria
  - Founder of Zillner IT-Security
  - Independent Security Consultant & Researcher
    - Consulting, Audit, Advisory, Training
  - Security Research
    - Internet of Things, Smart Homes
    - Wireless Security
  - www.zillner.tech
- SDR Enthusiast

ZILLNER IT SECURITY

WHAT IS IT ABOUT?

AND WHY DO WE NEED IT?

ZILLNER
IT SECURITY

OUTLINE

- Introduction & Background
- Virtual Machine Introspection (VMI)
- Use cases
- Prototype
- Summary

ZILLNER IT SECURITY

MOTIVATION

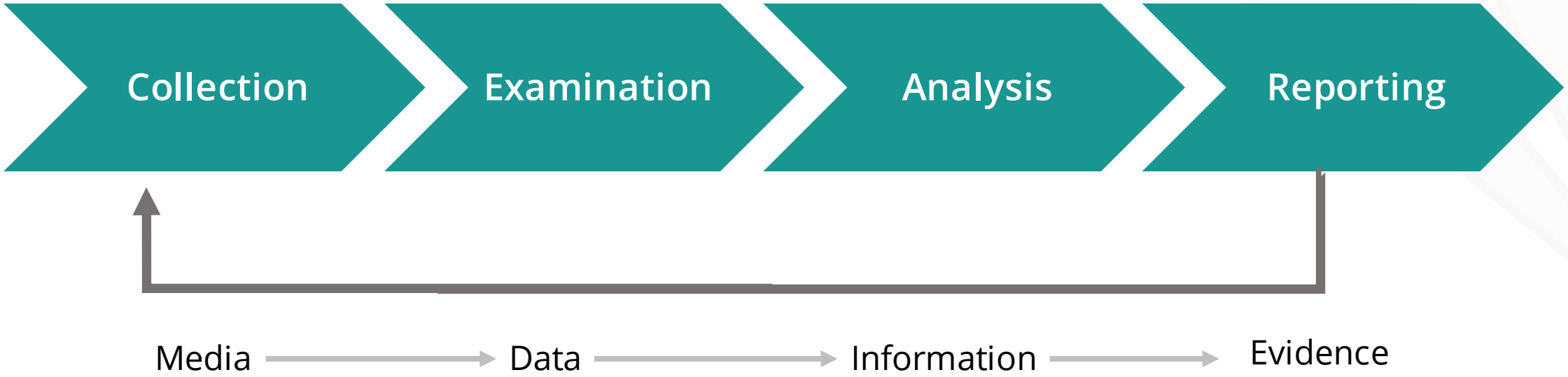
- Relocation of systems and services into cloud environments is on the rise
- Users loose direct access / control over their systems
- Forensic methods are limited in the cloud
- Enable the user to perform their own forensic investigations
- Forensic as a Service

ZILLNER IT SECURITY

MEMORY FORENSICS & VIRTUAL MACHINE INTROSPECTION

ZILLNER IT SECURITY

FORENSIC PROCESS

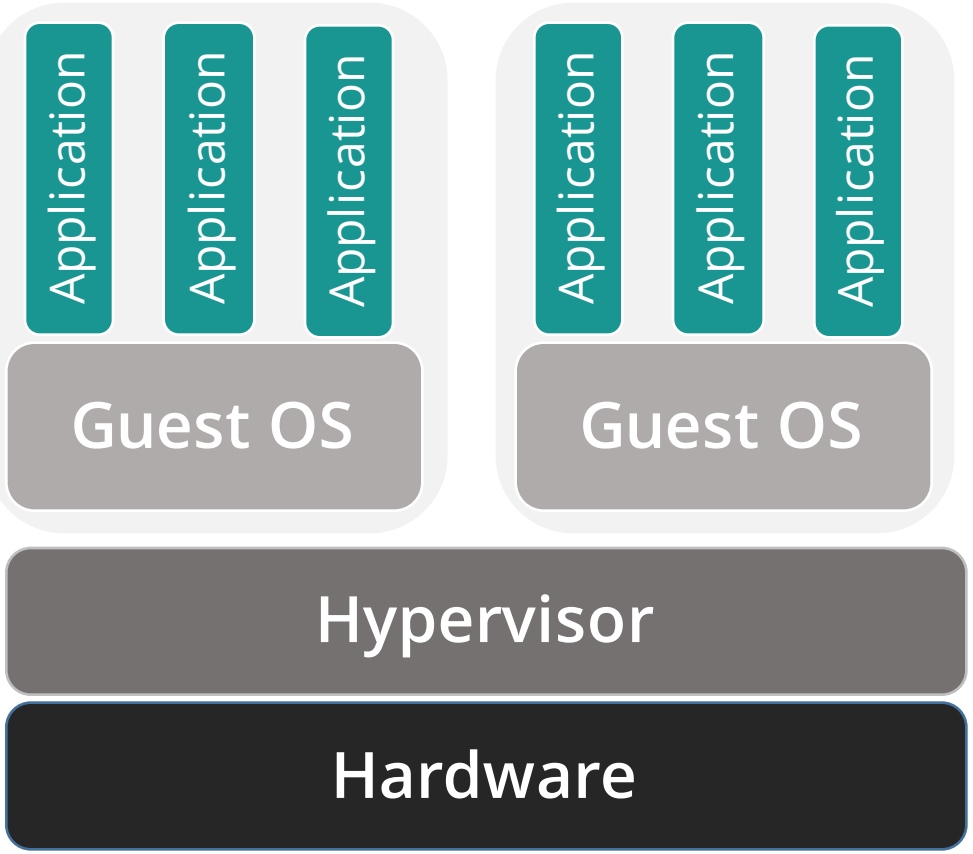


ZILLNER IT SECURITY

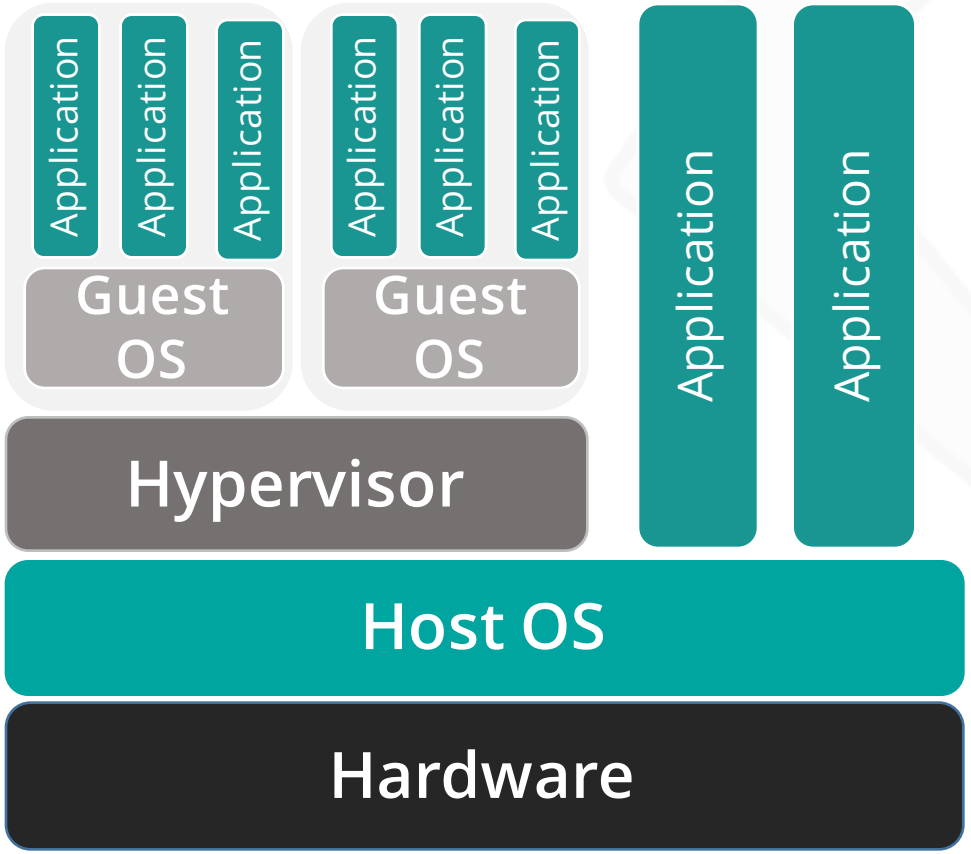
HARDWARE VIRTUALIZATION

- One / Multiple guest OS on virtualized hardware
- Managed by Virtual Machine Monitor (VMM) – Hypervisor
- Provides interfaces and controls interactions with hardware
  - CPU, memory, network, storage,…
- Hypervisor on own OS – Host OS

NATIVE VS. HOSTED VIRTUALIZATION



Native virtualization



Hosted virtualization

ZILLNER IT SECURITY

VIRTUAL MACHINE INTROSPECTION

- *"Virtual Introspection (VI) is the process by which the state of a virtual machine (VM) is observed from either the Virtual Machine Monitor (VMM), or from some virtual machine other than the one being examined. "* [1]

[1]: Brian Hay and Kara Nance. Forensics examination of volatile system data using virtual introspection. SIGOPS Oper. Syst. Rev., 42(3):74 82, April 2008

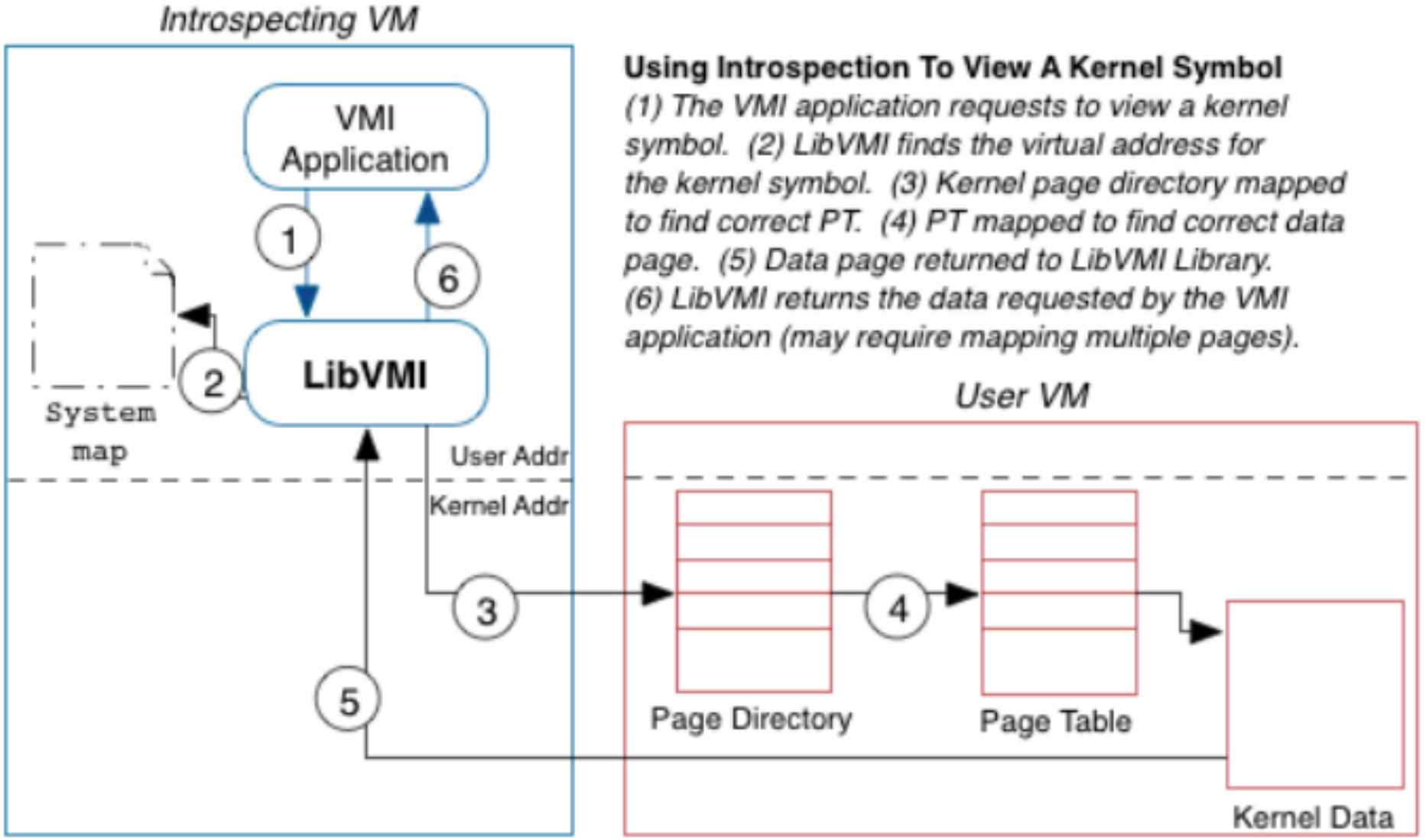
ZILLNER IT SECURITY

SEMANTIC GAP

- Difference between the presentation of data from volatile memory by the OS and the raw data format
- Requires VMI to perform the same translation of the the raw memory data as the OS
- At least some knowledge about the guest OS is necessary

ZILLNER IT SECURITY

HOW DOES IT WORK?



Using Introspection To View A Kernel Symbol

(1) The VMI application requests to view a kernel symbol. (2) LibVMI finds the virtual address for the kernel symbol. (3) Kernel page directory mapped to find correct PT. (4) PT mapped to find correct data page. (5) Data page returned to LibVMI Library. (6) LibVMI returns the data requested by the VMI application (may require mapping multiple pages).

http://libvmi.com/docs/gcode-intro.html

ADVANTAGES

- No altering of the target system
- Very hard to detect the monitoring
- Live analysis of memory content
- Data size for analysis (storage much larger than memory)
- Detection of advanced memory only malware
- More reliable data
  - No data corruption through malware

ZILLNER IT SECURITY

COUNTERMEASURES

- Detection
  - **Timing analysis -** unusual patterns in the frequency at which it is scheduled for execution
  - **Page fault analysis** - the target VM may be able to detect unusual patterns in the distribution of page faults

- Direct Kernel Structure Manipulation (DKSM)
  - VMI assumes that OS implement certain kernel- and data structures
  - DKSM modifies this structures and prevents monitoring
  - **Sytanx based:** targeted deletion/addition/manipulation of data structures
  - **Sematic:** semantics of the data structures are changed
  - **Combined:** mix of syntax and semantics manipulation

ZILLNER IT SECURITY

FIELDS OF APPLICATION

EXAMPLES

- Rootkit detection
  - Manipulation of memory access
  - Interception of system calls
- Cryptographic key extraction
  - On the fly encrypted container
  - Network forensics
- IDS / IPS

ZILLNER IT SECURITY

PROTOTYPE

ZILLNER
IT SECURITY

SOLUTION APPROACH

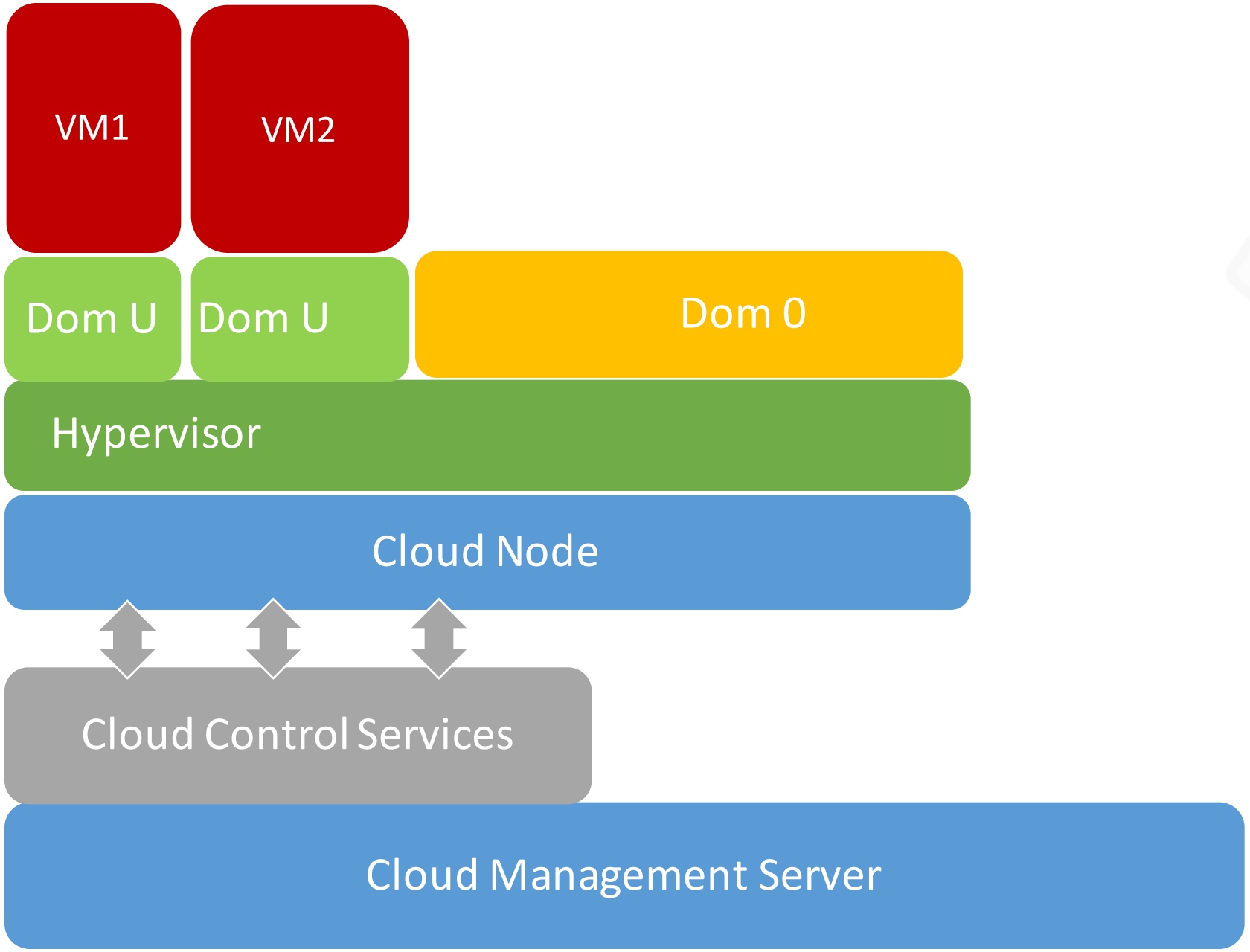
- Combining existing tools for a novel approach
- Open Source
- Minimal overhead
- Transparent for the user

ZILLNER IT SECURITY

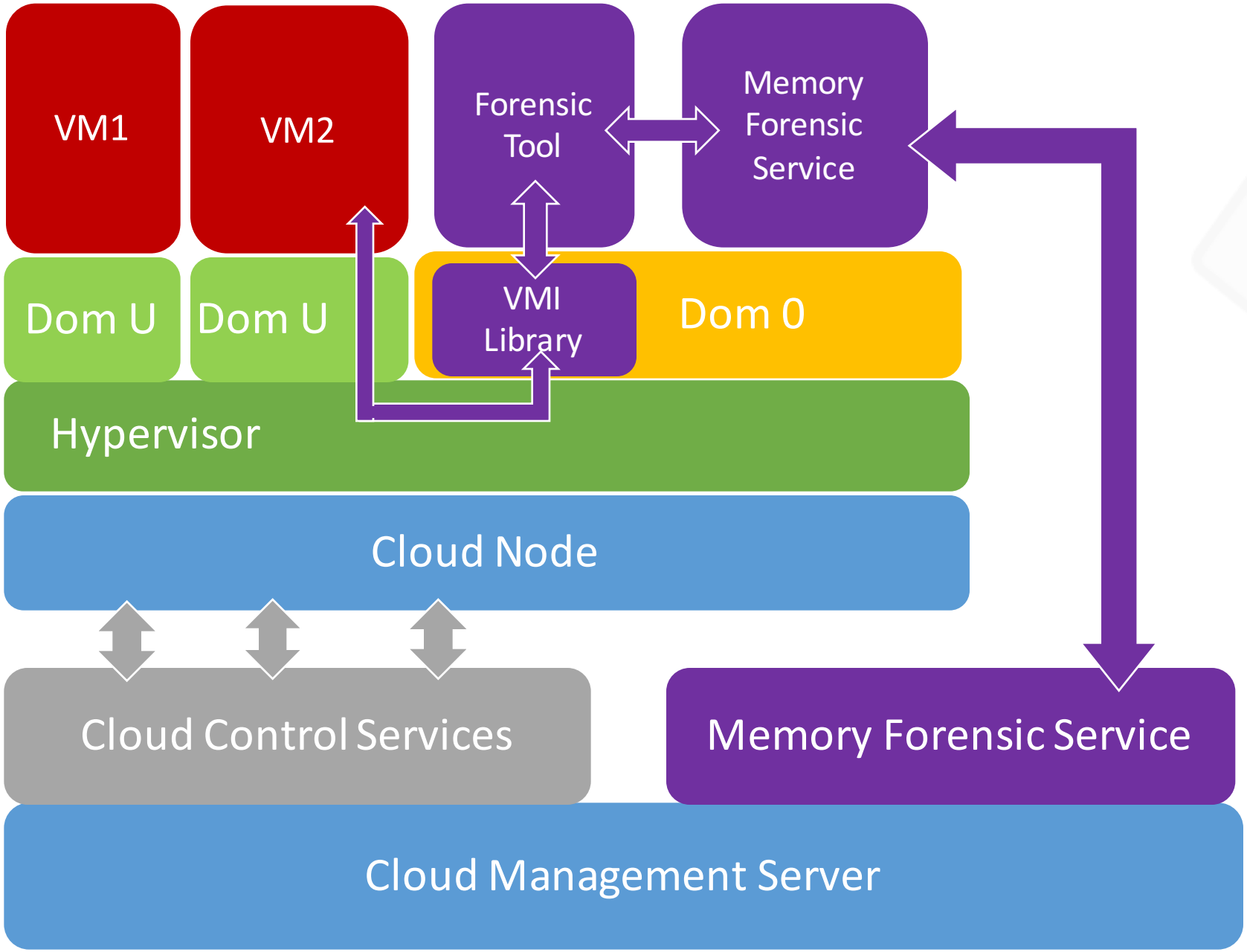
ARCHITECTURE

- Cloud Solution
  - Open Nebula
- Cloud Management Server
- Cloud Node
- Host OS – Ubuntu
- Guest VM
- Memory Forensic Services
- VMI Library – LibVMI
- Forensic Tool – Volatility
- Hypervisor - Xen

ZILLNER IT SECURITY

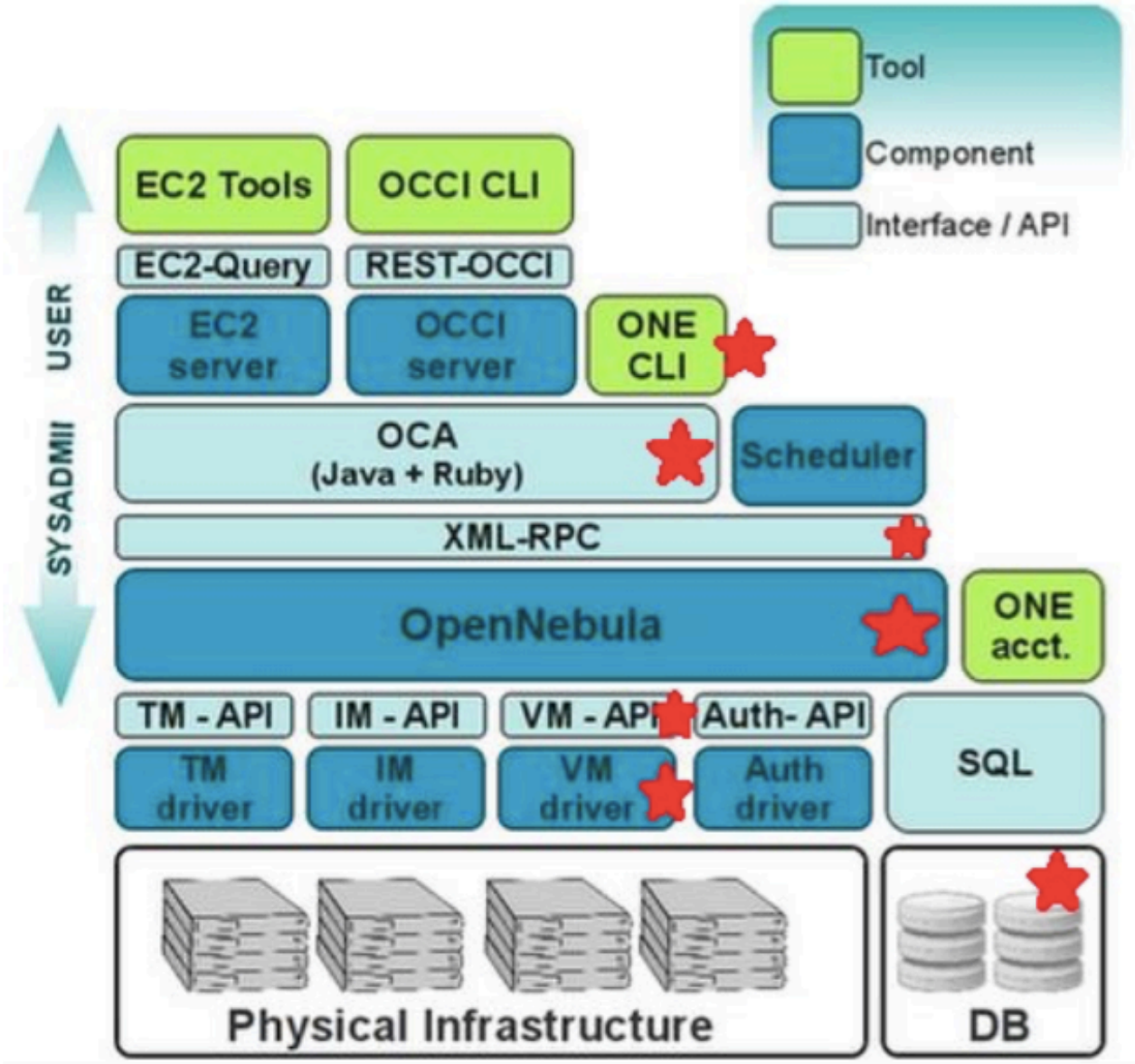


ZILLNER IT SECURITY



ZILLNER IT SECURITY

OPEN NEBULA EXTENSIONS



www.opennebula.org

ZILLNER IT SECURITY

MEMORY FORENSIC SERVICES

- Self developed management and control services
- Client – Server model
- Platform independent
- PKI for secure communication
- Command whitelisting

```
1 onevm memfor <VMID> <Volatiliy Profil> <Volatility Kommando>
```

ZILLNER IT SECURITY

FORENSIC PROCESS

Media → Data → Information → Evidence

ZILLNER IT SECURITY

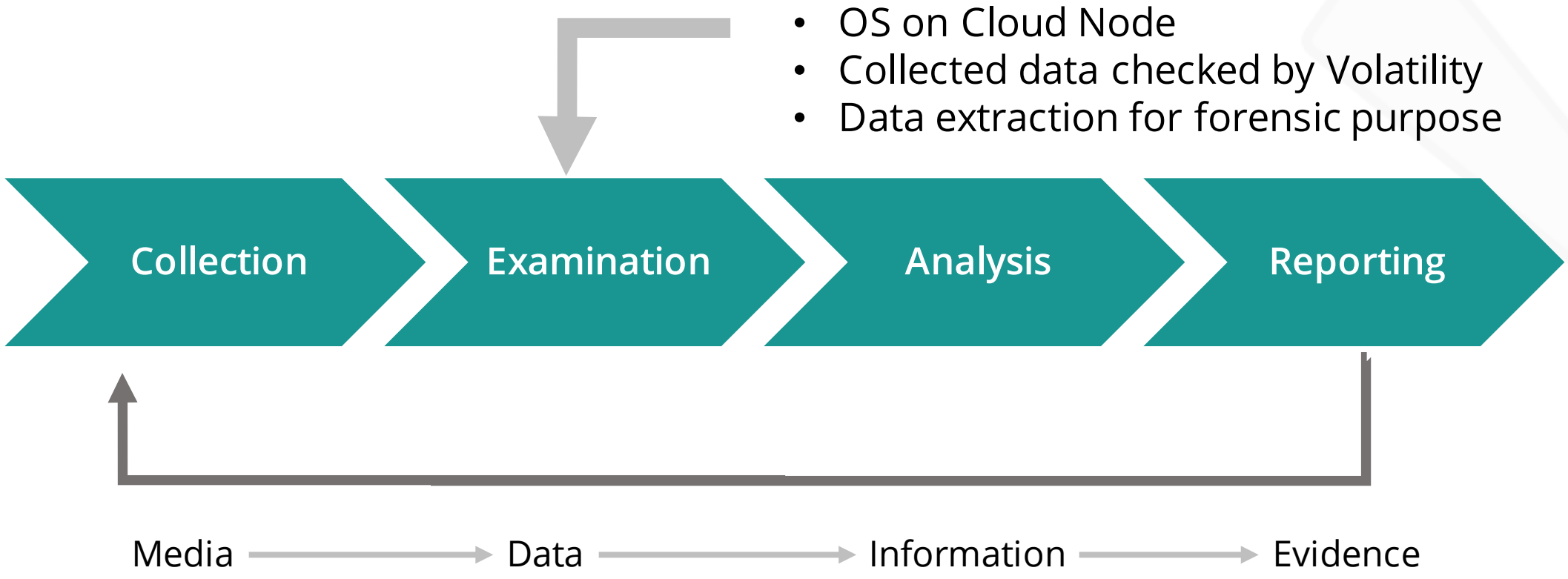
FORENSIC PROCESS

- OS on Cloud Node
- Data provided by LibVMI
- Collected by Volatility

Media → Data → Information → Evidence

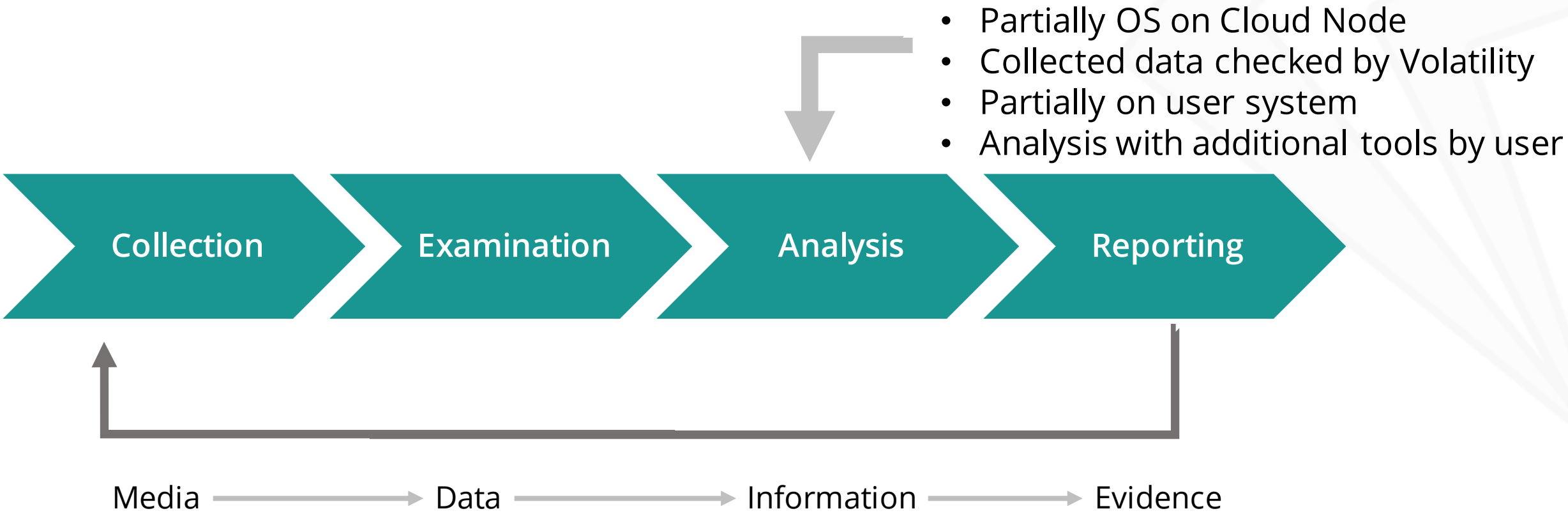
ZILLNER IT SECURITY

FORENSIC PROCESS



ZILLNER IT SECURITY

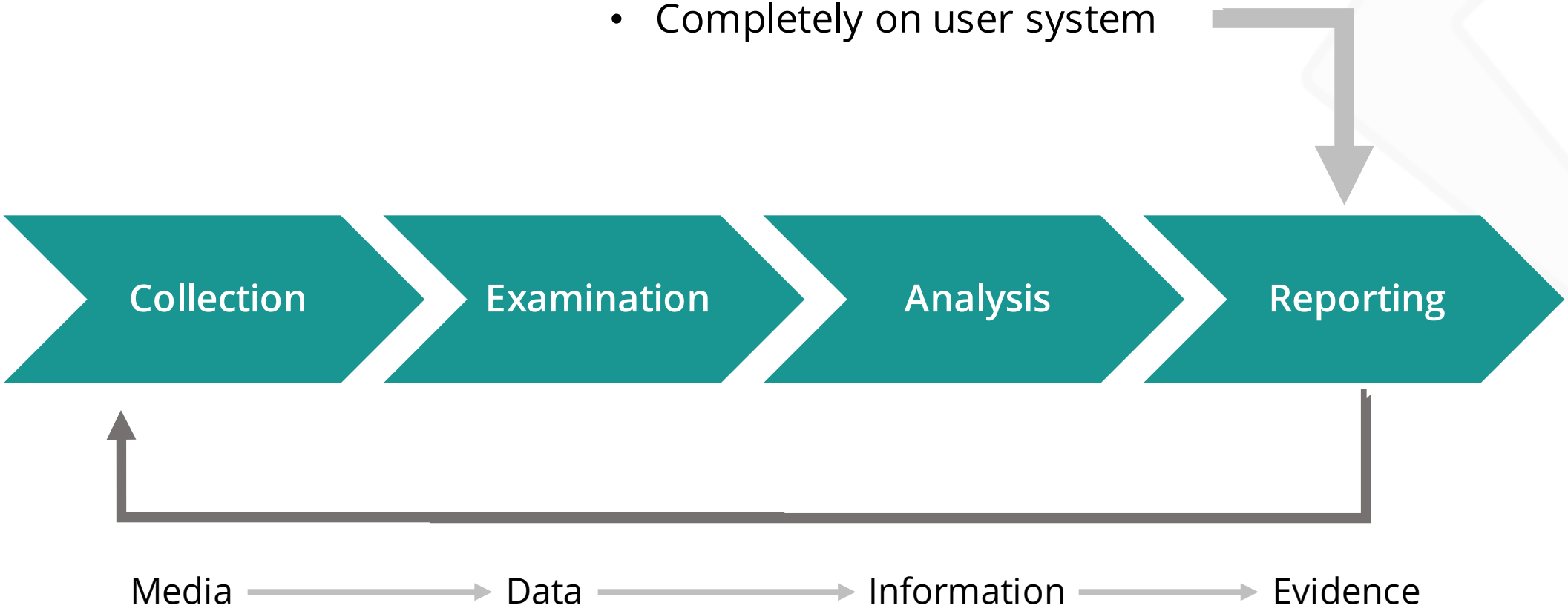
FORENSIC PROCESS



ZILLNER IT SECURITY

FORENSIC PROCESS

- Completely on user system



ZILLNER IT SECURITY

ADVANTAGES

- User gets easy access to the data
- No changes on the target VM necessary
- Memory analysis not on the possibly compromised system
- No stop/pausing of the analyzed machine required
- Operation of the VM does not get influenced
- Analysis can be done either local or over the network
  - Reduction of local load / network load
- Usage of existing authentication and authorization system

ZILLNER IT SECURITY

DISADVANTAGES

- Configuration necessary
- Knowledge about the guest OS required
- Installation overhead for cloud provider

- Additional attack surface
- Security is crucial for the added services
- User segregation is very important

ZILLNER IT SECURITY

LIBVMI CONFIG EXAMPLE

```
1  UbuntuLucid {
2          sysmap = "/usr/local/libvmi-0.8/Systemmaps/vm1lucid/
               System.map-2.6.32-45-server";
3          ostype = "Linux";
4          linux_name = 0x490;
5          linux_tasks = 0x258;
6          linux_mm = 0x290;
7          linux_pid = 0x2b8;
8          linux_pgd = 0x50;
9          linux_addr = 0x100;
10 }
```

ZILLNER IT SECURITY

VOLATILITY / LIBVMI USAGE

```
1 python vol.py -l vmi://win7 pslist  # win7 is the target
```

ZILLNER IT SECURITY

USE CASE

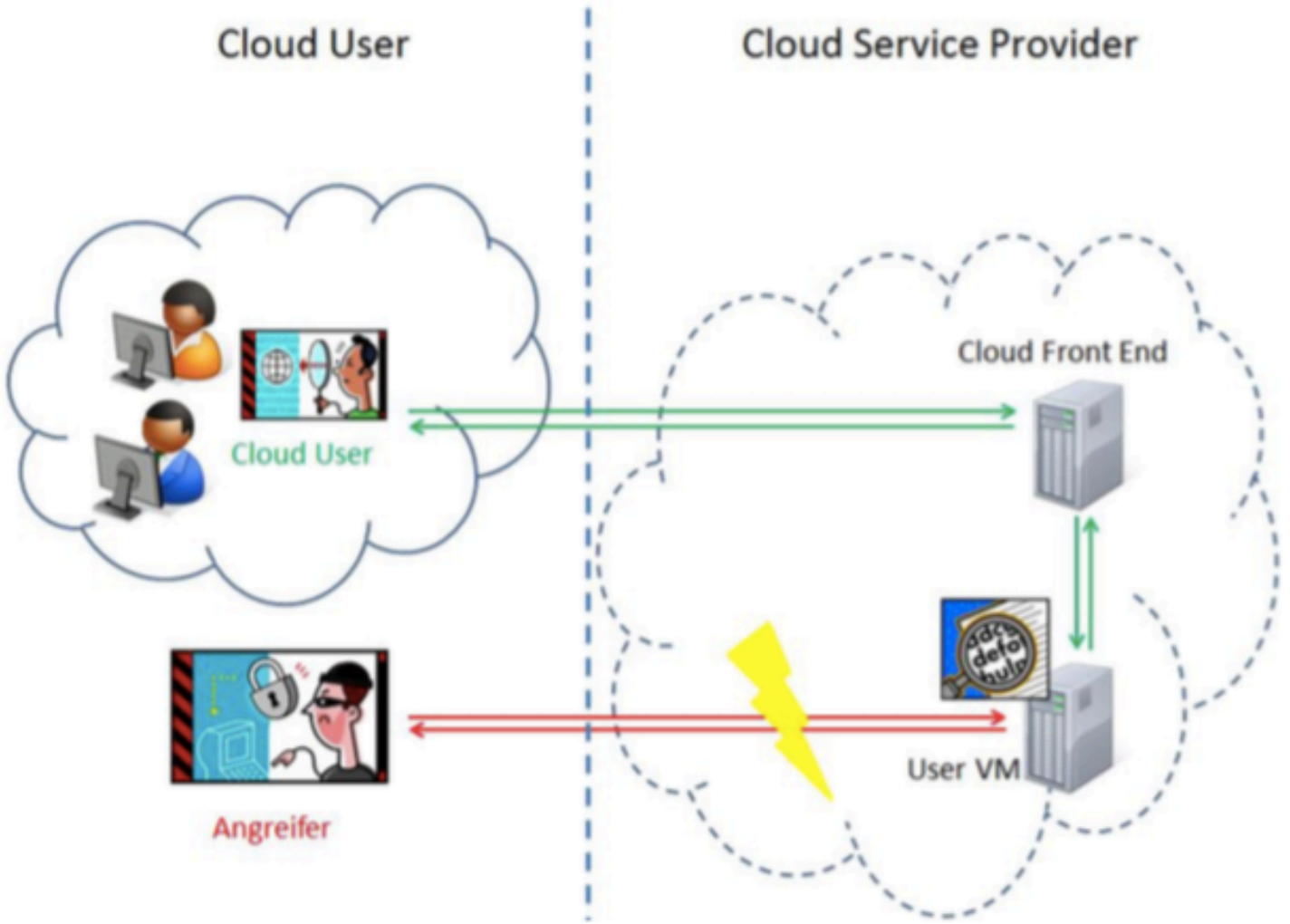
KERNEL LEVEL ROOT KIT DETECTION

- Modifying of data structures, which display the processes currently running on the system
- System call interception
- Interrupt hooking
- Modifying the kernel memory image
- Intercepting calls handled by the VFS
- Virtual memory subversion

ZILLNER IT SECURITY

USE CASE

ENDUSER VM IN IAAS CLOUD



ZILLNER IT SECURITY

DEMO

ZILLNER
IT SECURITY

SUMMARY

ZILLNER
IT SECURITY

SUMMARY

- Investigations in cloud environments get more and more common
- Hypervisor forensics VMI is a very interesting solution approach
- Fully Open Source based working prototype
- Enables fast responses to security incidents
- Lot of room for enhancements
- Different use cases for VMI in clouds possible

ZILLNER IT SECURITY

BLACK HAT SOUND BYTES

- Hypervisor forensics / VMI are very powerful and interesting technologies
- FaaS gives power to the end user
- Memory analysis is a huge benefit for forensic investigations

ZILLNER IT SECURITY

Q & A

Please fill out the Black Hat Feedback Form

ZILLNER
IT SECURITY

Contact

Tobias Zillner
tobias@zillner.tech
www.zillner.tech
+43 664 8829 8290