Hochschule Wismar

Fakultät für Ingenieurwissenschaften

WINGS Studiengang IT Sicherheit und Forensik

Master-Thesis

Capabilities of Live/Memory-Forensics on Compartmentalized Systems Using Bromium, Qubes and Proxmox as Examples

Eingereicht am:	31. August 2019
von:	Ute Schüller
Matrikelnummer:	XXXXXX
Erstgutachter:	Prof. Dr.-Ing. A. Raab-Düsterhöft
Zweitgutachter:	Prof. Dr.-Ing. A. Ahrens

Scope of Work

With the constantly increasing digitalization of daily life, there is also a rising number of attacks, against private as well as company computers. One of the options to reduce those risks, is the usage of operating systems or software, that provide compartments (aka. sandboxes or containers), when working on a network-connected computer. Despite the reduced risks, sometimes the need for live/memory-forensics arises on those systems, as well. How to perform such an analysis, is well-documented for e.g. VMware; nevertheless, for Bromium, Qubes and Proxmox, there is not much literature available. Consequently, the aim of this master thesis is to investigate the available options for those 3 operating systems. On one Hand, this includes examining the systems for already present IT-forensic capabilities, and, on the other hand, investigating additional possibilities to collect live/memory forensic data, that can be gathered, using freely available software like Rekall or Proxmox. Additionally, commercial software (Axiom, EnCase, X-Ways and Nuix) should be looked at, to find out, which possibilities are available, within those programs. Finally, if possible, a half-automated data collection on Qubes should be implemented, for later IT-forensic investigation.

In GERMAN:

Titel:

Möglichkeiten der Live/Memory-Forensik auf kompartimentierten Systeme am Beispiel von Bromium, Qubes und Proxmox

Aufgabenstellung

Mit der stetig wachsenden Digitalisierung des täglichen Lebens, wächst auch die Anzahl der Angriffe auf private, wie auch auf Firmen-Computer. Eine der Möglichkeiten den entsprechenden Risiken entgegenzuwirken, ist die Nutzung von Betriebssystemen oder Software, die Kompartimente (bzw. Sandboxen oder Container) zur Verfügung stellen. Trotz der geringeren Risiken besteht auch auf solchen Systemen immer wieder die Notwendigkeit zur Live-/Memory-Forensik. Diese ist z.B. für VMware recht gut dokumentiert, jedoch gibt es wenig entsprechende Literatur zu Bromium, Qubes und Proxmox. Daher ist es das Ziel dieser Masterarbeit, die Live/Memory-forensischen Möglichkeiten auf diesen drei Betriebssystemen zu untersuchen. Einerseits sollen dabei die bereits vorhandenen Analyse- bzw. Auswertungsmöglichkeiten der Systeme betrachtet werden und andererseits, welche weiteren Möglichkeiten sich bei Nutzung von frei verfügbarer Software wie Rekall und Volatility ergeben. Zusätzlich soll betrachtet werden inwieweit kommerzielle Software, wie Axiom, EnCase, X-Ways und Nuix für eine entsprechende Untersuchung in Frage kommen. Darüber hinaus, sollte möglichst eine halb-automatisierte Datensammlung in Qubes implementiert werden, die dann für spätere IT-forensischen Untersuchungen genutzt werden kann.

Abstract

This thesis discusses memory forensics on compartmentalized systems. For specialized systems like Bromium, Qubes and Proxmox there is not much related literature available. Consequently those 3 operating systems should be focused.

With the focus on the specials of compartmentalized systems, possible options, like creating memory images within the guest systems and performing memory forensics on those images or live analysis within the VMs, will not be investigated, as an analysis of this type, does not much differ from an analysis performed for a non-compartmentalized system. Additionally, as this work tries to find possible options for memory forensics in these environments, it does not focus on gathering court-proof evidence.

As has been found during this work, memory forensics is very fragile, slight changes in installed libraries, kernel versions, etc. might immediately break the whole process. On the windows based Bromium installation, there are the best chances, to gather memory images and to perform virtual machine introspection (VMI). On Proxmox, it is at least possible, to inspect the running host system; whereas tools like vmscan in Rekall failed to find the EPT values for further inspection of the running VMs. With Qubes, it was not even possible to generate reliable memory images and consequently, none of the tools was able to perform memory forensics. And without any possibility to generate reliable memory images on Qubes (not even manually), implementing an automated approach was not possible, either.

Kurzreferat

Diese Masterarbeit behandelt Memory Forensik auf kompartimentierten Systemen. Für spezialisierte Systemen, wie Bromium, Qubes und Proxmox gibt es nur wenig entsprechende Literatur. zu. Daher sollen diese drei Systeme untersucht werden.

Da der Fokus auf den Besonderheiten von kompartimentierten Systemen liegt, werden mögliche Optionen, wie das Erzeugen von Memory Images innerhalb von Gast-Systemen (VMs) und das Analysieren dieser Images oder das Durchführen von Live Memory Forensik innerhalb der VMs nicht betrachtet, da sie kaum Unterschiede zur Analyse auf nicht-kompartimentierten Systemen aufweisen. Auch will diese Arbeit Möglichkeiten der Memory Forensik in kompartimentierten Umgebungen finden. Der Fokus liegt hingegen nicht darauf, dass die Ergebnisse bereits gerichtsverwertbare Beweise liefern.

Wie bei dieser Arbeit herausgefunden wurde, ist Memory Forensik eine sehr fehleranfällige Sache. Bereits kleine Abweichungen bei den installierten Libraries oder Kernel Versionen, können umgehend dazu führen, dass der gesamte Analyse-Prozess nicht mehr funktioniert.

Auf einer windows-basierten Bromium Installation bestehen generell die besten Möglichkeiten, Memory Images zu erzeugen und die zugehörigen virtuellen Maschinen zu inspizieren. Auf Proxmox ist es immerhin möglich, das Host-System zu analysieren, jedoch sind Tools wie vmscan dort nicht in der Lage, EPT-Werte zur weiteren Untersuchung der laufenden VMs zu finden. Und mit Qubes war es noch nicht einmal möglich, vertrauenswürdige Memory Images zu erzeugen und somit war keines der verwendeten Tools in der Lage, hier Memory Forensik durchzuführen. Ohne die Möglichkeit (wenigstens manuell) vertrauenswürdige Memory Images zu erzeugen, war es folglich auch nicht möglich, diese automatisch zu generieren.

Table of Content

1 Introduction

1.1 Motivation

With the constantly increasing digitalization of daily life, there is also a rising number of attacks, against private as well as company computers. One of the options to reduce those risks, is the usage of operating systems or software, that provide compartments (aka. sandboxes or containers), when working on a network-connected computer. In practice, this means, running certain tasks, like reading emails, unpacking ZIP-files or browsing WEB-pages within compartments or containers. Consequently, the corresponding processes are not allowed to access all resources, that the user and the operating systems would have access to, but only those resources, that are allowed to be accessed by the corresponding compartment.

As an example, there is an email attachment warning (in German), that was release in August 2018 by LKA Niedersachsen (it references job application emails, that have a ZIP attachment, which contains an executable, that then tries to encrypt the hard disk of the user) [1]. Similar warnings are available e.g. by the FBI [2] .

In an environment, that uses compartmentalization, the email program might be granted access to the network (to access the email server) as well as to email related files, but not to any other files of the user, e.g. other office documents like Word or Excel files. Consequently, the executable might be able to encrypt the email related files, but without the needed access, it is not able to encrypt any additional files, like other documents or pictures of the user. Therefore, it is impossible, that this program encrypts the complete harddisk.

Even, if the risks are reduced, when using compartmentalization, sometimes the need for live/memory-forensics arises on those systems, as well. How to perform such an analysis, is relatively well-documented for e.g. VMware; nevertheless, for Bromium, Qubes and Proxmox, there is not much literature available.

1.2 Goals

For standard Windows or Linux systems, there are several options for forensic memory analysis, at least, if these operating systems are somehow standard and not too new. Even for compartmentalized systems, using Xen or KVM, there are some tools available. But how does it look with latest versions of these operating systems? And especially with systems like Bromium, Qubes and Proxmox, that are specialized on security and surely not as widely spread as standard KVM or Xen implementations?

The aim of this master thesis is to investigate the available IT forensic options for the three compartmentalized systems:

- Bromium (installed on Windows)
- Qubes
- Proxmox

On one hand, it should examine, whether these systems provide IT forensic information on their own. On the other hand, this thesis investigates, whether live/memory forensic data can be gathered on these systems, with freely available tools, like Rekall and Volatility and also with commercial tools like X-Ways, Axiom and Nuix.

The goal of this thesis is to look at possible options for memory analysis on these systems, whereas it is NOT, to make sure, that these options provide court-proof evidence.

1.3 Project Partners

- Dipl.-Wirt.-Inf. Martin Wundram (DigiTrace); tutor
- Bromium

1.4 Acknowledgments

Special thanks to Mr. Wundram, for the idea and subject for this thesis and for establishing the contact to Bromium.

Many thanks to Bromium and especially to my personal contact, Mr. Dominik Vidas, who provided the test setup for the Bromium software and also provided answers to many Bromium related questions.

Without their very appreciated help, this work would not have been possible.

1.5 Structure of this thesis

This thesis will start with a description of fundamental terms, the operating systems, that are installed on the test systems, short descriptions of used (memory) forensics software and tools for memory image creation.

The next section will focus on tests, whether and how the forensic software can be used to gather live/memory forensic information on the test systems, followed by a section, focusing on the specials in relation to memory image collection on Qubes.

And finally, the thesis will be closed with an evaluation of the test results.

2 Basic Information

2.1 Fundamental Terms

Compartment: In general a compartment is a room, section or some part of a subdivided area [3]. In relation to operating systems, it refers to solutions provided by tools like virtual machines or containers, that separate applications from each other, within the operating system.

Compartmentalization: in computer science, "means organizing resources into groups (also called compartments or zones). each of which is isolated from the others" [4, p. 4]. It is a technique, that is widely used e.g. in networking, where it refers to separating the network into different subsections or in operating systems, where it separates application from each other, by using tools like virtual machines, or containers.

Virtual Machine: is a computer process, which runs a complete (virtual) operating system on virtualized hardware, within a host operating system (OS), which has the physical HW and real HW-access. Several virtual machines can run within the same host operating system. And even nested configurations, with a hypervisor running within a virtual machine, are sometimes possible.

Hypervisor: is the process, that handles one or several virtual machines within one host operating system.

Hardware Virtual Machine: "A Hardware Virtual Machine allows the host OS to not be aware of the existence of the Virtual Client. The Virtual Client will require no modification to run on the Host OS. In the past, speed to network and storage had lower performance since the hardware access was being emulated. The case is no longer true and HVM is equal, if not better than PV. HVM requires a CPU which supports virtualization, either Intel VT-x or AMD-V."[5]

Paravirtualization (PV): "The VC or Guest requires modifications to run properly. The modifications are meant to allow hardware access at near native speed as if the Guest were the Host OS. Each Linux kernel used in PV are specifically designed to be used for PV. The kernel is not a typical kernel but modified. PV does not require the use of a CPU which supports virtualization. The OS itself must support PV."[5]

Container: Within an operating system, a container is a subsection of an operating system. Its content is somehow separated from the rest of the operating system. In contrary to a virtual machine, it does not run a complete operating system; rather, it shares the kernel with the host, and only runs separate libraries and application processes and typically has somehow limited permissions.

Container Engine: is the process, that handles one or several containers within the same host operating system.

Computer Forensics: "Computer forensics is the practice of identifying, extracting and considering evidence from digital media such as computer hard drives. Digital evidence is both fragile and volatile and requires the attention of a certified specialist to ensure that materials of evidentiary value can be effectively isolated and extracted in a scientific manner that will bear the scrutiny of a court of law." [6] Sometimes Computer Forensics is also referred to as IT Forensics.

Live Forensics: is a subdiscipline of computer forensics, that is focused on analyzing live/running computer systems.

Memory Forensics: is also a subdiscipline of computer forensics. Its intention is to extract information from system memory or memory images.

EPT: "The extended page-table mechanism (EPT) is a feature that can be used to support the virtualization of physical memory. When EPT is in use, certain addresses that would normally be treated as physical addresses (and used to access memory) are instead treated as guest-physical addresses. Guest-physical addresses are translated by traversing a set of EPT paging structures to produce physical addresses that are used to access memory."[7]

DTB: The directory table base is used for Windows' page translation process and therefore needed by forensic tools to access the kernels address space. On a live system, the DTB is stored in the CR3 register.

PDB Files: (aka Program Database, from Microsoft) contain symbol or debug information for programs, which may be used by debuggers.[8] In Memory Forensics, e.g. Rekall uses the kernel's PDB to find the DTB.

Virtual Machine Introspection: Virtual machine introspection (VMI) is a technique for externally monitoring the runtime state of a system-level virtual machine. Monitors can be placed in another virtual machine, within the hypervisor, or within any other part of the virtualization architecture. For virtual machine introspection, the runtime state can be defined broadly to include processor registers, memory, disk, network, and any other hardware-level events.[9]

Direct Memory Access: For Direct Memory Access (DMA), a DMA controller is used, to allow direct memory access between storage hardware and memory, without using the CPU. Typically, this is used to fast transfer of large amounts of data without noticeable effects on the running operating system. An example for a valid application is taking disk/filesystem snapshots for backup purposes in large storage devices.

DMA Attack: During a DMA Attack, the DMA capabilities of the hardware are misused to store the system's memory onto disk, circumventing the access privileges, that are otherwise enforced by the operating system.

2.2 General Description of the Operating/Test Systems

2.2.1 Bromium

Bromium is a commercial Windows-based endpoint security software. It is intended for mid-size to big organizations, which can use the associated central management system, to define policies for the windows end user systems.

The Bromium client software then implements and enforces the corresponding rules on the end users' systems, increasing the organization's system security by using compartmentalization, which is implemented using Micro-Xen.

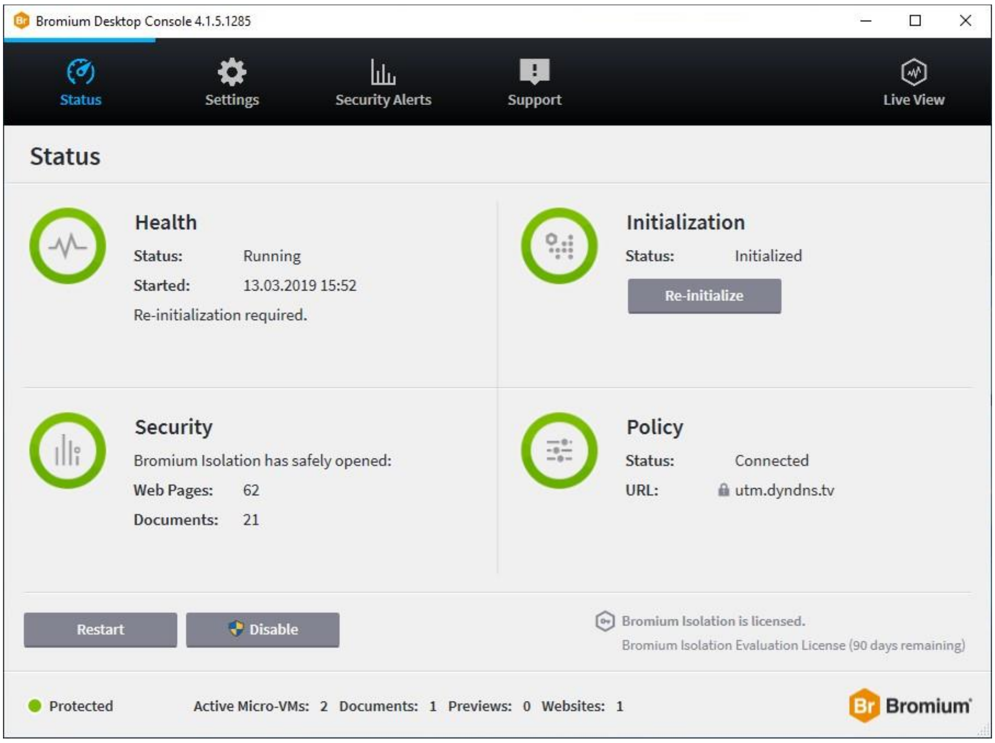
One of the main features of Bromium is, that each TAB of a WEB browser runs within its own compartment, another one, that downloaded documents are marked unsafe, per default, and consequently opened, read and modified within their own compartments.

For this thesis, a Windows 10 Pro 64-bit installation is used, version: 1809 (OS Build 17763) together with Bromium version 4.1.5.1285 .

In total, Bromium has 3 administrative GUIs, two on the client ("Bromium Desktop Console" and "Bromium Live View") and one on the admin server, where the policies for the clients are defined.

The Bromium Desktop Console runs on the Bromium client and provides the current status, settings, installed software versions, etc.

In the "Security Alerts" section, it also presents information about notable events, that happened, but only, as long as the client is not connected to the admin server. As soon as there is a connection to the admin server, the information silently vanishes from the client and becomes visible in the admin GUI on the server, instead.



Bromium Desktop Console 4.1.5.1285

Status

Health
Status: Running
Started: 13.03.2019 15:52
Re-initialization required.

Initialization
Status: Initialized

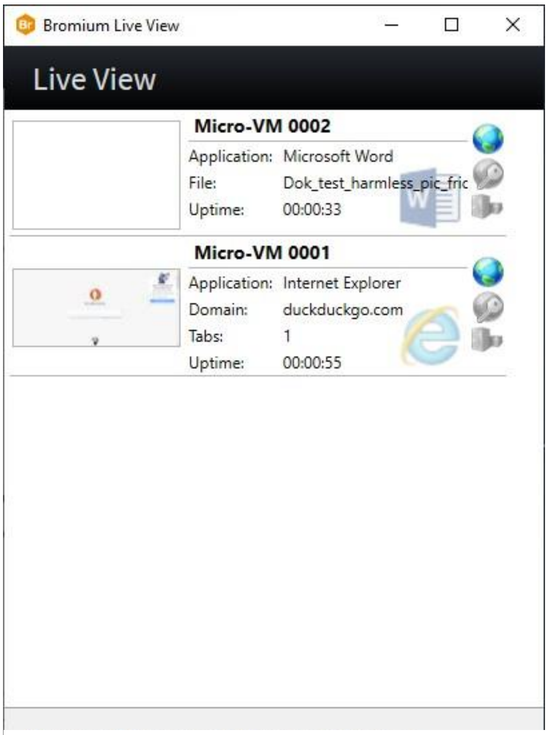
Security
Bromium Isolation has safely opened:
Web Pages: 62
Documents: 21

Policy
Status: Connected
URL: utm.dyndns.tv

Bromium Isolation is licensed.
Bromium Isolation Evaluation License (90 days remaining)

Protected Active Micro-VMs: 2 Documents: 1 Previews: 0 Websites: 1

Figure 1: Bromium Desktop Console



Bromium Live View

Live View

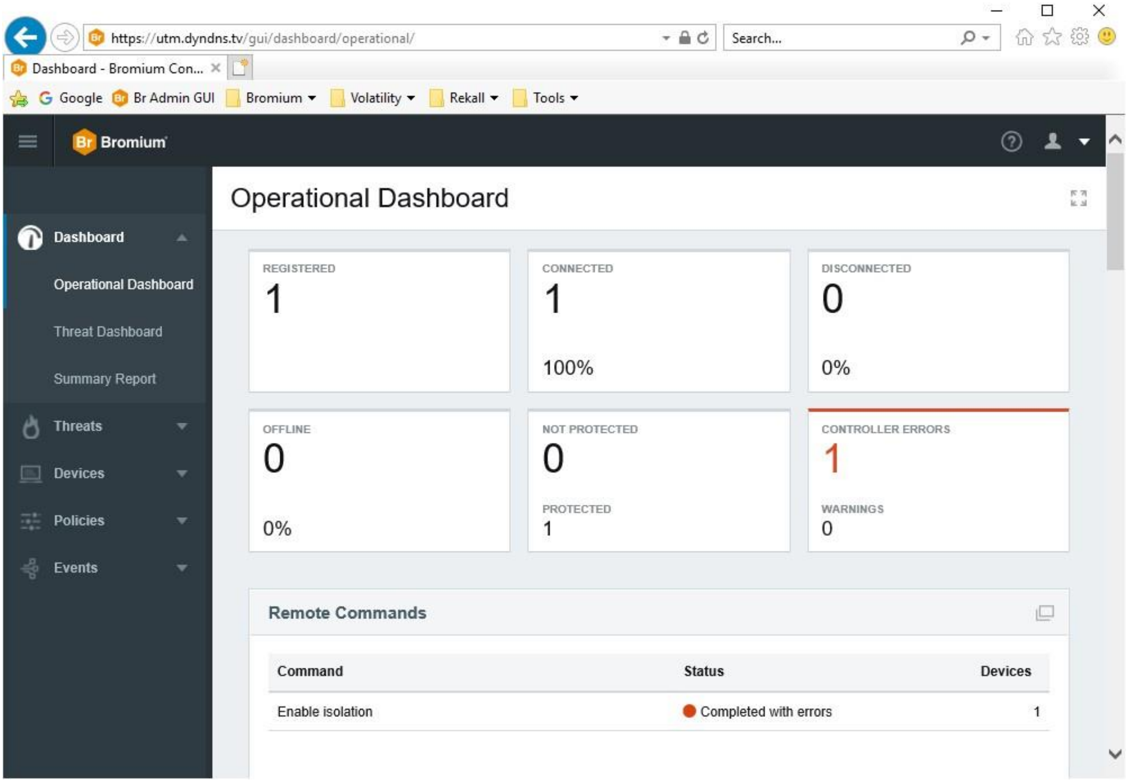
Micro-VM 0002
Application: Microsoft Word
File: Dok_test_harmless_pic_fric
Uptime: 00:00:33

Micro-VM 0001
Application: Internet Explorer
Domain: duckduckgo.com
Tabs: 1
Uptime: 00:00:55

Figure 2: Bromium Live View

Bromium Live View provides information about running Micro-VMs. These consist of running web browser tabs or documents, that previously have been marked unsecure and consequently are opened in separate Micro-WMs. All files, that are downloaded from the Web or received via email are marked unsecure, automatically; whereas files, that are created locally, are not marked as unsecure, by default.

The Bromium Admin GUI runs on the central Bromium admin server. It allows the administrator, to get an overview via the Dashboard, to view threats, that have been reported by the clients, to configure devices (e.g. group devices or run remote commands), configure policies (rules) for the clients, and to view events, which are informational messages, e.g. about threat isolation, system events, securing files, etc.



Operational Dashboard

REGISTERED 1

CONNECTED 1 100%

DISCONNECTED 0 0%

OFFLINE 0 0%

NOT PROTECTED 0 PROTECTED 1

CONTROLLER ERRORS 1 WARNINGS 0

Remote Commands

Command	Status	Devices
Enable isolation	● Completed with errors	1

Figure 3: Bromium Admin GUI

For this thesis, the admin server is provided as a web service by Bromium, running on their own systems, and a basic ruleset, also provided by Bromium, is used as policy.

Details about the Bromium test system installation, are available in appendix A.1 "Bromium".

2.2.2 Qubes

Qubes is a freely available linux based operating systems, which can be used by private, as well as, commercial users. It provides the possibility to run certain tasks, sealed within compartments, and consequently, the corresponding processes are granted access to the resources that are related to their compartment, but access to all other resources of the operation system, is denied.

Qubes is based on Fedora Linux and uses Xen for compartmentalization.

Besides Fedora (its own base operating system), it also provides predefined compartments for newer versions of Fedora, as well as Debian and Whonix. Additionally, there is a QubesBuilder, which simplifies creating individual template VMs and instructions on how to create templates with Ubuntu or for Windows7 applications [10]. Natively, Qubes' compartments are either called "domains" or "qubes".

The intended users of Qubes are end users and therefore, it is equipped with a graphical user interface (GUI; Xfce4) by default.

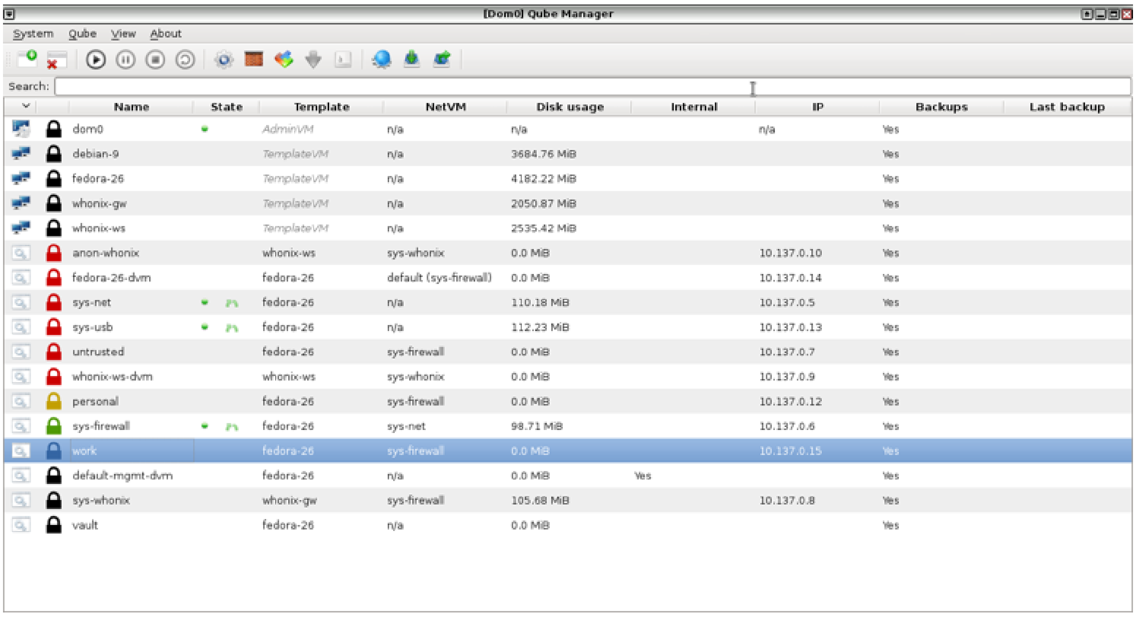
For this thesis, Qubes has been installed in version: 4.0, which is based on Fedora 25. After latest updates in March 2019, it is basically a Qubes 4.0.1 and runs the kernel version: 4.14.103-1.pvops.qubes.x86_64. Xen version is: 4.8.5.

Details about the Qubes test system installation, are listed in appendix A.2 "Qubes".

A typical Qubes 4.0 configuration consists of:

- dom0, which is the AdminVM (that holds the main operating system)
- some services (sys-net, sys-firewall, sys-usb), which allow access to networking and USB devices
- some template VMs (debian-9, fedora-26 and two VMs for whonix/Tor networking)
- some predefined VMs (untrusted, personal, work, vault, ...), which allow separating work related tasks from personal actions, or storing files in the vault VM, that has no network access or running potentially unsecure operations in the untrusted VM

All the corresponding settings can be configured with the "Qube Manager" software, that shows the configured VMs, as well as their templates, networking, Disk usage, etc.

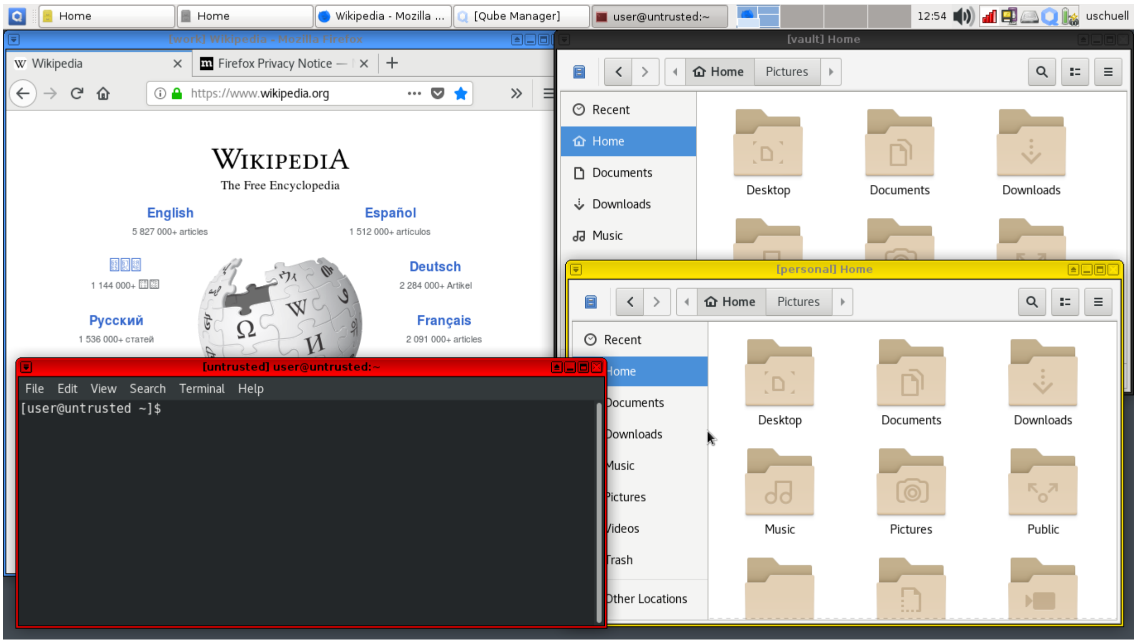


[Dom0] Qube Manager

Name	State	Template	NetVM	Disk usage	Internal	IP	Backups	Last backup
dom0	●	AdminVM	n/a	n/a		n/a	Yes	
debian-9		TemplateVM	n/a	3684.76 MiB			Yes	
fedora-26		TemplateVM	n/a	4182.22 MiB			Yes	
whonix-gw		TemplateVM	n/a	2050.87 MiB			Yes	
whonix-ws		TemplateVM	n/a	2535.42 MiB			Yes	
anon-whonix		whonix-ws	sys-whonix	0.0 MiB		10.137.0.10	Yes	
fedora-26-dvm		fedora-26	default (sys-firewall)	0.0 MiB		10.137.0.14	Yes	
sys-net	●	fedora-26	n/a	110.18 MiB		10.137.0.5	Yes	
sys-usb	●	fedora-26	n/a	112.23 MiB		10.137.0.13	Yes	
untrusted		fedora-26	sys-firewall	0.0 MiB		10.137.0.7	Yes	
whonix-ws-dvm		whonix-ws	sys-whonix	0.0 MiB		10.137.0.9	Yes	
personal		fedora-26	sys-firewall	0.0 MiB		10.137.0.12	Yes	
sys-firewall	●	fedora-26	sys-net	98.71 MiB		10.137.0.6	Yes	
work		fedora-26	sys-firewall	0.0 MiB		10.137.0.15	Yes	
default-mgmt-dvm		fedora-26	n/a	0.0 MiB	Yes		Yes	
sys-whonix		whonix-gw	sys-firewall	105.68 MiB		10.137.0.8	Yes	
vault		fedora-26	n/a	0.0 MiB			Yes	

Figure 4: Qube Manager

The Qube Manager also allows to configure, which applications can be started via GUI for a certain VM. To simplify identification (to indicate which window belongs to which VM), the windows are marked with different colors, which can also be configured with the Qube Manager.



```
[user@untrusted ~]$
```

Figure 5: Qubes GUI

Qubes also provides the possibility to run so-called disposable VMs, that can be used to inspect a potentially harmful document and that are simply removed, as soon as the intended task is finished.

Additionally, there is a function to create "trusted PDFs", which opens a PDF document in a disposable VM, extracts its content as graphics, that are then stored in a new secure PDF.

By default, there is no cut & paste or moving files between the different VMs; nevertheless, there are corresponding options. E.g. a cut & paste is possible using a system-wide clipboard and the keyboard combinations Shift-Ctrl-C/Shift-Ctrl-V and also moving or copying files between VMs, can be done with special commands.

In general, a configured VM shares the root-filesystem of its TemplateVM in read-only mode and only keeps its own private data. Consequently, software can be installed in a VM, but will be lost, as soon as the VM is restarted. To permanently install software for a VM, it needs to be installed in the corresponding Template VM and will become visible in the depending VMs, as soon as those are restarted.

2.2.3 Proxmox

Similar to Qubes, Proxmox is a freely available linux based operating system. Nevertheless, apart from sharing the linux base, there are certain differences:

Proxmox is based on Debian linux [11, p. 19] and uses Qemu/KVM as hypervisor [11, p. 117], if using Proxmox virtual machines. On the other hand, Proxmox also provides the possibility to configure containers and uses LXD as the corresponding container manager [11, p. 169]. It is clearly intended for server applications, like WEB or database servers. As a consequence, it does not provide a system GUI, per default.

Although it is highly discouraged to install and run a GUI like Xfce in Proxmox, it is still possible to do so. The main reason behind this recommendation is, that it might break the update capabilities. Nevertheless, as the environment should remain as stable as possible, for the duration of this master thesis, this is no real disadvantage and having the possibility, to run the Admin Web-Server and the corresponding Web-GUI on the same system, is a big advantage [12].

Details about the Proxmox test system installation, are available in appendix A.3 "Proxmox".

In contrary to Qubes, Proxmox does not provide any preconfigured VMs or containers during installation.

2.2.4 OS Overview / Comparison

Table 1: OS overview / comparison

	Bromium	Qubes	Proxmox
OS-basis	Windows Kernel: 1809 (Build:17763) OS Name: Windows 10 Professional	Linux: Fedora Kernel: 4.14.103-1.pvops.qubes or 4.14.123-1.pvops.qubes OS Name: Fedora-25	Linux: Debian Kernel: 4.15.18-10-pve OS Name: Debian GNU/Linux 9 (stretch)
intended usage	commercial	end-users (private & commercial)	server admins (private & commercial)
Hypervisor	Micro-Xen	Xen	Qemu/KVM (VMs) LXD (containers)
Sys. GUI	yes	yes	no
Adm. GUI	yes	yes	yes

2.2.5 SIFT Workstation

The SIFT workstation [13] is a forensic analysis system, provided by SANS [14]. It can be installed either on a physical system or alternatively in a virtual system, e.g. in Virtual Box. It provides many preinstalled tools for digital forensics. As it is already equipped with Volatility and Rekall, it has been installed as an additional test/analysis system in a Virtual Box, on Windows 10 Pro, in order to have another system for possible comparison of Rekall or Volatility outputs.

2.3 Tools for (Memory) Forensics

2.3.1 Volatility (free)

The first version of the Volatility Framework has been released in 2007.

"Up until that point, digital investigations had focused primarily on finding contraband within hard drive images. Volatility introduced people to the power of analyzing the runtime state of a system using the data found in volatile storage (RAM).

It also provided a cross-platform, modular, and extensible platform to encourage further work into this exciting area of research. Another major goal of the project was to encourage the collaboration, innovation, and accessibility to knowledge that had been common within the offensive software communities." [15]

Volatility provides many plugins, which allow to collect a lot of different information from a system's memory dump. Examples are process-listing or -tree, network connections, open files, memory associated with specific processes and many more.

By itself, Volatility is not able to detect Virtual Machines (VMs) and the related memory sections; nevertheless, in conjunction with additional plugins like Actaeon [16] it might be used for Virtual Machine Introspection (VMI).

2.3.2 Rekall (free)

"Rekall is an advanced forensic and incident response framework. While it began life purely as a memory forensic framework, it has now evolved into a complete platform. Rekall implements the most advanced analysis techniques in the field, while still being developed in the open, with a free and open source license. Many of the innovations implemented within Rekall have been published in peer reviewed papers.

Rekall provides an end-to-end solution to incident responders and forensic analysts. From state of the art acquisition tools, to the most advanced open source memory analysis framework." [17]

Originally, Rekall started as a fork of Volatility [18] in the year 2013. As a logical consequence, both tools are quite similar and share many common plugins. The biggest difference between those tools is the live memory forensic capability, that is available in Rekall, exclusively. And together with the live capabilities, Rekall also provides options to capture memory images, that are missing in Volatility, as well.

For VMI, the "vmscan" plugin [19] is the most important feature, as it scans for virtualization technologies, by using memory structures, that are typically present, when using Intel's VT-x with EPT. The resulting EPT values can later be used to access those sections in memory, that are related to VMs.

2.3.3 Magnet Axiom (commercial)

Magnet Axiom [20] is a commercial forensic software, which provides memory analysis, filesystem analysis, timelines et al.

According to Magnet, the manufacturer, Axiom provides its very own memory analysis tools, as well as an integrated Volatility to analyze memory images. These memory images should be loaded in one of the supported formats, which currently are: raw, Crash Dumps, Virtual Machine Saved State, Virtual Box Core Dumps (ELF) and other Volatility-Supported Formats. Guessing from the available plugins and data formats, it is unlikely, that Axiom is able to perform VMI on either Bromium, Qubes or Rekall.

On the university's test LAB, Axiom is installed in version 3.1.0.14142 .

2.3.4 Nuix (commercial)

Nuix investigate/engine [21] is a commercial forensic software. It provides options to analyze disk images, some databases or even virtual machine disk images.

NUIX has been looked at in version 7.6.6, which has been accessed through the test LAB of the university. Trying to add memory images as evidence to a case in NUIX, these images were all treated as files or disk images. Therefore, NUIX was not of any help for the purpose of this thesis.

Unfortunately, without access to the password protected NUIX support documents and online manuals, it could not be verified, whether NUIX is intended to perform memory forensics and if yes, for which OS versions.

2.3.5 X-Ways Forensics (commercial)

X-Ways Forensics [22] is a commercial forensic software, which is provided by the German company X-Ways AG and often used by German investigators.

As is listed on X-Ways Web-Presence, X-Ways provides "Very powerful main memory analysis for local RAM or memory dumps of Windows 2000, XP, Vista, 2003 Server, 2008 Server, Windows 7" [23]

So, X-Ways is neither capable of handling memory dumps captured from Windows 10 systems nor from linux systems. Therefore, it is not an option for testing together with Bromium, Qubes or Proxmox.

2.3.6 EnCase Forensic (commercial)

EnCase Forensic is a commercial forensic software, that is provided by Guidance Software, which meanwhile belongs to "opentext". According to their webpage, there are two different plugins available for memory forensics:

- a Volatility Reporting Plugin [24], which integrates Volatility Standalone vers. 2.4 with EnCase Forensic vers. 7.10
- MemoryAnalysis [25], is an EnScript, that has been tested with EnCase

Forensic vers. 7.06

For this thesis, EnCase version 8.04.00.129 has been provided via the test LAB of the university. On this test installation, the above listed plugins could not be started because of license issues. And apart from those plugins, the installed EnCase version did not come with any other memory forensics tools.

2.3.7 SIFT Workstation

The SIFT workstation [13] is a forensic analysis system, provided by SANS [14]. It can be installed either on a physical system or alternatively in a virtual system, e.g. in Virtual Box. It provides many preinstalled tools for digital forensics. As it is already equipped with Volatility and Rekall, it has been installed as an additional test/analysis system in a Virtual Box, on Windows 10 Pro, in order to have another system for possible comparison of Rekall or Volatility outputs.

2.4 Tools for Memory Image Creation

Tools for creation of memory images typically differ in relation to correctness, atomicity and integrity [26, p. 1]. Nevertheless, the aim of this thesis is to look at which data can be collected in memory on compartmentalized systems, and not, to focus on differences in memory image collection; therefore, mainly those tools, that are suggested by the Volatility and Rekall Communities, are considered. Consequently, winpmem [27],[28] or Rekall will be used for memory capturing on Windows and Rekall, /proc/kcore [29], linpmem or LiME [30],[28] will be used on Linux.

Typically, all these tools, need either root capabilities on Linux or Administrator access on windows, in order to get access to the physical memory of the system.

Quite a good overview of available options is listed at Forensicswiki [28].

2.4.1 winpmem

The tool winpmem is a memory capturing tool for windows, which originated from the Rekall project and further evolved into the AFF4 project [31]. The latest stable version is 3.2, which is available at GitHub [32].

On one hand, winpmem is capable of creating complete windows memory images in AFF4 format and on the other hand, it can be loaded as a driver, with option "-L", which then allows online memory access for Rekall via \\.\pmem or the "live" option.

If an aff4 Image has been created without compression, then the raw memory dump can be extracted, treating the aff4 file as a zip file and extracting "PhysicalMemory", otherwise, Rekall is needed with the "imagecopy" option, to extract a raw memory image.

2.4.2 Rekall

With Rekall, the plugins aff4acquire, ewfacquire, imagecopy or memdump may be used, to either dump the system's memory or memory associated with a specified process or to convert between formats. Here are some examples:

Creating a complete raw image:

```
C:\ rekal live
Live (Memory)> imagecopy output-image="mem_image.raw"
```

Collecting an aff4 memory dump:

```
C:\ rekal aff4acquire memdump.aff4
```

Extracting a raw dump from the aff4 format:

```
C:\ rekal -f memdump.aff4 imagecopy --output-image="memdump.raw"
```

Creating a complete dump in ewf format:

```
C:\ rekal ewfacquire memdump.e01
```

Converting an aff4 dump into ewf format:

```
C:\ rekal -f memdump.aff4 ewfacquire --destination="memdump.e01"
```

Additionally, if vmscan is able to detect EPT values for running guest VMs, rekall can be used, to create a raw image of the guest VM's memory:

```
C:\ rekal -f memdump.aff4 --ept=<EPT-value> imagecopy --output-
image="guest-image.raw"
```

2.4.3 pmem / linpmem

The tools pmem and linpmem are automatically installed together with the Rekall on linux and linpmem is also provided in standalone versions at rekall-forensics [33] or in newer versions at GitHub [32]. The collected data does not much differ from that, created with Rekall and aff4acquire. Its main advantage is, that the tool is quite small and therefore it is easier to transfer linpmem to a system, that does not already have any of these tools pre-installed.

As described in the Rekall blog "The pmem suite of memory acquisition tools" [34] provides two different methods of memory collection. Either /proc/kcore can be used by linpmem to collect memory, or if /proc/kcore is not enabled in the kernel, then the pmem kernel module can be used to get access to physical memory. Typically, memory images collected with pmem are in aff4-format.

2.4.4 LiME

"LiME (formerly DMD) is a Loadable Kernel Module (LKM), which allows the acquisition of volatile memory from Linux and Linux-based devices, such as those, powered by Android. The tool supports acquiring memory either to the file system of the device or over the network." [35]

A memory image can be collected by loading the lime.ko kernel module, together with options for an output file and the format, that should be used for data collection. Typically, Rekall and Volatility work fine with memory images, taken in lime-format, but seem to have issues, using images in raw format, written by lime.

Example:

```
# insmod lime.ko "path=/tmp/mem.lime format=lime"
```

2.4.5 DMA Attack

Another method, besides using software, to store memory dumps, is to present another system as a storage unit (e.g. Firewire) to the host and to use DMA (Direct Memory Access) to store the content of the memory into a file. This is called a DMA attack.

If available, on virtualized systems, this represents a potentially more reliable form of storing memory images, as it is not dependent on the access limitations of the running host operating system [36, p. 14] and its kernel, and typically does not cause a lot of changes to the memory (by running additional processes), that should be acquired. Nevertheless, in general, this form of memory acquisition is considered as not completely reliable [36, p. 15].

This method often has the limitation of using only the first 4 GB of memory [37] and all the used test systems have 8 GB of memory. On the other hand, there are GitHub projects like pcileech [38], which list hardware, that is natively able to access more than only the first 4 GB of memory.

Because of the hardware and software needs, the related costs and the time, needed to get it adapted, using DMA attacks extends the scope of this thesis and will not be tested.

And, at least for Qubes, there seem to be some limitations to use DMA attacks, at all, as is listed in the FAQs from Qubes [39].

2.4.6 VM Dumps - for Xen (Qubes) or QEMU/KVM (Proxmox)

On a system, running with Xen, it is possible to create a dump of a VM, using either "xl dump-core" or "virsh dump". Similarly, on KVM systems, "virsh dump" can be used to dump a VM.

Unfortunately, forensic software, as listed in chapter 2.3 "Tools for (Memory) Forensics" in not capable of analyzing these VM dumps. Instead, possibilities to get further information out of these dumps, might be using either GDB or the crash utility [40].

2.4.7 Summary

To have exactly the same memory image data in raw format, as well as in aff4 format, it makes sense, to first use Rekall and aff4acquire to create an aff-image, whenever possible. And later, in a second step, to use Rekall and "imagecopy" against this image, to extract a raw-image out of the aff4-image.

On the other hand, there are systems, like Qubes, which uses a Xen-Hypervisor, where dom0 does not have access to the complete memory sections of the VMs, as this access is possibly handled by the hypervisor, exclusively. There, options like memory dumps of the VMs, initiated by the hypervisor (with e.g. with "xl dump-core" in Xen) or DMA attacks seem to be the only options, to get access to all the VMs' memory sections, at all. Nevertheless, the VM dumps cannot be accessed with the examined forensic software and need other tools for analysis.

2.5 General Options in Relation to Memory Forensics on Compartmentalized Systems

2.5.1 From within a Compartment

Running live forensics or data acquisition from within a compartment is surely an option, that could be investigated; at least on the linux platforms, where access and installations within the compartments are possible (with Bromium, this is not an option, at all, as no additional software can be installed within the provided containers and no command line access to the hypervisor is available). On the other hand, apart from needing access to the compartment, itself, and possibly building a specific profile for the kernel, that is running within the compartment, this method is very similar to memory forensics on a non-compartmentalized system. Therefore, this option will not be considered, in this thesis.

2.5.2 From outside of the Compartments (Hypervisor/Container Engine)

Having the possibility to look at a container from the outside, has the advantage, that the investigator does not need to have any access to the container, itself. They only need to access the compartment's memory from the hypervisor or container engine to collect data from the compartment.

According to the current research, there are several projects for freely available tools, that might achieve this goal:

- VM discovery in Rekall (based on VT-x) [19]
- Actaeon [16]
- libVMI [41]
- KVM-VMI [42]
- drakvuf [43]

The biggest problem of VM introspection is caused by the fact, that kernel structures in host OS and guest OS are often different. Typically without knowing the exact details about the guest OS, it can be nearly impossible to get access to the internal structures of the guest OS [44, p. 2]. Additionally, it has to be noted, that apart from the "VM discovery" in Rekall and "drakvuf", all of the aforementioned tools are available on linux, only.

Actaeon is unfortunately too old, to work with the latest Volatility and therefore cannot be used in this work. Additionally, it does not support the processor, that is installed in the test systems. An installation of Actaeon has been tested on the SIFT workstation, but did not work successfully and was unable to produce any results. Details are listed in appendix B.8 "SIFT: Testing to install/run Actaeon".

Both LibVMI and LVM-VMI have a lot of prerequisites and especially need a patched KVM/QEMU in order to be able to inspect VMs. A patch for KVM/QEMU as well as other software versions needed, have not been found in versions, that are compatible with the current Qubes or Proxmox installations. Therefore, these libraries/extensions have not been tested with either Qubes or Proxmox.

Drakvuf consists of a collection of plugins, that do not replace or extend any functions, like pslist or files to VMs, rather, they are an addition, allowing the examiner, to get additional information, about system calls, or file- or socket-access [45].

3 Testing: Review of existing IT-forensic Capabilities on the test systems

3.1 Bromium

3.1.1 On the Client

As long as the Bromium client has a network connection to the admin server, no data about recognized incidents is visible on the client, itself. Nevertheless, in the event. that there is no connection available to the admin server, information about recognized incidents, is stored and visible locally. Later, as soon, as the connection to the server is possible, again, this information gets transferred to the admin server and afterwards vanishes from the client.

The information is shown in the so-called "Security Alert Inspector", which can be activated from the "Bromium Desktop Console". Within the inspector, the information is shown as listing of events, as well as in graphical format. Additionally, there is an option, to export the data into a CSV file.

Additionally, Bromium provides the possibility to generate debug logfiles, that can be generated on the client. Those are intended for Bromium support and consist of 7zip files, which can be looked at manually [46].

3.1.2 On the Admin Server

Whenever the Bromium client has access to the admin server via network, the information about an incident is stored on the server instead of the client. It also consists of the graphical view, that is available in the "Security Alert Inspector", as well as the option, to export the data in CSV format.

3.2 Qubes

Qubes does not provide any real forensic capabilities; nevertheless, there are several logfiles available, that provide useful information, about the different guest qubes, configured on the system. All of these logfiles reside within the directories /var/log/xen/console and /var/log/qubes in the qube dom0. For each qube, the "Qube Manager" provides access to a file guest-<qube>.log in the xen/console section and the files guid.<qube>.log and qrexec.<qube>.log in the qubes section.

For dom0, there is an additional file: /var/log/xen/console/hypervisor.log . Using the terminal in dom0, it is also possible, to access the aforementioned logfiles, as well as some additional logfiles, that are not visible through the "Qube Manager".

3.3 Proxmox

Similar to Qubes, Proxmox does not provide any real forensic capabilities on its own. Depending on the OS, that is used within the virtual machines or containers, there are the typical OS related logs, like e.g. /var/log/messages or the journal-log in newer Linux systems. Additionally, the Web-Admin-GUI also provides read access to /var/log/messages of the Proxmox host system. Apart from that, no additional logging is provided by Proxmox.

3.4 Summary

Overall, none of the test systems are pre-equipped with any real IT-forensic capabilities.

On the linux-based systems Qubes and Proxmox, system logs are the only available options to get information, about the processes in the Virtual Machines or containers.

With Bromium, there is a bit more information available, as it comes with its own event logging and additionally provides "incident reports" for notable events.

These are available in a graphical presentation and can be exported to CSV format, as well.

4 Testing: Memory Analysis with Forensic Tools

4.1 Bromium

The general issues with memory forensics on Bromium, are:

- Bromium is a commercial software and consequently, source code is not available, which makes it difficult to get details, about the internals of the µ-xen (aka uxen, pronounced as "micro-xen") compartments, that are used within Bromium.
- Bromium is running a customized version of Xen on a Windows host. (Xen support is implemented in Rekall, but only for the linux version, not for Windows).
- Neither Rekall nor Volatility supports the latest windows versions, if pre-compiled versions are used; consequently, Rekall and Volatility need to be installed from sources, if they should be able to analyze current windows versions and also with Bromium, running on a current Windows 10 installation.

4.1.1 Bromium - Testing with Rekall

Installing Rekall on a Bromium system works as on any other Windows system. Once installed, it can be run in live mode or against memory images and provides the typical data about the host system, like process lists, open files, network connections, etc.

With its *vmscan* plugin, Rekall is able to find extended page table entries for running VMs in memory or memory images.

a. using vmscan and resulting EPT (Extended Page Table) in Live Mode:

Running Rekall in live mode, the *vmscan* plugin is able to detect VMs and the corresponding EPTs. Nevertheless, when specifying the option --*ept* with live-Rekall, and running commands like *pstree* or *imageinfo*, the output is the same, as without the *--ept* option. Consequently, it has to be concluded, that live-Rekall might ignore the *--ept* option. Getting access to the data within the VMs in live mode, does not work out of the box on a windows installation.

b. using vmscan and resulting EPT against a Memory Image:

When using a memory image, Rekall obviously uses the *--ept* option (as subsequent runs of *version_scan* produce different lists for different EPTs.

But, using different EPTs (without specifying the correct profile, that is unknown at this time), results in failures. Depending on the order in which the plugins (*imageinfo*, *dtbscan*, *kdbgscan pstree*, *pslist*, ...) are used, they either fail with "`unable to find a valid profile`" or "`A DTB value was found but failed to verify`".

The first reason for this behavior is, that the guest virtual profile needs to be specified, as noted in the *vmscan* section of the Rekall Online documentation: "Once EPT values are found, you can use them to inspect virtual machines with any of the Rekall modules by using the *--ept* parameter and specifying the guest virtual machine profile." [47]

Even, when testing with the list of PDBs, returned by the *version_scan* command (and consequently being able to specify a profile), it first seemed to be impossible to access the data structures within the VMs, at all.

One thought was, that Xen uses a different way of addressing [48] called "direct mapping" and Xen support in Rekall is only implemented for Linux [49]. As a consequence, using the resulting PDB-Files together with the profile option, does not enable Rekall (installed on Windows) to get access to any additional data structures within the VM.

Using the EPT value together with the *find_dtb* plugin (which lists _EPROCESS addresses, while searching for a valid kernel DTB), it seemed at least possible to get an idea about the processes, running within a VM; nevertheless, it neither provided any detailed data nor allowed access to the VMs memory, using Rekall's plugins.

With the information (provided by Bromium), that the kernel used within a Micro-Xen VM, is the same kernel, as the kernel used by the host system, it was then possible, to provide the needed profile option and with that parameter given, some more information could be extracted from the memory dumps.

Still, plugins like *pstree*, *pslist*, *netscan* or *netstat*, produced errors or empty outputs. But others, like *psscan*, *psxview*, *filescan* and *services* started to produce some output, that looked at least partially useful.

What is remarkable is, that even with different EPT values given, the *psscan* outputs looked nearly identical (having the same PIDs and timestamps), but with some processes, being listed only in one or some of the outputs and not in all outputs.

On the other hand, process IDs, as well as timestamps, differ obviously from those, that are returned, when running the *ps*-related plugins without a given EPT. But, still, the output does not look trustworthy, as the memory dump, that returned the data, was taken on June 3rd, and the Micro-Xen VMs for the IE-tabs were started the very same day. Unexpectedly, the timestamps returned by the *psscan* plugin mostly show May 30th. Some of the different process related outputs are listed in appendix G.2.2 "Testing with Rekall against the Host System" and appendix G.2.3 "Testing with Rekall against VMs".

Also for *filescan*, used with different EPT values, the outputs look very similar, whereas when running *filescan* without giving an EPT, the returned list is a lot longer and has many references e.g. to C:\Users\, that are not in the *filescan*-outputs, when specifying an EPT (and consequently looking at the memory sections, that are used by the VMs).

Even with a valid profile, it only worked for a few memory images. For most other images, the Rekall plugins returned: "`A DTB value was found but failed to verify`". It is not clear right now, why some memory images can be used with an EPT value to return some data via *psscan*, whereas others do not return any data during analysis.

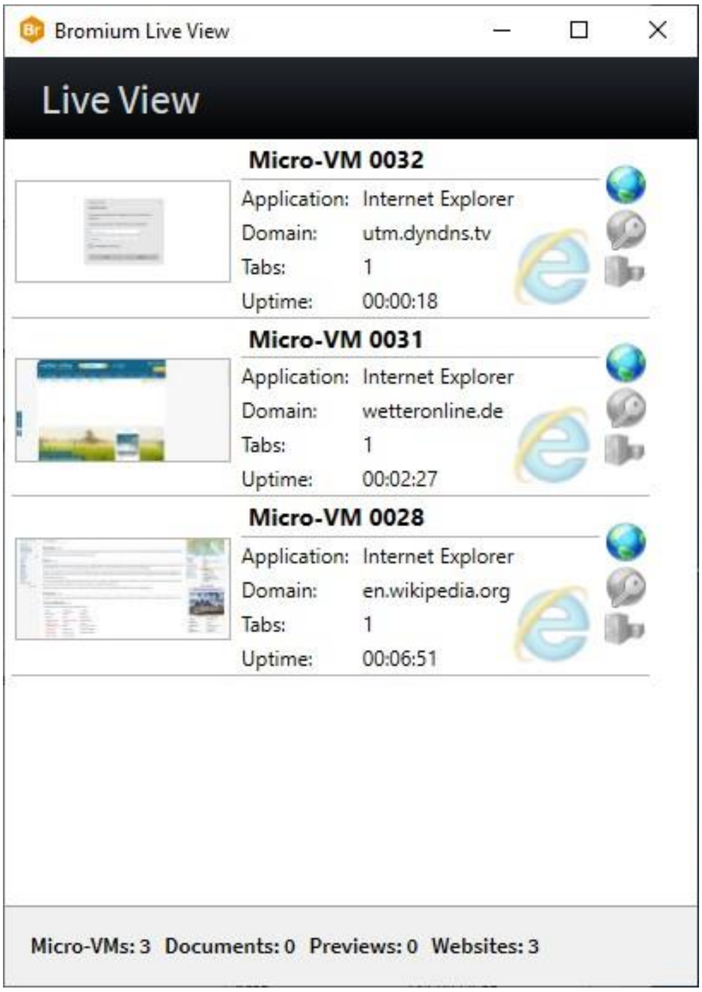
One idea was, that it might be related on whether Bromium is running or not when analyzing the memory image. But even after a reboot, and switching off Bromium, some memory images, like that from June 3rd can be partially analyzed and others, like that from August 17th, do not produce any useful output, at all.

Another idea is, that Bromium with its Micro-Xen implementation might somehow block the access to the memory, that is related to the active Micro-Xen VMs and consequently, access possible via the EPT values might point to remnants of older Micro-VMs, that are not really active anymore. This theory can be supported by the fact, that the *pstree* output (gathered from the VM host), typically shows two new processes (`Br-uxendm.exe` together with a `conhost.exe` child-process, which most likely handles the user input/output) for each newly started VM. On the other hand, if the system has not been booted recently, but did run for a longer time, there are also `Br-uxendm.exe` processes, without a sub-process. For an example output, visit the *pstree* listing on page: 39 . So, it is likely, that `Br-uxendm.exe` processes, that have no `conhost.exe` child-process, are possible leftovers from formerly running VMs.

4.1.2 Bromium - Testing with Volatility

Volatility, by itself, does not provide any functions for inspection of virtual machines. Therefore, installed on Windows, it allows to get information about processes, open network connections, open files, etc. of the Windows host system, but does not provide any information about the processes within the virtual machines, based on Micro-Xen, that are running in the Bromium environment.

Both, Rekall and Volatility show `Br-uxendm.exe` processes for each running InternetExplorer-TAB (IE-Tab) and as long as the IE-Tab is still visible, and has a Web-address-line, there is also a `conhost.exe` subprocess, for the input:



Bromium Live View

Live View

Micro-VM 0032

Application: Internet Explorer
Domain: utm.dyndns.tv
Tabs: 1
Uptime: 00:00:18

Micro-VM 0031

Application: Internet Explorer
Domain: wetteronline.de
Tabs: 1
Uptime: 00:02:27

Micro-VM 0028

Application: Internet Explorer
Domain: en.wikipedia.org
Tabs: 1
Uptime: 00:06:51

Micro-VMs: 3 Documents: 0 Previews: 0 Websites: 3

Figure 6: Bromium Live View with 3 IE-Tabs

```
(rekal) C:\DFIR\rekall>rekal -f memdump_190603_IE-tabs_3x.aff4 pstree
No handlers could be found for logger "rekall.plugins.tools"
_EPROCESS                                ppid  thd_count hnd_count       create_time
---------------------------------------- ------ --------- --------- ------------------------
 [...]
.. 0xe302c9e733c0 BrRemoteMgmtSv (3544)     672         9         - 2019-05-20 12:21:42Z
.. 0xe302c9e753c0 BrService.exe (3568)      672        22         - 2019-05-20 12:21:42Z
... 0xe302cbcc1080 BrHostSvr.exe (2580)    3568       105         - 2019-05-24 12:19:15Z
.... 0xe302cf668080 Br-uxendm.exe (7012)   2580        21         - 2019-06-03 11:34:56Z
..... 0xe302cc00c080 conhost.exe (1004)    7012         4         - 2019-06-03 11:34:56Z
.... 0xe302ccab5080 Br-uxendm.exe (8212)   2580        24         - 2019-06-03 11:34:34Z
..... 0xe302cb518080 conhost.exe (6288)    8212         4         - 2019-06-03 11:34:34Z
.... 0xe302cb840080 Br-uxendm.exe (8928)   2580        25         - 2019-06-03 11:30:12Z
..... 0xe302c8da7080 conhost.exe (10948)   8928         4         - 2019-06-03 11:30:12Z
... 0xe302cb16d540 Br-uxendm.exe (8312)    3568         0         - 2019-05-30 16:27:23Z
... 0xe302ca49d080 Br-uxendm.exe (8940)    3568         0         - 2019-05-20 12:22:05Z
... 0xe302caa4e080 Br-uxendm.exe (11808)   3568         0         - 2019-05-24 12:19:32Z
... 0xe302cc7130c0 Br-uxendm.exe (11968)   3568         0         - 2019-05-21 11:37:01Z
```

4.1.3 Bromium - Testing with commercial Forensic Software

To give commercial forensic software a chance, the aforementioned memory image from June 3rd, has been converted into raw format and EWF format with *imagecopy* and *ewfacquire* in Rekall.

Details about the different tools are listed in appendices G.2.4 to G.2.7 .

None of the forensic tools (Axiom, NUIX, Encase, X-Ways) was capable of performing memory forensics on the memory image(s) taken on Bromium, at all. Axiom looked most promising, but also failed to analyze even the Bromium host system, not to mention to recognize virtual machines and to perform VMI. The main reason, why Axiom did not even manage to inspect at least the Bromium host system, was the fact, that the provided Axiom version comes with Volatility and it's profiles and is equipped with a profile Win10x64_17134 which is older than the profile Win10x64_17763, which would be compatible with the Bromium system.

4.1.4 Bromium - Test Summary

Rekall (installed on Windows; on the Bromium System) is capable of finding EPTs (extended page table entries) for running Bromium VMs and at least sometimes, Rekall is also able to further analyze the VMs using the found EPT values.

Volatility is not able to recognize VMs on its own and the Actaeon plugin, that provides such capabilities is only available on linux and for an older version of Volatility, which on the other hand is not able to analyze current windows versions.

None of the commercial forensic software, tested together with this thesis, was capable of performing memory forensics on the Bromium system.

4.2 Qubes

The two main questions with memory forensics, especially on Qubes, are:

- Does Dom0 have access to the memory sections, that are used by the guest compartments? If yes, how/where to access the memory? And if not, how/where to get access to the guest's memory regions? Especially, when running the latest version of Qubes, it uses Intel's VT-d together with HVM (hardware based VMs), by default, and therefore prevents DMA attacks from being effective.
- Dom0 does not have network access and consequently, installations can only be done with the qubes-dom0-update command, that uses a proxy via the sys-firewall domain and only provides access to software packages, that have been provided by Qubes. Nevertheless, commands, like "git clone" or "pip install", that are typically used, during installation of Rekall or Volatility, are not available due to the indirect network access.

As has been tested on Qubes, neither is */proc/kcore* available in dom0 nor is */dev/mem* readable after the first MB. Consequently, there is no native access to collect memory dumps, apart from "*xl dump-core*", which creates a XEN memory dump.

Besides the Xen dump, it is also possible, to compile a kernel, which enables */proc/kcore*. To achieve this goal, it is necessary, to configure a Qubes Development VM to run qubes-builder and compile a new kernel in this environment. The new kernel package created in the Development VM then needs to be copied to dom0 and installed there. With */proc/kcore* available, it is then possible to collect memory images with linpmem (but only with an older version of linpmem, as latest versions fail with missing libraries for Fedora 25, for GLIBC versions 2.25 or 2.27 .

Additionally, a zip-archive of LiME can be placed in dom0 and compiled there. This allows to collect memory archives with the lime kernel module. (In the performed tests, this originally worked with the 4.14.123 kernel in dom0 and started to crash the system, after the installation of Rekall, finally succeeded, with all the changes, like installing a new kernel and lots of additional SW packages.)

Volatility can be placed in dom0 of Qubes via USB stick, either in binary version, or using the GitHub repository as a ZIP-archive. This archive then needs to be unzipped and Volatility can be started with "*python vol.py*" (even without performing an installation, using *setup.py*). To achieve its full functionality, the zip-archive version would also need several prerequisites to be installed; nevertheless, even without the prerequisites, the unpacked zip-archive allows creation of a profile, that is needed for memory analysis.

Getting Rekall into the Qubes environment is a lot more difficult. Installing Rekall into a standalone Fedora-25 VM, generally works, but then, there is no configuration for *pyinstaller* and the Debian-intended *dpkg-buildpackage*, which is documented for Rekall (in tools/installers/README) [18]. After several unsuccessful tests, manually creating a standalone binary of Rekall, has been abandoned.

Also, just downloading the GitHub repository as a zip-archive and unpacking it, is not working for Rekall. A more realistic approach (given the missing network access in dom0) has been documented by Michael Cohen for a presentation for DFRWS USA 2016 [50], [51]. He uses a standalone installation mode with "*pip download*" instead of "*pip install*", which (using a USB stick for transfer) is also possible for dom0 in Qubes. Later on, Rekall is then installed in dom0 using the downloaded pip-sources instead of the standard *pypi* index on the internet. In addition, the missing needed RPM packages (for libraries, Perl functionality, etc.) also have to be downloaded via the standalone Fedora-25 VM.

4.2.1 Qubes - Testing with Rekall

Generally, Rekall is able to run, but somehow, it is not able to work with the memory images or with the profile, generated on the Qubes system. Typically, running Rekall failed with an error message:

```
"A DTB value could be found but failed to verify"
```

4.2.2 Qubes - Testing with Volatility

Similarly, to Rekall, Volatility fails with errors like "`No valid DTB found`".

4.2.3 Qubes - Testing with commercial Forensic Software

During testing, it was not possible to create reliable memory images in Qubes.

The images, that have been acquired were all too small in size and could not even be analyzed with Rekall or Volatility on the Qubes system itself. (For details, refer to page 48.) Consequently, it made no sense, to test these images against commercial forensic software.

4.2.4 Qubes - Test Summary

Without reliable access to memory or memory images on Qubes, no memory forensic analysis was possible at all.

4.3 Proxmox

4.3.1 Proxmox - Testing with Rekall

Installing Rekall on the Proxmox host system works as expected. Creating memory images with Rekall is also possible. Plugins like *pslist* or *pstree* return the process list of the system, independent of whether live mode or a memory image is used. Unfortunately, *vmscan* is not able to find any VMs and their corresponding EPT values. Consequently, VMI did not work from this point on.

Just to make sure, the inability of *vmscan* to find any VMs, is not a problem of the local Rekall installation, on the Proxmox system, itself, a memory image, created with Rekall on Proxmox (and tested with the *pslist* plugin with the Rekall installation on the Proxmox system) has also been verified on the Rekall installation on the Bromium system, which was able to find VMs for Bromium memory images. But even there, the *vmscan* did not detect any VMs for Proxmox images. As a consequence, it has to be concluded, that something in Proxmox is special and prevents the *vmscan* plugin of Rekall from finding the VM structures in memory.

4.3.2 Proxmox - Testing with Volatility

Installing and running Volatility on Proxmox is possible and Volatility can be used to get data from the host system, using memory images, created either with *imagecopy* in Rekall or with LiME. Plugins like *linux_pslist* or *linux_netstat* produce the expected outputs.

By itself, Volatility is not able analyze data from VMs.

With Volatility, installed on a SIFT workstation, it has been tested to install the Actaeon plugin. Unfortunately, the latest version, available at GitHub has been created for Volatility version 2.1 and is not compatible with the latest Volatility version 2.6.1, which on the other hand is needed for current OS versions and kernels. And additionally, Actaeon in not able to work with the installed Haswell CPU.

4.3.3 Proxmox - Testing with commercial Forensic Software

As has been found out during testing with memory images from the Bromium system, none of the used commercial forensic programs provides any possibilities to detect running VMs in memory. Consequently, the images acquired on the Proxmox system, have not been tested against the commercial programs.

Axiom, that looked most promising for memory forensics, only supports Windows and OSX and no linux analysis.

4.3.4 Proxmox - Test Summary

On Proxmox, the host system can be analyzed with either Rekall or Volatiliy, whereas the tested commercial software is unable to analyze the gathered memory images. Analyzing the running VMs on Proxmox was not possible at all.

5 Memory Image Collection in Qubes

As has been described in chapter 4.2 "Qubes", collecting a memory image in Qubes (dom0) is not an easy task. Therefore, instead of creating a half-automated data collection for Qubes, this chapter will describe, how to possibly collect memory images in Qubes 4.0 and how to possibly install Rekall.

This will include the following topics:

- building a LiME kernel module (to collect memory images) in dom0
- building a kernel with "CONFIG_PROC_KCORE=y" for dom0,
  which enables */proc/kcore* for possible memory access
  (this procedure includes: setting up a qubes-builder environment, as well as installing and booting the modified kernel)
- providing linpmem (to collect aff4 memory images) in dom0
- using the aforementioned modifications to finally create memory images in aff4 or raw format

5.1 Installing qubes-builder, Creating/Installing a new Kernel in dom0

The most reliable procedure, to compile a modified kernel for dom0 in Qubes, is to use a qubes-builder environment, create the needed installation RPMs, there, and to install the new kernel from the RPM. Instructions for this procedure are available in the Qubes documentation "Building Qubes OS ISO" [52] and from instructions at GitHub (constantoverride, "recompiling Qubes kernel for AppVMs (or dom0 too? maybe)") [53].

The complete procedure, that has been tested, is listed in appendix F.1 "Installation of a modified Kernel in dom0".

In short, the procedure consists of the following steps:

- installing a standalone VM based on Fedora-28
- loading the qubes-builder environment from GitHub
- configuring qubes-builder to build a kernel for a Fedora-25 dom0, with

the needed change (CONFIG_PROC_KCORE=y)

- install/make dependencies
- compile the new kernel(s) and create the corresponding RPM(s), which is done by qubes-builder, using changeroot into a Fedora-25 environment
- copy the needed RPMs to dom0 ("kernel-414-4.14.123-1.pvops.qubes.x86_64.rpm" and "kernel-414-devel-4.14.123-1.pvops.qubes.x86_64.rpm ")
- install the RPMs in dom0 (using "*dnf install*"), which automatically includes building a new initramfs-image and also provides the needed entries in the grub2 configuration
- reboot dom0 to activate and test the new kernel

As is clearly visible, this procedure includes a lot of changes and especially a reboot of the complete Qubes environment; consequently, this procedure is totally unusable for ad-hoc memory forensics. It only makes sense, to test this procedure in memory forensics, in general preparation for future events or in case of analyzing reoccurring behavior.

5.2 Compiling the LiME Kernel Module

LiME is available from GitHub [35]. It just needs to be unzipped, and then using the make-command in the src sub-directory, a new module named *lime-<kernel-vers-details>.ko* is generated. This new module can then be used with *insmod*, to write kernel-images to disk. Details are listed in appendix D.1 "Qubes: LiME - creating a LiME module on Qubes (dom0)".

Originally, this procedure worked as expected. But, after the installation of additional software, that was needed to get Rekall installed on the Qubes system in dom0, using *insmod* with the lime.ko module reproducibly resulted in immediate system crashes.

5.3 Providing linpmem

All versions of linpmem need to have */proc/kcore* available on the system, in order to collect memory images. Consequently, this option only works after building a new kernel, as described in chapter 5.1 "Installing qubes-builder, Creating/Installing a new Kernel in dom0"

The tool linpmem is available as an executable at GitHub "Velocidex" [32] in version: 3.3.rc1 and in an older version (version 2.1.post4) at rekall-forensics [33].

To use the 2.1 version, it simply has to be copied to dom0 or provided/mounted via USB stick.

The 3.3 version additionally requires updating glibc to at least version 2.25 (glibc 2.24 is installed in Qubes 4.0), which is not available for Fedora25 and consequently also unavailable for Qubes 4.0. Nevertheless, under the (incorrect) title "Release 1.0 RC2" at Velocidex, there is also a version 3.0rc2 . This version first seemed to run on Qubes, without the need of installing any additional software packages. But later failed with errors about missing AFF4 storage. Hence, the linpmem version 2.1post4 is the latest release, that runs on Qubes, without additional major changes to the system, apart from the updated kernel for */proc/kcore* .

Details are listed in appendix E.2 "Qubes: linpmem Installation & Usage".

5.4 Creating Memory Images in Qubes

Originally, it looked, as if memory images in Qubes could be acquired with LiME or linpmem, but as can be seen later, these images were all too small, in comparison to the 8 GB of memory, that the Qubes system has in total.

Then, after the updates and installations needed to test getting Rekall to work, the LiME kernel module did not work any longer and when loaded with *insmod*, directly caused a system crash.

As a result, the *aff4acquire*, *ewfacquire* and *imagecopy* options of Rekall remained for taking memory images. Nevertheless, the generated memory images are also too small, to be trustworthy.

5.5 Summary

Unfortunately, all the research done and described earlier in this chapter, might be totally useless for memory forensics in Qubes, as Qubes uses a Xen-Hypervisor, which handles the memory access of the VMs and consequently dom0 might not have access to the VMs' memory sections. This is described by e.g. Researchgate [54].

The created memory images do not look trustworthy, as the test system has 8GB of RAM and all the created memory images are a lot smaller in raw format. The following list compares the sizes of memory images generated with *aff4acquire* and *imagecopy* in Rekall against the used memory as listed in "*xl lists*"; nevertheless; other memory acquisition tools did show similar size issues.

Table 2: Memory Image Sizes on Qubes

# of additional active VMs	active VMs (from "xl list")									Mem in MB				
	dom0	sys-net	sys-net-dm	sys-usb	sys-usb-dm	sys-firewall	sum system	personal	Forensics DEV	work	sum add	sum all	aff4	raw
0	4074	384	144	284	144	1220	6250				0	6250	2296	5458
1	3234	384	144	284	144	693	4883	2903			2903	7786	2118	5458
2	2890	384	144	284	144	613	4459	2572	705		3277	7736	1422	5458
3	2508	384	144	284	144	508	3972	2143	625	943	3711	7683	1348	5458

Other options, that remain, are:

- creating memory images via DMA access; but, this needs additional specialized hardware and guessing from their FAQs, Qubes works heavily on DMA protection [39]
- using Xen memory dumps of the VMs, created with "*xl dump-core*"; for this option to work, it needs to be examined, how the ELF dump file is

structured and how to possibly extract a raw dump from this file or how and with which tools do analyze a XEN VM dump

6 Test Results / Evaluation

The following criteria has been selected to compare memory forensic possibilities on the compartmentalized systems:

- possibility to use available precompiled binaries
- ability to generate reliable memory images
- ability to perform memory forensics on the host
- ability to detect VMs (e.g. with vmscan)
- ability to extract data from VMs
  (e.g. process list, network connections, open files)

Table 3: Evaluation of Forensic Tools

Evaluation	Bromium			Qubes	Proxmox		
possibility to use precompiled binaries	--			--	--		
possibility to generate reliable memory images	+			--	+		
Memory Forensics	host	detect VMs	analyze VMs		host	detect VMs	analyze VMs
Rekall	+	+	?		+	--	
Volatility	+	--			+	--	
Magnet Axiom	--				--		
Nuix	--				--		
X-Ways	--				--		
EnCase	--				--		

Comparing the tested compartmentalized operating systems, is it most difficult, to get any data for analysis out of **Qubes**. With the typical tools for memory image creation, no reliable memory images could be generated. With Qubes trying to prevent DMA attacks and to separate memory of different VMs, looking deeper into VM dumps ("*xl dump-core*") and their ELF format, might be more promising, to get more information out of running VMs.

With **Proxmox**, VMI basically failed, as none of the available tools was able to detect the running VMs. Here, looking deeper into the *vmscan* plugin of Rekall and possibly getting this plugin to detect VMs might be a chance for further investigation.

With **Bromium**, it was always possible to detect some running VMs with *vmscan* in Rekall and at least sometimes possible, to perform VMI and get a process list or list of open files out of a VM. Nevertheless, the timestamps found with *psscan* did not match the expected dates. So, it is possible, this data was extracted from VMs, that ran some time earlier and not from the VMs, that were expected to be seen. Future work in this area might focus on why it did not work reliably, why the timestamps of the processed did not match the expected date and how to get additional plugins like pslist or netstat to work, too.

Comparing the memory forensic tools, clearly, Rekall and especially its *vmscan* plugin did show the best options for memory forensics on compartmentalized systems with current OS versions. Volatility had a similar add-on with Actaeon, but this tool is currently too old to work with the latest Volatility version or with current CPUs and consequently with current windows or linux kernels.

Looking at the commercial tools, they sometimes simply implement Volatility (also not in the latest version) and overall, they are not able to detect VMs in a memory image.

7 Conclusion / Summary

Performing a live memory forensic analysis was not possible at all.

And even with some slight modifications on **Proxmox** and rather severe modifications in **Qubes**, it was not possible to perform any virtual machine introspection (VMI) on those systems.

On a **Bromium** system, it was always possible to detect running VMs with the *vmscan* plugin in Rekall and at least sometimes possible to even get further information out of the running VMs, with plugins like *psscan*, *psxview* or *filescan*. But the timestamps of the processes, that were found, did not match the expected date. Consequently, this data is also unreliable and needs further investigation.

Additional plugins or libraries like Actaeon, LibVMI, KVM-VMI, have been considered, but were not compatible with either Windows (for Bromium) or the available software on the Qubes and Proxmox systems or the installed processor. Therefore, these options have not been tested any further.

8 Outlook / future work

8.1 Bromium

On Bromium it is possible to inspect the host system with either Rekall or Volatility and to detect VMs with Rekall and the vmscan plugin.

It might be worth investigating, why using the resulting EPT values only sometimes allow to analyze the VMs and for other similar memory images failed, with the message "`a DTB value could be found but failed to verify`" or simply produced no output.

Another point is looking at the date and timestamps of the data, that could be extracted. Why do the time stamps not match with the expected dates?

Additionally, many plugins, like *pstree*, *pslist*, *netstat*, etc. did not produce any results, at all, when used against the VMs. Future work could concentrate on getting these plugins to work for VMs, if used together with the *--ept* option.

8.2 Qubes

On Qubes, it was not possible to create any reliable memory images with either Rekall (*acquire-commands), LiME or linpmem. Questions, that could be investigated in future work, are:

- Is it possible (and how) to reconfigure Qubes, to get access to the complete memory of the system?
- How can Xen dumps (generated with "*xl dump-core*") be analyzed?
- Is it possible to extract raw memory images from Xen dumps? (And is this an option to get memory images, that could be used with memory forensic tools, afterwards?)

8.3 Proxmox

On Proxmox, Rekall and Volatility could be used to analyze the host system. Nevertheless, the *vmscan* plugin of Rekall failed to find any VMs and to provide the corresponding EPT values.

Open questions, related to Proxmox and memory forensics, are:

- Is it possible to modify the *vmscan* plugin to find the VMs in Proxmox?
- Might it be necessary to patch QEMU/KVM to allow for VMI, similar as described in the prerequisites for LibVMI? [41]
- Are there chances to get LibVMI or KVM-VMI modified, in order to work with Proxmox? [41] [42]

9 Glossary/Abbreviations

DMA	Direct Memory Access
DTB	Directory Table Base
EPT	Extended Page Table
GUI	Graphical User Interface
HVM	Hardware Virtual Machine
KVM	Kernel-based Virtual Machines
LKM	Loadable Kernel Module
LXD	name of a container technology
OS	Operating System
PDB	Program Database
PV	Paravirtualization
VC	Virtual Client (guest)
VM	Virtual Machine
VMI	Virtual Machine Introspection

References

[1] Polizei Niedersachsen, “LKA Niedersachsen: Der Ratgeber Internetkriminalität der Polizei Niedersachsen: Bewerbungsmail mit Schadsoftware im Anhang.” [Online]. Available: https://www.polizei-praevention.de/aktuelles/bewerbungsmail-mit-schadsoftware-im-anhang.html?type=98. [Accessed: 11-Jan-2019]

[2] FBI, “Ransomware Prevention and Response for CISOs,” *Federal Bureau of Investigation*. [Online]. Available: https://www.fbi.gov/file-repository/ransomware-prevention-and-response-for-cisos.pdf/view. [Accessed: 01-Apr-2019]

[3] Unknown, “compartment - Wiktionary.” [Online]. Available: https://en.wiktionary.org/wiki/compartment. [Accessed: 22-Mar-2019]

[4] David Basin, Patrick Schaller, and Michael Schläpfer, “Security Principles,” in *Applied Information Security*, Springer, Berlin, Heidelberg, 2011, pp. 1–16 [Online]. Available: https://doi.org/10.1007/978-3-642-24474-2_1. [Accessed: 22-Mar-2019]

[5] Jarret B, “Hardware Virtual Machine (HVM) and Paravirtualization (PV),” *Linux.org*. [Online]. Available: https://www.linux.org/threads/hardware-virtual-machine-hvm-and-paravirtualization-pv.12475/. [Accessed: 23-May-2019]

[6] Unknown, “Computer forensics - ForensicsWiki.” [Online]. Available: https://forensicswiki.org/wiki/Computer_forensics. [Accessed: 22-Mar-2019]

[7] S. Karvandi, “Hypervisor From Scratch – Part 4: Address Translation Using Extended Page Table (EPT),” *Sina & Shahriar’s Blog*, 05-Oct-2018. [Online]. Available: https://rayanfam.com/topics/hypervisor-from-scratch-part-4/. [Accessed: 21-May-2019]

[8] Microsoft, “What’s inside a PDB File?,” *C++ Team Blog*, 08-Feb-2016. [Online]. Available: https://devblogs.microsoft.com/cppblog/whats-inside-a-pdb-file/. [Accessed: 21-May-2019]

[9] B. D. Payne, “Virtual Machine Introspection,” in *Encyclopedia of Cryptography and Security*, H. C. A. van Tilborg and S. Jajodia, Eds. Boston, MA: Springer US, 2011, pp. 1360–1362 [Online]. Available: https://doi.org/10.1007/978-1-4419-5906-5_647. [Accessed: 28-May-2019]

[10] Qubes, “Installing a Windows VM,” *Qubes OS*. [Online]. Available: https://www.qubes-os.org/doc/windows-vm/. [Accessed: 02-Apr-2019]

[11] “Proxmox VE Administration Guide.pdf.” [Online]. Available: https://pve.proxmox.com/pve-docs/pve-admin-guide.pdf. [Accessed: 20-Mar-2019]

[12] Proxmox Server Solutions Gmbh, “Developer Workstations with Proxmox VE and X11 - Proxmox VE.” [Online]. Available: https://pve.proxmox.com/wiki/Developer_Workstations_with_Proxmox_VE_and_X11. [Accessed: 20-Mar-2019]

[13] Unknown, “SIFT Workstation Download.” [Online]. Available: https://digital-forensics.sans.org/community/downloads. [Accessed: 21-Aug-2019]

[14] SANS, “Digital Forensics Training | Incident Response Training | SANS.” [Online]. Available: https://digital-forensics.sans.org/. [Accessed: 21-Aug-2019]

[15] Volatility Foundation, “volatilityfoundation | About,” *The Volatility Foundation - Open Source Memory Forensics*. [Online]. Available: https://www.volatilityfoundation.org/about. [Accessed: 01-Apr-2019]

[16] emdel, “Actaeon.” [Online]. Available: http://s3.eurecom.fr/tools/actaeon/. [Accessed: 16-Jul-2019]

[17] Rekall Forensics, “Rekall Forensics.” [Online]. Available: http://www.rekall-forensic.com/. [Accessed: 11-Jan-2019]

[18] Unknown, *Rekall Memory Forensic Framework - on GitHub*. Google, 2019 [Online]. Available: https://github.com/google/rekall. [Accessed: 02-Apr-2019]

[19] J. Sanchez, “VM discovery and introspection with Rekall.” [Online]. Available: http://blog.rekall-forensic.com/2014/10/vm-discovery-and-introspection-with.html. [Accessed: 01-Apr-2019]

[20] Magnet Forensics, “Magnet AXIOM - Digital Investigation Platform,” *Magnet Forensics*. [Online]. Available: https://www.magnetforensics.com/products/magnet-axiom/. [Accessed: 29-Jul-2019]

[21] NUIX, “Nuix Investigate | Nuix.” [Online]. Available: https://www.nuix.com/fact-sheets/nuix-investigate. [Accessed: 29-Jul-2019]

[22] X-Ways Software Technology AG, “X-Ways Forensics: Integrated Computer Forensics Software.” [Online]. Available: http://www.x-ways.net/forensics/index-m.html. [Accessed: 29-Jul-2019]

[23] X-Ways Software Technology AG, “X-Ways Forensics: Integrated Computer Forensics Software.” [Online]. Available: http://www.x-ways.net/forensics/index-m.html. [Accessed: 23-Aug-2019]

[24] GUIDANCE Software, “Volatility Reporting Plugin.” [Online]. Available: https://www.guidancesoftware.com/app/Volatility-Reporting-Plugin. [Accessed: 04-Aug-2019]

[25] GUIDANCE Software, “MemoryAnalysis.” [Online]. Available: https://www.guidancesoftware.com/app/MemoryAnalysis. [Accessed: 04-Aug-2019]

[26] M. Gruhn and F. C. Freiling, “Evaluating atomicity, and integrity of correct memory acquisition methods,” *Digit. Investig.*, vol. 16, pp. S1–S10, Mar. 2016 [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S1742287616000049. [Accessed: 01-Apr-2019]

[27] Rekall Forensics, “Rekall Tutorial - Rekall Forensics.” [Online]. Available: http://www.rekall-forensic.com/documentation-1/rekall-documentation/tutorial. [Accessed: 01-Apr-2019]

[28] Unknown, “Tools:Memory Imaging - ForensicsWiki.” [Online]. Available: http://www.forensicswiki.org/wiki/Tools:Memory_Imaging. [Accessed: 08-Mar-2019]

[29] Unknown, “Rekall Memory Forensic Framework.” [Online]. Available: http://web.rekall-innovations.com/docs/Manual/tutorial.html. [Accessed: 05-Jun-2019]

[30] Volatility Foundation, “volatilityfoundation | FAQ,” *The Volatility Foundation - Open Source Memory Forensics*. [Online]. Available: https://www.volatilityfoundation.org/faq. [Accessed: 01-Apr-2019]

[31] Unknown, “Rekall Forensics blog.” [Online]. Available: http://blog.rekall-forensic.com/. [Accessed: 01-Apr-2019]

[32] Unknown, *An AFF4 C++ implementation. . Contribute to Velocidex/c-aff4 development by creating an account on GitHub*. Velocidex, 2019 [Online]. Available: https://github.com/Velocidex/c-aff4. [Accessed: 20-Jun-2019]

[33] Rekall Forensics, “releases.rekall-forensic.com.” [Online]. Available: http://releases.rekall-forensic.com/. [Accessed: 07-Jun-2019]

[34] Unknown, “The pmem suite of memory acquisition tools.” [Online]. Available: http://blog.rekall-forensic.com/2016/05/the-pmem-suite-of-memory-acquisition.html. [Accessed: 23-Jun-2019]

[35] Unknown, *LiME*. 504ENSICS Labs, 2019 [Online]. Available: https://github.com/504ensicsLabs/LiME. [Accessed: 20-Jun-2019]

[36] “Graziano et al. - 2013 - Hypervisor Memory Forensics.pdf.” [Online]. Available: http://www.s3.eurecom.fr/docs/raid13_graziano.pdf. [Accessed: 17-Jul-2019]

[37] iMHLv2, “Github - Volatility - Firewire Address Space,” 25-May-2014. [Online]. Available: https://github.com/volatilityfoundation/volatility/wiki/Firewire-Address-Space. [Accessed: 08-Apr-2019]

[38] U. Frisk, “Direct Memory Access (DMA) Attack Software.,” 02-Aug-2019. [Online]. Available: https://github.com/ufrisk/pcileech. [Accessed: 04-Aug-2019]

[39] Unknown, “Frequently Asked Questions,” *Qubes OS*. [Online]. Available: https://www.qubes-os.org/faq/. [Accessed: 14-Aug-2019]

[40] S. Seyfried, “Configuring and Analyzing Kernel Crash Dumps,” p. 6.

[41] Unknown, *The official home of the LibVMI project is at https://github.com/libvmi/libvmi.: libvmi/libvmi*. libvmi, 2019 [Online]. Available: https://github.com/libvmi/libvmi. [Accessed: 27-May-2019]

[42] Unknown, *KVM-based Virtual Machine Introspection. Contribute to KVM-VMI/kvm-vmi development by creating an account on GitHub*. KVM Virtual Machine Introspection, 2019 [Online]. Available: https://github.com/KVM-VMI/kvm-vmi. [Accessed: 27-May-2019]

[43] Unknown, “DRAKVUF^{TM} Black-box Binary Analysis System.” [Online]. Available: https://drakvuf.com/. [Accessed: 27-May-2019]

[44] J. Xiao, L. Lu, H. Wang, and X. Zhu, “HyperLink: Virtual Machine Introspection and Memory Forensic Analysis without Kernel Source Code,” in *2016 IEEE International Conference on Autonomic Computing (ICAC)*, Wuerzburg, Germany, 2016, pp. 127–136 [Online]. Available: http://ieeexplore.ieee.org/document/7573124/. [Accessed: 27-May-2019]

[45] T. K. Lengyel, “Drakvuf GitHUB Wiki Plugins.” [Online]. Available: https://github.com/tklengyel/drakvuf/wiki/DRAKVUF-Plugin-Documentation. [Accessed: 13-Aug-2019]

[46] Bromium, “Gathering logs for Bromium products.” [Online]. Available: https://support.bromium.com/s/article/How-to-enable-verbose-debug-level-log-bundles. [Accessed: 08-Aug-2019]

[47] Unknown, “Rekall Memory Forensic Framework - vmscan.” [Online]. Available: http://web.rekall-innovations.com/docs/Manual/Plugins/General/VmScan.html. [Accessed: 04-Jul-2019]

[48] J. Sanchez, “XEN ParaVirtualization support in Rekall.” [Online]. Available: http://blog.rekall-forensic.com/2015/08/xen-paravirtualization-support-in-rekall.html. [Accessed: 22-May-2019]

[49] Unknown, “Announcing Rekall Release 1.3.1 (Dammastock).” [Online]. Available: http://blog.rekall-forensic.com/2015/04/announcing-rekall-release-131-dammastock.html. [Accessed: 23-May-2019]

[50] Michael Cohen, “Using GRR and Rekall for Scalable Memory Analysis (part 1),” *dfrws*, 25-May-2016. [Online]. Available: https://www.dfrws.org/conferences/dfrws-usa-2016/sessions/using-grr-and-rekall-scalable-memory-analysis-part-1. [Accessed: 25-Jun-2019]

[51] Michael Cohen, “pres_using_grr_and_rekall_for_scalable_memory_analysis.pdf,” *dfrws*. [Online]. Available: https://www.dfrws.org/file/839. [Accessed: 25-Jun-2019]

[52] Unknown, “Qubes ISO Building,” *Qubes OS*. [Online]. Available: https://www.qubes-os.org/doc/qubes-iso-building/. [Accessed: 20-Jun-2019]

[53] constantoverride, “recompiling Qubes kernel for AppVMs (or dom0 too? maybe),” *Gist*. [Online]. Available: https://gist.github.com/constantoverride/825717e0136f804aa6ebf66293234b57. [Accessed: 20-Jun-2019]

[54] M. A. Bamiah, “Figure 4. Xen architecture. For each DomU VM, CPU and memory access...,” *ResearchGate*. [Online]. Available: https://www.researchgate.net/figure/Xen-architecture-For-each-DomU-VM-CPU-and-memory-access-operations-are-handled-directly_fig4_241195554. [Accessed: 28-Jul-2019]

[55] M. Cary, “Installing Volatility on Windows,” *DFIR on the Mountain*. 29-Oct-2018 [Online]. Available: https://dfironthemountain.wordpress.com/2018/10/29/installing-volatility-on-windows/. [Accessed: 21-Feb-2019]

List of Figures

List of Tables

Declaration of Academic Integrity

Hiermit erkläre ich, dass ich die hier vorliegende Arbeit selbstständig, ohne unerlaubte fremde Hilfe und nur unter Verwendung der in der Arbeit aufgeführten Hilfsmittel angefertigt habe.

Ort, Datum (Unterschrift)

Theses

- **Live memory forensics with VMI does not work** on either of the systems (Bromium, Qubes or Rekall).
- **Bromium** systems **can be analyzed with Rekall** (at least sometimes and using memory images). Then, plugins like "psscan" and "filescan" are able to extract information from the running Micro-Xen VMs on a Bromium system.
- With **Qubes**, all the tested tools for memory dumping, **did not produce reliable memory dumps** and therefore, using memory forensics tools does not provide any insight into running VMs.
- On **Qubes**, testing with VM-dumps (instead of memory dumps), created e.g. with "*xl dump-core*", and finding out, which other tools, like GDB or the crash utility, might work on those dumps or how to extract a raw memory dump out of the VM dump, might provide additional VM information.
- On **Proxmox**, forensic tools can be used to inspect the host system and generate memory images. Nevertheless, none of the tested tools was able to detect and inspect VMs.
- Overall, **memory forensics is very "fragile"**. Even slight changes, or automatic updates in windows (that can be switched off; but get automatically reenabled by Microsoft after 3 months) might make the difference between a plugin, that works successfully and one, that does not.

Appendix

A Installation Instructions for the Test Systems

A.1 Bromium

1. Load SW from: https://bromiuminc-my.sharepoint.com/:f:/g/personal/XXXXX
   (the download Share provided by Bromium is already expired)
2. Download: x64 version
3. Directory: C:\Program Files\Bromium\vSentry
4. URL for the controller: https://utm.dyndns.tv
5. => install => question about allowing modifications => confirm
6. check for requirements => one Warning, that Office is not installed
   => „Continue“ (use TAB; mouse did not work at this point)
7. open an Admin CMD and run:
   - cd %brs%
   - BrManage config set --name=BMS.IgnoreInvalidServerCertificate --value=1
8. in IE: „Add-On „Bromium Isolation Plugin_4_1_5_1285“ from 'Bromium, Inc.'“ => „activate"

A.2 Qubes

1. Load ISO from: https://www.qubes-os.org/downloads/ => Version 4.0
2. Download: Rufus (Version 3.4) (see: https://rufus.akeo.ie/)
3. use Rufus to DD the ISO to a USB stick
4. go to BIOS and enable VT-x and VT-d (within the "Advanced" section)
5. boot from the Qubes-4.0-USB-Stick
6. add/select German keyboard
7. select target disk (regain disk space to get rid of former installations on the disk)
8. add a user: "uschuell" with PW "forensics"
9. start installation (de-select disk encryption)
10. after the reboot => select defaults for configuration
11. perform steps from https://www.qubes-os.org/doc/installation-guide/
    => "Qubes 4.0 Warning" section
    - Steps for dom0 updates:
      1. Open the Qubes Menu by clicking on the “**Q**” icon in the top-left corner of the screen.
      2. Select **Terminal Emulator**.
      3. In the window that opens, enter this command:
         - **sudo vi /etc/yum.repos.d/qubes-dom0.repo**
      4. This opens the vi editor. Change all four instances of **http** to **https**.
      5. Save and exit
      6. Check for updates normally. (**sudo qubes-dom0-update**)
    - Steps for Fedora 26 TemplateVM updates:
      1. Open the Qubes Menu by clicking on the “**Q**” icon in the top-left corner of the screen.
      2. Select **Template: fedora-26**, then **fedora-26: Termina**l.
      3. In the window that opens, enter the command for your version:
         - ~~[Qubes 3.2] sudo gedit /etc/yum.repos.d/qubes-r3.repo~~
         - [Qubes 4.0] **sudo gedit /etc/yum.repos.d/qubes-r4.repo** (possibly use vi, here, too)

   4. This opens the gedit text editor in a window. Change all four instances of **http** to **https**.
   5. Click the "**Save**" button in the top-right corner of the window.
   6. Close the window.
   7. Check for updates normally. (connect to WiFi; then run "**sudo dnf upgrade**")
   8. Shut down the TemplateVM.

12. update to the latest versions:
    - launch the "Updater" (yellow star icon from the top right menu)
      => install all available updates (last done on March 26th 2019)
    - alternatively run the "sudo qubes-dom0-update" and "sudo dnf upgrade" from step 11, again

A.3 Proxmox

Basic Installation:

1. Load ISO from: https://www.proxmox.com/de/downloads => Version 5.3
2. Download: Rufus (Version 3.4) (see: https://rufus.akeo.ie/)
3. use Rufus to DD the ISO to a USB stick
4. go to BIOS and enable VT-x and VT-d (within the "Advanced" section; „Device Configuration")
5. also in BIOS (within the "Advanced" section; „Boot Options") enable
   1. „**UEFI Native (without CMS)**", if all OSes are able to boot from UEFI
   2. „**UEFI Hybrid (with CMS)**", if some OSes boot from UEFI (as Proxmox does) and others require legacy mode / MBR
6. boot from the Proxmox-5.3-USB-Stick
7. select „Install Proxmox VE"
8. agree to EULA
9. select target disk (/dev/sdb; KingDian; 238 GB) => Next
10. set Country to „Germany" (Keyboard Layout, too) => Next
11. add a PW "linux" (twice) and email address
12. configure network:
    1. Mgmt-Interface: enp0s25
    2. FQDN: us-pve.hpd.pdn
    3. IP-Address: 10.0.10.50
    4. Netmask: 255.255.128.0
    5. GW: 10.0.2.1
    6. DNS: 10.0.0.3
    7. => Install
13. reboot and select „M.2 SSD drive (UEFI)"
14. logon as root/linux
15. Connect to the Admin Web Interface via: https://10.0.10.50:8006 or: https://us-pve.hpd.pdn:8006
16. Get rid of the "no valid subscription" message in Proxmox:
    ```
    sed -i.bak "s/data.status !== 'Active'/false/g" \
    /usr/share/javascript/proxmox-widget-toolkit/proxmoxlib.js && \
    systemctl restart pveproxy.service
    ```
    Source:
    https://www.sysorchestra.com/remove-proxmox-5-3-no-valid-subscription-message/

Additional Installation Steps 1-4:

1. Configure Updates for the No-Subscription Installation:

```
# cat /etc/apt/sources.list.d/pve-enterprise.list
      deb https://enterprise.proxmox.com/debian/pve stretch pve-enterprise
```

```
# cd /etc/apt/sources.list.d/
# cp pve-enterprise.list pve-no-subscription.list
```

=> comment entry in pve-enterprise.list

=> change entry in pve-no-subscription.list from:

```
https://enterprise.proxmox.com/debian/pve
```

to:

```
http://download.proxmox.com/debian/pve stretch pve-no-subscription
```

Make sure IPv4 is used for Updates from the internet (otherwise connection to Debian will fail on my network):

```
# echo 'Acquire::ForeceIPv4 „true‟;' > /etc/apt/apt.conf.d/42ipv4
```

```
# apt update && apt disc-upgrade
```

Activate the new kernel:

```
# reboot
```

2. Additional User and X11:

```
# vi /etc/defaults/useradd
```

=> set SHELL to /bin/bash

```
# vi /etc/login.defs
```

=> add: CREATE_HOME yes

```
# adduser -m -s /bin/bash uschuell
# passwd uschuell
```

=> set to „forensics“

```
# echo "alias ll='ls -l'" >> /root/.bashrc
# . /root/.bashrc
```

Uncomment the ll-alias in /home/uschuell/.bashrc

Install the GUI, browser, login-manager and terminal:

```
# apt install xfce4 chromium lightedm xfce4-terminal
```

```
# reboot
```

3. Proxmox Admin Configuration:

Logon to the GUI as uschuell/forensics

Start the Web-Browser (Chromium), connect to https://localhost:8006 and logon as root/linux

Also, open a terminal:

```
# su
# apt install sudo
# vi /etc/sudoers
```
=> add:
```
    uschuell    ALL=(ALL:ALL)    ALL
```

Create Admin Users for the Web-GUI:
```
# pveum groupadd admin -comment „System Administrators‟
# pveum aclmod / -group admin -role Administrator
```

Back in the Web-GUI, add the user:
=> go to „Datacenter“ , „Permissions“, „Users“ => „Add
=> add:
User name: uschuell
Group: admin
=> leave other defaults or add Firstname or email, etc.
=> press „Add“

Finally, deactivate root logons within the terminal
```
# passwd -l root
```

4. Configure PCI Passthrough:
(for INTEL CPUs; allows giving hardware components to VMs and therefore making them unavailable on the host; also CONFIGURED at my host)

=> needs to be enabled in BIOS (check for either VTD or AMD-iommu)

```
# vi /etc/default/grub
```
=> change the line for GRUB_CMDLINE_LINUX_DEFAULT from:
```
      GRUB_CMDLINE_LINUX_DEFAULT=„quiet"
```
to:
```
      GRUB_CMDLINE_LINUX_DEFAULT=„intel_iommu=on"
```

```
# update-grub
```

```
# vi /etc/mopdules
```
=> add:
```
      vfio
      vfio_iommu_type1
      vfio_pci
      vfio_virqfd
```

=> Reboot from GUI or Web-GUI

=> Verify the changed configuration:
```
# sudo -i
# dmesg | grep -e DMAR -e IOMMU
```

=> watch out for:
Intel(R) Virtualization Technology for Direct I/O

```
# lsmod | grep vfio
```

=> should return something similar to:
```
      vfio_pci            45056   0
      vfio_virqfd         16384   1 vfio_pci
      irqbypass           16384   2 vfio_pci, kvm
      vfio_iommu_type1    24567   0
      vfio_               28672   1 vfio_iommu_type1,vfio_pci
```

A.4 Forensic Workstation (SIFT)

Windows-Installation:

1. remove the 2nd disk (M.2 SATA)
2. boot from Windows 10 installation USB stick (Vers. 1809; 17763)
    - select: Advanced / User Defined installation
    - start
4. After the reboot:
    - Region: Germany
    - Keyboard Layout: German
    - select WLAN network
5. Configuration selections:
    - select: personal usage (not organisation)
    - enter Microsoft User ID
    - create/enter PIN (forensics)
    - enter phone number for mobile connection => select: do it later
    - select: save new files only on this PC only (not in the cloud)
    - want to use Cortana (voice assistant) => deny
    - share activity between devices? => deny
    - voice input? => do not use => continue
    - send position to Microsoft => no => continue
    - search device / position => yes
    - diagnosis data? => simple => continue
    - freehand and input improvement? => no => continue
    - tips (using diagnosis data)? => no => continue
    - apps advertising ID? => no => continue
6. After the installation
    1. Make Internet Explorer the default (instead of Edge)
        - settings
        - apps&features
        - standard-apps
        - Webbrowser: Internet Explorer
        2. start Internet Explorer and "pin to task bar"
    3. change computer language to English with German keyboard in settings

SIFT Installation as VM in VirtualBox

5. install virtual box from: https://www.virtualbox.org (version 6.0)
6. download SIFT appliance in .ova format from: https://digital-forensics.sans.org/download-sift/3.0
7. Start VirtualBox and import:
    1. File
    2. Import Appliance
    3. select the downloaded .ova file
8. start the SIFT Workstation VM
    1. open: settings
        - select: Language support
            - add language: German
        - select: Keyboard ; then "Text Entry"
            - add German
            - move German to position 1 of the list
    2. run offered automatic updates
    3. reboot the SIFT workstation
9. In Virtual Box within "Settings", "Storage" add a SATA optical drive pointing to
    1. C:\Program Files\Oracle\VirtualBox\VBoxGuestAddition.iso
    2. accept the auto install of the VM guest additions

    3. shutdown the SIFT Workstation
10. In Virtual Box within "Settings", "Shared Folders" add:
    1. Name: SHARED
    2. Path: C:\Users\utesc\VirtualBox VMs\SHARED
    3. Access: Full
    4. Auto Mount: Yes
    5. At: /SHARED
11. start the Virtual Box
    1. find the shared folder: "df" => /SHARED
    2. make the SHARED folder accessible for the User: sansforensics; in a terminal
        - sudo -i
        - usermod -a -G vboxsf sansforensics
        - reboot
        - cd /SHARED; ls
        - cd ~/Desktop
        - ln -s /SHARED SHARED
    3. on the PC put a PIC(ture) into the SHARED folder
    4. make sure, the sansforensic user can access the PIC within the VM

B Rekall and Volatility Installations

B.1 Bromium Client: Volatility Installation

In short, the installation is based on:
https://dfironthemountain.wordpress.com/2018/10/29/installing-volatility-on-windows/

=> this nearly worked out of the box

=> additionally, the copy_data_block_order branch of volatility had to be incorporated;

see: https://github.com/volatilityfoundation/volatility/issues/583

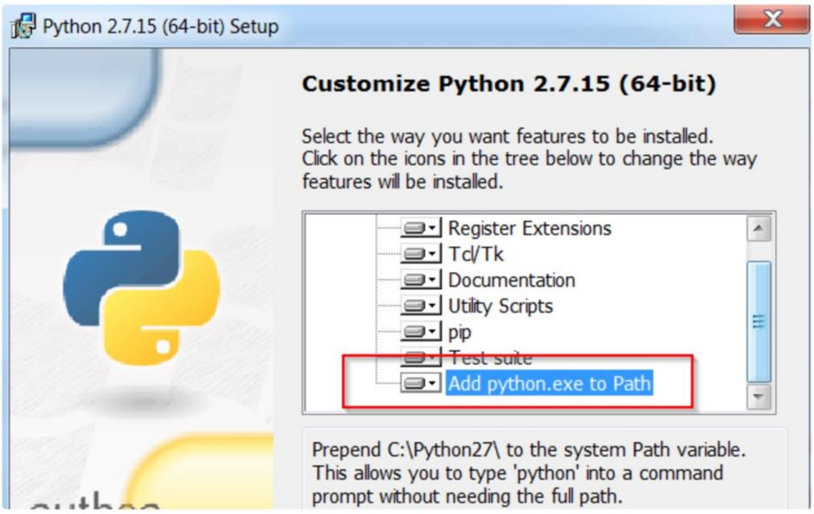
and: https://github.com/volatilityfoundation/volatility/compare/copy_data_block_order

Complete Installation instructions for Volatility on Windows 10:

These instructions are based on "Installing Volatility on Windows" by Mike Cary [55] , with slight modifications:

1. Download and install Python 2.7. (The Volatility setup script doesn’t currently support Python 3).

   ** Make sure to enable the option to add Python to Path during the installation as shown below. **



Python 2.7.15 (64-bit) Setup

Customize Python 2.7.15 (64-bit)

Select the way you want features to be installed.
Click on the icons in the tree below to change the way features will be installed.

Prepend C:\Python27\ to the system Path variable. This allows you to type 'python' into a command prompt without needing the full path.

2. Download the Volatility source from: https://github.com/volatilityfoundation/volatility/archive/master.zip

   and extract the files to \DFIR\volatility

3. Open a command prompt and create a virtual environment for Volatility:
   - `cd \Python27`
   - `pip install virtualenv`
   - `python -m pip install --upgrade pip`

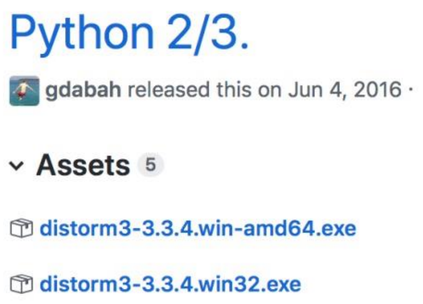
  - also install the virtualenv wrapper for windows, refer to: http://timmyreilly.azurewebsites.net/python-pip-virtualenv-installation-on-windows/
    `pip install virtualenvwrapper-win`
  - Create and Activate the Virtual Environment:
    - `mkvirtualenv vol`
    - `cd \DFIR\volatility`
    - `setprojectdir .`
  - other commands for Virtual Env:
    - `deactivate`
    - `workon vol`

4. Navigate to the location you extracted the Volatility source to and run “`setup.py install`”

5. Deactivate Virtualenv at this point: `deactivate`

6. Running “vol.py -h” at this point, would result in an error indicating that several dependencies are not installed. Use the links and commands below to install the following dependencies.

   - **diStorm3**: Download from https://github.com/gdabah/distorm/releases and run the executable to install

     Python 2/3.

     gdabah released this on Jun 4, 2016 ·

     Assets 5

     distorm3-3.3.4.win-amd64.exe

     distorm3-3.3.4.win32.exe

   - **pyCrypto**: Might be available from the install link on the Volatility Github for the pyCrypto: http://www.voidspace.org.uk/python/modules.shtml#pycrypto. If not, we can use pip to install, but will need to install the Microsoft C++ Compiler for Python 2.7 prior to doing so.

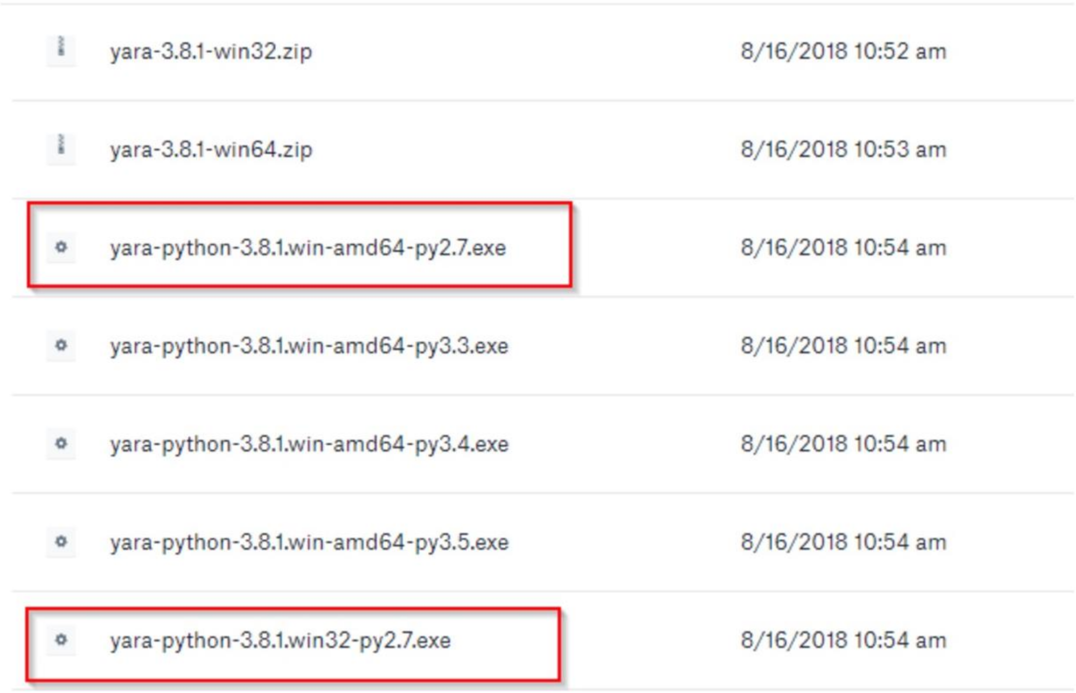
     => I used the pip install method:

     - Download and install Visual C++ Compiler for Python 2.7 from: https://www.microsoft.com/en-us/download/details.aspx?id=44266
     - From the command line type “`pip install pycrypto`”

   - **Yara**:

     https://www.dropbox.com/sh/umip8ndplytwzj1/AADdLRsrpJL1CM1vPVAxc5JZa?dl=0&lst=

     The dropbox link seems sketchy but that’s where the Volatility Github points to, when selecting the option for binary installers. There are several options on this page. Make sure to select one of the py2.7.exe options. Once downloaded, run the executable to install.



yara-3.8.1-win32.zip	8/16/2018 10:52 am
yara-3.8.1-win64.zip	8/16/2018 10:53 am
yara-python-3.8.1.win-amd64-py2.7.exe	8/16/2018 10:54 am
yara-python-3.8.1.win-amd64-py3.3.exe	8/16/2018 10:54 am
yara-python-3.8.1.win-amd64-py3.4.exe	8/16/2018 10:54 am
yara-python-3.8.1.win-amd64-py3.5.exe	8/16/2018 10:54 am
yara-python-3.8.1.win32-py2.7.exe	8/16/2018 10:54 am

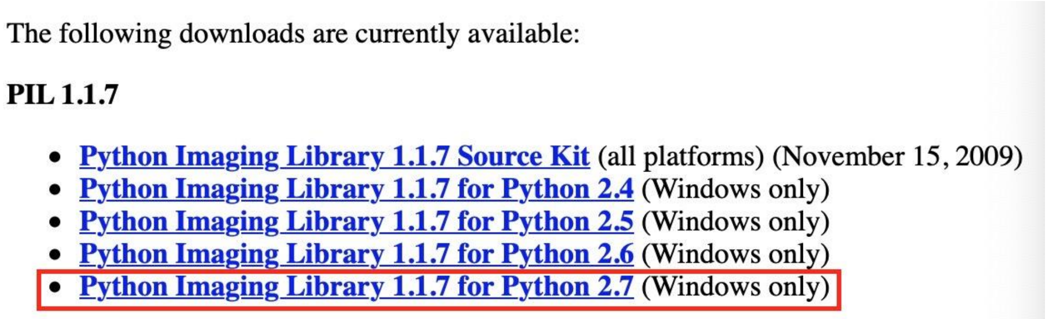
- **openpyxl:** There are no compiled Windows binaries so we will install by running

  “`pip install openpyxl`” from the command line

- **ujson**: There is no compiled binary installer for this one either so we will use PIP to install here too: “`pip install ujson`”

- **PIL**: Python Image Library can be found here:

  http://www.pythonware.com/products/pil/



The following downloads are currently available:

PIL 1.1.7

- Python Imaging Library 1.1.7 Source Kit (all platforms) (November 15, 2009)
- Python Imaging Library 1.1.7 for Python 2.4 (Windows only)
- Python Imaging Library 1.1.7 for Python 2.5 (Windows only)
- Python Imaging Library 1.1.7 for Python 2.6 (Windows only)
- Python Imaging Library 1.1.7 for Python 2.7 (Windows only)

=> download and install; but did not work on Windows 10...

Found the following hints:

https://stackoverflow.com/questions/20060096/installing-pil-with-pip/21151777

used: "`pip install pillow`"

7. Now activate the Virtualenv again: `workon vol`

8. At this point, Volatility is able to run, and "`vol.py --info`" shows the latest Win10 Profiles;

   nevertheless, Volatility still has trouble with the pslist output example:

```
(vol) C:\DFIR\volatility> vol.py -f c:\DFIR\DATA\memory_dump.raw
--profile=Win10x64_17763 pslist

Volatility Foundation Volatility Framework 2.6.1

Offset(V)    Name     PID     PPID     Thds     Hnds     Sess     Wow64
   Start     Exit

---------    -----   ----    -----    -----    -----    -----    -----
-   ------   -----

Traceback (most recent call last):

  File "C:\DFIR\volatility\vol.py", line 192, in <module> main()

  File "C:\DFIR\volatility\volatility\commands.py", line 147, in
execute func(outfd, data)

  File "C:\DFIR\volatility\volatility\plugins\taskmods.py", line
199, in render_text for task in data:

  File "C:\DFIR\volatility\volatility\win32\tasks.py", line 88,
in pslist for p in

       get_kdbg/addr_space).processes():

  File
"C:\DFIR\volatility\volatility\plugins\overlays\windows\kdbg_vty
pes.py", line 42, in processes

       raise AttributeError("(Could not list tasks, please
verify your --profile with kdbgscan")

       AttributeError: Could not list tasks, please verify your
--profile with kdbgscan

(vol) C:\DFIR\volatility>
```

=> to get rid of the issue, the "copy_data_block_order" fix needs to be installed; see: https://github.com/volatilityfoundation/volatility/issues/583

Changes in File: volatility/volatility/plugins/overlays/windows/win8_kdbg.py:

```
182 -    if (not block_encoded and op.mnemonic == "CMP" and
182 +    if (not (block_encoded or kdbg_block or wait_never or
wait_always) and
183 +    op.mnemonic == "CMP" and
192 -    elif (not kdbg_block and op.mnemonic == "LEA" and
193 +    elif (not (kdbg_block or wait_never or wait_always) and
194 +    op.mnemonic == "LEA" and
199 -    elif (not wait_never and op.mnemonic == "MOV" and
201 +    elif (not (wait_never or wait_always) and
202 +    op.mnemonic == "MOV" and
210 -    elif (not wait_always and op.mnemonic in ["MOV", "XOR"]
and
213 +    elif (not wait_always and
214 +    op.mnemonic in ["MOV", "XOR"] and
```

9. Test the installation, e.g. with

   ```
   vol.py -f c:\DFIR\DATA\memory_dump.raw --profile=Win10x64_17763
   pslist
   ```

10. Leave the Virtual Environment with

    ```
    deactivate
    ```

From now on, when leaving the environment, use: "`deactivate`" and to reactivate the environment, again, use: "`workon vol`"

B.2 Bromium: Rekall Installation

Installation References:

https://github.com/google/rekall

http://rekall-forensic.blogspot.ch/2015/09/installing-rekall-on-windows.html

- Python 2.7 => already installed for Volatility
- get the "Git Clone" Tool from: https://git-scm.com/download/win
  => downloads `GIT-2.20.1-64-bit.exe`
  => run the EXE
- Visual C++ for Python => already installed for Volatility
- `mkdir \DFIR\rekall`
- `mkdir \DFIR\GIT`
- `git clone https://github.com/google/rekall.git \DFIR\GIT\rekall`
- Virtualenv and mkVirtualenv are already installed for Volatility
  - `mkvirtualenv rekal`
  - `cd \DFIR\rekall`
  - `setprojectdir .`
- `pip install --upgrade setuptools pip wheel`
- `pip install --editable \DFIR\GIT\rekall\rekall-lib`
- `pip install --editable \DFIR\GIT\rekall\rekall-core`

  => received an error message:

  ```
  rekall-efilter 1.6.0 has requirement future==0.16.0, but you'll
  have future 0.17.2 which is incompatile
  ```

  => deinstall/reinstall:
  - `pip list`
  - `pip uninstall future`
  - `pip install future==0.16.0`
- `pip install --editable \DFIR\GIT\rekall\rekall-agent`

- `pip install --editable \DFIR\GIT\rekall`
- Now, let's test ☺
  - `winpmem -L`
  - `rekall --live Memory`
    - `pslist`
    - `psscan` (=> returns no data)
    - `netscan`
    - `imageinfo`
    - `kdbgscan` (=> returns no data)
    - `pstree`
- deactivate the Virtual Env for Rekall:
  - `deactivate`
  - to reactivate the Virtual Env, if needed, run:
    - `workon rekal`

B.3 Qubes: Volatility Installation

As we already know, that Volatility is not able to perform VMI, is might not be very useful on Qubes. Nevertheless, in order to have another tool for comparison, Volatility has been copied to Qubes as binary (source: https://www.volatilityfoundation.org/26) using a USB stick and in order to be able, to generate a profile for Volatility, the volatility-master has been copied from GitHUB (source: https://github.com/volatilityfoundation/volatility) as ZIP file (using the "Clone or download" button) and unzipped on Qubes.

B.4 Qubes: Rekall Installation in dom0

Prerequisites:

- collection of **PIP Packages**
- and **RPMs**

(as described in appendix F3 - "Using a Fedora-25 standalone VM, to collect PIP and RPM packages (for a Rekall Installation in dom0)"

1. Installing Rekall:
   ```
   [root@dom0 ~]# cd rekall
   [root@dom0 rekall]# source bin/activate

   (rekall) [root@dom0 rekall]# export PIP_FIND_LINKS=/root/pip-
   rekall-sources
   (rekall) [root@dom0 rekall]# export PIP_NO_INDEX=y
   (rekall) [root@dom0 rekall]# pip3 install pybindgen
   (rekall) [root@dom0 rekall]# pip3 install rekall-agent rekall
   future==0.16.0
   ```
   => seems to work, now

2.
   ```
   (rekall) [root@dom0 rekall]# bin/rekal live
   ```
   => fails with: `TypeError: collect() missing 1 required positional argument: 'renderer'`

3. `(rekall) [root@dom0 rekall]# bin/rekal --live Memory`
   `[1] Live(/proc/kcore) 08:35:32> pslist`
   => fails with: `RuntimeError: Unable to find a valid profile for this image.`
4. `(rekall) [root@dom0 rekall]# bin/rekal --live Memory -p ../Qubes-4.14.123-1.json pslist`
   => fails with: `CRITICAL:rekall.1:A DTB value was found but failed to verify.`

5. `see:` [`https://github.com/google/rekall/issues/493`](https://github.com/google/rekall/issues/493)
   `pip3 install pyaff4==0.26.post6`
   or: `pip install pyaff4==0.26.post6`
   => no success; still the same failure:
   `(rekall) [root@dom0 rekall]# bin/rekal live`
   => fails with: `TypeError: collect() missing 1 required positional argument: 'renderer'`

6. Testing memory collection with Rekall:
   `(rekall) [root@dom0 rekall]# bin/rekall aff4acquire rekall_dump.aff4`
   => seems to work; creates file: `rekall_dump.aff4`

7. Testing fix:
   [`https://github.com/google/rekall/commit/c5c6deead604aea70a1bd40f2a83bb22ce1d1afc`](https://github.com/google/rekall/commit/c5c6deead604aea70a1bd40f2a83bb22ce1d1afc)
   file: `rekall-core/rekall/plugins/linux/elf.py`
   in line 120:
   ```
   - yield dict(symbol=symbol_record,
   + yield dict(elf64_sym=symbol_record,
   ```

   file: `rekall-core/rekall/plugins/tools/disassembler.py`
   in line 580 add 2 new lines:
   ```
   + if data == None:
   + return
   => does not work, as *.py files look different!
   ```

B.5 Proxmox: Volatility Installation

Installation via apt install:

References:
https://github.com/volatilityfoundation/volatility/wiki/Installation
https://linoxide.com/linux-how-to/setup-volatility-memory-analysis/

- `# sudo -i`
- `# apt list --installed | grep volatility`
  => nothing installed
- `# apt list | grep volatility`
  ```
  volatility/stable 2.6-1 all
  volatility-tools/stable 2.6-1 all
  ```
- `# apt list --installed  grep python2`
  => shows python 2.7 is installed
- `# apt install volatility`
  => installs volatility and needed dependencies as e.g. dwarfdump
- `# find / -name vol.py`
  => /usr/share/volatility/vol.py
- `# /usr/share/volatility/vol.py`
  Volatility Foundation Framework 2.6

Building a Profile for the Proxmox System for Volatility:

- Check for needed packages:
  ```
  # apt list --installed dwarfdump build-essential
  ```
  (the Kernel Package "`pve-headers-4.15.18-10-pve`" has already been installed for Rekall)

-
  ```
  # cd /usr/share/volatility/tools/linux
  ```

-
  ```
  # make
  ```
  => creates: `module.dwarf`

-
  ```
  # ls /boot/System.map*
  ```
  => verify, that the file: ``System.map`uname -r` `` is present
- Find out, where volatility is installed:
  ```
  # find / -name overlays
  ```
  => shows: `/usr/lib/python2.7/dist-packages/volatility/plugins/overlays`
- Find out, which Linux Profiles are already available:
  ```
  # /usr/share/volatility/vol.py --info | grep Linux
  ```
- Create profile:
  ```
  # cd /usr/lib/python2.7/dist-packages/volatility/plugins/overlays/linux/
  # zip Debian9-Proxmox.zip \
  /usr/share/volatility/tools/linux/module.dwarf \
  /boot/System.map-`uname -r`
  ```
- Verify the new Profile Debian9-Proxmox has been created and is visible:
  ```
  # /usr/share/volatility/vol.py --info | grep LinuxDebian
  ```
  => returns: `LinuxDebian9-Proxmoxx64 - A Profile for Linux Debian9-Proxmox x64`

Creating a memory dump with LiME:

(Note: LiME had already been installed previously => just use it, here)

-
  ```
  # insmod /root/LiME/src/lime-4.15.18-10-pve.ko “path=/root/mem-images/mem-image_190418.lime format=lime”
  ```
  => seems to work;
  file: `/root/mem-images/mem-image_190418.lime` has been created

-
  ```
  # rmmod lime
  ```

Running/testing Volatility:

-
  ```
  # /usr/share/volatility/vol.py \
  -f /root/mem-images/mem-image_190418.lime \
  --profile=LinuxDebian9-Proxmoxx64 linux_pslist
  ```
  => returns an error about: `self.is_valid_profile` and `KeyError: 'DW_AT_byte_size'`
- Hints:
  - https://github.com/volatilityfoundation/volatility/issues/414
    Problem is caused for kernel version >= 4.9
    => solution: install latest volatility version from github

Consequently, tested a second installation with VirtEnv and GitHub:

-
  ```
  # sudo -i
  ```

-
  ```
  # virtualenv -p python2 /root/VirtEnv/volatility
  ```

-
  ```
  # echo "alias start-volatility-env='cd /root/VirtEnv/volatility;
  source /root/VirtEnv/volatility/bin/activate'" >> /root/.bashrc
  ```

-
  ```
  # . /root/.bashrc
  ```

-
  ```
  # start-volatility-env
  ```

-
  ```
  # git clone https://github.com/volatilityfoundation/volatility
  ```

-
  ```
  # cd volatility
  ```

-
  ```
  # chmod 755 vol.py
  ```

-
  ```
  # ./vol.py --info | grep Linux
  ```
  => the Proxmox Profile is not yet in the needed location

-
  ```
  # find /root/VirtEnv/volatility -name overlays
  ```

-
  ```
  # cp /usr/lib/python2.7/dist-packages/volatility/plugins/overlays/linux/Debian9-Proxmox.zip \
  volatility/plugins/overlays/linux/
  ```

-
  ```
  # ./vol.py --info | grep Linux
  ```
  => OK, now there is a Profile: `LinuxDebian9-Proxmoxx64`
- Testing Volatility, again:
  ```
  # /root/VirtEnv/volatility/volatility/vol.py \
  -f /root/mem-images/mem-image_190418.lime \
  --profile=LinuxDebian9-Proxmoxx64 linux_pslist
  ```
  => but still returns warnings: Failed to import volatility.plugins
  => see: https://github.com/volatilityfoundation/volatility/issues/535
- installing Prerequisites:
  ```
  # apt install yara
  # pip install distorm3 pycrypto openpyxl Pillow
  ```
- Testing Volatility, again:
  ```
  # /root/VirtEnv/volatility/volatility/vol.py \
  -f /root/mem-images/mem-image_190418.lime \
  --profile=LinuxDebian9-Proxmoxx64 linux_pslist
  ```
  => works, without returning any warnings, anymore...

B.6 Proxmox: Rekall Installation

-
  ```
  # sudo -i
  ```
- Verify and install needed packages:
  ```
  # apt list --installed | grep -e python3 -e virtualenv \
  -e build-essential -e python-dev -e python3-dev \
  -e libncurses5-dev -e pve-headers-4.15.18-10-pve -e zip \
  -e libelf-dev
  # apt install python3 virtualenv build-essential \
  python-dev python3-dev libncurses5-dev \
  pve-headers-4.15.18-10-pve zip libelf-dev
  ```
- Set up a Virtual Environment:
  ```
  # virtualenv -p python3 /root/VirtEnv/rekall
  ```
- Activate the virtual environment:
  ```
  # source /root/VirtEnv/rekall/bin/activate
  ```
  (to leave the virtual env later, just run: "`deactivate`")
- Install Rekall:
  ```
  # pip3 install --upgrade setuptools pip wheel
  # pip3 install PyBindGen
  # pip3 install rekall-agent rekall future==0.16.0
  ```
- Create a Rekall profile for the current system:

  -
    ```
    # cd /tmp/VirtEnv/rekall/tools/linux
    # make profile
    ```
    => works and creates `4.15.18-10-pve.zip`

  -
    ```
    # cd /tmp/VirtEnv/rekall
    ```

  -
    ```
    # rekall convert_profile tools/linux/4.15.18-10-pve.zip Proxmox-4.15.18-10-pve.json
    ```
- Test Rekall:
  ```
  rekall -p Proxmox-pve.json --live Memory pslist
  ```
  => returns a typical pslist-output, but shows timestamps with 1970
  => this will might to be dealt with, later

B.7 SIFT (Volatility and Rekall Updates)

The SIFT workstation has Rekall and Volatility preinstalled; nevertheless, in order to be able to work withmemory images of latest Windows and Linux Kernels, both, Rekall and Volatility need to be updated to the latest available versions.

- To get rid of the warning
  ```
  (in vol.py): "RequestsDependencyWarnings: Old version of
  cryptography ([1, 2, 3]) may cause slowdown."
  while running vol.py
  ```
  - `pip install --upgrade cryptography`
  - to get rid of the pip-is-old messge:
    `pip install --upgrade pip`
    - And to get rid of the additional, new error:
      ```
      "Failed to import
      volatility.plugins.community.TranVienHa.osint
      (AttributeError: 'module' object has no attribute
      'SSL_ST_INIT')"
      ```
      also run:
      `python -m easy_install --upgrade pyOpenSSL`
    - Source info:
      https://stackoverflow.com/questions/51283708/python-pip-package-requestsdependencywarning-when-installing-elastic-search-cura
- Update Volatility to the latest version to get the latest Windows10 Profiles
  - Find the directory, in which volatiltiy is installed:
    `# find / -name volatility`
    => /usr/lib/python2.7/dist-packages/volatility
  - Copy the latest volatility sources:
    # git clone https://github.com/volatilityfoundation/volatility.git /root/volatility_2.6_190412
  - Remove the former Volatility installation:
    - rm -rf /usr/lib/python2.7/dist-packages/volatility*
    - rm /usr/bin/vol.py
    - rm /usr/contrib/plugins/examples.py
  - Install the latest version if Volatility
    - git clone https://github.com/volatilityfoundation/volatility.git /root/volatility_2.6_190412
    - cd /root/volatility_2,6_190412
    - chmod 755 setup.py
    - This installs Volatility to:
      - /usr/local/bin/vol.py
      - /usr/local/lib/python2.7/dist-packages/volatility-2.6.1-py2.7.egg
    - POSSIBLE - BUT definitely no good choice, as the end result is a mixed Volatility installation...:
      Update Volatility to the latest version (See: https://www.youtube.com/watch?v=Us1gbPqtdtY; 10Cubes "Volatility Profiles and Windows 10)
      as root:
      - git clone https://github.com/volatilityfoundation/volatility.git
      - cd volatility
      - chmod 755 setup.py
      - ./setup.py install
      - chmod 755 vol.py
      - vol.py --info | grep Win10
- create a Linux Profile for Volatility (see: "Profiles - how to create")
- create a Linux Profile for Rekall (see: "Profiles - how to create")
- install Lime:

  - `sudo -i`
  - `git clone https://github.com/504ensicsLabs/LiME`
  - `cd LiME/src`
  - `make` (=> creates the file: `lime-<Kernel-Version>.ko`)
  - `cd ~`
  - **`insmod ./LiME/src/lime-<Kernel-Version>.ko "path=<output of dump> format=lime [dio=0|1]"`**
    - **path** the output file for the memory dump
    - **format** is the type of dump
      - **raw** – concatenates all System RAM ranges
      - **padded** – pads all non-System RAM ranges with 0s
      - **lime** – each range prepended with fixed-size header containing address space info
    - **dio**
      - 0 – default, do not attempt Direct IO
      - 1 – attempt to enable Direct IO
  - `rmmod lime`
- Creating Memory Images on SIFT:
  - LiME/raw:

    ```
    insmod /root/LiME/src/lime-4.4.0-97-generic.ko
    "path=/tmp/mem-image-lime.raw format=raw"
    rmmod lime
    ```
  - LiME/lime:

    ```
    insmod /root/LiME/src/lime-4.4.0-97-generic.ko
    "path=/tmp/mem-image-lime.lime format=lime"
    rmmod lime
    ```
  - pmem/raw:

    ```
    insmod /opt/rekall/tools/linux/pmem.ko
    ll /dev/pmem
    dd if//dev/pmem of=/tmp/mem-image-pmem.raw
    rmmod pmem
    ```
  - Find Memory images:

    ```
    ll /tmp/mem*
    -r--r--r-- 1 root root 2147019840 Apr 12 19:57 /tmp/mem-
    image-lime.lime
    -r--r--r-- 1 root root 2147019776 Apr 12 19:53 /tmp/mem-
    image-lime.raw
    -rw-r--r-- 1 root root 2147418111 Apr 12 20:00 /tmp/mem-
    image-pmem.raw
    ```
- Testing the images with Volatility:
  - `vol.py --file=/tmp/mem-image-lime.lime --profile=LinuxUbuntu1604.SIFTx64 linux_pslist`
    => works
  - `vol.py --file=/tmp/mem-image-lime.raw --profile=LinuxUbuntu1604.SIFTx64 linux_pslist`
    => returns error: `Failed valid address Space check`
  - `vol.py --file=/tmp/mem-image-pmem.raw --profile=LinuxUbuntu1604.SIFTx64 linux_pslist`
    => works
- Testing the images with Rekall:
  - `rekall --filename /tmp/mem-image-lime.lime --profile /opt/rekall/Ubuntu1604-SIFT.json pslist`
    => works
  - `rekall --filename /tmp/mem-image-lime.raw --profile /opt/rekall/Ubuntu1604-SIFT.json pslist`
    => returns error: `A DTB Value was found but failed to verify.`
  - `rekall --filename /tmp/mem-image-pmem.raw --profile /opt/rekall/Ubuntu1604-SIFT.json pslist`
    => works
- Updating Rekall (14.4.2019) => DON'T DO THIS!!! => breaks pslist on mem-images !!!

- `rekall --live Memory --profile /opt/rekall/Ubuntu1604-SIFT.json pslist`
  => returns timestamps from 1970 (!!!)
- `pip install fastchunking --no-cache-dir`
- `pip install PyYAML==3.12`
- `pip install --upgrade rekall -no-cache-dir future==0.16.0 --ignore-installed PyYAML`
- possibly, see: https://github.com/teamdfir/sift/issues/323

B.8 SIFT: Testing to install/run Actaeon

18.07.2019

Actaeon => see: http://s3.eurecom.fr/tools/actaeon/

19.07.19 - Testing with Volatility , Actaeon, ...

Testing an Actaeon Installation:

```
# wget https://raw.github.com/eurecom-s3/actaeon/master/utils/actaeon_setup.sh
# bash actaeon_setup.sh
```

=> fails because of missing svn-command / subversion product

```
# apt install subversion

# bash actaeon_setup.sh
```

=> fails again because http://hyperdbg/googlecoce.com/svn/trunk is not available, anymore

=> OK, let's try it, based on the manual installation...

```
:: Manual Installation
[-- Step 0x00
# cd /root
# git clone git://github.com/eurecom-s3/actaeon.git
```

=> OK, works

```
[-- 0x01 - Dumper
# cd root
# svn checkout http://hyperdbg.googlecode.com/svn/trunk/ hyperdbg-read-only
```

=> fails, as hyperdbg.googlecode.com is not available anymore
found: https://github.com/rmusser01/hyperdbg

```
# git clone https://github.com/rmusser01/hyperdbg
cd hyperdbg
cp /root/actaeon/dumper/hdbg.diff .
patch -p0 < hdbg.diff
=> fails... version not compatible...
```

```
[-- 0x02 - Volatility patch
cd /root/volatility
cp /root/actaeon/vol_patch/*.diff .
patch -p0 < intel.diff
```

=> fails completely => not compatible with Volatility 2.6

=> Cleanup:

```
# cd volatility/plugins/addrspaces
# mv intel.py.orig intel.py
# rm intel.py.rej
```

skip the next step, as it won't work anyway:

```
# patch -p0 < windows.diff
```

=> not run at all...

```
[-- 0x03 - Volatility Plugin
ll /root/actaeon/plugin/hypervisors
cd /root/volatility/volatility/plugins/
cp -R /root/actaeon/plugin/hypervisors/ ./
```

running a diff on the outputs of the command

```
      "python volatility/vol.py -f /SHARED/Proxmox_190711_mem.raw"
```

before and after the Actaeon patch installation shows: two new options:

```
-e EPT, --ept=EPT      EPT pointer from VMCS
hyperls                Detect hypervisors using Intel VT-x technology.
```

Running the hyperls Plugin without additional options, returns, that only westmere, sandy and penryn are supported platforms.

But neither of them is able to find a valid VMCS in the Proxmox dump:

```
# python volatility/vol.py -f /SHARED/Proxmox_190711_mem.raw hyperls \
-m westmere
Volatility Foundation Volatility Framework 2.6.1

:: Looking for VMCS0N...
INFO    : volatility.debug    : >> Possible VMCS at 0xf21d8000
INFO    : volatility.debug    : [32 bit PAE] VMCS 0x1f21d8000 has not ben validated

:: Counting the hypervisors in the dump...
        |_ There are 0 hypervisors:

:: Counting Guests in the dump...
>> Via EPT  [FAIL]

# python volatility/vol.py -f /SHARED/Proxmox_190711_mem.raw hyperls -m sandy
Volatility Foundation Volatility Framework 2.6.1

:: Looking for VMCS0N...
:: Counting the hypervisors in the dump...
        |_ There are 0 hypervisors:

:: Counting Guests in the dump...
>> Via EPT  [FAIL]
```

```
# python volatility/vol.py -f /SHARED/Proxmox_190711_mem.raw hyperls \
-m penryn
Volatility Foundation Volatility Framework 2.6.1

:: Looking for VMCS0N...
INFO    : volatility.debug    : >> Possible VMCS at 0x158cb8000
INFO    : volatility.debug    : [64 bit] VMCS 0x158cb8000 has not ben validated

:: Counting the hypervisors in the dump...
        |_ There are 0 hypervisors:

:: Counting Guests in the dump...
>> Via EPT  [FAIL]
```

But the plugins/hypervisors/vmcs_layout subdirectoy also lists files for haswell and nehalem => possibly only the db.py file needs to be updated to activated those, too?

Testing with option -G for generic:

```
# python volatility/vol.py -f /SHARED/Proxmox_190711_mem.raw hyperls -G
Volatility Foundation Volatility Framework 2.6.1

:: Looking for VMCS0N...
Traceback (most recent call last):
  File "volatility/vol.py", line 192, in <module>
    main()
  File "volatility/vol.py", line 183, in main
    command.execute()
 File "/root/volatility/volatility/commands.py", line 147, in execute
    func(outfd, data)
 File "/root/volatility/volatility/plugins/hypervisors/vmm.py", line 1083, in
render_text
    for i in data:
 File "/root/volatility/volatility/plugins/hypervisors/vmm.py", line 763, in calculate
    self.find_prevalent_microarch(generic_vmcs,phy_space)
 File "/root/volatility/volatility/plugins/hypervisors/vmm.py", line 638, in
find_prevalent_microarch
    for key in layouts.revision_id_db.keys()
AttributeError: 'module' object has no attribute 'revision_id_db'
```

C Generating Profiles, needed for Volatility/Rekall on Linux

Generally, creating profiles for Linux systems works always in the same way. For Proxmox, the installed versions of Rekall and Volatility have been used (installations described in appendix B.5 and appendix B.6) And for Qubes, the un-zipped volatility-master and the installed Rekall have been used (see appendix F.4 or F.5 for details).

As can be seen during the examples, profile have also been created on the SIFT workstation.

C.1 Generating a Profile - Volatility

Source: https://www.youtube.com/watch?v=qoplmHxmOp4

and: https://github.com/volatilityfoundation/volatility/wiki/Linux

```
# cd /root/volatility/tools/linux
(# make clean)
```

Check for dwarf, build-essentials and linux-headers-generic:

```
# apt list --installed | grep -e dwarf -e build-essential \
  -e linux-headers-generic
```

if not present; install:

```
# apt install dwarfdump
(or: apt install libdwarf-tools)
# apt install build-essential
# apt install linux-headers-generic
```

Then run make to generate module.dwarf:

```
# make
# ll
```

=> verify, there is a new module.dwarf

Check for /boot/System.map-<version>-generic:

```
# ls  /boot/System.map*
(example: System.map-4.4.0-142-generic)
# cd /root/volatility
# zip volatility/plugins/overlays/linux/Ubuntu1604-<version>.zip \
  tools/linux/module.dwarf /boot/System.map-<version>-generic
```

Now, the new profile should be visible with --info in Volatility:

```
# /root/volatility/vol.py --info | grep Linux
```

=> check for the newly created Linux Profile

If the local original Volatility Version is installed in:

/usr/local/lib/python2.7/dist-packages/volatility-2.6.1-py2.7.egg/volatility

then run:

```
# cp /root/volatility/volatility/plugins/overlays/linux/Ubuntu1604-<version>.zip \

/usr/local/lib/python2.7/dist-packages/volatility-2.6.1-
py2.7.egg/volatility/plugins/overlays/linux/Ubuntu1604-<version>.zip
```

and then run (to verify):

```
# vol.py --info | grep Linux
```

Check for linux plugins:

```
# /root/volatility/vol.py --info | grep linux_
```

Testing the new profile:

```
# /root/volatility/vol.py --file=/tmp/SIFT_lime_image.lime \
  --profile=LinuxUbuntu1604-440-142x64 linux_pslist
```

=> works with Volatility

```
# /root/volatility/vol.py --file=/tmp/SIFT_lime_image.raw \
  --profile=LinuxUbuntu1604-440-142x64 linux_pslist
```

=> results in an error: `No suitable address space mapping found`

```
# /root/volatility/vol.py --file=/tmp/SIFT_pmem_image.raw \
  --profile=LinuxUbuntu1604-440-142x64 linux_pslist
```

=> works with Volatility

As has been seen several times, Volatility is able to work with raw images generated with pmem or with lime-formatted images, generated with LiME.
raw-formatted images, generated with LiME do not work...

ATTENTION: There are possible issues with Fedora:
https://github.com/volatilityfoundation/volatility/wiki/Linux
https://code.google.com/archive/p/volatility/issues/355

C.2 Generating a Profile - Rekall

see: http://blog.rekall-forensic.com/2014/02/the-rekall-profile-repository-and.html

by default, Profiles created/collected/provided by the Rekall team, are used

nevertheless, for unusual profiles or, if the analysis system has no network access, local repositories can be used

install and update a local profile repository:

```
$ git clone https://code.google.com/p/rekall.profiles

# It is possible to update the local mirror with the latest public
profiles.

$ cd rekall.profiles
$ git pull

# Now we can tell Rekall to use the local repository

$ rekall -f memory_vm_10_7.dd.E01 \
  --profile_path full_path_to_rekall.profiles \
  --profile OSX/10.7.4_AMD -v pslist
```

Example, manually creating a **Linux Profile** (e.g. on the SIFT workstation):

Source: http://memory-analysis.rekall-forensic.com/www/06-Linux_Memory_analysis/

use the provided module and makefile from 'tools/linux' in the Rekall source tree

```
# find / -name rekall
```

=> /opt/rekall

```
# cd /opt/rekall/tools/linux
```

continue with the instructions from the README file (which is located in the rekall/linux/tools dir)

```
# make profile
```

=> results in additional files, including: 4.4.0-142-generic.zip

```
# cd /opt/rekall
# rekall convert_profile tools/linux/4.4.0-142-generic.zip Ubuntu-
4.4.0-142-generic.json
```

afterwards, rekall can be used, like this:

```
# rekall --profile Ubuntu-4.4.0-142-generic.json -f my-image.dd
```

or:

```
# rekall --profile Ubuntu-4.4.0-142-generic.json --live Memory
```

D LiME Installations

D.1 Qubes: LiME - creating a LiME module on Qubes (dom0)

- get sources as ZIP from: https://github.com/504ensicsLabs/LiME
- unzip LiME-master.zip in /root

-
  ```
  [root@dom0 ~]# cd LiME-master/src
  ```

-
  ```
  [root@dom0 src]# make
  ```
  => this generates the LKM: `lime-4.14.123-1.pvops.qubes.x86_64.ko`

-
  ```
  [root@dom0 src]# insmod lime-4.14.123-1.pvops.qubes.x86_64.ko
  "path=/root/Qubes_lime_image_190619.lime format=lime"
  ```

-
  ```
  [root@dom0 src]# rmmod lime
  ```

-
  ```
  [root@dom0 src]# insmod lime-4.14.123-1.pvops.qubes.x86_64.ko
  "path=/root/Qubes_lime_image_190619.raw format=raw"
  ```

-
  ```
  [root@dom0 src]# rmmod lime
  ```

D.2 Proxmox: LiME Installation

-
  ```
  # sudo -i
  ```

-
  ```
  # git clone https://github.com/504ensicsLabs/LiME
  ```
  => results in an error about missing "git"; therefore:
  ```
  # apt install git
  # git clone https://github.com/504ensicsLabs/LiME
  ```

-
  ```
  # cd Lime/src
  ```
- `# make` (=> creates the File: `lime-<Kernel-Version>.ko`)

-
  ```
  # cd ~
  ```

-
  ```
  # insmod /root/LiME/src/lime-4.15.18-10-pve.ko \
  “path=/tmp/mem-image.lime format=lime”
  ```
  => seems to work (in contrary to the pmem module)

-
  ```
  # rmmod lime
  ```
- Testing the lime-format-image with Rekall:
  ```
  # cd /root/VirtEnv/rekall
  # source /root/VirtEnv/rekall/bin/activate
  # rekall -p Proxmox-pve.json -f /tmp/mem-image.lime pslist
  ```
  => results in errors:
  ```
  TypeError: Set() missing 1 required positional argument: 'value'
  ```

-
  ```
  # insmod /root/LiME/src/lime-4.15.18-10-pve.ko “path=/tmp/mem-
  image.raw format=raw”
  ```

-
  ```
  # rmmod lime
  ```
- Testing the raw-format image with Rekall:
  ```
  # cd /root/VirtEnv/rekall
  # source /root/VirtEnv/rekall/bin/activate
  # rekall -p Proxmox-pve.json -f /tmp/mem-image.raw pslist
  ```
  => also results in errors:
  ```
  # TypeError: Set() missing 1 required positional argument: 'value'
  ```

E PMEM Installations

E.1 Bromium: winpmem Installation

- ATTENTION:
  error: "Imaging failed with error: -8"
  => might occur, if pmem is already in use, e.g. by a live rekal session
- `winpmem -L` (to load winpmem)
- `winpmem -U` (to unload winpmem)
- # collect a raw memory dump
  ```
  winpmem -m --format raw -o c:\memory_dump.zip
  ```
- => open the ZIP-File and extract `PhysicalMemory` to `memory_dump.raw`
- => can be used with either Rekall or Volatility
- or, according to: https://www.forensicswiki.org/wiki/Rekall (not verified, yet; 11.4.2019)
  - `winpmem.exe c:\DFIR\DATA\image.raw`
- `and according to the SANS Cheat-Sheet:` https://digital-forensics.sans.org/media/rekall-memory-forensics-cheatsheet.pdf
  - CREATING AN AFF4
    - (Open cmd.exe as Administrator)
      ```
      C:\> winpmem.exe -o output.aff4
      ```
    - INCLUDE PAGE FILE:
      ```
      C:\> winpmem.exe -p c:\pagefile.sys -o output.aff4
      ```
  - EXTRACTING TO RAW MEMORY IMAGE, FROM AFF4
    ```
    C:\> winpmem.exe output.aff4 --export PhysicalMemory \
         -o memory.img
    ```
  - EXTRACTING TO RAW, USING REKALL
    ```
    C:\> rekal -f win7.aff4 imagecopy \
         --output-image="/cases/win7.img"
    ```

E.2 Qubes: linpmem Installation & Usage

- Download linpmem binaries from:
  - http://releases.rekall-forensic.com/v1.5.1/linpmem-2.1.post4
  - https://github.com/Velocidex/c-aff4/releases/download/1.0.rc2/linpmem_3.0rc2.bin
  - https://github.com/Velocidex/c-aff4/releases/download/v3.3.rc1/linpmem-v3.3.rc1
- put files on a USB stick and attach to the Qubes system
- `[root@dom0 ~]# qvm-block attach dom0 sys-usb:sda2`
- `[root@dom0 ~]# mount /dev/xvdi mnt`
- `[root@dom0 ~]# cp mnt/linpmem* ./`
- `[root@dom0 ~]# umount mnt`
- `[root@dom0 ~]# qvm-block detach dom0 sys-usb:sda2`
- `[root@dom0 ~]# ./linpmem-2.1.post4 -o mem.aff4`

  => seems to work

- `[root@dom0 ~]# ./linpmem_3.0rc2.bin -o mem.aff4`

  => seems to work in the beginning, but then returns:

  ```
  E Unable to find storage for AFFDirectory file:///boot/efi/
  E Unable to find file:///boot/efi/
  E Unable to find storage for AFFDirectory file:///boot/grub/
  E Unable to find file:///boot/grub/
  E Unable to find storage for AFFDirectory file:///boot/grub2/
  E Unable to find file:///boot/grub2/
  E Unable to find storage for AFFDirectory file:///boot/loader/
  E Unable to find file:///boot/loader/
  E Unable to find storage for AFFDirectory file:///boot/lost%2Bfound/
  E Unable to find file:///boot/lost%2Bfound/
  ```

- `[root@dom0 ~]# ./linpmem-v3.3.rc1 -o mem.aff4`

  => directly fails with:

  ```
  ./linpmem-v3.3.rc1: /lib64/libc.so.6: version 'GLIBC_2.27' not found
  (required by ./linpmem-v3.3.rc1)
  ./linpmem-v3.3.rc1: /lib64/libc.so.6: version 'GLIBC_2.25' not found
  (required by ./linpmem-v3.3.rc1)
  ```

E.3 Linux: pmem Driver (provided with Rekall) - Compilation & Usage

Source: https://www.forensicswiki.org/wiki/Rekall#pmem

- Drivers can be found under: `rekall/tools/linux`
- To build the kernel module for the current kernel version, make sure you have a working build environment and the kernel headers installed. Change into this directory and run make:
  - `cd rekall/tools/linux`
  - `make`
- the acquisition driver is named: `pmem.ko`
- to load the driver: **`insmod pmem.ko`**
- to verify the driver is running: `lsmod | grep pmem`
- the driver creates a new device file: `/dev/pmem`
- to acquire a memory image: **`dd if=/dev/pmem of=/tmp/image.raw`**
- to unload the driver: **`rmmod pmem.ko`**
- for further information refer to: `rekall/tools/linux/README`

F Additional Installations and Memory Acquisition on Qubes

F.1 Installation of a modified Kernel in dom0

The given procedure uses a Fedora 28 standalone VM.

1. start a terminal in dom0 and become root:
   ```
   [user@dom0 ~]$ sudo -i
   ```
2. install the Fedora 28 template:
   ```
   [root@dom0 ~]# qubes-dom0-update qubes-template-fedora-28
   ```
3. from the main menue in Qubes, select "Create Qubes VM", with options:
   1. Name and label: "`QubesDEV`" (color: blue)
   2. Type: "`Standalone qube based on a template`"
   3. Template: "`fedora-28`"
   4. Networking: "`default (sys-firewall)`"
   5. and confirm with: OK
4. run "Qube Manager" and increase "Private storage max size" and "System storage max size" to 20480 MiB
5. launch terminal from the new qube: QubesDEV
6. => ATTENTION: all the following steps need to be executed as USER!
   ```
   [user@QubesDEV ~]$ git clone https://github.com/QubesOS/qubes-builder
   ```

7.
   ```
   [user@QubesDEV ~]$ cd qubes-builder
   ```

8.
   ```
   [user@QubesDEV ~]$ cp example-configs/qubes-os-master.conf builder.conf
   ```
9. configure the builder.conf file:
   ```
   [user@QubesDEV ~]$ vi builder.conf
   ```
   add the following lines at the end:
   ```
   COMPONENTS=linux-kernel builder-rpm builder
   GIT_URL_linux_kernel = $(GIT_BASEURL)/QubesOS/qubes-linux-kernel.git
   BRANCH_linux_kernel = stable-4.14
   ```
   Modify the following lines:
   "`DIST_DOM0 ?= fc29`" to "`DIST_DOM0 ?= fc25`"
   "`DIST_VM ?= fc29 stretch`" to "`DIST_VM ?= fc26 fc28`"

10.
    ```
    [user@QubesDEV ~]$ time make get-sources NO_CHECK=1
    ```
    ATTENTION: `warning: Macro expanded in comment on line9 %define _unpackaged_files_terminate_build 0`
    SOLUTION: `edit qubes-srv/linux-kernel/kernel.spec line: 9`
    => change "`^#%define`" to "`^%define`" => fixes the make error in step 16 (had "`^# %define`" in between, but then the rpm built after the kernel built failed)

11.
    ```
    [user@QubesDEV ~]$ time make install-deps
    ```
12. make sure https is used (instead of http):
    ```
    [user@QubesDEV ~]$ cd qubes-src/builder-rpm/repos/
    [user@QubesDEV ~]$ for file in qubes-repo-*-fedora.repo
    do
    echo "Processing file: ${file}"
    cp $file ${file}.org
    cat ${file}.org | sed -e 's/http:/https:/g' > $file
    diff $file ${file}.org
    done
    [user@QubesDEV ~]$ grep http qubes-repo-*-fedora.repo
    [user@QubesDEV ~]$ rm qubes-repo-*-fedora.repo.org
    ```

13.
    ```
    [user@QubesDEV ~]$ cd -
    ```

14. find out, where the KCORE parameter needs to be set:
    ```
    [user@QubesDEV ~]$ grep KCORE qubes-src/linux-kernel/config-*
    ```
15. Adjust the KCORE setting:
    ```
    [user@QubesDEV ~]$ vi qubes-src/linux-kernel/config-qubes
    ```
    => change the line, that contains KCORE to `"CONFIG_PROC_KCORE=y"`

16.
    ```
    [user@QubesDEV ~]$ time make linux-kernel
    USE_QUBES_REPO_VERSION=4.0 USE_QUBES_REPO_TESTING=1
    ```
    ERROR from "make linux-kernel":
    ```
      make[1]: *** [/home/user/qubes-builder/qubes-src/builder-
    rpm/Makefile-legacy.rpmbuilder:35:
                                /home/user/qubes-builder/chroot-dom0-
    fc25/home/user/.prepared_base] Error 1
        make[1]: Leaving directory '/home/user/qubes-builder'
        make_ *** [Makefile:231: linux-kernel-dom0] Error 1
    ```
    => see step 10 for the fix
17. Progress of the kernel build is visible in: `/home/user/qubes-builder/build-logs/linux-kernel-dom0-fc25.log`
    ```
    [root@QubesDEV ~]# tail -f /home/user/qubes-builder/build-
    logs/linux-kernel-dom0-fc25.log
    ```
    and the temporary kernel is built in directory:
    ```
    /home/user/qubes-builder/chroot-dom0-
    fc25/home/user/rpmbuild/BUILD/kernel-414-4.14.123/
    ```
18. Unfortunately, the kernel build failed; with an error message:
    error: Installed (but unpackaged) file(s) found:
    /usr/lib/debug/lib/modules/4.14.123-1.pvops.qubes.x86-64/vmlinux
    => see step 10 for the fix:
    `edit qubes-srv/linux-kernel/kernel.spec line: 9`
    => change `"^#%define"` to `"^%define"`
19. This time it worked, final rpms are:
    ```
    kernel-414-4.14.123-1.pvops.qubes.x86_64.rpm
    kernel-414-devel-4.14.123-1.pvops.qubes.x86_64.rpm
    kernel-414-qubes-vm-4.14.123-1.pvops.qubes.x86_64.rpm
    ```
    in directory: `/home/user/qubes-builder/qubes-src/linux-kernel/pkgs/dom0-fc25/x86_64/`

20.
    ```
    [user@dom0 ~]$ qvm-run --pass-io QubesDEV 'cat ~/qubes-
    builder/qubes-src/linux-kernel/pkgs/dom0-fc25/x86_64/kernel-414-
    4.14.123-1.pvops.qubes.x86_64.rpm' > kernel-414-4.14.123-
    1.pvops.qubes.x86_64.rpm
    [user@dom0 ~]$ qvm-run --pass-io QubesDEV 'cat ~/qubes-
    builder/qubes-src/linux-kernel/pkgs/dom0-fc25/x86_64/kernel-414-
    devel-4.14.123-1.pvops.qubes.x86_64.rpm' > kernel-414-devel-
    4.14.123-1.pvops.qubes.x86_64.rpm
    [user@dom0 ~]$ qvm-run --pass-io QubesDEV 'cat ~/qubes-
    builder/qubes-src/linux-kernel/pkgs/dom0-fc25/x86_64/kernel-414-
    qubes-vm-4.14.123-1.pvops.qubes.x86_64.rpm' > kernel-414-
    qubes.vm-4.14.123-1.pvops.qubes.x86_64.rpm
    ```
    (continued on 19.6.2019)
21. increase DNF's limit of simultaneously installed packages from 3 to 5:
    ```
    [user@dom0 ~]$ sudo -i
    [root@dom0 ~]# vi /etc/dnf/dnf.conf
    ```
    change "installonly_limit" line to: `installonly_limit=5`
22. install the new kernel package:
    ```
    [root@dom0 ~]# dnf install /home/uschuell/kernel-414-4.14.123-
    1.pvops.qube-x86_64.rpm
    ```
23. verify new kernel is installed:
    ```
    [root@dom0 ~]# ll /boot/vmlinuz-4.14.123*
    -rw-r--r-- 1  root root 6177536 Jun 12 16:16 /boot/linux-
    4.14.123.1.pvopy.qubes.x86_64
    ```
24. verify new boot entries have been configured:
    ```
    [root@dom0 ~]# date
    ```

```
Wed Jun 19 11:20:04 EDT
[root@dom0 ~]# ll /boot/grub2/grub2.conf
-rw-r--r--.  1 root root 21153 Jun 19 11:16 /boot/grub2/grub.cfg
[root@dom0 ~]# ll /boot/initramfs-4.14.123*
-rw------- 1  root root 22774731 Jun 19 11:16 /boot/initramfs-
4.14.123-1.pvops.qubes.x86_64.img
```

25. reboot and verify new kernel is activated and /proc/kcore is present:

```
[user@dom0 ~]$ sudo -i
[root@dom0 ~]# uname -r
4.14.123-1.pvops.qubes.x86_64
[root@dom0 ~]# ll /proc/kcore
-r--------  1 root root 140737477885952 Jun 19 11:27 /proc/kcore
```

26. Test creating an aff4 image with linpmem:

```
[root@dom0 ~]# ./linpmem --output mem.aff4
=> creates tons of output:
[root@dom0 ~]# ll mem.aff4
-rwx-rx-rx  1 root root 3408024238 Jun 19 11:35 mem.aff4
```

F.2 Creating a Fedora-25 standalone VM and Installing Rekall in this VM

installation process:

1. start a terminal in dom0 and become root:
   ```
   [user@dom0 ~]$ sudo -i
   ```
2. install the Fedora 25 template:
   ```
   [root@dom0 ~]# qubes-dom0-update qubes-template-fedora-25
   ```
3. from the main menu in Qubes, select "`Create Qubes VM`", with options:
   1. Name and label: "`Forensics-DEV`" (color: red)
   2. Type: "`Standalone qube based on a template`"
   3. Template: "`fedora-25`"
   4. Networking: "`default (sys-firewall)`"
   5. and confirm with: OK
4. run "`Qube Manager`" and
   increase "`Private storage max size`" to 10240 MiB,
   as is already set for "`System storage max size`"
5. launch terminal from the new qube: Forensics-DEV
   => fails with message:
   "`cannot connect to qrexec agent for 60 sec, see /var/log/xen/console/guest-Forensics-DEV.log`"
   => see: https://github.com/QubesOS/qubes-issues/issues/4442
   => increased initial memory
6. Check for Prerequisites:
   ```
   [user@Forensics-DEV ~]$ sudo -i
   ```
   Known pre-requisites for Rekall are:
   ```
   python3 virtualenvpython-dev python-dev
   libncurses5-dev libelf-dev zip
   build-essentials kernel-headers
   ```
   (=> as many of those are named differently on Fedora
   => just try installing and then fix the missing packages
   => results => see step 7)
7. install missing SW:
   ```
   [root@Forensics-DEV ~]# dnf install python3-virtualenv
   [root@Forensics-DEV ~]# dnf install python2-virtualenv
   [root@Forensics-DEV ~]# dnf install ncurses-devel
   [root@Forensics-DEV ~]# dnf install patch
   [root@Forensics-DEV ~]# dnf install redhat-rpm-config
   [root@Forensics-DEV ~]# dnf install automake
   [root@Forensics-DEV ~]# dnf install gcc-c++
   ```
8. => ATTENTION: all the following steps need to be executed as USER!

```
[user@Forensics-DEV ~]$ virtualenv -p python3 rekall
[user@Forensics-DEV ~]$ source rekall/bin/activate
[user@Forensics-DEV ~]$ pip3 install --upgrade setuptools pip
wheel
[user@Forensics-DEV ~]$ pip3 install pybindgen
[user@Forensics-DEV ~]$ pip3 install rekall-agent rekall
future==0.16.0
[user@Forensics-DEV ~]$ pip install pyaff4==0.26.post6
```

=> seems to work...

9. Try working on the mem-dumps and with the partial profile from dom0...
10. copy partial profile from dom0 to Forensics-DEV and make it a real profile

    ```
    [root@dom0 ~]# qvm-copy-to-vm Forensics-DEV \
    rekall-master/tootls/linux/4.14.123-1.pvops.qubes.x86_64.zip
    [user@Forensics-DEV rekall]$ mv ~/QubesIncoming/dom0/4.14.*.zip \
    /home/user/rekall/
    [user@Forensics-DEV rekall]$ rekal convert_profile \
    4.14.123-1.pvops.qubes.x86_64.zip Qubes-4.14.123-1.json
    ```

    => seems to work => File: `Qubes-4.14.123-1.json` has been created in dir: `/home/user/rekall`
11. For testing, copy a memory image from dom0 to Fortensics-DEV:

    ```
    [root@dom0 ~]# qvm-copy-to-vm Forensics-DEV mem.aff4
    ```

Testing to make a rekall binary for dom0

https://github.com/google/rekall/tree/master/tools/installers

```
[root@Forensics-DEV rekall]# pwd
/home/user/rekall
[root@Forensics-DEV ~]# dpkg-buildpackage
bash: dpkg-buildpackage: command not found...
Install package "dpkg-dev" to provide command "dpkg-build-package"?
[N/y] y
```

=> but dpkg-build-package still fails with an error:

```
tail: cannot open 'debian/changelog' for reading: No such file or
directory
```

Unfortunately, pyinstaller scripts are only available for windows and darwin (OSX)

=> next idea => pack the rekall directory and put it on a USB stick:

```
[user@Forensics-DEV ~]$ cd /home/user
[user@Forensics-DEV ~]$ tar -cvf rekall.tar rekall
[user@Forensics-DEV ~]$ gzip rekall.tar
```

=> put rekall.tar.gz on USB

=> on dom0; copy tar file to a local directory and continue:

```
[root@dom0 ~]# gunzip rekall.tar.gz
[root@dom0 ~]# tar -xvf rekall.tar
```

=> fails with tons of messages about:
`"cannot change ownership to UID 1000, gid 1000"`
or: `"cannot create symlink to '/usr/lib64/python3.5/...'"`

Continue with dpkg-packagebuilder on Forensics-DEV:

```
[root@Forensics-DEV ~]# dnf install debhelper
```

manually created the files: `changelog, compat, control, copyright, rules` as copies from GitHub

```
[user@Forensics-DEV ~]$ dpkg-buildpackage
```

=> fails with:

```
dpkg-buildpackage: source package rekall-forensic
dpkg-buildpackage: source version 1.7.2
dpkg-buildpackage: source distribution RELEASED
dpkg-buildpackage: source changed by Rekall Team <rekall-
dev@googlegroups.com>
dpkg-buildpackage: host architecture amd64
 dpkg-source --before-build rekall
dpkg-checkbuilddeps: Unmet build dependencies: build-essential:native
debhelper (>= 9) python
dpkg-buildpackage: warning: build dependencies/conflicts unsatisfied;
aborting
dpkg-buildpackage: warning: (Use -d flag to override.)
```

Summary:

Building a standalone rekall binary on Fedora 25 does not seem possible. Neither is a pyinstaller script provided for linux, nor does the dpkg-buildpackage command work on Fedora.

F.3 Using a Fedora-25 standalone VM, to collect PIP and RPM packages (for a Rekall Installation in dom0)

The idea of using pip download and dnf download has been presented by Michael Cohen at DFRWS 2016 USA [50]. Based in this idea, the following installation method has been used:

```
[user@Forensics-DEV ~]$ mkdir pip
[user@Forensics-DEV ~]$ cd pip

[user@Forensics-DEV pip]$ pip3 download setuptools pip wheel
[user@Forensics-DEV pip]$ pip3 download pybindgen rekall-agent rekall
[user@Forensics-DEV pip]$ pip3 download future==0.16.0
[user@Forensics-DEV pip]$ pip3 pyaff4==0.26.post6
[user@Forensics-DEV pip]$ pip3 download virtualenv

[user@Forensics-DEV pip]$ cd ..
[user@Forensics-DEV ~]$ tar -cvhzf pip-rekall-sources.tgz \
--owner=root --group=root pip
```

=> write pip-rekall-source.tgz to USB and attach this stick to dom0, e.g. /root/mnt

```
[root@dom0 ~]# cp mnt/pip-rekall-sources.tgz ./
[root@dom0 ~]# tar -tvzf pip-rekall-sources.tgz
[root@dom0 ~]# tar -xvzf pip-rekall-sources.tgz
[root@dom0 ~]# mv pip pip-rekall-sources

[root@dom0 ~]# export PIP_FIND_LINKS=/root/pip-rekall-sources
[root@dom0 ~]# export PIP_NO_INDEX=y
[root@dom0 ~]# pip3 install virtualenv
 (=> or without variables: pip3 install -f pip-rekall-sources --no-index
virtualenv)
[root@dom0 ~]# virtualenv -p python3 rekall
[root@dom0 ~]# cd rekall
[root@dom0 ~]# source bin/activate
```

```
[root@dom0 ~]# export PIP_FIND_LINKS=/root/pip-rekall-sources
[root@dom0 ~]# export PIP_NO_INDEX=y
[root@dom0 ~]# pip3 install pybindgen
[root@dom0 ~]# pip3 install rekall-agent rekall future==0.16.0
```

=> this fails because of missing RPM packages, which include at least:
`ncurses-dev, patch, redhat-rpm-config automake gcc-c++`

As these packages are not available for dom0 / from the Qubes sources, they will also have to be collected in Forensic-DEV (the Fedora 25 standalone VM), using "`dnf download`":

1. use dnf download:
   ```
   [user@Forensics-DEV ~]$ mkdir RPMs
   [user@Forensics-DEV ~]$ cd RPMs
   [user@Forensics-DEV RPMs]$ dnf download python3-virtualenv
   [user@Forensics-DEV RPMs]$ dnf download python2-virtualenv
   [user@Forensics-DEV RPMs]$ dnf download ncurses-devel
   [user@Forensics-DEV RPMs]$ dnf download patch
   [user@Forensics-DEV RPMs]$ dnf download redhat-rpm-config
   [user@Forensics-DEV RPMs]$ dnf download automake
   [user@Forensics-DEV RPMs]$ dnf download gcc-c++
   ```
2. provide RPMs to dom0 using a USB stick
3. References: https://unix.stackexchange.com/questions/249167/install-locally-using-dnf-in-fedora-without-using-internet-connection
   https://linoxide.com/linux-how-to/example-install-rpm-to-a-different-location-or-directory/
4. check whether installing RPMs from the local rpm-files to a different location is possible:
   ```
   [user@Forensics-DEV ~]$ rpm -qpi *.rpm | grep -e Name -e Arch -e
   Relo
   Name        : automake
   Architecture: noarch
   Relocations : (not relocatable)
   Name        : gcc-c++
   Architecture: x86_64
   Relocations : (not relocatable)
   Name        : ncurses-devel
   Architecture: x86_64
   Relocations : (not relocatable)
   Name        : patch
   Architecture: x86_64
   Relocations : (not relocatable)
   Name        : python2-virtualenv
   Architecture: noarch
   Relocations : (not relocatable)
   Name        : python3-virtualenv
   Architecture: noarch
   Relocations : (not relocatable)
   Name        : redhat-rpm-config
   Architecture: noarch
   Relocations : (not relocatable)
   ```

   => Result: RPMs are not relocatable => only local installations in standard root are possible...

Test installing RPMs locally in dom0 (using USB to transfer from Forensics-DEV):

1. install RPMs in dom0 (using USB)
   ```
   [user@Forensics-DEV ~] tar -cvzf RPM.tgz RPMs/
   ```

```
[root@dom0 ~]# tar -xvzf RPM.tgz
[root@dom0 ~]# dnf --disablerepo='*' install XXX.rpm
```

```
Error: nothing provides dwz >= 0.4 needed by redhat-rpm-config-45-1.fc25.noarch.
nothing provides python3-devel needed by python3-virtualenv-15.0.3-2.fc25.noarch.
nothing provides python2-devel needed by python2-virtualenv-15.0.3-2.fc25.noarch.
nothing provides libncurses++.so.6()(64bit) needed by ncurses-devel-6.0-
6.20160709.fc25.noarch.
nothing provides gcc = 6.4.1-1.fc25 needed by gcc-c++-6.4.1-1.fc25.noarch.
nothing provides autoconf => 2.65 needed by automake-1.15-7.fc25.noarch.
(try to add '--allowerasing' to command line to replace conflicting packages)
```

2. get missing RPMs in Forensics-DEV:

```
[user@Forensics-DEV RPMs]$ dnf download dwz
[user@Forensics-DEV RPMs]$ dnf download python3-devel
[user@Forensics-DEV RPMs]$ dnf download python2-devel
[user@Forensics-DEV RPMs]$ dnf download ncurses-c++-libs
[user@Forensics-DEV RPMs]$ dnf download gcc
[user@Forensics-DEV RPMs]$ dnf download autoconf
[user@Forensics-DEV RPMs]$ rm *.i6*
[user@Forensics-DEV RPMs]$ cd ..
[user@Forensics-DEV ~] tar -cvzf RPM.tgz RPMs/
```

3.

```
[root@dom0 ~]# tar -xvzf mnt/RPM.tgz
[root@dom0 ~]# dnf --disablerepo='*' install RPMs/XXX.rpm
```

```
Error: nothing provides fpc-srpm-macros >= 0.4 needed by redhat-rpm-config-45-
1.fc25.noarch.
nothing provides python-rpm-macros needed by python3-devel-3.5.4-2.fc25.noarch.
nothing provides python-rpm-macros needed by python-devel-2.7.13-3.fc25.noarch.
nothing provides libstdc++ = 6.4.1-1.fc25 needed by gcc-c++-6.4.1-1.fc25.noarch.
nothing provides cpp = 6.4.1-1.fc25 needed by gcc-c++-6.4.1-1.fc25.noarch.
nothing provides perl(Thread::Queue) needed by automake-1.15-7.fc25.noarch.
nothing provides m4 >= 1.4.14 needed by autoconf-2.69-22.fc24.noarch.
nothing provides python-rpm-macros needed by python-devel-2.7.13-3.fc25.noarch.
nothing provides python-rpm-macros needed by python3-devel-3.5.4-2.fc25.noarch.
(try to add '--allowerasing' to command line to replace conflicting packages)
```

4.

```
[user@Forensics-DEV RPMs]$ dnf download fpc-srpm-macros
[user@Forensics-DEV RPMs]$ dnf download python-rpm-macros
[user@Forensics-DEV RPMs]$ dnf download libstdc++
[user@Forensics-DEV RPMs]$ dnf download cpp
[user@Forensics-DEV RPMs]$ dnf download m4
[user@Forensics-DEV RPMs]$ rm *.i6*
```

5.

```
[root@dom0 ~]# tar -xvzf mnt/RPM.tgz
[root@dom0 ~]# dnf --disablerepo='*' install RPMs/XXX.rpm
```

=> tested this several times and found the following missing RPMs:

```
ghc-srpm-macros, python3-rpm-macros, python-srpm-macros,
python2-rpm-macros, libstdc++-devel, libgomp, gnat-srpm-macros,
go-srpm-macros, ocaml-srpm-macros, perl-Thread-Queue,
perl-srpm-macros, qt5-srpm-macros
```

6.

```
[user@Forensics-DEV RPMs]$ dnf download ghc-srpm-macros
[user@Forensics-DEV RPMs]$ dnf download python3-rpm-macros
[user@Forensics-DEV RPMs]$ dnf download python2-rpm-macros
[user@Forensics-DEV RPMs]$ dnf download python-srpm-macros
[user@Forensics-DEV RPMs]$ dnf download libstdc++-devel
[user@Forensics-DEV RPMs]$ dnf download libgomp
[user@Forensics-DEV RPMs]$ dnf download gnat-srpm-macros
[user@Forensics-DEV RPMs]$ dnf download go-srpm-macros
[user@Forensics-DEV RPMs]$ dnf download ocaml-srpm-macros
[user@Forensics-DEV RPMs]$ dnf download perl-Thread-Queue
```

```
[user@Forensics-DEV RPMs]$ rm *.i6*
```

7.
   ```
   [root@dom0 ~]# tar -xvzf mnt/RPM.tgz
   [root@dom0 ~]# dnf --disablerepo='*' install RPMs/XXX.rpm
   ```
   => it finally worked

F.4 Creating a Profile for Rekall in dom0

- get the Rekall sources as ZIP from: https://github.com/google/rekall
-
  ```
  [root@dom0 ~]# unzip rekall-master.zip
  ```
-
  ```
  [root@dom0 ~]# cd rekall-master
  ```
-
  ```
  [root@dom0 rekall-master]# cd tools/linux
  ```
-
  ```
  [root@dom0 linux]# make profile
  ```
  => this fails with:
  ```
  /usr/src/linux-headers-4.14.123-1.pvops.qubes.x86_64: No such
  file or directory. Stop.
  recipe for target 'dwarf' failed
  ```
- => uncomment line 10 in Makefile for "Fedora" and comment line 5
-
  ```
  [root@dom0 linux]# make profile
  ```
  => works and creates new files, including `4.14.123-1.pvops.qubes.x86_64.zip`
-
  ```
  [root@dom0 linux]# cd /root/rekall-master/rekall-core/rekall
  ```
-
  ```
  [root@dom0 rekall]# python rekal.py convert_profile
  tools/linux/4.14.123-1.pvops.qubes.x86_64.zip Qubes-
  4.14.123.json
  ```
  => fails with:
  ```
  File "rekal.py", line 31, in <module>
  import rekall
  ImportError: No module named rekall
  ```
- To generate the needed Profile: `Qubes-4.14.123.json`
  the zip file: `4.14.123-1.pvops.qubes.x86_64.zip`
  can be taken to any working Rekall version and converted, there:
  ```
  # rekall convert_profile 4.14.123-1.pvops.qubes.x86_64.zip \
  Qubes-4.14.123.json
  ```

F.5 Creating a Profile for Volatility in dom0

1. see: https://github.com/volatilityfoundation/volatility/wiki/Linux
2. dwarfdump is not unavailable for Fedora-25, but libdwarf-tools are available
   => install:
   ```
   root@dom0 ~]# qubes-dom0-update libdwarf-tools
   ```
3. get the Volatility sources as ZIP
   from: https://github.com/volatilityfoundation/volatility
   and unpack the ZIP in /root:
   ```
   [root@dom0 ~]# qvm-block attach dom0 sys-usb:sda1
   ```

```
[root@dom0 ~]# mount /dev/xvdi mnt
[root@dom0 ~]# cp mnt/volatility-2.6.zip ./
[root@dom0 ~]# umount mnt
[root@dom0 ~]# qvm-block detach dom0 sys-usb.sda1
[root@dom0 ~]# unzip volatility-2.6.zip
```

4. for the "`generic-kernel-headers`" package on Fedora (Qubes) install the kernel-devel rpm (which has been generated together with the new kernel; see F.1 (Installation of a modified Kernel in dom0:

```
[root@dom0 ~]# dnf install /home/uschuell/kernel-414-devel-
4.14.123-1.pvops.qubes.x86_64.rpm
```

5. compile "module.dwarf":

```
[root@dom0 ~]# cd volatility-master/tools/linux
[root@dom0 linux]# make
```

   this creates the new file module.dwarf

```
[root@dom0 linux]# ll module.dwarf
-rw-r--r-- 1 root root 3030276 Jun 19 16:12 module.dwarf
```

6. pack the profile for Qubes 40 and kernel 4.14:

```
[root@dom0 linux]# zip /root/Qubes40-414.zip module.dwarf \
/boot/System.map-4.14.123-2.pvops.qubes.x86_64
[root@dom0 ~]# cd /root
[root@dom0 ~]# mkdir vol-profiles
[root@dom0 ~]# mv Qubes40-414.zip vol-profiles
[root@dom0 ~]# ./volatility_2.6-standalone \
--plugins=/root/vol-profiles --info | grep Linux
```

   => shows new profile: `LinuxQubes40-414x64`

7. Testing with the new profile fails:

```
[root@dom0 ~]# ./volatility_2.6-standalone \
--plugins=/root/vol-profiles -f mem.raw imageinfo
```

   results in: `KeyError: 'DW_AT_byte_size'`
   or:

```
[root@dom0 ~]# cp vol-profiles/Qubes40-414.zip \
volatility-master/volatility/plugins/overlays/linux/
[root@dom0 ~]# cd volatility_master
[root@dom0 volatility_master]# python vol.py -f ../mem.raw \
imageinfo
```

   also results in: `KeyError: 'DW_AT_byte_size'`

   This might be an issue in older Volatility Version together with newer Linux Kernels:
   https://github.com/teamdfir/sift/issues/305
   or:
   https://github.com/volatilityfoundation/volatility/pull/335

8. testing Volatility (extracted ZIP) with updated dwarf.py:

   - download the new dwarf.py from:
     https://github.com/volatilityfoundation/volatility/blob/master/volatility/dwarf.py
     as `dwarf.py_181008` put on USB stick and transfer to Qubes

   -
```
[root@dom0 ~]# cd /root
[root@dom0 ~]# qvm-block attach dom0
[root@dom0 ~]# mount /dev/xvdi mnt
[root@dom0 ~]# cp mnt/dwarf.py_181008 \
volatility-master/volatility
[root@dom0 ~]# cd /root/volatility-master/volatility
[root@dom0 volatility]# cp dwarf.py dwarf.py_161227
[root@dom0 volatility]# diff dwarf.py_*
51a52
```

```
> 'ssizetype' : 'long',
135d135
<
138d137
<
204c203,204
< self.vtypes[name = [ int(data['DW_AT_byte_size'], self.base), {} ]
---
> if 'DW_AT_declaration' not in data:
> self.vtypes[name = [ int(data['DW_AT_byte_size'], self.base), {} ]
[root@dom0 volatility]# cp dwarf.py_181008 dwarf.py
[root@dom0 volatility]# cd /root/volatility-master
[root@dom0 volatility-master]# python vol.py \
-f ../proc/kcore --profile=LinuxQubes40-414x64 linus_pslist
```

=> returns a new error `'No suitable address space mapping found'`

=> seems the Profile works, now, but the memory dump from linpmem does not fit...

G Testing

G.1 HW Setup

For testing, 2 Notebooks have been installed. The hardware details of each of these systems are:

- Hardware: HP EliteBook 820 G1
- CPU: Intel(R) CORE (TM) i5-4300U CPU @ 1.90GHz (which belongs to the Haswell family)
- 8GB RAM
- 256GB SSD (internal)
- 240 GB M.2 SATA/SSD (KingDian_N400)
- VTx and VTd are enabled in BIOS

Each of the SSDs and M.2 SATAs has been installed with one separate system:

Laptop No.	Serial	Disk	OS
1	5CG43520VK	internal SSD	Windows 10 Pro / Bromium
		M.2 SATA	Qubes
2	5CG4501QDV	internal SSD	Windows 10 Pro / SIFT
		M.2 SATA	PROXMOX

G.2 Bromium

G.2.1 Test-Setup

The test installation consists of:

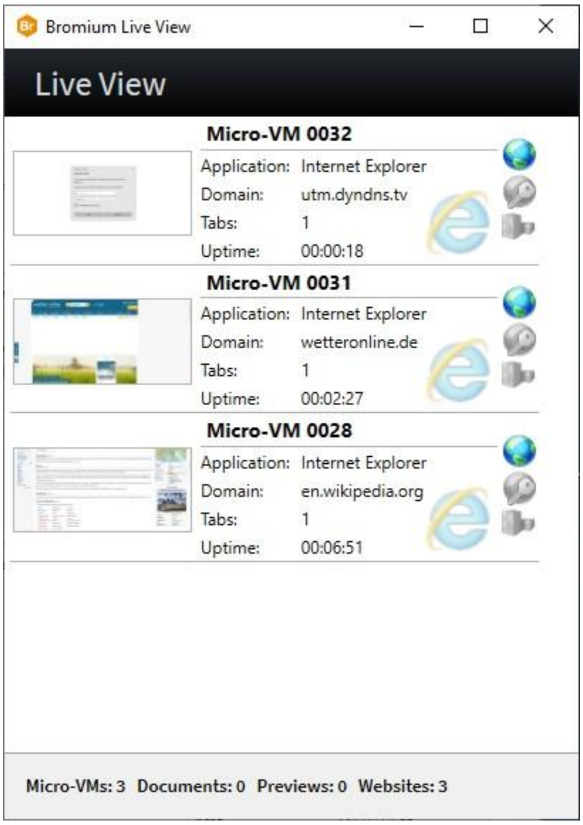
- Windows 10 Pro (OS Build 17763.678)
- Bromium (version:

For testing, an Internet Explorer 11 (IE11) has been started with 3 tabs, pointing to:

- "Dolomites" in Wikipedia (https://en.wikipedia.org/wiki/Dolomites)

- "Wetter aktuell" (https://www.wetteronline.de)
- and the Bromium-Admin GUI (https://utm.dyndns.tv/gui).

This is visible in the Bromium Live View:



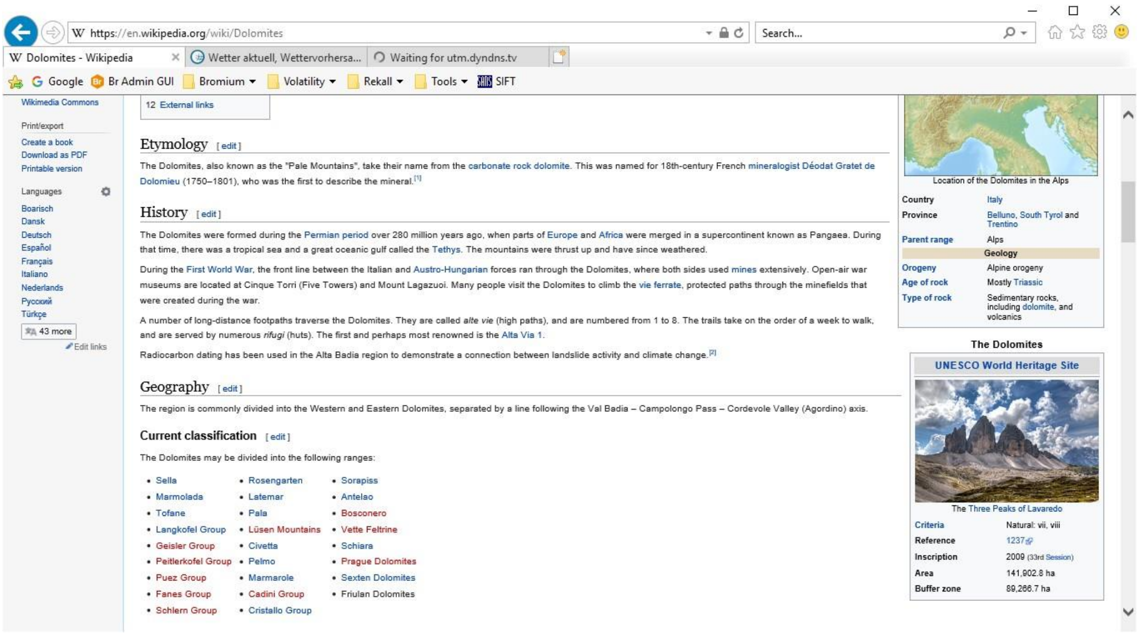
Micro-VM 0032
Application: Internet Explorer
Domain: utm.dyndns.tv
Tabs: 1
Uptime: 00:00:18

Micro-VM 0031
Application: Internet Explorer
Domain: wetteronline.de
Tabs: 1
Uptime: 00:02:27

Micro-VM 0028
Application: Internet Explorer
Domain: en.wikipedia.org
Tabs: 1
Uptime: 00:06:51

Micro-VMs: 3 Documents: 0 Previews: 0 Websites: 3

as well as in IE11:



Etymology [edit]

The Dolomites, also known as the "Pale Mountains", take their name from the carbonate rock dolomite. This was named for 18th-century French mineralogist Déodat Gratet de Dolomieu (1750–1801), who was the first to describe the mineral.[1]

History [edit]

The Dolomites were formed during the Permian period over 280 million years ago, when parts of Europe and Africa were merged in a supercontinent known as Pangaea. During that time, there was a tropical sea and a great oceanic gulf called the Tethys. The mountains were thrust up and have since weathered.

During the First World War, the front line between the Italian and Austro-Hungarian forces ran through the Dolomites, where both sides used mines extensively. Open-air war museums are located at Cinque Torri (Five Towers) and Mount Lagazuoi. Many people visit the Dolomites to climb the vie ferrate, protected paths through the minefields that were created during the war.

A number of long-distance footpaths traverse the Dolomites. They are called *alte vie* (high paths), and are numbered from 1 to 8. The trails take on the order of a week to walk, and are served by numerous *rifugi* (huts). The first and perhaps most renowned is the Alta Via 1.

Radiocarbon dating has been used in the Alta Badia region to demonstrate a connection between landslide activity and climate change.[2]

Geography [edit]

The region is commonly divided into the Western and Eastern Dolomites, separated by a line following the Val Badia – Campolongo Pass – Cordevole Valley (Agordino) axis.

Current classification [edit]

The Dolomites may be divided into the following ranges:

- Sella
- Marmolada
- Tofane
- Langkofel Group
- Geisler Group
- Peitlerkofel Group
- Puez Group
- Fanes Group
- Schlern Group
- Rosengarten
- Latemar
- Pala
- Lüsen Mountains
- Civetta
- Pelmo
- Marmarole
- Cadini Group
- Cristallo Group
- Sorapiss
- Antelao
- Bosconero
- Vette Feltrine
- Schiara
- Prague Dolomites
- Sexten Dolomites
- Friulan Dolomites

Location of the Dolomites in the Alps

Country	Italy
Province	Belluno, South Tyrol and Trentino
Parent range	Alps
Geology	
Orogeny	Alpine orogeny
Age of rock	Mostly Triassic
Type of rock	Sedimentary rocks, including dolomite, and volcanics

The Dolomites

UNESCO World Heritage Site

The Three Peaks of Lavaredo

Criteria	Natural: vii, viii
Reference	1237
Inscription	2009 (33rd Session)
Area	141,902.8 ha
Buffer zone	89,266.7 ha

G.2.2 Testing with Rekall against the Host System

Bromium / Win109 started with IE, running 3 TABS:

- Wikipedia Dolomites
- Wetteronline
- Bromium Admin GUI

=> ram images taken:

```
03.06.2019  13:41     5.000.398.463 memdump_190603_IE-tabs_3x.aff4
```

(generated with affacquire in rekall)

```
03.06.2019  14:23     9.644.802.048 memdump_190603_IE-tabs_3x.raw
```

(generated with imageinfo)

=> 2 TABs closed (only "Wikipedia Dolomites" remains open)

```
03.06.2019  13:43     5.871.504.221 memdump_190603_IE-tabs_1x.aff4
```

(generated with affacquire in rekall)

```
03.06.2019  14:07     9.644.802.048 memdump_190603_IE-tabs_1x.raw
```

(generated with imageinfo)

```
(rekal) C:\DFIR\rekall>rekal -f memdump_190603_IE-tabs_1x.aff4 pstree
No handlers could be found for logger "rekall.plugins.tools"
_EPROCESS                                 ppid  thd_count hnd_count       create_time
---------------------------------------- ------ --------- --------- ------------------------
0xe302c6c09080 csrss.exe (508)            456        12         - 2019-05-20 12:21:38Z
0xe302c6f64080 wininit.exe (600)          456         1         - 2019-05-20 12:21:38Z
. 0xe302c6fa41c0 services.exe (672)         600         9         - 2019-05-20 12:21:38Z
.. 0xe302cab8e540 svchost.exe (68)          672         8         - 2019-05-24 12:19:19Z
.. 0xe302c8c49340 svchost.exe (812)         672         2         - 2019-05-20 12:21:38Z
.. 0xe302c8c52340 svchost.exe (848)         672        21         - 2019-05-20 12:21:38Z
... 0xe302c9053080 SearchUI.exe (2076)      848        33         - 2019-05-24 12:19:21Z
... 0xe302cb1ad080 RuntimeBroker. (2228)    848         5         - 2019-05-24 13:24:56Z
... 0xe302cc7c5340 YourPhone.exe (2376)     848        17         - 2019-05-29 22:22:00Z
... 0xe302cab2a080 RuntimeBroker. (3036)    848        12         - 2019-05-24 12:19:23Z
... 0xe302ca079080 dllhost.exe (3360)       848         5         - 2019-05-24 14:21:42Z
... 0xe302cb8a1080 MicrosoftEdgeC (3412)    848        17         - 2019-05-24 12:20:21Z
... 0xe302cbf08500 dllhost.exe (4292)       848         2         - 2019-05-24 12:19:54Z
... 0xe302cd3a74c0 RuntimeBroker. (4372)    848         6         - 2019-05-24 12:20:00Z
... 0xe302c9d7d2c0 HxCalendarAppI (4708)    848        34         - 2019-05-31 12:39:00Z
... 0xe302c9e0d080 RuntimeBroker. (4848)    848         4         - 2019-05-29 22:22:00Z
... 0xe302cbed0540 ApplicationFra (4924)    848         7         - 2019-05-24 12:19:53Z
... 0xe302cabb2540 browser_broker (4976)    848         2         - 2019-05-24 12:20:20Z
... 0xe302ccee9540 Microsoft.Phot (5088)    848        18         - 2019-05-24 13:12:06Z
... 0xe302cbd74080 smartscreen.ex (6484)    848         9         - 2019-05-24 12:19:34Z
... 0xe302cac7a080 SettingSyncHos (6764)    848         9         - 2019-05-24 12:19:23Z
... 0xe302d398d080 WmiPrvSE.exe (6892)      848        10         - 2019-06-03 11:43:12Z
... 0xe302cab68500 dllhost.exe (7076)       848         6         - 2019-05-20 12:21:53Z
... 0xe302cb606080 WinStore.App.e (7204)    848        24         - 2019-05-24 12:19:59Z
... 0xe302ca057500 RuntimeBroker. (7648)    848         8         - 2019-05-24 13:24:53Z
... 0xe302c3373080 RuntimeBroker. (7876)    848         7          - 2019-05-24 12:19:25Z
... 0xe302c8e66080 RuntimeBroker. (8172)    848         7          - 2019-05-31 12:38:50Z
... 0xe302caf44540 XboxApp.exe (8540)       848        12          - 2019-05-29 22:01:02Z
... 0xe302cd371080 RuntimeBroker. (8676)    848         1          - 2019-05-24 12:20:21Z
.... 0xe302cc12f240 MicrosoftEdgeS (8688)   8676         9          - 2019-05-24 12:20:21Z
... 0xe302cd1a4540 ShellExperienc (8728)    848        19          - 2019-05-24 12:19:21Z
... 0xe302cba4b540 WindowsInterna (9012)    848        33          - 2019-05-24 13:17:11Z
... 0xe302cbd92080 MicrosoftEdge. (9028)    848         0          - 2019-05-20 12:25:27Z
... 0xe302ca084540 MicrosoftEdge. (9472)    848         0          - 2019-05-21 11:52:30Z
... 0xe302cb49e540 HxTsr.exe (10036)        848        17          - 2019-05-31 12:39:00Z
... 0xe302cbef4540 RuntimeBroker. (10116)    848         6          - 2019-05-24 13:12:06Z
... 0xe302cd171540 RuntimeBroker. (10120)    848        21          - 2019-05-24 12:19:23Z
... 0xe302cad950c0 SkypeApp.exe (10348)     848        40          - 2019-05-24 13:24:53Z
... 0xe302cc1172c0 LockApp.exe (10724)      848         8          - 2019-05-24 12:19:25Z
... 0xe302c3690080 SkypeBackgroun (11192)    848         2          - 2019-05-24 13:41:01Z
... 0xe302caab1080 SystemSettings (11260)    848        25          - 2019-05-27 14:31:38Z
... 0xe302c33a42c0 dllhost.exe (11468)      848         8          - 2019-05-24 12:19:22Z
... 0xe302ca506080 backgroundTask (11496)    848         0          - 2019-05-27 14:31:38Z
... 0xe302c3279080 MicrosoftEdge. (12164)    848        35          - 2019-05-24 12:20:20Z
.. 0xe302c8c23540 svchost.exe (932)         672        15          - 2019-05-20 12:21:39Z
.. 0xe302c8d1f340 svchost.exe (972)         672         7          - 2019-05-20 12:21:39Z
.. 0xe302c8f9e3c0 svchost.exe (1096)        672         7          - 2019-05-20 12:21:40Z
.. 0xe302c8fa0400 svchost.exe (1104)        672         3          - 2019-05-20 12:21:40Z
.. 0xe302c8fcf400 svchost.exe (1148)        672         4          - 2019-05-20 12:21:40Z
.. 0xe302c8fd0080 svchost.exe (1156)        672         6          - 2019-05-20 12:21:40Z
.. 0xe302c9052400 svchost.exe (1280)        672         3          - 2019-05-20 12:21:40Z
.. 0xe302c9059080 svchost.exe (1288)        672         3          - 2019-05-20 12:21:40Z
.. 0xe302c90a13c0 svchost.exe (1344)        672         2          - 2019-05-20 12:21:40Z
.. 0xe302c90eb400 svchost.exe (1456)        672         8          - 2019-05-20 12:21:40Z
.. 0xe302c9104080 svchost.exe (1508)        672         3          - 2019-05-20 12:21:40Z
.. 0xe302c9107400 svchost.exe (1520)        672         5          - 2019-05-20 12:21:40Z
.. 0xe302c90d4080 svchost.exe (1540)        672         5          - 2019-05-20 12:21:40Z
.. 0xe302c913c340 svchost.exe (1552)        672         3          - 2019-05-20 12:21:40Z
.. 0xe302c91413c0 svchost.exe (1596)        672         3          - 2019-05-20 12:21:40Z
.. 0xe302c93ad400 svchost.exe (1672)        672         8          - 2019-05-20 12:21:40Z
.. 0xe302c91b93c0 svchost.exe (1700)        672         4          - 2019-05-20 12:21:40Z
.. 0xe302c91d9080 svchost.exe (1752)        672         2          - 2019-05-20 12:21:40Z
.. 0xe302cb3ec300 svchost.exe (1764)        672         4          - 2019-05-24 12:21:16Z
.. 0xe302c9219340 svchost.exe (1800)        672        10          - 2019-05-20 12:21:40Z
... 0xe302cb3e4540 BrConsole.exe (3084)    1800         9          - 2019-05-24 12:19:20Z
.... 0xe302ce771540 BrStatusMonito (8916)   3084        16          - 2019-06-03 11:37:14Z
... 0xe302cac76080 taskhostw.exe (11464)   1800         6          - 2019-05-24 12:19:16Z
.. 0xe302c926f400 svchost.exe (1888)        672         7          - 2019-05-20 12:21:40Z
.. 0xe302c9270340 igfxCUIService (1896)     672         8          - 2019-05-20 12:21:40Z
.. 0xe302cab42080 svchost.exe (1912)        672         3          - 2019-05-24 12:20:23Z
.. 0xe302c92cc380 svchost.exe (1976)        672         4          - 2019-05-20 12:21:40Z
.. 0xe302c92ef400 svchost.exe (2000)        672         7          - 2019-05-20 12:21:40Z
.. 0xe302c93b3340 svchost.exe (2104)        672        13          - 2019-05-20 12:21:40Z
.. 0xe302c9499400 svchost.exe (2220)        672        11          - 2019-05-20 12:21:41Z
.. 0xe302c94942c0 svchost.exe (2236)        672         9          - 2019-05-20 12:21:41Z
... 0xe302cc62b540 sihost.exe (11160)      2236        11          - 2019-05-24 12:19:15Z
.. 0xe302c95b9400 svchost.exe (2368)        672         9          - 2019-05-20 12:21:41Z
.. 0xe302c96da400 svchost.exe (2504)        672        12          - 2019-05-20 12:21:41Z
.. 0xe302c96ba080 svchost.exe (2524)        672         6          - 2019-05-20 12:21:41Z
.. 0xe302c9860400 svchost.exe (2548)        672        15          - 2019-05-20 12:21:41Z
.. 0xe302c9c17340 spoolsv.exe (2584)        672         7          - 2019-05-20 12:21:42Z
.. 0xe302c98c4380 svchost.exe (2620)        672         7          - 2019-05-20 12:21:41Z
.. 0xe302c98c6380 svchost.exe (2648)        672         7          - 2019-05-20 12:21:41Z
.. 0xe302c98dd400 svchost.exe (2740)        672         4          - 2019-05-20 12:21:41Z
.. 0xe302c999c380 svchost.exe (2868)        672         9          - 2019-05-20 12:21:41Z
.. 0xe302c9a4f400 svchost.exe (2956)        672        10          - 2019-05-20 12:21:41Z
.. 0xe302c9a5f380 svchost.exe (3000)        672         7          - 2019-05-20 12:21:42Z
.. 0xe302c9c21080 svchost.exe (3116)        672        13          - 2019-05-20 12:21:42Z
.. 0xe302c9c303c0 svchost.exe (3168)        672         5         - 2019-05-20 12:21:42Z
.. 0xe302c9cb4340 svchost.exe (3312)        672         3         - 2019-05-20 12:21:42Z
.. 0xe302c9e6f3c0 armsvc.exe (3520)         672         2         - 2019-05-20 12:21:42Z
.. 0xe302c9e733c0 BrRemoteMgmtSv (3544)     672         9         - 2019-05-20 12:21:42Z
.. 0xe302c9e753c0 BrService.exe (3568)      672        23         - 2019-05-20 12:21:42Z
... 0xe302cbcc1080 BrHostSvr.exe (2580)    3568        98         - 2019-05-24 12:19:15Z
.... 0xe302cb840080 Br-uxendm.exe (8928)   2580        25         - 2019-06-03 11:30:12Z
..... 0xe302c8da7080 conhost.exe (10948)   8928         4         - 2019-06-03 11:30:12Z
.... 0xe302c9fc5080 Br-uxendm.exe (12456)   2580        16         - 2019-06-03 11:42:02Z
..... 0xe302cc00c080 conhost.exe (1652)   12456         3         - 2019-06-03 11:42:02Z
... 0xe302cb16d540 Br-uxendm.exe (8312)    3568         0         - 2019-05-30 16:27:23Z
... 0xe302ca49d080 Br-uxendm.exe (8940)    3568         0         - 2019-05-20 12:22:05Z
... 0xe302caa4e080 Br-uxendm.exe (11808)   3568         0         - 2019-05-24 12:19:32Z
... 0xe302cc7130c0 Br-uxendm.exe (11968)   3568         0         - 2019-05-21 11:37:01Z
.. 0xe302c9e77340 svchost.exe (3576)        672         5         - 2019-05-20 12:21:42Z
.. 0xe302c9e793c0 svchost.exe (3592)        672         3         - 2019-05-20 12:21:42Z
.. 0xe302c9eba340 svchost.exe (3644)        672        14         - 2019-05-20 12:21:42Z
.. 0xe302c9ebc400 svchost.exe (3652)        672         7         - 2019-05-20 12:21:42Z
.. 0xe302c9ebe400 svchost.exe (3660)        672        17         - 2019-05-20 12:21:42Z
.. 0xe302c9ed7340 svchost.exe (3704)        672         5         - 2019-05-20 12:21:43Z
.. 0xe302c9ed5300 ibtsiva.exe (3724)        672         1         - 2019-05-20 12:21:43Z
.. 0xe302c9ef13c0 svchost.exe (3788)        672         2         - 2019-05-20 12:21:43Z
.. 0xe302c9f1e380 svchost.exe (3824)        672         3         - 2019-05-20 12:21:43Z
.. 0xe302c9f1c3c0 SynTPEnhServic (3852)     672         3         - 2019-05-20 12:21:43Z
... 0xe302cbaeb080 SynTPEnh.exe (1164)     3852         0         - 2019-05-21 11:36:49Z
... 0xe302ca7e7080 SynTPEnh.exe (1772)     3852         0         - 2019-05-23 19:13:23Z
... 0xe302ca77e4c0 SynTPEnh.exe (5640)     3852         0         - 2019-05-20 12:21:50Z
... 0xe302cc133080 SynTPEnh.exe (7244)     3852         0         - 2019-05-23 19:13:28Z
... 0xe302cb6de540 SynTPEnh.exe (10600)    3852         9         - 2019-05-24 12:19:15Z
.... 0xe302cc0dd080 SynTPEnh.exe (9712)   10600         0         - 2019-05-24 12:19:16Z
..... 0xe302ca5c2080 SynTPHelper.ex (7632)   9712         1         - 2019-05-24 12:19:17Z
... 0xe302cbcc4080 SynTPEnh.exe (11680)    3852         0         - 2019-05-20 20:22:40Z
.. 0xe302c9f460c0 svchost.exe (3872)        672         6         - 2019-05-20 12:21:43Z
.. 0xe302c9ef0080 svchost.exe (3880)        672         8         - 2019-05-20 12:21:43Z
.. 0xe302c9f43440 MsMpEng.exe (3904)        672        29         - 2019-05-20 12:21:43Z
.. 0xe302ca07b400 svchost.exe (4072)        672         3         - 2019-05-20 12:21:43Z
.. 0xe302c656f0c0 svchost.exe (4324)        672        12         - 2019-05-20 12:21:43Z
.. 0xe302ca584400 svchost.exe (4616)        672         2         - 2019-05-20 12:21:46Z
.. 0xe302cab660c0 svchost.exe (5116)        672         4         - 2019-05-20 12:21:51Z
... 0xe302cb1bd540 ctfmon.exe (5188)       5116        10         - 2019-05-24 12:19:20Z
.. 0xe302c947e2c0 svchost.exe (5436)        672        11         - 2019-05-20 12:21:50Z
.. 0xe302cb474240 svchost.exe (5484)        672         7         - 2019-05-20 12:21:58Z
.. 0xe302ccc89080 svchost.exe (5732)        672         3         - 2019-05-22 17:58:43Z
.. 0xe302cabb9080 svchost.exe (6060)        672         9         - 2019-05-20 12:21:51Z
.. 0xe302caaaa380 svchost.exe (6076)        672         3         - 2019-05-20 12:21:50Z
.. 0xe302cacc6500 SearchIndexer. (6096)     672        47         - 2019-06-02 12:56:47Z
... 0xe302cc7b6080 SearchProtocol (4128)   6096         9         - 2019-06-03 11:42:06Z
... 0xe302ce2d8080 SearchFilterHo (12712)   6096         5         - 2019-06-03 11:41:52Z
... 0xe302d398e540 SearchProtocol (12760)   6096         6         - 2019-06-03 11:41:51Z
.. 0xe302caaa6340 svchost.exe (6140)        672         8         - 2019-05-20 12:21:50Z
.. 0xe302cabc1400 svchost.exe (6176)        672         6         - 2019-05-20 12:21:51Z
.. 0xe302cb3e8540 svchost.exe (6328)        672         5         - 2019-05-24 13:25:08Z
.. 0xe302caba7540 OfficeClickToR (6352)     672        20         - 2019-06-02 12:55:55Z
... 0xe302ca441080 AppVShNotify.e (1324)   6352         1         - 2019-06-02 12:56:11Z
... 0xe302cae64080 AppVShNotify.e (1824)   6352         1         - 2019-06-02 12:56:11Z
.. 0xe302cfb13340 svchost.exe (6772)        672         3         - 2019-06-03 11:39:40Z
.. 0xe302cac07540 svchost.exe (6864)        672        11         - 2019-05-20 12:21:53Z
.. 0xe302ca66e080 svchost.exe (7116)        672         7         - 2019-05-24 12:19:16Z
.. 0xe302cad49100 SgrmBroker.exe (7400)     672         5         - 2019-05-20 12:23:44Z
.. 0xe302caf7a240 svchost.exe (7472)        672         8         - 2019-05-20 12:21:56Z
.. 0xe302cc7c6080 svchost.exe (8040)        672         6         - 2019-05-20 12:23:44Z
.. 0xe302cb46d4c0 NisSrv.exe (8144)         672         7         - 2019-05-20 12:21:58Z
.. 0xe302cb1a8540 svchost.exe (8768)        672         0         - 2019-05-20 12:22:01Z
.. 0xe302cb813080 WUDFHost.exe (8984)       672        10         - 2019-05-24 13:08:01Z
.. 0xe302cb1ee400 SecurityHealth (9300)     672        11         - 2019-05-20 12:22:09Z
.. 0xe302cbfed080 svchost.exe (9720)        672         6         - 2019-05-20 12:22:15Z
.. 0xe302cc56b080 svchost.exe (10104)       672         5         - 2019-05-20 12:41:21Z
.. 0xe302cbbb6080 svchost.exe (10112)       672        10         - 2019-05-20 12:22:06Z
.. 0xe302cb1bf540 svchost.exe (10524)       672         4         - 2019-05-24 13:07:56Z
.. 0xe302c6bf1540 svchost.exe (10556)       672        17         - 2019-05-24 12:19:15Z
.. 0xe302cbd91080 BemAgent.exe (11320)      672         6         - 2019-05-20 12:23:43Z
. 0xe302c6fe5180 lsass.exe (684)            600         8         - 2019-05-20 12:21:38Z
. 0xe302c8c50240 fontdrvhost.ex (840)       600         5         - 2019-05-20 12:21:38Z
0xe302c3284040 System (4)                   0       202        - 2019-05-20 12:21:36Z
. 0xe302c32eb080 Registry (96)                4         4         - 2019-05-20 12:21:33Z
. 0xe302c641c240 smss.exe (360)               4         2         - 2019-05-20 12:21:36Z
. 0xe302c91d6080 MemCompression (1724)        4        82         - 2019-05-20 12:21:40Z
0xe302cc0f2080 winlogon.exe (2908)       5968         5        - 2019-05-23 19:13:22Z
. 0xe302c363f2c0 LogonUI.exe (2380)        2908         0         - 2019-05-24 12:42:21Z
. 0xe302cceee540 LogonUI.exe (2952)        2908         0         - 2019-05-28 14:54:29Z
. 0xe302cb3ee540 LogonUI.exe (9948)        2908         0         - 2019-05-27 16:28:59Z
. 0xe302cd1a5080 dwm.exe (9992)            2908        14         - 2019-05-23 19:13:23Z
. 0xe302cad9f080 fontdrvhost.ex (10572)    2908         5         - 2019-05-23 19:13:23Z
. 0xe302caec1540 userinit.exe (11612)      2908         0         - 2019-05-24 12:19:17Z
.. 0xe302cba29540 explorer.exe (8456)     11612        97         - 2019-05-24 12:19:18Z
... 0xe302cc736080 iexplore.exe (224)      8456        20         - 2019-06-03 11:30:25Z
.... 0xe302cb955080 iexplore.exe (10124)    224        16         - 2019-06-03 11:30:25Z
... 0xe302ccde5080 cmd.exe (1028)          8456         4         - 2019-05-24 13:43:49Z
.... 0xe302c9899540 conhost.exe (9500)     1028         5         - 2019-05-24 13:43:49Z
.... 0xe302cfbab540 rekal.exe (10180)      1028         1         - 2019-06-03 11:42:26Z
..... 0xe302ce7ac540 python.exe (12868)   10180       107         - 2019-06-03 11:42:26Z
... 0xe302cb89d080 ScreenHunter7P (2792)   8456         3         - 2019-05-24 12:19:43Z
... 0xe302c6b73080 SecurityHealth (9876)   8456         1         - 2019-05-24 12:19:35Z
... 0xe302caae3080 cmd.exe (11220)         8456         3         - 2019-05-27 14:20:46Z
.... 0xe302cbbc0080 conhost.exe (7372)    11220         4         - 2019-05-27 14:20:46Z
... 0xe302c60d8080 i_view64.exe (12264)    8456         6         - 2019-06-03 11:43:06Z
... 0xe302cfb550c0 i_view64.exe (12324)    8456         3         - 2019-06-03 11:39:41Z
. 0xe302cbbc7080 LogonUI.exe (11748)       2908         0         - 2019-05-23 19:13:23Z
0xe302cb3e5080 csrss.exe (6880)          5968        13        - 2019-05-23 19:13:22Z
0xe302caef3080 OneDrive.exe (8468)       8780        25        - 2019-05-30 12:22:48Z
0xe302cb9e8080 WzPreloader.ex (8428)     7732         5        - 2019-05-24 14:20:40Z
0xe302c333c540 igfxTray.exe (424)       10652         2        - 2019-05-24 12:19:19Z
0xe302ca318540 igfxEM.exe (1436)        10652         6        - 2019-05-24 12:19:18Z
0xe302cb24f540 igfxHK.exe (6264)        10652         2        - 2019-05-24 12:19:19Z

(rekal) C:\DFIR\rekall>
```

```
(rekal) C:\DFIR\rekall>rekal -f memdump_190603_IE-tabs_1x.aff4
imageinfo
No handlers could be found for logger "rekall.plugins.tools"
key           value
-------------------- -----
Kernel DTB           0x1aa002
NT Build             17763.rs5_release.180914-1434
NT Build Ex          17763.1.amd64fre.rs5_release.180914-1434
Signed Drivers       -
Time (UTC)           2019-06-03 11:43:13Z
Time (Local)         2019-06-03 13:43:13+0200
Sec Since Boot       1207300.53125
```

```
NtSystemRoot         C:\WINDOWS
```

```
(rekal) C:\DFIR\rekall>rekal -f memdump_190603_IE-tabs_1x.aff4
imagecopy --output-image="memdump_190603_IE-tabs_1x.raw"
No handlers could be found for logger "rekall.plugins.tools"
```

```
(rekal) C:\DFIR\rekall>rekal -f memdump_190603_IE-tabs_3x.aff4
imagecopy --output-image="memdump_190603_IE-tabs_3x.raw"
No handlers could be found for logger "rekall.plugins.tools"
```

```
(rekal) C:\DFIR\rekall>dir memdump_1906*
Volume in drive C has no label.
Volume Serial Number is 1CFF-E879

Directory of C:\DFIR\rekall

03.06.2019  13:43     5.871.504.221 memdump_190603_IE-tabs_1x.aff4
03.06.2019  14:07     9.644.802.048 memdump_190603_IE-tabs_1x.raw
03.06.2019  13:41     5.000.398.463 memdump_190603_IE-tabs_3x.aff4
03.06.2019  14:23     9.644.802.048 memdump_190603_IE-tabs_3x.raw
       4 File(s) 30.161.506.780 bytes
       0 Dir(s)  29.770.809.344 bytes free
```

```
(rekal) C:\DFIR\rekall>rekal -f memdump_190603_IE-tabs_1x.aff4 psxview
_EPROCESS            name          pid  PsActiveProcessHead CSRSS PspCidTable Sessions Handles PSScan
Thrdproc
-------------- -------------------- ----- ------------------- ----- ----------- -------- ------- ------ -----
---
0xe302c3284040 System                   4 True                False True        False    False   False  True
0xe302cab8e540 svchost.exe             68 True                True  True        True     True    False  False
0xe302c32eb080 Registry                96 True                False True        False    True    False  True
0xe302cc736080 iexplore.exe           224 True                True  True        True     True    False  True
0xe302c641c240 smss.exe               360 True                False True        False    True    False  True
0xe302c333c540 igfxTray.exe           424 True                True  True        True     True    False  False
0xe302c6c09080 csrss.exe              508 True                False True        True     True    False  True
0xe302c6f64080 wininit.exe            600 True                True  True        True     True    False  True
0xe302c6fa41c0 services.exe           672 True                True  True        True     True    False  True
0xe302c6fe5180 lsass.exe              684 True                True  True        True     True    False  False
0xe302c8c49340 svchost.exe            812 True                True  True        True     True    False  True
0xe302c8c50240 fontdrvhost.ex         840 True                True  True        True     True    False  True
0xe302c8c52340 svchost.exe            848 True                True  True        True     True    False  True
0xe302c8c23540 svchost.exe            932 True                True  True        True     True    False  True
0xe302c8d1f340 svchost.exe            972 True                True  True        True     True    False  True
0xe302ccde5080 cmd.exe               1028 True                True  True        True     True    False  False
0xe302c8f9e3c0 svchost.exe           1096 True                True  True        True     True    False  False
0xe302c8fa0400 svchost.exe           1104 True                True  True        True     True    False  False
0xe302c8fcf400 svchost.exe           1148 True                True  True        True     True    False  True
0xe302c8fd0080 svchost.exe           1156 True                True  True        True     True    False  True
```

```
0xe302cbaeb080 SynTPEnh.exe          1164 True                 False True        True     False   False  False
0xe302c9052400 svchost.exe           1280 True                 True  True        True     True    False  False
0xe302c9059080 svchost.exe           1288 True                 True  True        True     True    False  True
0xe302ca441080 AppVShNotify.e        1324 True                 True  True        True     True    False  False
0xe302c90a13c0 svchost.exe           1344 True                 True  True        True     True    False  False
0xe302ca318540 igfxEM.exe            1436 True                 True  True        True     True    False  False
0xe302c90eb400 svchost.exe           1456 True                 True  True        True     True    False  True
0xe302c9104080 svchost.exe           1508 True                 True  True        True     True    False  True
0xe302c9107400 svchost.exe           1520 True                 True  True        True     True    False  True
0xe302c90d4080 svchost.exe           1540 True                 True  True        True     True    False  True
0xe302c913c340 svchost.exe           1552 True                 True  True        True     True    False  False
0xe302c91413c0 svchost.exe           1596 True                 True  True        True     True    False  False
0xe302cc00c080 conhost.exe           1652 True                 True  True        True     True    False  True
0xe302c93ad400 svchost.exe           1672 True                 True  True        True     True    False  False
0xe302c91b93c0 svchost.exe           1700 True                 True  True        True     True    False  False
0xe302c91d6080 MemCompression        1724 True                 False True        False    True    False  True
0xe302c91d9080 svchost.exe           1752 True                 True  True        True     True    False  False
0xe302cb3ec300 svchost.exe           1764 True                 True  True        True     True    False  False
0xe302ca7e7080 SynTPEnh.exe          1772 True                 False True        True     False   False  False
0xe302c9219340 svchost.exe           1800 True                 True  True        True     True    False  True
0xe302cae64080 AppVShNotify.e        1824 True                 True  True        True     True    False  False
0xe302c926f400 svchost.exe           1888 True                 True  True        True     True    False  True
0xe302c9270340 igfxCUIService        1896 True                 True  True        True     True    False  False
0xe302cab42080 svchost.exe           1912 True                 True  True        True     True    False  False
0xe302c92cc380 svchost.exe           1976 True                 True  True        True     True    False  False
0xe302c92ef400 svchost.exe           2000 True                 True  True        True     True    False  True
0xe302c9053080 SearchUI.exe          2076 True                 True  True        True     True    False  True
0xe302c93b3340 svchost.exe           2104 True                 True  True        True     True    False  True
0xe302c9499400 svchost.exe           2220 True                 True  True        True     True    False  True
0xe302cb1ad080 RuntimeBroker.        2228 True                 True  True        True     True    False  False
0xe302c94942c0 svchost.exe           2236 True                 True  True        True     True    False  True
0xe302c95b9400 svchost.exe           2368 True                 True  True        True     True    False  False
0xe302cc7c5340 YourPhone.exe         2376 True                 True  True        True     True    False  False
0xe302c363f2c0 LogonUI.exe           2380 True                 False True        True     False   False  False
0xe302c96da400 svchost.exe           2504 True                 True  True        True     True    False  True
0xe302c96ba080 svchost.exe           2524 True                 True  True        True     True    False  False
[...]
```

```
(rekal) C:\DFIR\rekall>rekal -f memdump_190603_IE-tabs_3x.aff4 pstree
No handlers could be found for logger "rekall.plugins.tools"
_EPROCESS                                  ppid thd_count hnd_count        create_time
---------------------------------------- ------ --------- --------- ------------------------
  0xe302c6c09080 csrss.exe (508)            456        12         - 2019-05-20 12:21:38Z
  0xe302c6f64080 wininit.exe (600)          456         1         - 2019-05-20 12:21:38Z
. 0xe302c6fa41c0 services.exe (672)         600         8         - 2019-05-20 12:21:38Z
.. 0xe302cab8e540 svchost.exe (68)          672         6         - 2019-05-24 12:19:19Z
.. 0xe302c8c49340 svchost.exe (812)         672         2         - 2019-05-20 12:21:38Z
.. 0xe302c8c52340 svchost.exe (848)         672        23         - 2019-05-20 12:21:38Z
... 0xe302c9053080 SearchUI.exe (2076)      848        33         - 2019-05-24 12:19:21Z
... 0xe302cb1ad080 RuntimeBroker. (2228)    848         7         - 2019-05-24 13:24:56Z
... 0xe302cc7c5340 YourPhone.exe (2376)     848        17         - 2019-05-29 22:22:00Z
... 0xe302cab2a080 RuntimeBroker. (3036)    848        12         - 2019-05-24 12:19:23Z
... 0xe302ca079080 dllhost.exe (3360)       848         5         - 2019-05-24 14:21:42Z
... 0xe302cb8a1080 MicrosoftEdgeC (3412)    848        17         - 2019-05-24 12:20:21Z
... 0xe302cbf08500 dllhost.exe (4292)       848         2         - 2019-05-24 12:19:54Z
... 0xe302cd3a74c0 RuntimeBroker. (4372)    848         6         - 2019-05-24 12:20:00Z
... 0xe302c9d7d2c0 HxCalendarAppI (4708)    848        34         - 2019-05-31 12:39:00Z
... 0xe302c9e0d080 RuntimeBroker. (4848)    848         4         - 2019-05-29 22:22:00Z
... 0xe302cbed0540 ApplicationFra (4924)    848         7         - 2019-05-24 12:19:53Z
... 0xe302cabb2540 browser_broker (4976)    848         2         - 2019-05-24 12:20:20Z
... 0xe302ccee9540 Microsoft.Phot (5088)    848         18         - 2019-05-24 13:12:06Z
... 0xe302d3952080 backgroundTask (5148)    848         13         - 2019-06-03 11:39:40Z
... 0xe302cbd74080 smartscreen.ex (6484)    848          9         - 2019-05-24 12:19:34Z
... 0xe302cac7a080 SettingSyncHos (6764)    848          9         - 2019-05-24 12:19:23Z
... 0xe302cab68500 dllhost.exe (7076)       848          4         - 2019-05-20 12:21:53Z
... 0xe302cb606080 WinStore.App.e (7204)    848         24         - 2019-05-24 12:19:59Z
... 0xe302ca057500 RuntimeBroker. (7648)    848          9         - 2019-05-24 13:24:53Z
... 0xe302c3373080 RuntimeBroker. (7876)    848          7         - 2019-05-24 12:19:25Z
... 0xe302c8e66080 RuntimeBroker. (8172)    848          7         - 2019-05-31 12:38:50Z
... 0xe302caf44540 XboxApp.exe (8540)       848         12         - 2019-05-29 22:01:02Z
... 0xe302cd371080 RuntimeBroker. (8676)    848          1         - 2019-05-24 12:20:21Z
.... 0xe302cc12f240 MicrosoftEdgeS (8688)   8676          9         - 2019-05-24 12:20:21Z
... 0xe302cd1a4540 ShellExperienc (8728)    848         19         - 2019-05-24 12:19:21Z
... 0xe302cba4b540 WindowsInterna (9012)    848         33         - 2019-05-24 13:17:11Z
... 0xe302cbd92080 MicrosoftEdge. (9028)    848          0         - 2019-05-20 12:25:27Z
... 0xe302ca084540 MicrosoftEdge. (9472)    848          0         - 2019-05-21 11:52:30Z
... 0xe302cb49e540 HxTsr.exe (10036)        848         17         - 2019-05-31 12:39:00Z
... 0xe302cbef4540 RuntimeBroker. (10116)    848          6         - 2019-05-24 13:12:06Z
... 0xe302cd171540 RuntimeBroker. (10120)    848         22         - 2019-05-24 12:19:23Z
... 0xe302cad950c0 SkypeApp.exe (10348)     848         40         - 2019-05-24 13:24:53Z
... 0xe302cc1172c0 LockApp.exe (10724)      848          8         - 2019-05-24 12:19:25Z
... 0xe302c3690080 SkypeBackgroun (11192)    848          2         - 2019-05-24 13:41:01Z
... 0xe302caab1080 SystemSettings (11260)    848         25         - 2019-05-27 14:31:38Z
... 0xe302c33a42c0 dllhost.exe (11468)      848          8         - 2019-05-24 12:19:22Z
... 0xe302ca506080 backgroundTask (11496)    848          0         - 2019-05-27 14:31:38Z
... 0xe302c3279080 MicrosoftEdge. (12164)    848         35         - 2019-05-24 12:20:20Z
... 0xe302cfb334c0 RuntimeBroker. (12292)    848          7         - 2019-06-03 11:39:41Z
.. 0xe302c8c23540 svchost.exe (932)         672         15         - 2019-05-20 12:21:39Z
.. 0xe302c8d1f340 svchost.exe (972)         672          7         - 2019-05-20 12:21:39Z
.. 0xe302c8f9e3c0 svchost.exe (1096)        672          7         - 2019-05-20 12:21:40Z
.. 0xe302c8fa0400 svchost.exe (1104)        672          3         - 2019-05-20 12:21:40Z
.. 0xe302c8fcf400 svchost.exe (1148)        672          4         - 2019-05-20 12:21:40Z
.. 0xe302c8fd0080 svchost.exe (1156)        672          6         - 2019-05-20 12:21:40Z
.. 0xe302c9052400 svchost.exe (1280)        672          3         - 2019-05-20 12:21:40Z
.. 0xe302c9059080 svchost.exe (1288)        672          3         - 2019-05-20 12:21:40Z
.. 0xe302c90a13c0 svchost.exe (1344)        672          2         - 2019-05-20 12:21:40Z
.. 0xe302c90eb400 svchost.exe (1456)        672          9         - 2019-05-20 12:21:40Z
.. 0xe302c9104080 svchost.exe (1508)        672          3         - 2019-05-20 12:21:40Z
.. 0xe302c9107400 svchost.exe (1520)        672          5         - 2019-05-20 12:21:40Z
.. 0xe302c90d4080 svchost.exe (1540)        672          5         - 2019-05-20 12:21:40Z
.. 0xe302c913c340 svchost.exe (1552)        672          3         - 2019-05-20 12:21:40Z
.. 0xe302c91413c0 svchost.exe (1596)        672          3         - 2019-05-20 12:21:40Z
.. 0xe302c93ad400 svchost.exe (1672)        672          8         - 2019-05-20 12:21:40Z
.. 0xe302c91b93c0 svchost.exe (1700)        672          4         - 2019-05-20 12:21:40Z
.. 0xe302c91d9080 svchost.exe (1752)        672          2         - 2019-05-20 12:21:40Z
.. 0xe302cb3ec300 svchost.exe (1764)        672          4         - 2019-05-24 12:21:16Z
.. 0xe302c9219340 svchost.exe (1800)        672         10         - 2019-05-20 12:21:40Z
... 0xe302cb3e4540 BrConsole.exe (3084)    1800          9         - 2019-05-24 12:19:20Z
.... 0xe302ce771540 BrStatusMonito (8916)   3084         17         - 2019-06-03 11:37:14Z
... 0xe302cac76080 taskhostw.exe (11464)   1800          6         - 2019-05-24 12:19:16Z
.. 0xe302c926f400 svchost.exe (1888)        672          7         - 2019-05-20 12:21:40Z
.. 0xe302c9270340 igfxCUIService (1896)     672          8         - 2019-05-20 12:21:40Z
.. 0xe302cab42080 svchost.exe (1912)        672          3         - 2019-05-24 12:20:23Z
.. 0xe302c92cc380 svchost.exe (1976)        672          4         - 2019-05-20 12:21:40Z
.. 0xe302c92ef400 svchost.exe (2000)        672          7         - 2019-05-20 12:21:40Z
.. 0xe302c93b3340 svchost.exe (2104)        672         12         - 2019-05-20 12:21:40Z
.. 0xe302c9499400 svchost.exe (2220)        672         11         - 2019-05-20 12:21:41Z
.. 0xe302c94942c0 svchost.exe (2236)        672          9         - 2019-05-20 12:21:41Z
... 0xe302cc62b540 sihost.exe (11160)      2236         11         - 2019-05-24 12:19:15Z
.. 0xe302c95b9400 svchost.exe (2368)        672          9         - 2019-05-20 12:21:41Z
.. 0xe302c96da400 svchost.exe (2504)        672         12         - 2019-05-20 12:21:41Z
.. 0xe302c96ba080 svchost.exe (2524)        672          6         - 2019-05-20 12:21:41Z
.. 0xe302c9860400 svchost.exe (2548)        672         15         - 2019-05-20 12:21:41Z
.. 0xe302c9c17340 spoolsv.exe (2584)        672        7         - 2019-05-20 12:21:42Z
.. 0xe302c98c4380 svchost.exe (2620)        672        5         - 2019-05-20 12:21:41Z
.. 0xe302c98c6380 svchost.exe (2648)        672        7         - 2019-05-20 12:21:41Z
.. 0xe302c98dd400 svchost.exe (2740)        672        5         - 2019-05-20 12:21:41Z
.. 0xe302c999c380 svchost.exe (2868)        672       10         - 2019-05-20 12:21:41Z
.. 0xe302c9a4f400 svchost.exe (2956)        672       10         - 2019-05-20 12:21:41Z
.. 0xe302c9a5f380 svchost.exe (3000)        672        7         - 2019-05-20 12:21:42Z
.. 0xe302c9c21080 svchost.exe (3116)        672       13         - 2019-05-20 12:21:42Z
.. 0xe302c9c303c0 svchost.exe (3168)        672        5         - 2019-05-20 12:21:42Z
.. 0xe302c9cb4340 svchost.exe (3312)        672        3         - 2019-05-20 12:21:42Z
.. 0xe302c9e6f3c0 armsvc.exe (3520)         672        2         - 2019-05-20 12:21:42Z
.. 0xe302c9e733c0 BrRemoteMgmtSv (3544)     672        9         - 2019-05-20 12:21:42Z
.. 0xe302c9e753c0 BrService.exe (3568)      672       22         - 2019-05-20 12:21:42Z
... 0xe302cbcc1080 BrHostSvr.exe (2580)    3568      105         - 2019-05-24 12:19:15Z
.... 0xe302cf668080 Br-uxendm.exe (7012)   2580       21         - 2019-06-03 11:34:56Z
..... 0xe302cc00c080 conhost.exe (1004)    7012        4         - 2019-06-03 11:34:56Z
.... 0xe302ccab5080 Br-uxendm.exe (8212)   2580       24         - 2019-06-03 11:34:34Z
..... 0xe302cb518080 conhost.exe (6288)    8212        4         - 2019-06-03 11:34:34Z
.... 0xe302cb840080 Br-uxendm.exe (8928)   2580       25         - 2019-06-03 11:30:12Z
..... 0xe302c8da7080 conhost.exe (10948)   8928        4         - 2019-06-03 11:30:12Z
... 0xe302cb16d540 Br-uxendm.exe (8312)    3568        0         - 2019-05-30 16:27:23Z
... 0xe302ca49d080 Br-uxendm.exe (8940)    3568        0         - 2019-05-20 12:22:05Z
... 0xe302caa4e080 Br-uxendm.exe (11808)   3568        0         - 2019-05-24 12:19:32Z
... 0xe302cc7130c0 Br-uxendm.exe (11968)   3568        0         - 2019-05-21 11:37:01Z
.. 0xe302c9e77340 svchost.exe (3576)        672        5         - 2019-05-20 12:21:42Z
.. 0xe302c9e793c0 svchost.exe (3592)        672        3         - 2019-05-20 12:21:42Z
.. 0xe302c9eba340 svchost.exe (3644)        672       13         - 2019-05-20 12:21:42Z
.. 0xe302c9ebc400 svchost.exe (3652)        672        7         - 2019-05-20 12:21:42Z
.. 0xe302c9ebe400 svchost.exe (3660)        672       17         - 2019-05-20 12:21:42Z
.. 0xe302c9ed7340 svchost.exe (3704)        672        5         - 2019-05-20 12:21:43Z
.. 0xe302c9ed5300 ibtsiva.exe (3724)        672        1         - 2019-05-20 12:21:43Z
.. 0xe302c9ef13c0 svchost.exe (3788)        672        2         - 2019-05-20 12:21:43Z
.. 0xe302c9f1e380 svchost.exe (3824)        672        3         - 2019-05-20 12:21:43Z
.. 0xe302c9f1c3c0 SynTPEnhServic (3852)     672        3         - 2019-05-20 12:21:43Z
... 0xe302cbaeb080 SynTPEnh.exe (1164)     3852        0         - 2019-05-21 11:36:49Z
... 0xe302ca7e7080 SynTPEnh.exe (1772)     3852        0         - 2019-05-23 19:13:23Z
... 0xe302ca77e4c0 SynTPEnh.exe (5640)     3852        0         - 2019-05-20 12:21:50Z
... 0xe302cc133080 SynTPEnh.exe (7244)     3852        0         - 2019-05-23 19:13:28Z
... 0xe302cb6de540 SynTPEnh.exe (10600)    3852        9         - 2019-05-24 12:19:15Z
.... 0xe302cc0dd080 SynTPEnh.exe (9712)   10600        0         - 2019-05-24 12:19:16Z
..... 0xe302ca5c2080 SynTPHelper.ex (7632)   9712        1         - 2019-05-24 12:19:17Z
... 0xe302cbcc4080 SynTPEnh.exe (11680)    3852        0         - 2019-05-20 20:22:40Z
.. 0xe302c9f460c0 svchost.exe (3872)        672        6         - 2019-05-20 12:21:43Z
.. 0xe302c9ef0080 svchost.exe (3880)        672        7         - 2019-05-20 12:21:43Z
.. 0xe302c9f43440 MsMpEng.exe (3904)        672       29         - 2019-05-20 12:21:43Z
.. 0xe302ca07b400 svchost.exe (4072)        672        3         - 2019-05-20 12:21:43Z
.. 0xe302c656f0c0 svchost.exe (4324)        672       12         - 2019-05-20 12:21:43Z
.. 0xe302ca584400 svchost.exe (4616)        672        2         - 2019-05-20 12:21:46Z
.. 0xe302cab660c0 svchost.exe (5116)        672        4         - 2019-05-20 12:21:51Z
... 0xe302cb1bd540 ctfmon.exe (5188)       5116       10         - 2019-05-24 12:19:20Z
.. 0xe302c947e2c0 svchost.exe (5436)        672       10         - 2019-05-20 12:21:50Z
.. 0xe302cb474240 svchost.exe (5484)        672        8         - 2019-05-20 12:21:58Z
.. 0xe302ccc89080 svchost.exe (5732)        672        3         - 2019-05-22 17:58:43Z
.. 0xe302cabb9080 svchost.exe (6060)        672        9         - 2019-05-20 12:21:51Z
.. 0xe302caaaa380 svchost.exe (6076)        672        3         - 2019-05-20 12:21:50Z
.. 0xe302cacc6500 SearchIndexer. (6096)     672       46         - 2019-06-02 12:56:47Z
... 0xe302d39ae480 SearchFilterHo (3680)   6096        4         - 2019-06-03 11:39:39Z
... 0xe302ca4aa280 SearchProtocol (7192)   6096        7         - 2019-06-03 11:36:16Z
... 0xe302ce2d8080 SearchProtocol (7484)   6096        0         - 2019-06-03 11:39:37Z
.. 0xe302caaa6340 svchost.exe (6140)        672        8         - 2019-05-20 12:21:50Z
.. 0xe302cabc1400 svchost.exe (6176)        672        6         - 2019-05-20 12:21:51Z
.. 0xe302cb3e8540 svchost.exe (6328)        672        5         - 2019-05-24 13:25:08Z
.. 0xe302caba7540 OfficeClickToR (6352)     672        20         - 2019-06-02 12:55:55Z
... 0xe302ca441080 AppVShNotify.e (1324)   6352         1         - 2019-06-02 12:56:11Z
... 0xe302cae64080 AppVShNotify.e (1824)   6352         1         - 2019-06-02 12:56:11Z
.. 0xe302cfb13340 svchost.exe (6772)        672         3         - 2019-06-03 11:39:40Z
.. 0xe302cac07540 svchost.exe (6864)        672         9         - 2019-05-20 12:21:53Z
.. 0xe302ca66e080 svchost.exe (7116)        672         9         - 2019-05-24 12:19:16Z
.. 0xe302cad49100 SgrmBroker.exe (7400)     672         5         - 2019-05-20 12:23:44Z
.. 0xe302caf7a240 svchost.exe (7472)        672         8         - 2019-05-20 12:21:56Z
.. 0xe302cc7c6080 svchost.exe (8040)        672         5         - 2019-05-20 12:23:44Z
.. 0xe302cb46d4c0 NisSrv.exe (8144)         672         7         - 2019-05-20 12:21:58Z
.. 0xe302cb1a8540 svchost.exe (8768)        672         0         - 2019-05-20 12:22:01Z
.. 0xe302cb813080 WUDFHost.exe (8984)       672        10         - 2019-05-24 13:08:01Z
.. 0xe302cb1ee400 SecurityHealth (9300)     672        11         - 2019-05-20 12:22:09Z
.. 0xe302cbfed080 svchost.exe (9720)        672         6         - 2019-05-20 12:22:15Z
.. 0xe302cc56b080 svchost.exe (10104)       672         5         - 2019-05-20 12:41:21Z
.. 0xe302cbbb6080 svchost.exe (10112)       672        10         - 2019-05-20 12:22:06Z
.. 0xe302cb1bf540 svchost.exe (10524)       672         4         - 2019-05-24 13:07:56Z
.. 0xe302c6bf1540 svchost.exe (10556)       672        15         - 2019-05-24 12:19:15Z
.. 0xe302cbd91080 BemAgent.exe (11320)      672         6         - 2019-05-20 12:23:43Z
. 0xe302c6fe5180 lsass.exe (684)            600         9         - 2019-05-20 12:21:38Z
. 0xe302c8c50240 fontdrvhost.ex (840)       600         5         - 2019-05-20 12:21:38Z
  0xe302c3284040 System (4)                   0       208         - 2019-05-20 12:21:36Z
. 0xe302c32eb080 Registry (96)                4         4         - 2019-05-20 12:21:33Z
. 0xe302c641c240 smss.exe (360)               4         2         - 2019-05-20 12:21:36Z
. 0xe302c91d6080 MemCompression (1724)        4        82         - 2019-05-20 12:21:40Z
  0xe302cc0f2080 winlogon.exe (2908)       5968         5         - 2019-05-23 19:13:22Z
. 0xe302c363f2c0 LogonUI.exe (2380)        2908         0         - 2019-05-24 12:42:21Z
. 0xe302cceee540 LogonUI.exe (2952)        2908         0         - 2019-05-28 14:54:29Z
. 0xe302cb3ee540 LogonUI.exe (9948)        2908         0         - 2019-05-27 16:28:59Z
. 0xe302cd1a5080 dwm.exe (9992)            2908        14         - 2019-05-23 19:13:23Z
. 0xe302cad9f080 fontdrvhost.ex (10572)    2908         5         - 2019-05-23 19:13:23Z
. 0xe302caec1540 userinit.exe (11612)      2908         0         - 2019-05-24 12:19:17Z
.. 0xe302cba29540 explorer.exe (8456)     11612        91         - 2019-05-24 12:19:18Z
... 0xe302cc736080 iexplore.exe (224)      8456        20         - 2019-06-03 11:30:25Z
.... 0xe302cb5600c0 iexplore.exe (9660)     224        21         - 2019-06-03 11:36:56Z
.... 0xe302cb955080 iexplore.exe (10124)    224        16         - 2019-06-03 11:30:25Z
.... 0xe302cca762c0 iexplore.exe (11384)    224        21         - 2019-06-03 11:34:24Z
... 0xe302ccde5080 cmd.exe (1028)          8456         4         - 2019-05-24 13:43:49Z
.... 0xe302c9899540 conhost.exe (9500)     1028         5         - 2019-05-24 13:43:49Z
.... 0xe302cb460080 rekal.exe (12876)      1028         1         - 2019-06-03 11:39:56Z
..... 0xe302c9fc5080 python.exe (12892)   12876       109         - 2019-06-03 11:39:56Z
... 0xe302cb89d080 ScreenHunter7P (2792)   8456         2         - 2019-05-24 12:19:43Z
... 0xe302c6b73080 SecurityHealth (9876)   8456         1         - 2019-05-24 12:19:35Z
... 0xe302caae3080 cmd.exe (11220)         8456         3         - 2019-05-27 14:20:46Z
.... 0xe302cbbc0080 conhost.exe (7372)    11220         4         - 2019-05-27 14:20:46Z
... 0xe302cfb550c0 i_view64.exe (12324)    8456         3         - 2019-06-03 11:39:41Z
. 0xe302cbbc7080 LogonUI.exe (11748)       2908         0         - 2019-05-23 19:13:23Z
  0xe302cb3e5080 csrss.exe (6880)          5968        12         - 2019-05-23 19:13:22Z
  0xe302cb9e8080 WzPreloader.ex (8428)     7732         5         - 2019-05-24 14:20:40Z
  0xe302c333c540 igfxTray.exe (424)       10652         2         - 2019-05-24 12:19:19Z
  0xe302ca318540 igfxEM.exe (1436)        10652         6         - 2019-05-24 12:19:18Z
  0xe302cb24f540 igfxHK.exe (6264)        10652         2         - 2019-05-24 12:19:19Z
  0xe302caef3080 OneDrive.exe (8468)       8780        26         - 2019-05-30 12:22:48Z

(rekal) C:\DFIR\rekall>
```

```
(rekal) C:\DFIR\rekall>rekal -f memdump_190603_IE-tabs_3x.aff4 psxview
```

```
  _EPROCESS            name          pid  PsActiveProcessHead CSRSS PspCidTable Sessions Handles PSScan
Thrdproc
-------------- -------------------- ----- ------------------- ----- ----------- -------- ------- ------ -----
---
0xe302c3284040 System                   4 True                False True        False    False   False  True
```

```
0xe302cab8e540 svchost.exe             68 True                True  True        True     True    False  False
0xe302c32eb080 Registry                96 True                False True        False    True    False  True
0xe302cc736080 iexplore.exe           224 True                True  True        True     True    False  True
0xe302c641c240 smss.exe               360 True                False True        False    True    False  True
0xe302c333c540 igfxTray.exe           424 True                True  True        True     True    False  False
0xe302c6c09080 csrss.exe              508 True                False True        True     True    False  True
0xe302c6f64080 wininit.exe            600 True                True  True        True     True    False  True
0xe302c6fa41c0 services.exe           672 True                True  True        True     True    False  True
0xe302c6fe5180 lsass.exe              684 True                True  True        True     True    False  False
0xe302c8c49340 svchost.exe            812 True                True  True        True     True    False  True
0xe302c8c50240 fontdrvhost.ex         840 True                True  True        True     True    False  True
0xe302c8c52340 svchost.exe            848 True                True  True        True     True    False  True
0xe302c8c23540 svchost.exe            932 True                True  True        True     True    False  True
0xe302c8d1f340 svchost.exe            972 True                True  True        True     True    False  True
0xe302cc00c080 conhost.exe           1004 True                True  True        True     True    False  False
0xe302ccde5080 cmd.exe               1028 True                True  True        True     True    False  False
0xe302c8f9e3c0 svchost.exe           1096 True                True  True        True     True    False  False
0xe302c8fa0400 svchost.exe           1104 True                True  True        True     True    False  False
0xe302c8fcf400 svchost.exe           1148 True                True  True        True     True    False  True
0xe302c8fd0080 svchost.exe           1156 True                True  True        True     True    False  True
0xe302cbaeb080 SynTPEnh.exe          1164 True                False True        True     False   False  False
0xe302c9052400 svchost.exe           1280 True                True  True        True     True    False  False
0xe302c9059080 svchost.exe           1288 True                True  True        True     True    False  True
0xe302ca441080 AppVShNotify.e        1324 True                True  True        True     True    False  False
0xe302c90a13c0 svchost.exe           1344 True                True  True        True     True    False  False
0xe302ca318540 igfxEM.exe            1436 True                True  True        True     True    False  False
0xe302c90eb400 svchost.exe           1456 True                True  True        True     True    False  True
0xe302c9104080 svchost.exe           1508 True                True  True        True     True    False  True
0xe302c9107400 svchost.exe           1520 True                True  True        True     True    False  True
0xe302c90d4080 svchost.exe           1540 True                True  True        True     True    False  True
0xe302c913c340 svchost.exe           1552 True                True  True        True     True    False  False
0xe302c91413c0 svchost.exe           1596 True                True  True        True     True    False  False
0xe302c93ad400 svchost.exe           1672 True                True  True        True     True    False  False
0xe302c91b93c0 svchost.exe           1700 True                True  True        True     True    False  False
0xe302c91d6080 MemCompression        1724 True                False True        False    True    False  True
0xe302c91d9080 svchost.exe           1752 True                True  True        True     True    False  False
[...]
```

```
(rekal) C:\DFIR\rekall> rekal -f memdump_190603_IE-tabs_3x.aff4 vmscan
No handlers could be found for logger "rekall.plugins.tools"
Description                                  Type          Valid   EPT
------------------------------------ -------------------- -------- ---
  VM [2 vCORE(s), AMD64]                               VM     True 0x2537901E
  VM [2 vCORE(s), AMD64]                               VM     True 0x1BC2F401E
  VM [2 vCORE(s), AMD64]                               VM     True 0x1CE71401E
```

```
(rekal) C:\DFIR\rekall>rekal -f memdump_190603_IE-tabs_1x.aff4 vmscan
No handlers could be found for logger "rekall.plugins.tools"
Description                                  Type          Valid   EPT
------------------------------------ -------------------- -------- ---
  VM [2 vCORE(s), AMD64]                               VM     True 0x2537901E
  VM [2 vCORE(s), AMD64]                               VM     True 0x8AD4901E
```

```
(rekal) C:\DFIR\rekall>rekal -f memdump_190603_IE-tabs_3x.aff4 netscan
    offset     protocol      local_addr               remote_addr                state        pid      owner
```

```
created
-------------- -------- -------------------- ------------------------------ ---------------- ----- ----------
---- -------
0xe302c64c8e10 TCPv4    0.0.0.0:135          0.0.0.0:0                      LISTENING          932
svchost.exe    2019-05-20 12:21:39Z
0xe302c64c8e10 TCPv6    :::135               :::0                           LISTENING          932
svchost.exe    2019-05-20 12:21:39Z
0xe302c8cd9310 TCPv4    0.0.0.0:5040         0.0.0.0:0                      LISTENING         6864
svchost.exe    2019-05-24 13:07:56Z
0xe302c90b3ae0 TCPv4    0.0.0.0:49665        0.0.0.0:0                      LISTENING         1456
svchost.exe    2019-05-20 12:21:40Z
0xe302c90b4170 TCPv4    0.0.0.0:49666        0.0.0.0:0                      LISTENING         1800
svchost.exe    2019-05-20 12:21:40Z
0xe302c90b4560 TCPv4    0.0.0.0:49666        0.0.0.0:0                      LISTENING         1800
svchost.exe    2019-05-20 12:21:40Z
0xe302c90b4560 TCPv6    :::49666             :::0                           LISTENING         1800
svchost.exe    2019-05-20 12:21:40Z
0xe302c90b4e90 TCPv4    0.0.0.0:49665        0.0.0.0:0                      LISTENING         1456
svchost.exe    2019-05-20 12:21:40Z
0xe302c90b4e90 TCPv6    :::49665             :::0                           LISTENING         1456
svchost.exe    2019-05-20 12:21:40Z
0xe302c9c4f910 TCPv4    0.0.0.0:445          0.0.0.0:0                      LISTENING            4 System
2019-05-20 12:21:43Z
0xe302c9c4f910 TCPv6    :::445               :::0                           LISTENING            4 System
2019-05-20 12:21:43Z
0xe302cf609c70 TCPv4    0.0.0.0:7680         0.0.0.0:0                      LISTENING         5436
svchost.exe    2019-05-24 13:07:59Z
0xe302cf609c70 TCPv6    :::7680              :::0                           LISTENING         5436
svchost.exe    2019-05-24 13:07:59Z
0xe302c3388c10 TCPv4    10.0.100.112:62790   52.114.132.73:443              CLOSED               0
-
0xe302c6bf9320 TCPv4    10.0.100.112:62999   93.184.220.29:80               ESTABLISHED          0
-
0xe302ca44b410 TCPv6    2003:a:144f:3c00:ddd 2a02:26f0:fb:1b6::3114:80      CLOSED               0
-
                        d:59e5:235b:2b5d:535
                        58
0xe302ca6d4410 TCPv6    2003:a:144f:3c00:ddd 2a02:26f0:e9:388::4106:443     CLOSED               0
-
                        d:59e5:235b:2b5d:535
                        52
0xe302caeaaaa0 TCPv4    10.0.100.112:60368   13.107.3.128:443               CLOSED               0
-
0xe302cbef5410 TCPv4    10.0.100.112:53554   13.107.42.12:443               CLOSED               0
-
0xe302ccee88a0 TCPv6    2003:a:144f:3c00:ddd 2a02:26f0:fb:1b6::3114:80      CLOSED               0
-
                        d:59e5:235b:2b5d:535
                        61
0xe302cdeec010 TCPv4    10.0.100.112:62791   52.114.132.73:443              CLOSED               0
-
0xe302c36526f0 UDPv4    0.0.0.0:0            *:*                                              2740
svchost.exe    2019-05-21 11:48:07Z
0xe302c36526f0 UDPv6    :::0                 *:*                                              2740
svchost.exe    2019-05-21 11:48:07Z
0xe302c3654520 UDPv4    0.0.0.0:0            *:*                                              2740
svchost.exe    2019-05-21 11:48:07Z
0xe302c3655240 UDPv4    0.0.0.0:58513        *:*                                              2504
svchost.exe    2019-05-21 11:36:53Z
0xe302c3655240 UDPv6    :::58513             *:*                                              2504
svchost.exe    2019-05-21 11:36:53Z
0xe302c3ffc2f0 UDPv4    0.0.0.0:0            *:*                                              3544
BrRemoteMgmtSv 2019-06-03 11:40:57Z
```

```
0xe302c3ffc2f0 UDPv6    :::0                 *:*                                               3544
BrRemoteMgmtSv 2019-06-03 11:40:57Z
0xe302c64c3590 UDPv4    127.0.0.1:54887      *:*                                               3704
svchost.exe    2019-05-20 12:21:43Z
0xe302c9c4dae0 UDPv4    0.0.0.0:0            *:*                                               2504
svchost.exe    2019-06-02 12:07:42Z
0xe302c9c4dae0 UDPv6    :::0                 *:*                                               2504
svchost.exe    2019-06-02 12:07:42Z
0xe302cf604090 UDPv4    10.0.100.112:138     *:*                                                  4 System
2019-05-24 13:07:58Z
0xe302cf6051a0 UDPv4    127.0.0.1:1900       *:*                                               9720
svchost.exe    2019-05-24 13:07:56Z
0xe302cf605590 UDPv6    ::1:60316            *:*                                               9720
svchost.exe    2019-05-24 13:07:56Z
0xe302cf606400 UDPv4    10.0.100.112:60317   *:*                                               9720
svchost.exe    2019-05-24 13:07:56Z
0xe302cf606a90 UDPv6    fe80::e830:a191:2f6a *:*                                               9720
svchost.exe    2019-05-24 13:07:56Z
                        :4698:1900
0xe302cf606be0 UDPv6    ::1:1900             *:*                                               9720
svchost.exe    2019-05-24 13:07:56Z
0xe302cf6073c0 UDPv4    10.0.100.112:1900    *:*                                               9720
svchost.exe    2019-05-24 13:07:56Z
0xe302cf608620 UDPv4    127.0.0.1:60318      *:*                                               9720
svchost.exe    2019-05-24 13:07:56Z
0xe302cf60d0f0 UDPv4    0.0.0.0:5355         *:*                                               2504
svchost.exe    2019-06-03 11:20:25Z
0xe302cf60d0f0 UDPv6    :::5355              *:*                                               2504
svchost.exe    2019-06-03 11:20:25Z
0xe302cf614470 UDPv4    0.0.0.0:5355         *:*                                               2504
svchost.exe    2019-06-03 11:20:25Z
0xe302cf617b90 UDPv4    0.0.0.0:0            *:*                                               7012 Br-
uxendm.exe  2019-06-03 11:37:00Z


(rekal) C:\DFIR\rekall>rekal -f memdump_190603_IE-tabs_1x.aff4 netscan
    offset     protocol      local_addr                remote_addr                 state         pid      owner
created
-------------- -------- -------------------- ------------------------------ ---------------- ----- ----------
---- -------
0xe302c64c8e10 TCPv4    0.0.0.0:135          0.0.0.0:0                       LISTENING          932
svchost.exe    2019-05-20 12:21:39Z
0xe302c64c8e10 TCPv6    :::135               :::0                            LISTENING          932
svchost.exe    2019-05-20 12:21:39Z
0xe302c8cd9310 TCPv4    0.0.0.0:5040         0.0.0.0:0                       LISTENING         6864
svchost.exe    2019-05-24 13:07:56Z
0xe302c90b3ae0 TCPv4    0.0.0.0:49665        0.0.0.0:0                       LISTENING         1456
svchost.exe    2019-05-20 12:21:40Z
0xe302c90b4170 TCPv4    0.0.0.0:49666        0.0.0.0:0                       LISTENING         1800
svchost.exe    2019-05-20 12:21:40Z
0xe302c90b4560 TCPv4    0.0.0.0:49666        0.0.0.0:0                       LISTENING         1800
svchost.exe    2019-05-20 12:21:40Z
0xe302c90b4560 TCPv6    :::49666             :::0                            LISTENING         1800
svchost.exe    2019-05-20 12:21:40Z
0xe302c90b4e90 TCPv4    0.0.0.0:49665        0.0.0.0:0                       LISTENING         1456
svchost.exe    2019-05-20 12:21:40Z
0xe302c90b4e90 TCPv6    :::49665             :::0                            LISTENING         1456
svchost.exe    2019-05-20 12:21:40Z
0xe302c9c4f910 TCPv4    0.0.0.0:445          0.0.0.0:0                       LISTENING            4 System
2019-05-20 12:21:43Z
0xe302c9c4f910 TCPv6    :::445               :::0                            LISTENING            4 System
```

```
2019-05-20 12:21:43Z
0xe302cf609c70 TCPv4    0.0.0.0:7680         0.0.0.0:0                      LISTENING         5436
svchost.exe    2019-05-24 13:07:59Z
0xe302cf609c70 TCPv6    :::7680              :::0                           LISTENING         5436
svchost.exe    2019-05-24 13:07:59Z
0xe302c3388c10 TCPv4    10.0.100.112:62790   52.114.132.73:443              CLOSED               0
-
0xe302ca44b410 TCPv6    2003:a:144f:3c00:ddd 2a02:26f0:fb:1b6::3114:80      CLOSED               0
-
                        d:59e5:235b:2b5d:535
                        58
0xe302ca6d4410 TCPv6    2003:a:144f:3c00:ddd 2a02:26f0:e9:388::4106:443     CLOSED               0
-
                        d:59e5:235b:2b5d:535
                        52
0xe302caeaaaa0 TCPv4    10.0.100.112:60368   13.107.3.128:443               CLOSED               0
-
0xe302cbef5410 TCPv4    10.0.100.112:53554   13.107.42.12:443               CLOSED               0
-
0xe302ccee88a0 TCPv6    2003:a:144f:3c00:ddd 2a02:26f0:fb:1b6::3114:80      CLOSED               0
-
                        d:59e5:235b:2b5d:535
                        61
0xe302cdeec010 TCPv4    10.0.100.112:62791   52.114.132.73:443              CLOSED               0
-
0xe302c36526f0 UDPv4    0.0.0.0:0            *:*                                              2740
svchost.exe    2019-05-21 11:48:07Z
0xe302c36526f0 UDPv6    :::0                 *:*                                              2740
svchost.exe    2019-05-21 11:48:07Z
0xe302c3654520 UDPv4    0.0.0.0:0            *:*                                              2740
svchost.exe    2019-05-21 11:48:07Z
0xe302c3655240 UDPv4    0.0.0.0:58513        *:*                                              2504
svchost.exe    2019-05-21 11:36:53Z
0xe302c3655240 UDPv6    :::58513             *:*                                              2504
svchost.exe    2019-05-21 11:36:53Z
0xe302c64c3590 UDPv4    127.0.0.1:54887      *:*                                              3704
svchost.exe    2019-05-20 12:21:43Z
0xe302c9c4dae0 UDPv4    0.0.0.0:0            *:*                                              2504
svchost.exe    2019-06-02 12:07:42Z
0xe302c9c4dae0 UDPv6    :::0                 *:*                                              2504
svchost.exe    2019-06-02 12:07:42Z
0xe302cf604090 UDPv4    10.0.100.112:138     *:*                                                 4 System
2019-05-24 13:07:58Z
0xe302cf6051a0 UDPv4    127.0.0.1:1900       *:*                                              9720
svchost.exe    2019-05-24 13:07:56Z
0xe302cf605590 UDPv6    ::1:60316            *:*                                              9720
svchost.exe    2019-05-24 13:07:56Z
0xe302cf606400 UDPv4    10.0.100.112:60317   *:*                                              9720
svchost.exe    2019-05-24 13:07:56Z
0xe302cf606a90 UDPv6    fe80::e830:a191:2f6a *:*                                              9720
svchost.exe    2019-05-24 13:07:56Z
                        :4698:1900
0xe302cf606be0 UDPv6    ::1:1900             *:*                                              9720
svchost.exe    2019-05-24 13:07:56Z
0xe302cf6073c0 UDPv4    10.0.100.112:1900    *:*                                              9720
svchost.exe    2019-05-24 13:07:56Z
0xe302cf608620 UDPv4    127.0.0.1:60318      *:*                                              9720
svchost.exe    2019-05-24 13:07:56Z
0xe302cf60d0f0 UDPv4    0.0.0.0:5355         *:*                                              2504
svchost.exe    2019-06-03 11:20:25Z
0xe302cf60d0f0 UDPv6    :::5355              *:*                                              2504
svchost.exe    2019-06-03 11:20:25Z
0xe302cf614470 UDPv4    0.0.0.0:5355         *:*                                              2504
```

```
svchost.exe    2019-06-03 11:20:25Z
```

```
(rekal) C:\DFIR\rekall>rekal -f memdump_190603_IE-tabs_3x.aff4
filescan
```

```
a     offset     ptr_no hnd_no access     Owner              name          pid  path
- -------------- ------ ------ ------ -------------- -------------------- ----- ----
  0xe302c3b37910     32      0 RW-rwd -              -                        - \Device\HarddiskVolume1\$Mft
  0xe302c3b37ed0     20      0 RW-rwd -              -                        -
\Device\HarddiskVolume1\$BitMap
  0xe302c3c07350     27      0 RW-rwd -              -                        - C:\$Extend\$UsnJrnl:$J:$DATA
  0xe302c3ec87f0     33      0 RW-rwd -              -                        - C:\$Directory
  0xe302c3ec8c40  32535      1 RW-r-d 0xe302c3284040 System                   4
C:\ProgramData\Bromium\vSentry\Logs\BrCow.etl.011
  0xe302c3ec8db0  32792      1 RWDr-d 0xe302c3284040 System                   4
C:\Windows\System32\LogFiles\WMI\RtBackup\EtwRTUBPM.etl
  0xe302c3ec9090     32      0 RW-rwd -              -                        -
C:\$Secure:$SDH:$INDEX_ALLOCATION
  0xe302c3ec97c0  32769      1 RW-rw- 0xe302c3284040 System                   4
\Device\clfs\SystemRoot\System32\Config\TxR\{6160866a-78aa-11e9-9b64-806e6f6e6963}.TM
  0xe302c3ec9930  32769      1 RWDr-d 0xe302c3284040 System                   4
C:\Windows\System32\LogFiles\WMI\RtBackup\EtwRTDiagLog.etl
  0xe302c3ec9aa0     33      0 RW-rwd -              -                        - C:\$Directory
  0xe302c3eca620     17      0 RW-rwd -              -                        - C:\$Directory
  0xe302c3eca900  32779      1 RWDr-d 0xe302c3284040 System                   4
C:\Windows\System32\LogFiles\WMI\RtBackup\EtwRTEventLog-System.etl
  0xe302c3ecabe0     24      0 RW-rwd -              -                        - C:\$Directory
  0xe302c3ecaec0     14      0 R--r-d -              -                        -
C:\Windows\System32\drivers\ksthunk.sys
  0xe302c3ecbd20  32781      1 R--r-d 0xe302c3284040 System                   4
C:\Windows\System32\drivers\de-DE\USBXHCI.SYS.mui
  0xe302c3ecc170     32      0 RW-rwd -              -                        - C:\$Directory
  0xe302c3ffe0a0      3      0 RW-rwd -              -                        -
C:\$Extend\$RmMetadata\$Repair:$Verify:$DATA
  0xe302c3ffe210  32763      1 R----- 0xe302c3284040 System                   4 C:\System Volume
Information\{3808876b-c176-4e48-b7ae-04046e6cc752}
  0xe302c3ffe380      3      0 RW-r-- -              -                        -
C:\$Extend\$RmMetadata\$Repair
  0xe302c3ffe4f0  32769      1 RWDrwd 0xe302c3284040 System                   4
\Device\clfs\Device\HarddiskVolume2\$Extend\$RmMetadata\$TxfLog\$TxfLog
  0xe302c3ffe660  32769      1 RW-r-- 0xe302c3284040 System                   4
C:\$Extend\$RmMetadata\$TxfLog\$TxfLog.blf
  0xe302c3ffed90     32      0 RW-rwd -              -                        - C:\$Directory
  0xe302c3fff070     26      0 RW-rwd -              -                        - C:\$Mft::$BITMAP
  0xe302c3fff1e0      3      0 RW-rwd -              -                        -
C:\$Extend:$I30:$INDEX_ALLOCATION
  0xe302c60fc660     17      0 RW-rwd -              -                        - C:\$Directory
  0xe302c60fc7d0  32769      1 RW-r-- 0xe302c3284040 System                   4
C:\$Extend\$RmMetadata\$TxfLog\$TxfLogContainer00000000000000000024
  0xe302c60fcab0  32769      1 RW-r-- 0xe302c3284040 System                   4 \Device\clfsTxfLog
  0xe302c60fcd90      3      0 RW-rwd -              -                        -
C:\$Extend\$RmMetadata\$TxfLog\$Tops
  0xe302c60fd070  32768      1 RW-rwd 0xe302c3284040 System                   4
C:\$Extend\$RmMetadata\$Txf:$I30:$INDEX_ALLOCATION
  0xe302c60fd1e0      4      0 RW-rwd -              -                        -
C:\$Extend\$RmMetadata\$TxfLog\$Tops:$T:$DATA
  0xe302c60fd350  32769      1 RW-r-- 0xe302c3284040 System                   4
C:\$Extend\$RmMetadata\$TxfLog\$TxfLogContainer00000000000000000012
  0xe302c60fd7a0  32768      1 RW-rwd 0xe302c3284040 System                   4
\Device\clfs\Device\HarddiskVolume2\$Extend\$RmMetadata\$TxfLog\$TxfLog
  0xe302c60fd910  32769      1 RWDrwd 0xe302c3284040 System                   4
\Device\clfs\Device\HarddiskVolume2\$Extend\$RmMetadata\$TxfLog\$TxfLog
```

```
  0xe302c60fda80     33      0 RW-rwd -              -                        - C:\$Directory
  0xe302c60fdbf0  32769      1 RW-rw- 0xe302c3284040 System                   4 \Device\clfsKtmLog
  0xe302c60fdd60      1      0 RW-rwd -              -                        - C:\:$I30:$INDEX_ALLOCATION
  0xe302c60fded0     29      0 RW-rwd -              -                        -
C:\$Extend\$Reparse:$R:$INDEX_ALLOCATION
  0xe302c6147940     32      0 RW-rwd -              -                        - C:\$Directory
  0xe302c61481e0     33      0 RW-rwd -              -                        - C:\$Directory
 [...]
```

```
(rekal) C:\DFIR\rekall>rekal -f memdump_190603_IE-tabs_1x.aff4
filescan
```

```
a     offset     ptr_no hnd_no access     Owner              name          pid  path
- -------------- ------ ------ ------ -------------- -------------------- ----- ----
  0xe302c3b37910     32      0 RW-rwd -              -                        - \Device\HarddiskVolume1\$Mft
  0xe302c3b37ed0     20      0 RW-rwd -              -                        -
\Device\HarddiskVolume1\$BitMap
  0xe302c3c07350     29      0 RW-rwd -              -                        - C:\$Extend\$UsnJrnl:$J:$DATA
  0xe302c3ec87f0     33      0 RW-rwd -              -                        - C:\$Directory
  0xe302c3ec8c40  32535      1 RW-r-d 0xe302c3284040 System                   4
C:\ProgramData\Bromium\vSentry\Logs\BrCow.etl.011
  0xe302c3ec8db0  32792      1 RWDr-d 0xe302c3284040 System                   4
C:\Windows\System32\LogFiles\WMI\RtBackup\EtwRTUBPM.etl
  0xe302c3ec9090     32      0 RW-rwd -              -                        -
C:\$Secure:$SDH:$INDEX_ALLOCATION
  0xe302c3ec97c0  32769      1 RW-rw- 0xe302c3284040 System                   4
\Device\clfs\SystemRoot\System32\Config\TxR\{6160866a-78aa-11e9-9b64-806e6f6e6963}.TM
  0xe302c3ec9930  32769      1 RWDr-d 0xe302c3284040 System                   4
C:\Windows\System32\LogFiles\WMI\RtBackup\EtwRTDiagLog.etl
  0xe302c3ec9aa0     33      0 RW-rwd -              -                        - C:\$Directory
  0xe302c3eca620     17      0 RW-rwd -              -                        - C:\$Directory
  0xe302c3eca900  32779      1 RWDr-d 0xe302c3284040 System                   4
C:\Windows\System32\LogFiles\WMI\RtBackup\EtwRTEventLog-System.etl
  0xe302c3ecabe0     24      0 RW-rwd -              -                        - C:\$Directory
  0xe302c3ecaec0     14      0 R--r-d -              -                        -
C:\Windows\System32\drivers\ksthunk.sys
  0xe302c3ecbd20  32781      1 R--r-d 0xe302c3284040 System                   4
C:\Windows\System32\drivers\de-DE\USBXHCI.SYS.mui
  0xe302c3ecc170     32      0 RW-rwd -              -                        - C:\$Directory
  0xe302c3ffe0a0      3      0 RW-rwd -              -                        -
C:\$Extend\$RmMetadata\$Repair:$Verify:$DATA
  0xe302c3ffe210  32763      1 R----- 0xe302c3284040 System                   4 C:\System Volume
Information\{3808876b-c176-4e48-b7ae-04046e6cc752}
  0xe302c3ffe380      3      0 RW-r-- -              -                        -
C:\$Extend\$RmMetadata\$Repair
  0xe302c3ffe4f0  32769      1 RWDrwd 0xe302c3284040 System                   4
\Device\clfs\Device\HarddiskVolume2\$Extend\$RmMetadata\$TxfLog\$TxfLog
  0xe302c3ffe660  32769      1 RW-r-- 0xe302c3284040 System                   4
C:\$Extend\$RmMetadata\$TxfLog\$TxfLog.blf
  0xe302c3ffed90     32      0 RW-rwd -              -                        - C:\$Directory
  0xe302c3fff070     26      0 RW-rwd -              -                        - C:\$Mft::$BITMAP
  0xe302c3fff1e0      3      0 RW-rwd -              -                        -
C:\$Extend:$I30:$INDEX_ALLOCATION
  0xe302c60fc660     17      0 RW-rwd -              -                        - C:\$Directory
  0xe302c60fc7d0  32769      1 RW-r-- 0xe302c3284040 System                   4
C:\$Extend\$RmMetadata\$TxfLog\$TxfLogContainer00000000000000000024
  0xe302c60fcab0  32769      1 RW-r-- 0xe302c3284040 System                   4 \Device\clfsTxfLog
  0xe302c60fcd90      3      0 RW-rwd -              -                        -
C:\$Extend\$RmMetadata\$TxfLog\$Tops
  0xe302c60fd070  32768      1 RW-rwd 0xe302c3284040 System                   4
C:\$Extend\$RmMetadata\$Txf:$I30:$INDEX_ALLOCATION
  0xe302c60fd1e0      4      0 RW-rwd -              -                        -
C:\$Extend\$RmMetadata\$TxfLog\$Tops:$T:$DATA
```

```
  0xe302c60fd350  32769      1 RW-r-- 0xe302c3284040 System                  4
C:\$Extend\$RmMetadata\$TxfLog\$TxfLogContainer00000000000000000012
  0xe302c60fd7a0  32768      1 RW-rwd 0xe302c3284040 System                  4
\Device\clfs\Device\HarddiskVolume2\$Extend\$RmMetadata\$TxfLog\$TxfLog
  0xe302c60fd910  32769      1 RWDrwd 0xe302c3284040 System                  4
\Device\clfs\Device\HarddiskVolume2\$Extend\$RmMetadata\$TxfLog\$TxfLog
  0xe302c60fda80     33      0 RW-rwd -              -                       - C:\$Directory
  0xe302c60fdbf0  32769      1 RW-rw- 0xe302c3284040 System                  4 \Device\clfsKtmLog
  0xe302c60fdd60      1      0 RW-rwd -              -                       - C:\:$I30:$INDEX_ALLOCATION
  0xe302c60fded0     29      0 RW-rwd -              -                       -
C:\$Extend\$Reparse:$R:$INDEX_ALLOCATION
[...]
```

```
#####################################################################
Summary/Extract (Bromium VM related processes):
#####################################################################
```

```
(rekal) C:\DFIR\rekall>rekal -f memdump_190603_IE-tabs_3x.aff4 pstree
_EPROCESS                                  ppid  thd_count hnd_count       create_time
------------------------------------------ ------ --------- --------- ------------------------

.. 0xe302c9e753c0 BrService.exe (3568)      672        22         - 2019-05-20 12:21:42Z
... 0xe302cbcc1080 BrHostSvr.exe (2580)    3568       105         - 2019-05-24 12:19:15Z

.... 0xe302cb840080 Br-uxendm.exe (8928)   2580        25         - 2019-06-03 11:30:12Z
.... 0xe302cf668080 Br-uxendm.exe (7012)   2580        21         - 2019-06-03 11:34:56Z
.... 0xe302ccab5080 Br-uxendm.exe (8212)   2580        24         - 2019-06-03 11:34:34Z

... 0xe302cb16d540 Br-uxendm.exe (8312)    3568         0         - 2019-05-30 16:27:23Z
... 0xe302ca49d080 Br-uxendm.exe (8940)    3568         0         - 2019-05-20 12:22:05Z
... 0xe302caa4e080 Br-uxendm.exe (11808)   3568         0         - 2019-05-24 12:19:32Z
... 0xe302cc7130c0 Br-uxendm.exe (11968)   3568         0         - 2019-05-21 11:37:01Z

... 0xe302cc736080 iexplore.exe (224)      8456        20         - 2019-06-03 11:30:25Z
.... 0xe302cb955080 iexplore.exe (10124)    224        16         - 2019-06-03 11:30:25Z
.... 0xe302cca762c0 iexplore.exe (11384)    224        21         - 2019-06-03 11:34:24Z
.... 0xe302cb5600c0 iexplore.exe (9660)     224        21         - 2019-06-03 11:36:56Z
```

```
(rekal) C:\DFIR\rekall>rekal -f memdump_190603_IE-tabs_1x.aff4 pstree
No handlers could be found for logger "rekall.plugins.tools"
_EPROCESS                                  ppid  thd_count hnd_count       create_time
------------------------------------------ ------ --------- --------- ------------------------
.. 0xe302c9e753c0 BrService.exe (3568)      672        23         - 2019-05-20 12:21:42Z
... 0xe302cbcc1080 BrHostSvr.exe (2580)    3568        98         - 2019-05-24 12:19:15Z

.... 0xe302cb840080 Br-uxendm.exe (8928)   2580        25         - 2019-06-03 11:30:12Z
.... 0xe302c9fc5080 Br-uxendm.exe (12456)  2580        16         - 2019-06-03 11:42:02Z

... 0xe302cb16d540 Br-uxendm.exe (8312)    3568         0         - 2019-05-30 16:27:23Z
... 0xe302ca49d080 Br-uxendm.exe (8940)    3568         0         - 2019-05-20 12:22:05Z
... 0xe302caa4e080 Br-uxendm.exe (11808)   3568         0         - 2019-05-24 12:19:32Z
... 0xe302cc7130c0 Br-uxendm.exe (11968)   3568         0         - 2019-05-21 11:37:01Z

... 0xe302cc736080 iexplore.exe (224)      8456        20         - 2019-06-03 11:30:25Z
```

```
.... 0xe302cb955080 iexplore.exe (10124)    224       16        - 2019-06-03 11:30:25Z
```

```
(rekal) C:\DFIR\rekall>rekal -f memdump_190603_IE-tabs_3x.aff4 psxview
  _EPROCESS           name       pid  PsActiveProcessHead CSRSS PspCidTable Sessions Handles PSScan Thrdproc
-------------- ----------------- ----- ------------------- ----- ----------- -------- ------- ------ --------
0xe302cbcc1080 BrHostSvr.exe      2580 True                True  True        True     True    False  True
0xe302cb3e4540 BrConsole.exe      3084 True                True  True        True     True    False  False
0xe302c9e733c0 BrRemoteMgmtSv     3544 True                True  True        True     True    False  True
0xe302c9e753c0 BrService.exe      3568 True                True  True        True     True    False  True
0xe302ce771540 BrStatusMonito     8916 True                True  True        True     True    False  True
0xe302cb840080 Br-uxendm.exe      8928 True                True  True        True     True    False  True
0xe302cf668080 Br-uxendm.exe      7012 True                True  True        True     True    False  True
0xe302ccab5080 Br-uxendm.exe      8212 True                True  True        True     True    False  True
0xe302cb16d540 Br-uxendm.exe      8312 True                False True        True     False   False  False
0xe302ca49d080 Br-uxendm.exe      8940 True                False True        True     False   False  False
0xe302caa4e080 Br-uxendm.exe     11808 True                False True        True     False   False  False
0xe302cc7130c0 Br-uxendm.exe     11968 True                False True        True     False   False  False
0xe302cc736080 iexplore.exe        224 True                True  True        True     True    False  True
0xe302cb5600c0 iexplore.exe       9660 True                True  True        True     True    False  True
0xe302cb955080 iexplore.exe      10124 True                True  True        True     True    False  True
0xe302cca762c0 iexplore.exe      11384 True                True  True        True     True    False  True
```

```
(rekal) C:\DFIR\rekall>rekal -f memdump_190603_IE-tabs_1x.aff4 psxview
_EPROCESS           name       pid  PsActiveProcessHead CSRSS PspCidTable Sessions Handles PSScan Thrdproc
-------------- ----------------- ----- ------------------- ----- ----------- -------- ------- ------ --------
0xe302cbcc1080 BrHostSvr.exe      2580 True                True  True        True     True    False  True
0xe302cb3e4540 BrConsole.exe      3084 True                True  True        True     True    False  False
0xe302c9e733c0 BrRemoteMgmtSv     3544 True                True  True        True     True    False  True
0xe302c9e753c0 BrService.exe      3568 True                True  True        True     True    False  True
0xe302ce771540 BrStatusMonito     8916 True                True  True        True     True    False  False
0xe302cb840080 Br-uxendm.exe      8928 True                True  True        True     True    False  True
0xe302c9fc5080 Br-uxendm.exe     12456 True                True  True        True     True    False  True

0xe302cb16d540 Br-uxendm.exe      8312 True                False True        True     False   False  False
0xe302ca49d080 Br-uxendm.exe      8940 True                False True        True     False   False  False
0xe302caa4e080 Br-uxendm.exe     11808 True                False True        True     False   False  False
0xe302cc7130c0 Br-uxendm.exe     11968 True                False True        True     False   False  False
0xe302cc736080 iexplore.exe        224 True                True  True        True     True    False  True
0xe302cb955080 iexplore.exe      10124 True                True  True        True     True    False  True
```

G.2.3 Testing with Rekall against VMs

```
C:\Users\Ute Schueller>workon rekal

 (rekal) C:\DFIR\rekall>rekal -f memdump_190603_IE-tabs_3x.aff4 vmscan
No handlers could be found for logger "rekall.plugins.tools"
Description                               Type          Valid   EPT
----------------------------------------- -------------------- -------- ---
  VM [2 vCORE(s), AMD64]                            VM      True 0x2537901E
  VM [2 vCORE(s), AMD64]                            VM      True 0x1BC2F401E
  VM [2 vCORE(s), AMD64]                            VM      True 0x1CE71401E
```

```
(rekal) C:\DFIR\rekall>rekall -f 190603_Bromium\memdump_190603_IE-
tabs_3x.aff4 --ept=0x2537901E version_scan --name_regex="krnl"
No handlers could be found for logger "rekall.plugins.tools"
    offset                   guid                              pdb
```

```
-------------- -------------------------------- ------------------------------
     0x1ba5090 A46DE659B91CC96DDFD3EC6313473A071 dxgkrnl.pdb
     0x2381e90 8CFB49428DC86A330CE257778E0C2F931 ntkrnlmp.pdb


(rekal) C:\DFIR\rekall>rekall -f 190603_Bromium\memdump_190603_IE-
tabs_3x.aff4 --ept=0x1BC2F401E version_scan --name_regex="krnl"
No handlers could be found for logger "rekall.plugins.tools"
    offset                guid                             pdb
-------------- -------------------------------- ------------------------------
     0x1ba5090 A46DE659B91CC96DDFD3EC6313473A071 dxgkrnl.pdb
     0x2381e90 8CFB49428DC86A330CE257778E0C2F931 ntkrnlmp.pdb



(rekal) C:\DFIR\rekall>rekall -f 190603_Bromium\memdump_190603_IE-
tabs_3x.aff4 --ept=0x1CE71401E version_scan --name_regex="krnl"
No handlers could be found for logger "rekall.plugins.tools"
    offset                guid                             pdb
-------------- -------------------------------- ------------------------------
     0x1ba5090 A46DE659B91CC96DDFD3EC6313473A071 dxgkrnl.pdb
     0x2381e90 8CFB49428DC86A330CE257778E0C2F931 ntkrnlmp.pdb


(rekal) C:\DFIR\rekall>rekall -f 190603_Bromium\memdump_190603_IE-
tabs_3x.aff4 version_scan --name_regex="krnl"
No handlers could be found for logger "rekall.plugins.tools"
    offset                guid                             pdb
-------------- -------------------------------- ------------------------------
     0x2997fe0 20C711BBD4C21AA5C45BC774BC7E04721 ntkrnlmp.pdb
     0x8dceb80 A2B0477A6F8A409DB798B482B3BEB5E91 ntoskrnl.pdb
    0xa336d090 A46DE659B91CC96DDFD3EC6313473A071 dxgkrnl.pdb
    0xaef934a8 A46DE659B91CC96DDFD3EC6313473A076 xgkrnl.pdb
               4882801
    0xb22eae90 8CFB49428DC86A330CE257778E0C2F931 ntkrnlmp.pdb
   0x1aef6657c A46DE659B91CC96DDFD3EC6313473A071 dxgkrnl.pdb
               3F40028
   0x201f0c090 F4AF6C0AF407EBC00A8F98FD3D89CA901 dxgkrnl.pdb
   0x232f7eb54 0AB8F18581AD0D0DD59CA7082AC9554E1 KrnlProv.pdb


(rekal) C:\DFIR\rekall>rekall -f 190603_Bromium\memdump_190603_IE-
tabs_3x.aff4 --profile=nt/GUID/8CFB49428DC86A330CE257778E0C2F931 --
ept=0x2537901E psscan
No handlers could be found for logger "rekall.plugins.tools"
a    offset_p            name          pid     offset_v     ppid       pdb       stat
create_time               exit_time
- -------------- -------------------- ----- -------------- ------ -------------- ---- --
---------------------- ------------------------
> c:\dfir\git\rekall\rekall-core\rekall\obj.py(2515)Object()
-> "Type name must be a string, not %s" % name.__class__)
(Pdb) quit
2019-08-21 20:50:04,361:ERROR:rekall.1:Failed to decode {'profile': {'mro':
u'Ntkrnlmp:Ntoskrnl:BasicPEProfile:RelativeOffsetMixin:newobject:BasicClasses:Profile:ob
ject', 'name': u'nt/GUID/20C711BBD4C21AA5C45BC774BC7E04721'}, 'count': 0, 'target':
u'Pointer', 'type_name': 'Array', 'vm': {'dtb': 1744896, 'base': u'PhysicalAS', 'mro':
u'WindowsAMD64PagedMemory:WindowsPagedMemoryMixin:AMD64PagedMemory:IA32PagedMemoryPae:IA
32PagedMemory:PagedReader:BaseAddressSpace:newobject:object', 'cls':
u'WindowsAMD64PagedMemory'}, 'mro': u'Array:BaseObject:object', 'offset':
272693869284448L, 'target_args': {'target': u'_OBJECT_TYPE'}, 'name': 'Array'}: Type
name must be a string, not <class 'future.types.newstr.newstr'>
  0x8609c9c4e040 System                 4 0x8609c9c4e040      0       0x1aa000 P
2019-05-30 16:24:55Z
  0x8609c9cda040 Registry              88 0x8609c9cda040      4      0x1900000 P
2019-05-30 16:24:54Z
  0x8609cda38440 dllhost.exe         2120 0x8609cda38440    720      0xa240000 P
2019-05-30 16:26:05Z
  0x8609cdf52080 wininit.exe          468 0x8609cdf52080    384       0xf40000 P
2019-05-30 16:25:00Z
  0x8609cdf91140 services.exe         588 0x8609cdf91140    468     0x142c0000 P
2019-05-30 16:25:01Z
  0x8609cdf92080 winlogon.exe         532 0x8609cdf92080    460      0xad40000 P
2019-05-30 16:25:00Z
  0x8609d0a40140 fontdrvhost.ex       744 0x8609d0a40140    532     0x16580000 P
```

```
2019-05-30 16:25:02Z
  0x8609d0ae1140 fontdrvhost.ex         752 0x8609d0ae1140    468     0x165c0000 P
2019-05-30 16:25:02Z
  0x8609d0b41300 svchost.exe            836 0x8609d0b41300    588     0x15b80000 P
2019-05-30 16:25:08Z
  0x8609d0c87080 ielowutil.exe         3664 0x8609d0c87080    720       0x380000
2019-06-03 11:34:22Z     2019-06-03 11:39:23Z
  0x8609d0cb84c0 iexplore.exe          3068 0x8609d0cb84c0   1768      0xcdc0000 P
2019-05-30 16:26:06Z
  0x8609d0d9c3c0 iexplore.exe          1768 0x8609d0d9c3c0    720     0x31600000 P
2019-05-30 16:26:04Z
  0x8609d0fa2240 svchost.exe            928 0x8609d0fa2240    588     0x1bf40000 P
2019-05-30 16:25:09Z
  0x8609d1160300 svchost.exe           1392 0x8609d1160300    588     0x1a540000 P
2019-05-30 16:25:09Z
  0x8609d121d240 svchost.exe           1640 0x8609d121d240    588     0x15fc0000 P
2019-05-30 16:25:10Z
  0x8609d1294280 BrUcvmService.        1916 0x8609d1294280    588     0x21c40000 P
2019-05-30 16:25:10Z
  0x8609d1615300 uxenclipboard.        2384 0x8609d1615300   1952      0x3a80000 P
2019-05-30 16:25:13Z
  0x8609d166b080 sppsvc.exe            2592 0x8609d166b080    588     0x1be40000 P
2019-05-30 16:25:13Z
  0x8609d1671080 net.exe               2644 0x8609d1671080   2564     0x25580000 P
2019-05-30 16:25:14Z     2019-05-30 16:25:14Z
  0x8609d16dc0c0 SppExtComObj.E        2748 0x8609d16dc0c0    720     0x32b00000 P
2019-05-30 16:25:35Z
  0x8609d1bcc3c0 firefox.exe           3520 0x8609d1bcc3c0    948     0x1cf40000 P
2019-05-30 16:26:11Z     2019-06-03 11:30:30Z
  0x8609d1d4a4c0 POWERPNT.EXE          3528 0x8609d1d4a4c0    948      0x8600000 P
2019-05-30 16:26:28Z     2019-06-03 11:30:28Z
  0x8609d1ef0480 AcroRd32.exe          3140 0x8609d1ef0480    948     0x21d00000 P
2019-05-30 16:26:34Z     2019-06-03 11:30:27Z
  0x8609d2092080 explorer.exe          4644 0x8609d2092080    948     0x1d400000 P
2019-05-30 16:26:55Z
  0x8609d209c400 rundll32.exe          4940 0x8609d209c400    720     0x1948b000 P
2019-05-30 16:26:51Z
  0x8609d225e340 DiagnosticsHub        4148 0x8609d225e340    588     0x24fc0000 P
2019-06-03 11:35:58Z


(rekal) C:\DFIR\rekall>rekall -f 190603_Bromium\memdump_190603_IE-
tabs_3x.aff4 --profile=nt/GUID/20C711BBD4C21AA5C45BC774BC7E04721  --
ept=0x2537901E psscan
No handlers could be found for logger "rekall.plugins.tools"
a    offset_p            name           pid     offset_v     ppid       pdb       stat
create_time                 exit_time
- -------------- -------------------- ----- -------------- ------ -------------- ---- --
---------------------- ------------------------
> c:\dfir\git\rekall\rekall-core\rekall\obj.py(2515)Object()
-> "Type name must be a string, not %s" % name.__class__)
(Pdb) quit
2019-08-21 19:24:48,371:ERROR:rekall.1:Failed to decode {'profile': {'mro':
u'Ntkrnlmp:Ntoskrnl:BasicPEProfile:RelativeOffsetMixin:newobject:BasicClasses:Profile:ob
ject', 'name': u'nt/GUID/8CFB49428DC86A330CE257778E0C2F931'}, 'count': 0, 'target':
u'Pointer', 'type_name': 'Array', 'vm': {'dtb': 1744896, 'base': u'PhysicalAS', 'mro':
u'WindowsAMD64PagedMemory:WindowsPagedMemoryMixin:AMD64PagedMemory:IA32PagedMemoryPae:IA
32PagedMemory:PagedReader:BaseAddressSpace:newobject:object', 'cls':
u'WindowsAMD64PagedMemory'}, 'mro': u'Array:BaseObject:object', 'offset':
272693869284448L, 'target_args': {'target': u'_OBJECT_TYPE'}, 'name': 'Array'}: Type
name must be a string, not <class 'future.types.newstr.newstr'>
  0x8609c9c4e040 System                   4 0x8609c9c4e040      0       0x1aa000 P
2019-05-30 16:24:55Z
  0x8609c9cda040 Registry                88 0x8609c9cda040      4      0x1900000 P
2019-05-30 16:24:54Z
  0x8609cda38440 dllhost.exe           2120 0x8609cda38440    720      0xa240000 P
2019-05-30 16:26:05Z
  0x8609cdf52080 wininit.exe            468 0x8609cdf52080    384       0xf40000 P
2019-05-30 16:25:00Z
  0x8609cdf91140 services.exe           588 0x8609cdf91140    468     0x142c0000 P
2019-05-30 16:25:01Z
  0x8609cdf92080 winlogon.exe           532 0x8609cdf92080    460      0xad40000 P
2019-05-30 16:25:00Z
  0x8609d0a40140 fontdrvhost.ex         744 0x8609d0a40140    532     0x16580000 P
2019-05-30 16:25:02Z
```

```
  0x8609d0ae1140 fontdrvhost.ex         752 0x8609d0ae1140    468     0x165c0000 P
2019-05-30 16:25:02Z
  0x8609d0b41300 svchost.exe            836 0x8609d0b41300    588     0x15b80000 P
2019-05-30 16:25:08Z
  0x8609d0c87080 ielowutil.exe         3664 0x8609d0c87080    720       0x380000
2019-06-03 11:34:22Z     2019-06-03 11:39:23Z
  0x8609d0cb84c0 iexplore.exe          3068 0x8609d0cb84c0   1768      0xcdc0000 P
2019-05-30 16:26:06Z
  0x8609d0d9c3c0 iexplore.exe          1768 0x8609d0d9c3c0    720     0x31600000 P
2019-05-30 16:26:04Z
  0x8609d0fa2240 svchost.exe            928 0x8609d0fa2240    588     0x1bf40000 P
2019-05-30 16:25:09Z
  0x8609d1160300 svchost.exe           1392 0x8609d1160300    588     0x1a540000 P
2019-05-30 16:25:09Z
  0x8609d121d240 svchost.exe           1640 0x8609d121d240    588     0x15fc0000 P
2019-05-30 16:25:10Z
  0x8609d1294280 BrUcvmService.        1916 0x8609d1294280    588     0x21c40000 P
2019-05-30 16:25:10Z
  0x8609d1615300 uxenclipboard.        2384 0x8609d1615300   1952      0x3a80000 P
2019-05-30 16:25:13Z
  0x8609d166b080 sppsvc.exe            2592 0x8609d166b080    588     0x1be40000 P
2019-05-30 16:25:13Z
  0x8609d1671080 net.exe               2644 0x8609d1671080   2564     0x25580000 P
2019-05-30 16:25:14Z     2019-05-30 16:25:14Z
  0x8609d16dc0c0 SppExtComObj.E        2748 0x8609d16dc0c0    720     0x32b00000 P
2019-05-30 16:25:35Z
  0x8609d1bcc3c0 firefox.exe           3520 0x8609d1bcc3c0    948     0x1cf40000 P
2019-05-30 16:26:11Z     2019-06-03 11:30:30Z
  0x8609d1d4a4c0 POWERPNT.EXE          3528 0x8609d1d4a4c0    948      0x8600000 P
2019-05-30 16:26:28Z     2019-06-03 11:30:28Z
  0x8609d1ef0480 AcroRd32.exe          3140 0x8609d1ef0480    948     0x21d00000 P
2019-05-30 16:26:34Z     2019-06-03 11:30:27Z
  0x8609d2092080 explorer.exe          4644 0x8609d2092080    948     0x1d400000 P
2019-05-30 16:26:55Z
  0x8609d209c400 rundll32.exe          4940 0x8609d209c400    720     0x1948b000 P
2019-05-30 16:26:51Z
  0x8609d225e340 DiagnosticsHub        4148 0x8609d225e340    588     0x24fc0000 P
2019-06-03 11:35:58Z
```

```
(rekal) C:\DFIR\rekall>rekall -f 190603_Bromium\memdump_190603_IE-
tabs_3x.aff4 --profile=nt/GUID/8CFB49428DC86A330CE257778E0C2F931 --
ept=0x1BC2F401E psscan
No handlers could be found for logger "rekall.plugins.tools"
a    offset_p            name           pid     offset_v     ppid       pdb       stat
create_time               exit_time
- -------------- -------------------- ----- -------------- ------ ------------- ---- --
---------------------- ------------------------
> c:\dfir\git\rekall\rekall-core\rekall\obj.py(2515)Object()
-> "Type name must be a string, not %s" % name.__class__)
(Pdb) quit
2019-08-21 20:53:39,921:ERROR:rekall.1:Failed to decode {'profile': {'mro':
u'Ntkrnlmp:Ntoskrnl:BasicPEProfile:RelativeOffsetMixin:newobject:BasicClasses:Profile:ob
ject', 'name': u'nt/GUID/20C711BBD4C21AA5C45BC774BC7E04721'}, 'count': 0, 'target':
u'Pointer', 'type_name': 'Array', 'vm': {'dtb': 1744896, 'base': u'PhysicalAS', 'mro':
u'WindowsAMD64PagedMemory:WindowsPagedMemoryMixin:AMD64PagedMemory:IA32PagedMemoryPae:IA
32PagedMemory:PagedReader:BaseAddressSpace:newobject:object', 'cls':
u'WindowsAMD64PagedMemory'}, 'mro': u'Array:BaseObject:object', 'offset':
272693869284448L, 'target_args': {'target': u'_OBJECT_TYPE'}, 'name': 'Array'}: Type
name must be a string, not <class 'future.types.newstr.newstr'>
  0x8609c9c4e040 System                   4 0x8609c9c4e040      0       0x1aa000
2019-05-30 16:24:55Z
  0x8609c9cda040 Registry                88 0x8609c9cda040      4      0x1900000
2019-05-30 16:24:54Z
  0x8609cda38440 dllhost.exe           2120 0x8609cda38440    720      0xa240000
2019-05-30 16:26:05Z
  0x8609cdf52080 wininit.exe            468 0x8609cdf52080    384       0xf40000
2019-05-30 16:25:00Z
  0x8609cdf91140 services.exe           588 0x8609cdf91140    468     0x142c0000
2019-05-30 16:25:01Z
  0x8609cdf92080 winlogon.exe           532 0x8609cdf92080    460      0xad40000
2019-05-30 16:25:00Z
  0x8609d0a40140 fontdrvhost.ex         744 0x8609d0a40140    532     0x16580000
2019-05-30 16:25:02Z
```

```
  0x8609d0ae1140 fontdrvhost.ex         752 0x8609d0ae1140    468     0x165c0000
2019-05-30 16:25:02Z
  0x8609d0b41300 svchost.exe            836 0x8609d0b41300    588     0x15b80000
2019-05-30 16:25:08Z
  0x8609d0bfc240 svchost.exe           2004 0x8609d0bfc240    588     0x2ecc0000
2019-05-30 16:26:03Z     2019-06-03 11:37:46Z
  0x8609d0cb84c0 iexplore.exe          3068 0x8609d0cb84c0   1768      0xcdc0000
2019-05-30 16:26:06Z
  0x8609d0d9c3c0 iexplore.exe          1768 0x8609d0d9c3c0    720     0x31600000
2019-05-30 16:26:04Z
  0x8609d0fa2240 svchost.exe            928 0x8609d0fa2240    588     0x1bf40000
2019-05-30 16:25:09Z
  0x8609d1160300 svchost.exe           1392 0x8609d1160300    588     0x1a540000
2019-05-30 16:25:09Z
  0x8609d121d240 svchost.exe           1640 0x8609d121d240    588     0x15fc0000
2019-05-30 16:25:10Z
  0x8609d1294280 BrUcvmService.        1916 0x8609d1294280    588     0x21c40000
2019-05-30 16:25:10Z
  0x8609d1615300 uxenclipboard.        2384 0x8609d1615300   1952      0x3a80000
2019-05-30 16:25:13Z
  0x8609d166b080 sppsvc.exe            2592 0x8609d166b080    588     0x1be40000
2019-05-30 16:25:13Z
  0x8609d1671080 net.exe               2644 0x8609d1671080   2564     0x25580000
2019-05-30 16:25:14Z     2019-05-30 16:25:14Z
  0x8609d16dc0c0 SppExtComObj.E        2748 0x8609d16dc0c0    720     0x32b00000
2019-05-30 16:25:35Z
  0x8609d1aaf300 audiodg.exe           3816 0x8609d1aaf300   1392     0x115c0000
2019-05-30 16:26:14Z     2019-06-03 11:41:09Z
  0x8609d1ad4080 WmiPrvSE.exe          3680 0x8609d1ad4080    720     0x275d6000
2019-05-30 16:26:13Z     2019-06-03 11:38:07Z
  0x8609d1bcc3c0 firefox.exe           3520 0x8609d1bcc3c0    948     0x1cf40000
2019-05-30 16:26:11Z     2019-06-03 11:37:02Z
  0x8609d1d4a4c0 POWERPNT.EXE          3528 0x8609d1d4a4c0    948      0x8600000
2019-05-30 16:26:28Z     2019-06-03 11:37:00Z
  0x8609d1d50540 TiWorker.exe          5056 0x8609d1d50540    720     0x10240000
2019-06-03 11:40:18Z
  0x8609d1ef0480 AcroRd32.exe          3140 0x8609d1ef0480    948     0x21d00000
2019-05-30 16:26:34Z     2019-06-03 11:37:00Z
  0x8609d2092080 explorer.exe          4644 0x8609d2092080    948     0x1d400000
2019-05-30 16:26:55Z
  0x8609d209c400 rundll32.exe          4940 0x8609d209c400    720     0x1948b000
2019-05-30 16:26:51Z
```

```
(rekal) C:\DFIR\rekall>rekall -f 190603_Bromium\memdump_190603_IE-tabs_3x.aff4 --profile=nt/GUID/20C711BBD4C21AA5C45BC774BC7E04721  --ept=0x1BC2F401E psscan
No handlers could be found for logger "rekall.plugins.tools"
a    offset_p            name           pid     offset_v     ppid       pdb       stat
create_time               exit_time
- -------------- -------------------- ----- -------------- ------ -------------- ---- --
---------------------- ------------------------
> c:\dfir\git\rekall\rekall-core\rekall\obj.py(2515)Object()
-> "Type name must be a string, not %s" % name.__class__)
(Pdb) quit
2019-08-21 19:16:16,405:ERROR:rekall.1:Failed to decode {'profile': {'mro':
u'Ntkrnlmp:Ntoskrnl:BasicPEProfile:RelativeOffsetMixin:newobject:BasicClasses:Profile:ob
ject', 'name': u'nt/GUID/20C711BBD4C21AA5C45BC774BC7E04721'}, 'count': 0, 'target':
u'Pointer', 'type_name': 'Array', 'vm': {'dtb': 1744896, 'base': u'PhysicalAS', 'mro':
u'WindowsAMD64PagedMemory:WindowsPagedMemoryMixin:AMD64PagedMemory:IA32PagedMemoryPae:IA
32PagedMemory:PagedReader:BaseAddressSpace:newobject:object', 'cls':
u'WindowsAMD64PagedMemory'}, 'mro': u'Array:BaseObject:object', 'offset':
272693869284448L, 'target_args': {'target': u'_OBJECT_TYPE'}, 'name': 'Array'}: Type
name must be a string, not <class 'future.types.newstr.newstr'>
  0x8609c9c4e040 System                   4 0x8609c9c4e040      0       0x1aa000
2019-05-30 16:24:55Z
  0x8609c9cda040 Registry                88 0x8609c9cda040      4      0x1900000
2019-05-30 16:24:54Z
  0x8609cda38440 dllhost.exe           2120 0x8609cda38440    720      0xa240000
2019-05-30 16:26:05Z
  0x8609cdf52080 wininit.exe            468 0x8609cdf52080    384       0xf40000
2019-05-30 16:25:00Z
  0x8609cdf91140 services.exe           588 0x8609cdf91140    468     0x142c0000
2019-05-30 16:25:01Z
  0x8609cdf92080 winlogon.exe           532 0x8609cdf92080    460      0xad40000
```

```
2019-05-30 16:25:00Z
  0x8609d0a40140 fontdrvhost.ex         744 0x8609d0a40140    532     0x16580000
2019-05-30 16:25:02Z
  0x8609d0ae1140 fontdrvhost.ex         752 0x8609d0ae1140    468     0x165c0000
2019-05-30 16:25:02Z
  0x8609d0b41300 svchost.exe            836 0x8609d0b41300    588     0x15b80000
2019-05-30 16:25:08Z
  0x8609d0bfc240 svchost.exe           2004 0x8609d0bfc240    588     0x2ecc0000
2019-05-30 16:26:03Z     2019-06-03 11:37:46Z
  0x8609d0cb84c0 iexplore.exe          3068 0x8609d0cb84c0   1768      0xcdc0000
2019-05-30 16:26:06Z
  0x8609d0d9c3c0 iexplore.exe          1768 0x8609d0d9c3c0    720     0x31600000
2019-05-30 16:26:04Z
  0x8609d0fa2240 svchost.exe            928 0x8609d0fa2240    588     0x1bf40000
2019-05-30 16:25:09Z
  0x8609d1160300 svchost.exe           1392 0x8609d1160300    588     0x1a540000
2019-05-30 16:25:09Z
  0x8609d121d240 svchost.exe           1640 0x8609d121d240    588     0x15fc0000
2019-05-30 16:25:10Z
  0x8609d1294280 BrUcvmService.        1916 0x8609d1294280    588     0x21c40000
2019-05-30 16:25:10Z
  0x8609d1615300 uxenclipboard.        2384 0x8609d1615300   1952      0x3a80000
2019-05-30 16:25:13Z
  0x8609d166b080 sppsvc.exe            2592 0x8609d166b080    588     0x1be40000
2019-05-30 16:25:13Z
  0x8609d1671080 net.exe               2644 0x8609d1671080   2564     0x25580000
2019-05-30 16:25:14Z     2019-05-30 16:25:14Z
  0x8609d16dc0c0 SppExtComObj.E        2748 0x8609d16dc0c0    720     0x32b00000
2019-05-30 16:25:35Z
  0x8609d1aaf300 audiodg.exe           3816 0x8609d1aaf300   1392     0x115c0000
2019-05-30 16:26:14Z     2019-06-03 11:41:09Z
  0x8609d1ad4080 WmiPrvSE.exe          3680 0x8609d1ad4080    720     0x275d6000
2019-05-30 16:26:13Z     2019-06-03 11:38:07Z
  0x8609d1bcc3c0 firefox.exe           3520 0x8609d1bcc3c0    948     0x1cf40000
2019-05-30 16:26:11Z     2019-06-03 11:37:02Z
  0x8609d1d4a4c0 POWERPNT.EXE          3528 0x8609d1d4a4c0    948      0x8600000
2019-05-30 16:26:28Z     2019-06-03 11:37:00Z
  0x8609d1d50540 TiWorker.exe          5056 0x8609d1d50540    720     0x10240000
2019-06-03 11:40:18Z
  0x8609d1ef0480 AcroRd32.exe          3140 0x8609d1ef0480    948     0x21d00000
2019-05-30 16:26:34Z     2019-06-03 11:37:00Z
  0x8609d2092080 explorer.exe          4644 0x8609d2092080    948     0x1d400000
2019-05-30 16:26:55Z
  0x8609d209c400 rundll32.exe          4940 0x8609d209c400    720     0x1948b000
2019-05-30 16:26:51Z
```

```
(rekal) C:\DFIR\rekall>rekall -f 190603_Bromium\memdump_190603_IE-
tabs_3x.aff4 --profile=nt/GUID/8CFB49428DC86A330CE257778E0C2F931 --
ept=0x1CE71401E psscan
No handlers could be found for logger "rekall.plugins.tools"
a    offset_p            name          pid     offset_v     ppid       pdb       stat
create_time               exit_time
- -------------- -------------------- ----- -------------- ------ ------------- ---- --
----------------------- ------------------------
> c:\dfir\git\rekall\rekall-core\rekall\obj.py(2515)Object()
-> "Type name must be a string, not %s" % name.__class__)
(Pdb) quit
2019-08-21 20:59:12,634:ERROR:rekall.1:Failed to decode {'profile': {'mro':
u'Ntkrnlmp:Ntoskrnl:BasicPEProfile:RelativeOffsetMixin:newobject:BasicClasses:Profile:ob
ject', 'name': u'nt/GUID/20C711BBD4C21AA5C45BC774BC7E04721'}, 'count': 0, 'target':
u'Pointer', 'type_name': 'Array', 'vm': {'dtb': 1744896, 'base': u'PhysicalAS', 'mro':
u'WindowsAMD64PagedMemory:WindowsPagedMemoryMixin:AMD64PagedMemory:IA32PagedMemoryPae:IA
32PagedMemory:PagedReader:BaseAddressSpace:newobject:object', 'cls':
u'WindowsAMD64PagedMemory'}, 'mro': u'Array:BaseObject:object', 'offset':
272693869284448L, 'target_args': {'target': u'_OBJECT_TYPE'}, 'name': 'Array'}: Type
name must be a string, not <class 'future.types.newstr.newstr'>
  0x8609c9c4e040 System                   4 0x8609c9c4e040      0       0x1aa000 P
2019-05-30 16:24:55Z
  0x8609c9cda040 Registry                88 0x8609c9cda040      4      0x1900000 P
2019-05-30 16:24:54Z
  0x8609cda38440 dllhost.exe           2120 0x8609cda38440    720      0xa240000 P
2019-05-30 16:26:05Z
  0x8609cdf52080 wininit.exe            468 0x8609cdf52080    384       0xf40000 P
```

```
2019-05-30 16:25:00Z
  0x8609cdf91140 services.exe           588 0x8609cdf91140    468     0x142c0000 P
2019-05-30 16:25:01Z
  0x8609cdf92080 winlogon.exe           532 0x8609cdf92080    460      0xad40000 P
2019-05-30 16:25:00Z
  0x8609d0a40140 fontdrvhost.ex         744 0x8609d0a40140    532     0x16580000 P
2019-05-30 16:25:02Z
  0x8609d0ae1140 fontdrvhost.ex         752 0x8609d0ae1140    468     0x165c0000 P
2019-05-30 16:25:02Z
  0x8609d0b41300 svchost.exe            836 0x8609d0b41300    588     0x15b80000 P
2019-05-30 16:25:08Z
  0x8609d0cb84c0 iexplore.exe          3068 0x8609d0cb84c0   1768      0xcdc0000 P
2019-05-30 16:26:06Z
  0x8609d0d9c3c0 iexplore.exe          1768 0x8609d0d9c3c0    720     0x31600000 P
2019-05-30 16:26:04Z
  0x8609d0fa2240 svchost.exe            928 0x8609d0fa2240    588     0x1bf40000 P
2019-05-30 16:25:09Z
  0x8609d1160300 svchost.exe           1392 0x8609d1160300    588     0x1a540000 P
2019-05-30 16:25:09Z
  0x8609d121d240 svchost.exe           1640 0x8609d121d240    588     0x15fc0000 P
2019-05-30 16:25:10Z
  0x8609d1294280 BrUcvmService.        1916 0x8609d1294280    588     0x21c40000 P
2019-05-30 16:25:10Z
  0x8609d1615300 uxenclipboard.        2384 0x8609d1615300   1952      0x3a80000 P
2019-05-30 16:25:13Z
  0x8609d166b080 sppsvc.exe            2592 0x8609d166b080    588     0x1be40000 P
2019-05-30 16:25:13Z
  0x8609d1671080 net.exe               2644 0x8609d1671080   2564     0x25580000 P
2019-05-30 16:25:14Z     2019-05-30 16:25:14Z
  0x8609d16dc0c0 SppExtComObj.E        2748 0x8609d16dc0c0    720     0x32b00000 P
2019-05-30 16:25:35Z
  0x8609d1bcc3c0 firefox.exe           3520 0x8609d1bcc3c0    948     0x1cf40000 P
2019-05-30 16:26:11Z     2019-06-03 11:34:54Z
  0x8609d1d4a4c0 POWERPNT.EXE          3528 0x8609d1d4a4c0    948      0x8600000 P
2019-05-30 16:26:28Z     2019-06-03 11:34:52Z
  0x8609d1ef0480 AcroRd32.exe          3140 0x8609d1ef0480    948     0x21d00000 P
2019-05-30 16:26:34Z     2019-06-03 11:34:51Z
  0x8609d2092080 explorer.exe          4644 0x8609d2092080    948     0x1d400000 P
2019-05-30 16:26:55Z
  0x8609d209c400 rundll32.exe          4940 0x8609d209c400    720     0x1948b000 P
2019-05-30 16:26:51Z
```

```
(rekal) C:\DFIR\rekall>rekall -f 190603_Bromium\memdump_190603_IE-
tabs_3x.aff4 --profile=nt/GUID/20C711BBD4C21AA5C45BC774BC7E04721  --
ept=0x1CE71401E psscan
No handlers could be found for logger "rekall.plugins.tools"
a    offset_p            name          pid     offset_v     ppid       pdb       stat
create_time               exit_time
- -------------- -------------------- ----- -------------- ------ -------------- ---- --
---------------------- ------------------------
> c:\dfir\git\rekall\rekall-core\rekall\obj.py(2515)Object()
-> "Type name must be a string, not %s" % name.__class__)
(Pdb) quit
2019-08-21 19:29:37,230:ERROR:rekall.1:Failed to decode {'profile': {'mro':
u'Ntkrnlmp:Ntoskrnl:BasicPEProfile:RelativeOffsetMixin:newobject:BasicClasses:Profile:ob
ject', 'name': u'nt/GUID/8CFB49428DC86A330CE257778E0C2F931'}, 'count': 0, 'target':
u'Pointer', 'type_name': 'Array', 'vm': {'dtb': 1744896, 'base': u'PhysicalAS', 'mro':
u'WindowsAMD64PagedMemory:WindowsPagedMemoryMixin:AMD64PagedMemory:IA32PagedMemoryPae:IA
32PagedMemory:PagedReader:BaseAddressSpace:newobject:object', 'cls':
u'WindowsAMD64PagedMemory'}, 'mro': u'Array:BaseObject:object', 'offset':
272693869284448L, 'target_args': {'target': u'_OBJECT_TYPE'}, 'name': 'Array'}: Type
name must be a string, not <class 'future.types.newstr.newstr'>
  0x8609c9c4e040 System                   4 0x8609c9c4e040      0       0x1aa000 P
2019-05-30 16:24:55Z
  0x8609c9cda040 Registry                88 0x8609c9cda040      4      0x1900000 P
2019-05-30 16:24:54Z
  0x8609cda38440 dllhost.exe           2120 0x8609cda38440    720      0xa240000 P
2019-05-30 16:26:05Z
  0x8609cdf52080 wininit.exe            468 0x8609cdf52080    384       0xf40000 P
2019-05-30 16:25:00Z
  0x8609cdf91140 services.exe           588 0x8609cdf91140    468     0x142c0000 P
2019-05-30 16:25:01Z
  0x8609cdf92080 winlogon.exe           532 0x8609cdf92080    460      0xad40000 P
```

```
2019-05-30 16:25:00Z
  0x8609d0a40140 fontdrvhost.ex         744 0x8609d0a40140    532     0x16580000 P
2019-05-30 16:25:02Z
  0x8609d0ae1140 fontdrvhost.ex         752 0x8609d0ae1140    468     0x165c0000 P
2019-05-30 16:25:02Z
  0x8609d0b41300 svchost.exe            836 0x8609d0b41300    588     0x15b80000 P
2019-05-30 16:25:08Z
  0x8609d0cb84c0 iexplore.exe          3068 0x8609d0cb84c0   1768      0xcdc0000 P
2019-05-30 16:26:06Z
  0x8609d0d9c3c0 iexplore.exe          1768 0x8609d0d9c3c0    720     0x31600000 P
2019-05-30 16:26:04Z
  0x8609d0fa2240 svchost.exe            928 0x8609d0fa2240    588     0x1bf40000 P
2019-05-30 16:25:09Z
  0x8609d1160300 svchost.exe           1392 0x8609d1160300    588     0x1a540000 P
2019-05-30 16:25:09Z
  0x8609d121d240 svchost.exe           1640 0x8609d121d240    588     0x15fc0000 P
2019-05-30 16:25:10Z
  0x8609d1294280 BrUcvmService.        1916 0x8609d1294280    588     0x21c40000 P
2019-05-30 16:25:10Z
  0x8609d1615300 uxenclipboard.        2384 0x8609d1615300   1952      0x3a80000 P
2019-05-30 16:25:13Z
  0x8609d166b080 sppsvc.exe            2592 0x8609d166b080    588     0x1be40000 P
2019-05-30 16:25:13Z
  0x8609d1671080 net.exe               2644 0x8609d1671080   2564     0x25580000 P
2019-05-30 16:25:14Z     2019-05-30 16:25:14Z
  0x8609d16dc0c0 SppExtComObj.E        2748 0x8609d16dc0c0    720     0x32b00000 P
2019-05-30 16:25:35Z
  0x8609d1bcc3c0 firefox.exe           3520 0x8609d1bcc3c0    948     0x1cf40000 P
2019-05-30 16:26:11Z     2019-06-03 11:34:54Z
  0x8609d1d4a4c0 POWERPNT.EXE          3528 0x8609d1d4a4c0    948      0x8600000 P
2019-05-30 16:26:28Z     2019-06-03 11:34:52Z
  0x8609d1ef0480 AcroRd32.exe          3140 0x8609d1ef0480    948     0x21d00000 P
2019-05-30 16:26:34Z     2019-06-03 11:34:51Z
  0x8609d2092080 explorer.exe          4644 0x8609d2092080    948     0x1d400000 P
2019-05-30 16:26:55Z
  0x8609d209c400 rundll32.exe          4940 0x8609d209c400    720     0x1948b000 P
2019-05-30 16:26:51Z
```

```
(rekal) C:\DFIR\rekall>rekall -f 190603_Bromium\memdump_190603_IE-
tabs_3x.aff4 --profile=nt/GUID/8CFB49428DC86A330CE257778E0C2F931 --
ept=0x2537901E find_dtb
No handlers could be found for logger "rekall.plugins.tools"
_EPROCESS (P)          name          pid       dtv         valid
-------------- -------------------- ----- -------------- ----------
0x000000ae2d0c cmd.exe                  0          0x18f True
0x00000255c9c0 Idle                     0       0x1aa000 True
0x0000033501bc cmd.exe                  0 0x48004300450052 True
0x00000673a078 csrss.exe            24989       0x600060 True
                                    93752
0x000008b08080 svchost.exe           2220      0xe140000 True
 Found possible _EPROCESS @ 0xAA5615C (In process cmd.exe        )
 (DTB: 0x9CECC0x00000aa5615c cmd.exe                  0 0x9ceccba800000010 True
0x00000b5f7c1c cmd.exe                  0 0x235000000000002 True
0x00000ce09080 lsass.exe              616     0x10f00000 True
0x00000db5d400 csrss.exe              392      0xdb00000 True
0x00000df2c300 svchost.exe            836     0x15b80000 True
0x000010692240 svchost.exe            720     0x19880000 True
0x000011426d4e System                   0 0x18f9b8237000000 True
0x000011427089 System                   0 0xb000000000e00 True
0x000011561240 svchost.exe           2004     0x2ecc0000 True
0x00001161ccc0 csrss.exe                0 0xffffa78298dfece0 True
0x00001161cdc0 csrss.exe            33685 0x84440f5710b9bf0a True
                                      504
0x00001161d640 svchost.exe          33685 0xc584c14560886485 True
                                      504
0x00001161d840 svchost.exe          33685 0xc584c1457dc14905 True
                                      504
0x00001161da80 svchost.exe          33685 0xc584c1457dd6ca85 True
                                      504
 Found possible _EPROCESS @ 0x140B9300 (In process svchost.exe    )
 (DTB: 0x11F40x0000140b9300 svchost.exe           1672     0x11f40000 True
 Found possible _EPROCESS @ 0x15073C7C (In process cmd.exe        )
 (DTB: 0x23410x000015073c7c cmd.exe             40275 0x2341500035415 True
                                    44083
 Found possible _EPROCESS @ 0x160F4080 (In process svchost.exe    )
 (DTB: 0x10F80x0000160f4080 svchost.exe           5084     0x10f80000 True
0x0000172fc240 svchost.exe           1640     0x15fc0000 True
0x000017e8b080 winlogon.exe           532      0xad40000 True
0x000018c7184c cmd.exe                  0 0x65006300690076 True
0x000019acf15c cmd.exe              10758          0x1b5 True
                                    38976
 Found possible _EPROCESS @ 0x1A864300 (In process svchost.exe    )
 (DTB: 0x15240x00001a864300 svchost.exe           1276     0x15240000 True
 Found possible _EPROCESS @ 0x1B667300 (In process svchost.exe    )
 (DTB: 0x1B600x00001b667300 svchost.exe           1092     0x1b600000 True
0x00001b80a240 svchost.exe            928     0x1bf40000 True
 Found possible _EPROCESS @ 0x1B900300 (In process svchost.exe    )
 (DTB: 0x1E2C0x00001b900300 svchost.exe           1008     0x1e2c0000 True
 Found possible _EPROCESS @ 0x1C299300 (In process svchost.exe    )
 (DTB: 0x1A540x00001c299300 svchost.exe           1392     0x1a540000 True
0x00001c426300 svchost.exe            884     0x1d150000 True
0x00001c45f280 svchost.exe           1168     0x1c180000 True
0x00001f549e5c cmd.exe                  0 0x88aa9a2000000000 True
0x00002035a778 csrss.exe                0       0xf8007c True
0x000020659648 csrss.exe            11161 0xffff8609cdebc178 True
                                    43456
0x000021708080 csrss.exe              476      0xf6c0000 True
0x0000249fa7e0 svchost.exe          11082 0x68636e75616c6c73 True
                                      696
0x000026fba2c0 svchost.exe           2304     0x2db80000 True
0x000027b36cc0 svchost.exe              0    0x100000008 True
0x00002b0ee240 svchost.exe           2352     0x2be00000 True
0x00002f87c540 svchost.exe           4272      0x5c40000 True
0x00003023eb60 lsass.exe            78644 0x32002d0031002d True
                                       21
0x000030aa5cac cmd.exe                  0 0x98f7b38000000000 True
 Found possible _EPROCESS @ 0x31CB7C4C (In process cmd.exe        )
 (DTB: 0x14460x000031cb7c4c cmd.exe              15693 0x1446391b7f104639 True
                                    12395
0x00003324e040 System                   4       0x1aa000 True
```

```
(rekal) C:\DFIR\rekall>rekall -f 190603_Bromium\memdump_190603_IE-
tabs_3x.aff4 --profile=nt/GUID/8CFB49428DC86A330CE257778E0C2F931 --
ept=0x1BC2F401E find_dtb
No handlers could be found for logger "rekall.plugins.tools"
_EPROCESS (P)          name          pid       dtv         valid
-------------- -------------------- ----- -------------- ----------
0x00000255c9c0 Idle                     0       0x1aa000 True
0x00000673a078 csrss.exe            24989       0x600060 True
                                    93752
0x000008b08080 svchost.exe           2220      0xe140000 True
0x00000aa5615c cmd.exe                  0 0x9ceccba800000010 True
0x00000b5f7c1c cmd.exe              11816 0x6e65726566655265 True
                                    42601
0x00000ce09080 lsass.exe              616     0x10f00000 True
0x00000db5d400 csrss.exe              392      0xdb00000 True
0x00000df2c300 svchost.exe            836     0x15b80000 True
0x0000106333bc cmd.exe                  0 0x6f006900740063 True
0x000010692240 svchost.exe            720     0x19880000 True
0x000011561240 svchost.exe           2004     0x2ecc0000 True       (DTB: 0x0)
0x00001161ccc0 csrss.exe                0 0xffffa78298dfece0 True
0x00001161cdc0 csrss.exe            33685 0x8c64d1b24b223734 True
                                      504
0x00001161d640 svchost.exe          33685 0xc584c14560886485 True
                                      504
0x00001161d840 svchost.exe          33685 0xc584c1457dc14905 True
                                      504
0x00001161d960 svchost.exe          33685 0xc584c14560807b65 True
                                      504
0x00001161da80 svchost.exe          33685 0xc584c1457dd6ca85 True
                                      504
0x0000140b9300 svchost.exe           1672     0x11f40000 True
 Found possible _EPROCESS @ 0x160F4080 (In process svchost.exe    )
 (DTB: 0x10F80x0000160f4080 svchost.exe           5084     0x10f80000 True
0x0000172fc240 svchost.exe           1640     0x15fc0000 True
0x000017e8b080 winlogon.exe           532      0xad40000 True
0x000018c7184c cmd.exe                  0 0x65006300690076 True
0x000019c66240 svchost.exe           1208     0x1a380000 True
```

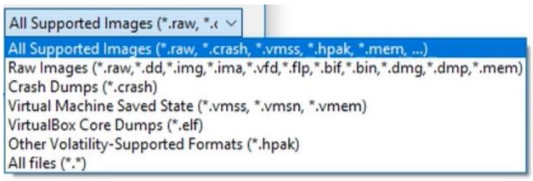
```
0x000019ed5c2e System               19720 0xf18b4818418b4840 True
                                    63232
0x00001a864300 svchost.exe           1276     0x15240000 True
 Found possible _EPROCESS @ 0x1B0792B5 (In process System          )
 (DTB: 0x80000x00001b0792b5 System                   0 0x800020821600b70 True
0x00001b667300 svchost.exe           1092     0x1b600000 True
0x00001b80a240 svchost.exe            928     0x1bf40000 True
0x00001b900300 svchost.exe           1008     0x1e2c0000 True
0x00001be99b60 lsass.exe            78644 0x32002d0031002d True
                                       21
 Found possible _EPROCESS @ 0x1C299300 (In process svchost.exe     )
 (DTB: 0x1A540x00001c299300 svchost.exe           1392     0x1a540000 True
0x00001c426300 svchost.exe            884     0x1d150000 True
0x00001c45f280 svchost.exe           1168     0x1c180000 True
0x00001cb7871c cmd.exe                  0 0x72006100740073 True
0x00002035a778 csrss.exe                0       0x780054 True
0x000020659648 csrss.exe            11144 0xffff8609cdebc178 True
                                    30352
 Found possible _EPROCESS @ 0x21708080 (In process csrss.exe       )
 (DTB: 0xF6C00x000021708080 csrss.exe              476      0xf6c0000 True
 Found possible _EPROCESS @ 0x22839B8C (In process cmd.exe         )
 (DTB: 0x70000x000022839b8c cmd.exe              78683 0x70007300200067 True
                                        8
0x000026fba2c0 svchost.exe           2304     0x2db80000 True
0x000027b36cc0 svchost.exe          47065 0x894888458b084d8b True
                                       08
0x000027b37680 svchost.exe          33685 0xc584c14560c45685 True
                                      504
0x00002b0ee240 svchost.exe           2352     0x2be00000 True
0x00002be2009c cmd.exe                  0 0x76005300580070 True
0x00002db8b7e0 svchost.exe          11082 0x68636e75616c6c73 True
                                      696
0x00002ede515c cmd.exe              10758          0x1b5 True       (DTB: 0x0)
                                    38976
0x0000302ec054 cmd.exe              43345   0x6c006c0064 True
                                      728
0x0000325cb1bc cmd.exe                  0 0x48004300450052 True
0x00003324e040 System                   4       0x1aa000 True
```

```
(rekal) C:\DFIR\rekall>rekall -f 190603_Bromium\memdump_190603_IE-
tabs_3x.aff4 --profile=nt/GUID/8CFB49428DC86A330CE257778E0C2F931 --
ept=0x1CE71401E find_dtb
No handlers could be found for logger "rekall.plugins.tools"
_EPROCESS (P)          name          pid       dtv         valid
-------------- -------------------- ----- -------------- ----------
 Found possible _EPROCESS @ 0x255C9C0 (In process Idle            )
 (DTB: 0x1AA000x00000255c9c0 Idle                     0       0x1aa000 True
 Found possible _EPROCESS @ 0x4AEF778 (In process csrss.exe       )
 (DTB: 0xF80070x000004aef778 csrss.exe                0       0xf8007c True
0x00000673a078 csrss.exe            24989       0x600060 True
                                    93752
 Found possible _EPROCESS @ 0x7A62D0C (In process cmd.exe         )
 (DTB: 0xBE33B0x000007a62d0c cmd.exe              23823 0xbe33b19e88004bec True
                                    91473
0x000008b08080 svchost.exe           2220      0xe140000 True
0x000009a6ceec cmd.exe              52428 0x5005f6000821fb84 True
                                        8
 Found possible _EPROCESS @ 0xAA5615C (In process cmd.exe         )
 (DTB: 0x9CECC0x00000aa5615c cmd.exe                  0 0x9ceccba800000010 True
0x00000b810e5c cmd.exe                  0 0xff13220fff6b8265 True
 Found possible _EPROCESS @ 0xCE09080 (In process lsass.exe       )
 (DTB: 0x10F000x00000ce09080 lsass.exe              616     0x10f00000 True
0x00000db5d400 csrss.exe              392      0xdb00000 True
 Found possible _EPROCESS @ 0xDF2C300 (In process svchost.exe     )
 (DTB: 0x15B800x00000df2c300 svchost.exe            836     0x15b80000 True
0x00000e37acf0 csrss.exe            42837 0xff97930effb2ad0b True
                                    28128
0x000010692240 svchost.exe            720     0x19880000 True
0x0000109f384c cmd.exe                  0 0x65006300690076 True
0x000011561240 svchost.exe           2004     0x2ecc0000 True       (DTB: 0x0)
0x00001161ccc0 csrss.exe                0 0xffffa78298dfece0 True
0x00001161cdc0 csrss.exe            33685 0x9f19698d4a9b570f True
                                      504
```

```
0x00001161d640 svchost.exe          33685 0xc584c14560886485 True
                                      504
0x00001161d840 svchost.exe          33685 0xc584c1457dc14905 True
                                      504
0x00001161d960 svchost.exe          33685 0xc584c14560807b65 True
                                      504
0x00001161da80 svchost.exe          33685 0xc584c1457dd6ca85 True
                                      504
0x000013d0ac3d System               57045 0x6100680073002000 True      B: 0x0)
                                     2736
0x0000140b9300 svchost.exe           1672     0x11f40000 True
0x000015073c7c cmd.exe              42782 0xfff7f7f7fff7f7f7 True
                                    11455
 Found possible _EPROCESS @ 0x160F4080 (In process svchost.exe    )
 (DTB: 0x10F80x0000160f4080 svchost.exe            5084     0x10f80000 True
 Found possible _EPROCESS @ 0x172FC240 (In process svchost.exe    )
 (DTB: 0x15FC0x0000172fc240 svchost.exe            1640     0x15fc0000 True
0x000017e8b080 winlogon.exe           532      0xad40000 True
0x000018cadc1c cmd.exe                  0 0xe36ee22800000165 True
0x000019c66240 svchost.exe           1208     0x1a380000 True
0x00001a864300 svchost.exe           1276     0x15240000 True
 Found possible _EPROCESS @ 0x1B667300 (In process svchost.exe    )
 (DTB: 0x1B600x00001b667300 svchost.exe            1092     0x1b600000 True
0x00001b80a240 svchost.exe            928     0x1bf40000 True
0x00001b900300 svchost.exe           1008     0x1e2c0000 True
0x00001c299300 svchost.exe           1392     0x1a540000 True
0x00001c426300 svchost.exe            884     0x1d150000 True
0x00001c45f280 svchost.exe           1168     0x1c180000 True
0x000020659648 csrss.exe            20957 0xffff8609cdebc178 True
                                    12247
 Found possible _EPROCESS @ 0x21708080 (In process csrss.exe      )
 (DTB: 0xF6C00x000021708080 csrss.exe               476      0xf6c0000 True
0x00002173bebc cmd.exe              52428     0x20666e57 True
                                        8
0x000023173080 svchost.exe           2052 0x7ff98edd4002 True
 Found possible _EPROCESS @ 0x25173B60 (In process lsass.exe      )
 (DTB: 0x32000x000025173b60 lsass.exe             78644 0x32002d0031002d True
                                       21
0x000026f44e8c cmd.exe              52428 0x65007400740069 True
                                        8
0x000026fba2c0 svchost.exe           2304     0x2db80000 True
 Found possible _EPROCESS @ 0x27B36CC0 (In process svchost.exe    )
 (DTB: 0x89480x000027b36cc0 svchost.exe           47065 0x894888458b084d8b True
                                       08
 Found possible _EPROCESS @ 0x28A0CC4C (In process cmd.exe        )
 (DTB: 0xD0EE0x000028a0cc4c cmd.exe                  0 0xd0ee71b00000000c True
 Found possible _EPROCESS @ 0x2B0EE240 (In process svchost.exe    )
 (DTB: 0x2BE00x00002b0ee240 svchost.exe            2352     0x2be00000 True
 Found possible _EPROCESS @ 0x2BC291BC (In process cmd.exe        )
 (DTB: 0x48000x00002bc291bc cmd.exe                  0 0x48004300450052 True
0x00002db8b7e0 svchost.exe          11082 0x68636e75616c6c73 True
                                      696
0x00002e85fcf0 csrss.exe            26101 0x7ff99b94a7b0 True
                                    75792
0x00002ede515c cmd.exe              10758          0x1b5 True
                                    38976
0x000030430d0c cmd.exe                  0          0x165 True
0x00003324e040 System                   4       0x1aa000 True
```

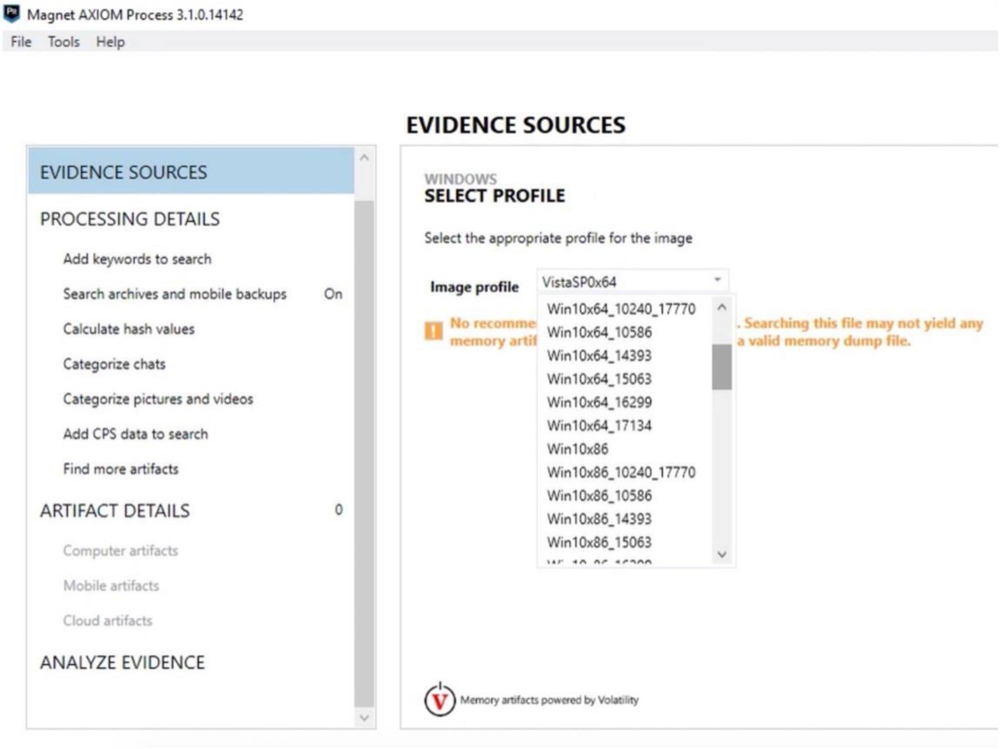
G.2.4 Testing with Magnet Axiom

First of all, Axiom has Volatility integrated and is generally able to read memory images. It even understands a lot of different formats:


All Supported Images (*.raw, *.c
All Supported Images (*.raw, *.crash, *.vmss, *.hpak, *.mem, ...)
Raw Images (*.raw,*.dd,*.img,*.ima,*.vfd,*.flp,*.bif,*.bin,*.dmg,*.dmp,*.mem)
Crash Dumps (*.crash)
Virtual Machine Saved State (*.vmss, *.vmsn, *.vmem)
VirtualBox Core Dumps (*.elf)
Other Volatility-Supported Formats (*.hpak)
All files (*.*)

And, Axiom is capable of reading a raw image from the Bromium system.

But that's it for the good news ;-) an automatic detection of the needed profile for analysis fails and trying to select a profile, manually, shows, that the latest available profile is for Windows 10, Build 17134:



Magnet AXIOM Process 3.1.0.14142

EVIDENCE SOURCES

WINDOWS
SELECT PROFILE

Select the appropriate profile for the image

Image profile: VistaSP0x64

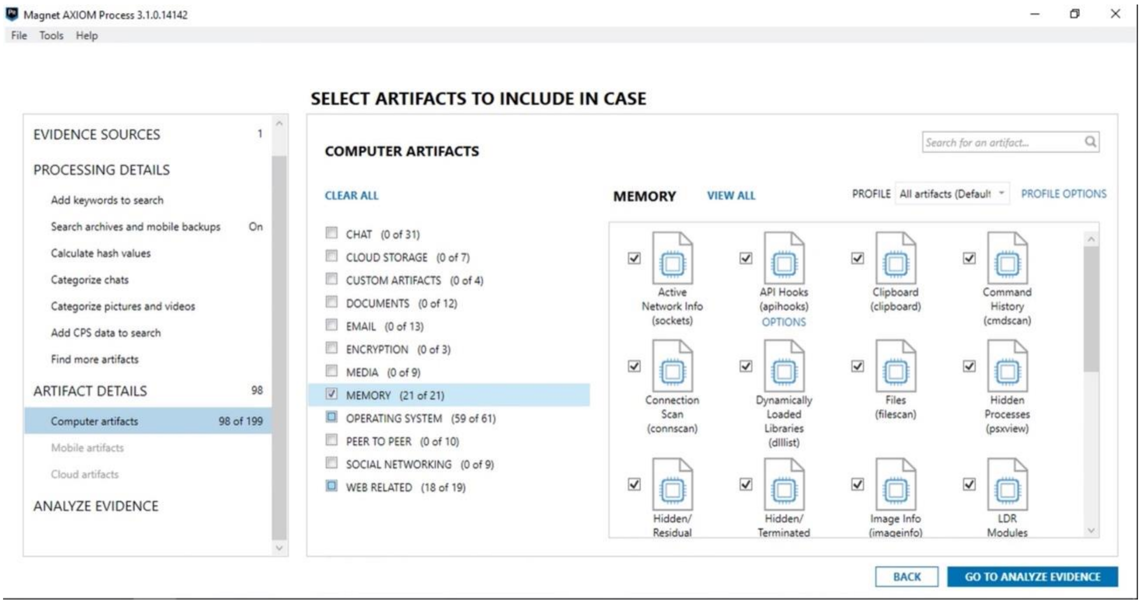
- Win10x64_10240_17770
- Win10x64_10586
- Win10x64_14393
- Win10x64_15063
- Win10x64_16299
- Win10x64_17134
- Win10x86
- Win10x86_10240_17770
- Win10x86_10586
- Win10x86_14393
- Win10x86_15063

No recomme memory artif . Searching this file may not yield any a valid memory dump file.

Memory artifacts powered by Volatility

As has been seen with Rekall and Volatility, before, the Bromium system runs with version 17763 and consequently a profile for 17134 it too old and cannot be used for analysis.

Nevertheless, just out of interest, further function in Axiom have been looked at:



Magnet AXIOM Process 3.1.0.14142

SELECT ARTIFACTS TO INCLUDE IN CASE

COMPUTER ARTIFACTS

Search for an artifact...

CLEAR ALL

- □ CHAT (0 of 31)
- □ CLOUD STORAGE (0 of 7)
- □ CUSTOM ARTIFACTS (0 of 4)
- □ DOCUMENTS (0 of 12)
- □ EMAIL (0 of 13)
- □ ENCRYPTION (0 of 3)
- □ MEDIA (0 of 9)
- ☑ MEMORY (21 of 21)
- ▣ OPERATING SYSTEM (59 of 61)
- □ PEER TO PEER (0 of 10)
- □ SOCIAL NETWORKING (0 of 9)
- ▣ WEB RELATED (18 of 19)

MEMORY

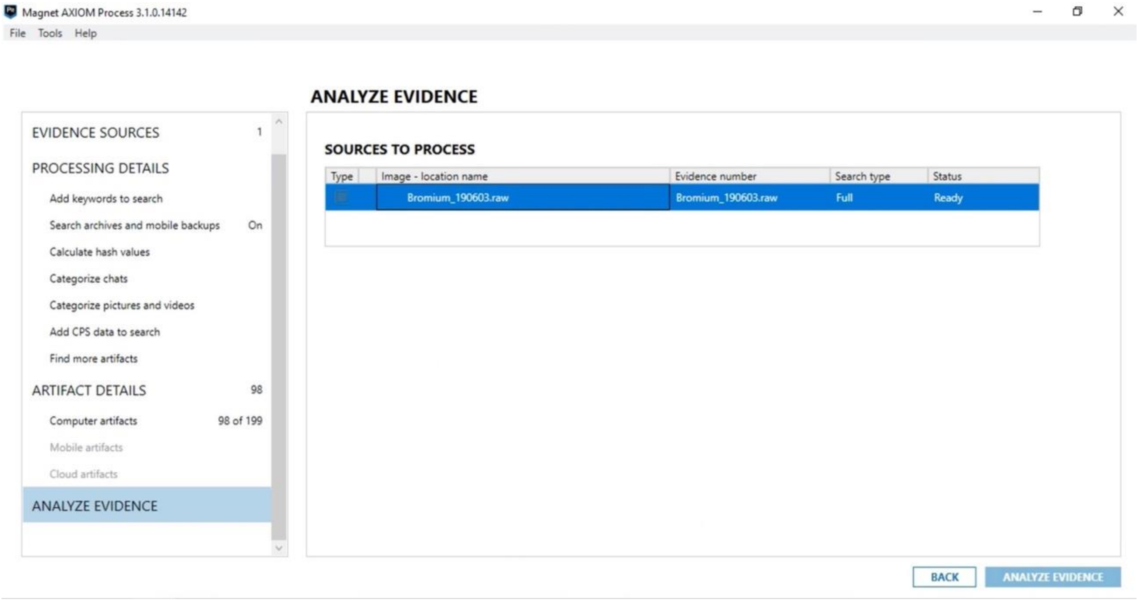
VIEW ALL

PROFILE: All artifacts (Default)

PROFILE OPTIONS

- ☑ Active Network Info (sockets)
- ☑ API Hooks (apihooks) OPTIONS
- ☑ Clipboard (clipboard)
- ☑ Command History (cmdscan)
- ☑ Connection Scan (connscan)
- ☑ Dynamically Loaded Libraries (dlllist)
- ☑ Files (filescan)
- ☑ Hidden Processes (psxview)
- ☑ Hidden/ Residual
- ☑ Hidden/ Terminated
- ☑ Image Info (imageinfo)
- ☑ LDR Modules

And as expected, Axiom refused to analyze the memory image with the given profile (the "ANALYZE EVIDENCE" button is greyed out):



ANALYZE EVIDENCE

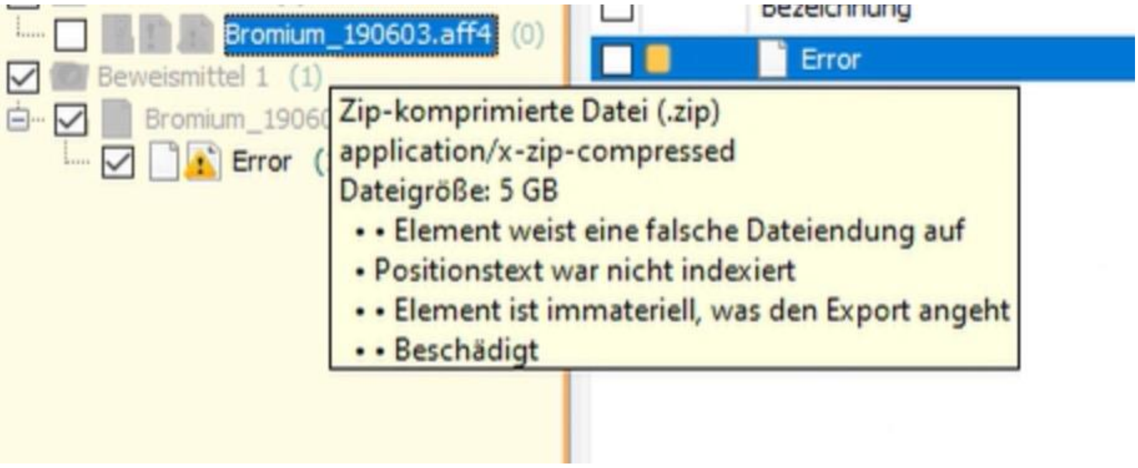
EVIDENCE SOURCES 1
PROCESSING DETAILS
Add keywords to search
Search archives and mobile backups On
Calculate hash values
Categorize chats
Categorize pictures and videos
Add CPS data to search
Find more artifacts
ARTIFACT DETAILS 98
Computer artifacts 98 of 199
Mobile artifacts
Cloud artifacts
ANALYZE EVIDENCE

SOURCES TO PROCESS

Type	Image - location name	Evidence number	Search type	Status
	Bromium_190603.raw	Bromium_190603.raw	Full	Ready

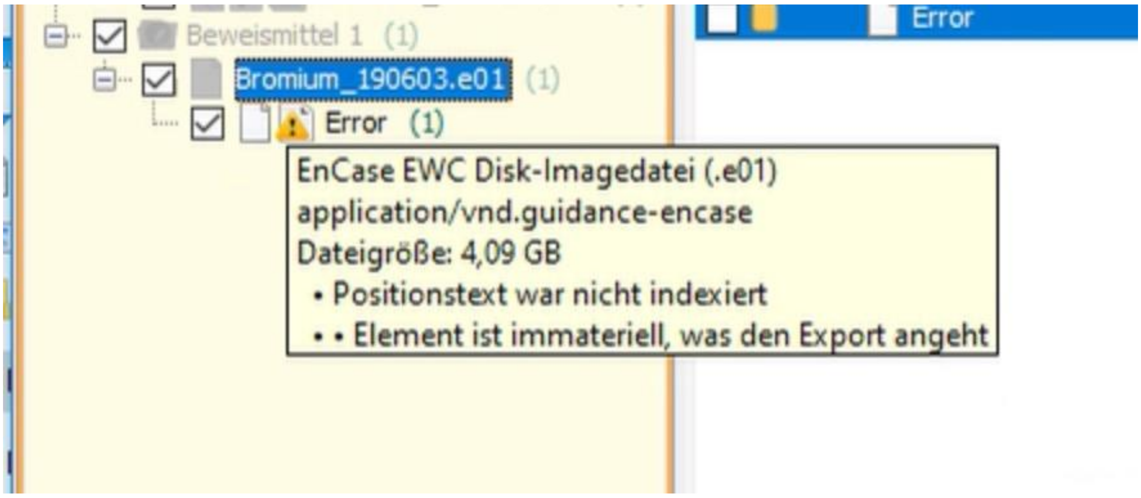
G.2.5 Testing with Nuix

First it looked, as if NUIX might be able to accept a Bromium memory image as evidence. But looking closer, all of the memory images were interpreted as files of disk images and it does not look, as if NUIX in the given version provides any memory analysis capabilities, at all:



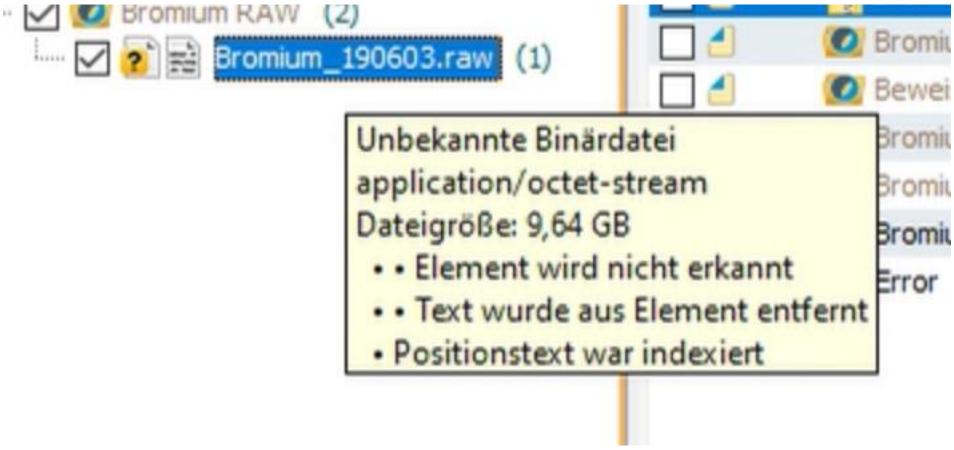
Zip-komprimierte Datei (.zip)
application/x-zip-compressed
Dateigröße: 5 GB

- • Element weist eine falsche Dateiendung auf
- Positionstext war nicht indexiert
- • Element ist immateriell, was den Export angeht
- • Beschädigt



EnCase EWC Disk-Imagedatei (.e01)
application/vnd.guidance-encase
Dateigröße: 4,09 GB

- Positionstext war nicht indexiert
- • Element ist immateriell, was den Export angeht



Unbekannte Binärdatei
application/octet-stream
Dateigröße: 9,64 GB

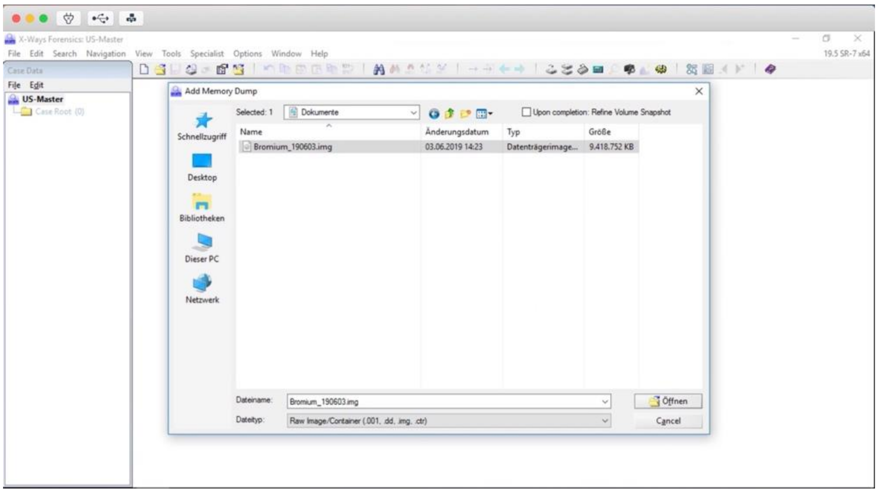
- • Element wird nicht erkannt
- • Text wurde aus Element entfernt
- Positionstext war indexiert

G.2.6 Testing with X-Ways

The X-Ways system at the university's test LAB has been looked at in version 19.5 and as was expected from the X-Ways web page, it was not capable ofinterpreting any memory image taken on the Bromium system, as this is based on Windows10.

Just to make sure, this was no incorrect assumtion, based on a possibly outdated web documentation, it has also been tested to load the Bromium memory images in EWF and raw format.

Loading the memory dump in raw format with the "add memory dump" function:



X-Ways Forensics: US-Master — 19.5 SR-7 x64

Add Memory Dump

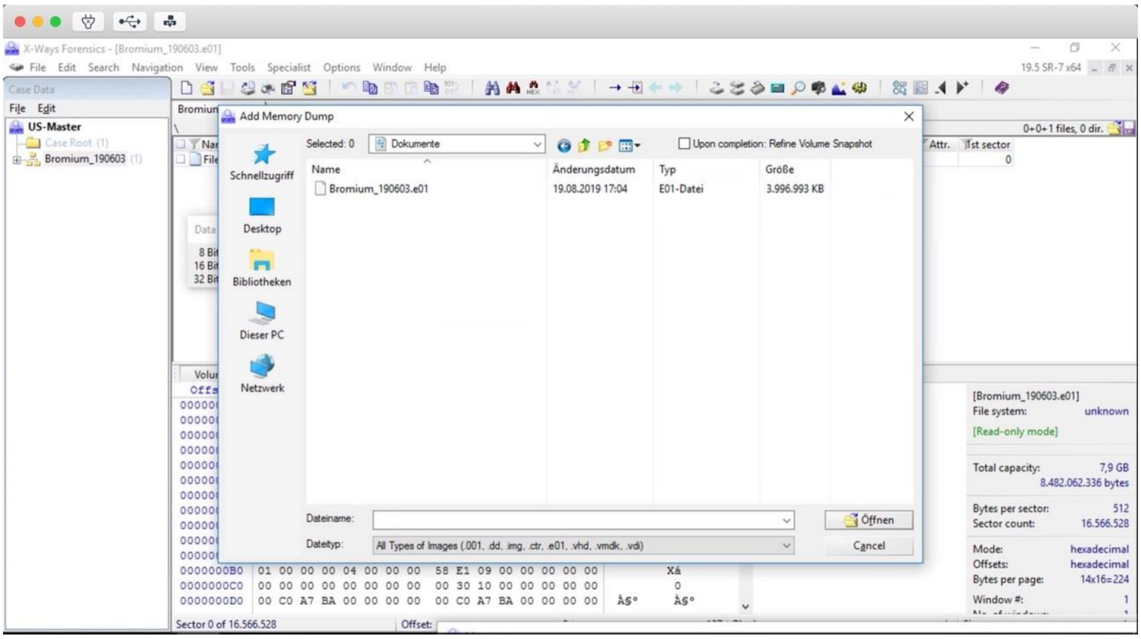
Selected: 1 — Dokumente — □ Upon completion: Refine Volume Snapshot

Name	Änderungsdatum	Typ	Größe
Bromium_190603.img	03.06.2019 14:23	Datenträgerimage...	9.418.752 KB

Dateiname: Bromium_190603.img

Dateityp: Raw Image/Container (.001, .dd, .img, .ctr)

Loading the memory dump in EWF format (e01):



X-Ways Forensics - [Bromium_190603.e01] — 19.5 SR-7 x64

Add Memory Dump

Selected: 0 — Dokumente — □ Upon completion: Refine Volume Snapshot

Name	Änderungsdatum	Typ	Größe
Bromium_190603.e01	19.08.2019 17:04	E01-Datei	3.996.993 KB

Dateiname:

Dateityp: All Types of Images (.001, .dd, .img, .ctr, .e01, .vhd, .vmdk, .vdi)

```
000000B0  01 00 00 00 04 00 00 00  58 E1 09 00 00 00 00 00      Xá
000000C0  00 00 00 00 00 00 00 00  00 30 10 00 00 00 00 00      0
000000D0  00 C0 A7 BA 00 00 00 00  00 C0 A7 BA 00 00 00 00  À§°  À§°
```

Sector 0 of 16.566.528

0+0+1 files, 0 dir.

[Bromium_190603.e01]
File system: unknown
[Read-only mode]

Total capacity: 7,9 GB
8.482.062.336 bytes

Bytes per sector: 512
Sector count: 16.566.528

Mode: hexadecimal
Offsets: hexadecimal
Bytes per page: 14x16=224

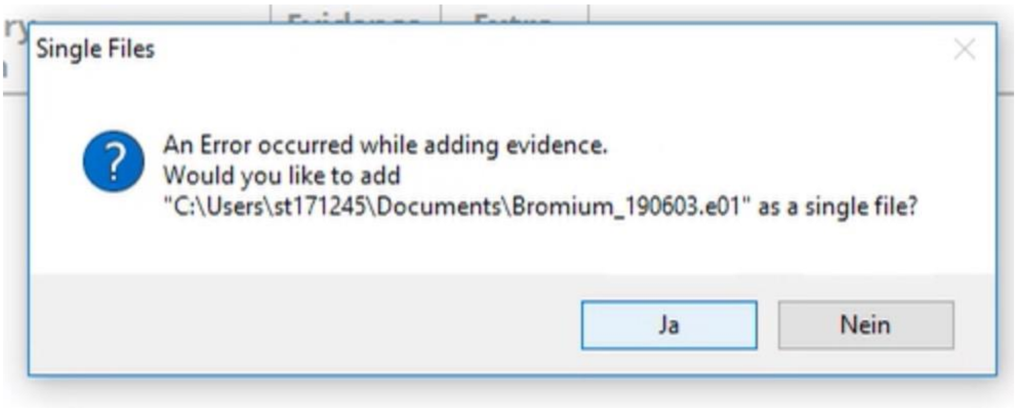
Window #: 1

As can be seen from the icon (left of "Bromium_190603"), the evidence is interpreted as file/disk image and not as memory.

G.2.7 Testing with EnCase

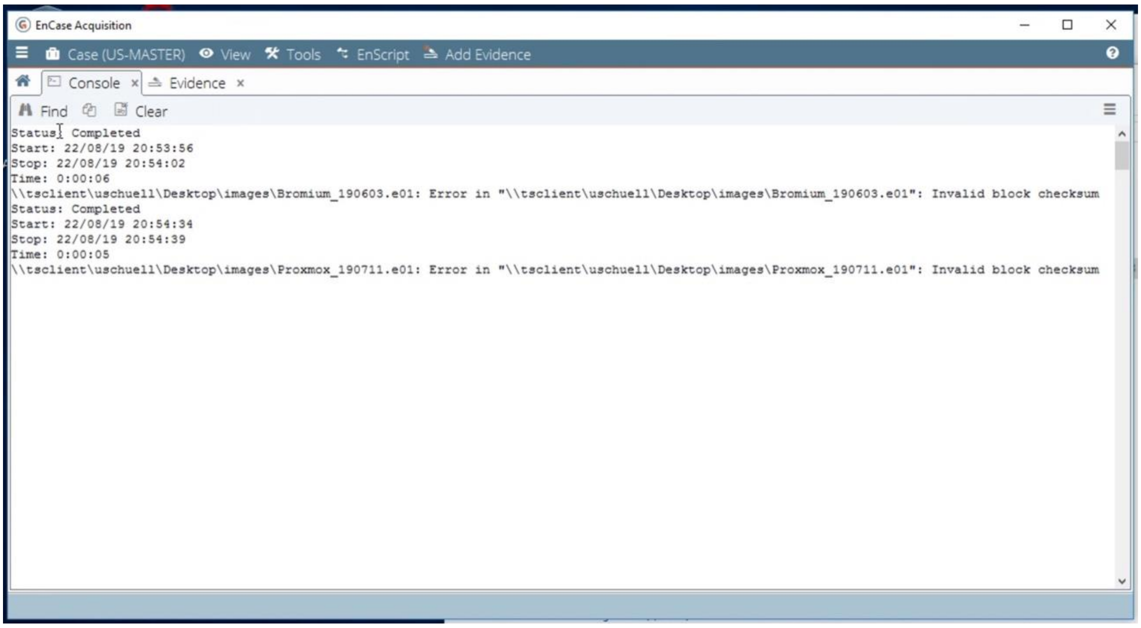
Using Encase has been tested with the only possibly matching format (EWF) and the corresponding file: Bromium_190603.e01 .

After a short message:



An Error occurred while adding evidence.
Would you like to add
"C:\Users\st171245\Documents\Bromium_190603.e01" as a single file?

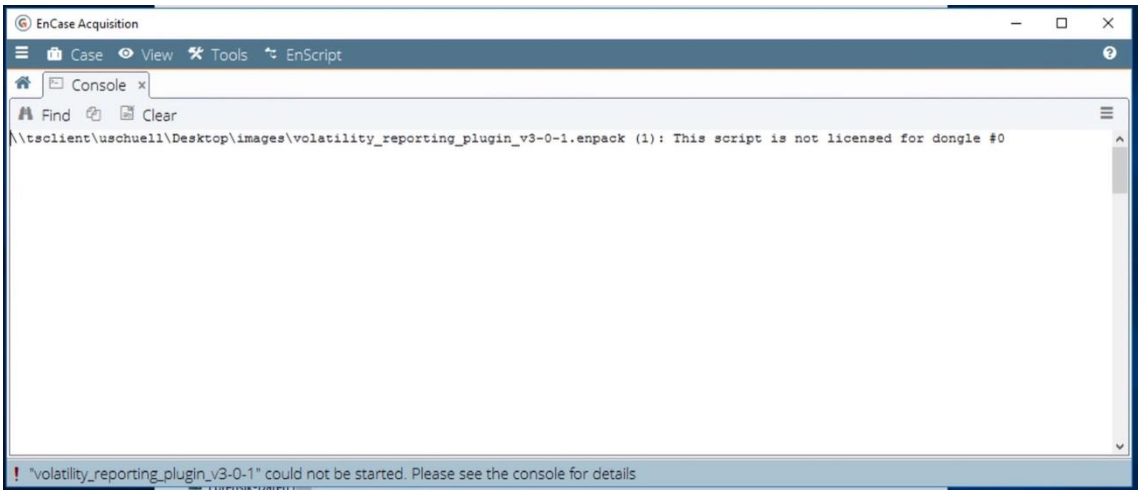
it was possible to load the memory image, but image verification failed:



```
Status: Completed
Start: 22/08/19 20:53:56
Stop: 22/08/19 20:54:02
Time: 0:00:06
\\tsclient\uschuell\Desktop\images\Bromium_190603.e01: Error in "\\tsclient\uschuell\Desktop\images\Bromium_190603.e01": Invalid block checksum
Status: Completed
Start: 22/08/19 20:54:34
Stop: 22/08/19 20:54:39
Time: 0:00:05
\\tsclient\uschuell\Desktop\images\Proxmox_190711.e01: Error in "\\tsclient\uschuell\Desktop\images\Proxmox_190711.e01": Invalid block checksum
```

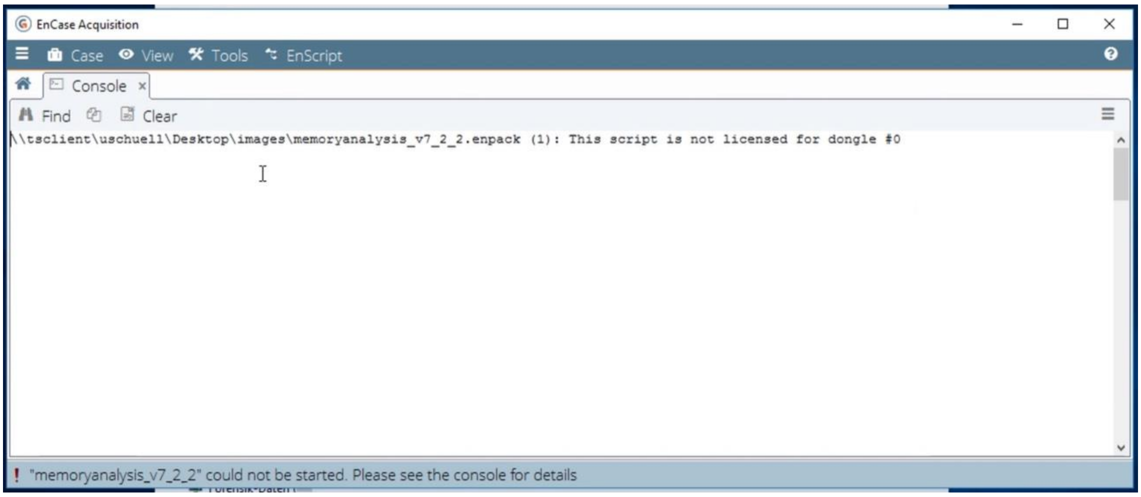
with an "Invalid block checksum" error.

And trying to use the possible NUIX additions for Volatility or Memory Analysis, that have been found on the NUIX home page, failed with license messages, anyway:



```
\\tsclient\uschuell\Desktop\images\volatility_reporting_plugin_v3-0-1.enpack (1): This script is not licensed for dongle #0
```

! "volatility_reporting_plugin_v3-0-1" could not be started. Please see the console for details



```
\\tsclient\uschuell\Desktop\images\memoryanalysis_v7_2_2.enpack (1): This script is not licensed for dongle #0
```

! "memoryanalysis_v7_2_2" could not be started. Please see the console for details

G.3 Qubes

G.3.1 Test-Setup

For testing, Qubes has been installed in Version 4. Details about the installation are available in appendix A.2 and in appendix F.

To make sure, that VMs are up and running, for testing, a Web-browser has been started in the personal qube, pointing to the wikipedia "dolimites" page.

G.3.2 Testing with Rekall

Neither in live-mode nor with a memory image, Rekall was able to work correctly on Qubes:

```
[root@dom0 rekall]# bin/rekall -f
/root/images/Qubes_190702_rekall_aff4acquire.aff4 \
-p /root/Qubes-4.14.123-1.json pslist
"A DTB value was found but failed to verify"
```

```
[root@dom0 rekall]# bin/rekall --live Memory \
-p /root/Qubes-4.14.123-1.json pslist
"A DTB value was found but failed to verify"
```

G.3.3 Testing with Volatility:

```
[root@dom0 ~]# ./volatility_2.6_lin64_standalone --plugins=/root/vol-
profiles -f /root/images/Qubes_190702_rekall_aff4acquire.raw imageinfo
Volatility Foundation Volatility Framework 2.6
INFO    : volatility.debug    : Determining profile based on KDBG search...
Traceback /most recent call last):
  File "vol.py", line 192, in <module>
  File "vol.py", line 183, in main
  File "volatility/commands.py", line 147, in execute
  File "volatility/plugins/imageinfo.py", line 45, in render_text
  File "volatility/plugins/imageinfo.py", line 55, in calculate
  File "volatility/plugins/kdbgscan.py", line 116, in calculate
  File "volatility/addrspace.py", line 378, in __init__
  File "volatility/addrspace.py", line 73, in __init__
  File "volatility/addrspace.py", line 98, in set_profile
  File "volatility/plugins/overlays/linux/linux.py", line 216, in __init__
  File "volatility/obj.py", line 862, in __init__
  File "volatility/plugins/overlays/linux/linux.py", line 227, in reset
  File "volatility/plugins/overlays/linux/linux.py", line 264, in load_vtypes
  File "volatility/dwarf.py", line 71, in __init__
  File "volatility/dwarf.py", line 162, in feed_line
  File "volatility/dwarf.py", line 204, in process_statement

KeyError: 'DW_AT_byte_size'
Failed to execute script vol
```

```
[root@dom0 ~]# ./volatility_2.6_lin64_standalone --plugins=/root/vol-
profiles -f /root/images/Qubes_190702_rekall_aff4acquire.raw pslist
Volatility Foundation Volatility Framework 2.6
No suitable address space mapping found
Tried to open image as:
 MachOAddressSpace: mac: need base
 LimeAddressSpace: lime: need base
 WindowsHiberFileSpace32: No base Address Space
 WindowsCrashDumpSpace64BitMap: No base Address Space
 VMWareMetaAddressSpace: No base Address Space
 WindowsCrashDumpSpace64: No base Address Space
 HPAKAddressSpace: No base Address Space
 VirtualBoxCoreDumpElf64: No base Address Space
 VMWareAddressSpace: No base Address Space
 QemuCoreDumpElf: No base Address Space
 WindowsCrashDumpSpace32: No base Address Space
 Win10AMD64PagedMemory: No base Address Space
 LinuxAMD64PagedMemory: No base Address Space
 AMD64PagedMemory: No base Address Space
 IA32PagedMemoryPAE: No base Address Space
 IA32PagedMemory: No base Address Space
 OSXPmemELF: No base Address Space
 MachOAddressSpace: MachO Header signature invalid
```

```
LimeAddressSpace: invalid Lime Header signature
WindowsHiberFileSpace32: No xpress signature found
WindowsCrashDumpSpace64BitMap: Header signature invalid
VMWareMetaAddressSpace: VMware metadata file is not valid
WindowsCrashDumpSpace64: Header signature invalid
HPAKAddressSpace: Invalid magiv found
VirtualBoxCoreDumpElf64: ELF Header signature invalid
VMWareAddressSpace: Invalid VMware signature: 0x0
QemuCoreDumpElf: ELF Header signature invalid
WindowsCrashDumpSpace32: Header signature invalid
Win10AMD64PagedMemory: Incompatible profile WinXPSP2c86 selected
WindowsAMD64PagedMemory: Incompatible profile WinXPSP2c86 selected
LinuxAMD64PagedMemory: Incompatible profile WinXPSP2c86 selected
AMD64PagedMemory: Incompatible profile WinXPSP2c86 selected
IA32PagedMemoryPAE: No valid DTB found
IA32PagedMemory: No valid DTB found
OSXPmemELF: ELF Headert signature invalid
FileAddressSpace: Must be first Address Space
ArmAddressSpace: No valid DTB found
```

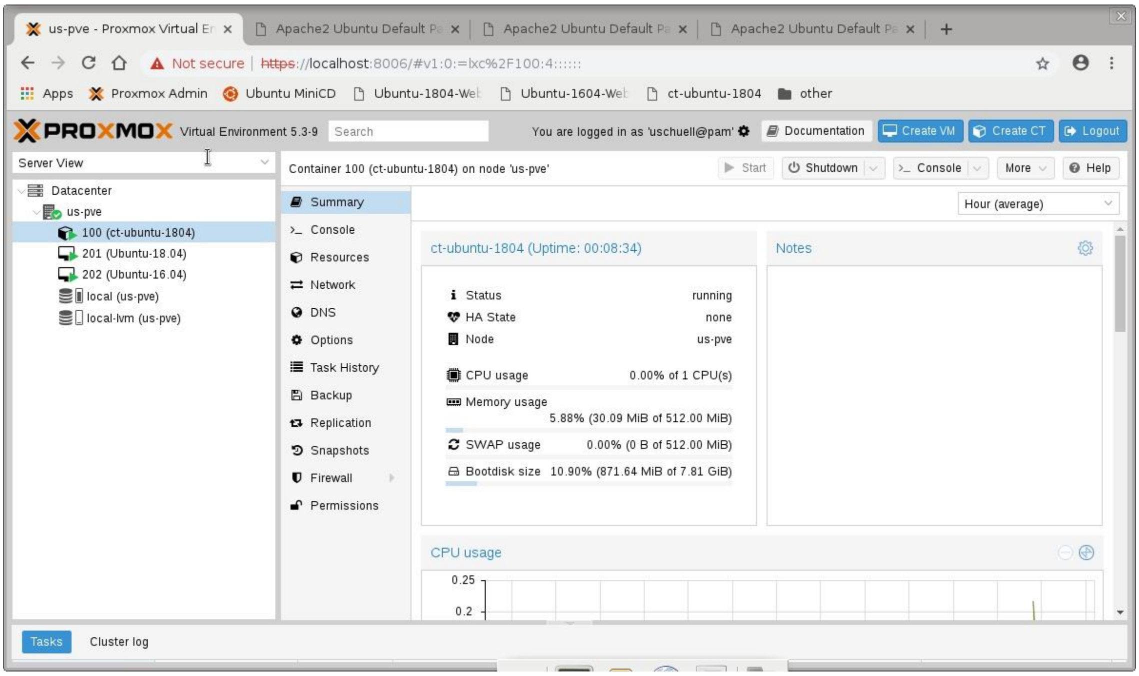
G.4 Proxmox

G.4.1 Test-Setup

For testing, Proxmox has been installed in version 5.3, as described in appendix A.3 .

To have virtual machines and containers running, two VMs, with Ubuntu 16.04 and 18.04 and one container, with Ubuntu 18.04 have been configured, all of then running Apache Web-Servers (with the main web page showing an identifying message), so, that connecting to the different Web-Server-Addresses can show, both VMs and the container are running.

This view of Proxmox Admin GUI shows the running Container:



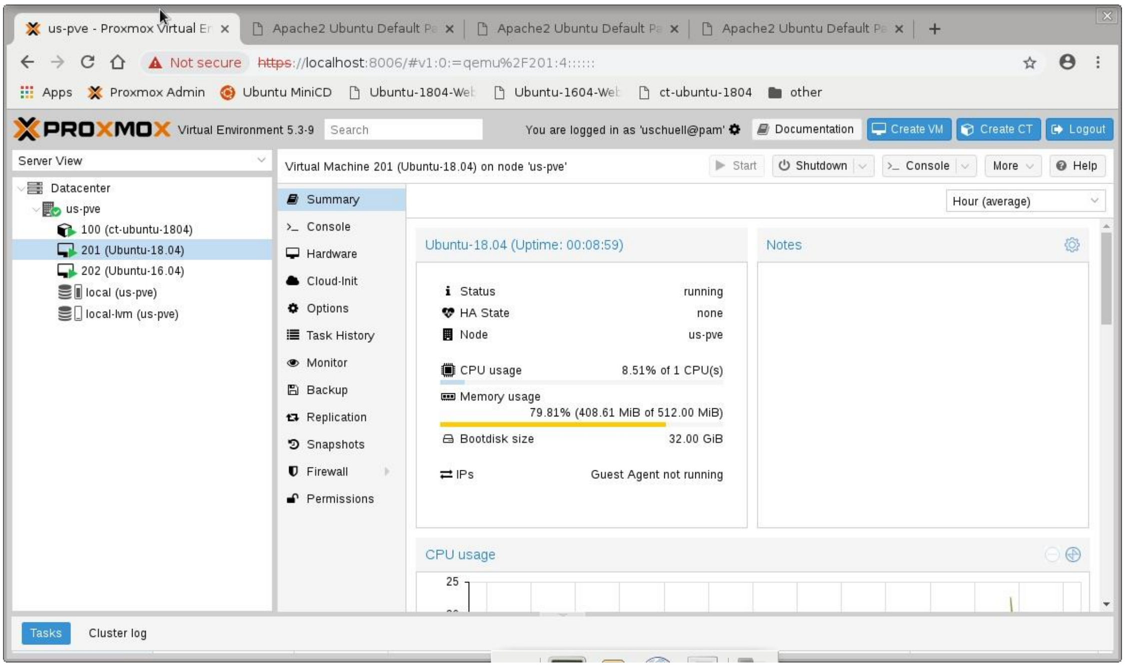
Not secure | https://localhost:8006/#v1:0:=lxc%2F100:4:::::

Container 100 (ct-ubuntu-1804) on node 'us-pve'

ct-ubuntu-1804 (Uptime: 00:08:34)

Status running
HA State none
Node us-pve
CPU usage 0.00% of 1 CPU(s)
Memory usage 5.88% (30.09 MiB of 512.00 MiB)
SWAP usage 0.00% (0 B of 512.00 MiB)
Bootdisk size 10.90% (871.64 MiB of 7.81 GiB)

Proxmox Admin GUI showing the running Ubuntu 18.04 VM:



https://localhost:8006/#v1:0:=qemu%2F201:4::::::

Virtual Environment 5.3-9

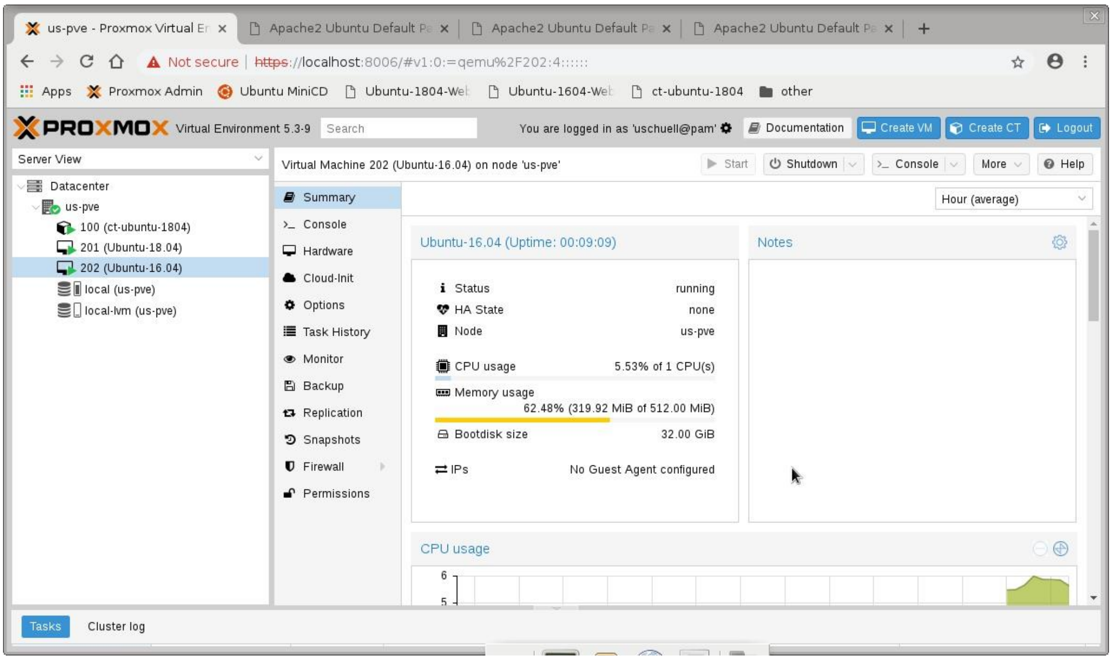
You are logged in as 'uschuell@pam'

Virtual Machine 201 (Ubuntu-18.04) on node 'us-pve'

Ubuntu-18.04 (Uptime: 00:08:59)

Status	running
HA State	none
Node	us-pve
CPU usage	8.51% of 1 CPU(s)
Memory usage	79.81% (408.61 MiB of 512.00 MiB)
Bootdisk size	32.00 GiB
IPs	Guest Agent not running

Proxmox Admin GUI showing the running Ubuntu 16.04 VM:



https://localhost:8006/#v1:0:=qemu%2F202:4::::::

Virtual Environment 5.3-9

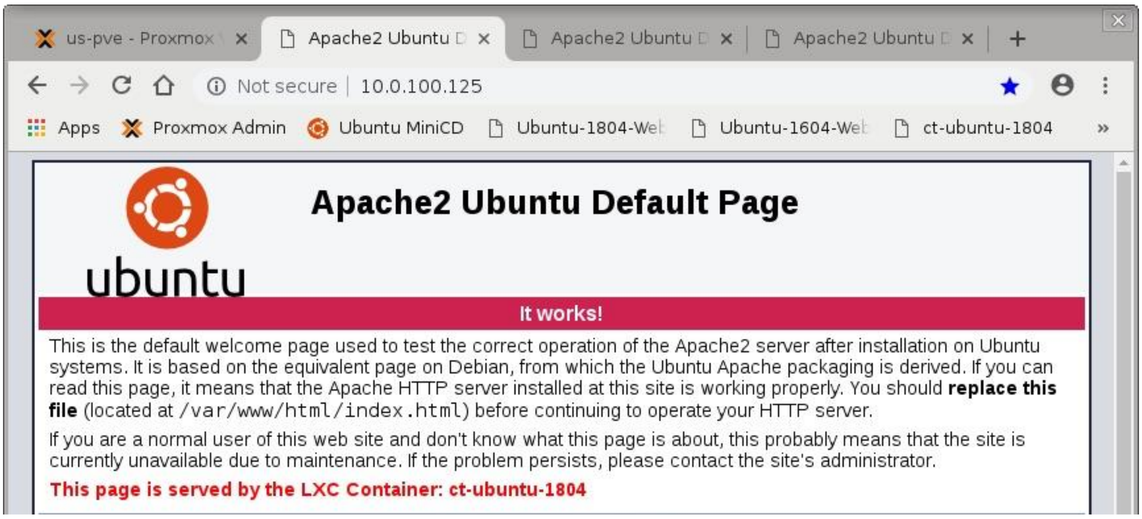
You are logged in as 'uschuell@pam'

Virtual Machine 202 (Ubuntu-16.04) on node 'us-pve'

Ubuntu-16.04 (Uptime: 00:09:09)

Status	running
HA State	none
Node	us-pve
CPU usage	5.53% of 1 CPU(s)
Memory usage	62.48% (319.92 MiB of 512.00 MiB)
Bootdisk size	32.00 GiB
IPs	No Guest Agent configured

Chromium Tab connected to the web server on the LXC container:



Apache2 Ubuntu Default Page

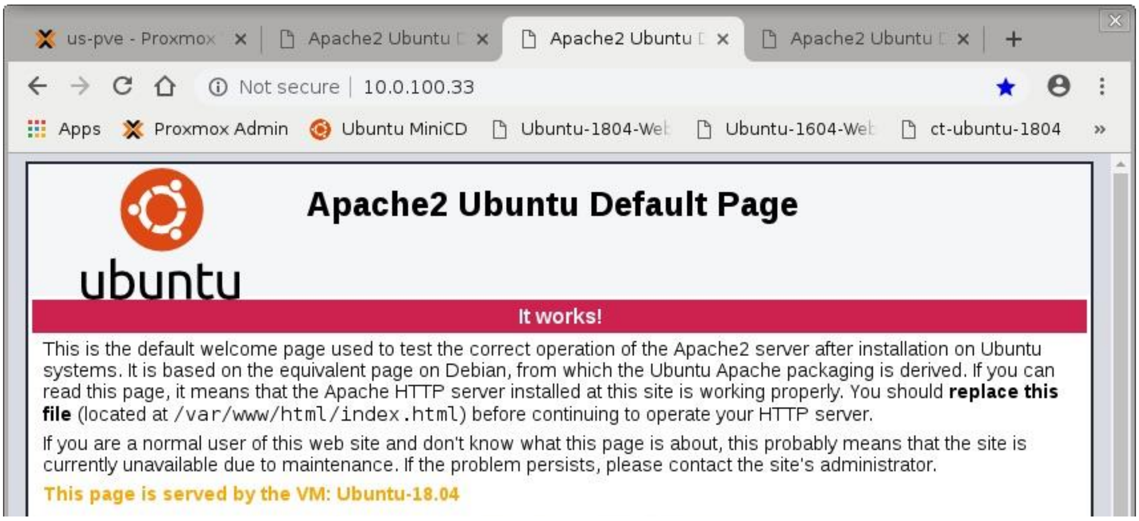
It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

This page is served by the LXC Container: ct-ubuntu-1804

Chromium Tab, connected to the web server on the Ubuntu 18.04 VM:



Apache2 Ubuntu Default Page

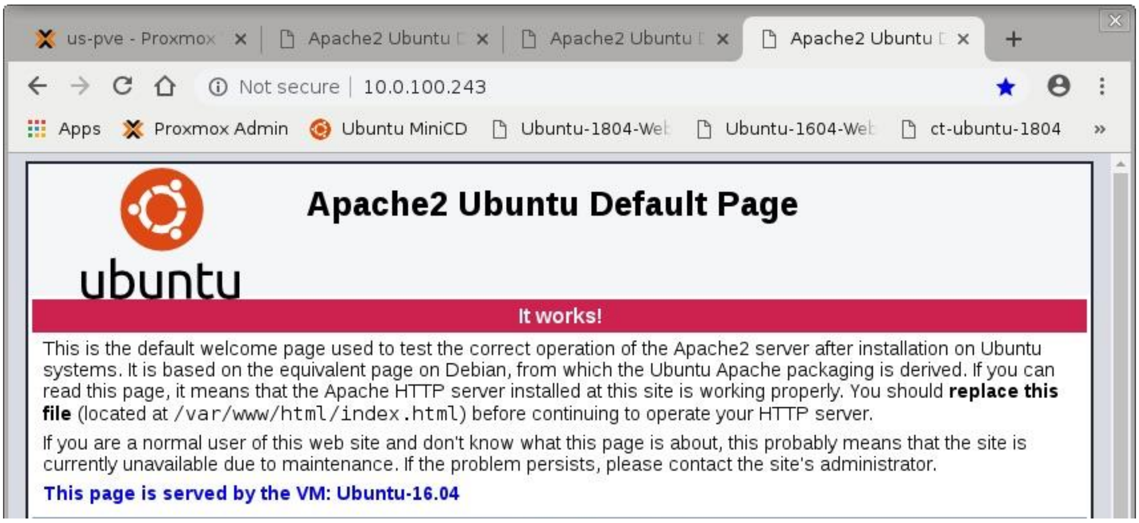
It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

This page is served by the VM: Ubuntu-18.04

Chromium Tab, connected to the web server on the Ubuntu 16.04 VM:



Apache2 Ubuntu Default Page

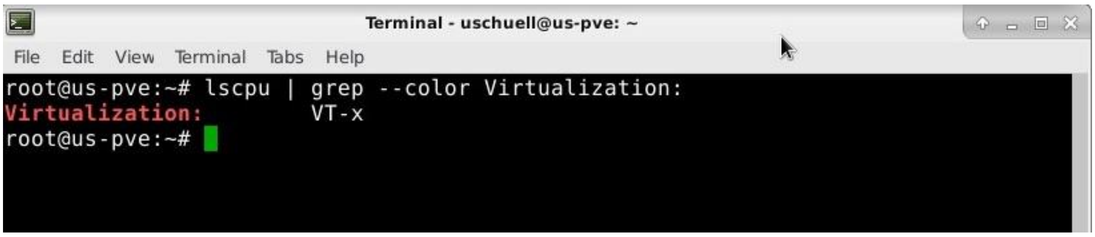
It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

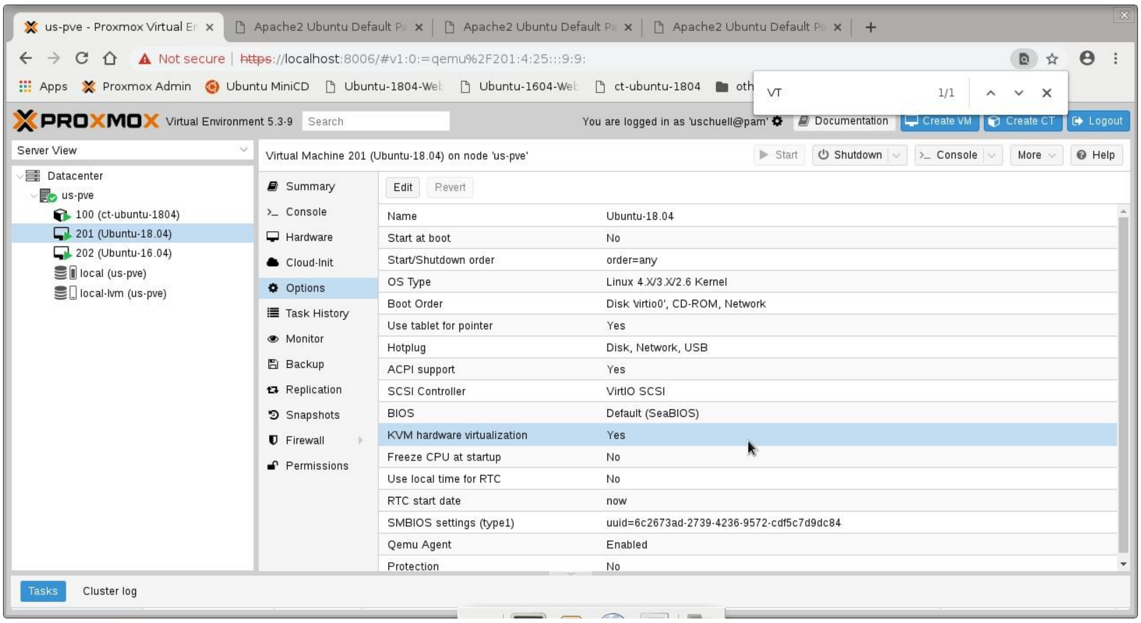
This page is served by the VM: Ubuntu-16.04

It has also been verified, that VT-x is active on the host:

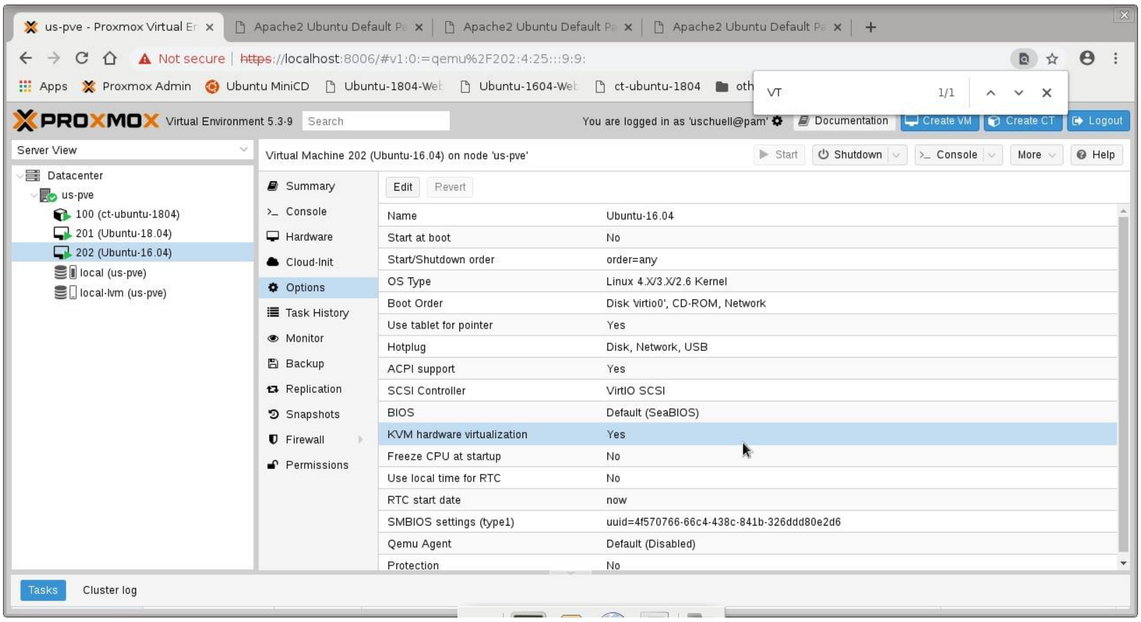

Terminal - uschuell@us-pve: ~
File Edit View Terminal Tabs Help

```
root@us-pve:~# lscpu | grep --color Virtualization:
Virtualization:        VT-x
root@us-pve:~#
```

as well as on the KVM based VMs:



Name	Ubuntu-18.04
Start at boot	No
Start/Shutdown order	order=any
OS Type	Linux 4.X/3.X/2.6 Kernel
Boot Order	Disk 'virtio0', CD-ROM, Network
Use tablet for pointer	Yes
Hotplug	Disk, Network, USB
ACPI support	Yes
SCSI Controller	VirtIO SCSI
BIOS	Default (SeaBIOS)
KVM hardware virtualization	Yes
Freeze CPU at startup	No
Use local time for RTC	No
RTC start date	now
SMBIOS settings (type1)	uuid=6c2673ad-2739-4236-9572-cdf5c7d9dc84
Qemu Agent	Enabled
Protection	No



Name	Ubuntu-16.04
Start at boot	No
Start/Shutdown order	order=any
OS Type	Linux 4.X/3.X/2.6 Kernel
Boot Order	Disk 'virtio0', CD-ROM, Network
Use tablet for pointer	Yes
Hotplug	Disk, Network, USB
ACPI support	Yes
SCSI Controller	VirtIO SCSI
BIOS	Default (SeaBIOS)
KVM hardware virtualization	Yes
Freeze CPU at startup	No
Use local time for RTC	No
RTC start date	now
SMBIOS settings (type1)	uuid=4f570766-66c4-438c-841b-326ddd80e2d6
Qemu Agent	Default (Disabled)
Protection	No

G.4.2 Testing with Rekall

```
# rekall aff4quire Proxmox_190711_mem.aff4
```

```
# rekall -f Proxmox_190711_mem.aff4 imagecopy \
  --output-image="Proxmox_190711_mem.raw"
```

```
# rekall -f Proxmox_190711_mem.aff4 vmscan
```

=> Unfortunately, the vmscan does not return any EPT values.

Apart from *vmscan*, other plugins, like *pslist* or *filescan*, that inspect the host system, work fine, with Rekall in Proxmox.

G.4.3 Testing with Rekall in SIFT

on the SIFT host system:

copy Proxmox-pve.json, Proxmox_190711_mem.aff4 and Proxmox_190711_mem.raw to the SHARED folder

start the SIFT workstation in Virtual box

on the SIFT workstation:

```
# sudo -i
```

```
# rekall -f /SHARED/Proxmox_190711_mem.aff4 -p /SHARED/Proxmox-pve.json
```

```
[1] Proxmox_190711_mem.aff4 12:19:53> vmscan
```

=> does not find any VMs (despite the fact, there are the 2 running VMs with Ubuntu 1804 and 1604 and the Container with Ubuntu 18.04)

```
[1] Proxmox_190711_mem.aff4 12:36:52> pstree
```

=> shows/finds all the running processes

```
[1] Proxmox_190711_mem.aff4 12:38:36> pslist
```

=> shows the processes, as well, only the timestamps are 1970...

List of commands and whether they return any data:

Command	**works/fails**
vmscan	no data
pstree	works
pslist	works (with 1970 timestamps)
version_scan	no data
lsof	works (lots of data)
netstat	no data
arp	works
bash	works

cpuinfo	fails
hostname	works
iomem	works
ifconfig	works
dmesg	works
find_dtb	works (returns: 0x00004fe0a000)
vadmap	works
lsmod	works

G.4.4 Testing with Rekall on the Bromium System

Running Rekall with a Proxmox-image on Bromium:

```
(rekal) C:\DFIR\rekall>rekall -f d:\Proxmox_190711_mem.aff4 vmscan
No handlers could be found for logger "rekall.plugins.tools"
Description                                          Type          Valid
EPT
---------------------------------------- -------------------- --------
```

=> This generates no output; as on Proxmox, itself, Rekall on Windows is not able to detect any VMs in the Proxmox memory image.

```
(rekal) C:\DFIR\rekall>rekall -f d:\Proxmox_190711_mem.aff4 ewfacquire
d:\Proxmox_190711.e01
No handlers could be found for logger "rekall.plugins.tools"
 Writing XXXXMb

(rekal) C:\DFIR\rekall>rekall -f d:\Proxmox_190711_mem.aff4 imagecopy
--output-image="Proxmox_190711.raw"
No handlers could be found for logger "rekall.plugins.tools"
 Writing offset X.XX GB
(rekal) C:\DFIR\rekall>move Proxmox_190711.raw d:\

(rekal) C:\DFIR\rekall>dir d:\Proxmox_190711*
 Volume in drive D is PROXMOX
 Volume Serial Number is 5D27-6CCA

 Directory of d:\

11.07.2019  19:10     5.234.028.741 Proxmox_190711_mem.aff4
11.07.2019  19:18     9.644.802.048 Proxmox_190711_mem.raw
```

```
21.08.2019  16:27     4.635.277.721 Proxmox_190711.e01
21.08.2019  16:47     9.644.802.048 Proxmox_190711.raw
21.08.2019  18:16             1.475 Proxmox_190711.txt
               5 File(s) 29.158.912.033 bytes
               0 Dir(s)  23.838.195.712 bytes free
```

G.4.5 Testing with Volatility on SIFT

```
# cp Debian9_proxmox.zip \
  /root/volatility/volatility/plugins/overlays/linux/
# python volatility/vol.py -f \
  /SHARED/Proxmox_190711_mem.raw --info | grep Linux
```

=> shows: LinuxDebian9-Proxmoxx64

```
# python volatility/vol.py -f \
  /SHARED/Proxmox_190711_mem.raw \
  --profile=LinuxDebian9-Proxmoxx64 linux_psaux \
  > pslist.txt
# grep -e Pid -e kvm -e lxc pslist.txt \
  > /SHARED/psaux_lxc_kvm.txt

# cat /SHARED/psaux_lxc_kvm.txt
Pid    Uid    Gid    Arguments

1021   0      0      /usr/bin/lxcfs /var/lib/lxcfs/
1183   0      0      /usr/lib/x86_64-linux-gnu/lxc/lxc-monitord
                      --daemon
16466  0      0      [lxc monitor] /v3ar/lib/lxc 100
17038  0      0      /usr/bin/dtach -A /var/run/dtach/vzctlconsole100
                     -r winch -z lxc-console -n 100 -e -1
17039  0      0      lxc-console -n 100 -e 1
-----
12761  0      0      /usr/bin/kvm -id 201 -name Ubuntu-18.04 ...
12794  0      0      /usr/bin/kvm -id 201 -name Ubuntu-18.04 ...
12797  0      0      [kvmpit/12761]

-----
28567  0      0      /usr/bin/kvm -id 202 -name Ubuntu-16.04 ...
28599  0      0      /usr/bin/kvm -id 202 -name Ubuntu-16.04 ...
28602  0      0      [kvmpit/28567]


# python volatility/vol.py \
  -f /SHARED/Proxmox_190711_mem.raw \
  --profile=LinuxDebian9-Proxmoxx64 linux_pslist \
  > pslist2.txt
```

```
# grep -e Pid -e kvm -e lxc pslist2.txt \
  > /SHARED/pslist_lxc_kvm.txt
Offset             Name                 Pid             PPid
  Uid        Gid    DTB                Start Time
0xffff8908700716c0 lxcfs                1021            1
  0          0      0x00000002303fa000 2019-07-09 14:39:00 UTC+0000
0xffff89086d828000 lxc-monitord         1183            1
  0          0      0x000000022d91a000 2019-07-09 14:39:00 UTC+0000
0xffff89078d88c440 lxc-start            16466           1
  0          0      0x00000002032dc000 2019-07-11 14:05:00 UTC+0000
0xffff8907c0048000 lxc-console          17039           17038
  0          0      0x000000022cc3c000 2019-07-11 14:05:12 UTC+0000
------------------
0xffff89081c558000 kvm                  12761           1
  0          0      0x000000019c758000 2019-07-09 16:19:48 UTC+0000
0xffff89082e8b2d80 kvm-pit/12761        12797           2
  0          0      ------------------ 2019-07-09 16:19:48 UTC+0000
------------------
0xffff89081c69c440 kvm                  28567           1
  0          0      0x000000012808c000 2019-07-11 14:42:04 UTC+0000
0xffff89086c21c440 kvm-pit/28567        28602           2
  0          0      ------------------ 2019-07-11 14:42:04 UTC+0000
```