Digital Investigation 10 (2013) S105–S115

Contents lists available at SciVerse ScienceDirect

Digital Investigation

journal homepage: www.elsevier.com/locate/diin

Anti-forensic resilient memory acquisition

Johannes Stüttgen [a,*], Michael Cohen [b]

[a] Department of Computer Science, Friedrich-Alexander University of Erlangen-Nuremberg, Martensstraße 3, 91058 Erlangen, Germany
[b] Google Inc., Brandschenkestrasse 110, Zurich, Switzerland

Keywords:
Memory forensics
Memory acquisition
Anti forensics
Live forensics
Malware
Computer security
Information security
Incident response

A B S T R A C T

Memory analysis has gained popularity in recent years proving to be an effective technique for uncovering malware in compromised computer systems. The process of memory acquisition presents unique evidentiary challenges since many acquisition techniques require code to be run on a potential compromised system, presenting an avenue for anti-forensic subversion. In this paper, we examine a number of simple anti-forensic techniques and test a representative sample of current commercial and free memory acquisition tools. We find that current tools are not resilient to very simple anti-forensic measures. We present a novel memory acquisition technique, based on direct page table manipulation and PCI hardware introspection, without relying on operating system facilities - making it more difficult to subvert. We then evaluate this technique's further vulnerability to subversion by considering more advanced anti-forensic attacks.

1. Introduction

Since host-based memory forensics was first proposed, rapid advances in the analysis techniques for memory images have taken place. Modern tools are capable of extracting detailed information about system state, configuration, and anomalies. In particular, memory analysis has proven useful for the detection of rootkits and other malware infecting the host, as well as the analysis of malicious software.

As this analytical capability matures, applications are emerging for application of memory analysis in many contexts, such as remote forensics (Cohen et al., 2011), malware classification, and even self healing of compromised systems (Grizzard, 2006). Direct memory access can be used in the forensic context to obtain a complete point-in-time static forensic image or to enable an external memory analysis module to perform runtime live analysis. In this paper we refer to the process of accessing the physical memory as "memory acquisition", regardless of its intent.

Memory analysis is attractive for malware analysis as it is seen as a way to examine the system from an external and impartial point of view. While malware may attempt to hide by hooking operating system services (Florio, 2005), analysis of memory images offers the opportunity to examine the rootkit's hooks and code outside of the path of the ordinary operating system functionality.

Anti-Forensics has been broadly defined as "any attempt to compromise the availability or usefulness of evidence to the forensic process" (Harris, 2006). Thus anti-forensic attacks fall into two broad categories – those techniques which prevent evidence from being acquired, and those techniques which remove data from the collected evidence such that the collected evidence can not be suitably analyzed.

A number of effective anti-forensic techniques against memory acquisition have been proposed. Substitution attacks, in which data fabricated by the attacker is substituted in place of valid data during the acquisition process have been implemented (Bilby, 2006; Milkovic, 2012). Alternatively a rootkit might disrupt the acquisition process altogether (e.g. hang the hardware) when detecting the

* Corresponding author.
E-mail addresses: johannes.stuettgen@cs.fau.de (J. Stüttgen), scudette@google.com (M. Cohen).

1742-2876/$ – see front matter © 2013 Johannes Stüttgen and Michael Cohen. Published by Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.diin.2013.06.012

presence of a forensic agent. This approach is especially effective against memory acquisition, since the volatility of the evidence does not permit the investigator to reacquire the memory under the same conditions.

Although there have been efforts to test memory forensic acquisition tools (Inoue et al., 2011), and even solidify the criteria by which these tools can be tested (Carrier and Grand, 2004; Vömel and Freiling, 2012), robustness of the tools against anti-forensic interference is not yet explored during these tests (Wundram et al., 2013). Thus, while one can gain assurances about the forensic soundness of acquisition tools under ideal lab conditions, it is impossible to extrapolate this to acquisition of a hostile system, potentially employing anti-forensic techniques.

Due to lack of research and understanding of anti-forensic techniques in memory acquisition, current commercial or free memory acquisition tools do not appear to implement mechanisms to protect their operations against anti-forensic attacks. Due to the increasing popularity of these tools, there is currently an invigorated research interest in developing anti-forensic techniques specifically targeting these tools (Milkovic, 2012; Haruyama and Suzuki, 2012).

Related Work: A number of memory acquisition techniques have been proposed in the literature (Vömel and Freiling, 2011). In assessing the exposure of different memory acquisition techniques to anti-forensic subversion, we can broadly divide techniques into those which rely on the operating system software integrity and those who rely on the hardware.

Many operating systems already present a view of physical memory through a special device or kernel API. For example, in Windows the operating system presents the section object `\\.\PhysicalMemory`, to allow reading from physical memory. Earlier memory acquisition tools directly opened this device from user-space (Garner, 2006). More recent versions of Windows deny direct access to the device from user space, necessitating a kernel driver to open the device from kernel space. A number of more direct kernel API routines are utilized in current tools, such as `MmMapIOSpace` and the undocumented `MmMapMemoryDumpMdl` (MoonSols, 2012).

Bypassing memory acquisition tools that depend on the operating system was demonstrated by the ddefy tool (Bilby, 2006). This tool hooks the physical memory device and filters certain pages from being read through this interface, providing instead a cached copy (prior to kernel modification). In principle, any OS facility can be hooked in a similar manner in order to subvert the acquisition tool. Additionally, many acquisition tools have a user mode process to write and process the image. This increases the attack surface of the tool, by allowing standard user space hooks to modify the memory image as it is written to disk (Milkovic, 2012).

Hardware-based solutions were proposed as being resilient to rootkit manipulation. Dedicated hardware can access the memory bus directly without CPU management (Carrier and Grand, 2004). It is even possible to re-purpose existing hardware to extract physical memory. For example, the Firewire hardware may be used for direct memory access (DMA) to the physical address space (Boileau, 2006).

Unfortunately, even hardware-based acquisition can be defeated using very low level manipulation of the memory controller's hardware registers (Rutkowska, 2007). By remapping some parts of the physical address space into an IO device, the CPU's view of this range is different from the hardware DMA view.

A more subversion resistant approach is taken by `BodySnatcher` (Schatz, 2007). It involves loading a new, trusted OS for acquisition. While avoiding rootkit interference and guaranteeing the atomicity of the image, the technique has severe drawbacks. For example the operating system of the host is halted, making it unsuitable for production environments. Additionally, the acquisition OS needs to have drivers for the device used to extract the memory image to (e.g. the network interface), making it highly platform dependent.

Recent advances in hardware virtualization allow running the acquisition software on a higher privilege level than the operating system. For example the Hypersleuth (Martignoni et al., 2010) and VIS (Yu et al., 2011) tools leverage the Intel VMX instruction set to virtualize the operating system on the fly. Running in VMX root-mode, the acquisition software essentially acts as a thin hypervisor and thus is not prone to subversion by operating system level rootkits. Also, the hypervisor based acquisition tool can guarantee the atomicity of the image, by adopting a copy-on-write based imaging approach. However, this approach depends on the ability to load a new hypervisor on the fly. In environments where a hypervisor is already running, this will not work unless nested hardware virtualization is supported and active. With Hyper-V being shipped with Windows 8 and many web servers being virtual instances, this is increasingly often the case. Also, the ability to virtualize the operating system on the fly means a rootkit can do the same thing, defeating the acquisition hypervisor (Rutkowska, 2006; Zovi, 2006; King and Chen, 2006).

A possible solution to this problem is to go even deeper, and execute memory acquisition software on a firmware level. By running in System Management Mode (SMM), the program is isolated from any operating system and even hypervisor based malware (Wang et al., 2011), while still being able to create an atomic memory image without completely halting the host. Unfortunately, only the BIOS can load code into SMM. The acquisition software thus has to be installed by flashing a new BIOS onto the target machine. This requires a reboot, making this technique unsuitable for ad-hoc analysis.

Contributions: In this paper, we advance the field of forensic memory acquisition by considering the efficacy of forensic tools when facing determined and skilled adversaries, willing to use anti-forensic techniques. We find that the current generation of forensic memory acquisition tools are ill equipped to face this adversarial challenge. By understanding the weaknesses present in current tools we are able to further the state of the art by developing more robust solutions, thereby increasing the complexity required by the attacker to effectively bypass forensic tools.

Forensic tool testing is a contemporary research topic. There have been attempts to quantify testable criteria by which to assess the correctness of memory acquisition