LESSON: Advanced DFIR Concepts

Primer

This module includes two labs: the first lab focuses on Examining Famous Malware Memory Dumps, while the second lab focuses on Process Investigation . We highly recommend that instructors plan ahead to allocate sufficient time for these labs to ensure students have ample opportunity to practice.

For this lesson and upcoming lessons, instructors are required to ensure the following activities are completed for each lesson

- Review the “Lesson Opener” and “Real World Scenario” with the learners prior to starting the module.
- Throughout the module, you will find “Consider the Real World Scenario” slides. Review the questions found on these slides, tie the concepts back to the scenario discussed at the start of the lesson as well as content you are presenting, and encourage the learners to share their thoughts.
- For each lesson, you will find a “Pulse Check” slide which is the opportunity for instructors to open a poll to gather feedback from the learners. Leave the poll open for about 1 minute and after you close the poll, share the results with the learners. Encourage the learners to share their thoughts. This information will help the instructors as well as the learners better understand where they are with regards to the lesson.
- Labs are to be demonstrated live for each module. The demonstration of labs is the top priority for the lead instructor. While demonstrating each lab, encourage students to participate and explore.
- At the end of each lesson, it is important to take a few minutes to review the key concepts for the lesson, provide guidance on what the learners can do to prepare for the next lesson, and wrap up with Q&A.

Summary

In this lesson, learners will discuss memory forensics, a critical aspect of cybersecurity. They will uncover the concealed activities of malware and user actions through the lens of memory analysis. Notable cases, such as Stuxnet, Zeus, and Cridex, will serve as case studies, showcasing the application of memory forensics in understanding and combating cyberthreats. The lesson will guide learners through essential frameworks like Volatility and Rekall, providing the tools needed for in-depth memory dump analysis. Emphasizing the complexity of memory forensics due to volatile RAM data and diverse OS structures, the lesson will underscore the importance of identifying the system's OS and hardware as a prerequisite for effective memory forensics. Learners will explore critical elements like KDBG for OS

version identification, format conversion requirements for memory dumps, and the role of VM metadata in virtualized environments. Networking information, socket details, environmental variables, registry keys, and command-line data will be examined, offering a comprehensive understanding of the wealth of insights that memory forensics can provide. The lesson will introduce Volatility's array of plugins tailored for specialized data extraction, with a focus on the user-friendly Volatility Workbench GUI, streamlining memory analysis, albeit with certain limitations in plugin loading. The exploration of process forensics will cap the lesson, enabling learners to identify, understand, and detect malicious activities through techniques like process dumping, DLL analysis, and file dumping. Through practical applications and case studies, this lesson will empower learners to navigate the intricate landscape of memory forensics in cybersecurity.

Objectives

- Recognize the importance of memory analysis in digital forensics.
- List famous memory forensics cases.
- Identify the challenges of memory forensics and the frameworks that help simplify the process.
- Explain how to use the Volatility framework for system profiling and memory format conversion.
- Identify the components of VM metadata in virtualized environments.
- Explain and illustrate how to analyze different components of a memory dump using Volatility.
- Describe the benefits of Volatility Workbench.
- Define the concept and goals of process forensics.
- Describe the use of the Volatility plugins pslist and pstree.
- Differentiate between a legitimate and a suspicious process tree.
- Explain how to perform process, DLL, and file dumping.

Lesson Activities and Teaching Strategies

Estimated Time	Lesson Portion	Directions
2 min	**Lesson Opener:** Advanced DFIR Concepts	● Introduce learners to the importance of DFIR advanced concepts in cybersecurity.
5 min	**Real World Scenario:** Advanced DFIR Concepts	● Review the real world scenario challenge and inform learners that you will be constantly coming back to this scenario throughout the lesson to discover how to solve and apply concepts to this real situation.
2 min	**Lesson Companion:** Advanced DFIR Concepts	● Review the lesson companion, and inform learners that you will be constantly coming back to this scenario throughout the lesson to discover how to solve and apply concepts to this real situation.

20 min	**Cyber Uncovered:** Memory Forensics	● Emphasize the significance of memory analysis in digital forensics for detecting malware activity, user actions, and elusive system states not found in disk-based artifacts. ● Discuss notable malware cases, including Stuxnet, Zeus, and Cridex, to provide real world context and showcase the impact of memory forensics in investigating cyberthreats. ● Introduce analysis frameworks like Volatility and Rekall, highlighting their role in conducting in-depth analysis of memory dumps through various plugins and scripts. ● Explain the challenges inherent in memory forensics, such as the volatile nature of data in RAM and the complexities arising from different operating systems, underscoring the need for specialized expertise. ● Break down the complexities of memory analysis by outlining the extraction of artifacts using frameworks like Volatility and Rekall, covering processes, DLL and handles, network connections, code injection, rootkits, and executables loaded in memory. ● Stress the necessity of profile identification, guiding learners to determine the specific operating system and hardware configuration before delving into memory forensics for accurate data interpretation. ● Introduce the kernel debugger block (KDBG) as a crucial element in system profiling, demonstrating how Volatility leverages this memory block to identify the version and configuration of the operating system in a memory dump. ● Recommend the use of built-in plugins during memory investigation for uncovering hidden processes, detecting malware, and revealing the system's behavior. Advanced users can explore Volshell for more granular control. ● Discuss the various memory formats, such as crash dumps or hibernation files, emphasizing the importance of conversion before analysis. Walk through the steps, including format identification, plugin listing, profile selection, dump conversion, and analysis using Volatility plugins. ● Be prepared to discuss the implication of the real world scenario presented at the beginning of class on Memory Forensics. There are specific prompts that you should ask learners to reflect on to apply this concept to the real world scenario.
5 min	**Real world scenario:** Memory Forensics	● Review the real world scenario challenge and inform learners that you will be constantly coming back to this scenario throughout the lesson to discover how to solve and apply concepts to this real situation.
5 min Break		
20 min	**Cyber Uncovered:** Memory Examination	● Begin the lesson by introducing the concept of VM metadata in virtualized environments, explaining its significance in forensic investigations.

		● Demonstrate the extraction of VM information for VMware using the provided command and discuss the potential variations in extracting information for other virtualization platforms like VirtualBox. ● Cover the topic of network connections in memory forensics, emphasizing the importance of networking information in identifying active connections, open ports, and network activities. ● Explore the examination of socket information in memory forensics, highlighting its role in revealing details about network sockets, including the state of connections and the associated applications. ● Discuss the analysis of environmental variables in a memory dump, focusing on how it contributes to understanding the runtime environment of processes and its relevance to system behavior alterations by malware. ● Introduce the examination of registry keys in memory forensics, explaining its significance in detecting configuration changes, auto-start entries, and other modifications indicative of system tampering or persistence mechanisms. ● Cover the topic of command-line data analysis in memory forensics, explaining its focus on extracting and examining command-line arguments, providing context about process execution, user actions, and potential malicious commands. ● Explore additional Volatility plugins such as evtlogs, getsids, iehistory, and modscan, explaining their specialized roles in extracting and analyzing specific types of data or artifacts from memory dumps. ● Conclude the lesson by introducing Volatility Workbench, emphasizing its role as a graphical user interface (GUI) that simplifies the memory analysis process. ● Discuss its features, such as loading memory dumps, selecting profiles, and running plugins, and highlight its potential downsides. ● Be prepared to discuss the implication of the real world scenario presented at the beginning of class on Memory Examination. There are specific prompts that you should ask learners to reflect on to apply this concept to the real world scenario.
5 min	**Real world scenario:** Memory Examination	● Review the real world scenario challenge and inform learners that you will be constantly coming back to this scenario throughout the lesson to discover how to solve and apply concepts to this real situation.
20 min	**Lab:** Examining Famous Malware Memory Dumps	● Remind learners to use this lab to practice and apply the concepts they have learned throughout the day. ● Learners will receive direct feedback on their lab to properly assess their knowledge and determine where they might need additional assistance.
5 min	**Pulse Check**	● Before you launch the pulse check, explain each section clearly, and encourage the learners to participate in the survey.

		● After administering the survey, share the poll results with learners and ask learners to provide feedback ● Encourage learners to attend office hours with the associate instructor.
5 min Break		
20 min	**Cyber Uncovered:** Process Forensics	● Emphasize the significance of memory dumps in capturing a snapshot of all running processes. ● Highlight the advantages of using memory dumps for process examination compared to live investigations. ● Clearly state the primary goal of process forensics: To analyze and understand processes during memory capture. ● Discuss the importance of identifying malicious or anomalous processes and understanding their impact on the system. ● Introduce the pslist plugin in Volatility for listing active processes in a memory dump. ● Explain the role of pstree in displaying processes in a tree structure, aiding in understanding parent-child relationships. ● Define a legitimate process tree as a representation of normal, safe processes in a healthy system. ● Provide an example, such as svchost.exe executed under services.exe, to illustrate a legitimate process hierarchy. ● Define a suspicious process tree in memory forensics, emphasizing unusual parent-child relationships or processes associated with malicious activities. ● Use the example of Stuxnet to illustrate processes with illegitimate PPIDs in a suspicious process tree. ● Explain the concept of process dumping in memory forensics, involving the extraction of a process's executable for detailed analysis. ● Provide the syntax for using the ProcDump plugin in Volatility and its role in extracting process executables. ● Discuss situations where adversaries inject specific DLLs instead of entire executables. ● Introduce Volatility's DllDump plugin for extracting dynamic-link libraries from the memory space of a process. ● Explain that memory can contain documents, executables, or temporary files, offering insights into activities during capture. ● Introduce the filescan command in Volatility and mention the possibility of using the dumpfiles command for extraction. ● Be prepared to discuss the implication of the real world scenario presented at the beginning of class on Forensics Processs. There are specific prompts that you should ask learners to reflect on to apply this concept to the real world scenario.
5 min	**Real world scenario:** Forensics Process	● Review the real world scenario challenge and inform learners that you will be constantly coming back to this scenario throughout the lesson to discover how to solve and apply concepts to this real situation.

20 min	**Lab:** Process Investigation	● Remind learners to use this lab to practice and apply the concepts they have learned throughout the day. ● Learners will receive direct feedback on their lab to properly assess their knowledge and determine where they might need additional assistance.
5 min	**End-of-Course Survey**	● Allocate 5 minutes to facilitate the completion of the End-of-Course Survey. ● Encourage learners to provide honest and constructive feedback about their learning experience.
15 min	**Lesson Closure**	● Encourage learners to read ahead of time ● Provide learners additional resources to read / practice and assign homework (e.g., future labs) before you demonstrate the labs during the next class ● Spend some time to highlight what are the key takeaways from today's lesson ● Important topics covered during the class includes ○ Highlight the key takeaway regarding Memory Forensics: ■ Memory analysis in forensics is essential for uncovering hidden malware activities and tracking user actions. Tools such as FTK Imager are used to capture a memory dump from a Windows system, while Volatility is employed for comprehensive analysis of the memory dump. ■ The first step in capturing a system's volatile memory is identifying the underlying operating system and hardware configuration. This ensures compatibility with the tools being used and allows forensic experts to tailor their approach based on the specific system architecture. ○ Highlight the key takeaway regarding Memory Examination: ■ Memory dumps contain various information such as such as running processes, network connections, sockets, and active user sessions, and URLs. ■ Analyzing environmental variables in memory dumps provides valuable insights into the runtime environment of processes. By examining these variables, we can understand how processes are configured, including their dependencies, execution paths, and system settings at the time of capture. This information is crucial for reconstructing events of the system and identifying potential security issues.

		○ Highlight the key takeaway regarding Process Forensics: ■ Process forensics focuses on identifying and understanding processes during memory capture, with the primary goal of detecting malicious activities. ■ By analyzing the process tree, you can identify legitimate processes and gain insights into the system's state when the memory dump was captured ■ Suspicious process trees, often marked by unusual relationships, may indicate potential security threats.
	Add Additional Time Filler	● Review using Kahoot or other similar platforms ● Conduct interview preparation conversations ● Continue discussions on real-world scenarios ● Demonstrate how to create users in Linux and grant them permissions ● Discuss different career paths in cybersecurity and highlight the roles that require Linux skills