508.3 - Memory Forensics in Incident Response and Threat

Incident Response 4-21

Enterprise IR challenges...4-5

Rapid-Response Tooling... 6

Enterprise IR Agents 7-20

Remote Access Agents 7

Remote Analysis Agents 8

F-Response 9-12

KAPE.. 13-15

Velociraptor.. 16-20

EDR.. 22-27

Why EDR? 24

EDR Challenges 25

Memory Forensics 26-27

Memory Forensics............................. 30-202

Why Memory Forensics?..................................30-32

Overview and possible findings/artifacts................31

Memory analysis advantages 32

Memory Acquisition..33-39

Windows... 34-37

Overview 34-35

hiberfil.sys 36-37

Virtual machines..38

Memory Analysis Introduction.......................40-118

Definition..40

Windows Analysis.. 42-44

KDBG / KPCR / PEB / VAD..42

Unlinking..43

Hooking..43

Memory structure simplified...................................44

Volatility.. 45-57

Overview 45

How-to 46-47

Help flag (-h) 48

Volatility Version 3.0...49

Volatility Profiles..51

Image Identification..52

Build Numbers (how-to)..52

kdbgscan...53-55

Hibernation file conversion.................................56-57

Starting analysis (finding the first hit)............. 58-118

Step 1: Identify rogue processes..........................59-87

Process Blocks (EPROCESS)...........................60-61

Analyzing Processes 62-64

Volatility Plugins 65-86

pslist (process overview)..........................66-68

psscan..71-74

pstree 75-79

Baseline 84-86

Know normal (Windows)...................................69

EchoTrail...70

WMI & PowerShell processes........................80-81

Overview...80-81

Suspicious WMI processes 82-83

wmiprvse.exe 82

scrcons.exe...83

Review...87

Step 2: Analyzing process objects 89-110

Process overview...90-91

Volatility plugins overview................................92

dlllist...93-95

Cobalt Strike Sacrificial Processes 96-97

getsids 98-101

Understanding SIDs..................................102-103

Handles 104-106

Named pipes..107

Mutants/Mutexes...................................108-109

Review...110

Step 3: Network Artifacts...............................111-118

Overview 112

Unusual behavior..113

Volatility plugins......................................114-117

Overview..114

netscan..115-117

Review..118

Code Injection, Rootkits and Extraction 120-198

Starting analysis (finding the first hit)....................121

Step 4: Look for Code Injection........................122-148

Why code injection?...123

Overview (DLL injection/Process hollowing)...124

Simple DLL Injection............................125-127

Volatility plugins......................................128-137

Overview..128

ldrmodules..129-137

Reflective Code Injection...........................138-139

Overview..138

Detecting reflective/hidden code injection 139

Detecting code injection...........................140-148

malfind 140-147

Review...148

Step 5: Hooking and Rootkit Detection.............149-169

Overview 150-151

SSDT/IDT/IAT/IRP................................150-151

Volatility plugins......................................152-168

Overview..152

ssdt 153-156

psxview (cross-view analysis).................158-159

modscan & modules............................161-163

driverbl..164-165

apihooks 166-168

Direct Kernel Object Manipulation (DKOM)........157

Review...169

Plugin use-cases...169

Step 6: Extracting objects from memory...........172-198

Volatility plugins......................................173-180

dlldump 174-175

moddump 176-178

procdump 179

memdump / vaddump................................180

strings 181

grep 182

Using strings and grep with memdump.......183-184

Command history and active consoles 185-188

cmdscan & consoles............................185-186

Windows 10 memory compression....................189

Extracting files...191-196

dumpfiles...191-193

filescan 192

filescan + dumpfiles....................................193

Extracting Registry..197

shimcachemem..197

Review...198

Scaling Analysis... 201-203

Live memory forensics................................... 201-202

Using IOCs..203

openioc_scan 203

yarascan 203

page_brute.py...203