CodeMachine Kernel Security Training Topics

Duration 10 Days
Dates 2-6 May, 2016 and 9-13 May, 2016
Location Canberra, ACT, Australia

Day Wise Course Schedule.

Day	Time	Topic	Content
1	AM	Kernel Debugger	Debugger Package, Debugger Parameters, Kernel Mode Debugging, Debugger Symbols, Debugger Command Types, Debugger Command Reference
1	AM	Kernel Architecture	NTOSKRNL, HAL & Drivers, Processes and Threads, System & System Idle Process, Process and Thread Data Structures, System Calls, System Service Dispatching, KUSER_SHARED_DATA, User vs Kernel Mode Execution, x64 Registers and Calling Convention
2	AM	Execution Contexts	Kernel Processor Control Region, Trap Frames, Interrupt Request Levels, Deferred Procedure Calls, Timers, Asynchronous Procedure Calls, System Worker Threads
2	PM	Synchronization	Dispatcher Objects, Thread Waits, Interlocked Operations, Mutexes, Fast Mutexes & Guarded Mutexes, Executive Resources, Spin Locks
3	AM	Memory Management	Virtual Address Space, Virtual Address Descriptors, Kernel Virtual Address Space, Address Translation, PTE and Session Space, PFN Database, System Cache, Kernel Mode Stacks, Kernel Pools, Memory Descriptor Lists, Memory Mapping
3	PM	Objects & Handles	Object Manager, Object Namespace, Objects and Handles, Object Layout, Object Header, Object Types, Object Type Callbacks
4	AM	Device Drivers	Driver architecture, I/O manager data structures (driver object, device object, file object, symbolic link), I/O requests (IRP and I/O stack location), I/O processing, IOCTL requests, data buffering mechanisms.
4	PM	Kernel Security Mitigations	NULL Page Allocation Prevention, Supervisor Mode Execution Protection, Safe Linking and Unlinking, Kernel Mode Code Signing, Kernel Patch Protection, Kernel Mode Data Execution Prevention, Non-Executable Pools, Kernel Address Space Layout Randomization
5	AM	Driver Development Environment	Windows Driver Kit, Building with Enterprise WDK, Targets, Platforms and Configurations, Kernel Debugging, Driver Symbols and Source Code, Driver Replacement Maps

5	PM	Driver Programming Basics	Driver Entry Points, Windows Version APIs, WDK Headers, NTSTATUS Codes, Debug Prints, Memory Allocation, Unicode Strings
6	AM	I/O Processing	I/O Manager Objects, User/Kernel Interface, Building IRPs, Object References
6	PM	Asynchronous Execution	DPC Routines, Kernel Timers, Worker Routines, Driver Threads
7	AM	Queues & Serialization	Linked Lists, Waitable Locks, Rundown protection, Spin Locks, Interlocked Operations
7	PM	Advanced Techniques	Mapping memory, Executive callbacks, Stack back-traces, Registry access, File access
8	AM	Kernel Security Bypass	Kernel Mode Address Leaks, SMEP Bypass, Kernel Mode Code Signing Bypass, Kernel Mode Shell Code, NTOSKRNL Exports
8	PM	Hooking Techniques	Interception Types, Patch Guard, Inline Hooking, Memory Protection, Import Hooking, Dispatch Table Hooking
9	AM	Filtering Mechanisms	IRP Filter, Process & Thread Filter, Object Access Filter, Image Load Notification, Registry Callback, File System Mini-Filter, Early Load Anti-Malware (ELAM)
9	PM	Covert Communications	NDIS driver types, NDIS internal data structures, net buffer lists (NBL), net buffers (NB), intermediate drivers (NDIS IM), lightweight filters (NDIS LWF), NDIS hooking, host firewall bypass.
10	AM	Stealth Behavior	Kernel structure manipulation, rootkit self-defense, anti-debugging techniques, anti-VM techniques, stealth user mode communication, stealth filtering, detection bypass.
10	PM	Detection Tools & Case Studies	Volatility framework, rootkit detectors, endpoint security products, Rustock, TDSS/TDL4, ZeroAccess Carberp, Regin.