

# Windows Internals 101

[ AFK.conf ]

PS > ls env:

- Offensive Security    Accenture
- Twitter / GitHub      @g0ttfrid
- LinkedIn                in/yurimaia

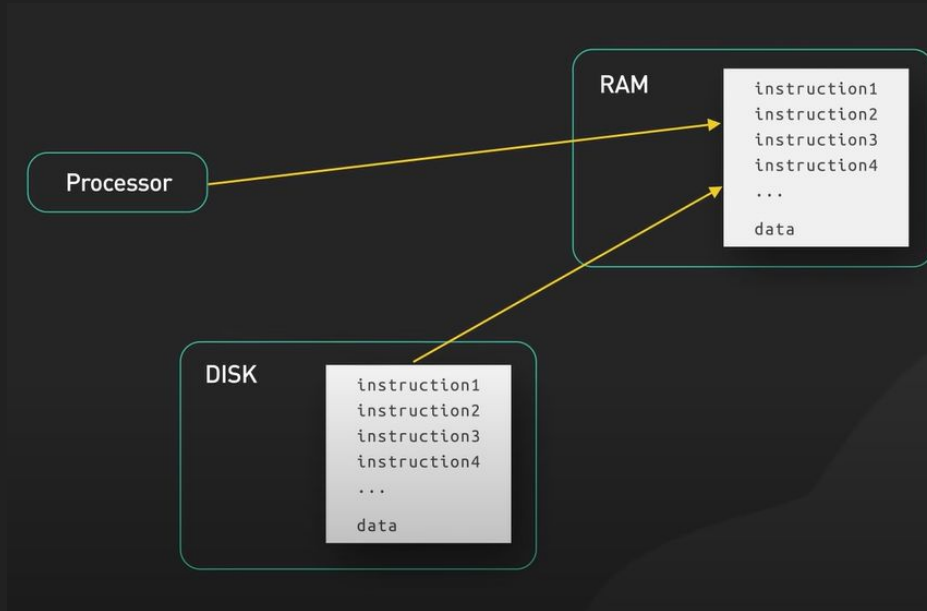


# PS > Agenda

- Process, Threads and Handles
- Windows API (Win32 API)
- User Mode vs Kernel Mode
- Syscalls

# PS > Process, Threads and Handles

- Program > Process
- Resources



Processor registers

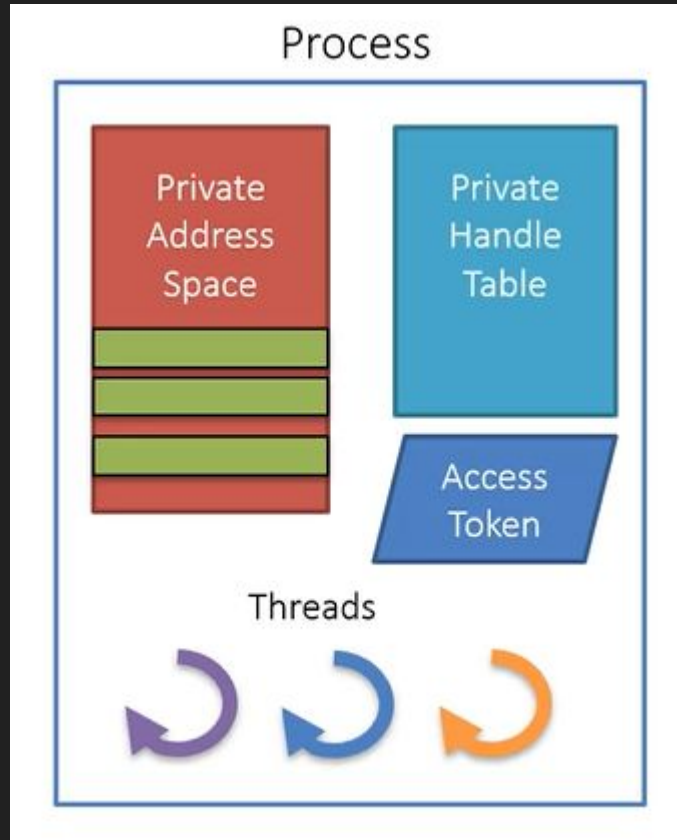
Program counters

Stack pointers

Memory pages

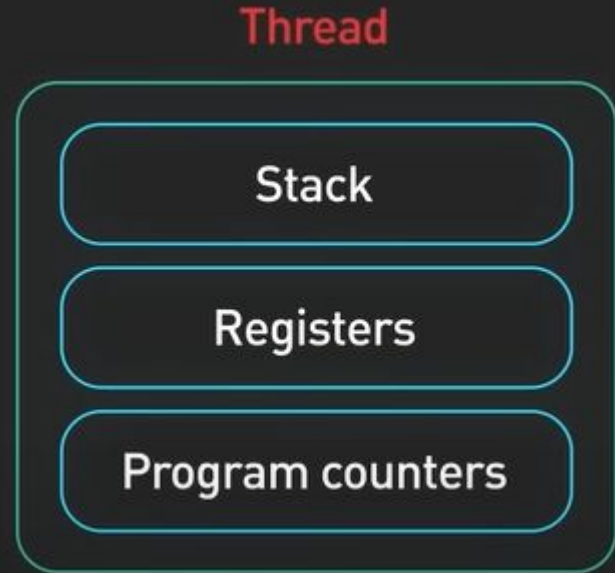
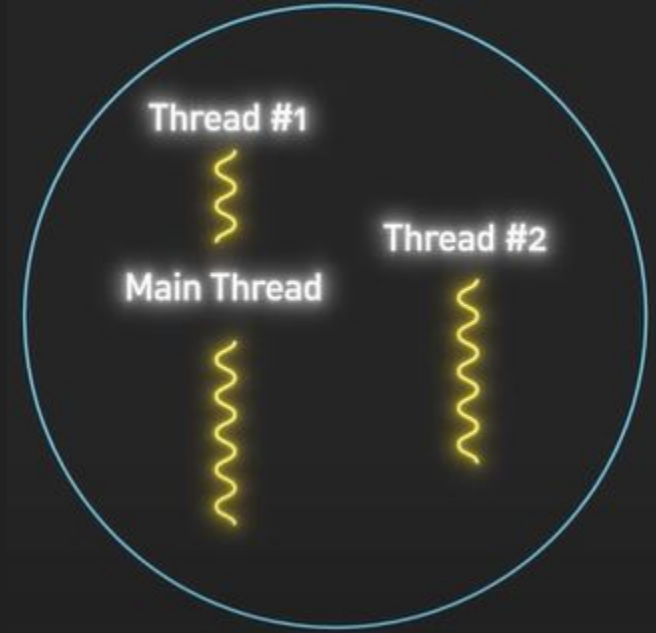
# PS > Process, Threads and Handles

- Address Space



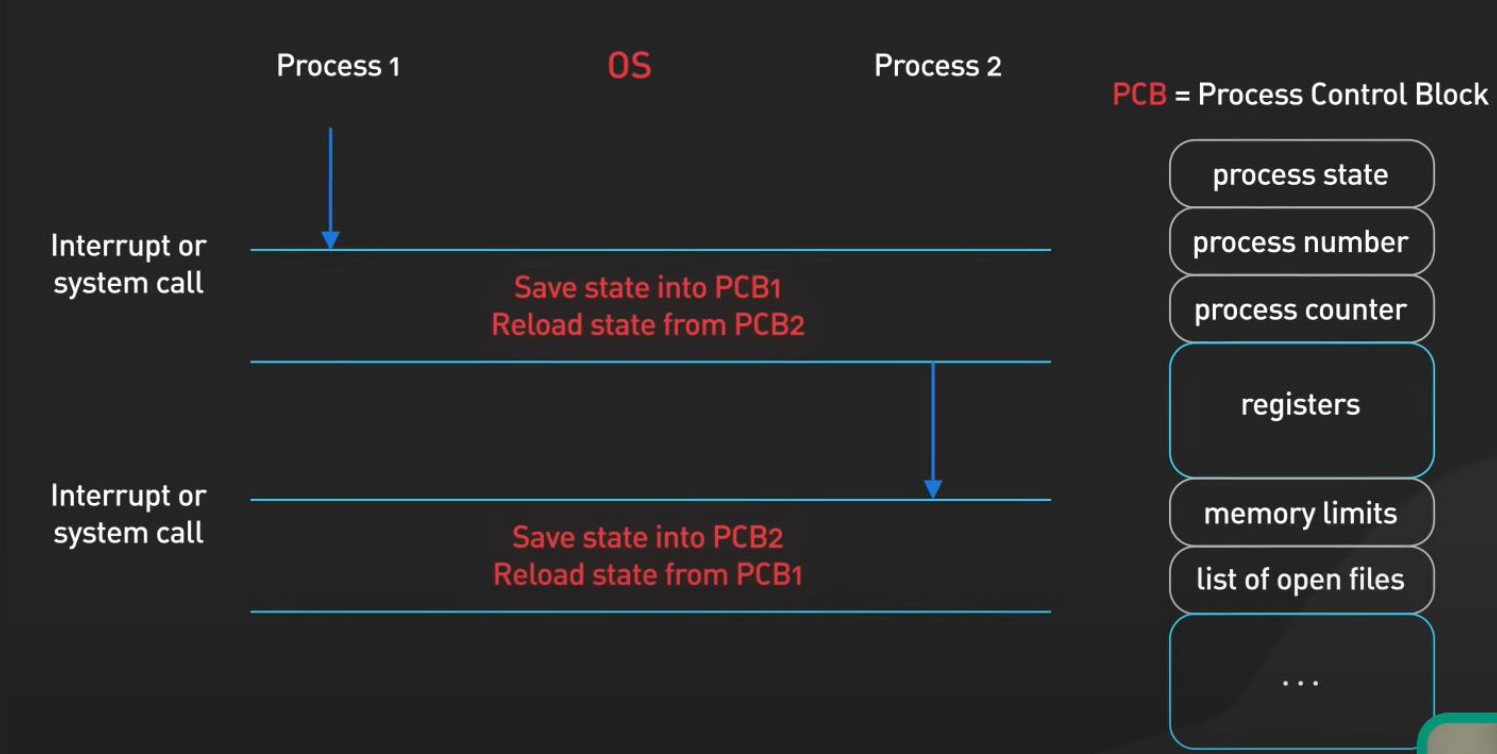
# PS > Process, Threads and Handles

- Threads



# PS > Process, Threads and Handles

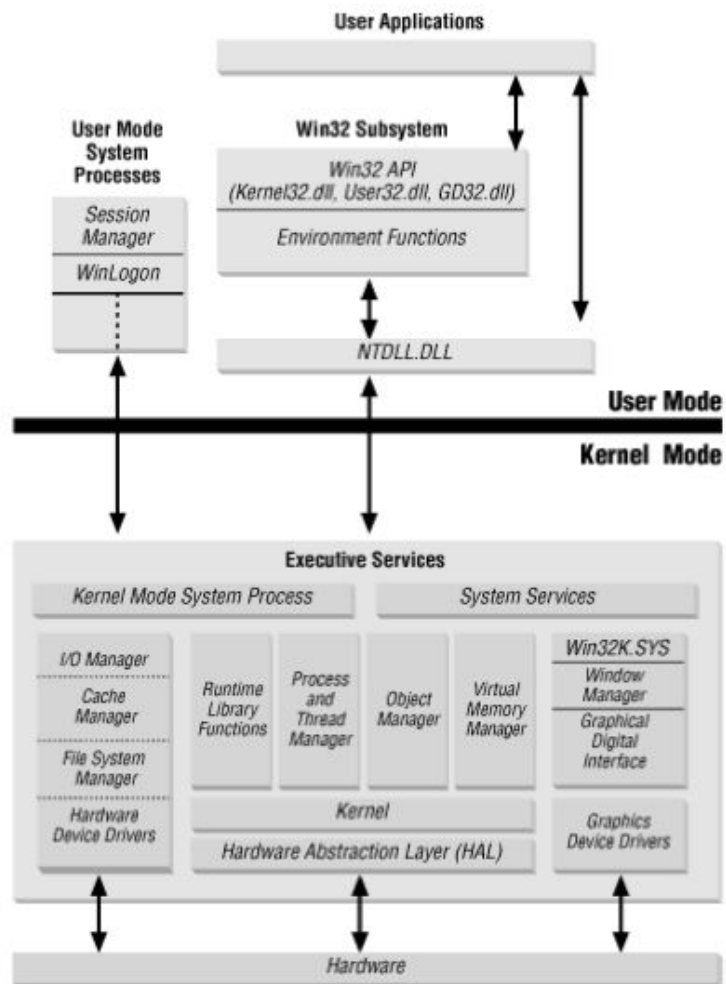
- Context Switch



## PS > Windows API (Win32 API)

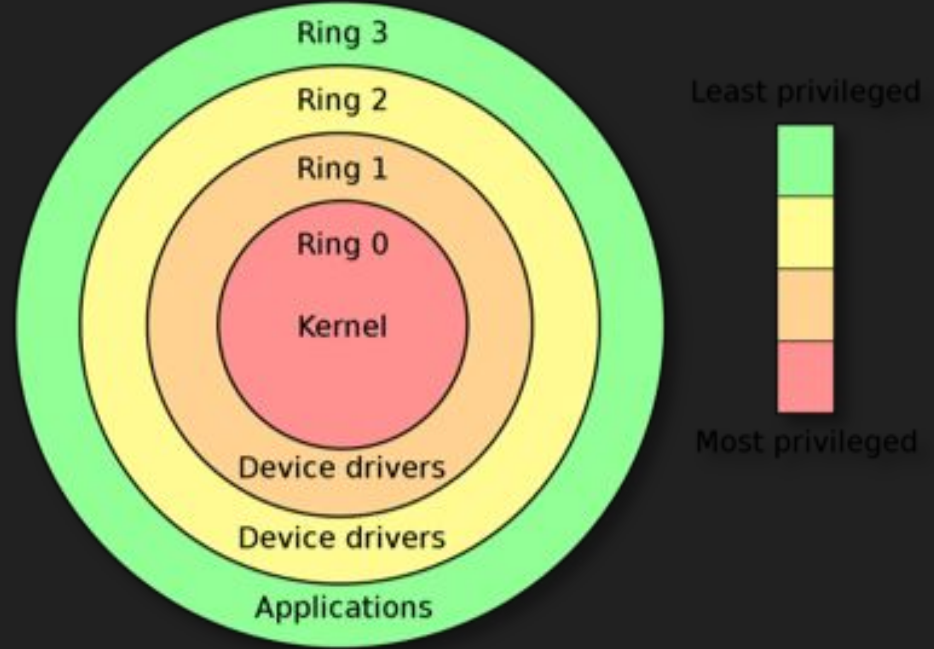
- Host enumeration
- Starting processes
- Process injection
- Token manipulation
- And more...





# PS > User Mode vs Kernel Mode

- User apps
- S.O. Services



# Event Properties

Event Process Stack

Frame	Module	Location	Address	Path
K 0	FLTMGR.SYS	FltDecodeParameters + 0x1c5d	0xffff8070833555d	C:\WINDOWS\System32\drivers\FLTMGR.SYS
K 1	FLTMGR.SYS	FltDecodeParameters + 0x17bc	0xffff807083350bc	C:\WINDOWS\System32\drivers\FLTMGR.SYS
K 2	FLTMGR.SYS	FltDecodeParameters + 0x1328	0xffff80708334c28	C:\WINDOWS\System32\drivers\FLTMGR.SYS
K 3	FLTMGR.SYS	FltDecodeParameters + 0x111e	0xffff80708334a1e	C:\WINDOWS\System32\drivers\FLTMGR.SYS
K 4	ntoskml.exe	IoFileCallDriver + 0x59	0xffff8005a332ae9	C:\WINDOWS\system32\ntoskml.exe
K 5	ntoskml.exe	NtQueryInformationFile + 0x1071	0xffff8005a8b0fa1	C:\WINDOWS\system32\ntoskml.exe
K 6	ntoskml.exe	NtWriteFile + 0x8bd	0xffff8005a8af18d	C:\WINDOWS\system32\ntoskml.exe
K 7	ntoskml.exe	setjmpex + 0x7805	0xffff8005a47085	C:\WINDOWS\system32\ntoskml.exe
U 8	ntdll.dll	NtWriteFile + 0x14	0x7ffc5f55f864	C:\WINDOWS\SYSTEM32\ntdll.dll
U 9	KERNELBASE.dll	WriteFile + 0x7a	0x7ffc5c04ebda	C:\WINDOWS\System32\KERNELBASE.dll
U 10	notepad.exe	notepad.exe + 0x5c0e	0x7f7306f5c0e	C:\WINDOWS\system32\notepad.exe
U 11	notepad.exe	notepad.exe + 0x5fd1	0x7f7306f5fd1	C:\WINDOWS\system32\notepad.exe
U 12	notepad.exe	notepad.exe + 0x28e5	0x7f7306f28e5	C:\WINDOWS\system32\notepad.exe
U 13	notepad.exe	notepad.exe + 0x4037	0x7f7306f4037	C:\WINDOWS\system32\notepad.exe
U 14	USER32.dll	DispatchMessageW + 0x6a6	0x7ffc5e42ca66	C:\WINDOWS\System32\USER32.dll
U 15	USER32.dll	DispatchMessageW + 0x1c2	0x7ffc5e42c582	C:\WINDOWS\System32\USER32.dll
U 16	notepad.exe	notepad.exe + 0x448d	0x7f7306f448d	C:\WINDOWS\system32\notepad.exe
U 17	notepad.exe	notepad.exe + 0x1ae07	0x7f73070ae07	C:\WINDOWS\system32\notepad.exe
U 18	KERNEL32.DLL	BaseThreadInitThunk + 0x14	0x7ffc5c997974	C:\WINDOWS\System32\KERNEL32.DLL
U 19	ntdll.dll	RtlUserThreadStart + 0x21	0x7ffc5f52a271	C:\WINDOWS\SYSTEM32\ntdll.dll

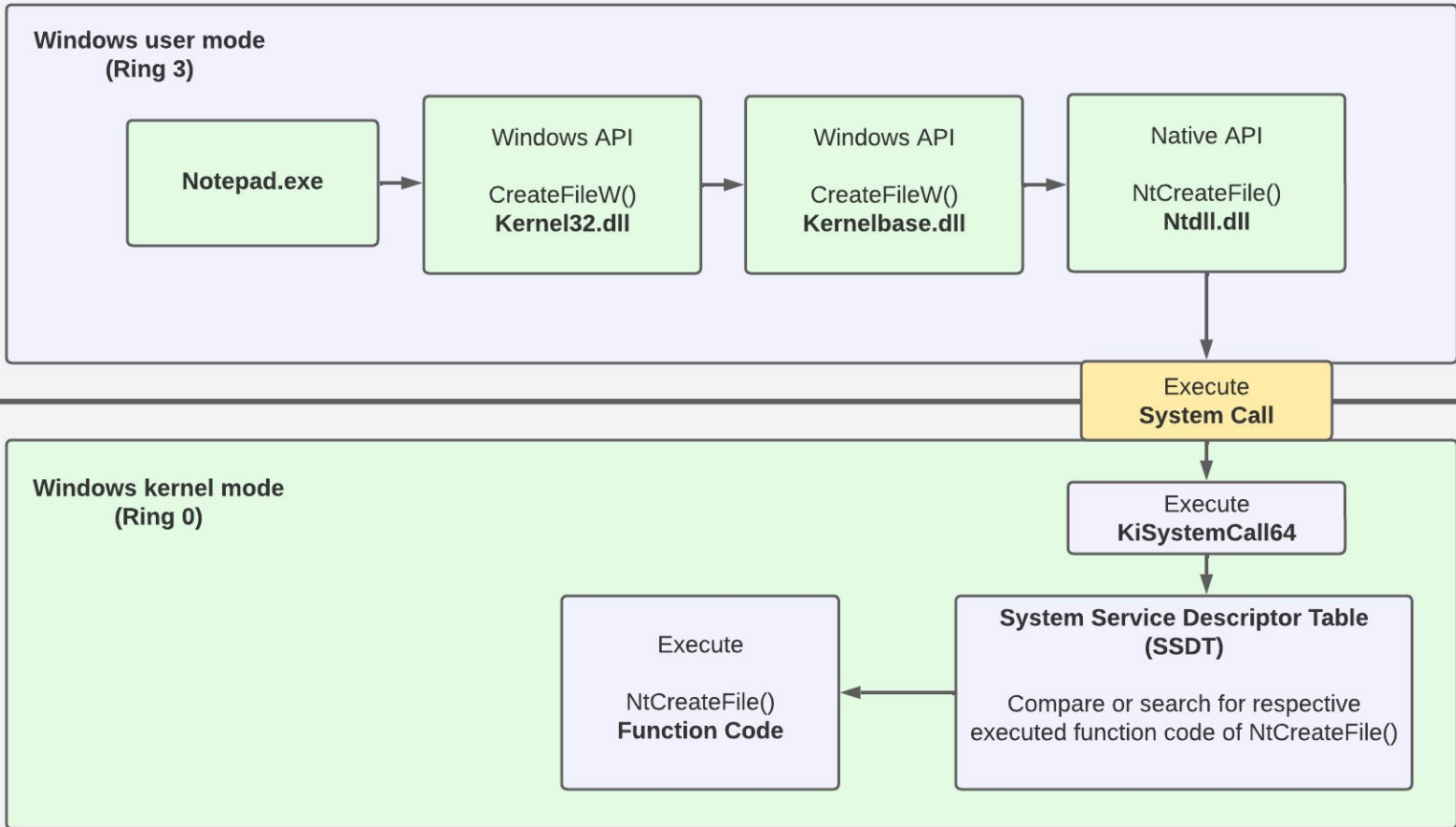
Kernel-mode

User-mode

Sys Call

## PS > Syscalls

- Access to hardware such as scanners and printers
- Network connections for sending and receiving data packets
- Reading and writing files
- and more...



The figure shows the transition from Windows user mode to kernel mode in the context of saving a file within notepad.exe.

# Hands-on



# PS > refs

<https://training.zeropointsecurity.co.uk/courses/red-team-ops-ii>

<https://learn.microsoft.com/pt-br/windows/win32/procthread/about-processes-and-threads>

<https://synzack.github.io/Blinding-EDR-On-Windows/>

<https://www.outflank.nl/blog/2019/06/19/red-team-tactics-combining-direct-system-calls-and-srdr-to-bypass-av-edr/>

<https://redops.at/en/blog/direct-syscalls-a-journey-from-high-to-low>