VolShell cheatsheet

By @LabibNag & @0xMohamedhasan

Basic		
Info	Command	Example
Help	vol.py -f 'mem.file' --profile 'profile' volshell -h	vol.py -f "test.mem" --profile WinXPSP2x86 volshell -h
Write mode	vol.py -f 'mem.file' --profile 'profile' volshell -w	vol.py -f "test.mem" --profile WinXPSP2x86 volshell -w
Specific Process ID	vol.py -f 'mem.file' --profile 'profile' volshell -p 'process_id'	vol.py -f "test.mem" --profile WinXPSP2x86 volshell -p 1337
Specific Process name	vol.py -f 'mem.file' --profile 'profile' volshell -n 'process_name'	vol.py -f "test.mem" --profile WinXPSP2x86 volshell -n notepad.exe
Specific object offset (V)	vol.py -f 'mem.file' --profile 'profile' volshell -o 'virtual address'	vol.py -f "test.mem" --profile WinXPSP2x86 volshell -o 0x7454122

Main APIs		
Help	hh(api)	hh(addrspace())
Active processes list	ps()	
Check current process context	sc()	
Modules list	modules()	
Search	find("str", max=n, shift=n, skip=n, count=false, length=n)	find("Nn" , max=5, shift=0, skip=0, count=False, length=5)
Show struct info	dt(objct_name, address=offset, space=(v)(p), =(list nested structs))	dt("_EPROCESS", address=0x822943c0, space=addrspace(), recursive=False))
Show struct layout	dt(objct_name)	dt("_LIST_ENTRY")

addrspace API		
Get current V address space	addrspace()	
Get current P address space	addrspace().base	
Read V address space	addrspace().read(offset , length)	addrspace().read(0x821E0A20 , 120)
Read P address space	addrspace().base.read(offset , length)	addrspace().base.read(0x23e0a20 , 120)
Write at V address space	addrspace().write(data , offset)	addrspace().write("test" , 0x821E0A20)
Write at P address space	addrspace().base.write(data , offset)	addrspace().base.write("test" , 0x23e0a20)
Convert V to P	addrspace().vtop(V.address)	addrspace().vtop(0x821E0A20)

Process APIs	
Get Current process V offset	proc()
Get Current process addrspace	proc().get_process_address_space()
Get Current process modules	proc().get_load_modules()

Data representation		
Show data in hex	db(address, length=n, space=V or P)	db(x821E0A20, length=128, space=addrspace().base)
Show data in dword	dd(address, length=n, space=V or P)	db(0x23e0a20, length=128, space=addrspace())
Show data in qword	dq(address, length=n, space=V or P)	db(x821E0A20, length=128, space=addrspace())
Disassemble	dis(address, length=n, space=V or P, mode=16bit,32bit,64bit)	dis(0x822943c0, length=128, space=addrspace(), mode=32bit)