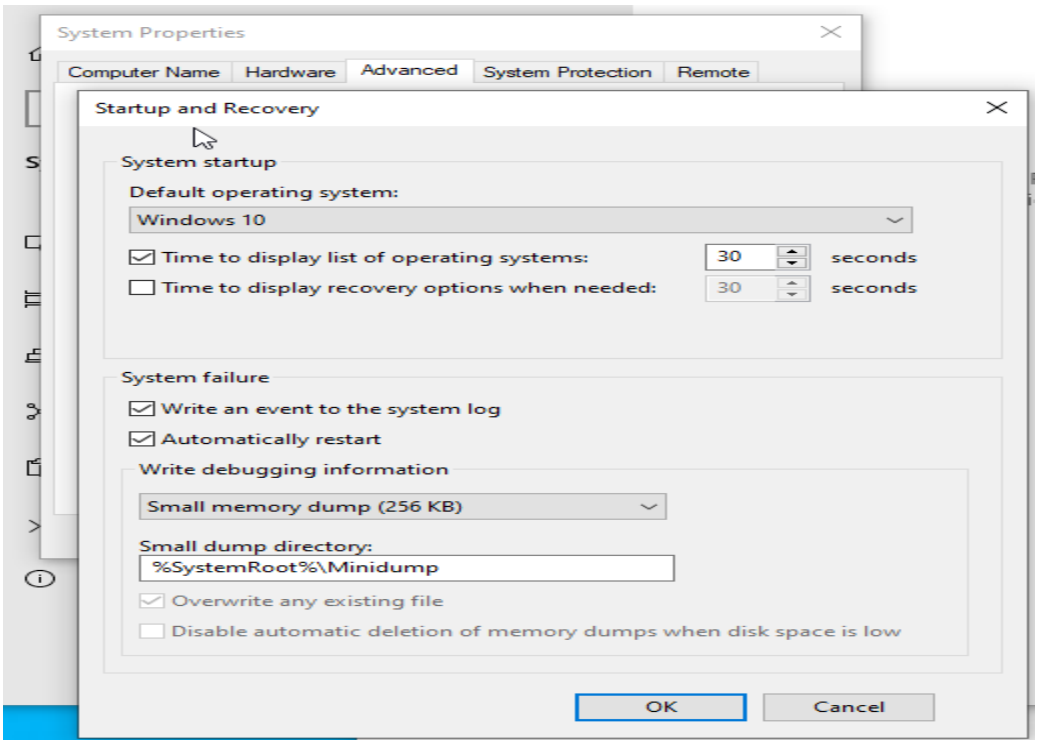
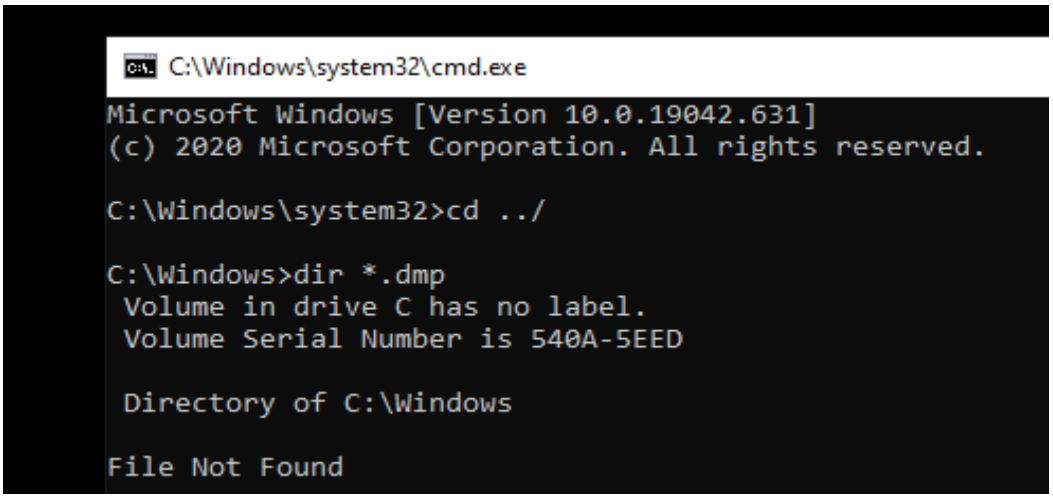
PART 1: WINDOWS CRASH DUMP

Task 1:

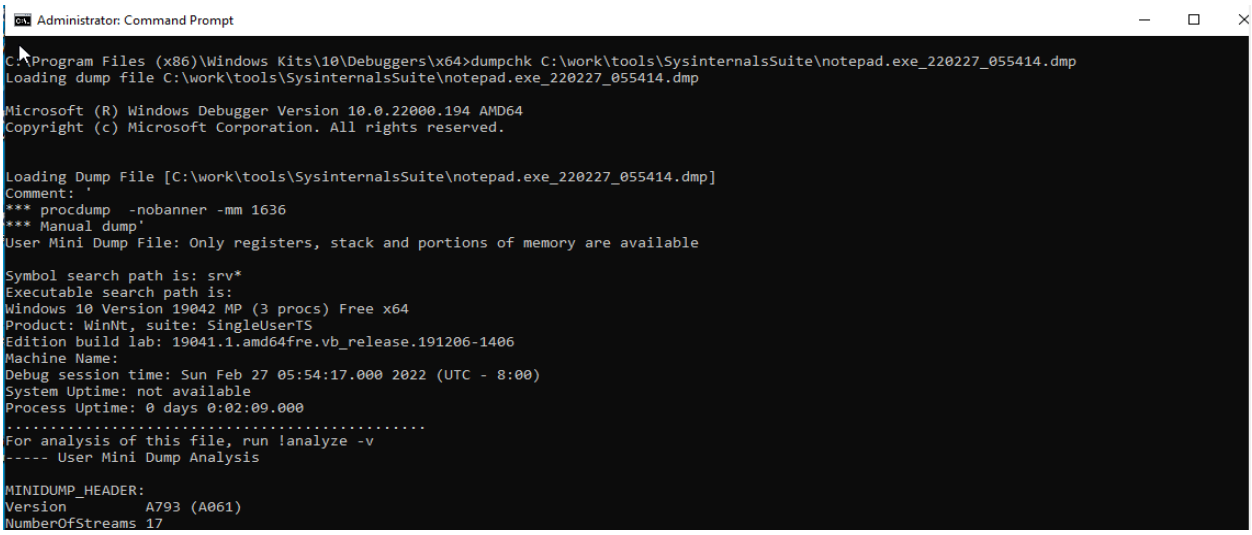
This step teaches us to create memory dump whenever window crashes so we can later investigate about the memory location, program status and other stuff.





Task 2:

Dumpchk *is a tool that is used to analyze the crash dump files.*



```
Administrator: Command Prompt

C:\Program Files (x86)\Windows Kits\10\Debuggers\x64>dumpchk C:\work\tools\SysinternalsSuite\notepad.exe_220227_055414.dmp
Loading dump file C:\work\tools\SysinternalsSuite\notepad.exe_220227_055414.dmp

Microsoft (R) Windows Debugger Version 10.0.22000.194 AMD64
Copyright (c) Microsoft Corporation. All rights reserved.


Loading Dump File [C:\work\tools\SysinternalsSuite\notepad.exe_220227_055414.dmp]
Comment: '
*** procdump  -nobanner -mm 1636
*** Manual dump'
User Mini Dump File: Only registers, stack and portions of memory are available

Symbol search path is: srv*
Executable search path is:
Windows 10 Version 19042 MP (3 procs) Free x64
Product: WinNt, suite: SingleUserTS
Edition build lab: 19041.1.amd64fre.vb_release.191206-1406
Machine Name:
Debug session time: Sun Feb 27 05:54:17.000 2022 (UTC - 8:00)
System Uptime: not available
Process Uptime: 0 days 0:02:09.000
.....................................................
For analysis of this file, run !analyze -v
----- User Mini Dump Analysis

MINIDUMP_HEADER:
Version         A793 (A061)
NumberOfStreams 17
```

PART 2: COLLECTING PROCESS INFORMATION

Task 1:

Pslist *is a tool that is used to display information about CPU usage and the processes running on the computer*

Administrator: Command Prompt

```
C:\work\tools\SysinternalsSuite>pslist -nobanner
Process information for DESKTOP-PG94K8K:

Name                 Pid Pri Thd  Hnd   Priv        CPU Time    Elapsed Time
Idle                   0   0   3    0     60      0:37:55.343     0:14:57.447
System                 4   8 141 2392    196      0:00:20.937     0:14:57.447
Registry             100   8   4    0   5440      0:00:00.687     0:15:01.759
smss                 344  11   2   53   1056      0:00:00.218     0:14:57.420
csrss                440  13  11  453   1648      0:00:00.609     0:14:39.818
wininit              516  13   1  164   1348      0:00:00.187     0:14:39.578
csrss                536  13  13  397   1756      0:00:04.140     0:14:39.566
winlogon             616  13   3  273   2596      0:00:00.218     0:14:39.504
services             636   9   5  603   4396      0:00:01.984     0:14:39.473
lsass                680   9  10 1150   6236      0:00:01.671     0:14:39.422
svchost              792   8   1   55    796      0:00:00.000     0:14:39.151
fontdrvhost          812   8   5   39   1272      0:00:00.031     0:14:39.138
fontdrvhost          820   8   5   39   1656      0:00:00.265     0:14:39.138
svchost              880   8  15 1041   9352      0:00:02.171     0:14:39.097
svchost              924   8  15 1027   5856      0:00:02.453     0:14:38.876
svchost              980   8   5  249   2020      0:00:00.359     0:14:38.831
dwm                  384  13  18  911  36848      0:00:10.484     0:14:38.695
svchost             1032   8   3  107   1232      0:00:00.015     0:14:38.496
svchost             1040   8   2  150   1384      0:00:00.125     0:14:38.496
svchost             1092   8   1  204   2000      0:00:00.156     0:14:38.475
svchost             1112   8   1  142   1576      0:00:00.015     0:14:38.472
svchost             1164   8   5  391   5348      0:00:00.781     0:14:38.450
svchost             1240   8   6  245   3264      0:00:00.140     0:14:38.380
```

```
TiWorker             3936   8   3  172  22256      0:00:00.750     0:01:24.852
UserOOBEBroker       3972   8   4  132   1852      0:00:00.062     0:01:03.492
svchost              5888   8   4  145   1740      0:00:00.046     0:00:51.989
smartscreen          1688   8   4  140   2348      0:00:00.046     0:00:51.893
SearchProtocolHost   5044   4  11  361   2732      0:00:00.109     0:00:08.959
SearchFilterHost      432   4   8  172   2224      0:00:00.078     0:00:08.903
notepad              1636   8   7  275   3480      0:00:00.296     0:00:07.657
SearchProtocolHost   5024   4   9  264   1944      0:00:00.046     0:00:06.992
dllhost              2052   8   7  127   1572      0:00:00.062     0:00:02.608
pslist               3828  13   4  222   2684      0:00:00.343     0:00:00.302
```

Here using **pslist** we found the **PID** of the notepad application which is **1636.**

Task 2:

```
C:\work\tools\SysinternalsSuite>procdump -nobanner -mm 1636
[05:54:14] Dump 1 initiated: C:\work\tools\SysinternalsSuite\notepad.exe_220227_055414.dmp
[05:54:17] Dump 1 complete: 1 MB written in 3.1 seconds
[05:54:17] Dump count reached.
```

prodump *is a tool used to dump the memory of a particular process. We use this command "**procdump -nobanner -mm 1636**"*

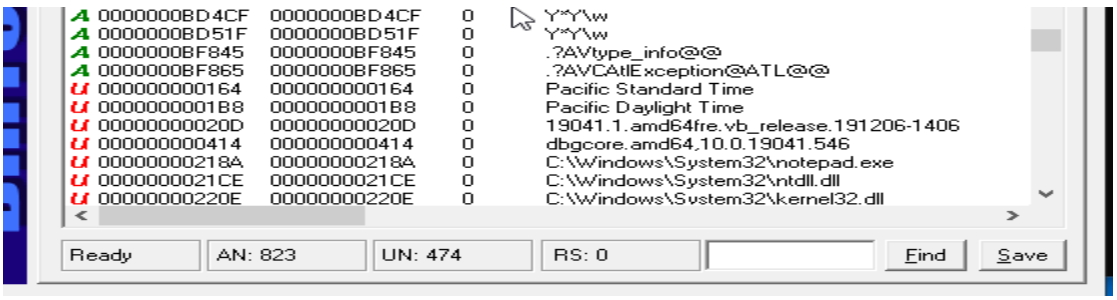
→ ***procdump*** *dumps the memory*

→ ***-nobanner*** *removes the banner*

→ ***-mm*** *mini dump*

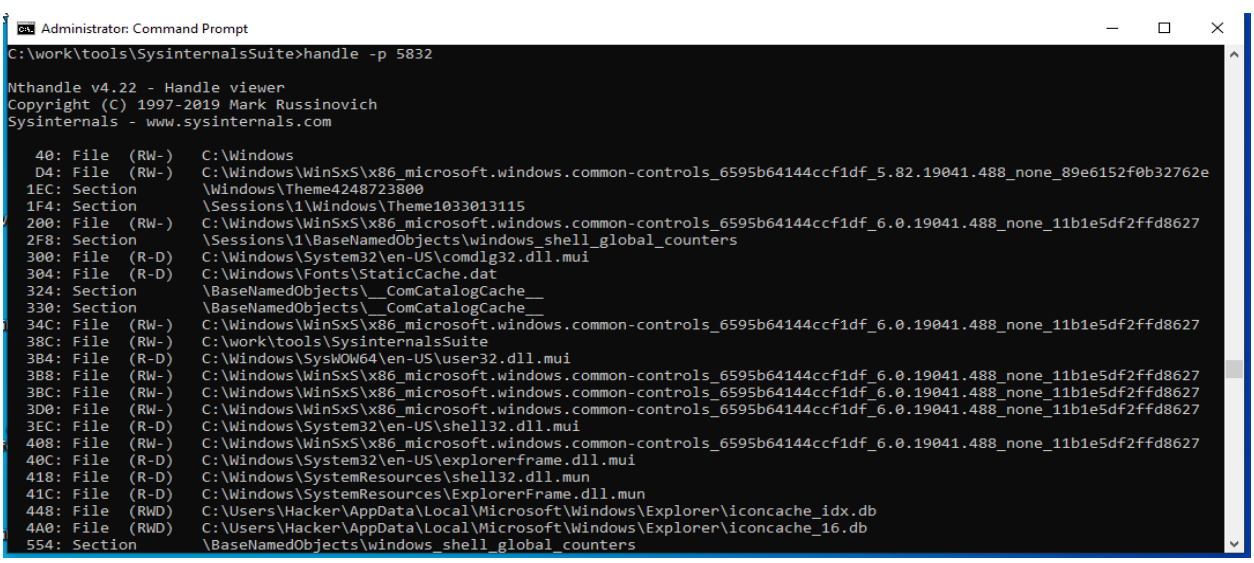
→ ***1636*** *is a **PID (**process id)*

Task 3:



*This is **Bintext** application that reads the memory dump file that we create.*

Task 4:



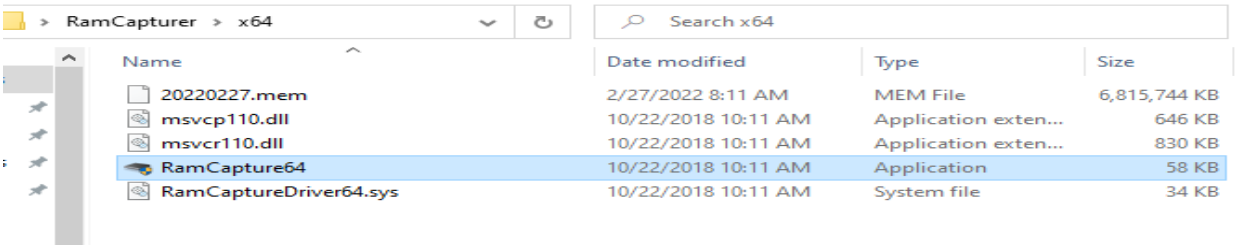
Handlers *are used just like pointers and gives us dump of the open files for all the processes.*

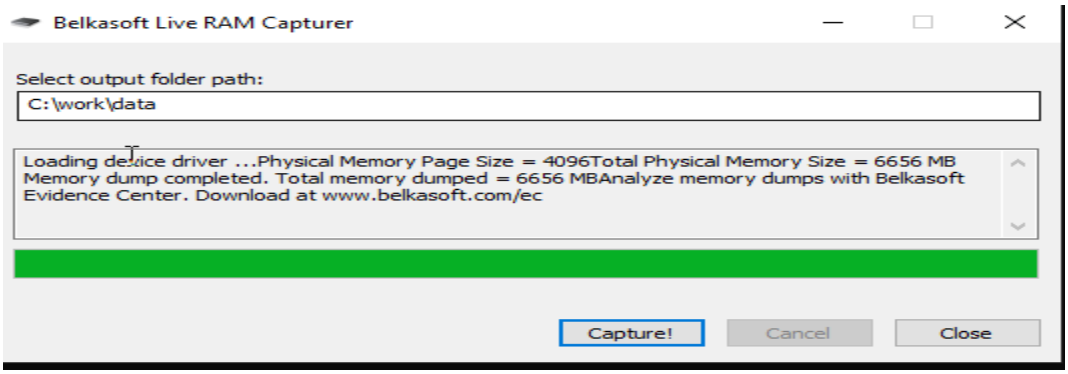
Task 5:

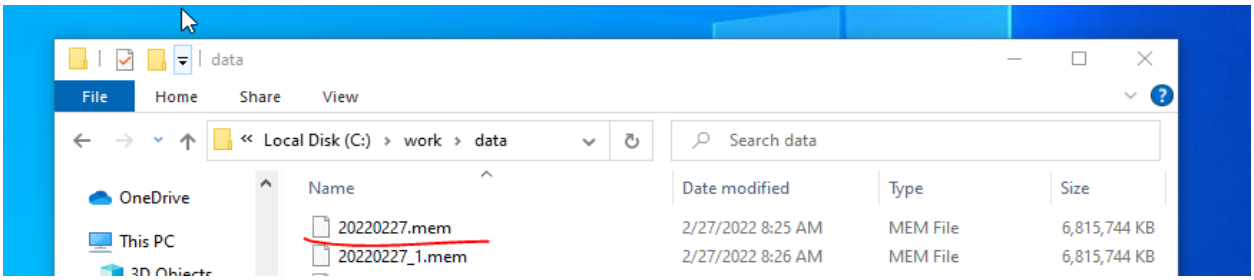
```
C:\work\tools\SysinternalsSuite>listdlls notepad.exe

Listdlls v3.2 - Listdlls
Copyright (C) 1997-2016 Mark Russinovich
Sysinternals

------------------------------------------------------------------------------
notepad.exe pid: 6604
Command line: "C:\Windows\system32\NOTEPAD.EXE" C:\Users\Hacker\Desktop\New Text Document.txt

Base               Size      Path
0x000000007ff50000  0x38000   C:\Windows\system32\NOTEPAD.EXE
0x0000000008130000  0x1f6000  C:\Windows\SYSTEM32\ntdll.dll
0x0000000006b40000  0xbd000   C:\Windows\System32\KERNEL32.DLL
0x00000000058a0000  0x2c8000  C:\Windows\System32\KERNELBASE.dll
0x00000000077f0000  0x2a000   C:\Windows\System32\GDI32.dll
0x0000000005cc0000  0x22000   C:\Windows\System32\win32u.dll
0x0000000005cf0000  0x109000  C:\Windows\System32\gdi32full.dll
0x0000000005c20000  0x9d000   C:\Windows\System32\msvcp_win.dll
0x0000000006090000  0x100000  C:\Windows\System32\ucrtbase.dll
0x0000000007e60000  0x1a0000  C:\Windows\System32\USER32.dll
0x0000000006c00000  0x355000  C:\Windows\System32\combase.dll
0x0000000006190000  0x124000  C:\Windows\System32\RPCRT4.dll
0x0000000007690000  0xae000   C:\Windows\System32\shcore.dll
0x0000000007c00000  0x9e000   C:\Windows\System32\msvcrt.dll
0x00000000ed260000  0x29b000  C:\Windows\WinSxS\amd64_microsoft.windows.common-controls_6595b64144ccf1df_6.0.19041.488_none_ca04af0
81b815d21\COMCTL32.dll
0x0000000007d80000  0x30000   C:\Windows\System32\IMM32.DLL
0x0000000006010000  0x7f000   C:\Windows\System32\bcryptPrimitives.dll
```

listdll *is another windows utility that list all the dynamic link library that are loaded into the processes.*

PART 3: RAM ACQUISITION







Ram acquisition *is the process of creating an active image of the content running on the RAM and storing it into a file such as "**.mem**"*

SUMMARY

This whole lab was divided into three parts. First part taught us about the windows crash dump, it showed us how to create memory dump if our system crashes accidently. It also taught us about the tool named dumpchk that reads the memory dump file created above. Second part taught us how to use the tools pslist, procdump, Bintext, handle and listdll and retrieve the information about the memory dump or processes. Third part taught us about the ram acquisition that can be done using a software called Belkasoft. This creates a copy of content stored in the RAM so it can be investigated later.