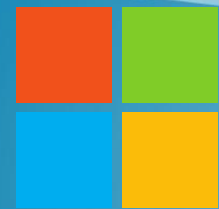


Microsoft Windows
Operating System

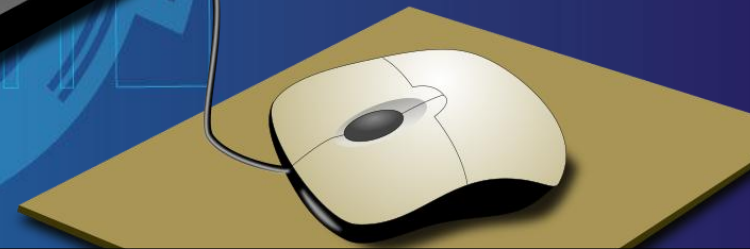
Introduction to USER mode & KERNEL mode

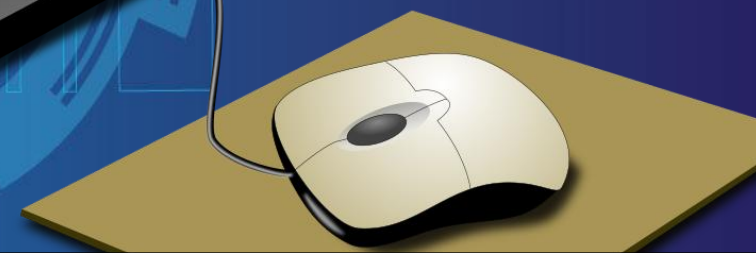
Youtube link: <https://youtu.be/RK8mRlf5bMg>

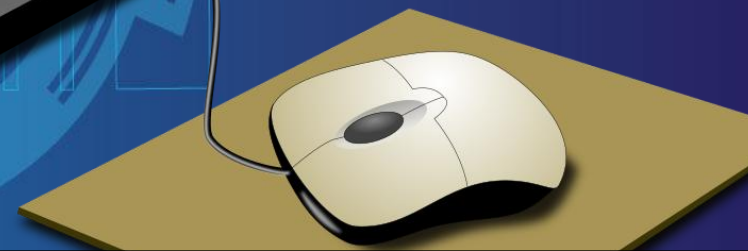
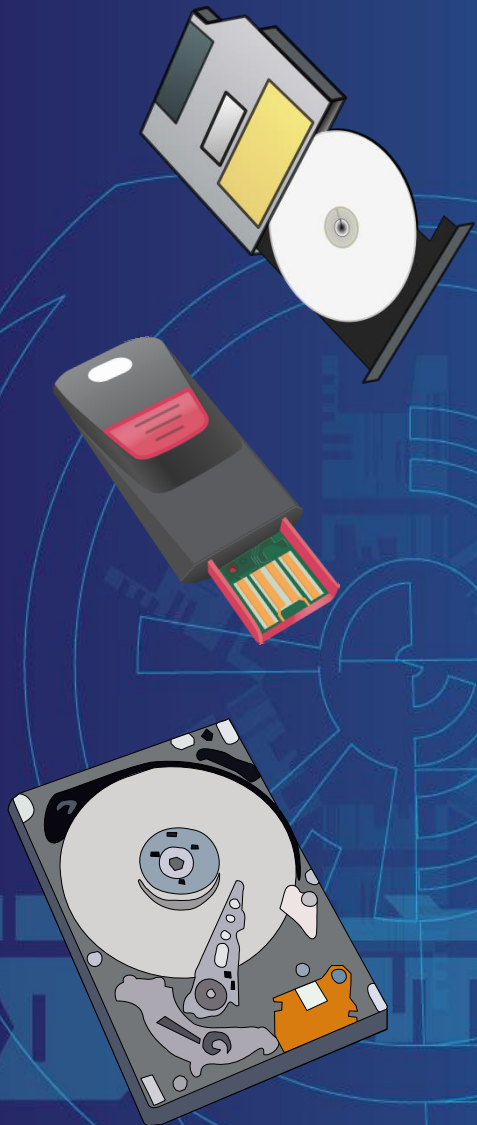


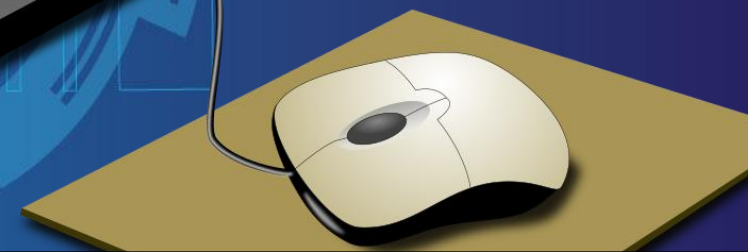
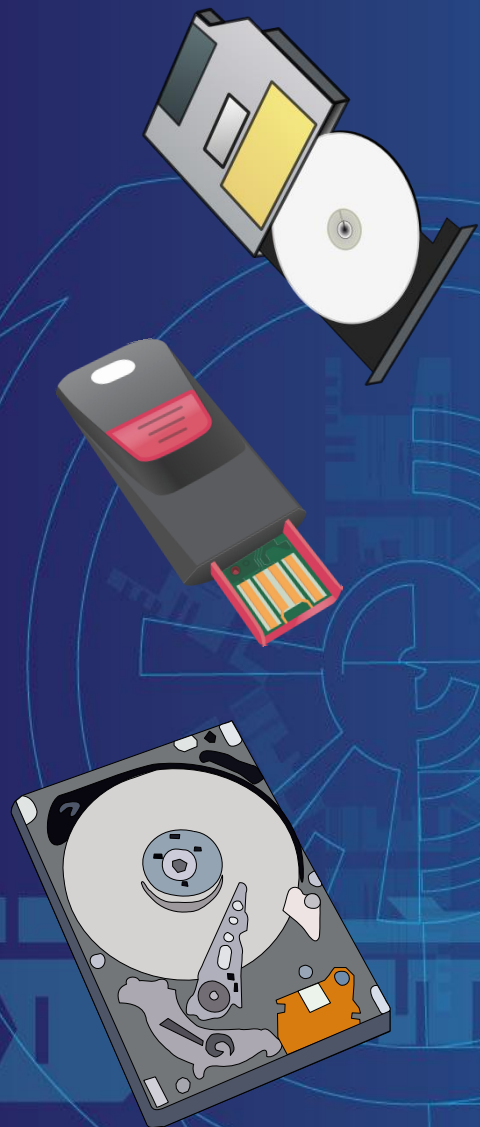
Microsoft Windows Operating System

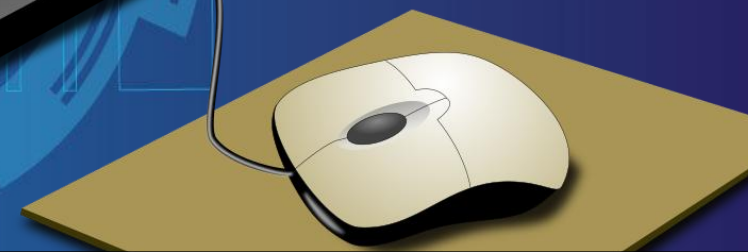
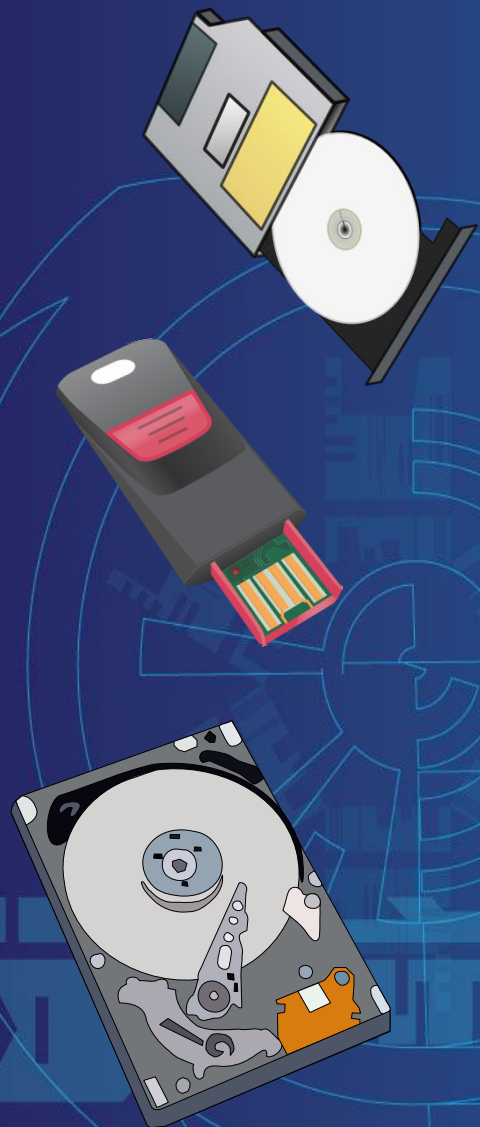
Youtube link: <https://youtu.be/RK8mRlf5bMg>





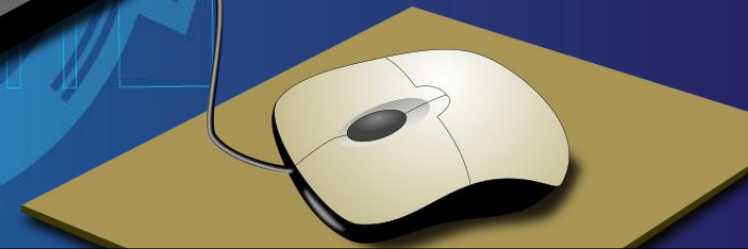
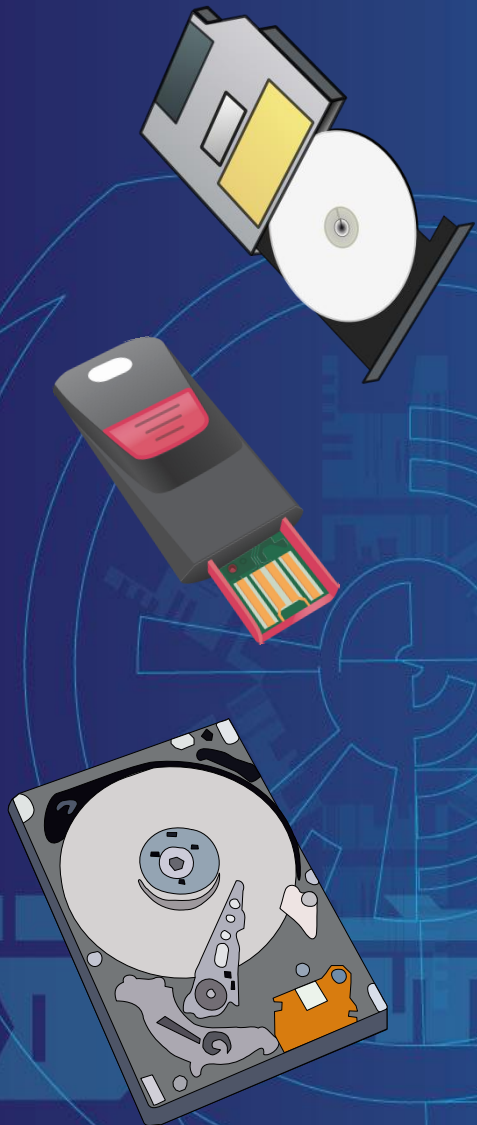






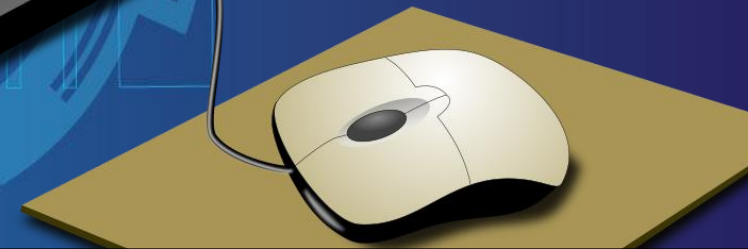
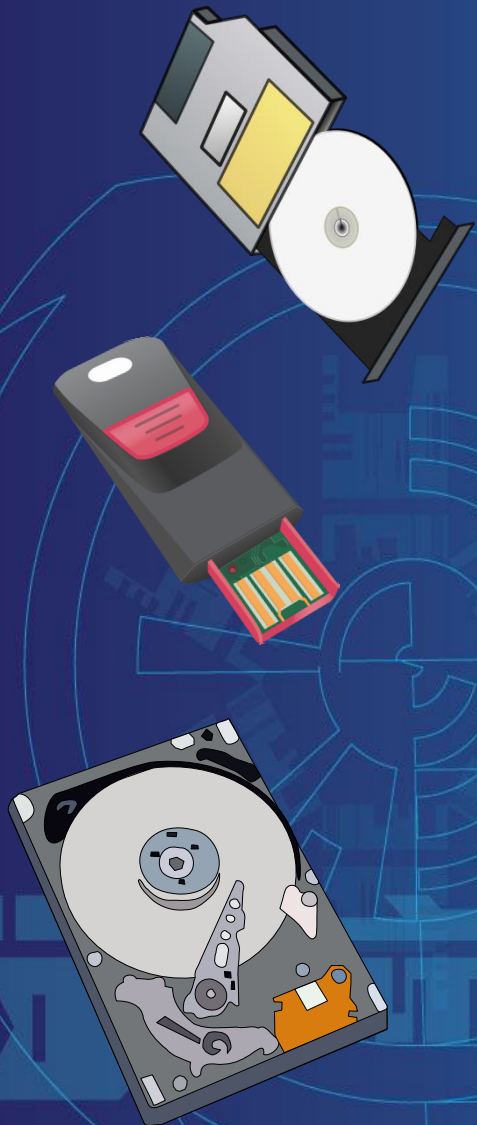
User Applications

Youtube link: <https://youtu.be/RK8mRlf5bMg>



User Applications

Youtube link: <https://youtu.be/RK8mRlf5bMg>



The image features a central computer monitor with a blue screen. On the screen, there are two rounded rectangular boxes: a light green one at the top and a grey one at the bottom. To the left of the monitor, there are icons for a CD/DVD drive, a USB drive, and a hard disk. To the right, there is an icon of a printer and a keyboard. In front of the monitor is a black keyboard and a white mouse on a yellow mousepad. The background is dark blue with faint, glowing circuit-like patterns.

User Applications

Windows Kernel

Youtube link: <https://youtu.be/RK8mRlf5bMg>



The diagram illustrates the Windows operating system architecture. A central monitor displays two stacked boxes: a green box at the top labeled 'User Applications' and a grey box at the bottom labeled 'Windows Kernel'. A large orange arrow points from the 'User Applications' box down to the 'Windows Kernel' box. The monitor is surrounded by various computer peripherals: a CD/DVD drive and a floppy disk on the top left, a USB drive on the middle left, a hard drive on the bottom left, a printer on the top right, a keyboard on the bottom right, and a mouse on the bottom right. The background is a dark blue with faint, stylized circuit patterns.

User Applications

Windows Kernel

Youtube link: <https://youtu.be/RK8mRlf5bMg>



The diagram illustrates the Windows operating system architecture. A central computer monitor displays two main components: 'User Applications' in a green box at the top and 'Windows Kernel' in a grey box at the bottom. A large orange arrow points from the User Applications box down to the Windows Kernel box. Surrounding the monitor are various hardware components: a CD/DVD drive with a disc, a USB drive, a hard disk, a printer, a keyboard, and a mouse. Dotted yellow lines connect these hardware components to the Windows Kernel box, indicating that they interact with the kernel. The background is a dark blue with faint, stylized circuit patterns.

User Applications

Windows Kernel

Youtube link: <https://youtu.be/RK8mRlf5bMg>



The diagram illustrates the Windows operating system architecture. At the center is a large monitor displaying two boxes: a green box at the top labeled 'User Applications' and a grey box at the bottom labeled 'Windows Kernel'. An orange arrow points from the User Applications box down to the Windows Kernel box, and a green arrow points from the Windows Kernel box up to the User Applications box. Surrounding the monitor are various hardware components: a CD/DVD drive, a USB drive, a hard disk, a printer, a keyboard, and a mouse. Dotted yellow lines connect these hardware components to the Windows Kernel box, indicating their interaction with the operating system. The background is a dark blue with faint circuit patterns.

User Applications

Windows Kernel

Youtube link: <https://youtu.be/RK8mRlf5bMg>

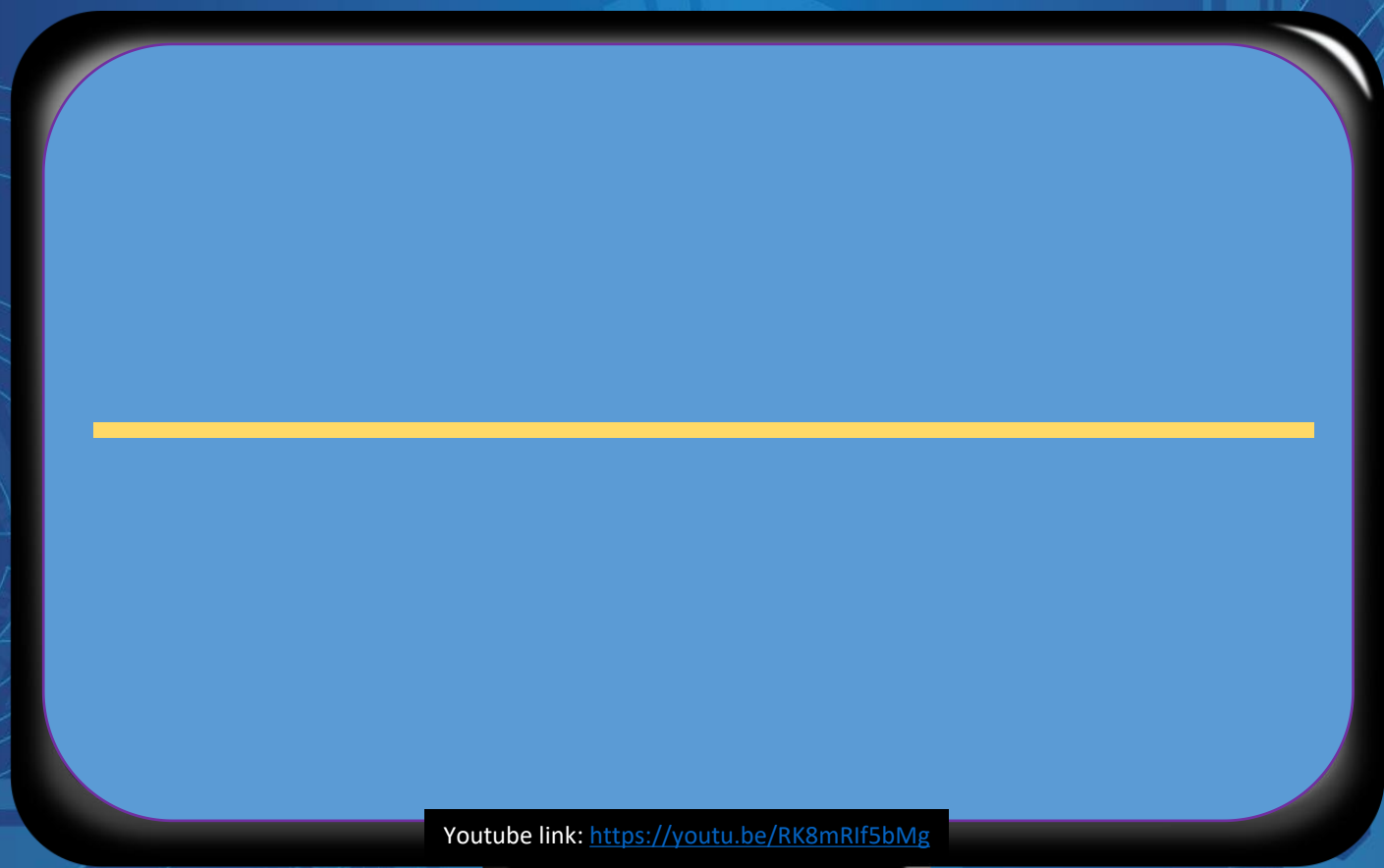


The diagram illustrates the Windows operating system architecture. At the center is a computer monitor displaying two main components: 'User Applications' in a green box at the top and 'Windows Kernel' in a grey box at the bottom. A thick orange arrow points from the User Applications box down to the Windows Kernel box, while a thick green arrow points from the Windows Kernel box up to the User Applications box, indicating a bidirectional flow of control and data. Surrounding the monitor are various hardware components: a CD/DVD drive with a disc, a USB flash drive, a hard disk drive, a printer, a keyboard, and a mouse. Dotted yellow lines connect these hardware components to the Windows Kernel box, signifying that they all interface with the kernel. The entire scene is set against a dark blue background with faint, stylized circuit patterns.

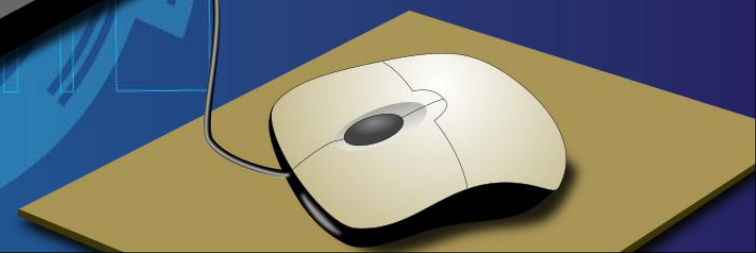
User Applications

Windows Kernel

Youtube link: <https://youtu.be/RK8mRlf5bMg>

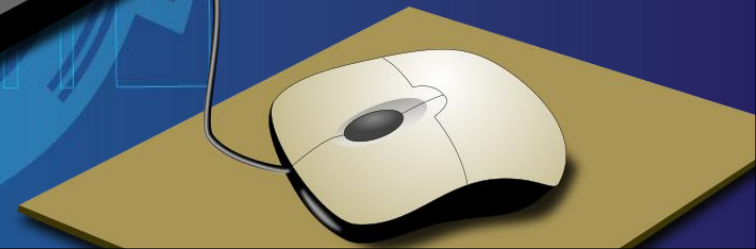


Youtube link: <https://youtu.be/RK8mRlf5bMg>



User Mode

Youtube link: <https://youtu.be/RK8mRlf5bMg>





The image shows a computer monitor with a black frame. The screen is divided into two horizontal sections. The top section is light blue and contains the text 'User Mode'. The bottom section is dark blue and contains the text 'Kernel Mode'. A yellow horizontal line separates the two sections. Below the monitor, there is a black keyboard and a white mouse on a yellow mousepad. The background is a dark blue gradient with faint, stylized circuitry and gear patterns.

User Mode

Kernel Mode

Youtube link: <https://youtu.be/RK8mRlf5bMg>



The image shows a computer monitor with a black frame. The screen is divided into two horizontal sections. The top section is light blue and contains the text 'User Mode'. The bottom section is dark blue and contains the text 'Kernel Mode'. A yellow horizontal line separates the two sections. Below the monitor, there is a black keyboard and a white mouse on a yellow mousepad. The background is a dark blue gradient with faint, stylized circuitry and gear patterns.

User Mode

Kernel Mode

Youtube link: <https://youtu.be/RK8mRlf5bMg>

User Mode

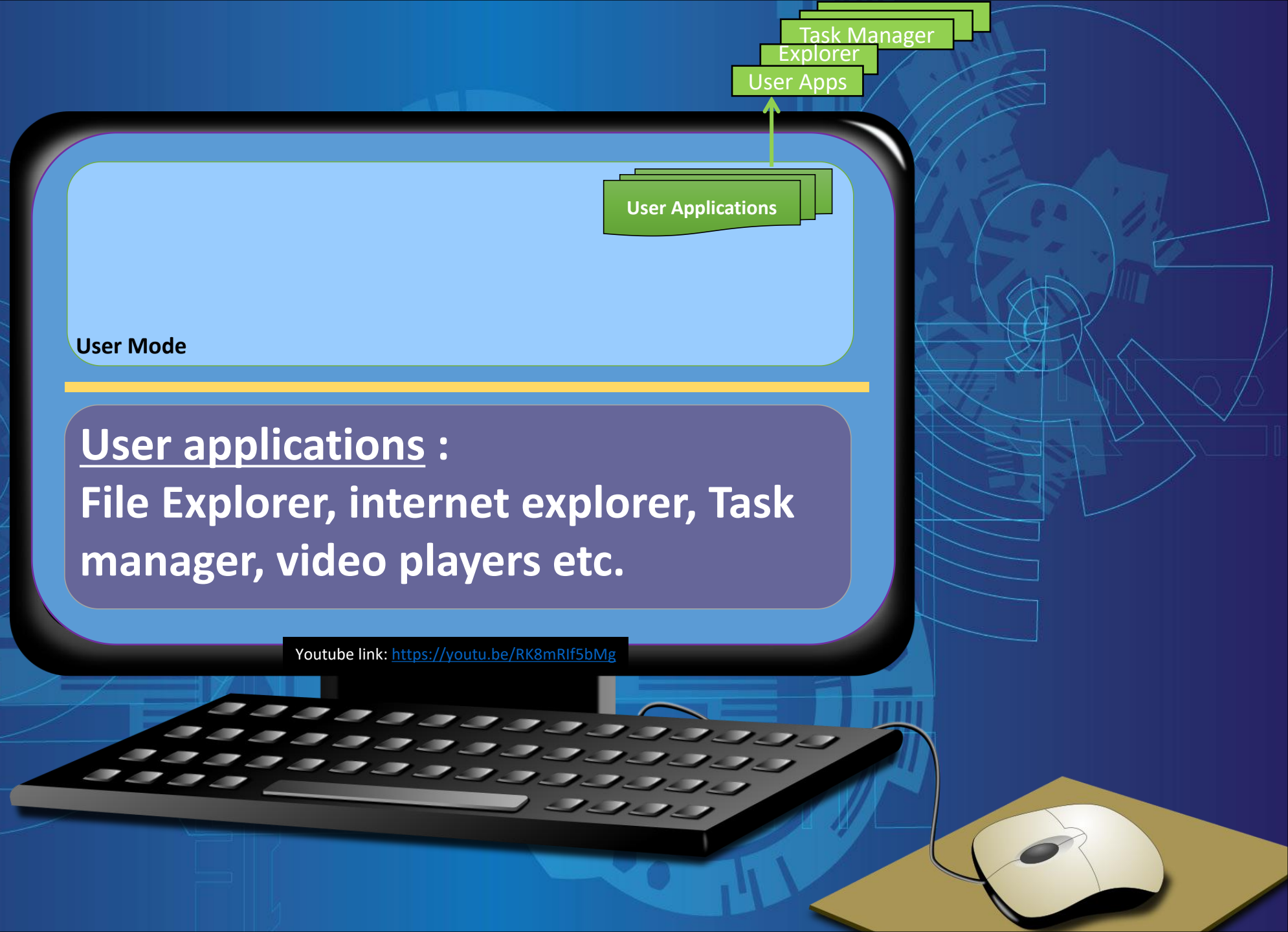
Youtube link: <https://youtu.be/RK8mRlf5bMg>

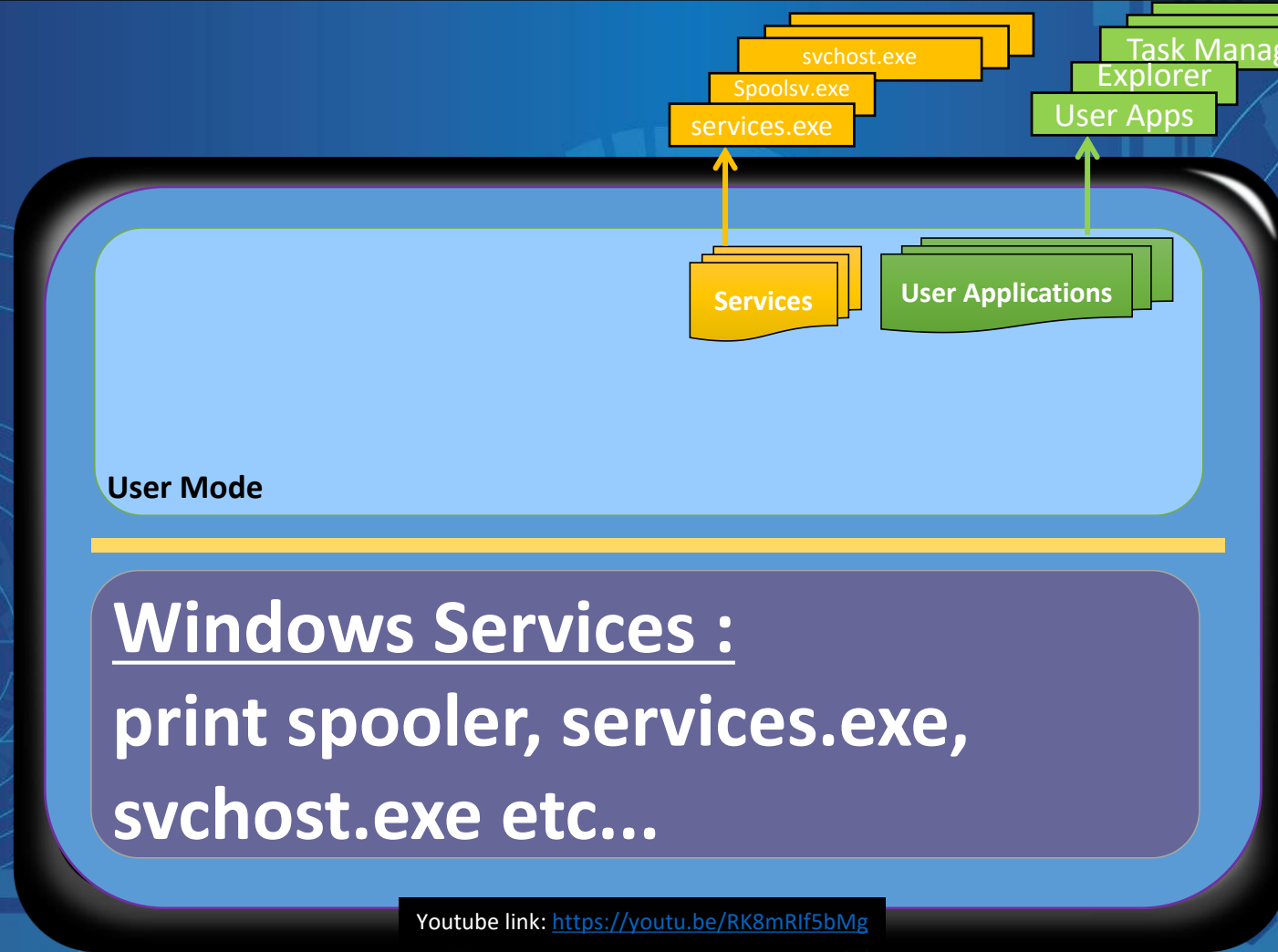


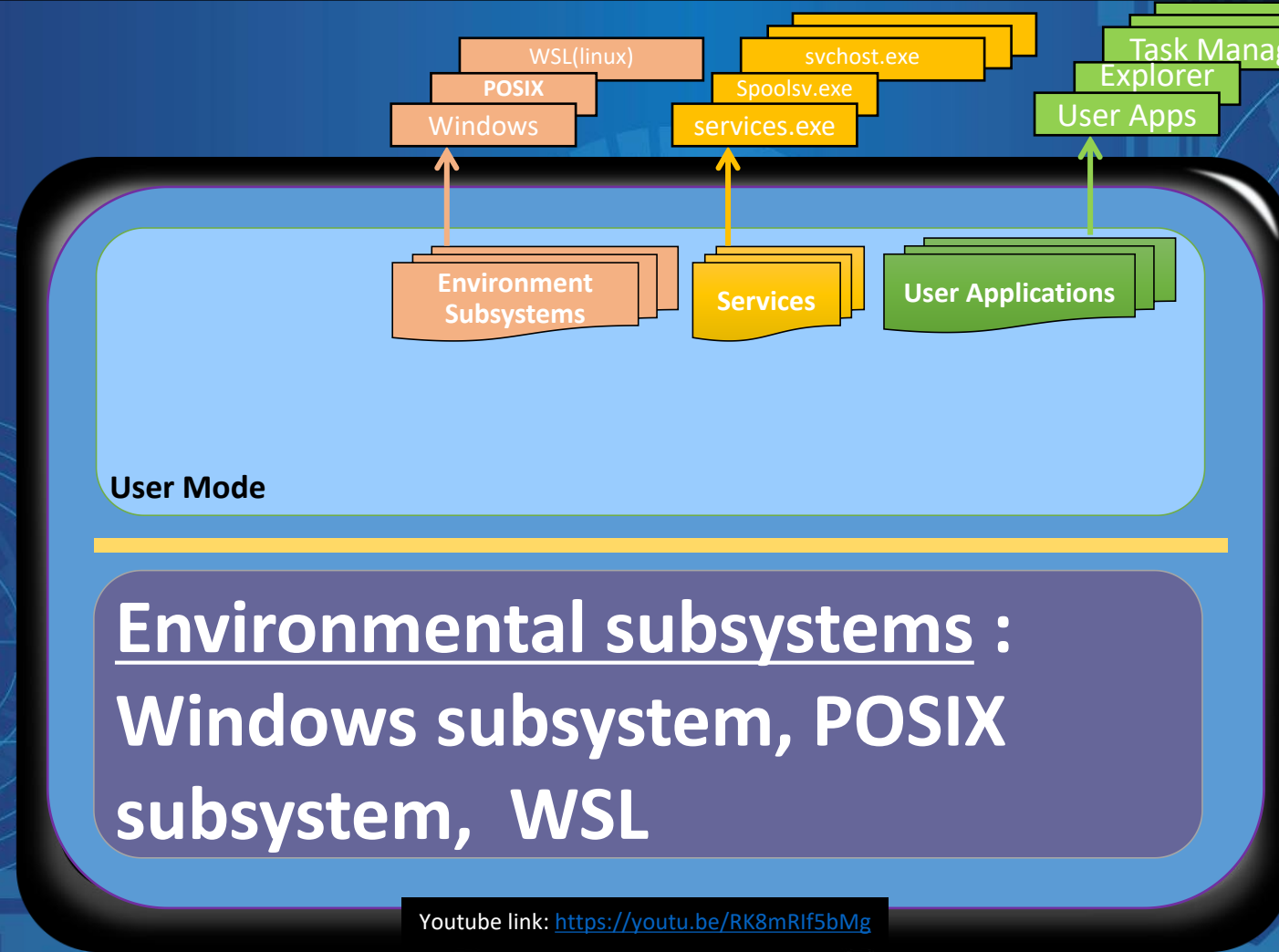
User Applications

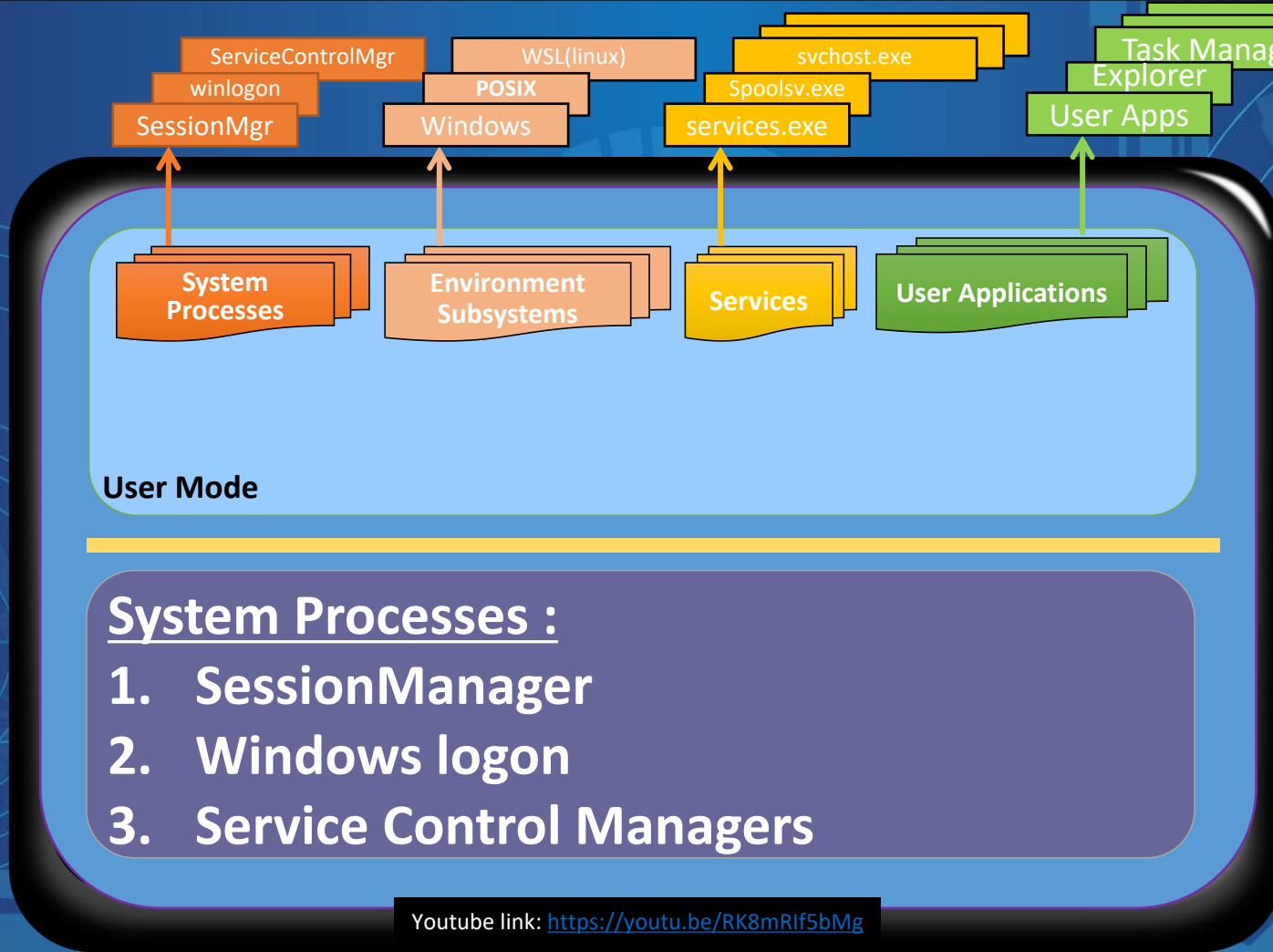
User Mode

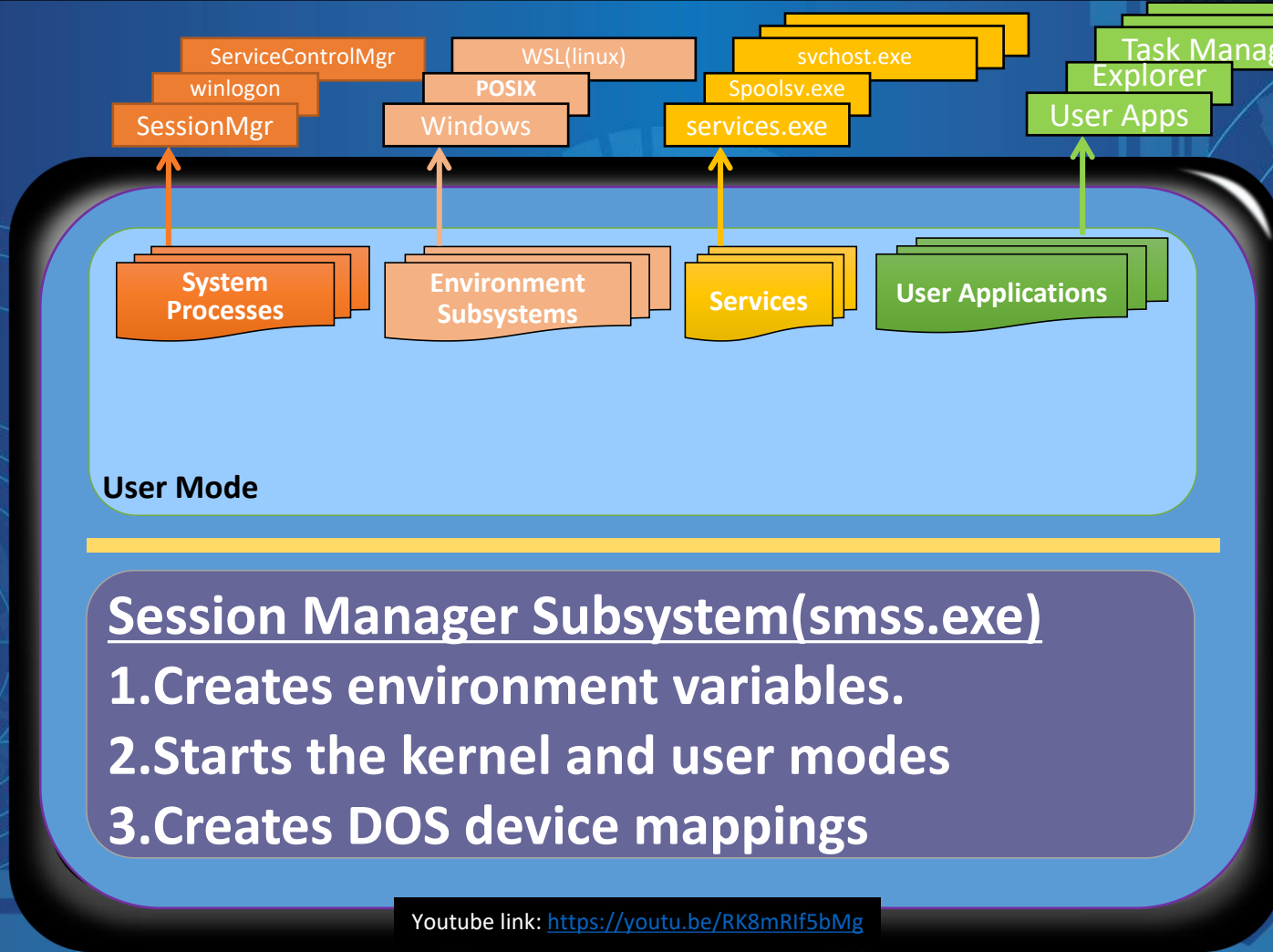
Youtube link: <https://youtu.be/RK8mRlf5bMg>

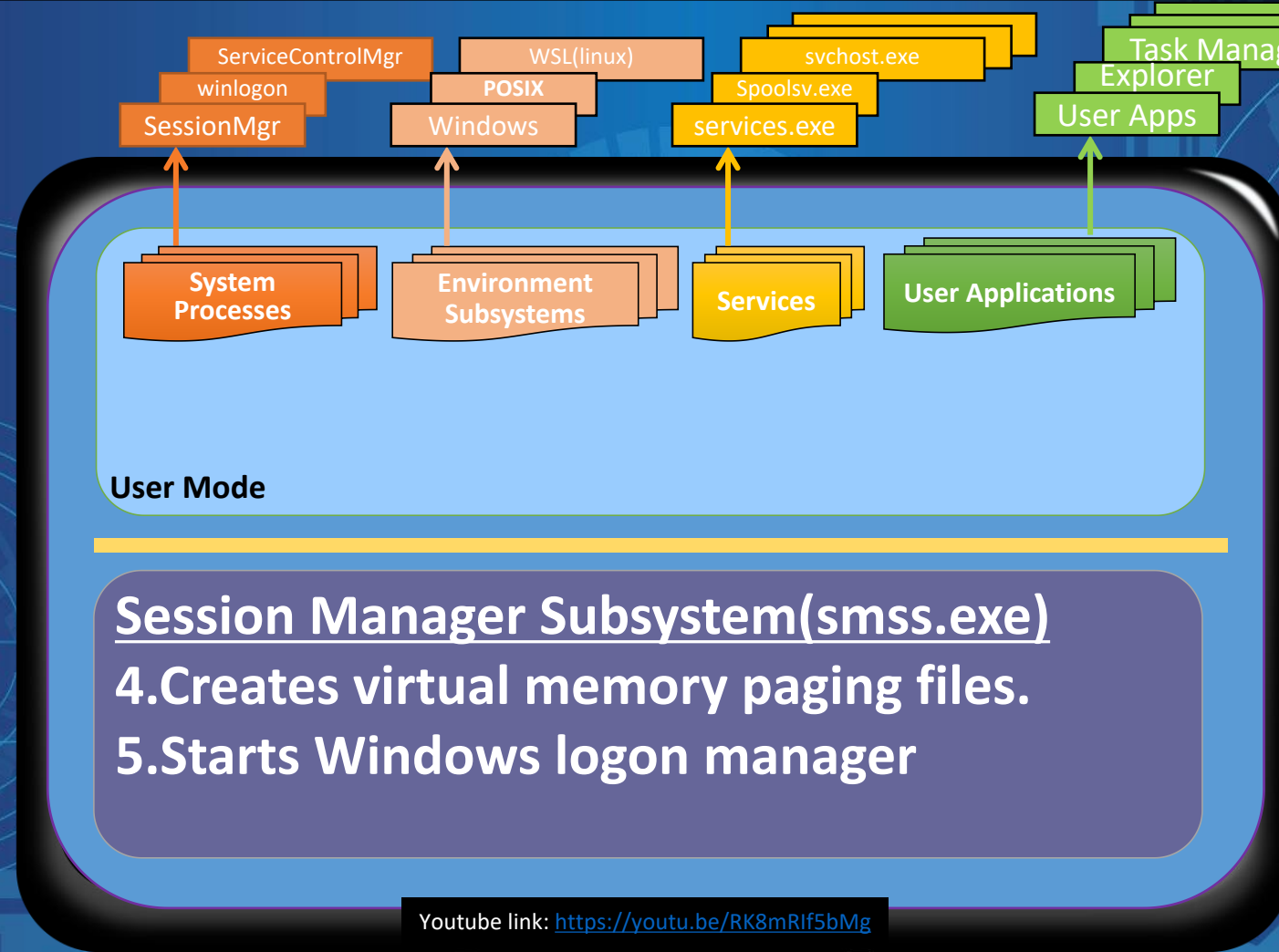


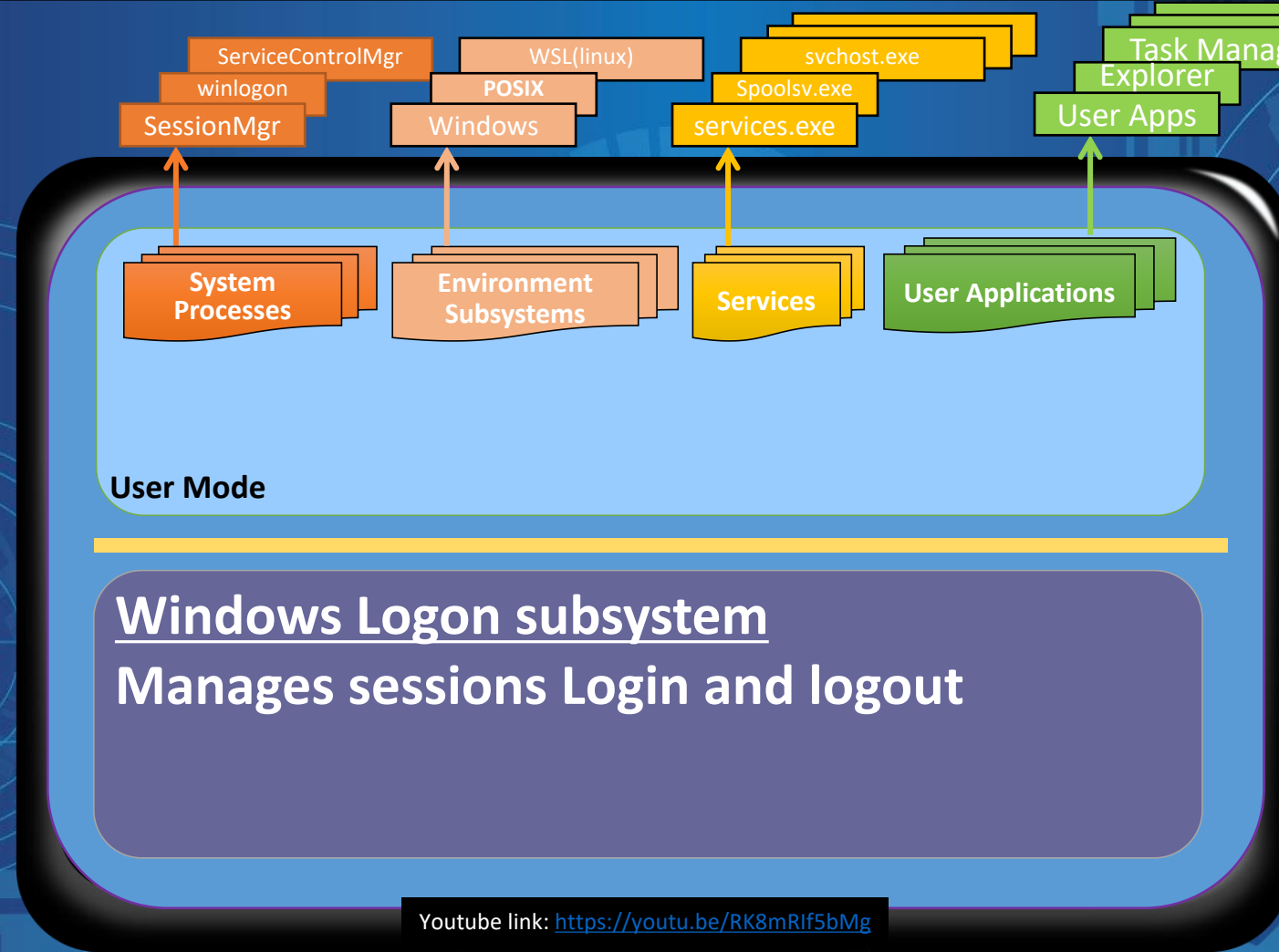


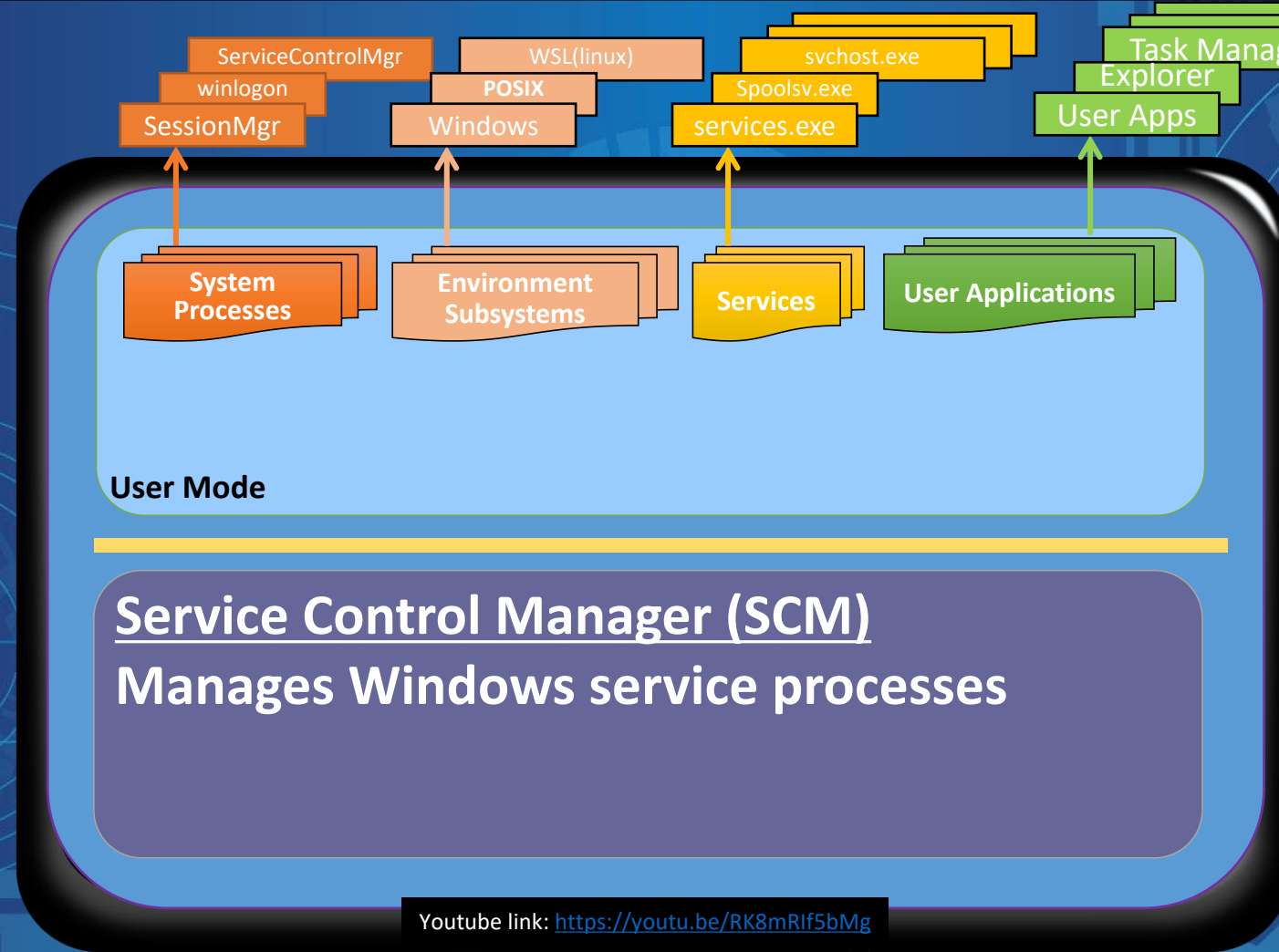


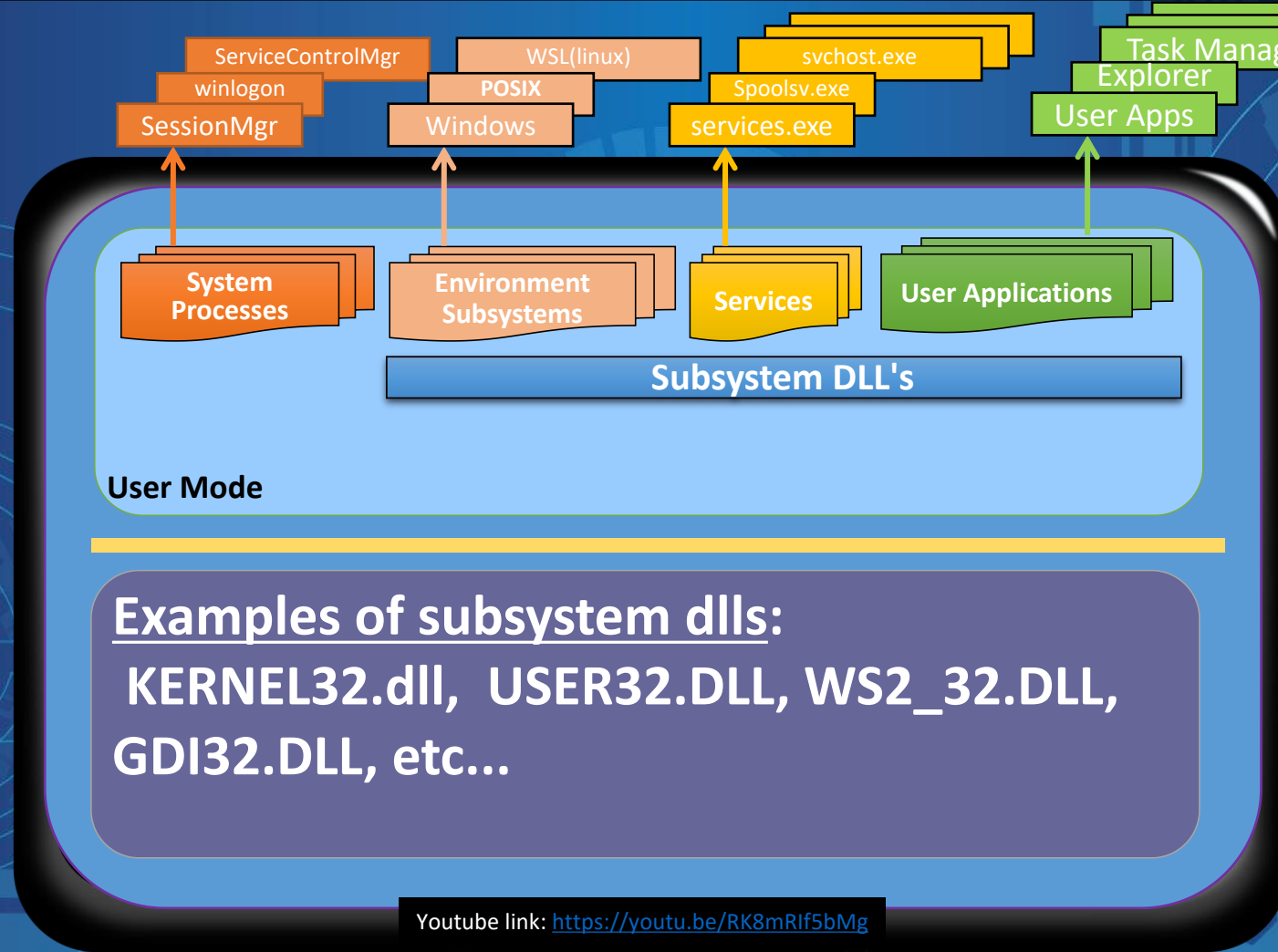








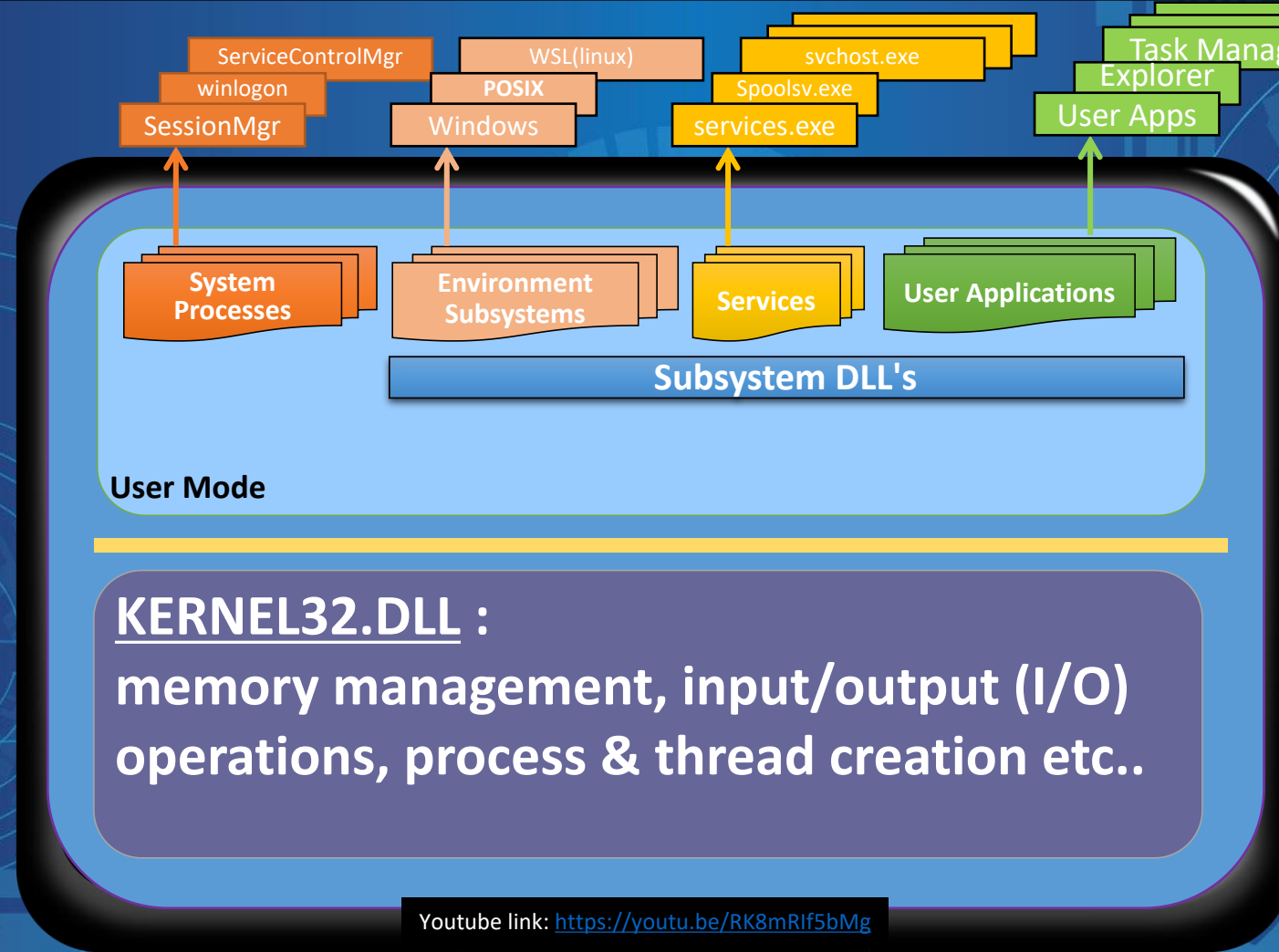


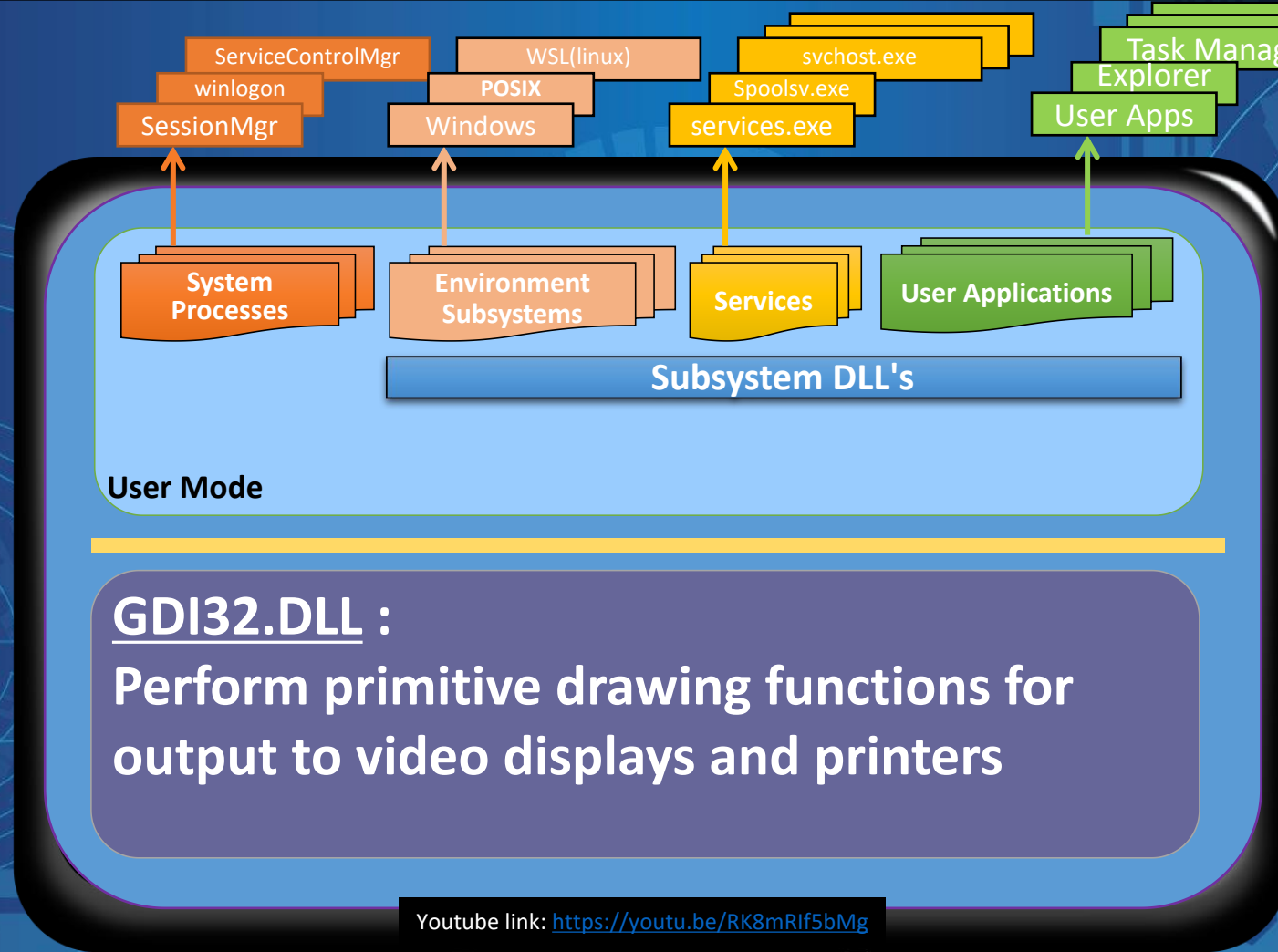


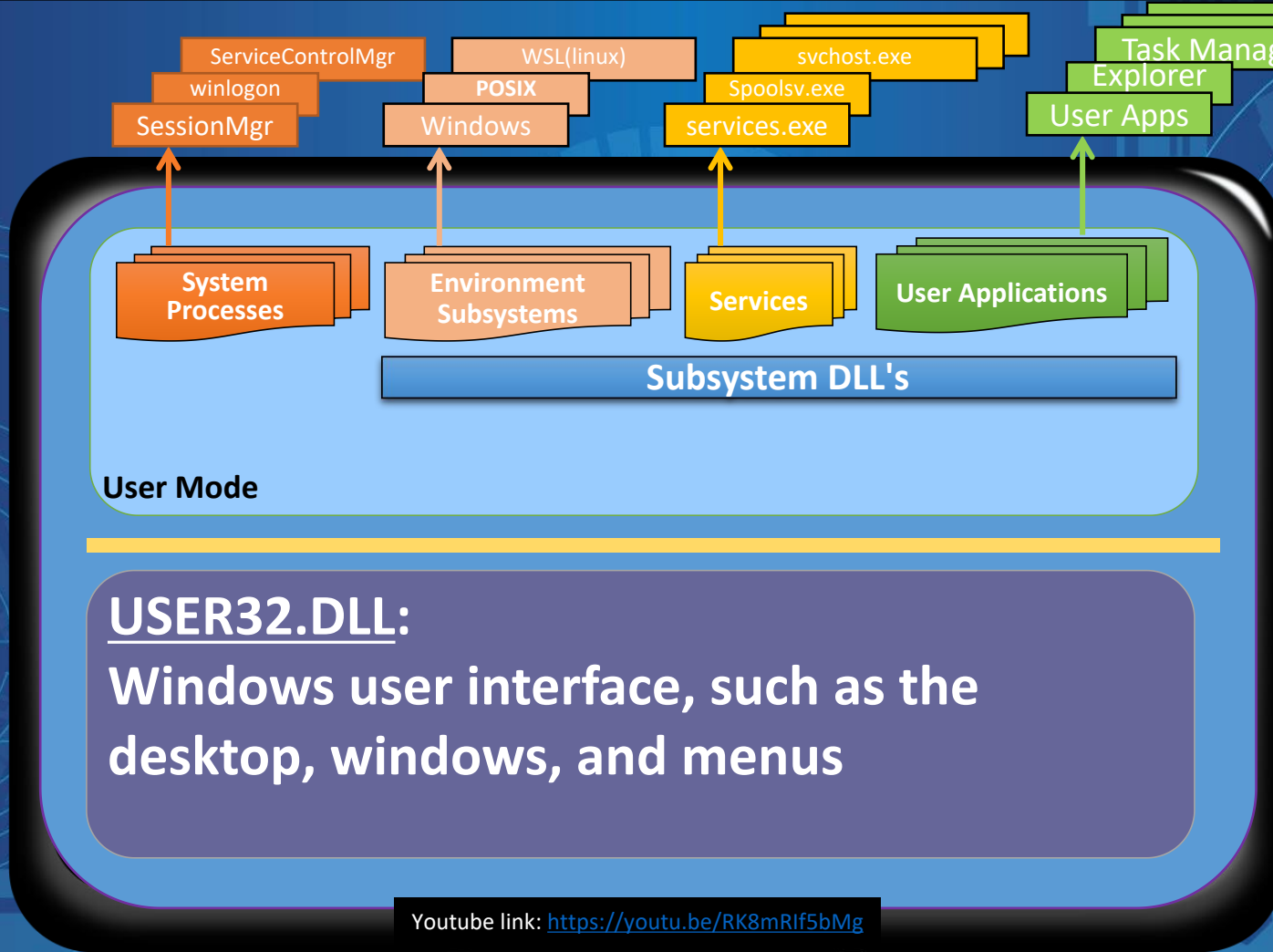
Examples of subsystem dlls:
KERNEL32.dll, USER32.DLL, WS2_32.DLL,
GDI32.DLL, etc...

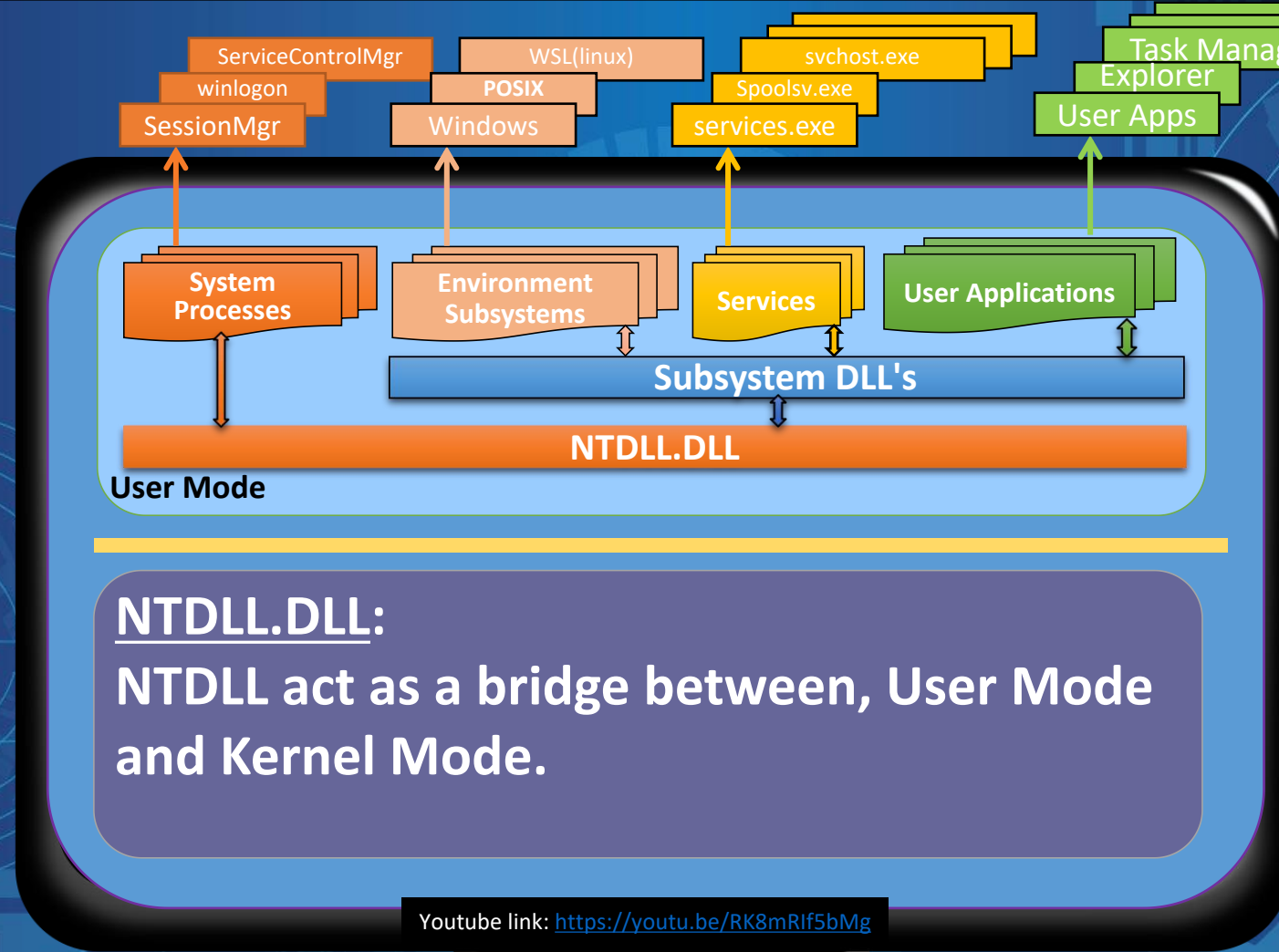
Youtube link: <https://youtu.be/RK8mRlf5bMg>

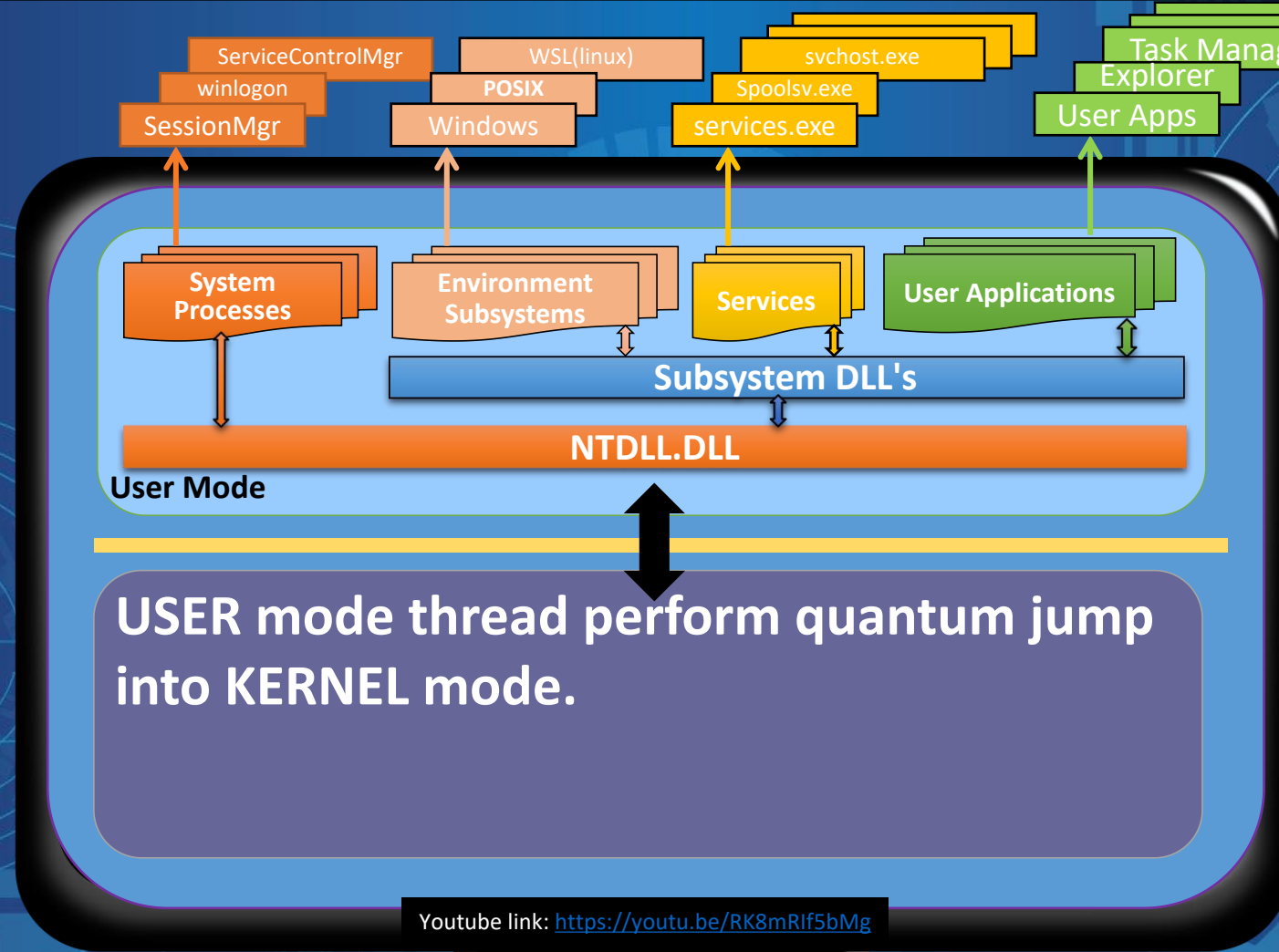






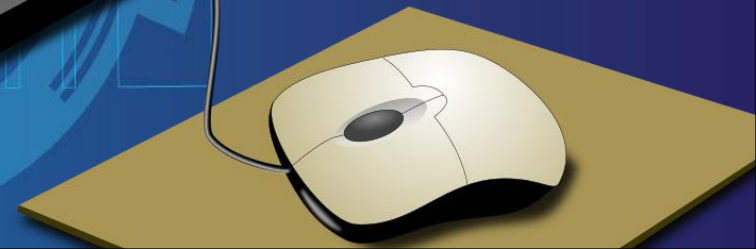






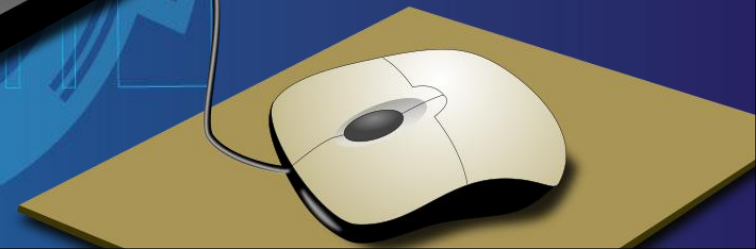
Kernel Mode

Youtube link: <https://youtu.be/RK8mRlf5bMg>



Kernel Mode

Youtube link: <https://youtu.be/RK8mRlf5bMg>

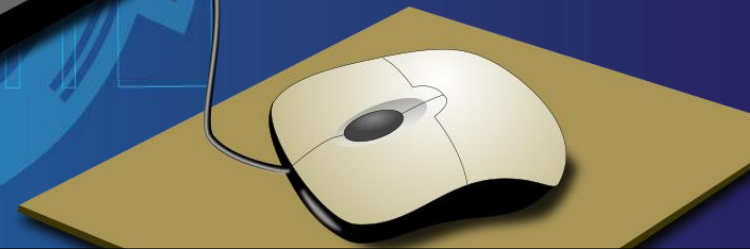




TRAP/SYSCALL

Kernel Mode

Youtube link: <https://youtu.be/RK8mRlf5bMg>



Kernel Mode

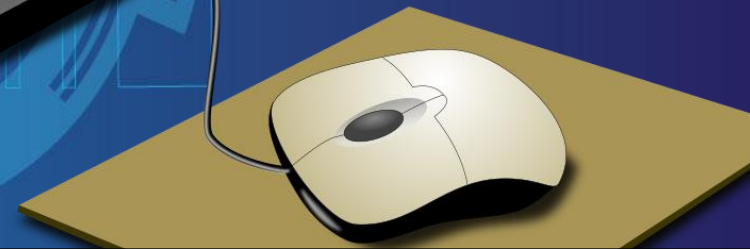


TRAP/SYSCALL

Executive

**Performs I/O, object management, security
and process management.**

Youtube link: <https://youtu.be/RK8mRlf5bMg>



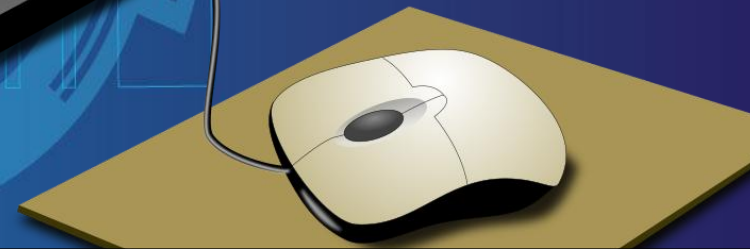
Kernel Mode



TRAP/SYSCALL

Executive

Youtube link: <https://youtu.be/RK8mRlf5bMg>



Kernel Mode

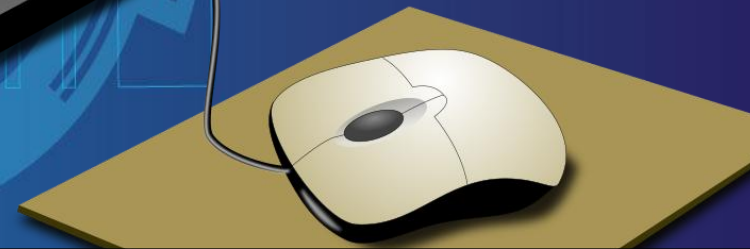


TRAP/SYSCALL

Executive Services

Executive

Youtube link: <https://youtu.be/RK8mRlf5bMg>



Kernel Mode



TRAP/SYSCALL

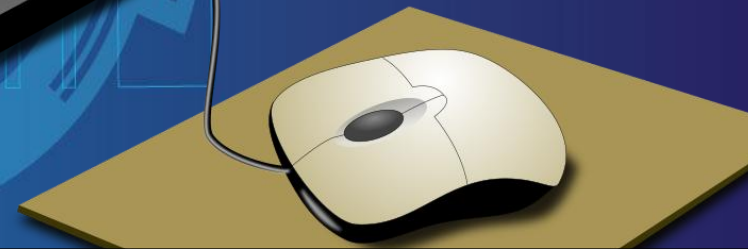
Executive Services

I/O
Mgr

Executive

**Manages : Keyboard, Mice, Disk drivers,
Audio/Video controllers, N/W port etc...**

Youtube link: <https://youtu.be/RK8mRlf5bMg>



Kernel Mode



TRAP/SYSCALL

Executive Services

I/O
Mgr

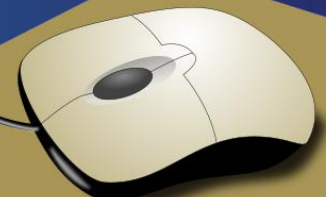
Memory
Mgr

Executive

Manages : memory alloc/dealloc-ation

Supports : memory mapped files, shared
memory etc...

Youtube link: <https://youtu.be/RK8mRlf5bMg>



Kernel Mode



TRAP/SYSCALL

Executive Services

**I/O
Mgr**

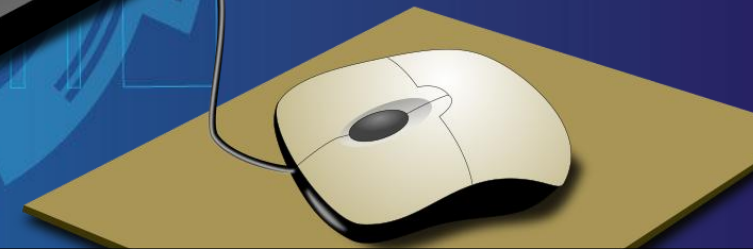
**Memory
Mgr**

**Process/
Thread
Mgr**

Executive

**Manages : Execution of all threads
in a process**

Youtube link: <https://youtu.be/RK8mRlf5bMg>



Kernel Mode



TRAP/SYSCALL

Executive Services

I/O
Mgr

Memory
Mgr

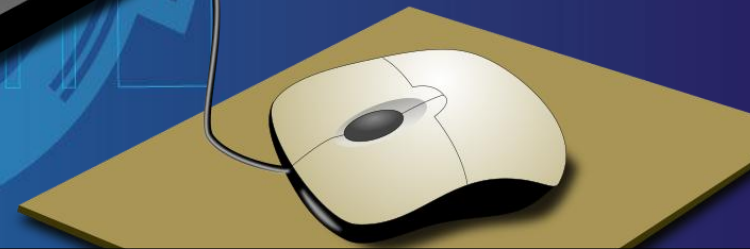
Process/
Thread
Mgr

Config
Mgr

Executive

Manages :Registry

Youtube link: <https://youtu.be/RK8mRlf5bMg>



Kernel Mode



TRAP/SYSCALL

Executive Services

I/O
Mgr

Memory
Mgr

Process/
Thread
Mgr

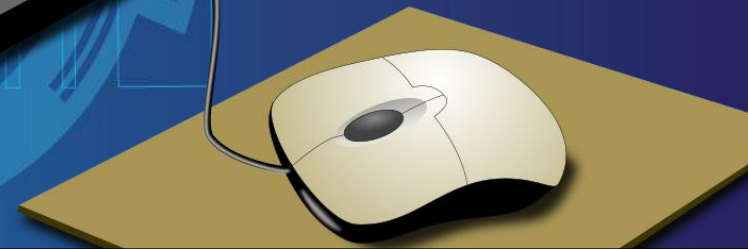
Config
Mgr

Security
Mgr

Executive

Manages : Routines for driver to work with access control (ACL). Access control list (ACL) determines which objects have what security

Youtube link: <https://youtu.be/RK8mRlf5bMg>



Kernel Mode



TRAP/SYSCALL

Executive Services

I/O
Mgr

Memory
Mgr

Process/
Thread
Mgr

Config
Mgr

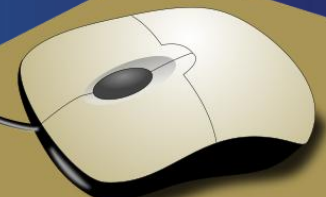
Security
Mgr

Object Manager

Manages:

1. Creation and destruction of Kernel objects

Youtube link: <https://youtu.be/RK8mRlf5bMg>



Kernel Mode



TRAP/SYSCALL

Executive Services

I/O
Mgr

Memory
Mgr

Process/
Thread
Mgr

...

Security
Mgr

Object Manager

Manages:

2. Object namespace database for tracking object information.

Youtube link: <https://youtu.be/RK8mRlf5bMg>



Kernel Mode



TRAP/SYSCALL

Executive Services

I/O
Mgr

Memory
Mgr

Process/
Thread
Mgr

...

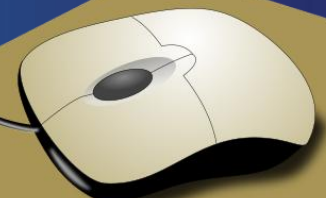
Security
Mgr

Object Manager

Manages:

3. Tracks resources assigned to each process.

Youtube link: <https://youtu.be/RK8mRlf5bMg>



Kernel Mode



TRAP/SYSCALL

Executive Services

I/O
Mgr

Memory
Mgr

Process/
Thread
Mgr

...

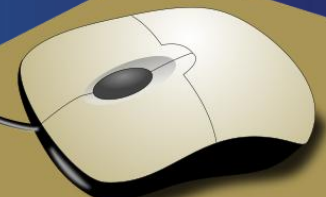
Security
Mgr

Object Manager

Manages:

**4. Access rights for specific objects to
provide security.**

Youtube link: <https://youtu.be/RK8mRlf5bMg>



Kernel Mode



TRAP/SYSCALL

Executive Services

I/O
Mgr

Memory
Mgr

Process/
Thread
Mgr

...

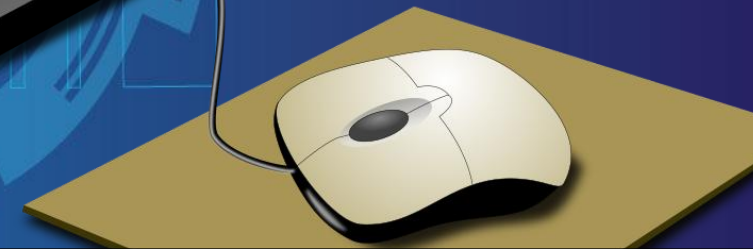
Security
Mgr

Object Manager

Manages:

5. lifetime of an object determining when an object will be automatically destroyed

Youtube link: <https://youtu.be/RK8mRlf5bMg>



Kernel Mode



TRAP/SYSCALL

Executive Services

I/O
Mgr

Memory
Mgr

Process/
Thread
Mgr

...

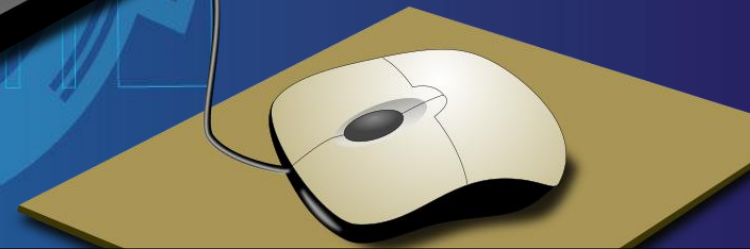
Security
Mgr

Object Manager

Other Device Drivers

Kernel Drivers

Youtube link: <https://youtu.be/RK8mRlf5bMg>



Kernel Mode



TRAP/SYSCALL

Executive Services

I/O
Mgr

Memory
Mgr

Process/
Thread
Mgr

...

Security
Mgr

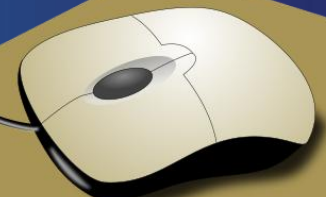
Object Manager

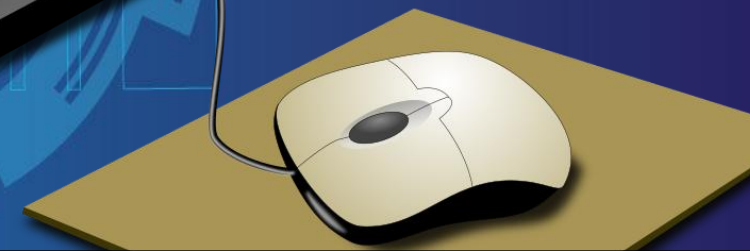
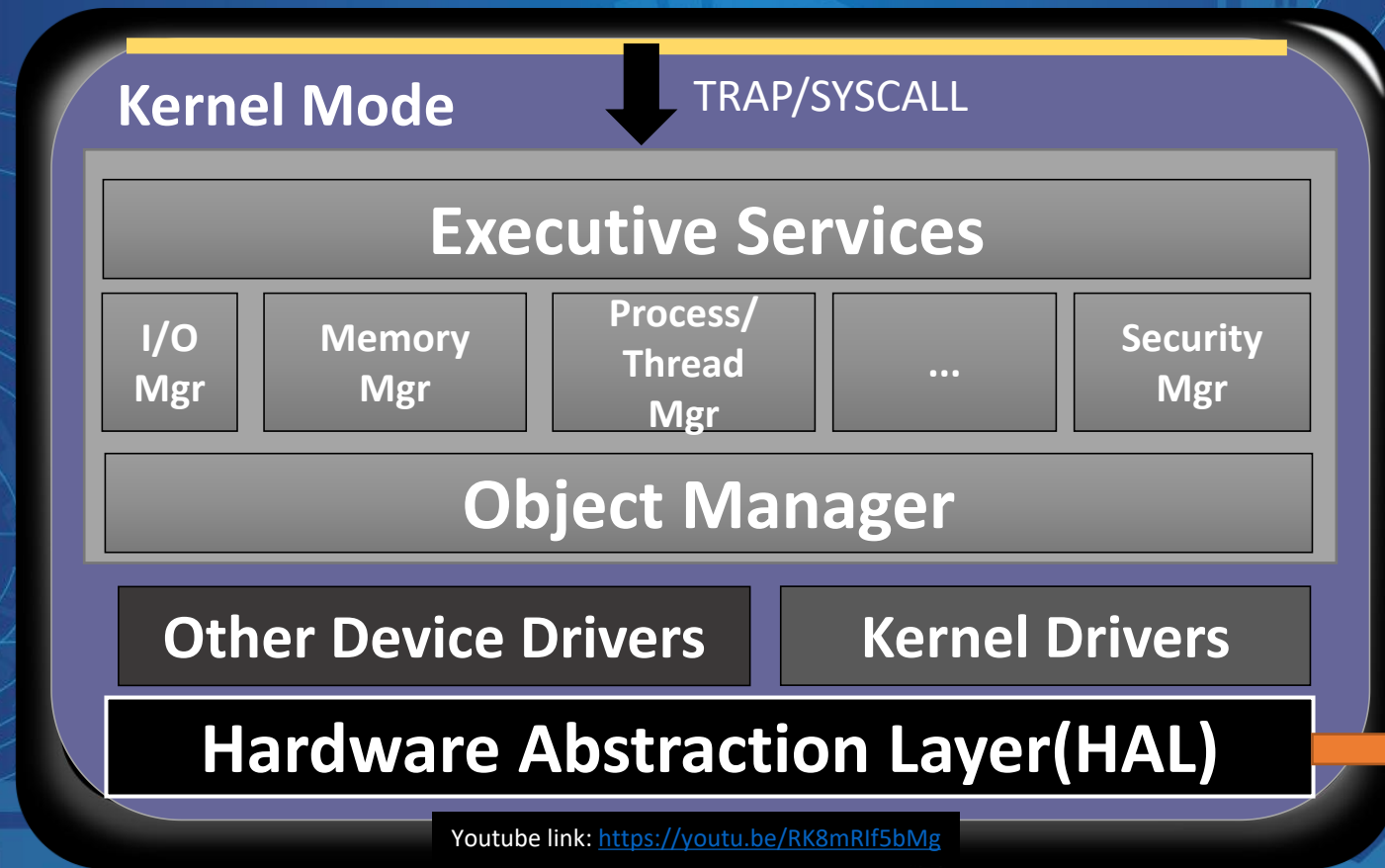
Other Device Drivers

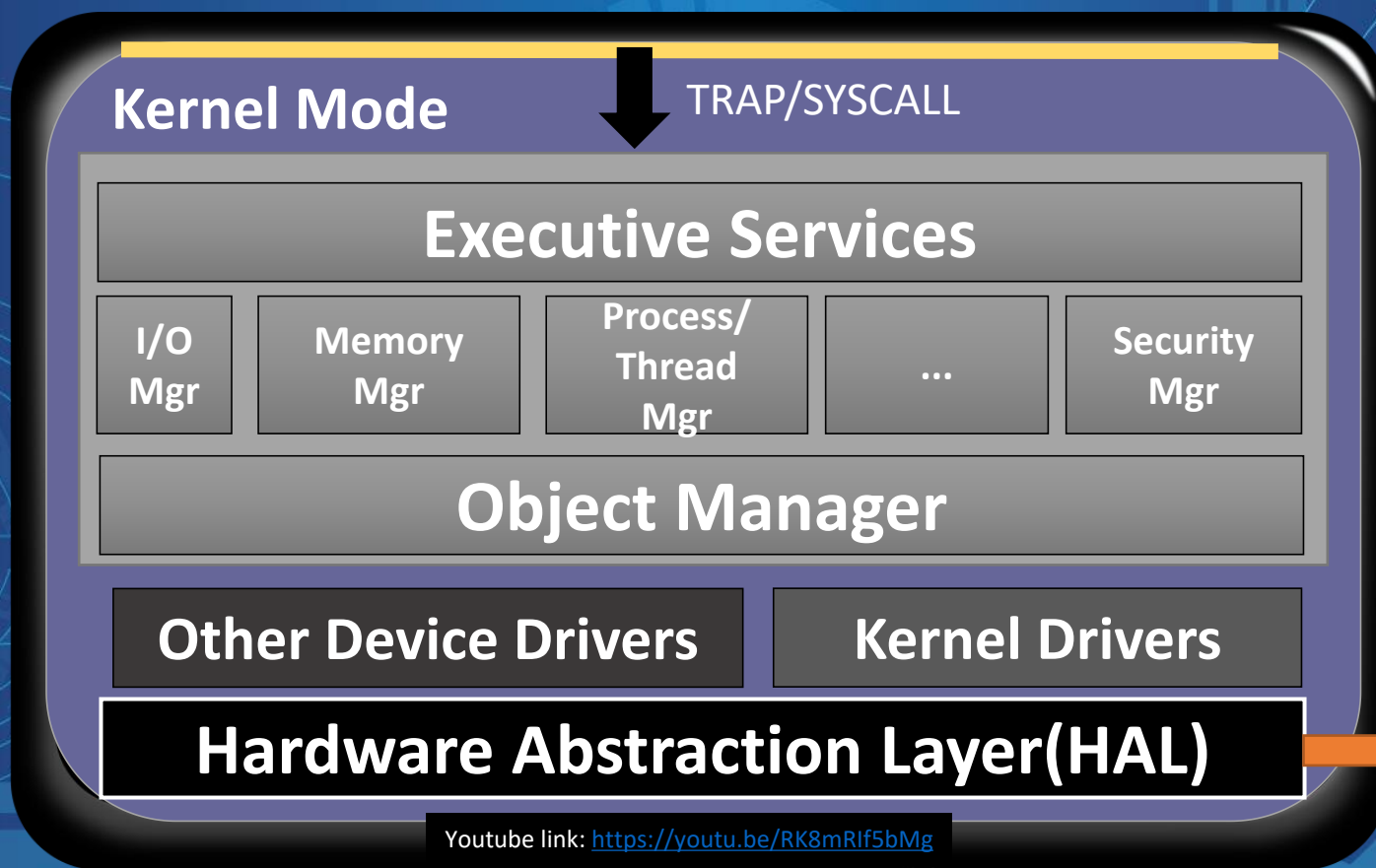
Kernel Drivers

Hardware Abstraction Layer(HAL)

Youtube link: <https://youtu.be/RK8mRlf5bMg>







Hardware



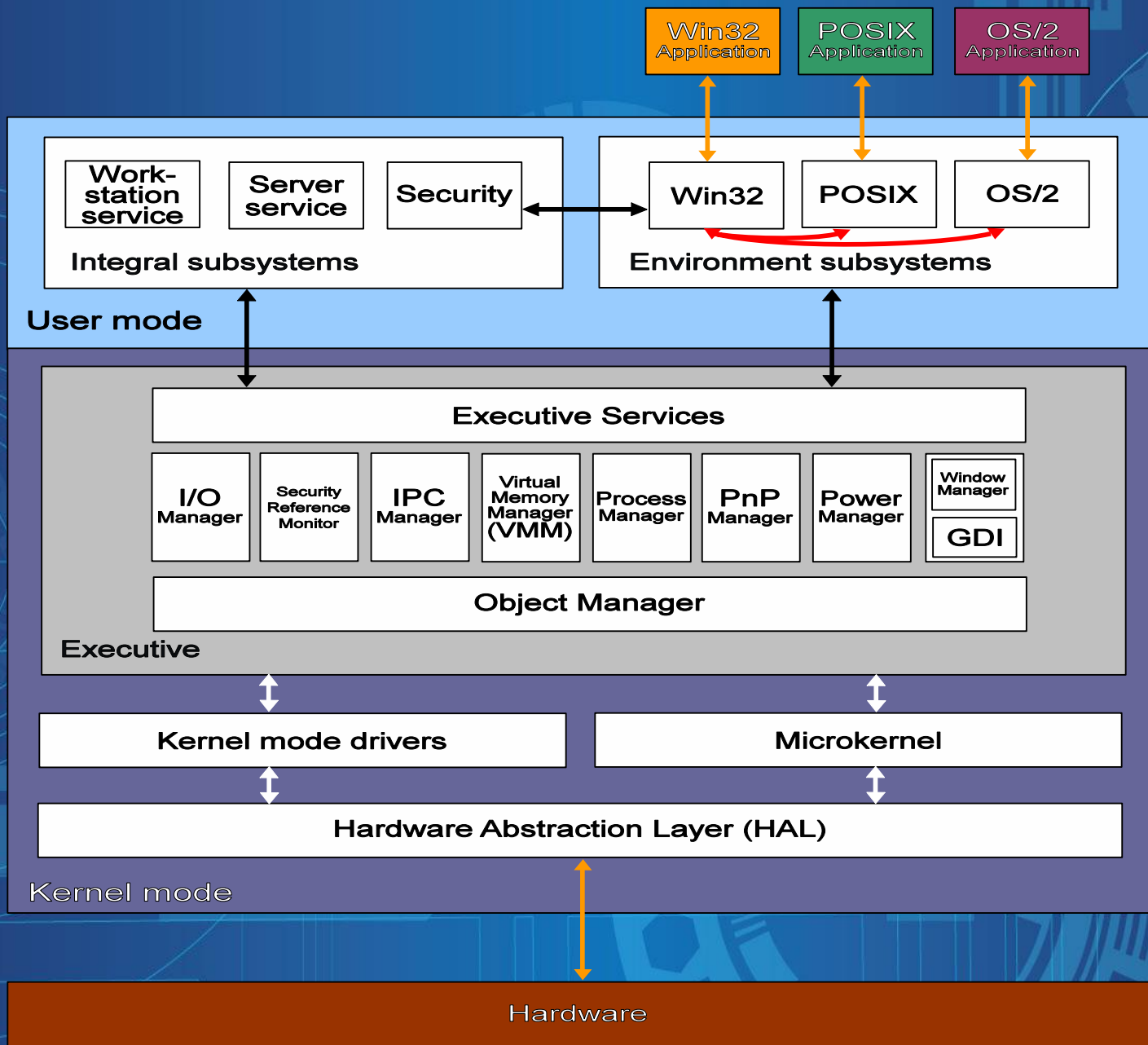


Image Attribution: https://commons.wikimedia.org/wiki/File:Windows_2000_architecture.svg

The original uploader was Grm wnr at English Wikipedia. Later versions were uploaded by Xyzzy n at en.wikipedia. / CC BY-SA (<http://creativecommons.org/licenses/by-sa/3.0/>)

Youtube link: <https://youtu.be/RK8mRlf5bMg>

Scenario : Open a file in hard disk

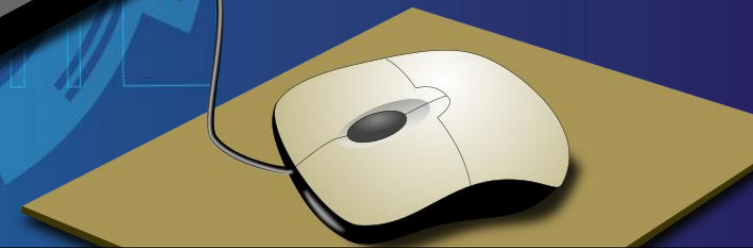
Youtube link: <https://youtu.be/RK8mRlf5bMg>



Scenario : Open a file in hard disk

```
#include <stdio.h>
int main()
{
    FILE* pFile = fopen ( "C:\\\\file.txt", "w+" );
    if( nullptr != pFile ){
        fclose( pFile );
    }
    return 0;
}
```

Youtube link: <https://youtu.be/RK8mRlf5bMg>



Scenario : Open a file in hard disk

```
#include <stdio.h>
int main()
{
    FILE* pFile = fopen ( "C:\\\\file.txt", "w+" );
    if( nullptr != pFile ){
        fclose( pFile );
    }
    return 0;
}
```

Youtube link: <https://youtu.be/RK8mRlf5bMg>



Scenario : Open a file in hard disk

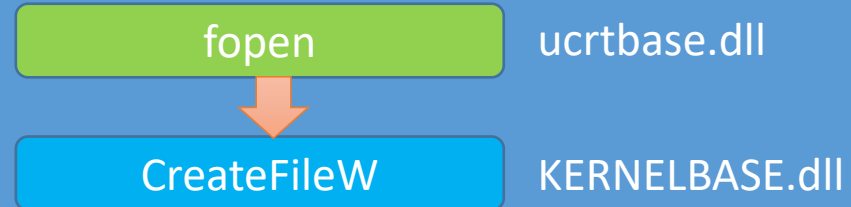
fopen

ucrtbase.dll

Youtube link: <https://youtu.be/RK8mRlf5bMg>

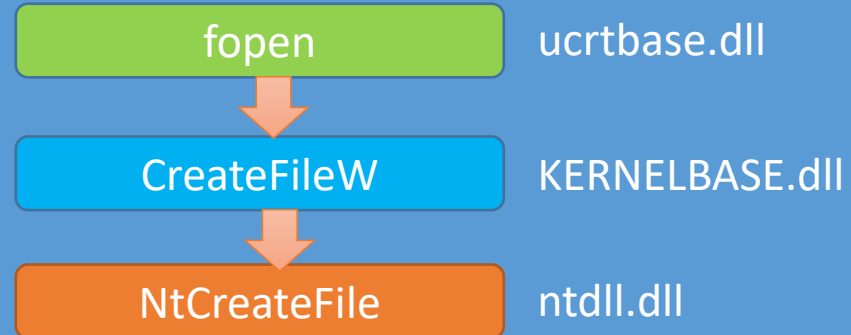


Scenario : Open a file in hard disk



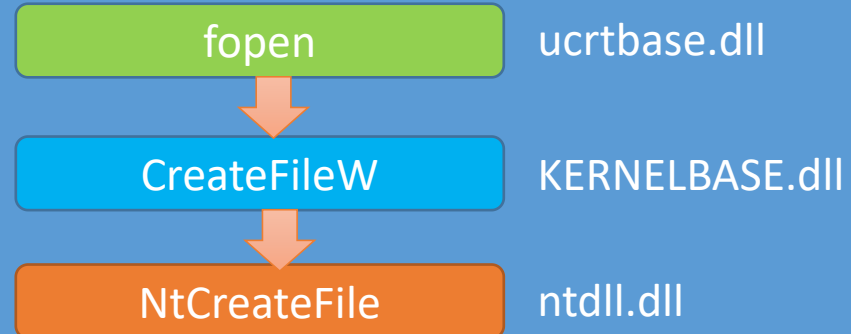
Youtube link: <https://youtu.be/RK8mRlf5bMg>

Scenario : Open a file in hard disk



Youtube link: <https://youtu.be/RK8mRlf5bMg>

Scenario : Open a file in hard disk



Creates a stack frame with arguments.
Places values in registers for specific service required

Youtube link: <https://youtu.be/RK8mRlf5bMg>

Scenario : Open a file in hard disk

fopen

ucrtbase.dll

CreateFileW

KERNELBASE.dll

NtCreateFile

ntdll.dll

Creates a stack frame with arguments.
Places values in registers for specific service required

Execute trap instruction

TRAP/SYSCALL

Youtube link: <https://youtu.be/RK8mRlf5bMg>



Scenario : Open a file in hard disk

fopen

ucrtbase.dll

CreateFileW

KERNELBASE.dll

NtCreateFile

ntdll.dll

Creates a stack frame with arguments.
Places values in registers for specific service required

Execute trap instruction

TRAP/SYSCALL

Youtube link: <https://youtu.be/RK8mRlf5bMg>



Scenario : Open a file in hard disk

```
Call Site
ntdll!NtCreateFile
KERNELBASE!CreateFileInternal+0x2f4
KERNELBASE!CreateFileW+0x66
ucrtbased!create_file+0x5f [minkernel\crts
ucrtbased!_wsopen_nolock+0x17a [minkernel\
ucrtbased!_sopen_nolock+0xad [minkernel\cr
ucrtbased!__crt_char_traits<char>::tsopen_
ucrtbased!common_sopen_dispatch<char>+0x29
ucrtbased!_sopen_dispatch+0x41 [minkernel\
ucrtbased!_sopen_s+0x42 [minkernel\crts\u
ucrtbased!__crt_char_traits<char>::tsopen_
ucrtbased!common_openfile<char>+0x18a [min
ucrtbased!_openfile+0x3e [minkernel\crts\u
ucrtbased!__crt_char_traits<char>::open_fi
ucrtbased!common_fsopen<char>+0x282 [minke
ucrtbased!fopen+0x23 [minkernel\crts\ucrt\
UserMode!main+0x3e [E:\MyPrograms\CPP\User
UserMode!invoke_main+0x39 [d:\agent\work\
UserMode!__scrt_common_main_seh+0x12e [d:\
UserMode!__scrt_common_main+0xe [d:\agent\
UserMode!mainCRTStartup+0x9 [d:\agent\wor
KERNEL32!BaseThreadInitThunk+0x14
ntdll!RtlUserThreadStart+0x21
```

Youtube link: <https://youtu.be/RK8mRlf5bMg>




Scenario : Open a file in hard disk

```
Call Site
ntdll!NtCreateFile
KERNELBASE!CreateFileInternal+0x2f4
KERNELBASE!CreateFileW+0x66
ucrtbased!create_file+0x5f [minkernel\crt
ucrtbased!_wsopen_nolock+0x17a [minkernel\
ucrtbased!_sopen_nolock+0xad [minkernel\cr
ucrtbased!__crt_char_traits<char>::tsopen
ucrtbased!common_sopen_dispatch<char>+0x29
ucrtbased!_sopen_dispatch+0x41 [minkernel\
ucrtbased!_sopen_s+0x42 [minkernel\crt\uc
ucrtbased!__crt_char_traits<char>::tsopen
ucrtbased!common_openfile<char>+0x18a [min
ucrtbased!_openfile+0x3e [minkernel\crt\vu
ucrtbased!__crt_char_traits<char>::open_fi
ucrtbased!common_fsopen<char>+0x282 [minke
ucrtbased!fopen+0x23 [minkernel\crt\ucrt\
UserMode!main+0x3e [E:\MyPrograms\CPP\User
UserMode!invoke_main+0x39 [d:\agent\work\
UserMode!__scrt_common_main_seh+0x12e [d:\
UserMode!__scrt_common_main+0xe [d:\agent\
UserMode!mainCRTStartup+0x9 [d:\agent\wor
KERNEL32!BaseThreadInitThunk+0x14
ntdll!RtlUserThreadStart+0x21
```

Youtube link: <https://youtu.be/RK8mRlf5bMg>



Scenario : Open a file in hard disk



```
Call Site
ntdll!NtCreateFile
KERNELBASE!CreateFileInternal+0x2f4
KERNELBASE!CreateFileW+0x66
ucrtbased!create_file+0x5f [minkernel\crt
ucrtbased!_wsopen_nolock+0x17a [minkernel\
ucrtbased!_sopen_nolock+0xad [minkernel\cr
ucrtbased!__crt_char_traits<char>::tsopen_
ucrtbased!common_sopen_dispatch<char>+0x29
ucrtbased!_sopen_dispatch+0x41 [minkernel\
ucrtbased!_sopen_s+0x42 [minkernel\crt\uc
ucrtbased!__crt_char_traits<char>::tsopen_
ucrtbased!common_openfile<char>+0x18a [min
ucrtbased!_openfile+0x3e [minkernel\crt\vu
ucrtbased!__crt_char_traits<char>::open_fi
ucrtbased!common_fsopen<char>+0x282 [minke
ucrtbased!fopen+0x23 [minkernel\crt\ucrt\
UserMode!main+0x3e [E:\MyPrograms\CPP\User
UserMode!invoke_main+0x39 [d:\agent\work\
UserMode!__scrt_common_main_seh+0x12e [d:\
UserMode!__scrt_common_main+0xe [d:\agent\
UserMode!mainCRTStartup+0x9 [d:\agent\wor
KERNEL32!BaseThreadInitThunk+0x14
ntdll!RtlUserThreadStart+0x21
```

Youtube link: <https://youtu.be/RK8mRlf5bMg>



Scenario : Open a file in hard disk



```
Call Site  
ntdll!NtCreateFile  
KERNELBASE!CreateFileInternal+0x2f4  
KERNELBASE!CreateFileW+0x66  
ucrtbased!create_file+0x5f [minkernel\crt  
ucrtbased!_wsopen_nolock+0x17a [minkernel\  
ucrtbased!_sopen_nolock+0xad [minkernel\cr  
ucrtbased!__crt_char_traits<char>::tsopen_  
ucrtbased!common_sopen_dispatch<char>+0x29  
ucrtbased!_sopen_dispatch+0x41 [minkernel\  
ucrtbased!_sopen_s+0x42 [minkernel\crt\suc  
ucrtbased!__crt_char_traits<char>::tsopen_  
ucrtbased!common_openfile<char>+0x18a [min  
ucrtbased!_openfile+0x3e [minkernel\crt\suc  
ucrtbased!__crt_char_traits<char>::open_fi  
ucrtbased!common_fsopen<char>+0x282 [minke  
ucrtbased!fopen+0x23 [minkernel\crt\ucrt\  
UserMode!main+0x3e [E:\MyPrograms\CPP\User  
UserMode!invoke_main+0x39 [d:\agent\work\  
UserMode!__scrt_common_main_seh+0x12e [d:\  
UserMode!__scrt_common_main+0xe [d:\agent\  
UserMode!mainCRTStartup+0x9 [d:\agent\wor  
KERNEL32!BaseThreadInitThunk+0x14  
ntdll!RtlUserThreadStart+0x21
```

Youtube link: <https://youtu.be/RK8mRlf5bMg>



Scenario : Open a file in hard disk

→ Call Site
ntdll!NtCreateFile
KERNELBASE!CreateFileInternal+0x2f4
KERNELBASE!CreateFileW+0x66
ucrtbased!create_file+0x5f [minkernel\crt
ucrtbased!_wsopen_nolock+0x17a [minkernel\
ucrtbased!_sopen_nolock+0xad [minkernel\cr
ucrtbased!__crt_char_traits<char>::tsopen_
ucrtbased!common_sopen_dispatch<char>+0x29
ucrtbased!_sopen_dispatch+0x41 [minkernel\
ucrtbased!_sopen_s+0x42 [minkernel\crt\uc
ucrtbased!__crt_char_traits<char>::tsopen_
ucrtbased!common_openfile<char>+0x18a [min
ucrtbased!_openfile+0x3e [minkernel\crt\vu
ucrtbased!__crt_char_traits<char>::open_fi
ucrtbased!common_fsopen<char>+0x282 [minke
ucrtbased!fopen+0x23 [minkernel\crt\ucrt\
UserMode!main+0x3e [E:\MyPrograms\CPP\User
UserMode!invoke_main+0x39 [d:\agent\work\
UserMode!__scrt_common_main_seh+0x12e [d:\
UserMode!__scrt_common_main+0xe [d:\agent\
UserMode!mainCRTStartup+0x9 [d:\agent\wor
KERNEL32!BaseThreadInitThunk+0x14
ntdll!RtlUserThreadStart+0x21

Youtube link: <https://youtu.be/RK8mRlf5bMg>



Scenario : Open a file in hard disk

```
0:000> u
ntdll!NtCreateFile:
00007ffc`4417cb00 4c8bd1      mov     r10,rcx
00007ffc`4417cb03 b855000000    mov     eax,55h
00007ffc`4417cb08 f604250803fe7f01 test    byte ptr [SharedUserData+0x308
00007ffc`4417cb10 7503         jne     ntdll!NtCreateFile+0x15 (00007f
00007ffc`4417cb12 0f05         syscall
00007ffc`4417cb14 c3           ret
00007ffc`4417cb15 cd2e         int     2Eh
00007ffc`4417cb17 c3           ret
```


CPU register eax is set as 55h

Youtube link: <https://youtu.be/RK8mRlf5bMg>



Scenario : Open a file in hard disk

```
0:000> u
ntdll!NtCreateFile:
00007ffc`4417cb00 4c8bd1      mov     r10,rcx
00007ffc`4417cb03 b855000000    mov     eax,55h
00007ffc`4417cb08 f604250803fe7f01 test    byte ptr [SharedUserData+0x308
00007ffc`4417cb10 7503         jne     ntdll!NtCreateFile+0x15 (00007f
00007ffc`4417cb12 0f05         syscall
00007ffc`4417cb14 c3           ret
00007ffc`4417cb15 cd2e         int     2Eh
00007ffc`4417cb17 c3           ret
```



Parameters are passed as stack frames

Youtube link: <https://youtu.be/RK8mRlf5bMg>



Scenario : Open a file in hard disk

```
0:000> u
ntdll!NtCreateFile:
00007ffc`4417cb00 4c8bd1      mov     r10,rcx
00007ffc`4417cb03 b855000000    mov     eax,55h
00007ffc`4417cb08 f604250803fe7f01 test    byte ptr [SharedUserData+0x308
00007ffc`4417cb10 7503         jne     ntdll!NtCreateFile+0x15 (00007f
00007ffc`4417cb12 0f05         syscall ←
00007ffc`4417cb14 c3           ret
00007ffc`4417cb15 cd2e         int     2Eh
00007ffc`4417cb17 c3           ret
```

Enters into kernel mode using 'syscall'

Youtube link: <https://youtu.be/RK8mRI5bMg>



Scenario : Open a file in hard disk

```
0:000> u
ntdll!NtCreateFile:
00007ffc`4417cb00 4c8bd1      mov     r10,rcx
00007ffc`4417cb03 b855000000    mov     eax,55h
00007ffc`4417cb08 f604250803fe7f01 test    byte ptr [SharedUserData+0x308
00007ffc`4417cb10 7503         jne     ntdll!NtCreateFile+0x15 (00007f
00007ffc`4417cb12 0f05         syscall ←
00007ffc`4417cb14 c3           ret
00007ffc`4417cb15 cd2e         int     2Eh
00007ffc`4417cb17 c3           ret
```

Enters into kernel mode using 'syscall'

Youtube link: <https://youtu.be/RK8mRI5bMg>

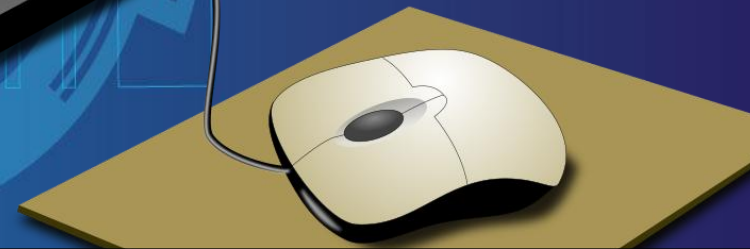


Kernel Mode



TRAP/SYSCALL

Youtube link: <https://youtu.be/RK8mRlf5bMg>



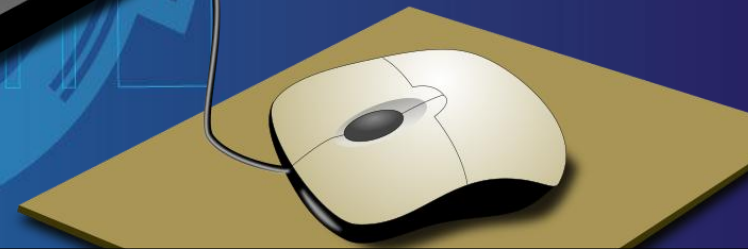
Kernel Mode

TRAP/SYSCALL

NtCreateFile

ntoskrnl.exe

Youtube link: <https://youtu.be/RK8mRlf5bMg>



Kernel Mode

TRAP/SYSCALL
↓

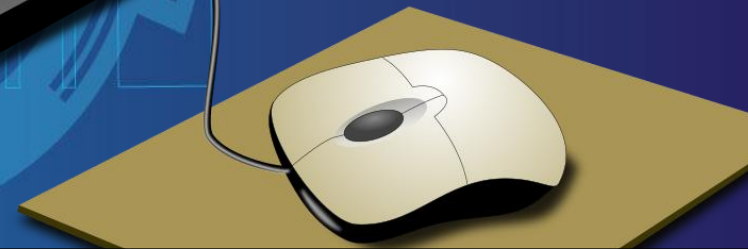
NtCreateFile

ntoskrnl.exe

HAL.dll

hal.dll

Youtube link: <https://youtu.be/RK8mRlf5bMg>



Kernel Mode

TRAP/SYSCALL

NtCreateFile

ntoskrnl.exe

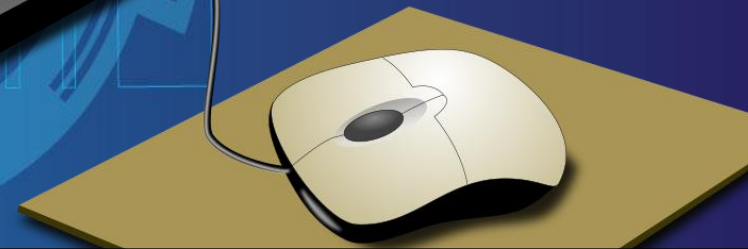
HAL.dll

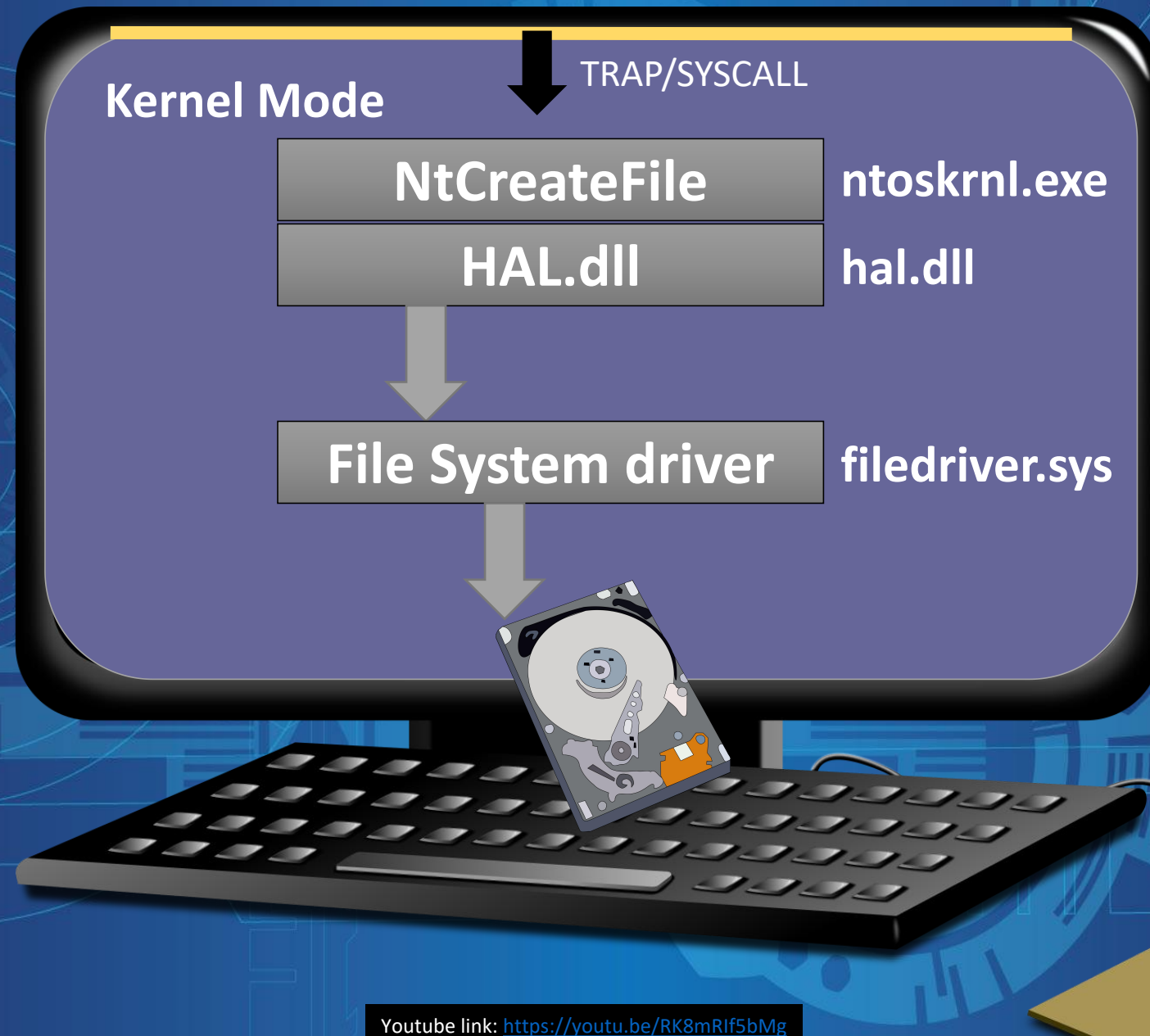
hal.dll

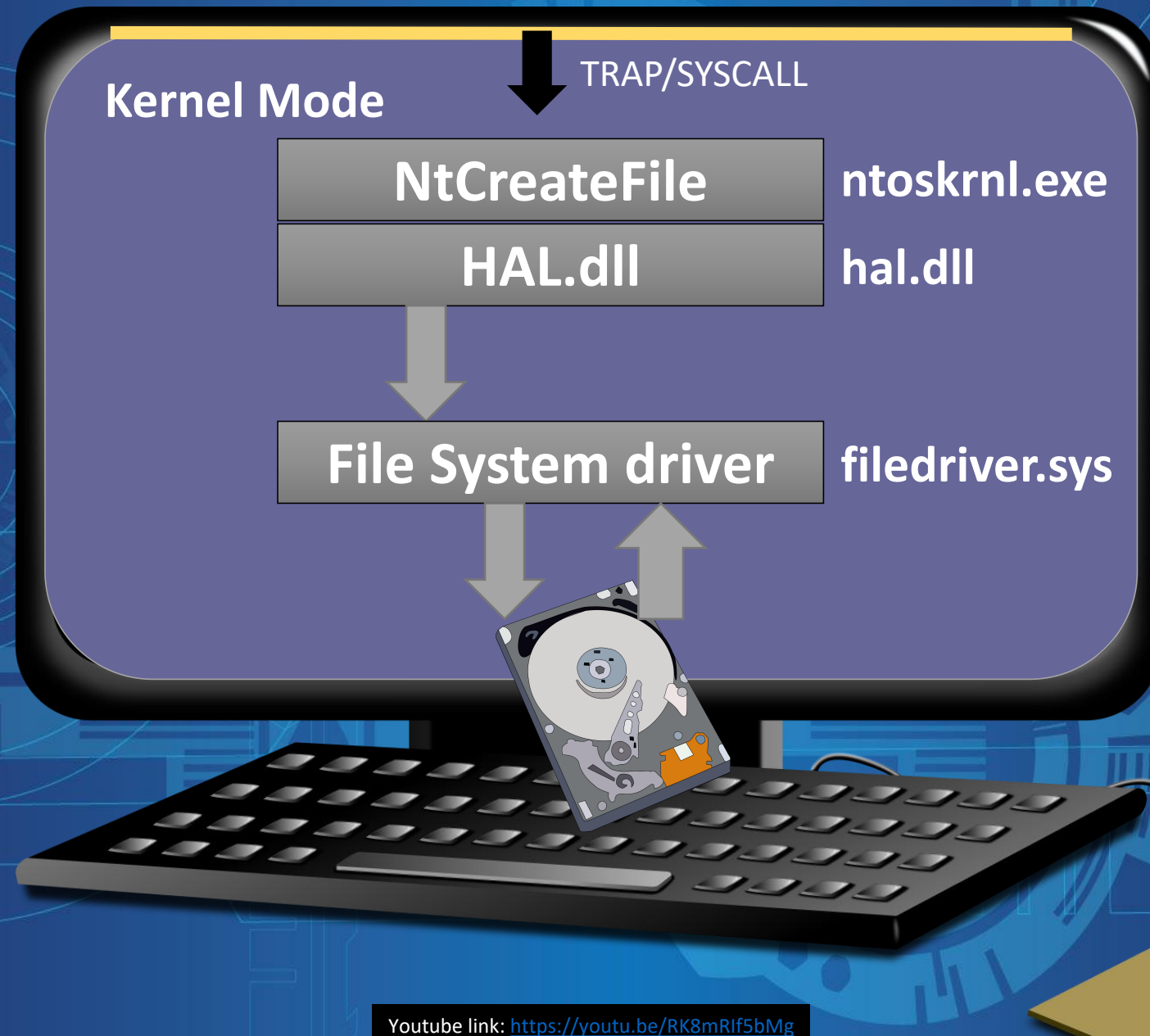
File System driver

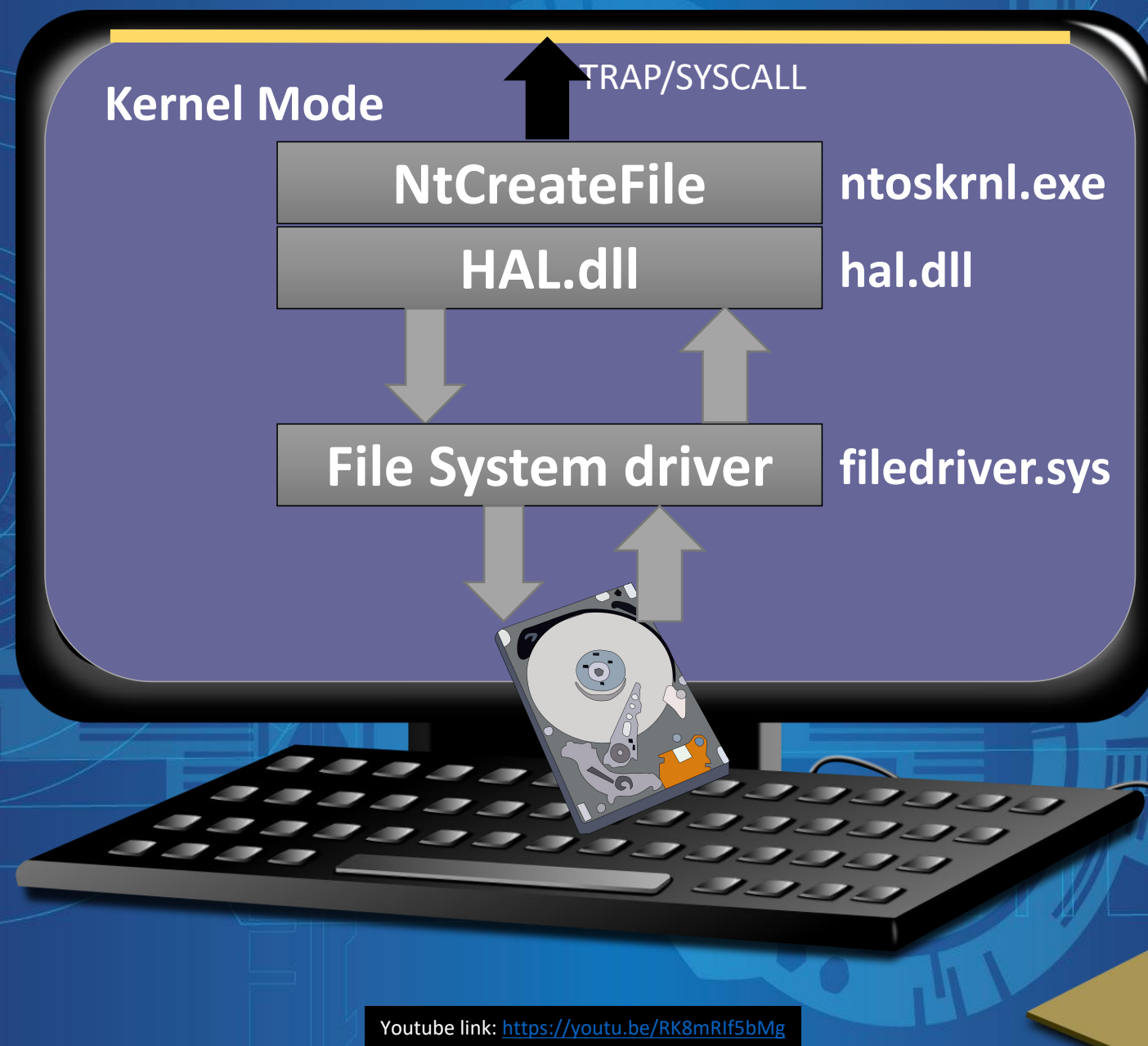
filedriver.sys

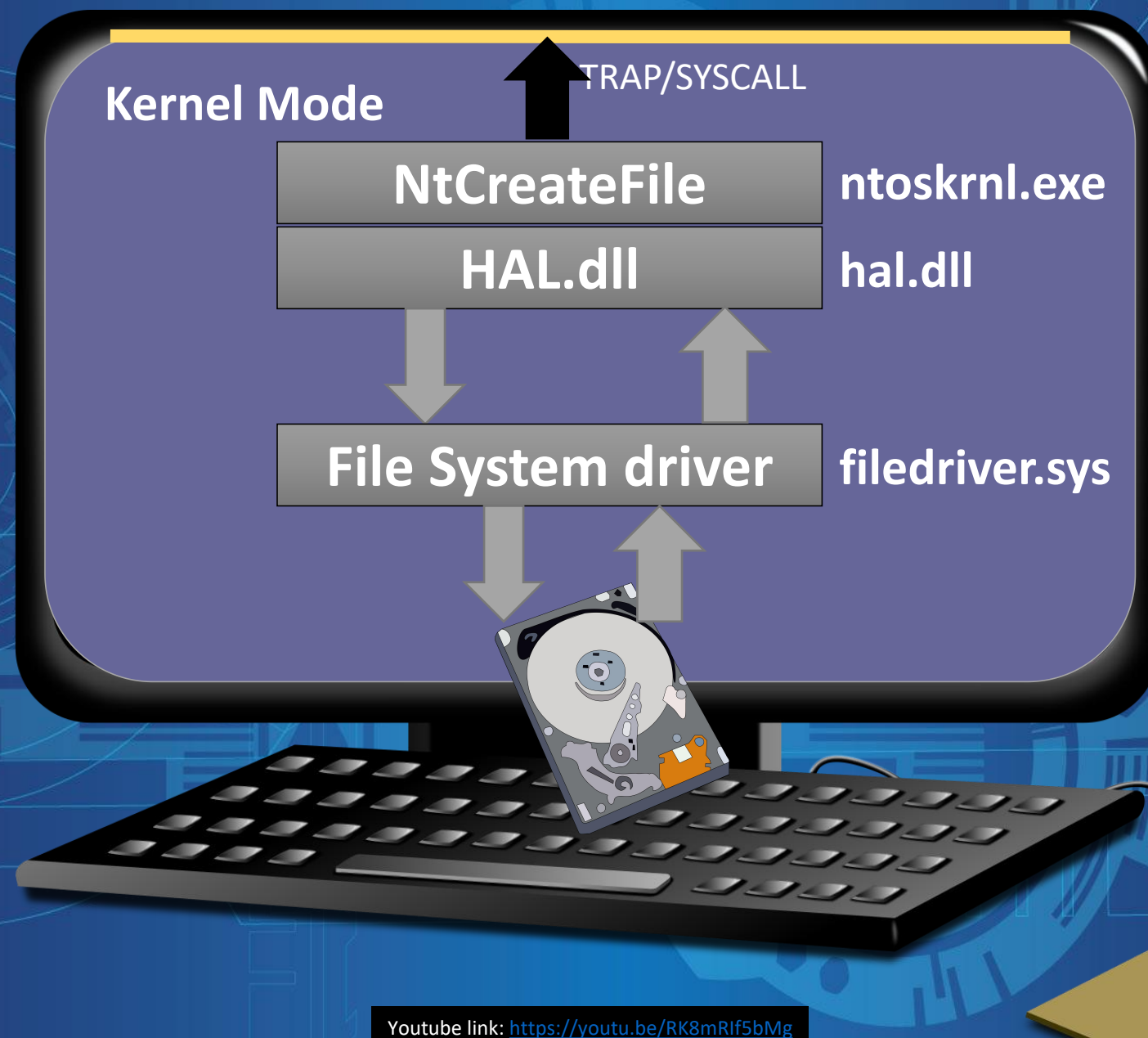
Youtube link: <https://youtu.be/RK8mRlf5bMg>











Kernel Mode

Dependency Walker - [ntoskrnl.exe]

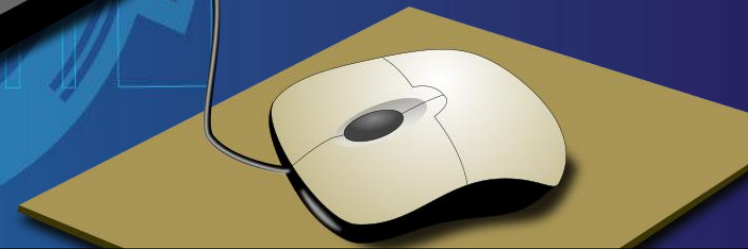
File Edit View Options Profile Window Help

NTOSKRNL.EXE

- HAL.DLL
- PSHED.DLL
- BOOTVID.DLL
- KDCOM.DLL
- CI.DLL

PI	Ordinal ^	Hint	Function
E	Ordinal ^	Hint	Function
C	1411 (0x0583)	1402 (0x057A)	NtCreateEnlistment
C	1412 (0x0584)	1403 (0x057B)	NtCreateEvent
C	1413 (0x0585)	1404 (0x057C)	NtCreateFile
C	1414 (0x0586)	1405 (0x057D)	NtCreateResourceManager

Youtube link: <https://youtu.be/RK8mRlf5bMg>



Scenario : Open a file in hard disk

fopen

ucrtbase.dll

CreateFileW

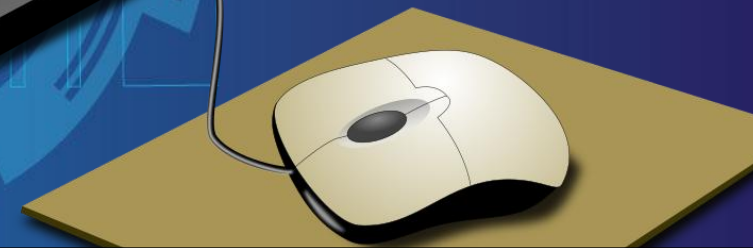
KERNELBASE.dll

NtCreateFile

ntdll.dll

↑
TRAP/SYSCALL

Youtube link: <https://youtu.be/RK8mRlf5bMg>



Scenario : Open a file in hard disk

fopen

ucrtbase.dll

CreateFileW

KERNELBASE.dll

NtCreateFile

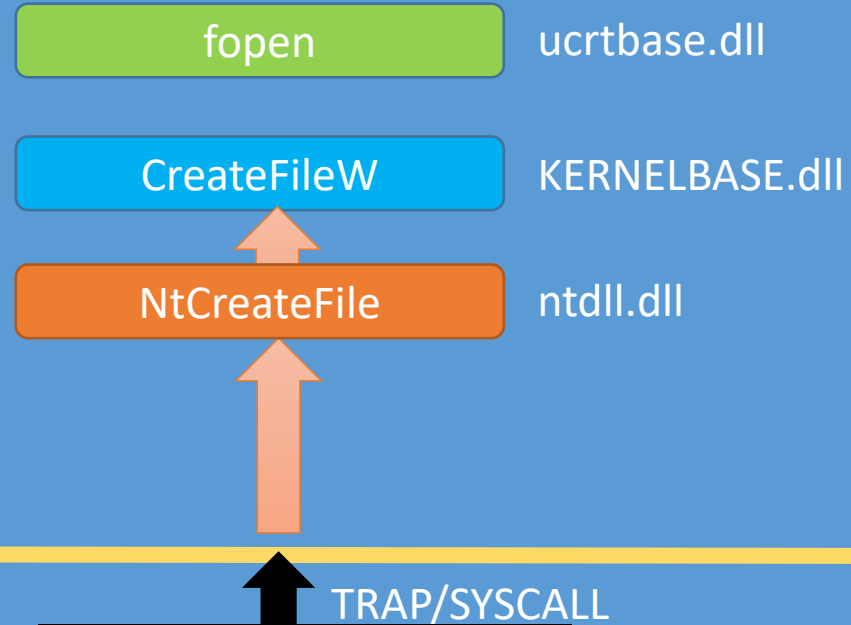
ntdll.dll

TRAP/SYSCALL

Youtube link: <https://youtu.be/RK8mRlf5bMg>

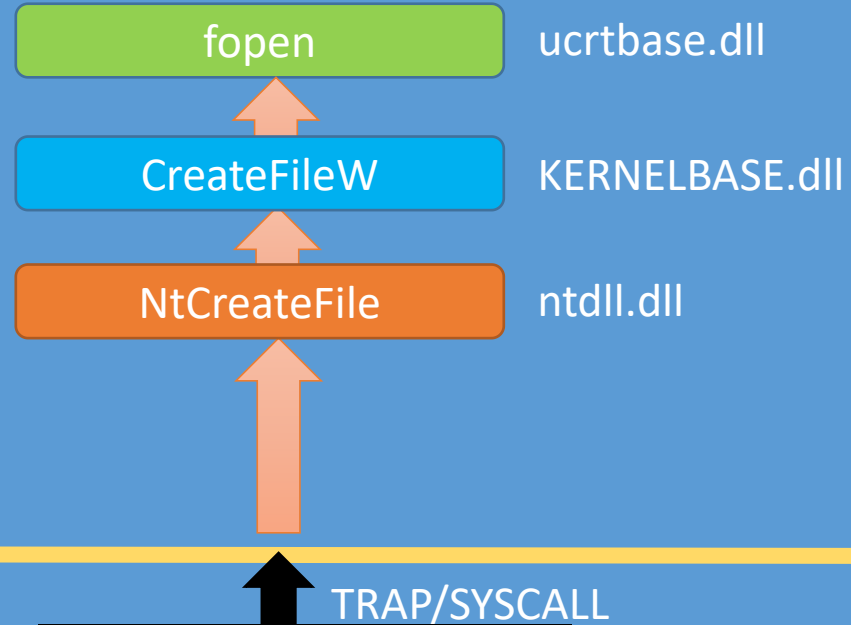


Scenario : Open a file in hard disk



Youtube link: <https://youtu.be/RK8mRlf5bMg>

Scenario : Open a file in hard disk



Youtube link: <https://youtu.be/RK8mRlf5bMg>



Thank you



Youtube link: <https://youtu.be/RK8mRlf5bMg>

