**Part 1. The Attack**

1. Start three apps:

* **mem\_allocator\_console\_ru.exe**
* **mem\_allocator\_console\_uk.exe**
* **mem\_attacker\_console.exe**

1. MemAttacker reads first two bytes of PE-header of MemAllocatorRU MemAllocatorUK drivers:

|  |  |
| --- | --- |
| **mem\_attacker\_console.exe** | read\_data <RU\_BASE\_ADDR>  read\_data <RU\_BASE\_ADDR+1> |

Copy base address of MemAllocatorUK driver and read its content:

|  |  |
| --- | --- |
| **mem\_attacker\_console.exe** | read\_data <UK\_BASE\_ADDR>  read\_data <UK\_BASE\_ADDR+1> |

1. Allocate memory and show that it is accessible:

|  |  |
| --- | --- |
| **mem\_allocator\_console\_ru.exe** | alloc Wassily Kandinsky  read\_data <RU\_ALLOC\_ADDR> |
| **mem\_allocator\_console\_uk.exe** | alloc Banksy  read\_data <UK\_ALLOC\_ADDR> |

1. Start **mem\_attacker\_console.exe** and read\write each of allocated memory:

|  |  |
| --- | --- |
| **mem\_attacker\_console.exe** | read\_data <RU\_ALLOC\_ADDR>  write\_data <RU\_ALLOC\_ADDR> 1111 |
| **mem\_allocator\_console\_ru.exe** | read\_data <RU\_ALLOC\_ADDR> |

And finally:

|  |  |
| --- | --- |
| **mem\_attacker\_console.exe** | read\_data <UK\_ALLOC\_ADDR>  write\_data <UK\_ALLOC\_ADDR> 2222 |
| **mem\_allocator\_console\_uk.exe** | read\_data <UK\_ALLOC\_ADDR> |

1. Start **cmd.exe** and get its PID:

|  |  |
| --- | --- |
| **cmd.exe** | tasklist | findstr cmd |

Now we have its PID. Copy PID. Run the following:

|  |  |
| --- | --- |
| **cmd.exe** | whoami  sc config MpsSvc start= disabled |

We can see the **cmd.exe** does not have enough privileges. Let’s elevate it:

|  |  |
| --- | --- |
| **mem\_attacker\_console.exe** | priv <PID> |

Let’s back to the **cmd.exe** and run the following:

|  |  |
| --- | --- |
| **cmd.exe** | whoami  sc config MpsSvc start= disabled  sc stop MpsSvc |

Wait 10 hours and conclude that memory has been corrupted without BSOD.

**Part 2. The Attack Prevention**

1. Start **memranger\_console.exe**
2. Start four apps:

* **mem\_allocator\_console\_ru.exe**
* **mem\_allocator\_console\_uk.exe**
* **mem\_allocator\_console\_us.exe**
* **mem\_attacker\_console.exe**

1. Read first two bytes of PE-header of each driver:

MemAttacker reads first two bytes of the MemAllocatorRU:

|  |  |
| --- | --- |
| **mem\_attacker\_console.exe** | read\_data <RU\_BASE\_ADDR>  read\_data <RU\_BASE\_ADDR+1> |

I have zero value.

MemAllocatorRU reads its memory:

|  |  |
| --- | --- |
| **mem\_allocator\_console\_ru.exe** | read\_data <RU\_BASE\_ADDR>  read\_data <RU\_BASE\_ADDR+1> |

I have MZ-value.

1. MemAttacker reads first two bytes of the MemAllocatorUK:

|  |  |
| --- | --- |
| **mem\_attacker\_console.exe** | read\_data <UK\_BASE\_ADDR>  read\_data <UK\_BASE\_ADDR+1> |

I have zero value.

MemAllocatorUK reads its memory:

|  |  |
| --- | --- |
| **mem\_allocator\_console\_uk.exe** | read\_data <UK\_BASE\_ADDR>  read\_data <UK\_BASE\_ADDR+1> |

I have MZ-value.

1. MemAllocatorUS reads the MemAllocatorUK base address:

|  |  |
| --- | --- |
| **mem\_allocator\_console\_us.exe** | read\_data <UK\_BASE\_ADDR>  read\_data <UK\_BASE\_ADDR+1> |

I have zero value. We can see that MemAllocatorUS driver also reads zero instead of MZ-value.

🡺Step I. The drivers’ files have been isolated.

1. Allocate memory buffers and show that they are accessible:

|  |  |
| --- | --- |
| **mem\_allocator\_console\_ru.exe** | alloc Wassily Kandinsky  read\_data <RU\_ALLOC\_ADDR> |
| **mem\_allocator\_console\_uk.exe** | alloc Banksy  read\_data <UK\_ALLOC\_ADDR> |
| **mem\_allocator\_console\_us.exe** | alloc Jackson Pollock  read\_data <US\_ALLOC\_ADDR> |

1. Read/write allocated memory using **mem\_attacker\_console.exe**:

|  |  |
| --- | --- |
| **mem\_attacker\_console.exe** | read\_data <RU\_ALLOC\_ADDR>  write\_data <RU\_ALLOC\_ADDR> 1111 |
| **mem\_allocator\_console\_ru.exe** | read\_data <RU\_ALLOC\_ADDR> |

We can see that the data is protected. Let’s check next:

|  |  |
| --- | --- |
| **mem\_attacker\_console.exe** | read\_data <UK\_ALLOC\_ADDR>  write\_data <UK\_ALLOC\_ADDR> 2222 |
| **mem\_allocator\_console\_uk.exe** | read\_data <UK\_ALLOC\_ADDR> |

Finally, MemAllocatorUS reads MemAllocatorUK allocation:

|  |  |
| --- | --- |
| **mem\_allocator\_console\_us.exe** | read\_data < UK\_ALLOC\_ADDR >  write\_data < UK\_ALLOC\_ADDR > 3333 |
| **mem\_allocator\_console\_us.exe** | read\_data <UK\_ALLOC\_ADDR> |

The allocated data cannot be accessible by any other driver.

Let’s show that the data can be changed only by owner of this data:

|  |  |
| --- | --- |
| **mem\_allocator\_ console\_ru.exe** | read\_data <RU\_ALLOC\_ADDR>  write\_data <RU\_ALLOC\_ADDR> Ivan Aivazovsky  read\_data <RU\_ALLOC\_ADDR> |

🡺Step II. The integrity and confidentially of allocated data has been protected.

1. Start **cmd.exe** and get its PID:

|  |  |
| --- | --- |
| **cmd.exe** | tasklist | findstr cmd |

Now we have its PID. Copy PID. Run the following:

|  |  |
| --- | --- |
| **cmd.exe** | whoami  sc config MpsSvc start= disabled |

We can see that the **cmd.exe** does not have enough privileges. Let’s elevate it.

1. Run the following from **mem\_attacker\_console.exe**

|  |  |
| --- | --- |
| **mem\_attacker\_console.exe** | priv <PID > |

1. Let’s back to the **cmd.exe** and run the following:

|  |  |
| --- | --- |
| **cmd.exe** | sc config MpsSvc start= disabled  sc stop MpsSvc |

**Cmd.exe** doesn’t have enough privileges.

🡺Step III. The OS internal EPROCESS structures have been protected.

Summary: MemoryRanger runs drivers in isolated memory enclosures. The OS memory is protected