

MALWARE ANALYSIS: OLLYDBG

S11 - L3

| | | | |
|----------|-----------------|--|---|
| 00401056 | . 52 | PUSH EDX | pProcessInfo pStartupInfo CurrentDir = NULL pEnvironment = NULL CreationFlags = 0 InheritHandles = TRUE pThreadSecurity = NULL pProcessSecurity = NULL CommandLine = "cmd" ModuleFileName = NULL CreateProcessA |
| 00401057 | . 8D45 A8 | LEA EAX,DWORD PTR SS:[EBP-58] | |
| 0040105A | . 50 | PUSH EAX | |
| 0040105B | . 6A 00 | PUSH 0 | |
| 0040105D | . 6A 00 | PUSH 0 | |
| 0040105F | . 6A 00 | PUSH 0 | |
| 00401061 | . 6A 01 | PUSH 1 | |
| 00401063 | . 6A 00 | PUSH 0 | |
| 00401065 | . 6A 00 | PUSH 0 | |
| 00401067 | . 68 30504000 | PUSH Malware_.00405030 | |
| 0040106C | . 6A 00 | PUSH 0 | |
| 0040106E | . FF15 04404000 | CALL DWORD PTR DS:[<&KERNEL32.CreateProcessA>] | |

All'indirizzo **0040106E** troviamo la funzione **CreateProcessA**. Nelle righe precedenti possiamo vedere gli argomenti preparati prima di iniziare questo processo, uno di questi è il parametro **CommandLine**, con il valore di **cmd**.

| | |
|----------|---------------------------------|
| 004015A3 | CALL DWORD PTR DS:[4052D4], EDX |
| 004015A4 | XOR EDX, EDX |
| 004015A5 | MOV DL, AH |
| 004015A7 | MOV DWORD PTR DS:[4052D4], EDX |
| 004015AD | MOV ECX, EAX |
| 004015AF | AND ECX, 0FF |

| Registers (FPU) | |
|-----------------|---|
| EAX | 0A280105 |
| ECX | 2EEDB000 |
| EDX | 00000A28 |
| EBX | 7FFDB000 |
| ESP | 0012FF94 |
| EBP | 0012FFC0 |
| ESI | FFFFFFFF |
| EDI | 7C910208 ntdll.7C910208 |
| EIP | 004015A3 Malware_.004015A3 |
| C 0 | ES 0023 32bit 0(FFFFFFFF) |
| P 1 | CS 001B 32bit 0(FFFFFFFF) |
| A 0 | SS 0023 32bit 0(FFFFFFFF) |
| Z 0 | DS 0023 32bit 0(FFFFFFFF) |
| S 0 | FS 003B 32bit 7FDF000(FFF) |
| T 0 | GS 0000 NULL |
| D 0 | |
| O 0 | LastErr ERROR_INVALID_HANDLE (00000006) |
| EFL | 00000206 (NO, NB, NE, A, NS, PE, GE, G) |
| ST0 | empty -UNORM BCBC 01050104 005C0030 |
| ST1 | empty +UNORM 0069 006E0069 002E0067 |
| ST2 | empty 0.0 |
| ST3 | empty 0.0 |
| ST4 | empty 0.0 |
| ST5 | empty 0.0 |
| ST6 | empty 0.0 |
| ST7 | empty 0.0 |
| FST 0000 | Cond 0 0 0 0 Err 0 0 0 0 0 0 0 0 (GT) |
| FCW 027F | Prec NEAR, 53 Mask 1 1 1 1 1 1 |



| Registers (FPU) | |
|-----------------|--|
| EAX | 0A280105 |
| ECX | 2EEDB000 |
| EDX | 00000000 |
| EBX | 7FFDB000 |
| ESP | 0012FF94 |
| EBP | 0012FFC0 |
| ESI | FFFFFFFF |
| EDI | 7C910208 ntdll.7C910208 |
| EIP | 004015A5 Malware_.004015A5 |
| C 0 | ES 0023 32bit 0(FFFFFFFF) |
| P 1 | CS 001B 32bit 0(FFFFFFFF) |
| A 0 | SS 0023 32bit 0(FFFFFFFF) |
| Z 1 | DS 0023 32bit 0(FFFFFFFF) |
| S 0 | FS 003B 32bit 7FDF000(FFF) |
| T 0 | GS 0000 NULL |
| D 0 | |
| O 0 | LastErr ERROR_INVALID_HANDLE (00000006) |
| EFL | 00000246 (NO, NB, E, BE, NS, PE, GE, LE) |
| ST0 | empty -UNORM BCBC 01050104 005C0030 |
| ST1 | empty +UNORM 0069 006E0069 002E0067 |
| ST2 | empty 0.0 |
| ST3 | empty 0.0 |
| ST4 | empty 0.0 |
| ST5 | empty 0.0 |
| ST6 | empty 0.0 |
| ST7 | empty 0.0 |
| FST 0000 | Cond 0 0 0 0 Err 0 0 0 0 0 0 0 0 (GT) |
| FCW 027F | Prec NEAR, 53 Mask 1 1 1 1 1 1 |

All'indirizzo **004015A3** si trova l'istruzione **XOR EDX, EDX**. XOR è l'istruzione e EDX è l'operando. Quando un'istruzione XOR viene eseguita da un operando a se stesso, l'operando viene "cancellato", ovvero in questo caso il registro **EDX** verrà cancellato e riportato a 0.

| | | |
|----------|-----------------|--------------------------------|
| 0040159D | . 33D2 | CALL DWORD PTR DS:[4052D4],EDX |
| 004015A3 | . 8AD4 | XOR EDX,EDX |
| 004015A5 | . 8915 | MOV DL,AH |
| 004015A7 | . 8BC8 | MOV DWORD PTR DS:[4052D4],EDX |
| 004015A9 | . 8BC8 | MOV ECX,ECX |
| 004015AF | . 81E1 FF000000 | AND ECX,0FF |

| Registers (FPU) | |
|-----------------|---|
| EAX | 00000000 |
| ECX | 0012FFB0 |
| EDX | 7C910208 ntdll.7C910208 |
| EBX | 7FFDE000 |
| ESP | 0012FFC4 |
| EBP | 0012FFC0 |
| ESI | FFFFFFFF |
| EDI | 7C910208 ntdll.7C910208 |
| EIP | 00401577 Malware_.<ModuleEntryPoint> |
| C 0 | ES 0023 32bit 0(FFFFFFFF) |
| P 1 | CS 001B 32bit 0(FFFFFFFF) |
| A 0 | SS 0023 32bit 0(FFFFFFFF) |
| Z 1 | DS 0023 32bit 0(FFFFFFFF) |
| S 0 | FS 003B 32bit 7FFDD000(FFF) |
| T 0 | GS 0000 NULL |
| D 0 | |
| O 0 | LastErr ERROR_INVALID_HANDLE (00000006) |
| EFL | 00000246 (NO,NB,E,BE,NS,PE,GE,LE) |
| ST0 | empty -UNORM BCBC 01050104 005C0030 |
| ST1 | empty +UNORM 0069 006E0069 002E0067 |
| ST2 | empty 0.0 |
| ST3 | empty 0.0 |
| ST4 | empty 0.0 |
| ST5 | empty 0.0 |
| ST6 | empty 0.0 |
| ST7 | empty 0.0 |
| FST 0000 | Cond 0 0 0 0 Err 0 0 0 0 0 0 0 0 (GT) |
| FCW 027F | Prec NEAR,53 Mask 1 1 1 1 1 1 |



| Registers (FPU) | |
|-----------------|---|
| EAX | 0A280105 |
| ECX | 00000005 |
| EDX | 00000000 |
| EBX | 7FFDE000 |
| ESP | 0012FF94 |
| EBP | 0012FFC0 |
| ESI | FFFFFFFF |
| EDI | 7C910208 ntdll.7C910208 |
| EIP | 004015B5 Malware_.004015B5 |
| C 0 | ES 0023 32bit 0(FFFFFFFF) |
| P 1 | CS 001B 32bit 0(FFFFFFFF) |
| A 0 | SS 0023 32bit 0(FFFFFFFF) |
| Z 0 | DS 0023 32bit 0(FFFFFFFF) |
| S 0 | FS 003B 32bit 7FFDD000(FFF) |
| T 0 | GS 0000 NULL |
| D 0 | |
| O 0 | LastErr ERROR_INVALID_HANDLE (00000006) |
| EFL | 00000206 (NO,NB,NE,A,NS,PE,GE,G) |
| ST0 | empty -UNORM BCBC 01050104 005C0030 |
| ST1 | empty +UNORM 0069 006E0069 002E0067 |
| ST2 | empty 0.0 |
| ST3 | empty 0.0 |
| ST4 | empty 0.0 |
| ST5 | empty 0.0 |
| ST6 | empty 0.0 |
| ST7 | empty 0.0 |
| FST 0000 | Cond 0 0 0 0 Err 0 0 0 0 0 0 0 0 (GT) |
| FCW 027F | Prec NEAR,53 Mask 1 1 1 1 1 1 |

All'indirizzo **004015F** c'è l'istruzione **AND EXC, 0FF**. In genere l'istruzione **AND** viene utilizzata per produrre un flag 0 o 1 a seconda che entrambi gli argomenti siano uguali, ma ha anche la capacità di cancellare i bit di un registro. Nel caso di **AND EXC, 0FF**: **0FF** cancella i primi bit del registro **EXC**, facendolo cambiare da **0012FFB0** a **00000005**.