DBatLoader

malpedia.caad.fkie.fraunhofer.de/details/win.dbatloader

This Delphi loader misuses Cloud storage services, such as Google Drive to download the Delphi stager component. The Delphi stager has the actual payload embedded as a resource and starts it.

[TLP:WHITE] win_dbatloader_auto (20201014 | autogenerated rule brought to you by yara-signator)

```
rule win_dbatloader_auto {

    meta:
        author = "Felix Bilstein - yara-signator at cocacoding dot com"
        date = "2020-10-14"
        version = "1"
        description = "autogenerated rule brought to you by yara-signator"
        tool = "yara-signator v0.5.0"
        signator_config = "callsandjumps;datarefs;binvalue"
        malpedia_reference =
"https://malpedia.caad.fkie.fraunhofer.de/details/win.dbatloader"
        malpedia_rule_date = "20201014"
        malpedia_hash = "a7e3bd57eaf12bf3ea29a863c041091ba3af9ac9"
        malpedia_version = "20201014"
        malpedia_license = "CC BY-SA 4.0"
        malpedia_sharing = "TLP:WHITE"

    /* DISCLAIMER
     * The strings used in this rule have been automatically selected from the
     * disassembly of memory dumps and unpacked files, using YARA-Signator.
     * The code and documentation is published here:
     * https://github.com/fxb-cocacoding/yara-signator
     * As Malpedia is used as data source, please note that for a given
     * number of families, only single samples are documented.
     * This likely impacts the degree of generalization these rules will offer.
     * Take the described generation method also into consideration when you
     * apply the rules in your use cases and assign them confidence levels.
     */


    strings:
        $sequence_0 = { 8b4504 03c5 8be8 8bc5 2b442414 }
            // n = 5, score = 600
            //   8b4504               | mov                 eax, dword ptr [ebp +
4]
            //   03c5                 | add                 eax, ebp
            //   8be8                 | mov                 ebp, eax
            //   8bc5                 | mov                 eax, ebp
            //   2b442414             | sub                 eax, dword ptr [esp +
0x14]

        $sequence_1 = { 6681e1ff0f 0fb7c9 03c1 0110 }
            // n = 4, score = 600
            //   6681e1ff0f           | and                 cx, 0xfff
            //   0fb7c9               | movzx               ecx, cx
            //   03c1                 | add                 eax, ecx
            //   0110                 | add                 dword ptr [eax], edx

        $sequence_2 = { 8b542404 8b5228 8bc6 03d0 8915???????? 6a00 6a01 }
            // n = 7, score = 600
            //   8b542404             | mov                 edx, dword ptr [esp +
4]
            //   8b5228               | mov                 edx, dword ptr [edx +
0x28]
            //   8bc6                 | mov                 eax, esi
            //   03d0                 | add                 edx, eax
            //   8915????????         |
            //   6a00                 | push                0
            //   6a01                 | push                1

        $sequence_3 = { e8???????? 89442418 8bdd 83c308 8b7c2418 4f }
            // n = 6, score = 600
            //   e8????????           |
            //   89442418             | mov                  dword ptr [esp +
0x18], eax
            //   8bdd                 | mov                  ebx, ebp
            //   83c308               | add                  ebx, 8
            //   8b7c2418             | mov                  edi, dword ptr [esp +
0x18]
            //   4f                   | dec                  edi

        $sequence_4 = { e8???????? 50 8b4308 50 8b430c 03c6 50 }
            // n = 7, score = 600
            //   e8????????           |
            //   50                   | push                 eax
            //   8b4308               | mov                  eax, dword ptr [ebx +
8]
            //   50                   | push                 eax
            //   8b430c               | mov                  eax, dword ptr [ebx +
0xc]
            //   03c6                 | add                  eax, esi
            //   50                   | push                 eax

        $sequence_5 = { 89442420 df6c241c d835???????? e8???????? 89442418 }
            // n = 5, score = 600
            //   89442420             | mov                  dword ptr [esp +
0x20], eax
            //   df6c241c             | fild                 qword ptr [esp +
0x1c]
            //   d835????????         |
            //   e8????????           |
            //   89442418             | mov                  dword ptr [esp +
0x18], eax

        $sequence_6 = { 8b4500 03c6 0fb70b 6681e1ff0f 0fb7c9 }
            // n = 5, score = 600
            //   8b4500               | mov                  eax, dword ptr [ebp]
            //   03c6                 | add                  eax, esi
            //   0fb70b               | movzx                ecx, word ptr [ebx]
            //   6681e1ff0f           | and                  cx, 0xfff
            //   0fb7c9               | movzx                ecx, cx

        $sequence_7 = { 8b5c2404 81c3f8000000 8b442404 0fb77806 }
            // n = 4, score = 600
            //   8b5c2404             | mov                  ebx, dword ptr [esp +
4]
            //   81c3f8000000         | add                  ebx, 0xf8
            //   8b442404             | mov                  eax, dword ptr [esp +
4]
            //   0fb77806             | movzx                edi, word ptr [eax +
6]

        $sequence_8 = { 8944240c 8b44240c 894834 8b44240c 05f8000000 }
            // n = 5, score = 200
            //   8944240c             | mov                  dword ptr [esp +
0xc], eax
            //   8b44240c             | mov                  eax, dword ptr [esp +
0xc]
            //   894834               | mov                  dword ptr [eax +
0x34], ecx
            //   8b44240c             | mov                  eax, dword ptr [esp +
0xc]
            //   05f8000000           | add                  eax, 0xf8
        $sequence_9 = { 50 8b442410 8b4034 50 e8???????? }
            // n = 5, score = 200
            //   50                   | push                eax
            //   8b442410             | mov                 eax, dword ptr [esp +
0x10]
            //   8b4034               | mov                 eax, dword ptr [eax +
0x34]
            //   50                   | push                eax
            //   e8????????           |

        $sequence_10 = { e9???????? 90 90 90 90 90 90 }
            // n = 7, score = 200
            //   e9????????           |
            //   90                   | nop
            //   90                   | nop
            //   90                   | nop
            //   90                   | nop
            //   90                   | nop
            //   90                   | nop

        $sequence_11 = { 90 90 81ca80000000 eb52 90 }
            // n = 5, score = 200
            //   90                   | nop
            //   90                   | nop
            //   81ca80000000         | or                  edx, 0x80
            //   eb52                 | jmp                 0x54
            //   90                   | nop

        $sequence_12 = { 90 90 90 83ca10 8bc2 c3 }
            // n = 6, score = 200
            //   90                   | nop
            //   90                   | nop
            //   90                   | nop
            //   83ca10               | or                  edx, 0x10
            //   8bc2                 | mov                 eax, edx
            //   c3                   | ret

        $sequence_13 = { 33c0 89442440 df6c243c d835???????? e8???????? }
            // n = 5, score = 200
            //   33c0                 | xor                 eax, eax
            //   89442440             | mov                 dword ptr [esp +
0x40], eax
            //   df6c243c             | fild                qword ptr [esp +
0x3c]
            //   d835????????         |
            //   e8????????           |

        $sequence_14 = { 8b44241c 8b4024 e8???????? 50 8b442420 8b4008 50 }
            // n = 7, score = 200
            //   8b44241c             | mov                 eax, dword ptr [esp +
0x1c]
            //   8b4024               | mov                 eax, dword ptr [eax +
0x24]
            //   e8????????           |
            //   50                   | push                eax
            //   8b442420             | mov                 eax, dword ptr [esp +
0x20]
            //   8b4008               | mov                 eax, dword ptr [eax +
8]
            //   50                   | push                eax

        $sequence_15 = { 83c4c0 8bf0 8d3c24 b910000000 }
```

```
            // n = 4, score = 200
            //   83c4c0               | add                 esp, -0x40
            //   8bf0                 | mov                 esi, eax
            //   8d3c24               | lea                 edi, [esp]
            //   b910000000           | mov                 ecx, 0x10

        $sequence_16 = { 52 50 8b442410 8b403c 99 }
            // n = 5, score = 200
            //   52                   | push                edx
            //   50                   | push                eax
            //   8b442410             | mov                 eax, dword ptr [esp +
0x10]
            //   8b403c               | mov                 eax, dword ptr [eax +
0x3c]
            //   99                   | cdq

        $sequence_17 = { 8bea 8bd8 90 90 }
            // n = 4, score = 200
            //   8bea                 | mov                 ebp, edx
            //   8bd8                 | mov                 ebx, eax
            //   90                   | nop
            //   90                   | nop

        $sequence_18 = { 022e 310500000000 0c00 0000 94 }
            // n = 5, score = 200
            //   022e                 | add                 ch, byte ptr [esi]
            //   310500000000         | xor                 dword ptr [0], eax
            //   0c00                 | or                  al, 0
            //   0000                 | add                 byte ptr [eax], al
            //   94                   | xchg                eax, esp

        $sequence_19 = { 6800200000 8b44240c 8b4050 50 }
            // n = 4, score = 200
            //   6800200000           | push                0x2000
            //   8b44240c             | mov                 eax, dword ptr [esp +
0xc]
            //   8b4050               | mov                 eax, dword ptr [eax +
0x50]
            //   50                   | push                eax

        $sequence_20 = { 8bd8 33c0 55 68???????? 64ff30 648920 90 }
            // n = 7, score = 200
            //   8bd8                 | mov                 ebx, eax
            //   33c0                 | xor                 eax, eax
            //   55                   | push                ebp
            //   68????????           |
            //   64ff30               | push                dword ptr fs:[eax]
            //   648920               | mov                 dword ptr fs:[eax],
esp
            //   90                   | nop

        $sequence_21 = { 51 b910000000 f3a5 59 }
            // n = 4, score = 200
            //   51                   | push                ecx
            //   b910000000           | mov                 ecx, 0x10
            //   f3a5                 | rep movsd           dword ptr es:[edi],
dword ptr [esi]
            //   59                   | pop                 ecx

        $sequence_22 = { 8b442418 8b403c 99 030424 13542404 }
            // n = 5, score = 200
            //   8b442418             | mov                 eax, dword ptr [esp +
0x18]
            //   8b403c                 | mov                 eax, dword ptr [eax +
0x3c]
            //   99                     | cdq
            //   030424                 | add                 eax, dword ptr [esp]
            //   13542404               | adc                 edx, dword ptr [esp +
4]

        $sequence_23 = { 8d3c24 b910000000 f3a5 90 90 90 }
            // n = 6, score = 200
            //   8d3c24                 | lea                 edi, [esp]
            //   b910000000             | mov                 ecx, 0x10
            //   f3a5                   | rep movsd           dword ptr es:[edi],
dword ptr [esi]
            //   90                     | nop
            //   90                     | nop
            //   90                     | nop

    condition:
        7 of them and filesize < 54214656
}
```

[TLP:WHITE] win_dbatloader_w0   (20211012 | Detects cryptographic routine)

```
rule win_dbatloader_w0 {

    meta:
        author = "Daniel Plohmann <daniel.plohmann<at>fkie.fraunhofer.de>"
        date = "2021-10-12"
        version = "1"
        description = "Detects cryptographic routine"
        malpedia_reference =
"https://malpedia.caad.fkie.fraunhofer.de/details/win.dbatloader"
        malpedia_rule_date = "20211012"
        malpedia_hash = ""
        malpedia_version = "20211012"
        malpedia_license = "CC BY-SA 4.0"
        malpedia_sharing = "TLP:WHITE"

    strings:
        /*
                                  loc_413930:
        8D 45 E4                  lea     eax, [ebp+var_1C]
        8B 55 F8                  mov     edx, [ebp+var_8]
        8A 54 1A FF               mov     dl, [edx+ebx-1]
        0F B7 CF                  movzx   ecx, di
        C1 E9 08                  shr     ecx, 8
        32 D1                     xor     dl, cl
        */
        $xor_decrypt = { 8D 45 E4  8B 55 F8  8A 54 1A FF  0F B7 CF  C1 E9 08  32
D1 }

    condition:
        all of them
}
```