

Laboratorul 5

Folosirea functiilor din wininet

Wininet (Win32 Internet Functions) este o librerie ce contine functii pentru transmis si citit date pe retea pentru protocoalele http, ftp si gopher(un protocol utilizat pentru cautare de documente, dar nu prea mai este folosit tinand cont ca au aparut motoarele de cautare pe http)

Fiecare protocol suportat de wininet are la baza tcp-ip. Deci, este posibil sa se transmita informatii catre un server http/ftp si fara wininet, prin implementarea directa a protocoalele http si ftp peste tcpip. Dar acest lucru este oarecum inutil si anevoios. Ideal este sa se foloseasca functiile din wininet pentru http si ftp, iar pentru alte protocoale sa se apeleze direct la tcp/ip sau udp.

Pentru a folosi functiile din wininet, trebuie intai sa fie inclusa libraria in proiect. Acest lucru se face fie prin (click de dreapta pe numele proiectului), Properties->Linker->Additional Dependencies si se inregistreaza wininet.lib, fie prin directiva: `#pragma comment(lib, "wininet")` care trebuie scrisa intr-un fisier sursa din proiect

De asemenea, trebuie inclus si fisierul `<wininet.h>` in proiect

Utilizarea wininet pentru HTTP si FTP

Pasii pentru a utiliza functii specifice http si ftp sunt comuni pana intr-un punct. Anumite functii vor returna un handle care este de tip HINTERNET. Aceste handle-uri se inchid prin functia `InternetCloseHandle()`

1. Primul pas este initializarea libreriei wininet prin apelul functiei **InternetOpen**.

```
HINTERNET InternetOpen(  
    _In_ LPCTSTR lpszAgent,  
    _In_ DWORD dwAccessType,  
    _In_ LPCTSTR lpszProxyName,  
    _In_ LPCTSTR lpszProxyBypass,  
    _In_ DWORD dwFlags  
);
```

lpszAgent	Reprezinta un sir de caractere prin care aplicatia se identifica la server. In felul acesta, serverul poate sa-i ofere un continut care sa fie formatat diferit. Poate fi orice.
dwAccessType	Cum sa fie rezolvate numele de domenii. (printr-un proxy, sau direct) Cel mai des folosit este: <code>INTERNET_OPEN_TYPE_PRECONFIG</code> Care indica faptul ca se va folosi metoda configurata

	pentru sistemul de operare
lpszProxyName	Daca la parametrul anterior s-a setat INTERNET_OPEN_TYPE_PROXY, atunci aici e numele proxy-ului. Altfel e NULL
lpszProxyBypass	0 lista de stringuri care sa nu fie rezolvate prin proxy (ex: 127.0.0.1) Daca nu s-a setat INTERNET_OPEN_TYPE_PROXY, atunci parametrul trebuie sa fie NULL
dwFlags	Poate fi specificat ca cererile catre servere sa fie rezolvate offline (dintr-un cache). Sau totul sa mearga asincron (prin setarea flagului INTERNET_FLAG_ASYNC. In cazul acesta trebuie apelata si InternetSetStatusCallback pentru a seta functia de callback). De obicei este NULL.

In caz de succes returneaza un handle, altfel NULL

2. Al dilea pas este conectarea la server si specificarea protocolului. Acesta lucru poate fi facut fie prin functia InternetConnect, fie prin InternetOpenUrl. Avantajul la InternetOpenUrl este ca protocolul si serverul vor fi dedusi din forma url-ului. De ex: http://www.myserver.com indica protocolul http si serverul myserver.com

```
HINTERNET InternetConnect(
    _In_ HINTERNET hInternet,
    _In_ LPCTSTR lpszServerName,
    _In_ INTERNET_PORT nServerPort,
    _In_ LPCTSTR lpszUsername,
    _In_ LPCTSTR lpszPassword,
    _In_ DWORD dwService,
    _In_ DWORD dwFlags,
    _In_ DWORD_PTR dwContext
);
```

hInternet	Handle-ul obtinut de la InternetOpen
lpszServerName	Un string ce contine adresa ip sau numele serverului
nServerPort	Port-ul pe care se face conexiunea: Valorile cele mai des folosite sunt: INTERNET_DEFAULT_FTP_PORT → 21 INTERNET_DEFAULT_HTTP_PORT → 80 INTERNET_DEFAULT_HTTPS_PORT → 443 INTERNET_INVALID_PORT_NUMBER → portul este dedus din tipul protocolului
lpszUsername	User-ul, daca este necesar. Daca este setat NULL, se va folosi user-ul standard (anonymous pentru ftp)

lpszPassword	Parola, daca este necesar. Pentru ftp, daca useru-ul este NULL, se va folosi anonymous. Altfel, se va folosi un string gol
dwService	Tipul protocolului Cele mai folosite valori sunt: INTERNET_SERVICE_FTP INTERNET_SERVICE_HTTP
dwFlags	Se poate seta INTERNET_FLAG_PASSIVE pentru a merge pe format passive la ftp
dwContext	0 valoare prin care sa se identifice firul care a facut apelul la functie, in caz ca se doreste ca apelul sa fie asincron.

```

HINTERNET InternetOpenUrl(
    _In_ HINTERNET hInternet,
    _In_ LPCTSTR lpszUrl,
    _In_ LPCTSTR lpszHeaders,
    _In_ DWORD dwHeadersLength,
    _In_ DWORD dwFlags,
    _In_ DWORD_PTR dwContext
);

```

hInternet	Handle-ul obtinut de la InternetOpe
lpszUrl	Un string ce contine adresa ip sau numele serverului impreuna cu protocolul. Ex: http://www.test.com Sunt suportate protocoalele http,https,ftp
lpszHeaders	Daca se doreste headere suplimentare
dwHeadersLength	Dimensiunea headere-lor noi adaugate
dwFlags	Se poate seta INTERNET_FLAG_PASSIVE pentru a merge pe format passive la ftp. Sau INTERNET_FLAG_SECURE pentru a trece pe HTTPS. Daca se seteaza parametrul INTERNET_FLAG_RAW_DATA (care este doar pentru FTP) informatiile despre fisiere vor fi returnate sub forma unei structure WIN32_FILE_DATA
dwContext	0 valoare prin care sa se identifice firul care a facut apelul la functie, in caz ca se doreste ca apelul sa fie asincron.

Daca functiile reusesc, vor returna un handle de tip HINTERNET. Handle-urile trebuie inchise in ordinea inversa in care au fost deschise. Intai handle-ul de la InternetConnect, apoi cel de la InternetOpen

Utilizarea functiilor pentru HTTP

Dupa initializarea conexiunii, urmeaza partea in care se vor face cereri catre serverul http si se va citi raspunsul. Sunt 3 pasi pentru a face asta. Prima data trebuie construit pachetul, apoi trebuie trimis, iar in final este citit.

3' Construirea pachetului http

Acest lucru se face prin intermediul functiei HTTPOpenRequest. Aici, nu se va trimite nici un request. Doar se construiesc un pachet. Functia HTTPOpenRequest va primi headerele principale care sunt definite in protocolul HTTP 1.1 in fiecare din parametrii sai. Pentru headere suplimentare se poate folosi functia HttpAddRequestHeaders

Mai jos este un header http pe care browser-ul(chrome) il trimite pentru infoiasi.ro/~orar

```
GET ~orar HTTP/1.1
Host: www.infoiasi.ro
Connection: keep-alive
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/41.0.2272.101 Safari/537.36
Accept-Encoding: gzip, deflate, sdch
Accept-Language: en-US,en;q=0.8,ro;q=0.6
Cookie: JSESSIONID=46C1DCA66576259BB8424CEDB3FF36CD
```

```
HINTERNET HttpOpenRequest(
    _In_ HINTERNET hConnect,
    _In_ LPCTSTR lpszVerb,
    _In_ LPCTSTR lpszObjectName,
    _In_ LPCTSTR lpszVersion,
    _In_ LPCTSTR lpszReferer,
    _In_ LPCTSTR *lplpszAcceptTypes,
    _In_ DWORD dwFlags,
    _In_ DWORD_PTR dwContext
);
```

hInternet	Handle-ul obtinut de la InternetConnect
lpszVerb	GET, POST sau PUT (GET pentru exemplul de mai sus)
lpszObjectName	Obiectul de pe server (~orar pentru exemplul de mai sus)
lpszVersion	Versiunea protocolului (1.1 pentru exemplul de mai sus)
lpszReferer	De pe ce pagina s-a ajuns aici. Lipseste in exemplul de mai sus, intrucat adresa a fost direct introdusa in browser. Poate fi de ex: www.google.com
lplpszAcceptTypes	Un vector de stringuri ce specifica tipul de informatii

	acceptat de client. Ultimul element trebuie sa fie NULL. Poate fi specificat si text/* pentru a permite format care poate fi sub forma de text sau */* pentru orice tip de format
dwFlags	Flaguri prin care se pot specifica diferiti parametrii specifici HTTP. De ex: INTERNET_FLAG_Secure pentru https
dwContext	0 valoare prin care sa se identifice firul care a facut apelul la functie, in caz ca se doreste ca apelul sa fie asincron.

Daca functia reuseste, se returneaza un handle de tip Hinternet.

4' Trimitearea pachetului http

Aceasta functie face conexiunea cu serverul.

```

BOOL HttpSendRequest(
    _In_ HINTERNET hRequest,
    _In_ LPCTSTR lpszHeaders,
    _In_ DWORD dwHeadersLength,
    _In_ LPVOID lpOptional,
    _In_ DWORD dwOptionalLength
);

```

hRequest	Handle-ul obtinu de la HttpOpenrequest
lpszHeaders	Daca se doresc headere suplimentare, pot fi scrise aici (cate unul pe o linie)
dwHeadersLength	Dimensiunea headerlor trimise anterior
lpOptional	Daca s-a specificat POST la verb pentru functia HTTPOpenRequest, atunci datele care trebuie trimise (nu mai apar in url) sunt specificate prin acest parametru. Datele vor fi puse la sfarsitul pachetului HTTP
dwOptionalLength	Dimensiunea datelor trimise prin post.

Daca functia reuseste, va returna o valoare diferita de 0, altfel va returna FALSE

5' Citirea raspunsului

Odata trimisa cererea catre server, poate fi citit raspunsul. Acest lucru se face prin intermediul functiei InternetReadFile. Parametrii sunt foarte asemenatori functiei ReadFile

```

BOOL InternetReadFile(
    _In_ HINTERNET hFile,
    _Out_ LPVOID lpBuffer,
    _In_ DWORD dwNumberOfBytesToRead,
    _Out_ LPDWORD lpdwNumberOfBytesRead
);

```

);

hFile	Handle-ul obtinut de la HttpOpenrequest
lpBuffer	O zona de memorie in care sa fie puse datele citite
dwNumberOfBytesToRead	Numarul maxim de bytes pe care sa-l citeasca. (maxim dimensiunea buffer-ului)
lpdwNumberOfBytesRead	Un pointer catre un dword unde sa puna numarul de bytes cititi

Utilizarea functiilor pentru FTP

Pe baza handle-ului obtinut la InternetConnect se pot apela functii FTP. Acestea sunt folosite asemanator functiilor pentru sistemul de fisiere (intai Createfile apoi ReadFile/WriteFile). Functiile au FTP in fata numelui.

Ex: FTPOpenFile pentru deschis un handle catre fisier, apoi InternetReadfile pentru citirea datelor, iar la sfarsit InternetClosehandle

```
HINTERNET FtpOpenFile(  
    _In_ HINTERNET hConnect,  
    _In_ LPCTSTR lpszFileName,  
    _In_ DWORD dwAccess,  
    _In_ DWORD dwFlags,  
    _In_ DWORD_PTR dwContext  
);
```

hConnect	Handle-ul obtinut de la InternetConnect
lpszFileName	Path-ul fisierului de pe ftp
dwAccess	GENERIC_READ sau GENERIC_WRITE
dwFlags	FTP_TRANSFER_TYPE_BINARY sau FTP_TRANSFER_TYPE_ASCII sunt cele mai folosite
dwContext	O valoare prin care sa se identifice firul care a facut apelul la functie, in caz ca se doreste ca apelul sa fie asincron.

Daca functia reuseste, va returna un handle. Acesta trebuie inchis cu InterentCloseHandle. Handle-ul obtinut va fi trimis catre functia InternetReadFile.

Exista si o comanda prin care se poate descarca un fisier, fara deschide intai handle-ul: FtpGetFile

```
BOOL FtpGetFile(  
    _In_ HINTERNET hConnect,  
    _In_ LPCTSTR lpszRemoteFile,  
    _In_ LPCTSTR lpszNewFile,  
    _In_ BOOL fFailIfExists,  
    _In_ DWORD dwFlagsAndAttributes,
```

```

    _In_   DWORD dwFlags,
    _In_   DWORD_PTR dwContext
);

```

hConnect	Handle-ul obtinut de la InternetConnect
lpszRemoteFile	Path-ul fisierului de pe ftp
lpszNewFile	Path-ul fisierului local
fFailIfExists	Indica daca sa suprascrie fisierul care se afla local
dwFlagsAndAttributes	Atributele pentru fisierul descarcat, de ex: FILE_ATTRIBUTES_NORMAL
dwFlags	FTP_TRANSFER_TYPE_BINARY sau FTP_TRANSFER_TYPE_ASCII sunt cele mai folosite
dwContext	O valoare prin care sa se identifice firul care a facut apelul la functie, in caz ca se doreste ca apelul sa fie asincron.

Asemănător cu această funcție este și funcție pentru upload de fișiere: FtpPutFile. Alte funcții utile sunt: FtpDeleteFile, FtpRenameFile, FtpRemoveDirectory, FtpCreateDirectory.

Pentru listare de fișiere, sunt funcțiile FTPFindFirstFile și InternetFindNextFile. Acestea sunt asemănătoare cu FindFirstFile și FindNextFile de la sistemul de fișiere. Diferența o constă în faptul că FTPFindFirstFile primește un handle către conexiunea ftp, iar ultimii 2 parametrii specifică o serie de flaguri și un context folosit în cazul în care acțiunile se fac asincron.

```

HINTERNET FtpFindFirstFile(
    _In_   HINTERNET hConnect,
    _In_   LPCTSTR lpszSearchFile,
    _Out_  LPWIN32_FIND_DATA lpFindFileData,
    _In_   DWORD dwFlags,
    _In_   DWORD_PTR dwContext
);

```

hConnect	Handle-ul obtinut de la InternetConnect
lpszSearchFile	O locatie unde sa se faca listarea. Poate contine si wildcarduri (asemanator ca la findfirstfile). Nu poate contine spatii
lpFindFileData	Pointer catre o structura unde vor fi puse informatiile despre fisierul gasit
dwFlags	INTERNET_FLAG_RELOAD este cel mai folosit. (refresh)
dwContext	O valoare prin care sa se identifice firul care a facut apelul la functie, in caz ca se doreste ca apelul sa fie asincron.

Handle-urile trebuie inchise in ordinea inversa in care au fost deschise. Toate handle-urile rezultate in urma unui apel a unei functii din wininet, sunt de tip HINTERNET si trebuie inchise cu InternetCloseHandle