



Upload e Download via FTP con Inet

http://www.vbsimple.net/cliserv/clser_03.htm

Difficoltà: 3 / 5

In questo esempio vedremo come [uploadare](#) o [downloadare](#) un file su un server [remoto](#) tramite FTP. L'esempio sfrutterà il controllo Inet; pertanto dall'elenco componenti (CTRL+T) del menu Progetto selezioneremo il controllo **Microsoft Internet Transfer Control**, detto anche Inet Control .

Se vogliamo uploadare un file sul nostro sito per poterlo mettere on line o per renderlo disponibile a terzi, oppure vogliamo scaricare un file da uno spazio FTP pubblico o del quale conosciamo la password, abbiamo due possibilità.

Se siamo degli smanettoni e ci piacciono le finestre testuali in stile **xterm** (un terminale testuale), apriamo il *prompt di MS-DOS* e cominciamo a digitare i comandi del protocollo FTP.

Se invece non abbiamo dimestichezza con comandi strani che non conosciamo e vogliamo comodamente premere due o tre pulsanti per completare la nostra operazione, usiamo il nostro client FTP preferito (io personalmente uso sia il prompt del DOS che il programma WS_FTP).

Il client però farà esattamente ciò che noi faremo in maniera testuale...

Cerchiamo di capire quindi cosa sta dietro la negoziazione di una sessione FTP. Tutti gli ISP (*Internet Service Provider*) dovrebbero dare uno spazio seppur esiguo. Se non siete a conoscenza di un host con servizio FTP potreste provare ad implementare uno scanner che tenti delle connessioni con utente **anonymous** seguendo il [tutorial sulla programmazione del Winsock](#) già pubblicato dal sottoscritto ;)

Apriamo quindi il prompt DOS e digitiamo "**ftp <nome-server>**"
Come messaggio di benvenuto otterremo qualcosa del genere:

Connesso a chili.iol.it

*220 chili.iol.it FTP server (Version wu-2.4-IOL(117) Thu Dec 18 18:48:39 MET 1997)
ready.*

Dobbiamo quindi specificare l'utente e la password che intendiamo utilizzare.

Se il server consente un accesso anonimo possiamo semplicemente usare "**anonymous**" come UserID e il nostro indirizzo e-mail come password.

Utente (chili.iol.it(none)): icarus

331 Password required for icarus.

Password: **nonveladico**

230- 19110322 bytes disponibili su 31457280.

230 User icarus logged in.

Ok, siamo loggati e adesso smazziamoci i comandi...ah già, ma non li conosciamo...

Come faccio a fare un client FTP in VB se non conosco neanche i comandi fondamentali?!

Che tristezza, che sconforto...

Meglio che continuo a sfondarmi di Quake3 e Diablo2.

Niente paura! Prima di cominciare a parlare del codice facciamo una piccola digressione riguardo al protocollo FTP :)

Il tutorial si allungherà un pochino, ma è necessario per andare avanti.

Chiediamo quindi aiuto proprio al [demone](#) FTP che ci ha risposto, digitando proprio la parola "**help**".

Help

I comandi possono essere abbreviati.

I comandi sono:

! delete literal prompt send

? debug ls put status

etc. etc.

Possiamo addirittura sapere nello specifico cosa fa un comando digitando

<help nome-comando>.

help mput

mput invia gruppi di file

Dopo aver negoziato la sessione col demone e aver immesso userID e password, se volessimo sapere quanto spazio abbiamo a disposizione e quali file sono già presenti basterebbe digitare il comando **dir** uguale a quello del DOS.

Dobbiamo poi decidere il modo in cui uploadare il nostro file. Se il nostro è un file di testo sceglieremo il modo "**ascii**", ma se è un Mp3 da mettere sul nostro sito dobbiamo scegliere il modo "**binary**". I comandi per la scelta della modalità sono rispettivamente **ascii** e **binary**. Digitiamo quindi il nome del modo che ci serve ed avremo la conferma dal demone.

Un comando utile per quando dobbiamo trasferire più files è **prompt**, che evita di farci ripetere sempre la domanda se siamo sicuri di voler caricare il file. Se i file sono 10 possiamo anche rispondere sempre di sì, ma se i file sono 100 possiamo anche andare a prenderci un caffè ed evitare di stare dietro all'upload continuando a rispondere "yes". Quindi digitando **prompt** otteniamo in risposta l'attivazione o la disattivazione della domanda di conferma. Lo stato che ci interessa raggiungere è "**prompt off**".

Supponiamo che il file icarus.vbp che vogliamo uploadare si trovi nella directory **c:\miei_files\vb**. Dobbiamo quindi cambiare la directory di lavoro [locale](#).

Per questo ci viene in aiuto il comando **lcd** che significa appunto 'local change directory'.

Se dovessimo invece cambiare directory all'interno dello spazio FTP useremmo il comando **cd** uguale al comando DOS.

Dopo tutti questi preparativi, noiosi ma necessari, possiamo passare alla vera fase di upload. Il comando di upload vero e proprio non è unico ma ne esistono due. Uno per uploadare una serie di file (ad esempio *.vbp) elencati utilizzando le wildcard e/o separandoli da spazio e uno per uploadare un file unico. I comandi sono rispettivamente **mput** e **put**.

Per quanto riguarda il download esistono comandi analoghi che, come possiamo comprendere spulciando l'elenco fornitoci con **help**, saranno **mget** e **get**.

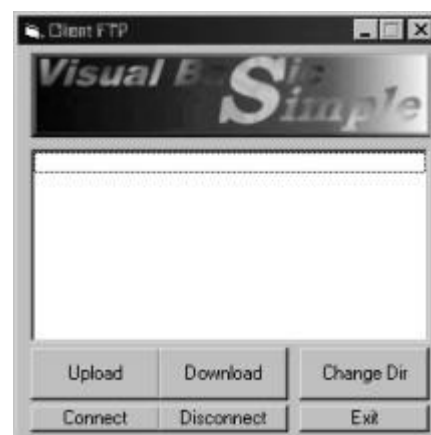
È possibile compiere anche altre operazioni come il cambio del nome, ma non elencheremo tutti i comandi uno per uno in questo tutorial. Basta affidarsi sempre all'help in linea.


Cominciamo quindi con il codice vero e proprio.


Per il design del nostro client utilizzeremo un paio di pulsanti in basso per connetterci, disconnetterci ed uscire dal programma, i cui loro rispettivi nomi saranno **cmdConnect**, **cmdDisconnect** e **cmdExit**.


Avremo anche una serie di pulsanti per tre funzioni elementari: **cmdUpload**, **cmdDownload** e **cmdChangeDir**.

Al centro abbiamo una Listbox  chiamata **ListOut** per elencare i files remoti.



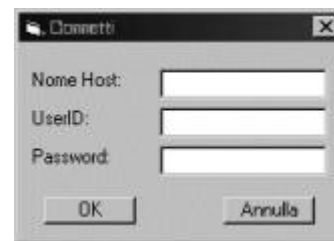
L'ideale da fare sarebbe usare due ListBox per selezionare i files e magari uploadarli e downloadarli con un doppio click, oppure con una funzione di *drag'n'drop*, ma direi che sono argomenti che esulano dal target di questo tutorial e aggiungerebbero solo confusione all'argomento che vogliamo trattare, cioè l'uso del controllo  **Internet Transfer Control**.

Proprio questo sarà la prima cosa da piazzare all'interno del form e gli daremo il nome Inet1; nella finestra dell proprietà specificheremo il valore **icFTP** nella proprietà  **Protocol** e **21** nella proprietà **RemotePort**. Le altre proprietà le imposteremo da codice.

Un ultimo controllo da inserire è un Timer  di nome **Timer1** che utilizzeremo per ritardare l'esecuzione di un comando. Impostiamo per esso le proprietà **Interval** a **1000** e la **Enabled** a **False**.

Mi pareva carino creare una dialog a parte per effettuare il login e così ho fatto.

Aggiungiamo quindi un form ed inseriamo in esso tre *Label* **A**, tre *TextBox* **ab1** denominate **txtHost**, **txtUserID** e **txtPassword** e due *CommandButton* **ab2** denominati **OKButton** e **CancelButton**.



La prima textbox servirà per immettere l'indirizzo dell'host a cui collegarci. Per inciso sarebbe la parte che viene dopo ftp://.

La seconda e la terza textbox serviranno per specificare UserID e Password. Se l'accesso al server non è anonimo occorrerebbe settare un carattere (di solito '*') per lo shadowing della password.

Pertanto impostiamo a "*" la proprietà PasswordChar della TextBox **txtPassword**.

Il contenuto di queste tre textbox verrà poi concatenato per formare qualcosa che assomiglia a **ftp://UserID:Password@serverFtp.com** ovvero un URL di accesso ad un sito FTP, comprendente UserID, Password e nome del server.

Se proviamo a digitare tale URL nel nostro browser dovremmo riuscire a navigare in modo grafico il nostro spazio FTP.


A questo punto ci si chiederebbe anche perché dobbiamo piagarci le dita sulla tastiera per battere codice per fare un client sgrauso e poco affidabile.

Questo tutorial è solo un esempio, *un pretesto* per imparare qualcosa dal quale poi tireremo fuori quello che vorremo a seconda della nostra creatività e del nostro ingegno.

E' utile imparare il maggior numero di cose e mettersele in tasca. Quando poi si dovrà risolvere un problema avremo a disposizione un carnet di nozioni e l'unica cosa da fare sarà scegliere la soluzione migliore. Imparare queste cose senza uno scopo preciso è comunque utile. Non tanto imparare per il gusto di imparare, ma per quello che un mio amico e mentore (Master SppMember) mi ha insegnato col nome di **Serendipity**.

In breve...

Può capitare che trapanandosi il cervello per risolvere un problema si trovi la soluzione o ancora meglio l'ispirazione per qualcosa che con il problema originale non ha nulla a che fare. Il caso, che porta alla scoperta o all'illuminazione, è la Serendipity. L'ingegno dell'uomo che sfoga la sua creatività in qualcosa, non si sa cosa, ma tutto il sapere che si ha in testa prima o poi sfocerà in qualcosa.

Utilizziamo quindi i tre valori che abbiamo inserito nelle textbox andando a vedere il primo e più importante metodo  del controllo Inet.

Execute

La sintassi è la seguente:

object.Execute url, operation, data, requestheader

Se come url passiamo l'indirizzo del server FTP e usiamo il metodo Execute solo per

connetterci allora gli altri parametri per adesso non serviranno.

Cominciamo a vedere il codice della semplice dialog per effettuare il login:

```
1. Option Explicit
2.
3. Private Sub CancelButton_Click()
4.     Me.Hide
5. End Sub
6.
7. Private Sub OKButton_Click()
8.     Dim URLACCESSO As String
9.     On Error GoTo ERRORE
10.    URLACCESSO = "ftp://" & txtUserID.Text & ":" & txtPassword.Text & "@" &
        txtHost.Text
11.    Ftp.ID = "DIR"
12.    Ftp.Inet1.Execute URLACCESSO
13.    Me.Hide
14.    Exit Sub
15. ERRORE:
16.    MsgBox "ERRORE: " & Err.Description, vbCritical + vbOKOnly
17. End Sub
```

Il click sul pulsante Annulla nasconderà il form in questione, restituendo il controllo al form principale che ha richiamato questo.

Il click sopra il pulsante OK dovrà tentare la connessione all'host specificato.

Alla riga 9 abbiamo una semplice istruzione di gestione degli errori.

Alla riga 10 viene costruito l'URL di accesso al server concatenando i valori delle Textbox con le apposite stringhe (':', '@').

Siamo nelle righe di codice della dialog e il controllo Inet che andiamo a utilizzare si trova nel form principale. Quindi per utilizzarlo dovremo effettuare il corretto riferimento al form principale, partendo dal nome del form per arrivare al nome del controllo.

Alla riga 11 viene impostata la variabile **ID** a "DIR" e dopo vedremo perché.

Alla riga 12 viene effettuata la connessione al server mediante l'utilizzo dell'URL appena formato. Fatto questo nasconderemo la dialog di connessione che ormai non ci serve più e usciamo dalla funzione.

Se il collegamento avesse generato un errore sarebbe stata mostrata una finestra esplicativa dell'errore (riga 16).

Passiamo quindi al codice del form principale.

```
1. Option Explicit
2. Public ID As String
3.
4. Private Sub cmdConnect_Click()
5.     Load login
6.     login.Show vbModal
7.     Unload login
8. End Sub
9.
```

```

10. Private Sub cmdDisconnect_Click()
11.     Inet1.Execute "", "CLOSE"
12. End Sub
13.
14. Private Sub cmdExit_Click()
15.     Unload Me
16. End Sub
17.

```

Alla riga 2 abbiamo dichiarato una variabile pubblica (cioè accessibile da ogni punto del codice) di nome **ID**. Essa verrà utilizzata per identificare i comandi da eseguire. In particolare verrà utilizzata per mostrare il contenuto della cartella remota ogni qual volta viene cambiata la cartella corrente oppure alla fine di un upload.

Il codice compreso tra le righe 4-8 caricano, mostrano e scaricano la finestra **dialog** per effettuare il login al server remoto. Questo sarà eseguito alla pressione del pulsante **Connect**.

Il click sopra il pulsante **Disconnect** dovrà disconnettere il client dal server e questo viene effettuato mediante l'invio del comando "**CLOSE**" (riga 11).


Il click sopra il pulsante **Exit** provocherà lo scaricamento del form corrente e quindi la chiusura del programma (riga 15).

```


18. Private Sub Inet1_StateChanged(ByVal State As Integer)
19.     Dim POSINVIO As Integer
20.     Dim RITORNO As String
21.     If (ID = "DIR") And (State = icResponseCompleted) Then
22.         RITORNO = Inet1.GetChunk(1024, icString)
23.         POSINVIO = InStr(1, RITORNO, vbNewLine)
24.         While POSINVIO > 0
25.             If Left(RITORNO, POSINVIO - 1) <> " " Then ListOut.AddItem Left(RITORNO,
POSINVIO - 1)
26.             RITORNO = Mid(RITORNO, POSINVIO + 2)
27.             POSINVIO = InStr(1, RITORNO, vbNewLine)
28.         Wend
29.         ID = " "
30.     End If
31. End Sub
32.

```

Ecco l'evento  **StateChanged** del controllo Inet.

Come dice il nome stesso riguarda il cambiamento nello stato della connessione. Viene fornito un valore State che indica un numero dell'[enumerazione](#)  **StateConstants**.

Il valore che utilizzeremo è l'**icResponseCompleted** ovvero il completamento dell'operazione. Pertanto, alla riga 12, viene controllato se il valore della variabile ID è "DIR" e lo stato è icResponseCompleted. Ciò significherà che il codice seguente sarà eseguito soltanto al completamento dell'operazione quando ID è uguale a "DIR". Più precisamente il codice sarà eseguito ogni volta che vogliamo aggiornare il contenuto della **ListOut**.


Alla riga 22 leggiamo i dati dal controllo Inet1 mediante l'utilizzo del metodo GetChunk. Tali dati saranno memorizzati nella variabile **RITORNO** e conterranno l'elenco dei files e delle cartelle nella directory di lavoro remota. Ogni file o cartella è separato dall'altro da un INVIO ([costante](#)  vbNewLine).

Pertanto, prima di presentare i dati all'utente, sarà necessario estrarre i singoli nomi dalla

variabile **RITORNO**. Per effettuare quest'operazione sarà utilizzato il ciclo alle righe 23-28 che ricercherà le posizioni di vbNewLine ed inserirà ogni voce nella ListBox **ListOut**.

Effettuata quest'operazione verrà azzerato il valore di ID, che verrà reimpostato ogni volta che si desidera elencare il contenuto della cartella remota.

```
33. Private Sub Timer1_Timer()  
34.     ID = "DIR"  
35.     Inet1.Execute "", "DIR"  
36.     Timer1.Enabled = False  
37.     ListOut.Clear  
38.     ListOut.AddItem "./"  
39.     ListOut.AddItem "../"  
40. End Sub  
41.
```

Il Timer  che abbiamo inserito sul form serve esclusivamente per elencare i files all'interno della cartella remota. Abbiamo dovuto inserirlo poiché se vogliamo elencare il contenuto della cartella subito immediatamente dopo essere entrati nella cartella, viene generato un errore. Il socket ha bisogno di una certa quantità di tempo prima di poter richiedere dati riguardo la cartella corrente. Non potremo cambiare cartella e al contempo elencarne il contenuto. Dovremo necessariamente attendere qualche decimo di secondo e poi richiamare il comando **DIR** di elencazione. Ecco il motivo di questo Timer.

Alla riga 34 impostiamo la variabile **ID** a DIR, per il motivo già visto alla riga 21. Fatto questo richiediamo il comando **"DIR"** mediante il solito metodo **Execute**, disattiviamo il timer, ripuliamo la ListOut (riga 37) ed aggiungiamo i due collegamenti simbolici "." e ".." per permettere il ritorno alla cartella precedente (righe 38 e 39).

```
42. Private Sub cmdDownload_Click()  
43.     On Error GoTo ERRORE  
44.     Inet1.Execute "", "GET " & ListOut.Text & " " & ListOut.Text  
45.     Exit Sub  
46. ERRORE:  
47.     MsgBox "ERRORE: " & Err.Description, vbCritical + vbOKOnly  
48. End Sub  
49.
```

Il click sopra il pulsante Download scaricherà il file selezionato nella ListBox. Alla riga 43 abbiamo un'istruzione di gestione degli errori. Subito dopo verrà eseguito il metodo Execute passandogli come valore il comando GET seguito dal nome del file da scaricare ed il nome da assegnare al file scaricato. Il nome in questione saranno estratti dalla proprietà Text della ListOut. Eseguita l'operazione si uscirà dalla funzione.

Alla riga 47 c'è la funzione che mostra una finestra di errore ogni qual volta si verifica un errore all'interno di questa Sub.

```
50. Private Sub cmdUpload_Click()  
51.     On Error GoTo ERRORE  
52.     Inet1.Execute "", "PUT ./ftp.frm ftp.frm"  
53.     Timer1.Enabled = True  
54.     Exit Sub  
55. ERRORE:  
56.     MsgBox "ERRORE: " & Err.Description, vbCritical + vbOKOnly  
57. End Sub  
58.
```


Il click sopra il pulsante Upload invierà il file FTP.FRM all'host remoto. Naturalmente in un vero client FTP il nome del file verrebbe scelto nel computer locale, ma questo esempio vuole soprattutto spiegare il meccanismo di collegamento tra i due computer, e non vuole essere un completo client FTP.

Alla riga 52 viene eseguito il metodo **Execute** fornendogli come comando **PUT** seguito dal nome del file locale e il nome che il file deve assumere nel computer remoto.

In seguito a questo viene attivato il Timer1 che provvederà autonomamente a richiedere l'elencazione della cartella remota dopo circa un secondo dall'attivazione. Questo per richiedere l'aggiornamento dell'elenco **ListOut**.

```
59. Private Sub cmdChangeDir_Click()  
60.     On Error Resume Next  
61.     Inet1.Execute "", "CD " & ListOut.Text  
62.     Timer1.Enabled = True  
63. End Sub
```

L'ultimo comando che vedremo sarà quello che permetterà il cambio di cartella di lavoro remota. Al click sopra il pulsante **ChangeDir** verrà inviato il comando **CD** seguito dal nome della cartella, letto dalla **ListOut**.

In seguito all'invio del comando viene attivato il Timer1 che, dopo un secondo, rielencherà il contenuto della nuova cartella remota.

Possiamo passare all'esecuzione del programma. Lanciamolo, premiamo il pulsante Connect, inseriamo il nome dell'host a cui collegarci, lo UserID e la Password e premiamo il pulsante OK. Se i dati inseriti saranno corretti, dovremmo vedere il contenuto della cartella remota.

Una volta connessi, sarà possibile navigare tra le cartelle, cliccando sul nome della cartella e premendo il pulsante ChangeDir, potremo Uploadare il file FTP.FRM sul server remoto e downloadare sul nostro computer il file selezionato.



Figura 3



Figura 4

Conoscendo il protocollo FTP possiamo associare a diversi pulsanti un metodo Execute con l'argomento appropriato, per [uploadare](#), [downloadare](#), rinominare, cancellare tutti i file che vogliamo sul nostro spazio FTP.

Le cose che si possono fare con questo controllo sono molteplici e sta alla fantasia del programmatore svilupparle.

[Alberto Alagna](#)
17 Marzo 2001



[Torna all'indice Client / Server](#)
