

# Støtte for kommandospråk for oppringingsskript

Copyright (c) 1997 Microsoft Corp.  
Shiva Corporation har levert en del av informasjonen i dette dokumentet.

## Slik bruker du dette dokumentet

Dette dokumentet er hovedsakelig skrevet for Internett-leverandører slik at de kan opprette skript som sluttbrukere kan endre for sine egne tilkoblinger.

## Slik bruker du skript i Internet Explorer

Med Iescript.exe kan sluttbrukere bruke skriptfiler (scp-filer) for å opprette tilkoblinger. Iescript legger til innstillinger i tilkoblingsfilen (con-filen) som vist nedenfor. Bruk Iescript når du redigerer tilkoblingsfilen. Ikke rediger tilkoblingsfilen manuelt.

Navn på avsnitt: **Skript**

Følgende funksjoner støttes:

### **ScriptEnabled**

Yes/No Angir om skriptverktøy skal aktiveres.

### **ScriptFileName**

En fullstendig kvalifisert streng som angir plasseringen til skriptfilen som er tilordnet tilkoblingsfilen.

### **ScriptTerminal**

Yes/No Angir om skriptterminalvinduet skal lastes under avspilling av skript.

### **ScriptRecord**

Yes/No Angir om den kjørbare skriptfilen skal gå over i innspillingsmodus når den aktiveres.

### **Eksempler på oppføringer i tilkoblingsfiler for skript**

#### **Eksempel 1:**

```
[Skript]
ScriptEnabled=Yes
ScriptFileName=C:\Shiva\Myscript.Scp
ScriptTerminal=Yes
ScriptRecord=No
```

#### **Eksempel 2:**

```
[Script]
ScriptEnabled=Yes
ScriptFileName=C:\Shiva\NewScrip.Scp
ScriptTerminal=Yes
ScriptRecord=Yes
```

**Obs!**

- Eksempel 2 viser hva som kommer frem når et skript spilles inn.

## Opprette et skript i Internet Explorer

Hvis du vil opprette en skriptfil for sluttbrukere, må du først spille inn en skriptfil med Scripter.exe. Etter at du har opprettet denne filen, kan sluttbrukeren redigere den og sette inn egne navn og passord.

### Innspilling og avspilling av skript med terminalvinduet

- 1 Kjør Scripter.exe. Filen ligger vanligvis i samme katalog som Internet Explorer-programmet.
- 2 Velg Spill inn på Fil-menyen.
- 3 Angi plassen for skriptfilen, og merk deretter skriptfilen (scp-filen) i listen over filer, eller skriv inn et nytt filnavn. Når du starter innspillingen, vises følgende vindu.
- 4 Åpne programgruppen for Internet Explorer, og velg deretter Ny tilkobling. Følg instruksjonene på skjermen for å opprette en ny tilkobling. (Du kan eventuelt velge et tilkoblingsikon du allerede har opprettet.)
- 5 Merk tilkoblingsikonet. Velg Egenskaper på Fil-menyen. Kontroller at det er merket av for Vis terminalvindu etter oppringing.
- 6 La tilkoblingen til Internett-serveren fullføres. ShivaRemote Scripter-vinduet viser ledeteksten "Started!!!".
- 7 Klikk i innspillingsvinduet (som nå fungerer som terminalvindu), og skriv deretter inn informasjonen som trengs for å logge på Internett-leverandørens server (ISP-serveren).
- 8 Etter at påloggingen er fullført, velger du Avslutt innspilling i skriptvinduet.
- 9 Velg OK i ShivaPPP Scripter-vinduet.
- 10 Hvis du vil avslutte samtaler, velger du Avslutt i dialogboksen for tilkobling.

**Obs!**

Ved bruk av skriptfilen må brukeren kontrollere at det ikke er merket av for Vis terminalvindu etter oppringing.

Når du er ferdig med å skrive inn den informasjonen som trengs i skriptet, velger du Stopp. Vinduet lukkes, og den vanlige PPP-forhandlingen begynner.

Skriptet som er opprettet av programmet, ser slik ut:

```
; C:\SCRIPTS\IE\LOGIN.SCP
; Opprettet: 1. juli 1996, 17:07:11
;
;
proc main
string szPassword
transmit "^M"
waitfor "Vertsnavn: ",matchcase until 15
transmit "spry01^M"
waitfor "UIC: ",matchcase until 6
transmit "spry053514^M"
waitfor "Passord: ",matchcase until 3
if $PASSWORD then
    transmit $PASSWORD
```

```

transmit "^M"
goto doneTxPassword
endif
getinput "Passord: " szPassword
transmit szPassword
transmit "^M"
doneTxPassword:
endproc

```

### Redigere skriptfilen basert på den innspilte filen, eller opprette en ny skriptfil

Scripter.exe spiller inn en skriptfil som inneholder brukerinformasjon, inkludert brukernavn og passord. Hvis du er Internett-leverandør, vil du kanskje tilby brukerne en standard skriptfil som kan redigeres og brukes av alle sluttbrukere.

Bruk eksemplet nedenfor som retningslinje hvis du skal opprette et standardskript som ikke inneholder bestemt brukerinformasjon.

```

;-----
; Filnavn: LOGIN01.SCP
; Opprettet dato:      18. oktober 1996
; Opprettet klokkeslett: 12:00:00
;-----

proc main

; Enkelte systemer krever et tastetrykk for å starte
påloggingsprosedyren
;-----
----

transmit "^M"

; Sender vertsnavnet, som er hardkodet av leverandøren
;-----

waitfor "Vertsnavn:"
transmit "hostname"
transmit "^M"

; Sender brukernavnet
; Hentes fra brukernavnfeltet i tilkoblingsfilen
;-----

waitfor "Brukernavn:"
transmit $USERID
transmit "^M"

```

```

; Sender passordet
; Hentes fra passordfeltet i tilkoblingsfilen
;-----

waitfor "Passord:"
transmit $PASSWORD
transmit "^M"

endproc

```

## Avansert informasjon om skript

### 1.0 Oversikt

Mange Internett-leverandører og elektroniske tjenester krever at brukeren skriver inn informasjon manuelt, for eksempel brukernavn og passord, for å opprette en tilkobling. Hvis brukeren har støtte for Ekstern pålogging, kan vedkommende skrive et skript for å automatisere denne prosessen.

Et skript er en tekstfil som inneholder mange kommandoer, parametre og uttrykk som kreves av Internett-leverandøren eller den elektroniske tjenesten for at tilkoblingen skal opprettes og tjenesten kunne brukes. Du kan bruke et hvilket som helst tekstredigeringsverktøy, for eksempel Notisblokk, når du skal opprette en skriptfil. Etter at du har opprettet skriptfilen, kan du tilordne den til en bestemt skripttilkobling for Ekstern pålogging ved hjelp av skriptverktøy for Ekstern pålogging.

### 2.0 Grunnleggende skriptstruktur

En kommando er den mest grunnleggende instruksjonen som skriptfiler inneholder. Noen kommandoer krever parametre som definerer mer nøyaktig hva kommandoen skal gjøre. Et uttrykk er en kombinasjon av operatører og argumenter som tilsammen utgjør en helhet. Uttrykk kan brukes som verdier i alle slags kommandoer. Noen eksempler på uttrykk er aritmetikk, relasjonssammenligninger og strengsammenkoblinger.

Nedenfor ser du den mest grunnleggende typen skript for Ekstern pålogging:

```

;
; En kommentar begynner med semikolon og gjelder til
; slutten av linjen.
;

proc main
    ; Et skript kan inneholde et hvilket som helst antall
    ; variabler og kommandoer

    variabeldeklarasjoner

    kommandoblokk

endproc

```

Et skript må ha en hovedprosedyre, angitt av **proc**-funksjonen, og en samsvarende **endproc**-funksjon, som viser når prosedyren er fullført.

Du må oppgi variabler før du legger til kommandoer. Den første kommandoen i hovedprosedyren kjøres, og deretter kjøres etterfølgende kommandoer i den rekkefølgen de vises i skriptet.

Skriptet avsluttes når du har kommet til slutten av hovedprosedyren.

### 3.0 Variabler

Skript kan inneholde variabler. Navn på variabler må begynne med en bokstav eller et understrekingstegn ( \_ ), og kan inneholde både store og små bokstaver samt tall og understrekingstegn i vilkårlig rekkefølge. Du kan ikke bruke reserverte ord som variabelnavn. Hvis du vil ha mer informasjon, se listen over reserverte ord på slutten av dette dokumentet.

Du må deklare variabler før du bruker dem. Når du deklarerer en variabel, må du også definere hvilken type det er. Noen variabler kan bare inneholde samme type verdier. Nedenfor ser du tre typer variabler som støttes:

<u>Type</u>	Integer
<u>Beskrivelse</u>	Et negativt eller positivt tall, for eksempel 7, -12 eller 5698.

<u>Type</u>	String
<u>Beskrivelse</u>	Flere tegn som står i doble anførselstegn, for eksempel "Hallo til verden!" eller "Skriv inn et passord:".

<u>Type</u>	Boolean
<u>Beskrivelse</u>	En logisk boolsk verdi som er SANN eller USANN.

Variabler er tilordnet verdier ved hjelp av tilordningsinstruksjonen nedenfor:

```
variable = uttrykk
```

Variabelen får det beregnede uttrykket.

For eksempel:

```
integer count = 5
integer timeout = (4 * 3)
integer i

boolean bDone = FALSE

string szIP = (getip 2)

set ipaddr szIP
```

Eksemplet ovenfor bruker skriptkommandoer som ikke støttes av Shiva.

### 3.1 Systemvariabler

Systemvariabler angis av skriptkommandoer eller bestemmes av informasjonen du skriver inn når du oppretter en tilkobling. Systemvariabler er skrivebeskyttet. De kan altså ikke endres i skriptet. Systemvariablene er:

<u>Navn</u>	\$USERID
<u>Type</u>	Streng
<u>Beskrivelse</u>	Brukeridentifikasjonen for gjeldende tilkobling. Denne variabelen er verdien til brukernavnet som er angitt i dialogboksen Koble til i tilkoblingsprogrammet.

<u>Navn</u>	\$PASSWORD_
<u>Type</u>	Streng

Beskrivelse Passordet for den gjeldende tilkoblingen. Denne variabelen er verdien til brukernavnet som er angitt i dialogboksen Koble til i tilkoblingsprogrammet.

Navn \$SUCCESS

Type Boolsk

Beskrivelse Denne variabelen er angitt av visse kommandoer for å vise om kommandoen lyktes eller ikke. Et skript kan ta beslutninger basert på verdien til denne variabelen.

Navn \$FAILURE

Type Boolsk

Beskrivelse Denne variabelen er angitt av visse kommandoer for å vise om kommandoen mislyktes eller ikke. Et skript kan ta beslutninger basert på verdien til denne variabelen.

Disse variablene kan brukes når samme type uttrykk brukes. For eksempel:

```
transmit $USERID
```

er en gyldig kommando fordi \$USERID er en strengvariabel.

## 4.0 Strenglitteraler

Skript for Ekstern pålogging støtter avbruddssekvenser og oversetting av cirkumflekstegn som beskrevet nedenfor.

Strenglitteral Beskrivelse  
Strenglitteral *^tegn*  
Beskrivelse Oversetting av cirkumflekstegn

Hvis *tegn* er en verdi mellom "@" og "\_", er tegnsekvensen oversatt til en enbitsverdi mellom 0 og 31. For eksempel blir ^M konvertert til linjeskift.

Hvis *tegn* er en verdi mellom "@" og "\_", er tegnsekvensen oversatt til en enbitsverdi mellom 0 og 31. ^M er for eksempel konvertert til linjeskift.

Hvis *tegn* er en verdi mellom "a" og "z", er tegnsekvensen oversatt til en enbitsverdi mellom 1 og 26.

Hvis *tegn* er en annen verdi, behandles ikke tegnsekvensen på noen spesiell måte.

Strenglitteral <cr>  
Beskrivelse Linjeskift

Strenglitteral <lf>  
Beskrivelse Linjemating

Strenglitteral \"  
Beskrivelse Doble anførselstegn

Strenglitteral \^  
Beskrivelse Cirkumflekstegn

Strenglitteral \<  
Beskrivelse Enkelt "mindre enn"-tegn (<)

Strenglitteral \\  
Beskrivelse Omvendt skråstrek

For eksempel:

```
transmit "^M"  
transmit "Ole^M"  
transmit "<cr><lf>"  
waitfor "<cr><lf>"
```

## 5.0 Uttrykk

Et uttrykk er en kombinasjon av operatører og argumenter som til sammen utgjør en helhet. Uttrykk kan brukes som verdier i alle slags kommandoer.

I uttrykk kan alle slags variabler, eller heltall, strenger eller boolske verdier settes sammen med alle monooperatører og binære operatører i tabellene nedenfor. Alle monooperatører prioriteres høyest. Prioriteten til binære operatører er angitt av deres plassering i tabellen.

Mono operatører er:

Operator -  
Operasjonstype Enkel minus

Operator !  
Operasjonstype Komplement

De binære operatorene står i prioriteringsrekkefølge i tabellen nedenfor. Operatører med høyest prioritet står først:

Operatører \* /  
Operasjonstype Multiplikasjon/divisjon  
Restriksjonstype Heltall

Operatører + -  
Operasjonstype Addisjon/subtraksjon av heltall  
Restriksjonstype Strenger (bare +)

Operatører < > <= >=  
Operasjonstype Sammenligning  
Restriksjonstype Heltall

Operatører == !=  
Operasjonstype Likhet  
Restriksjonstype Heltall, strenger, boolsk type

Operatører and  
Operasjonstype Logisk AND  
Restriksjonstype Boolsk

Operatører or  
Operasjonstype Logisk OR  
Restriksjonstype Boolsk

For eksempel:

```
count = 3 + 5 * 40
transmit "Velkommen" + " hit"
delay 24 / (7 - 1)
```

## 6.0 Kommentarer

All tekst på en linje som står etter et semikolon, ignoreres.

For eksempel:

```
; dette er en kommentar

transmit "Velkommen"           ; overfører strengen "Velkommen"
```

## 7.0 Funksjoner

Funksjoner angir skriptets struktur. I motsetning til kommandoer utfører de ikke handlinger. Funksjonene er oppført nedenfor.

### **proc** *name*

Angir begynnelsen på en prosedyre. Alle skript må ha en hovedprosedyre (**proc main**). Skriptkjøring starter med hovedprosedyren og stopper ved slutten av hovedprosedyren.

### **endproc**

Angir slutten av prosedyren. Når skriptet kjøres til **endproc**-instruksjonen for hovedprosedyren, starter skript for Ekstern pålogging PPP eller SLIP.

### **integer** *navn* [ = *verdi* ]

Angir en heltallvariabel. Du kan bruke alle talluttrykk eller variabler til å initialisere variabelen.

### **string** *navn* [ = *verdi* ]

Angir en strengvariabel. Du kan bruke alle strenglitteraler eller variabler til å initialisere variabelen.

### **boolean** *navn* [ = *verdi* ]

Angir en boolsk variabel. Du kan bruke alle boolske uttrykk eller variabler til å initialisere variabelen.

## 8.0 Kommandoer

Den første implementeringen av skriptspråket for Oppringing for Internet Explorer er delsett av kommandospråket for skript for Ekstern pålogging som det er angitt i Microsoft Windows(R) 95.

Alle kommandoene er reserverte ord. Dette betyr at du ikke kan angi variabler som har samme navn som kommandoene. Kommandoene står oppført nedenfor:

### **delay** *nSekunder*

Stopper i det antall sekunder som er angitt i *nSekunder* før neste kommando i skriptet kjøres.



For eksempel:

```
delay 2      ; venter 2 sekunder
delay x * 3  ; venter x * 3 sekunder
```

### **goto etikett**

Hopper til det stedet i skriptet som er angitt ved *etikett* og fortsetter å kjøre de neste kommandoene.

For eksempel:

```
waitfor "Prompt>" until 10
if !$SUCCESS then
    goto BailOut ; hopper til BailOut og utfører
                  ; kommandoene som følger
endif

transmit "bbs^M"
goto End

BailOut:
transmit "^M"
```

### **halt**

Stopper skriptet. Denne kommandoen fjerner ikke terminalvinduet. Du må velge Fortsett for å opprette tilkoblingen. Du kan ikke starte skriptet på nytt.

### **if betingelse then**

*kommandoer*

**endif**

Kjører flere *kommandoer* hvis *betingelsen* er SANN.

For eksempel:

```
if $USERID == "John" then
    transmit "Johnny^M"
endif
```

### **label :**

Angir det stedet i skriptet det skal hoppes til. En etikett må være et entydig navn og følge navnekonvensjonene for variabler.

### **transmit streng [ , raw ]**

Sender tegnene som er angitt med *streng*, til den eksterne datamaskinen.

Den eksterne datamaskinen gjenkjenner avbruddssekvenser og oversetting av

cirkumflekstegn hvis du ikke inkluderer **raw**-parameteren i kommandoen. **Raw**-parameteren er nyttig når du overfører systemvariablene \$USERID og \$PASSWORD og brukernavnet og passordet inneholder tegnsekvenser som ville blitt tolket som cirkumflekstegn eller avbruddssekvenser uten **raw**-parameteren.

For eksempel:

```
transmit "slip" + "^M"  
transmit $USERID, raw
```

**waitfor** *streng* [ , **matchcase** ] [ **then etikett**  
{ , *streng* [ , **matchcase** ] **then etikett** } ]  
[ **until tid** ]

Venter til datamaskinen mottar én eller flere av de angitte strengene fra den eksterne datamaskinen. *Streng*-parameteren skiller ikke mellom små og store bokstaver hvis du ikke inkluderer parameteren **matchcase**.

Hvis en samsvarende streng er mottatt, og etikettparameteren **then** brukes, hopper denne kommandoen til det stedet i skriptfilen som er angitt med *etikett*.

Den valgfrie parameteren **until tid** angir maksimalt antall sekunder som datamaskinen venter på å motta strengen før den kjører neste kommando. Uten denne parameteren venter datamaskinen evig.

Hvis datamaskinen mottar en av de angitte strengene, settes systemvariabelen \$SUCCESS til SANN. Den angis derimot til USANN hvis antallet sekunder som er angitt ved *tid*, forløper før strengen er mottatt.

For eksempel:

```
waitfor "Brukernavn:"  
  
waitfor "Passord?", matchcase  
  
waitfor "prompt>" until 10  
  
waitfor  
    "Brukernavn:" then DoLogin,  
    "Passord:"    then DoPassword,  
    "BBS:"        then DoBBS,  
    "Annet:"      then DoOther  
until 10
```

Kommandoen nedenfor er et Shiva-tillegg til Microsofts skriptkommandosett.

**getinput** "*Vist streng* " *szInndata*

Ber brukeren om å skrive inn informasjon mens skriptet kjøres.

For eksempel:

```
Getinput "Systempassord" szSystemPassword
```

## 9.0 Reserverte ord

Ordene nedenfor er reserverte og kan ikke brukes som variabelnavn:

and	boolean	databits	delay
do	endif	endproc	endwhile
even	FALSE	getip	goto
halt	if	integer	ipaddr
keyboard	mark	matchcase	none
odd	off	on	or
parity	port	proc	raw
screen	set	space	stopbits
string	then	transmit	TRUE
until	waitfor	while	

## 10.0 Shiva-kommandoord som ikke støttes av Scripter

getip  
port databits  
port parity  
port stopbits  
set screen keyboard  
ipaddr  
while/endwhile