

Puesta en marcha y configuración del servicio Monser

BASE100

BASE 100, S.A.
www.base100.com

Índice

1.	INTRODUCCIÓN	3
2.	PUESTA EN MARCHA	4
3.	PROTOCOLO DE COMUNICACIONES.....	5
3.1	COMANDO DE VALIDACIÓN	5
3.2	COMANDOS DISPONIBLES	5
3.3	CÓDIGOS DE RESPUESTA	7

1. Introducción

MONSER es un servicio TCP/IP situado en el puerto **1099** del servidor CTSQL. Es accesible abriendo una conexión remota de cualquier máquina que soporte TCP/IP.

Para conectarnos a MONSER disponemos de varias opciones:

1. Utilizar la herramienta monitora de Cosmos o MultiBase:

monitor.exe para Windows.

monitor para plataformas Unix.

2. Abrir un **STREAM Remoto** de MultiBase o Cosmos al puerto 1099 (servicio MONSER).
3. Conectarnos directamente utilizando la herramienta **TELNET**.

Por ejemplo:

```
C:\> Telnet foo.es monser
```

(Tenemos que tener la entrada "monser" en el fichero "services".)

O bien directamente

```
C:\> Telnet foo.es 1099
```

2. Puesta en marcha

Unix

Para poner en marcha el servicio de monitorización remota tenemos que ejecutar el programa *monser*.

Si el servicio se ha lanzado correctamente nos devolverá el PID y el puerto de conexión que utiliza.

Para cerrar el servicio “*mataremos*” el proceso haciendo un *Kill* del PID.

Windows

Para poner en marcha el servicio de monitorización remota tenemos que dar de alta un servicio LSTMonser.

Este proceso se realiza desde la herramienta de instalación de servicios **mwlisten**:

```
mwlisten -install monser
```

A continuación deberemos configurarlo para que se active de forma automática, asociándole un usuario, que será el mismo que utilizemos en el servicio CTSQL. Para ello ejecutaremos:

```
mwlisten -d monser
```

Para borrar el servicio ejecutaremos:

```
mwlisten -remove monser
```

Windows y Unix

Para conectarnos desde una estación cliente tendremos que validarnos con un usuario válido en la máquina Unix. Debe de estar dado de alta en el fichero de configuración *monitor.ini* del servidor, en la sección GENERAL, cláusula RemoteUser.

3. Protocolo de comunicaciones

3.1 Comando de validación

Un vez conectados al servicio tendremos que validarnos para poder hacer uso de los comandos de administración. Para validarnos haremos uso del comando VALIDATE:

```
VALIDATE user passwd \r\n
```

Tendremos que introducir un nombre de usuario y una palabra clave (password) válidos en la máquina servidora y que estén contemplados en el fichero de configuración *monitor.ini*, en la sección GENERAL, cláusula RemoteUser:

```
[General]
...
RemoteUser = ctl, dctl
...
```

Una vez validados nos aparecerá la respuesta del servidor:

```
200 User Authorized. \r\n
```

Si la validación es errónea el servidor denegará el servicio, cerrando la conexión:

```
-201 User not authorized.
```

NOTA: Si estamos utilizando la herramienta TELNET no aparecerán los caracteres `\r\n` [chr(13) + chr(10)], pero desde una aplicación programada por nosotros sí deberemos tenerlos en cuenta.

3.2 Comandos disponibles

Una vez validados para el servidor podremos ejecutar una serie de comandos de administración para interrogar sobre las sesiones activas y actuar sobre ellas. Los comandos disponibles se pueden consultar ejecutando el comando HELP:

Por ejemplo:

```
VALIDATE user passwd \r\n
200 User Authorized. \r\n
HELP
270+ Help command.
270 HELP - Monitor Remote Commands.
270 -----
270 VALIDATE user passwd
270 STATUS - START - STOP
270 SESSIONS - LIST - GET num [param]
270 WHOAMI - KILL nun
```

Como podemos observar en el ejemplo anterior, todas las líneas de respuesta a los comandos van precedidas por un número de 3 cifras. La primera línea indica si va sola o no, ya que tendrá un signo “+” detrás del número en caso de que le sigan más líneas. Para detectar que ya no sigue más información utilizaremos la línea con el punto, es decir, nos encontraremos con líneas de información hasta “\r\n.\r\n”.

Las respuestas de error devuelven un número negativo de tres cifras:

```
LISTT
-501 Unknown Command (listt).
GET 0
-231 No Session Info.
SESSIONS
230 0 active sessions.
```

La lista de comandos soportados es la siguiente:

VALIDATE user passwd

Permite validarnos con el servicio y tener acceso a los demás comandos. Si la validación es errónea cierra la conexión.

STATUS

Informa del estado del servidor, si está START o STOP, es decir, si podemos abrir conexiones o no.

START

Comando para activar el servidor.

STOP

Comando para cerrar el servidor. No se puede cerrar si hay sesiones conectadas.

SESSIONS

Indica el número de sesiones.

LIST

Lista las sesiones activas.

GET num [param]

Proporciona información de una sesión, pudiendo especificar la información que deseemos.

WHOAMI

Permite identificar sesiones provenientes de la máquina desde la que nos hemos conectado.

KILL nun

Permite “matar” una sesión remota.

QUIT

Desconecta la conexión al *monser*.

3.3 Códigos de respuesta

Códigos de respuesta simples

```
200 User Authorized. (VALIDATE)
210 Monitor STARTED. (START)
220 Monitor STOPED. (STOP)
230 %d active sessions. (SESSIONS)
240 %s -> información básica de sesión (LIST)
270 %s -> línea de ayuda. (HELP)
280 %s -> línea de información de sesión. (GET)
290 %d killed. (KILL)
300 %d - %s %s ->línea de respuesta a WHOAMI ( id - clienthostname(ip)
user )
310 bye. (QUIT)
```

Códigos de respuesta complejos (respuesta de más de una línea)

```
240+ %d active sessions. -> acompañado de códigos 240 de sesiones
(LIST)
270+ Help command. (HELP)
280+ Session Info. primera línea de GET.
300+ Session List. primera línea de WHOAMI, si hay más de una.
```

Códigos de respuesta de errores

```
-201 User Not Authorized.
-202 Validate first please. "VALIDATE user passwd".
-211 Unable to START.
-221 Unable to STOP.
-231 No Session Info.
-291 Unable to kill %d Session.
-301 No session matches to you.
-311 bye unable disconnect.
-501 Unknown Command (%s).
-502 %s. ->otros posibles errores
```

NOTA: %s: texto; %d: caracteres con contenido numérico.