加载对应版本的 SOS。

如果被调试程序使用的是.NET 1.0 运行时，那么它的 SOS 模块位于 WinDBG 程序目录中的 clr10 子目录下。具体来说，可以使用如下命令。

```
.load clr10\sos.dll
```

从.NET Framework 1.1 开始，.NET 运行时中就包含了与其配套的 SOS.dll，比如下面便是两个与运行时放在一起的 sos.dll。

```
C:\WINDOWS\Microsoft.NET\Framework\v1.1.4322\sos.dll
C:\WINDOWS\Microsoft.NET\Framework\v2.0.50727\sos.dll
```

如果使用.load 命令加载上面的 sos.dll，路径很长，不容易记住。这时可以通过观察运行时核心模块 mscorwks.dll 的位置来得到 CLR 的位置，然后复制并粘贴。

```
0:000> lmvm mscorwks
start    end        module name
791b0000 79419000   mscorwks   (pdb symbols)          C:\WINDOWS\Microsoft.NET\
Framework\v1.1.4322\mscorwks.dll
```

但是这样做有点麻烦，更简洁的方法是使用.loadby 命令，即：

```
.loadby sos mscorwks
```

意思是加载与 mscorwks 模块相同位置的 sos 扩展模块，如果执行时没有任何提示信息，那么便执行成功了。

无论使用以上哪种方法加载，加载后，都可以使用.chain 命令来观察已经加载的扩展模块。

```
0:000> .chain
Extension DLL chain:
C:\WINDOWS\Microsoft.NET\Framework\v1.1.4322\sos: API 1.0.0, built Thu Jul 15
10:46:07 2004 [path: C:\WINDOWS\Microsoft.NET\Framework\v1.1.4322\sos.dll]
[以上信息做过删减]
```

加载成功后，可以执行!help 命令来显示 SOS 的帮助信息。对于 2.0 或者更高版本的 SOS，可以在!help 后面加上要了解的命令来得到关于某个命令的详细解释。

下面再讲另一种情况，也就是在使用 WinDBG 打开 EXE 时该如何加载 SOS 扩展。因为当被调试进程加载了 CLR 后，SOS 才能工作和有意义，所以应该在 CLR 的核心模块 MSCORWKS.dll 或者 clr.dll 加载后，再加载 SOS。那么如何知道 MSCORWKS/clr 何时被加载呢？这可以通过定制 WinDBG 模块加载事件的方式来实现，执行 sxe 命令。

```
sxe ld:mscorwks.dll 或者 sxe ld:clr.dll
```

上述命令的含义是让 WinDBG 收到被调试进程加载 mscorwks.dll/clr.dll 模块的事件时，中断下来。中断下来后，便可以执行.loadby sos mscorwks 或者.loadby sos clr 命令了。

也可以使用下面这条命令把上面所说的两个动作合在一起。

```
sxe -c ".loadby sos mscorwks;g" ld mscorwks.dll
```

其含义是当收到加载 mscorwks.dll 的事件后执行-c 后面跟的命令，也就是双引号内的内容——先加载 sos，然后恢复目标继续运行（g）。

7.6.2 设置断点

可以使用 SOS 中的 BPMD 命令来针对托管代码设置断点，它有两种格式。先来看第一种。