

Congelamientos por software

Cuando las aplicaciones generan inestabilidad

YA DETALLAMOS LAS CAUSAS RELACIONADAS CON CUELGUES PRODUCIDOS POR EL HARDWARE. AHORA ANALIZAREMOS QUÉ OCURRE CUANDO UN PROGRAMA PROVOCA ESTE PROBLEMA.

Las causas de cuelgues por software pueden ser muchísimas y, a veces, estar producidas por el mal funcionamiento de un dispositivo de hard. Ante esta situación, el mismo sistema operativo presenta indicaciones que contienen toda la información sobre lo ocurrido. En este apartado detallaremos los errores producidos por el **volcado de memoria** (*memory dump*) que genera Windows. Generalmente, se manifiesta con la caída de sistema o la **pantalla azul**.

La pantalla azul de la muerte (Blue Screen Of Death)

Quienes alguna vez han utilizado un sistema operativo de Microsoft habrán tenido la oportunidad de conocer el Error de detención (*Stop Error*) de Windows, también conocido como La pantalla azul de la muerte. Se trata de un mensaje de error que se produce cuando el sistema **no puede recuperarse de un error crítico**. A pesar de que nunca nos fijamos en esta pantalla debido a que presenta gran cantidad de números y códigos que muchas veces no entendemos, prestarle atención es una de las formas más rápidas de llegar a diagnosticar el motivo de un congelamiento.

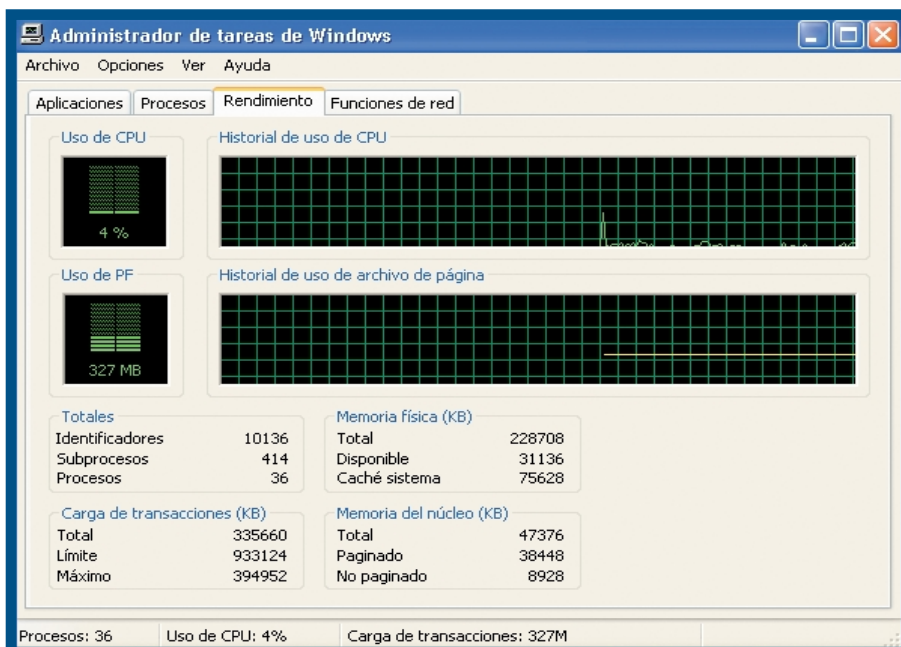
FALLA EVIDENTE

Si al momento de ejecutar un determinado programa, el sistema se congela y no responde ante ninguna orden, estamos frente a una falla sistemática del software en cuestión, y no habrá más remedio que desinstalarlo, reiniciar la PC y volver a instalarlo de una manera limpia, sin errores. Así no quedarán dudas acerca del funcionamiento de la aplicación.

Detectar el programa que provoca cuelgues

El usuario de PC suele instalar software ignorando ciertos aspectos que pueden ocasionar problemas en el sistema. En general, no se tienen en cuenta el espacio disponible y los requisitos ideales para el funcionamiento del programa. Además, suele suceder que la instalación no se realiza de manera limpia –es decir, sin errores– y, en consecuencia, comienza a producirse cierta inestabilidad. Por este motivo, aquí veremos una de las opciones disponibles para descubrir cuál es el programa que está generando el conflicto, y proceder a desinstalarlo.

En el momento en que la PC se congela, debemos acudir al Administrador de tareas de Windows para verificar cuál de todas las aplicaciones que se están ejecutando es la causa. Luego tendremos que obligarla a finalizar su



EN ESTE ADMINISTRADOR ENCONTRAREMOS OPCIONES PARA VERIFICAR LOS PROCESOS DEL SISTEMA, EL RENDIMIENTO DEL EQUIPO Y LAS FUNCIONES DE RED.

tarea y, así, darle al sistema el respiro necesario para que pueda volver a funcionar.

Instalaciones de software

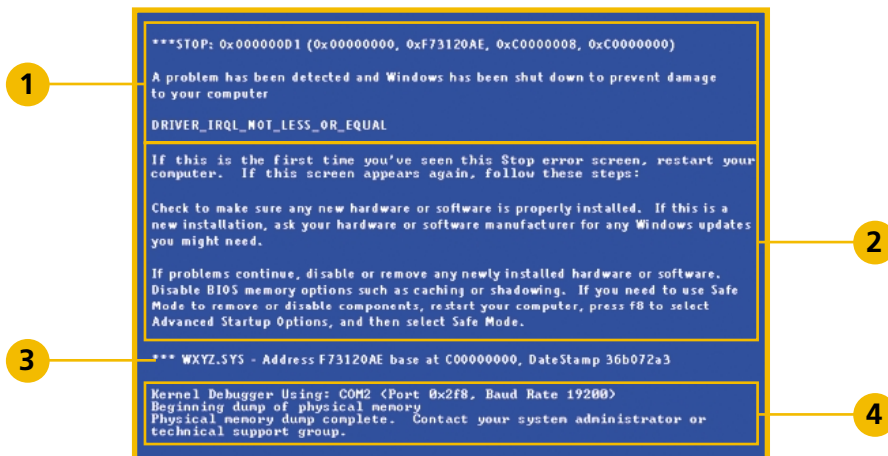
Muchos de los usuarios menos experimentados muestran un comportamiento compulsivo: instalar cualquier cosa que caiga en sus manos sin importar su procedencia. Ya sea un parche del sistema, un software, un emulador etc., instalar software de ese modo sólo puede llevar a dañar el S.O. y, por lo tanto, a producir cuelgues. Es decir que las instalaciones de drivers, parches o programas deben realizarse con precaución y **de a una por vez**. Esto se debe a que muchos programas que se instalan en una PC suelen incluir miniaplicaciones que pueden actuar de forma maliciosa –como spyware o virus–, seguramente, provocarán inestabilidad en el sistema o, peor aún, harán que terminemos perdiendo datos de suma importancia, como documentos personales. Luego de cada instalación, es recomen-

dable probar la PC para asegurarse de que el sistema no haya sufrido daños y todo siga funcionando como corresponde. También es fundamental que, a la hora de desinstalar una aplicación, verifiquemos que el proceso se realice correctamente y que no queden residuos. Para controlar esto, podemos recurrir a algún tipo de software que permita realizar una búsqueda minuciosa en el Registro de Windows, con el fin de eliminar todas aquellas entradas que no sean necesarias. Este tipo de trabajo se conoce como limpieza del sistema operativo. Cuando hablamos de limpieza en este caso, nos estamos refiriendo, justamente, a mantener el S.O. libre de virus y spyware. y a procurar que el Registro goce de buena salud. Para esto, necesitamos diferentes aplicaciones, como **AVG** (antivirus), **Ad-Aware** (antispyware) y **RegCleaner** (limpiador de Registro). De más está decir que todos estos programas requieren actualizaciones frecuentes.

DAÑOS AL SISTEMA

Si los cuelgues o reinicios coinciden con alguna acción importante o vital del S.O. –como durante una instalación o desinstalación de un programa–, seguramente terminaremos por “romper” la estructura del sistema, lo cual hará que los congelamientos o reinicios sean aún más frecuentes. Esto se convertirá en una gran bola de nieve imparable, y lo más probable es que debamos formatear el disco y reinstalar el S.O. Una opción valedera es ejecutar la herramienta de restauración (como Restaurar sistema, en el caso de Windows XP), que intentará volver a la última configuración estable conocida. En este proceso, también podemos elegir diferentes fechas de restauración y, así, decidir a qué día volver. Luego, la PC iniciará la restauración del Registro a la fecha elegida y reiniciará. Si todo se hizo correctamente, no tendremos que formatear el disco. En la página 64 analizamos en detalle este tema.

LA PANTALLA AZUL DE WINDOWS



- 1 Bugcheck:** Contiene el número del error en formato hexadecimal (números y letras), información en formato texto que indica por qué el sistema se detuvo.
- 2 Acción recomendada:** Es bastante genérica e incluye información acerca de cómo deberíamos proceder para solucionar el inconveniente. Siempre intenta lo mismo para casi todos los errores: “reinicie su equipo”.

- 3 Información de driver:** Es, tal vez, la parte más importante. Si algún driver está relacionado con el error que detuvo el sistema, éste figurará en esta sección.
- 4 Puerto de depuración e información de estado:** Al detenerse el sistema, Windows intentará enviar información a un archivo en disco o a alguno de los puertos COM. En esta zona se verá la acción tomada.

Programas conflictivos

La importancia del Administrador de tareas

CUANDO UNA APLICACIÓN GENERA CUELQUES, NADA MEJOR QUE RECURRIR AL ADMINISTRADOR DE TAREAS, QUE PERMITIRÁ CONOCER SUS PROCESOS Y SOLUCIONAR EL PROBLEMA.

El profesional del hardware debe entender cada proceso y cada programa que se está ejecutando en el sistema, con el fin de reconocer posibles inconvenientes, como aplicaciones que consumen demasiados recursos y pueden causar problemas. Para localizar la lista de los procesos, debemos acceder al Administrador de tareas de Windows (presionando <Ctrl> + <Alt> + <Supr>, y luego, ir a la solapa [Procesos]). Allí encontraremos información dividida en cuatro columnas: procesos que se están ejecutando, nombre de usuario de la sesión, recursos de la CPU y recursos que ese proceso está tomando de la memoria.

Cada proceso en detalle

Es poco lo que se sabe acerca de los procesos que podemos observar en el Administrador de tareas, y son muchos los mitos sobre los cuales se basa la información que se ofrece en los sitios de Internet con respecto a este tema. Por eso, al hablar sobre los procesos en ejecución, suele hacerse referencia a una "zona os-

cura". En estas páginas, además de iluminar esta zona, veremos cómo diferenciar los procesos verdaderos de los falsos, provocados por programas maliciosos.

→ **SVCHOST.EXE:** Microsoft Windows utiliza esta aplicación para encargarse de los procesos ejecutados desde DLLs. Cabe destacar que no se recomienda detenerlo, ya que puede ser necesario para el buen funcionamiento del sistema.

→ **TASKMGR.EXE:** Es el **Manejador de Tareas de Windows**, cuya función es mostrar los procesos que están en ejecución. Presionando <Ctrl> + <Alt> + <Supr>, el archivo taskmgr.exe abre la ventana y nos muestra la lista de procesos.

→ **SPOOLS.V.EXE:** El sistema operativo delega a esta aplicación tareas relacionadas con los procesos que son ejecutados por las impresoras locales.



EL ADMINISTRADOR DE TAREAS SUELE RESULTAR UNA HERRAMIENTA FUNDAMENTAL AL MOMENTO DE ANALIZAR COMPORTAMIENTOS ERRÓNEOS DE SOFTWARE. ADEMÁS, SI UN PROCESO ESTÁ UTILIZANDO DEMASIADOS RECURSOS, PODEMOS FINALIZARLO EN EL MOMENTO.



→ **WDFMGR.EXE:** Está íntimamente relacionado con el **Reproductor de Windows Media**, y su función es disminuir ciertos problemas de incompatibilidad.

→ **ISASS.EXE:** Genera los procesos de autenticación de usuarios para **Winlogon**. Es utilizado por paquetes de autenticación, tales como Msgina.dll. Si el procedimiento tiene éxito, Isass.exe genera los tokens de acceso (identificadores de seguridad que establecen los privilegios y derechos) para el usuario que son utilizados para lanzar el shell inicial (programa de atención al usuario). Los otros procesos que el usuario inicia heredan estos tokens.

→ **SERVICES.EXE:** Windows gestiona la operación de iniciar y detener servicios a través de esta aplicación. Además, es el encargado de controlar los servicios automáticos que se cargan en el inicio.

→ **IEXPLORE.EXE:** Se ocupa de gestionar los programas de Windows, como el shell gráfico, incluyendo el menú de Inicio, la barra de tareas, el Escritorio y el explorador de archivos. Al detener este proceso, se pierde la interfaz gráfica de Windows, con lo cual se inhabilita la interacción con el sistema.

→ **WINLOGON.EXE:** Gestiona los procedimientos de **login** (conexión) y **logout** (desconexión) al sistema. Cabe destacar que es un proceso elemental del sistema y no debe ser finalizado.

→ **CSRSS.EXE:** Es el ejecutable principal para Microsoft Client / Server Runtime Server Subsystem. Es decir que este proceso maneja la mayoría de los comandos gráficos. Además, es fundamental para la estabilidad y la seguridad, por lo que no debería ser detenido.

→ **SRVANY.EXE:** Es una aplicación que permite a un ejecutable correr como un servicio. No es un proceso esencial del sistema, pero no debería ser detenido a menos que causara problemas.

→ **MDM.EXE:** Está asociado con el sistema de **"debugeo"** (descubrir y solu-



➔ **CONOCIENDO LOS PROCESOS DETALLADOS EN ESTE APARTADO, EL EXPERTO PODRÁ REALIZAR DIAGNÓSTICOS QUE LE PERMITIRÁN SOLUCIONAR EL PROBLEMA.**

cionar bugs) de Windows. Permite al usuario corregir errores de Internet Explorer mediante el uso de una herramienta de **scripting**.

→ **ALG.EXE:** Windows administra este proceso para el servicio de conexión compartida a Internet y el firewall. Es importante para la estabilidad y la seguridad del sistema, y no debería ser detenido, a no ser que estas funcionalidades de Windows sean reemplazadas por otros programas.

→ **WZQKPIK.EXE:** Es la abreviación de WinZip QuickPick, que hace referencia al icono de WinZip alojado en la barra de tareas, si es que está instalada.

→ **SMSS.EXE:** Es un proceso de Windows denominado **Session Manager SubSystem** (subsistema de manejo de sesiones) y es responsable de gestionar las sesiones en el S.O.

→ **CTFMON.EXE:** Corresponde a la suite Microsoft Office; activa el *Alternative User Input Text Input Processor* (TIP) y la barra de lenguaje de Office XP.

→ **SYSTRAY.EXE:** Es un proceso de background que muestra información tal como la fecha y la hora del sistema.

→ **WINOA386.MOD:** Este proceso genera una consola de MS-DOS dentro del entorno Windows de 32 bits y posibilita el acceso a líneas de comandos del sistema operativo del disco (DOS).

Dependiendo de los programas instalados, es posible encontrar otros procesos, como el del programa antivirus, el firewall, el soft de mensajería, etc. Con lo ya visto, el experto podrá realizar diagnósticos que permitirán solucionar el problema. Si al ejecutar el programa vuelve a fallar, será conveniente desinstalarlo.

VIRUS OPTIX

En ocasiones, el archivo Isass.exe es generado por el virus Optix.Pro, que tiene la capacidad de deshabilitar firewalls y otras protecciones locales del sistema; también provee un backdoor para futuras intrusiones. Este proceso es un riesgo de seguridad y debe ser eliminado del sistema.

Cuelgues por virus

Detección de procesos generados por malware

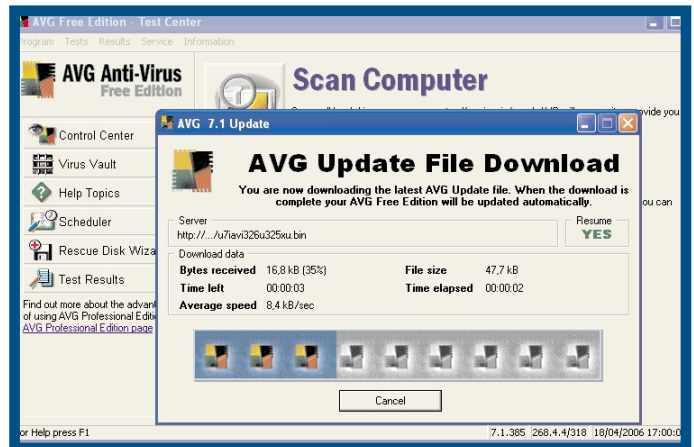
En el apartado anterior, hemos detallado los procesos que podemos observar desde el Administrador de tareas de Windows. Algunos de ellos son ordinarios y parte elemental del sistema; sin embargo, podemos encontrar otros extraordinarios que pueden resultar riesgos de seguridad, dado que son parte de un virus o son generados por ellos. Por eso, es fundamental tener en claro cuáles son los procesos normales y aquellos que está ejecutando un software malicioso.

- **HBINST.EXE:** Su función es monitorear hábitos de uso en Internet; es decir, rastrear las páginas más visitadas para un posterior análisis de mercado.
- **IEXPLORER.EXE:** Puede llegar a ser una variación de Rapid Blaster, cuya función es mostrar publicidad en el navegador. No hay que confundirlo con IEXPLORE.EXE, que es el proceso del navegador de Microsoft.
- **JDBGMRG.EXE:** Si encontramos este proceso ejecutándose, deberemos eliminarlo inmediatamente, ya que se trata del virus **TROJ_DASMIN.B**.
- **START.EXE:** Al igual que tantos otros, este proceso controla hábitos de navegación y ejecuta ventanas con publicidad en el navegador.

PARA TENER EN CUENTA

Hay que tener en cuenta que algunos virus generan archivos con nombres similares:

- **SCVHOST.EXE:** Es producto de un virus denominado **W32/Agobot-S**. Estamos hablando de un gusano y troyano backdoor de IRC (*Internet Relay Chat*, protocolo de comunicación en tiempo real) que se copia a sí mismo para aprovechar los recursos compartidos.
- **SVCHOSTS.EXE:** Es una instalación del virus **Sdbot-N**. En este caso, se trata también de un troyano backdoor que permite a un usuario remoto controlar nuestra máquina a través de IRC. Se ejecuta en background; trata de conectarse a un canal específico de un servidor de IRC y, luego, queda a la escucha de ciertos comandos para llevar a cabo sus acciones.
- **SVSHOST.EXE:** Es instalado por el virus **Worm.P2P.Spybot.gen**. Se trata de un gusano que se propaga generalmente a través del P2P (Kazaa) y del correo electrónico. Esta versión también posee características de troyano, ya que utiliza los puertos traseros para permitirle al intruso dominar el equipo.



- **DCOMX.EXE:** Es parte de un virus denominado **CIREBOT** y deberá ser eliminado del sistema cuanto antes.
- **FSG.EXE:** Se trata de un spyware que pone en riesgo la seguridad de los datos, producto de aplicaciones freeware o shareware.
- **SHOWBEHIND.EXE:** Este programa monitorea los hábitos de navegación y ejecuta ventanas en el navegador.
- **MSVXD.EXE:** Si este proceso está en ejecución, se debe a que el sistema está infectado por el virus **W32/Datom-A** y debe ser eliminado.
- **MAPISVC32.EXE:** Es producto de un virus conocido como **KX** y deberá ser eliminado del sistema cuanto antes.
- **PGMONITR.EXE:** Este spyware, instalado generalmente por programas como Kazaa, no sólo muestra publicidades indeseadas y recolecta información personal, sino que, además, ocasiona inconvenientes con la conexión a Internet y puede dificultar el arranque de la PC.
- **ADAWARE.EXE:** Es probable que si encontramos este proceso en ejecución, sea producto de un virus conocido como **Rapid Blaster**, que tiene la habilidad de copiarse a sí mismo y propagarse en otros directorios.
- **LOADER.EXE:** También conocido como **Hijacker**, este programa cambia la página de inicio configurada en el navegador, además de otros parámetros personales.
- **ARR.EXE:** Hay que prestar mucha atención a este proceso, ya que su función es discar un número telefónico, generalmente de otro país, para acceder a diferentes tipos de material no deseado. Cabe destacar que el usuario deberá pagar la llamada de larga distancia.

Los procesos maliciosos detallados son sólo algunos de los apuntados en la lista y sirven de ejemplo para saber cuál es el efecto que causan en el sistema. Sin embargo, cualquiera de los comentados en esta página evidencia la ejecución de programas maliciosos en el sistema y deberá ser eliminado de inmediato; recordemos que estos procesos corresponden a software dañino (virus, gusanos y spyware) y pueden ser detectados por medio de los antivirus y antispyware más reconocidos.

Cuelgues atribuidos al S.O.

Cuando el sistema operativo es el responsable

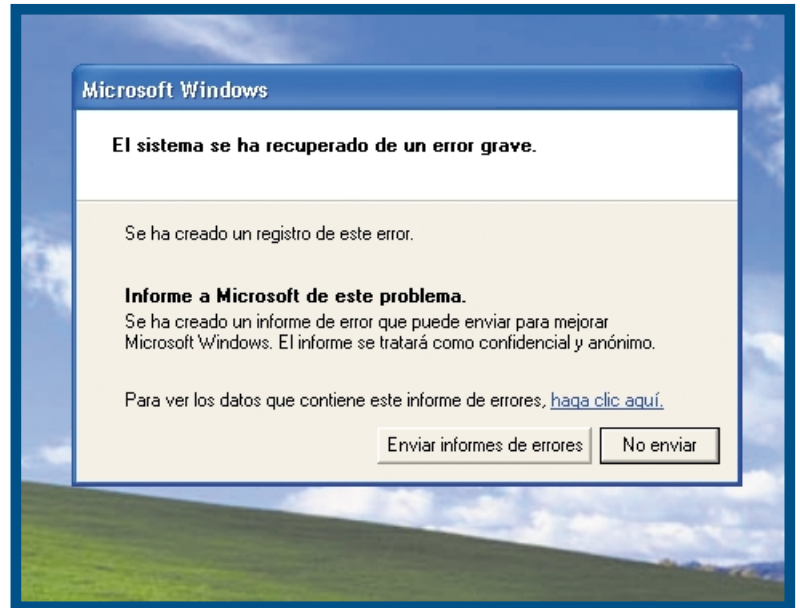
YA REVISAMOS LOS PROBLEMAS DE HARDWARE Y LOS ATRIBUIDOS A LOS PROGRAMAS. VEAMOS AHORA QUÉ OCURRE CUANDO EL SISTEMA OPERATIVO ES EL QUE PRODUCE EL CONGELAMIENTO Y QUÉ HERRAMIENTAS TENEMOS PARA SOLUCIONARLO.

El cuelgue más frecuente asociado al S.O. es el llamado **Access Violation** (violación de acceso), cuyo efecto es la muy temida pantalla azul. Pero ¿por qué sucede esto?

Las aplicaciones, durante su ejecución, necesitan recordar ciertos valores, que pueden ser pequeñas cantidades de información, así como también grandes paquetes de datos. Cuando un programa almacena cierta cantidad de información en memoria, precisa saber, de alguna forma, en dónde se localiza, para lo cual se implementa un método basado en una serie de **punteros**. Un puntero es, básicamente, una dirección de memoria que contiene los datos guardados por la aplicación en cuestión. Por otra parte, tenemos que aclarar que los sistemas operativos pueden trabajar en 16 bits, 32 bits o 64 bits, y este hecho define la cantidad de espacio de memoria que pueden direccionar. Por ejemplo, un sistema operativo que trabaje en 16 bits podrá direccionar 65.536 lugares en memoria (2 elevado a la 16) o, lo que es lo mismo, 64 Kb de memoria.

El problema que suele presentarse es que, en algunas ocasiones, ciertas fallas de programación en los sistemas o en las aplicaciones pueden provocar un mal funcionamiento de los punteros, los cuales pasan a almacenar direcciones de memoria que no pertenecen al programa o que no contienen la información que éste trata de buscar en ese momento. El resultado es que, cuando dicho programa quiere quitar o poner información en esa porción de la memoria, no puede hacerlo, lo que provoca la famosa violación de acceso.

El problema principal de las versiones de Windows 9X (incluida la Millennium) es que el siste-



ESTA PANTALLA NOS PERMITIRÁ IDENTIFICAR QUE SE TRATA DE UN PROBLEMA CON EL SISTEMA OPERATIVO. DEBEREMOS PROVOCAR UN VOLCADO DE MEMORIA PARA TENER MAYOR PRECISIÓN ACERCA DE LA CAUSA.

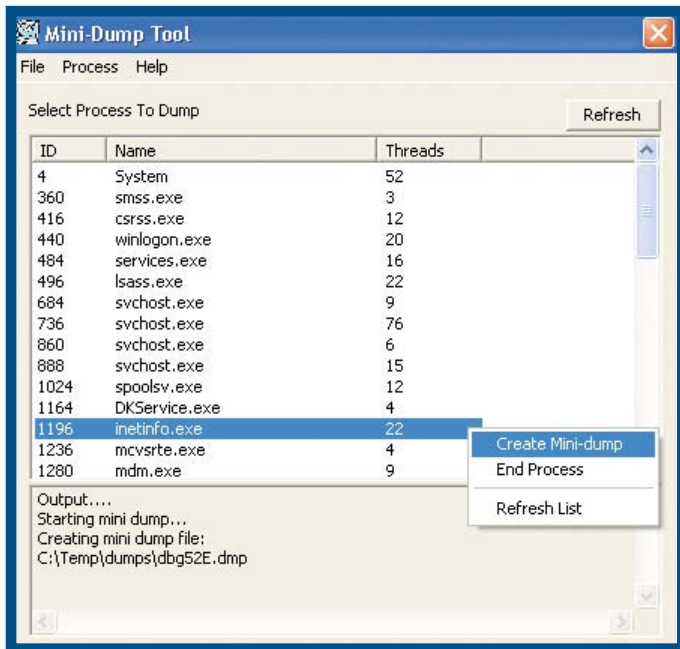
ma operativo no contempla una solución aceptable para ese tipo de situaciones, lo que da lugar a una cadena de errores cuando un programa quiere –y se le permite– escribir información en una parte de la memoria ocupada por otro programa, y este otro, a su vez, provoca un error en otro proceso, hasta que el sistema, simplemente, deja de responder o se reinicia.

¿Qué pasa en cada sistema?

Los sistemas operativos de las versiones NT de Microsoft, como Windows 2000 –uno de los más estables de toda la familia–, ofrecen una solución bastante efectiva con respecto al conflicto mencionado, ya que su mecanismo de seguridad no permite casi ninguna violación de acceso.

En otra clase de sistemas, como UNIX, este tipo de problemas reciben el nombre de Errores de Bus, si la falla es del sistema, o Error de Segmentación, si la causa es el programa. Lo que hacen estos sistemas operativos es eliminar la tarea que provoca el conflicto, con lo cual protegen al resto de los procesos que están en ejecución (y evitan que todo el sistema se congele).

→ UN PUNTERO NO ES MÁS QUE UNA DIRECCIÓN DE MEMORIA QUE CONTIENE LOS DATOS ALMACENADOS POR UNA APLICACIÓN.



ALGUNAS APLICACIONES, COMO MINI-DUMP TOOL (BILL.ATWILL.COM), PERMITEN GENERAR VOLCADOS DE MEMORIA PEQUEÑOS.



→ **VOLCADO DEL NÚCLEO (kernel dump)**

Por lo general, su tamaño es igual a la cantidad de memoria utilizada por el kernel del sistema. Para **Windows XP con 512 MB de RAM** es de aproximadamente 60 MB. Éste es el tipo de volcado más útil, ya que como vimos antes, para que se produzca una caída del sistema, la falla tiene que haber sido en el kernel, con lo cual se descarta el resto de la información, que no nos sería del todo útil.

→ **VOLCADO COMPLETO (full dump)**

Su tamaño es igual a la cantidad de memoria que se utilizaba al momento de detenerse el sistema. Contiene todos los ejecutables que estaban en memoria.

Tradicionalmente, los S.O. basados en UNIX son mucho más estables que los de Microsoft, ya que sus componentes están mucho más modularizados, menos integrados que en Windows, de manera que un error en un segmento del programa tiene menos probabilidades de afectar a otros segmentos, y entonces resulta más simple aislar los problemas.

Como conclusión, podemos decir que, cuando elegimos las aplicaciones para una computadora, desde el sistema operativo hasta los más pequeños programas, debemos tener en cuenta si éstos ya han sido lo suficientemente testeados como para considerarlos estables, y no instalar aplicaciones sólo por tener la última novedad en software.

Los volcados de memoria

Un volcado de memoria es una imagen de lo que Windows tenía en memoria en el momento en que se detuvo el sistema, de modo que presenta **información de gran valor**. Cabe destacar que Windows puede crear tres tipos distintos de volcados de memoria. Veamos las diferencias:

→ **VOLCADO PEQUEÑO (mini dump)**

Es un pequeño archivo –de unos 64 Kb en sistemas de 32 bits y de 128 Kb en los de 64 bits– que no contiene ninguno de los ejecutables que estaban en memoria, sino que incluye puntualmente la siguiente información respecto del problema: el mensaje de detención, sus parámetros y otros datos, una lista de controladores cargados, el contexto en el cual el procesador (PRCB) se detuvo, la información de proceso y contexto del núcleo (EPROCESS) del proceso que se detuvo, la información de proceso y contexto del núcleo (ETHREAD) del subproceso que se detuvo y, finalmente, la pila de llamadas del modo de núcleo para el subproceso que se detuvo.

Tengamos en cuenta que, en todos los tipos vistos, es necesario que el archivo de paginación de Windows esté en la misma unidad donde se almacenará el volcado (y no, por ejemplo, en otro disco); de lo contrario, se creará un volcado de memoria corrupto.

Herramientas que podemos usar

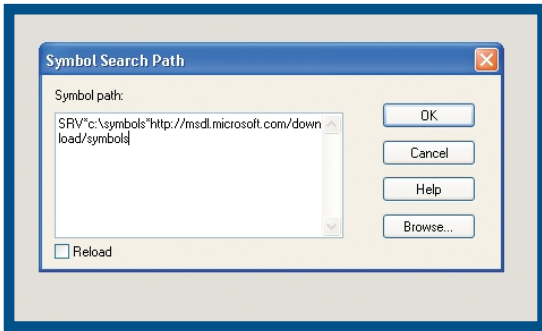
En primer lugar, podemos acceder a la pantalla de configuración de inicio y recuperación de Windows. Para hacerlo, vamos a [Panel de control/Sistema]; en la solapa [Opciones Avanzadas], veremos el panel de Inicio y recuperación. Allí vamos a [Configuración].

Debemos tener en cuenta que lo ideal sería no sobrescribir los volcados, con el fin de tener varias referencias para poder comprobar si en todos los casos el sistema se detuvo en las mismas circunstancias. Por último, el archivo donde se guarda el volcado de memoria está dentro del directorio **WINDOWS** bajo el nombre **MEMORY.DMP**.

Otra herramienta muy útil es **Debugger Tools for Windows** (también conocida como **Windows Debugger**). Se

VOLCADOS DE NÚCLEO

Para optimizar nuestro trabajo, es recomendable configurar el sistema operativo para que realice volcados de núcleo; de esta manera, siempre contamos con toda la información necesaria en disco sin tener que pasar horas analizando entre cientos de megas de información. Podremos realizar este proceso accediendo al menú [Inicio/Configuración] de Windows XP.



En esta ventana, deberemos indicar la ruta donde se encuentra la información vinculada al volcado de memoria ocasionado.

trata de un depurador que, en general, se incluye con Visual Studio o se puede descargar en forma gratuita desde el sitio web de Microsoft: www.microsoft.com/whdc/devtools/debugging/installx86.mspx. Básicamente, esta herramienta permite desensamblar **el volcado de memoria para poder analizarlo y sacar conclusiones**. Como veremos a continuación, esta tarea puede parecer más compleja de lo que realmente es. Una vez que bajamos e instalamos el programa, deberemos ejecutarlo, ir al menú **[File]** y seleccionar la opción **[Symbol file path]** (es la ruta donde deberá buscar los símbolos para poder interpretar el volcado de memoria). Allí deberemos colocar lo siguiente:

SRV*c:\symbols*http://msdl.microsoft.com/download/symbols

Una tabla de símbolos es una lista de identificadores, sus ubicaciones en el programa y sus atributos. Esta tabla se crea al compilar el programa y, por lo general, no se entrega al usuario en el producto final, ya que sólo se utiliza durante el proceso de depuración. Con la ruta detallada anteriormente en el depurador, indicamos que se deben descargar las tablas de símbolos correspondientes desde el servidor de Microsoft y guardarlas dentro de la carpeta C:\symbols de nuestra máquina. Es muy importante crear la carpeta **symbols** antes de ejecutar este programa, dado que ahora ya tenemos la información. Sólo debemos saber cómo y cuándo provocar un volcado e interpretarla.

Cuelgues causados por el usuario

En algunas ocasiones, un cuelgue congela la pantalla de modo que impide realizar cualquier tipo de acción; sin embargo, no se muestra una pantalla azul ni se realiza el volcado de memoria. En este tipo de situaciones, es prácticamente imposible determinar cuál es el motivo específico del cuelgue. Por suerte, a partir de Windows 2000 existe la posibilidad de generar manualmente un error crítico y forzar

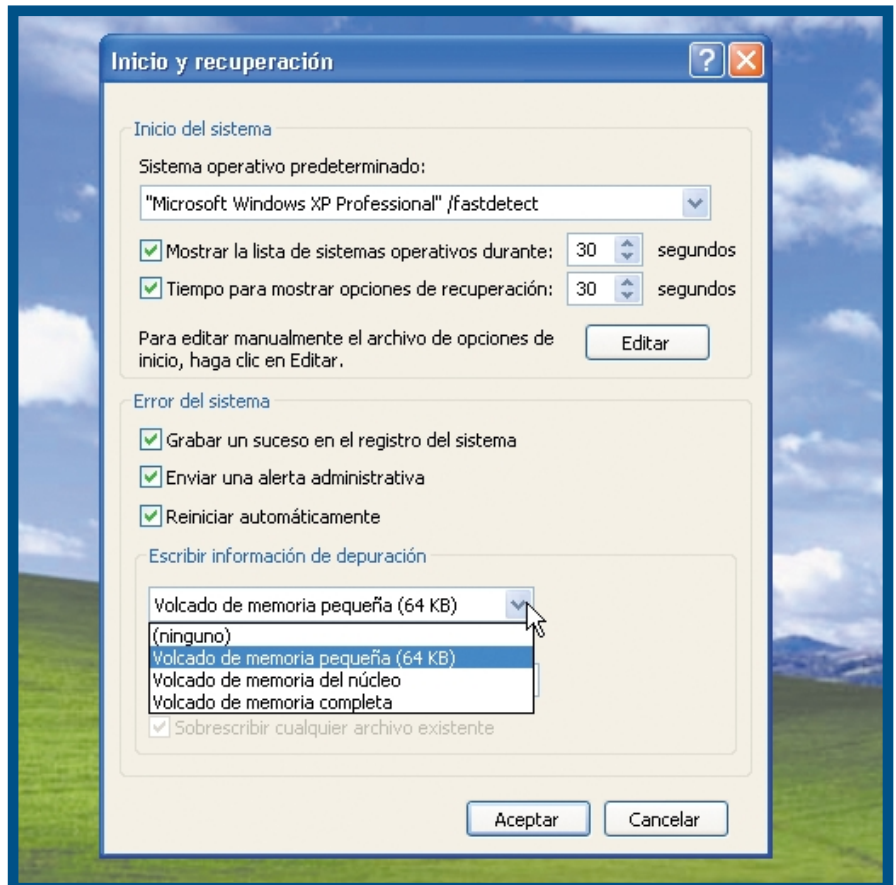
al sistema a detenerse por completo mostrando la pantalla azul y provocando el volcado de memoria.

Generar el volcado de memoria

Para provocar un volcado de memoria, debemos **editar el Registro** de manera que, al presionar las teclas **<Ctrl> (derecha)** y **<Bloq Despl> (Scroll Lock) dos veces**, Windows sepa que queremos hacerlo. Tenemos que buscar la siguiente rama en el Registro:

HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\i8042prt\Parameters

Allí agregamos un nuevo valor de tipo **DWORD** con el nombre **"CrashOn CtrlScroll"** y establecemos su contenido en **1 (uno)**. Luego, reiniciamos el equipo y, en el momento en que sea necesario, se podrá



➔ **AL DEPURAR EL VOLCADO DE LA MEMORIA, EN LA CABECERA SE NOS INDICA CUÁL ES EL ESTADO: CORRUPTO O NO, SI ESTÁ CORRUPTO NO NOS SERÁ ÚTIL.**

generar un volcado presionando simultáneamente las teclas <Ctrl> + <Bloq Despl> (dos veces).

El análisis

Una vez que tenemos todo lo necesario para comenzar, cargamos el volcado de memoria en el depurador, yendo a [File/Open Crash Dump...].

Cuando seleccionamos el archivo, el depurador procederá a descargar las tablas de símbolos necesarias. Al depurar el volcado de memoria, en la cabecera se nos indica cuál es el estado: corrupto o no. Si está corrupto no nos será útil. Una vez abierto el archivo que contiene el volcado de memoria (dump) y luego de comprobar que está en perfectas condiciones, procedemos a realizar un análisis profundo sobre él para **encontrar la causa de la falla**. Para realizar esta tarea, ejecutamos uno de los tantos comandos que posee el depurador de Windows (windbg): **analyze**. Este comando nos mostrará en pantalla información detallada sobre las circunstancias que desencadenaron el cuelgue o detenimiento del sistema. Entre todos esos datos, deberemos saber rescatar aquello que nos interesa.

Finalizado el proceso de análisis y con la información que hemos recogido del proceso, debemos prestar atención a los siguientes puntos:

➔ **DESCRIPCION_DE_LA_EXCEPCION (xx):** Es un texto que describe el **bugcheck** que ha producido el detenimiento del sistema. Los números que aparecen entre paréntesis en formato hexadecimal indican el número identificador del bugcheck. Podemos notar que es más fácil comprender un mensaje del tipo "KMODE_EXCEPTION_NOT_HANDLED (1e)" (excepción de modo kernel no manejada o no controlada), que uno del tipo "STOP:0x0000001e". En las páginas siguientes veremos cómo interpretar estos mensajes.



DESDE ESTA HERRAMIENTA, PODREMOS CARGAR EL VOLCADO DE MEMORIA MEDIANTE LA OPCIÓN FILE/OPEN CRASH DUMP. LUEGO DE SELECCIONAR EL ARCHIVO, EL DEPURADOR DESCARGARÁ LAS TABLAS DE SÍMBOLOS NECESARIAS.

➔ **ARGUMENTS (Argumentos):** En esta sección se encuentra información sobre determinados valores relacionados con el código de error que produjo el detenimiento del sistema. Son **cuatro** en total y cada uno contiene información específica:

Arg1: Dirección de memoria a la que se hace referencia o **código de excepción** (depende del tipo de error que sea).

Arg2: Dirección de memoria donde **ocurrió la excepción** o identificador del IRQL (*Interrupt ReQuest Level* o nivel de pedido de interrupción).

Arg3: Primer parámetro de la excepción, que indica si lo que se intentó realizar fue un proceso de lectura (0x00000000) o de escritura (0x00000001).

Arg4: Dirección de memoria (origen) desde donde se trató de realizar el proceso que generó la **excepción**.

➔ **DEFAULT_BUCKET_ID:** Define la categoría general dentro de la cual se clasifica la falla.

➔ **IMAGE_NAME:** Éste es el nombre del ejecutable o driver que tenía el control al momento exacto de producirse el congelamiento en el sistema.

➔ **STACK_TEXT:** Contiene información sobre la pila al momento de producirse la falla. Podemos utilizarla para determinar cuál fue el proceso que desencadenó el problema y buscar una solución adecuada.

