

Fortificando el Sistema con AppArmor

JAULA DORADA

Tras penetrar en un sistema remoto, los intrusos podrían pensar que ya han pasado la parte más dura y están a salvo, pero AppArmor les va a estropear la diversión, encerrándolos en una jaula virtual.

POR RALF SPENNEBERG

Nadie es perfecto, algo que es particularmente cierto en el mundo del software. Cualquier aplicación que no sea trivial contendrá errores de programación que serán aprovechados por los hackers para hacerse con el control del software, haciendo que los programas realicen tareas que sus desarrolladores jamás llegaron a pensar. La situación comienza a ser crítica cuando la aplicación posee privilegios diferentes de los privilegios del atacante.

Por ejemplo, el comando *ping* necesita de los privilegios de superusuario para poder enviar el formato especial de paquete que utiliza. Aunque teóricamente es posible hacer un mal uso de estos privilegios para causar toda clase de problemas. A pesar de que el comando *ping* es un programa con un buen comportamiento, un atacante capaz de hacerse con la herramienta tendría acceso sin restricciones al resto del sistema.

AppArmor [1] modifica esta situación. En vez de permitir a un programa de

root el acceso sin restricciones a todo el sistema, asigna límites con la intención de obtener un balance entre efectividad y complejidad. AppArmor utiliza un mecanismo simple y transparente para proporcionar un estándar alto de protección, similar al de Systrace. No intenta

Listado 1: Compilando un kernel compatible con AppArmor

```
01 tar xjf linux-2.6.15.tar.bz2
02 cd linux-2.6.15
03 patch -p1
  <../aa_2.0-2.6.15.patch
04 patch -p1
  <../aa_namespace_sem-2.6.15.
  patch
05 make oldconfig
06 make bzImage
07 make modules
08 make modules_install
09 make install
10 rmdir /subdomain
11 ln -s /sys/kernel/security/
  subdomain /subdomain
```

competir con sistemas más complejos como SELinux o RSBAC, pero de nuevo, la configuración de estas alternativas requiere más conocimientos sobre la administración de sistemas.

Fortificado

Además de todos los obstáculos adicionales, la primera regla de seguridad es evitar las vulnerabilidades. En un ordenador protegido con AppArmor, o en cualquier sistema de esta clase, deben deshabilitarse todos los servicios que no se necesiten, tener los últimos parches instalados y utilizar una configuración a medida cuidadosamente estudiada. Esto deja a AppArmor con un sistema libre de vulnerabilidades conocidas hasta la fecha y libre de exploits.

AppArmor monitoriza los ficheros y las aplicaciones accedidas y el tipo de acceso de que se trate; al mismo tiempo, gobierna el uso de los privilegios de superusuario. Dependiendo de la versión del kernel, Linux puede distinguir entre 29 capacidades diferentes (véase *man 7 capabilities*). Por ejemplo, *CAP_KILL* se refiere a la habilidad del superusuario de terminar un proceso y *CAP_NET_RAW* a la de crear paquetes de red arbitrarios.

En el caso del comando *ping*, AppArmor le asignaría el uso de *CAP_NET_RAW*, pero le denegaría el uso de *CAP_KILL*. Esto impediría a un



Figura 1: AppArmor mantiene un perfil para cada aplicación protegida.

atacante la posibilidad de matar otros procesos.

AppArmor en Linux

Las distribuciones Novell SLES9 y Suse Linux 10.0 vienen con el sistema AppArmor por defecto. AppArmor no era libre por

entonces (véase el cuadro “Immunix”). Tras la aparición de la versión de AppArmor bajo la licencia GPL, Novell ha anunciado ahora que integrará AppArmor en OpenSUSE 10.1. Si no se desea esperar, puede utilizarse OpenSUSE 10.0, aunque la instalación es bastante compleja. Entre otras cosas habrá

que modificar y recompilar el kernel, por lo que la actualización no está recomendada para usuarios sin experiencia.

Los RPMs de AppArmor para OpenSUSE 10.0 están disponibles en Novell Forge [3]. Aunque Suse/Novell compilaron los RPMs para OpenSUSE 10.1 Alpha, también funcionan en OpenSUSE 10.0. La instalación sigue los pasos habituales, `rpm -ivh nombre-paquete.rpm`. El kernel también requiere el soporte para AppArmor. Novell dispone de los parches necesarios en [4]; los parches están diseñados para la versión original del kernel 2.6.15 [5]. Para compilar un kernel compatible con AppArmor, hay que cargar tanto el kernel original como los parches `aa_2.0-2.6.15.patch` y `aa_namespace_sem-2.6.15.patch`. Luego tan sólo hay que seguir los pasos del Listado 1.

También es posible instalar AppArmor en sistemas que no sean Suse, como Debian o Fedora. Sin embargo, esto implica la compilación de los archivos con el código fuente y realizarlo sin un GUI, como GUI se entiende que corre Yast 2.

Arranque y Parada

Suse dispone de controles basados en el GUI para ejecutar AppArmor. Se ejecuta

Immunix

Novell adquirió Immunix a mediados de 1995. Immunix se ha especializado en el desarrollo de soluciones de seguridad durante años. La compañía modificó el GCC, conocido como StackGuard, para que compilase las aplicaciones de modo que impidiese los diversos tipos de problemas relacionados con los desbordamientos de búfers. Para ello, StackGuard utiliza el denominado *canario*. Este sistema genera números aleatorios cuando el programa se ejecuta. Antes de cada llamada a una subrutina almacena los valores canarios en la pila. Si el valor se ve modificado cuando el programa regresa, se termina con la sospecha de un desbordamiento de búfer. (El término canario viene de la minería, cuando los mineros utilizaban canarios para asegurarse de que la atmósfera estaba libre de acumulaciones de monóxido de carbono).

Immunix también lideró el desarrollo de la interfaz LSM (Linux Security Modules [2]) en el kernel 2.6. Esta interfaz le permite a los módulos del kernel monitorizar los eventos críticos con respecto a la seguridad en diversos puntos. Algunos sistemas seguros utilizan LSM, como LIDS (Linux Intrusion Detection System) y SELinux (Security Enhanced Linux). Este último desarrollado por la NSA (National Security Agency, USA) implementa un sistema MAC (Mandatory Access Control) que permite a los administradores definir políticas detalladas de permisos de accesos. Este conjunto restrictivo de políticas puede incluso monitorizar y restringir al superusuario, root, y todas sus actividades.

Como las distribuciones actuales de Novell/Suse Linux poseen el soporte a nivel del kernel para el programa SELinux, no necesita las políticas necesarias para que funcione.

El sistema AppArmor es también de Immunix. Novell ha posicionado a AppArmor como una alternativa simple y efectiva a SELinux. SELinux es una solución que requiere una compleja configuración, sin embargo, AppArmor simplemente tiene como objetivo las aplicaciones individuales y los eventos críticos. A finales de enero de 2006, Novell sacó a la luz el código fuente de AppArmor bajo la licencia GPL e inmediatamente publicó el código en su propio sitio web [3].

Perfiles

El paquete de AppArmor contiene perfiles para los siguientes servidores:

- Postfix
- Apache (en modo prefork)
- Squid
- OpenSSH server
- NTP server
- Name Service Caching Daemon (nscd)
- Identd
- Protocol services Klogd and Syslogd

También dispone de perfiles para diversos programas clientes:

- Acrobat Reader
- Ethereal
- Opera
- Firefox
- Evolution
- Gaim
- Realplayer
- Man
- Netstat
- Ping
- Traceroute

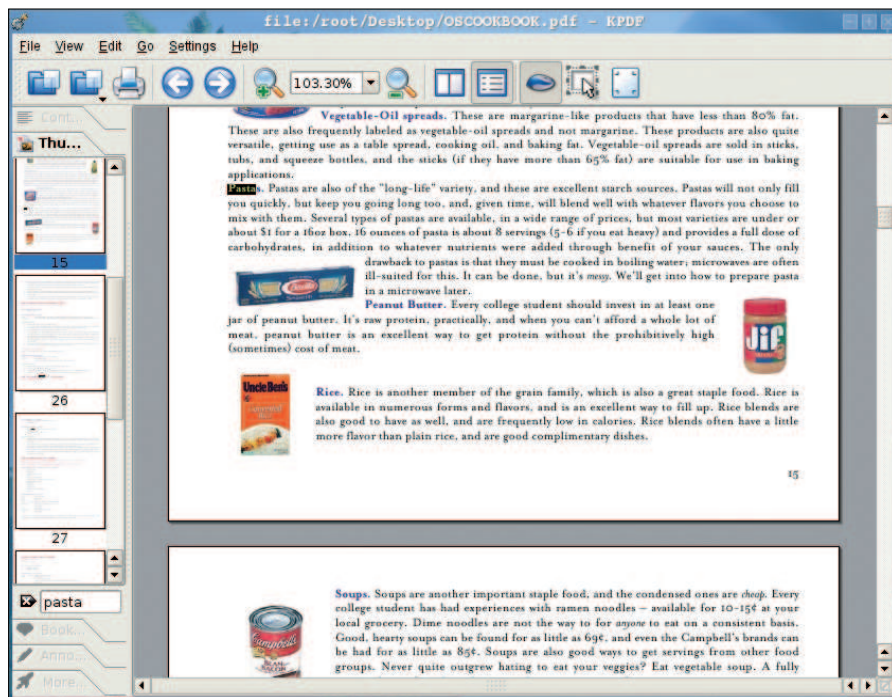


Figura 2: Kpdf mostrando un documento PDF. Si el documento fuera de un atacante, éste podría explotar alguna vulnerabilidad en el PDF.

Yast y se selecciona AppArmor en la columna de la izquierda. Luego aparece en la derecha la barra de control de AppArmor. Aquí es donde se puede comprobar el estado actual de AppArmor y donde se puede habilitar. Si se prefiere puede utilizar se la línea de comandos: introduciendo `rcsubdomain start` y `rcsubdomain stop` (como root).

Para que AppArmor funcione, la herramienta debe ejecutarse antes de que arranquen las aplicaciones protegidas. Esto es por lo que AppArmor se ejecuta en el arranque del sistema. El programa también necesita un fichero de perfil en `/etc/subdomain.d` para cada aplicación que vaya a proteger.

Autoprotección

Novell posee perfiles para una gran cantidad de comandos críticos (véase el cuadro "Perfiles"). Voy a utilizar el visor de ficheros Kpdf (Figura 2) para mostrar lo fácil que resulta la generación de un

Probando AppArmor

Cuando se utilice el asistente para crear un perfil, no incluirá el comando `print`. Esto significa que el asistente no incluirá la función en el perfil. De este modo cuando se use posteriormente Kpdf, se observará que todas las funciones están disponibles como siempre, excepto que no se podrá imprimir.

perfil gracias al modo de aprendizaje de AppArmor.

En los últimos años se han descubierto diversos errores de programación en varios visores PDF, como Xpdf y Kpdf. Un atacante que supiera de estos errores podría manipular un fichero PDF para inyectar y ejecutar un código dañino y hacerse con el control del visor de PDF.

Para añadir Kpdf a la lista de programas que AppArmor monitoriza, hay que ejecutar Yast 2 y seleccionar el asistente de perfiles bajo *AppArmor*. Se comienza introduciendo el nombre de la aplicación y su ruta completa. Si no se conoce la ruta, se puede teclear `which kpdf` para averiguarla. La ruta en Suse Linux es `/opt/kde3/bin/kpdf`.

Se ejecuta la aplicación y se maneja durante un rato. Hay que asegurarse de utilizar todas las funciones de Kpdf. Pero también hay que asegurarse de que en la fase de aprendizaje es imposible que se produzca un ataque. AppArmor posteriormente permitirá todas las funciones que Kpdf utilice ahora. Tras la ejecución de la lista completa de funciones, se puede cerrar la aplicación. Ahora pueden analizarse los resultados grabados en el perfil del asistente. Para ello, se selecciona `Scan system log for AppArmor events` (Figura 3).

Procesos Hijos

Tras completar el análisis de eventos, que puede llevar unos cuantos minutos, el asistente pregunta si se desean permitir todos los tipos de accesos, sugiriendo una acción para cada uno. Si el programa monitorizado invoca a otro programa, por ejemplo, el asistente de perfiles proporciona las siguientes opciones:

- Inherit: Las mismas restricciones de Kpdf se aplican a la nueva aplicación *Kdialog*.
- Profile: Esta aplicación posee su propio perfil.
- Unconfined: AppArmor no monitorizará este programa.
- Deny: Se impedirá la ejecución de la nueva aplicación.

Unconfined es una opción, ya que el Kpdf utiliza el programa *KDialog* para abrir y cerrar ficheros. Como esto da al programa completa libertad, el asistente advierte de posibles vulnerabilidades (Figura 4). Podría ser mejor crear un perfil para *KDialog* para restringir el acceso del programa sólo a los ficheros PDF.

Listado 2: Incluyendo Abstractionsv

```
01 # vim:syntax=subdomain
02 # Last Modified: Sun Jan 22
    10:16:55 2006
03 /opt/kde3/bin/kpdf
    flags=(complain) {
04 #include
    <abstractions/authentication>
05 #include <abstractions/base>
06 #include <abstractions/bash>
07 #include <abstractions/gnome>
08 #include <abstractions/kde>
09 #include
    <abstractions/namespace>
10 #include
    <abstractions/user-write>
    11
    12 / r,
    13 /etc r,
    14 /etc/X11/.kstylerc.lock rw,
    15
    /etc/X11/.qt_plugins_3.3rc.lock
    k rw,
    16 /etc/X11/.qtrc.lock rw,
    17 /etc/exports r,
    18 /etc/rpc r,
    19 ...
    20 }
```



Figura 3: AppArmor grabando eventos para el análisis.

Acceso a Fichero

Tras tomar una decisión sobre cada aplicación que Kpdf llama, el asistente pregunta sobre los ficheros usados por Kpdf. Se puede escoger *Allow* para permitir el acceso a la mayoría de los ficheros. El asistente tiene una directiva incluida para ciertos ficheros.

Muchas aplicaciones necesitan acceder a los ficheros de configuración de KDE. En vez de permitir el acceso a cada fichero individualmente y saturar innecesariamente el perfil, simplemente se le puede añadir una plantilla a éste. Para ello se utiliza la línea `#include <abstractions/kde>`. Las plantillas para los perfiles son referidas como abstracciones en la jerga de AppArmor.

AppArmor viene con una colección de abstracciones adicionales para el intérprete de comandos Bash y para el DNS, por ejemplo. Tras contestar todas las preguntas se regresa a la pantalla de bienvenida del asistente. El perfil se almacena en `/etc/subdomain.d/opt.kde3.bin.kpdf` (véase el Listado 2 para una muestra). Ahora ya se puede quitar el asistente de perfiles y cerrar la aplicación.

Afinando

Si la aplicación falla a la hora de funcionar como se esperaba, sólo hay que ejecutar el asistente de perfiles de nuevo y repetir el proceso de aprendizaje. El asistente primero analiza el perfil existente y luego actualiza los cambios efectuados en él. Cualquier

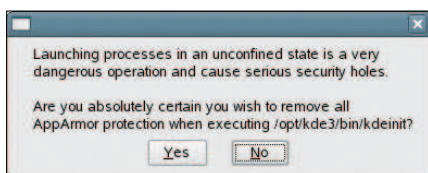


Figura 4: Cuando se selecciona Unconfined, el asistente advierte sobre el potencial riesgo de seguridad.

entrada que se haya añadido manualmente por medio de un editor se conservará. Después de cada entrada manual, hay que reejecutar AppArmor para indicarle a la herramienta que cargue de nuevo el perfil. Como alternativa, se podría utilizar Yast 2 y elegir o bien actualizar el perfil o seleccionar el icono con el bolígrafo para editar un perfil (Figura 5).

Como los servicios de red están expuestos al peligro constantemente, Novell proporciona el programa *unconfined*, que descubre los servicios de red que se están ejecutando en el sistema y muestra su estado bajo AppArmor. La salida proporcionada en el Listado 3 muestra que el sistema del ejemplo está ejecutando CUPS y que el portmapper RPC no está siendo monitorizado. Novell no tiene perfiles para estos servicios.

En las próximas semanas y meses, se espera que Novell amplíe los perfiles disponibles. Si se está interesado en seguirle la pista

Listado 3: Mostrando el estado de AppArmor

```
01 # unconfined
02 7988 /usr/lib/postfix/master
   confined by
   '/usr/lib/postfix/master
   (enforce)'
03 7988 /usr/lib/postfix/master
   confined by
   '/usr/lib/postfix/master
   (enforce)'
04 8025 /usr/sbin/cupsd not
   confined
05 8025 /usr/sbin/cupsd not
   confined
06 8081 /sbin/portmap not
   confined
07 8081 /sbin/portmap not
   confined
08 8109 /usr/sbin/sshd confined
   by '/usr/sbin/sshd (enforce)'
```

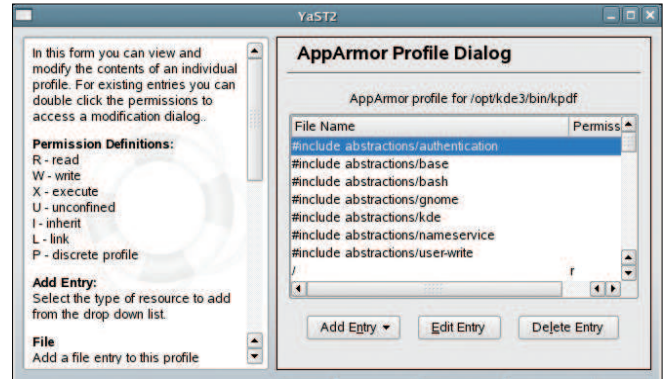


Figura 5: Yast 2 permite la edición del perfil de AppArmor.

a los desarrollos, hay que revisar la lista de correo [6] y consultar de vez en cuando el sitio web de AppArmor [1].

Bien Protegido

AppArmor monitoriza las aplicaciones críticas. Los programas sólo tienen permitido el acceso a los ficheros especificados y únicamente pueden invocar a una serie de comandos específicos. Si la aplicación tiene un agujero de seguridad que pudiera permitirle a un atacante ejecutar un intérprete de comandos u otros comandos con los privilegios de la víctima, surge AppArmor para proteger el sistema. La aplicación se ejecuta en una especie de caja de arena, o jaula, y es incapaz de salir de ella.

AppArmor no puede evitar las vulnerabilidades, pero puede impedir que los atacantes puedan explotarlas en su beneficio. Esto protege de forma efectiva a los usuarios de aquéllos. AppArmor está altamente recomendado para los programas que son accesibles a través de la red o que manejen datos de fuentes poco fiables como correos electrónicos, imágenes, vídeos o documentos ofimáticos. ■

INFO

- [1] AppArmor: <http://www.opensuse.org/AppArmor>
- [2] LSM: <http://lsm.immunix.org>
- [3] Paquetes AppArmor: <http://forge.novell.com/modules/xfcontent/downloads.php/apparmor/Stable>
- [4] Parches del Kernel para AppArmor: <http://forge.novell.com/modules/xfcontent/downloads.php/apparmor/Development/>
- [5] Kernel repository: <http://www.kernel.org>
- [6] Lista de correo de AppArmor: <http://forge.novell.com/mailman/listinfo/apparmor-general>