Sys Admin
the journal for UNIX systems administrators

Fun with SunScreen

Peter Baer Galvin

Solaris 9 has many security enhancements, including some supported ones like secure shell, a secure LDAP client, Kerberos KDC and administration tools, IKE, enhanced RBAC, and Xserver connection security. Seemingly, one of the biggest improvements is the inclusion of the previously commercial SunScreen V. 3.2 firewall. A Solaris 9 installation with the SunScreen features enabled should provide a good platform for evaluating its utility. In this column, I will first take a look at what SunScreen is supposed to do, and then describe how to install, configure, and use it to find out the fact and the fiction.

Features

SunScreen is considered by Sun to be a full-featured firewall. It was sold as a separate software product by Sun for a few years before being included with Solaris 9. It has an extensive feature set and provides the bulk of the features found in other major firewalls. It's stateful and dynamic, and at its core is a packet-filtering system like Checkpoint Firewall-1. The documentation describes these features:

- 130 multithreaded stateful packet filters
- Complete network address translation (NAT) abilities
- SKIP and IPSec/IKE VPN client support
- Ordered rule sets
- Many integrated authentication methods
- Detailed logging with extensive log management
- Multiple firewall management
- A Java applet GUI
- Complete command-line control
- Proxies for FTP, HTTP, SMTP, and Telnet
- Ability to integrate with Trend Micro's VirusWall for content scanning

The SMTP proxy is especially useful, allowing control over spam and

relaying. For example, SunScreen could be used on an email server, simplifying the sendmail (or smap, or other SMTP server) configuration.

One unique and useful feature of SunScreen it that it can act as a bridge as well as a router. This “stealth mode” has two ramifications, as follows:

1. The firewall can be added to an existing network without any renumbering or reconfiguring. It can be dropped into the middle of a production environment, for example, and no one would be the wiser.
2. The system does not present IP addresses, and thus cannot be detected or attacked easily. In fact, unless remote administration is needed, no network interfaces even need to be configured. In this mode, “only” 15 network interfaces are supported.

The default implementation mode is “routing”, in which each interface has an address on a different network, and the system acts as a filtering router. This mode supports an unlimited number of network interfaces (including 10Mb, 100Mb, 1000Mb, ATM, token ring, and FDDI).

Because of the stealth mode feature and the availability of low-cost Sun servers, such as the Netra V100, it’s conceivable that firewalls could be deployed at many points throughout an environment, rather than just at the edges, as is the case with most firewall deployments. Of course, performance, reliability, and manageability must be considered and tested before a full-force deployment such as this. A system can even be used for routing on some interfaces and in stealth mode on others, but that is more complex and has some limitations.

Installation

So the theory is good, but what about the practice? (As Yogi Berra pointed out, “In theory, there is no difference between theory and practice; in practice, however, there is.”). I installed SunScreen as part of the Solaris 9 full operating system installation. By selecting to perform the full install and the companion software install, it was automatically added to the system. There were some error messages about the SUNWeu* packages not being found, but the software installed successfully, as indicated by the log file in /var/sadm/install/logs. These error messages were in pop-up screens and could easily be missed. The installation documents (discussed later) indicate that these packages are required, so I installed them from the Solaris 9 Software 1 of 2 CD-ROM, from the /s0/Solaris_9/Product directory. SunScreen can also be installed manually from a secret directory

— /Solaris_9/ExtraValue/CoBundled, which can be found on the Solaris 9 DVD or the Software 2 of 2 CD. I reinstalled SunScreen after installing the language packages to avoid problems.

After installation, things got a little sticky. Where was SunScreen and how could it be run and managed?

The Solaris 9 documentation set includes the “System Administration Guide, Security Services”. Although this document has information on RBAC, SEAM, secure shell, and other general security tips, it includes nothing for SunScreen. Likewise, there is no information in the Solaris 9 installation guide or the Solaris 9 Operating Environment Package List regarding SunScreen.

Oddly enough, even in the “What’s New in the Solaris 9 Operating Environment” document, with its own security section, SunScreen is mentioned only under the “additional security software” section. In another example of the less-than-complete SunScreen integration, there is no interface to it from the Solaris Management Console, which is supposed to be the centralized administration point for Solaris systems. It appears that the addition of SunScreen to the standard Solaris 9 installation was either hastily done or just poorly executed. I expect that in future releases, SunScreen will be more fully integrated.

A direct search on docs.sun.com (or within the installed documentation) for “SunScreen” was more fruitful. There are several dedicated manuals, including “SunScreen 3.2 Administrator’s Overview”, “SunScreen 3.2 Configuration Examples”, and “SunScreen 3.2 Installation Guide”. There is also a separate SKIP User’s Guide. These documents are must reading for SunScreen users. There are also man pages associated with the SunScreen packages, but they are stored separately in /usr/share/man/man4sunscreen. SunScreen management is quite an undertaking, unfortunately, as just the Overview is 356 pages. (As a side note, installation information is included on adding SunScreen to Trusted Solaris 8. There might be licensing ramifications from taking the SunScreen on the Solaris 9 CD and installing it on TS8, but I’m not aware of any.)

For my test, I added SunScreen to my desktop workstation (SunBlade 100) to increase its security. For “real” firewall use, only the minimum Solaris software packages should be installed (either “core” or “end user”). The system-level security of a Solaris box running SunScreen in stealth mode is implemented via the JASS toolkit. Furthermore, security can be enhanced

by running the script /usr/lib/sunscreen/lib/harden_os. There is no way to undo the hardening changes, so be certain you want them before executing the script. All security improvements should be done before the machine is attached to a network, for maximum security.

SunScreen requires about 300 MB of available disk space to run, which includes the binaries as well as configuration files (/etc/sunscreen), log and temporary files (/var/sunscreen), internal files (/usr/lib/sunscreen), and man pages. The software only uses 32 MB of memory, but 64 MB are recommended for the administration software.

Depending on your operating system release and the type of encryption you desire, some downloading might be required from:

http://www.sun.com/software/solaris/encryption/download.html

For Solaris 9, DES and 3DES are in-built, so only AES and Blowfish need to be downloaded if desired. As always, be sure to install the latest operating system security patches for the operating system release in question.

Administration can occur remotely or locally, via a Web browser with a Java plug-in or via the command line. For local GUI administration, simply run a browser on the firewall and connect to port 3852. Remote administration requires establishing trust between the remote machine and the firewall. The remainder of my evaluation took place using local GUI administration.

Administration

The initial screen allows a login, with a choice of viewing information (e.g., read-only) or managing policies. The first task is to change the password of the administrator. A login as user “admin” with password “admin” in “manage policies mode” brings me to a policies list. SunScreen installs with a default policy already in place. Choosing the “initial” policy and pushing “edit” brings a screen with a “common objects” section and a “policy rules” section. Under “common objects”, choosing “authorized user” and “search” reveals the admin user in the results field. Selecting that shows the details on the admin user. Choosing “edit” brings up a dialog box that (finally!) allows a password change.

But we’re not done yet. Completing the dialog creates a new policy rule,

version 2 of “initial”. The “activating policy” button (as you might guess) activates the new version of the policy. There is also a “verify policy” button to perform a sanity check of the new policy. You can only view the currently running policy, not edit it. Edits can only be made to a new version of the policy that is then activated. This fail-safe methodology is a bit cumbersome, but version control usually more than pays for its added inconvenience through the ability to audit changes and return to previous states.

The next area of exploration is objects, which are used as the source, destination, and service entries of rules. The initial rule allows “common” services from any source to any destination. But what are those? The section of the screen labeled “Common Objects” provided the information by selecting “type -> service”, and choosing “search”. The results window showed all preconfigured groups. Selecting “common” from that drop-down list showed all the services in that group. I wouldn’t call this the most intuitive interface, but given the limits of a Web browser as a management console, it’s at least workable.

Creating a new rule is relatively simple. For example, to disable ftp access to the local host, the “add new rule” button in the “policy rules” section gave another dialog box. Rule ordering is important, so the deny ftp rule must be moved above the default allow rule. Another way to make this change would be to edit the “common” object and remove ftp, but adding an explicit rule is, well, more explicit. Activating the policy resulted in denied attempts to ftp to the machine. In the new rule dialog, I selected the detailed logging option. But where is that log? Along the top of the screen of the administration console (i.e., the Web browser), the “information” button shows the firewall status, as well as “logs” and “statistics” tabs. Under “logs”, each policy installation is logged, as well as authentication events and rule events that are configured to log. The denial events are here as expected.

Unexpected but welcome is the bi-directional functionality of the firewall. A rule to disallow ftp from the machine to a remote machine worked, as did a rule to disallow ftp from that remote machine to the test Sun machine. This functionality is available in some commercial security products, but is now available for free with Solaris. It is an excellent feature that can be used to thoroughly secure a Sun host.

Conclusions

Once the interface quirks are understood, managing a SunScreen Solaris box is very straightforward. If SunScreen is going to be added to an existing host, I recommend using only “allow” rules to start, with logging, to determine exactly what is happening on the host. Then introduce “deny” rules as you determine that protocols are not being (or should not be) used. Of course, testing in a controlled environment is the best way to evaluate the utility and impact of SunScreen.

SunScreen is a very complete firewall solution. I do not expect that it will take sales away from the commercial firewall hardware products, although it might in cost-conscious environments. It is certainly an excellent addition to standard Solaris 9 on Sun machines. It can be used just for monitoring of traffic and network activities on a host, or it can be used to limit services, inbound and out. Where tcp_wrappers is a partial solution, SunScreen can be a full solution to connection management.

There are many aspects of SunScreen that I didn’t cover here. For example, it can be run in a high-availability configuration (one node is active and the other passive until the active node fails). It can be remotely administered by a machine running SKIP or IKE. Also, proxy settings can allow only specific, authenticated users to telnet or ftp, or only specific Web sites to be visited by a given set of users. SecurID and radius can be used for authentication, as another example of the completeness of the SunScreen firewall solution.

On the plus side, the ever-present “documentation” button is useful for bringing up context-sensitive help. Also, the firewall software itself was stable throughout the testing, Unfortunately, the Java GUI crashed once and hung once during very light use. Perhaps command-line is the best way to go.

*Peter Baer Galvin (**http://www.petergalvin.org**) is the Chief Technologist for Corporate Technologies (**www.cptech.com**), a premier systems integrator and VAR. Before that, Peter was the systems manager for Brown University’s Computer Science Department. He has written articles for* Byte *and other magazines, and previously wrote Pete’s Wicked World, the security column, and Pete’s Super Systems, the systems management column for* Unix Insider *(**http://www.unixinsider.com**). Peter is coauthor of the* Operating Systems Concepts *and* Applied Operating Systems Concepts *textbooks. As a consultant and trainer, Peter has taught tutorials and given talks on security and systems administration worldwide.*
