GIAC
CERTIFICATIONS

Global Information Assurance Certification Paper

Copyright SANS Institute
Author Retains Full Rights

This paper is taken from the GIAC directory of certified professionals. Reposting is not permited without express written permission.

Interested in learning more?

Check out the list of upcoming events offering
"Security Essentials (Security 401)"
at http://www.giac.org/registration/gsec

SANS Security Essentials
GSEC Practical Assignment
Version 1.2d

Bob Mahoney

GSEC v1.2e
Portsmouth, NH

© SANS Institute 2000 - 2002, Author retains full rights.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

Welcoming MacOS X:
a Example of Practical Threat Assessment
In a University Environment

This paper intends to provide an example of the general process of assessing the potential security threat of a new computer platform, focusing primarily on a university environment. While a university network is unusual in many ways, the information here will hopefully be of some benefit to other security environments as well. New systems and usage scenarios are a regular feature on most large networks, and these changes to the security environment represent a major challenge to network security staff.

I will use the recent arrival of Apple's "MacOS X"[1] operating system as an example of a new platform that might require some forethought and planning from the security staff of a large research university. I will review some of the security risks MacOS X is likely to pose in such an environment, and hopefully shed some light on the possible effects the deployment of MacOS X systems might bring to a campus network.

Apple Computer's most recent operating system was released on March 24th, 2001. This product is a radical departure from earlier versions of MacOS, and is based on a combination of the Mach kernel (3.0) developed at Carnegie Mellon University, and operating system services based on 4.4BSD Lite 2 and FreeBSD. MacOS X layers a user friendly Macintosh-style GUI over a Unix kernel, to generally positive effect. This release represents the future of Apple operating systems, and is not a niche product. The target audience is the ordinary Mac user, and while it will be some time before a majority of existing customers will have migrated, the direction is clear, and Apple is aggressively moving in this new direction. (The upcoming MacOS 9.2 release is assumed to be the final "Classic MacOS" product)

A note on names: Web research will reveal quite a bit of information relating to "MacOS X". However, whether the author means to reference the original "MacOS X Server", the "Public Beta" release, or one of the several developer builds that were released, can often only be inferred by the date of the article and context. For the purposes of

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

this discussion I have limited myself to the publicly-available commercial product, patched to the latest level (as of this writing, 10.0.3). Where I reference earlier releases I will note the exact versions involved.

Who will use MacOS X?

A good starting point in any threat assessment is to determine who the players are. Who will use the system in question, and who else is likely to be affected by security situations involving it? At the moment, and probably for the near future, two types of users will be most likely to run MacOS X systems:

1) Those early adopters and "power users" already familiar with earlier MacOS versions (The so-called "Classic" environment), and interested in trying "the next thing".

2) The UNIX-adept trying out a new take on BSD. Of these two groups, it's seems more likely that, initially at least, the MacOS crowd will be more likely to be consistent users, rather than experimenters. However, while many of these users will be highly experienced in the "Classic" Macintosh world, they may have had little exposure to Unix, particularly from a system administration or security perspective.

Both of the above sets of users are unlikely to utilize MacOS X as a primary desktop, at least not initially. The reason for this is that at the time of this writing (Spring/Early Summer 2001), many services desired by the Macintosh-savvy are not yet available, or still in beta release. For those approaching MacOS X from the solely UNIX-savvy perspective, enough desired tools or services are similarly unready or must be built manually to cause very few such users to make MacOS X their primary environment.

As for those who might be affected by security problems on MacOS X, the easy answer for most such situations remains: "Anyone on the network anywhere". Remote compromises, and DOS attacks originating from compromised systems, is probably the most common observed security scenario on the network.

What is the environment of the potential threat?

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

First, let me discuss the local environment and my assumptions. The MIT network, like the network at some of the other large research schools, has historically not been firewalled from the Internet at large.[2] The reasons for this range from a local focus on host-based security, to avoiding restrictions that might interfere with computer and networking research.

The MIT network consists of approximately thirty thousand individual hosts, most of which are not maintained by "professional" system administrators. These are largely laboratory and desktop systems belonging to various staff, faculty, and students. In general, system security is not the highest priority of these users, whose focus is on individual research and academic efforts. However, most of these individuals consider themselves to be skilled users to some degree or another, so they frequently run programs and host services on these machines beyond those utilities that ship with the base operating system in question. It is very common for a machine at MIT to be a Web server, a mail server, and an ftp server, even simple desktop machines in student dorm rooms. One aspect of the local security landscape is that these systems are infrequently updated to address newly discovered security concerns.

My assumptions are as follows:

• Most initial users of MacOS X will be upgrading from previous versions of MacOS on systems they control, rather than being entirely new to the Macintosh. Apple computer has only recently (May 21, 2001) begun shipping new systems with MacOS X. These systems are set up to dual boot, with Classic being the default.

• Those users already adept at UNIX, but not currently MacOS users, will likely not represent any new security concerns, apart from MacOS X-specific problems. In other words, they will create no more security problems with MacOS X than they would running any other BSD-based system, unless an existing service specific to MacOS X carries some new vulnerability.

• Most of these users will consider themselves to be reasonably expert in the operation of "Classic" MacOS. (Version 9 and earlier systems)

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

• Users will be likely to gravitate to the services and functionality that are "new" to the Mac. These include the ability to create user accounts, run a web or ftp server, and other functions more common to the multi-user UNIX world.

So while the above network services have been available on earlier versions of MacOS in one form or another for some time[3], their deployment has before now been uncommon, and generally not been of great concern.

So we are faced with a situation where users will be running services they may or may not fully understand, but which are implemented on an operating system they are not yet familiar with. This represents not only the traditional threat associated with such service vulnerabilities as may arise, but also the concern that users will enable services and access mechanisms in their use of the system that will expose the system to abuse or compromise. It has been all too common in these situations for users to enable web or ftp servers, and then forget that they are active, neglecting to restrict access or apply needed patches.

System Updates and Patching

Updating running systems with security patches and updates is a significant challenge in our environment. Part-time system managers tend to neglect updating their systems on a regular basis, and this often results in large numbers of systems falling victim to recently disclosed vulnerabilities. This is a major issue in a university environment. In recent releases several vendors have worked to address this problem, either by adding update subscription services or manual update mechanisms. Recent Windows and Red Hat Linux releases have taken steps along this line, with varying degrees of success.

Apple has continued to improve on a service first seen in the initial MacOS 9 release: a simple software update procedure. While individual packages are still available at the Apple Software Update site (http://asu.info.apple.com/), the bundled Software Update procedure is set up by default to check the installed system weekly against a list of the latest versions available from Apple, and offer the user a chance to download and update their system. (The service can also be

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

set up to automatically update without user intervention, and manual update checks are available at any time) Under MacOS 9.x systems this is controlled via the Software Update control panel.

The 'root' account in MacOS X

Apple has taken steps to provide a reasonably secure, as-installed environment for MacOS X. By default, the root account is disabled, and users must instead use sudo[4] to execute privileged commands. The first account created on the system will have administrator privileges, but the user of that account is still required to use sudo to execute root-level commands at the system prompt. (In the GUI control windows, the user if prompted by the system for the administrator password for required services)

It is possible to enable the root account for direct use, however, via the "Netinfo Manager" application. Many operations a typical user will attempt, such as installing software, will ask for this password, and it must be expected that many users will enable the root account for "ease of use". Dangers here range from bad password discipline for the root account, to users remaining logged in as root and unintentionally bypassing user-level safeguards, with a variety of obvious bad effects. It is important to remember that many MacOS X users will have little or no experience on true multi-user systems, and may be assumed to have an incomplete understanding of many system components and processes.

Knocking on doors: Nmapping a new MacOS X install

It is important in evaluating a new system to understand its default configuration. Open and listening network ports are always a significant question. The following is a basic TCP connect scan of a MacOS X system (version 10.0.3, default configuration)

```
bash# nmap wingnut.example.com

Starting nmap V. 2.53 by fyodor@insecure.org (
www.insecure.org/nmap/ )
Interesting ports on wingnut.example.com (10.10.0.187):
(The 1520 ports scanned but not shown below are in state:
closed)
Port       State       Service
```

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

```
111/tcp     open        sunrpc
764/tcp     open        omserv
767/tcp     open        phonebook
```

As the SANS top-ten list indicates, RPC services are of some real concern, and should be turned off entirely, if not necessary to the user. Remote exploits of these services have been all too common in the recent past.

It is likely that a "fully operational" MacOS X system might have such services as ftpd, httpd, and telnetd running. In such cases it is important that the version and configuration be checked for the existence of known vulnerabilities. Both the CERT Coordination Center (http://www.cert.org/) and Packet Storm (http://packetstorm.securify.com/) have searchable archives of vulnerabilities, alerts, and system patches. It is generally easy to determine if there are outstanding issues by doing a few simple searches at these and other such sites. One bright spot here is that Apple has used common code from FreeBSD and other efforts, so that alerts of a general nature, such as with the popular Apache web server, and be easily identified as relevant to MacOS X systems.

It is important to remember when scanning operational systems (as opposed to test installs such as used for this paper) that Nmap cannot distinguish between services intended to be run by the user, and those resulting from compromise. While reasonable assumptions may often be made, in an academic environment that includes computer and security research efforts, it is important to remember that odd or traditionally unwelcome services may be running intentionally for some legitimate purpose.

Hostile Visitations?

The MIT network is also regularly assaulted by hostile vulnerability scans and attack sweeps targeting a variety of systems. It is not unusual to see an attacker's sweep of our entire address space be poorly tolerated by a particular service or platform[5]. These presumably unintended negative effects have often resulted in hung services or crashed systems. So far at least, the small number of MacOS X systems actually deployed have not been observed to be so affected, but it is likely that the future will reveal some services which are susceptible to these side effects.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

Local user exploits

Whether by misuse by legitimate users, or an intruder compromise of account passwords, local exploits are of concern as well. It is commonly assumed that once a user (legitimate or otherwise) has a local account on a system, proceeding to a root-level compromise is simplified. There have been a number of system vulnerabilities in common services that allow users to escalate privileges and gain root access.

In my searching, I was only able to find one MacOS X-specific local exploit. "Malevolence"[6] is a local user exploit for MacOS X that allows a user to capture an unshadowed version of the /etc/password file. This allows the user to attempt to crack the passwords for other local accounts on the machine. Obviously, if the root account has been enabled by the user, it too is subject to compromise in this way.

Remote Exploits

In a university environment, without a border firewall to hold off intruders, remote exploits are of great concern. Such networks are subjected to automated vulnerability scans which can download a specific exploit as soon as an appropriate system is found by the scan. In this way, an intruder can compromise large numbers of systems quite rapidly. The only protective mechanism available to most system administrators in this scenario is to constantly monitor and update their system software with all relevant system and service releases.

Apple has acknowledged two remote exploit vulnerabilities since the release of version 10.0.0[7]: An NTP buffer overflow vulnerability, described in FreeBSD-SA-01:31, and an FTP File Globbing vulnerability described in CERT® Advisory CA-2001-07, both patched in release 10.0.2. Like other system updates, this release was available automatically via the Software Update service, which by default will check for and install available updates weekly.

Conclusion

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

The process of evaluating a security situation will vary somewhat in each instance, but will always include a core set of questions:

- Who are the likely participants?
- Who else may see related security effects?
- How does this new situation differ from the existing conditions?
- What are the likely vulnerabilities introduced? What are the fringe cases?
- How does the new situation effect existing practices and systems? Are current systems threatened?
- What new remote vulnerabilities are likely?
- What new local vulnerabilities are likely?
- What mechanisms are available to mitigate new risk?

Or in the shortest possible form:
What has changed? And what does it mean?

For the example of MacOS X on the MIT environment, I feel that the risk is acceptable to MIT[8]. Secure tools available for common UNIX systems (ssh, kerberos, etc) are available for or included with MacOS X. Update mechanisms are generally superior to those in common use on other local platforms. General Mac and UNIX support/education efforts on campus are available to support user questions and concerns. While some increased user education will be required for those users unfamiliar with the characteristics of multi-user systems, the overall cost associated with supporting this platform for secure use on MITnet is in line with other local platforms, and may in fact provide important lessons in how such change is reconciled with existing support and security practices.

---

[1] Pronounced "Ten". See MacOS X home at Apple Computer: http://www.apple.com/macosx/

[2] The "Three Myths of Firewalls": http://web.mit.edu/kerberos/www/firewalls.html

[3] A search via http://www.versiontracker.com/ will reveal several ftp servers for MacOS Classic, and "Personal Web Sharing" appeared in MacOS 9.1. Other examples abound.

[4] Sudo home page: http://www.courtesan.com/sudo/

[5] This behavior is not unknown to those doing such scans. He author of http://www.sans.org/infosecFAQ/audit/nmap.htm notes in the section "Implications of using Nmap to scan networks" that some older systems on his network crashed in reaction to nmap scanning. The MIT network has seen many similar examples, including crashed printers and even networked HVAC controller gear. (The HVAC vendor, when questioned, revealed that system security had not been a design concern…)

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

[6] While the author's web site is currently unavailable, http://packetstorm.securify.com/ has a copy of the exploit.
[7] A listing of vulnerabilities and links to their patches can be found at: http://www.apple.com/support/security/security_updates.html
[8] Not that I have any control as to what systems are deployed here…

Sources used

SANs Institute, How To Eliminate The Ten Most Critical Internet Security Threats The Experts' Consensus, v 1.32 URL: http://www.sans.org/topten.htm (January 18, 2001)

Mac OS X FTP Vulnerability? (TidBITS#577/23-Apr-01), http://www.tidbits.com/tb-issues/TidBITS-577.html

Mac OS X: The Future Is Here - Coming Soon! (TidBITS#573/26-Mar-01), http://www.tidbits.com/tb-issues/TidBITS-573.html

Mac OS X 10.0.2 Fixes FTP Vulnerability (TidBITS#597/07-May-01), http://www.tidbits.com/tb-issues/TidBITS-579.html

Security Alert: Updating sudo on Mac OS X 10 http://www.stepwise.com/Articles/Workbench/2001-05-01.01.html

The Challenges of Integrating the Unix and Mac OS Environments (USENIX 2000 Invited Talks Presentation) http://www.mit.edu/people/wsanchez/papers/USENIX_2000/

Apple Computer Product Security Incident Response http://www.apple.com/support/security/security_updates.html

Things Macintosh http://magicpubs.com/mac/internet.php

MacOS X a Radical Departure from Its Predecessors http://cc.uoregon.edu/cnews/spring2001/osx.html

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.