MAC OS X VERSION 10.5 "LEOPARD"

Graduate Student: **Alexandru Gavril Bardas**
James Madison University
Computer Science - Secure Software Systems

Overview

Mac OS X version 10.5 "Leopard" is the sixth major release of Mac OS X. This operating system is the successor of Mac OS X v10.4 "Tiger". Leopard was released on 26 October 2007, and is available in two variants:

- a desktop version (for personal computers)
- a server version (Mac OS X Server)

Leopard only builds on the successes of previous Mac OS X versions. There is no need to worry about a lot of viruses, spyware, or service pack releases. The system may go for months or years without a system crash (Wikipedia, 2008c).

This happens because the translucent desktop of Mac OS X is Unix, the industrial-strength, rock-solid OS that drives many a Web site and university. Unix is not new at all, it is decades old, and has been polished by generations of programmers (Pogue, 2007). David Pogue states that this is the very reason why Steve Jobs, Apple's current CEO, and his team chose it as the basis for the NeXT operating system (that Jobs worked on during his twelve years away from Apple), which Apple bought in 1997 to turn into Mac OS X.

Steve Jobs announced at *Macworld 2008* that over 20% of Macs use Leopard as their operating system. According to Apple, the new operating system contains over three hundred changes and enhancements, covering core operating system components as well as included applications and developer tools.

Why is Mac OS X version 10.5 called Leopard?

Generally software companies develop their products in secret, using code names for new products to throw outsiders off the scent. Apple's code names for Mac OS X and its descendants have been named after big cats: Mac OS X was Cheetah, 10.1 was Puma, 10.2 was Jaguar, 10.3 was Panther, and 10.4 was Tiger. Mac OS X 10.5 is called Leopard (Pogue, 2007).

The code name is dropped as soon as the product is complete and the marketing department comes up with a new name. For instance Microsoft's Windows Vista was called Longhorn during the development process. In Mac OS X's case, though, Apple thinks that it can retain the cat names for the finished product. In 2009 Apple plans release Mac OS X v10.6 "Snow Leopard".

New or Improved Features in Leopard

Apple states that it added over 300 new features. They created even a website where they present the added features (http://www.apple.com/macosx/features/300.html).

The added features are however not "all equally breathtaking" (Pogue, 2007). Apple includes among the features the new built-in fonts (for example, Arial Unicode, Microsoft Sans Serif, Tahoma, Papyrus Condensed, and Wingdings), a Polish spelling checker, a **Portuguese** spelling checker, or the Arabesque screen saver.

Other examples of more important modified or new features in Mac OS X 10.5 are:

- The Time Machine is a real breakthrough. It keeps the entire Mac backed up, 24 hours a day. Everything: files, folders, settings, photos, email, programs, even Mac OS X itself. It's automatically enabled and after the first back-up that can last up to a few hours then the update of the back-up happens automatically. The user does not even know that the files that he is working on have been added to the back-up.
- Quick Look enables tapping the Space bar to view a highlighted document at full size, right at the desktop, without having to open the program that created it.
- The Spaces feature allows the user to work on two, four, eight, or 16 full-size virtual, external monitors.
- The enhanced Parental Controls let user set time limits for children's computer use and even make the Mac lock itself at bedtime. A log tracks the children's activities, including sent emails and visited Web sites.
- Screen Sharing over the network or Internet gives the users the possibility to control the other systems as if they were in front of those systems.
- The iChat text/voice/video chat program has some new features that give users a lot of options when having a video conference. Users can display documents, presentations, or movies to other users. Screen sharing is available too, so that users can assist friend users across the globe.
- The new Safari Web browser offers resizable text boxes on Web pages; a PDF viewer right in the browser window and more flexibility with tabbed windows.
- Mail now offers To Do lists and Notes (which also appear in the iCal calendar), stationery templates, RSS feeds, and Data Detectors, which find phone numbers, addresses, and dates in your email and offer to put them into your Address Book and calendar (Apple Inc., 2008d).
- Other Mac OS X programs have all been revamped, too. Humble little Preview offers now the possibility to cut pieces out of one photograph and paste them into another, just like Photoshop. TextEdit includes autosave, auto curly quotes, and auto Web links.
- Stacks are arcs or grids of icons that spring out of a Dock folder when the user selects it, making it easy for users to see what's inside. Unfortunately, only a relatively small number of icons appear (Pogue, 2007).
- The see-through menus are another visual effect that usually looks good. However sometimes it is hard to read because of certain backgrounds.
- Classic is gone. Users can no longer run Mac OS 9 programs in Leopard (Pogue, 2007).

- Boot Camp supports the most popular 32-bit releases of Windows XP and Windows Vista. Windows applications will run at native speed and they have full access to multiple processors and multiple cores, accelerated 3D graphics, and high-speed connections like USB, FireWire, Wi-Fi, and Gigabit Ethernet (Apple Inc., 2008d).
- The Dictionary program is able now to search on Wikipedia.
- A dozen of new keyboard shortcuts in the Finder make navigating without the mouse even faster and easier.
- Redesigned panels of System Preferences incorporate the functions of what used to be confusing little add-on programs (like Internet Connect and Printer Setup Utility).

There are a lot of more changes in Leopard but most of them are only improved older features from the previous version of OS X.

The Technology used in Leopard

XNU

XNU is the computer operating system kernel that Apple Inc. acquired and developed for use in the Mac OS X operating system and released as free and open source software as part of the Darwin operating system.

Originally XNU was developed by NeXT for the NEXTSTEP operating system. It was a hybrid kernel combining version 2.5 of the Mach kernel developed at Carnegie Mellon University with components from 4.3BSD and an object-oriented API for writing drivers called Driver Kit (Wikipedia, 2008a).

After Apple acquired NeXT, the Mach component was upgraded to version 3.0, the BSD components were upgraded with code from the FreeBSD project and the Driver Kit was replaced with a C++ API for writing drivers called I/O Kit.

XNU is a hybrid that contains features of both monolithic and micro kernels. It attempts to make the best use of both technologies, such as the message passing capability of micro kernels enabling greater modularity and larger portions of the OS to benefit from protected memory, as well as retaining the speed of monolithic kernels for certain critical tasks (Wikipedia, 2008a).

The core of the XNU kernel, Mach, was originally conceived as a simple microkernel. In this context it is able to run the core of an operating system as separated processes, which allows a great flexibility because one computer could run several operating systems in parallel above the Mach core. Unfortunately the performance is often reduced because of the time consuming kernel/user mode context switches and overhead stemming from mapping or copying messages between the address spaces of the microkernel and that of the service daemons. With Mac OS X, the designers have attempted to streamline certain tasks and thus BSD functionalities were built into the core with Mach. The result is a combination of Mach and a classical BSD kernel.

Mach provides kernel threads, processes, pre-emptive multitasking, message-passing (used in inter-process communication), protected memory, virtual memory management, kernel debugging support, and console I/O (Amit, 2006).

The Berkeley Software Distribution (BSD) portion of the kernel provides the POSIX API (BSD system calls), the Unix process model atop Mach tasks, basic security policies, user and group ids, permissions, the network stack, the virtual file system code, Network File System (NFS), cryptographic framework, UNIX System V inter-process communication (IPC), Audit subsystem, Mandatory Access Control and some of the locking primitives (The FreeBSD Foundation, 2008).

I/O Kit is the device driver framework, written in a subset of C++. Using its object-oriented design, features common to any class of driver are provided within the framework itself. Therefore device drivers are written more quickly and using less code. The I/O Kit is multi-threaded, symmetric multiprocessing (SMP)-safe, and allows for pluggable devices automatic, dynamic device configuration.

Many drivers can be written to run from user-space. This fact further enhances the stability of the system; if a user-space driver crashes, it will not crash the kernel. However, if a kernel-space driver crashes it will crash the kernel. Examples include Parallels, EyeTV and the Apple USB driver.

Privileges modes

Modern processor architectures have CPU modes that allow the OS to run at different privilege levels. Therefore tasks are tagged with a privilege level. Resources (for example segments, pages, ports) and privileged instructions are tagged with a demanded privilege level. When a task attempts to use a resource, or execute a privileged instruction, the processor determines whether it has the permission. In case it has not the required permission a "protection fault" interrupt is generated. This prevents user tasks from damaging the OS or each other (Apple Inc., 2008a).

Mac OS X, including Leopard, inherits its permissions model from UNIX. Apple has enhanced this security model by disabling the *root* account by default, “running with least privileges” method. By running code with the minimum necessary level of privileges, Mac OS X helps protect the system from inadvertent or deliberate damage.

Mac OS X has three types of user accounts, three privileges levels:

User

The user account is the account that has the least privileges in the Mac OS X system. The user can modify settings only for his or her account, not for the entire system. It is considered a good security practice to have all users operate at this level of permissions. If further privileges are required to install software or modify system settings, an administrator can be authenticated when needed. Also additional limits can be placed on user accounts to prevent them from: opening System Preferences**,** removing items from the Dock**,** changing passwords etc. (Apple Inc., 2008a).

These limits can be managed using either parental control in Leopard or managed preferences in Leopard Server.

Administrator

Mac OS X establishes an administrator user account when the system is first installed. An admin user can perform most of the operations normally associated with the root user, except directly adding, modifying, or deleting files in the system domain. However, an administrator can use the Installer or Software Update applications for this purpose (Apple Inc., 2008a)..

Root

Mac OS X (like most UNIX operating systems) has a superuser, named *root*. The superuser has full permissions to access all files on the system.

Unlike traditional UNIX systems, this account is disabled by default. This precaution helps to limit the extent of harmful changes that viruses or unauthorized users could make to the operating system (Apple Inc., 2008a).

In addition to user accounts, Mac OS X uses less privileged system accounts for some system services and software that require specialized access to certain system components, but not login access. To prevent unauthorized users from altering the system in an undesirable way, new users do not have administrative privileges unless assigned to them by the administrator. As users are added to the system, Mac OS X assigns them nonadministrative user accounts and prompts them to choose a password, providing means of authentication.

In Leopard, privileged access (such as use of the *sudo* command) and remote access are not allowed for users with no password.

Preemptive multitasking

Multitasking means "doing more than one thing at once." Most probably computer users will say that for years, Macs have been capable of making a printout, downloading a file, and letting them type away in a word processor, all at the same time.

Mac OS 7/8/9 (and Windows 95/98/Me) version of multitasking work by the “rule of the playground”(Pogue, 2007). If one of the running programs insists on hogging the attention of the processor (for example the program is crashing), “it leaves the other programs gasping for breath”. This kind of arrangement is called cooperative multitasking. It works fine only if the running programs are in fact cooperating with each other (which is not the case when one of the programs is crashing).

Mac OS X's preemptive multitasking system “brings a teacher to the playground to make sure that every program gets a fair amount of time from the Mac's processor” (Pogue, 2007). The result is that the programs will have a certain time to use the resources that it needs, for instance a crash of one of the programs does not block the resources needed by other programs.

Multithreading

Multithreading can be described as "doing more than one thing at once," too, but in this case it's referring to a single program. However not all programs will be able to use this feature because of the fact that a lot of programs are being “carbonized”. That means that the software companies put some effort into getting with Mac OS X program. They simply adapted, or updated their existing software so that it works with Mac OS X. The resulting software looks and feels almost like a true Mac OS X program, it contains crash protection, the good looks, the Save sheets etc., but behind the scenes, the bulk of the computer programming is the same as it was in Mac OS 9.

Multithreading will be most likely used by “Cocoa” programs, programs that were written from scratch exclusively for Mac OS X.

Symmetrical multiprocessing

Nowadays Mac computers offer astounding performance with up to eight cores of processing power. But before Mac OS X, only specially written software, for example Adobe Photoshop filters benefited from the speed boost.

Mac OS X automatically capitalizes on multiple processors, sharing the workload of multiple programs (or even multithreaded tasks within a single program). That means that every Mac OS X program gets accelerated. Apple claims that the new Leopard scheduler is very efficient at allocating tasks across multiple cores and processors. So Leopard spends less time managing tasks and more time performing computations.

Apple engineers overhauled the operating system's "thread scheduler" to improve performance. In addition, "processor affinity" now attempts to keep a thread running on the same processor to improve cache performance and let the thread spend less time switching back and forth.

What changes these changes mean for the user?

Substantial improvement will be noticed in both speed and system responsiveness. Single processor systems seem to benefit from more granular execution of well-threaded system functions, as well; tests showed that Mail and Spotlight in particular were quicker and more responsive on a PowerBook G4 than they were in Tiger.

64 –Bit

Mac OS X 10.4 added limited support for 64-bit memory addressing, which is very important when working with huge data sets used in science, research, and even some kinds of graphics manipulation. This support was limited to command-line applications.

The Apple-recommended workaround was the creation of a 32-bit GUI application that communicated with a 64-bit command line application. This was not a convenient solution for developers and users.

Leopard is fully 64-bit capable (assuming that the processor is too) and includes the graphical interface. Every supporting Mac OS X Cocoa framework and library is available in both 32-bit and 64-bit mode. Developers no longer have to painstakingly combine separate 64-bit and 32-bit programs, and they

can take full advantage of Leopard's high performance 64-bit math libraries. This all makes it faster and easier to develop 64-bit applications.

Memory allocation & memory protection

The memory allocation in OS X is dynamic. Mac OS X programs don't have fixed RAM allotments. The operating system allocates memory for programs in real time, so that no RAM is wasted. For a Mac OS X user this system means better stability, less hassle (Pogue, 2007).

Memory protection can be described as the fact that every program runs in its own “indestructible memory bubble” (Pogue, 2007) in Mac OS X. This is one of the reasons why Mac OS X is so much more stable than its predecessors. If one program crashes, it isn't allowed to poison the well of RAM that other programs might need to use. Programs may still freeze or quit unexpectedly, but instead of encountering a message that says, "Save open documents and restart," the user will be happy to find out that he can go right on working. The user can even open up the program that just died and get back to work.

Core Animation

Core Animation is described by Apple as being a new technology that in used in Leopard. Core Animation is actually a framework that makes it simple for Mac developers to add visually stunning user interfaces, graphics, and animations to applications.

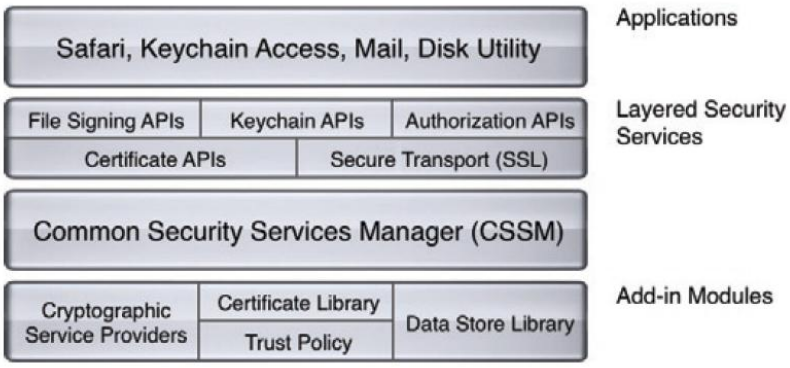
UNIX certification

Leopard is an Open Brand UNIX 03 Registered Product, conforming to the SUSv3 and POSIX 1003.1 specifications for the C API, Shell Utilities, and Threads. Since Leopard can compile and run all existing UNIX code, it can be deployed in environments that demand full conformance — complete with hooks to maintain compatibility with existing software.

Security in Leopard

Security Architecture in OS X

Mac OS X security services are built on the Common Data Security Architecture (CDSA), with support for cryptography, certificate management, trust policy management, and key recovery. It is a layered security infrastructure that makes it easy for Apple and Mac OS X developers to integrate leading-edge security features, such as authentication and encryption, into their applications.



Apple claims that security has never been a more important consideration when selecting a computer platform.

Leopard contains new security features that intend to provide better internal resiliency to successful attacks, in addition to preventing attacks from being successful in the first place.

It has a radical new approach to network firewalling, a new application-signing infrastructure and also memory randomization and sandboxing can make a big difference in preventing security exploits from getting anywhere.

Library Randomization

Leopard implements library randomization, which randomizes the locations of some libraries in memory. Vulnerabilities that corrupt program memory often rely on known addresses for these library routines, which allow injected code to launch processes or change files (Rich, 2007).

Application Layer Firewall

Leopard ships with two firewall engines: the original BSD IPFW (the initialism of Indiana University-Purdue University Fort Wayne), which was present in earlier releases of Mac OS X, and the new Leopard Application Layer Firewall. ALF (Application Layer Firewall) provides more fine-grained security than the simple packet-filtering firewall built into Tiger.

Unlike IPFW, which intercepts and filters IP datagram before the kernel performs significant processing, the Application Layer Firewall operates at the socket layer, bound to individual processes.

The Application Layer Firewall can therefore make filtering decisions on a per-application basis. Of the two-firewall engines, only the Application Layer Firewall is fully exposed in the Leopard user interface. The new firewall offers less control over individual packet decisions (users can decide to allow or deny connections system wide or to individual applications, but must use IPFW to set fine-grained TCP/IP header level policies). It also makes several policy exceptions for system processes: neither mDNSResponder (Bonjour feature process) nor programs running with superuser privileges are filtered (Wikipedia, 2008c).

Sandboxes

Leopard includes kernel-level support for role-based access control (RBAC). RBAC is intended to prevent, for instance, an application like Mail from editing the password database (MacInTouch, 2007).

Application Signing

Leopard provides a framework to use public key signatures for code signing to verify, in some circumstances that code has not been tampered with. Signatures can also be used to ensure that one program replacing another is truly an "update", and carry any special security privileges across to the new version. This reduces the number of user security prompts, and the likelihood of the user being trained to simply clicking "OK" to everything (Wikipedia, 2008c).

Secure Guest Account

Guests can be given access to a Leopard system with an account that the system erases and resets at logout (Wikipedia, 2008c).

Conclusion

Mac OS X v10.5 Leopard is the sixth release of Mac OS X in the last seven years. It is the richest version of Mac OS X to date. According to Apple is has over three hundred new features and delivers numerous innovations that make the Mac even more enjoyable to use.

Also the security features in Mac OS X provide solutions for securing data at all levels—from the operating system to applications and networks.

Furthermore, Leopard is built on new advanced software technologies that take full advantage of the latest hardware.

All in all Mac OS X v10.5 Leopard is a well achieved operating system.