Unit 7: Operating System Security

Functions of an Operating System (OS):

1) It provides a number of interfaces;
   - To the user - Graphical User Interface (GUI) or Command Line Interface (CLI)
   - To applications - Application Programming Interface (API)
   - To the hardware – via device drivers.
2) It provides resource management and scheduling
3) It provides an abstraction or virtualisation layer
4) It provides services and security for users and application

Resource Management

Resource management has several aspects namely: memory management, process management, energy management, communication management and File management. The operating system Kernel is responsible for these aspects

There are 3 classical models for a kernel – monolithic, micro-kernel and exo- kernel. Most modern commercial OS models tend to adopt a hybrid approach.

The main mechanism for resource allocation is the process which consists of a block memory that contains information. A process consists of one or more threads and are managed by the OS using process control blocks. Each process can have on or more threads and these threads are used to schedule the execution of the process. Threads are tracked and managed by the OS.

Different process states as assigned by the OS are:

New: when the application is converted from its on-disk state to an executable process.
Ready: then status is set to ready
Waiting: when it is added to the queue waiting for the schedule to allocate CPU time
Running: when it is loaded into the CPU (stays in that state until either the time-slice expires or input/output is requested)
Terminated: when it has finished its task or a user quits the application

A time slice is the amount of CPU time allocated to each task. There are different types of algorithms used for time allocation. These include: First come first served, priority, round robin, multi-level queues

Abstraction/Virtualisation
OS virtualisation is a very common concept here

Service and Security

Brief History of Operating Systems
In the 50S, security was not much of a concern. To infiltrate the system then, one would need physical access.

The Compatible Time Sharing System (CTSS) - first interactive operating system developed at MIT in 1961; it was compatible with batch programs that could also be running in the background on the same hardare

1964 – IBM system S/360 ran applications across different models and generation of systems. One of the first recorded 'worms' - the CHRISTMA worm started on IBM mainframes. It was written in one of the first shell/ macro languages – REXX.

Multics was later developed

1969 – Unix was developed; a descendant of Multics. Security concerns associated with Unix can be classified into 3 categories: user expectations, software quality and add-on integration. Open source projects like GNU have re-written large portions of Unix to address security concerns.

1975 – Control program for Microcomputers (CP/M), first operating system for the microcomputers was developed

1989 – The GUIs. Apple produced some of the first mainstream GUI OS such as the LisaOS and the MacOS. GUIs increased the attack surface of systems as It meant running of applications/services that has several weaknesses that an attacker could exploit.

1998 – Mobile Operating Systems
SymbianOS was the main OS used for smartphones powering more than 60% of handsets sold in 2006. Later came AndroisOS which now has more than 70% of the market and IOS about 26%. Security concerns with smartphones are the bluetooth and Wi-Fi communication protocols. Information could be intercepted when transmitted via these protocols. Phones can also be hacked.

Security Mechanism used in operating systems

Saltzer and Schroeder Principles

- Economy of Mechanism
- Least privilege
- Open design
- Fail safe defaults
- Separation of privilege
- Least common mechanism
- Complete mediation
- Psychological Acceptability

- Work factor
- Compromise Recordings