Parul® University
Vadodara, Gujarat

NAAC GRADE A++

Information and Communication Technology

Processes, Threads, Scheduling & Secure Execution

Prof. Khushal Bhoyar
Department of CSE (Cyber Security)
PIET, Parul University

2.1 Process Management Fundamentals

- A process represents a program in execution.
- It consists of:
- The executable code.
- Current activity.
- Associated system resources.

- Processes transition through various states:
- New (creation).
- Ready (awaiting CPU allocation).
- Running (executing on CPU).
- Waiting (awaiting event completion).
- Terminated (finished execution).

- **The Process Control Block (PCB)** serves as the kernel data structure containing all information about a process.

-cont

- **The PCB contains:**
- Process state.
- Program counter.
- CPU registers.
- Scheduling information.
- Memory management data.
- Accounting information.
- I/O status.
- **Context switching** involves saving the state of the current process and restoring the state of the next process to execute.
- **Context switching** requires careful management of:
- CPU registers.
- Program counters.
- Memory management data.

2.2 Thread Architecture and Implementation

- Threads represent the smallest unit of execution within a process.
- Threads share the process's:
- Code.
- Data.
- Resources.
- Threads have independent execution streams.
- **User-level threads:**
- Are managed by user-space libraries without kernel involvement.
- Offer fast creation and switching.
- Lack true parallelism on multiprocessors.
- **Kernel-level threads:**
- Are managed directly by the OS kernel.
- Enable true parallel execution across multiple CPUs.
- Have higher management overhead.

-cont

- **Multithreading benefits include:**
- Improved responsiveness (blocking one thread doesn't block entire process).
- Resource sharing (threads share memory and resources efficiently).
- Economy (thread creation cheaper than process creation).
- Scalability (better utilization of multiprocessor architectures).

Modern systems typically implement hybrid threading models combining advantages of both approaches.

2.3 CPU Scheduling Algorithms

- CPU scheduling determines which ready process receives CPU time.

It is based on specific scheduling objectives:

- Maximizing CPU utilization.
- Ensuring fairness.
- Minimizing response time.
- Maintaining system balance.

First-Come-First-Served (FCFS):

- Executes processes in arrival order.
- Simple but suffers from convoy effect.

Shortest Job First (SJF):

- Selects the process with smallest next CPU burst.
- Optimal for minimizing average waiting time.
- Requires accurate burst time estimation.

-cont

Round Robin (RR):
Allocates fixed time quantum to each process in circular order.
Ensures fairness with reasonable response times.

Priority scheduling:
Assigns priority levels to processes.
Executes highest priority ready process first.
Potentially causes starvation of low-priority processes without aging mechanisms.

Modern systems typically implement multi-level feedback queues.
These queues combine multiple algorithms with priority adjustments based on process behavior.

2.4 Security in Process Execution

Process isolation ensures that processes cannot interfere with each other's memory or execution.
It is implemented through:
Hardware memory protection.
Virtual address spaces.
Sandboxing creates restricted execution environments that limit system access for untrusted code.
Sandboxing uses techniques like:
System call filtering.
Resource quotas.
Namespace isolation.
Privilege separation divides programs into components with minimal necessary privileges.
This limits damage from compromised components.

-cont

Address Space Layout Randomization (ASLR) randomizes memory locations of key data areas to thwart memory corruption attacks.

Data Execution Prevention (DEP) marks memory pages as non-executable unless explicitly containing code.
DEP prevents execution of injected malicious code.
These mechanisms collectively form a defense-in-depth approach to process security.
This approach makes successful exploitation increasingly difficult.

1. https://www.druva.com/glossary/what-is-a-disaster-recovery-plan-definition-and-related-faqs

2. https://www.konverge.co.in/virtualization-in-cloud-computing-need-types-and-importance/

3. https://www.crowdstrike.com/cybersecurity-101/cloud-security/cloud-application-security/

Parul® University

NAAC GRADE A++

https://paruluniversity.ac.in/