# Introduction of Server

Servers are dedicated computers for a specific purpose, and unlike the Desktop computer these computers are made for durability, longevity and for working longer periods of time as compared to a desktop PC.

## Functions of a Server :

A server is made to provide services and functionality to other computers. The other computers which are using the server are called “clients” and this model of sharing is called “client-server” model.

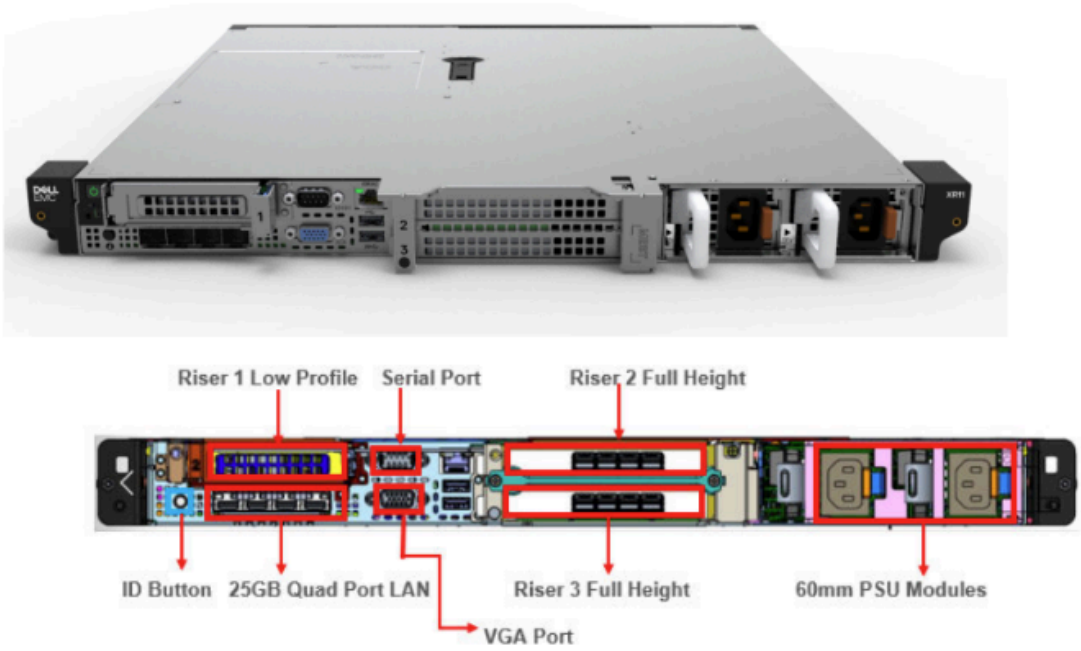
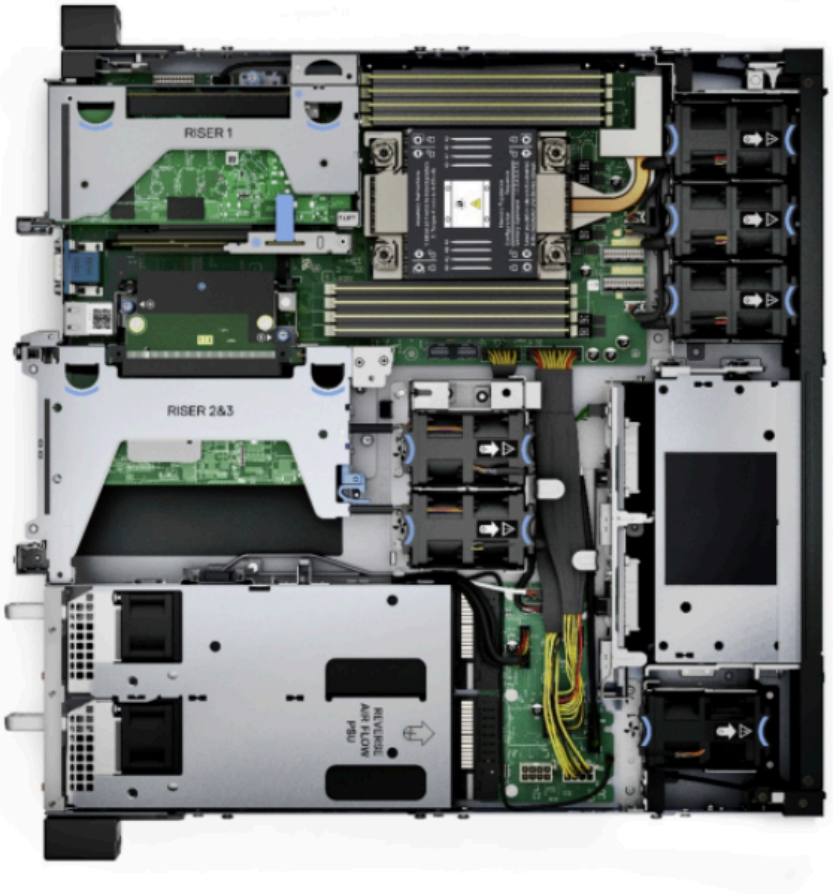
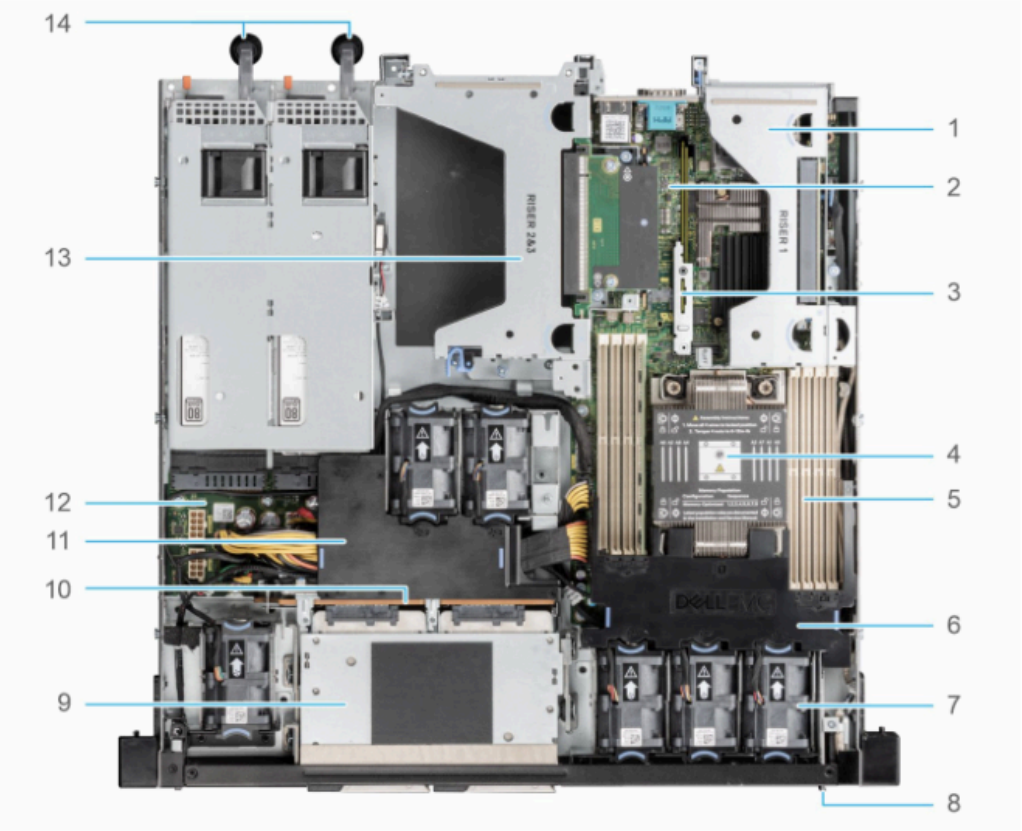



**Figure 1. XR11 Front accessed chassis front views**

## Inside the XR11 front accessed chassis



# Inside the system



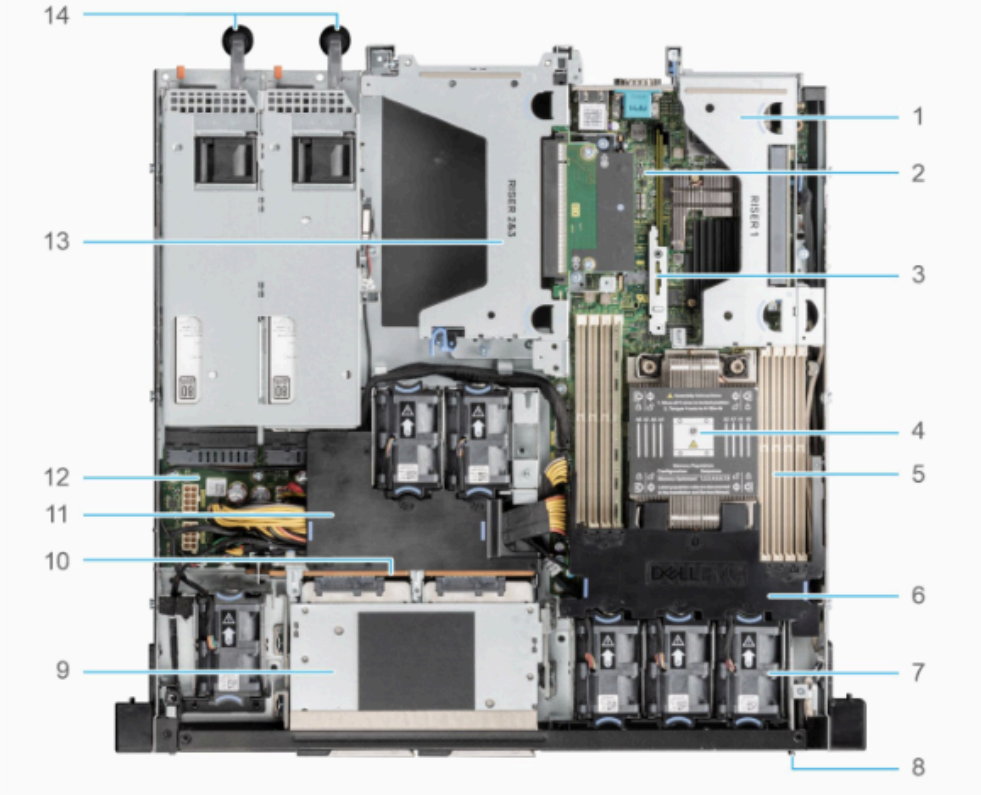



**Figure 8. Inside the system - Rear Accessed configuration**

1. Riser 1
2. System board
3. BOSS-S1 card (optional)
4. Processor and heat sink module
5. Memory module slots (8)
6. Processor air shroud
7. Cooling fans (6)
8. Information tag
9. Drive cage
10. Backplane
11. PCI air shroud
12. Power Interposer Board (PIB)
13. Riser 2 and 3
14. Power supply units

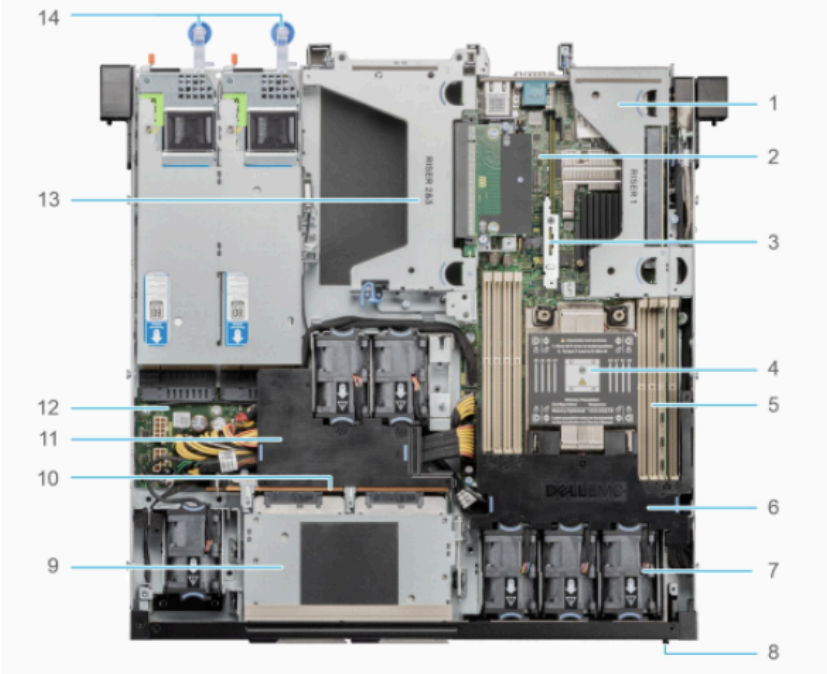
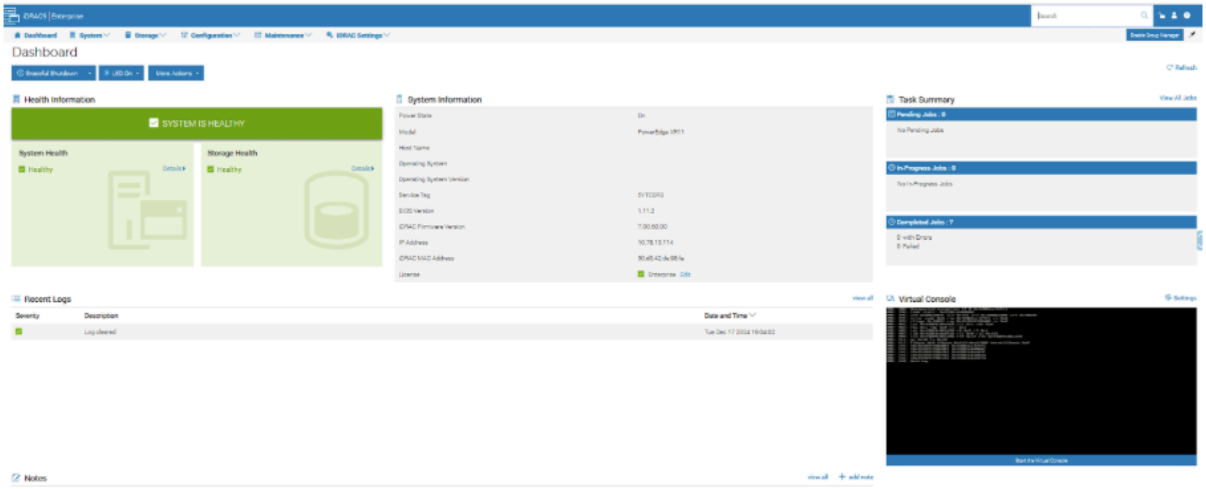
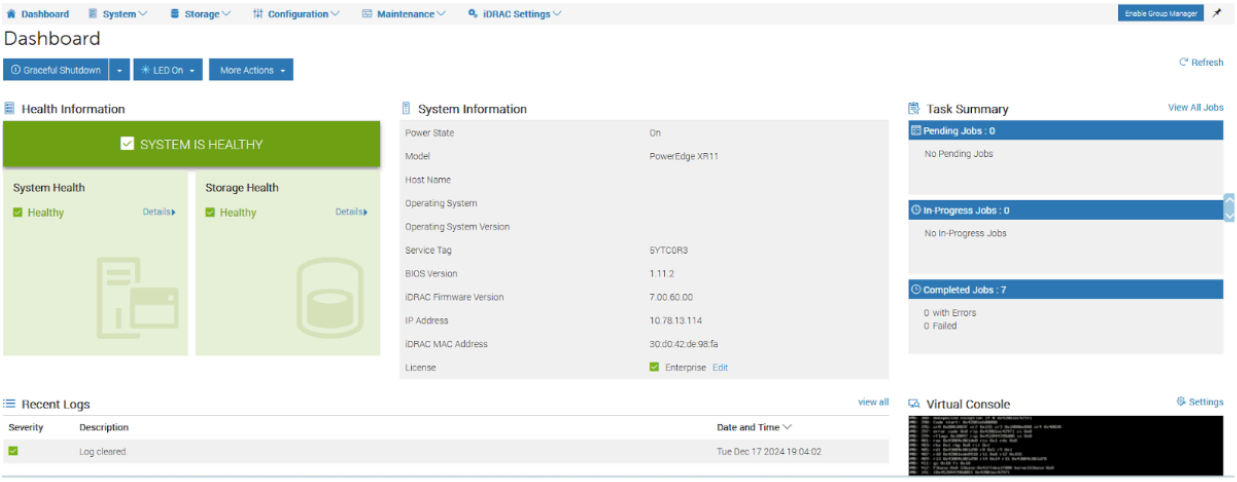
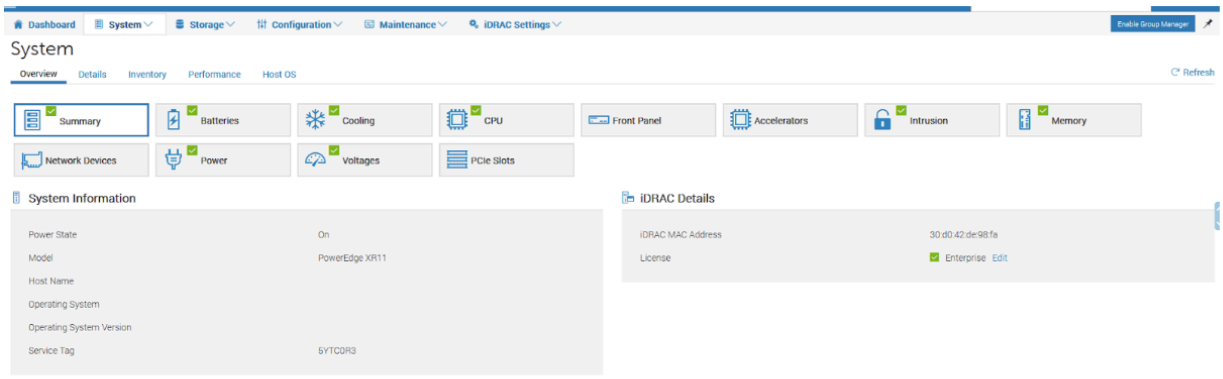


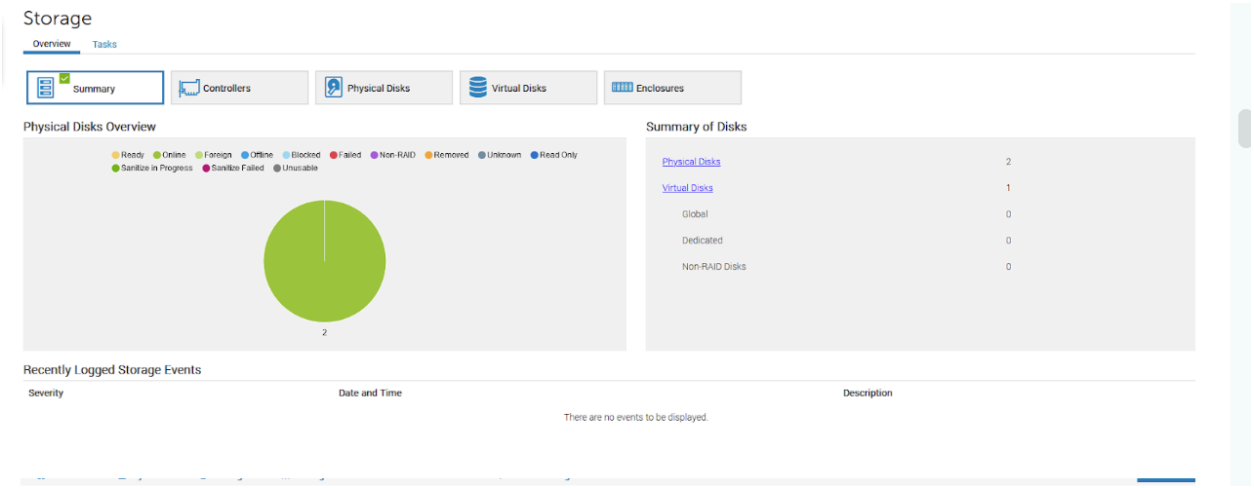

**Figure 9. Inside the system - Front Accessed configuration**

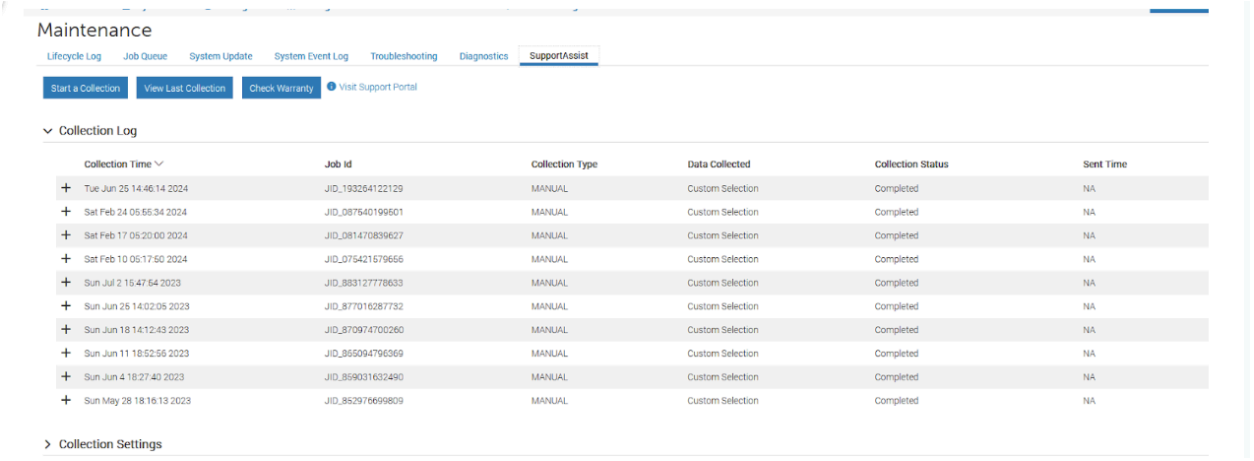
1. Riser 1
2. System board
3. BOSS-S1 card (optional)
4. Processor and heat sink module
5. Memory module slots (8)
6. Processor air shroud
7. Cooling fans (6)
8. Information tag
9. Drive cage
10. Backplane
11. PCI air shroud
12. Power Interposer Board (PIB)
13. Riser 2 and 3
14. Power supply units

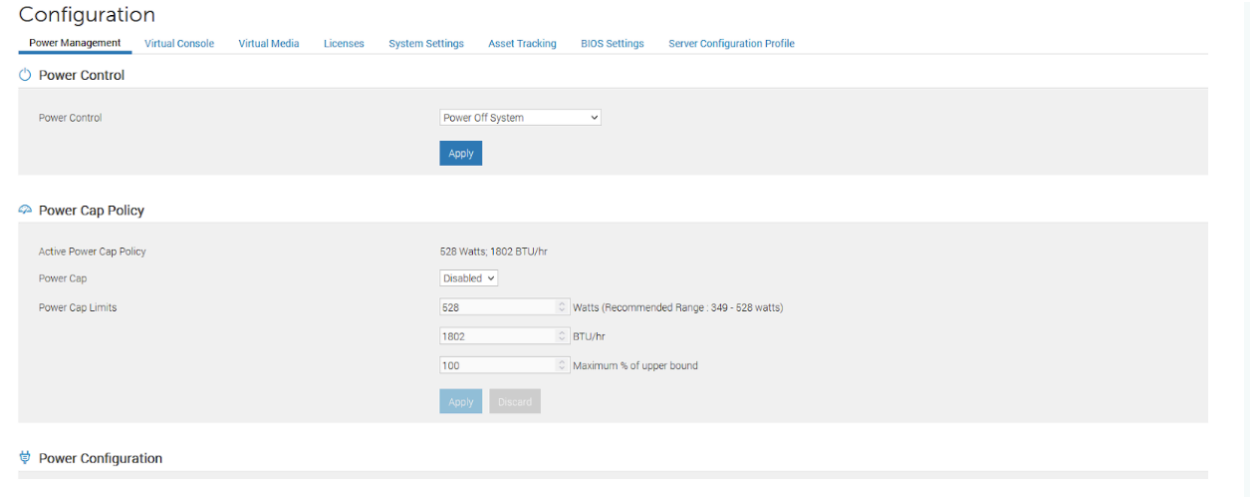


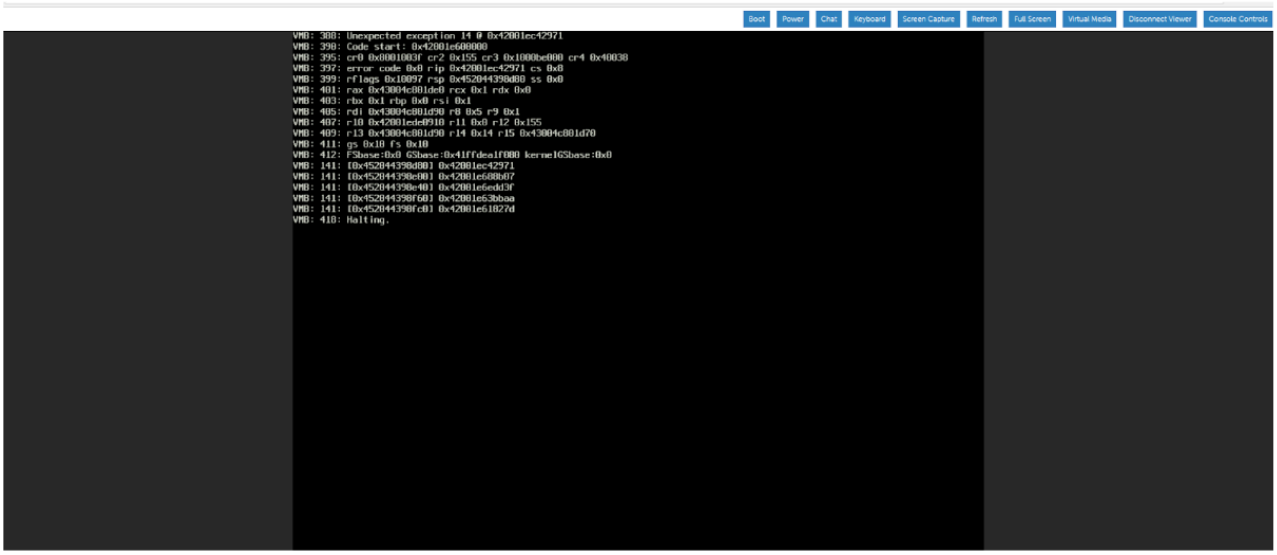

Dashboard
System
Storage
Configuration
Maintenance
iDRAC Settings
Enable Group Manager
Dashboard
Graceful Shutdown
LED On
More Actions
Refresh
Health Information
SYSTEM IS HEALTHY
System Health
Healthy
Details
Storage Health
Healthy
Details
System Information
Power State On
Model PowerEdge XR11
Host Name
Operating System
Operating System Version
Service Tag 5YTC0R3
BIOS Version 1.11.2
iDRAC Firmware Version 7.00.60.00
IP Address 10.78.13.114
iDRAC MAC Address 30:d0:42:de:98:fa
License Enterprise Edit
Task Summary
View All Jobs
Pending Jobs : 0
No Pending Jobs
In-Progress Jobs : 0
No In-Progress Jobs
Completed Jobs : 7
0 with Errors
0 Failed
Recent Logs
view all
Severity
Description
Date and Time
Log cleared.
Tue Dec 17 2024 19:04:02
Virtual Console
Settings

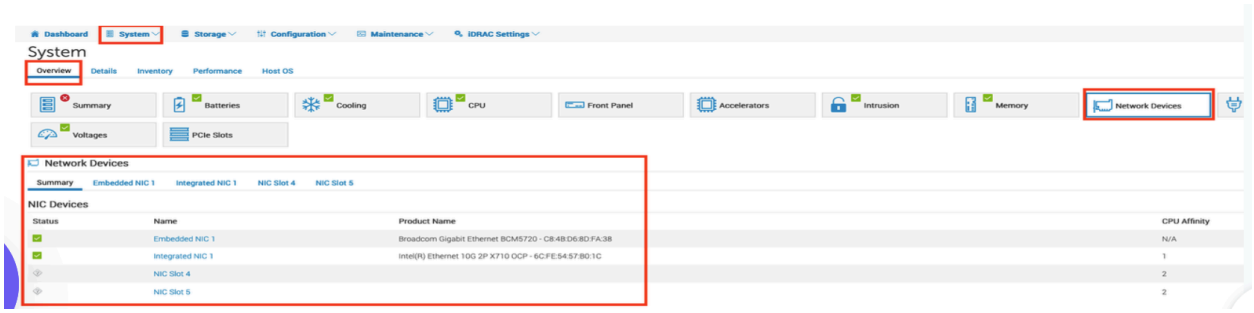

Dashboard
System
Storage
Configuration
Maintenance
iDRAC Settings
Enable Group Manager
System
Overview
Details
Inventory
Performance
Host OS
Refresh
Summary
Batteries
Cooling
CPU
Front Panel
Accelerators
Intrusion
Memory
Network Devices
Power
Voltages
PCIe Slots
System Information
Power State On
Model PowerEdge XR11
Host Name
Operating System
Operating System Version
Service Tag 5YTC0R3
iDRAC Details
iDRAC MAC Address 30:d0:42:de:98:fa
License Enterprise Edit


Storage
Overview
Tasks
Summary
Controllers
Physical Disks
Virtual Disks
Enclosures
Physical Disks Overview
Ready
Online
Foreign
Offline
Blocked
Failed
Non-RAID
Removed
Unknown
Read Only
Sanitize in Progress
Sanitize Failed
Unusable
2
Summary of Disks
Physical Disks 2
Virtual Disks 1
Global 0
Dedicated 0
Non-RAID Disks 0
Recently Logged Storage Events
Severity
Date and Time
Description
There are no events to be displayed.


Maintenance
Lifecycle Log
Job Queue
System Update
System Event Log
Troubleshooting
Diagnostics
SupportAssist
Start a Collection
View Last Collection
Check Warranty
Visit Support Portal
Collection Log
Collection Time
Job Id
Collection Type
Data Collected
Collection Status
Sent Time
Tue Jun 25 14:46:14 2024 JID_193264122129 MANUAL Custom Selection Completed NA
Sat Feb 24 05:55:34 2024 JID_087540199501 MANUAL Custom Selection Completed NA
Sat Feb 17 05:20:00 2024 JID_081470839627 MANUAL Custom Selection Completed NA
Sat Feb 10 05:17:50 2024 JID_075421579656 MANUAL Custom Selection Completed NA
Sun Jul 2 15:47:54 2023 JID_883127778633 MANUAL Custom Selection Completed NA
Sun Jun 25 14:02:05 2023 JID_877016287732 MANUAL Custom Selection Completed NA
Sun Jun 18 14:12:43 2023 JID_870974700260 MANUAL Custom Selection Completed NA
Sun Jun 11 18:52:56 2023 JID_865094796369 MANUAL Custom Selection Completed NA
Sun Jun 4 18:27:40 2023 JID_859031632490 MANUAL Custom Selection Completed NA
Sun May 28 18:16:13 2023 JID_852976699809 MANUAL Custom Selection Completed NA
Collection Settings


Configuration
Power Management
Virtual Console
Virtual Media
Licenses
System Settings
Asset Tracking
BIOS Settings
Server Configuration Profile
Power Control
Power Control
Power Off System
Apply
Power Cap Policy
Active Power Cap Policy
528 Watts; 1802 BTU/hr
Power Cap
Disabled
Power Cap Limits
528
Watts (Recommended Range : 349 - 528 watts)
1802
BTU/hr
100
Maximum % of upper bound
Apply
Discard
Power Configuration




Typically, every computer can be turned into a server using its OS features which allows it to do so. But they won't be able to support many a huge number of connections due to their hardware limitations and also because of their OS limitations. Server computers use the same parts as of the desktop PC but these parts are designed for durability and non-stop working conditions.

## Hardware requirement of Servers :

The hardware functionality is similar to a desktop PC, parts used in servers are of special grade.

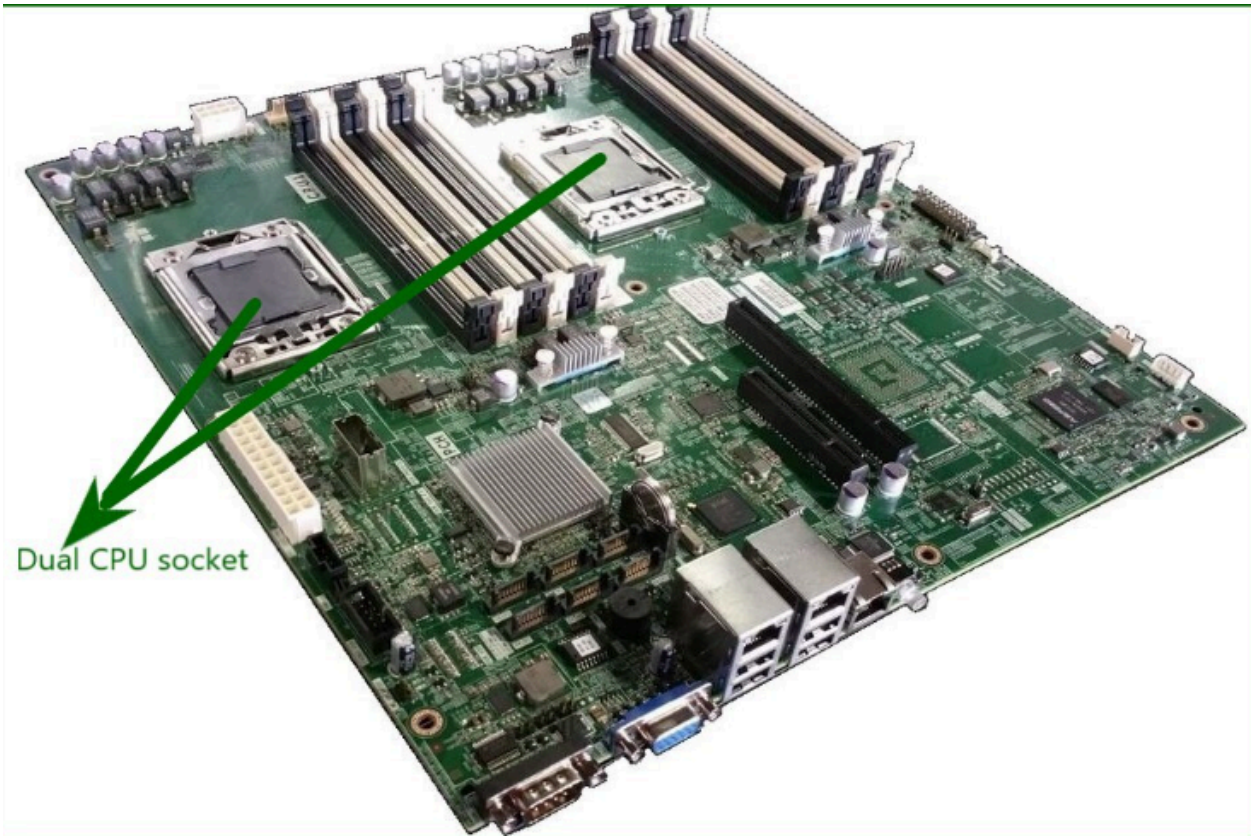
## Example:

The hard disk for server is more resistant to wear, tear and vibrations and can easily withstand

those vibrations for years and indeed is more expensive. Unlike a cheap desktop PC drive which is less resistant to wear and tear and vibrations.

A server can have many hard-drives and all of these drives are connected using a RAID configuration. RAID software automatically distributes all the data to each drive and in case of failure of a drive, It builds up the data back on to the new drive installed in place of the failed one.


The motherboard also is expensive and comes with a dual or quad or even octa- CPU configuration and supports large amount of RAM (up to 1-2TB of Ram). Also, server motherboard can support many hard drives.




Servers use ECC rams (Error correcting ram), that is if there occurs an error the ram itself checks for errors and corrects it making the servers more reliable. The server ram comes in large amount (up to 100GB or 128GB stick).

Servers have redundant power supplies, to keep the server up and running in case of power failure. It use a server grade OS like, Linux, Windows server, macOS server etc….These OS's are configured to handle thousands of concurrent connections.

## Uses of Server Computer:

The servers can be used as: –

1. **Application server –**

   Mainly provides the access to the applications to the user, eliminating the need to install a copy of the app on their respective PC. Example: -Like a web application server, it can be a www server or a local server.

2. **Catalog servers –**

   It keeps and maintains an index or table of content of information which can be accessed by other computers throughout the World Wide Web.

   **Example:** directory servers and name servers are some of the examples.

3. **Computing Server –**

   This type of server shares its extensive amount of computing resources like CPU, Ram and storage with its clients over a network.

4. **Database server –**

   This type of server maintains a specific type of database which can be accessed by a user if the user has access to it. Example: – the database can be of books, videos, pictures, music etc.Like this there are Fax servers, File servers, Media servers, Mail servers, Game servers, Print servers, Proxy servers etc. In this way there can be any number of servers which can be setup to make use of.

## Difference between server and PC :

| Servers | PC |
| --- | --- |
| It is designed for redundancy and non-stop working hours. | It is not designed for redundancy and longer working hours. |
| The hardware is expensive. | Hardware is not that expensive. |
| Uses specialized hardware only. | Does not use specialized hardware. |
| For different tasks different servers are used. Example: – Email server, file server etc… | A PC can be used for any number of tasks. |

## Difference Between Basic Input/Output System (BIOS) and Unified Extensible Firmware Interface (UEFI)

Out of the many firmware interfaces in the field of computer technology, Basic Input/Output System (BIOS) and Unified Extensible Firmware Interface (UEFI) are the most vital when it comes to the booting and handling of the system's hardware. Said interfaces are important for any person engaged in computing because the behaviour, compatibility, and security of a system depend upon this aspect. This article aims at giving elaborate descriptions on BIOS and UEFI separate from laying down their roles and differences.

## What is BIOS?

BIOS stands for Basic Input/Output System. It is a fundamental firmware interface that is found in computers. It is a tiny piece of code that is stored on a special chip called EPROM (Erasable Programmable Read-Only Memory). It contains the necessary instructions for the computer to start and load the operating system. The main objective of BIOS is to check the preferences that are stored in a CMOS chip to understand how the user wants the computer to operate.

## What is UEFI?

UEFI stands for Unified Extensible Firmware Interface. It is a modern firmware interface that replaces the older BIOS in computers. Its main job is to keep important information about the device initialisation, which is stored on a special part of the disk. The primary reason for creating UEFI is to tackle the limitations of BIOS and shorten the system boot time. It makes the computer start faster. It bridges the gap between the computer's hardware and operating system, allowing for better compatibility, security, and functionality.

## Difference Between BIOS and UEFI

The differences between BIOS and UEFI are mentioned below.

| Category | BIOS | UEFI |
| --- | --- | --- |
| **Release Date** | 1981 | 2002 |
| **User Interface** | Basic UI navigation using the keyboard. | User-friendly graphical UI with mouse support. |
| **Operating Mode** | 16-bit | 32-bit/64-bit. |
| **Partition Support** | It supports upto 4 physical partitions, upto 2 TB. | It supports upto 128 physical partitions, upto 18 exabytes. |
| **Security** | It has basic security features, having no inherent security feature. | It supports secure boot, and prevents unauthorized OS. |
| **Performance** | It has slower boot times and limited hardware support. | It has faster boot times and is optimized for modern hardware. |

# BIOS and UEFI: Which One is Better?

BIOS and UEFI are two completely different bootloaders, but the main work is done depending on BIOS and input from the user and the current hardware system present in the computer. However, for older systems or systems that are not very busy, the basic structure, which is BIOS, is enough; as for UEFI, it has some benefits for the modern systems. First, the improved BIOS can support greater capacity of the hard disc and faster booting, and better security which is provided by the UEFI, which is much more suitable for newly built computers and those users who are more professional.

## Hardware Compatibility

**BIOS**: Built for legacy/previously/BIOS, old hardware and all systems that do not need the advantages of Unified Extensible Firmware Interface. If your system is on the older side or you are managing some legacy software then BIOS may be more suitable for you.

**UEFI:** UEFI is created specifically for contemporary hardware and has a higher live capacity over 2 TB and a faster start compared to BIOS. It is suitable mostly to contemporary systems, including those with new hardware devices.

## Boot Time

- BIOS: Usually a longer boot up time because it has lesser features and a comparatively older design.
- UEFI: Reduces boot times heartily and enhances boot management process thus beginning the start up process with greater ease.

## User Interface

- **BIOS**: Daft text with icon based menu, control by keyboard only. In fact, it appears less convenient for a contemporary user, who is used to working with colorful operating systems, for example.
- **UEFI**: Is characterized by a graphical user interface (GUI) that can be operated using a mouse hence providing users with easy configurations.

### Security

- **BIOS**: It has no copy bearing protection system which make it even vulnerable to attacks and enhanced access.
- **UEFI**: This also comprises of things like Secure Boot that assists in preventing unauthorized firmware, operating system, as well as drivers, which increases the security of the system.

### What does BIOS stand for?

BIOS stand for Basic Input/Output System.

### Is the BIOS software or hardware?

BIOS is a software but, it's stored on a hardware component on the motherboard.

### Is BIOS the same for all computers?

No, BIOS is not the same for all computers.

### Is BIOS on the motherboard or CPU?

BIOS is located on a chip on the motherboard, not the CPU.

# Difference between Firmware and Operating System

Last Updated : 15 Jun, 2021

**Firmware :**

It is a piece of programming code embedded in a particular hardware. It is a modified version of the software. Firmware is equivalent to unmodified or fixed code.

**Examples** –

Firmware resides in keyboards, video cards, routers, webcams, motherboards, mice, microwave ovens, refrigerators, washing machines, etc. All these examples contain the pre-programmed embedded firmware where all the functions are performed by the instructions given by firmware.

**Operations in firmware :**

Firmware carries low-level operations which are responsible for the functions of any device. All such devices and hardware are controlled by the firmware. In such a way, every piece of hardware contains some kind of firmware installed into it.

All the software is written in high-level languages but the firmware is written in low-level language, i.e. assembly or machine language. These types of languages are understood by the hardware.

**Commands are given by firmware :**

Main memory and secondary memory both get the commands from the firmware. While booting the devices, firmware gives commands to RAM to take OS code and another command to ROM to give a copy of its OS code to RAM.

**Where firmware resides in memory?**

Firmware resides in the non-volatile memory (ROM or flash memory).

**Firmware code changing:**

Previously, changing used to be rarely done or not done, but changing is possible after its manufacturing.

**Example** –

Suppose a brand wants to add some extra features to its existing devices, then they can change its firmware so that changes will automatically be reflected in the device of that particular brand.

**Operating System :**

It is system software that operates as the foundation layer on a computer (or computing device). It acts as the interface between the hardware and the end-user of the computing devices. It manages all the resources such as I/O devices, processors, secondary storage devices, etc.

Initially, it is loaded onto the computer with the help of a boot program, and then it manages all the application programs. It also provides user interface components so that users can efficiently perform desired functions.

**Examples –**

Windows, Linux, etc

**Where OS resides?**

It resides in mass storage devices.

**Need of Operating System :**

There are multiple reasons behind the need for the OS –

- The operating system is needed for any kind of operation such as accessing any devices, executing any job, etc. It is the same as printing any documents, the print command is given for invoking the printer.
- Suppose a user is accessing hardware devices and at the same time another user wants to access the same devices, these types of ambiguity are avoided by the OS to maintain data consistency.

So there are multiple reasons for using the OS.

**Functional Area :**

The operating system is responsible for multiple functions, which are specified below:

- **Resource Management** –

  Allocating resources such as compilers, assemblers, utility programs, etc.
- **Processor Management** –

  Allocating the processors for different jobs.
- **Memory Management** –

  Allocations of main memory and other storage devices.
- **File Management** –

  Storing records of files on various storage devices.
- **Security and privacy** –

  maintaining the authenticity of the system.
- **Input / Output Management** –

  It is helpful in maintaining interaction and allocation of various i/o devices.

**Some useful points about the differences between firmware and operating systems :**

- Firmware typically resides in the ROM whereas the OS lives on disk.
- Firmware is a small program but the OS is a huge one.
- Firmware is usually fixed but the OS is often updated on a regular basis.
- Firmware is low-level operations, whereas the OS is high-level interfaces.
- Firmware is used for a single purpose, but the OS is used for general purposes, which allows any kind of software to run on multiple types of hardware.

| Sl. No- | Firmware | Operating System |
|---|---|---|
| 1. | It is a piece of programming code embedded in a particular hardware. | It is system software that operates as the foundation layer on a computer (or computing device). |
| 2. | It resides in ROM. | It resides on a disk. |
| 3. | It is a small program. | It is a huge program. |
| 4. | It is usually fixed. | It is often updated on a regular basis. |
| 5. | It is a low-level operation. | It is a high-level interface. |
| 6. | It has a single purpose. | It is a general-purpose system that allows any kind of software to run on multiple types of hardware. |
| 7. | **Examples** – It resides in keyboards, video cards, routers, webcams, motherboards, mice, microwave ovens, refrigerators, washing machines, etc. | **Examples** – Apple macOS, Microsoft Windows, Google's Android OS, Linux Operating System, and Apple iOS. |

# Kernel in Operating System

A kernel is a central component of an operating system that manages the operations of computers and hardware. It basically manages operations of memory and CPU time. It is a core component of an operating system. Kernel acts as a bridge between applications and data processing performed at the hardware level using inter-process communication and system calls.

## What is Kernel?

A kernel is the core part of an operating system. It acts as a bridge between software applications and the hardware of a computer. The kernel manages system resources, such as the CPU, memory, and devices, ensuring everything works together smoothly and efficiently. It handles tasks like running programs, accessing files, and connecting to devices like printers and keyboards.

## Types of Kernel

The kernel manages the system's resources and facilitates communication between hardware and software components. These kernels are of different types let's discuss each type along with its advantages and disadvantages:

## 1. Monolithic Kernel

It is one of the types of kernel where all operating system services operate in kernel space. It has dependencies between systems components. It has huge lines of code which is complex.

### Example:

```
Unix, Linux, Open VMS, XTS-400 etc.
```

## Advantages

- **Efficiency:** Monolithic kernels are generally faster than other types of kernels because they don't have to switch between user and kernel modes for every system call, which can cause overhead.
- **Tight Integration:** Since all the operating system services are running in kernel space, they can communicate more efficiently with each other, making it easier to implement complex functionalities and optimizations.
- **Simplicity:** Monolithic kernels are simpler to design, implement, and debug than other types of kernels because they have a unified structure that makes it easier to manage the code.
- **Lower latency:** Monolithic kernels have lower latency than other types of kernels because system calls and interrupts can be handled directly by the kernel.

## Disadvantages

- **Stability Issues:** Monolithic kernels can be less stable than other types of kernels because any bug or security vulnerability in a kernel service can affect the entire system.
- **Security Vulnerabilities:** Since all the operating system services are running in kernel space, any security vulnerability in one of the services can compromise the entire system.
- **Maintenance Difficulties:** Monolithic kernels can be more difficult to maintain than other types of kernels because any change in one of the services can affect the entire system.
- **Limited Modularity:** Monolithic kernels are less modular than other types of kernels because all the operating system services are tightly integrated into the kernel space. This makes it harder to add or remove functionality without affecting the entire system.

# 2. Micro Kernel

It is kernel types which has minimalist approach. It has virtual memory and thread scheduling. It is more stable with less services in kernel space. It puts rest in user space. It is use in small os.

**Example :**

```
Mach, L4, AmigaOS, Minix, K42 etc.
```

## Advantages

- **Reliability:** Microkernel architecture is designed to be more reliable than monolithic kernels. Since most of the operating system services run outside the kernel space, any bug or security vulnerability in a service won't affect the entire system.
- **Flexibility**：Microkernel architecture is more flexible than monolithic kernels because it allows different operating system services to be added or removed without affecting the entire system.
- **Modularity:** Microkernel architecture is more modular than monolithic kernels because each operating system service runs independently of the others. This makes it easier to maintain and debug the system.
- **Portability:** Microkernel architecture is more portable than monolithic kernels because most of the operating system services run outside the kernel space. This makes it easier to port the operating system to different hardware architectures.

### Disadvantages

- **Performance:** Microkernel architecture can be slower than monolithic kernels because it requires more context switches between user space and kernel space.
- **Complexity:** Microkernel architecture can be more complex than monolithic kernels because it requires more communication and synchronization mechanisms between the different operating system services.
- **Development Difficulty:** Developing operating systems based on microkernel architecture can be more difficult than developing monolithic kernels because it requires more attention to detail in designing the communication and synchronization mechanisms between the different services.
- **Higher Resource Usage:** Microkernel architecture can use more system resources, such as memory and CPU, than monolithic kernels because it requires more communication and synchronization mechanisms between the different operating system services

## 3. Hybrid Kernel

It is the combination of both monolithic kernel and microkernel. It has speed and design of monolithic kernel and modularity and stability of microkernel.

### Example :

```
Windows NT, Netware, BeOS etc.
```

### Advantages

- **Performance:** Hybrid kernels can offer better performance than microkernels because they reduce the number of context switches required between user space and kernel space.
- **Reliability:** Hybrid kernels can offer better reliability than monolithic kernels because they isolate drivers and other kernel components in separate protection domains.

- **Flexibility:** Hybrid kernels can offer better flexibility than monolithic kernels because they allow different operating system services to be added or removed without affecting the entire system.
- **Compatibility:** Hybrid kernels can be more compatible than microkernels because they can support a wider range of device drivers.

### Disadvantages

- **Complexity:** Hybrid kernels can be more complex than monolithic kernels because they include both monolithic and microkernel components, which can make the design and implementation more difficult.
- **Security:** Hybrid kernels can be less secure than microkernels because they have a larger attack surface due to the inclusion of monolithic components.
- **Maintenance:** Hybrid kernels can be more difficult to maintain than microkernels because they have a more complex design and implementation.
- **Resource Usage:** Hybrid kernels can use more system resources than microkernels because they include both monolithic and microkernel components.

## 4. Exo Kernel

It is the type of kernel which follows end-to-end principle. It has fewest hardware abstractions as possible. It allocates physical resources to applications.

### Example :

```
Nemesis, ExOS etc.
```

## Advantages

- **Flexibility:** Exokernels offer the highest level of flexibility, allowing developers to customize and optimize the operating system for their specific application needs.
- **Performance:** Exokernels are designed to provide better performance than traditional kernels because they eliminate unnecessary abstractions and allow applications to directly access hardware resources.
- **Security:** Exokernels provide better security than traditional kernels because they allow for fine-grained control over the allocation of system resources, such as memory and CPU time.
- **Modularity:** Exokernels are highly modular, allowing for the easy addition or removal of operating system services.

## Disadvantages

- **Complexity:** Exokernels can be more complex to develop than traditional kernels because they require greater attention to detail and careful consideration of system resource allocation.
- **Development Difficulty:** Developing applications for exokernels can be more difficult than for traditional kernels because applications must be written to directly access hardware resources.
- **Limited Support:** Exokernels are still an emerging technology and may not have the same level of support and resources as traditional kernels.
- **Debugging Difficulty:** Debugging applications and operating system services on exokernels can be more difficult than on traditional kernels because of the direct access to hardware resources.

# Nano Kernel

It is the type of kernel that offers hardware abstraction but without system services. Micro Kernel also does not have system services therefore the Micro Kernel and Nano Kernel have become analogous.

### Example :

`EROS etc.`

## Advantages

- **Small Size**: Nanokernels are designed to be extremely small, providing only the most essential functions needed to run the system. This can make them more efficient and faster than other kernel types.
- **High Modularity:** Nanokernels are highly modular, allowing for the easy addition or removal of operating system services, making them more flexible and customizable than traditional monolithic kernels.
- **Security:** Nanokernels provide better security than traditional kernels because they have a smaller attack surface and a reduced risk of errors or bugs in the code.
- **Portability:** Nanokernels are designed to be highly portable, allowing them to run on a wide range of hardware architectures.

## Disadvantages

- **Limited Functionality:** Nanokernels provide only the most essential functions, making them unsuitable for more complex applications that require a broader range of services.
- **Complexity:** Because nanokernels provide only essential functionality, they can be more complex to develop and maintain than other kernel types.

- **Performance:** While nanokernels are designed for efficiency, their minimalist approach may not be able to provide the same level of performance as other kernel types in certain situations.
- **Compatibility:** Because of their minimalist design, nanokernels may not be compatible with all hardware and software configurations, limiting their practical use in certain contexts.

# Functions of Kernel

The kernel is responsible for various critical functions that ensure the smooth operation of the computer system. These functions include:

## 1. Process Management

- Scheduling and execution of processes.
- Context switching between processes.
- Process creation and termination.

## 2. Memory Management

- Allocation and deallocation of memory space.
- Managing virtual memory.
- Handling memory protection and sharing.

## 3. Device Management

- Managing input/output devices.
- Providing a unified interface for hardware devices.
- Handling device driver communication.

## 4. File System Management

- Managing file operations and storage.
- Handling file system mounting and unmounting.
- Providing a file system interface to applications.

## 5. Resource Management

- Managing system resources (CPU time, disk space, network bandwidth)
- Allocating and deallocating resources as needed
- Monitoring resource usage and enforcing resource limits

## 6. Security and Access Control

- Enforcing access control policies.
- Managing user permissions and authentication.
- Ensuring system security and integrity.

### 7. Inter-Process Communication

- Facilitating communication between processes.
- Providing mechanisms like message passing and shared memory.

## Working of Kernel

- A kernel loads first into memory when an operating system is loaded and remains in memory until the operating system is shut down again. It is responsible for various tasks such as disk management , task management, and memory management .
- The kernel has a process table that keeps track of all active processes
- The process table contains a per-process region table whose entry points to entries in the region table.
- The kernel loads an executable file into memory during the ‘exec’ system call’.
- It decides which process should be allocated to the processor to execute and which process should be kept in the main memory to execute. It basically acts as an interface between user applications and hardware. The major aim of the kernel is to manage communication between software i.e. user-level applications and hardware i.e., CPU and disk memory.

## Objectives of Kernel

- To establish communication between user-level applications and hardware.
- To decide the state of incoming processes.
- To control disk management.
- To control memory management.
- To control task management.

## Conclusion

Kernels are the heart of operating systems , managing how hardware and software communicate and ensuring everything runs smoothly. Different types of kernels—like monolithic, microkernels, hybrid kernels, and others—offer various ways to balance performance, flexibility, and ease of maintenance. Understanding these kernel types helps us appreciate how operating systems work and how they handle the complex tasks required to keep our computers and devices running efficiently. Each type of kernel has its own strengths and weaknesses, but all play a crucial role in the world of computing.

# Difference between Operating System and Kernel

In the world of computing, two terms that are frequently mentioned are Operating System (OS) and Kernel. While they are closely related, they serve different purposes within a computer system. Understanding the difference between the two can help clarify how computers function and why both are essential for smooth operation. In this article, we will explore the key differences between the OS and the Kernel, their functions, and how they work together to manage hardware and software

## What is an Operating System?

An Operating System (OS) is the software that manages computer hardware, and software, and provides services for computer programs. It acts as an intermediary between the user and the hardware, making it easier for users to interact with the system without needing to understand the complex details of the hardware.

# What is a Kernel?

The Kernel is the core part of the operating system. It manages communication between the hardware and the software. Think of the kernel as the “brain” of the OS, controlling everything from memory allocation to device management.