

# Diseño e Implementación de un File Server Alternativo para Minix

## INTRODUCCIÓN

El diseño de **Minix** refleja un conjunto de particularidades aptas para su estudio en el entorno académico. Su tercera versión mantiene la función pedagógica de las anteriores pero incorpora las nociones de seguridad, eficiencia y confiabilidad necesarias para hacer un sistema operativo de alta fiabilidad. El presente trabajo busca extender las funcionalidades de Minix 3 mediante la incorporación de un **File Server Alternativo**. Este nuevo servidor se encarga de gestionar un sistema de archivos ubicado en un disco RAM.

AUTORES

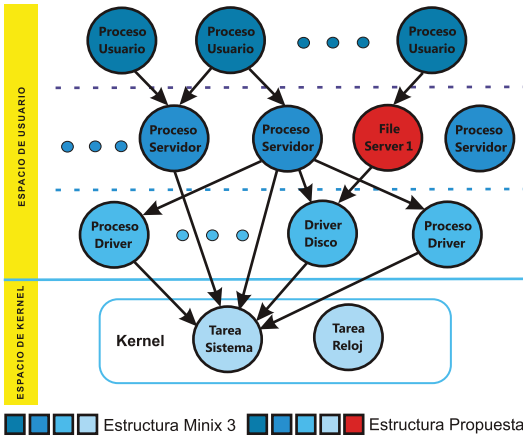
**BLAS, María Julia**  
mariajuliablas@gmail.com  
**PROLA, Gabino Ignacio**  
gabinoprola@gmail.com  
**SARLI, Juan Leonardo**  
juanleonardosarli@gmail.com

TRABAJO DE CÁTEDRA

Cátedra  
**Diseño e Implementación de Sistemas Operativos**  
Docentes  
**Mg Pablo Andrés Pessolani**  
**Ing. Cesar Ballardini**

## ELEMENTOS Y METODOLOGÍA DE TRABAJO

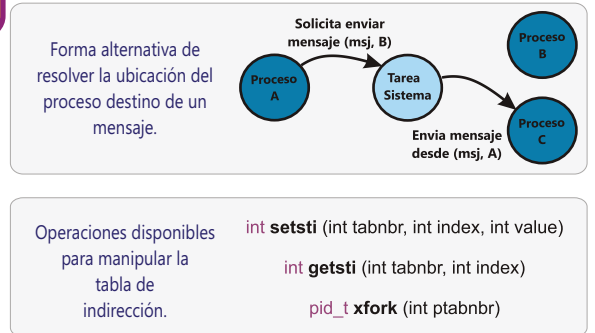
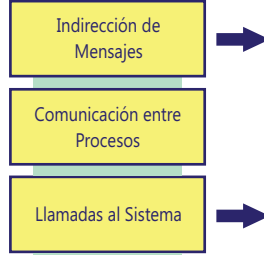
### 1. Arquitectura Objetiva



### 2. Soporte

Teniendo en cuenta que el nuevo servidor debe interactuar con un grupo selecto de procesos, se utilizaron herramientas que contribuyen a estos fines.

#### Conceptos Básicos Utilizados en la Implementación



### 3. Desarrollo e Implementación

En esta etapa se desarrolló un programa de usuario capaz de testear al nuevo servidor, se codificó la solución propuesta y se la integró a la estructura de Minix. Como resultado se obtuvo una nueva versión del sistema operativo preparada para operar con dos FS en simultáneo.

#### 1. PROGRAMA USUARIO DE

```
int main(int argc, char* argv[])
{
    /* Se determina en p_nr del fs. */
    endpoint = ObtenerEndpoint(fs);
    p_nr = ObtenerNumero(endpoint);
    /* Se altera el contenido de la tabla. */
    setsti(1, p_nr+NR_TASKS, argumento);
    /*
     * Se crea un proceso que trabaje sobre
     * la tabla 1.
     */
    pid = xfork(1);
    if(pid==0)
    {
        /* Crea, abre y escribe 'archivo1'. */
        archivo1= Abrir("archivo1");
        Escribir(archivo1, texto);
        Cerrar(archivo1);

        /* Abre y lee 'archivo1'. */
        archivo1= Abrir("archivo1");
        texto = Leer(archivo1);
        Cerrar(archivo1);

        /* Crea, abre y escribe 'archivo2'. */
        archivo2= Abrir("archivo2");
        Escribir(archivo2, texto);
        Cerrar(archivo2);
    }
}
```

#### 2. CODIFICACIÓN NUEVO FILE

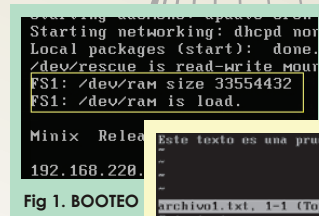
```
PUBLIC int main()
{
    Inicializar();
    Mientras(TRUE)
    {
        mensaje = ObtenerTrabajo();
        proceso = DeterminarEmisor(mensaje);
        Si(proceso = INIT)
        {
            Abrir(discoram);
            SetearTamaño(discoram, tamaño);
            Copiar(particionbooteo, discoram);
            InicializarDispositivo(discoram);
        }
        ChequearMensaje(mensaje);
        resultado = RealizarTrabajo(mensaje);
        EnviarResultados(resultado, proceso);
    }
}
```

#### 3. INTEGRACIÓN

Dada la naturaleza de la relación **PM-FS**, se decidió mantener esta interacción para el **FS1**. Posteriormente se añadió el **FS1** a la **imagen de boteo** para que inicie como un servidor del sistema.

## RESULTADOS

Al bootear el sistema, dentro del conjunto de mensajes iniciales se observaron los indicadores que referencian el accionar del FS1 (Fig. 1). Una ejecución del programa de prueba que utiliza al FS1 (p\_nr=8) como servidor de archivos dió como resultado la Fig. 2 y Fig. 3. De esta manera, puede establecerse que los resultados obtenidos han sido satisfactorios.



index	( 0)	( 1)	( 2)	( 3)
( -1)	-4	-4	-4	-4
( -3)	-3	-3	-3	-3
( -2)	-2	-2	-2	-2
( -1)	-1	-1	-1	-1
( 0)	0	0	0	0
( 1)	1	1	1	1
( 2)	2	2	2	2
( 3)	3	3	3	3
( 4)	4	4	4	4
( 5)	5	5	5	5
( 6)	6	6	6	6

Fig 2. TABLA DE INDIRECCIÓN



## CONCLUSIONES

Se ha demostrado que es posible incorporar un servidor alternativo a la estructura de Minix 3.1.2a. Mediante el uso de la indirección de mensajes, un proceso usuario ha logrado utilizar los servicios brindados por servidor desarrollado, el cual se encarga de crear y administrar un disco RAM que contiene un sistemas de archivos. De esta manera, este trabajo puede ser tomado como base para la implementación de procesos servidores similares relacionados a otros ámbitos.