

Cómo convertí mi Nintendo 64 en una bóveda criptográfica

En este tutorial vamos a programar una aplicación real para N64 que permite escribir mensajes, elegir una clave de seguridad con el Joystick y guardar los datos cifrados en el [Controller Pak](#) usando el algoritmo XOR.

1██ Requisitos Previos

- **Hardware:** Una Nintendo 64 y un [Controller Pak](#) (Memory Card).
 - **Software:** Docker instalado en tu ordenador.
 - **Flashcart:** Un EverDrive o similar para cargar tu creación en la consola.
-

2██ Configuración del Entorno (macOS/Linux)

Si usas un Mac con chip M1, M2 o M3, este paso es vital para evitar errores de arquitectura.

Abrimos la terminal y ejecutamos:

Bash

```
# Crear la carpeta del proyecto
mkdir -p ~/n64_vault && cd ~/n64_vault

# Forzar plataforma compatible para Libdragon
export DOCKER_DEFAULT_PLATFORM=linux/amd64

# Inicializar el entorno de desarrollo
```

```
libdragon init
```

3██ Estructura del Proyecto

Debes tener tus archivos organizados así dentro de la carpeta n64_vault:

- Makefile (Instrucciones de compilación)
 - src/main.c (Código fuente del juego)
-

4██ El Archivo de Compilación (Makefile)

Crea un archivo llamado Makefile y pega esto:

Makefile

```
PROG_NAME = n64_vault_beauty
SRCDIR = src
BUILDDIR = build
```

```
N64_INST = /n64_toolchain
include $(N64_INST)/include/n64.mk
```

```
OBJS = $(BUILDDIR)/main.o
CFLAGS += -Iinclude -falign-functions=32
```

```
all: $(PROG_NAME).z64
```

```
$(PROG_NAME).elf: $(OBJS)
```

```
$(PROG_NAME).z64: N64_ROM_TITLE = "CRYPTO VAULT N64"
```

```
$(BUILDDIR)/%.o: $(SRCDIR)/%.c
    @mkdir -p $(dir $@)
    $(CC) $(CFLAGS) -c -o $@ $<
```

clean:

```
rm -rf $(BUILDDIR) *.z64 *.elf
```

.PHONY: clean

5██ El Código Fuente (src/main.c)

Crea la carpeta src, dentro crea main.c y pega este código con la interfaz «Aesthetic»:

```
C

#include <stdio.h>
#include <string.h>
#include <libdragon.h>

#define SAVE_BLOCK 0x0600
#define MSG_LEN 16

int main(void) {
    timer_init();
    display_init(RESOLUTION_320x240, DEPTH_16_BPP, 2,
GAMMA_NONE, ANTIALIAS_RESAMPLE);
    joypad_init();

    uint8_t buffer[MSG_LEN] __attribute__((aligned(16)));
    char mensaje[MSG_LEN] = "TEXT0_SECRETO";
    uint8_t llave = 0x55;
    int char_index = 0;
    char status[64] = "SISTEMA LISTO";

    while (1) {
        joypad_poll();

        joypad_inputs_t    input    =
joypad_get_inputs(JOYPAD_PORT_1);
        joypad_buttons_t   btn      =
```

```

joypad_get_buttons_pressed(JOYPAD_PORT_1);

    if (input.stick_y > 40) llave++;
    if (input.stick_y < -40) llave--;
    if (btn.d_right) char_index = (char_index + 1) %
(MSG_LEN - 1);
    if (btn.d_left) char_index = (char_index - 1 +
(MSG_LEN - 1)) % (MSG_LEN - 1);
    if (btn.d_up) mensaje[char_index]++;
    if (btn.d_down) mensaje[char_index]--;

    if (btn.z) {
        uint8_t  cifrado[MSG_LEN]
__attribute__((aligned(16)));
        for(int i=0; i<MSG_LEN; i++) cifrado[i] =
(uint8_t)mensaje[i] ^ llave;
        if (joybus_accessory_write(JOYPAD_PORT_1,
SAVE_BLOCK, cifrado) == 0)
            strcpy(status, "DATOS CIFRADOS Y GUARDADOS");
        else
            strcpy(status, "ERROR: PAK NO DETECTADO");
    }
    if (btn.l) {
        if (joybus_accessory_read(JOYPAD_PORT_1,
SAVE_BLOCK, buffer) == 0) {
            for(int i=0; i<MSG_LEN; i++) mensaje[i] =
(char)(buffer[i] ^ llave);
            strcpy(status, "DESCRIPCION COMPLETADA");
        }
    }

    surface_t *disp = display_get();
    graphics_fill_screen(disp, 0x2104); // Fondo Azul
Marino

    graphics_draw_box(disp, 0, 0, 320, 40, 0x421F);
    graphics_draw_text(disp, 90, 15, "N64 CRIPTO-VAULT
V2.0");

    graphics_draw_box(disp, 190, 60, 110, 60, 0x0000);
    graphics_draw_text(disp, 200, 75, "KEY ID:");

```

```

char hex_llave[10];
sprintf(hex_llave, "0x%02X", llave);
graphics_draw_text(dis, 220, 95, hex_llave);

graphics_draw_text(dis, 30, 60, "MENSAJE:");
graphics_draw_line(dis, 30, 135, 170, 135, 0xFFFF);
graphics_draw_text(dis, 35, 120, mensaje);
static int timer = 0;
if ((timer++ / 10) % 2)
    graphics_draw_box(dis, 35 + (char_index * 8),
130, 8, 2, 0xF800);

graphics_draw_box(dis, 0, 180, 320, 60, 0x0000);
graphics_draw_line(dis, 0, 180, 320, 180, 0xF800);
graphics_draw_text(dis, 20, 195, "ESTADO:");
graphics_draw_text(dis, 80, 195, status);
graphics_draw_text(dis, 20, 220, "Z: GUARDAR  L: LEER
STICK: CLAVE");

    display_show(dis);
}
}

```

6██ Compilación Final

Para generar tu juego, simplemente ejecuta en la terminal:

Bash

```

libdragon make clean
libdragon make

```

Esto creará el archivo **n64_vault_beauty.z64**. ¡Cópialo a tu EverDrive y a disfrutar!

□ ¿Cómo funciona el cifrado?

El programa utiliza una operación matemática llamada **XOR**. Cuando pulsas **Z**, el texto se mezcla con la posición del Joystick (la llave). Si la llave no es la correcta al intentar leer con **L**, el resultado será «basura» visual. Solo tú, que conoces el valor hexadecimal de la llave, podrás recuperar el mensaje original.