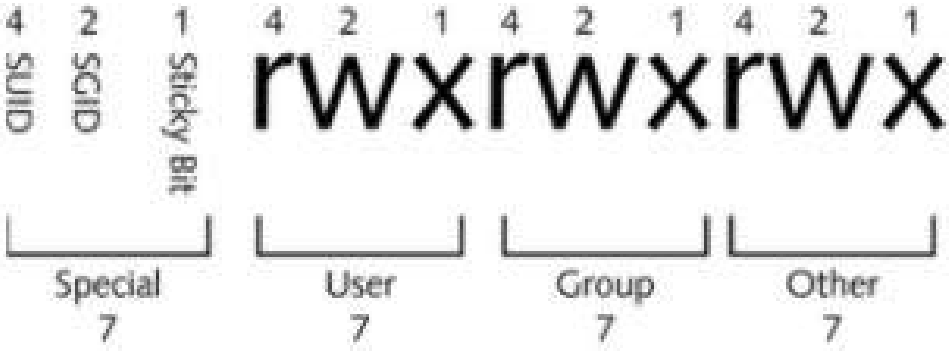
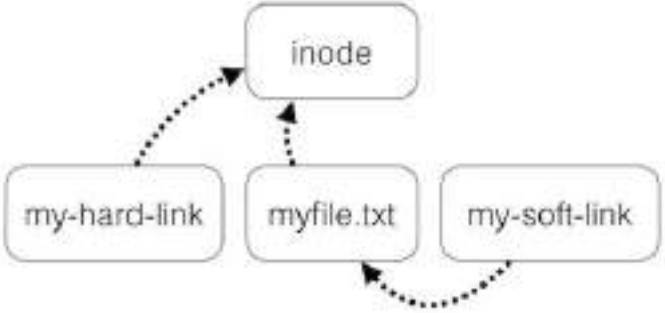
Permisos en Unix



Links en Unix



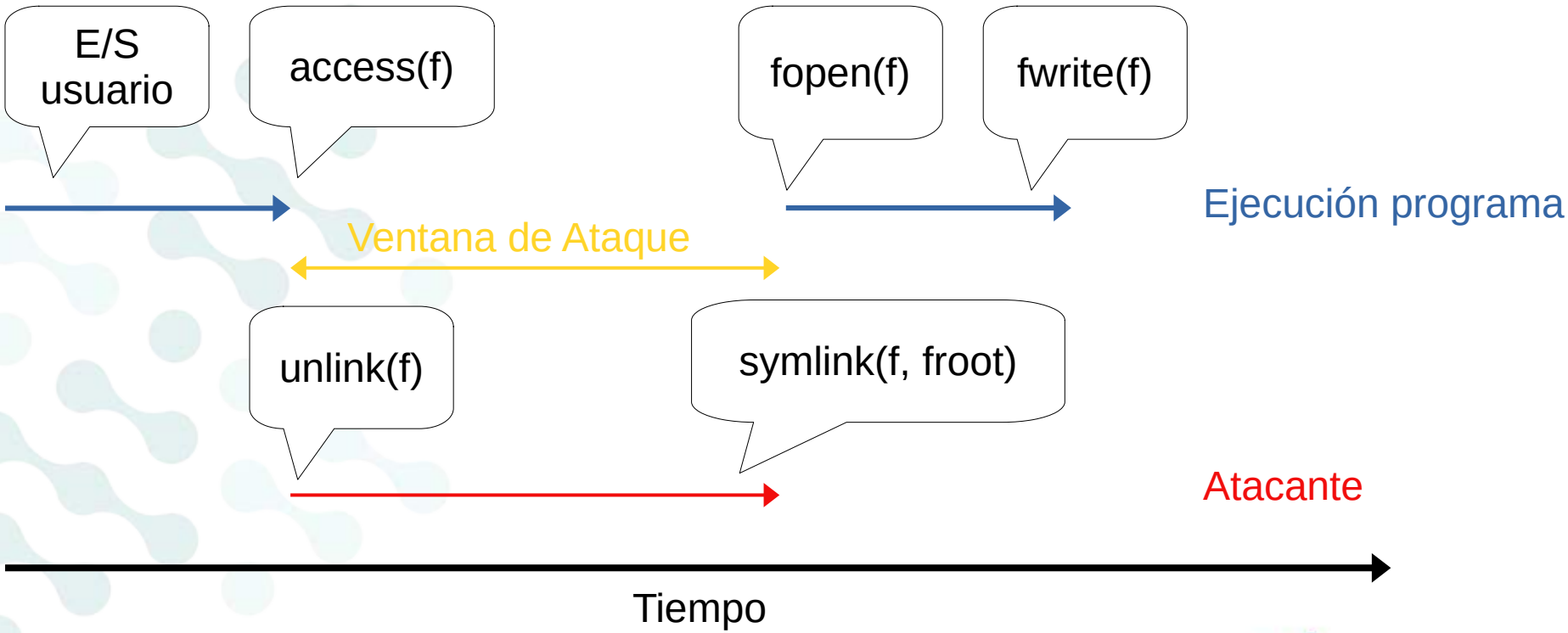
Ejemplo de Explotación mediante Symlinks

- Objetivo: binario SUID con un TOCTOU

```c
#include <stdio.h>
#include <unistd.h>
#include <string.h>

int main(int argc, char * argv[])
{
        char *file = argv[1];
        char buffer[51];
        FILE *fd;

        /* get user input */
        scanf("%50s", buffer);

        /* check access */
        if(!access(file, W_OK)) {
                /* write user input to file ¨*/
                fd = fopen(file, "a+");
                fwrite(buffer, sizeof(char), strlen(buffer), fd);
                fclose(fd);
        } else {
                printf("User does not have access\n");

        }

        return 0;
}
```

Ejemplo de Explotación mediante Symlinks

- Patrón de explotación:



Ejercicio

- Crear un programa que:
  - Tome como nombre del archivo a editar el primer parámetro de la línea de comandos
  - Defina un texto a escribir en el archivo
  - Verifique permisos del archivo:
    - Si el usuario TIENE permisos:
      - Simule el delay (procesamiento del archivo)
      - Abra el archivo, escriba el mensaje y lo cierre
    - Si el usuario NO TIENE permisos:
      - De un mensaje de error: “Sin permisos”

Simulación del Delay (aka “procesamiento”)

```
#define DELAY 99999999

/* Simular la demora entre la verificacion y la accion */
for(i = 0; i < DELAY;i++) {
        int a = i^2; /* Se podría utilizar cualquier otra operación, es solo un ejemplo */
}
```

Armado Ambiente de Prueba

- Crear un archivo (no vacio) con el usuario común
- Cambiarle el dueño a root
- Compilar el programa vulnerable
- Cambiarle el dueño a root al programa vulnerable
- Agregar permisos SUID root programa vulnerable
- Dar permisos de ejecución al script de prueba

Ejecución del Ataque

- Se provee el script exploit-toctou.sh que:
  - Toma como primer parámetro el nombre del programa vulnerable
  - Toma como segundo parámetro el nombre del archivo protegido
  - Lleva adelante el ataque creando el link simbólico y ejecutando el programa vulnerable

¿Qué pasa si ejecutan múltiples veces seguidas el exploit? ¿Por qué?

Ayudas...

- Correr algo como root en Linx

  `sudo comando`
- Hacer SUID un programa en Linux

  `chmod u+s archivo`
- Cambiar dueño de un archivo a root en Linux

  `chown root:root archivo`
- Compilación

  `clang archivo.c -o archivo_ejecutable`

Ayudas cont.

- Agregar contenido a un archivo

  `echo “contenido” > archivo`

- Dar permisos de ejecución

  `chmod +x archivo`

OJO: ¡Al recompilar repetir asignación de permisos!