Análisis de carga

Javier Falcón (2016-5265)

1 Ejercicio 1

1. ¿Cuál es la primera instrucción en ejecutar?
   La primera instrucción es un *long jump* (ljmp $0x3630,$0xf000e05b)

2. ¿En qué dirección física se encuentra esa instrucción?
   Se encuentra en la dirección 0xffff0

3. ¿Cuál es el propósito de la primera instrucción? ¿Por qué fue situada?
   Posicionar el puntero a la primera intrucción del segmento de código del cargador.

4. ¿Cuáles son las próximas tres instrucciones?

   - cmpw $0xffc8,%cs:(%esi)
   - jne 0xd241d416
   - xor %edx,%edx

2 Ejercicio 2

1. ¿Cómo el cargador lee sectores de disco? En particular, ¿qué interrupción de BIOS es utilizada?

   Se define una subrutina, en la cual se toma el número de disco y se carga en el registo $\%dl$, así como también el número del sector se carga en el registro $\%ebx$. Luego se lee el contenido en memoria del sector en $ES : 0000$.
   La instrucción *int* $0x13 genera la llamada a la tabla de interrupciones. Tiene relación a procesos de lectura y escritura en el disco.

2. ¿Cómo el cargador decide si encuentra el núcleo de Pintos?
   Mientras va leyendo cada partición, se chequean 3 condiciones: si la partición no está utilizada, si es una partición del kernel de Pintos y, finalmente, si es una partición cargable. Si cumple con las tres condiciones, entonces encontró el núcleo de Pintos.

3. ¿Qué sucede cuando el cargador no puede encontrar el núcleo de Pintos?
   Imprime que no pudo encontrar el kernel de Pintos y llama a la interrupción $0x18$, invocada cuando no se encuentra ningún disco booteable.

4. ¿En qué punto el cargador le transfiere control al núcleo Pintos?

   Se hace justo después de que el se carga el código del kernel en memoria, en $ES : 0000$. Se convierte la dirección de 32-bits al formato *segment:offset* para el "modo real".

3 Ejercicio 3

Se modificó el fichero *init.c* y se agregó el archivo fuente *pintosShell.c*. Ambos contenidos en la ruta **pintos/src/threads**