Bomba José Manuel Pérez Léndinez

Se ha realizado con gbd ya que ddd me dio muchos problemas.

Compresión del mecanismo de encriptación para la contraseña.

Lo primero que vemos al analizar el código es que después de la petición de introducir la contraseña se encuentra un bucle. Analizando su código se puede ver que es el encargado de cifrar la contraseña.

Antes de entrar del bucle se ve que en la posición 0x20(%esp) tenemos un 0 y sera usada como iterador y en %eax se tiene el tamaño de la cadena -1 (con la instrucción sub para no modificar el \n que señala el fin de la cadena) que se introdujo y se almacena en 0x24(%esp) que se usa para la expresión de fin del bucle.

En la linea 155 junto a la 159 conforman expresión que comprara que el iterador 0x20(%esp),al que se le suma 1 en la linea 146, almacenado en %eax es menor que el 0x24(%esp) (tamaño de la cadena -1) y si es así continua con el bucle. Si se cumple eso salta al cuerpo del bucle.

Con la linea 118 trae de memoria la contraseña introducida.

Con la linea 122 eax tiene el valor del iterador.

Con las lineas de la 122 a la 128 nos quedamos con la letra que corresponde a la pasada del bucle.

Con la linea 101 se suma 1 a la letra correspondiente de esta pasada

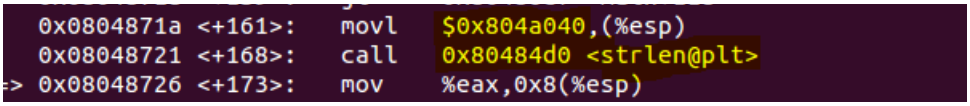
Después de la linea 134 a la 142 se realiza el mismo proceso de antes pero en vez de cargar en edx la contraseña se almacena en ecx y en edx se carga la pasada actual del bucle. Para luego dejar en de la posición de memoria que corresponde a la letra de esta pasada.

Con la linea 144 se añade la letra modificada a la contraseña.

Después se empiezan otra vez las comprobaciones de si el bucle a terminado o no.

Con esto se ve por tanto que se esta haciendo un algoritmo cesar a la contraseña introducida por el usuario en el que se cambia cada carácter de la contraseña por el siguiente carácter en la tabla ascii.

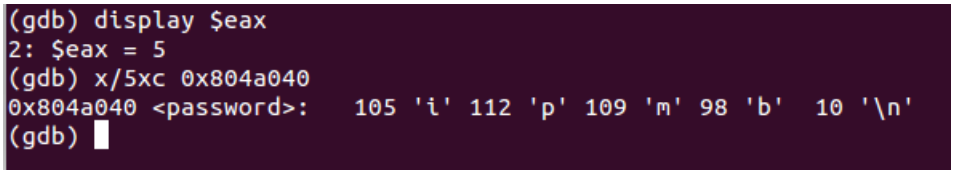
Obtener contraseña



La contraseña se pasa a strlen para ver su tamaño justo antes de la llamada colocando la en la pila.

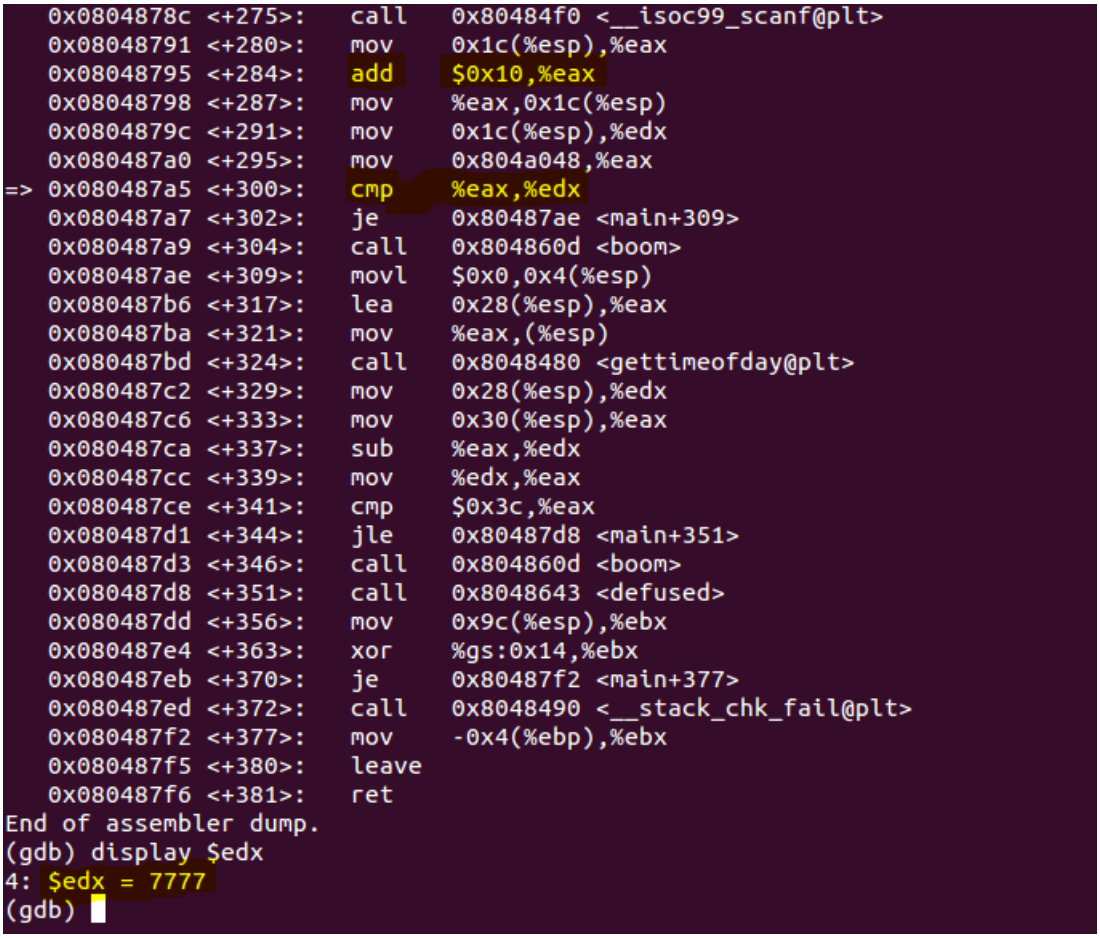
El tamaño de la contraseña no sabremos también justo después de terminar la llamada a strlen y se almacena en eax.

Con estos dos datos podemos mostrar perfectamente la contraseña desde esa dirección y con el tamaño de esta con el siguiente comando.



Solo nos quedaria mirar en la tabla assci el valor de los caracteres anteriores de la contraseña. En nuestro caso corresponderia con hola.

Compresión del mecanismo de encriptación para la pidcode y descifrado.



Al pid code introducido por el usuario se le suma 0x10 que corresponde con 16 en la linea 284. Y teniendo en cuenta que en la linea 300 se comprara con %edx que almacena el pidcode de la bomba.

Por tanto para saber el pid code solo hace falta mostrar el registro edx y restarle 16. El pidcode seria 7761.

Desactivar bomba.

Solo hay que cambiar las sentencia je por jmp con el programa ghx. Primero se busca con objdump -d y el nombre del ejecutable las lineas que aparecen con las cadenas del jn y se modifica la sentencia de je que el 74 por eb de esa forma salta siempre y no llama nunca a la función boom