**Lobko Artem, 12 group. Lab4\_SECURE\_CODE**

**Level1**

size\_t strlen(const char \*str)

Так как функция возвращает size\_t, size\_t может принимать положительные целые значения. Тогда как char ограничен от -128 до 127. То есть ввод argv[1] длина более 127 символов вызывает переполнение char.

**Level2**

unsigned long int strtoul(const char \*str, char \*\*endptr, int base)

В strtoul первым параметром передается предполагаемое целое число для приведения. Отрицательное число допускается, минус игнорируется функцией, однако возвращается unsigned long int. Отрицательные числа не входят в диапазон данного типа и возвращается MAX\_VALUE – value, где value – значение, переданное первым параметром.

**Level3**

int atoi(const char \*str)

Похожая ситуация со вторым уровнем. Функция atoi может принимать отрицательные значения, минус игнорируется, возвращается int. Однако во время присваивания к unsigned int, вылетает за пределы и присваивает MAX\_VALUE – value.

**Leve4**

Ввел Integer.MAX\_VALUE – 2;

**Level5**

В стеке лежат переменная a и массив buf[], переполнив buf[], изменится значение a.

**Level6**

CreateFile пытается открыть консоль, консоль находится по пути C:\Windows\System32\cmd.exe.

**Level7**

Как в 5, но строго 16 элементов. То есть сбросить только 8 младший битов числа b.

**Level8**

Достал стек через %x

**Level9**

Переполняем пароль, за счет этого в key записывается больше, чем нужно. Значит, стек переполнится и в i запишется другое значение.

**Level10**

Ввел полный путь к exe- файлу. Длина больше размера массива, стек переполнился и операция buf[16] = 0, записала 0 в i.

**Level11**

В стеке лежит saved\_eip, saved\_ebp, buf[80], тобишь надо переполнить buf, затем 4 байта для saved\_ebp, и в saved\_eip записать 0x41424344, а это символы A B C D.

Но т.к. стек LIFO, записать их надо как D C B A.

**Level12**

0x40100f = (0, 64, 16, 15) \0,@,^P,^O

EIP (указатель на точку возврата), EBP (указатель на начало стека), buf[80]

Надо сдвинуть данные в стеке таким образом, чтобы в EIP оказался указатель на нашу функцию, поэтому вводим 80 символов для переполнения буфера + 4 для переполнения регистра EBP + символы, которые определят указатель на нашу функцию (в обратном порядке).