Floristería

Autor

`@pwnedshell`

Descripción

Sam va tan lentín que te ha pedido que le compres unas flores para su amada. Dicen que la propia floristería te puede vender una flag, a ver cómo lo haces para conseguirla...

Solución

Se concede una conexión a un servidor mediante **netcat** y un zip que contiene el código del reto, listo para levantarse con **docker**.

```
docker-compose up --build
```

Al conectarnos al puerto con **netcat** observamos lo siguiente:

```
❯ nc ********* ****
```

```
-------------------------
1) Mirar la cartera
2) Ir a trabajar
3) Comprar flag (10000€)
4) Comprar cactus (24€)
5) Comprar nassau (56€)
6) Comprar orquídea (44€)
7) Comprar merlot (75€)
8) Comprar ficus (82€)
9) Salir
¿Qué deseas hacer?
```

Podemos mirar la cartera:

```
-------------------------
1) Mirar la cartera
2) Ir a trabajar
3) Comprar flag (10000€)
4) Comprar cactus (24€)
5) Comprar nassau (56€)
6) Comprar orquídea (44€)
7) Comprar merlot (75€)
8) Comprar ficus (82€)
9) Salir
¿Qué deseas hacer?
1

Tu saldo actual es de 100€
```

Observamos que solo tenemos 100€. ¿Qué pasa si vamos a trabajar?

```
-------------------------
1) Mirar la cartera
2) Ir a trabajar
3) Comprar flag (10000€)
4) Comprar cactus (24€)
5) Comprar nassau (56€)
6) Comprar orquídea (44€)
7) Comprar merlot (75€)
8) Comprar ficus (82€)
9) Salir
¿Qué deseas hacer?
2

A trabajar otra vez ...

Has trabajado como un verdadero campeón, toma 3€
Tu saldo actual es de 103€ para gastar en flores
```

Tras esperar 15 segundos se obtiene la enorme cantidad de 3€. No parece que merezca la pena el uso de fuerza bruta para llegar a 10000€.

Por último se puede comprobar la funcionabilidad de **Comprar.**

```
-------------------------
1) Mirar la cartera
2) Ir a trabajar
3) Comprar flag (10000€)
4) Comprar cactus (24€)
5) Comprar nassau (56€)
6) Comprar orquídea (44€)
7) Comprar merlot (75€)
8) Comprar ficus (82€)
9) Salir
¿Qué deseas hacer?
4

¿Cuantas cactus quieres?
15
No puedes quedarte sin dinero por que tienes que comprar el pan

-------------------------
1) Mirar la cartera
2) Ir a trabajar
3) Comprar flag (10000€)
4) Comprar cactus (24€)
5) Comprar nassau (56€)
6) Comprar orquídea (44€)
7) Comprar merlot (75€)
8) Comprar ficus (82€)
9) Salir
¿Qué deseas hacer?
4

¿Cuantas cactus quieres?
2
Que bonitos son los 2 cactus que has comprado
Te ha costado 48€
Tu saldo actual es de 55€
```

Efectivamente no podemos comprar 15 cactus por que no tenemos suficiente dinero y hay que comprar el pan. Sin embargo 2 sí (obvio).

Pasemos a lo interesante, mirar el código.

```
package main

import (
  "fmt"
  "math"
  "os"
  "strconv"
```

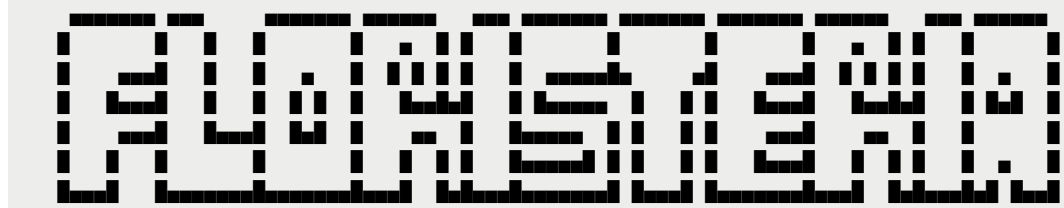
```
  "strings"
  "time"

  "github.com/dixonwille/wmenu/v5"
)

type flor struct {
  Nombre string
  Precio int16
}

var saldo = int16(100)

func main() {
  fmt.Println(`
```



```
  `)

  menu := wmenu.NewMenu("¿Qué deseas hacer?")
  menu.Option("Mirar la cartera", nil, false, mirar)
  menu.Option("Ir a trabajar", nil, false, trabajar)
  menu.Option("Comprar flag (10000€)", nil, false, flag)
  menu.Option("Comprar cactus (24€)", flor{Nombre: "cactus", Precio: int16(24)}, false, comprar)
  menu.Option("Comprar nassau (56€)", flor{Nombre: "nassau", Precio: int16(56)}, false, comprar)
  menu.Option("Comprar orquídea (44€)", flor{Nombre: "orquídea", Precio: int16(44)}, false, comprar)
  menu.Option("Comprar merlot (75€)", flor{Nombre: "merlot", Precio: int16(75)}, false, comprar)
  menu.Option("Comprar ficus (82€)", flor{Nombre: "ficus", Precio: int16(82)}, false, comprar)
  menu.Option("Salir", nil, false, func(opt wmenu.Opt) error {
    os.Exit(0)
    return nil
  })
  for {
    fmt.Println("-------------------------")
    err := menu.Run()
    if err != nil {
      if strings.Contains(err.Error(), "invalid response") {
        fmt.Println("Respuesta inválida")
      } else {
        fmt.Println("Ha ocurrido un error que no debería ocurrir")
      }
    }
  }
}

func trabajar(opt wmenu.Opt) error {
  fmt.Println("\nA trabajar otra vez ...")
  time.Sleep(15 * time.Second)
  fmt.Printf("\nHas trabajado como un verdadero campeón, toma 3€\n")
  saldo = int16(math.Abs(float64(saldo)) + 3)
  fmt.Printf("Tu saldo actual es de %d€ para gastar en flores\n\n", saldo)
  return nil
}
```

```
func mirar(opt wmenu.Opt) error {
  fmt.Printf("\nTu saldo actual es de %d€\n\n", saldo)
  return nil
}

func flag(opt wmenu.Opt) error {
  if saldo < 10000 {
    fmt.Println("No puedes quedarte sin dinero por que tienes que comprar el pan")
  } else {
    fmt.Println("HackOn{Fake_Flag_4_Testing}")
    os.Exit(1)
  }
  return nil
}

func comprar(opt wmenu.Opt) error {
  var input string
  fmt.Printf("\n¿Cuantas %s quieres?\n", opt.Value.(flor).Nombre)
  fmt.Scanf("%s", &input)
  if len(input) < 1 {
    fmt.Printf("Indtroduce algo ¿no?\n\n")
    return nil
  }
  if input[0] == '-' {
    fmt.Printf("No puedes introducir un número negativo\n\n")
    return nil
  }
  i, err := strconv.Atoi(input)
  if err != nil {
    fmt.Printf("Algo ha pasado al convertir el input a entero\n\n")
    return nil
  }
  if i == 0 {
    fmt.Printf("Hmmm ¿No quieres?\n\n")
    return nil
  }
  if i > 400 {
    fmt.Printf("No tenemos tanto stock, lo sentimos\n\n")
    return nil
  }
  total := int16(i) * opt.Value.(flor).Precio
  if saldo < total {
    fmt.Printf("No puedes quedarte sin dinero por que tienes que comprar el pan\n\n")
  } else {
    saldo = saldo - int16(total)
    if i > 1 {
      fmt.Printf("Que bonitos son los %d %s que has comprado\n", i, opt.Value.(flor).Nombre)
    } else {
      fmt.Printf("Vaya %s más precioso has comprado\n", opt.Value.(flor).Nombre)
    }
    fmt.Printf("Te ha costado %d€\n", total)
    fmt.Printf("Tu saldo actual es de %d€\n\n", saldo)
  }
  return nil
}
```

Para ir al grano, en este reto hay que darse cuenta de varias cosas.

1. El reto está hecho en GO

No es lo más relevante pero sí que permite investigar un poco cómo funciona este lenguaje y cómo maneja este los desbordamientos de enteros. (Spoiler, es un **integer overflow**).

2. Dónde está la flag

Efectivamente la **flag** se consigue desde la función `flag()` llamada al pedir la compra de una flag.

Se observa cómo en la línea 73 se realiza la comprobación `saldo < 10000` y en caso de que el **saldo** sea mayor o igual se devolvería la **flag** y acabaría el programa.

3. Int16

En este programa se opera constantemente con **int16**, lo quiere decir que vamos a trabajar con enteros de rango `-32,768 a 32,767`.

4. El camino del input

En la función `comprar()` el input que introduce el usuario cuando es preguntado por la cantidad que desea se registra como un **string**.

Posteriormente se comprueba que no empieza por el carácter `-` y se convierte a entero (por defecto la conversión de la función `strconv.Atoi` devuelve un **int64**).

Se comprueba si es 0, menor, **o igual** a 400 y en dichos casos acaba.

Necesitamos por lo tanto un número entre 1 y 400 incluido.

Una vez se tenga ese número se transforma a **int16** y se multiplica por el valor de la flor elegida.

Solamente si el **saldo** es mayor al **total** se restará el total al saldo y se mostrará por pantalla la compra.

5. Algo extraño en la función trabajar

Aunque parezca que la función `trabajar()` es únicamente para ganar 3 miseros euros a cambio de 15 segundos de tu vida, es importante observar cómo (sin venir a cuento) se transforma el saldo obtenido a su valor absoluto.

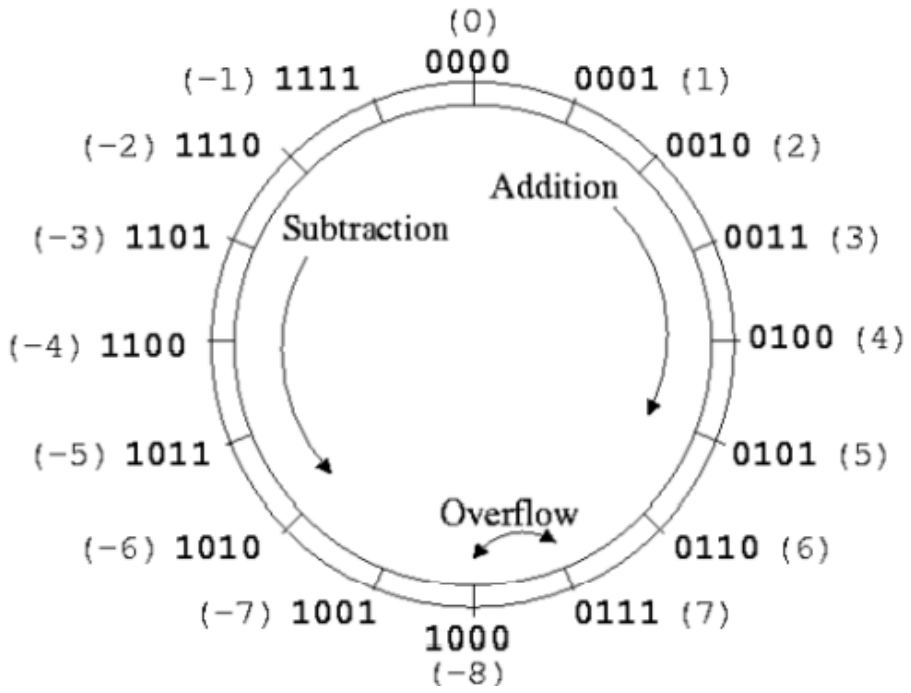
Hmmm, ¿por qué haría eso?

Integer overflow

Se puede definir como el resultado de intentar almacenar en memoria una variable de algún tipo (int, char, short…) con un valor que sobrepase el rango máximo representable del mismo

Colisiones

Las colisiones son originadas al sobrepasar el rango máximo que permite una variable (o en el caso del ejemplo, una representación de un número de bits), por lo que a partir de ese valor tope, los valores empiezan a repetirse siguiendo un esquema similar a la estructura de un reloj (como en la figura que se adjunta a continuación), en la que la mitad derecha corresponde a los números positivos y la mitad izquierda a los negativos



La idea del reto

La idea del reto es darse cuenta de que, para poder explotar un **integer overflow** de **int16,** se necesita un número mayor a `32,767` y que se puede pedir un máximo de 400 unidades de x flor.

¿Qué pasa si dividimos 32768 entre todos los precios de las flores? (redondeando para abajo)

- 32768 / 24 cactus = 1365
- 32768 / 56 nassau = 585
- 32768 / 44 orquídea = 743

- 32768 / 75 merlot = 436
- 32768 / 82 ficus = **399 (vualá)**

Si se compran 400 por lo tanto, se superará el máximo permitido para un **int16** y al ser entero con signo, se obtendrá un número negativo (recuerda que la comprobación del negativo se hace mucho antes de llegar a este paso y aquí ya no sirve de nada).

```
400 * 82 = 32800 // total := int16(i) * opt.Value.(flor).Precio
int16(32800) = -32736 // saldo = saldo - int16(total)
100 - -32736 = -32700 // saldo = saldo - int16(total) se suma y vuelve a sobrepasar el límite
// saldo = -32700
```

Compremos 400 ficus.

```
-------------------------
1) Mirar la cartera
2) Ir a trabajar
3) Comprar flag (10000€)
4) Comprar cactus (24€)
5) Comprar nassau (56€)
6) Comprar orquídea (44€)
7) Comprar merlot (75€)
8) Comprar ficus (82€)
9) Salir
¿Qué deseas hacer?
8

¿Cuantas ficus quieres?
400
Que bonitos son los 400 ficus que has comprado
Te ha costado -32736€
Tu saldo actual es de -32700€
```

Observamos como tenemos un saldo (aunque negativo) muy grande.

Si pedimos una flag, el saldo va a seguir siendo negativo, si seguimos pidiendo de 400 en 400 ficus, llegará un momento en el que, por ~~huevos~~ matemáticas, no se podrá seguir avanzando ni comprando nada.

Es el momento de recordad que al trabajar, el saldo es transformado a su valor absoluto. Trabajemos pues.

```
-------------------------
1) Mirar la cartera
2) Ir a trabajar
```

```
3) Comprar flag (10000€)
4) Comprar cactus (24€)
5) Comprar nassau (56€)
6) Comprar orquídea (44€)
7) Comprar merlot (75€)
8) Comprar ficus (82€)
9) Salir
¿Qué deseas hacer?
2

A trabajar otra vez ...

Has trabajado como un verdadero campeón, toma 3€
Tu saldo actual es de 32703€ para gastar en flores
```

Vaya, parece que ahora nuestro saldo es positivo y...

¡Podemos comprar la flag!

```
-------------------------
1) Mirar la cartera
2) Ir a trabajar
3) Comprar flag (10000€)
4) Comprar cactus (24€)
5) Comprar nassau (56€)
6) Comprar orquídea (44€)
7) Comprar merlot (75€)
8) Comprar ficus (82€)
9) Salir
¿Qué deseas hacer?
3
HackOn{Eres_la_fl0r_que_p0rta_la_espina_que_desgarra_mi_coras0n}
```

Flag

```
HackOn{Eres_la_fl0r_que_p0rta_la_espina_que_desgarra_mi_coras0n}
```