Golang RE Intro

Montrehack August 2019
WiFi : GoogleGuest

Golang

- Compiled language
- Aims to replace older low-level languages like C
- Has packages (similar to Python modules or Rust crates)
- Built-in concurrency

Why reversing Golang?

- Real software
  - Docker
  - Kubernetes
- Malware
  - Elf.Lady
  - Mirai’s server
  - Cryptominers (reversed one yesterday!)

Identifying Golang binaries

```
$ file Level1

ELF 64-bit LSB executable, x86-64, version 1 (SYSV),
statically linked, Go
BuildID=5yB10_vLiY-4fhyHKD_-/nC453_wpnPdfWn44-oLj

/76Pq_KDkNSyUw2vA6KxE/28GNz8qYhc1QnaP8_OEN, not stripped
```

Statically linked

```
[svieg@primarch Level1]$ wc -l level1.go

29 level1.go

[svieg@primarch Level1]$ ls -lah Level1

-rwxr-xr-x 1 svieg svieg 2.0M Jul 29 23:52 Level1
```

Try not to get lost in the package code!

Experiment!

Try the first challenge at

svieg.com/Level1.{exe,elf}

Solution at 6:45

Hints

- You want the look at the **main** package
  - main_* or main.*

Strings

One giant string!

```
aGoFindTheFlagA db '`Go` find the flag and enter it here: arg size to reflect.call mo'
                                          ; DATA XREF: .rodata:main_statictmp_1↓o
                db 're than 1GBcan not access a needed shared libraryconcurrent map i'
                db 'teration and map writegcBgMarkWorker: blackening not enabledmakec'
                db 'han: invalid channel element typeruntime: blocked read on free po'
                db 'lldescruntime: sudog with non-false isSelect277555756156289135105'
```

Strings - References

format: <pointer><size>

```
main_statictmp_1 dq offset aGoFindTheFlagA
                                        ; DATA XREF: main_main+4C↑o
                                        ; "`Go` find the flag and enter it here: a"...
                 dq 26h
```

Sections

.gopclntab

Function names

Follow the pattern <package_name>_<function_name> or <package_name>.<function_name>

I.e.:

- main_main or main.main
- os_NewFile or os.NewFile

Helpful scripts

IDA 7.x (not freeware): https://github.com/sibears/IDAGolangHelper

Ghidra:
https://github.com/ghidraninja/ghidra_scripts/blob/master/golang_renamer.py

Source - Boilerplate

```
package main; // functions main_*

import (

    "fmt"; // functions fmt_*

    "os"; // functions os_*

    "bufio"; // functions bufio_*

)
```

Source - Main

```
func main() {

    fmt.Print("`Go` find the flag and enter it here: ");

    user_input,_,err := bufio.NewReader(os.Stdin).ReadLine();

    if err != nil {

            fmt.Println("Invalid input :/ , ",err);

    }

    if check_flag(user_input) {

        fmt.Println("Congrats!");

    } else {

        fmt.Println("That's not the flag :( ");

    }}
```

Source - check_flag

In the binary: Function **main_check_flag**

```
func check_flag(user_input []byte) bool {

    scrambled_flag := []string{"!", "-", "0", "0", "0", "3", "4", "A", "F", "G", "G", "L", "L",
"T", "W", "_", "_", "c", "e", "g", "l", "m", "n"};

    flag := string(scrambled_flag[8] + scrambled_flag[11] + scrambled_flag[7] + scrambled_flag[9] +
scrambled_flag[1] + scrambled_flag[14] + scrambled_flag[5] + scrambled_flag[20] +
scrambled_flag[17] + scrambled_flag[2] + scrambled_flag[21] + scrambled_flag[18] +
scrambled_flag[15] + scrambled_flag[13] + scrambled_flag[2] + scrambled_flag[15] +
scrambled_flag[9] + scrambled_flag[2] + scrambled_flag[11] + scrambled_flag[6] + scrambled_flag[22]
+ scrambled_flag[19] + scrambled_flag[0])

    return string(user_input) == flag;

}
```

Solution - Static

- Without running the binary with IDA, Ghidra, etc.
- main_main calls main_check_flag
- From Ghidra:

```
  main.check_flag();
  if (local_150 == '\0')
    fmt.Fprintln();
  }
  else {
    fmt.Fprintln();
  }
  return;
}
```

Identifying the []byte

```
sub     rsp, 338h
mov     [rsp+338h+var_8], rbp
lea     rbp, [rsp+338h+var_8]
lea     rdi, [rsp+338h+var_2E8]
lea     rsi, main_statictmp_0 ; []byte (byte slice)
mov     [rsp+338h+var_348], rbp
lea     rbp, [rsp+338h+var_348]
call    slice_operation
```

Byte slice to bytes

```
main_statictmp_0 db  41h ; A
                 db 0B1h
                 db  4Bh ; K
                 db    0
                 db    0
```

Change to offset (select + hit ‘d’ x 4)

```
main_statictmp_0 dq offset byte_4BB141
                 db    1
                 db    0
                 db    0
                 db    0
                 db    0
                 db    0
```

Byte slice to bytes (cont.)

```
unk_4BB141      db  21h ; !
                db  28h ; (
```

Define byte plus change to char (select, ‘d’ then ‘r’)

```
byte_4BB141     db '!'
```

```
sub_451E0E(&v14, &main_statictmp_0);
sub_451B51(&v49, 0.0);
v52 = v28;
v51 = v27;
v54 = v32;
v53 = v31;
v56 = v26;
v55 = v25;
v58 = v30;
v57 = v29;
v60 = v17;
v59 = v16;
v62 = v36;
v61 = v35;
v64 = v22;
v63 = v21;
v66 = v46;
v65 = v45;
v68 = v40;
v67 = v39;
v70 = v19;
v69 = v18;
v72 = v48;
v71 = v47;
v74 = v42;
v73 = v41;
v76 = v38;
v75 = v37;
v78 = v34;
v77 = v33;
v80 = v19;
v79 = v18;
v82 = v38;
v81 = v37;
v84 = v30;
v83 = v29;
v86 = v19;
v85 = v18;
v88 = v32;
v87 = v31;
v90 = v24;
v89 = v23;
v92 = v50;
v91 = v49;
v94 = v44;
v93 = v43;
v96 = v15;
v95 = v14;
```

Solution - Dynamic

- Running the tool with a little bit of static analysis
- ltrace, strace
- GDB

Ltrace and strace

```
write(1, "`Go` find the flag and enter it "..., 38`Go` find the flag and enter it here: ) = 38
read(0, allo
"allo\n", 4096)                 = 5
futex(0x55d0d0, FUTEX_WAKE_PRIVATE, 1)  = 1
write(1, "That's not the flag :( \n", 24That's not the flag :(
) = 24
exit_group(0)                           = ?
+++ exited with 0 +++
hugen@hugen:~/Downloads$ ltrace ./Level1.elf
Couldn't find .dynsym or .dynstr in "/proc/202457/exe"
hugen@hugen:~/Downloads$ `Go` find the flag and enter it here: Invalid input :/ ,  read /dev/stdin: input/output error
That's not the flag :(
```

GDB

- Bunch of memory manipulation
- Then a branch
- Breakpoint on the branch
- Check memory

```
.text:0000000000488DCB mov     rax, [rsp+338h+var_188]
.text:0000000000488DD3 mov     rcx, [rsp+338h+var_180]
.text:0000000000488DDB mov     [rsp+338h+var_30], rcx
.text:0000000000488DE3 mov     [rsp+338h+var_38], rax
.text:0000000000488DEB mov     rax, [rsp+338h+var_1B0]
.text:0000000000488DF3 mov     rcx, [rsp+338h+var_1B8]
.text:0000000000488DFB mov     [rsp+338h+var_20], rax
.text:0000000000488E03 mov     [rsp+338h+var_28], rcx
.text:0000000000488E0B mov     rax, [rsp+338h+var_2E8]
.text:0000000000488E10 mov     rcx, [rsp+338h+var_2E0]
.text:0000000000488E15 mov     [rsp+338h+var_10], rcx
.text:0000000000488E1D mov     [rsp+338h+var_18], rax
.text:0000000000488E25 lea     rax, [rsp+338h+var_308]
.text:0000000000488E2A mov     [rsp+338h+var_338], rax
.text:0000000000488E2E lea     rax, [rsp+338h+var_178]
.text:0000000000488E36 mov     [rsp+338h+var_330], rax
.text:0000000000488E3B mov     [rsp+338h+var_328], 17h
.text:0000000000488E44 mov     [rsp+338h+var_320], 17h
.text:0000000000488E4D call    runtime_concatstrings
.text:0000000000488E52 mov     rax, [rsp+338h+var_318]
.text:0000000000488E57 mov     rcx, [rsp+338h+arg_8]
.text:0000000000488E5F cmp     [rsp+338h+var_310], rcx
.text:0000000000488E64 jz      short loc_488E7F
```

GDB (cont.)

- Breakpoint

```
(gdb) b main
main                main.check_flag  main.init        main.main
(gdb) b *main.check_flag+0x36f
```

- Check memory

```
(gdb) x/100wx $rsp
0xc0000a2ae8:   0x000a2b18      0x000000c0      0x000a2ca8      0x000000c0
0xc0000a2af8:   0x00000017      0x00000000      0x00000017      0x00000000
0xc0000a2b08:   0x000a2b18      0x000000c0      0x00000017      0x00000000
0xc0000a2b18:   0x47414c46      0x6c33572d      0x656d3063      0x5f30545f
0xc0000a2b28:   0x344c3047      0x0021676e      0x00000000      0x00000000
0xc0000a2b38:   0x004bb141      0x00000000      0x00000001      0x00000000
```

- Print flag

```
(gdb) x/s 0xc0000a2b18
0xc0000a2b18:   "FLAG-W3lc0me_T0_G0L4ng!"
```

Experiment!

Try the second challenge at

svieg.com/Level2.{exe,elf}

Solution at 7:30

Hints

- Search for “8b9035807842a4e4dbe009f3f1478127”

Solution

- “Custom” MD5 implementation!
- copied from crypto/md5 package
- Lacks some constants!

var int a0 := 0x67452301 → 0x00000001

var int b0 := 0xefcdab89 //B → same

var int c0 := 0x98badcfe //C → same

var int d0 := 0x10325476 //D → 0x10000000

Solution - main_main

From the documentation

```
func (b *Reader) ReadLine() (line []byte, isPrefix bool, err error)
```

```
bufio___Reader__ReadLine((__int64)&var_58, (__int64)&v45, v5, v6,
v7, v8, *(__int128 *)&v27);

[...]

  main_check_user_key((__int64)&var_58, v12, (__int64)v35, v34,
v9, v10, v36, v34);
```

Solution - main_check_user_key

function prototype:

main_check_user_key(__int64 a1, __int64 a2, __int64 a3, __int64 a4, __int64 a5, __int64 a6, **unsigned __int8 *a7**, unsigned __int64 a8)

a7 is our byte slice (user input)

main_Sum(a1, a2, a3, (__int64)a7, a5, a6, v16, 6LL, **a7**, v8);

main_sum

main_Sum(__int64 a1, __int64 a2, __int64 a3, __int64 a4, __int64 a5, __int64 a6, __int128 a7, __int64 a8, **unsigned __int8 *a9**, unsigned __int64 a10)

[...]

if (!a10 || (v11 = ***a9**, a10 <= 1) || (v11 = (**a9[1]** << 16) + ((_DWORD)v11 << 24), a10 <= 2)) **[...]**

v20 = v11 + (**a9[2]** << 8) **+ 1**;

if (a10 <= 3 || (v13 = **a9[3]**, a10 <= 4) || (v13 = (**a9[4]** << 8) + ((_DWORD)v13 << 16), a10 <= 5)

v22 = v13 + **a9[5] + 0x10000000**;

Solution

Key is the values in the ascii-range of some MD5 constants:

```
hugen@hugen:~/Downloads$ ./Level2.elf

Hey, do you have the key?: gE#2Tv

Flag: FLAG-W41t_Th4ts_Ju5t_Md5
```

Experiment!

Try the third challenge at

ssh level3@svieg.com password: level3

svieg.com/Level3.{exe,elf}

Solution at 8:30

Hints

- You control the URL so you control the server!
- Get request has a custom timeout!

Solution

- Concurrency is easy!

```
filename2 := "FLAG.txt"
go func() {
    time.Sleep(20 * time.Second)
    filename = filename2
}()
```

- Race condition!
- Two threads:
  - Main thread
  - Thread changing the name of the file
- Things you have control over: a URL (and the server that this points to!)

Normal Flow

1. Main thread gets started
2. Ask for input
3. Second thread started with a 20 seconds sleep
4. Server takes less than 20 seconds to respond (probably 404’s because it doesn’t have /check_internet)
5. Read “NO_FLAG.txt”

Abused Flow

1. Main thread gets started
2. Ask for input
3. Second thread started with a 20 seconds sleep
4. Server takes **more** than 20 seconds to respond
5. Second thread changes the filename to “FLAG.txt”
6. Request times at 25 seconds out and returns
7. Read “FLAG.txt”

Solution - main_main

- NO_FLAG.txt
- FLAG.txt
- runtime_newproc

```
lea     rax, aHomeFlagFlagTx+1B88h ; "/home/level3/NO_FLAG.txt/usr/share/lib/"...
mov     [rdi], rax
```

```
loc_62FCCC:
lea     rax, aHomeFlagFlagTx+1B88h ; "/home/
call    runtime_gcWriteBarrier
jmp     loc_62FA51
```

```
loc_62FA51:
nop
mov     [rsp+1A0h+var_1A0], rax
mov     [rsp+1A0h+var_198], rax
mov     [rsp+1A0h+var_190], 18h
call    runtime_memequal
cmp     byte ptr [rsp+1A0h+var_188], 0
jz      loc_62FCBC
```

```
mov     dword ptr [rsp+1A0h+var_1A0], 18h
lea     rax, off_6CDB30
mov     [rsp+1A0h+var_198], rax
lea     rax, aHomeFlagFlagTx ; "/home/flag/FLAG.txt/usr/lib/locale/TZ/1"...
mov     [rsp+1A0h+var_190], rax
mov     [rsp+1A0h+var_188], 13h
mov     rax, [rsp+1A0h+var_120]
mov     [rsp+1A0h+var_180], rax
call    runtime_newproc
lea     rax, [rsp+1A0h+var_150]
```

```
loc_62FCBC:
mov     rbp, [rsp+1A0h+var_8]
add     rsp, 1A0h
retn
```

Solution - off_C6DB30

The new thread function!

```
off_6CDB30    |    dq offset main_main_func1
```

Solution - main_main_func1

- Sleep for 20 seconds
- assign arg_8 to arg_10

```
sub     rsp, 10h
mov     [rsp+10h+var_8], rbp
lea     rbp, [rsp+10h+var_8]
mov     rax, 20000000000 ; 20 seconds
mov     [rsp+10h+var_10], rax
call    time_Sleep
mov     rax, [rsp+10h+arg_8]
mov     rdi, [rsp+10h+arg_10]
mov     [rdi+8], rax
cmp     cs:runtime_writeBarrier, 0
jnz     short loc_62FD49
```

Solution - main_main (cont.)

```
call    main_check_internet_connection

[...]

call    io_ioutil_ReadFile
```

Solution - main_check_internet_connection

- runtime_newobject
- 25 seconds
- rax is object
- Sets timeout value

```
sub     rsp, 58h
mov     [rsp+58h+var_8], rbp
lea     rbp, [rsp+58h+var_8]
lea     rax, unk_695980
mov     [rsp+58h+var_58], rax
call    runtime_newobject
mov     rax, [rsp+58h+var_50]
mov     [rsp+58h+var_10], rax
cmp     cs:runtime_writeBarrier, 0
jnz     loc_62F773
```

```
loc_62F792:
call    runtime_morestac
main_check_internet_conn
```

```
m0
ptr [rax], xmm0
ptr [rax+10h], xmm0
ptr [rax+20h], xmm0
```

```
loc_62F773:
lea     rcx, unk_695980
mov     [rsp+58h+var_58], rcx
mov     [rsp+58h+var_50], rax
call    runtime_typedmemclr
mov     rax, [rsp+58h+var_10]
jmp     loc_62F656      ; 25 seconds
```

```
loc_62F656:             ; 25 seconds
mov     rcx, 25000000000
mov     [rax+28h], rcx
mov     [rsp+58h+var_58], 0
mov     rcx, [rsp+58h+arg_0]
mov     [rsp+58h+var_50], rcx
mov     rcx, [rsp+58h+arg_8]
mov     [rsp+58h+var_48], rcx
lea     rcx, unk_6BC8E1
mov     [rsp+58h+var_40], rcx
mov     [rsp+58h+var_38], 0Fh
call    runtime_concatstring2
mov     rax, [rsp+58h+var_28]
mov     rcx, [rsp+58h+var_30]
mov     rdx, [rsp+58h+var_10]
mov     [rsp+58h+var_58], rdx
mov     [rsp+58h+var_50], rcx
mov     [rsp+58h+var_48], rax
call    net_http___Client__Get
```

Abused Flow

1. Main thread gets started
2. Ask for input
3. Second thread started with a 20 seconds sleep
4. Server takes **more** than 20 seconds to respond
5. Second thread changes the filename to “FLAG.txt”
6. Request times at 25 seconds out and returns
7. Read “FLAG.txt”

Thanks!

We’re hiring :)

Slides at: svieg.com/montrehack-slides.pdf